SQIsign2D-East: A New Signature Scheme Using 2-Dimensional Isogenies

Kohei Nakagawa[1(✉)], Hiroshi Onuki[2], Wouter Castryck[3], Mingjie Chen[3], Riccardo Invernizzi[3], Gioella Lorenzon[3], and Frederik Vercauteren[3]

[1] NTT Social Informatics Laboratories, Tokyo, Japan
kohei.nakagawa@ntt.com
[2] The University of Tokyo, Tokyo, Japan
[3] COSIC, ESAT, KU Leuven, Leuven, Belgium

Abstract. Isogeny-based cryptography is cryptographic schemes whose security is based on the hardness of a mathematical problem called the isogeny problem, and is attracting attention as one of the candidates for post-quantum cryptography. A representative isogeny-based cryptography is the signature scheme called SQIsign, which was submitted to the NIST PQC standardization competition. SQIsign has attracted much attention because of its very short signature and key size among the candidates for the NIST PQC standardization. Recently, a lot of new schemes have been proposed that use high-dimensional isogenies. Among them, the signature scheme called SQIsignHD has an even shorter signature size than SQIsign. However, it requires 4-dimensional isogeny computations for the signature verification. In this paper, we propose a new signature scheme, SQIsign2D-East, which requires only two-dimensional isogeny computations for verification, thus reducing the computational cost of verification though increasing the signing cost. First, we generalized an algorithm called RandIsogImg, which computes a random isogeny of non-smooth degree. Then, by using this generalized RandIsogImg, we construct a new signature scheme SQIsign2D-East.

1 Introduction

In recent years, isogeny-based cryptography has been actively studied as one of the candidates for post-quantum cryptography (PQC). One of the representative isogeny-based cryptographies is the signature scheme called SQIsign [13], which was submitted to the NIST PQC standardization competition. SQIsign has attracted much attention because of its very short signature and key size among the candidates for the NIST PQC standardization. Another well-known isogeny-based cryptography is SIDH [20], which is proposed by De Feo and Jao. Additionally, SIKE [1], a key encapsulation scheme based on SIDH, remained

Initially, it was a paper by Nakagawa and Onuki, but the security issue described in Sect. 4.3 were pointed out by Wouter Castryck, Mingjie Chen, Riccardo Invernizzi, Gioella Lorenzon and Frederik Vercauteren and a solution was also proposed by them. Therefore, we marged these results into a single paper.

K.-M. Chung and Y. Sasaki (Eds.): ASIACRYPT 2024, LNCS 15486, pp. 272–303, 2025.
https://doi.org/10.1007/978-981-96-0891-1_9

an alternative candidate for the NIST PQC standardization competition until Round 4. However, recent attacks [5,24,28] broke the security of SIDH and SIKE. These attacks find the secret isogeny from the two point images under the isogeny by computing high dimensional isogenies.

In response, a number of cryptographic applications of attacks on SIDH have been studied, such as SQIsignHD [11], FESTA [3], QFESTA [26] SCALLOP-HD [7], and IS-CUBE [25]. Among them, SQIsignHD is a variant of SQIsign that has a shorter signature size and higher signing performance than SQIsign. However, it requires 4-dimensional isogeny computations for signature verification, which leads to a large computational cost. Since NIST calls for signature schemes that have short signatures and fast verification, reducing the verification cost of SQIsignHD is an important issue.

1.1 Contributions

In this paper, we make the following contributions:

1. We construct a new algorithm `GenRandIsogImg`, which is a generalization of the algorithm called `RandIsogImg` proposed in [26], which computes the codomain and point images of a given degree isogeny from a *special* elliptic curve E_0. Our `GenRandIsogImg` computes the codomain and point images of a given degree isogeny from *any* elliptic curve E.
2. Using `GenRandIsogImg` as a building block, we propose a new variant of SQIsignHD, which only requires 2-dimensional isogeny computations for the verification. We name this signature scheme 'SQIsign2D-East'.
3. We give concrete parameters of SQIsign2D for the NIST security level 1, 3, and 5. Under these parameter settings, we analyse the signature sizes and show that our signature sizes are smaller than SQIsign and larger than SQIsignHD.
4. We analyse the computational cost of SQIsign2D-East under the parameter for the NIST security level 1 and show that the verification cost of SQIsign2D is smaller than that of SQIsignHD though the signing cost is larger.

1.2 Related Works

At the same time as this work, [2] and [17] also proposed a variant of SQIsignHD based on 2-dimensional isogenies. The former is called 'SQIsign2D-West' and the later is called 'SQIPrime'. These protocols are similar to ours, but they were proposed independently of us. Our protocol has a stronger security assumption than their protocol but seems to be more efficient. We leave the comparison with their protocol as future work.

Recently, [27] proposed an algorithm called `IdealToIsogenyIQO` that makes the key generation and the signing procedure in SQIsign at least twice as fast. However, their costs are still larger than SQIsignHD and SQIsign2D-East as described in their paper.

1.3 Organization

In Sect. 2, we give some notation and background knowledge used in our protocol. In Sect. 3, we construct a generalized `RandIsogImg`. In Sect. 4, we propose our new signature scheme SQIsign2D-East and its security is analysed in Sect. 5. In Sect. 6, we give some concrete parameters for SQIsign2D-East and analyse the data size and the computational cost of SQIsign2D-East. Finally, in Sect. 7, we give the conclusion of this paper.

2 Preliminaries

In this section, we summarize some background knowledge used in our protocol.

2.1 Notation

Throughout this paper, we use the following notation. We let p be a prime number of cryptographic size, i.e., p is at least about 2^{256} and let λ be a security parameter. Let $f(x)$ and $g(x)$ be real functions. We write $f(x) = O(g(x))$ if there exists a constant $c \in \mathbb{R}$ such that $f(x)$ is bounded by $c \cdot g(x)$ for sufficiently large x. For a finite set S, we write $x \in_U S$ if x is sampled uniformly at random from S. Let $\perp$ be the symbol indicating failure of an algorithm.

2.2 Abelian Varieties and Isogenies

In this paper, we mainly use principally polarized superspecial abelian varieties of dimension one or two defined over a finite field of characteristic p. Such a variety is isomorphic to a supersingular elliptic curve, the product of two supersingular elliptic curves, or a Jacobian of a superspecial hyperelliptic curve of genus two, and always has a model defined over $\mathbb{F}_{p^2}$. Therefore, we only consider varieties defined over $\mathbb{F}_{p^2}$.

Basic Facts. An *isogeny* is a rational map between abelian varieties which is a surjective group homomorphism and has finite kernel. The *degree* of an isogeny φ is its degree as a rational map and is denoted by $\deg \varphi$. An isogeny φ is *separable* if $\#\ker \varphi = \deg \varphi$. A separable isogeny is uniquely determined by its kernel up to post-composition of an isomorphism. For an isogeny $\varphi : A \to B$ between principally polarized abelian varieties, there exists a unique *dual isogeny* $\hat{\varphi}$ such that $\hat{\varphi} \circ \varphi$ is equal to the multiplication-by-$\deg \varphi$ map on A.

Let $\varphi : A \to B$, $\psi : A \to C$, and $\psi' : B \to D$ be isogenies such that $\deg \varphi$ is coprime to $\deg \psi$. If $\ker \psi' = \varphi(\ker \psi)$ holds, we say that ψ' is the push-forward of ψ by φ and denote it by $\psi' = [\varphi]_* \psi$. Under the same situation, we say that ψ is the pull-back of ψ' by φ and denote it by $\psi = [\varphi]^* \psi$.

Let A and B be principally polarized abelian varieties. If there exists an isogeny between A and B then the dimensions of A and B are the same. If A is

superspecial then there exists an isogeny between A and B if and only if B is a superspecial abelian variety of the same dimension as A.

Let A be a principally polarized abelian variety and ℓ a positive integer. An *ℓ-isotropic subgroup* of A is a subgroup of the ℓ-torsion subgroup $A[\ell]$ of A on which the ℓ-Weil pairing is trivial. An ℓ-isotropic subgroup G is *maximal* if there is no other ℓ-isotropic subgroup containing G. A separable isogeny whose kernel is a maximal ℓ-isotropic subgroup is called an *ℓ-isogeny* if the dimension of the domain is one or an *(ℓ, ℓ)-isogeny* if the dimension of the domain is two.

Let E be an elliptic curve defined over $\mathbb{F}_{p^2}$. Among the isomorphism class of E, we can chose a Montgomery curve as a canonical representative by using [6, Algorithm 1]. We call this curve the *normalized curve* of E. In this paper, we assume that all elliptic curves are normalized. Moreover, we can compute a canonical basis of the n-torsion subgroup $E[n]$ defined over $\mathbb{F}_{p^2}$ by using [6, Algorithm 3]. Especially when $n = 2^k$ for a positive integer k, we can compute a canonical basis of $E[n]$ by the algorithm proposed in [9, Section 5.1].

Computing Isogenies. Let A be a principally polarized abelian variety, ℓ a positive integer, and G a maximal ℓ-isotropic subgroup of A.

If the dimension of A is one then we can compute an ℓ-isogeny φ with kernel G by Vélu's formulas [32]. More precisely, given A, ℓ, G, Vélu's formulas give a method to compute the codomain of φ in $O(\ell)$ operations on a field containing the points in G. In addition, for additional input $P \in A$, we can compute $\varphi(P)$ in $O(\ell)$ operations on a field containing the points in G and P. These computational costs are improved to $\tilde{O}(\sqrt{\ell})$ by Bernstein, De Feo, Leroux, and Smith [4].

For an isogeny $\varphi : A \to B$, we say that information $\mathcal{I}_\varphi$ is an *efficient representation* of φ when we can compute $\varphi(P)$ in polynomial time from a given point $P \in A$ and the information $\mathcal{I}_\varphi$. For example, the tuple (A, ℓ, G) described above is an efficient representation of ℓ-isogeny $\varphi : A \to B$ when ℓ is smooth.

If A is the Jacobian of a hyperelliptic curve of genus two and $\ell = 2$ then we can compute $(2, 2)$-isogeny by formulas in Smith's Ph.D thesis [30]. Formulas of $(2, 2)$-isogenies for the case A is the product of two elliptic curves is given by Howe, Leprévost, and Poonen [19]. In 2023, more efficient formulas of $(2, 2)$-isogenies is proposed by Dartois, Maino, Pope, and Robert [12]. In addition, an efficient formulas of $(3, 3)$-isogenies is proposed by Corte-Real Santos, Costello and Smith [29]. An algorithm to compute (ℓ, ℓ)-isogenies for a general ℓ was given by [10] and later improved by [23]. The computational cost of this algorithm is $O(\ell^2)$ operations on a field containing the points in G.

2.3 Quaternion Algebras and the Deuring Correspondence

Quaternion Algebras. A *quaternion algebra* over $\mathbb{Q}$ is a division algebra defined by $\mathbb{Q} + \mathbb{Q}\mathbf{i} + \mathbb{Q}\mathbf{j} + \mathbb{Q}\mathbf{k}$ and $\mathbf{i}^2 = a, \mathbf{j}^2 = b, \mathbf{ij} = -\mathbf{ji} = \mathbf{k}$ for $a, b \in \mathbb{Q}^*$. We denote it by $H(a, b)$. We say $H(a, b)$ is *ramified* at a place v of $\mathbb{Q}$ if $H(a, b) \otimes_{\mathbb{Q}} \mathbb{Q}_v$ is not isomorphic to the algebra of the 2×2 matrices over $\mathbb{Q}_v$. There exists a quaternion algebra ramified exactly at p and ∞. Such an algebra is unique up to isomorphism. We denote it by $\mathcal{B}_{p,\infty}$.

Let $\alpha = x + y\mathbf{i} + z\mathbf{j} + t\mathbf{k} \in H(a,b)$ with $x, y, z, t \in \mathbb{Q}$. The *conjugate* of α is $x - y\mathbf{i} - z\mathbf{j} - t\mathbf{k}$ and denoted by $\bar{\alpha}$. The *reduced norm* of α is $\alpha\bar{\alpha}$ and denoted by $n(\alpha)$.

