Pairing Cryptography Meets Isogeny: A New Framework of Isogenous Pairing Groups

Takeshi Koshiba and Katsuyuki Takashima
December 8, 2016

Saitama Univ. and Mitsubishi Electric

Abstract. We put forth a new mathematical framework called *Isogenous Pairing Groups (IPG)* and new intractable assumptions in the framework, the *Isogenous DBDH (Isog-DBDH) assumption* and its variants. Three operations, i.e., exponentiation, pairing and isogeny on elliptic curves are treated under a unified notion of *trapdoor homomorphisms*, and combinations of the operations have potential new cryptographic applications, in which the compatibility of pairing and isogeny is a main ingredient in IPG. As an example, we present constructions of (small and large universe) key-policy attribute-based encryption (KP-ABE) schemes *secure against pre-challenge quantum adversaries* in the quantum random oracle model (QROM). Note that our small universe KP-ABE has *asymptotically the same efficiency* as Goyal et al.'s small universe KP-ABE, which has only classical security. As a by-product, we also propose *practical* (hierarchical) identity-based encryption ((H)IBE) schemes *secure against pre-challenge quantum adversaries* in the QROM from isogenies, which are based on the Boneh-Franklin IBE and the Gentry-Silverberg HIBE, respectively.

Keywords: Attribute-Based Encryption, Post-Quantum Crypto., Isogeny

1 Introduction

1.1 Backgrounds

Since the seminal work of Boneh-Franklin, pairings on elliptic curves have many applications, i.e., identity-based encryption, short (group) signatures, attribute-based encryption, efficient anonymous credential systems, etc. Elliptic curves have another interesting operation called isogeny, which has been used for quantum-resistant cryptosystems [22, 33, 19]. This paper establishes a unified framework for these operations, namely, a notion of isogenous pairing groups (IPG). Using this framework, we can raise pairing cryptosystems to quantum resistant ones in a weak sense. As an example, we construct secure (H)IBE and ABE in a weak form of post-quantum security model.

(Key-policy) attribute-based encryption (KP-ABE) is a powerful and useful generalization of identity-based encryption (IBE). In a KP-ABE system, ciphertexts are associated with sets of attributes and user secret keys distributed by an authority are associated with formulas over attributes. A user should be able to decrypt a ciphertext if and only if the secret key formula is satisfied by the ciphertext attributes. Hence, there exists a hierarchical structure in keys, namely, master secret keys and user-level keys, where the master secret key can generate a user key for any formula. Therefore, leakage of the master secret key is more serious than the leakage of each user key.

All pairing-based ABE proposals (e.g., [29, 35]) would be totally broken by the emergence of quantum computers [38], in particular, the master secret key would be revealed. In the post-quantum era, it is important to confirm whether classical cryptographic techniques are still secure against quantum adversary. Recently, strong security notions and constructions against quantum computers have been intensively studied [9, 49, 48, 13, 14, 21, 20, 4]. All existing quantum-resistant IBE and ABE schemes have been constructed only from lattices [26, 2, 28, 11]. However, it is important to propose IBE/ABE schemes with quantum resistance from another mathematical foundation since, for example, a very recent NIST report [16, Sec. 2] (draft) mentions that "..., it has proven difficult to give precise estimates of the security of lattice schemes against even known cryptanalysis techniques."

Here, we demonstrate a power of isogenies on elliptic curves for the issue and open a new research avenue (or let pairing-based IBE/ABE (partially) survive in the quantum world). In [15], Charles et al. constructed a hash function based on the intractability of computing a (large degree) isogeny between two supersingular elliptic curves given only the two curves. Hence, the isogeny one-way function is another quantum-resistant mathematical tool, and several post-quantum cryptosystems have been proposed by using isogenies [22, 33, 19, 25]. But, these proposals achieve just limited functionalities and security from several reasons. First, the previous isogeny cryptosystems use only isogenies. Here, we also employ pairing operations since supersingular curves are suitable for deploying pairing crypto as well. Second, there exist only limited foundational isogeny-related computational assumptions for security, i.e., isogeny computation hardness [15, 22] and supersingular isogeny DH (SIDH) hardness assumptions [22, 19]. Both assumptions are basic, but have limited applications. For enjoying various advantages of pairing assumptions in isogeny crypto, we establish a new mathematical framework, i.e., isogenous pairing groups (IPG).

Admittedly, we cannot achieve a full quantum security when using bilinear pairings. Therefore, we first formalize a weak form of quantum security against pre-challenge quantum adversaries. Roughly speaking, the adversary in the model can attempt *quantum* attack before the challenge phase, but can execute only classical attacks after the challenge. Although this notion is weaker than full quantum security, we can present efficient KP-ABE constructions with pre-challenge quantum security. For example, our small-universe KP-ABE scheme has *asymptotically the same efficiency* as Goyal et al.'s simple classically secure KP-ABE. The security notion also has practical merits: In particular, we can protect a critical master secret key (as described above) by using a pre-challenge quantum secure IBE/ABE. This is because if an adversary obtains the master secret, then it can attack any challenge ciphertext just by executing *classical* decryption algorithm by using the master secret.

1.2 Our Results

We put forth a new mathematical framework called *Isogenous Pairing Groups (IPG)* and new intractable assumptions in the framework, the *Isogenous DBDH (Isog-DBDH) assumption* and its variants. As an application of IPG, we propose an ABE scheme secure against pre-challenge quantum adversaries, whose security is reduced from a newly established intractability assumption, i.e., a variant of Isog-DBDH assumption. We achieve the result step-by-step as described below. (For mathematical backgrounds on IPG, refer to Section A.)

- We establish a mathematical framework for enjoying both merits of pairings and isogenies, called Isogenous Pairing Groups (IPG). Moreover, we define new assumptions in the framework for proving the above security for our IBE and ABE schemes: the Isog-DBDH assumption and its variants. The assumptions seem to have more applications in future, and it is of independent interest. For the details, see Section 2.
- We formulate a new security definition against pre-challenge quantum adversaries (Section 3), which is based on the framework given in [49]. In the security game, an adversary consists of two parts, quantum and classical machines, i.e., $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$. The task of the first quantum machine $\mathcal{A}_1$ is to analyze the public key and pre-challenge information, and his/her result is passed to the second classical machine $\mathcal{A}_2$, whose task is the same as usual, i.e., to distinguish the challenge bit with asking auxiliary key queries.
- We present an anonymous IBE construction secure against pre-challenge quantum adversaries in the quantum random oracle model (QROM) in Section 4. Our IBE scheme is based on the Boneh-Franklin IBE (BF-IBE), and has an efficiency (practicality) comparable to the BF-IBE. One of the main differences is that a public master key has two elliptic curve parameters for using isogeny in the cryptographic construction. Since BF-IBE was adopted as an international standard [32], our IBE is quite practical with respect to the data sizes, encryption and decryption times. We can also extend our construction to HIBE based on the Gentry-Silverberg HIBE [27] (Appendix C).
- We construct small and large universe KP-ABE schemes secure against pre-challenge quantum adversaries, which are selective-attribute secure in the QROM (Section 5). First, we construct a small universe KP-ABE secure against pre-challenge quantum adversaries in the QROM, and then obtain the large universe construction by hierarchically combining two instantiations of the small universe ABE. The proposed KP-ABE schemes are based on the GPSW06 KP-ABE [29]. We note that all sizes of public parameters, secret keys and ciphertxts of our small universe KP-ABE is *asymptotically the same* as GPSW06 small universe KP-ABE.

1.3 Key Techniques

New Mathematical Framework of Isogenous Pairing Groups (IPG) We first observe a similarity of scalar multiplications and isogenies on elliptic curves. We unify the two homomorphic operations as a notion of Trapdoor Homomorphism (TH), which is defined in Section 2.1. Informally, a randomly chosen TH cannot be computed efficiently (without a trapdoor), but, once having a trapdoor, the homomorphic computation turns out to be easy.

For most pairing cryptosystems, the bilinear property $e(g_0, \hat{g}_0^\alpha) = e(g_0, \hat{g}_0)^\alpha$ is a key point, which is considered as a compatibility condition on pairing with (public key) $\hat{g}_0^\alpha$ and scalar multiplication with (secret key) α. Based on the above similarity, for our IBE and ABE, a compatibility of pairing and isogeny, e.g., $e_0(g_0, \hat{g}_0) = e_1(\phi(g_0), \hat{g}_1)$, is required. Note that since we use multiple elliptic curves, pairings e_0 and e_1 are defined on different E_0 and $E_1 := \phi(E_0)$, respectively. Based on the compatibility, we formulate a notion of Isogenous Pairing Groups, an extension of that of pairing groups. In the system, multiple pairing groups of the same prime order are employed, where efficient homomorphisms between them are hidden from adversaries. It is schematically presented in Fig. 1 in Section 2.2.

(H)IBE and ABE on IPG By using the above similarity, we replace a part of master key pair of BF-IBE, $(\mathsf{pk} := (\hat{g}_0, \hat{g}_0^\alpha),\ \mathsf{sk} := \alpha)$ for a group element $\hat{g}_0 \in \hat{\mathbb{G}}_0$ and a random scalar α, by $(\mathsf{pk} := (\hat{g}_0, \phi(\hat{g}_0)),\ \mathsf{sk} := \phi))$ for a randomly chosen isogeny ϕ. The important difference of scalar multiplications and isogenies is that the former security is assured only against classical adversaries but the latter security is assured even against quantum adversaries (at the present knowledge). The difference leads to security against pre-challenge quantum adversaries.

Moreover, for achieving the security, we should not include two (or more) different elements in one same group in the public key. Otherwise, the quantum adversary reveals the secret exponent (discrete log) relating the two elements. The public key condition restricts our ciphertext construction. For example, our small universe KP-ABE has a simple ciphertext as $\mathsf{ct}_{\mathsf{tag},\,\Gamma} := (\{\hat{g}_t^\zeta\}_{t\in\Gamma}, e_0(H(\mathsf{tag}), \hat{g}_0)^\zeta)$ with a uniformly random $\zeta \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$ (and hash H) for tag tag and attributes Γ. With this strong restriction, we obtain provably secure small and large universe KP-ABE schemes against pre-challenge quantum adversaries.

New Assumptions on IPG: Isog-DBDH and Variants We formulate new assumptions for our proposals, namely, Isog-DBDH assumption and its extension, d-qIsog-DBDH. The usual DBDH instance consists of $(g, g^\alpha, g^\beta, g^\gamma, h_T)$ for distinguishing whether h_T is $g_T^{\alpha\beta\gamma}$ or random where $g_T := e(g, g)$. An instance for the Isog-DBDH problem consists of $(g, g^\alpha, \phi(g), \phi(g)^\beta, h_T)$ for distinguishing whether h_T is $g_T^{\alpha\beta}$ or random, where $g \in E_0$ and $\phi(g) \in E_1$ are encoded on different (isogenous) elliptic curves E_0 and E_1, respectively. Therefore, informally, an adversary without knowing ϕ cannot obtain a meaningful pairing value $g_T^{\alpha\beta}$ from g^α and $\phi(g)^\beta$. Our IBE (resp., small, large universe ABE) scheme is secure under the 1-qIsog-DBDH (resp., d-qIsog-DBDH, $2dn$-qIsog-DBDH) assumption, where d is small universe size and n is bitlength of an attribute.

1.4 Related Works

Previous IBEs have been constructed by three types of mathematical problems: pairing, factoring, and lattice (See Table 1). While lattice-based IBE is considered to be quantum-secure, factoring-based IBEs are never converted to be quantum-secure. Our proposal is placed in the middle of the two situations. By using isogenies, we can achieve security against pre-challenge quantum adversaries. For the comparison, see Table 1.

Most of lattice-based IBE and ABE schemes are believed to be quantum secure, but recently, Biasse and Song [7] demonstrated an effective quantum attack for some ideal based cryptosystems. While Ducas et al. [24] improved a lattice-based IBE for practical use, lattice-based ABE schemes [28, 11] are inefficient far from practical use. Some efficient pairing-based ABEs are implemented in the publicly available software, e.g., Charm [3], and our isogeny-based technique can

Table 1. Comparison with existing IBE schemes, where LWE stands for Learning With Errors and the right most column represents the type of the adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$ in the security definition as is given in Def. 10, in which c (resp., q) means "classical ppt (resp., quantum polynomial-time) machine".

	Math. Primitive	Assumption	Sec. Model	Type of $(\mathcal{A}_1, \mathcal{A}_2)$
BF01 [10]	pairing	DBDH	ROM	(c, c)
Wat09 [45]		DLIN	adaptive	(c, c)
Coc01 [18]	factoring	Quad. Residuosity	ROM	(c, c)
BGH07 [12]			ROM	(c, c)
GPV08 [26]	lattice	LWE	ROM	(q, q)
ABB10 [2]			selective	(q, q)
Proposed	pairing & isogeny	1-qIsog-DBDH	QROM	(q, c)

Table 2. Comparison with existing large universe KP-ABE schemes, where d, n are size parameters of the large universe for our KP-ABE in Section 5.3 and the type of $(\mathcal{A}_1, \mathcal{A}_2)$ column represents the same as in Table 1.

	Math. Primitive	Assumption	Sec. Model	Type of $(\mathcal{A}_1, \mathcal{A}_2)$
GPSW06 [29]	pairing	DBDH	selective	(c, c)
OT10 [35]		DLIN	adaptive	(c, c)
GVW13 [28]	lattice	LWE	selective	(q, q)
BGG$^+$14 [11]			selective	(q, q)
Proposed	pairing & isogeny	$2dn$-qIsog-DBDH	selective in QROM	(q, c)

be applied to the legacy IBE/ABE cryptosystems with comparable efficiency. For the comparison of our ABE and previous ones, see Table 2.

For the security proofs based on the simulation, the rewinding techniques have been commonly used. In the quantum setting, the naive usage of the rewinding techniques does not work for the proofs. Watrous [46] devised a technique to affirmatively resolve the rewinding problem. His technique has contributed to provide security proof of classical cryptographic protocols against quantum adversaries [4, 42, 30, 43]. Showing security proofs against quantum adversaries that can invoke *oracles* is another important issue. Since quantum adversaries can make quantum-superposition queries to the oracle, simulators (for security proofs) have to respond with the corresponding quantum-superposition answers. For this problem, Damgård et al. [21] gave a positive solution in the case of secret sharing. Especially, the quantum superposition attacks in the ROM is problematic. Boneh et al. [9] first considered a ROM security definition against quantum computers. Based on the definition, Zhandry proposed a construction of secure identity-based encryption in the quantum random oracle model (QROM) [49], then quantum random functions in the model [48]. Moreover, the authors proposed quantum-safe MAC, signatures and CCA-secure encryption constructions [13, 14]. For other cryptographic techniques, the security against quantum adversaries has been discussed [20, 41].

1.5 Notations

When A is a random variable or distribution, $y \overset{\mathsf{R}}{\leftarrow} A$ denotes that y is randomly selected from A according to its distribution. When A is a set, $y \overset{\mathsf{U}}{\leftarrow} A$ denotes that y is uniformly selected from A. We denote the finite field of order q by $\mathbb{F}_q$. Let $[n] := \{1, .., n\}$ and $[0, n] := \{0\} \cup [n] := \{0, .., n\}$ for any $n \in \mathbb{Z}_{>0}$. For two vectors $\vec{y} = (y_i)_{i\in[r]}$ and $\vec{v} = (v_i)_{i\in[r]}$, $\vec{y} \cdot \vec{v}$ denotes the inner-product $\sum_{i=1}^{r} y_i v_i$.

2 Isogenous Pairing Groups

For mathematical backgrounds on IPG, refer to Section A.

2.1 Trapdoor Homomorphisms

Definition 1 (Trapdoor Homomorphism (TH)). *A (randomly chosen) function $\phi := \phi_\xi : G_0 \to G_1$ with two (randomly chosen) cyclic groups G_0, G_1 of a prime order q is called a* trapdoor homomorphism *if the following conditions hold.*

- *(Homomorphism) ϕ is a non-trivial (e.g., non-zero for an additive group) homomorphism.*
- *(TH-DH (TH-Diffie-Hellman) intractability assumption) Any probabilistic polynomial-time (ppt) machine $\mathcal{B}$ computes $\phi(g)$ only with a negligible probability when given $(g_0, \phi(g_0), g)$ for randomly chosen ϕ and $g_0, g \overset{\mathsf{U}}{\leftarrow} G_0$.*
- *(Polynomial-size trapdoor) There exists a ppt machine $\mathcal{B}$ which computes $\phi(g)$ for any $g \in G_0$ given a polynomial-size trapdoor ξ for $\phi := \phi_\xi$.*

Examples. By using elliptic curves, we have three examples of THs.

1. (Exponentiation) $G_0 := G_1 := \mathbb{G}$ is an elliptic curve cyclic group and $\phi := \phi_\xi$ is an exponentiation on $\mathbb{G}$ (i.e., scalar multiplication on the curve), i.e., $\phi_\xi : g \mapsto g^\xi$, where ξ is a scalar. TH-DH input and output are given as $(g_0, \phi(g_0), g) = (g_0, g_0^\xi, g)$ and $\phi(g) = g^\xi$, respectively, and then TH-DH intractability is the same as the usual computational DH assumption.
2. (Pairing) $G_0 := \mathbb{G}, G_1 := \mathbb{G}_T$ is a pairing group and $\phi := \phi_\xi$ is a pairing operation on $\mathbb{G}$, i.e., $\phi_\xi : g \mapsto e(g, \xi)$, where ξ is an element in $\mathbb{G}$ in the symmetric pairing case (or $\hat{\mathbb{G}}$ in the asymmetric pairing case). TH-DH input and output are given as $(g_0, \phi(g_0), g) = (g_0, e(g_0, \xi), g)$ and $\phi(g) = e(g, \xi)$, respectively, and then TH-DH intractability is reduced to the computational BDH (CBDH) assumption (Lemma 1).
3. (Isogeny) $G_0 := \mathbb{G}_0, G_1 := \mathbb{G}_1$ are two different elliptic curve cyclic groups obtained from two curves E, E', respectively, and $\phi := \phi_\xi$ is an isogeny from $\mathbb{G}_0$ to $\mathbb{G}_1$, i.e., $\phi_\xi : E \to E' := E/C$, where $\xi := C$ is a (cyclic) subgroup in E. For the details, see Section A.3, in particular, Algorithms 1 and 2. The TH-DH intractability is another kind of natural extensions of the DH assumption obtained by using isogenies other than that given in [22].

Remark 1 (Trapdoor Extraction Intractability). Trapdoor Extraction Intractability assumption is formulated as follows: Any ppt machine $\mathcal{B}$ computes a trapdoor ξ only with a negligible probability given a $(g_0, \phi(g_0))$ for $g_0 \overset{\mathsf{U}}{\leftarrow} G_0$. As is easily seen, the trapdoor extraction is intractable if the TH-DH is intractable.

For the exponentiation case, it is the Discrete Logarithm Problem (DLP) assumption, and for the pairing case, it is the Pairing Inversion (PI) assumption. For the isogeny case, it is called the Isogeny assumption. (Refer to Def. 5.)

