Tight Adaptive Simulation Security for Identity-based Inner-Product FE in the (Quantum) Random Oracle Model

Tennma Edamura[1] and Atsushi Takayasu ✉[1,2][0000−0002−9310−6976]

The University of Tokyo, Japan,
National Institute of Advanced Industrial Science and Technology, Japan,
{edamura-tenma0821, takayasu-a}@g.ecc.u-tokyo.ac.jp

Abstract. Abdalla et al. (ASIACRYPT 2020) introduced a notion of *identity-based inner-product functional encryption* (IBIPFE) that combines identity-based encryption and inner-product functional encryption (IPFE). Thus far, several pairing-based and lattice-based IBIPFE schemes have been proposed. However, there are two open problems. First, there are no known IBIPFE schemes that satisfy the *adaptive simulation-based security*. Second, known IBIPFE schemes that satisfy the adaptive indistinguishability-based security or the selective simulation-based security do not have tight reductions. In this paper, we propose lattice-based and pairing-based IBIPFE schemes that satisfy the tight adaptive simulation-based security. At first, we propose a generic transformation from an indistinguishability-based secure $(L+1)$-dimensional (IB)IPFE scheme to a simulation-based secure L-dimensional (IB)IPFE scheme. The proposed transformation improves Agrawal et al.'s transformation for plain IPFE (PKC 2020) that requires an indistinguishability-based secure $2L$-dimensional scheme. Then, we construct a lattice-based IBIPFE scheme that satisfies the tight adaptive indistinguishability-based security under the LWE assumption in the quantum random oracle model. We apply the proposed transformation and obtain the first lattice-based IBIPFE scheme that satisfies adaptive simulation-based security. Finally, we construct a pairing-based IBIPFE scheme that satisfies the tight adaptive simulation-based security under the DBDH assumption in the random oracle model. The pairing-based scheme does not use the proposed transformation towards the best efficiency.

Keywords: (identity-based) inner-product functional encryption, simulation-based security, tight reduction, (quantum) random oracle model, lattice, pairing.

1 Introduction

1.1 Background

Functional encryption (FE) [12,31] is a generalization of the traditional public key encryption by prohibiting all-or-nothing decryption. If a user decrypts a ciphertext $\mathsf{FE.ct}_x$ that is an encryption of x by using a secret key $\mathsf{FE.sk}_f$ associated with a function f, the decryption result is not x itself but $f(x)$. There are *indistinguishability* (IND)-based and *simulation* (SIM)-based security definitions. Briefly speaking, IND-based security ensures that an adversary that is given several $\mathsf{FE.sk}_{f_i}$ such that $f_i(x_0^\star) = f_i(x_1^\star)$ cannot distinguish $\mathsf{FE.ct}_{x_0^\star}$ and $\mathsf{FE.ct}_{x_1^\star}$, while the SIM-based security ensures that an adversary that is given $\mathsf{FE.ct}_{x^\star}$ and several $\mathsf{FE.sk}_{f_i}$ cannot learn anything besides $f_i(x^\star)$. Thus, SIM-based security is the stronger than IND-based security and *adaptive* (AD) SIM security is the most desirable security definition [12,31].

Although there are known FE schemes for all circuits [18,19], strong assumptions such as multilinear maps [17] and indistinguishability obfuscation [18] are required. In turn, FE for simple functionalities under standard assumptions has been studied. One of the most fundamental research topics should be arguably *inner-product functional encryption* (IPFE) introduced by Abdalla et al. [1]. A decryption result of a ciphertext $\mathsf{IPFE.ct}_{\mathbf{x}}$ associated with an L-dimensional vector $\mathbf{x}$ by a secret key $\mathsf{IPFE.sk}_{\mathbf{y}}$ associated with an L-dimensional vector $\mathbf{y}$ is their inner product $\langle \mathbf{x}, \mathbf{y} \rangle$. After Abdalla et al. [1] proposed selective (SEL)-IND-secure IPFE schemes under the decisional Diffie-Hellman (DDH) assumption and the learning with errors (LWE) assumption, Agrawal et al. [7] proposed AD-IND-secure IPFE schemes under DDH, LWE, and the decision composite residuosity DCR assumption. In this paper, we focus on DDH and LWE-based schemes. Specifically, Agrawal et al. [7] proposed two LWE-based schemes that compute inner products either over the integers or modulo a prime p. Then, Abdalla et al. [3] (resp. Wee [37]) proved that Agrawal et al.'s DDH-based scheme [7] satisfies SEL-SIM (resp. semi-adaptive SIM) security. Finally, Agrawal et al. [6] proved that the DDH-based scheme [7] satisfies the AD-SIM security. In contrast, Agrawal et al. [4] improved the efficiency of Agrawal et al.'s LWE-based schemes [7] by introducing a weaker security model. Wang et al. [36] also improved the efficiency in the same AD-IND security of Agrawal et al. [7]. Then, Agrawal et al. [6] modified the LWE-based scheme module a prime p [7,36] by introducing a generic transformation from a $2L$-dimensional IND-secure IPFE scheme modulo p to an L-dimensional SIM-secure scheme with stateful key generations. Therefore, the SIM-secure schemes are less efficient than the IND-secure schemes. Although Lin and Luo [28] improved the efficiency of AD-SIM-secure LWE-based schemes by introducing a transformation from an $(L+1)$-dimensional IND-secure scheme to an L-dimensional SIM-secure scheme, their proposed AD-SIM security is weaker than Agrawal et al. [6] since a simulator of Lin and Luo [28] can receive additional information compared with Agrawal et al. [6]. We note that most IPFE schemes have *tight reductions* since the most technical steps are usually not computational but information theoretical arguments.

Abdalla et al. [2] extended IPFE to IPFE *with fine-grained access control* such as identity-based IPFE (IBIPFE) and attribute-based IPFE, where we focus on IBIPFE throughout the paper. Abdalla et al. [2] proposed several IBIPFE schemes under the symmetric external Diffie-Hellman (SXDH) assumption and the LWE assumption. Their SXDH-based schemes satisfy either AD-IND security or SEL-SIM security, while their LWE-based schemes satisfy either AD-IND security in the random oracle model (ROM) or SEL-IND security in the standard model. Then, Lai et al. [27] proposed AD-IND-secure LWE-based IBIPFE scheme in the standard model. However, AD-SIM-secure IBIPFE schemes have not been proposed so far. Moreover, all the above AD-IND-secure or SEL-SIM-secure IBIPFE schemes [2,27] do not have tight reduction although there are several AD-secure IPFE scheme [6,7,36] and AD-secure identity-based encryption (IBE) schemes [9,10,13,15,16,20,21,22,23,24,26] with (almost) tight reductions under standard assumptions. Among the above LWE-based IBIPFE schemes, Abdalla et al.'s scheme in the ROM [2] that combines Gentry et al.'s IBE scheme [20] and Wang et al.'s IPFE scheme [36] is the most efficient. However, their proof should be refined in the quantum random oracle model (QROM) since an encryption scheme secure under post-quantum assumptions in the ROM may be vulnerable against quantum adversaries [38].

1.2 Our Contribution

In this paper, we propose two efficient AD-SIM-secure IBIPFE schemes with tight reductions under standard assumptions in the (Q)ROM. To construct an AD-SIM-secure LWE-based scheme, we borrow the idea of Agrawal et al. [6] and propose a transformation from an IND-secure IBIPFE scheme to a SIM-secure scheme. For this purpose, we do not just extend Agrawal et al.'s transformation [6] to the identity-based setting but improve it. In particular, our transformation for an L-dimensional SIM-secure scheme has stateless key generations and utilizes not $2L$ but only an $(L+1)$-dimensional IND-secure scheme that computes inner products modulo a prime p. The proposed transformation is slightly more efficient than Lin and Luo [28] due to shorter ciphertexts although the former achieves the stronger security. Therefore, the proposed transformation provides the most efficient AD-SIM-secure LWE-based (plain) IPFE scheme. We note that a proof of the proposed transformation is not the same as Lin and Luo [28] since additional information from a simulator and slightly larger ciphertexts are essential for the latter proof.

Although the proposed transformation looks sufficient to construct AD-SIM-secure IBIPFE schemes at a glance, it is not the case for LWE-based schemes since Abdalla et al.'s [2] and Lai et al.'s [27] IBIPFE scheme do not compute inner products modulo a prime p but over the integers. Therefore, we modify Abdalla et al.'s IBIPFE scheme in the ROM [2] to compute inner products modulo a prime p since the scheme is the most efficient among known LWE-based IBIPFE schemes [2,27]. The modification itself is not very impressive since we just borrow the idea of Agrawal et al. [7] that modified their proposed LWE-based IPFE scheme computing inner products over integers to be a scheme modulo a prime p. Then, we refine a proof of Abdalla et al. [2] with a tight reduction in the QROM. Intuitively, Abdalla et al.'s proof modifies answers of secret key queries and random oracle queries by following Gentry et al.'s proof of their IBE scheme [20], further modifies an answer of a challenge query by following Wang et al.'s proof of their IPFE scheme [36], then employ Agrawal et al.'s information-theoretic argument [7]. In contrast, we utilize Katsumata et al.'s proof [26] that proves the tight AD security of Gentry et al.'s IBE scheme [20] in the QROM.

Intuitively, our proof modifies answers to quantum random oracle queries and secret key queries by following Katsumata et al.'s proof [26], further modifies an answer of a challenge query by combining Katsumata et al.'s proof [26] and Wang et al.'s proof [36], then employ Agrawal et al.'s information-theoretic argument [7]. As a result, we prove the tight AD-IND security of the proposed modification of Abdalla et al.'s IBIPFE scheme [2] in the QROM. Finally, we apply the proposed IND-to-SIM transformation and obtain the first AD-SIM-secure LWE-based IBIPFE scheme that has a tight reduction in the QROM.

Since Abdalla et al.'s AD-IND-secure SXDH-based IBIPFE scheme [2] computes inner products modulo a prime p, we can apply the proposed IND-to-SIM transformation and obtain the first AD-SIM-secure pairing-based IBIPFE scheme under the SXDH assumption in the standard model. However, the reduction is not tight. To achieve a tight reduction and improved efficiency, we propose an AD-IND-secure pairing-based IBIPFE scheme under the decisional bilinear Diffie-Hellman (DBDH) assumption in the ROM. Towards the best efficiency, the proposed pairing-based construction is direct in the sense that we do not rely on the proposed IND-to-SIM transformation that requires $(L+1)$-dimensional IBIPFE scheme as a building block. The proposed DBDH-based IBIPFE scheme is a combination of Coron's tightly AD-secure DBDH-based IBE scheme in ROM [16] and Agrawal et al.'s AD-SIM-secure DDH-based IPFE scheme [7]. We use Coron's IBE scheme among various (almost) tightly AD-secure pairing-based IBE schemes [9,10,15,16,21,22,23,24] since the scheme is the most efficient to the best of our knowledge. Moreover, structures of Coron's IBE scheme and Agrawal et al.'s IPFE scheme [7] are compatible to prove the tight AD-SIM security of the proposed IBIPFE scheme. Indeed, the proof is simple. Our proof modifies answers to random oracle queries, secret key queries, and a challenge query by following Coron's proof [16], then employing Agrawal et al.'s information-theoretic argument [7].

Table 1 compares the proposed IBIPFE schemes and known IBIPFE schemes [2,27]. Abdala et al.'s schemes [2] denoted by ACGU20 consist of two lattice-based schemes and two pairing-based schemes, where the former schemes satisfy either AD-IND security in the ROM or SEL-IND security in the standard model and the latter schemes satisfy either SEL-SIM security or AD-IND security in the standard model. Lai et al.'s scheme [27] denoted by LLW21 is a lattice-based scheme that satisfies AD-IND security in the standard model. The proposed schemes consist of two lattice-based schemes and two pairing-based schemes. The first lattice-based scheme is a mild modification of the AD-IND-secure Abdala et al.'s scheme [2] in the ROM, while the second scheme is the same as the first scheme applied by our proposed IND-to-SIM transformation. The third pairing-based scheme is the same as the AD-IND-secure Abdala et al.'s scheme [2] in the standard model applied by our proposed IND-to-SIM transformation, while the fourth pairing-based scheme or our original scheme. Among them, our second lattice-based scheme, our third pairing-based scheme, and our fourth pairing-based scheme are the only schemes satisfying AD-SIM security. Although Abdala et al.'s lattice-based SEL-IND-secure scheme is the only known scheme with a

Table 1. Security comparison among IBIPFE schemes

Scheme	security	reduction loss	model	assumption
ACGU20 [2]	AD-IND	$O(Q_{\mathsf{H}}^2)$	ROM	LWE
	SEL-IND	$O(1)$	standard	LWE
	SEL-SIM	$O(Q_{\mathsf{sk}})$	standard	SXDH
	AD-IND	$O(Q_{\mathsf{sk}})$	standard	SXDH
LLW21 [27]	AD-IND	$O(Q_{\mathsf{H}})$	standard	LWE
Section 4	AD-IND	$O(1)$	QROM	LWE
Section 3 + Section 4	AD-SIM	$O(1)$	QROM	LWE
Section 3 + ACGU20 [2]	AD-SIM	$O(Q_{\mathsf{sk}})$	standard	SXDH
Section 5	AD-SIM	$O(1)$	ROM	DBDH

Q_{H} and Q_{sk} denote the number of random oracle hash queries and secret key queries, respectively.

tight reduction, our first lattice-based scheme, our second lattice-based scheme, and our fourth pairing-based scheme satisfy either AD-IND security or AD-SIM security with tight reductions.

1.3 Technical Overview

We explain an overview of our proposed IND-to-SIM transformation. For simplicity, we explain an overview in the case of plain IPFE.
Agrawal et al.'s Transformation [6]. At first, we explain Agrawal et al.'s IND-to-SIM transformation [6]. Let $\mathbf{x}^\star \in \mathbb{Z}_p^L$ denote a challenge plaintext and $\mathbf{y}_1, \dots, \mathbf{y}_{L-1} \in \mathbb{Z}_p^L$ denote linearly independent vectors on which the adversary makes secret key queries in this order. In the real security game, a SIM-secure challenge ciphertext is an IND-secure encryption of a vector

$$\mathsf{R}.\mathbf{x}^\star = [\mathbf{x}^\star, \underbrace{0, \dots, 0}_{L\ 0\text{'s}}] \in \mathbb{Z}_p^{2L},$$

while a SIM-secure i-th secret key for $\mathbf{y}_i$ is an IND-secure secret key of a vector

$$\mathsf{R}.\mathbf{y}_i = [\mathbf{y}_i, \underbrace{0, \dots, 0}_{i-1\ 0\text{'s}}, 1, \underbrace{0, \dots, 0}_{L-1-i\ 0\text{'s}}, t_i] \in \mathbb{Z}_p^{2L},$$

where t_i is a $\mathbb{Z}_p$ random element. There are enough slots in $2L$-dimensional vectors to answer secret key queries on at most $L-1$ linearly independent vectors.

Hereafter, we explain the case of the ideal security game. Let Q_{pre} denote the number of adversary's pre-challenge secret key queries. Upon an adversary's pre-challenge secret key query on $\mathbf{y}_i$, the simulator creates an IND-secure secret key of a vector

$$\mathsf{I}.\mathbf{y}_i = [\mathbf{y}_i, \underbrace{0,\ldots,0}_{i-1\ 0\text{'s}}, 1, \underbrace{0,\ldots,0}_{L-1-i\ 0\text{'s}}, t_i] \in \mathbb{Z}_p^{2L}$$

as the case of the real security game. In the ideal security game, the simulator should answer a challenge query without the knowledge of a challenge plaintext $\mathbf{x}^\star$. In turn, the adversary declares $\{z_i\}_{i\in[Q_{\mathsf{pre}}]}$ such that

$$z_i = \langle \mathbf{x}^\star, \mathbf{y}_i \rangle \mod p.$$

Then, the simulator creates an IND-secure encryption of a vector

$$\mathsf{I}.\mathbf{x}^\star = [\widehat{\mathbf{x}}, -t_1, \ldots, -t_{L-1}, 1] \in \mathbb{Z}_p^{2L}$$

such that

$$\langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle = z_i \mod p$$

for all $i \in [Q_{\mathsf{pre}}]$. Upon an adversary's post-challenge secret key query on $\mathbf{y}_i$, , the adversary declares $z_i = \langle \mathbf{x}^\star, \mathbf{y}_i \rangle \mod p$ in addition to $\mathbf{y}_i$. Then, the simulator creates an IND-secure secret key of a vector

$$\mathsf{I}.\mathbf{y}_i = [\mathbf{y}_i, \underbrace{0,\ldots,0}_{i-1\ 0\text{'s}}, 1, \underbrace{0,\ldots,0}_{L-1-i\ 0\text{'s}}, t_i + z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle] \in \mathbb{Z}_p^{2L},$$

where t_i is a $\mathbb{Z}_p$ random element.

The proof of indistinguishability between real and ideal security games consists of two steps. At first, we show that the distribution of the real security game does not change even when secret key queries are answered as in the ideal security game. To this end, we have to ensure that decryption results are consistent between the challenge ciphertext in the real security game and secret keys in the ideal security game. We can easily check the fact since it holds that

$$\langle \mathsf{R}.\mathbf{x}^\star, \mathsf{I}.\mathbf{y}_i \rangle = \langle \mathbf{x}^\star, \mathbf{y}_i \rangle \mod p$$

since all the last L coordinates of $\mathsf{R}.\mathbf{x}^\star$ are 0. Moreover, although the last coordinate t_i of $\mathsf{R}.\mathbf{y}_i$ in the real security game is replaced with $t_i + z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle \mod p$ of post-challenge $\mathsf{I}.\mathbf{y}_i$, their distributions are the same due to the randomness of t_i.

To complete the proof, we have to show that the modified security game and the ideal security game are computationally indistinguishable. To utilize the IND security, we have to ensure that decryption results are consistent between ciphertexts and secret keys in the ideal security game. In the case of pre-challenge secret keys, we have

$$\langle \mathsf{I}.\mathbf{x}^\star, \mathsf{I}.\mathbf{y}_i \rangle = \langle [\widehat{\mathbf{x}}, -t_1, \ldots, -t_{L-1}, 1], [\mathbf{y}_i, \underbrace{0,\ldots,0}_{i-1\ 0\text{'s}}, 1, \underbrace{0,\ldots,0}_{L-1-i\ 0\text{'s}}, t_i] \rangle \mod p$$

$$
\begin{aligned}
&= \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle + \langle [-t_1, \ldots, -t_{L-1}, 1], [\underbrace{0, \ldots, 0}_{i-1\ 0\text{'s}}, 1, \underbrace{0, \ldots, 0}_{L-1-i\ 0\text{'s}}, t_i] \rangle \mod p \\
&= \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle \mod p \\
&= z_i \mod p.
\end{aligned}
$$

In the case of post-challenge secret keys, we have

$$
\begin{aligned}
&\langle \mathsf{I}.\mathbf{x}^\star, \mathsf{I}.\mathbf{y}_i \rangle \\
&= \langle [\widehat{\mathbf{x}}, -t_1, \ldots, -t_{L-1}, 1], [\mathbf{y}_i, \underbrace{0, \ldots, 0}_{i-1\ 0\text{'s}}, 1, \underbrace{0, \ldots, 0}_{L-1-i\ 0\text{'s}}, t_i + z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle] \rangle \mod p \\
&= \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle + \langle [-t_1, \ldots, -t_{L-1}, 1], [\underbrace{0, \ldots, 0}_{i-1\ 0\text{'s}}, 1, \underbrace{0, \ldots, 0}_{L-1-i\ 0\text{'s}}, t_i + z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle] \rangle \mod p \\
&= \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle + z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle \mod p \\
&= z_i \mod p.
\end{aligned}
$$

Then, we can utilize the IND security to show that two games are computationally indistinguishable.

Proposed Transformation. We observe how Agrawal et al.'s transformation [6] utilizes the additional L dimensions. To answer all secret key queries, we have to use $L-1$ $\mathbb{Z}_p$ random elements $t_1, \ldots, t_{L-1}$ since their randomness ensured that $\mathsf{R}.\mathbf{y}_i$ and post-challenge $\mathsf{I}.\mathbf{y}_i$ follow the same distribution. It was easy to design a real ciphertext so that decryption results are consistent between ciphertexts in the real security game and secret keys in the ideal security game by setting 0 for all additional L coordinates. Then, the additional L dimensions are used to ensure that decryption results are consistent between ciphertexts and secret keys in the ideal security game. In the challenge ciphertext, the additional $L-1$ coordinates are used to embed $t_1, \ldots, t_{L-1}$, while the last coordinate is 1. Since pre-challenge $\mathsf{I}.\mathbf{y}_i$ is the same as $\mathsf{R}.\mathbf{y}_i$, the additional L dimensions are used to ensure that inner products between $\mathsf{I}.\mathbf{x}$ and $\mathsf{I}.\mathbf{y}_i$ in the additional L dimensions becomes zero. Similarly, post-challenge $\mathsf{I}.\mathbf{y}_i$ utilize the additional L dimensions so that the inner products $z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle$ ensure that decryption results are consistent. Therefore, the additional L dimensions play a vital role in ensuring decryption consistency.