An *order* $\mathcal{O}$ of $H(a,b)$ is a subring of $H(a,b)$ that is also a $\mathbb{Z}$-lattice of rank 4. This means that $\mathcal{O} = \mathbb{Z}\alpha_1 + \mathbb{Z}\alpha_2 + \mathbb{Z}\alpha_3 + \mathbb{Z}\alpha_4$ for a basis $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ of $H(a,b)$. We denote such an order by $\mathbb{Z}\langle\alpha_1, \alpha_2, \alpha_3, \alpha_4\rangle$. An order $\mathcal{O}$ is said to be *maximal* if there is no larger order that contains $\mathcal{O}$.

For a maximal order $\mathcal{O}$, an (integral) *left $\mathcal{O}$-ideal* I is a $\mathbb{Z}$-lattice of rank 4 satisfying $I \subset \mathcal{O}$ and $\mathcal{O} \cdot I \subset I$. An *right $\mathcal{O}$-ideal* is similarly defined. For an ideal I, we denote its conjugate by $\bar{I} = \{\bar{\alpha} \mid \alpha \in I\}$. We denote by $n(I)$ the *reduced norm* of ideal I, defined as (the unique positive generator of) the $\mathbb{Z}$-module generated by the reduced norms of the elements of I. A left $\mathcal{O}$-ideal I of integer norm can be written as $I = \mathcal{O}\alpha + \mathcal{O}n(I)$ for some $\alpha \in I$. We denote such I by $I = \mathcal{O}\langle\alpha, n(I)\rangle$. The *ideal equivalence* denoted by $I \sim J$ means that there exists $\beta \in \mathcal{B}^*_{p,\infty}$ such that $I = J\beta$.

Deuring Correspondence. Deuring [16] showed that the endomorphism ring of a supersingular elliptic curve over $\mathbb{F}_{p^2}$ is isomorphic to a maximal order of $\mathcal{B}_{p,\infty}$ and gave a correspondence (the *Deuring correspondence*) where a supersingular elliptic E curve over $\mathbb{F}_{p^2}$ corresponding to a maximal order isomorphic to $\mathrm{End}(E)$.

Suppose $p \equiv 3 \pmod 4$. This is the setting we use in our protocol. Then we can take $\mathcal{B}_{p,\infty} = H(-1,-p)$ and an elliptic curve over $\mathbb{F}_{p^2}$ with j-invariant 1728 is supersingular. Let E_0 be the elliptic curve over $\mathbb{F}_{p^2}$ defined by $y^2 = x^3 + x$. Then $j(E_0) = 1728$, so E_0 is supersingular. We define endomorphisms $\iota : (x,y) \mapsto (-x, \sqrt{-1}y)$ and $\pi : (x,y) \mapsto (x^p, y^p)$ of E_0, where $\sqrt{-1}$ is a fixed square root of -1 in $\mathbb{F}_{p^2}$. The endomorphism ring of E_0 is isomorphic to $\mathcal{O}_0 := \mathbb{Z}\langle 1, \mathbf{i}, \frac{\mathbf{i}+\mathbf{j}}{2}, \frac{1+\mathbf{k}}{2}\rangle$. This isomorphism is given by $\iota \mapsto \mathbf{i}$ and $\pi \mapsto \mathbf{j}$. From now on, we identify $\mathrm{End}(E_0)$ with $\mathcal{O}_0$ by this isomorphism.

Some isogeny-based protocols, e.g., SQISign [13], need to compute the image under an element in $\mathcal{O}_0$ represented by the coefficients with respect to the basis $(1, \mathbf{i}, \frac{\mathbf{i}+\mathbf{j}}{2}, \frac{1+\mathbf{k}}{2})$. Let $P \in E_0(\mathbb{F}_{p^2})$ and $\alpha = x + y\mathbf{i} + z\frac{\mathbf{i}+\mathbf{j}}{2} + t\frac{1+\mathbf{k}}{2}$ for $x, y, z, t \in \mathbb{Z}$. Given P and x, y, z, t, one can compute $\alpha(P)$ in $O(\log \max\{|x|, |y|, |z|, |t|\})$ operations on $\mathbb{F}_{p^2}$ and $O(\log p)$ operations on $\mathbb{F}_{p^4}$. The latter operations on $\mathbb{F}_{p^4}$ is necessary only for the case when the order of P is even. We need to compute $\alpha(P_0)$ and $\alpha(Q_0)$ for a fixed basis P_0, Q_0 of $E_0[2^a]$ for some integer a in our protocol. In this case, by precomputing the images of P_0 and Q_0 under $\mathbf{i}$, $\frac{\mathbf{i}+\mathbf{j}}{2}$, and $\frac{1+\mathbf{k}}{2}$, we can compute $\alpha(P_0)$ and $\alpha(Q_0)$ by scalar multiplications by x, y, z, t and additions.

The Deuring correspondence also gives a correspondence between isogenies and ideals. Let E_1 be a supersingular elliptic curve over $\mathbb{F}_{p^2}$ and let $\mathcal{O}_1$ be a maximal order of $\mathcal{B}_{p,\infty}$ such that $\mathcal{O}_1 \cong \mathrm{End}(E_1)$. Let $\phi : E_1 \to E_2$ be an N-isogeny, then the isogeny ϕ can be associated to a left $\mathcal{O}_1$-ideal I_ϕ. This ideal I_ϕ is also a right $\mathcal{O}_2$-ideal for a maximal order $\mathcal{O}_2$ satisfying $\mathcal{O}_2 \cong \mathrm{End}(E_2)$. Such an ideal I_ϕ is called a *connecting ideal* from $\mathcal{O}_1$ to $\mathcal{O}_2$. Furthermore, it is

known that its norm $n(I_\phi)$ is equal to the degree N of ϕ. The order $\mathfrak{O}$ denoted by $\mathfrak{O} = \mathcal{O}_1 \cap \mathcal{O}_2$ is called the *Eichler order* and $\mathfrak{O} = \mathbb{Z} + I_\phi$ holds. Moreover, two isogenies $\phi, \psi : E_1 \to E_2$ that have the same domain and codomain correspond to equivalent ideals $I_\phi \sim I_\psi$.

Let I_τ be a connecting ideal of norm d from $\mathcal{O}_0 \cong \mathrm{End}(E_0)$ to $\mathcal{O}_1 \cong \mathrm{End}(E_1)$ and let $\tau : E_0 \to E_1$ be the corresponding isogeny. In our protocol, we need to compute the image under an endomorphism $\alpha_1 \in \mathrm{End}(E_1)$ represented as an element $\alpha \in \mathcal{O}_0 \cap \mathcal{O}_1$. Since $\alpha \in \mathcal{O}_0$, we can compute the image under the corresponding endomorphism $\alpha_0 \in \mathrm{End}(E_0)$ as described above. Then, if the order n of $P \in E_1$ is coprime to d, we can compute $\alpha_1(P)$ as follows:

$$\alpha_1(P) = \frac{1}{d}\tau \circ \alpha_0 \circ \hat{\tau}(P),$$

where $\dfrac{1}{d}$ is an inverse of d modulo n.

Algorithms Using Quaternion Algebras. As in the above, we let $\mathcal{O}_0$ be the maximal order of $\mathcal{B}_{p,\infty}$ with basis $(1, \mathbf{i}, \frac{\mathbf{i}+\mathbf{j}}{2}, \frac{1+\mathbf{k}}{2})$. Here, we introduce some existing algorithms using quaternion algebras necessary for the construction of our SQIsign2D-East. These algorithms are used in SQISign (see the official document [6] for details).

- $\texttt{FullRepresentInteger}_{\mathcal{O}_0}(M)$: Take an integer $M > p$ as input, output $\alpha \in \mathcal{O}_0$ such that $n(\alpha) = M$.
- $\texttt{EichlerModConstraint}(I, \gamma, \delta)$: Take a left-$\mathcal{O}_0$ ideal I of prime norm N and $\gamma, \delta \in \mathcal{O}_0$ as input, output $(C_0 : D_0) \in \mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$ such that $\gamma(C_0\mathbf{j} + D_0\mathbf{k})\delta \in \mathbb{Z} + I$.
- $\texttt{StrongApproximation}_M(N, C_0, D_0)$: Take integers M, N, C_0 and D_0 as input, output $\mu \in \mathcal{O}_0$ such that $n(\mu) = M$ and $\mu = m(C_0\mathbf{j} + D_0\mathbf{k}) + N\mu_1$, where $m \in \mathbb{Z}$ and $\mu_1 \in \mathcal{O}_0$.

2.4 Computing Isogenies of Dimension One from Dimension Two

In this subsection, we give an algorithm to compute isogenies of dimension one by using an isogeny of dimension two, which is an important sub-algorithm for our protocol. This algorithm comes from recent attacks to SIDH by [5,24,28]. We use the following theorem, which is based on Kani's criterion [21].

Theorem 1 ([24, Theorem 1]). *Let N_1, N_2, and D be pairwise coprime integers such that $D = N_1 + N_2$, and let E_0, E_1, E_2, and E_3 be elliptic curves connected by the following diagram of isogenies:*

$$\begin{array}{ccc} E_0 & \xrightarrow{\psi_2} & E_2 \\ {\scriptstyle\psi_1}\downarrow & \overset{f}{\nearrow} & \downarrow{\scriptstyle\psi_1'} \\ E_1 & \xrightarrow[\psi_2']{} & E_3, \end{array}$$

where $\psi_2' \circ \psi_1 = \psi_1' \circ \psi_2$, $f = \psi_2 \circ \hat{\psi}_1$, $\deg(\psi_1) = \deg(\psi_1') = N_1$, *and* $\deg(\psi_2) = \deg(\psi_2') = N_2$. *Then, the isogeny*

$$\Phi = \begin{pmatrix} \hat{\psi}_1 & -\hat{\psi}_2 \\ \psi_2' & \psi_1' \end{pmatrix} : E_1 \times E_2 \to E_0 \times E_3 \tag{1}$$

is a (D, D)*-isogeny with respect to the natural product polarizations on* $E_1 \times E_2$ *and* $E_0 \times E_3$, *and has kernel* $\{([N_2]P, f(P)) \mid P \in E_1[D]\}$.

Conversely, a (D, D)-isogeny with kernel $\{([N_2]P, f(P)) \mid P \in E_1[D]\}$ is of the form $\iota \circ \Phi$ with an isomorphism ι from $E_0 \times E_3$. To construct algorithms to evaluate the isogenies in the matrix in Eq. (1), we need to restrict the possibility of ι. In particular, we assume that the codomain E_3 of ψ_1' and ψ_2' is not isomorphic to E_0. This assumption is plausible because there exist about $p/12$ supersingular elliptic curves over $\mathbb{F}_{p^2}$ up to isomorphism and ψ_1' seems to be a random isogeny unless we intend to have $E_1 \cong E_3$. Under this assumption, an isomorphism from $E_0 \times E_3$ is represented by $\begin{pmatrix} \iota_0 & 0 \\ 0 & \iota_3 \end{pmatrix}$ or $\begin{pmatrix} 0 & \iota_3 \\ \iota_0 & 0 \end{pmatrix}$, where ι_0 is an isomorphism from E_0 and ι_3 is an isomorphism from E_3. Since we assume that E_0 and E_3 are normalized, we can determine the codomain of Φ in only two ways: $E_0 \times E_3$ or $E_3 \times E_0$.

Using Theorem 1 and assuming the above assumption, we construct an algorithm to evaluate the isogenies in the matrix in Equation (1) by computing a (D, D)-isogeny. We denote the algorithm by `KaniCod`.