Lemma 1. *The TH-DH assumption for pairing is reduced to the CBDH assumption. That is, for any adversary $\mathcal{B}$ for TH-DH, there exists a probabilistic machine $\mathcal{C}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{TH\text{-}DH}}(\lambda) = \mathsf{Adv}_{\mathcal{C}}^{\mathsf{CBDH}}(\lambda)$, where $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{TH\text{-}DH}}$ (resp., $\mathsf{Adv}_{\mathcal{C}}^{\mathsf{CBDH}}$) is an advantage of $\mathcal{B}$ for the TH-DH (resp., $\mathcal{C}$ for the CBDH) problem.*

Proof. A ppt machine $\mathcal{C}$ is given a CBDH instance $(g, g^\alpha, g^\beta, g^\gamma)$ where $\alpha, \beta, \gamma \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$. $\mathcal{C}$ sends $(g, e(g^\alpha, g^\beta) = e(g, g^{\alpha\beta}), g^\gamma)$ to a TH-DH adversary $\mathcal{B}$, where implicitly setting $\xi := g^{\alpha\beta}$. If $\mathcal{B}$ outputs $e(g^\gamma, g^{\alpha\beta}) = e(g, g)^{\alpha\beta\gamma}$, then $\mathcal{C}$ outputs it. □

By combining the TH-DH intractability assumptions for isogeny and pairing, we have the following definition.

Definition 2 (Isog-CBDH). *The Isog-CBDH assumption is as follows: For two (randomly chosen) cyclic symmetric pairing groups $\mathbb{G}_0, \mathbb{G}_1$ (in two different elliptic curves) and a (randomly chosen) isogeny $\phi : \mathbb{G}_0 \to \mathbb{G}_1$, any ppt machine $\mathcal{B}$ computes $h_T := e_1(\phi(g), \phi(h))$ only with a negligible probability given $(g_0, \phi(g_0), g, \phi(h))$ for uniformly generated $g_0, g, h \overset{\mathsf{U}}{\leftarrow} \mathbb{G}_0$.*

Lemma 2. *The TH-DH assumption for isogeny is reduced to the Isog-CBDH assumption. That is, for any adversary $\mathcal{B}$ for TH-DH, there exists a ppt machine $\mathcal{C}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security param.* λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{TH\text{-}DH}}(\lambda) = \mathsf{Adv}_{\mathcal{C}}^{\mathsf{Isog\text{-}CBDH}}(\lambda)$, *where* $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{TH\text{-}DH}}$ *(resp.,* $\mathsf{Adv}_{\mathcal{C}}^{\mathsf{Isog\text{-}CBDH}}$*) is an advantage of $\mathcal{B}$ for the TH-DH (resp., $\mathcal{C}$ for the Isog-CBDH) problem.*

Proof. A ppt machine $\mathcal{C}$ is given an Isog-BDH instance $(g_0, \phi(g_0), g, \phi(h))$ where $g_0, g, h \overset{\mathsf{U}}{\longleftarrow} \mathbb{G}_0$. $\mathcal{C}$ sends $(g_0, \phi(g_0), g)$ to TH-DH adversary $\mathcal{B}$. If $\mathcal{B}$ outputs $\phi(g)$, then $\mathcal{C}$ outputs $e_1(\phi(g), \phi(h))$. □

The decisional version of the Isog-CBDH assumption is given below.

Definition 3 (Isog-DBDH). *The Isog-DBDH assumption is as follows: For two (randomly chosen) cyclic symmetric pairing groups $\mathbb{G}_0, \mathbb{G}_1$ (in two different elliptic curves) and a (randomly chosen) isogeny $\phi : \mathbb{G}_0 \to \mathbb{G}_1$, any ppt machine $\mathcal{B}$ guesses whether $h_T = e_1(\phi(g), \phi(h))$ or random in $\mathbb{G}_T$ only with a negligible probability given $(g_0, \phi(g_0), g, \phi(h), h_T)$ for $g_0, g, h \overset{\mathsf{U}}{\longleftarrow} \mathbb{G}_0$ and $h_T \in \mathbb{G}_T$.*

2.2 Isogenous Pairing Groups (IPG)

Combining the three trapdoor homomorphic structures, we propose a useful cryptographic framework called "Isogenous Pairing Groups (IPG)". After establishing the framework, we define a natural and useful computational assumption on IPG called "Isog-DBDH", which is a generalization of that in Def. 3. When applying IPG framework and Isog-DBDH to crypto constructions, "compatibility" of pairings on different groups (or elliptic curves) and isogenies is a main ingredient. For the symmetric pairing groups $\mathbb{G}_0, \mathbb{G}_1$ in Def. 3, it is described as

$$e_0(g, h) = e_1(\phi(g), \phi(h)), \tag{1}$$

where e_0 (resp., e_1) is an efficiently computable pairing on $\mathbb{G}_0$ (resp., $\mathbb{G}_1$), i.e., on the curve E_0 (resp., E_1). For the correctness, see Prop. 2 in Section A.4. The above property (1) is extended to among multiple curves $\{E_t\}_{t\in[0,d]}$ or multiple asymmetric pairing groups $(\mathbb{G}_t, \hat{\mathbb{G}}_t)_{t\in[0,d]}$ as is given in Eq. (2) below.

Definition 4 (Isogenous Pairing Groups (IPG)). *Isogenous Pairing Groups (IPG) generator generates a random instance as follows:*

$$\mathsf{Gen}^{\mathsf{IPG}}(1^\lambda, d) \xrightarrow{\mathsf{R}} (\,\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t\in[0,d]}, \mathbb{G}_T),\ \mathsf{sk}^{\mathsf{IPG}} := (\phi_t)_{t\in[d]}\,),$$

where $(\mathbb{G}_t, \hat{\mathbb{G}}_t, e_t, \mathbb{G}_T)$ are asymmetric pairing groups of a prime order q with pairings $e_t : \mathbb{G}_t \times \hat{\mathbb{G}}_t \to \mathbb{G}_T$, $\hat{g}_t \in \hat{\mathbb{G}}_t$, trapdoor homomorphisms $\phi_t : \mathbb{G}_0 \to \mathbb{G}_t$ (given by isogenies between different elliptic curves), and $g_t = \phi_t(g_0) \in \mathbb{G}_t$. The isogenous pairing groups satisfy

$$\textbf{Compatibility}: \quad e_0(g_0, \hat{g}_0) = e_t(g_t, \hat{g}_t) = e_t(\phi_t(g_0), \hat{g}_t) \text{ for any } t \in [d]. \tag{2}$$

$$\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t\in[0,d]}, \mathbb{G}_T), \quad \mathsf{sk}^{\mathsf{IPG}} := (\phi_t)_{t\in[d]}$$

$$\begin{tikzcd}
\mathbb{G}_0 \times \hat{\mathbb{G}}_0 \arrow[rr, "e_0"] \arrow[dr, "\phi_1"] \arrow[ddr, "\phi_d"'] & \circlearrowleft & \mathbb{G}_T \\
& \mathbb{G}_1 \times \hat{\mathbb{G}}_1 \arrow[ur, "e_1"'] \arrow[d, dash, dotted] & \\
& \mathbb{G}_d \times \hat{\mathbb{G}}_d \arrow[uur, "e_d"'] &
\end{tikzcd}$$

$$\begin{aligned}
&e_0(g_0, \hat{g}_0) = \\
&e_1(g_1, \hat{g}_1) = \\
&\quad e_1(\phi_1(g_0), \hat{g}_1) = \\
&\quad \vdots \\
&e_d(g_d, \hat{g}_d) = \\
&\quad e_d(\phi_d(g_0), \hat{g}_d) \\
&\quad = g_T \in \mathbb{G}_T
\end{aligned}$$

Fig. 1. Compatibility of IPG

We denote the common non-trivial pairing value by g_T, i.e., $g_T = e_0(g_0, \hat{g}_0) \neq 1$. See Fig. 1. Moreover, we require that $\mathbb{G}_t \neq \hat{\mathbb{G}}_t$. (Namely, points in $\mathbb{G}_t$ and $\hat{\mathbb{G}}_t$ generate the group of q-torsion points on the t-th elliptic curve.)

For simulation in security proof, the simulation algorithm $\mathsf{SimGen}^{\mathsf{IPG}}(\mathbb{G}_0, \hat{\mathbb{G}}_0, g_0, \hat{g}_0, e_0)$ outputs a newly isogenous group $(\mathbb{G}, \hat{\mathbb{G}}, g, \hat{g}, e, \phi)$ such that $\phi : \mathbb{G}_0 \to \mathbb{G}$ is an efficiently computable group isomorphism, $g = \phi(g_0)$, and the compatibility holds, i.e., $e_0(g_0, \hat{g}_0) = e(\phi(g_0), \hat{g}) = e(g, \hat{g})$.

Remark 2. A concrete instantiation of IPG by supersingular elliptic curves is given in Appendix B.1. The above compatibility is also assured by Prop. 2 in Sec. A.4.

1. An isogeny ϕ_t above is calculated by using Algorithm 1 or 2 in Section A.3. A trapdoor ξ for Algorithm 1 (resp., 2) is given by a torsion point R on the elliptic curve (resp., a walk data $\omega \in \{0,1\}^{\kappa}$). An isogeny ϕ_t in a secret key $\mathsf{sk}^{\mathsf{IPG}}$ is given by the above trapdoor ξ_t depending on Algorithm 1 or 2 for efficient computing of ϕ_t.
2. We note that each isogeny ϕ_t is defined from the elliptic curve E_0 to E_t, hence, ϕ_t is defined from the rank two torsion group $\mathbb{G}_0 \oplus \hat{\mathbb{G}}_0$ to $\mathbb{G}_t \oplus \hat{\mathbb{G}}_t$ such that $\mathbb{G}_t = \phi_t(\mathbb{G}_0)$ and $\hat{\mathbb{G}}_t = \phi_t(\hat{\mathbb{G}}_0)$. However, since ϕ_t is enough to be defined only on $\mathbb{G}_0$ for our schemes in Sections 4 and 5, ϕ_t is defined only on $\mathbb{G}_0$ in the present definition, Def. 4 for IPG.

Assumptions The most basic security requirement is given by the intractability of the simple isogeny computation problem: calculate (trapdoors of) any non-zero $(\phi_t)_{t\in[d]}$ s.t. $\phi_t : \mathbb{G}_0 \to \mathbb{G}_t$ when given $(\mathbb{G}_t)_{t\in[0,d]}$. The most basic case is given when $d = 1$: calculate ϕ_1 when given $\mathbb{G}_0$ and $\mathbb{G}_1$. Based on the fundamental assumption, we define useful assumptions on IPG for cryptographic use below. The isogeny problem on IPG is similarly given such that a ppt machine should answer the (master) secret key when given the (master) public key from $\mathsf{Gen}^{\mathsf{IPG}}$.

Definition 5 (Isogeny Problem on IPG (for $d = 1$)). *Let* $(\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t=0,1}, \mathbb{G}_T),\ \mathsf{sk}^{\mathsf{IPG}} := \phi_1) \xleftarrow{\mathsf{R}} \mathsf{Gen}^{\mathsf{IPG}}(1^\lambda, 1)$.
If a ppt adversary $\mathcal{B}$ *outputs* ϕ_1 *when given* $\mathsf{pk}^{\mathsf{IPG}}$*,* $\mathcal{B}$ *wins.*

While the problem instance has auxiliary inputs, points on groups, that is, $(g_t \in \mathbb{G}_t, \hat{g}_t \in \hat{\mathbb{G}}_t)_{t=0,1}$, it is believed to have *no* efficient quantum adversary at present. We use the intractability for the *pre-challenge quantum security*.

In Def. 15 in Section A.3, De Feo et al.'s assumption, Computational Supersingular Isogeny (CSSI) assumption (for smooth order pairing groups) by using a prime p with $p + 1 = \ell_A^{\kappa_A} \ell_B^{\kappa_B} \cdot f$ and isogeny generation $\mathrm{Isog}_{\ell,\kappa}^{\mathsf{djp}}$ (Algorithm 1) with $\ell := \ell_A, \kappa := \kappa_A$, is given as a special instantiation of the isogeny assumption given in Def. 5. The CSSI problem is believed to have *no* efficient quantum attack, which gives a reasonable justification for our isogeny assumption on IPG.

For provable security of our schemes, we define new problems, Isog-DBDH problem on IPG. The adversary against the Isog-DBDH problem has two parts, the first is quantum and the second is classical. We first formulate a classical adversary form of our Isog-DBDH, for simplicity (for $d = 1$).

Definition 6 (Isog-DBDH Assumption (on IPG)). *Let* $\mathcal{B}$ *be a classical ppt machine adversary. For* $(\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t=0,1}, \mathbb{G}_T),\ \mathsf{sk}^{\mathsf{IPG}} := \phi_1) \xleftarrow{\mathsf{R}} \mathsf{Gen}^{\mathsf{IPG}}(1^\lambda, 1)$ *and* $\alpha, \beta, \delta \xleftarrow{\mathsf{U}} \mathbb{F}_q$*,* $\mathcal{B}$ *receives* $\mathcal{X}_\mathfrak{b}$ *for* $\mathfrak{b} \xleftarrow{\mathsf{U}} \{0, 1\}$*, that is defined by*

$$\mathcal{X}_0 := (\ \mathsf{pk}^{\mathsf{IPG}},\ g_0^{\alpha},\ \hat{g}_1^{\beta},\ g_T^{\alpha\beta}\ ) \quad \text{and} \quad \mathcal{X}_1 := (\ \mathsf{pk}^{\mathsf{IPG}},\ g_0^{\alpha},\ \hat{g}_1^{\beta},\ g_T^{\delta}\ ), \tag{3}$$

where $g_T := e_0(g_0, \hat{g}_0)$*.* $\mathcal{B}$ *outputs a guess bit* $\mathfrak{b}'$*. If* $\mathfrak{b} = \mathfrak{b}'$*,* $\mathcal{B}$ *wins. The Isog-DBDH assumption is: For any ppt adversary* $\mathcal{B}$*, the advantage of* $\mathcal{B}$ *for the Isog-DBDH problem is negligible in* λ*.*

In other words, $\mathcal{X}_0$ and $\mathcal{X}_1$ contain four group elements $(g_0,\ \hat{g}_1,\ g_0^{\alpha},\ \hat{g}_1^{\beta})$ and the problem asks whether the target $\mathbb{G}_T$ element is $g_T^{\alpha\beta}$ or random. Here, we note that two elements in $\mathbb{G}_0$ and $\hat{\mathbb{G}}_1$ cannot be paired, in particular, $\mathcal{B}$ cannot obtain the pairing value $g_T^{\alpha\beta}$ from g_0^{α} and $\hat{g}_1^{\beta} = \phi_1(\hat{g}_0)^{\beta}$ (by Item 2 of Remark 2). Compared with the usual DBDH instance, one scalar multiplication is replaced by another trapdoor homomorphism ϕ_1 in the Isog-DBDH instance.

Security of our schemes is based on the next assumptions, qIsog-DBDH and d-qIsog-DBDH, where an adversary $\mathcal{B}$ is given by two machines $(\mathcal{B}_1, \mathcal{B}_2)$, where $\mathcal{B}_1$ is modeled as a polynomial-time quantum adversary, $\mathcal{B}_2$ a classical ppt machine.

Definition 7 (qIsog-DBDH Assumption (on IPG)). *Let* $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ *be an adversary, where* $\mathcal{B}_1$ *is modeled as a polynomial-time quantum adversary,* $\mathcal{B}_2$ *a classical ppt machine. For* $(\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t=0,1}, \mathbb{G}_T), \mathsf{sk}^{\mathsf{IPG}} := \phi_1) \xleftarrow{\mathsf{R}} \mathsf{Gen}^{\mathsf{IPG}}(1^\lambda, 1)$*,* $\mathcal{B}_1$ *outputs* $\mathsf{state} \xleftarrow{\mathsf{R}} \mathcal{B}_1(\mathsf{pk}^{\mathsf{IPG}})$*. Then, for* $\alpha, \beta, \delta \xleftarrow{\mathsf{U}} \mathbb{F}_q$*,* $\mathcal{B}_2$ *receives* $\mathcal{X}_\mathfrak{b}$ *for* $\mathfrak{b} \xleftarrow{\mathsf{U}} \{0, 1\}$*, that is defined by Eq. (3).* $\mathcal{B}_2$ *outputs a guess bit* $\mathfrak{b}'$*. If* $\mathfrak{b} = \mathfrak{b}'$*,* $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ *wins. The qIsog-DBDH assumption is defined in a similar manner as in Def. 6.*

For a general positive integer $d > 0$, d-qIsog-DBDH problem is given as follows:

Definition 8 (d-qIsog-DBDH Assumption (on IPG)). *Let $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ be an adversary, where $\mathcal{B}_1$ is modeled as a polynomial-time quantum adversary, $\mathcal{B}_2$ a classical ppt machine. For $(\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t \in [0,d]}, \mathbb{G}_T)$, $\mathsf{sk}^{\mathsf{IPG}} := (\phi_t)_{t \in [d]}) \xleftarrow{\mathsf{R}} \mathsf{Gen}^{\mathsf{IPG}}(1^\lambda, d)$, $\mathcal{B}_1$ outputs $\mathsf{state} \xleftarrow{\mathsf{R}} \mathcal{B}_1(\mathsf{pk}^{\mathsf{IPG}})$. Then, for $\alpha, \beta, \delta \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\mathcal{B}_2$ receives $\mathcal{X}_{\mathfrak{b}}$ for $\mathfrak{b} \xleftarrow{\mathsf{U}} \{0,1\}$, that is defined by*

$$\mathcal{X}_0 := (\ \mathsf{state},\ g_0^{\alpha},\ (\hat{g}_t^{\beta})_{t \in [d]},\ g_T^{\alpha\beta}\ ) \quad \text{and} \quad \mathcal{X}_1 := (\ \mathsf{state},\ g_0^{\alpha},\ (\hat{g}_t^{\beta})_{t \in [d]},\ g_T^{\delta}\ ),$$

where $g_T := e_0(g_0, \hat{g}_0)$. $\mathcal{B}_2$ outputs a guess bit $\mathfrak{b}'$. If $\mathfrak{b} = \mathfrak{b}'$, $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ wins. The advantage of adversary $\mathcal{B}$ in the experiment is defined as $\mathsf{Adv}_{\mathcal{B}}^{d\text{-}\mathsf{qIsog}\text{-}\mathsf{DBDH}}(\lambda) :=$ $\Pr[\mathcal{B}$ wins$] - 1/2$ for any security parameter λ. The d-qIsog-DBDH assumption is: For any ppt $\mathcal{B}$, the advantage of $\mathcal{B}$, $\mathsf{Adv}_{\mathcal{B}}^{d\text{-}\mathsf{qIsog}\text{-}\mathsf{DBDH}}(\lambda)$, is negligible in λ.

A concrete instantiation of the d-qIsog-DBDH assumption by supersingular elliptic curves is given in Appendix B.2.

3 Definitions of IBE and ABE Secure Against Pre-challenge Quantum Adversaries

3.1 Intuition

As is described in Section 1.4, there already exist several security models against quantum adversary. For achieving efficient (or practical) quantum-secure IBE and ABE, we define a new, intermediate notion between classical and (full-)quantum security ones, i.e., security against *pre-challenge quantum adversary*. In the definitions (of IBE and ABE), an adversary $\mathcal{A}$ consists of two machines $(\mathcal{A}_1, \mathcal{A}_2)$, the first $\mathcal{A}_1$ is modeled by a quantum machine for the pre-challenge phase and the second $\mathcal{A}_2$ a classical machine for the post-challenge phase. (That is, $\mathcal{A}_1$ is stronger than $\mathcal{A}_2$, which is analogous to the concept of non-adaptive chosen ciphertext attack (CCA1) security [6] in literatures.) In the IBE security game, $\mathcal{A}_1$ is given a target master public key pk, makes random oracle and key generation queries (with some restrictions) and outputs some state information denoted by state. After obtaining state, $\mathcal{A}_2$ makes a challenge encryption query, obtains a challenge ciphertext and then makes ID-secret key queries (with other restrictions than the pre-challenge phase) and $\mathcal{A}_2$'s task is guessing the random bit b (encoded on the ciphertext) as usual. As is easily seen, if both $(\mathcal{A}_1, \mathcal{A}_2)$ are classical (resp. quantum), the notion is classical (resp. quantum). Therefore, our security notion is an intermediate one of the two notions.

