SQIsign2DPush: Faster Signature Scheme Using 2-Dimensional Isogenies

Kohei Nakagawa[1] and Hiroshi Onuki[2][0000−0002−0202−8918]

[1] NTT Social Informatics Laboratories, Japan kohei.nakagawa@ntt.com
[2] The University of Tokyo, Japan hiroshi-onuki@g.ecc.u-tokyo.ac.jp

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 for additional signature. SQIsign has attracted much attention because of its very short signature and key size among candidates for the NIST PQC standardization. Recently, many new signature schemes using high-dimensional isogenies have been proposed, such as: SQIsignHD, SQIsign2D-West, SQIsgn2D-East, and SQIPrime. Last year, SQIsign advanced to Round 2 of the NIST competition and was updated to version 2.0 (we call it SQIsign-v2.0), which is based on SQIsign2D-West. SQIsign-v2.0 achieves smaller signature sizes and faster verification. However, the signing costs are relatively high. In this paper, we propose a new signature scheme 'SQIsign2DPush', which has a smaller signing cost than SQIsign-v2.0 while the signature size and the verification cost are almost the same.

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 for additional signature [8]. SQIsign has attracted much attention because of its very short signature and key size among the candidates for the NIST PQC additional signature.

SIDH [22] and SIKE [21] were also representative isogeny-based cryptographies. However, recent attacks [7,26,30] broke the security of SIDH and SIKE. These attacks show that any isogeny can be efficiently recovered from its evaluation on sufficiently large torsion subgroup by computing high-dimensional isogenies. In response, a lot of cryptographic applications using high-dimensional isogenies have been studied: encryption schemes such as FESTA [5], QFESTA [27], and POKÉ [4], and variants of SQIsign such as SQIsignHD [11], SQIsign2D-West [3], SQIsign2D-East [28], and SQIPrime [16].

SQIsignHD is the first signature scheme using high-dimensional isogenies and achieves much smaller signature sizes and higher singing performance than SQIsign. However, it requires 4 or 8-dimensional isogeny computations for verification, which leads to a large verification cost. On the other hand, SQIsign2D-West, SQIsign2D-East, and SQIPrime use 2-dimensional isogenies for verification. As a result, all of these schemes achieved high verification speeds, although the signing costs have slowed down compared to SQIsignHD. In particular, SQIsign2D-West was adopted as the updated version of SQIsign [1] (we call it SQIsign-v2.0) when SQIsign advanced to Round 2 of the NIST competition. However, the signing cost of SQIsign-v2.0 remains relatively slow compared to other NIST candidates. One of the dominant factors in the signing cost of SQIsign-v2.0 is the computation of an isogeny called the 'auxiliary isogeny'. This is an isogeny of a certain (generally non-smooth) degree from a given elliptic curve. In SQIsign-v2.0, many 2-dimensional isogenies are required to compute this auxiliary isogeny.

1.1 Contributions

In this paper, we make the following contributions.

- We construct a new algorithm named `PushRandIsog`, which computes an auxiliary isogeny, i.e., a given (generally non-smooth) degree isogeny from a given elliptic curve E.
- Using `PushRandIsog` as a building block, we construct a new variant of SQIsign. We name our new signature scheme 'SQIsign2DPush'.
- Our SQIsign2DPush has smaller signing costs than SQIsign-v2.0 while the signature sizes and the verification costs are almost the same.
- We prove the security of SQIsign2DPush using the *hint-assisted wHVZK* framework [1], which is also used in SQIsign-v2.0.
- We give concrete parameters of SQIsign2DPush for the NIST security level 1, 3, and 5. Under these parameter settings, we analyse the signature sizes and computational costs and compare with some protocols such as SQIsign-v2.0.

Our SQIsign2DPush is based mainly on SQIsignHD [11] rather than SQIsign-v2.0 [1]. In fact, the key generation, the commitment, and the challenge are almost the same as SQIsignHD. In the response, however, we compute an *auxiliary isogeny* of a given (probabily non-smooth) degree from a given curve as in SQIsign-v2.0. To compute such an isogeny, we construct a new algorithm named `PushRandIsog`. Let E_0 be a supersingular elliptic curve whose endomorphism ring is special p-extremal [24, Section 2.3]. Our `PushRandIsog` takes an integer d and an isogeny $\psi : E_0 \to E$ as input, and outputs a d-isogeny φ from E. Our main idea for `PushRandIsog` is to compute a d-isogeny φ_0 from E_0 using the idea of QFESTA [27, Algorithm 2] and then *push* the isogeny φ_0 by the given isogeny $\psi : E_0 \to E$. Then, we obtain a d-isogeny φ from E. For this, we use `FastCommit` from SQIsignHD [11, Algorithm 1] and a technique from POKÉ [4].

1.2 Organizations

In Section 2, we give some notation and background knowledge used in this paper. In Section 3, we propose our new algorithm `PushRandIsog`. In Section 4, we propose our new signature scheme SQIsign2DPush and its security is analysed in Section 5. In Section 6, we give some concrete parameters for SQIsign2DPush and analyse the signature sizes and the computational costs of SQIsign2DPush. Finally, in Section 7, we give the conclusion of this paper.

2 Preliminaries

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

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. 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 that indicates the failure of an algorithm. We denote the neutral element of an elliptic curve by O.

2.2 Abelian varieties and Isogenies

In this paper, we mainly use principally polarized superspecial abelian varieties defined over a finite field of characteristic p of dimension one or two. 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 with 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 isomorphism. For an isogeny $\varphi : A \to B$ between principally polarized abelian varieties, there exists an isogeny $B \to A$ called the *dual isogeny* (with respect to the polarization). We denote the dual isogeny of φ by $\hat{\varphi}$.

Let $\varphi : A \to B$ and $\psi : A \to C$ be separable isogenies of coprime degrees. If a separable isogeny $\psi' : B \to D$ satisfies $\ker \psi' = \varphi(\ker \psi)$, 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 $\mathbb{F}_{p^2}$ and let n be an integer such that the n-torsion subgroup $E[n]$ is contained in $E(\mathbb{F}_{p^2})$. Then we can compute a deterministic basis of $E[n]$ by using deterministic algorithms such as [8, Algorithm 3] and [1, Algorithm 2.2].

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.

If the dimension of A is two and $\ell = 2$, then we can compute $(2, 2)$-isogeny with kernel G by the recent formulas proposed by Dartois, Maino, Pope, and Robert [12].

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. In this paper, we assume that an isogeny is given by an efficient representation.

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 $\mathcal{B}_{p,\infty}$. In this paper, we only consider $\mathcal{B}_{p,\infty}$.

Let $\alpha = x + y\mathbf{i} + z\mathbf{j} + t\mathbf{k} \in \mathcal{B}_{p,\infty}$ with $x, y, z, t \in \mathbb{Q}$. The *conjugate* of α is $x - y\mathbf{i} - z\mathbf{j} - t\mathbf{k}$ and is denoted by $\bar{\alpha}$. The *reduced norm* of α is $\alpha\bar{\alpha}$ and is denoted by $n(\alpha)$. In this paper, we simply call it the *norm.*

An *order* $\mathcal{O}$ of $\mathcal{B}_{p,\infty}$ is a subring of $\mathcal{B}_{p,\infty}$ 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 $\mathcal{B}_{p,\infty}$. 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 $\mathcal{O}$, the (integral) *left $\mathcal{O}$-ideal* I is a $\mathbb{Z}$-lattice of rank 4 satisfying $I \subseteq \mathcal{O}$ and $\mathcal{O} \cdot I \subseteq I$. The *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 norms of the elements of I. In this paper, we simply call it the *norm*.

The *ideal equivalence* denoted by $I \sim J$ means that there exists $\beta \in \mathcal{B}_{p,\infty}^*$ such that $J = I\beta$. In particular, the ideal $J = I\frac{\bar{\alpha}}{n(I)}$ for $\alpha \in I$ is equivalent to I and has norm $n(\alpha)/n(I)$. In this paper, we call the value $n(\alpha)/n(I)$ the *normalized norm* of $\alpha \in I$ and denote it by $q_I(\alpha)$. It is known that there always exists $\alpha \in I$ such that $q_I(\alpha) \leq 2\sqrt{2p}/\pi$, that is, there exists an ideal $J \sim I$ of norm $\leq 2\sqrt{2p}/\pi$.

Deuring Correspondence. Deuring [15] 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 (*Deuring correspondence*) where a supersingular elliptic curve E over $\mathbb{F}_{p^2}$ corresponds 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 the 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 $\mathcal{O}_0 \coloneqq \mathbb{Z}\langle 1, \mathbf{i}, \frac{\mathbf{i}+\mathbf{j}}{2}, \frac{1+\mathbf{k}}{2}\rangle$. An isomorphism is given by $\iota \mapsto \mathbf{i}$ and $\pi \mapsto \mathbf{j}$. From now on, we assume that $\mathrm{End}(E_0)$ and $\mathcal{O}_0$ always correspond to each other through this isomorphism.

Let $P \in E_0(\mathbb{F}_{p^2})$ and $\alpha = x + y\mathbf{i} + z\frac{\mathbf{i}+\mathbf{j}}{2} + t\frac{1+\mathbf{k}}{2} \in \mathcal{O}_0$ for $x,y,z,t \in \mathbb{Z}$. Given P and x,y,z,t, we 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 torsion basis P_0, Q_0 of E_0 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 isogeny and ideal. 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 $\mathcal{O}_1 \cong \mathrm{End}(E_1)$. We fix an isomorphism ι between $\mathrm{End}(E_1)$ and $\mathcal{O}_1$. Let $\varphi : E_1 \to E_2$ be a d-isogeny, then the isogeny φ can be associated to a left $\mathcal{O}_1$-ideal I_φ of norm d via the isomorphism ι as follows:

$$I_\varphi = \{\iota(\psi \circ \varphi) \mid \psi \in \mathrm{Hom}(E_2, E_1)\}.$$

This ideal I_φ is also a right $\mathcal{O}_2$-ideal for a maximal order $\mathcal{O}_2$ satisfying $\mathcal{O}_2 \cong \mathrm{End}(E_2)$. In particular, φ induces an isomorphism from $\mathrm{End}(E_2)$ to $\mathcal{O}_2$ defined by

$$\alpha \mapsto \frac{1}{\deg \varphi}\iota(\widehat{\varphi} \circ \alpha \circ \varphi).$$

Moreover, two isogenies $\varphi, \psi : E_1 \to E_2$ with the same domain and codomain correspond to the equivalent ideals $I_\varphi \sim I_\psi$.

2.4 Kani's Lemma

In this subsection, we introduce Kani's lemma [23], which is used in the attacks to SIDH [7,26,30].

Theorem 1 ([26, Theorem 1]) *Let d_1, d_2, and D be pairwise coprime integers such that $D = d_1 + d_2$, and let E, E_1, E_2, and F be elliptic curves connected by the following diagram of isogenies:*

$$\begin{tikzcd}
E \arrow[r, "\varphi_2"] \arrow[d, "\varphi_1"'] & E_2 \arrow[d, "\varphi_1'"] \\
E_1 \arrow[r, "\varphi_2'"'] \arrow[ur, "f"] & F,
\end{tikzcd}$$

where $\varphi_2' \circ \varphi_1 = \varphi_1' \circ \varphi_2$, $f = \varphi_2 \circ \hat{\varphi}_1$, and $\deg(\varphi_i) = \deg(\varphi_i') = d_i$ for $i \in \{1, 2\}$. Then, the map

$$\Phi = \begin{pmatrix} \hat{\varphi}_1 & -\hat{\varphi}_2 \\ \varphi_2' & \varphi_1' \end{pmatrix} : E_1 \times E_2 \to E \times F \tag{1}$$

is a (D, D)-isogeny with respect to the natural product polarizations on $E_1 \times E_2$ and $E \times F$, and has kernel $\{([d_2]P, f(P)) \mid P \in E_1[D]\}$. Conversely, a (D, D)-isogeny with this kernel is equal to Φ up to isomorphism.

2.5 Existing Algorithms

In this subsection, we introduce the existing algorithms used in this paper. Let p be a prime such that $p \equiv 3 \mod 4$ and let $E_0/\mathbb{F}_{p^2}$ be a supersingular elliptic curve defined by $y^2 = x^3 + x$. We also write $\mathcal{O}_0 = \mathbb{Z}\langle 1, \mathbf{i}, \frac{\mathbf{i}+\mathbf{j}}{2}, \frac{1+\mathbf{k}}{2}\rangle$, which is a maximal order of $\mathcal{B}_{p,\infty}$ isomorphic to $\mathrm{End}(E_0)$.

$\texttt{FullRepresentInteger}_{\mathcal{O}_0}(M)$ [14, Algorithm 1]: This algorithm takes an integer $M > p$ as input and outputs $\alpha \in \mathcal{O}_0$ of norm M. From the Deuring correspondence, we can see this α as an endomorphism of E_0 of degree M.

$\texttt{FastDoublePath}_{2^{e_1},3^{e_2}}/\texttt{FastCommit}_{2^{e_1},3^{e_2}}$ [11, Algorithm 1]: These algorithms are used in the setting where $p = c \cdot 2^{e_1} 3^{e_2} - 1$. Then, `FastDoublePath` takes 2^{e_1} and 3^{e_2} as input and outputs a 2^{2e_1}-isogeny $\varphi_1 : E_0 \to E$, a 3^{2e_2}-isogeny $\varphi_2 : E_0 \to E$, and their corresponding left $\mathcal{O}_0$-ideals I_1, I_2, where these two isogenies have the same codomain E. `FastCommit` is similar to `FastDoublePath`, but it only outputs a 3^{2e_2}-isogeny $\varphi_2 : E_0 \to E$ and its corresponding ideal I_2. We decompose φ_1 into $\varphi_1'' \circ \varphi_1'$ such that $\deg \varphi_1' = \deg \varphi_1'' = 2^{e_1}$. Then, φ_1 is efficiently represented by generators of $\ker \varphi_1'$ and $\ker \varphi_1''$ since $E_0[2^{e_1}]$ is $\mathbb{F}_{p^2}$-rational. φ_2 is similarly represented.

$\texttt{IsogToIdeal}(\psi, \varphi, I_\varphi)$ [11, Algorithm 10]: This is an algorithm to convert an isogeny to its corresponding ideal. Such an algorithm is used broadly in many isogeny-based cryptographies. In this paper, we use the algorithm proposed in SQIsignHD. It takes as input an isogeny ψ, an isogeny φ from E_0 to the domain of ψ, and a left $\mathcal{O}_0$-ideal I_φ corresponding to φ, where $\deg\varphi$ is smooth and coprime to $\deg\psi$ and $E_0[\deg\varphi]$ is in $E_0(\mathbb{F}_{p^2})$. Then it outputs the ideal I_ψ corresponding to ψ via the isomorphism induced by φ.