Let N_1, N_2 be integers coprime with each other and $D = N_1 + N_2$. Let E_1, E_2 supersingular elliptic curves over $\mathbb{F}_{p^2}$, (P_1, Q_1) a basis of $E_1[D]$, (P_2, Q_2) a basis of $E_2[D]$, S_1 a finite subset of E_1, and S_2 a finite subset of E_2. If there exist isogenies $\psi_1 : E_0 \to E_1$ and $\psi_2 : E_0 \to E_2$ such that $\deg \psi_1 = N_1$ $\deg \psi_2 = N_2$, $P_2 = \psi_2 \circ \hat{\psi}_1(P_1)$, and $Q_2 = \psi_2 \circ \hat{\psi}_1(Q_1)$, then `KaniCod` with input $(N_1, N_2, E_1, E_2, P_1, Q_1, P_2, Q_2; S_1; S_2)$ returns the curve E_0, the image of S_1 under $\hat{\psi}_1$, and the image of S_2 under $\hat{\psi}_2$. If such isogenies do not exist then `KaniCod` returns $\perp$. The procedure for `KaniCod` is as follows:

1. Compute a (D, D)-isogeny Φ with kernel $\langle([N_2]P_1, P_2), ([N_2]Q_1, Q_2)\rangle$.
2. If the codomain of Φ is not the product of elliptic curves then return $\perp$.
3. Otherwise let $F_1 \times F_2$ be the codomain of Φ.
4. Let P_1' and Q_1' be first components of $\Phi((P_1, O_{E_2}))$ and $\Phi((Q_1, O_{E_2}))$.
5. Compute the D-Weil pairings $e_D(P_1, Q_1)$ and $e_D(P_1', Q_1')$.
6. If $e_D(P_1, Q_1)^{N_1} = e_D(P_1', Q_1')$ then return F_1 and the first components of $\Phi((R_1, O_{E_2}))$ and $\Phi((O_{E_1}, R_2))$ for $R_1 \in S_1$ and $R_2 \in S_2$.
7. If $e_D(P_1, Q_1)^{N_2} = e_D(P_1', Q_1')$ then return F_2 and the second components of $\Phi((R_1, O_{E_2}))$ and $\Phi((O_{E_1}, R_2))$ for $R_1 \in S_1$ and $R_2 \in S_2$.
8. Otherwise, return $\perp$.

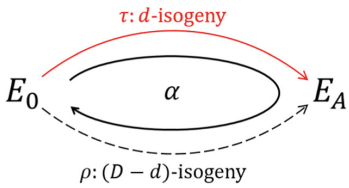
When D is smooth, $P_1, Q_1 \in E_1(\mathbb{F}_{p^2})$, $S_1 \subset E_1(\mathbb{F}_{p^2})$, $P_2, Q_2 \in E_2(\mathbb{F}_{p^2})$, and $S_2 \subset E_2(\mathbb{F}_{p^2})$ the computational costs of `KaniCod` are $O((\#S_1 + \#S_2) \log D)$ operations on $\mathbb{F}_{p^2}$ by using the methods stated in Sect. 2.2. Especially, D is a power of 2 in our case.

2.5 RandIsogImg

Here, we recall the algorithm RandIsogImg which evaluates the codomain of a random isogeny of *non-smooth* degree and some point images under the isogeny. This algorithm was proposed in the paper of QFESTA [26] and is an important component of our SQIsign2D-East.

Let E_0 be the elliptic curve over $\mathbb{F}_{p^2}$ defined as $E_0 : y^2 = x^3 + x$. Let D be a smooth integer satisfying $E_0[D] \subset E_0(\mathbb{F}_{p^2})$ and $D \approx p$, and let d be an integer coprime to D satisfying $D - d \approx p$. RandIsogImg takes integers d, D satisfying these conditions and a finite subset S of E_0 as input, and outputs the codomain of a random d-isogeny τ and the images of the points in S under τ.

In this algorithm, we first compute an endomorphism $\alpha \in \mathrm{End}(E_0)$ of degree $d \cdot (D-d)$ using FullRepresentInteger and decompose it into $\alpha = \hat{\rho} \circ \tau$, where τ and ρ are the isogenies whose domains are E_0 and whose degrees are d and $D - d$, respectively. (See the following diagram.) Since $\deg \tau + \deg \rho = D$ and $\gcd(\deg \tau, \deg \rho) = 1$, we can evaluate point images under the isogeny τ by using KaniCod. We describe the pseudo code of RandIsogImg in Algorithm 1.



Algorithm 1. $\texttt{RandIsogImg}_{\mathcal{O}_0}(d, D; S)$

Input: Relatively prime Integers d, D such that $D - d \approx p$ and $E_0[D] \subset E_0(\mathbb{F}_{p^2})$ and a finite subset $S \subset E_0$.

Output: $(E_A, \tau(S))$ for a random d-isogeny $\tau : E_0 \to E_A$.

1: Let $\alpha \leftarrow \texttt{FullRepresentInteger}_{\mathcal{O}_0}(d \cdot (D - d))$.
2: Take a basis P_0, Q_0 of $E_0[D]$.
3: $(E_A, \tau(S), \emptyset) \leftarrow \texttt{KaniCod}(d, D - d, E_0, E_0, P_0, Q_0, \alpha(P_0), \alpha(Q_0); S; \emptyset)$.
4: **return** $(E_A, \tau(S))$.

In addition, we can compute the left $\mathcal{O}_0$-ideal $I_\tau = \mathcal{O}_0\langle \alpha, d \rangle$, which corresponds to the isogeny τ. We denote the algorithm which outputs $(E_A, \tau(S), I_\tau)$ by RandIsogImgWithIdeal.

2.6 SQIsignHD

SQIsignHD is a signature scheme proposed in [11] in 2023, which is based on SQIsign and utilizes an attack on SIDH to achieve a smaller signature length than SQIsign. There are two types of SQIsignHD, one using 4-dimensional isogenies and the other using 8-dimensional isogenies for the verification. In this section, we introduce an overview of SQIsignHD using 4-dimensional isogenies. For more details, we refer to [11].

First, we show the system parameters of SQIsignHD. Let a, b be integers satisfying $2^a \approx 3^b \approx 2^\lambda$, and let p be a prime satisfying $p = 2^a 3^b f - 1$ for a sufficiently small integer f. Let E_0 be the elliptic curve over $\mathbb{F}_{p^2}$ defined as $E_0 : y^2 = x^3 + x$. Furthermore, we say that an odd integer q is 2^a-*good* if there exist integers m_1, m_2 satisfying $m_1^2 + m_2^2 = 2^a - q$.

SQIsignHD is obtained by applying the Fiat-Shamir transform [18] on the identification scheme based on the following diagram.

$$\begin{array}{ccc} E_0 & \xrightarrow[com]{\psi} & E_1 \\ {\scriptstyle sk}\,\big\downarrow{\scriptstyle \tau} & & {\scriptstyle ch}\,\big\downarrow{\scriptstyle \phi} \\ E_A & \xrightarrow[resp]{\sigma} & E_2 \end{array}$$

In the following, we describe the overview of SQIsignHD identification protocol, which is similar to our protocol.

keygen: The prover generates a random 3^{2b}-isogeny $\tau : E_0 \to E_A$ and publishes the curve E_A as the public key.

commit: The prover generates a random 3^{2b}-isogeny $\psi : E_0 \to E_1$ and sends E_1 to the verifier as the commitment.

challenge: The verifier generates a random 3^b-isogeny $\phi : E_1 \to E_2$ and sends it to the prover.

response: The prover computes the ideal J corresponding to $\phi \circ \psi \circ \hat{\tau}$ and finds a random equivalent ideal $I_\sigma \sim J$ whose norm q is 2^a-good. Then, the prover sends to the verifier an efficient representation of the q-isogeny $\sigma : E_A \to E_2$ corresponding to I_σ.

verify: The verifier checks that the response send by the prover correctly represents a q-isogeny $\sigma : E_A \to E_2$.

As an efficient representation of the q-isogeny σ, the prover sends $(q, \sigma|_{E_A[2^a]})$. Then, the verifier recovers the isogeny σ using Theorem 1. To apply Theorem 1, the verifier needs to compute a $(2^a - q)$-isogeny from E_A. However, this task is hard since the degree $2^a - q$ is generally non-smooth. The verifier instead computes the 2-dimensional endomorphism over $E_A \times E_A$ of degree $2^a - q$ as follows:

1. Find two integers m_1, m_2 satisfying $m_1^2 + m_2^2 + q = 2^a$.
2. Let ω be the 2-dimensional endomorphism of degree $m_1^2 + m_2^2 = 2^a - q$ defined as follows:
$$\omega = \begin{pmatrix} m_1 & -m_2 \\ m_2 & m_1 \end{pmatrix}.$$

Let I_2 be the 2×2 identity matrix. Under the following diagram, the verifier can recover σ by computing 4-dimensional 2^a-isogeny. In this step, the verifier uses an extension of Theorem 1 to higher dimension by Robert [28].

$$\begin{array}{ccc} E_A \times E_A & \xrightarrow{\sigma I_2} & E_2 \times E_2 \\ \downarrow{\scriptstyle \omega} & & \downarrow{\scriptstyle \omega'} \\ E_A \times E_A & \xrightarrow[\sigma I_2]{} & E_2 \times E_2. \end{array}$$

3 Building Block for SQIsign2D-East

In this section, we give an algorithmic building block for SQIsign2D-East. We assume that we have a prime $p = 2^{a+b}f - 1$ with $a \approx b \approx \lambda$ and $f \in \mathbb{N}$ as small as possible. We use the same notation $q := \deg(\sigma)$ as in Subsect. 2.6. Note that the degree q is approximately $p^{1/2}$. In SQIsignHD, the verifier required a 4-dimensional isogeny computations since the auxiliary path ω of degree $2^a - q$ is a 2-dimensional isogeny. Our main idea is to generate the auxiliary path ω as 1-dimensional isogeny of degree $2^a - q$ by using `RandIsogImg`. However, the original `RandIsogImg` can only compute an isogeny from a specific elliptic curve E_0. Since the auxiliary path we need is the isogeny from the public key E_A, we have to construct a generalized `RandIsogImg`.

3.1 Generalized `RandIsogImg`

We construct a generalized `RandIsogImg` so that we can compute an isogeny from arbitrary curves. Let E be an elliptic curve isogenous to E_0 and let $\mathcal{O} \cong \mathrm{End}(E)$. Let τ be an N_τ-isogeny from E_0 to E and let I_τ be a left $\mathcal{O}_0$-ideal corresponding to τ. We propose an algorithm to compute an isogeny of non-smooth degree from E.

In the procedure of $\texttt{RandIsogImg}_{\mathcal{O}_0}(d, D; S)$, we use $\mathcal{O}_0$ only in step 1, where we find $\alpha \in \mathcal{O}_0$ satisfying $n(\alpha) = d \cdot (D - d)$. Therefore, to construct a generalized `RandIsogImg`, we have to find $\alpha \in \mathcal{O}$ satisfying $n(\alpha) = d \cdot (D - d)$. This can be achieved by using `EichlerModConstraint` and `StrongApproximation` as follows:

1. Using $\texttt{EichlerModConstraint}(I_\tau, 1, 1)$, obtain $(C_0 : D_0) \in \mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z})$ such that $C_0 j + D_0 k \in \mathbb{Z} + I_\tau = \mathcal{O}_0 \cap \mathcal{O}$.
2. Using $\texttt{StrongApproximation}_{d(D-d)}(N_\tau, C_0, D_0)$, we can find $\alpha \in \mathcal{O}_0 \cap \mathcal{O}$ satisfying $n(\alpha) = d(D - d)$.

The above approach is also used in the key generation and signing algorithm of SQIsign [15]. Since we use `StrongApproximation`, the degree N_τ of τ must be prime and $d(D-d) > pN_\tau^3$ must hold. If we assume that $D - d \approx p$ as with the original `RandIsogImg`, the requirement on the degree d will be $d > N_\tau^3$. In addition, if we fix D around p, the condition $D - d \approx p$ holds for almost all d satisfying $d < D$.