One example scenario is as follows: Emergence of quantum computers leads to security breaches. However, in the initial phase of development of quantum computers, like classical computer emergence, the speed of deployment must be relatively slow. At least, handy quantum computers are not so spreading and the machine cost is expensive. IBE (resp. KP-ABE) have a two-level hierarchical structure in secret keys, i.e., master secret keys and ID (resp. policy) secret keys. If the master secret key is revealed, the adversary can generate any ID (resp. policy) secret key by using a real key generation algorithm, which is a classical ppt algorithm. Therefore, the adversary first targets the master secret key for an effective attack. On the other hand, if the master secret key is protected, the attacker should break each ciphertext one-by-one. A quantum adversary can attack each ciphertext, but, if he/she can attack the master secret key at the first stage of the attack, the strategy is considered to be effective. From this point, the defense of the master key from a quantum adversary is important and our security notion called PH-PQ (Payload-Hiding against Pre-challenge Quantum adversaries) in Definitions 10 and 14 is motivated by this observation.

3.2 Quantum Computation

A quantum algorithm is an algorithm executed on a quantum computer that produces a classical output. Quantum algorithms can keep and operate quantum states in the registers, where the quantum states can be represented as a linear combination of distinct states. A classical output that a quantum algorithm produces is one of the distinct states that represent the final quantum state. Refer the reader to [34] for a more thorough discussion.

Here we remind a few basic facts about quantum computation necessary for understanding our results in a manner similar to [49].

Fact 1 Any classical computation can be implemented on a quantum computer.

Fact 2 Any function that has an efficient classical algorithm computing it can be implemented efficiently as a quantum-accessible oracle.

Fact 3 Given a quantum algorithm $\mathcal{A}$ with oracle access to an oracle O, each oracle O defines a probability distribution of the outputs of $\mathcal{A}$. Hence, any probability amplitude D of oracles leads to a probability distribution of outputs of $\mathcal{A}$, and if two probability amplitudes D_1 and D_2 are identical, the probability distributions of the outputs of $\mathcal{A}$ under these amplitudes are also identical.

3.3 Identity-Based Encryption (IBE)

Definition 9 (Identity-Based Encryption: IBE). *An identity-based encryption scheme consists of probabilistic polynomial-time algorithms* $\mathsf{Setup}, \mathsf{KeyGen}$, Enc *and* Dec. *They are given as follows:*

Setup *takes as input a security parameter* 1^λ. *It outputs public parameters* pk *and master secret key* sk.

KeyGen *takes as input public parameters* pk*, a master secret key* sk*, and an identity* ID. *It outputs a corresponding secret key* $\mathsf{sk}_{\mathsf{ID}}$.

Enc *takes as input public parameters* pk*, a message* m *in some associated message space* msg*, and an identity* ID. *It outputs a ciphertext* $\mathsf{ct}_{\mathsf{ID}}$.

Dec *takes as input public parameters* pk*, a secret key* $\mathsf{sk}_{\mathsf{ID}}$ *for an identity* ID*, and a ciphertext* $\mathsf{ct}_{\mathsf{ID}'}$ *that was encrypted under an identity* ID'. *It outputs either* $m' \in \mathsf{msg}$ *or the distinguished symbol* $\perp$.

An IBE scheme should have the following correctness property: for all $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^\lambda)$, all identities ID, all secret keys $\mathsf{sk}_{\mathsf{ID}} \xleftarrow{\mathsf{R}} \mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \mathsf{ID})$, all messages m, all identities ID', all ciphertexts $\mathsf{ct}_{\mathsf{ID}'} \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk}, m, \mathsf{ID}')$, it holds that $m = \mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_{\mathsf{ID}}, \mathsf{ct}_{\mathsf{ID}'})$ if $\mathsf{ID} = \mathsf{ID}'$. Otherwise, it holds with negligible probability.

We define the security notion of Payload-Hiding against Pre-challenge Quantum adversary (PH-PQ) for IBE.

Definition 10 (PH-PQ for IBE). *Let* $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *be an IBE scheme and let* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ *be a stateful adversary, where* $\mathcal{A}_1$ *is modeled as a polynomial-time quantum adversary. Consider the experiment* $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{ibe,ph\text{-}pq}}[\lambda]$ *below:*

$$\begin{aligned}
&\mathbf{Exp}_{\mathcal{A}}^{\mathsf{ibe,ph\text{-}pq}}[\lambda] : \ (\mathsf{sk}, \mathsf{pk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^\lambda), \quad \mathsf{state} \xleftarrow{\mathsf{R}} \mathcal{A}_1^{\mathsf{RO}(\cdot),\, \mathsf{KeyGen}(\mathsf{sk},\cdot)}(\mathsf{pk}),\\
&\quad / * \ \mathsf{ID}^* \text{ is not queried to } \mathsf{RO} \text{ in the pre-challenge phase by } \mathcal{A}_1,\\
&\qquad\qquad \text{and not queried to } \mathsf{KeyGen} \text{ in any phase } * /\\
&\quad (\mathsf{ID}^*, m_0, m_1) \xleftarrow{\mathsf{R}} \mathcal{A}_2^{\mathsf{RO}(\cdot),\, \mathsf{KeyGen}(\mathsf{sk},\cdot)}(\mathsf{state}),\ b \xleftarrow{\mathsf{U}} \{0,1\},\ \mathsf{ct}^* \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk}, m_b, \mathsf{ID}^*),\\
&\quad b' \xleftarrow{\mathsf{R}} \mathcal{A}_2^{\mathsf{RO}(\cdot),\, \mathsf{KeyGen}(\mathsf{sk},\cdot)}(\mathsf{ct}^*), \quad \text{output } b'.
\end{aligned}$$

Here, RO *is quantum-accessible (i.e., with quantum superposed inputs and outputs) and* KeyGen *is classical-accessible. If* $b = b'$*,* $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$ *wins. The advantage of adversary* $\mathcal{A}$ *in the experiment is defined as* $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ibe,ph\text{-}pq}}(\lambda) := \Pr[\mathcal{A} \text{ wins}] - 1/2$ *for any security parameter* λ*. An IBE scheme is* payload-hiding against pre-challenge quantum adversary (PH-PQ) *if all adversaries* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$*, where* $\mathcal{A}_1$ *be a polynomial-time quantum machine and* $\mathcal{A}_2$ *a classical ppt machine, achieve at most a negligible advantage in the above security game (or experiment).*

The security notion of anonymous-ID secure against pre-challenge quantum adversary for IBE in Definition 17 in Appendix D.

3.4 Key-Policy Attribute-Based Encryption (KP-ABE)

For a polynomial $d = d(\lambda)$, a sub-universe $\mathcal{U}_t$ $(\subset \{0,1\}^*)$ is assigned for $t \in [d]$. Each attribute is expressed by a pair of sub-universe id and value of attribute, i.e., (t, v), where $t \in [d]$ and $v \in \mathcal{U}_t$. Let $\mathcal{U}_t := \{1\}$ for the small universe case and $\mathcal{U}_t := \{0,1\}^n$ for the large universe case with a polynomial $n := n(\lambda)$. Thus, the small universe $[d] \times \{1\}$ is identified with $[d]$ and the large universe is given by $[d] \times \{0,1\}^n$.

Span Programs and (Monotone) Access Structures

Definition 11 (Span Programs [5]). *A span program over* $\mathbb{F}_q$ *is a labeled matrix* $\mathbb{S} := (M, \rho)$ *where* M *is an* $(l \times r)$ *matrix over* $\mathbb{F}_q$ *and* ρ *is a labeling of the rows of* M *by an attribute from* $\{(t, v), (t', v'), \ldots\}$ *(every row is labeled by one attribute), i.e.,* $\rho : \{1, \ldots, l\} \to \{(t, v), (t', v'), \ldots\}$*. A span program accepts or rejects an input by the following criterion. Let* Γ *be a set of attributes, i.e.,*

$\Gamma := \{(t_j, x_j)\}_{1 \le j \le d'}$ ($x_j \in \mathcal{U}_{t_j}$). The span program $\mathbb{S}$ accepts Γ, denoted by $R(\mathbb{S}, \Gamma) = 1$, if and only if $\vec{1} \in \mathsf{span}\langle (M_i)_{\rho(i) \in \Gamma} \rangle$, i.e., some linear combination of the rows $(M_i)_{\rho(i) \in \Gamma}$ gives the all one vector $\vec{1}$.

No row M_i (for $i \in [l]$) of the matrix M is zero.

Definition 12. *A secret-sharing scheme for span program $\mathbb{S} := (M, \rho)$ is:*

1. *Let M be an $l \times r$ matrix and a column vector $\vec{f} := (f_1, \ldots, f_r) \xleftarrow{\mathsf{U}} \mathbb{F}_q^r$. Then, $s_0 := \vec{1} \cdot \vec{f} = \sum_{k=1}^{r} f_k$ is the secret to be shared, and $\vec{s} := (s_1, \ldots, s_l)^{\mathrm{T}} := M \cdot \vec{f}^{\mathrm{T}}$ is the l shares of the secret s_0 and the share s_i belongs to $\rho(i)$.*
2. *If span program $\mathbb{S} := (M, \rho)$ accepts Γ, i.e., $\vec{1} \in \mathsf{span}\langle (M_i)_{\rho(i) \in \Gamma} \rangle$, there exist constants $\{\sigma_i \in \mathbb{F}_q \mid i \in I\}$ such that $I \subseteq \{i \in [l] \mid \rho(i) \in \Gamma\}$ and $\sum_{i \in I} \sigma_i s_i = s_0$. Furthermore, these constants $\{\sigma_i\}$ can be computed in time polynomial in the size of the matrix M.*

KP-ABE In KP-ABE, encryption (resp. a secret key) is associated with attributes Γ (resp. access structure $\mathbb{S}$). In fact, our KP-ABEs are tagged schemes, and the relation R^+ for the tagged KP-ABE is defined as $R^+((\mathsf{tag}, \mathbb{S}), (\mathsf{tag}', \Gamma)) := Eq(\mathsf{tag}, \mathsf{tag}') \wedge R(\mathbb{S}, \Gamma)$ for a key parameter $(\mathsf{tag}, \mathbb{S})$ and ciphertext parameter (tag', Γ), where $R(\mathbb{S}, \Gamma) = 1$ iff $\mathbb{S}$ accepts Γ and $Eq(\mathsf{tag}, \mathsf{tag}') = 1$ iff $\mathsf{tag} = \mathsf{tag}'$.

Definition 13 (Key-Policy Attribute-Based Encryption: KP-ABE). *A key-policy attribute-based encryption scheme consists of probabilistic polynomial-time algorithms* Setup, KeyGen, Enc *and* Dec*. They are given as follows:*

Setup *takes as input a security parameter 1^λ. It outputs public parameters* pk *and master secret key* sk*.*

KeyGen *takes as input a public parameters* pk*, a master secret key* sk*, a tag* tag*, and an access structure $\mathbb{S} := (M, \rho)$. It outputs a corresponding secret key $\mathsf{sk}_{\mathsf{tag}, \mathbb{S}}$.*

Enc *takes as input a public parameters* pk*, a message m in some associated message space* msg*, a tag* tag′*, and a set of attributes, $\Gamma := \{(t_j, x_j)\}_{1 \le j \le d'}$. It outputs a ciphertext $\mathsf{ct}_{\mathsf{tag}', \Gamma}$.*

Dec *takes as input a public parameters* pk*, a secret key $\mathsf{sk}_{\mathsf{tag}, \mathbb{S}}$ for access structure $\mathbb{S}$ and tag* tag*, and a ciphertext $\mathsf{ct}_{\mathsf{tag}', \Gamma}$ that was encrypted under a set of attributes Γ and tag* tag′*. It outputs either $m' \in \mathsf{msg}$ or the symbol $\perp$.*

A KP-ABE scheme should have the following correctness property: for all $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^\lambda)$, all tags tag, all access structures $\mathbb{S}$, all secret keys $\mathsf{sk}_{\mathsf{tag}, \mathbb{S}} \xleftarrow{\mathsf{R}} \mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \mathsf{tag}, \mathbb{S})$, all messages m, all tags tag′, all attribute sets Γ, all ciphertexts $\mathsf{ct}_{\mathsf{tag}', \Gamma} \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk}, m, \mathsf{tag}', \Gamma)$, it holds that $m = \mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_{\mathsf{tag}, \mathbb{S}}, \mathsf{ct}_{\mathsf{tag}', \Gamma})$ if $\mathsf{tag} = \mathsf{tag}'$ and $\mathbb{S}$ accepts Γ. Otherwise, it holds with negligible probability.

Definition 14 (PH-PQ for KP-ABE). *Let* (Setup, KeyGen, Enc, Dec) *be a tagged key-policy attribute based encryption scheme and let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be a* *stateful adversary, where $\mathcal{A}_1$ is modeled as a polynomial-time quantum adversary. Consider the experiment* $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{abe,ph\text{-}pq}}[\lambda]$ *below:*

$$
\begin{aligned}
&\mathbf{Exp}_{\mathcal{A}}^{\mathsf{abe,ph\text{-}pq}}[\lambda]: \ \Gamma^* \xleftarrow{\mathsf{R}} \mathcal{A}_1(1^\lambda), \quad (\mathsf{sk},\mathsf{pk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^\lambda),\\
&\quad (\mathsf{state},\mathsf{tag}^*) \xleftarrow{\mathsf{R}} \mathcal{A}_1^{\mathsf{RO}(\cdot),\,\mathsf{KeyGen}(\mathsf{sk},\cdot)}(\mathsf{pk}),\\
&\quad\ /*\ \ \mathsf{tag}^* \text{ is not queried to } \mathsf{RO} \text{ nor } \mathsf{KeyGen} \text{ in the pre-challenge phase by } \mathcal{A}_1 */\\
&\quad (m_0,m_1) \xleftarrow{\mathsf{R}} \mathcal{A}_2^{\mathsf{RO}(\cdot),\,\mathsf{KeyGen}(\mathsf{sk},\cdot)}(\mathsf{state}),\ b \xleftarrow{\mathsf{U}} \{0,1\},\ \mathsf{ct}^* \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk},m_b,\mathsf{tag}^*,\Gamma^*),\\
&\quad b' \xleftarrow{\mathsf{R}} \mathcal{A}_2^{\mathsf{RO}(\cdot),\,\mathsf{KeyGen}(\mathsf{sk},\cdot)}(\mathsf{ct}^*), \quad \text{output } b'.\\
&\quad\ /* \text{ if } (\mathsf{tag}^*,\mathbb{S}) \text{ is queried to } \mathsf{KeyGen} \text{ in the post-challenge phase by } \mathcal{A}_2,\\
&\qquad\qquad \mathbb{S} \text{ does not accept } \Gamma^* */
\end{aligned}
$$

Here, RO *is quantum-accessible (i.e., with quantum superposed inputs and outputs) and* KeyGen *is classical-accessible. If $b = b'$, $\mathcal{A} := (\mathcal{A}_1,\mathcal{A}_2)$ wins. The advantage of adversary $\mathcal{A}$ in the experiment is defined as* $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{abe,ph\text{-}pq}}(\lambda) := \Pr[\mathcal{A} \text{ wins}] - 1/2$ *for any security parameter λ. A tagged KP-ABE scheme is (selective-attribute)* payload-hiding against pre-challenge quantum adversaries (PH-PQ) *if all adversaries $\mathcal{A} = (\mathcal{A}_1,\mathcal{A}_2)$, where $\mathcal{A}_1$ be a polynomial-time quantum machine and $\mathcal{A}_2$ a classical ppt machine, achieve at most a negligible advantage in the above security game (or experiment).*

4 Proposed Anonymous IBE against Pre-challenge Quantum Adversaries

4.1 Construction

The proposed IBE scheme is a Boneh-Franklin type IBE. A master secret key sk is given by an isogeny from $\mathbb{G}_0$ to $\mathbb{G}_1$. A hash function $H : \mathbb{F}_q \to \mathbb{G}_0$ maps an arbitrary $\mathsf{ID} \in \mathbb{F}_q$ to a point of $\mathbb{G}_0$. Note that the size of $\mathbb{F}_q$ is exponential in λ.

$$
\begin{aligned}
&\mathsf{Setup}(1^\lambda): (\ \mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t,\hat{\mathbb{G}}_t,g_t,\hat{g}_t,e_t)_{t=0,1},\mathbb{G}_T),\ \mathsf{sk}^{\mathsf{IPG}} := \phi_1\ ) \xleftarrow{\mathsf{R}} \mathsf{Gen}^{\mathsf{IPG}}(1^\lambda,1),\\
&\quad \text{generate a random hash } H : \mathbb{F}_q \to \mathbb{G}_0 \text{ with the identity space } \mathbb{F}_q,\\
&\quad \text{return } \ \mathsf{pk} := ((\mathbb{G}_t,\hat{\mathbb{G}}_t,\hat{g}_t,e_t)_{t=0,1},\mathbb{G}_T,\ H), \quad \mathsf{sk} := \phi_1.\\
&\mathsf{KeyGen}(\mathsf{pk},\mathsf{sk},\mathsf{ID}): \ h_0 := H(\mathsf{ID}) \in \mathbb{G}_0,\ h_1 := \phi_1(h_0), \ \text{ return } \mathsf{sk}_{\mathsf{ID}} := h_1.\\
&\mathsf{Enc}(\mathsf{pk},m,\mathsf{ID}): \ h_0 := H(\mathsf{ID}), \ \ \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}, \ \ c := \hat{g}_1^{\zeta}, \ \ z := e_0(h_0,\hat{g}_0)^{\zeta}, \ \ c_T := z \cdot m,\\
&\quad \text{return } \ \mathsf{ct}_{\mathsf{ID}} := (c,c_T).\\
&\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{\mathsf{ID}} = h_1,\ \mathsf{ct}_{\mathsf{ID}'} = (c,c_T)): \ \text{if } \mathsf{ID} = \mathsf{ID}',\ z' := e_1(h_1,c),\\
&\quad m' := c_T \cdot (z')^{-1}, \text{ return } m', \ \text{ otherwise, return } \perp.
\end{aligned}
$$

Dec correctly decrypts since $z' = e_1(h_1,c) = e_1(\phi_1(h_0),\hat{g}_1^{\zeta}) = e_1(\phi_1(h_0),\hat{g}_1)^{\zeta} = e_0(h_0,\hat{g}_0)^{\zeta} = z$ if $\mathsf{ID} = \mathsf{ID}'$. Here, we use the compatibility $e_1(\phi_1(h_0),\hat{g}_1) = e_0(h_0,\hat{g}_0)$ in Definition 4.