`KaniEval`$(d_1, d_2, P_1, Q_1, P_2, Q_2; \mathcal{S}_1; \mathcal{S}_2)$ [28, Section 2.4]: Here, we use the same notation in Theorem 1. According to Theorem 1, we can evaluate a d_1-isogeny $\hat{\varphi}_1$ and a d_2-isogeny $\hat{\varphi}_2$ from the knowledge of $\varphi_2 \circ \hat{\varphi}_1|_{E_1[D]}$. We denote the algorithm by `KaniEval`. More precisely, `KaniEval` takes coprime integers d_1, d_2, a basis P_1, Q_1 of $E_1[D]$, $P_2 = \varphi_2 \circ \hat{\varphi}_1(P_1), Q_2 = \varphi_2 \circ \hat{\varphi}_1(Q_1) \in E_2[D]$, a finite subset $\mathcal{S}_1 \subseteq E_1$, and a finite subset of $\mathcal{S}_2 \subseteq E_2$ as input. Then, `KaniEval` returns the codomain E of $\widehat{\varphi}_1$ and $\widehat{\varphi}_2$, the image of $\mathcal{S}_1$ under $\hat{\varphi}_1$, and the image of $\mathcal{S}_2$ under $\hat{\varphi}_2$. In this algorithm, we compute a (D, D)-isogeny, which can be computed efficiently when D is smooth. In this paper, we consider the case where D is a power of 2.

2.6 Signature Schemes in High-Dimension

Here, we briefly describe three signature schemes using high-dimensional isogenies. First, we introduce SQIsignHD, which is the first signature scheme using high-dimensional isogenies. In our protocol, we use several algorithms proposed in SQIsignHD. Secondly, we introduce SQIsign-v2.0, one of the NIST candidates. Finally, we introduse SQIsign2D-East, which is faster than SQIsign-v2.0.

SQIsignHD SQIsignHD is a signature scheme proposed in [11] in 2023. There are two types of SQIsignHD, one using 4-dimensional isogenies and the other using 8-dimensional isogenies for the verification. In this paper, we only consider SQIsignHD using 4-dimensional isogenies. For more details, refer to [11].

First, we show the public parameters of SQIsignHD. Let $p = c \cdot 2^{e_1} \cdot 3^{e_2} - 1$ be a cryptographic size prime with $e_1, e_2 \in \mathbb{N}$ and $c \in \mathbb{N}$ as small as possible. Let $E_0 : y^2 = x^3 + x$ be the elliptic curve defined over $\mathbb{F}_{p^2}$. Furthermore, we say that an odd integer d is 2^{e_1}-*good* if there exist integers m_1, m_2 satisfying $m_1^2 + m_2^2 = 2^{e_1} - d$.

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

$$\begin{tikzcd}
E_0 \arrow[r, shift left, "\varphi_{\mathrm{sk1}}"] \arrow[r, shift right, "\varphi_{\mathrm{sk2}}"'] \arrow[d, "\varphi_{\mathrm{com}}"'] & E_{\mathrm{pk}} \arrow[d, "\varphi_{\mathrm{chl}}"] \\
E_{\mathrm{com}} \arrow[r, "\varphi_{\mathrm{rsp}}"'] & E_{\mathrm{chl}}
\end{tikzcd}$$

In the following, we describe the overview of SQIsignHD Σ-protocol.

gen: The prover generates a random 2^{2e_1}-isogeny $\varphi_{\mathrm{sk1}} : E_0 \to E_{\mathrm{pk}}$, a random 3^{2e_2}-isogeny $\varphi_{\mathrm{sk2}} : E_0 \to E_{\mathrm{pk}}$, and the corresponding ideals $I_{\mathrm{sk1}}, I_{\mathrm{sk2}}$ via $\texttt{FastDoublePath}_{2^{e_1},3^{e_2}}$. Then the prover publishes the curve E_{pk} as the public key.

com: The prover generates a random 3^{2e_2}-isogeny $\varphi_{\mathrm{com}} : E_0 \to E_{\mathrm{com}}$ and the corresponding ideal I_{com} via $\texttt{FastCommit}_{2^{e_1},3^{e_2}}$. The prover then sends E_{com} to the verifier as the commitment.

chl: The verifier generates a random 3^{e_2}-isogeny $\varphi_{\mathrm{chl}} : E_{\mathrm{pk}} \to E_{\mathrm{chl}}$ and sends it to the prover.

rsp: The prover first computes the ideal I_{chl} corresponding to φ_{chl} via the isomorphism induced by φ_{sk1} using $\texttt{IsogToIdeal}(\varphi_{\mathrm{chl}}, \varphi_{\mathrm{sk1}}, I_{\mathrm{sk1}})$. Let $\gamma \in \mathcal{O}_0$ be a quaternion corresponding to $\widehat{\varphi}_{\mathrm{sk2}} \circ \varphi_{\mathrm{sk1}}$. Then the prover computes $I = \bar{I}_{\mathrm{com}} \cdot I_{\mathrm{sk1}} \cdot I_{\mathrm{chl}} \cdot \frac{\bar{\gamma}}{2^{2e_1}}$, which corresponds to $\varphi_{\mathrm{chl}} \circ \varphi_{\mathrm{sk2}} \circ \hat{\varphi}_{\mathrm{com}}$ and finds a random ideal I_{rsp} equivalent to I of 2^{e_1}-good norm d_{rsp}. Finally, the prover sends to the verifier an efficient representation of the d_{rsp}-isogeny $\varphi_{\mathrm{rsp}} : E_{\mathrm{com}} \to E_{\mathrm{chl}}$ corresponding to I_{rsp}.

verf: The verifier checks whether the response sent by the prover correctly represents a d_{rsp}-isogeny $\varphi_{\mathrm{rsp}} : E_{\mathrm{com}} \to E_{\mathrm{chl}}$. If it does, the verifier accepts the response.

As an efficient representation of the d_{rsp}-isogeny φ_{rsp}, the prover sends d_{rsp}, $\varphi_{\mathrm{rsp}}(P_{\mathrm{com}})$, and $\varphi_{\mathrm{rsp}}(Q_{\mathrm{com}})$ for the deterministic basis $(P_{\mathrm{com}}, Q_{\mathrm{com}})$ of $E_{\mathrm{com}}[2^{e_1}]$. The idea of the verification is to use Theorem 1 to recover φ_{rsp}. To apply Theorem 1, the verifier needs to compute a $(2^{e_1} - d_{\mathrm{rsp}})$-isogeny from E_{com}. However, this task is hard since the degree $2^{e_1} - d_{\mathrm{rsp}}$ is generally non-smooth. The verifier instead computes the 2-dimensional endomorphism over $E_{\mathrm{com}} \times E_{\mathrm{com}}$ of degree $2^{e_1} - d_{\mathrm{rsp}}$ as follows:

1. Find two integers m_1, m_2 satisfying $m_1^2 + m_2^2 = 2^{e_1} - d_{\mathrm{rsp}}$.
2. Let Φ_{aux} be the 2-dimensional endomorphism of degree $m_1^2 + m_2^2 = 2^{e_1} - q$ defined as follows:
$$\Phi_{\mathrm{aux}} = \begin{pmatrix} m_1 & -m_2 \\ m_2 & m_1 \end{pmatrix}.$$

Let I_2 be the 2×2 identity matrix. Under the following diagram, the verifier can recover φ_{rsp} by computing 4-dimensional 2^{e_1}-isogeny. In this step, the verifier uses an extension of Theorem 1 to dimension 4 by Robert [30].

$$\begin{tikzcd}
E_{\mathrm{com}} \times E_{\mathrm{com}} \arrow[r, "\varphi_{\mathrm{rsp}} \cdot \mathrm{I}_2"] \arrow[d, "\Phi_{\mathrm{aux}}"'] & E_{\mathrm{chl}} \times E_{\mathrm{chl}} \arrow[d, "\Phi_{\mathrm{aux}}"] \\
E_{\mathrm{com}} \times E_{\mathrm{com}} \arrow[r, "\varphi_{\mathrm{rsp}} \cdot \mathrm{I}_2"'] & E_{\mathrm{chl}} \times E_{\mathrm{chl}}.
\end{tikzcd}$$

SQIsign-v2.0 SQIsign-v2.0 is obtained by applying Fiat-Shamir transform on the Σ-protocol based on the following diagram.

$$\begin{tikzcd}
E_0 \arrow[r, "\varphi_{\mathrm{sk}}"] \arrow[d, "\varphi_{\mathrm{com}}"'] & E_{\mathrm{pk}} \arrow[d, "\varphi_{\mathrm{chl}}"] \\
E_{\mathrm{com}} \arrow[r, "\varphi_{\mathrm{rsp}}"'] \arrow[d, "\varphi_{\mathrm{aux}}"'] & E_{\mathrm{chl}} \arrow[d, "\varphi'_{\mathrm{aux}}"] \\
E_{\mathrm{aux}} & E'_{\mathrm{aux}}
\end{tikzcd}$$

As a public parameter of SQIsign-v2.0 [1], we use a prime of the form $p = c \cdot 2^e - 1$ with $e \approx 2\lambda$ and $c \in \mathbb{N}$ as small as possible. In addition, we take an integer e_{rsp} such that $2^{e_{\mathrm{rsp}}} \geq 2\sqrt{2p}/\pi$, which is the upper bound of φ_{rsp}. The main difference of SQIsign-v2.0 from SQIsignHD is that the verifier embeds φ_{rsp} in a 2-dimensional isogeny. To do so, the prover computes a 1-dimensional isogeny φ_{aux} from E_{com} of degree $d_{\mathrm{aux}} := 2^{e_{\mathrm{rsp}}} - \deg \varphi_{\mathrm{rsp}}$. In this paper, we call such an isogeny an 'auxiliary isogeny.'

Before describing the overview of the SQIsign-v2.0 Σ-protocol, we briefly introduce an algorithm called `IdealToIsogeny` proposed in [3, Algorithm 3]. This is an algorithm to convert a left $\mathcal{O}_0$-ideal I to its corresponding isogeny φ_I. `IdealToIsogeny` consists of three steps: (1) find two elements in I with suitable norms; (2) compute two isogenies from E_0 using Theorem 1; (3) apply Theorem 1 again to obtain φ_I. In total, `IdealToIsogeny` requires approximately 4λ calls to compute a $(2, 2)$-isogeny, two calls to `FullRepresentInteger`, and one call to a lattice reduction algorithm.

Now, we show an overview of the SQIsign-v2.0 Σ-protocol. Note that we will only describe the case where $d_{\mathrm{rsp}} := \deg \varphi_{\mathrm{rsp}}$ is odd for simplicity. For more detail, see [3,1].

gen: The prover generates a random left $\mathcal{O}_0$-ideal I_{sk} of norm N_{sk}, where N_{sk} is an odd integer $\approx 2^{4\lambda}$. Then the prover computes the isogeny $\varphi_{\mathrm{sk}} : E_0 \to E_{\mathrm{pk}}$ corresponding to I_{sk} via `IdealToIsogeny`. Then the prover publishes the curve E_{pk} as the public key.

com: The prover generates a random left $\mathcal{O}_0$-ideal I_{com} of norm N_{com}, where N_{com} is an odd integer $\approx 2^{4\lambda}$. Then the prover computes the isogeny $\varphi_{\mathrm{com}} : E_0 \to E_{\mathrm{com}}$ corresponding to I_{com} via `IdealToIsogeny`. Then the prover publishes the curve E_{com} as the commitment.

chl: The verifier generates a random 2^e-isogeny $\varphi_{\mathrm{chl}} : E_{\mathrm{pk}} \to E_{\mathrm{chl}}$ and sends it to the prover.

rsp: The prover performs as follows:

1. As in SQIsignHD, compute the ideal I corresponding to $\varphi_{\mathrm{chl}} \circ \varphi_{\mathrm{sk}} \circ \hat{\varphi}_{\mathrm{com}}$ and find a random ideal I_{rsp} equivalent to I of norm $d_{\mathrm{rsp}} \leq 2^{e_{\mathrm{rsp}}}$.

2. Generate a random left $\mathcal{O}_0$-ideal I''_{aux} of norm $d_{\mathrm{aux}} := 2^{e_{\mathrm{rsp}}} - d_{\mathrm{rsp}}$ and let $J = (I_{\mathrm{com}} I_{\mathrm{rsp}}) \cap I''_{\mathrm{aux}}$. Then J corresponds to $\varphi'_{\mathrm{aux}} \circ \varphi_{\mathrm{rsp}} \circ \varphi_{\mathrm{com}}$ for a d_{aux}-isogeny $\varphi'_{\mathrm{aux}} : E_{\mathrm{chl}} \to E'_{\mathrm{aux}}$.
3. Compute an isogeny φ_J corresponding to J via `IdealToIsogeny`.
4. From the knowledge of $\varphi_J \circ \widehat{\varphi}_{\mathrm{com}} = [N_{\mathrm{com}}]\varphi'_{\mathrm{aux}} \circ \varphi_{\mathrm{rsp}}$, compute the pull-back φ_{aux} of φ'_{aux} by φ_{rsp} via `KaniEval`. Since $\deg \varphi_{\mathrm{rsp}} + \deg \varphi'_{\mathrm{aux}} = 2^{e_{\mathrm{rsp}}}$, we require $e_{\mathrm{rsp}} \approx \lambda$ times $(2,2)$-isogeny computations.
5. Send $\widehat{\varphi}_{\mathrm{aux}} \circ \varphi_{\mathrm{rsp}}|_{E_{\mathrm{aux}}[2^{e_{\mathrm{rsp}}}]}$ as a response.

verf: The verifier tries to recover the isogeny φ_{rsp} from $\widehat{\varphi}_{\mathrm{aux}} \circ \varphi_{\mathrm{rsp}}|_{E_{\mathrm{aux}}[2^{e_{\mathrm{rsp}}}]}$ via `KaniEval`. If it succeeds, the verifier accepts the response.

SQIsign2D-East SQIsign2D-East [28] is also obtained by applying Fiat-Shamir transform on the Σ-protocol based on the following diagram.

$$\begin{tikzcd}
E_0 \arrow[r, "\varphi_{\mathrm{com}}"] \arrow[d, "\varphi_{\mathrm{sk}}"'] & E_{\mathrm{com}} \arrow[d, "\varphi_{\mathrm{chl}}"] \\
E_{\mathrm{pk}} \arrow[r, "\varphi_{\mathrm{rsp}}"'] \arrow[d, "\varphi_{\mathrm{aux}}"'] & E_{\mathrm{chl}} \\
E_{\mathrm{aux}} &
\end{tikzcd}$$

SQIsign2D-East is similar to SQIsign-v2.0. The main difference of SQIsign2D-East from SQIsign-v2.0 is the algorithm to compute an auxiliary isogeny φ_{aux}. They use the algorithm named `GenRandIsogImg` [28, Algorithm 2], which takes an integer d_{aux} and an isogeny $\varphi_{\mathrm{sk}} : E_0 \to E_{\mathrm{pk}}$ and outputs a d_{aux}-isogeny φ_{aux} from E_{pk}.