To design IND-to-SIM transformation by using $(L+1)$-dimensional IND-secure IPFE scheme, the main idea is how to use not only the additional dimensions but the first L dimensions to ensure the decryption consistency. In the real security game, we use vectors

$$\mathsf{R}.\mathbf{x}^\star = [\mathbf{x}^\star, 0] \in \mathbb{Z}_p^{L+1}, \qquad \mathsf{R}.\mathbf{y}_i = [\mathbf{y}_i, t_i] \in \mathbb{Z}_p^{L+1}$$

to create a challenge ciphertext which is an encryption of $\mathbf{x}^\star$ and i-th secret keys of $\mathbf{y}_i$ in the real security game, respectively. In the ideal security game, we use vectors

$$\mathsf{I}.\mathbf{y}_i = [\mathbf{y}_i, t_i], \qquad \mathsf{I}.\mathbf{x}^\star = [\widehat{\mathbf{x}}, 1], \qquad \mathsf{I}.\mathbf{y}_i = [\mathbf{y}_i, z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle],$$

to create pre-challenge i-th secret keys of $\mathbf{y}_i$, a challenge ciphertext, and post-challenge i-th secret keys of $\mathbf{y}_i$ in the ideal security game, respectively. If we set $\widehat{\mathbf{x}}$ such that

$$\langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle = z_i - t_i \mod p$$

for $i \in [Q_{\mathsf{pre}}]$, decryption results are consistent between ciphertexts and secret keys in the ideal security game. In the case of pre-challenge secret keys, we have

$$\begin{aligned} \langle \mathsf{I}.\mathbf{x}^\star, \mathsf{I}.\mathbf{y}_i \rangle &= \langle [\widehat{\mathbf{x}}, 1], [\mathbf{y}_i, t_i] \rangle \mod p \\ &= \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle + t_i \mod p \\ &= z_i - t_i + t_i \mod p \\ &= z_i \mod p. \end{aligned}$$

In the case of post-challenge secret keys, we have

$$\begin{aligned} \langle \mathsf{I}.\mathbf{x}^\star, \mathsf{I}.\mathbf{y}_i \rangle &= \langle [\widehat{\mathbf{x}}, 1], [\mathbf{y}_i, z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle] \rangle \mod p \\ &= \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle + z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle \mod p \\ &= z_i \mod p. \end{aligned}$$

However, this approach is problematic since the last coordinates $z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle$ of post-challenge $\mathsf{I}.\mathbf{y}_i$ are not random. Thus, distributions in the real and ideal security games are not the same.

To resolve the issue, we show that $\widehat{\mathbf{x}}$ can be defined so that the vector has sufficient entropy. We recall that a proof of Agrawal et al.'s transformation [6] consists of two steps, where the first step ensures that a distribution of the real security game does not change even when secret key queries are answered as in the ideal security game. In this step, we observe that the information of $\widehat{\mathbf{x}}$ itself is not given, while only $\langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle$ appears in the last coordinates of post-challenge $\mathsf{I}.\mathbf{y}_i$. Therefore, we set $\widehat{\mathbf{x}}$ as a random vector as long as it does not violate decryption consistency. Let $\bar{\mathbf{x}}, \mathbf{x}^\perp_{Q_{\mathsf{pre}}+1}, \ldots, \mathbf{x}^\perp_L \in \mathbb{Z}_p^L$ be vectors such that

$$\langle \bar{\mathbf{x}}, \mathbf{y}_i \rangle = z_i - t_i \mod p, \qquad \langle \mathbf{x}^\perp_j, \mathbf{y}_i \rangle = 0 \mod p$$

for $i \in [Q_{\mathsf{pre}}]$ and $\{\mathbf{x}^\perp_{Q_{\mathsf{pre}}+1}, \ldots, \mathbf{x}^\perp_L\}$ is linearly independent. We sample $\mathbb{Z}_p$ random $s_{Q_{\mathsf{pre}}+1}, \ldots, s_L$ and set

$$\widehat{\mathbf{x}} = \bar{\mathbf{x}} + \sum_{j \in [Q_{\mathsf{pre}}+1, L]} s_j \cdot \mathbf{x}^\perp_j \mod p.$$

Since it holds that

$$\langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle = \langle \bar{\mathbf{x}}, \mathbf{y}_i \rangle = z_i - t_i \mod p$$

for $i \in [Q_{\mathsf{pre}}]$, this $\widehat{\mathbf{x}}$ does not violate the decryption consistency as the above analysis. Moreover, the randomness of $s_{Q_{\mathsf{pre}}+1}, \ldots, s_L \in \mathbb{Z}_p$ ensure that

$$\langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle = \langle \bar{\mathbf{x}}, \mathbf{y}_i \rangle + \sum_{j \in [Q_{\mathsf{pre}}+1, L]} s_j \cdot \langle \mathbf{x}^\perp_j, \mathbf{y}_i \rangle \mod p$$

for $i \in [Q_{\mathsf{pre}}+1, L-1]$ follow the uniform distribution over $\mathbb{Z}_p^{L-Q_{\mathsf{pre}}-1}$. Therefore, distributions in the real and ideal security games are the same since the last coordinates $z_i - \langle \widehat{\mathbf{x}}, \mathbf{y}_i \rangle$ of post-challenge $\mathsf{I}.\mathbf{y}_i$ are random as in $\mathsf{R}.\mathbf{y}_i$. Thus, we can complete the first step of the proof. We can also prove the second step in the same way as Agrawal et al.'s transformation [6].

1.4 Organization

In Section 3, we propose the IND-to-SIM transformation. In Section 4, we propose our lattice-based IBIPFE scheme. In Section 5, we propose our pairing-based IBIPFE scheme.

2 Preliminaries

Notation. Let $\lambda \in \mathbb{N}$ denote the security parameter throughout the paper. Let uppercase (resp. lowercase) bold letter $\mathbf{A}$ (resp. $\mathbf{a}$) denote a matrix (resp. column vector). For a matrix $\mathbf{R} \in \mathbb{R}^{n \times n}$, let $\|\mathbf{R}\|$ denote the length of the longest column of $\mathbf{R}$ and let $\|\mathbf{R}\|_{\mathrm{GS}}$ denote the length of the longest column of the Gram-Schmidt orthogonalization of $\mathbf{R}$. Let $\langle \mathbf{x}, \mathbf{y} \rangle$ denote an inner product of $\mathbf{x}$ and $\mathbf{y}$. For vectors $\mathbf{a}_1, \ldots, \mathbf{a}_m \in \mathbb{Z}_p^n$, let $[\mathbf{a}_1 \mid \cdots \mid \mathbf{a}_m] \in \mathbb{Z}_p^{n \times m}$ denote their horizontal concatenation. For vectors $\mathbf{a} = [a_1, \ldots, a_n] \in \mathbb{Z}_p^n$ and $\mathbf{b} = [b_1, \ldots, b_m] \in \mathbb{Z}_p^m$, let $[\mathbf{a} \parallel \mathbf{b}] = [a_1, \ldots, a_n, b_1, \ldots, b_m] \in \mathbb{Z}_p^{n+m}$ denote their vertical concatenation. For a finite set S, let $s \leftarrow_R S$ denote the operation of sampling s from S uniformly at random. For a probability distribution $\mathcal{S}$, let $s \leftarrow \mathcal{S}$ denote the operation of sampling s according to $\mathcal{S}$. For two random variables X and Y over S, the statistical distance $\Delta(X, Y)$ between X and Y is defined as $\Delta(X, Y) \coloneqq \sum_{s \in S} |\Pr[X = s] - \Pr[Y = s]|$. We say that the two distributions X and Y are statistically close when $\Delta(X, Y)$ is negligible in the security parameter. The min-entropy of a random variable X is defined as $\mathbf{H}_\infty \coloneqq -\log(\max_x \Pr[X = x])$. For two sets $\mathcal{X}$ and $\mathcal{Y}$, let $\mathsf{Func}(\mathcal{X}, \mathcal{Y})$ denote the set of all functions from $\mathcal{X}$ to $\mathcal{Y}$. For an algorithm $\mathcal{A}$, let $\mathsf{Time}(\mathcal{A})$ denote the running time of $\mathcal{A}$.

Pseudo-Random Function.

Definition 1 (Pseudo-Random Function (PRF)). *Let $\mathcal{F} = \{f_k\}_{k \in \mathcal{K}}$ denote a function family such that $f_k : \mathcal{X} \to \mathcal{Y}$. A function family $\mathcal{F}$ is said to be a* PRF *family if for any quantum polynomial-time $\mathcal{A}$,*

$$\mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{F},\mathcal{A}}(\lambda) = \left| \Pr\left[\mathcal{A}^{f_k(\cdot)}(\lambda) \mid k \leftarrow_R \mathcal{K}\right] - \Pr\left[\mathcal{A}^{f(\cdot)}(\lambda) \mid f \leftarrow_R \mathsf{Func}(\mathcal{X}, \mathcal{Y})\right] \right|$$

is negligible in λ.

Identity-based Inner-Product Functional Encryption.

Syntax. An identity-based inner-product functional encryption (IBIPFE) scheme Π computing inner products modulo a prime number p consists of four PPT algorithms ($\mathsf{Setup}, \mathsf{Enc}, \mathsf{KGen}, \mathsf{Dec}$) defined as follows:

```
$(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, 1^L)$
$((\mathsf{id}_0^\star, \mathbf{x}_0^\star), (\mathsf{id}_1^\star, \mathbf{x}_1^\star)) \leftarrow \mathcal{A}^{\mathsf{KGen}(\mathsf{msk},\cdot,\cdot)}(\mathsf{mpk})$
$\mathsf{ct}^\star \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}_\beta^\star, \mathbf{x}_\beta^\star); \beta \leftarrow_R \{0,1\}$
$\widehat{\beta} \leftarrow \mathcal{A}^{\mathsf{KGen}(\mathsf{msk},\cdot,\cdot)}(\mathsf{mpk}, \mathsf{ct}^\star)$
```

Fig. 1. The AD-IND security game

$\mathsf{Setup}(1^\lambda, 1^L) \to (\mathsf{mpk}, \mathsf{msk})$**:** On input the security parameter λ and a dimension L of an inner product, output a master public/secret key pair $(\mathsf{mpk}, \mathsf{msk})$, where mpk implicitly contains an identity space $\mathcal{ID}$ and a prime number p.

$\mathsf{Enc}(\mathsf{mpk}, \mathsf{id}, \mathbf{x}) \to \mathsf{ct}_{\mathsf{id},\mathbf{x}}$**:** On input the master public key mpk, an identity $\mathsf{id} \in \mathcal{ID}$, and a vector $\mathbf{x} \in \mathbb{Z}_p^L$, output a ciphertext $\mathsf{ct}_{\mathsf{id},\mathbf{x}}$.

$\mathsf{KGen}(\mathsf{msk}, \mathsf{id}, \mathbf{y}) \to \mathsf{sk}_{\mathsf{id},\mathbf{y}}$**:** On input the master secret key msk, an identity $\mathsf{id} \in \mathcal{ID}$, and a vector $\mathbf{y} \in \mathbb{Z}_p^L$, output a secret key $\mathsf{sk}_{\mathsf{id},\mathbf{y}}$.

$\mathsf{Dec}(\mathsf{mpk}, \mathsf{ct}_{\mathsf{id},\mathbf{x}}, \mathsf{sk}_{\mathsf{id}',\mathbf{y}}) \to \langle \mathbf{x}, \mathbf{y} \rangle / \bot$**:** On input a master public key mpk, a ciphertext $\mathsf{ct}_{\mathsf{id},\mathbf{x}}$, and a secret key $\mathsf{sk}_{\mathsf{id}',\mathbf{y}}$, output a decryption result $\langle \mathbf{x}, \mathbf{y} \rangle \in \mathbb{Z}_p$ or a failure symbol $\bot$.

Correctness. For all the security parameter $\lambda \in \mathbb{N}$, a master public/secret key pair $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, 1^L)$, an identity $\mathsf{id} \in \mathcal{ID}$, and two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{Z}_p^L$, it holds that

$$\mathsf{Dec}(\mathsf{mpk}, \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}, \mathbf{x}), \mathsf{KGen}(\mathsf{msk}, \mathsf{id}, \mathbf{y})) = \langle \mathbf{x}, \mathbf{y} \rangle \mod p$$

with overwhelming probability.

Security. We review the adaptive indistinguishability ($\mathsf{AD\text{-}IND}$) security and the adaptive simulation ($\mathsf{AD\text{-}SIM}$) security. To define them, we use the following function

$$\mathsf{Eval}(\mathsf{id}^\star, \mathbf{x}^\star, \mathsf{id}, \mathbf{y}) = \begin{cases} \langle \mathbf{x}^\star, \mathbf{y} \rangle & \text{if } \mathsf{id} = \mathsf{id}^\star \\ \bot & \text{if } \mathsf{id} \neq \mathsf{id}^\star \end{cases}.$$

The AD-IND security ensures that any PPT adversary $\mathcal{A}$ cannot distinguish encryptions of $(\mathsf{id}_0^\star, \mathbf{x}_0^\star)$ and $(\mathsf{id}_1^\star, \mathbf{x}_1^\star)$ even when $\mathcal{A}$ can receive polynomially many secret keys for $(\mathsf{id}, \mathbf{y})$ such that $\mathsf{Eval}(\mathsf{id}_0^\star, \mathbf{x}_0^\star, \mathsf{id}, \mathbf{y}) = \mathsf{Eval}(\mathsf{id}_1^\star, \mathbf{x}_1^\star, \mathsf{id}, \mathbf{y})$. If $\mathsf{id}_0^\star \neq \mathsf{id}_1^\star$ holds, $\mathcal{A}$ cannot receive secret keys for any $(\mathsf{id}, \mathbf{y})$ such that $\mathsf{id} \in \{\mathsf{id}_0^\star, \mathsf{id}_1^\star\}$. If $\mathsf{id}_0^\star = \mathsf{id}_1^\star = \mathsf{id}^\star$ holds, let $\mathbf{y}_{\mathsf{id}^\star,1}, \ldots, \mathbf{y}_{\mathsf{id}^\star,Q_{\mathsf{id}^\star}} \in \mathbb{Z}_p^L$ denote all vectors such that $\mathcal{A}$ receives secret keys for $(\mathsf{id}^\star, \mathbf{y}_{\mathsf{id}^\star,i})$ throughout the security game. To satisfy $\mathsf{Eval}(\mathsf{id}_0^\star, \mathbf{x}_0^\star, \mathsf{id}, \mathbf{y}) = \mathsf{Eval}(\mathsf{id}_1^\star, \mathbf{x}_1^\star, \mathsf{id}, \mathbf{y})$, a set $\{\mathbf{y}_{\mathsf{id}^\star,1}, \ldots, \mathbf{y}_{\mathsf{id}^\star,Q_{\mathsf{id}^\star}}\}$ contains at most $L-1$ linearly independent vectors. We note that the challenge ciphertext cannot hide the information of $\mathsf{id}^\star$ unlike IBE if $\mathsf{id}_0^\star = \mathsf{id}_1^\star = \mathsf{id}^\star$ holds since $\mathcal{A}$ can receive secret keys associated with $\mathsf{id}^\star$.

Definition 2 (AD-IND **Security).** *The* AD-IND *security is defined by the security game between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ summarized in Figure 1. $\mathcal{A}$ can access an oracle* KGen *that takes* $(\mathsf{id}, \mathbf{y})$ *as input and outputs* $\mathsf{sk}_{\mathsf{id},\mathbf{y}} \leftarrow \mathsf{KGen}(\mathsf{msk}, \mathsf{id}, \mathbf{y})$*, where it is required that* $\mathsf{Eval}(\mathsf{id}_0^\star, \mathbf{x}_0^\star, \mathsf{id}, \mathbf{y}) = \mathsf{Eval}(\mathsf{id}_1^\star, \mathbf{x}_1^\star, \mathsf{id}, \mathbf{y})$ *throughout the game. An* IBIPFE *scheme Π is said to satisfy the* AD-IND *security if for any PPT $\mathcal{A}$,*

$$\mathsf{Adv}_{\Pi,\mathcal{A}}^{\mathsf{AD\text{-}IND}}(\lambda) = \left|\Pr\left[\widehat{\beta} = 1 \mid \beta = 0\right] - \Pr\left[\widehat{\beta} = 1 \mid \beta = 1\right]\right|$$

is negligible in λ.

Although the AD-IND security ensures that any PPT adversary $\mathcal{A}$ cannot distinguish encryptions of $(\mathsf{id}_0^\star, \mathbf{x}_0^\star)$ and $(\mathsf{id}_1^\star, \mathbf{x}_1^\star)$, the fact does not ensure that the adversary cannot learn anything besides $\mathsf{Eval}(\mathsf{id}_0^\star, \mathbf{x}_0^\star, \mathsf{id}, \mathbf{y}) = \mathsf{Eval}(\mathsf{id}_1^\star, \mathbf{x}_1^\star, \mathsf{id}, \mathbf{y})$. In contrast, the AD-SIM security ensures that there is a PPT simulator $\mathcal{S}$ that does not take $\mathbf{x}^\star$ but only evaluation results $\{\mathsf{Eval}(\mathsf{id}^\star, \mathbf{x}^\star, \mathsf{id}, \mathbf{y})\}_{(\mathsf{id},\mathbf{y})}$ between the challenge ciphertext and all secret keys $\mathcal{A}$ receives as input and simulates the challenge ciphertext $\mathsf{ct}^\star \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}^\star, \mathbf{x}^\star)$. In other words, the AD-SIM security ensures that $\mathcal{A}$ cannot extract any additional information from the challenge ciphertext since $\mathcal{A}$ also knows $\{\mathsf{Eval}(\mathsf{id}^\star, \mathbf{x}^\star, \mathsf{id}, \mathbf{y})\}_{(\mathsf{id},\mathbf{y})}$. We follow Agrawal et al.'s definition of AD-SIM security for IPFE [6] and define the following AD-SIM security for IBIPFE.

Definition 3 (AD-SIM **Security).** *The* AD-SIM *security is defined by two security games summarized in Figure 2, where the left (resp. right) is the real security game* $\mathsf{SIM}_{\mathsf{Real}}$ *(resp. ideal security game* $\mathsf{SIM}_{\mathsf{Ideal}}$*).*

- *In* $\mathsf{SIM}_{\mathsf{Real}}$ *between $\mathcal{A}$ and a challenger $\mathcal{C}$, $\mathcal{C}$ runs real algorithms* $(\mathsf{Setup}, \mathsf{Enc}, \mathsf{KGen})$ *of an* IBIPFE *scheme Π and $\mathcal{A}$ can access an oracle* KGen *that takes* $(\mathsf{id}, \mathbf{y})$ *as input and outputs* $\mathsf{sk}_{\mathsf{id},\mathbf{y}} \leftarrow \mathsf{KGen}(\mathsf{msk}, \mathsf{id}, \mathbf{y})$.
- *In* $\mathsf{SIM}_{\mathsf{Ideal}}$ *between $\mathcal{A}$ and a simulator $\mathcal{S}$, $\mathcal{S}$ runs PPT simulation algorithms* $(\mathsf{Setup}^\star, \mathsf{KGen}_0^\star, \mathsf{Enc}^\star, \mathsf{KGen}_1^\star)$ *and $\mathcal{A}$ can access an oracle* $\mathsf{KGen}_0^\star$ *(resp.* $\mathsf{KGen}_1^\star$*) that takes* $(\mathsf{id}, \mathbf{y})$ *(resp.* $(\mathsf{id}, \mathbf{y}, \mathsf{Eval}(\mathsf{id}^\star, \mathbf{x}^\star, \mathsf{id}, \mathbf{y}))$*) as input and outputs* $\mathsf{sk}_{\mathsf{id},\mathbf{y}}^\star \leftarrow \mathsf{KGen}_0^\star(\mathsf{mpk}^\star, \mathsf{msk}^\star, \mathsf{id}, \mathbf{y}, \mathsf{st})$ *(resp.* $\mathsf{sk}_{\mathsf{id},\mathbf{y}}^\star \leftarrow \mathsf{KGen}_1^\star(\mathsf{mpk}^\star, \mathsf{msk}^\star, \mathsf{id}, \mathbf{y}, \mathsf{Eval}(\mathsf{id}^\star, \mathbf{x}^\star, \mathsf{id}, \mathbf{y}), \mathsf{st})$*). Moreover, let*

$$\mathcal{V} = (\mathbf{y}_{\mathsf{id}^\star,i}, z_i = \langle \mathbf{x}^\star, \mathbf{y}_{\mathsf{id}^\star,i} \rangle)_{i \in [Q_{\mathsf{id}^\star}]},$$

where $\{\mathbf{y}_{\mathsf{id}^\star,i}\}_{i \in [Q_{\mathsf{id}^\star}]}$ *denote all vectors on which $\mathcal{A}$ has made* $\mathsf{KGen}_0^\star$ *oracle queries associated with* $\mathsf{id}^\star$.