Note that a secret key $\mathsf{sk}_{\mathsf{ID}}$ (resp., ciphertext $\mathsf{ct}_{\mathsf{ID}}$) consists of one element of $\mathbb{G}_0$ (resp., one $\hat{\mathbb{G}}_1$ element and one $\mathbb{G}_T$ element), which are almost the same as those in the original BF-IBE. However, size of public parameters is double of that in BF-IBE, i.e., two elliptic curve parameters. This shows that our IBE is quite practical. A concrete instantiation of our IBE by supersingular elliptic curves is given in Appendix B.3.

4.2 Security

The following proposition assures the correctness of a simulation strategy of the random oracle H by using a random degree $2\nu_1$ polynomial for a quantum adversary which makes quantum random oracle queries at most ν_1 times. Cf. Fact 3 in Section 3.2.

Proposition 1. *[49] Let H be an oracle drawn from a $2\nu_1$-wise independent distribution. Then, the behavior of any quantum algorithm making at most ν_1 quantum queries to H is identical to the behavior of the quantum algorithm making at most ν_1 quantum queries to a truly random function.*

Remark 3 (Lemma 6.4 in [13]). Furthermore, we can show that for an oracle H drawn from a $(2\nu_1 + \nu_2)$-wise independent distribution, the behavior of any quantum algorithm making at most ν_1 quantum and ν_2 classical queries to H is identical to the behavior of the quantum algorithm making at most ν_1 quantum and ν_2 classical queries to a truly random function.

Theorem 1. *The proposed IBE scheme is PH-PQ secure under the 1-qIsog-DBDH assumption in the quantum random oracle model.*

For any adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, there exists an adversary $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ for the 1-qIsog-DBDH problem, where $\mathcal{B}_1$ is a polynomial-time quantum machine and $\mathcal{B}_2$ is a classical ppt machine, such that for any security parameter λ, $\mathsf{Adv}^{\mathsf{ibe,ph\text{-}pq}}_{\mathcal{A}}(\lambda) \leq \nu_2 \cdot \mathsf{Adv}^{\mathsf{1\text{-}qIsog\text{-}DBDH}}_{\mathcal{B}}(\lambda)$, where ν_2 is the maximum number of the random oracle queries of $\mathcal{A}_2$.

Our IBE can be shown to be anonymous ID secure against pre-challenge quantum adversaries. (The security notion is defined in Appendix D.)

Theorem 2. *The proposed IBE scheme is anonymous ID secure against pre-challenge quantum adversaries under the 1-qIsog-DBDH assumption in the quantum random oracle model.*

Theorem 2 is proven in a manner similar to Lemma 4.3 in [1] (given in Appendix D). Anonymous IBE has an application to searchable encryption as is well known. In the application, if the master secret key is revealed by a quantum attacker, all the trapdoor keys for encrypted search can be generated by a classical attacker. Our pre-challenge quantum security prevents such a devastating attack scenario (see Section 3.1).

Proof of Theorem 1. We will employ two different simulation strategies for two phases, the pre-challenge phase for the quantum $\mathcal{A}_1$ and the post-challenge phase for the classical $\mathcal{A}_2$. Note that the target ID^* is never queried to the random oracle nor key generation queries in the pre-challenge phase. Hence, the simulation in the phase is simpler than that given in [49]. Let ν_1 (resp., ν_2) the maximum number of random oracle queries in the pre-challenge phase (resp., in the post-challenge phase), and $\nu := 2\nu_1 + \nu_2$. We can simulate the random oracle (and then key generation) by using a random degree ν polynomial $F(X)$ since a ν-wise independent function is enough for oracle simulation for $\mathcal{A}_1$ and $\mathcal{A}_2$, which is obtained from Remark 3.

We will simulate the environment of the second part of the adversary, $\mathcal{A}_2$, in a similar manner to the indistinguishability proof of the Boneh-Franklin IBE [10].

In order to prove Theorem 1, we construct a probabilistic machine $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ against the 1-qIsog-DBDH Problem using an adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$ in a security game as a black box as follows:

1. **Pre-challenge phase simulation for the quantum machine $\mathcal{A}_1$:** $\mathcal{B}_1$ is given a public parameter for the 1-qIsog-DBDH problem, $\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t=0,1}, \mathbb{G}_T)$, then, $\mathcal{B}_1$ provides the quantum adversary $\mathcal{A}_1$ the public key $\mathsf{pk} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, \hat{g}_t, e_t)_{t=0,1}, \mathbb{G}_T)$ and answers to quantum random oracle access for H and classical key queries. Let $F(X)$ be a random degree ν polynomial, i.e., $F(X) \overset{\mathsf{U}}{\leftarrow} \oplus_{i=0}^{\nu} \mathbb{F}_q X^i$ with $\nu := 2\nu_1 + \nu_2$. A quantum superposition random oracle query RO (for H) is answered by the superposition of $g_0^{\tau_{\mathsf{ID}}}$ using g_0 in $\mathsf{pk}^{\mathsf{IPG}}$ and $\tau_{\mathsf{ID}} := F(\mathsf{ID})$. A classical (not superposed) key generation query for ID is answered by $g_1^{\tau_{\mathsf{ID}}}$ using g_1 in $\mathsf{pk}^{\mathsf{IPG}}$ and $\tau_{\mathsf{ID}} := F(\mathsf{ID})$.
2. When $\mathcal{A}_1$ outputs $\mathsf{state} \overset{\mathsf{R}}{\leftarrow} \mathcal{A}_1(\mathsf{pk})$ and sends state to $\mathcal{A}_2$, $\mathcal{B}_1$ obtains state and outputs $\mathsf{state}' := (\mathsf{state}, F(X))$ (and then gives state' to $\mathcal{B}_2$).
3. **Post-challenge phase simulation for the classical ppt machine $\mathcal{A}_2$:** $\mathcal{B}_2$ is given $\mathcal{X}_{\mathfrak{b}} := (\ \mathsf{state}',\ g_0^{\alpha},\ \hat{g}_1^{\beta},\ g_T^{\theta}\ )$, where $\theta = \alpha\beta$ if $\mathfrak{b} = 0$ and $\theta \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$ if $\mathfrak{b} = 1$ and $\mathsf{state}' := (\mathsf{state}, F(X))$. $\mathcal{B}_2$ plays a role of the challenger in the security game against adversary $\mathcal{A}_2$, and $\mathcal{B}_2$ sends state to $\mathcal{A}_2$. $\mathcal{B}_2$ chooses a random ν_0 ($1 \leq \nu_0 \leq \nu_2$) for embedding the problem instance to classical random oracle answers, where ν_2 is the maximum number of random oracle queries of classical $\mathcal{A}_2$.
4. When the ι-th random oracle query is issued for an identity ID,
   (a) if $\iota \neq \nu_0$, $\mathcal{B}_2$ generates $\tau_{\mathsf{ID}} := F(\mathsf{ID})$ and calculates $h_{\mathsf{ID}} := g_0^{\tau_{\mathsf{ID}}}$, then returns the value h_{ID} to $\mathcal{A}_2$.
   (b) if $\iota = \nu_0$, $\mathcal{B}_2$ obtains the hash value $h_{\mathsf{ID}_{\nu_0}} := h_0 := g_0^{\alpha}$ from $\mathcal{X}_{\mathfrak{b}}$, i.e., set $\tau_{\mathsf{ID}_{\nu_0}} := \alpha$ implicitly, then returns the value $h_{\mathsf{ID}_{\nu_0}}$ to $\mathcal{A}_2$.
5. When a key query is issued for identity ID, $\mathcal{B}_2$ generates $\tau_{\mathsf{ID}} := F(\mathsf{ID})$ and calculates $\mathsf{sk}_{\mathsf{ID}} := g_1^{\tau_{\mathsf{ID}}}$, then returns the value $\mathsf{sk}_{\mathsf{ID}}$ to $\mathcal{A}_2$.
6. When $\mathcal{B}_2$ receives an encryption query with challenge identity ID^* and plaintexts $(m^{(0)}, m^{(1)})$ from $\mathcal{A}_2$, if the current number ν^* of the RO queries in the the post-challenge phase is greater than or equal to ν_0, first $\mathcal{B}_2$ checks whether $\mathsf{ID}^* = \mathsf{ID}_{\nu_0}$ or not, where ID_{ν_0} is the ν_0-th queried identity to the

RO. If it does not hold, $\mathcal{B}_2$ aborts the game. If $\nu^* < \nu_0$ or ($\nu^* \geq \nu_0$ and $\mathsf{ID}^* = \mathsf{ID}_{\nu_0}$), $\mathcal{B}_2$ selects (challenge) bit $b \xleftarrow{\mathsf{U}} \{0,1\}$. $\mathcal{B}_2$ generates the challenge ciphertext $\mathsf{ct}_{\mathsf{ID}^*} := (c, c_T)$ such that $c := \hat{g}_1^{\beta}$ is obtained from the input $\mathcal{X}_{\mathfrak{b}}$ (which implicitly sets $\zeta := \beta$) and $c_T := g_T^{\theta} \cdot m_b$ where g_T^{θ} is also obtained from the input $\mathcal{X}_{\mathfrak{b}}$. $\mathcal{B}_2$ returns $\mathsf{ct}_{\mathsf{ID}^*}$ to $\mathcal{A}_2$.

7. When the ι-th random oracle query for identity ID is issued by $\mathcal{A}_2$ after the encryption query, $\mathcal{B}_2$ executes the same procedure as that of step 4a if $\iota \neq \nu_0$. When $\iota = \nu_0$, if $\mathsf{ID} = \mathsf{ID}^*$, $\mathcal{B}_2$ executes the same procedure as that of step 4b, otherwise, $\mathcal{B}_2$ aborts the game.
   When a key query is issued for identity ID, $\mathcal{B}_2$ generates and returns $\mathsf{sk}_{\mathsf{ID}}$ to $\mathcal{A}_2$ as in the same way as in step 5.
8. $\mathcal{A}_2$ finally outputs bit b'. If $b = b'$, $\mathcal{B}_2$ outputs $\mathfrak{b}' := 0$. Otherwise, $\mathcal{B}_2$ outputs $\mathfrak{b}' := 1$.

When $\mathfrak{b} = 0$ (resp. $\mathfrak{b} = 1$), the view of $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$ is equivalent to that in the real game (resp., the random ciphertext game). Moreover, the advantage of $\mathcal{A}$ in the latter is equal to zero since the value of b is independent from the adversary's view in the game. We obtain the inequality in Theorem 1, and this completes the proof of Theorem 1. □

5 Proposed PH-PQ Secure KP-ABE

5.1 Small Universe Construction

The proposed KP-ABE scheme is based on the GPSW06 KP-ABE [29]. By identifying $(t, 1) \in [d] \times \{1\} = [d] \times \mathcal{U}$ with $t \in [d]$, an attribute is considered as an element of the polynomial-size universe $[d]$, i.e., attribute set $\Gamma \subset [d]$. The IPG with $(d + 1)$ pairing groups is used.

Key Idea in Constructing the Proposed KP-ABE In GPSW06 KP-ABE, the public parameters (resp., master secret key) are given as $\mathsf{pk} := ((\hat{\mathbb{G}}, \hat{g}_t := \hat{g}^{\alpha_t})_{t \in [d]}, \mathbb{G}_T, g_T^{y})$ (resp., $\mathsf{sk} := ((\alpha_t)_{t \in [d]}, y)$), where $\hat{g}_t := \hat{g}^{\alpha_t}$ are group elements in the *same* group $\hat{\mathbb{G}}$ and $\alpha_t, y \xleftarrow{\mathsf{U}} \mathbb{F}_q$. The exponentiation-based pk is vulnerable to quantum attack. Instead, we encode these group elements on *different* groups (i.e., different elliptic curves) such as $(\hat{g}_t \in \hat{\mathbb{G}}_t)_{t \in [d]}$, where $\hat{g}_t \in \hat{\mathbb{G}}_t$ are defined as $\hat{g}_t = \phi_t(\hat{g}_0) \in \hat{\mathbb{G}}_t = \phi_t(\hat{\mathbb{G}}_0)$ using master secret isogenies ϕ_t. Informally, the public parameters are invulnerable to quantum attack from the quantum hardness of the isogeny problem (Def. 5). (The security of the proposed scheme is formally proved under the d-qIsog-DBDH assumption in Theorem 3.) From the compatibility of IPG (Eq. (2)), we decrypt ciphertexts correctly.

For achieving pre-challenge quantum security, we should not include two (or more) different elements in one same group in the public parameters. Otherwise, the quantum adversary reveals the secret exponent (discrete log) relating the two elements. It restricts our ciphertext construction. For example, our small universe

KP-ABE has a simple ciphertext as $\mathsf{ct}_{\mathsf{tag},\varGamma} := (\{\hat{g}_t^{\zeta}\}_{t\in\varGamma}, e_0(H(\mathsf{tag}),\hat{g}_0)^{\zeta})$ with a uniformly random $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$ (and hash H).

Note that pk includes $d+1$ pairing groups $(\mathbb{G}_t, \hat{\mathbb{G}}_t)_{t\in[0,d]}$ as well as d elements $(\hat{g}_t)_{t\in[d]}$ and components of secret key $\mathsf{sk}_{\mathsf{tag},\mathbb{S}}$ and ciphertext $\mathsf{ct}_{\mathsf{tag},\varGamma}$ have similar structures as the original GPSW06 KP-ABE. Therefore, all sizes of public parameters, secret keys and ciphertxts of our KP-ABE are *asymptotically the same* as GPSW06 KP-ABE. A concrete instantiation of our KP-ABE by supersingular elliptic curves is given in Appendix B.4.

Construction

```
$\mathsf{Setup}(1^\lambda)$ :
    $(\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t\in[0,d]}, \mathbb{G}_T),\ \mathsf{sk}^{\mathsf{IPG}} := (\phi_t)_{t\in[d]}) \xleftarrow{\mathsf{R}} \mathsf{Gen}^{\mathsf{IPG}}(1^\lambda, d)$,
    generate a random hash $H : \mathbb{F}_q \to \mathbb{G}_0$ with the tag space $\mathbb{F}_q$,
    return $\mathsf{pk} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, \hat{g}_t, e_t)_{t\in[0,d]}, \mathbb{G}_T,\ H)$,   $\mathsf{sk} := (\phi_t)_{t\in[d]}$.
$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \mathsf{tag},\ \mathbb{S} := (M,\rho))$ :  $h_0 := H(\mathsf{tag}) \in \mathbb{G}_0$,
    choose random $\vec{u}$ such that $\vec{1}\cdot\vec{u} = 1$,
    for $i \in [l]$,  $s_i := M_i \cdot \vec{u}$,  $t := \rho(i)$,   $k_i := \phi_t(h_0)^{s_i}$,
    return $\mathsf{sk}_{\mathsf{tag},\mathbb{S}} := \{k_i\}_{i\in[l]}$.
$\mathsf{Enc}(\mathsf{pk},\ m,\ \mathsf{tag},\ \varGamma)$ :  $h_0 := H(\mathsf{tag}) \in \mathbb{G}_0$,  $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$,    for $t \in \varGamma$,  $c_t := \hat{g}_t^{\zeta}$,
    $z := e_0(h_0, \hat{g}_0)^{\zeta}$,  $c_T := z \cdot m$,  return  $\mathsf{ct}_{\mathsf{tag},\varGamma} := (\{c_t\}_{t\in\varGamma}, c_T)$.
$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{\mathsf{tag},\mathbb{S}} := \{k_i\}_{i\in[l]},\ \mathsf{ct}_{\mathsf{tag}',\varGamma} := (\{c_t\}_{t\in\varGamma}, c_T))$ :
   if $\mathsf{tag} = \mathsf{tag}'$ and $\mathbb{S} := (M,\rho)$ accepts $\varGamma := \{t\}$, then compute $\{\sigma_i\}_{\rho(i)\in\varGamma}$
     such that  $\vec{1} = \sum_{\rho(i)\in\varGamma} \sigma_i M_i$,  where $M_i$ is the $i$-th row of $M$,
   $z' := \prod_{t:=\rho(i)\in\varGamma} e_t(k_i, c_t)^{\sigma_i}$,   return  $m' := c/z'$,   otherwise,  return $\perp$.
```

[Correctness]: If $\mathsf{tag} = \mathsf{tag}'$ and $\mathbb{S}$ accepts $\varGamma$,

$$z' = \textstyle\prod_{t:=\rho(i)\in\varGamma} e_t(k_i, c_t)^{\sigma_i} = \prod_{t:=\rho(i)\in\varGamma} e_t\left(\phi_t(h_0)^{s_i}, \hat{g}_t^{\zeta}\right)^{\sigma_i}$$
$$= \textstyle\prod_{t:=\rho(i)\in\varGamma} e_t\left(\phi_t(h_0), \hat{g}_t\right)^{\zeta\sigma_i s_i} = \prod_{\rho(i)\in\varGamma} e_0(h_0, \hat{g}_0)^{\zeta\sigma_i s_i} = e(h_0, \hat{g}_0)^{\zeta} = z.$$

5.2 Security of Our Small Universe KP-ABE

Theorem 3. *The proposed KP-ABE scheme is (selective-attribute) PH-PQ secure under the d-qIsog-DBDH assumption in the quantum random oracle model.*

For any adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, there exists an adversary $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ for the d-qIsog-DBDH problem, where $\mathcal{B}_1$ is a polynomial-time quantum machine and $\mathcal{B}_2$ is a classical ppt machine, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{abe,ph\text{-}pq}}(\lambda) \le \mathsf{Adv}_{\mathcal{B}}^{d\text{-}\mathsf{qIsog\text{-}DBDH}}(\lambda)$.

Remark 4. We will employ two different simulation strategies for two phases, the pre-challenge phase for the quantum $\mathcal{A}_1$ and the post-challenge phase for the classical $\mathcal{A}_2$, in a similar manner to the case of the proposed IBE. In particular, we note that we can simulate the random oracle (and then key generation) by using a random degree ν polynomial $F(X)$ since a ν-wise independent function is enough for oracle simulation for $\mathcal{A}_1$, which is obtained from Remark 3, where ν_1 (resp., ν_2) is the maximum number of random oracle queries in the pre-challenge phase (resp., in the post-challenge phase) and $\nu := 2\nu_1 + \nu_2$. We make two remarks on why challenge Γ^* and tag^* should be declared beforehand.

- Our simulated public parameters given in Eq. (4) below have two parts depending on whether $t \in \Gamma^*$ or not. Therefore, challenge attributes Γ^* should be declared at the beginning of the game.
- Our access relation is given by the conjunctive combination of span program and tag matching, i.e., $R^+((\mathsf{tag}, \mathbb{S}), (\mathsf{tag}^*, \Gamma)) := Eq(\mathsf{tag}, \mathsf{tag}^*) \wedge R(\mathbb{S}, \Gamma)$. We cannot determine which part of the condition does not hold for a key query with a parameter $(\mathsf{tag}, \mathbb{S})$ before knowing the challenge tag tag^*. Therefore, challenge tag tag^* should be declared before the post-challenge phase simulation by $\mathcal{B}_2$.