The cost of `GenRandIsogImg` is smaller than `IdealToIsogeny`, but it requires strong constraints on d_{aux} and $\deg \varphi_{\mathrm{sk}}$. In additon, the distribution of φ_{aux} output by `GenRandIsogImg` is not uniform but depends on φ_{sk}. Therefore, SQIsign2D-East is faster than SQIsign-v2.0 but its security assumption is stronger.

Another difference of SQIsign2D-East from SQIsign-v2.0 is that they use `RandIsogImg` proposed in QFESTA [27, Algorithm 2] to generate φ_{sk} and φ_{com}.

Summarizing Three Protocols Finnaly, we summarize the above three protocols in Table 1, which shows the algorithm each protocol uses to compute $\varphi_{\mathrm{sk}}(\varphi_{\mathrm{sk1}}$ and φ_{sk2} for SQIsignHD), φ_{com}, and φ_{aux}.

3 New Algorithm: `PushRandIsog`

In this section, we propoes a new algorithm named `PushRandIsog`, which is used to compute an auxiliary isogeny in our new scheme SQIsign2DPush. From now on, we use the following notation.

Table 1. The algorithm each protocol uses to compute each isogeny.

Protocol	$\varphi_{\mathrm{sk}}(\varphi_{\mathrm{sk1}}$ and φ_{sk2})	φ_{com}	φ_{aux}
SQIsignHD	`FastDoublePath`	`FastCommit`	-
SQIsign-v2.0	`IdealToIsogeny`	`IdealToIsogeny`	`IdealToIsogeny`
SQIsign2D-East	`RandIsogImg`	`RandIsogImg`	`GenRandIsogImg`

- A cryptographic size prime $p = c \cdot 2^{e_1} \cdot 3^{e_2} - 1$ with $e_1, e_2 \in \mathbb{N}$ and $c \in \mathbb{N}$ as small as possible.
- The elliptic curve $E_0 : y^2 = x^3 + x$ defined over $\mathbb{F}_{p^2}$.
- The deterministic basis P_0, Q_0 of $E_0[2^{e_1}3^{e_2}]$.
- Maximal order $\mathcal{O}_0 \coloneqq \mathbb{Z}\langle 1, \mathbf{i}, \frac{\mathbf{i}+\mathbf{j}}{2}, \frac{1+\mathbf{k}}{2}\rangle$ of $\mathcal{B}_{p,\infty}$. This order $\mathcal{O}_0$ is isomorphic to $\mathrm{End}(E_0)$ as described in Section 2.3.
- For integers ℓ, m, n, we denote by $(m, n)_\ell$ a set of integers x such that $m < x < n$ and $\gcd(x, \ell) = 1$.
- For an isogeny $\varphi : E \to F$ and an integer n, we denote by $\varphi|_n$ the restriction of φ to $E[n]$.

In addition, we let $(P_0^1, Q_0^1) \coloneqq [3^{e_2}](P_0, Q_0)$, which is a basis of $E_0[2^{e_1}]$ and $(P_0^2, Q_0^2) \coloneqq [2^{e_1}](P_0, Q_0)$, which is a basis of $E_0[3^{e_2}]$.

3.1 Overview of `PushRandIsog`

Here, we give an overview of our new algorithm `PushRandIsog`. The input and output of `PushRandIsog` are as follows:

- Input: An integer $d \in (0, 2^{e_1})_6$ and a 3^{2e_2}-isogeny $\psi : E_0 \to E$.
- Output: E' and $\varphi|_{2^{e_1}}$ for a d-isogeny $\varphi : E \to E'$.

From now on, we decompose ψ into two 3^{e_2}-isogenies $\psi_0 : E_0 \to E_m$ and $\psi_m : E_m \to E$. Then, we can assume that ψ is given by generators of $\ker \psi_0$ and $\ker \psi_m$ since $E_0[3^{e_2}]$ is $\mathbb{F}_{p^2}$-rational. We denote these generators by K_0 and K_m, respectively. In addition, we let $P_m^1 \coloneqq \psi_0(P_0^1)$, $Q_m^1 \coloneqq \psi_0(Q_0^1)$, $P^1 \coloneqq \psi_m(P_m^1)$, and $Q^1 \coloneqq \psi_m(Q_m^1)$. Our algorithm can be broadly divided into the following three steps.

(i) Compute a $d(2^{e_1'} - d)$-isogeny $\rho_0 \circ \varphi_0 : E_0 \to E_0''$ for an integer $e_1' \leq e_1$ such that $3 \nmid 2^{e_1'} - d > 0$.
(ii) Compute a push-forward $\rho_m \circ \varphi_m$ of $\rho_0 \circ \varphi_0$ by ψ_0.
(iii) Compute a push-forward φ of φ_m by ψ_m.

The following is the commutative diagram of isogenies that our algorithm is based on. Note that the symbol in each parentheses represents the degree of each isogeny.

$$\begin{tikzcd}
E_0 \arrow[r, "\varphi_0"] \arrow[r, "(d)"'] \arrow[d, "\psi_0"'] & E'_0 \arrow[r, "\rho_0"] \arrow[r, "(2^{e'_1}-d)"'] & E''_0 \arrow[d, "\psi''_0"] \\
E_m \arrow[r, "\varphi_m"] \arrow[r, "(d)"'] \arrow[d, "\psi_m"'] & E'_m \arrow[r, "\rho_m"] \arrow[r, "(2^{e'_1}-d)"'] \arrow[d, "\psi'_m"] & E''_m \\
E \arrow[r, "\varphi"] \arrow[r, "(d)"'] & E' &
\end{tikzcd}$$

The integer $e'_1 \le e_1$ such that $3 \nmid 2^{e'_1} - d > 0$ might not exist in general (e.g. when $d = 2^{e_1} - 3$), but we here assume that there exists such an integer e'_1. Note that in our protocol setting, we can always find such e'_1. We will discuss it in Remark 1 of Section 4.

3.2 Details of PushRandIsog

Now we show the details of `PushRandIsog`. First, the computations related to the upper half of the commutative diagram (namely step (i) and (ii)) are performed in the same manner as the key generation and encryption in POKÉ [4, Algorithm 3, Algorithm 4]. We show the concrete algorithm for step (i) and (ii) in Algorithm 1.

We show that $(R''_m, S''_m) = (\rho_m \circ \varphi_m([2^\delta]P^1_m), \rho_m \circ \varphi_m([2^\delta]Q^1_m))$. In Step 6, we obtain integers u, v such that $\varphi_0(K_0) = [u]\hat{\rho}_0(P''_0) + [v]\hat{\rho}_0(Q''_0) = \hat{\rho}_0([u]P''_0 + [v]Q''_0)$ hold. Then, we obtain $\rho_0 \circ \varphi_0(K_0) = [\deg \rho_0]([u]P''_0 + [v]Q''_0) = K''_0$, which is a generator of ψ''_0. From the commutativity of the upper half of the diagram, we have the following equation:

$$\begin{aligned}
R''_m = \psi''_0(R''_0) &= \psi''_0 \circ \rho_0 \circ \varphi_0([2^\delta]P^1_0) \\
&= \rho_m \circ \varphi_m \circ \psi_0([2^\delta]P^1_0) \\
&= \rho_m \circ \varphi_m([2^\delta]P^1_m).
\end{aligned}$$

Similarly, $S''_m = \rho_m \circ \varphi_m([2^\delta]Q^1_m)$ also holds. This means that we obtain $\rho_m \circ \varphi_m|_{2^{e'_1}}$ since $([2^\delta]P^1_m, [2^\delta]Q^1_m)$ is a basis of $E_m[2^{e'_1}]$.

Unlike POKÉ, we additionally need to compute the push-forward φ of the isogeny φ_m by ψ_m (step (iii)). We can compute the push-forward φ as follows:

1. Compute $R'_m = \varphi_m(P^1_m)$, $S'_m = \varphi_m(Q^1_m)$, and $K'_m = \varphi_m(K_m)$ from $\rho_m \circ \varphi_m|_{2^{e'_1}}$ via `KaniEval`.
2. Compute a 3^{e_2}-isogeny $\psi'_m : E'_m \to E'$ with kernel $\langle K'_m \rangle$.
3. Let $R' \leftarrow \psi'_m(R'_m)$ and $S' \leftarrow \psi'_m(S'_m)$.

From the commutativity of the lower half of the diagram, we have $R' = \psi'_m \circ \varphi_m(P^1_m) = \varphi \circ \psi_m(P^1_m) = \varphi(P^1)$ and similarly, $S' = \varphi(Q^1)$. We show the detailed algorithm of `PushRandIsog` in Algorithm 2.

Algorithm 1 Compute a $d(2^{e_1'} - d)$-isogeny $\rho_m \circ \varphi_m$

```
Require: An integer $d \in (0, 2^{e_1})_6$, a generator $K_0$ of $\psi_0$, an integer $e_1' \leq e_1$ such that
    $3 \nmid 2^{e_1'} - d > 0$, and $\delta := e_1 - e_1'$.
Ensure: $E_m''$ and $(\rho_m \circ \varphi_m([2^\delta]P_m^1), \rho_m \circ \varphi_m([2^\delta]Q_m^1))$ for a $d(2^{e_1'} - d)$-isogeny $\rho_m \circ \varphi_m$ :
    $E_m \to E_m''$.
Let $\alpha \leftarrow \texttt{FullRepresentInteger}_{\mathcal{O}_0}(d(2^{e_1'} - d)2^\delta 3^{e_2})$.
    // Write $\alpha = \widehat{\tau}_0 \circ \rho_0 \circ \varphi_0$, where $\deg \varphi_0 = d, \deg \rho_0 = 2^{e_1'} - d, \deg \tau_0 = 2^\delta 3^{e_2}$.
Let $\tau_0 : E_0 \to E_0''$ be a $2^\delta 3^{e_2}$-isogeny with kernel $E_0[2^\delta 3^{e_2}] \cap \ker(\hat{\alpha})$.
Let $R_0'', S_0'' \leftarrow \tau_0 \circ \alpha(P_0), \tau_0 \circ \alpha(Q_0)$.
    // $R_0'' = \rho_0 \circ \varphi_0([2^\delta P_0^1])$, $S_0'' = \rho_0 \circ \varphi_0([2^\delta Q_0^1])$.
Let $P_0'', Q_0''$ be a basis of $E_0''[3^{e_2}]$.
$(E_0'; K_0'; P_0', Q_0') \leftarrow \texttt{KaniEval}(d, 2^{e_1'} - d, [2^\delta]P_0^1, [2^\delta]Q_0^1, R_0'', S_0''; K_0; P_0'', Q_0'')$.
    // $K_0' = \varphi_0(K_0)$, $P_0' = \hat{\rho}_0(P_0'')$, $Q_0' = \hat{\rho}_0(Q_0'')$.
Find two integers $u, v \in \mathbb{Z}/3^{e_2}\mathbb{Z}$ such that $K_0' = [u]P_0' + [v]Q_0'$. //Solve BiDLP.
Let $K_0'' \leftarrow [2^{e_1'} - d]([u]P_0'' + [v]Q_0'')$.
Compute a $3^{e_2}$-isogeny $\psi_0'' : E_0'' \to E_m''$ with kernel $\langle K_0'' \rangle$ and let $R_m'' \leftarrow \psi_0''(R_0'')$,
    $S_m'' \leftarrow \psi_0''(S_0'')$.
return $E_m'', (R_m'', S_m'')$.
```

4 New Signature Scheme: SQIsign2DPush

In this section, we describe our new signature scheme SQIsign2DPush.

4.1 SQIsign2DPush Σ-Protocol

We first describe the Σ-protocol underlying SQIsign2DPush. The Σ-protocol is based on the following diagram.

$$\begin{tikzcd}
E_0 \arrow[r, shift left, "\varphi_{\mathrm{sk1}}"] \arrow[r, shift right, "\varphi_{\mathrm{sk2}}"'] \arrow[d, "\varphi_{\mathrm{com}}"'] & E_{\mathrm{pk}} \arrow[dr, "\varphi_{\mathrm{chl}}"] & \\
E_{\mathrm{com}} \arrow[r, "\varphi_{\mathrm{rsp}}^{(1)}"'] \arrow[rr, bend left, "\varphi_{\mathrm{rsp}}"] \arrow[d, "\varphi_{\mathrm{aux}}^{(1)}"] \arrow[dd, bend right, "\varphi_{\mathrm{aux}}"'] & E_{\mathrm{chl}}' \arrow[r, "\varphi_{\mathrm{rsp}}^{(0)}"'] & E_{\mathrm{chl}} \\
E_{\mathrm{aux}}' \arrow[d, "\varphi_{\mathrm{aux}}^{(0)}"] & & \\
E_{\mathrm{aux}} & &
\end{tikzcd}$$

Parameter setting. We use the same notation as in Section 3, assuming that $2^{e_1} \approx 3^{e_2} \approx 2^\lambda$, and $2^{e_1} \geq 2\sqrt{2p}/\pi$. Then, the public parameters of SQIsign2DPush are taken as param $= (p, e_1, e_2, E_0, P_0, Q_0, \mathcal{O}_0)$. Note that these parameters are the same as SQIsignHD.

```
**Algorithm 2** $\texttt{PushRandIsog}_{\mathcal{O}_0}(d, \psi)$
**Require:** An integer $d \in (0, 2^{e_1})_6$ and a $3^{2e_2}$-isogeny $\psi : E_0 \to E$.
**Ensure:** $(E'; \varphi(P^1), \varphi(Q^1))$ for a $d$-isogeny $\varphi : E \to E'$ and two points $(P^1, Q^1) = (\psi(P_0^1), \psi(Q_0^1))$.
  Obtain an elliptic curve $E_m$ and points $K_0 \in E_0, K_m \in E_m$ such that $\psi$ is the composition of two $3^{e_2}$-isogenies $\psi_0 : E_0 \to E_m$ with kernel $\langle K_0 \rangle$ and $\psi_m : E_m \to E$ with kernel $\langle K_m \rangle$.
  Let $P_m^1, Q_m^1 \leftarrow \psi_0(P_0^1), \psi_0(Q_0^1)$.
  Find the smallest integer $e_1' \le e_1$ such that $3 \nmid 2^{e_1'} - d > 0$ and let $\delta \leftarrow e_1 - e_1'$.
  Compute $E_m''$ and $(R_m'', S_m'') = (\rho_0 \circ \varphi_0([2^\delta]P_m^1), \rho_0 \circ \varphi_0([2^\delta]Q_m^1))$ for a $d(2^{e_1'} - d)$-isogeny $\rho_m \circ \varphi_m : E_m \to E_m''$ via Algortihm 1.
  $(E_m'; R_m', S_m', K_m'; \emptyset)$
    $\leftarrow \texttt{KaniEval}(d, 2^{e_1'} - d, [2^\delta]P_m^1, [2^\delta]Q_m^1, R_m'', S_m''; P_m^1, Q_m^1, K_m; \emptyset)$.
    $//R_m' = \varphi_m(P_m^1)$, $S_m' = \varphi_m(Q_m^1)$, $K_m' = \varphi_m(K_m)$.
  Compute the $3^{e_2}$-isogeny $\psi_m' : E_m' \to E'$ with kernel $\langle K_m' \rangle$ and let
    $R' \leftarrow \psi_m'(R_m'), S' \leftarrow \psi_m'(S_m')$.
  **return** $(E'; R', S')$.
```

Key generation, commitment, and challenge. The algorithms for key generation, commitment, and challenge are almost the same as SQIsignHD.