In the following, we show there is an additional hidden constraint on d. First, $\texttt{StrongApproximation}_{d(D-d)}(N_\tau, C_0, D_0)$ outputs $\mu \in \mathcal{O}_0$ such that

$$n(\mu) = d(D-d) \text{ and } \mu = m(C_0\mathbf{j} + D_0\mathbf{k}) + N_\tau\mu_1,$$

where $m \in \mathbb{Z}$ and $\mu_1 \in \mathcal{O}_0$. Therefore, the following equation holds:

$$d(D-d) = n(\mu) \equiv m^2 p(C_0^2 + D_0^2) \mod N_\tau.$$

For such an integer m to exist, the following condition must be satisfied:

$$\left(\frac{d(D-d)}{N_\tau}\right) = \left(\frac{p(C_0^2 + D_0^2)}{N_\tau}\right),$$

where $\left(\frac{\cdot}{\cdot}\right)$ is the quadratic residue symbol. On the other hand, from the definition of `EichlerModConstraint`, there exists an integer m' satisfying

$$m' + C_0\mathbf{j} + D_0\mathbf{k} \in I_\tau.$$

Hence, we have

$$n(m' + C_0\mathbf{j} + D_0\mathbf{k}) = (m')^2 + p(C_0^2 + D_0^2) \equiv 0 \mod N_\tau,$$

which means that

$$\left(\frac{p(C_0^2 + D_0^2)}{N_\tau}\right) = \left(\frac{-1}{N_\tau}\right).$$

Summarizing the above discussion, d must satisfy

$$\left(\frac{d(D-d)}{N_\tau}\right) = \left(\frac{-1}{N_\tau}\right). \tag{2}$$

However, if we use the degree d satisfying this condition in our protocol, we face a security issue. We explain this issue in Sect. 4.3. To avoid this security issue, we instead require that $3 \mid d(D-d)$ and that 3 is not a square modulo N_τ, i.e., we require $N_\tau \equiv 5, 7 \bmod 12$. Then, our two new conditions together allow us to modify as follows:

- if d satisfies condition (2), then we call `StrongApproximation` with target norm $M = d(D-d)$;
- otherwise, we call `StrongApproximation` with target norm $d(D-d)/3$ to obtain an endomorphism α'. In this case we have that $d(D-d)/3$ satisfies

$$\left(\frac{d(D-d)/3}{N_\tau}\right) = \left(\frac{-1}{N_\tau}\right).$$

After that, we compute a random degree 3 isogeny $\alpha'' : E \to E''$ using Vélu's formulas and we compose it with α' to finally obtain an isogeny α of degree $d(D-d)$ from E to E''.

From the above argument, a generalized `RandIsogImg` for E is as shown in Algorithm 2.

Algorithm 2. $\texttt{GenRandIsogImg}_{\tau,I_\tau}(d, D; S)$

Input: An isogeny $\tau : E_0 \to E$ of prime degree N_τ, its corresponding ideal I_τ, relatively prime integers d, D such that $3 \mid d(D-d)$, $N^3 < d < D \approx p$, and $E[D] \subseteq E(\mathbb{F}_{p^2})$, and a finite set $S \subseteq E$,
Output: $(F; \iota(S))$ for a random d-isogeny $\iota : E \to F$.

$(C_0 : D_0) \leftarrow \texttt{EichlerModConstraint}(I_\tau, 1, 1)$
Let P, Q be a basis of $E[D]$.
if d satisfies $\left(\frac{d(D-d)}{N_\tau}\right) = \left(\frac{-1}{N_\tau}\right)$ **then**
  $\alpha \leftarrow \texttt{StrongApproximation}_{d(D-d)}(N_\tau, C_0, D_0)$
  $(F; \iota(S); \emptyset) \leftarrow \texttt{KaniCod}(d, D-d, E, E, P, Q, \alpha(P), \alpha(Q); S; \emptyset)$
else
  $\alpha' \leftarrow \texttt{StrongApproximation}_{d(D-d)/3}(N_\tau, C_0, D_0)$
  $\alpha'' \leftarrow$ random 3-isogeny $E \to E''$, computed using Vlu's formulas.
  $\alpha \leftarrow \alpha'' \circ \alpha'$
  $(F; \iota(S); \emptyset) \leftarrow \texttt{KaniCod}(d, D-d, E, E'', P, Q, \alpha(P), \alpha(Q); S; \emptyset)$
end if
return $(F; \iota(S))$

3.2 Computing Auxiliary Path

Unfortunately, the requirement $d > N_\tau^3$ is too strong to compute an auxiliary path of degree $r = 2^a - q \approx p^{1/2}$. To allow the use of smaller degree, we take the following approach:

1. Let D_1 be a smooth integer such that $r(D_1 - r) > N_\tau^3$ and $r(D_1 - r) < D$.
2. Compute a $r(D_1 - r)$-isogeny using `GenRandIsogImg`.
3. By computing a (D_1, D_1)-isogeny, obtain a r-isogeny.

Then, the lower bound of r decreases from N_τ^3 to approximately N_τ^3/D_1.

Remark 1. Strictly speaking, the lower bound of r is $B = D_1/2 - \sqrt{(D_1/2)^2 - N_\tau^3}$ $= (D_1/2) \cdot (1 - \sqrt{1 - 4N_\tau^3/D_1^2})$. Especially when $D_1^2 \gg N_\tau^3$, we have $B \approx N_\tau^3/D_1$, where we used $\sqrt{1-\epsilon} \approx 1 - \epsilon/2$ for $\epsilon \ll 1$.

Algorithm to compute an auxiliary path is given in Algorithm 3. Especially in our protocol, we use $D_1 = 2^a \approx p^{1/2}$ and $D = 2^{a+b} \approx p$. Since the degree $r = 2^a - q$ of the auxiliary path we need is around $p^{1/2}$, we have $r(D_1 - r) \approx p$ for almost all $r < D_1$. Hence, the condition $r(D_1 - r) > N_\tau^3$ is satisfied when $N_\tau < p^{1/3}$.

Algorithm 3. $\texttt{AuxiliaryPath}_{\tau,I_\tau}(r, D_1, D; S)$

Input: An isogeny $\tau : E_0 \to E$ of prime degree N_τ, its corresponding ideal I_τ, integers r, D_1, D such that $\gcd(r, D_1 D) = 1$, $N_\tau^3 < r(D_1 - r) < D \approx p$, $3 \mid d(D - d)$ for $d = r(D_1 - r)$, and $E[D] \subset E(\mathbb{F}_{p^2})$, and a finite set $S \subset E$.
Output: $(F; \omega(S))$ for a random r-isogeny $\omega : E \to F$.
1: Let P, Q be a basis of $E[D_1]$.
2: $(F'; \iota(P), \iota(Q)) \leftarrow \texttt{GenRandIsogImg}_{\tau,I_\tau}(r(D_1 - r), D; P, Q)$.
3: $(F; \omega(S); \emptyset) \leftarrow \texttt{KaniCod}(r, D_1 - r, E, F', P, Q, \iota(P), \iota(Q); S; \emptyset)$.
4: **return** $(F; \omega(S))$.

In the following, let $M(q) \coloneqq q(2^a - q)(2^{a+b} - q(2^a - q))$. From the above argument, the requirements on the degree q are as follows:

$$\begin{aligned} &q \text{ is odd integer smaller than } 2^a, \\ &q(2^a - q) < 2^{a+b}, \\ &3 \mid M(q). \end{aligned}$$

Definition 1. *We say that a positive integer q is '$(2^a, 2^b)$-nice' if q is an odd integer smaller than 2^a and satisfying $q(2^a - q) < 2^{a+b}$. In addtion, we say that a positive integer q is '$(2^a, 2^b)_3$-nice' if q is $(2^a, 2^b)$-nice and satisfies $3 \mid M(q)$.*

Remark 2. The odd integer $q < 2^a$ is always $(2^a, 2^b)$-nice when $a - b \leq 2$ from the following inequality:

$$q \cdot (2^a - q) = 2^{2a-2} - (2^{a-1} - q)^2 < 2^{2a-2} \leq 2^{a+b}.$$

Remark 3. From the following facts, the probability that $3 \mid M(q)$ is 2/3 or 1.

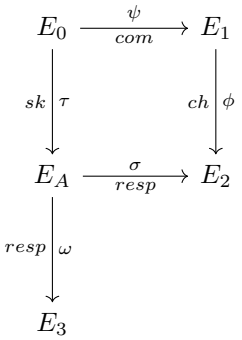
- if $a \equiv b \bmod 2$ then $3 \mid M(q)$,
- if $a \equiv 0 \bmod 2$ and $b \equiv 1 \bmod 2$ then $3 \nmid M(q)$ if and only if $q \equiv 2 \bmod 3$,
- if $a \equiv 1 \bmod 2$ and $b \equiv 0 \bmod 2$ then $3 \nmid M(q)$ if and only if $q \equiv 1 \bmod 3$.

4 New Signature Scheme: SQIsign2D-East

In this section, we describe our new signature scheme SQIsign2D-East. First, we describe the detailed algorithm for SQIsign2D-East and then we propose its variant named 'CompactSQIsign2D-East', which has smaller signature size than the original SQIsign2D-East.

4.1 Description of SQIsign2D-East

We first describe the identification protocol underlying SQIsign2D-East. SQIsign-2D-East identification protocol is based on the following diagram.



We show the algorithms for the SQIsign2D-East identification scheme below.

Parameter Setting. The public parameter of SQIsign2D-East is taken as follows:

1. Let p be a prime of the form $p = 2^{a+b}f - 1$, where f is a small integer and $a \approx b \approx \lambda$.
2. Let E_0 be the elliptic curve over $\mathbb{F}_{p^2}$ defined as $E_0 : y^2 = x^3 + x$.
3. Let P_0, Q_0 be a basis of $E_0[2^{a+b}]$.
4. Let $\mathcal{O}_0 = \mathbb{Z}\langle 1, \mathbf{i}, \frac{\mathbf{i}+\mathbf{j}}{2}, \frac{1+\mathbf{k}}{2}\rangle$, which is isomorphic to $\mathrm{End}(E_0)$.
5. Let $\mathrm{param} = (p, a, b, E_0, P_0, Q_0, \mathcal{O}_0)$.

Key Generation. As we stated in Subsect. 3.2, we have to take the degree N_τ of the secret isogeny τ smaller than $p^{1/3}$. Fortunately, we can take N_τ as small as approximately $p^{1/4}$ while achieving λ-bits security as follows:

1. Take a random prime $N_\tau < p^{1/4}$ such that $\left(\frac{3}{N_\tau}\right) = -1$.
2. Compute a random N_τ-isogeny $\tau : E_0 \to E$.

The method to use a random degree smaller than $p^{1/4}$ is also used in the key generation of SQIsign [13].

Since N_τ is a large prime, we cannot compute τ efficiently from $\ker \tau$ using Vélu's formulas. Instead, we compute an efficient representation $(N_\tau, \tau(P_0), \tau(Q_0))$ of τ using `RandIsogImg`. By using $(N_\tau, \tau(P_0), \tau(Q_0))$, we can efficiently compute $\tau(T_0)$ for any $T_0 \in E_0[2^{a+b}]$ as follows:

1. Find $s, t \in \mathbb{Z}/2^{a+b}\mathbb{Z}$ such that $T_0 = sP_0 + tQ_0$.
2. Return $\tau(T_0) = s\tau(P_0) + t\tau(Q_0)$.

Now we show the key generation algorithm in Algorithm 4.

Algorithm 4. keygen(param) $\rightarrow (pk, sk)$

Input: Public parameter param $= (p, a, b, E_0, P_0, Q_0, \mathcal{O}_0)$.
Output: Public key pk and secret key sk.
1: Take a random prime $N_\tau < p^{1/4}$.
2: $(E_A, R_A, S_A, I_\tau) \leftarrow \texttt{RandIsogImgWithIdeal}_{\mathcal{O}_0}(N_\tau, 2^{a+b}; P_0, Q_0)$.
3: **return** $pk = E_A, sk = (\tau = (N_\tau, R_A, S_A), I_\tau)$.

Commitment. The commitment phase is similar to the key-generation. However, the commitment degree N_ψ need not to be prime smaller than $p^{1/4}$ unlike N_τ. Hence, we just chose a raodom odd integer N_ψ smaller than 2^{a+b}.

As with the key generation, we compute $(N_\psi, \psi(P_0), \psi(Q_0))$ as an efficient representation of ψ using `RandIsogImg`. As described above, we can efficiently evaluate ψ over the 2^{a+b}-torsion subgroup using this representation. In addition, we can compute $\hat{\psi}(T_1)$ for any $T_1 \in E_1[2^{a+b}]$, where E_1 is the codomain of ψ as follows:

1. Find $s, t \in \mathbb{Z}/2^{a+b}\mathbb{Z}$ such that $T_1 = s\psi(P_0) + t\psi(Q_0)$.
2. Return $\hat{\psi}(T_A) = sN_\psi P_0 + tN_\psi Q_0$.

Now, we show the commitment algorithm in Algorithm 5.

Algorithm 5. commit(param) $\rightarrow (com, s)$

Input: Public parameter param.
Output: Commitment com and secret information s.
1: Take a random odd integer $N_\psi < 2^{a+b}$.
2: $(E_1, R_1, S_1, I_\psi) \leftarrow \texttt{RandIsogImgWithIdeal}_{\mathcal{O}_0}(N_\psi, 2^{a+b}; P_0, Q_0)$.
3: **return** $com = E_1, s = (\psi = (N_\psi, R_1, S_1), I_\psi)$.