Proof. In order to prove Theorem 3, we construct a probabilistic machine $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ against the d-qIsog-DBDH Problem using an adversary $\mathcal{A}$ in a security game as a black box as follows:

1. **Pre-challenge phase simulation for the quantum machine $\mathcal{A}_1$:** $\mathcal{B}_1$ is given a public parameter for the d-qIsog-DBDH problem, $\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t\in[0,d]}, \mathbb{G}_T)$, and $\mathcal{B}_1$ plays a role of the challenger in the security game against adversary $\mathcal{A}_1$.
2. $\mathcal{A}_1$ declares the challenge attributes Γ^*, then, $\mathcal{B}_1$ provides the quantum adversary $\mathcal{A}_1$ the public key pk which are generated as: Generate $(\mathbb{G}'_t, \hat{\mathbb{G}}'_t, g'_t, \hat{g}'_t, e'_t, \phi'_t) \leftarrow \mathsf{SimGen}^{\mathsf{IPG}}(\mathbb{G}_0, \hat{\mathbb{G}}_0, g_0, \hat{g}_0, e_0)$ for $t \notin \Gamma^*$ (and $t \in [d]$). Then, $\mathcal{B}_1$ provides the adversary $\mathcal{A}_1$
$$\mathsf{pk} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, \hat{g}_t, e_t)_{t\in\{0\}\cup\Gamma^*},\ (\mathbb{G}'_t, \hat{\mathbb{G}}'_t, \hat{g}'_t, e'_t)_{t\notin\Gamma^*},\ \mathbb{G}_T), \tag{4}$$
where $(\mathbb{G}_t, \hat{\mathbb{G}}_t, \hat{g}_t, e_t)_{t\in\{0\}\cup\Gamma^*}$ are obtained from $\mathsf{pk}^{\mathsf{IPG}}$ of the d-qIsog-DBDH instance, and sends it to $\mathcal{A}_1$.
3. $\mathcal{B}_1$ simulates quantum random oracle access for H in the pre-challenge phase. Let $F(X)$ be a random degree ν polynomial, i.e., $F(X) \xleftarrow{\mathsf{U}} \oplus_{i=0}^{\nu} \mathbb{F}_q X^i$ with $\nu := 2\nu_1 + \nu_2$. A quantum superposition random oracle query RO (for H) is answered by the superposition of $g_0^{\tau_{\mathsf{tag}}}$ using g_0 in $\mathsf{pk}^{\mathsf{IPG}}$ and $\tau_{\mathsf{tag}} := F(\mathsf{tag})$. A classical (not superposed) key generation query is answered as follows: $\mathcal{B}_1$ chooses random $\vec{u} \in \mathbb{F}_q^r$ such that $\vec{1} \cdot \vec{u} = 1$, and for $i \in [l]$, sets $s_i := M_i \cdot \vec{u}$, then returns $(k_i := g_{\rho(i)}^{\tau_{\mathsf{tag}} \cdot s_i})_{i\in[l]}$ to $\mathcal{A}_1$. Here, l (resp., r) is the row (resp., column) number of the access structure matrix M.
4. When $\mathcal{A}_1$ outputs $(\mathsf{state}, \mathsf{tag}^*) \xleftarrow{\mathsf{R}} \mathcal{A}_1(\mathsf{pk})$ and sends state to $\mathcal{A}_2$, $\mathcal{B}_1$ obtains state and outputs $\mathsf{state}' := (\mathsf{state}, \mathsf{tag}^*, F(X))$ (and then gives state' to $\mathcal{B}_2$).

5. **Post-challenge phase simulation for the classical ppt machine $\mathcal{A}_2$:** $\mathcal{B}_2$ is given $\mathcal{X}_{\mathfrak{b}} := (\ \mathsf{state}',\ g_0^{\alpha},\ (\hat{g}_t^{\beta})_{t\in[d]},\ g_T^{\theta}\ )$, where $\theta = \alpha\beta$ if $\mathfrak{b} = 0$ and $\theta \overset{\mathsf{U}}{\longleftarrow} \mathbb{F}_q$ if $\mathfrak{b} = 1$ and $\mathsf{state}' := (\mathsf{state}, \mathsf{tag}^*, F(X))$. $\mathcal{B}_2$ plays a role of the challenger in the security game against adversary $\mathcal{A}_2$, and $\mathcal{B}_2$ sends state to $\mathcal{A}_2$.
6. When a random oracle query is issued for a tag tag,
   (a) if $\mathsf{tag} \neq \mathsf{tag}^*$, $\mathcal{B}_2$ generates $\tau_{\mathsf{tag}} := F(\mathsf{tag})$ and calculates $h_{\mathsf{tag}} := h_0 := g_0^{\tau_{\mathsf{tag}}}$, then returns the value h_{tag} to $\mathcal{A}_2$.
   (b) if $\mathsf{tag} = \mathsf{tag}^*$, $\mathcal{B}_2$ obtains the hash value $h_{\mathsf{tag}^*} := h_0 := g_0^{\alpha}$ from $\mathcal{X}_{\mathfrak{b}}$, set $\tau_{\mathsf{tag}^*} := \alpha$ implicitly, then returns the value h_{tag^*} to $\mathcal{A}_2$.
7. When a key query is issued for a tag tag and access structure $\mathbb{S} := (M, \rho)$, first $\mathcal{B}_2$ checks whether $\mathsf{tag} = \mathsf{tag}^*$ or not, where tag^* is the challenge tag declared by $\mathcal{A}_1$.
   (a) If it does not hold, $\mathcal{B}_2$ executes the same procedure of KeyGen as that of step 3.
   (b) If it holds, $\mathcal{B}_2$ generates a vector $\vec{v} \overset{\mathsf{U}}{\longleftarrow} \mathbb{F}_q^r$. Pick $\vec{w} \overset{\mathsf{U}}{\longleftarrow} \{\vec{w} \in \mathbb{F}_q^r \,|\, \vec{w} \cdot M_i = 0 \text{ if } \rho(i) \in \Gamma^*,\ \mu := \vec{w} \cdot \vec{1} \neq 0\}$. Finally, define the vector $\vec{u}\,' := \vec{v} + \chi\vec{w}$, where $\chi := \frac{\alpha - \vec{v}\cdot\vec{1}}{\mu}$, and α is defined in the instance $\mathcal{X}_{\mathfrak{b}}$, then $\vec{u}\,' \cdot \vec{1} = \alpha$. Thus, let $\vec{u} := \vec{u}\,'/\alpha$ and $s_i := M_i \cdot \vec{u}$, then $\vec{u} \cdot \vec{1} = 1$ and $M_i \cdot \vec{u}\,' = \alpha s_i$. For $i \in [l]$, $\mathcal{B}_2$ calculates

$$k_i := \begin{cases} g_t^{\eta_{1,i}} & \text{if } \rho(i) \in \Gamma^*, \\ \phi_t'\left((g_0^{\alpha})^{\eta_{2,i}} \cdot g_0^{\eta_{3,i}}\right) & \text{if } \rho(i) \notin \Gamma^*. \end{cases}$$

   where $\eta_{1,i} := M_i \cdot \vec{v}$, $\eta_{2,i} := \frac{M_i \cdot \vec{w}}{\mu}$, $\eta_{3,i} := M_i \cdot \vec{v} - \frac{(M_i\cdot\vec{w})\cdot(\vec{v}\cdot\vec{1})}{\mu}$. The $\{k_i\}$ is a legitimate key since
   if $\rho(i) \in \Gamma^*$, $M_i \cdot \vec{u}\,' = M_i \cdot (\vec{v} + \chi\vec{w}) = M_i \cdot \vec{v} = \eta_{1,i}$, then $k_i = g_t^{\eta_{1,i}} = g_t^{M_i\cdot\vec{u}\,'} = g_t^{\alpha s_i} = \phi_t(h_{\mathsf{tag}^*})^{s_i}$, and
   if $\rho(i) \notin \Gamma^*$, $M_i \cdot \vec{u}\,' = M_i \cdot (\vec{v} + \chi\vec{w}) = M_i \cdot \vec{v} + \frac{\alpha - \vec{v}\cdot\vec{1}}{\mu}(M_i \cdot \vec{w}) = \alpha\frac{M_i\cdot\vec{w}}{\mu} + M_i\cdot\vec{v} - \frac{(M_i\cdot\vec{w})\cdot(\vec{v}\cdot\vec{1})}{\mu} = \alpha\eta_{2,i} + \eta_{3,i}$, then $k_i = \phi_t'\left((g_0^{\alpha})^{\eta_{2,i}} \cdot g_0^{\eta_{3,i}}\right) = \phi_t'\left(g_0^{\alpha\eta_{2,i}+\eta_{3,i}}\right) = (g_t')^{\alpha\eta_{2,i}+\eta_{3,i}} = (g_t')^{M_i\cdot\vec{u}\,'} = (g_t')^{\alpha s_i} = \phi_t'(h_{\mathsf{tag}^*})^{s_i}$.
   Then $\mathcal{B}_2$ returns the value $\mathsf{sk}_{\mathsf{tag}^*,\mathbb{S}} := \{k_i\}_{i\in[l]}$ to $\mathcal{A}_2$.
8. When $\mathcal{B}_2$ receives an encryption query with plaintexts $(m^{(0)}, m^{(1)})$ (and challenge the tag tag^*, attributes Γ^*) from $\mathcal{A}_2$, first $\mathcal{B}_2$ selects (challenge) bit $b \overset{\mathsf{U}}{\longleftarrow} \{0,1\}$. $\mathcal{B}_2$ generates the challenge ciphertext $\mathsf{ct}_{\mathsf{tag}^*,\Gamma^*} := ((c_t)_{t\in\Gamma^*}, c_T)$ such that $(c_t := \hat{g}_t^{\beta})_{t\in\Gamma^*}$ are obtained from the input $\mathcal{X}_{\mathfrak{b}}$ and $c_T := g_T^{\theta} \cdot m_b$ where g_T^{θ} is also obtained from the input $\mathcal{X}_{\mathfrak{b}}$. $\mathcal{B}_2$ returns $\mathsf{ct}_{\mathsf{tag}^*,\Gamma^*}$ to $\mathcal{A}_2$.
9. When a random oracle (resp. key) query is issued by $\mathcal{A}_2$ after the encryption query, $\mathcal{B}_2$ executes the same procedure as that of step 6 (resp. step 7).
10. $\mathcal{A}_2$ finally outputs bit b'. If $b = b'$, $\mathcal{B}_2$ outputs $\mathfrak{b}' := 0$. Otherwise, $\mathcal{B}_2$ outputs $\mathfrak{b}' := 1$.

When $\mathfrak{b} = 0$ (resp. $\mathfrak{b} = 1$), the view of $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$ is equivalent to that in in the real game (resp., the random ciphertext game). Moreover, the advantage of $\mathcal{A}$ in the latter is equal to zero since the value of b is independent from the adversary's view in the game. This completes the proof of Theorem 3. □

5.3 Large Universe Construction

An attribute $x_t := (x_{t,j})_{j\in[n]}$ for any sub-universe id t is an element in $\mathcal{U} := \{0,1\}^n$, and our construction has a hierarchical structure for $t \in [d]$ and $j \in [n]$ with two instantiations of the small universe ABE. In the low level instantiation, a special form of n-out-of-$2n$ secret sharing predicate is used for identity-matching for the length-n binary identities x_t. The IPG with $(2dn+1)$ pairing groups is used.

Goyal et al. [29] also present a large universe KP-ABE scheme (with no quantum security). The scheme encodes each attribute using a degree-d polynomial. For that, several *base* group elements are included in public parameters. Apparently, the polynomial evaluations in exponents during encryption need multiplication of the base elements. Therefore, the base group elements cannot be encoded on different groups if the polynomial encoding is employed. Then we should avoid the polynomial encoding for achieving quantum resistance. Instead, for n-bit attribute, we include n groups (elliptic curves) in public parameters and encode the j-th bit using the j-th group for $j \in [n]$. Therefore, sizes of public parameters, secret keys and ciphertexts of our scheme are n times of those of the original GPSW large universe KP-ABE scheme.

```
$\mathsf{Setup}(1^\lambda)$ :
    $\Big(\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_0, \hat{\mathbb{G}}_0, g_0, \hat{g}_0, e_0), (\mathbb{G}_{t,j,\iota}, \hat{\mathbb{G}}_{t,j,\iota}, g_{t,j,\iota}, \hat{g}_{t,j,\iota}, e_{t,j,\iota})_{\iota\in[0,1]}^{t\in[d],j\in[n]}, \mathbb{G}_T),$
        $\mathsf{sk}^{\mathsf{IPG}} := (\phi_{t,j,\iota})_{\iota\in[0,1]}^{t\in[d],j\in[n]}\Big) \xleftarrow{\mathsf{R}} \mathsf{Gen}^{\mathsf{IPG}}(1^\lambda, 2dn),$
    generate a random hash $H : \mathbb{F}_q \to \mathbb{G}_0$ with the tag space $\mathbb{F}_q$,
    $\mathsf{pk} := (((\mathbb{G}_0, \hat{\mathbb{G}}_0, \hat{g}_0, e_0), (\mathbb{G}_{t,j,\iota}, \hat{\mathbb{G}}_{t,j,\iota}, \hat{g}_{t,j,\iota}, e_{t,j,\iota})_{\iota\in[0,1]}^{t\in[d],j\in[n]}, \mathbb{G}_T,\ H),$
    return $\mathsf{pk},\ \mathsf{sk} := \mathsf{sk}^{\mathsf{IPG}}$.
$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \mathsf{tag},\ \mathbb{S} := (M, \rho))$ :  $h_0 := H(\mathsf{tag}) \in \mathbb{G}_0$,
    choose random $\vec{u}$ such that $\vec{1} \cdot \vec{u} = 1$,
    for $i \in [l],\ s_i := M_i \cdot \vec{u}$, choose random $\vec{\tau}_i := (\tau_{i,j})$ such that $s_i = \sum_{j=1}^{n} \tau_{i,j}$,
    if $\rho(i) = (t, v_i := (v_{i,j}) \in \{0,1\}^n), \quad k_{i,j} := \phi_{t,j,v_{i,j}}(h_0)^{\tau_{i,j}}$,
    return $\mathsf{sk}_{\mathsf{tag},\mathbb{S}} := \{k_{i,j}\}_{i\in[l],j\in[n]}$.
$\mathsf{Enc}(\mathsf{pk},\ m,\ \mathsf{tag},\ \mathit{\Gamma})$ :  $h_0 := H(\mathsf{tag}) \in \mathbb{G}_0,\ \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$,
    for $(t, x_t := (x_{t,j}) \in \{0,1\}^n) \in \mathit{\Gamma}, \quad c_{t,j} := \hat{g}_{t,j,x_{t,j}}^{\zeta}$,
    $z := e_0(h_0, \hat{g}_0)^{\zeta},\ c_T := z \cdot m$,  return  $\mathsf{ct}_{\mathsf{tag},\mathit{\Gamma}} := (\{c_{t,j}\}_{(t,\cdot)\in\mathit{\Gamma},j\in[n]}, c_T)$.
$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{\mathsf{tag},\mathbb{S}} := \{k_{i,j}\}_{i\in[l],j\in[n]},\ \mathsf{ct}_{\mathsf{tag}',\mathit{\Gamma}} := (\{c_{t,j}\}_{(t,\cdot)\in\mathit{\Gamma},j\in[n]}, c_T))$ :
   if $\mathsf{tag} = \mathsf{tag}'$ and $\mathbb{S} := (M, \rho)$ accepts $\mathit{\Gamma} := \{(t, x_t)\}$, then compute $\{\sigma_i\}_{\rho(i)\in\mathit{\Gamma}}$
```

such that $\vec{1} = \sum_{\rho(i) \in \Gamma} \sigma_i M_i$, where M_i is the i-th row of M,

$$z' := \prod_{\rho(i)=(t,(v_{i,j}))\in\Gamma} \left(\prod_{j=1}^{n} e_{t,j,v_{i,j}}(k_{i,j}, c_{t,j}) \right)^{\sigma_i}, \quad \text{return} \quad m' := c/z'.$$

otherwise, return $\perp$.

[Correctness]: If $\mathsf{tag} = \mathsf{tag}'$ and $\mathbb{S}$ accepts Γ,

$$z' = \textstyle\prod_{\rho(i)=(t,(v_{i,j}))\in\Gamma} \left(\prod_{j=1}^{n} e_{t,j,v_{i,j}}(k_{i,j}, c_{t,j}) \right)^{\sigma_i}$$
$$= \textstyle\prod_{\rho(i)=(t,(v_{i,j}))\in\Gamma} \left(\prod_{j=1}^{n} e_{t,j,v_{i,j}}(\phi_{t,j,v_{i,j}}(h_0)^{\tau_{i,j}}, \hat{g}_{t,j,v_{i,j}}^{\zeta}) \right)^{\sigma_i}$$
$$= \textstyle\prod_{\rho(i)=(t,\cdot)\in\Gamma} \left(\prod_{j=1}^{n} e_0(h_0, \hat{g}_0)^{\zeta\tau_{i,j}} \right)^{\sigma_i} = \prod_{\rho(i)=(t,\cdot)\in\Gamma} \left(e_0(h_0, \hat{g}_0)^{\zeta s_i} \right)^{\sigma_i} = z$$

Theorem 4. *The proposed KP-ABE scheme is (selective-attribute) PH-PQ secure under the $2dn$-qIsog-DBDH assumption in the quantum random oracle model.*

For any adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, there exists an adversary $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ for the $2dn$-qIsog-DBDH problem, where $\mathcal{B}_1$ is a polynomial-time quantum machine and $\mathcal{B}_2$ is a classical ppt machine, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{abe,ph\text{-}pq}}(\lambda) \leq \mathsf{Adv}_{\mathcal{B}}^{2dn\text{-}\mathsf{qIsog\text{-}DBDH}}(\lambda)$.

Theorem 4 is proven in a similar manner to Theorem 3. The proof is given in the full version of this paper.

6 Concluding Remarks

We have several open problems arising in the IPG framework. First, can we construct a pre-challenge quantum secure *ciphertext-policy (CP)*-ABE scheme on IPG ? In KP-ABE, ciphertexts are associated with attributes, but a policy (or access structure) in CP-ABE, which has a more complicated structure than an attribute set. Hence, it seems difficult to encode such a complicated object with restricted public parameters as described in Section 1.3.

Second, although we can formulate isogeny-related DLIN (Isog-DLIN) and q-type assumptions on IPG in a manner similar to the Isog-DBDH assumptions (in Section 2.2), we have not yet a new, interesting cryptosystem (for the first time) from such assumptions on IPG. Can we present some interesting application from the new assumptions ? For example, since there exist no *adaptively secure* (KP-)ABE for expressive access structures *against quantum adversaries* in the standard model (even from lattices !), can we construct an adaptively secure, (pre-challenge) quantum secure (KP-)ABE scheme on IPG for span program access structures in the standard model ?