- **gen**$(1^\lambda) \to (\mathrm{pk}, \mathrm{sk})$: We compute a 2^{2e_1}-isogeny $\varphi_{\mathrm{sk1}} : E_0 \to E_{\mathrm{pk}}$, a 3^{2e_2}-isogeny $\varphi_{\mathrm{sk2}} : E_0 \to E_{\mathrm{pk}}$, and their corresponding ideals I_{sk1}, and I_{sk2} using $\texttt{FastDoublePath}_{2^{e_1}, 3^{e_2}}$. Return $\mathrm{pk} = E_{\mathrm{pk}}$ and $\mathrm{sk} = (\varphi_{\mathrm{sk1}}, \varphi_{\mathrm{sk2}}, I_{\mathrm{sk1}}, I_{\mathrm{sk2}})$.
- **com**$(\mathrm{pk}, \mathrm{sk}) \to (\mathrm{com}, \mathrm{st})$: We compute a 3^{2e_2}-isogeny $\varphi_{\mathrm{com}} : E_0 \to E_{\mathrm{com}}$ and its corresponding ideal I_{com} via $\texttt{FastCommit}_{2^{e_1}, 3^{e_2}}$. Let ψ_0 and ψ_m be the isogenies such that $\varphi_{\mathrm{com}} = \psi_m \circ \psi_0$ and $\deg \varphi_0 = \deg \varphi_m = 3^{e_2}$. Since we use $\texttt{FastCommit}$, we have generators K_0 of $\ker \psi_0$ and K_m of $\ker \psi_m$ as an efficient representation of φ_{com}. Return $\mathrm{com} = E_{\mathrm{com}}$ and a state $\mathrm{st} = (\varphi_{\mathrm{com}}, I_{\mathrm{com}})$.
- **chl** $\to$ chl: Return an integer $\mathrm{chl} \in_U [0, 3^{e_2})$. This integer describes the kernel of the challenge isogeny $\varphi_{\mathrm{chl}} : E_{\mathrm{pk}} \to E_{\mathrm{chl}}$ such that $\ker(\varphi_{\mathrm{chl}}) = \langle P_{\mathrm{pk}} + [\mathrm{chl}]Q_{\mathrm{pk}} \rangle$, where $P_{\mathrm{pk}}, Q_{\mathrm{pk}}$ is the deterministic basis of $E_{\mathrm{pk}}[3^{e_2}]$.

Response. As in SQIsign-v2.0, we compute a response isogeny $\varphi_{\mathrm{rsp}} : E_{\mathrm{com}} \to E_{\mathrm{chl}}$ and an auxiliary isogeny $\varphi_{\mathrm{aux}} : E_{\mathrm{com}} \to E_{\mathrm{aux}}$.

- **rsp**$(\mathrm{pk}, \mathrm{sk}, \mathrm{com}, \mathrm{st}, \mathrm{chl}) \to \mathrm{rsp}/\perp$:
  1. Compute the ideal I_{chl} corresponding to φ_{chl} via the isomorphism induced by φ_{sk1} using $\texttt{IsogToIdeal}(\varphi_{\mathrm{chl}}, \varphi_{\mathrm{sk1}}, I_{\mathrm{sk1}})$.
  2. Let $\gamma \in \mathcal{O}_0$ be the quaternion corresponding to $\widehat{\varphi}_{\mathrm{sk2}} \circ \varphi_{\mathrm{sk1}}$ and let

$$I = \bar{I}_{\mathrm{com}} \cdot I_{\mathrm{sk1}} \cdot I_{\mathrm{chl}} \cdot \frac{\bar{\gamma}}{2^{2e_1}},$$

  which corresponds to $\varphi_{\mathrm{chl}} \circ \varphi_{\mathrm{sk2}} \circ \widehat{\varphi}_{\mathrm{com}}$ via the isomorphism induced by φ_{com}.

3. Sample an ideal I_{rsp} equivalent to I of norm $\in (0, 2^{e_1})_3$, i.e., let $I_{\mathrm{rsp}} = I \cdot \frac{\bar{\alpha}}{n(I)}$ for $\alpha \in I$ such that $q_I(\alpha) \in (0, 2^{e_1})_3$. If there is no such ideal, return $\perp$, which means a failure of response.
4. Since $\alpha \in I \subseteq \mathcal{O}_0$, we can consider the endomorphism of E_0 corresponding to α. We denote it by α_{E_0}. Let φ_{rsp} be the isogeny corresponding to I_{rsp} via the isomorphism induced by φ_{com}. Then, we can compute $\varphi_{\mathrm{rsp}}|_{2^{e_1}}$ using the equation:
$$\varphi_{\mathrm{rsp}} = [3^{-5e_2}] \circ \varphi_{\mathrm{chl}} \circ \varphi_{\mathrm{sk2}} \circ \hat{\alpha}_{E_0} \circ \hat{\varphi}_{\mathrm{com}}$$
since $\alpha_{E_0} = \widehat{\varphi}_{\mathrm{com}} \circ \widehat{\varphi}_{\mathrm{rsp}} \circ \varphi_{\mathrm{chl}} \circ \varphi_{\mathrm{sk2}}$.
5. Let d_{rsp} be the norm of I_{rsp} and write $d_{\mathrm{rsp}} = d'_{\mathrm{rsp}} \cdot 2^{n_1}$ for an odd integer d'_{rsp}. We decompose φ_{rsp} into $\varphi^{(0)}_{\mathrm{rsp}} \circ \varphi^{(1)}_{\mathrm{rsp}}$ such that $\deg(\varphi^{(1)}_{\mathrm{rsp}}) = d'_{\mathrm{rsp}}$ and $\deg(\varphi^{(0)}_{\mathrm{rsp}}) = 2^{n_1}$. We denote the codomain of $\varphi^{(1)}_{\mathrm{rsp}}$ by E'_{chl}.
6. Let $d_{\mathrm{aux}} = 2^{e_1 - n_1} - d'_{\mathrm{rsp}}$ and write $d_{\mathrm{aux}} = d'_{\mathrm{aux}} \cdot 3^{n_2}$ for an integer d'_{aux} coprime to 3.
7. Since $d'_{\mathrm{aux}} \in (0, 2^{e_1})_6$, we can compute $\varphi^{(1)}_{\mathrm{aux}}|_{2^{e_1}}$ for a d'_{aux}-isogeny $\varphi^{(1)}_{\mathrm{aux}} : E_{\mathrm{com}} \to E'_{\mathrm{aux}}$ via `PushRandIsog`.
8. If n_2 is odd, compute a random 3-isogeny φ_3 from E'_{aux} and let $\varphi^{(0)}_{\mathrm{aux}} = [3^{(n_2-1)/2}] \circ \varphi_3$. Otherwise, let $\varphi^{(0)}_{\mathrm{aux}} = [3^{n_2/2}]$. Let E_{aux} be the codomain of $\varphi^{(0)}_{\mathrm{aux}}$.
9. Let $\varphi_{\mathrm{aux}} = \varphi^{(0)}_{\mathrm{aux}} \circ \varphi^{(1)}_{\mathrm{aux}}$, whose degree is d_{aux}.
10. Take the deterministic basis $P_{\mathrm{aux}}, Q_{\mathrm{aux}}$ of $E_{\mathrm{aux}}[2^{e_1}]$.
11. Let $(P_{\mathrm{chl}}, Q_{\mathrm{chl}}) \leftarrow [d^{-1}_{\mathrm{aux}}](\varphi_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}(P_{\mathrm{aux}}), \varphi_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}(Q_{\mathrm{aux}}))$.
12. Return $\mathrm{rsp} = (E_{\mathrm{aux}}, P_{\mathrm{chl}}, Q_{\mathrm{chl}}, n_1)$.

The response algorithm is shown in Algorithm 3. In Step 13, we solve BiDLP on E_{aux} twice to find the matrix M. Since $(R_{\mathrm{aux}}, S_{\mathrm{aux}}) = (\varphi_{\mathrm{aux}}(P^1_{\mathrm{com}}), \varphi_{\mathrm{aux}}(Q^1_{\mathrm{com}}))$, we have

$$(P_{\mathrm{aux}}, Q_{\mathrm{aux}}) = (\varphi_{\mathrm{aux}}(P^1_{\mathrm{com}}), \varphi_{\mathrm{aux}}(Q^1_{\mathrm{com}}))M.$$

By taking $\varphi_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}$ on both side, we have

$$\begin{aligned}(\varphi_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}(P_{\mathrm{aux}}), \varphi_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}(Q_{\mathrm{aux}})) &= [d_{\mathrm{aux}}](\varphi_{\mathrm{rsp}}(P^1_{\mathrm{com}}), \varphi_{\mathrm{rsp}}(Q^1_{\mathrm{com}}))M \\ &= [d_{\mathrm{aux}}](R_{\mathrm{chl}}, S_{\mathrm{chl}})M.\end{aligned}$$

Therefore, in Step 14, we obtain

$$(P_{\mathrm{chl}}, Q_{\mathrm{chl}}) = [d^{-1}_{\mathrm{aux}}](\varphi_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}(P_{\mathrm{aux}}), \varphi_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}(Q_{\mathrm{aux}})).$$

Remark 1. In `PushRandIsog` in Step 10 of Algorithm 3, we have to find an integer $e'_1 \leq e_1$ such that $3 \nmid 2^{e'_1} - d'_{\mathrm{aux}} > 0$ (see Algorithm 2). We show that we can always find such an integer. (i) If $n_2 = 0$: $d'_{\mathrm{aux}} = 2^{e_1 - n_1} - d'_{\mathrm{rsp}}$ holds and we have $2^{e_1 - n_1} - d'_{\mathrm{aux}} = d'_{\mathrm{rsp}} > 0$. Since $d_{\mathrm{rsp}} = d'_{\mathrm{rsp}} \cdot 2^{n_1}$ is coprime to 3, d'_{rsp} is also coprime to 3. Therefore, we can choose $e'_1 = e_1 - n_1$. (ii) If $n_2 \geq 1$: we have $d'_{\mathrm{aux}} = (2^{e_1 - n_1} - d'_{\mathrm{rsp}})/3^{n_2} < (2^{e_1 - n_1} - d'_{\mathrm{rsp}})/2 < 2^{e_1 - n_1 - 1}$. Hence, $2^{e_1 - n_1} - d'_{\mathrm{aux}} > 2^{e_1 - n_1 - 1} - d'_{\mathrm{aux}} > 0$ holds. Since $2^{e_1 - n_1} - d'_{\mathrm{aux}} \not\equiv 2^{e_1 - n_1 - 1} - d'_{\mathrm{aux}} \bmod 3$, at least one of them is coprime to 3. Therefore, we can choose $e'_1 = e_1 - n_1$ or $e'_1 = e_1 - n_1 - 1$.

Algorithm 3 $\mathbf{rsp}(\mathrm{pk}, \mathrm{sk}, \mathrm{com}, \mathrm{st}, \mathrm{chl}) \to \mathrm{rsp}/\perp$

```
Require: A publick key pk, a secret key sk, a commitment com, a state st, and a challenge chl.
Ensure: Response rsp or $\perp$.
Let $I_{\mathrm{chl}} \leftarrow \texttt{IsogToIdeal}(\varphi_{\mathrm{chl}}, \varphi_{\mathrm{sk1}}, I_{\mathrm{sk1}})$.
Let $\gamma \in \mathcal{O}_0$ be the quaternion corresponding to $\widehat{\varphi}_{\mathrm{sk2}} \circ \varphi_{\mathrm{sk1}}$.
Let $I \leftarrow \bar{I}_{\mathrm{com}} \cdot I_{\mathrm{sk1}} \cdot I_{\mathrm{chl}} \cdot \frac{\bar{\gamma}}{2^{2e_1}}$.
Find all $\alpha \in I$ such that $d_{\mathrm{rsp}} \coloneqq q_I(\alpha) \in (0, 2^{e_1})_3$ by lattice enumeration and choose one of them uniformly. If there is no such $\alpha$, return $\perp$.
Let $I_{\mathrm{rsp}} = I\frac{\bar{\alpha}}{n(I)}$ and let $\varphi_{\mathrm{rsp}} : E_{\mathrm{com}} \to E_{\mathrm{chl}}$ be its corresponding isogeny.
Let $R_{\mathrm{chl}} \leftarrow [3^{-5e_2}]\varphi_{\mathrm{chl}} \circ \varphi_{\mathrm{sk2}} \circ \hat{\alpha}_{E_0} \circ \hat{\varphi}_{\mathrm{com}}(P^1_{\mathrm{com}})$. $//R_{\mathrm{chl}} = \varphi_{\mathrm{rsp}}(P^1_{\mathrm{com}})$.
Let $S_{\mathrm{chl}} \leftarrow [3^{-5e_2}]\varphi_{\mathrm{chl}} \circ \varphi_{\mathrm{sk2}} \circ \hat{\alpha}_{E_0} \circ \hat{\varphi}_{\mathrm{com}}(Q^1_{\mathrm{com}})$. $//S_{\mathrm{chl}} = \varphi_{\mathrm{rsp}}(Q^1_{\mathrm{com}})$.
Let $d_{\mathrm{rsp}} = d'_{\mathrm{rsp}} \cdot 2^{n_1}$, where $2 \nmid d'_{\mathrm{rsp}}$ and let $d_{\mathrm{aux}} = 2^{e_1 - n_1} - d'_{\mathrm{rsp}}$.
Let $d_{\mathrm{aux}} = d'_{\mathrm{aux}} \cdot 3^{n_2}$, where $3 \nmid d'_{\mathrm{aux}}$.
Let $(E'_{\mathrm{aux}}, R'_{\mathrm{aux}}, S'_{\mathrm{aux}}) \leftarrow \texttt{PushRandIsog}_{\mathcal{O}_0}(d'_{\mathrm{aux}}, \varphi_{\mathrm{com}})$.
    $//\ R'_{\mathrm{aux}} = \varphi^{(1)}_{\mathrm{aux}}(P^1_{\mathrm{com}})$, $S'_{\mathrm{aux}} = \varphi^{(1)}_{\mathrm{aux}}(Q^1_{\mathrm{com}})$.
If $n_2$ is odd, compute a random 3-isogeny $\varphi_3 : E'_{\mathrm{aux}} \to E_{\mathrm{aux}}$ and let $\varphi^{(0)}_{\mathrm{aux}} = [3^{(n_2-1)/2}] \circ \varphi_3$. Otherwise, let $\varphi^{(0)}_{\mathrm{aux}} = [3^{n_2/2}]$.
Let $R_{\mathrm{aux}}, S_{\mathrm{aux}} \leftarrow \varphi^{(0)}_{\mathrm{aux}}(R'_{\mathrm{aux}}), \varphi^{(0)}_{\mathrm{aux}}(S'_{\mathrm{aux}})$.
Let $P_{\mathrm{aux}}, Q_{\mathrm{aux}}$ be the deterministic basis of $E_{\mathrm{aux}}[2^{e_1}]$ and find a matrix $M \in (\mathbb{Z}/2^{e_1}\mathbb{Z})^{2\times 2}$ such that $(P_{\mathrm{aux}}, Q_{\mathrm{aux}}) = (R_{\mathrm{aux}}, S_{\mathrm{aux}})M$.
Let $(P_{\mathrm{chl}}, Q_{\mathrm{chl}}) \leftarrow (R_{\mathrm{chl}}, S_{\mathrm{chl}})M$.
return $(E_{\mathrm{aux}}, P_{\mathrm{chl}}, Q_{\mathrm{chl}}, n_1)$.
```

Verify. The verification is also similar to SQIsign-v2.0. But we have to check that $\ker(\widehat{\varphi}_{\mathrm{rsp}} \circ \varphi_{\mathrm{chl}})$ does not contain $E_{\mathrm{pk}}[3]$. This is to achieve the special soundness (see the proof of Theorem 2 for details).