An IBIPFE *scheme Π is said to satisfy the* AD-SIM *security if for any PPT $\mathcal{A}$,*

$$\mathsf{Adv}_{\Pi,\mathcal{A}}^{\mathsf{AD\text{-}IND}}(\lambda) = |\Pr[\beta = 1 \mid \mathsf{SIM}_{\mathsf{Real}}] - \Pr[\beta = 1 \mid \mathsf{SIM}_{\mathsf{Ideal}}]|$$

is negligible in λ.

In this paper, we may call $\mathcal{A}$'s oracle access to $\mathsf{KGen}_0^\star$ (resp. $\mathsf{KGen}_1^\star$) a pre-challenge (resp. post-challenge) secret key query.

$\mathsf{SIM}_{\mathsf{Real}}$	$\mathsf{SIM}_{\mathsf{Ideal}}$
$(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, 1^L)$	$(\mathsf{mpk}^\star, \mathsf{msk}^\star) \leftarrow \mathsf{Setup}^\star(1^\lambda, 1^L)$
$(\mathsf{id}^\star, \mathbf{x}^\star) \leftarrow \mathcal{A}^{\mathsf{KGen}(\mathsf{msk},\cdot,\cdot)}(\mathsf{mpk})$	$(\mathsf{id}^\star, \mathcal{V}) \leftarrow \mathcal{A}^{\mathsf{KGen}_0^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\cdot,\cdot,\mathsf{st})}(\mathsf{mpk}^\star)$
$\mathsf{ct}^\star \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}^\star, \mathbf{x}^\star)$	$\mathsf{ct}^\star \leftarrow \mathsf{Enc}^\star(\mathsf{mpk}^\star, \mathsf{id}^\star, \mathcal{V})$
$\beta \leftarrow \mathcal{A}^{\mathsf{KGen}(\mathsf{msk},\cdot,\cdot)}(\mathsf{mpk}, \mathsf{ct}^\star)$	$\beta \leftarrow \mathcal{A}^{\mathsf{KGen}_1^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\cdot,\cdot,\cdot,\mathsf{st})}(\mathsf{mpk}^\star, \mathsf{ct}^\star)$

Fig. 2. The AD-SIM security game

Remark 1. Lin and Luo [28] claimed that they proposed a transformation from an $(L+1)$-dimensional IND-secure IPFE scheme to an L-dimensional SIM-secure IPFE scheme. However, their definition of the AD-SIM security is weaker than Agrawal et al. [6]. Briefly speaking, not only $\mathsf{KGen}_1^\star$ but also $\mathsf{KGen}_0^\star$ can take $\mathsf{Eval}(\mathsf{id}^\star, \mathbf{x}^\star, \mathsf{id}, \mathbf{y})$ in Lin and Luo's definition. In other words, the definition is not fully adaptive since candidates of the challenge $\mathbf{x}^\star$ decrease by accessing the $\mathsf{KGen}_0^\star$ oracle.

3 IND-to-SIM Transformation

In this section, we propose our generic transformation from an $(L + 1)$-dimensional AD-IND-secure IBIPFE scheme to an L-dimensional AD-SIM-secure IBIPFE scheme computing inner products modulo a prime p. We give a construction in Section 3.1 and prove the security in Section 3.2.

3.1 Construction

We use an $(L+1)$-dimensional AD-IND-secure IBIPFE scheme $\Pi_{\mathsf{IND}} = (\mathsf{IND.Setup}, \mathsf{IND.Enc}, \mathsf{IND.KGen}, \mathsf{IND.Dec})$ that computes an inner product modulo a prime p and construct an L-dimensional AD-SIM-secure IBIPFE scheme $\Pi_{\mathsf{SIM}} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{KGen}, \mathsf{Dec})$ that computes an inner product modulo a prime p as follows.

$\mathsf{Setup}(1^\lambda, 1^L) \rightarrow (\mathsf{mpk}, \mathsf{msk})$**:** Run $(\mathsf{IND.mpk}, \mathsf{IND.msk}) \leftarrow \mathsf{IND.Setup}(1^\lambda, 1^{L+1})$, choose an index $k \leftarrow_R \mathcal{K}$ for a function family $\mathcal{F} = \{f_k\}_{k\in\mathcal{K}}$ such that $f_k : \mathcal{ID} \rightarrow \mathbb{Z}_p^L$, and output

$$\mathsf{mpk} = \mathsf{IND.mpk}, \qquad \mathsf{msk} = (\mathsf{IND.msk}, k).$$

$\mathsf{Enc}(\mathsf{mpk}, \mathsf{id}, \mathbf{x}) \rightarrow \mathsf{ct}_{\mathsf{id},\mathbf{x}}$**:** Parse $\mathsf{mpk} = \mathsf{IND.mpk}$. Run

$$\mathsf{IND.ct}_{\mathsf{id},[\mathbf{x}\|0]} \leftarrow \mathsf{IND.Enc}(\mathsf{IND.mpk}, \mathsf{id}, [\mathbf{x} \parallel 0]),$$

and output $\mathsf{ct}_{\mathsf{id},\mathbf{x}} = \mathsf{IND.ct}_{\mathsf{id},[\mathbf{x}\|0]}$.

$\mathsf{KGen}(\mathsf{msk}, \mathsf{id}, \mathbf{y}) \to \mathsf{sk}_{\mathsf{id},\mathbf{y}}$**:** Parse $\mathsf{msk} = (\mathsf{IND.msk}, k)$. Compute $f_k(\mathsf{id}) = \mathbf{r}_{\mathsf{id}} \in \mathbb{Z}_p^L$, set

$$\widehat{\mathbf{y}} = [\mathbf{y} \parallel \langle \mathbf{r}_{\mathsf{id}}, \mathbf{y} \rangle \mod p],$$

run

$$\mathsf{IND.sk}_{\mathsf{id},\widehat{\mathbf{y}}} \leftarrow \mathsf{IND.KGen}(\mathsf{msk}, \mathsf{id}, \widehat{\mathbf{y}})$$

and output $\mathsf{sk}_{\mathsf{id},\mathbf{y}} = (\widehat{\mathbf{y}}, \mathsf{IND.sk}_{\mathsf{id},\widehat{\mathbf{y}}})$.

$\mathsf{Dec}(\mathsf{mpk}, \mathsf{ct}_{\mathsf{id},\mathbf{x}}, \mathsf{sk}_{\mathsf{id},\mathbf{y}}) \to \langle \mathbf{x}, \mathbf{y} \rangle$**:** Parse $\mathsf{ct}_{\mathsf{id},\mathbf{x}} = \mathsf{IND.ct}_{\mathsf{id},[\mathbf{x}\parallel 0]}$ and $\mathsf{sk}_{\mathsf{id},\mathbf{y}} = (\widehat{\mathbf{y}}, \mathsf{IND.sk}_{\mathsf{id},\widehat{\mathbf{y}}})$. Output the result of $\mathsf{IND.Dec}(\mathsf{IND.mpk}, \mathsf{IND.ct}_{\mathsf{id},[\mathbf{x}\parallel 0]}, \mathsf{IND.sk}_{\mathsf{id},\widehat{\mathbf{y}}})$.

Correctness. Since it holds that

$$\langle [\mathbf{x} \parallel 0], [\mathbf{y} \parallel \langle \mathbf{r}_{\mathsf{id}}, \mathbf{y} \rangle \mod p] \rangle = \langle \mathbf{x}, \mathbf{y} \rangle \mod p,$$

the correctness of the underlying IBIPFE scheme $\Pi_{\mathsf{IND}} = (\mathsf{IND.Setup}, \mathsf{IND.Enc}, \mathsf{IND.KGen}, \mathsf{IND.Dec})$ ensures the correctness of the proposed scheme $\Pi_{\mathsf{SIM}} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{KGen}, \mathsf{Dec})$.

3.2 Security

To conclude this section, we prove the following theorem.

Theorem 1. *If the underlying* IBIPFE *scheme* Π_{IND} *satisfies the* AD-IND *security and* $\mathcal{F}$ *is a* PRF *family, then the proposed* IBIPFE *scheme* Π_{SIM} *in Section 3.1 satisfies the tight* AD-SIM *security. In particular, for any PPT* $\mathcal{A}$ *that breaks the* AD-SIM *security of* Π_{SIM}*, there exist PPT* $\mathcal{B}_1$ *and* $\mathcal{B}_2$ *such that*

$$\mathsf{Adv}^{\mathsf{AD\text{-}SIM}}_{\Pi_{\mathsf{SIM}},\mathcal{A}}(\lambda) \leq \mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{F},\mathcal{B}_1}(\lambda) + \mathsf{Adv}^{\mathsf{AD\text{-}IND}}_{\Pi_{\mathsf{IND}},\mathcal{B}_2}(\lambda)$$

and

$$\max\{\mathsf{Time}(\mathcal{B}_1), \mathsf{Time}(\mathcal{B}_2)\} = \mathsf{Time}(\mathcal{A}) + \mathsf{poly}(\lambda, L).$$

Proof of Theorem 1. At first, we define the following simulation algorithms $(\mathsf{Setup}^\star, \mathsf{KGen}_0^\star, \mathsf{Enc}^\star, \mathsf{KGen}_1^\star)$:

$\mathsf{Setup}^\star(1^\lambda, 1^L) \to (\mathsf{mpk}^\star, \mathsf{msk}^\star)$**:** Run $(\mathsf{IND.mpk}, \mathsf{IND.msk}) \leftarrow \mathsf{IND.Setup}(1^\lambda, 1^{L+1})$ and output

$$\mathsf{mpk}^\star = \mathsf{IND.mpk}, \qquad \mathsf{msk}^\star = \mathsf{IND.msk}.$$

$\mathsf{KGen}_0^\star(\mathsf{mpk}^\star, \mathsf{msk}^\star, \mathsf{id}, \mathbf{y}, \mathsf{st}) \to \mathsf{sk}^\star_{\mathsf{id},\mathbf{y}}$**:** Parse $\mathsf{mpk}^\star = \mathsf{IND.mpk}$ and $\mathsf{msk}^\star = \mathsf{IND.msk}$. Retrieve $(\mathsf{id}, \{\mathbf{y}_{\mathsf{id},i}, t_{\mathsf{id},i}\}_{i\in[Q]}) \in \mathsf{st}$, where $\mathbf{y}_{\mathsf{id},1}, \ldots, \mathbf{y}_{\mathsf{id},Q}$ denote all linearly independent vectors on which $\mathcal{A}$ has made secret key queries associated with id so far.

– If there exist $c_1, \dots, c_Q \in \mathbb{Z}_p$ such that $\mathbf{y} = \sum_{i \in [Q]} c_i \cdot \mathbf{y}_{\mathsf{id},i} \mod p$, set

$$\widehat{\mathbf{y}} = [\mathbf{y} \parallel \sum_{i \in [Q]} c_i \cdot t_{\mathsf{id},i} \mod p].$$

– If there do not exist $c_1, \dots, c_Q \in \mathbb{Z}_p$ such that $\mathbf{y} = \sum_{i \in [Q]} c_i \cdot \mathbf{y}_{\mathsf{id},i} \mod p$, sample $t_{\mathsf{id},Q+1} \leftarrow_R \mathbb{Z}_p$, set $\mathbf{y}_{\mathsf{id},Q+1} = \mathbf{y}$ and

$$\widehat{\mathbf{y}} = [\mathbf{y}_{\mathsf{id},Q+1} \parallel t_{\mathsf{id},Q+1}],$$

and update $(\mathsf{id}, \{\mathbf{y}_{\mathsf{id},i}, t_{\mathsf{id},i}\}_{i \in [Q]}) \in \mathsf{st}$ by $(\mathsf{id}, \{\mathbf{y}_{\mathsf{id},i}, t_{\mathsf{id},i}\}_{i \in [Q+1]})$.

Run

$$\mathsf{IND.sk}_{\mathsf{id},\widehat{\mathbf{y}}} \leftarrow \mathsf{IND.KGen}(\mathsf{msk}, \mathsf{id}, \widehat{\mathbf{y}})$$

and output $\mathsf{sk}^{\star}_{\mathsf{id},\mathbf{y}} = \mathsf{IND.sk}_{\mathsf{id},\widehat{\mathbf{y}}}$.

$\mathsf{Enc}^{\star}(\mathsf{mpk}^{\star}, \mathsf{msk}^{\star}, \mathsf{id}^{\star}, \mathcal{V}, \mathsf{st}) \to \mathsf{ct}^{\star}$: Parse $\mathsf{mpk}^{\star} = \mathsf{IND.mpk}$, $\mathsf{msk}^{\star} = \mathsf{IND.msk}$, and

$$\mathcal{V} = (\mathbf{y}_{\mathsf{id}^{\star},i}, z_i^{\mathsf{pre}} = \langle \mathbf{x}^{\star}, \mathbf{y}_{\mathsf{id}^{\star},i} \rangle)_{i \in [Q_{\mathsf{id}^{\star}}]}.$$

Suppose that a set $\{\mathbf{y}_{\mathsf{id}^{\star},1}, \dots, \mathbf{y}_{\mathsf{id}^{\star},Q_{\mathsf{id}^{\star}}}\}$ contains Q_{pre} linearly independent vectors and $\{\mathbf{y}_{\mathsf{id}^{\star},1}, \dots, \mathbf{y}_{\mathsf{id}^{\star},Q_{\mathsf{pre}}}\} \subseteq \{\mathbf{y}_{\mathsf{id}^{\star},1}, \dots, \mathbf{y}_{\mathsf{id}^{\star},Q_{\mathsf{id}^{\star}}}\}$ is linearly independent for simplicity. Retrieve $(\mathsf{id}^{\star}, \{\mathbf{y}_{\mathsf{id}^{\star},i}, t_{\mathsf{id}^{\star},i}\}_{i \in [Q_{\mathsf{pre}}]}) \in \mathsf{st}$. Let

$$\mathbf{Y}^{\mathsf{pre}} = [\mathbf{y}_{\mathsf{id}^{\star},1} \mid \cdots \mid \mathbf{y}_{\mathsf{id}^{\star},Q_{\mathsf{pre}}}] \in \mathbb{Z}_p^{L \times Q_{\mathsf{pre}}},$$
$$\mathbf{z}^{\mathsf{pre}} = [z_1^{\mathsf{pre}}, \dots, z_{Q_{\mathsf{pre}}}^{\mathsf{pre}}] \in \mathbb{Z}_p^{Q_{\mathsf{pre}}}, \qquad \mathbf{t}^{\mathsf{pre}}_{\mathsf{id}^{\star}} = [t_{\mathsf{id}^{\star},1}, \dots, t_{\mathsf{id}^{\star},Q_{\mathsf{pre}}}] \in \mathbb{Z}_p^{Q_{\mathsf{pre}}}.$$

Compute $\bar{\mathbf{x}}, \mathbf{x}^{\perp}_{Q_{\mathsf{pre}}+1}, \dots, \mathbf{x}^{\perp}_L \in \mathbb{Z}_p^L$ such that

$$\mathbf{Y}^{\mathsf{pre}\top} \cdot \bar{\mathbf{x}} = \mathbf{z}^{\mathsf{pre}} - \mathbf{t}^{\mathsf{pre}}_{\mathsf{id}^{\star}} \mod p, \qquad \mathbf{Y}^{\mathsf{pre}\top} \cdot \mathbf{x}^{\perp}_j = \mathbf{0} \mod p$$

for $j \in [Q_{\mathsf{pre}} + 1, L]$ and $\{\mathbf{x}^{\perp}_{Q_{\mathsf{pre}}+1}, \dots, \mathbf{x}^{\perp}_L\}$ is linearly independent, sample $s_{Q_{\mathsf{pre}}+1}, \dots, s_L \leftarrow_R \mathbb{Z}_p$, and set

$$\widehat{\mathbf{x}} = \bar{\mathbf{x}} + \sum_{j \in [Q_{\mathsf{pre}}+1, L]} s_j \cdot \mathbf{x}^{\perp}_j \mod p,$$

where it holds that

$$\begin{aligned}\langle [\widehat{\mathbf{x}} \parallel 1], [\mathbf{y}_{\mathsf{id}^{\star},i} \parallel t_{\mathsf{id}^{\star},i}] \rangle &= \underbrace{\langle \bar{\mathbf{x}}, \mathbf{y}_{\mathsf{id}^{\star},i} \rangle}_{= z_i^{\mathsf{pre}} - t_{\mathsf{id}^{\star},i} \mod p} + \sum_{j \in [Q_{\mathsf{pre}}+1, L]} s_j \cdot \underbrace{\langle \mathbf{x}^{\perp}_j, \mathbf{y}_{\mathsf{id}^{\star},i} \rangle}_{=0 \mod p} + t_{\mathsf{id}^{\star},i} \\ &= z_i^{\mathsf{pre}} \mod p\end{aligned}$$

for $i \in [Q_{\mathsf{pre}}]$. Run

$$\mathsf{IND.ct}_{\mathsf{id}^{\star},[\widehat{\mathbf{x}} \parallel 1]} \leftarrow \mathsf{IND.Enc}(\mathsf{IND.mpk}, \mathsf{id}^{\star}, [\widehat{\mathbf{x}} \parallel 1])$$

and output $\mathsf{ct}^{\star} = \mathsf{IND.ct}_{\mathsf{id}^{\star},[\widehat{\mathbf{x}} \parallel 1]}$.

$\mathsf{KGen}^{\star}_{1}(\mathsf{mpk}^{\star}, \mathsf{msk}^{\star}, \mathsf{id}, \mathbf{y}, z = \mathsf{Eval}(\mathsf{id}^{\star}, \mathbf{x}^{\star}, \mathsf{id}, \mathbf{y}), \mathsf{st}) \to \mathsf{sk}_{\mathsf{id},\mathbf{y}}$**:** Parse $\mathsf{mpk}^{\star} = \mathsf{IND.mpk}$ and $\mathsf{msk}^{\star} = \mathsf{IND.msk}$. Retrieve $(\mathsf{id}^{\star}, \{\mathbf{y}_{\mathsf{id}^{\star},i}, t_{\mathsf{id}^{\star},i}\}_{i\in[Q]}) \in \mathsf{st}$, where $\mathbf{y}_{\mathsf{id}^{\star},1}, \ldots, \mathbf{y}_{\mathsf{id}^{\star},Q}$ denote all linearly independent vectors on which $\mathcal{A}$ has made secret key queries associated with $\mathsf{id}^{\star}$ so far.

- If $\mathsf{id} \neq \mathsf{id}^{\star}$ holds or there exist $c_1, \ldots, c_Q \in \mathbb{Z}_p$ such that $\mathbf{y} = \sum_{i\in[Q]} c_i \cdot \mathbf{y}_{\mathsf{id}^{\star},i} \mod p$, set
$$\widehat{\mathbf{y}} = [\mathbf{y} \parallel \sum_{i\in[Q]} c_i \cdot t_{\mathsf{id}^{\star},i} \mod p]$$
as in $\mathsf{KGen}^{\star}_{0}$.
- If $\mathsf{id} = \mathsf{id}^{\star}$ holds and there do not exist $c_1, \ldots, c_Q \in \mathbb{Z}_p$ such that $\mathbf{y} = \sum_{i\in[Q]} c_i \cdot \mathbf{y}_{\mathsf{id}^{\star},i} \mod p$, set
$$t_{\mathsf{id}^{\star},Q+1} = z - \langle \widehat{\mathbf{x}}, \mathbf{y} \rangle \in \mathbb{Z}_p$$
so that
$$\langle [\widehat{\mathbf{x}} \parallel 1], [\mathbf{y} \parallel t_{\mathsf{id}^{\star},Q+1}] \rangle = z,$$
further set $\mathbf{y}_{\mathsf{id}^{\star},Q+1} = \mathbf{y}$ and
$$\widehat{\mathbf{y}} = [\mathbf{y}_{\mathsf{id}^{\star},Q+1} \parallel t_{\mathsf{id}^{\star},Q+1}],$$
and update $(\mathsf{id}^{\star}, \{\mathbf{y}_{\mathsf{id}^{\star},i}, t_{\mathsf{id}^{\star},i}\}_{i\in[Q]}) \in \mathsf{st}$ by $(\mathsf{id}^{\star}, \{\mathbf{y}_{\mathsf{id}^{\star},i}, t_{\mathsf{id}^{\star},i}\}_{i\in[Q+1]})$.