References

1. Abdalla, M., Bellare, M., Catalano, D., Kiltz, E., Kohno, T., Lange, T., Malone-Lee, J., Neven, G., Paillier, P., Shi, H.: Searchable encryption revisited: Consistency

properties, relation to anonymous ibe, and extensions. J. Cryptology 21(3), 350–391 (2008)

2. Agrawal, S., Boneh, D., Boyen, X.: Efficient lattice (H)IBE in the standard model. In: EUROCRYPT 2010. pp. 553–572 (2010)
3. Akinyele, J.A., Garman, C., Miers, I., Pagano, M.W., Rushanan, M., Green, M., Rubin, A.D.: Charm: a framework for rapidly prototyping cryptosystems. J. Cryptographic Engineering 3(2), 111–128 (2013)
4. Ambainis, A., Rosmanis, A., Unruh, D.: Quantum attacks on classical proof systems: The hardness of quantum rewinding. In: FOCS 2014. pp. 474–483 (2014)
5. Beimel, A.: Secure schemes for secret sharing and key distribution. PhD Thesis, Israel Institute of Technology, Technion, Haifa (1996)
6. Bellare, M., Desai, A., Pointcheval, D., Rogaway, P.: Relations among notions of security for public-key encryption schemes. In: CRYPTO '98. pp. 26–45 (1998)
7. Biasse, J.F., Song, F.: On the quantum attacks against schemes relying on the hardness of finding a short generator of an ideal in Qpn. CACR Report 2015, 12 (2015), `http://cacr.uwaterloo.ca/techreports/2015/cacr2015-12.pdf`
8. Biasse, J., Jao, D., Sankar, A.: A quantum algorithm for computing isogenies between supersingular elliptic curves. In: INDOCRYPT 2014. pp. 428–442 (2014)
9. Boneh, D., Dagdelen, Ö., Fischlin, M., Lehmann, A., Schaffner, C., Zhandry, M.: Random oracles in a quantum world. In: ASIACRYPT 2011. pp. 41–69 (2011)
10. Boneh, D., Franklin, M.K.: Identity-based encryption from the Weil pairing. In: CRYPTO 2001. pp. 213–229 (2001)
11. Boneh, D., Gentry, C., Gorbunov, S., Halevi, S., Nikolaenko, V., Segev, G., Vaikuntanathan, V., Vinayagamurthy, D.: Fully key-homomorphic encryption, arithmetic circuit ABE and compact garbled circuits. In: EUROCRYPT 2014. pp. 533–556 (2014)
12. Boneh, D., Gentry, C., Hamburg, M.: Space-efficient identity based encryption without pairings. In: FOCS 2007. pp. 647–657 (2007)
13. Boneh, D., Zhandry, M.: Quantum-secure message authentication codes. In: EUROCRYPT 2013. pp. 592–608 (2013)
14. Boneh, D., Zhandry, M.: Secure signatures and chosen ciphertext security in a quantum computing world. In: CRYPTO 2013, Part II. pp. 361–379 (2013)
15. Charles, D., Lauter, K., Goren, E.: Cryptographic hash functions from expander graphs. J. Crypt. 22(1), 93–113 (2009), preliminary version: *IACR Cryptology eprint Archiv*, 2006:021, 2006
16. Chen, L., Jordan, S., Liu, Y.K., Moody, D., Peralta, R., Perlner, R., Smith-Tone, D.: Report on Post-Quantum Cryptography. NISTIR 8105 (Draft) (2016)
17. Childs, A., Jao, D., Soukharev, V.: Constructing elliptic curve isogenies in quantum subexponential time. J. Math. Crypt. 8(1), 1–29 (2014)
18. Cocks, C.: An identity based encryption scheme based on quadratic residues. In: Cryptography and Coding, 8th IMA International Conference. pp. 360–363 (2001)
19. Costello, C., Longa, P., Naehrig, M.: Efficient algorithms for supersingular isogeny Diffie-Hellman. In: CRYPTO 2016, Part I. pp. 572–601 (2016)
20. Dagdelen, Ö., Fischlin, M., Gagliardoni, T.: The Fiat-Shamir transformation in a quantum world. In: ASIACRYPT 2013, Part II. pp. 62–81 (2013)
21. Damgård, I., Funder, J., Nielsen, J.B., Salvail, L.: Superposition attacks on cryptographic protocols. In: ICITS 2013. pp. 142–161 (2013)
22. De Feo, L., Jao, D., Plût, J.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. J. Math. Crypt. 8(3), 209–247 (2014), preliminary version: *IACR Cryptology eprint Archiv*, 2011:506, 2011

23. Delfs, C., Galbraith, S.D.: Computing isogenies between supersingular elliptic curves over Fp. Des. Codes Cryptography 78(2), 425–440 (2016)
24. Ducas, L., Lyubashevsky, V., Prest, T.: Efficient identity-based encryption over NTRU lattices. In: ASIACRYPT 2014, Part II. pp. 22–41 (2014)
25. Galbraith, S.D., Petit, C., Shani, B., Ti, Y.B.: On the security of supersingular isogeny cryptosystems. IACR Cryptology ePrint Archive 2016, 859 (2016), to appear in ASIACRYPT 2016
26. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: STOC 2008. pp. 197–206 (2008)
27. Gentry, C., Silverberg, A.: Hierarchical ID-based cryptography. In: ASIACRYPT 2002. pp. 548–566 (2002)
28. Gorbunov, S., Vaikuntanathan, V., Wee, H.: Attribute-based encryption for circuits. In: STOC 2013. pp. 545–554 (2013)
29. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: ACM CCS 2006. pp. 89–98 (2006)
30. Hallgren, S., Smith, A., Song, F.: Classical cryptographic protocols in a quantum world. In: CRYPTO 2011. pp. 411–428 (2011)
31. Hoory, S., Linial, N., Wigderson, A.: Expander graphs and their application. Bulletin of AMS 43(4), 439–561 (2006)
32. ISO/IEC 18033-5:2015: Information technology - Security techniques - Encryption algorithms - Part 5: Identity-based ciphers. ISO/IEC (2015)
33. Jao, D., Soukharev, V.: Isogeny-based quantum-resistant undeniable signatures. In: PQCrypto 2014. pp. 160–179 (2014)
34. Nielsen, M.A., Chuang, I.: Quantum Computation and Quantum Information. Cambridge University Press, 10-th anniversary edn. (2010)
35. Okamoto, T., Takashima, K.: Fully secure functional encryption with general relations from the decisional linear assumption. In: CRYPTO 2010. pp. 191–208 (2010), full version is available at `http://eprint.iacr.org/2010/563`
36. Pizer, A.: Ramanujan graphs and Hecke operators. Bull. AMS 23(1), 127–137 (1990)
37. Rostovtsev, A., Stolbunov, A.: Public-key cryptosystem based on isogenies. IACR Cryptology ePrint Archive 2006, 145 (2006), `http://eprint.iacr.org/2006/145`
38. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J. Comput. 26(5), 1484–1509 (1997)
39. Silverman, J.: The Arithmetic of Elliptic Curves, GTM, vol. 106. Springer Verlag, 2nd edn. (2009)
40. Sutherland, A.: Identifying supersingular elliptic curves. LMS J. Comp. and Math. 15, 317–325 (2012)
41. Targhi, E.E., Unruh, D.: Post-quantum security of the Fujisaki-Okamoto and OAEP transforms. IACR Cryptology ePrint Archive 2015, 1210 (2015), to appear in TCC 2016-B
42. Unruh, D.: Universally composable quantum multi-party computation. In: EUROCRYPT 2010. pp. 486–505 (2010)
43. Unruh, D.: Non-interactive zero-knowledge proofs in the quantum random oracle model. In: EUROCRYPT 2015, Part II. pp. 755–784 (2015)
44. Vélu, J.: Isogénies entre courbes elliptiques. C.R. Acad. Sc. Paris, Séries A. 273, 238–241 (1971)
45. Waters, B.: Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In: CRYPTO 2009. pp. 619–636 (2009)
46. Watrous, J.: Zero-knowledge against quantum attacks. SIAM J. Comput. 39(1), 25–58 (2009)

47. Yoshida, R., Takashima, K.: Computing a sequence of 2-isogenies on supersingular elliptic curves. IEICE Trans. Fundamentals 96-A(1), 158–165 (2013), preliminary version is available in ICISC 2008, LNCS, vol.5461, pp.52–65 (2008)
48. Zhandry, M.: How to construct quantum random functions. In: FOCS 2012. pp. 679–687 (2012)
49. Zhandry, M.: Secure identity-based encryption in the quantum random oracle model. In: CRYPTO 2012. pp. 758–775 (2012)

A Mathematical Backgrounds on IPG

Here, we show mathematical backgrounds for IPG constructions. Section A.1 introduces several basic facts on elliptic curves, and gives Vélu's formulas for isogeny as fundamental operations. Section A.2 gives a basic one-way function from isogeny computation as is explained in [15, 22]. Section A.3 gives two explicit constructions of trapdoor homomorphisms from isogenies, Algorithms 1 and 2. A trapdoor of Algorithm 1 (resp. 2) is given by a point generating the kernel of the isogeny (resp. a walk data indicating the kernel). Section A.4 gives a proposition for compatibility between the Weil pairing and isogeny.

A.1 Isogenies between Elliptic Curves

We summarize facts about elliptic curves. For details, see [39], for example.

Let p be a prime greater than 3 and $\mathbb{F}_p$ be the finite field with p elements. In this paper, we consider only primes p with $q \,|\, p+1$, where q is the prime order of a large cyclic group. Let $\overline{\mathbb{F}}_p$ be its algebraic closure. An elliptic curve E over $\overline{\mathbb{F}}_p$ is given by the Weierstrass normal form

$$E : y^2 = x^3 + Ax + B \tag{5}$$

for A and $B \in \overline{\mathbb{F}}_p$ where the discriminant of the RHS of (5) is non-zero. We denote the point at infinity on E by $\mathcal{O}_E$. Elliptic curves are endowed with a unique algebraic group structure, with $\mathcal{O}_E$ as neutral element. The j-invariant of E is $j(A,B) = 1728\frac{4A^3}{4A^3+27B^2}$. Conversely, for $j \neq 0, 1728 \in \overline{\mathbb{F}}_p$, set $A = A(j) = \frac{3j}{1728-j}$, $B = B(j) = \frac{2j}{1728-j}$. Then, the obtained E in (5) has j-invariant j. Two elliptic curves over $\overline{\mathbb{F}}_p$ are isomorphic if and only if they have the same j-invariant. For a positive integer n, the set of n-torsion points of E is $E[n] = \{P \in E(\overline{\mathbb{F}}_p) \,|\, nP = \mathcal{O}_E\}$.

Given two elliptic curves E and E' over $\overline{\mathbb{F}}_p$, a homomorphism $\phi : E \to E'$ is a morphism of algebraic curves that sends $\mathcal{O}_E$ to $\mathcal{O}_{E'}$. A non-zero homomorphism is called an isogeny, and a separable isogeny with the cardinality ℓ of the kernel is called ℓ-isogeny. We consider only *separable* isogenies in this paper, i.e., any isogeny is separable here.

An elliptic curve E over $\overline{\mathbb{F}}_p$ is called supersingular if there are no points of order p, i.e., $E[p] = \{\mathcal{O}_E\}$. The j-invariants of supersingular elliptic curves lie in $\mathbb{F}_{p^2}$ (see [39, Chap. V, Th. 3.1]), and $E[q] \subset E(\mathbb{F}_{p^2})$.

Vélu's Formulas We compute the ℓ-isogeny by using Vélu's formulas for a small prime $\ell = 2, 3, \ldots$. Vélu gave in [44] the explicit formulas of the isogeny

$\psi : E \to E'$ and the equation of the form (6) of E' when E is given by (5) and $C = \ker \psi$ is explicitly given. Then there exists a unique isogeny $\psi : E \to E'$ s.t. $C = \ker \psi$, and we denote E' by E/C.

For an elliptic curve E and a cyclic group C of order ℓ, Vélu's formula [44] gives an isogenous curve E/C and the associated isogeny $E \ni (x, y) \mapsto (X, Y) \in E/C$. For computing it, for $E : y^2 = x^3 + Ax + B$ and point $Q = (x_Q, y_Q) \neq O_E \in C$, we define $g^x_Q = 3x_Q^2 + a, g^y_Q = -2y_Q$, and $t_Q = 2g^x_Q$ if $Q \in E[2]$, $t_Q = g^x_Q$ if $Q \notin E[2]$, $u_Q = (g^y_Q)^2$. For $S = (C - \{\mathcal{O}_E\})/\pm 1$, let $t = \sum_{Q \in S} t_Q, w = \sum_{Q \in S}(u_Q + x_Q t_Q), A' = A - 5t, B' = B - 7w$, then,

$$E/C : Y^2 = X^3 + A'X + B', \quad X = x + \sum_{Q \in S} \left(\frac{t_Q}{x - x_Q} + \frac{u_Q}{(x - x_Q)^2} \right),$$

$$Y = y - \sum_{Q \in S} \left(\frac{2u_Q y}{x - x_Q} + \frac{t_Q(y - y_Q) - g^x_Q g^y_Q}{(x - x_Q)^2} \right) \tag{6}$$

gives the curve and isogeny.

In particular, the $\ell = 2$ case is given compactly as indicated below. For a subgroup $C = \langle (\vartheta, 0) \rangle \subset E[2]$ of order 2, the elliptic curve E/C is given by the equation

$$Y^2 = X^3 - (4A + 15\vartheta^2)X + (8B - 14\vartheta^3). \tag{7}$$

Therefore, E/C is also defined over $\mathbb{F}_{p^2}$ when E is supersingular. Moreover, the isogeny $\psi := \psi_\vartheta$: $E \to E/C$ is given by

$$\psi : (x, y) \mapsto (X, Y) := \left(x + \frac{(3\vartheta^2 + A)}{x - \vartheta}, y - \frac{(3\vartheta^2 + A)y}{(x - \vartheta)^2} \right), \tag{8}$$

$\psi(\mathcal{O}_E) = \mathcal{O}_{E/C}$ and $\psi((\vartheta, 0)) = \mathcal{O}_{E/C}$. Clearly, ψ is also defined over $\mathbb{F}_{p^2}$ for supersingular E.

A.2 Basic One-way Function from Isogeny Computation

We consider a graph consisting of ℓ-isogenies between supersingular elliptic curves. The graph has an expanding property (expander graph), and is called a Pizer graph [36, 15] or an isogeny graph [40].

Expander Graph Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a (directed) graph with vertex set $\mathcal{V}$ and edge set $\mathcal{E}$. A graph $\mathcal{G}$ is an expander graph with expansion constant $c > 0$ if, for any subset $\mathcal{U} \subset \mathcal{V}$ s.t. $|\mathcal{U}| \leq \frac{|\mathcal{V}|}{2}$, then $|\Gamma(\mathcal{U})| \geq c\,|\mathcal{U}|$ where $\Gamma(\mathcal{U})$ is the boundary of $\mathcal{U}$ (which is all neighbors of $\mathcal{U}$ minus all elements of $\mathcal{U}$). Any expander graph is connected. A random walk on an expander graph has rapidly mixing property. After $O(\log(|\mathcal{V}|))$ steps, the last point of the random walk approximates the uniform distribution on $\mathcal{V}$.

Such a property is useful for cryptography. Therefore, there exist several cryptographic constructions using an expander graph ([15, 37] etc.). For details of expander graphs, see [31].

Pizer Graph The Pizer graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a small prime ℓ consists of isomorphism classes of supersingular elliptic curves over $\overline{\mathbb{F}}_p$ as vertex set $\mathcal{V}$, and (informally) their ℓ-isogenies as edge set $\mathcal{E}$. Precisely, the vertex set $\mathcal{V}$ is the set of supersingular j-invariants and edges $(j, j') \in \mathbb{F}_{p^2}^2$ present with multiplicity k whenever j' is a root of $\Phi_\ell(j, Y)$ with multiplicity k, where $\Phi_\ell(X, Y)$ is the classical modular polynomial (see Definition 1 in [40]). Equivalently, $\ell + 1$ edges are coming from any vertex in $\mathcal{V}$, and when the vertex is represented by an elliptic curve E, they are associated with $\ell + 1$ ℓ-torsion cyclic subgroups on E. For each edge from E, the other vertex is the quotient curve E/C where C represents the corresponding subgroup of order ℓ.

The graph is directed, in which the direction of the edge associated with (j, j') (resp. (E, C)) is defined to be from j to j' (resp. from E to $E' = E/C$). The in-degree and out-degree of any vertex are $\ell + 1$, and it has a multi-edge (j, j') when $\Phi_\ell(j, Y)$ has a multiple root $Y = j'$ with multiplicity ≥ 2. Moreover, it has a self-loop (j, j) when $\Phi_2(X, X)$ has a root $X = j$.

$\mathcal{G}$ is known to have a rapidly mixing property. In particular, this is called a Ramanujan graph, a special type of expander graph. For details, see [36, 15], for example.

One-wayness of Isogeny Computation against Quantum Computers In summary, we have a one-way function (9) from isogeny (sequence) computation

$$\text{Isogeny} \quad (E, C) \overset{\text{easy}}{\underset{\text{hard}}{\rightleftarrows}} (E, E/C), \tag{9}$$

where E is a supersingular elliptic curve (EC) and $C \subset E[\ell^\kappa]$ is an order-ℓ^κ cyclic torsion subgroup where $p = \Theta(2^\lambda)$, $\kappa = \Theta(\log p) = \Theta(\lambda)$ for the security parameter λ. First, from the expanding property (or rapidly mixing property) explained above, by walking just $\kappa = \Theta(\log p)$-times iteratively, our ending point E has almost uniform distribution in the isogeny graph. This improves efficiency of the forward direction function evaluation in Eq. (9).

Childs et al. [17] proposed a *subexponential time quantum algorithm* for the inverse direction function given in Eq. (9), i.e., for the isogeny problem between *ordinary* elliptic curves. However, there exists *no* subexponential time quantum algorithm for the isogeny problem between *supersingular* elliptic curves while Biasse et al. [23, 8] made a progress on the exponential time algorithm based on a Grover-type quantum search. Therefore, our isogeny function given in Eq. (9) is considered as one-way even against quantum adversaries at present.

A.3 Trapdoor Homomorphisms from Isogeny Sequence Algorithms

At present, there exist two types of algorithms for computing isogeny sequences, one by Charles et al. [15] and another by De Feo et al. [22]. The first is given in a mathematically clear manner according to Eq. (9) and the second has an advantage of no additional restriction for the system parameter p. Therefore, for our purpose, it would be better to use the second algorithm, Algorithm 2, denoted by $\mathrm{Isog}^{\mathsf{clg}}_{\ell,\kappa}$.

Isogeny Sequence Algorithm by De Feo et al. [22] Let ℓ be a small prime, for example, $\ell = 2, 3, \ldots$, and a large prime p satisfies $\ell^\kappa \,|\, p+1$. Then, a supersingular EC has a rational subgroup $(\mathbb{Z}/\ell^\kappa\mathbb{Z})^2 \subseteq E(\mathbb{F}_{p^2}) \simeq (\mathbb{Z}/(p+1)\mathbb{Z})^2$. In other words, all the ℓ^κ-torsion points are defined over $\mathbb{F}_{p^2}$. For using a point R in $E[\ell^\kappa]$, we can compute an isogeny $\phi : E \to E/\langle R\rangle$ by iteratively using Vélu's formula for ℓ-isogenies. (In [22], the algorithm is used for establishing a DH type key exchange.)

In our parameter selection, first generate a random point R in $E[\ell^\kappa]$, then set $E_0 := E, R_0 := R$ and, for $0 \le i < \kappa$, let

$$E_{i+1} := E_i/\langle \ell^{\kappa-i-1}R_i\rangle, \quad \psi_i : E_i \to E_{i+1}, \quad R_{i+1} := \psi_i(R_i),$$

where $R_i \in E_i[\ell^{\kappa-i}]$, $\ell^{\kappa-i-1}R_i$ is in $E_i[\ell]$ and then ψ_i is an ℓ-isogeny. The composition gives the desired

$$\phi := \psi_{\kappa-1}\cdots\psi_0 : E = E_0 \to E_\kappa = E/\langle R\rangle.$$

We describe the algorithm in Algorithm 1 and call it $\mathrm{Isog}^{\mathsf{djp}}_{\ell,\kappa}$ after De Feo, Jao, and Plût. A trapdoor ξ of trapdoor homomorphisms indicated in Def. 1 is given by the kernel generating point R.