- **verf**(pk, com, chl, rsp) →accept/reject: We first compute the isogeny $\varphi_{\mathrm{chl}} : E_{\mathrm{pk}} \to E_{\mathrm{chl}}$ with kernel $\langle P_{\mathrm{pk}} + [\mathrm{chl}]Q_{\mathrm{pk}} \rangle$ and let K_{bt} be a generator of $\ker \widehat{\varphi}_{\mathrm{chl}} \cap E_{\mathrm{chl}}[3]$, which is the kernel of the first step of the dual $\widehat{\varphi}_{\mathrm{chl}}$. (See Remark 2 for how to compute such K_{bt}.) Next, we compute the isogeny $\varphi : E_{\mathrm{chl}} \to E'_{\mathrm{chl}}$ with kernel $[2^{e_1 - n_1}]\langle P_{\mathrm{chl}}, Q_{\mathrm{chl}} \rangle$, which corresponds to the dual of $\varphi^{(0)}_{\mathrm{rsp}}$ and let $P'_{\mathrm{chl}} = \varphi(P_{\mathrm{chl}}), Q'_{\mathrm{chl}} = \varphi(Q_{\mathrm{chl}})$, and $K'_{\mathrm{bt}} = \varphi(K_{\mathrm{bt}})$. Then we compute the isogeny Φ with kernel

$$\begin{aligned}&\langle([2^{n_1}]P_{\mathrm{aux}}, P'_{\mathrm{chl}}), ([2^{n_1}]Q_{\mathrm{aux}}, Q'_{\mathrm{chl}})\rangle \\ &= [2^{n_1}]\langle([d_{\mathrm{aux}}]P_{\mathrm{aux}}, \varphi^{(1)}_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}(P_{\mathrm{aux}})), ([d_{\mathrm{aux}}]Q_{\mathrm{aux}}, \varphi^{(1)}_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}(Q_{\mathrm{aux}}))\rangle.\end{aligned}$$

  From Theorem 1, Φ should be a $(2^{e_1 - n_1}, 2^{e_1 - n_1})$-isogeny from $E_{\mathrm{aux}} \times E'_{\mathrm{chl}}$ to $E_{\mathrm{com}} \times F$ for a curve F. Otherwise, we return reject. Finally, we check whether the kernel of $\widehat{\varphi}_{\mathrm{chl}} \circ \varphi_{\mathrm{rsp}}$ contains $E_{\mathrm{com}}[3]$: compute $K_{\mathrm{com}} = \widehat{\varphi}^{(1)}_{\mathrm{rsp}}(K'_{\mathrm{bt}}) = \widehat{\varphi}_{\mathrm{rsp}}(K_{\mathrm{bt}})$ and check if $K_{\mathrm{com}} = O$. If $K_{\mathrm{com}} = O$, return reject. Otherwise, return accept.

We show the verification algorithm in Algorithm 4.

Algorithm 4 **verf**(pk, com, chl, rsp) → accept/reject

```
Require: A public key pk, a commitment com, a challenge chl, and a response rsp.
Ensure: accept or reject.
 1: Let $P_{\rm pk}, Q_{\rm pk}$ be the deterministic basis of $E_{\rm pk}[3^{e_2}]$.
 2: Compute a $3^{e_2}$-isogeny $\varphi_{\rm chl} : E_{\rm pk} \to E_{\rm chl}$ with kernel $\langle P_{\rm pk} + [{\rm chl}]Q_{\rm pk} \rangle$.
 3: Let $K_{\rm bt}$ be a generator of $\ker \widehat{\varphi}_{\rm chl} \cap E_{\rm chl}[3]$.
 4: Compute a $2^{n_1}$-isogeny $\varphi : E_{\rm chl} \to E'_{\rm chl}$ with kernel $[2^{e_1-n_1}]\langle P_{\rm chl}, Q_{\rm chl} \rangle$ and let
    $P'_{\rm chl} \leftarrow \varphi(P_{\rm chl}), Q'_{\rm chl} \leftarrow \varphi(Q_{\rm chl}), K'_{\rm bt} \leftarrow \varphi(K_{\rm bt})$.
 5: Let $P_{\rm aux}, Q_{\rm aux}$ be the deterministic basis of $E_{\rm aux}[2^{e_1}]$.
 6: Compute a $(2^{e_1-n_1}, 2^{e_1-n_1})$-isogeny $\Phi : E_{\rm aux} \times E'_{\rm chl} \to A$ with kernel
    $\langle ([2^{n_1}]P_{\rm aux}, P'_{\rm chl}), ([2^{n_1}]Q_{\rm aux}, Q'_{\rm chl}) \rangle$.
 7: if not $A \cong E_{\rm com} \times F$ for a curve $F$ then
 8:     return reject.
 9: end if
10: Let $\eta : A \to E_{\rm com} \times F$ be the isomorphism and let $(K_{\rm com}, -) \leftarrow \eta \circ \Phi((O, K'_{\rm bt}))$.
11: if $K_{\rm com} \neq O$ then
12:     return reject.
13: end if
14: return accept.
```

Remark 2. We explain how to compute $K_{\rm bt}$ in the verification. Let $\varphi : E_1 \to E_2$ be a 3-isogeny. Then, we can compute a generator $K_{\rm bt}$ of $\ker \widehat{\varphi}$ from the Montgomery coefficients of E_1 and E_2. Let A_1 and A_2 be the Montgomery coefficients of E_1 and E_2, respectively. From the formulas in [9, Theorem 1], the x-coordinate $x_{\rm bt}$ of $K_{\rm bt}$ satisfies $6x_{\rm bt} + A_2 x_{\rm bt}^2 - 6x_{\rm bt}^3 = A_1$. In addition, from the 3-division polynomial of Montgomery curves, we have $3x_{\rm bt}^4 + 4A_2 x_{\rm bt}^3 + 6x_{\rm bt}^2 - 1 = 0$. From these two equations, we obtain

$$x_{\rm bt} = \frac{A_2^3 + 18A_2^2A_1 - 3A_2A_1^2 + 48A_2 - 112A_1}{4A_2^3A_1 + 114A_2^2 + 12A_2A_1 + 2A_1^2 - 576}.$$

Correctness Assume that the response is generated honestly. As we mentioned in the description of **verf**, the map Φ computed in Step 6 should be a $(2^{e_1-n_1}, 2^{e_1-n_1})$-isogeny from $E_{\rm aux} \times E'_{\rm chl}$ to $E_{\rm com} \times F$ for a curve F. In addition, $\widehat{\varphi}_{\rm chl} \circ \varphi_{\rm rsp}$ does not contain $E_{\rm com}[3]$ since $\varphi_{\rm chl}$ is a cyclic 3^{e_2}-isogeny and $\deg \varphi_{\rm rsp}$ is coprime to 3. Therefore, **verf** always accepts the honest response.

4.2 Fiat-Shamir with Bounded Aborts

To obtain a signature scheme from our Σ-protocol, we apply the Fiat-Shamir transform. However, our Σ-protocol may fail to generate a response, i.e. **rsp** may output $\perp$ when there is no ideal $I_{\rm rsp}$ of desired norm. Therefore, we apply the Fiat-Shamir transform with bounded aborts (FSwBA) [25]. Let $H : \{0,1\}^* \to [0, 3^{e_2})$ be a cryptographic hash function and B be a positive integer. Then, FSwBA with the bound B is as follows.

Fiat-Shamir transform with bounded aborts

```
**Sign**(pk, sk, $m$):
$\kappa \leftarrow 1, \mathrm{rsp} \leftarrow \perp$.
**while** $\mathrm{rsp} = \perp$ and $\kappa \leq B$:
    $(\mathrm{com}, \mathrm{st}) \leftarrow \mathbf{com}(\mathrm{pk}, \mathrm{sk})$.
    $\mathrm{chl} \leftarrow H(\mathrm{pk}, m, \mathrm{com})$.
    $\mathrm{rsp} \leftarrow \mathbf{rsp}(\mathrm{pk}, \mathrm{sk}, \mathrm{com}, \mathrm{st}, \mathrm{chl})$.
    $\kappa \leftarrow \kappa + 1$.
**if** $\mathrm{rsp} = \perp$, **return** $\perp$.
**return** $\mathrm{sig} = (\mathrm{com}, \mathrm{rsp})$.
```

```
**Verf**(pk, $m$, sig):
Parse $\mathrm{sig} = (\mathrm{com}, \mathrm{rsp})$.
$\mathrm{chl} \leftarrow H(\mathrm{pk}, m, \mathrm{com})$.
**return** $\mathbf{verf}(\mathrm{pk}, \mathrm{com}, \mathrm{chl}, \mathrm{rsp})$.
```

By using a sufficiently large bound B, the probability of failure of **Sign** can be negligible. To determine an appropriate bound, it is necessary to evaluate the failure probability of **rsp**. We will evaluate the probability in Section 4.3.

4.3 Failure Probability of the Response

In this subsection, we evaluate the failure probability of the response. The failure occurs when there exists no ideal I_{rsp} equiavlent to I of norm $\in (0, 2^{e_1})_3$.

From [11, Lemma 12], if $2^{e_1} \geq 2\sqrt{2p}/\pi$, then there exists I_{rsp} such that $n(I_{\mathrm{rsp}}) < 2^{e_1}$. However, in our protocol, it is necessary that $3 \nmid n(I_{\mathrm{rsp}})$, and such an ideal I_{rsp} may not always exist. For example, the following case can be considered.

- Let $(\alpha_1, \ldots, \alpha_4)$ be a Minkowski reduced basis of I for the quadratic form q_I.
- $q_I(\alpha_1)$ and $q_I(\alpha_2)$ are divisible by 3 and smaller than 2^{e_1}.
- $q_I(\alpha_3)$ and $q_I(\alpha_4)$ are larger than 2^{e_1}.

Since $\frac{8p^2}{3\pi^4} \leq \prod_{i=1}^{4} q_I(\alpha_i) \leq \frac{64p^2}{\pi^4}$ holds from the Minkowski's second theorem (see the proof of [11, Lemma 12]), such an ideal can exist. In this case, any $\alpha \in I$ satisfying $q_I(\alpha) < 2^{e_1}$ is in $\mathbb{Z}\langle \alpha_1, \alpha_2 \rangle$, and therefore $q_I(\alpha)$ is divisible by 3. Thus, there exists no $\alpha \in I$ such that $q_I(\alpha) \in (0, 2^{e_1})_3$.

It has been experimentally confirmed that the probability of such an occurrence is small but not negligible. We show our experimental results in Table 2. We also show the bound B for the FSwBA transform that ensures the failure probability of **Sign** is at most $2^{-\lambda}$. Note that no failures were recorded in the experiment for Level 5. Therefore, the failure probability is considered to be extremely small, but a more detailed evaluation is left for future work. We also leave the theoretical evaluation of the failure probability as an open problem.

4.4 Signature Encoding and Sizes

The signature of SQIsign2DPush consists of $(E_{\mathrm{com}}, E_{\mathrm{aux}}, P_{\mathrm{chl}}, Q_{\mathrm{chl}}, n_1)$, but the signature size can be compressed similarly to SQIsign-v2.0.

Table 2. Experimental failure count in 4,000,000 trials.

NIST level	e_1	e_2	c	# of failure	failure rate	bound B
1 ($\lambda = 128$)	131	78	1	1	$\approx 2^{-22}$	6
3 ($\lambda = 192$)	191	117	1	13	$\approx 2^{-18}$	11
5 ($\lambda = 256$)	263	156	1	0	0	1

First, two points $P_{\mathrm{chl}}, Q_{\mathrm{chl}}$ can be represented by four elements of $\mathbb{Z}/2^{e1}\mathbb{Z}$ (that are the coefficients of $P_{\mathrm{chl}}, Q_{\mathrm{chl}}$ on the deterministic basis of $E_{\mathrm{chl}}[2^{e_1}]$). To compute the coefficients, the signer additionally needs to solve BiDLP on $E_{\mathrm{chl}}[2^{e_1}]$ twice.

Moreover, we can reduce the size of the signature by including the challenge chl $\in [0, 3^{e_2})$ instead of the commitment E_{com}. In that case, the verifier first computes the isogeny φ_{chl} from chl and obtains its codomain E_{chl}. The verifier then computes $\varphi : E_{\mathrm{chl}} \to E'_{\mathrm{chl}}$ and $\Phi : E_{\mathrm{aux}} \times E'_{\mathrm{chl}} \to A$ as in steps 1-6 of Algorithm 4. The verifier then checks if $A \cong F_1 \times F_2$ for curves F_1 and F_2. By computing two hash values $H(\mathrm{pk}, m, j(F_1))$ and $H(\mathrm{pk}, m, j(F_2))$ and comparing them with chl, the verifier can determine whether F_1 or F_2 is isomorphic to the commitment curve E_{com}. After obtaining the curve E_{com}, the verifier performs Steps 10-14 of Algorithm 4 in the same manner.

As in SQIsign-v2.0, we include two hints in the signature to allow the verifier to compute the deterministic basis of $E_{\mathrm{chl}}[2^{e_1}]$ and $E_{\mathrm{aux}}[2^{e_1}]$ faster. We use [1, Algorithm 2.1] to generate a hint and use [1, Algorithm 2.2] to compute the basis using the hint. The size of each hint is 1 byte. We also include a hint for the deterministic basis of $E_{\mathrm{pk}}[3^{e_3}]$ in the public key. For this, we use the method of [10, Section 3.3] and [3, Section 7.1]. The hint for $E_{\mathrm{pk}}[3^{e_3}]$ costs 2 bytes.