Challenge. We just compute a random 2^b-isogeny from E_1 using Vélu's formulas. We show the challenge algorithm in Algorithm 6.

Response. In the response phase, we first compute the ideal I_ϕ. This can be done by using `IsogToIdeal` algorithm [11, Algorithm 10], which takes two isogenies $\psi : E_0 \rightarrow E_1$ and $\phi : E_1 \rightarrow E_2$ and the ideal I_ψ corresponding to ψ as input and return the ideal I_ϕ corresponding to ϕ. Then, we compute the ideal J

corresponding to $\phi \circ \psi \circ \hat{\tau}$. Next, we find all $\alpha \in J$ such that $q := n(\alpha)/n(J)$ is $(2^a, 2^b)_3$-nice by lattice enumeration (e.g., see [8, Algorithm 2.7.5]) and choose one of them uniformly. Then, we let $I_\sigma = J\frac{\bar{\alpha}}{n(J)}$ and compute an efficient representation of the q-isogeny $\sigma : E_A \to E_2$ corresponding to I_σ. Finally, we generate an auxiliary path $\omega : E_A \to E_3$ and return an efficient representation of $\sigma \circ \hat{\omega}$.

Algorithm 6. challenge$(pk, \text{param}) \to ch$

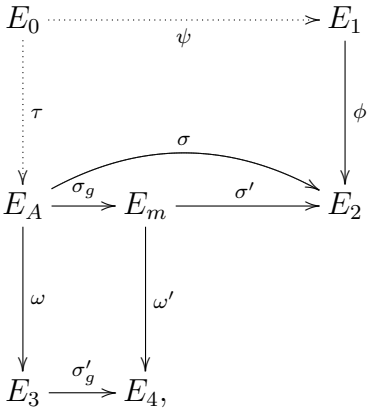
Input: Public key pk and public parameter param.
Output: Challenge ch.
1: Take a random integer $u \in_U \mathbb{Z}/2^b\mathbb{Z}$ and a bit $\text{bin} \in_U \{0, 1\}$.
2: Let P'_1, Q'_1 be the canonical basis of $E_1[2^b]$.
3: If $\text{bin} = 0$, $K'_1 \leftarrow P'_1 + uQ'_1$, otherwise, $K'_1 \leftarrow uP'_1 + Q'_1$.
4: **return** $ch = K'_1$, a generator of the kernel of $\phi : E_1 \to E_2$.

If there is no ideal I_σ whose norm q is $(2^a, 2^b)_3$-nice, we need to go back to the commitment phase. In the following, we discuss how to avoid this. From now on, we assume that $a - b \leq 2$, which means that at least $2/3$ of odd integers smaller than 2^a are $(2^a, 2^b)_3$-nice (see Remark 2 and Remark 3). To avoid the failure in finding I_σ, we consider using $q' = q/\gcd(q, f)$ instead of q. This reduces the constraint of q from $q < 2^a$ to $q' < 2^a \Leftrightarrow q < \gcd(q, f) \cdot 2^a$.

Definition 2. *We say that a positive integer q is '$(2^a, 2^b, f)$-nice' when $q' = q/\gcd(q, f)$ is $(2^a, 2^b)$-nice. Similarly, we say that q is '$(2^a, 2^b, f)_3$-nice' when $q' = q/\gcd(q, f)$ is $(2^a, 2^b)_3$-nice.*

Let σ be a q-isogeny computed in the response phase. Assume that q is $(2^a, 2^b, f)_3$-nice and let $g = \gcd(q, f)$, $q' = q/g$, and $r = 2^a - q'$. We formally decompose the q-isogeny σ to a g-isogeny $\sigma_g : E_A \to E'_A$ and a q'-isogeny $\sigma' : E'_A \to E_2$ and take the following procedures:

1. Compute $\ker \sigma_g$ by evaluating σ over $E_A[g]$.
2. Compute $\sigma_g : E_A \to E_m$ by using Vélu's formulas.
3. Obtain an r-isogeny $\omega : E_A \to E_3$ by using `AuxiliaryPath`.
4. Let $\sigma'_g = [\omega]_* \sigma_g$ and compute $\ker \sigma'_g = \omega(\ker \sigma_g)$.
5. Compute $\sigma'_g : E_3 \to E_4$ by using Vélu's formulas.
6. Evaluate σ' and ω' over $E_m[2^a]$ by using the relationships: $\sigma' \circ \sigma_g = \sigma$ and $\omega' \circ \sigma_g = \sigma'_g \circ \omega$.



Note that there is a concern that $\deg \sigma_g = g$ is not coprime to $\deg \omega = r$. This means that the degree of ω' may not be equal to r but reduces to $\tilde{r} = r/h$ for a factor h of $\gcd(g, r)$. In this case, we additionally compute a random h-isogeny ι from E_4 and use $\iota \circ \omega'$ as an auxiliary path. For simplicity, we consider the case $h = 1$ in the following.

The response algorithm is given in Algorithm 7. To compute R'_2, S'_2 in step 5, we use the following equation:

$$\sigma \circ [2^b] = \frac{1}{N_\tau N_\psi} \phi \circ \psi \circ \hat{\tau} \circ \hat{\alpha},$$

which is obtained by applying the Deuring correspondence on the equation $I_\sigma = \bar{I}_\tau I_\psi I_\phi \cdot \frac{\bar{\alpha}}{N_\tau N_\psi 2^b}$. Then, we can compute R'_2 as follow:

$$R'_2 = \frac{1}{N_\psi N_\tau} \phi \circ \tau \circ \hat{\psi} \circ \hat{\alpha}(P_A) = \sigma(2^b P_A) = \sigma(P'_A).$$

We can compute S'_2 similarly.

In step 6, we compute $R'_3 = \omega(P'_A)$ and $S'_3 = \omega(Q'_A)$ for an r-isogeny $\omega : E_A \to E_3$. Since $K'_g = 2^a(R'_3 + \ell S'_3) = \omega(K_g)$ holds, we have $\sigma'_g = [\omega]_* \sigma_g$. Therefore, the following equation holds:

$$R'_4 = \sigma'_g(gR'_3) = \sigma'_g \circ \omega(gP'_A) = \omega' \circ \sigma_g(gP'_A) = \omega'(R'_m),$$

where $\omega' = [\sigma_g]_* \omega$. Similarly, $S'_4 = \omega'(S'_m)$ also holds.

From the equation $(P'_4, Q'_4) = (R'_4, S'_4)M = (\omega'(R'_m), \omega'(S'_m))M$ in step 11, the following equation holds:

$$(\hat{\omega'}(P'_4), \hat{\omega'}(Q'_4)) = r(R'_m, S'_m)M = -q(R'_m, S'_m)M,$$

Algorithm 7. response$(sk, s, ch, \text{param}) \to resp$

Input: Secret key sk, secret information s, challenge ch, and public parameter param.
Output: Response $resp$.

1: Let $I_\phi \leftarrow \texttt{IsogToIdeal}(\phi, \psi, I_\psi)$.
2: Let $J = \bar{I}_\tau I_\psi I_\phi$.
3: Find all $\alpha \in J$ such that $q \coloneqq n(\alpha)/n(J)$ is $(2^a, 2^b, f)_3$-nice by lattice enumeration and choose one of them uniformly.
4: Let $I_\sigma = J\frac{\bar{\alpha}}{n(J)}$.
5: Let $q = n(I_\sigma), g = \gcd(q, f), q' = q/g$ and $r = 2^a - q'$.
6: Let P_A, Q_A be the canonical basis of $E_A[2^{a+b}g]$ and let $(P'_A, Q'_A) = 2^b(P_A, Q_A)$.
7: Compute $R'_2 = \sigma(P'_A)$ and $S'_A = \sigma(Q'_A)$.
8: Let $(E_3, R'_3, S'_3) \leftarrow \texttt{AuxiliaryPath}_{I_\tau}(r, 2^a, 2^{a+b}; P'_A, Q'_A)$.
9: Find an integer ℓ such that $2^a(R'_2 + \ell S'_2) = O$ (or $2^a(\ell R'_2 + S'_2) = O$) and let $K_g = 2^a(P'_A + \ell Q'_A)$ (or $K_g = 2^a(\ell P'_A + Q'_A)$).
10: Compute $\sigma_g : E_A \to E_m = E_A/\langle K_g \rangle$, $R'_m = \sigma_g(gP'_A), S'_m = \sigma_g(gQ'_A)$.
11: Let $K'_g = 2^a(R'_3 + \ell S'_3)$ (or $K_g = 2^a(\ell P'_A + Q'_A)$).
12: Compute $\sigma'_g : E_3 \to E_4 = E_A/\langle K'_g \rangle$, $R'_4 = \sigma'_g(gR'_3), S'_4 = \sigma'_g(gS'_3)$.
13: Let P'_4, Q'_4 be the canonical basis of $E_4[2^a]$ and compute the change of basis matrix M such that $(P'_4, Q'_4) = (R'_4, S'_4)M$.
14: Compute $(U_2, V_2) = -g(R'_2, S'_2)M$.
15: **return** $resp = (K_g, E_4, U_2, V_2)$.

where we used $r = 2^a - q' \equiv -q' \mod 2^a$. By taking the image under the isogeny σ' of both sides, we obtain

$$\begin{aligned}(\sigma' \circ \hat{\omega'}(P'_4), \sigma' \circ \hat{\omega'}(Q'_4)) &= -q(\sigma'(R'_m), \sigma'(S'_m))M \\ &= -q(\sigma' \circ \sigma_g(gP'_A), \sigma' \circ \sigma_g(gQ'_A))M \\ &= -qg(\sigma(P'_A), \sigma(Q'_A))M \\ &= -qg(R'_2, S'_2)M = q(U_2, V_2).\end{aligned}$$

Therefore, we obtain the following equation:

$$(U_2, V_2) = \left(\frac{1}{q}\sigma' \circ \hat{\omega'}(P'_4), \frac{1}{q}\sigma' \circ \hat{\omega'}(Q'_4)\right). \tag{3}$$

Verify. We show the verification algorithm in Algorithm 8. We prove that SQIsign2D-East identification protocol is complete. Assume here that the prover computes the response honestly. From Eq. 3, the subgroup K of $E_A \times F$ satisfies

Algorithm 8. verify$(pk, com, ch, resp, \text{param}) \rightarrow$ accept/reject

Input: Public key pk, commitment com, challenge ch, response $resp$, and public parameter param.
Output: accept or reject.
1: Compute $\sigma_g : E_A \rightarrow E_m = E_A/\langle K_g \rangle$.
2: Let P'_4, Q'_4 be the canonical basis of $E_3[2^a]$.
3: Compute a $(2^a, 2^a)$-isogeny $\Phi : E_4 \times E_2 \rightarrow A$ with kernel $K = \langle (P'_3, U_2), (Q'_3, V_2) \rangle$.
4: **if** $A \cong E_m \times F$ for an elliptic curve F **then**
5: **return** accept.
6: **else**
7: **return** reject.
8: **end if**

the following equation:

$$\begin{aligned} K &= \langle (P'_4, U_2), (Q'_4, V_2) \rangle \\ &= \left\langle \left(P'_4, \frac{1}{q}\sigma' \circ \hat{\omega'}(P'_4) \right), \left(Q'_4, \frac{1}{q}\sigma' \circ \hat{\omega'}(Q'_4) \right) \right\rangle \\ &= \langle (qP'_4, \sigma' \circ \hat{\omega'}(P'_4)), (qQ'_4, \sigma' \circ \hat{\omega'}(Q_3 4)) \rangle. \end{aligned}$$

Let $\sigma'' = [\omega']_* \sigma', \omega'' = [\sigma']_* \omega'$, and F be the codomain of σ'' and ω''. From Theorem 1, a $(2^a, 2^a)$-isogeny Φ with kernel K has the following form:

$$\Phi = \begin{pmatrix} \hat{\omega'} & -\hat{\sigma'} \\ \sigma'' & \omega'' \end{pmatrix} : E_4 \times E_2 \rightarrow E_m \times \text{ F}$$

up to isomorphism. Therefore, the verifier accepts the honest response.