Run
$$\mathsf{IND.sk}_{\mathsf{id}^{\star},\widehat{\mathbf{y}}} \leftarrow \mathsf{IND.KGen}(\mathsf{msk}, \mathsf{id}^{\star}, \widehat{\mathbf{y}})$$
and output $\mathsf{sk}^{\star}_{\mathsf{id}^{\star},\mathbf{y}} = \mathsf{IND.sk}_{\mathsf{id}^{\star},\widehat{\mathbf{y}}}$.

Hereafter, we prove the indistinguishability of a game sequence $\mathsf{SIM}_{\mathsf{Real}} = \mathsf{Game}_0, \ldots, \mathsf{Game}_4 = \mathsf{SIM}_{\mathsf{Ideal}}$.

Game_0**.** This is the real security game $\mathsf{SIM}_{\mathsf{Real}}$ of the AD-SIM security.

Game_1**.** This is the same as Game_0 except that a pseudo-random $f_k(\mathsf{id}) = \mathbf{r}_{\mathsf{id}} \in \mathbb{Z}_p^L$ is replaced with $\mathbf{r}_{\mathsf{id}} \leftarrow_R \mathbb{Z}_p^L$. In particular, $\mathcal{C}$ samples $\mathbf{r}_{\mathsf{id}} \leftarrow_R \mathbb{Z}_p^L$ upon $\mathcal{A}$'s first secret key query associated with id and stores $(\mathsf{id}, \mathbf{r}_{\mathsf{id}}) \in \mathsf{st}$. Since $\mathcal{F}$ is a PRF family, Game_0 and Game_1 are computationally indistinguishable.

Game_2**.** This is the same as Game_1 except that $\mathcal{C}$ does not run KGen but $\mathsf{KGen}^{\star}_0$ to answer both pre- and post-challenge secret key queries. The only change between Game_1 and Game_2 is that $\langle \mathbf{r}_{\mathsf{id}}, \mathbf{y} \rangle \mod p$ in Game_1 is replaced with $t_{\mathsf{id},i}$ or $\sum_{i\in[Q]} c_i \cdot t_{\mathsf{id},i} \mod p$ in Game_2. Suppose that $(\mathsf{id}, \mathbf{r}_{\mathsf{id}}) \in \mathsf{st}$ (resp. $(\mathsf{id}, \{\mathbf{y}_{\mathsf{id},i}, t_{\mathsf{id},i}\}_{i\in[Q_{\mathsf{id}}]}) \in \mathsf{st}$) is stored at the end of Game_1 (resp. Game_2). Both $(\langle \mathbf{r}_{\mathsf{id}}, \mathbf{y}_{\mathsf{id},1} \rangle, \ldots, \langle \mathbf{r}_{\mathsf{id}}, \mathbf{y}_{\mathsf{id},Q_{\mathsf{id}}} \rangle)$ in Game_1 and $(t_{\mathsf{id},1}, \ldots, t_{\mathsf{id},Q_{\mathsf{id}}})$ in Game_2 follow the uniform distribution over $\mathbb{Z}_p^{Q_{\mathsf{id}}}$ since $\{\mathbf{y}_{\mathsf{id}^{\star},1}, \ldots, \mathbf{y}_{\mathsf{id}^{\star},Q_{\mathsf{id}}}\}$ is linearly independent. Thus, answers of secret key queries on $(\mathsf{id}, \mathbf{y}_{\mathsf{id},i})$ for $\mathbf{y}_{\mathsf{id},i} \in \mathsf{st}$ are the same between Game_1 and Game_2. Suppose that there exist $c_1, \ldots, c_{Q_{\mathsf{id}}} \in \mathbb{Z}_p$ such that

$\mathbf{y} = \sum_{i \in [Q_{\mathsf{id}}]} c_i \cdot \mathbf{y}_{\mathsf{id},i} \mod p$ upon a secret key query on $(\mathsf{id}, \mathbf{y})$ for $\mathbf{y}_{\mathsf{id},i} \notin \mathsf{st}$. In Game_1, it holds that

$$\langle \mathbf{r}_{\mathsf{id}}, \mathbf{y} \rangle = \langle \mathbf{r}_{\mathsf{id}}, \sum_{i \in [Q]} c_i \cdot \mathbf{y}_{\mathsf{id},i} \rangle = \sum_{i \in [Q]} c_i \cdot \langle \mathbf{r}_{\mathsf{id}}, \mathbf{y}_{\mathsf{id},i} \rangle \mod p.$$

Since $(\langle \mathbf{r}_{\mathsf{id}}, \mathbf{y}_{\mathsf{id},1} \rangle \mod p, \ldots, \langle \mathbf{r}_{\mathsf{id}}, \mathbf{y}_{\mathsf{id},Q_{\mathsf{id}}} \rangle \mod p)$ in Game_1 and $(t_{\mathsf{id},1}, \ldots, t_{\mathsf{id},Q})$ in Game_2 follow the same distribution, $\langle \mathbf{r}_{\mathsf{id}}, \mathbf{y} \rangle \mod p$ in Game_1 and $\sum_{i \in [Q_{\mathsf{id}}]} c_i \cdot t_{\mathsf{id},i} \mod p$ in Game_2 follow the same distribution. Thus, answers of secret key queries on $(\mathsf{id}, \mathbf{y})$ are the same between Game_1 and Game_2. Therefore, Game_1 and Game_2 follow the same distribution.

Game_3**.** This is the same as Game_2 except that $\mathcal{C}$ does not run $\mathsf{KGen}_0^\star$ but $\mathsf{KGen}_1^\star$ to answer post-challenge secret key queries. To run $\mathsf{KGen}_1^\star$ in Game_3, $\mathcal{C}$ computes $\widehat{\mathbf{x}} \in \mathbb{Z}_p^L$ upon the challenge query by running $\mathsf{Enc}^\star$ although $\mathcal{C}$ answers the query in the same way as Game_2 by running Enc. Suppose that $(\mathsf{id}^\star, \{\mathbf{y}_{\mathsf{id}^\star,i}, t_{\mathsf{id}^\star,i}\}_{i \in [Q_{\mathsf{pre}} + Q_{\mathsf{post}}]}) \in \mathsf{st}$ is stored at the end of the security game. The only changes between Game_2 and Game_3 are answers of post-challenge secret key queries on $(\mathsf{id}^\star, \mathbf{y}_{\mathsf{id}^\star, Q_{\mathsf{pre}}+1}), \ldots, (\mathsf{id}^\star, \mathbf{y}_{\mathsf{id}^\star, Q_{\mathsf{pre}}+Q_{\mathsf{post}}})$. In particular, $t_{\mathsf{id}^\star, Q_{\mathsf{pre}}+1}, \ldots, t_{\mathsf{id}^\star, Q_{\mathsf{pre}}+Q_{\mathsf{post}}} \leftarrow_R \mathbb{Z}_p$ in Game_2 are replaced with

$$t_{\mathsf{id}^\star,i} = z_i^{\mathsf{post}} - \langle \widehat{\mathbf{x}}, \mathbf{y}_{\mathsf{id}^\star,i} \rangle \in \mathbb{Z}_p$$

for $i \in [Q_{\mathsf{pre}} + 1, Q_{\mathsf{pre}} + Q_{\mathsf{post}}]$ in Game_3, where $z_i^{\mathsf{post}} = \mathsf{Eval}(\mathsf{id}^\star, \mathbf{x}^\star, \mathsf{id}^\star, \mathbf{y}_{\mathsf{id}^\star,i}) = \langle \mathbf{x}^\star, \mathbf{y}_{\mathsf{id}^\star,i} \rangle$. Let

$$\begin{aligned}
\mathbf{X}^\perp &= [\mathbf{x}_{Q_{\mathsf{pre}}+1}^\perp \mid \ldots \mid \mathbf{x}_L^\perp] \in \mathbb{Z}_p^{L \times (L - Q_{\mathsf{pre}})}, \\
\mathbf{s} &= [s_{Q_{\mathsf{pre}}+1}, \ldots, s_L] \in \mathbb{Z}_p^{L - Q_{\mathsf{pre}}}, \\
\mathbf{Y}^{\mathsf{post}} &= [\mathbf{y}_{\mathsf{id}^\star, Q_{\mathsf{pre}}+1} \mid \cdots \mid \mathbf{y}_{\mathsf{id}^\star, Q_{\mathsf{pre}}+Q_{\mathsf{post}}}] \in \mathbb{Z}_p^{L \times Q_{\mathsf{post}}}, \\
\mathbf{z}^{\mathsf{post}} &= [z_{Q_{\mathsf{pre}}+1}^{\mathsf{post}}, \ldots, z_{Q_{\mathsf{pre}}+Q_{\mathsf{post}}}^{\mathsf{post}}] \in \mathbb{Z}_p^{Q_{\mathsf{post}}}, \\
\mathbf{t}_{\mathsf{id}^\star}^{\mathsf{post}} &= [t_{\mathsf{id}^\star, Q_{\mathsf{pre}}+1}, \ldots, t_{\mathsf{id}^\star, Q_{\mathsf{pre}}+Q_{\mathsf{post}}}] \in \mathbb{Z}_p^{Q_{\mathsf{post}}}.
\end{aligned}$$

Observe that

$$\begin{aligned}
\mathbf{t}_{\mathsf{id}^\star}^{\mathsf{post}} &= \mathbf{z}^{\mathsf{post}} - {\mathbf{Y}^{\mathsf{post}}}^\top \cdot \widehat{\mathbf{x}} \mod p \\
&= \mathbf{z}^{\mathsf{post}} - {\mathbf{Y}^{\mathsf{post}}}^\top \cdot \left(\bar{\mathbf{x}} + \sum_{i \in [Q_{\mathsf{pre}}+1, L]} s_i \cdot \mathbf{x}_i^\perp \right) \mod p \\
&= \mathbf{z}^{\mathsf{post}} - {\mathbf{Y}^{\mathsf{post}}}^\top \cdot \bar{\mathbf{x}} - {\mathbf{Y}^{\mathsf{post}}}^\top \cdot \mathbf{X}^\perp \cdot \mathbf{s} \mod p.
\end{aligned}$$

Since a rank of ${\mathbf{Y}^{\mathsf{post}}}^\top \cdot \mathbf{X}^\perp$ is Q_{post} and $\mathbf{s}$ follows the uniform distribution, ${\mathbf{Y}^{\mathsf{post}}}^\top \cdot \mathbf{X}^\perp \cdot \mathbf{s} \mod p \in \mathbb{Z}_p^{Q_{\mathsf{post}}}$ follows the uniform distribution. Therefore, Game_2 and Game_3 follow the same distribution.

Game_4**.** This is the same as Game_3 except that $\mathcal{C}$ does not run Enc but $\mathsf{Enc}^\star$ to answer the challenge query. In particular, $\mathsf{Game}_4 = \mathsf{SIM}_{\mathsf{Ideal}}$ holds. Due to the AD-IND security of (IND.Setup, IND.Enc, IND.KGen, IND.Dec), Game_3 and Game_4 are computationally indistinguishable. □

4 Lattice-based IBIPFE Scheme

In this section, we propose a lattice-based IBIPFE scheme computing an inner product modulo a prime p satisfying the tight AD-IND security under the LWE assumption in the QROM. In Section 4.1, we review preliminaries on lattices. Then, we give a construction of the proposed scheme in Section 4.2 and prove the security in Section 4.3.

4.1 Preliminaries on Lattice-based Cryptography

An integer lattice Λ is an additive discrete subgroup of $\mathbb{Z}^m$. For integers n, m, and q such that $q \geq 2$, matrices $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{U} \in \mathbb{Z}_q^{\ell\times n}$, let $\Lambda_q^{\perp}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m \mid \mathbf{A}\mathbf{x} = \mathbf{0} \mod q\}, \Lambda_q^{\mathbf{U}}(\mathbf{A}) = \{\mathbf{X} \in \mathbb{Z}^m \mid \mathbf{A}\mathbf{X} = \mathbf{U} \mod q\}$.

Discrete Gaussian and Sampling Algorithms. Let $\mathcal{D}_{\Lambda,\sigma}$ denote a discrete Gaussian distribution over Λ with a Gaussian parameter σ. In the following, we review some basic properties of discrete Gaussian distributions.

Lemma 1 ([20]). *Let n, m, q be positive integers such that $m \geq 2n \log q$. Let σ be any positive real number such that $\sigma \geq \sqrt{n + \log m}$. For $\mathbf{A} \leftarrow_R \mathbb{Z}_q^{n\times m}$ and $\mathbf{e} \leftarrow D_{\mathbb{Z}^m,\sigma}$, a distribution of $\mathbf{u} = \mathbf{A}\mathbf{e} \mod q$ is statistically close to uniform over $\mathbb{Z}_q^n$. Furthermore, for a fixed $\mathbf{u} \in \mathbb{Z}_q^n$, a conditional distribution of $\mathbf{e} \leftarrow D_{\mathbb{Z}^m,\sigma}$, given $\mathbf{A}\mathbf{e} = \mathbf{u} \mod q$ for a uniformly random $\mathbf{A}$ in $\mathbb{Z}_q^{n\times m}$ is statistically close to $D_{\Lambda_q^{\mathbf{u}}(\mathbf{A}),\sigma}$.*

Lemma 2 ([20,30]). *Let $\sigma > 16\sqrt{\log 2m/\pi}$ and $\mathbf{u}$ be any vector in $\mathbb{Z}_q^n$. Then, for all but q^{-n} fraction of $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, we have $\Pr_{\mathbf{x}\leftarrow D_{\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A}),\sigma}}[\|\mathbf{x}\| \geq \sigma\sqrt{m}] \leq 2^{-(m-1)}$.*

Lemma 3 ([20,32,33]). *Let $\sigma > 16\sqrt{\log 2m/\pi}$ and $\mathbf{u}$ be any vector in $\mathbb{Z}_q^n$. Then, for all but q^{-n} fraction of $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, we have $\mathbf{H}_\infty(D_{\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A}),\sigma}) \geq m - 1$.*

Lemma 4 (Noise Re-randomization, [25], Lemma 1). *Let q, ℓ, m be positive integers and r be a positive real number satisfying $r > \max\{\omega(\sqrt{\log m}), \omega(\sqrt{\log \ell})\}$. Let $\mathbf{b} \in \mathbb{Z}_q^m$ be arbitrary and $\mathbf{z}$ chosen from $D_{\mathbb{Z}^m,r}$. Then, there exists a PPT algorithm* ReRand *such that for any $\mathbf{V} \in \mathbb{Z}^{m\times\ell}$ and positive real number $\sigma > \|\mathbf{V}\|_2$,* $\mathsf{ReRand}(\mathbf{V}, \mathbf{b}+\mathbf{z}, r, \sigma)$ *outputs $\mathbf{b}'^{\top} = \mathbf{b}^{\top}\mathbf{V} + \mathbf{z}'^{\top} \in \mathbb{Z}_q^{\ell}$ where a distribution of $\mathbf{z}'$ is statistically close to $D_{\mathbb{Z}^{\ell},2r\sigma}$.*

Lemma 5 ([5,8,14,29]). *Let $n, m, q > 0$ be positive integers with $m \geq 3n\lceil \log q \rceil$ and q a prime. Then, we have the following polynomial time algorithms:*

$\mathsf{TrapGen}(1^n, 1^m, q) \to (\mathbf{A}, \mathbf{T_A})$: *a PPT algorithm that outputs a full rank matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a basis $\mathbf{T_A} \in \mathbb{Z}^{m\times m}$ for $\Lambda_q^{\perp}(\mathbf{A})$ such that a distribution of $\mathbf{A}$ is statistically close to uniform and $\|\mathbf{T_A}\|_{\text{GS}} = O(\sqrt{n \log q})$.*

$\mathsf{SamplePre}(\mathbf{A}, \mathbf{T}_{\mathbf{A}}, \mathbf{u}, \sigma) \to \mathbf{e}$: *a PPT algorithm that is given a full rank matrix* $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$*, a basis* $\mathbf{T}_{\mathbf{A}} \in \mathbb{Z}^{m\times m}$ *of a lattice* $\Lambda_q^{\perp}(\mathbf{A})$*, a vector* $\mathbf{u} \in \mathbb{Z}_q^n$*, and* $\sigma \geq \|\mathbf{T}_{\mathbf{A}}\|_{\mathrm{GS}} \cdot \omega(\sqrt{\log m})$*, and outputs a vector* $\mathbf{e} \in \mathbb{Z}^m$ *sampled from a distribution statistically close to* $D_{\Lambda_q^{\mathbf{U}}(\mathbf{A}),\sigma}$.

$\mathsf{Sample}\mathbb{Z}(\sigma)$: *a PPT algorithm that is given* $\sigma > \omega(\sqrt{\log m})$ *and outputs a vector* $\mathbf{e} \in \mathbb{Z}^m$ *sampled from a distribution statistically close to* $D_{\mathbb{Z}^m,\sigma}$.

Quantum Computation. Let $|0\rangle := [1, 0]^{\top}$ and $|1\rangle := [0, 1]^{\top}$ denote the state of 1 qubit. Let $|\psi\rangle = \sum_{x\in\{0,1\}^n} \alpha_x|x\rangle \in \mathbb{C}^{2^n}$ denote the state of n qubits, where $\alpha_x \in \mathbb{C}$ satisfying $\sum_{x\in\{0,1\}^n} |\alpha_x|^2 = 1$ and $|x\rangle = |x_1x_2\cdots x_n\rangle = |x_1\rangle \otimes |x_2\rangle \otimes \cdots \otimes |x_n\rangle$ for $x_1, x_2, \ldots, x_n \in \{0, 1\}$ is an orthonormal basis on $\mathbb{C}^{2^n}$ called the computational basis. If we measure the state $|\psi\rangle$ in the computational basis, the classical bit $x \in \{0, 1\}^n$ is observed with probability $|\alpha_x|^2$ and the state becomes $|x\rangle$. An arbitrary evolution of quantum state from $|\psi\rangle$ to $|\psi'\rangle$ is described by a unitary matrix U, where $|\psi'\rangle = U|\psi\rangle$. In short, a quantum algorithm is described by quantum evolutions that consist of evolutions with unitary matrices and measurements. The running time $\mathsf{Time}(\mathcal{A})$ of a quantum algorithm $\mathcal{A}$ is defined to be the number of universal gates and measurements required for running $\mathcal{A}$. If $\mathcal{A}$ is a quantum oracle algorithm, we assume that $\mathcal{A}$ runs in a unit time. Any efficient classical computation can be achieved by a quantum computation efficiently. In particular, for any function f that is classically computable, there exists a unitary matrix U_f such that $U_f|x, y\rangle = |x, f(x) \oplus y\rangle$, and the number of universal gates to express U_f is linear in the size of a classical circuit that computes f.

Quantum Random Oracle Model. The notion of the QROM was introduced by Boneh et al. [11] as a quantum extension of the ROM. As in the case of the ROM, the QROM is an idealized model in the sense that a hash function is idealized to be an oracle that simulates a random function. However, the hash function in the QROM is a quantumly accessible oracle, unlike the case of the ROM. In security proofs in the QROM, a random function $\mathsf{H} : \mathcal{X} \to \mathcal{Y}$ is uniformly chosen at the beginning, and an adversary can make queries on a quantum state $\sum_{x,y} \alpha_{x,y}|x\rangle|y\rangle$ to the oracle and receive $\sum_{x,y} \alpha_{x,y}|x\rangle|\mathsf{H}(x) \oplus y\rangle$.