As a result, the signature of our protocol requires $\lceil \log_2 3^{e_2} \rceil$ bits for chl, $\lceil \log_2 p^2 \rceil$ bits for E_{aux}, $4\lceil \log_2 2^{e_1} \rceil$ bits for $P_{\mathrm{chl}}, Q_{\mathrm{chl}}$, $\lceil \log_2 e_1 \rceil$ bits for n_1, and 2 bytes for two hints. In total, the signature size is approximately $9\lambda + \log_2 \lambda + 16$ bits since $2^{e_1} \approx 3^{e_2} \approx 2^{\lambda}$.

5 Security Analysis

In this section, we analize the security of SQIsign2DPush. To ensure that our protocol is EUF-CMA secure, we prove that the underlying Σ-protocol is knowledge-sound and honest-verifier zero-knowledge. Throughout the security analysis, we denote the Σ-protocol by Σ_{Push}.

5.1 Special Soundness

We show that Σ_{Push} is special-sound with respect to the following relation:

$$\mathcal{R}_{\texttt{OneEnd}} = \{(E, \alpha) \mid E : \text{supersingular curve over } \mathbb{F}_{p^2}, \alpha \in \mathrm{End}(E) \backslash \mathbb{Z}\}.$$

Theorem 2 Σ_{Push} *satisfies the special-soundness with respect to the relation* $\mathcal{R}_{\texttt{OneEnd}}$.

Proof. The proof of the soundness of our protocol is similar to that of SQIsignHD [11, Proposition 17]. Let $(E_{\mathrm{com}}, \mathrm{chl}, \mathrm{rsp} = (E_{\mathrm{aux}}, P_{\mathrm{chl}}, Q_{\mathrm{chl}}, n_1))$ and $(E_{\mathrm{com}}, \widetilde{\mathrm{chl}}, \widetilde{\mathrm{rsp}} = (\widetilde{E}_{\mathrm{aux}}, \widetilde{P}_{\mathrm{chl}}, \widetilde{Q}_{\mathrm{chl}}, \widetilde{n}_1))$ be two Σ_{Push} transcripts against the statement E_{pk} with the same commitment E_{com} but different challenges $\mathrm{chl} \neq \widetilde{\mathrm{chl}}$. From chl and $\widetilde{\mathrm{chl}}$, we can compute $\varphi_{\mathrm{chl}} : E_{\mathrm{pk}} \to E_{\mathrm{chl}}$ and $\widetilde{\varphi}_{\mathrm{chl}} : E_{\mathrm{pk}} \to \widetilde{E}_{\mathrm{chl}}$. From rsp and $\widetilde{\mathrm{rsp}}$, we can compute efficient representations of $\varphi_{\mathrm{rsp}} : E_{\mathrm{com}} \to E_{\mathrm{chl}}$ and $\widetilde{\varphi}_{\mathrm{rsp}} : E_{\mathrm{com}} \to \widetilde{E}_{\mathrm{chl}}$.

Therefore, we obtain an efficient representation of $\beta = \widehat{\varphi}_{\mathrm{chl}} \circ \varphi_{\mathrm{rsp}} \circ \widehat{\widetilde{\varphi}}_{\mathrm{rsp}} \circ \widetilde{\varphi}_{\mathrm{chl}} \in \mathrm{End}(E_{\mathrm{pk}})$. We now show that β is non-scalar. If it were, we would have $\beta = [\nu]$ for an integer ν. Then we have $\nu^2 = \deg \beta = d\tilde{d} \cdot 3^{2e_2}$, where $d = \deg \varphi_{\mathrm{rsp}}$ and $\tilde{d} = \deg \widetilde{\varphi}_{\mathrm{rsp}}$. We write $d = 3^m \cdot d'$ and $\tilde{d} = 3^n \cdot \tilde{d}'$, where d' and $\tilde{d}'$ are coprime to 3. Without loss of generality, we can assume $m \geq n$. Since $\nu^2 = d\tilde{d} \cdot 3^{2e_2} = d'\tilde{d}' \cdot 3^{m+n+2e_2}$, we have $\nu = \nu' \cdot 3^{(m+n)/2+e_2}$ for $\nu' = \sqrt{d'\tilde{d}'} \in \mathbb{Z}$. Note that ν' is coprime to 3 since d' and $\tilde{d}'$ is coprime to 3. Now we have $[\nu']\widehat{\varphi}_{\mathrm{rsp}} \circ \varphi_{\mathrm{chl}} = [3^{(m-n)/2} \cdot d']\widehat{\widetilde{\varphi}}_{\mathrm{rsp}} \circ \widetilde{\varphi}_{\mathrm{chl}}$. Let $\widetilde{K}_{\mathrm{chl}}$ be a generator of $\ker \widetilde{\varphi}_{\mathrm{chl}}$. Then, $\widehat{\varphi}_{\mathrm{rsp}} \circ \varphi_{\mathrm{chl}}(\widetilde{K}_{\mathrm{chl}}) = [(\nu')^{-1} \cdot 3^{(m-n)/2} \cdot d']\widehat{\widetilde{\varphi}}_{\mathrm{rsp}} \circ \widetilde{\varphi}_{\mathrm{chl}}(\widetilde{K}_{\mathrm{chl}}) = O$ holds. Since $\ker(\widehat{\varphi}_{\mathrm{rsp}} \circ \varphi_{\mathrm{chl}})$ does not contain $E_{\mathrm{chl}}[3]$, we have $\varphi_{\mathrm{chl}}(\widetilde{K}_{\mathrm{chl}}) = O$. This means that $\ker \varphi_{\mathrm{chl}} = \ker \widetilde{\varphi}_{\mathrm{chl}}$. This contradicts $\mathrm{chl} \neq \widetilde{\mathrm{chl}}$. □

5.2 Honest-Verifier Zero-Knowledge

We now show that Σ_{Push} is weak honest-verifier zero-knowledge (wHVZK). As in SQIsign-v2.0, we use the *hint-assisted wHVZK* framwork [2, Definition 3.2], where the zero-knowledge simulator receives some additional information called *hints* sampled from a hint distribution [2, Definition 3.1].

Let $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$ be a relation. A *hint distribution* $\mathcal{H}$ for $\mathcal{R}$ is a collection of distributions $\{\mathcal{H}_x\}_{x \in \mathcal{X}}$ on a set of hints for x (see [2, Definition 3.1] for the formal definition). Then we define the hint-assisted wHVZK as follows.

Definition 1 ([2, Definition 3.2]) *Let $\Sigma = (\mathbf{com}, \mathbf{chl}, \mathbf{rsp}, \mathbf{verf})$ be a Σ-protocol and let $\mathcal{H}$ be a hint distribution. We say that Σ is $\mathcal{H}$-hint-assisted wHVZK if there exists a PPT algorithm, called the simulator $\mathcal{S}$, such that for all $q = \mathrm{poly}(\lambda)$, the following two distributions are computationally indistinguishable.*

- $\mathcal{D}_0 = (\mathrm{pk}, \pi_1, \ldots, \pi_q)$, *where* $(\mathrm{pk}, \mathrm{sk}) \leftarrow \mathbf{gen}(1^\lambda), (\mathrm{com}_i, \mathrm{st}_i) \leftarrow \mathbf{com}(\mathrm{pk}, \mathrm{sk}), \mathrm{chl}_i \leftarrow \mathbf{chl}, \mathrm{rsp}_i \leftarrow \mathbf{rsp}(\mathrm{pk}, \mathrm{sk}, \mathrm{com}_i, \mathrm{st}_i, \mathrm{chl}_i), \pi_i = (\mathrm{com}_i, \mathrm{chl}_i, \mathrm{rsp}_i)$ *for* $i \in \{1, \ldots, q\}$.
- $\mathcal{D}_1 = (\mathrm{pk}, \pi_1, \ldots, \pi_q)$, *where* $(\mathrm{pk}, \mathrm{sk}) \leftarrow \mathbf{gen}(1^\lambda), h_i \leftarrow \mathcal{H}_{\mathrm{pk}}, \pi_i \leftarrow \mathcal{S}(\mathrm{pk}, h_i)$ *for* $i \in \{1, \ldots, q\}$.

Next, we give some oracles to define the hint distribution in the following.

$\underline{D_{\mathtt{Chl}}(E) \to (\mathrm{chl}, \varphi)\text{:}}$
Input: A supersingular elliptic curve E over $\mathbb{F}_{p^2}$.
Output: An integer $\mathrm{chl} \in [0, 3^{e_2})$ and a 3^{e_2}-isogeny φ from E.

```
1. chl $\in_U [0, 3^{e_2})$.
2. Take the deterministic basis $P, Q$ of $E[3^{e_2}]$.
3. Compute the isogeny $\varphi : E \to E'$ with $\ker \varphi = \langle P + [\mathrm{chl}]Q \rangle$.
4. return $(\mathrm{chl}, \varphi)$.
```

$\underline{D_{\texttt{RspIsog}}(E, E') \to \varphi/\perp:}$
Input: Supersingular elliptic curves E, E' over $\mathbb{F}_{p^2}$.
Output: An isogeny $\varphi : E \to E'$ of degree $\in (0, 2^{e_1})_3$ or $\perp$.

```
1. Try to sample an isogeny $\varphi' : E \to E'$ uniformly among isogenies $E \to E'$
   of degree $\in (0, 2^{e_1})_3$.
2. If there is no such isogeny, return $\perp$.
3. Otherwise, write $\varphi' = [m] \circ \varphi$ with $m \in \mathbb{Z}$ and $\varphi$ is cyclic.
4. return $\varphi$.
```

$\underline{D_{\texttt{AuxIsog}}(E, \psi, d) \to \varphi:}$
Input: A supersingular elliptic curve E over $\mathbb{F}_{p^2}$, a 3^{2e_2}-isogeny $\psi : E_0 \to E$, and an integer $d \in (0, 2^{e_1})_2$.
Output: A d-isogeny φ from E.

```
1. Write $d = 3^n d'$, where $d'$ coprime to 3.
2. Compute a $d'$-isogeny $\varphi' : E \to E'$ via $\texttt{PushRandIsog}_{\mathcal{O}_0}(d', \psi)$.
3. If $n$ is odd, compute a 3-isogeny $\varphi'' : E' \to E''$ uniformly and let $\varphi'' \leftarrow$
   $[3^{(n-1)/2}] \circ \varphi''$. Otherwise, let $\varphi'' = [3^{n/2}]$.
4. Let $\varphi \leftarrow \varphi'' \circ \varphi'$
5. return $\varphi$.
```

Then, the hint distribution $\mathcal{H}^{\texttt{sim}}$ is defined as follows.

Hint distribution $\mathcal{H}^{\texttt{sim}}$ for our protocol

$\underline{\mathcal{H}_E^{\texttt{sim}} \text{ for a supersingular elliptic curve } E/\mathbb{F}_{p^2}:}$

```
1. chl, $\varphi_1 \leftarrow D_{\texttt{Chl}}(E)$. Let $E_1$ be the codomain of $\varphi_1$.
2. $\psi, I \leftarrow \texttt{FastCommit}_{2^{e_1}, 3^{e_2}}$. Let $E_2$ be the codomain of $\psi$.
3. $\varphi_2 \leftarrow D_{\texttt{RspIsog}}(E_1, E_2)$, where $\varphi_2 : E_1 \to E_2$ or $\varphi_2 = \perp$.
4. If $\varphi_2 = \perp$, go to Step 1.
5. Otherwise, write $\deg \varphi_2 = 2^n d'$ with $d'$ odd.
6. $\varphi_3 \leftarrow D_{\texttt{AuxIsog}}(E_2, \psi, 2^{e_1 - n} - d')$.
7. return $(\mathrm{chl}, \varphi_2, \varphi_3)$.
```

Now, we prove that Σ_{Push} is *$\mathcal{H}^{sim}$-hint-assisted wHVZK.*

Theorem 3 *Σ_{Push} is $\mathcal{H}^{sim}$-hint-assisted wHVZK.*

Proof. We define a simulator $\mathcal{S}(E_{\mathrm{pk}}, h) \to (E_{\mathrm{com}}, \mathrm{chl}, \mathrm{rsp})$ as follows.
Input: A public key E_{pk} and a hint $h = (\mathrm{chl}, \widehat{\varphi}_{\mathrm{rsp}}, \varphi_{\mathrm{aux}}) \leftarrow \mathcal{H}^{\texttt{sim}}_{E_{\mathrm{pk}}}$.
Output: A transcript $(E_{\mathrm{com}}, \mathrm{chl}, \mathrm{rsp})$.

1. Let E_{com} be the domain of φ_{rsp} and φ_{aux}, E_{chl} be the codomain of φ_{rsp}, and E_{aux} be the codomain of φ_{aux}.
2. Compute $\deg \varphi_{\mathrm{rsp}}$. Since $\deg \varphi_{\mathrm{rsp}} < 2^{e_1}$, it is determined by the 2^{e_1}-Weil pairing on E_{com} and E_{chl}.
3. Write $\deg \varphi_{\mathrm{rsp}} = 2^{n_1} d'_{\mathrm{rsp}}$, where d'_{rsp} odd and let $d_{\mathrm{aux}} = 2^{e_1 - n_1} - d'_{\mathrm{rsp}}$.
4. Take the deterministic basis $P_{\mathrm{aux}}, Q_{\mathrm{aux}}$ of $E_{\mathrm{aux}}[2^{e_1}]$.
5. Let $P_{\mathrm{chl}}, Q_{\mathrm{chl}} \leftarrow [d_{\mathrm{aux}}^{-1}]\varphi_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}(P_{\mathrm{aux}}), [d_{\mathrm{aux}}^{-1}]\varphi_{\mathrm{rsp}} \circ \widehat{\varphi}_{\mathrm{aux}}(Q_{\mathrm{aux}})$.
6. **return** $(E_{\mathrm{com}}, \mathrm{chl}, \mathrm{rsp} = (E_{\mathrm{aux}}, P_{\mathrm{chl}}, Q_{\mathrm{chl}}, n_1))$.

By construction, the distribution of the transcript $(E_{\mathrm{com}}, \mathrm{chl}, \mathrm{rsp})$ output by the simulator $\mathcal{S}$ is exactly the same as that of the real transcript output by Σ_{Push}. □

5.3 More Reasonable Hint Distribution

In order to prove the EUF-CMA security of our protocol, we need to assume that $(\mathcal{R}_{\texttt{OneEnd}}, \mathcal{H}^{\texttt{sim}})$ is a hard relation with hints [2, Definition 3.3] (see Section 5.5 for more details). That is equal to the hardness of the following problem.

Problem 1 (q-$\mathcal{H}^{\texttt{sim}}$-hint-OneEnd) *Given a supersingular elliptic curve E over $\mathbb{F}_{p^2}$ and q hints $h_1, \ldots, h_q \leftarrow \mathcal{H}_E^{\texttt{sim}}$, find an endomorphism $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$.*

However, the hint distribution $\mathcal{H}^{\texttt{sim}}$ we used is too powerful: it solves an isogeny problem in $D_{\texttt{RspIsog}}$. Thus, we introduce another hint distribution $\mathcal{H}^{\texttt{unif}}$: rather than sampling a random curve and generating a connecting isogeny, the new hint distribution $\mathcal{H}_E^{\texttt{unif}}$ samples random isogenies directly from E. We show the hint distribution $\mathcal{H}^{\texttt{unif}}$ below.