Algorithm 1 $\mathrm{Isog}^{\mathsf{djp}}_{\ell,\kappa}$: Generate a random supersingular EC which is ℓ^κ-isogenous to E_0 (given in [22])

```
Input :  An initial elliptic curve $E_0$.
Output :  An isogenous $E$ and a kernel generator $R$ in $E_0[\ell^\kappa]$, that is,
    a trapdoor $\xi$ for computing the isogeny $\phi := \phi_\xi : E_0 \to E$.
1: generate a random point $R$ in $E_0[\ell^\kappa]$, then set $R_0 := R$
2: for $0 \le i < \kappa$ do
3:     compute $E_{i+1} := E_i/\langle \ell^{\kappa-i-1}R_i\rangle$, $\psi_i : E_i \to E_{i+1}$, and $R_{i+1} := \psi_i(R_i)$ by Vélu's
       formula, where $R_i \in E_i[\ell^{\kappa-i}]$, $\ell^{\kappa-i-1}R_i$ is in $E_i[\ell]$ and then $\psi_i$ is an $\ell$-isogeny.
4: end for
5: we set the composition $\phi := \psi_{\kappa-1}\cdots\psi_0 : E_0 \to E_\kappa = E_0/\langle R\rangle$.
   return $E := E_\kappa$ (or $j(E_\kappa)$) and $\xi := R$.
```

Isogeny Sequence Algorithm by Charles et al. [15] When $\ell = 2$, we can use a base prime p *without the restriction (for Algorithm 1) that* $p+1 = f \cdot \ell^\kappa$ *with* $\kappa = \Theta(\log p)$. From the cryptographic perspective, it would be better to use a general prime p from security and efficiency reasons. Since $2 \,|\, p+1$ for any odd prime p, a supersingular EC defined over $\mathbb{F}_{p^2}$ with $E(\mathbb{F}_{p^2}) \simeq (\mathbb{Z}/(p+1)\mathbb{Z})^2 \supseteq E[2] \simeq (\mathbb{Z}/2\mathbb{Z})^2$. In other words, all the 2-torsion points are always defined over $\mathbb{F}_{p^2}$. Therefore, by iteratively choosing 2-isogenies in a random manner, we can obtain a random ℓ^κ-isogeny from any supersingular E.

We consider computing a 2-isogeny sequence

$$E_0 \xrightarrow{\psi_0} E_1 \xrightarrow{\psi_1} \cdots \xrightarrow{\psi_{\kappa-2}} E_{\kappa-1} \xrightarrow{\psi_{\kappa-1}} E_\kappa \tag{10}$$

where E_i are supersingular without backtracking, i.e., $\psi_i \neq \widehat{\psi}_{i+1}$ for $i = 0, \ldots, \kappa - 2$ and all ψ_i are given by Vélu's formulas (7). The isogeny sequence starting from E_0 is determined by a bit sequence. As is explained before, a supersingular elliptic curve E_0 and the 2-torsion points on E_0 are defined over $\mathbb{F}_{p^2}$.

As the out-degree of any vertex is three and the walk we consider has no backtracking, we have 2 possibilities to proceed to the next vertex in $\mathcal{V}$ at $i \geq 1$. For $i = 0$, we fix 2 possibilities $(\psi_{0,0}, \psi_{0,1})$ from E_0 at the beginning. For each $i = 0, \ldots, \kappa - 1$, a next step is determined by a lexicographical order in $\mathbb{F}_{p^2}$ for choosing a next j-invariant. That is, we associate a walk data $\omega = \omega_0\omega_1 \cdots \omega_{\kappa-1} \in \{0,1\}^\kappa$ with a sequence (10) where $\psi_0 = \psi_{0,0}$ or $\psi_{0,1}$. Our goal is to compute the j-invariant $j_\kappa = j(E_\kappa)$ from $j_0 = j(E_0)$ and a walk data $\omega \in \{0,1\}^\kappa$ that determines the next vertices. For the details, see [15, 47].

We describe the algorithm in Algorithm 2 and call it $\mathsf{Isog}^{\mathsf{clg}}_{\ell,\kappa}$ after Charles, Lauter, and Goren. A trapdoor ξ of trapdoor homomorphisms indicated in Def. 1 is given by the walk data ω.

Algorithm 2 $\mathsf{Isog}^{\mathsf{clg}}_{\ell,\kappa}$: Generate a random supersingular EC which is ℓ^κ-isogenous to E_0 (given in [15, 47]) when $\ell = 2$

```
Input : An initial elliptic curve $E_0$.
Output : An isogenous $E$ and all the selector bits $\omega := \{\omega_i\}_{0 \leq i < \kappa}$, that is,
    a trapdoor $\xi$ for computing the isogeny $\phi := \phi_\xi : E_0 \to E$.
for $0 \leq i < \kappa$ do
    generate a random bit $\omega_i \in \{0,1\}$ for selecting a next kernel point $R_i$,
    which is either of two points in $K_i := E_i[\ell] \setminus \psi_{i-1}(E_{i-1}[\ell])$ if $i \neq 0$
    (resp., in $K_i := \{$ some fixed two points in $E_i[\ell] \setminus \{\mathcal{O}_{E_i}\}\}$ if $i = 0$) since $\ell = 2$.
    compute $E_i/\langle R_i \rangle$ and the $j$-invariants $j(E_i/\langle R_i \rangle)$ by Vélu's formula for two candidates $R_i \in K_i$.
    $j(E_i/\langle R_i \rangle)$ i.e., $R_i$, is determined from $\omega_i$ by a lexicographic order in $\mathbb{F}_{p^2}$.
    we set $\psi_i : E_i \to E_{i+1} := E_i/\langle R_i \rangle$ for the selected $R_i$.
end for
we set the composition $\phi := \psi_{\kappa-1} \cdots \psi_0 : E_0 \to E_\kappa$.
return $E := E_\kappa$ (or $j(E_\kappa)$) and all the selector bits $\xi := \omega := \{\omega_i\}_{0 \leq i < \kappa}$.
```

CSSI Problem by IPG notation Using a prime p with $p + 1 = \ell_A^{\kappa_A} \ell_B^{\kappa_B} \cdot f$ and $\mathsf{Isog}^{\mathsf{djp}}_{\ell,\kappa}$ with $\ell := \ell_A, \kappa := \kappa_A$, De Feo et al.'s assumption, Computational Supersingular Isogeny (CSSI) assumption is given as a special instance of our isogeny assump. in Def. 5. We give the CSSI problem [22] by using our IPG notation. It has been believed to have *no* efficient quantum attack for it since the first publication of [22] in 2011 on IACR ePrint.

Definition 15 (CSSI Problem [22] by IPG notation).

Let $(\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t=0,1}, \mathbb{G}_T),\ \mathsf{sk}^{\mathsf{IPG}} := \phi_1) \xleftarrow{\mathsf{R}} \mathsf{Gen}^{\mathsf{IPG}}(1^\lambda, 1)$*, where*

1. *a prime* p *with* $p + 1 = \ell_A^{\kappa_A} \ell_B^{\kappa_B} \cdot f$ *is used*
2. $\mathrm{Isog}^{\mathsf{djp}}_{\ell,\kappa}$ *with* $\ell := \ell_A, \kappa := \kappa_A$ *is used*
3. $\mathbb{G}_t$*,* $\hat{\mathbb{G}}_t$ *and* $\hat{\mathbb{G}}_T$ *have order* $\ell_B^{\kappa_B}$*, not prime but a smooth prime power.*

If adversary $\mathcal{B}$ *outputs* ϕ_1 *(or a point* $R_A \in E_0[\ell_A^{\kappa_A}]$ *s.t.* $\ker \phi_1 = \langle R_A \rangle$*) when given* $\mathsf{pk}^{\mathsf{IPG}}$*,* $\mathcal{B}$ *wins.*

A.4 Compatibility between Pairing and Isogeny for IPG

The Weil Pairing under Isogeny The Weil pairing is compatible with isogenies as in the following proposition. It is a key fact for our construction of Isogeneous Pairing Groups (IPG).

Proposition 2. [39, Chap. III, Thm. 6.1 & Prop. 8.2] *For any* $P, Q \in E_0[q]$ *and any (non-constant) isogeny* $\phi : E_0 \to E$*, it holds* $e_{\mathsf{weil}}(\phi(P), \phi(Q)) = e_{\mathsf{weil},0}(P, Q)^{\deg \phi}$*, where* $e_{\mathsf{weil},0}$ *(resp.* e_{weil}*) is the Weil pairing on* E_0 *(resp.* E*).*

B Instantiations of Our Constructions by Elliptic Curves

B.1 Instantiation of IPG from Supersingular Elliptic Curves

We instantiate an IPG from compatibly generated supersingular elliptic curves, where isogeny generator $\mathrm{Isog}_{\ell,\kappa} = \mathrm{Isog}^{\mathsf{djp}}_{\ell,\kappa}$ or $\mathrm{Isog}^{\mathsf{clg}}_{\ell,\kappa}$ is defined by Algorithm 1 or 2, respectively.

```
$\mathsf{Gen}^{\mathsf{IPG}}(1^\lambda, d)$ : Generate a random supersingular elliptic curve $E_0/\mathbb{F}_{p^2}$
        with a sufficiently large, odd prime $p$, generate a suitable $(\ell, \kappa)$,
    $(\mathbb{G}_0, \hat{\mathbb{G}}_0, \mathbb{G}_T; e_0)$ : a system of asymmetric pairing groups of order $r$ from
        subgroups of $E_0$, where $e_0$ is defined by $e_0(h_0, \hat{h}_0) := e_{\mathsf{weil},0}(h_0, \hat{h}_0)^{l^\kappa}$
        for any $h_0 \in \mathbb{G}_0, \hat{h}_0 \in \hat{\mathbb{G}}_0$, from the Weil pairing $e_{\mathsf{weil},0}$ on $E_0$
    $g_0 \xleftarrow{\mathsf{U}} \mathbb{G}_0,\ \hat{g}_0 \xleftarrow{\mathsf{U}} \hat{\mathbb{G}}_0$,
    for $t \in [d]$, $(E_t, \xi_t) \xleftarrow{\mathsf{R}} \mathrm{Isog}_{\ell,\kappa}(E_0)$, $\phi_t := \phi_{\xi_t}$, $\mathbb{G}_t := \phi_t(\mathbb{G}_0)$, $\hat{\mathbb{G}}_t := \phi_t(\hat{\mathbb{G}}_0)$,
        $g_t := \phi_t(g_0)$, $\hat{g}_t := \phi_t(\hat{g}_0)$, $e_t(h_t, \hat{h}_t) := e_{\mathsf{weil},t}(h_t, \hat{h}_t)$ for any $h_t \in \mathbb{G}_t, \hat{h}_t \in \hat{\mathbb{G}}_t$,
        where $e_{\mathsf{weil},t}$ is the Weil pairing on $E_t$,
    return $\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t\in[0,d]}, \mathbb{G}_T)$, $\mathsf{sk}^{\mathsf{IPG}} := (\xi_t)_{t\in[d]}$ for $(\phi_t)_{t\in[d]}$.
$\mathsf{SimGen}^{\mathsf{IPG}}(\mathbb{G}_0, \hat{\mathbb{G}}_0, g_0, \hat{g}_0, e_0)$ :
    $(E, \xi) \xleftarrow{\mathsf{R}} \mathrm{Isog}_{l,\kappa}(E_0)$, where $E_0$ is the elliptic curve including $\mathbb{G}_0, \hat{\mathbb{G}}_0$,
    $\phi := \phi_\xi$, $g := \phi(g_0)$, $\hat{g} := \phi(\hat{g}_0)$, $\mathbb{G} := \phi(\mathbb{G}_0)$, $\hat{\mathbb{G}} := \phi(\hat{\mathbb{G}}_0)$,
    $e(h, \hat{h}) := e_{\mathsf{weil}}(h, \hat{h})$ for any $h \in \mathbb{G}, \hat{h} \in \hat{\mathbb{G}}$, where $e_{\mathsf{weil}}$ is the Weil pairing on $E$,
    return $(\mathbb{G}, \hat{\mathbb{G}}, g, \hat{g}, e, \xi$ for $\phi)$.
```

B.2 Instantiation of d-qIsog-DBDH Assumption

Definition 16 (d-qIsog-DBDH Problem). *Let $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ be an adversary, where $\mathcal{B}_1$ is modeled as a quantum adversary, $\mathcal{B}_2$ a classical ppt machine.*

Generate a random supersingular elliptic curve $E_0/\mathbb{F}_{p^2}$ with a sufficiently large, odd prime p, and generate a suitable (ℓ, κ).

Generate $(\mathbb{G}_0, \hat{\mathbb{G}}_0, \mathbb{G}_T; e_0)$: a system of asymmetric pairing groups of order q from subgroups of E_0, where e_0 is defined by $e_0(h_0, \hat{h}_0) := e_{\mathsf{weil},0}(h_0, \hat{h}_0)^{\ell^\kappa}$ for any $h_0 \in \mathbb{G}_0, \hat{h}_0 \in \hat{\mathbb{G}}_0$, from the Weil pairing $e_{\mathsf{weil},0}$ on E_0, $g_0 \xleftarrow{\mathsf{U}} \mathbb{G}_0$, $\hat{g}_0 \xleftarrow{\mathsf{U}} \hat{\mathbb{G}}_0$.

For $t \in [d]$, $(E_t, \xi_t) \xleftarrow{\mathsf{R}} \mathrm{Isog}_{\ell,\kappa}(E_0)$, $\phi_t := \phi_{\xi_t}$, $\mathbb{G}_t := \phi_t(\mathbb{G}_0)$, $\hat{\mathbb{G}}_t := \phi_t(\hat{\mathbb{G}}_0)$, $g_t := \phi_t(g_0)$, $\hat{g}_t := \phi_t(\hat{g}_0)$, $e_t(h_t, \hat{h}_t) := e_{\mathsf{weil},t}(h_t, \hat{h}_t)$ for any $h_t \in \mathbb{G}_t, \hat{h}_t \in \hat{\mathbb{G}}_t$, where $e_{\mathsf{weil},t}$ is the Weil pairing on E_t.

Set $\mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t\in[0,d]}, \mathbb{G}_T)$ (and $\mathsf{sk}^{\mathsf{IPG}} := (\phi_t)_{t\in[d]}$), and $\mathcal{B}_1$ outputs $\mathsf{state} \xleftarrow{\mathsf{R}} \mathcal{B}_1(\mathsf{pk}^{\mathsf{IPG}})$. For $\alpha, \beta, \delta \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\mathcal{B}_2$ receives $\mathcal{X}_{\mathfrak{b}}$ for $\mathfrak{b} \xleftarrow{\mathsf{U}} \{0,1\}$, that is defined by

$$\mathcal{X}_0 := (\ \mathsf{state},\ \alpha{\cdot}g_0,\ (\beta{\cdot}g_t)_{t\in[d]},\ g_T^{\alpha\beta}\ )\ \text{ and }\ \mathcal{X}_1 := (\ \mathsf{state},\ \alpha{\cdot}g_0,\ (\beta{\cdot}g_t)_{t\in[d]},\ g_T^{\delta}\ ),$$

where $g_T := e_0(g_0, \hat{g}_0)$. $\mathcal{B}_2$ outputs a guess bit $\mathfrak{b}'$. If $\mathfrak{b} = \mathfrak{b}'$, $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ wins.

B.3 Instantiation of IBE in Section 4

We give an elliptic curve based description of our IBE scheme in Section 4. A master secret key sk is given by an isogeny from E_0 to E_1. Exponentiations on $(\mathbb{G}_t, \hat{\mathbb{G}}_t)_{t=0,1}$ in Section 4 are given by scalar multiplications on elliptic curves.

$\mathsf{Setup}(1^\lambda)$: Generate a random supersingular elliptic curve $E_0/\mathbb{F}_{p^2}$
with a sufficiently large, odd prime p, generate a suitable (ℓ, κ),
$(\mathbb{G}_0, \hat{\mathbb{G}}_0, \mathbb{G}_T; e_0)$: a system of asymmetric pairing groups of order q from
subgroups of E_0, where e_0 is defined by $e_0(h_0, \hat{h}_0) := e_{\mathsf{weil},0}(h_0, \hat{h}_0)^{\ell^\kappa}$
for any $h_0 \in \mathbb{G}_0, \hat{h}_0 \in \hat{\mathbb{G}}_0$, from the Weil pairing $e_{\mathsf{weil},0}$ on E_0
$(E_1, \xi_1) \xleftarrow{\mathsf{R}} \mathrm{Isog}_{\ell,\kappa}(E_0)$, $\phi_1 := \phi_{\xi_1}$, $\mathbb{G}_1 := \phi_1(\mathbb{G}_0)$, $\hat{\mathbb{G}}_1 := \phi_1(\hat{\mathbb{G}}_0)$,
$e_t(h_1, \hat{h}_1) := e_{\mathsf{weil},1}(h_1, \hat{h}_1)$ for any $h_1 \in \mathbb{G}_1, \hat{h}_1 \in \hat{\mathbb{G}}_1$, where $e_{\mathsf{weil},1}$
is the Weil pairing on E_1, $\hat{g}_0 \xleftarrow{\mathsf{U}} \hat{\mathbb{G}}_0$, $\hat{g}_1 := \phi_1(\hat{g}_0)$,
generate a random hash $H : \mathbb{F}_q \to \mathbb{G}_0$ with the identity space $\mathbb{F}_q$,
return $\mathsf{pk} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, \hat{g}_t, e_t)_{t=0,1}, \mathbb{G}_T,\ H)$, $\mathsf{sk} := \xi_1$ for ϕ_1.

$\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \mathsf{ID})$: $h_0 := H(\mathsf{ID}) \in \mathbb{G}_0$, $h_1 := \phi_1(h_0)$, return $\mathsf{sk}_{\mathsf{ID}} := h_1$.

$\mathsf{Enc}(\mathsf{pk}, m, \mathsf{ID})$: $h_0 := H(\mathsf{ID})$, $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times$, $c := \zeta \cdot \hat{g}_1$, $z := e_0(h_0, \hat{g}_0)^{\zeta\cdot\ell^\kappa}$,
$c_T := z \cdot m$, return $\mathsf{ct}_{\mathsf{ID}} := (c, c_T)$.

$\mathsf{Dec}(\mathsf{sk}_{\mathsf{ID}}, \mathsf{ct}_{\mathsf{ID}'})$: if $\mathsf{ID} = \mathsf{ID}'$, $z' := e_1(h_1, c)$,
$m' := c_T \cdot (z')^{-1}$, return m', otherwise, return $\perp$.

Dec correctly decrypts since $z' = e_1(h_1, c) = e_{\mathsf{weil},1}(h_1, c) = e_{\mathsf{weil},1}(\phi_1(h_0), \zeta \cdot \hat{g}_1) = e_{\mathsf{weil},1}(\phi_1(h_0), \phi_1(\hat{g}_0))^{\zeta} = e_{\mathsf{weil},0}(h_0, \hat{g}_0)^{\ell^{\kappa}\zeta} = e_0(h_0, \hat{g}_0)^{\zeta} = z$ if $\mathsf{ID} = \mathsf{ID}'$. Here, we use Prop. 2 and $\deg(\phi_1) = \ell^{\kappa}$.