Lemma 6. *([35, Lem. 2.2]) Let* ℓ *be an integer. Let* $\mathsf{H}\colon \{0, 1\}^{\ell} \times \mathcal{X} \to \mathcal{Y}$ *and* $\mathsf{H}'\colon \mathcal{X} \to \mathcal{Y}$ *be two independent random functions. If an unbounded time quantum adversary* $\mathcal{A}$ *makes queries to* H *at most* Q_{H} *times, then we have*

$$\left|\Pr\left[\mathcal{A}^{|\mathsf{H}\rangle,|\mathsf{H}(K,\cdot)\rangle}(1^{\lambda}) = 1 \mid K \leftarrow \{0,1\}^{\ell}\right] - \Pr\left[\mathcal{A}^{|\mathsf{H}\rangle,|\mathsf{H}'\rangle}(1^{\lambda}) = 1\right]\right| \leq Q_{\mathsf{H}} \cdot 2^{\frac{-\ell+1}{2}}.$$

LWE Assumption relative to the QROM. We review the LWE assumption against adversaries that can access a quantum random oracle defined in [26]. If we assume the existence of a quantum-accessible PRF, the LWE assumption relative to the QROM in Definition 4 is tightly reduced from the LWE assumption [34].

Definition 4 (Learning with Errors relative to the QROM). *For integers* $n = n(\lambda), m = m(n)$*, a prime* $q = q(n) > 2$*, an error distribution* $\chi = \chi(n)$ *over*

$\mathbb{Z}$, some positive integers a, b, and a quantum polynomial time algorithm $\mathcal{A}$, the advantage for the learning with errors problem $\mathsf{LWE}_{n,m,q,\chi}$ *of $\mathcal{A}$* relative to a quantum random oracle *is defined as follows:*

$$\mathsf{Adv}^{\mathsf{LWE}_{n,m,q,\chi}}_{\mathcal{A},\mathbf{QRO}_{a,b}}(\lambda) := \left| \Pr\left[\mathcal{A}^{|\mathsf{H}\rangle}\left(\mathbf{A}, \mathbf{A}^\top \mathbf{s} + \mathbf{z}\right) = 1\right] - \Pr\left[\mathcal{A}^{|\mathsf{H}\rangle}\left(\mathbf{A}, \mathbf{w} + \mathbf{z}\right) = 1\right] \right|$$

where $\mathbf{A} \leftarrow_R \mathbb{Z}_q^{n\times m}$, $\mathbf{s} \leftarrow_R \mathbb{Z}_q^n$, $\mathbf{w} \leftarrow_R \mathbb{Z}_q^m$, $\mathbf{z} \leftarrow \chi^m$, $\mathsf{H} \leftarrow_R \mathsf{Func}(\{0,1\}^a, \{0,1\}^b)$. We say that the LWE *assumption relative to an (a, b)-quantum random oracle holds if $\mathsf{Adv}^{\mathsf{LWE}_{n,m,q,\chi}}_{\mathcal{A},\mathbf{QRO}_{a,b}}(\lambda)$ is negligible for all quantum polynomial-time $\mathcal{A}$.*

4.2 Construction

We modify Abdala et al.'s IBIPFE scheme that computes inner products over the integers [2] by following Agrawal et al. [7] so that we can compute inner products modulo a prime p.

$\mathsf{Setup}(1^\lambda, 1^L) \to (\mathsf{mpk}, \mathsf{msk})$**:** Set integers $n, m, p, q = p^k$ for a prime p and positive real α, α', σ, and choose a cryptographic hash function $\mathsf{H} : \mathcal{ID} \to \mathbb{Z}_q^{n\times L}$. Run $(\mathbf{A}, \mathbf{T_A}) \leftarrow \mathsf{TrapGen}(n, m, q)$, where $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, and output

$$\mathsf{mpk} = (\mathbf{A}, \mathsf{H}), \qquad \mathsf{msk} = \mathbf{T_A}.$$

$\mathsf{Enc}(\mathsf{mpk}, \mathsf{id}, \mathbf{x}) \to \mathsf{ct}_{\mathsf{id},\mathbf{x}}$**:** Parse $\mathsf{mpk} = (\mathbf{A}, \mathsf{H})$. Compute $\mathsf{H}(\mathsf{id}) = \mathbf{U}_{\mathsf{id}} \in \mathbb{Z}_q^{n\times L}$, sample $\mathbf{s} \leftarrow_R \mathbb{Z}_q^n$, $\mathbf{e}_1 \leftarrow \mathcal{D}_{\mathbb{Z}^m,\alpha' q}$, $\mathbf{e}_2 \leftarrow \mathcal{D}_{\mathbb{Z}^L,\alpha' q}$, and output

$$\mathsf{ct}_{\mathsf{id},\mathbf{x}} = \left(\mathbf{c}_1 = \mathbf{A}^\top \mathbf{s} + \mathbf{e}_1, \qquad \mathbf{c}_2 = \mathbf{U}_{\mathsf{id}}^\top \mathbf{s} + \mathbf{e}_2 + p^{k-1} \cdot \mathbf{x}\right) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^L.$$

$\mathsf{KGen}(\mathsf{msk}, \mathsf{id}, \mathbf{y}, \mathsf{st}) \to \mathsf{sk}_{\mathsf{id},\mathbf{y}}$**:** Parse $\mathsf{mpk} = (\mathbf{A}, \mathsf{H})$ and $\mathsf{msk} = \mathbf{T_A}$. If this is the first key generation associated with $\mathsf{id} \in \mathcal{ID}$, compute $\mathsf{H}(\mathsf{id}) = \mathbf{U}_{\mathsf{id}} \in \mathbb{Z}_q^{n\times L}$, run

$$\mathbf{Z}_{\mathsf{id}} \leftarrow \mathsf{SamplePre}(\mathbf{A}, \mathbf{T_A}, \sigma, \mathbf{U}_{\mathsf{id}}),$$

where $\mathbf{A}\mathbf{Z}_{\mathsf{id}} = \mathbf{U}_{\mathsf{id}} \mod q$, output

$$\mathsf{sk}_{\mathsf{id},\mathbf{y}} = (\overline{\mathbf{y}} = \mathbf{y}, \qquad \mathbf{k}_{\mathsf{id},\mathbf{y}} = \mathbf{Z}_{\mathsf{id}} \cdot \overline{\mathbf{y}}) \in \mathbb{Z}^L \times \mathbb{Z}^m,$$

set $\overline{\mathbf{y}}_{\mathsf{id},1} = \mathbf{y}$ and $\mathbf{k}_{\mathsf{id},1} = \mathbf{k}_{\mathsf{id},\mathbf{y}}$, and store $\left(\mathsf{id}, \mathbf{Z}_{\mathsf{id}}, \left(\overline{\mathbf{y}}_{\mathsf{id},1}, \mathbf{k}_{\mathsf{id},1}\right)\right) \in \mathsf{st}$. Otherwise, retrieve $\left(\mathsf{id}, \mathbf{Z}_{\mathsf{id}}, \left(\overline{\mathbf{y}}_{\mathsf{id},i}, \mathbf{k}_{\mathsf{id},i}\right)_{i\in[Q_{\mathsf{id}}]}\right) \in \mathsf{st}$, where $\overline{\mathbf{y}}_{\mathsf{id},1}, \ldots, \overline{\mathbf{y}}_{\mathsf{id},Q_{\mathsf{id}}}$ denote all linearly independent vectors which secret keys associated with id have been created so far.

- If there exist $c_1, \ldots, c_{Q_{\mathsf{id}}} \in \mathbb{Z}_p$ such that $\mathbf{y} = \sum_{i\in[Q_{\mathsf{id}}]} c_i \cdot \overline{\mathbf{y}}_{\mathsf{id},i} \mod p$, output

$$\mathsf{sk}_{\mathsf{id},\mathbf{y}} = \left(\overline{\mathbf{y}} = \sum_{i\in[Q_{\mathsf{id}}]} c_i \cdot \overline{\mathbf{y}}_{\mathsf{id},i}, \qquad \mathbf{k}_{\mathsf{id},\mathbf{y}} = \sum_{i\in[Q_{\mathsf{id}}]} c_i \cdot \mathbf{k}_{\mathsf{id},i}\right) \in \mathbb{Z}^L \times \mathbb{Z}^m.$$

– If there do not exist $c_1, \ldots, c_{Q_{\mathsf{id}}} \in \mathbb{Z}_p$ such that $\mathbf{y} = \sum_{i \in [Q_{\mathsf{id}}]} c_i \cdot \overline{\mathbf{y}}_{\mathsf{id},i} \mod p$, output

$$\mathsf{sk}_{\mathsf{id},\mathbf{y}} = (\overline{\mathbf{y}} = \mathbf{y}, \qquad \mathbf{k}_{\mathsf{id},\mathbf{y}} = \mathbf{Z}_{\mathsf{id}} \cdot \overline{\mathbf{y}}) \in \mathbb{Z}^L \times \mathbb{Z}^m,$$

set $\overline{\mathbf{y}}_{\mathsf{id},Q_{\mathsf{id}}+1} = \mathbf{y}$ and $\mathbf{k}_{\mathsf{id},Q_{\mathsf{id}}+1} = \mathbf{k}_{\mathsf{id},\mathbf{y}}$, and update $\left(\mathsf{id}, \left(\overline{\mathbf{y}}_{\mathsf{id},i}, \mathbf{k}_{\mathsf{id},i}\right)_{i \in [Q_{\mathsf{id}}]}\right) \in \mathsf{st}$ by $\left(\mathsf{id}, \left(\overline{\mathbf{y}}_{\mathsf{id},i}, \mathbf{k}_{\mathsf{id},i}\right)_{i \in [Q_{\mathsf{id}}+1]}\right) \in \mathsf{st}$.

$\mathsf{Dec}(\mathsf{mpk}, \mathsf{ct}_{\mathsf{id},\mathbf{x}}, \mathsf{sk}_{\mathsf{id},\mathbf{y}}) \to \langle \mathbf{x}, \mathbf{y} \rangle$**:** Parse $\mathsf{ct}_{\mathsf{id},\mathbf{x}} = (\mathbf{c}_1, \mathbf{c}_2)$ and $\mathsf{sk}_{\mathsf{id},\mathbf{y}} = (\overline{\mathbf{y}}, \mathbf{k}_{\mathsf{id},\mathbf{y}})$. Compute

$$\mu = \langle \mathbf{c}_2, \overline{\mathbf{y}} \rangle - \langle \mathbf{c}_1, \mathbf{k}_{\mathsf{id},\mathbf{y}} \rangle \mod q$$

and output $\arg\min_{z \in \mathbb{Z}_p} |\mu - p^{k-1} \cdot z|$.

Correctness. Since $\mathbf{A}\mathbf{Z}_{\mathsf{id}} = \mathbf{U}_{\mathsf{id}} \mod q$ holds, it holds that

$$\mathbf{A}\mathbf{k}_{\mathsf{id},\mathbf{y}} = \mathbf{A}\mathbf{Z}_{\mathsf{id}} \cdot \overline{\mathbf{y}} = \mathbf{U}_{\mathsf{id}} \cdot \overline{\mathbf{y}} \mod q$$

if $\overline{\mathbf{y}} = \mathbf{y} \in \mathbb{Z}^L$ and $\mathbf{k}_{\mathsf{id},\mathbf{y}} = \mathbf{Z}_{\mathsf{id}} \cdot \mathbf{y} \in \mathbb{Z}^m$. Similarly, it holds that

$$\mathbf{A}\mathbf{k}_{\mathsf{id},\mathbf{y}} = \mathbf{A}\left(\sum_{i \in [Q_{\mathsf{id}}]} c_i \cdot \mathbf{Z}_{\mathsf{id}} \cdot \overline{\mathbf{y}}_{\mathsf{id},i}\right) = \mathbf{U}_{\mathsf{id}} \cdot \overline{\mathbf{y}} \mod q$$

if $\overline{\mathbf{y}} = \sum_{i \in [Q_{\mathsf{id}}]} c_i \cdot \overline{\mathbf{y}}_{\mathsf{id},i} \in \mathbb{Z}^L$ and $\mathbf{k}_{\mathsf{id},\mathbf{y}} = \sum_{i \in [Q_{\mathsf{id}}]} c_i \cdot \mathbf{k}_{\mathsf{id},i} = \mathbf{Z}_{\mathsf{id}} \cdot \left(\sum_{i \in [Q_{\mathsf{id}}]} c_i \cdot \overline{\mathbf{y}}_{\mathsf{id},i}\right) \in \mathbb{Z}^m$. Since it holds that

$$\begin{aligned}\langle \mathbf{c}_2, \overline{\mathbf{y}} \rangle &= \left(\mathbf{U}_{\mathsf{id}}^\top \mathbf{s} + \mathbf{e}_2 + p^{k-1} \cdot \mathbf{x}\right)^\top \overline{\mathbf{y}} = \mathbf{s}^\top \mathbf{U}_{\mathsf{id}} \cdot \overline{\mathbf{y}} + \mathbf{e}_2^\top \overline{\mathbf{y}} + p^{k-1} \cdot \langle \mathbf{x}, \mathbf{y} \rangle, \\ \langle \mathbf{c}_1, \mathbf{k}_{\mathsf{id},\mathbf{y}} \rangle &= \left(\mathbf{A}^\top \mathbf{s} + \mathbf{e}_1\right)^\top \mathbf{k}_{\mathsf{id},\mathbf{y}} = \mathbf{s}^\top \mathbf{U}_{\mathsf{id}} \cdot \overline{\mathbf{y}} + \mathbf{e}_1^\top \mathbf{k}_{\mathsf{id},\mathbf{y}},\end{aligned}$$

we have

$$\mu = p^{k-1} \cdot \langle \mathbf{x}, \mathbf{y} \rangle + \underbrace{\mathbf{e}_2^\top \overline{\mathbf{y}} - \mathbf{e}_1^\top \mathbf{k}_{\mathsf{id},\mathbf{y}}}_{\text{error terms}} \mod q.$$

Since $\mathbf{e}_1 \leftarrow \mathcal{D}_{\mathbb{Z}^m,\alpha' q}$ and $\mathbf{e}_2 \leftarrow \mathcal{D}_{\mathbb{Z}^L,\alpha' q}$, it holds that $\|\mathbf{e}_1\| \leq \alpha' q \sqrt{m}$ and $\|\mathbf{e}_2\| \leq \alpha' q \sqrt{L}$ with overwhelming probability from Lemma 2. Since $c_1, \ldots, c_{Q_{\mathsf{id}}} \in \mathbb{Z}_p$ and $\overline{\mathbf{y}}_{\mathsf{id},1}, \ldots, \overline{\mathbf{y}}_{\mathsf{id},Q_{\mathsf{id}}} \in \mathbb{Z}_p^L$, it holds that $\|\overline{\mathbf{y}}\| \leq p^2 \sqrt{L}$. Since $\mathbf{Z}_{\mathsf{id}} = [\mathbf{z}_{\mathsf{id},1} \mid \cdots \mid \mathbf{z}_{\mathsf{id},L}] \leftarrow \mathsf{SamplePre}(\mathbf{A}, \mathbf{T}_{\mathbf{A}}, \sigma, \mathbf{U}_{\mathsf{id}})$, the distribution is statistically close to $D_{\Lambda_q^{\mathbf{U}}(\mathbf{A}),\sigma}$. Thus, it holds that $\|\mathbf{z}_{\mathsf{id},\ell}\| \leq \sigma \sqrt{m}$ with overwhelming probability from Lemma 2. Moreover, $\|\mathbf{k}_{\mathsf{id},\mathbf{y}}\| = \|\mathbf{Z}_{\mathsf{id}} \cdot \overline{\mathbf{y}}\| \leq p^2 \sigma \sqrt{Lm}$. Therefore, the absolute value of the error term is upper bounded by $\alpha' q \sqrt{L} \cdot p^2 \sqrt{L} + \alpha' q \sqrt{m} \cdot p^2 \sigma \sqrt{Lm} = \alpha' p^2 q \sqrt{L}(\sqrt{L} + m)$. If the error term is upper bounded by $p^{k-1}/2$, the correctness holds with overwhelming probability.

4.3 Tight AD-IND Security in the QROM

To conclude this section, we prove the following theorem.

Theorem 2. *If the* LWE *assumption holds and* $\mathcal{F}$ *is a* PRF *family, the proposed* IBIPFE *scheme* Π_{LWE} *in Section 4.2 satisfies the tight* AD-IND *security in the quantum random oracle model. In particular, for any quantum* $\mathcal{A}$ *that breaks the* AD-IND *security of* Π_{LWE}*, there exist quantum* $\mathcal{B}_1$ *and* $\mathcal{B}_2$ *such that*

$$\mathsf{Adv}^{\mathsf{AD\text{-}SIM}}_{\Pi_{\mathsf{LWE}},\mathcal{A}}(\lambda) \leq \mathsf{Adv}_{\mathcal{F},\mathcal{B}_1}(\lambda) + \mathsf{Adv}^{\mathsf{LWE}}_{\mathcal{B}_2}(\lambda).$$

and

$$\max\{\mathsf{Time}(\mathcal{B}_1), \mathsf{Time}(\mathcal{B}_2)\} = \mathsf{Time}(\mathcal{A}) + \mathsf{poly}(\lambda, L).$$

A proof of Theorem 2 is a combination of Katsumata et al.'s proof [26] for Gentry et al.'s IBE scheme [20], Wang et al.'s proof for their IPFE scheme [36], and Agrawal et al.'s information-theoretic argument for their IPFE scheme [7]. We use a game sequence $\mathsf{Game}_0, \dots, \mathsf{Game}_6$. In Game_1 and Game_2, we follow Katsumata et al.'s proof [26] and change answers of quantum random oracle queries and secret key queries, respectively. In Game_3, we change the way to compute mpk as the standard argument of lattice-based cryptography. In Game_4–Game_6, we follow Katsumata et al.'s proof [26] and Wang et al.'s proof [36], and change an answer of a challenge query. Finally, we apply Agrawal et al.'s information-theoretic argument [7] and conclude the proof.

Proof of Theorem 2. We prove the computational indistinguishability of a game sequence $\mathsf{Game}_0, \dots, \mathsf{Game}_6$.

Game_0**.** This is the security game of the AD-IND security.

Game_1**.** This is the same as Game_0 except for answers to quantum random oracle queries. In Game_0, $\mathcal{C}$ samples $\mathsf{H} \leftarrow_R \mathsf{Func}(\mathcal{ID}, \mathbb{Z}_q^{n\times L})$ at beginning of the game and sets $\mathsf{H}(\mathsf{id}) = \mathbf{U}_{\mathsf{id}} \in \mathbb{Z}_q^{n\times L}$ to simulate a random oracle. In Game_2, $\mathcal{C}$ samples $\widehat{\mathsf{H}} \leftarrow_R \mathsf{Func}(\mathcal{ID}, \mathcal{R})$ at the beginning of the game and defines $\mathsf{H}(\mathsf{id}) = \mathbf{A}\widehat{\mathbf{Z}}_{\mathsf{id}} \in \mathbb{Z}_q^{n\times L}$, where $\widehat{\mathbf{Z}}_{\mathsf{id}} = \mathsf{Sample}\mathbb{Z}(\sigma; \widehat{\mathsf{H}}(\mathsf{id}))$. In Game_0 and Game_1, distributions of $\mathsf{H}(\mathsf{id})$ are statistically close from Lemma 1. Thus, the difference of $\mathcal{A}$'s advantage between Game_0 and Game_1 are negligible from Lemma 6.

Game_2**.** This is the same as Game_1 except for answers to secret key queries. In particular, $\mathcal{C}$ does not run $\mathbf{Z}_{\mathsf{id}} \leftarrow \mathsf{SamplePre}(\mathbf{A}, \mathbf{T}_{\mathbf{A}}, \sigma, \mathbf{U}_{\mathsf{id}})$ to answer $\mathcal{A}$'s first secret key query associated with id but sets $\mathbf{Z}_{\mathsf{id}} = \widehat{\mathbf{Z}}_{\mathsf{id}} = \mathsf{Sample}\mathbb{Z}(\sigma; \widehat{\mathsf{H}}(\mathsf{id}))$. Therefore, $\mathcal{C}$ does not use $\mathsf{msk} = \mathbf{T}_{\mathbf{A}}$ anymore. Game_1 and Game_2 are statistically indistinguishable from Lemma 1.

Game_3**.** This is the same as Game_2 except the way $\mathcal{C}$ computes $\mathbf{A} \in \mathsf{mpk}$. In particular, $\mathcal{C}$ does not run $(\mathbf{A}, \mathbf{T}_{\mathbf{A}}) \leftarrow \mathsf{TrapGen}(n, m, q)$ but samples $\mathbf{A} \leftarrow_R \mathbb{Z}_q^{n\times m}$. Game_2 and Game_3 are statistically indistinguishable from Lemma 5.