Hint distribution $\mathcal{H}^{\texttt{unif}}$

$\underline{\mathcal{H}_E^{\texttt{unif}}:}$

1. Sample an integer d from a weighted distribution on $(0, 2^{e_1})_3$, where each integer d, with prime factorization $\prod_{i=1}^t p_i^{m_i}$, has weight $\prod_{i=1}^t (p_i + 1)^{m_i}$.
2. Sample an isogeny $\psi'_1 : E \to E_1$ uniformly among the (possibly non-cyclic) isogenies from E of degree d.
3. Write ψ_1 for the cyclic component of ψ'_1.
4. Write $\deg \psi'_1 = 2^{n_1} \cdot d_1$, where d_1 odd.
5. Write $2^{e_1 - n_1} - d_1 = 3^{n_2} \cdot d_2$, where d_2 coprime to 3.
6. Sample an isogeny $\psi_2 : E_1 \to E'_2$ uniformly among the isogenies from E_1 of degree d_2.
7. **return** (ψ_1, ψ_2).

Using a hint $h = (\psi_1, \psi_2)$ sampled from $\mathcal{H}_E^{\texttt{unif}}$, we can make something that looks like a hint from $\mathcal{H}_E^{\texttt{sim}}$ in polynomial time as follows:

1. Let E_1 be the codomain of ψ_1, and let E_2 be the codomain of ψ_2.
2. Let $(\mathrm{chl}, \varphi) \leftarrow D_{\mathrm{chl}}(E)$.

3. Write $\psi = \psi_2 \circ \psi_1 : E \to E_2$.
4. Compute the push-forward $\psi' : E' \to E_2'$ of ψ by φ. Since we can compute a generator of $\ker \varphi$ from chl, we can compute the push-forward as in Algorithm 2.
5. Decompose ψ' into $\psi' = \psi_2' \circ \psi_1'$, where $\deg \psi_i' = \deg \psi_i$ for $i \in \{1, 2\}$. We can compute ψ_1' and ψ_2' using `KaniEval`.
6. Compute $d \coloneqq \deg \psi_1'$. Since $\deg \psi_1' < 2^{e_1}$, it is determined by the 2^{e_1}-Weil pairing on E and E_1.
7. Write $d = 2^{n_1} \cdot d_1$, where d_1 odd.
8. Write $2^{e_1 - n_1} - d_1 = 3^{n_2} \cdot d_2$, where d_2 coprime to 3.
9. If n_2 is odd, let $\psi_2'' \leftarrow [3^{(n_2-1)/2}] \circ \varphi$ for a randomly chosen 3-isogeny φ uniformly among 3-isogenies from E_2'. Otherwise, let $\psi_2'' \leftarrow [3^{n_2/2}]$.
10. **return** $(\mathrm{chl}, \psi_1', \psi_2'' \circ \psi_2')$.

We denote this algorithm by `MakeHint`(h). Now, we define the problem to distinguish hints sampled from $\mathcal{H}^{\mathtt{sim}}$ and the outputs of `MakeHint`.

Problem 2 (q-hint-dist) *Given a supersingular elliptic curve E over $\mathbb{F}_{p^2}$ and q hints $h_1, \ldots, h_q$ sampled with probability 1/2 from either distribution:*

- $\mathcal{D}_0 = \{(h_1, \ldots, h_q) \mid h_i \leftarrow \mathcal{H}_E^{sim}$ *for* $i \in \{1, \ldots, q\}\}$.
- $\mathcal{D}_1 = \{(h_1, \ldots, h_q) \mid h_i' \leftarrow \mathcal{H}_E^{unif}, h_i \leftarrow$ ***MakeHint***(h_i') *for* $i \in \{1, \ldots, q\}\}$.

Determine from which distribution the hints are sampled.

Under the hardness of Prioblem 2, the hardness of Problem 1 is equal to the hardness of the following problem.

Problem 3 (q-$\mathcal{H}^{\mathtt{unif}}$-hint-OneEnd) *Given a supersingular elliptic curve E over $\mathbb{F}_{p^2}$ and q hints $h_1, \ldots, h_q \leftarrow \mathcal{H}_E^{unif}$, find an endomorphism $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$.*

Therefore, the hardness of Problem 2 and 3 induces the hardness of Problem 1.

5.4 Hardness Analysis

We discuss the hardness of Probelem 2 and 3. We denote by $\mathcal{E}$ the set of all supersingular elliptic curves over $\mathbb{F}_{p^2}$ up to isomorphism. For a curve E, we let

$$\mathcal{C}_E \coloneqq \{E' \in \mathcal{E} \mid \exists\ \mathrm{chl} \in (0, 3^{e_2}),\ \exists\ \varphi : E \to E' \text{ s.t.} \ker \varphi = \langle P + [\mathrm{chl}]Q \rangle\},$$

where P, Q be the deterministic basis of $E[3^{e_2}]$. We also denote the uniform distribution over $\mathcal{E}$ by $\mathcal{E}^{\mathtt{unif}}$.

Hardness of Problem 2: Let $h = (\mathrm{chl}, \varphi_2, \varphi_3)$ be one of the q hints sampled from either $\mathcal{D}_0$ or $\mathcal{D}_1$. Let E_1 be the domain of φ_2, let E_2 be the codomain of φ_2 (equal to the domain of φ_3), and let E_3 be the codomain of φ_3. In addition, we let φ_1 be the isogeny with kernel $\langle P + [\mathrm{chl}]Q \rangle$ for the deterministic basis (P, Q) of $E[3^{e_2}]$. Then, φ_1 is the isogeny from E to E_1.

First, we consider the case where $\mathcal{D}_{\texttt{RspIsog}}$ does not return $\perp$ in $\mathcal{H}_E^{\texttt{sim}}$. Then, the distribution of φ_1 is exactly the same in both $\mathcal{D}_0$ and $\mathcal{D}_1$. Next, the conditional distribution of φ_2 given E_1 and E_2 is also the same in both cases: uniform among isogenies $E_1 \to E_2$ of degree $\in (0, 2^{e_1})_3$. Therefore, the only distinguishing factor for φ_2 is the distribution of E_2. In $\mathcal{D}_0$, the curve E_2 is the output of $\texttt{FastCommit}_{2^{e_1},3^{e_2}}$ and its distribution is considered close to $\mathcal{E}^{\texttt{unif}}$ as discussed in [11, Section 5.2]. In $\mathcal{D}_1$, the curve E_2 is the codomain of a random isogeny φ_2 from E_1 of bounded degree and its distribution is also considered close to $\mathcal{E}^{\texttt{unif}}$ as discussed in [2, Remark 5.1]. Now, the only distinguishing factor is φ_3. Write $\deg \varphi_3 = 3^n d'$, where d' coprime to 3 and decompose φ_3 into $\varphi_3^{(0)} \circ \varphi_3^{(1)}$ such that $\deg \varphi_3^{(0)} = 3^n$ and $\deg \varphi_3^{(1)} = d'$. Since the 3^n-isogeny $\varphi_3^{(0)}$ is sampled in the same way in both cases, the only difference is in the distribution of $\varphi_3^{(1)}$: in $\mathcal{D}_0$ is output of `PushRandIsog`, while in $\mathcal{D}_1$ is distributed uniformly among d'-isogenies from E_2. Recall that `PushRandIsog` computes the push-forward of a d'-isogeny φ_0 by an isogeny $\psi : E_0 \to E_2$, where φ_0 is determined by an endomorphism computed by `FullRepresentInteger` (see Algorithm 1). The distribution of φ_0 is considered close to the unifrom distribution among d'-isogenies from E_0 by the similar discussion for the output distribution of `RandIsogImg` [27, Section 3.1]. Therefore, the output ditribution of `PushRandIsog` is also considered close to unifrom.

Next, we discuss the effect of rejection in $\mathcal{D}_{\texttt{RspIsog}}$. As in the previous paragraph, we assume that the distribution of E_2 computed by `FastCommit` is close to $\mathcal{E}^{\texttt{unif}}$. In $\mathcal{H}_E^{\texttt{sim}}$, a pair of curves (E_1, E_2) is repeatedly sampled from $\mathcal{C}_E \times \mathcal{E}$ uniformly until there exists an isogeny $\varphi_2 : E_1 \to E_2$ of degree $\in (0, 2^{e_1})_3$. Therefore, the distribution of (E_1, E_2) is uniform over $\mathcal{P}_E \subseteq \mathcal{C}_E \times \mathcal{E}$ defined as follows:

$$\mathcal{P}_E \coloneqq \{(E_1, E_2) \in \mathcal{C}_E \times \mathcal{E} \mid \exists \text{ isogeny } \varphi_2 : E_1 \to E_2 \text{ of degree } \in (0, 2^{e_1})_3\}.$$

We let $\mathcal{C}_E(E_2) \coloneqq \{E_1 \in \mathcal{C}_E \mid (E_1, E_2) \in \mathcal{P}_E\}$ for every $E_2 \in \mathcal{E}$. Then, for each $E_2' \in \mathcal{E}$, we have

$$\Pr[E_2 = E_2' \mid (E_1, E_2) \in_U \mathcal{P}_E] = \frac{\#\mathcal{C}_E(E_2')}{\#\mathcal{P}_E}.$$

We denote this distribution by $\mathcal{E}^{\texttt{sim}}$, which is the distribution of E_2 sampled in $\mathcal{H}_E^{\texttt{sim}}$. Then, the statistical distance between $\mathcal{E}^{\texttt{sim}}$ and $\mathcal{E}^{\texttt{unif}}$ is

$$\begin{aligned}
\varDelta(\mathcal{E}^{\texttt{sim}}, \mathcal{E}^{\texttt{unif}}) &= \frac{1}{2} \sum_{E_2 \in \mathcal{E}} \left| \frac{\#\mathcal{C}_E(E_2)}{\#\mathcal{P}_E} - \frac{1}{\#\mathcal{E}} \right| \\
&\leq \frac{1}{2} \#\mathcal{E} \max_{E_2, E_2' \in \mathcal{E}} \left| \frac{\#\mathcal{C}_E(E_2)}{\#\mathcal{P}_E} - \frac{\#\mathcal{C}_E(E_2')}{\#\mathcal{P}_E} \right| \\
&= \frac{1}{2} \frac{\#(\mathcal{C}_E \times \mathcal{E})}{\#\mathcal{P}_E} \frac{1}{\#\mathcal{C}_E} \max_{E_2, E_2' \in \mathcal{E}} |\#\mathcal{C}_E(E_2) - \#\mathcal{C}({}_E E_2')| \\
&= \frac{\max_{E_2, E_2' \in \mathcal{E}} |\#\mathcal{C}_E(E_2) - \#\mathcal{C}_E(E_2')|}{2 \Pr[D_{\texttt{RspIsog}} \text{ does not output } \perp]} \cdot \frac{1}{\#\mathcal{C}_E},
\end{aligned}$$

where we used $\Pr[D_{\texttt{RspIsog}}$ does not output $\perp] = \frac{\#\mathcal{P}_E}{\#(\mathcal{C}_E \times \mathcal{E})}$. From the results in Section 4.3, $\Pr[D_{\texttt{RspIsog}}$ does not output $\perp] \approx 1$. In addition, $\#\mathcal{C}_E \approx 2^\lambda$ holds. Therefore, if $|\#\mathcal{C}_E(E_2) - \#\mathcal{C}_E(E_2')|$ is bounded by $o(1)$, the distance $\Delta(\mathcal{E}^{\texttt{sim}}, \mathcal{E}^{\texttt{unif}})$ is negligible.

Similarly, we let $\mathcal{E}(E_1) := \{E_2 \in \mathcal{E} \mid (E_1, E_2) \in \mathcal{P}_E\}$ for $E_1 \in \mathcal{C}_E$. Then, the distribution of E_1 sampled in $\mathcal{H}_E^{\texttt{sim}}$ is statistically close to the uniform distribution over $\mathcal{C}_E$ if $|\#\mathcal{E}(E_1) - \#\mathcal{E}(E_1')|$ is bounded by $o(1)$. We believe that $|\#\mathcal{E}(E_1) - \#\mathcal{E}(E_1')|$ does not become so large, but the detailed analysis is a future work.

Hardness of Problem 3: Our definition of $\mathcal{H}^{\texttt{unif}}$ is similar to that of SQIsign-v2.0 [1, Algorithm 5]: the essencial difference is that we sample an integer d to be coprime to 3. We do not expect this difference provides any help in finding an endomorphism. Therefore, q-$\mathcal{H}^{\texttt{unif}}$-hint-OneEnd is considered to be as hard as OneEnd (without hints) from the same discussion in [2, Section 5.5]. OneEnd is believed to computationally hard from [33,29]

5.5 EUF-CMA Security

Now, we discuss the EUF-CMA security of SQIsign2DPush. From [2, Theorem 1], if we can apply the Fiat-Shamir transform on Σ_{Push}, the resulting signature is EUF-CMA secure under the following conditions.

1. Σ_{Push} has high commitment entropy.
2. Σ_{Push} has a challenge space of exponential size in λ.
3. Σ_{Push} is special sound (Theorem 2).
4. Σ_{Push} is $\mathcal{H}^{\texttt{sim}}$-hint-asisted wHVZK (Theorem 3).
5. $(\mathcal{R}_{\texttt{OneEnd}}, \mathcal{H}^{\texttt{sim}})$ is a hard relation with hints (hardness of Problem 2 and 3).

We have already discussed conditions 3, 4, and 5 in the previous sections. Condition 2 is obvious. Regarding condition 1, as mentioned in the hardness analysis of Problem 2, the distribution of the commitment is expected to be close to $\mathcal{E}^{\texttt{unif}}$, thus a high commitment entropy is expected.

However, we apply the Fiat-Shamir transform with bounded aborts (FSwBA) on Σ_{Push} to obtain SQIsign2DPush. Therefore, we need to discuss whether [2, Theorem 1] similarly holds even when applying FSwBA. In particular, the distributions of commitment and challenge are affected by this change. However, as we mentioned in the hardness analysis of Problem 2, these distributions are close to those without considering aborts under the heuristics that $|\#\mathcal{C}_E(E_2) - \#\mathcal{C}_E(E_2')|$ and $|\#\mathcal{E}(E_1) - \#\mathcal{E}(E_1')|$ is bounded by $o(1)$. Hence, SQIsign2DPush is expected to be EUF-CMA secure, but a rigorous discussion is left for future work.

5.6 λ-Bit Security

We ensure that our protocol achieves λ-bit security under our parameter settings. From the previous discussion, q-$\mathcal{H}^{\texttt{sim}}$-hint-OneEnd is considered to be as hard as OneEnd. The best known algorithm to solve OneEnd requires $\tilde{O}(\sqrt{p})$ computations [17]. Since $p \approx 2^{2\lambda}$ in our setting, solving OneEnd requires $\tilde{O}(2^{\lambda})$ computations. In this attack, the most computationally intensive task involves searching for a path in the supersingular isogeny graph to a curve within a specific set, which has an approximate cardinality of $\tilde{O}(p^{1/2})$. Therefore, a quantum adversary has the potential to reduce the computational costs of these attacks to $\tilde{O}(p^{1/4}) = \tilde{O}(2^{\lambda/2})$ using the Grover's algorithm.