B.4 Instantiation of Small Universe KP-ABE in Section 5.1

An attribute set Γ is a subset of $[d]$. $(d+1)$ isogenous ECs $(E_t)_{t \in [0,d]}$ are used.

```
$\mathsf{Setup}(1^{\lambda})$ : Generate a random supersingular elliptic curve $E_0/\mathbb{F}_{p^2}$
      with a sufficiently large, odd prime $p$, generate a suitable $(\ell, \kappa)$,
   $(\mathbb{G}_0, \hat{\mathbb{G}}_0, \mathbb{G}_T; e_0)$ : a system of asymmetric pairing groups of order $q$ from
      subgroups of $E_0$, where $e_0$ is defined by $e_0(h_0, \hat{h}_0) := e_{\mathsf{weil},0}(h_0, \hat{h}_0)^{\ell^{\kappa}}$
      for any $h_0 \in \mathbb{G}_0, \hat{h}_0 \in \hat{\mathbb{G}}_0$, from the Weil pairing $e_{\mathsf{weil},0}$ on $E_0$
   $g_0 \xleftarrow{\mathsf{U}} \mathbb{G}_0$, $\hat{g}_0 \xleftarrow{\mathsf{U}} \hat{\mathbb{G}}_0$,
   for $t \in [d]$, $(E_t, \xi_t) \xleftarrow{\mathsf{R}} \mathrm{Isog}_{\ell,\kappa}(E_0)$, $\phi_t := \phi_{\xi_t}$, $\mathbb{G}_t := \phi_t(\mathbb{G}_0)$, $\hat{\mathbb{G}}_t := \phi_t(\hat{\mathbb{G}}_0)$,
      $g_t := \phi_t(g_0)$, $\hat{g}_t := \phi_t(\hat{g}_0)$, $e_t(h_t, \hat{h}_t) := e_{\mathsf{weil},t}(h_t, \hat{h}_t)$ for any $h_t \in \mathbb{G}_t, \hat{h}_t \in \hat{\mathbb{G}}_t$,
      where $e_{\mathsf{weil},t}$ is the Weil pairing on $E_t$,
   generate a random hash $H : \mathbb{F}_q \to \mathbb{G}_0$ with the tag space $\mathbb{F}_q$,
   return $\mathsf{pk} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, \hat{g}_t, e_t)_{t \in [0,d]}, \mathbb{G}_T, H)$, $\mathsf{sk} := (\xi_t)_{t \in [d]}$ for $(\phi_t)_{t \in [d]}$.
$\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \mathsf{tag}, \mathbb{S} := (M, \rho))$ : $h_0 := H(\mathsf{tag}) \in \mathbb{G}_0$,
   choose random $\vec{u}$ such that $\vec{1} \cdot \vec{u} = 1$,
   for $i \in [l]$, $s_i := M_i \cdot \vec{u}$, $t := \rho(i)$, $k_i := s_i \cdot \phi_t(h_0)$,
   return $\mathsf{sk}_{\mathsf{tag},\mathbb{S}} := \{k_i\}_{i \in [l]}$.
$\mathsf{Enc}(\mathsf{pk}, m, \mathsf{tag}, \Gamma)$ : $h_0 := H(\mathsf{tag}) \in \mathbb{G}_0$, $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$, for $t \in \Gamma$, $c_t := \zeta \cdot \hat{g}_t$,
   $z := e_0(h_0, \hat{g}_0)^{\zeta}$, $c_T := z \cdot m$, return $\mathsf{ct}_{\mathsf{tag},\Gamma} := (\{c_t\}_{t \in \Gamma}, c_T)$.
$\mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_{\mathsf{tag},\mathbb{S}} := \{k_i\}_{i \in [l]}, \mathsf{ct}_{\mathsf{tag}',\Gamma} := (\{c_t\}_{t \in \Gamma}, c_T))$ :
  if $\mathsf{tag} = \mathsf{tag}'$ and $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{t\}$, then compute $\{\sigma_i\}_{\rho(i) \in \Gamma}$
    such that $\vec{1} = \sum_{\rho(i) \in \Gamma} \sigma_i M_i$, where $M_i$ is the $i$-th row of $M$,
  $z' := \prod_{t := \rho(i) \in \Gamma} e_t(k_i, c_t)^{\sigma_i}$, return $m' := c/z'$, otherwise, return $\perp$.
```

[Correctness]: If $\mathsf{tag} = \mathsf{tag}'$ and $\mathbb{S}$ accepts Γ,

$z' = \prod_{t := \rho(i) \in \Gamma} e_t(k_i, c_t)^{\sigma_i} = \prod_{t := \rho(i) \in \Gamma} e_{\mathsf{weil},t}(k_i, c_t)^{\sigma_i}$

$= \prod_{t := \rho(i) \in \Gamma} e_{\mathsf{weil},t}\left(s_i \cdot \phi_t(h_0),\ \zeta \cdot \hat{g}_t\right)^{\sigma_i} = \prod_{t := \rho(i) \in \Gamma} e_{\mathsf{weil},t}\left(\phi_t(h_0), \phi_t(\hat{g}_0)\right)^{\zeta \sigma_i s_i}$

$= \prod_{\rho(i) \in \Gamma} e_{\mathsf{weil},0}(h_0, \hat{g}_0)^{\ell^{\kappa} \zeta \sigma_i s_i} = e_{\mathsf{weil},0}(h_0, \hat{g}_0)^{\ell^{\kappa} \zeta} = e_0(h_0, \hat{g}_0)^{\zeta} = z$

by Prop. 2 and $\deg(\phi_t) = \ell^{\kappa}$.

We obtain an elliptic curve instantiation of large universe KP-ABE in Section 5.3 in a similar manner as above (which seems to be described in a rather complicated manner).

C Proposed PH-PQ Secure HIBE

C.1 Construction

The proposed HIBE scheme is a Gentry-Silverberg type HIBE. A master secret key sk is given by an isogeny from $\mathbb{G}_0$ to $\mathbb{G}_1$. Hash functions $H_t : \mathbb{F}_q \to \mathbb{G}_t$ $(t = 0, 1)$ map an arbitrary ID sequence (ID_i), which is represented as an element in $\mathbb{F}_q$, to a point of $\mathbb{G}_t$. Note that the size of $\mathbb{F}_q$ is exponential in λ.

$\mathsf{Setup}(1^\lambda) :$ $(\ \mathsf{pk}^{\mathsf{IPG}} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, g_t, \hat{g}_t, e_t)_{t=0,1}, \mathbb{G}_T),\ \mathsf{sk}^{\mathsf{IPG}} := \phi_1\ ) \xleftarrow{\mathsf{R}} \mathsf{Gen}^{\mathsf{IPG}}(1^\lambda, 1)$,
generate a random hash $H_t : \mathbb{F}_q \to \mathbb{G}_t$ with the identity space $\mathbb{F}_q$,
return $\mathsf{pk} := ((\mathbb{G}_t, \hat{\mathbb{G}}_t, \hat{g}_t, e_t, H_t)_{t=0,1},\ \mathbb{G}_T), \quad \mathsf{sk} := \phi_1$.

$\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, (\mathsf{ID}_i)_{i\in[j]}) :$ for $i \in [j-1]$, $r_i \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \hat{d}_i := \hat{g}_1^{r_i} \in \hat{\mathbb{G}}_1$,
$h_1 := H_0(\mathsf{ID}_1) \in \mathbb{G}_0$, for $i \in [2, j]$, $h_i := H_1(\mathsf{ID}_1, \dots, \mathsf{ID}_i) \in \mathbb{G}_1$,

$$d_j := \phi_1\,(h_1) \cdot \prod_{i=2}^{j} h_i^{r_{i-1}}, \quad \text{return } \mathsf{sk}_{(\mathsf{ID}_i)_{i\in[j]}} := ((\hat{d}_i \in \hat{\mathbb{G}}_1)_{i\in[j-1]},\ d_j \in \mathbb{G}_1).$$

$\mathsf{Enc}(\mathsf{pk}, m, (\mathsf{ID}_i)_{i\in[j]}) :$ $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times, \quad \hat{c}_0 := \hat{g}_1^\zeta$,
for $i \in [2, j]$, $h_i := H_1(\mathsf{ID}_1, \dots, \mathsf{ID}_i) \in \mathbb{G}_1,\ c_i := h_i^\zeta$,
$h_1 := H_0(\mathsf{ID}_1) \in \mathbb{G}_0, \quad z := e_0(h_1, \hat{g}_0)^\zeta, \quad c_T := z \cdot m$,
return $\mathsf{ct}_{(\mathsf{ID}_i)_{i\in[j]}} := (\ \hat{c}_0 \in \hat{\mathbb{G}}_1,\ (c_i \in \mathbb{G}_1)_{i\in[2,j]},\ c_T\ )$.

$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{(\mathsf{ID}_i)_{i\in[j]}} = ((\hat{d}_i), d_j),\ \mathsf{ct}_{(\mathsf{ID}'_i)_{i\in[j]}} = (\hat{c}_0, (c_i), c_T)) :$

$$\text{if } (\mathsf{ID}_i) = (\mathsf{ID}'_i),\ z' := \frac{e_1(d_j, \hat{c}_0)}{\prod_{i=2}^{j} e_1(c_i, \hat{d}_{i-1})},$$

$m' := c_T \cdot (z')^{-1}$, return m', otherwise, return $\perp$.

$\mathsf{Delegate}_j(\mathsf{pk}, \mathsf{sk}_{(\mathsf{ID}_i)_{i\in[j]}} = ((\hat{d}_i), d_j),\ \ \mathsf{ID}_{j+1}) :$
for $i \in [j]$, $r'_i \xleftarrow{\mathsf{U}} \mathbb{F}_q$, for $i \in [j-1]$, $\hat{d}'_i := \hat{d}_i \cdot \hat{g}_1^{r'_i} \in \hat{\mathbb{G}}_1, \quad \hat{d}'_j := \hat{g}_1^{r'_j} \in \hat{\mathbb{G}}_1$,
for $i \in [2, j+1]$, $h_i := H_1(\mathsf{ID}_1, \dots, \mathsf{ID}_i) \in \mathbb{G}_1$,

$$d'_{j+1} := d_j \cdot \prod_{i=2}^{j+1} h_i^{r'_{i-1}}, \quad \text{return } \mathsf{sk}_{(\mathsf{ID}_i)_{i\in[j+1]}} := ((\hat{d}'_i \in \hat{\mathbb{G}}_1)_{i\in[j]},\ d'_{j+1} \in \mathbb{G}_1).$$

Dec correctly decrypts since

$$\begin{aligned} z' &= \frac{e_1(d_j, \hat{c}_0)}{\prod_{i=2}^{j} e_1(c_i, \hat{d}_{i-1})} = \frac{e_1(\phi_1\,(h_1) \cdot \prod_{i=2}^{j} h_i^{r_{i-1}},\ \hat{g}_1^\zeta)}{\prod_{i=2}^{j} e_1(h_i^\zeta, \hat{g}_1^{r_{i-1}})} = e_1(\phi_1(h_1), \hat{g}_1)^\zeta \\ &= e_0(h_1, \hat{g}_0)^\zeta = z \end{aligned}$$

if $(\mathsf{ID}_i) = (\mathsf{ID}'_i)$. Here, we use the compatibility $e_1(\phi_1(h_1), \hat{g}_1) = e_0(h_1, \hat{g}_0)$ in Definition 4.

C.2 Security

Theorem 5. *The proposed HIBE scheme is PH-PQ secure under the 1-qIsog-DBDH assumption in the quantum random oracle model.*

For any adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, there exists an adversary $\mathcal{B} := (\mathcal{B}_1, \mathcal{B}_2)$ for the 1-qIsog-DBDH problem, where $\mathcal{B}_1$ is a polynomial-time quantum machine and $\mathcal{B}_2$ is a classical ppt machine, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{hibe,ph\text{-}pq}}(\lambda) \leq \nu_2 \cdot \mathsf{Adv}_{\mathcal{B}}^{\mathsf{1\text{-}qIsog\text{-}DBDH}}(\lambda)$, where ν_2 is the maximum number of the random oracle queries of $\mathcal{A}_2$.

The proof is given in the full version of this paper.

Remark 5. Our HIBE scheme can be modified to be level-1 anonymous ID secure against pre-challenge quantum adversaries as in the modified GS-HIBE given in Section 5.3 in [1].

D Anonymity of Our IBE : Proof of Theorem 2

First, we define the security notion of Anonymous-ID secure against Pre-challenge Quantum adversary (AI-PQ) for IBE.

Definition 17 (AI-PQ for IBE). *Let* $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *be an IBE scheme and let* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ *be a stateful adversary, where* $\mathcal{A}_1$ *is modeled as a polynomial-time quantum adversary. Consider the experiment* $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{ibe,ai\text{-}pq}}[\lambda]$ *below:*

$$
\begin{aligned}
&\mathbf{Exp}_{\mathcal{A}}^{\mathsf{ibe,ai\text{-}pq}}[\lambda] : \ (\mathsf{sk}, \mathsf{pk}) \overset{\mathsf{R}}{\leftarrow} \mathsf{Setup}(1^{\lambda}), \ \ \mathsf{state} \overset{\mathsf{R}}{\leftarrow} \mathcal{A}_1^{\mathsf{RO}(\cdot),\, \mathsf{KeyGen}(\mathsf{sk},\cdot)}(\mathsf{pk}), \\
&\ (\mathsf{ID}^{(0)}, \mathsf{ID}^{(1)}, m^{(0)}, m^{(1)}) \overset{\mathsf{R}}{\leftarrow} \mathcal{A}_2^{\mathsf{RO}(\cdot),\, \mathsf{KeyGen}(\mathsf{sk},\cdot)}(\mathsf{state}), \quad b \overset{\mathsf{U}}{\leftarrow} \{0,1\}, \\
&\ \mathsf{ct}^* \overset{\mathsf{R}}{\leftarrow} \mathsf{Enc}(\mathsf{pk}, m^{(b)}, \mathsf{ID}^{(b)}), \ b' \overset{\mathsf{R}}{\leftarrow} \mathcal{A}_2^{\mathsf{RO}(\cdot),\, \mathsf{KeyGen}(\mathsf{sk},\cdot)}(\mathsf{ct}^*), \quad \text{output } b'. \\
&\quad /* \text{ None of } \mathsf{ID}^{(0)} \text{ and } \mathsf{ID}^{(1)} \text{ are queried to } \mathsf{RO} \text{ in the pre-challenge phase by } \mathcal{A}_1, \\
&\qquad\quad \text{and not queried to } \mathsf{KeyGen} \text{ in any phase } */
\end{aligned}
$$

Here, RO *is quantum-accessible (i.e., with quantum superposed inputs and outputs) and* KeyGen *is classical-accessible. If* $b = b'$, $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$ *wins. The advantage of adversary* $\mathcal{A}$ *in the experiment is defined as* $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ibe,ai\text{-}pq}}(\lambda) := \Pr[\mathcal{A} \text{ wins}] - 1/2$ *for any security parameter* λ. *An IBE scheme is* anonymous-ID secure against pre-challenge quantum adversary (AI-PQ) *if all adversaries* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, *where* $\mathcal{A}_1$ *be a polynomial-time quantum machine and* $\mathcal{A}_2$ *a classical ppt machine, achieve at most a negligible advantage in the above security game (or experiment).*

Proof Sketch of Theorem 2. Theorem 2 is proven in a manner similar to Lemma 4.3 in [1]. We have

$$\begin{aligned}
&\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ibe,ai\text{-}pq}}(\lambda) = \Pr[\mathcal{A} \text{ wins}] - 1/2\\
&\quad = \Pr[b' = 1 \land b = 1 \text{ in } \mathbf{Exp}_{\mathcal{A}}^{\mathsf{ibe,ai\text{-}pq}}] + \Pr[b' = 0 \land b = 0 \text{ in } \mathbf{Exp}_{\mathcal{A}}^{\mathsf{ibe,ai\text{-}pq}}] - 1/2\\
&\quad = 1/2 \cdot (\Pr[b' = 1|b = 1 \text{ in } \mathbf{Exp}_{\mathcal{A}}^{\mathsf{ibe,ai\text{-}pq}}] + \Pr[b' = 0|b = 0 \text{ in } \mathbf{Exp}_{\mathcal{A}}^{\mathsf{ibe,ai\text{-}pq}}]) - 1/2\\
&\quad = 1/2 \cdot (\Pr[b' = 1|b = 1] + (1 - \Pr[b' = 1|b = 0])) - 1/2\\
&\quad = 1/2 \cdot (\Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0]).
\end{aligned}$$

Therefore, we will show that $\mathsf{Adv}'_{\mathcal{A}}(\lambda) := \Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0]$ for the experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{ibe,ai\text{-}pq}}$ is negligible. First, we note that the proof of Theorem 1 shows that the adversary $\mathcal{A}$ has only negligible advantage in "the real-or-random game", where $\mathcal{A}$'s task is to distinguish the real ciphertext of the message that $\mathcal{A}$ asked or the random ciphertext of a random message. Then, we can make a standard hybrid argument for the indistinguishability of $\mathsf{Adv}'_{\mathcal{A}}$ as

$$\begin{aligned}
&\mathsf{Adv}'_{\mathcal{A}}(\lambda) = \Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0]\\
&\quad = \Pr[b' = 1|b = 1]\\
&\qquad\quad - \Pr[b' = 1| \text{ challenge ciphertext is for a random message and } \mathrm{ID}_1\ ]\\
&\qquad + \Pr[b' = 1| \text{ challenge ciphertext is for a random message and } \mathrm{ID}_1\ ]\\
&\qquad\quad - \Pr[b' = 1| \text{ challenge ciphertext is for a random message and } \mathrm{ID}_0\ ]\\
&\qquad + \Pr[b' = 1| \text{ challenge ciphertext is for a random message and } \mathrm{ID}_0\ ]\\
&\qquad\quad - \Pr[b' = 1|b = 0]\\
&\quad \le \mathsf{Adv}_{\mathcal{A}}^{\mathsf{ibe,ph\text{-}pq}}(\lambda) + \mathsf{Adv}''_{\mathcal{A}}(\lambda) + \mathsf{Adv}_{\mathcal{A}}^{\mathsf{ibe,ph\text{-}pq}}(\lambda),
\end{aligned}$$

where

$$\begin{aligned}
&\mathsf{Adv}''_{\mathcal{A}}(\lambda) := \Pr[b' = 1| \text{ challenge ciphertext is for a random message and } \mathrm{ID}_1\ ]\\
&\qquad - \Pr[b' = 1| \text{ challenge ciphertext is for a random message and } \mathrm{ID}_0\ ].
\end{aligned}$$

Here, note that the ciphertext of our IBE has the form of

$$\mathsf{ct}_{\mathsf{ID}} := (c := \hat{g}_1^{\zeta},\ c_T := e_0(H(\mathsf{ID}), \hat{g}_0)^{\zeta} \cdot m),$$

with $\zeta \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$. Thus, if the encrypted message m is random and hidden, then c_T is also random and perfectly hides $H(\mathsf{ID})$ from the adversary's view. Therefore, $\mathsf{Adv}''_{\mathcal{A}}(\lambda) = 0$ and $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ibe,ai\text{-}pq}}(\lambda)\,(= 1/2 \cdot \mathsf{Adv}'_{\mathcal{A}}(\lambda))$ is negligible in λ under the 1-qIsog-DBDH assumption. This completes the proof of Theorem 2. □