Game_4**.** This is the same as Game_4 except the way $\mathcal{C}$ computes the challenge ciphertext $\mathsf{ct}^\star$. Upon $\mathcal{A}$'s challenge query on $((\mathsf{id}_0^\star, \mathbf{x}_0^\star), (\mathsf{id}_1^\star, \mathbf{x}_1^\star))$ in Game_4, $\mathcal{C}$

samples $\mathbf{s} \leftarrow_R \mathbb{Z}_q^n$, $\mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z}^m,\alpha q}$, and computes

$$\mathbf{v} = \mathbf{A}^\top \mathbf{s} + \mathbf{e} \in \mathbb{Z}_q^m.$$

Then, $\mathcal{C}$ runs

$$[\widehat{\mathbf{c}}_1 \parallel \widehat{\mathbf{c}}_2] \leftarrow \mathsf{ReRand}\left([\mathbf{I}_m \mid \widehat{\mathbf{Z}}_{\mathsf{id}_\beta^\star}], \mathbf{v}, \alpha q, \frac{\alpha'}{2\alpha} \right)$$

in Lemma 4, where $\widehat{\mathbf{c}}_1 \in \mathbb{Z}_q^m$ and $\widehat{\mathbf{c}}_2 \in \mathbb{Z}_q^L$, samples $\beta \leftarrow_R \{0,1\}$, and computes

$$\mathsf{ct}^\star = \big(\mathbf{c}_1 = \widehat{\mathbf{c}}_1, \qquad \mathbf{c}_2 = \widehat{\mathbf{c}}_2 + p^{k-1} \cdot \mathbf{x}_\beta^\star\big) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^L.$$

Since it holds that

$$[\mathbf{I}_m \mid \widehat{\mathbf{Z}}_{\mathsf{id}_\beta^\star}]^\top \mathbf{A}^\top \mathbf{s} = [\mathbf{A} \mid \mathbf{A}\widehat{\mathbf{Z}}_{\mathsf{id}_\beta^\star}]^\top \mathbf{s} = [\mathbf{A} \mid \mathsf{H}(\mathsf{id}_\beta^\star)]^\top \mathbf{s},$$

Game_3 and Game_4 are statistically indistinguishable from Lemma 4.

Game_5. This is the same as Game_5 except for the way $\mathcal{C}$ computes $\mathbf{v} \in \mathbb{Z}_q^m$ to answer the challenge query. In particular, $\mathcal{C}$ samples $\mathbf{b} \leftarrow_R \mathbb{Z}_q^m$, $\mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z}^m,\alpha q}$, and computes

$$\mathbf{v} = \mathbf{b} + \mathbf{e} \in \mathbb{Z}_q^m.$$

Game_4 and Game_5 are computationally indistinguishable by assuming the hardness of $\mathsf{LWE}_{n,m,q,\alpha'}$ relative to a quantum random oracle $\widehat{\mathsf{H}} \in \mathsf{Func}(\mathcal{ID}, \mathcal{R})$.

Game_6. This is the same as Game_5 except for the way $\mathcal{C}$ computes $[\widehat{\mathbf{c}}_1 \mid \widehat{\mathbf{c}}_2]$ to answer the challenge query. In particular, $\mathcal{C}$ samples $\mathbf{b} \leftarrow_R \mathbb{Z}_q^m$, $\mathbf{e}_1 \leftarrow \mathcal{D}_{\mathbb{Z}^m,\alpha' q}$, $\mathbf{e}_2 \leftarrow \mathcal{D}_{\mathbb{Z}^L,\alpha' q}$, and computes

$$[\widehat{\mathbf{c}}_1 \parallel \widehat{\mathbf{c}}_2] = [\mathbf{I}_m \mid \widehat{\mathbf{Z}}_{\mathsf{id}_\beta^\star}]^\top \mathbf{b} + [\mathbf{e}_1 \parallel \mathbf{e}_2] \in \mathbb{Z}_q^{m+L}.$$

Game_5 and Game_6 are statistically indistinguishable from Lemma 4.

Finally, we can conclude that the challenge ciphertext $\mathsf{ct}^\star$ in Game_6 is (almost) independent of β by following Agrawal et al.'s information-theoretic agrument [7]. □

5 Pairing-based IBIPFE Scheme

In this section, we propose a pairing-based IBIPFE scheme computing an inner product modulo a prime p satisfying the tight AD-SIM security under the DBDH assumption in the ROM. In Section 5.1, we review bilinear groups and the DBDH assumption. Then, we give a construction of the proposed scheme in Section 5.2 and prove the security in Section 5.3.

5.1 Bilinear Groups

Let $\mathcal{G}$ denote a symmetric bilinear groups generator, i.e., $\mathcal{G}(1^\lambda) \to (p, \mathbb{G}, \mathbb{G}_T, g, e)$, where the input λ is the security parameter, $p = \Theta(2^\lambda)$ is a prime number, $\mathbb{G}$ and $\mathbb{G}_T$ are cyclic groups of order p, g is a generator of $\mathbb{G}$, and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is an efficiently computable non-degenerate bilinear map. Thus, $e(g, g)$ is a generator of $\mathbb{G}_T$ and it holds that $e(g^a, g^b) = e(g, g)^{ab}$ for any $a, b \in \mathbb{Z}_p$. For simplicity, we use $\mathcal{G}(1^\lambda)$ to denote the output of the generator. We will use the following decisional bilinear Diffie-Hellman (DBDH) assumption.

Definition 5 (DBDH Assumption). *Let $(p, \mathbb{G}, \mathbb{G}_T, g, e) \leftarrow \mathcal{G}(1^\lambda)$ and $a, b, c, d \leftarrow_R \mathbb{Z}_p$. We say that the* DBDH *assumption holds if the advantage*

$$\begin{aligned} &\mathsf{Adv}_{\mathcal{A}}^{\mathsf{DBDH}}(\lambda) \\ = &\left|\Pr\left[\mathcal{A}(\mathcal{G}(1^\lambda), g^a, g^b, g^c, e(g,g)^{abc}) \to 1\right] - \Pr\left[\mathcal{A}(\mathcal{G}(1^\lambda), g^a, g^b, g^c, e(g,g)^{d}) \to 1\right]\right| \end{aligned}$$

is negligible in λ for any PPT adversary $\mathcal{A}$.

5.2 Construction

We propose an IBIPFE scheme Π_{DBDH} over a symmetric bilinear group as follows.

$\mathsf{Setup}(1^\lambda, 1^L) \to (\mathsf{mpk}, \mathsf{msk})$**:** Run $(p, \mathbb{G}, \mathbb{G}_T, g, e) \leftarrow \mathcal{G}(1^\lambda)$, sample $a, b \leftarrow_R \mathbb{Z}_p$, set $h_1 = g^a, h_2 = g^b$, choose a cryptographic hash function $\mathsf{H} : \mathcal{ID} \to \mathbb{G}^L$ and an index $k \leftarrow_R \mathcal{K}$ for a function family $\mathcal{F} = \{f_k\}_k$ such that $f_k : \mathcal{ID} \to \mathbb{Z}_p^L$, and output

$$\mathsf{mpk} = (p, \mathbb{G}, \mathbb{G}_T, g, e, h_1, h_2, \mathsf{H}), \qquad \mathsf{msk} = (a, k).$$

$\mathsf{Enc}(\mathsf{mpk}, \mathsf{id}, \mathbf{x}) \to \mathsf{ct}_{\mathsf{id},\mathbf{x}}$**:** Parse $\mathbf{x} = (x_1, \ldots, x_L) \in \mathbb{Z}_p^L$. Compute $\mathsf{H}(\mathsf{id}) = (h_{\mathsf{id},1}, \ldots, h_{\mathsf{id},L}) \in \mathbb{G}^L$, sample $s \leftarrow_R \mathbb{Z}_p$, and output

$$\mathsf{ct}_{\mathsf{id},\mathbf{x}} = \big(C = g^s, D = e(h_1, h_2)^s, (E_\ell = e(g,g)^{x_\ell} \cdot e(h_1, h_{\mathsf{id},\ell})^s)_{\ell \in [L]}\big).$$

$\mathsf{KGen}(\mathsf{msk}, \mathsf{id}, \mathbf{y}) \to \mathsf{sk}_{\mathsf{id},\mathbf{y}}$**:** Parse $\mathbf{y} = [y_1, \ldots, y_L] \in \mathbb{Z}_p^L$. Compute $\mathsf{H}(\mathsf{id}) = (h_{\mathsf{id},1}, \ldots, h_{\mathsf{id},L}) \in \mathbb{G}^L$ and $f_k(\mathsf{id}) = \mathbf{r}_{\mathsf{id}} = [r_{\mathsf{id},1}, \ldots, r_{\mathsf{id},L}] \in \mathbb{Z}_p^L$, and output

$$\mathsf{sk}_{\mathsf{id},\mathbf{y}} = \left(\mathbf{y}, \sigma = \prod_{\ell \in [L]} (h_{\mathsf{id},\ell} \cdot h_2^{-r_{\mathsf{id},\ell}})^{a y_\ell}, \kappa = \langle \mathbf{r}_{\mathsf{id}}, \mathbf{y} \rangle \mod p\right).$$

$\mathsf{Dec}(\mathsf{mpk}, \mathsf{ct}_{\mathsf{id},\mathbf{x}}, \mathsf{sk}_{\mathsf{id},\mathbf{y}}) \to \langle \mathbf{x}, \mathbf{y} \rangle$**:** Parse $\mathsf{ct}_{\mathsf{id},\mathbf{x}} = \big(C, D, (E_\ell)_{\ell \in [L]}\big)$ and $\mathsf{sk}_{\mathsf{id},\mathbf{y}} = (\mathbf{y}, \sigma, \kappa)$. Compute

$$E_{\langle \mathbf{x}, \mathbf{y} \rangle} = \frac{\prod_{\ell \in [L]} {E_\ell}^{y_\ell}}{e(C, \sigma) \cdot D^\kappa}$$

and output $\log_{e(g,g)} E_{\langle \mathbf{x}, \mathbf{y} \rangle}$.

Correctness. Since it holds that

$$\begin{aligned}
\prod_{\ell\in[L]} {E_\ell}^{y_\ell} &= \prod_{\ell\in[L]} (e(g,g)^{x_\ell}\cdot e(h_1,h_{\mathsf{id},\ell})^s)^{y_\ell} = e(g,g)^{\langle \mathbf{x},\mathbf{y}\rangle}\cdot \prod_{\ell\in[L]} e(h_1,h_{\mathsf{id},\ell})^{sy_\ell},\\
e(C,\sigma) &= e(g^s, \prod_{\ell\in[L]} (h_{\mathsf{id},\ell}\cdot h_2^{-r_{\mathsf{id},\ell}})^{ay_\ell})\\
&= \prod_{\ell\in[L]} e(h_1,h_{\mathsf{id},\ell})^{sy_\ell}\cdot \prod_{\ell\in[L]} e(h_1,h_2)^{-s\langle \mathbf{r}_{\mathsf{id}},\mathbf{y}\rangle},\\
D^\kappa &= e(h_1,h_2)^{s\langle \mathbf{r}_{\mathsf{id}},\mathbf{y}\rangle},
\end{aligned}$$

we have

$$\begin{aligned}
E_{\langle \mathbf{x},\mathbf{y}\rangle} &= \frac{e(g,g)^{\langle \mathbf{x},\mathbf{y}\rangle}\cdot \prod_{\ell\in[L]} e(h_1,h_{\mathsf{id},\ell})^{sy_\ell}}{\prod_{\ell\in[L]} e(h_1,h_{\mathsf{id},\ell})^{sy_\ell}\cdot \prod_{\ell\in[L]} e(h_1,h_2)^{-s\langle \mathbf{r}_{\mathsf{id}},\mathbf{y}\rangle}\cdot e(h_1,h_2)^{s\langle \mathbf{r}_{\mathsf{id}},\mathbf{y}\rangle}}\\
&= e(g,g)^{\langle \mathbf{x},\mathbf{y}\rangle}.
\end{aligned}$$

5.3 Tight AD-SIM Security in the ROM

To conclude this section, we prove the following theorem.

Theorem 3. *If the* DBDH *assumption holds and* $\mathcal{F}$ *is a* PRF *family, the proposed* IBIPFE *scheme* Π_{DBDH} *in Section 5.2 satisfies the tight* AD-SIM *security in the random oracle model. In particular, for any PPT* $\mathcal{A}$ *that breaks the* AD-SIM *security of* Π_{DBDH}*, there exist PPT* $\mathcal{B}_1$ *and* $\mathcal{B}_2$ *such that*

$$\mathsf{Adv}^{\mathsf{AD\text{-}SIM}}_{\Pi_{\mathsf{DBDH}},\mathcal{A}}(\lambda) \le \mathsf{Adv}_{\mathcal{F},\mathcal{B}_1}(\lambda) + \mathsf{Adv}^{\mathsf{DBDH}}_{\mathcal{B}_2}(\lambda)$$

and

$$\max\{\mathsf{Time}(\mathcal{B}_1),\mathsf{Time}(\mathcal{B}_2)\} = \mathsf{Time}(\mathcal{A}) + \mathsf{poly}(\lambda, L).$$

A proof of Theorem 3 is a combination of Coron's proof for their IBE scheme [16] and Agrawal et al.'s information-theoretic argument for their IPFE scheme [7]. We use a game sequence $\mathsf{Game}_0,\dots,\mathsf{Game}_5$. Game_1 is a simple change. In Game_2 (resp. Game_3), we follow Coron's proof [16] and change answers of random oracle queries and secret key queries (resp. challenge query). In Game_4 and Game_5, we apply Agrawal et al.'s information-theoretic argument [7] (with slight modification) and conclude the proof.

Proof of Theorem 3. At first, we define the following simulation algorithms $(\mathsf{Setup}^\star, \mathsf{KGen}^\star_0, \mathsf{Enc}^\star, \mathsf{KGen}^\star_1)$. In advance, we show how to simulate the random oracle at the end of the proof. Upon a query on id, we sample $\mathbf{r}_{\mathsf{id}} = [r_{\mathsf{id},1},\dots,r_{\mathsf{id},L}], \mathbf{t}_{\mathsf{id}} = [t_{\mathsf{id},1},\dots,t_{\mathsf{id},L}] \leftarrow_R \mathbb{Z}_p^L$, set $\mathsf{H}(\mathsf{id}) = (h_{\mathsf{id},1},\dots,h_{\mathsf{id},L}) \in \mathbb{G}^L$;

$$h_{\mathsf{id},\ell} = g^{t_{\mathsf{id},\ell}}\cdot h_2^{r_{\mathsf{id},\ell}}$$

for $\ell \in [L]$, and store $(\mathsf{id},\mathbf{r}_{\mathsf{id}},\mathbf{t}_{\mathsf{id}}) \in \mathsf{st}$.

$\mathsf{Setup}^\star(1^\lambda, 1^L) \to (\mathsf{mpk}^\star, \mathsf{msk}^\star)$**:** Run $(p, \mathbb{G}, \mathbb{G}_T, g, e) \leftarrow \mathcal{G}(1^\lambda)$, sample $a, b \leftarrow_R \mathbb{Z}_p$, set $h_1 = g^a, h_2 = g^b$, and output

$$\mathsf{mpk}^\star = (p, \mathbb{G}, \mathbb{G}_T, g, e, h_1, h_2), \qquad \mathsf{msk}^\star = (a, b).$$

$\mathsf{KGen}_0^\star(\mathsf{mpk}^\star, \mathsf{msk}^\star, \mathsf{id}, \mathbf{y}, \mathsf{st}) \to \mathsf{sk}_{\mathsf{id},\mathbf{y}}^\star$**:** Parse $\mathbf{y} = [y_1, \ldots, y_L] \in \mathbb{Z}_p^L$. Retrieve $(\mathsf{id}, \mathbf{r}_{\mathsf{id}}, \mathbf{t}_{\mathsf{id}}) \in \mathsf{st}$ if it is stored. Otherwise, sample $\mathbf{r}_{\mathsf{id}} = [r_{\mathsf{id},1}, \ldots, r_{\mathsf{id},L}], \mathbf{t}_{\mathsf{id}} = [t_{\mathsf{id},1}, \ldots, t_{\mathsf{id},L}] \leftarrow_R \mathbb{Z}_p^L$ and store $(\mathsf{id}, \mathbf{r}_{\mathsf{id}}, \mathbf{t}_{\mathsf{id}}) \in \mathsf{st}$. Then, output

$$\mathsf{sk}_{\mathsf{id},\mathbf{y}}^\star = \Big(\mathbf{y}, \sigma^\star = h_1^{\langle \mathbf{t}_{\mathsf{id}}, \mathbf{y} \rangle}, \kappa^\star = \langle \mathbf{r}_{\mathsf{id}}, \mathbf{y} \rangle \mod p\Big).$$

$\mathsf{Enc}^\star(\mathsf{mpk}^\star, \mathsf{msk}^\star, \mathsf{id}^\star, \mathcal{V}, \mathsf{st}) \to \mathsf{ct}^\star$**:** Parse $\mathsf{mpk}^\star = (p, \mathbb{G}, \mathbb{G}_T, g, e, h_1, h_2)$, $\mathsf{msk}^\star = (a, b)$, and

$$\mathcal{V} = (\mathbf{y}_{\mathsf{id}^\star,i}, z_i^{\mathsf{pre}} = \langle \mathbf{x}^\star, \mathbf{y}_{\mathsf{id}^\star,i} \rangle \mod p)_{i \in [Q_{\mathsf{id}^\star}]}.$$

Suppose that a set $\{\mathbf{y}_{\mathsf{id}^\star,1}, \ldots, \mathbf{y}_{\mathsf{id}^\star,Q_{\mathsf{id}^\star}}\}$ contains Q_{pre} linearly independent vectors and $\big\{\mathbf{y}_{\mathsf{id}^\star,1}, \ldots, \mathbf{y}_{\mathsf{id}^\star,Q_{\mathsf{pre}}}\big\} \subseteq \{\mathbf{y}_{\mathsf{id}^\star,1}, \ldots, \mathbf{y}_{\mathsf{id}^\star,Q_{\mathsf{id}^\star}}\}$ is linearly independent for simplicity. Retrieve $(\mathsf{id}^\star, \mathbf{r}_{\mathsf{id}^\star}, \mathbf{t}_{\mathsf{id}^\star}) \in \mathsf{st}$. Let

$$\mathbf{Y}^{\mathsf{pre}} = \big[\mathbf{y}_{\mathsf{id}^\star,1} \mid \ldots \mid \mathbf{y}_{\mathsf{id}^\star,Q_{\mathsf{pre}}}\big] \in \mathbb{Z}_p^{L \times Q_{\mathsf{pre}}}, \qquad \mathbf{z}^{\mathsf{pre}} = \big[z_1^{\mathsf{pre}}, \ldots, z_{Q_{\mathsf{pre}}}^{\mathsf{pre}}\big] \in \mathbb{Z}_p^{Q_{\mathsf{pre}}},$$

where it holds that

$$\mathbf{Y}^{\mathsf{pre}\top} \mathbf{x}^\star = \mathbf{z}^{\mathsf{pre}}.$$

Compute an arbitrary $\widehat{\mathbf{x}} = [\hat{x}_1, \ldots, \hat{x}_L] \in \mathbb{Z}_p^L$ such that

$$\mathbf{Y}^{\mathsf{pre}\top} \widehat{\mathbf{x}} = \mathbf{z}^{\mathsf{pre}} \mod p.$$

Sample $s, s' \leftarrow_R \mathbb{Z}_p$ and output

$$\mathsf{ct}^\star = \begin{pmatrix} C^\star = g^s, D^\star = e(h_1, h_2)^{s'}, \\ (E_\ell^\star = e(g,g)^{\hat{x}_\ell} \cdot e(C^\star, h_1^{t_{\mathsf{id}^\star,\ell}}) \cdot D^{\star\, r_{\mathsf{id}^\star,\ell}})_{\ell \in [L]} \end{pmatrix}$$

and store $(\widehat{\mathbf{x}}, s, s') \in \mathsf{st}$.