4.2 Reducing Signature Size

Applying the Fiat-Shamir transform, the signature of our protocol is made of the data $(E_1, K_g, E_4, U_2, V_2)$, where E_1 is the commitment elliptic curve, E_4 is the codomain of the auxiliary path, $K_g \in E_A[g]$, and $U_2, V_2 \in E_2[2^a]$. E_1 and E_4 can be determined by their j-invariant $j(E_1), j(E_4) \in \mathbb{F}_{p^2}$. Therefore, storing E_1 and E_4 takes $2 \log_2 p^2 \approx 8\lambda$ bits. The points U_2 and V_2 can be compressed as in SIKE. Using this compression, U_2 and V_2 requires $3a \approx 3\lambda$ bits. Similarly, the point K_g can be compressed and it requires about $\log_2 g\lambda$ bits. Totally, the signature size is 11λ bits.

Actually, we can reduce the signature size by about 2λ bits by using the same method as SQIsign: include the information about $\hat{\phi}$ instead of the commitment E_1 in the signature. We name this variant 'CompactSQIsign2D-East'. To apply this method, we compute $\omega'' = [\sigma']_* \omega'$ using `KaniCod`. Now we explain how

CompactSQIsign2D-East works. Let $H : \{0,1\}^* \times \mathbb{F}_{p^2} \to \mathbb{Z}/2^b\mathbb{Z} \times \{0,1\}$ be a cryptographic hash function and let `GenKer` be an algorithm defined as follows:

`GenKer`$(m, E_1) \to K_1'$:

1. $h, \mathrm{bin} \leftarrow H(m, j(E_1))$.
2. Let P_1', Q_1' be the canonical basis of $E_1[2^b]$.
3. If $\mathrm{bin} = 0$, return $K_1' = hP_1' + Q_1'$.
4. Otherwise, return $K_1' = P_1' + hQ_1'$.

In the following, we regard $\mathbb{F}_{p^2}$ as a totally ordered set under an appropriate order relation. We show the explicit algorithms for CompactSQIsign2D-East in Algorithm 9 and 10. Note that the key generation algorithm for CompactSQIsign2D-East is same as Algorithm 4.

Algorithm 9. CompactSign$(pk, sk, m, \mathrm{param}) \to sig$

Input: The public key pk, the secret key sk, the message m, and the public parameter param.
Output: The signature sig.

1: $(E_1, N_\psi, R_1, S_1, I_\psi) \leftarrow, (param)$.
2: Let $K_1 \leftarrow$ `GenKer`(m, E_1) and $\hat{\phi} : E_1 \to E_2$.
3: Let K_2 be a generator of $\ker \hat{\phi}$.
4: Find a 2^b-torsion point P_2' linearly independent with K_2 deterministically.
5: Find $t \in \mathbb{Z}/2^b\mathbb{Z}$ satisfying $K_1 = t\hat{\phi}(P_2')$.
6: Compute P_4', Q_4', R_m', S_m', and $resp = (K_g, E_4, U_2, V_2)$ using Algorithm 7.
7: $(F; \emptyset; U, V) \leftarrow$ `KaniCod`$(q', r, E_4, E_2, P_4', Q_4', qU_2, qV_2; \emptyset; R_m', S_m')$.
8: Let M and M_F be the Montgomery coefficient of E_m and F, respectively.
9: **if** $M \leq M_F$ **then**
10: bin $\leftarrow 0$.
11: **else**
12: bin $\leftarrow 1$.
13: **end if**
14: **return** $sig = (K_g, F, U, V, K_2, t, \mathrm{bin})$.

Since the point $K_2 \in E_2[2^b]$ can be represented by a single $\mathbb{Z}/2^b\mathbb{Z}$ element, the size of (K_2, t) is about $2b$ bits. Therefore, the total signature size is about $\log_2 p^2 + 3a + 2b \approx 9\lambda$ bits.

4.3 Security Issue

We discuss the security issue when we use only $(2^a, 2^b)$-nice degrees q satisfying Eq. (2) for $d = q(2^a - q)$ and $D = 2^{a+b}$.

As a first step, we observe that an adversary can evaluate σ at any input; the degree q can be then recovered using a pairing computation combined with

Algorithm 10. CompactVerify$(pk, m, sig, \text{param}) \rightarrow$ accept/reject

Input: The public key pk, the message m, the signature sig, and the public parameter param.
Output: accept or reject.

Let P_A, Q_A be the canonical basis of $E_A[2^{a+b}g]$.
Compute $\sigma_g : E_A \rightarrow E_m = E_A/\langle K_g \rangle$, $R'_m = \sigma_g(2^b g P_A), S'_m = \sigma_g(2^b g Q_A)$.
Compute a $(2^a, 2^a)$-isogeny $\Phi : E_m \times f \rightarrow A$ with kernel $\langle (R'_m, U), (S'_m, V) \rangle$.
if not $A \cong F_0 \times F_1$ for elliptic curves F_0 and F_1 **then**
    return reject.
end if
Let M_0 and M_1 be the Montgomery coefficient of F_0 and F_1, respectively.
if $M_0 > M_1$ **then**
    $F_0, F_1 \leftarrow F_1, F_0$.
end if
$E_2 \leftarrow F_{\text{bin}_2}$.
Find a 2^b-torsion point P'_2 linearly independent with K_2 deterministically.
Compute a 2^a-isogeny $\hat{\phi} : E_2 \rightarrow E_1 = E_2/\langle K_2 \rangle$ and $L_1 = \hat{\phi}(P'_2)$.
Let $K_1 \leftarrow \texttt{GenKer}(m, E_1)$.
if $K_1 = tL'_1$ **then**
    return accept.
else
    return reject.
end if

an easy discrete log computation. Therefore it can be assumed that q is known. It is important to note that q varies with every signature and is essentially random subject to the above condition. Hence for each signature the adversary learns that $M(q)$ has the same quadratic residuosity as $-1 \bmod N_\tau$. From Dirichlet's theorem on arithmetic progressions it follows that, as soon as $M(q)$ is not an exact square, the density of primes N_τ satisfying (2) is 50%. Thus, heuristically, we expect that N_τ is uniquely determined by about $\lambda/2$ values of q. This means that after seeing roughly $\lambda/2$ signatures we should be able to find N_τ by simply brute-forcing over all primes in $(0, p^{1/4})$ and testing whether (2) holds for each of the corresponding values of q. Ignoring polynomial overhead, this step therefore has a complexity of $O(2^{\lambda/2})$.

Given the norm N_τ of the secret ideal I_τ, we can recover I_τ by enumerating all left $\mathcal{O}_0$-ideals of norm N_τ and check whether the corresponding isogenies have codomain isomorphic to E_A. There will be $O(2^{\lambda/2})$ such ideals and they can be enumerated using the bijection from [22, Lemma 7.2]. Therefore, the cost of this step is $\tilde{O}(2^{\lambda/2})$.

4.4 On Sampling a Response Ideal

Lemma 1. *Let $f \in \mathbb{Z}_{>0}$. As $x \to \infty$, the proportion of integers $q \in (x, fx)$ satisfying $q < \gcd(q, f)x$ converges to*

$$\frac{P(f) - f}{f(f-1)},$$

where $P(f)$ denotes the gcd-sum function (also known as Pillai's arithmetical function):

$$P(f) = \sum_{k=0}^{f-1} \gcd(k, f) = \sum_{d|f} d\varphi(f/d).$$

Proof. For any $k = 0, \ldots, f-1$, the number of integers $q \in (x, fx)$ such that

$$q \bmod f = k \quad \text{and} \quad q < \gcd(q, f)x$$

is asymptotic to

$$\frac{1}{f} \cdot \frac{\gcd(k, f) - 1}{f - 1},$$

so the lemma follows by summing over all congruence classes mod f.

For a detailed study of the gcd-sum function, we refer to [31]. It is a multiplicative function which at prime powers $f = \ell^e$ takes the values $P(\ell^e) = (e+1)\ell^e - e\ell^{e-1}$. On "average", it can be shown that

$$P(f) \approx \frac{3f \log f}{\pi^2},$$

although its concrete values fluctuate largely with f.

Heuristic. *Let J be a left ideal of $\mathcal{O}_A$ and assume that $0 \le a - b \le 2$. If*

$$\delta \pi^2 2^{a-b} P(f) > f$$

where

$$\delta = \begin{cases} 1 & \text{if } a \equiv b \bmod 2, \\ 2/3 & \text{if not} \end{cases}$$

(see Remark 3), then on average we expect there to exist at least one left ideal $I_\sigma \sim J$ such that $q := n(I_\sigma)$ is odd, $M(q)$ is divisible by 3, $q < \gcd(q, f)2^a$ and $q(2^a - q) < 2^{a+b}$. More quantitatively, the probability that no such ideal exists can be estimated as

$$\left(1 - \delta \frac{P(f)}{2f^2}\right)^{2\pi^2 f 2^{a-b}} \cdot \left(1 - \delta \frac{P(f) - f}{3f(f-1)}\right)^{2\pi^2 f 2^{a-b}}.$$

Explanation. First note that the assumption $0 \leq a-b \leq 2$ implies that $q(2^a-q) < 2^{a+b}$ as soon as q is odd, so the last condition is of no concern. The Gaussian heuristic says that in any sufficiently general lattice $\Lambda \subset \mathbb{R}^4$, we expect

$$\#\{\,\alpha \in \Lambda \mid \|\alpha\| < R\,\} \approx \frac{\frac{\pi^2}{2}R^4}{\mathrm{Vol}(\Lambda)},$$

where the numerator on the right is just the volume of a ball in $\mathbb{R}^4$ with radius R. Applying this heuristic to $\Lambda = J$, which has Euclidean covolume $n(J)^2 p/4$, and to $R = \sqrt{f2^a n(J)}$, we find an expected number of

$$\frac{2\pi^2 f^2 2^{2a}}{p} \approx 2\pi^2 f 2^{a-b}$$

elements $\alpha \in J$ whose quaternion norm is smaller than $f2^a n(J)$. Assuming $\mathcal{O}_R(J)^\times = \{\pm 1\}$, from [14, Lemma 1] it follows that there should be about $\pi^2 f 2^{a-b}$ left ideals $I_\sigma \subset \mathcal{O}_A$ satisfying $I_\sigma \sim J$ and $n(I_\sigma) < f2^a$.

If we furthermore assume that the norms of these I_σ's behave as independent uniform variables in $(0, f2^a) \cap \mathbb{Z}$, then we expect a proportion of $1/2$ to be odd, a proportion of δ to satisfy $3 \mid M(q)$, and a proportion of $P(f)/f^2$ to meet the bound $q < \gcd(q, f)2^a$, leading to

$$\delta \frac{\pi^2 2^{a-b} P(f)}{f}$$

ideals whose norm q is of the desired type.

Remark 4. The count is slightly off in case f is even, because the condition $q < \gcd(q, f)2^a$ is not independent of the condition that q is odd. A similar remark applies in case $\delta = 2/3$ and $3 \mid f$. For simplicity, we ignore this issue here, although it is taken into account in the failure rates listed in Table 1.

A similar reasoning shows the failure rate of the signing procedure (i.e., the probability of having to go back to the commitment phase): this is

$$\left(1 - \delta \frac{P(f)}{2f^2}\right)^{2\pi^2 f 2^{a-b}}. \tag{4}$$

For the concrete parameter sets shown in Sect. 6.1, this gives (Fig. 1):

5 Security Analysis

We now discuss the security of our SQIsign2D-East.

NIST level	a	b	f	failure rate
1	127	126	27	2^{-78}
3	191	189	35	2^{-195}
5	254	253	153	

Fig. 1. Heuristical rates of failure to find an equivalent ideal of the desired norm type. For NIST level 3 this is formula (4). For the other NIST levels the formula was tweaked so as to take into account Remark 4.

5.1 On the Distribution of Auxiliary Paths

Let $\tau : E_0 \to E_A$ be an N_τ-isogeny and I_τ be the left $\mathcal{O}_0$-ideal corresponding to τ. Given the right order $\mathcal{O}_A$ of I_τ, we use $\mathcal{S}_{I_\tau,M}$ to denote the distribution on

$$\mathcal{O}_M := \{\alpha \in \mathcal{O}_0 \cap \mathcal{O}_A \mid n(\alpha) = M\}$$

that are outputs of the algorithm consisting of first getting $(C_0 : D_0) \in \mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z})$ by running $\texttt{EichlerModConstraint}(I_\tau, 1, 1)$, then getting $\alpha \in \mathcal{O}_0 \cap \mathcal{O}_A$ with norm M by running $\texttt{StrongApproximation}_M(N_\tau, C_0, D_0)$.