Next, we consider the attack to find an isogeny between E_0 and E_{pk} (or E_{com}). Between these two curves, there exist 2^{2e_1}-isogeny and 3^{2e_2}-isogeny. Therefore, using the meet-in-the-middle attack [20], one can find an isogeny with a cost of $\tilde{O}(2^{e_1})$ or $\tilde{O}(3^{e_2})$. Since $2^{e_1} \approx 3^{e_2} \approx 2^{\lambda}$ in our setting, this attack also requires $\tilde{O}(2^{\lambda})$ computations. In the case of a quantum adversary, by using the method in [31], the cost is reduced to $\tilde{O}(2^{2\lambda/3})$.

Finally, we consider the attack against the hash function that outputs a challenge. Our challenge space is $[0, 3^{e_2})$, and its size is approximately $3^{e_2} \approx 2^{\lambda}$, which is sufficient to achieve λ-bit security. However, the actual parameters shown in Section 6.1 indicate that $\log_2 3^{e_2}$ is slightly smaller than λ. Therefore, we apply the *grinding technique*: to compensate for the $\lambda - \log_2 3^{e_2}$ missing bits, the hash function is constructed by iterating $\lceil 2^{\lambda}/3^{e_2} \rceil$ times a standard hash function. Note that the grinding technique is also used in SQIsign-v2.0. In the case of a quantum adversary, the Grover's algorithm reduces the cost to $O(2^{\lambda/2})$.

6 Efficiency

In this section, we analyse the efficiency of SQIsign2DPush. First, we provide concrete parameters of our protocol for NIST security level 1, 3, and 5. Then we compare the data sizes of our protocol with SQIsign2D-East and SQIsign-v2.0. Finally, we compare the computational cost of SQIsign2DPush. Note that we consider Compact-SQIsign2D-East for SQIsign2D-East.

6.1 Parameters

We give concrete parameters of SQIsign2DPush satisfying the NIST security level 1, 3, and 5. We also show the number of iterations of the hash function for the grinding technique.

Security	e_1	e_2	c	p	Hash iterations
Level 1	131	78	1	$2^{131} \cdot 3^{78} - 1$	21
Level 3	191	117	1	$2^{191} \cdot 3^{117} - 1$	95
Level 5	263	156	1	$2^{263} \cdot 3^{156} - 1$	430

6.2 Data Sizes

In this subsection, we compare the data sizes of SQIsign2D-East, SQIsign-v2.0, and our SQIsign2DPush using the above parameters. Table 3 shows the sizes of each public key and signature. As shown in Table 3, the signature size of SQIsign2DPush is comparable to SQIsign-v2.0 and SQIsign2D-East. The public key size of our protocol is almost the same as SQIsign-v2.0 and SQIsign2D-East.

Table 3. Public key and signature size comparison in bytes

Security	Protocol	Public key	Signature
Level 1	SQIsign2D-East	66	164
	SQIsign-v2.0	65	148
	SQIsign2DPush	**66**	**151**
Level 3	SQIsign2D-East	98	245
	SQIsign-v2.0	97	224
	SQIsign2DPush	**98**	**219**
Level 5	SQIsign2D-East	128	331
	SQIsign-v2.0	129	292
	SQIsign2DPush	**130**	**297**

6.3 Computational Cost

We compare the computational costs in security level 1. Table 4 shows the number of isogeny computations of each degree.

Table 4. Number of isogeny computations of each degree. The numbers in parentheses mean that they may vary slightly depending on the case.

Protocol (Security level 1)		2	3	$(2,2)$
SQIsign2D-East	**Gen**	-	-	253
	Sign	127	(2)	770
	Verf	127	(2)	129
SQIsign-v2.0	**Gen**	-	-	744
	Sign	(248)	-	1488
	Verf	(248)	-	(126)
SQIsign2DPush	**Gen**	393	234	-
	Sign	(262)	468	(262)
	Verf	(0)	78	(131)

As for the signing, the number of $(2,2)$-isogeny computations of our protocol is less than 1/3 of SQIsign2D-East and less than 1/5 of SQIsign-v2.0. Though the number of 1-dimensional isogeny computations is larger than SQIsign-v2.0 and SQIsign2D-East, these costs are expected to be smaller than that of $(2,2)$-isogeny computations. Therefore, the total signing cost of our protocol expected to be smaller than both.

As for the verification, the number of $(2,2)$-isogeny computations of our protocol is comparable to SQIsign-v2.0 and SQIsign2D-East. However, we need to compute one point image during the $(2,2)$-isogeny computations in our protocol. Therefore, our computational cost is considered to be slightly higher than SQIsign-v2.0 and SQIsign2D-East.

In addition, the computational cost related to the quaternion algebra is also reduced in our protocol. Our protocol requires twice `FullRepresentInteger` and once lattice enunmuration for signing. On the other hand, SQIsign-v2.0 requires four `FullRepresentInteger` to execute `IdealToIsogeny` [1, Algorithm 3.13] twice and several computations to find an ideal of proper norm [1, Algorithm3.10, Algorithm 3.16]. SQIsign2D-East also requires some heavy computations of quaternion in `GenRandIsogImg` [28, Algorithm 2]. Although not as costly as $(2,2)$-isogeny computations, the computational cost related to the quaternion algebra still accounts for a considerable portion of the signing cost, which makes this advantage appear valuable as well.

Finally, we developed a proof-of-concept implementation of SQIsign2DPush in Julia language [6] with its computer algebra package Nemo [19]. The implementation is available at:

`https://github.com/hiroshi-onuki/SQIsign2D-Push`.

Table 5 shows the computational times of the implementation. These are the averages of 100 run times. The computational times are measured on a computer with an Intel Core i7-10700K CPU@3.70Hz without Turbo Boost. The cost evaluation through an optimized implementation is a future work.

Table 5. Computational times (sec.)

Security	**Gen**	**Sign**	**Verf**
Level 1	0.28	0.85	0.14
Level 3	0.50	1.62	0.33
Level 5	1.04	2.60	0.45

7 Conclusion

In this paper, we introduce SQIsign2DPush, a new variant of SQIsign with faster signing. As a building block of our protocol, we construct a new algorithm named `PushRandIsog`, which computes a given (generally non-smooth) degree isogeny from a given elliptic curve. Using `PushRandIsog`, we compute an auxiliary isogeny faster than the other variants of SQIsign and reduce the computational cost of signing.

Indeed, the number of $(2,2)$-isogeny computations for signing of our protocol is less than $1/3$ of SQIsign2D-East and less than $1/5$ of SQIsign-v2.0. The verification cost of our protocol is almost the same as SQIsign2D-East and SQIsign-v2.0. In addition, the signature size of our protocol is comparable to SQIsign2D-East and SQIsign-v2.0.

As a future work, we need to evaluate the failure probability of the response in more detail. We also need to provide a complete proof of EUF-CMA security. The cost evaluation through an optimized implementation is also a future work.

Acknowledgement

This work was supported by JST K Program Grant Number JPMJKP24U2, Japan. We would like to thank Haruhisa Kosuge for providing valuable information on the Fiat-Shamir transform with aborts.

References

1. Marius A. Aardal, Gora Adj, Diego F. Aranha, Andrea Basso, Isaac Andrés Canales Martínez, Jorge Chávez-Saab, Maria Corte-Real Santos, Pierrick Dartois, Luca De Feo, Max Duparc, Jonathan Komada Eriksen, Tako Boris Fouotsa, Décio Luiz Gazzoni Filho, Basil Hess, David Kohel, Antonin Leroux, Patrick Longa, Luciano Maino, Michael Meyer, Kohei Nakagawa, Hiroshi Onuki, Lorenz Panny, Sikhar Patranabis, Christophe Petit, Giacomo Pope, Krijn Reijnders, Damien Robert, Francisco Rodríguez Henríquez, Sina Schaeffler, and Benjamin Wesolowski. SQIsign. Technical report, National Institute of Standards and Technology, 2024. available at `https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures`.
2. Marius A. Aardal, Andrea Basso, Luca De Feo, Sikhar Patranabis, and Benjamin Wesolowski. A complete security proof of SQIsign. Cryptology ePrint Archive, Report 2025/379, 2025.
3. Andrea Basso, Pierrick Dartois, Luca De Feo, Antonin Leroux, Luciano Maino, Giacomo Pope, Damien Robert, and Benjamin Wesolowski. SQIsign2D-West - the fast, the small, and the safer. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part III*, volume 15486 of *LNCS*, pages 339–370. Springer, Singapore, December 2024.
4. Andrea Basso and Luciano Maino. POKÉ: A compact and efficient pke from higher-dimensional isogenies. In Serge Fehr and Pierre-Alain Fouque, editors, *EUROCRYPT2025*, volume 15602 of *LNCS*, pages 94–123. Springer, Cham, May 2025.
5. Andrea Basso, Luciano Maino, and Giacomo Pope. FESTA: Fast encryption from supersingular torsion attacks. In Jian Guo and Ron Steinfeld, editors, *ASIACRYPT 2023, Part VII*, volume 14444 of *LNCS*, pages 98–126. Springer, Singapore, December 2023.
6. Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017.
7. Wouter Castryck and Thomas Decru. An efficient key recovery attack on SIDH. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part V*, volume 14008 of *LNCS*, pages 423–447. Springer, Cham, April 2023.
8. 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. Technical report, National Institute of Standards and Technology, 2023. available at `https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures`.
9. Craig Costello and Hüseyin Hisil. A simple and compact algorithm for SIDH with arbitrary degree isogenies. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part II*, volume 10625 of *LNCS*, pages 303–329. Springer, Cham, December 2017.

10. Craig Costello, David Jao, Patrick Longa, Michael Naehrig, Joost Renes, and David Urbanik. Efficient compression of SIDH public keys. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part I*, volume 10210 of *LNCS*, pages 679–706. Springer, Cham, April / May 2017.
11. Pierrick Dartois, Antonin Leroux, Damien Robert, and Benjamin Wesolowski. SQIsignHD: New dimensions in cryptography. In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part I*, volume 14651 of *LNCS*, pages 3–32. Springer, Cham, May 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. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part III*, volume 15486 of *LNCS*, pages 304–338. Springer, Singapore, December 2024.
13. Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski. SQISign: Compact post-quantum signatures from quaternions and isogenies. In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part I*, volume 12491 of *LNCS*, pages 64–93. Springer, Cham, December 2020.
14. Luca De Feo, Antonin Leroux, Patrick Longa, and Benjamin Wesolowski. New algorithms for the Deuring correspondence - towards practical and secure SQISign signatures. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part V*, volume 14008 of *LNCS*, pages 659–690. Springer, Cham, April 2023.
15. Max Deuring. Die typen der multiplikatorenringe elliptischer funktionenkörper. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 14:197–272, 1941.
16. Max Duparc and Tako Boris Fouotsa. SQIPrime: A dimension 2 variant of SQISignHD with non-smooth challenge isogenies. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part III*, volume 15486 of *LNCS*, pages 396–429. Springer, Singapore, December 2024.
17. Kirsten Eisenträger, Sean Hallgren, Chris Leonardi, Travis Morrison, and Jennifer Park. Computing endomorphism rings of supersingular elliptic curves and connections to path-finding in isogeny graphs. *Open Book Series*, 4(1):215–232, 2020.
18. Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In Andrew M. Odlyzko, editor, *CRYPTO'86*, volume 263 of *LNCS*, pages 186–194. Springer, Berlin, Heidelberg, August 1987.
19. Claus Fieker, William Hart, Tommy Hofmann, and Fredrik Johansson. Nemo/hecke: Computer algebra and number theory packages for the julia programming language. In *Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation*, ISSAC '17, pages 157–164, New York, NY, USA, 2017. ACM.
20. Steven D Galbraith. Constructing isogenies between elliptic curves over finite fields. *LMS Journal of Computation and Mathematics*, 2:118–138, 1999.
21. David Jao, Reza Azarderakhsh, Matthew Campagna, Craig Costello, Luca De Feo, Basil Hess, Amir Jalali, Brian Koziel, Brian LaMacchia, Patrick Longa, Michael Naehrig, Joost Renes, Vladimir Soukharev, David Urbanik, Geovandro Pereira, Koray Karabina, and Aaron Hutchinson. SIKE. Technical report, National Institute of Standards and Technology, 2020. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions`.
22. David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In Bo-Yin Yang, editor, *Post-Quantum Cryp-*

tography - 4th International Workshop, PQCrypto 2011, pages 19–34. Springer, Berlin, Heidelberg, November / December 2011.

23. Ernst Kani. The number of curves of genus two with elliptic differentials. *Journal für die reine und angewandte Mathematik*, 1997(485):93–122, 1997.
24. David Kohel, Kristin Lauter, Christophe Petit, and Jean-Pierre Tignol. On the quaternion-isogeny path problem. *LMS Journal of Computation and Mathematics*, 17(A):418–432, 2014.
25. Vadim Lyubashevsky. Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures. In Mitsuru Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 598–616. Springer, Berlin, Heidelberg, December 2009.
26. Luciano Maino, Chloe Martindale, Lorenz Panny, Giacomo Pope, and Benjamin Wesolowski. A direct key recovery attack on SIDH. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part V*, volume 14008 of *LNCS*, pages 448–471. Springer, Cham, April 2023.
27. Kohei Nakagawa and Hiroshi Onuki. QFESTA: Efficient algorithms and parameters for FESTA using quaternion algebras. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part V*, volume 14924 of *LNCS*, pages 75–106. Springer, Cham, August 2024.
28. Kohei Nakagawa, Hiroshi Onuki, Wouter Castryck, Mingjie Chen, Riccardo Invernizzi, Gioella Lorenzon, and Frederik Vercauteren. SQIsign2D-East: A new signature scheme using 2-dimensional isogenies. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part III*, volume 15486 of *LNCS*, pages 272–303. Springer, Singapore, December 2024.
29. Aurel Page and Benjamin Wesolowski. The supersingular endomorphism ring and one endomorphism problems are equivalent. In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part VI*, volume 14656 of *LNCS*, pages 388–417. Springer, Cham, May 2024.
30. Damien Robert. Breaking SIDH in polynomial time. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part V*, volume 14008 of *LNCS*, pages 472–503. Springer, Cham, April 2023.
31. Seiichiro Tani. Claw finding algorithms using quantum walk. *Theoretical Computer Science*, 410(50):5285–5297, 2009.
32. Jacques Vélu. Isogénies entre courbes elliptiques. *Comptes-Rendus de l'Académie des Sciences*, 273:238–241, 1971.
33. Benjamin Wesolowski. The supersingular isogeny path and endomorphism ring problems are equivalent. In *62nd FOCS*, pages 1100–1111. IEEE Computer Society Press, February 2022.