$\mathsf{KGen}_1^\star(\mathsf{mpk}^\star, \mathsf{msk}^\star, \mathsf{id}, \mathbf{y}, z = \mathsf{Eval}(\mathsf{id}^\star, \mathbf{x}^\star, \mathsf{id}, \mathbf{y}), \mathsf{st}) \to \mathsf{sk}_{\mathsf{id},\mathbf{y}}^\star$**:** Parse $\mathsf{mpk}^\star = (p, \mathbb{G}, \mathbb{G}_T, g, e, h_1, h_2)$ and $\mathsf{msk}^\star = (a, b)$. If $\mathsf{id} \neq \mathsf{id}^\star$ holds, run $\mathsf{sk}_{\mathsf{id},\mathbf{y}}^\star \leftarrow \mathsf{KGen}_0^\star(\mathsf{mpk}^\star, \mathsf{msk}^\star, \mathsf{id}, \mathbf{y}, \mathsf{st})$ and output $\mathsf{sk}_{\mathsf{id},\mathbf{y}}^\star$. Otherwise, retrieve $(\mathsf{id}^\star, \mathbf{r}_{\mathsf{id}^\star}, \mathbf{t}_{\mathsf{id}^\star}) \in \mathsf{st}$ and $(\widehat{\mathbf{x}}, s, s') \in \mathsf{st}$. Output

$$\mathsf{sk}_{\mathsf{id}^\star,\mathbf{y}}^\star = \left(\mathbf{y}, \sigma^\star = h_1^{\langle \mathbf{t}_{\mathsf{id}^\star}, \mathbf{y} \rangle - \frac{\langle \widehat{\mathbf{x}}, \mathbf{y} \rangle - z}{a(s'-s)}}, \kappa^\star = \langle \mathbf{r}_{\mathsf{id}^\star}, \mathbf{y} \rangle + \frac{\langle \widehat{\mathbf{x}}, \mathbf{y} \rangle - z}{ab(s'-s)} \mod p\right).$$

We check that decryption results are consistent for $(\mathbf{y}, \sigma^\star, \kappa^\star) \leftarrow \mathsf{KGen}_0^\star(\mathsf{mpk}^\star, \mathsf{msk}^\star, \mathsf{id}, \mathbf{y}, \mathsf{st})$ and $(\mathbf{y}, \sigma^\star, \kappa^\star) \leftarrow \mathsf{KGen}_1^\star(\mathsf{mpk}^\star, \mathsf{msk}^\star, \mathsf{id}, \mathbf{y}, z = \mathsf{Eval}(\mathsf{id}^\star, \mathbf{x}^\star, \mathsf{id}, \mathbf{y}), \mathsf{st})$. For $\big(C, D, (E_\ell)_{\ell \in [L]}\big) \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}, \mathbf{x})$, we have

$$\prod_{\ell \in [L]} {E_\ell}^{y_\ell} = \prod_{\ell \in [L]} \big(e(g,g)^{x_\ell} \cdot e(h_1, g^{t_{\mathsf{id},\ell}} \cdot h_2^{r_{\mathsf{id},\ell}})^s\big)^{y_\ell}$$

$$= e(g,g)^{\langle \mathbf{x},\mathbf{y}\rangle + as\langle \mathbf{t}_{\mathsf{id}},\mathbf{y}\rangle + abs\langle \mathbf{r}_{\mathsf{id}},\mathbf{y}\rangle}.$$

For $(\mathbf{y},\sigma^\star,\kappa^\star) \leftarrow \mathsf{KGen}_0^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id},\mathbf{y},\mathsf{st})$ and $(\mathbf{y},\sigma^\star,\kappa^\star) \leftarrow \mathsf{KGen}_1^\star(\mathsf{mpk}^\star,$ $\mathsf{msk}^\star,\mathsf{id},\mathbf{y},z=\mathsf{Eval}(\mathsf{id}^\star,\mathbf{x}^\star,\mathsf{id},\mathbf{y}),\mathsf{st})$ for $\mathsf{id}\neq\mathsf{id}^\star$, we have

$$\begin{aligned} e(C,\sigma^\star) &= e(g^s, h_1^{\langle \mathbf{t}_{\mathsf{id}},\mathbf{y}\rangle}) = e(g,g)^{as\langle \mathbf{t}_{\mathsf{id}},\mathbf{y}\rangle}, \\ D^{\kappa^\star} &= e(h_1,h_2)^{s\langle \mathbf{r}_{\mathsf{id}},\mathbf{y}\rangle} = e(g,g)^{abs\langle \mathbf{r}_{\mathsf{id}},\mathbf{y}\rangle}. \end{aligned}$$

Thus, it holds that

$$\begin{aligned} &\mathsf{Dec}(\mathsf{mpk},\mathsf{Enc}(\mathsf{mpk},\mathsf{id},\mathbf{x}),\mathsf{KGen}_0^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id},\mathbf{y},\mathsf{st})) \\ =\ &\mathsf{Dec}(\mathsf{mpk},\mathsf{Enc}(\mathsf{mpk},\mathsf{id},\mathbf{x}),\mathsf{KGen}_1^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id},\mathbf{y},z=\mathsf{Eval}(\mathsf{id}^\star,\mathbf{x}^\star,\mathsf{id},\mathbf{y}),\mathsf{st})) \\ =\ &\langle \mathbf{x},\mathbf{y}\rangle \mod p. \end{aligned}$$

For $(\mathbf{y},\sigma^\star,\kappa^\star) \leftarrow \mathsf{KGen}_1^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id}^\star,\mathbf{y},z=\mathsf{Eval}(\mathsf{id}^\star,\mathbf{x}^\star,\mathsf{id}^\star,\mathbf{y}),\mathsf{st})$, we have

$$\begin{aligned} e(C,\sigma^\star) &= e(g^s, h_1^{\langle \mathbf{t}_{\mathsf{id}^\star},\mathbf{y}\rangle - \frac{\langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z}{a(s'-s)}}) = e(g,g)^{as\langle \mathbf{t}_{\mathsf{id}^\star},\mathbf{y}\rangle - \frac{s}{s'-s}(\langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z)}, \\ D^{\kappa^\star} &= e(h_1,h_2)^{s\left(\langle \mathbf{r}_{\mathsf{id}^\star},\mathbf{y}\rangle + \frac{\langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z}{ab(s'-s)}\right)} = e(g,g)^{abs\langle \mathbf{r}_{\mathsf{id}^\star},\mathbf{y}\rangle + \frac{s}{s'-s}(\langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z)}. \end{aligned}$$

Thus, it holds that

$$\begin{aligned} &\mathsf{Dec}(\mathsf{mpk},\mathsf{Enc}(\mathsf{mpk},\mathsf{id}^\star,\mathbf{x}),\mathsf{KGen}_1^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id}^\star,\mathbf{y},z=\mathsf{Eval}(\mathsf{id}^\star,\mathbf{x}^\star,\mathsf{id},\mathbf{y}),\mathsf{st})) \\ =\ &\langle \mathbf{x},\mathbf{y}\rangle \mod p. \end{aligned}$$

For $\left(C^\star, D^\star, (E_\ell^\star)_{\ell\in[L]}\right) \leftarrow \mathsf{Enc}^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id}^\star,\mathcal{V},\mathsf{st})$, we have

$$\begin{aligned} \prod_{\ell\in[L]} E_\ell^{\star\, y_\ell} &= \prod_{\ell\in[L]} \left(e(g,g)^{\hat{x}_\ell}\cdot e(C^\star, h_1^{t_{\mathsf{id}^\star,\ell}})\cdot D^{\star\, r_{\mathsf{id}^\star,\ell}}\right)^{y_\ell} \\ &= e(g,g)^{\langle \hat{\mathbf{x}},\mathbf{y}\rangle + as\langle \mathbf{t}_{\mathsf{id}^\star},\mathbf{y}\rangle + abs'\langle \mathbf{r}_{\mathsf{id}^\star},\mathbf{y}\rangle}. \end{aligned}$$

For $(\mathbf{y},\sigma^\star,\kappa^\star) \leftarrow \mathsf{KGen}_0^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id}^\star,\mathbf{y},\mathsf{st})$, we have

$$\begin{aligned} e(C^\star,\sigma^\star) &= e(g^s, h_1^{\langle \mathbf{t}_{\mathsf{id}^\star},\mathbf{y}\rangle}) = e(g,g)^{as\langle \mathbf{t}_{\mathsf{id}^\star},\mathbf{y}\rangle}, \\ D^{\star\,\kappa^\star} &= e(h_1,h_2)^{s'\langle \mathbf{r}_{\mathsf{id}^\star},\mathbf{y}\rangle} = e(g,g)^{abs'\langle \mathbf{r}_{\mathsf{id}^\star},\mathbf{y}\rangle}. \end{aligned}$$

Thus, it holds that

$$\begin{aligned} &\mathsf{Dec}(\mathsf{mpk},\mathsf{Enc}^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id}^\star,\mathsf{st}),\mathsf{KGen}_0^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id}^\star,\mathbf{y},\mathsf{st})) \\ =\ &\langle \widehat{\mathbf{x}},\mathbf{y}\rangle \mod p \\ =\ &\langle \mathbf{x}^\star,\mathbf{y}\rangle \mod p. \end{aligned}$$

For $(\mathbf{y},\sigma^\star,\kappa^\star) \leftarrow \mathsf{KGen}_1^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id}^\star,\mathbf{y},z=\mathsf{Eval}(\mathsf{id}^\star,\mathbf{x}^\star,\mathsf{id}^\star,\mathbf{y}),\mathsf{st})$, we have

$$e(C^\star,\sigma^\star) = e(g^s, h_1^{\langle \mathbf{t}_{\mathsf{id}^\star},\mathbf{y}\rangle - \frac{\langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z}{a(s'-s)}}) = e(g,g)^{as\langle \mathbf{t}_{\mathsf{id}^\star},\mathbf{y}\rangle - \frac{s}{s'-s}\cdot(\langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z)},$$

$$D^{\star\kappa^\star} = e(h_1,h_2)^{s'\left(\langle \mathbf{r}_{\mathsf{id}^\star},\mathbf{y}\rangle + \frac{\langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z}{ab(s'-s)}\right)} = e(g,g)^{abs'\langle \mathbf{r}_{\mathsf{id}^\star},\mathbf{y}\rangle + \frac{s'}{s'-s}(\langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z)}.$$

Moreover, we have

$$\begin{aligned} e(C^\star,\sigma^\star)\cdot D^{\star\kappa^\star} &= e(g,g)^{as\langle \mathbf{t}_{\mathsf{id}^\star},\mathbf{y}\rangle + abs'\langle \mathbf{r}_{\mathsf{id}^\star},\mathbf{y}\rangle - \frac{s}{s'-s}\cdot(\langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z) + \frac{s'}{s'-s}(\langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z)} \\ &= e(g,g)^{as\langle \mathbf{t}_{\mathsf{id}^\star},\mathbf{y}\rangle + abs'\langle \mathbf{r}_{\mathsf{id}^\star},\mathbf{y}\rangle + \langle \widehat{\mathbf{x}},\mathbf{y}\rangle - z}. \end{aligned}$$

Thus, it holds that

$$\begin{aligned} &\mathsf{Dec}(\mathsf{mpk},\mathsf{Enc}^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id}^\star,\mathsf{st}),\mathsf{KGen}_1^\star(\mathsf{mpk}^\star,\mathsf{msk}^\star,\mathsf{id}^\star,\mathbf{y},z,\mathsf{st})) \\ &= z \mod p \\ &= \langle \mathbf{x}^\star,\mathbf{y}\rangle \mod p. \end{aligned}$$

Hereafter, we prove the computational indistinguishability of a game sequence $\mathsf{SIM}_{\mathsf{Real}} = \mathsf{Game}_0,\dots,\mathsf{Game}_5 = \mathsf{SIM}_{\mathsf{Ideal}}$. For simplicity, we assume that $\mathcal{A}$ makes a random oracle query on id before it makes secret key queries on $(\mathsf{id},\mathbf{y})$ and challenge query on $(\mathsf{id}^\star = \mathsf{id},\mathbf{x}^\star)$ throughout the game.

Game_0. This is the $\mathsf{SIM}_{\mathsf{Real}}$ experiment. At the beginning of the game, $\mathcal{C}$ runs $(p,\mathbb{G},\mathbb{G}_T,g,e) \leftarrow \mathcal{G}(1^\lambda)$, samples $a,b \leftarrow_R \mathbb{Z}_p$, chooses a random function $\mathsf{H} \leftarrow_R \mathsf{Func}(\mathcal{ID},\mathbb{G}^L)$ and an index of a function family $k \leftarrow_R \mathcal{K}$ such that $f_k : \mathcal{ID} \to \mathbb{Z}_p^L$, and sets

$$\mathsf{mpk} = \big(p,\mathbb{G},\mathbb{G}_T,g,e,h_1 = g^a,h_2 = g^b\big), \qquad \mathsf{msk} = (a,k).$$

Then, $\mathcal{C}$ answers $\mathcal{A}$'s queries as follows.

- Upon $\mathcal{A}$'s random oracle queries on $\mathsf{id} \in \mathcal{ID}$, $\mathcal{C}$ computes $\mathsf{H}(\mathsf{id}) = (h_{\mathsf{id},1},\dots,h_{\mathsf{id},L}) \in \mathbb{G}^L$ and returns $(h_{\mathsf{id},1},\dots,h_{\mathsf{id},L})$.
- Upon $\mathcal{A}$'s secret key query on $(\mathsf{id},\mathbf{y} = [y_1,\dots,y_L])$, $\mathcal{C}$ computes $\mathsf{H}(\mathsf{id}) = (h_{\mathsf{id},1},\dots,h_{\mathsf{id},L}) \in \mathbb{G}^L$ and $f_k(\mathsf{id}) = \mathbf{r}_{\mathsf{id}} \in \mathbb{Z}_p^L$, and returns

$$\mathsf{sk}_{\mathsf{id},\mathbf{y}} = \left(\mathbf{y},\sigma = \prod_{\ell\in[L]} (h_{\mathsf{id},\ell}\cdot h_2^{-r_{\mathsf{id},\ell}})^{ay_\ell},\kappa = \langle \mathbf{r}_{\mathsf{id}},\mathbf{y}\rangle \mod p\right).$$

- Upon $\mathcal{A}$'s challenge query on $(\mathsf{id}^\star,\mathbf{x}^\star = (x_1^\star,\dots,x_L^\star))$, $\mathcal{C}$ computes $\mathsf{H}(\mathsf{id}^\star) = (h_{\mathsf{id}^\star,1},\dots,h_{\mathsf{id}^\star,L}) \in \mathbb{G}^L$, samples $s \leftarrow_R \mathbb{Z}_p$, and returns

$$\mathsf{ct}^\star = \Big(C = g^s, D = e(h_1,h_2)^s, (E_\ell = e(g,g)^{x_\ell^\star}\cdot e(h_1,h_{\mathsf{id}^\star,\ell})^s)_{\ell\in[L]}\Big).$$

Game_1. This is the same as Game_0 except that a pseudo-random $f_k(\mathsf{id}) = \mathbf{r}_{\mathsf{id}} \in \mathbb{Z}_p^L$ is replaced with $\mathbf{r}_{\mathsf{id}} \leftarrow_R \mathbb{Z}_p^L$. In particular, $\mathcal{C}$ samples $\mathbf{r}_{\mathsf{id}} \leftarrow_R \mathbb{Z}_p^L$ upon $\mathcal{A}$'s first secret key query associated with id and stores $(\mathsf{id},\mathbf{r}_{\mathsf{id}}) \in \mathsf{st}$. Since $\mathcal{F}$ is a PRF family, Game_0 and Game_1 are computationally indistinguishable.

Game_2. This game is the same as Game_1 except for the way $\mathcal{C}$ answers $\mathcal{A}$'s random oracle queries and secret key queries. Upon $\mathcal{A}$'s random oracle query on id, $\mathcal{C}$ samples $\mathbf{r}_{\mathsf{id}} = [r_{\mathsf{id},1}, \ldots, r_{\mathsf{id},L}], \mathbf{t}_{\mathsf{id}} = [t_{\mathsf{id},1}, \ldots, t_{\mathsf{id},L}] \leftarrow_R \mathbb{Z}_p^L$ and stores $(\mathsf{id}, \mathbf{r}_{\mathsf{id}}, \mathbf{t}_{\mathsf{id}}) \in \mathsf{st}$. Then, $\mathcal{C}$ computes

$$h_{\mathsf{id},\ell} = g^{t_{\mathsf{id},\ell}} \cdot h_2^{r_{\mathsf{id},\ell}}$$

for $\ell \in [L]$ and returns $(h_{\mathsf{id},1}, \ldots, h_{\mathsf{id},L}) \in \mathbb{G}^L$. Upon $\mathcal{A}$'s secret key query on $(\mathsf{id}, \mathbf{y} = (y_1, \ldots, y_L))$, $\mathcal{C}$ retrieves $(\mathsf{id}, \mathbf{r}_{\mathsf{id}}, \mathbf{t}_{\mathsf{id}}) \in \mathsf{st}$ and uses the same $\mathbf{r}_{\mathsf{id}}$ to answer the query. Therefore, we have

$$\sigma = \prod_{\ell \in [L]} (h_{\mathsf{id},\ell} \cdot h_2^{-r_{\mathsf{id},\ell}})^{a y_\ell} = \prod_{\ell \in [L]} g^{a t_{\mathsf{id},\ell} y_\ell} = h_1^{\langle \mathbf{t}_{\mathsf{id}}, \mathbf{y} \rangle}.$$

Due to the fresh randomness of $\mathbf{t}_{\mathsf{id}}$ (resp. $\mathbf{r}_{\mathsf{id}}$), $\mathcal{C}$'s answers of random oracle queries (resp. secret key queries) follow the same distribution between Game_1 and Game_2. Thus, Game_1 and Game_2 follow the same distribution.

Game_3. This game is the same as Game_2 except for the way $\mathcal{C}$ answers $\mathcal{A}$'s challenge query. Upon $\mathcal{A}$'s challenge query on $(\mathsf{id}^\star, \mathbf{x}^\star = [x_1^\star, \ldots, x_L^\star])$, $\mathcal{C}$ samples $s \leftarrow_R \mathbb{Z}_p$, computes

$$C = g^s, \qquad D = e(h_1, h_2)^s,$$

and returns

$$\mathsf{ct}^\star = \Big(C, D, (E_\ell = e(g,g)^{x_\ell^\star} \cdot e(C, h_1^{t_{\mathsf{id}^\star,\ell}}) \cdot D^{r_{\mathsf{id}^\star,\ell}})_{\ell \in [L]}\Big).$$

Due to the bilinearity of e and the modification in Game_2, we have

$$\begin{aligned} E_\ell &= e(g,g)^{x_\ell^\star} \cdot e(g^s, h_1^{t_{\mathsf{id}^\star,\ell}}) \cdot e(h_1, h_2)^{s r_{\mathsf{id}^\star,\ell}} \\ &= e(g,g)^{x_\ell^\star} \cdot e(h_1, g^{t_{\mathsf{id}^\star,\ell}})^s \cdot e(h_1, h_2^{r_{\mathsf{id}^\star,\ell}})^s \\ &= e(g,g)^{x_\ell^\star} \cdot e(h_1, g^{t_{\mathsf{id}^\star,\ell}} \cdot h_2^{r_{\mathsf{id}^\star,\ell}})^s \\ &= e(g,g)^{x_\ell^\star} \cdot e(h_1, h_{\mathsf{id}^\star,\ell})^s. \end{aligned}$$

Thus, Game_2 and Game_3 follow the same distribution.

Game_4. This game is the same as Game_3 except for the way $\mathcal{C}$ computes C and D upon $\mathcal{A}$'s challenge query. Upon the query on $(\mathsf{id}^\star, \mathbf{x}^\star = [x_1^\star, \ldots, x_L^\star])$, $\mathcal{C}$ samples $s, s' \leftarrow_R \mathbb{Z}_p$ and computes

$$C = g^s, \qquad D = e(h_1, h_2)^{s'},$$

while $\mathcal{C}$ computes $(E_\ell)_{\ell \in [L]}$ in the same way as in Game_3 by using C and D. Due to the DBDH assumption, Game_3 and Game_4 are computationally indistinguishable. To prove the computational indistinguishability, we construct a reduction algorithm $\mathcal{B}_2$ that utilizes $\mathcal{A}$ to distinguish Game_3 and Game_4, and solves the DBDH problem. Given $((p, \mathbb{G}, \mathbb{G}_T, g, e), g^a, g^b, g^c, T)$, where $T = e(g,g)^{abc}$ or $T = e(g,g)^d$, $\mathcal{B}_2$ sends

$$\mathsf{mpk} = \big(p, \mathbb{G}, \mathbb{G}_T, g, e, h_1 = g^a, h_2 = g^b\big)$$

to $\mathcal{A}$. Then, $\mathcal{B}_2$ answers $\mathcal{A}$'s queries as follows.