For a fixed q, we define

$$\texttt{Iso}(E_A, q) := \{\varphi : E_A \to \star \text{ such that } \deg\varphi = 2^a - q\},$$

and we consider the following distributions on $\texttt{Iso}(E_A, q)$:

$\mathcal{D}_\mathcal{U}$: The uniform distribution $\mathcal{U}_{\texttt{Iso}(E_A,q)}$.
$\mathcal{D}_1$: For q such that $d = q(2^a - q)$ satisfies Eq. (2): a factor of θ_α of degree $2^a - q$ where $\alpha \sim \mathcal{S}_{I_\tau,M(q)}$ and $\theta_\alpha \in \mathrm{End}(E_A)$ is the corresponding endomorphism.
$\mathcal{D}_2$: For q such that $d = q(2^a - q)$ does not satisfy Eq. (2): a factor of $\theta_\alpha \circ \theta''$ of degree $2^a - q$ where $\alpha \sim \mathcal{S}_{I_\tau,M(q)/3}$, $\theta_\alpha \in \mathrm{End}(E_A)$ is the corresponding endomorphism and θ'' is a random isogeny of degree 3 with domain E_A.
$\mathcal{D}_{\mathcal{AP}}$: $\mathcal{D}_{\mathcal{AP}} = \mathcal{D}_1$ if $d = q(2^a - q)$ satisfies Eq. (2), and $\mathcal{D}_{\mathcal{AP}} = \mathcal{D}_2$ otherwise. Note that this is the same distribution as the outputs of Algorithm 3 with $d = q, D_1 = 2^a$ and $D = 2^{a+b}$.

Finally, we define a distribution $\mathcal{Q}$ on $\mathbb{Z}$, which is the distribution of reduced norm of the response ideals I_σ.

Problem 1. Let a be a fixed integer as in the parameter choices and E_A be the public curve. Let $S = \{\omega : E_A \to \star \text{ of degree } 2^a - q\}$ be a set of size $M > \log N_\tau$ where either

1. S is sampled by first sampling $q \sim \mathcal{Q}$, then sampling ω from $\mathcal{D}_\mathcal{U}$;
2. S is sampled by first sampling $q \sim \mathcal{Q}$, then sampling ω from $\mathcal{D}_{\mathcal{AP}}$.

The problem is, given E_A, a, S, to distinguish between the two cases with a polynomial number of queries to $\mathcal{Q}$, FIDIO and to $\mathcal{D}_{\mathcal{AP}}$.

Remark 5. It seems that the most natural way to distinguish the two cases in Problem 1 is to reverse engineer the algorithm that underlies the distribution $\mathcal{D}_{\mathcal{AP}}$. That means, given an isogeny $E_A \to F$ of degree $2^a - q$, one tries to complete the diagrams in Figs. 2 and 3. In the first case, it means to come up with an isogeny from F to E_A of degree $q(2^{a+b} - q(2^a - q))$. This gives rise to an endomorphism on E_A, then one recovers the quaternion element corresponding to this endomorphism and check whether the quaternion element is sampled from $\mathcal{S}_{I_\tau, M(q)}$. The second case is similar, except that one finds an isogeny from F to some curve E'' that is away from E_A by a degree 3 isogeny. This process, requires at least the knowledge of both the endomorphism rings of E_A and F. Therefore, it seems reasonable to assume that an algorithm to solve Problem 1 requires at least $O(2^\lambda)$ time complexity.

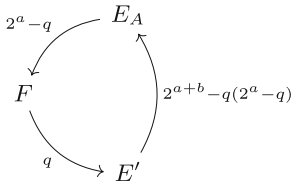


Fig. 2. A diagram that illustrates the computation of the auxiliary path from E_A in the case when Eq. (2) holds for $d = q(2^a - q)$ and $D = 2^{a+b}$.

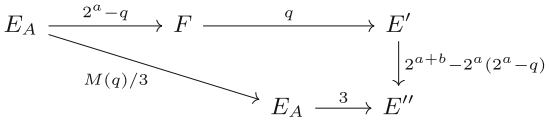


Fig. 3. A diagram that illustrates the computation of the auxiliary path from E_A in the case when Eq. (2) does not hold.

5.2 Soundness of SQIsign2D-East

The proof of soundness of our protocol is quite similar to that of SQIsignHD. Let $(E_1, \phi, K_g, E_4, U_2, V_2)$ and $(E_1, \phi', K'_g, E'_4, U'_2, V'_2)$ are two SQIsign2D-East transcripts with the same commitment E_1 but different challenges $\phi \neq \phi'$. From (K_g, E_4, U_2, V_2) and (K'_g, E'_4, U'_2, V'_2), we can compute efficient representations of $\sigma : E_A \to E_2$ and $\sigma' : E_A \to E'_2$, where E_2 and E'_2 are codomains of ϕ and ϕ', respectively.

Therefore, we obtain an efficient representation of $\alpha = \hat{\sigma'} \circ \phi' \circ \hat{\phi} \circ \sigma \in \mathrm{End}(E_A)$. Finally, the proof that α is non-scalar is exactly same as SQIsignHD since it depends only on the fact that $q = \deg(\sigma)$ and $q' = \deg(\sigma')$ are coprime to $\deg(\phi) = \deg(\phi')$.

5.3 Zero-Knowledge of SQIsign2D-East

We now conclude this section with a proof of the zero-knowledge property of our SQIsign2D-East.

Definition 3. *Given parameters f and a, a random uniform nice degree isogeny oracle (*RUNDIO*) is an oracle taking as input a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$ and returning an efficient representation of a random isogeny $\sigma : E \to E'$ of $(2^a, 2^b, f)_3$-nice degree prime such that:*

(i) The distribution of E' is uniform in the supersingular isogeny graph.
(ii) The conditional distribution of σ given E' is uniform among isogenies $E \to E'$ of $(2^a, 2^b, f)_3$-nice degree.

The existence of RUNDIO is based on the Heuristic assumption Sect. 4.4 applied to our choices of parameter sets.

Definition 4. *A fixed degree isogeny oracle (*FIDIO*) is an oracle taking as input a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$ and an integer N, and outputs a uniformly random isogeny $\varphi : E \to E'$ (in efficient representation) with domain E and degree N.*

Theorem 2. *Assuming that the commitment curve E_1 is both computationally indistinguishable from an elliptic curve chosen uniformly at random in the supersingular isogeny graph, and the hardness of Problem 1. Then the SQIsign2D-East identification protocol is computationally honest-verifier zero-knowledge in the RUNDIO and FIDIO model.*

In other words, there exists a polynomial time simulator $\mathcal{S}$ with access to a RUNDIO and a FIDIO that produces random transcripts which are computationally indistinguishable from honest transcripts.

Proof. A transcript of SQIsign2D-East consists of $(E_1, \phi, K_g, E_4, U_2, V_2)$, where E_1 is a commitment, ϕ is a challenge, (K_g, E_4, U_2, V_2) can be uniquely computed from a q-isogeny σ and a $(2^a - q)$-isogeny ω. (See Algorithm 7 for detail.) The simulator proceeds as follows:

1. Call the RUNDIO on input E_A to get an isogeny $\sigma' : E_A \to E_2'$ of $(2^a, 2^b, f)_3$-nice degree q.
2. Generate an isogeny $\hat{\phi}' : E_2' \to E_1'$ of degree 2^b uniformly at random.
3. Call the FIDIO on input $(E_A, 2^a - q)$, resulting in the isogeny $\omega' : E_A \to E_3'$.
4. Compute (K_g', E_4', U_2', V_2') from (σ', ω').

Then the procedure above gives rise to a simulated transcript as $(E_1', \phi', K_g', E_4', U_2', V_2')$.

Let $(E_1, \phi, K_g, E_4, U_2, V_2)$ be a real transcript where (K_g, E_4, U_2, V_2) is computed from the response isogeny $\sigma : E_A \to E_2$ of degree q and the auxiliary path $\omega : E_A \to E_3$ of degree $2^a - q$. From the properties of the RUNDIO and FIDIO and the assumptions we made in the theorem, we can see that:

1. By the definition of the RUNDIO, E'_2 is uniformly random in the supersingular isogeny graph. Since $\hat{\phi}'$ is a uniformly random isogeny from E'_2 of degree 2^b, its codomain curve E'_1 is also uniformly random in the graph. By assumption, E_1 and E'_1 are computationally indistinguishable.
2. ϕ and ϕ' follow the same distribution as they are generated the same way.
3. Conditional to E'_2, σ' is uniformly random among isogenies between E_A and E'_2 of $(2^a, 2^b)_3$-nice degree by the definition of RUNDIO. Conditional to E_2, σ has the same distribution by construction.
4. Assuming the hardness of Problem 1, conditional to q, ω is computationally indistinguishable from a random isogeny of degree $2^a - q$ from E_A.
5. Item 3,4 combined shows that (K_g, E_4, U_2, V_2) is computationally indistinguishable from (K'_g, E'_4, U'_2, V'_2) as the distributions of (σ, ω) and (σ', ω') are computationally indistinguishable. □

Remark 6. The assumption on the distribution of the commitment curve E_1 made in Theorem 2 is about analyzing the distribution of the outputs of the algorithm `RandIsogImg` given the input norm size. This has been discussed in great detail in [26] where this algorithm was first introduced. Based on the discussions there, we believe this assumption is reasonable.

The Previous Attack Strategy Does Not Apply. To run the attack as in Sect. 4.3 on SQIsign2D-East, we need to able to solve the following problem:

Problem 2. Let $\omega : E_A \to \star$ of degree $2^a - q$ where either

1. ω is sampled from $\mathcal{D}_1$,
2. ω is sampled from $\mathcal{D}_2$.

The problem is, given E_A, ω, to distinguish with success rate 1 between the two cases with a polynomial number of queries to $\mathcal{D}_{\mathcal{AP}}$.

We prove in Proposition 1 that Problem 2 is no easier than Problem 1 assuming that the best algorithm to solve Problem 1 has complexity $O(2^{\lambda'})$ where $\lambda' \geq \lambda$. This seems a reasonable assumption as discussed in Remark 5, and a necessary condition to have our protocol achieve λ-bits security. Proposition 1 then implies that our assumption on the hardness of Problem 1 ensures the hardness of Problem 2, therefore we do not need to make an extra assumption on Problem 2. This agrees with our intuition that if Problem 2 were easy, then our SQIsign2D-East would not be zero-knowledge.

Proposition 1. *If solving Problem 1 requires $O(2^{\lambda'})$ time complexity with $\lambda' \geq \lambda$, then solving Problem 2 requires at least $O(2^{\lambda'})$ time complexity.*

Proof. We prove by contradiction. Suppose there is an algorithm $\mathcal{A}$ that solves Problem 2 in $O(2^{\lambda''})$ where $\lambda'' < \lambda$. Now in Problem 1, we are given with M samples with $M > \log N_\tau$ such that they are either from $\mathcal{D}_{\mathcal{U}}$ or $\mathcal{D}_{\mathcal{AP}}$. We run the distinguishing algorithm $\mathcal{A}$ on around $\log N_\tau \approx \lambda/2$ number of samples to get enough Legendre symbol values with respect to N_τ to uniquely determine

N_τ. These values allows us to recover N_τ in time $O(2^{\lambda/2})$. Given the value of N_τ, then we check whether the remaining $M - \log N_\tau$ samples gives rise to correct Legendre symbols values. In the case when the M samples are from $\mathcal{D}_{\mathcal{U}}$, this fails with a non-negligible probability; and in the case when M samples are from $\mathcal{D}_{\mathcal{AP}}$, this always succeeds. This leads to an algorithm that solves Problem 1 in time $\tilde{O}(2^{\lambda''} + 2^{\lambda/2})$ which is less than $O(2^{\lambda'})$, a contradiction. □

Remark 7. Although the additional 3-isogeny computation will probably be very fast if compared to the rest of the response step, it still introduces a conditional step that is performed only when q fails to satisfy some Legendre symbol condition with respect to N_τ. This creates a side channel that may be exploited leading to a restoration of the original attack. We leave this solution as a future work.