- Upon $\mathcal{A}$'s random oracle queries on $\mathsf{id} \in \mathcal{ID}$, $\mathcal{B}_2$ samples $\mathbf{r}_{\mathsf{id}} = [r_{\mathsf{id},1}, \ldots, r_{\mathsf{id},L}], \mathbf{t}_{\mathsf{id}} = [t_{\mathsf{id},1}, \ldots, t_{\mathsf{id},L}] \leftarrow_R \mathbb{Z}_p^L$ and stores $(\mathsf{id}, \mathbf{r}_{\mathsf{id}}, \mathbf{t}_{\mathsf{id}}) \in \mathsf{st}$. Then, $\mathcal{B}_2$ computes

$$h_{\mathsf{id},\ell} = g^{t_{\mathsf{id},\ell}} \cdot h_2^{r_{\mathsf{id},\ell}}$$

for $\ell \in [L]$ and returns $(h_{\mathsf{id},1}, \ldots, h_{\mathsf{id},L}) \in \mathbb{G}^L$.
- Upon $\mathcal{A}$'s secret key query on $(\mathsf{id}, \mathbf{y} = [y_1, \ldots, y_L])$, $\mathcal{B}_2$ retrieves $(\mathsf{id}, \mathbf{r}_{\mathsf{id}}, \mathbf{t}_{\mathsf{id}}) \in \mathsf{st}$, and returns

$$\mathsf{sk}_{\mathsf{id},\mathbf{y}} = \Big(\mathbf{y}, \sigma = h_1^{\langle \mathbf{t}_{\mathsf{id}}, \mathbf{y} \rangle}, \kappa = \langle \mathbf{r}_{\mathsf{id}}, \mathbf{y} \rangle \mod p\Big).$$

- Upon $\mathcal{A}$'s challenge query on $(\mathsf{id}^\star, \mathbf{x}^\star = [x_1^\star, \ldots, x_L^\star])$, $\mathcal{B}_2$ retrieves $(\mathsf{id}^\star, \mathbf{r}_{\mathsf{id}^\star}, \mathbf{t}_{\mathsf{id}^\star}) \in \mathsf{st}$, sets

$$C = g^c, \qquad D = T,$$

and returns

$$\mathsf{ct}^\star = \Big(C, D, (E_\ell = e(g,g)^{x_\ell^\star} \cdot e(C, h_1^{t_{\mathsf{id}^\star,\ell}}) \cdot D^{r_{\mathsf{id}^\star,\ell}})_{\ell \in [L]}\Big).$$

We have completed the description of $\mathcal{B}_2$. If $T = e(g,g)^{abc}$ holds, $\mathsf{ct}^\star$ is distributed according to Game_3 by implicitly setting $s = c$ since it holds that $D = e(g,g)^{abc} = e(h_1, h_2)^s$.If $T = e(g,g)^d$ holds, $\mathsf{ct}^\star$ is distributed according to Game_4 by implicitly setting $s' = d/(ab)$ since it holds that $D = e(g,g)^d = e(h_1, h_2)^{s'}$.Thus, Game_3 and Game_4 are computationally indistinguishable from $\mathcal{A}$'s view.

Game_5. This is the same as Game_4 except that $\mathcal{C}$ does not use $(\mathsf{Enc}, \mathsf{KGen})$ but $(\mathsf{KGen}_0^\star, \mathsf{Enc}^\star, \mathsf{KGen}_1^\star)$ to answer $\mathcal{A}$'s queries. In particular, $\mathsf{Game}_5 = \mathsf{SIM}_{\mathsf{Ideal}}$ holds.

We conclude the proof by showing that Game_4 and Game_5 follow the same distribution. Let $\widehat{\mathbf{x}} = [\hat{x}_1, \ldots, \hat{x}_L] \in \mathbb{Z}_p^L$ denote a vector which $\mathcal{C}$ computes during $\mathsf{Enc}^\star$, where it holds that

$$\mathbf{Y}^{\mathsf{pre}\top} \widehat{\mathbf{x}} = \mathbf{z}^{\mathsf{pre}} \mod p.$$

Let $\Delta\mathbf{x} = [\varDelta x_1, \ldots, \varDelta x_L] \in \mathbb{Z}_p^L$ denote a vector such that

$$\Delta\mathbf{x} = \widehat{\mathbf{x}} - \mathbf{x}^\star \mod p,$$

where it holds that

$$\langle \Delta\mathbf{x}, \mathbf{y} \rangle = \langle \widehat{\mathbf{x}}, \mathbf{y} \rangle - \langle \mathbf{x}^\star, \mathbf{y} \rangle = 0 \mod p$$

for $\mathcal{A}$'s pre-challenge secret key queries on $(\mathsf{id}^\star, \mathbf{y})$. Thus, we have

$$\mathbf{Y}^{\mathsf{pre}\top} \Delta\mathbf{x} = \mathbf{Y}^{\mathsf{pre}\top} \widehat{\mathbf{x}} - \mathbf{Y}^{\mathsf{pre}\top} \mathbf{x}^\star = \mathbf{z}^{\mathsf{pre}} - \mathbf{z}^{\mathsf{pre}} = \mathbf{0} \mod p.$$

We can replace $\mathbf{r}_{\mathsf{id}^\star}, \mathbf{t}_{\mathsf{id}^\star} \leftarrow_R \mathbb{Z}_p^L$ in Game_5 with

$$\widehat{\mathbf{r}}_{\mathsf{id}^\star} = \mathbf{r}_{\mathsf{id}^\star} - \frac{1}{ab(s'-s)} \cdot \Delta\mathbf{x}, \qquad \widehat{\mathbf{t}}_{\mathsf{id}^\star} = \mathbf{t}_{\mathsf{id}^\star} + \frac{1}{a(s'-s)} \cdot \Delta\mathbf{x}$$

since $\widehat{\mathbf{r}}_{\mathsf{id}^\star} = [\hat{r}_{\mathsf{id}^\star,1}, \dots, \hat{r}_{\mathsf{id}^\star,L}]$ and $\widehat{\mathbf{t}}_{\mathsf{id}^\star} = [\hat{t}_{\mathsf{id}^\star,1}, \dots, \hat{t}_{\mathsf{id}^\star,L}]$ also follow the uniform distribution over $\mathbb{Z}_p^L$. Then, all $\mathcal{C}$'s answers upon $\mathcal{A}$'s queries become the same between Game_4 and Game_5. $\mathcal{C}$'s answer upon $\mathcal{A}$'s random oracle query on $\mathsf{id}^\star$ is the same since it holds that

$$\begin{aligned} g^{\hat{t}_{\mathsf{id}^\star,\ell}} \cdot h_2^{\hat{r}_{\mathsf{id}^\star,\ell}} &= g^{t_{\mathsf{id}^\star,\ell} + \frac{1}{a(s'-s)} \cdot \Delta x_\ell} \cdot h_2^{r_{\mathsf{id}^\star,\ell} - \frac{1}{ab(s'-s)} \cdot \Delta x_\ell} \\ &= g^{t_{\mathsf{id}^\star,\ell}} \cdot h_2^{r_{\mathsf{id}^\star,\ell}} \cdot g^{\frac{1}{a(s'-s)} \cdot \Delta x_\ell} \cdot h_2^{-\frac{1}{ab(s'-s)} \cdot \Delta x_\ell} \\ &= g^{t_{\mathsf{id}^\star,\ell}} \cdot h_2^{r_{\mathsf{id}^\star,\ell}}. \end{aligned}$$

$\mathcal{C}$'s answers upon $\mathcal{A}$'s pre-challenge secret key queries are the same since it holds that $\langle \widehat{\mathbf{r}}_{\mathsf{id}^\star}, \mathbf{y} \rangle = \langle \mathbf{r}_{\mathsf{id}^\star}, \mathbf{y} \rangle \mod p$ and $\langle \widehat{\mathbf{t}}_{\mathsf{id}^\star}, \mathbf{y} \rangle = \langle \mathbf{t}_{\mathsf{id}^\star}, \mathbf{y} \rangle \mod p$. $\mathcal{C}$'s answers upon $\mathcal{A}$'s post-challenge secret key queries are the same since it holds that

$$\begin{aligned} \sigma^\star &= h_1^{\langle \widehat{\mathbf{t}}_{\mathsf{id}^\star}, \mathbf{y} \rangle - \frac{\langle \widehat{\mathbf{x}}, \mathbf{y} \rangle - z}{a(s'-s)}} \\ &= h_1^{\langle \mathbf{t}_{\mathsf{id}^\star} + \frac{1}{a(s'-s)} \cdot \Delta\mathbf{x}, \mathbf{y} \rangle - \frac{\langle \widehat{\mathbf{x}}, \mathbf{y} \rangle - \langle \mathbf{x}^\star, \mathbf{y} \rangle}{a(s'-s)}} \\ &= h_1^{\langle \mathbf{t}_{\mathsf{id}^\star}, \mathbf{y} \rangle}, \\ \kappa^\star &= \langle \widehat{\mathbf{r}}_{\mathsf{id}^\star}, \mathbf{y} \rangle + \frac{\langle \widehat{\mathbf{x}}, \mathbf{y} \rangle - z}{ab(s'-s)} \mod p \\ &= \langle \mathbf{r}_{\mathsf{id}^\star} - \frac{1}{ab(s'-s)} \cdot \Delta\mathbf{x}, \mathbf{y} \rangle + \frac{\langle \widehat{\mathbf{x}}, \mathbf{y} \rangle - \langle \mathbf{x}^\star, \mathbf{y} \rangle}{ab(s'-s)} \mod p \\ &= \langle \mathbf{r}_{\mathsf{id}^\star}, \mathbf{y} \rangle \mod p. \end{aligned}$$

$\mathcal{C}$'s answers upon $\mathcal{A}$'s challenge query is the same since it holds that

$$\begin{aligned} &e(g,g)^{\hat{x}_\ell} \cdot e(C^\star, h_1^{\hat{t}_{\mathsf{id}^\star,\ell}}) \cdot D^{\star \hat{r}_{\mathsf{id}^\star,\ell}} \\ =\ &e(g,g)^{x^\star_\ell + \Delta x_\ell} \cdot e(g^s, h_1^{t_{\mathsf{id}^\star,\ell} + \frac{1}{a(s'-s)} \cdot \Delta x_\ell}) \cdot e(h_1, h_2)^{s'\left(r_{\mathsf{id}^\star,\ell} - \frac{1}{ab(s'-s)} \cdot \Delta x_\ell\right)} \\ =\ &e(g,g)^{x^\star_\ell} \cdot e(g^s, h_1^{t_{\mathsf{id}^\star,\ell}}) \cdot e(h_1, h_2)^{s' r_{\mathsf{id}^\star,\ell}} \cdot e(g,g)^{\Delta x_\ell \left(1 + \frac{s}{s'-s} - \frac{s'}{s'-s}\right)} \\ =\ &e(g,g)^{x^\star_\ell} \cdot e(g^s, h_1^{t_{\mathsf{id}^\star,\ell}}) \cdot e(h_1, h_2)^{s' r_{\mathsf{id}^\star,\ell}} \\ =\ &e(g,g)^{x^\star_\ell} \cdot e(C, h_1^{t_{\mathsf{id}^\star,\ell}}) \cdot D^{r_{\mathsf{id}^\star,\ell}}. \end{aligned}$$

Therefore, Game_4 and Game_5 follow the same distribution. □

Acknowledgement. We would like to thank anonymous reviewers of PKC 2025 for their valuable comments. This work was partially supported by JST CREST Grant Number JPMJCR2113, Japan and JSPS KAKENHI Grant Number 24K02939, Japan.

References

1. 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, Berlin, Heidelberg (Mar / Apr 2015). https://doi.org/10.1007/978-3-662-46447-2_33
2. Abdalla, M., Catalano, D., Gay, R., Ursu, B.: Inner-product functional encryption with fine-grained access control. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020, Part III. LNCS, vol. 12493, pp. 467–497. Springer, Cham (Dec 2020). https://doi.org/10.1007/978-3-030-64840-4_16
3. 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, Cham (Apr / May 2017). https://doi.org/10.1007/978-3-319-56620-7_21
4. Agrawal, S., Bhattacherjee, S., Phan, D.H., Stehlé, D., Yamada, S.: Efficient public trace and revoke from standard assumptions: Extended abstract. In: Thuraisingham, B.M., Evans, D., Malkin, T., Xu, D. (eds.) ACM CCS 2017. pp. 2277–2293. ACM Press (Oct / Nov 2017). https://doi.org/10.1145/3133956.3134041
5. Agrawal, S., Boneh, D., Boyen, X.: Efficient lattice (H)IBE in the standard model. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 553–572. Springer, Berlin, Heidelberg (May / Jun 2010). https://doi.org/10.1007/978-3-642-13190-5_28
6. Agrawal, S., Libert, B., Maitra, M., Titiu, R.: Adaptive simulation security for inner product functional encryption. In: Kiayias, A., Kohlweiss, M., Wallden, P., Zikas, V. (eds.) PKC 2020, Part I. LNCS, vol. 12110, pp. 34–64. Springer, Cham (May 2020). https://doi.org/10.1007/978-3-030-45374-9_2
7. Agrawal, S., Libert, B., Stehlé, D.: Fully secure functional encryption for inner products, from standard assumptions. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016, Part III. LNCS, vol. 9816, pp. 333–362. Springer, Berlin, Heidelberg (Aug 2016). https://doi.org/10.1007/978-3-662-53015-3_12
8. Ajtai, M.: Generating hard instances of the short basis problem. In: Wiedermann, J., van Emde Boas, P., Nielsen, M. (eds.) ICALP 99. LNCS, vol. 1644, pp. 1–9. Springer, Berlin, Heidelberg (Jul 1999). https://doi.org/10.1007/3-540-48523-6_1
9. Attrapadung, N., Hanaoka, G., Yamada, S.: A framework for identity-based encryption with almost tight security. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part I. LNCS, vol. 9452, pp. 521–549. Springer, Berlin, Heidelberg (Nov / Dec 2015). https://doi.org/10.1007/978-3-662-48797-6_22
10. Blazy, O., Kiltz, E., Pan, J.: (Hierarchical) identity-based encryption from affine message authentication. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 408–425. Springer, Berlin, Heidelberg (Aug 2014). https://doi.org/10.1007/978-3-662-44371-2_23
11. Boneh, D., Dagdelen, Ö., Fischlin, M., Lehmann, A., Schaffner, C., Zhandry, M.: Random oracles in a quantum world. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 41–69. Springer, Berlin, Heidelberg (Dec 2011). https://doi.org/10.1007/978-3-642-25385-0_3
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, Berlin, Heidelberg (Mar 2011). https://doi.org/10.1007/978-3-642-19571-6_16

13. Boyen, X., Li, Q.: Towards tightly secure lattice short signature and id-based encryption. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 404–434. Springer, Berlin, Heidelberg (Dec 2016). https://doi.org/10.1007/978-3-662-53890-6_14
14. Brakerski, Z., Langlois, A., Peikert, C., Regev, O., Stehlé, D.: Classical hardness of learning with errors. In: Boneh, D., Roughgarden, T., Feigenbaum, J. (eds.) 45th ACM STOC. pp. 575–584. ACM Press (Jun 2013). https://doi.org/10.1145/2488608.2488680
15. Chen, J., Wee, H.: Fully, (almost) tightly secure IBE and dual system groups. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 435–460. Springer, Berlin, Heidelberg (Aug 2013). https://doi.org/10.1007/978-3-642-40084-1_25
16. Coron, J.S.: A variant of boneh-franklin IBE with a tight reduction in the random oracle model. DCC **50**(1), 115–133 (2009). https://doi.org/10.1007/s10623-008-9218-2
17. Garg, S., Gentry, C., Halevi, S.: Candidate multilinear maps from ideal lattices. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 1–17. Springer, Berlin, Heidelberg (May 2013). https://doi.org/10.1007/978-3-642-38348-9_1
18. 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). https://doi.org/10.1109/FOCS.2013.13
19. 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, Berlin, Heidelberg (Jan 2016). https://doi.org/10.1007/978-3-662-49099-0_18
20. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Ladner, R.E., Dwork, C. (eds.) 40th ACM STOC. pp. 197–206. ACM Press (May 2008). https://doi.org/10.1145/1374376.1374407
21. Gong, J., Chen, J., Dong, X., Cao, Z., Tang, S.: Extended nested dual system groups, revisited. In: Cheng, C.M., Chung, K.M., Persiano, G., Yang, B.Y. (eds.) PKC 2016, Part I. LNCS, vol. 9614, pp. 133–163. Springer, Berlin, Heidelberg (Mar 2016). https://doi.org/10.1007/978-3-662-49384-7_6
22. Gong, J., Dong, X., Chen, J., Cao, Z.: Efficient IBE with tight reduction to standard assumption in the multi-challenge setting. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 624–654. Springer, Berlin, Heidelberg (Dec 2016). https://doi.org/10.1007/978-3-662-53890-6_21
23. Han, S., Liu, S., Qin, B., Gu, D.: Tightly CCA-secure identity-based encryption with ciphertext pseudorandomness. DCC **86**(3), 517–554 (2018). https://doi.org/10.1007/s10623-017-0339-3
24. Hofheinz, D., Jia, D., Pan, J.: Identity-based encryption tightly secure under chosen-ciphertext attacks. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 190–220. Springer, Cham (Dec 2018). https://doi.org/10.1007/978-3-030-03329-3_7
25. Katsumata, S., Yamada, S.: Partitioning via non-linear polynomial functions: More compact IBEs from ideal lattices and bilinear maps. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 682–712. Springer, Berlin, Heidelberg (Dec 2016). https://doi.org/10.1007/978-3-662-53890-6_23

26. Katsumata, S., Yamada, S., Yamakawa, T.: Tighter security proofs for GPV-IBE in the quantum random oracle model. Journal of Cryptology **34**(1), 5 (Jan 2021). https://doi.org/10.1007/s00145-020-09371-y
27. Lai, Q., Liu, F.H., Wang, Z.: New lattice two-stage sampling technique and its applications to functional encryption - stronger security and smaller ciphertexts. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part I. LNCS, vol. 12696, pp. 498–527. Springer, Cham (Oct 2021). https://doi.org/10.1007/978-3-030-77870-5_18
28. Lin, H., Luo, J.: Succinct and adaptively secure ABE for ABP from k-Lin. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020, Part III. LNCS, vol. 12493, pp. 437–466. Springer, Cham (Dec 2020). https://doi.org/10.1007/978-3-030-64840-4_15
29. Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 700–718. Springer, Berlin, Heidelberg (Apr 2012). https://doi.org/10.1007/978-3-642-29011-4_41
30. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. In: 45th FOCS. pp. 372–381. IEEE Computer Society Press (Oct 2004). https://doi.org/10.1109/FOCS.2004.72
31. O'Neill, A.: Definitional issues in functional encryption. Cryptology ePrint Archive, Report 2010/556 (2010), https://eprint.iacr.org/2010/556
32. Peikert, C.: Limits on the hardness of lattice problems in ell _p norms. In: (CCC 2007). pp. 333–346. IEEE Computer Society (2007). https://doi.org/10.1109/CCC.2007.12
33. Peikert, C., Rosen, A.: Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 145–166. Springer, Berlin, Heidelberg (Mar 2006). https://doi.org/10.1007/11681878_8
34. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Gabow, H.N., Fagin, R. (eds.) 37th ACM STOC. pp. 84–93. ACM Press (May 2005). https://doi.org/10.1145/1060590.1060603
35. Saito, T., Xagawa, K., Yamakawa, T.: Tightly-secure key-encapsulation mechanism in the quantum random oracle model. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part III. LNCS, vol. 10822, pp. 520–551. Springer, Cham (Apr / May 2018). https://doi.org/10.1007/978-3-319-78372-7_17
36. Wang, Z., Fan, X., Liu, F.H.: FE for inner products and its application to decentralized ABE. In: Lin, D., Sako, K. (eds.) PKC 2019, Part II. LNCS, vol. 11443, pp. 97–127. Springer, Cham (Apr 2019). https://doi.org/10.1007/978-3-030-17259-6_4
37. Wee, H.: Attribute-hiding predicate encryption in bilinear groups, revisited. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part I. LNCS, vol. 10677, pp. 206–233. Springer, Cham (Nov 2017). https://doi.org/10.1007/978-3-319-70500-2_8
38. Yamakawa, T., Zhandry, M.: Classical vs quantum random oracles. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part II. LNCS, vol. 12697, pp. 568–597. Springer, Cham (Oct 2021). https://doi.org/10.1007/978-3-030-77886-6_20