6 Efficiency

In this section, we analyse the efficiency of SQIsign2D-East and CompactSQIsign-2D-East. First, we provide concrete parameters for these protocols, then compare the data sizes of these protocols such as public key size and ciphertext size with SQIsign and SQIsignHD. Finally, we analyse the computational cost of SQIsign2D-East and CompactSQIsign2D-East.

6.1 Parameters

In the following, we give concrete parameters for SQIsign2D-East and CompactSQIsign2D-East satisfying the NIST security level 1, 3, and 5.

NIST level	a	b	f	p
1	127	126	27	$2^{253} \cdot 27 - 1$
3	191	189	35	$2^{380} \cdot 35 - 1$
5	254	253	153	$2^{507} \cdot 153 - 1$

Remark 8. To fit primes into 64-bit limbs, it preferable to use smaller primes such as: $p = 2^{248} \cdot 5 - 1$, $p = 2^{376} \cdot 65 - 1$, and $p = 2^{500} \cdot 27 - 1$ used in SQIsign2D-West [2]. However, if we choose such primes, the challenge length b becomes quite smaller than λ. (e.g. $b = 123 < 128$ for Level 1.) Therefore, we need to extend the challenge length in some way. For example, if there exists a smooth integer $c \mid (p-1)$, we can extend the challenge degree from 2^b to $2^b \cdot c$ by using an additional c-isogeny. This change requires to evaluate points of order c under ψ, which is computed by a 2-dimensional isogeny. In the gluing step of the theta algorithm by [12], we need to compute the x-coordinate of the sum of an evaluated point and a point of order 4. This requires the arithmetic on $\mathbb{F}_{p^4}$. We leave the efficient computation to future work.

6.2 Data Sizes

In this subsection, we compare the signature sizes of SQIsign, SQIsignHD, SQIsign- 2D-East, and CompactSQIsign2D-East using the above parameters. Table 1 shows each signature size. Note that we do not give the signature size of SQIsignHD for the level 3 and 5 since sufficient information to evaluate the signature sizes are not given in [11].

Table 1. Signature size comparison

Security	Protocol	Signature (bytes)
Level 1	SQIsign	177
	SQIsignHD	109
	SQIsign2D-East	**182**
	CompactSQIsign2D-East	**150**
Level 3	SQIsign	263
	SQIsignHD	–
	SQIsign2D-East	**271**
	CompactSQIsign2D-East	**223**
Level 5	SQIsign	335
	SQIsignHD	–
	SQIsign2D-East	**359**
	CompactSQIsign2D-East	**295**

As shown in Table 1, the signature size of SQIsign2D-East is larger than both SQIsign and SQIsignHD for every security level. On the other hand, the signature size of CompactSQIsign2D-East is smaller than SQIsign and larger than SQIsignHD for every security level.

6.3 Computational Cost

We compare the computational costs of SQIsignHD, SQIsign2D-East, and CompactSQIsign2D-East for the security level 1. Table 2 shows the number of isogeny computations of each degree. As Table 2 shows, our protocol does not require any 4-dimensional isogeny computation for the verification. In addition, the number of 2-dimensional isogeny computations is smaller than the number of 4-dimensional isogeny computations in SQIsignHD. Therefore, the verification cost of our protocol is clearly smaller than that of SQIsignHD. As for the key generation and signing, our protocol requires 2-dimensional isogeny computations, whereas SQIsignHD only requires 1-dimensional isogeny computations. Therefore, our protocol is likely to have a larger cost for the key generation and signing.

Finally, in Table 3, we show the actual computational times of SQIsign2D-East and CompactSQIsign2D-East implemented in Julia. The implementation is available as supplementary materials. These are the averages of 100 run times.

Table 2. Number of isogeny computations of each degree

Protocol (Security level 1)		2	3	$(2,2)$	$(2,2,2,2)$
SQIsignHD	keygen	378	234	–	–
	sign	252	312	–	–
	verify	–	78	–	142
SQIsign2D-East	keygen	–	–	253	–
	sign	126	0–4	633	–
	verify	126	0–4	127	–
CompactSQIsign2D-East	keygen	–	–	253	–
	sign	126	0–4	760	–
	verify	126	0–4	127	–

Table 3. Computational times (sec.)

Security	Protocol	keygen	sign	verify
Level 1	SQIsign2D-East	0.50	1.50	0.24
	CompactSQIsign2D-East	0.52	1.87	0.32
Level 3	SQIsign2D-East	1.02	2.91	0.51
	CompactSQIsign2D-East	1.03	3.21	0.56
Level 5	SQIsign2D-East	1.52	4.21	0.72
	CompactSQIsign2D-East	1.57	4.97	0.80

The computational times are measured on a computer with an Intel Core i7-10700K CPU@3.70 Hz without Turbo Boost. The cost evaluation through an optimized implementation is a future work.

7 Conclusion

In this paper, we introduce SQIsign2D-East, a new variant of SQIsignHD, which requires only 2-dimensional isogeny computations for the verification, while SQI-signHD requires 4-dimensional isogeny computations. As a building block of SQIsign2D-East, we construct a new algorithm, which is a generalization of the conventional algorithm called `RandIsogImg`. In addition, we propose CompactSQI-sign2D-East, which has shorter signature size but has larger signing cost.

Both SQIsign2D-East and CompactSQIsign2D-East have less verification costs than SQIsignHD. On the other hand, the signing costs are expected to be larger than SQIsignHD though they are expected to be smaller than SQIsign. The signature size of SQIsign2D-East is longer than both SQIsign and SQIsignHD. The signature size of CompactSQIsign2D-East is shorter than SQIsign but longer than SQIsignHD.

Acknowledgments. We would like to thank Andrea Basso, Luca De Feo, Pierrick Dartois, Antonin Leroux, Luciano Maino, and Benjamin Wesolowski for sharing their work on SQIsign2D-West, and Max Duparc and Tako Boris Fouotsa for sharing their work on SQIPrime with us prior to publication. We also thank the anonymous Asiacrypt 2024 reviewers for their valuable and constructive feedbacks.

References

1. Reza Azarderakhsh, Matthew Campagna, Craig Costello, Luca De Feo, Basil Hess, Amir Jalali, David Jao, Brian Koziel, Brian LaMacchia, Patrick Longa, et al. Supersingular isogeny key encapsulation.*Submission to the NIST Post-Quantum Standardization project*, 152:154–155, 2017.
2. Andrea Basso, Luca De Feo, Pierrick Dartois, Antonin Leroux, Luciano Maino, Giacomo Pope, Damien Robert, and Benjamin Wesolowski. SQIsign2D-West: the Fast, the Small, and the Safer. Cryptology ePrint Archive, Paper 2024/760, 2024. https://eprint.iacr.org/2024/760.
3. Andrea Basso, Luciano Maino, and Giacomo Pope. FESTA: Fast encryption from supersingular torsion attacks. In *ASIACRYPT 2023*, pages 98–126, 2023.
4. Daniel J Bernstein, Luca De Feo, Antonin Leroux, and Benjamin Smith. Faster computation of isogenies of large prime degree. *Open Book Series*, 4(1):39–55, 2020.
5. Wouter Castryck and Thomas Decru. An efficient key recovery attack on SIDH. In *EUROCRYPT 2023*, pages 423–447, 2023.
6. Jorge Chavez-Saab, Maria Corte-Real Santos, Luca De Feo, Jonathan Komada Eriksen, Basil Hess, David Kohel, Antonin Leroux, Patrick Longa, Michael Meyer, Lorenz Panny, Sikhar Patranabis, Christophe Petit, Francisco Rodríguez Henríquez, Sina Schaeffler, and Benjamin Wesolowski. SQIsign. Submission to NIST standardization of additional digital signature schemes. https://sqisign.org, 2023.
7. Mingjie Chen, Antonin Leroux, and Lorenz Panny. SCALLOP-HD: group action from 2-dimensional isogenies. In *PKC 2024*, pages 190–216. Springer, 2024.
8. Henri Cohen. *A course in computational algebraic number theory*, volume 138. Springer Science & Business Media, 2013.
9. Maria Corte-Real Santos, Jonathan Komada Eriksen, Michael Meyer, and Krijn Reijnders. Apréssqi: extra fast verification for sqisign using extension-field signing. In *EUROCRYPT 2024*, pages 63–93. Springer, 2024.
10. Romain Cosset and Damien Robert. Computing (l, l)-isogenies in polynomial time on Jacobians of genus 2 curves. *Mathematics of Computation*, 84(294):1953–1975, 2015.
11. Pierrick Dartois, Antonin Leroux, Damien Robert, and Benjamin Wesolowski. SQISignHD: new dimensions in cryptography. In *EUROCRYPT 2024*, pages 3–32. Springer, 2024.
12. Pierrick Dartois, Luciano Maino, Giacomo Pope, and Damien Robert. An Algorithmic Approach to $(2, 2)$-isogenies in the Theta Model and Applications to Isogeny-based Cryptography. Cryptology ePrint Archive, Paper 2023/1747, 2023. https://eprint.iacr.org/2023/1747.
13. Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski. SQISign: Compact post-quantum signatures from quaternions and isogenies. In *ASIACRYPT 2020*, pages 64–93, 2020.

14. Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski. SQISign: Compact post-quantum signatures from quaternions and isogenies. In *Asiacrypt Vol. 1*, volume 12491 of *Lecture Notes of Computer Science*, pages 64–93. Springer, 2020.
15. Luca De Feo, Antonin Leroux, Patrick Longa, and Benjamin Wesolowski. New algorithms for the deuring correspondence: towards practical and secure sqisign signatures. In *EUROCRYPT 2023*, pages 659–690. Springer, 2023.
16. Max Deuring. Die typen der multiplikatorenringe elliptischer funktionenkörper. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 14:197–272, 1941.
17. Max Duparc and Tako Boris Fouotsa. SQIPrime: A dimension 2 variant of SQISignHD with non-smooth challenge isogenies. Cryptology ePrint Archive, Paper 2024/773, 2024. https://eprint.iacr.org/2024/773.
18. Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *CRYPTO 1986*, pages 186–194. Springer, 1986.
19. Everett W. Howe, Franck Leprévost, and Bjorn Poonen. Large torsion subgroups of split Jacobians of curves of genus two or three. *Forum Mathematicum*, 12(3):315–364, 2000.
20. David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In *PQCrypto 2011*, pages 19–34, 2011.
21. Ernst Kani. The number of curves of genus two with elliptic differentials. *Journal für die reine und angewandte Mathematik*, 485:93–122, 1997.
22. Markus Kirschmer and John Voight. Algorithmic enumeration of ideal classes for quaternion orders. *SIAM Journal on Computing*, 39(5):1714–1747, 2010.
23. David Lubicz and Damien Robert. Computing isogenies between abelian varieties. *Compositio Mathematica*, 148(5):1483–1515, 2012.
24. Luciano Maino, Chloe Martindale, Lorenz Panny, Giacomo Pope, and Benjamin Wesolowski. A direct key recovery on SIDH. *EUROCRYPT 2023*, pages 448–471, 2023.
25. Tomoki Moriya. IS-CUBE: An isogeny-based compact KEM using a boxed SIDH diagram. Cryptology ePrint Archive, Paper 2023/1506, 2023. https://eprint.iacr.org/2023/1506.
26. Kohei Nakagawa and Hiroshi Onuki. QFESTA: Efficient algorithms and parameters for FESTA using quaternion algebras. In *Annual International Cryptology Conference*, pages 75–106. Springer, 2024.
27. Hiroshi Onuki and Kohei Nakagawa. Ideal-to-isogeny algorithm using 2-dimensional isogenies and its application to SQIsign. Cryptology ePrint Archive, Paper 2024/778, 2024. https://eprint.iacr.org/2024/778.
28. Damien Robert. Breaking SIDH in polynomial time. In *EUROCRYPT 2023*, pages 472–503, 2023.
29. Maria Corte-Real Santos, Craig Costello, and Benjamin Smith. Efficient (3,3)-isogenies on fast kummer surfaces. Cryptology ePrint Archive, Paper 2024/144, 2024.
30. Benjamin Andrew Smith. *Explicit endomorphisms and correspondences*. Phd thesis, University of Sydney, 2005.
31. Lázlo Tóth. A survey of gcd-sum functions. *Journal of Integer Sequences*, 13:article 10.8.1, 2010.
32. Jacques Vélu. Isogénies entre courbes elliptiques. *Comptes-Rendus de l'Académie des Sciences*, 273:238–241, 1971.