Ideal-to-isogeny algorithm using 2-dimensional isogenies and its application to SQIsign

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

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

Abstract. The Deuring correspondence is a correspondence between supersingular elliptic curves and quaternion orders. Under this correspondence, an isogeny between elliptic curves corresponds to a quaternion ideal. This correspondence plays an important role in isogeny-based cryptography and several algorithms to compute an isogeny corresponding to a quaternion ideal (ideal-to-isogeny algorithms) have been proposed. In particular, SQIsign is a signature scheme based on the Deuring correspondence and uses an ideal-to-isogeny algorithm. In this paper, we propose a novel ideal-to-isogeny algorithm using isogenies of dimension 2. Our algorithm is based on Kani's reducibility theorem, which gives a connection between isogenies of dimension 1 and 2. By using the characteristic p of the base field of the form $2^f g - 1$ for a small odd integer g, our algorithm works by only 2-isogenies and $(2, 2)$-isogenies in the operations in $\mathbb{F}_{p^2}$. We apply our algorithm to SQIsign and compare the efficiency of the new algorithm with the existing one. Our analysis shows that the key generation and the signing in our algorithm are at least twice as fast as those in the existing algorithm at the NIST security level 1. This advantage becomes more significant at higher security levels. In addition, our algorithm also improves the efficiency of the verification in SQIsign.

Keywords: post-quantumcryptography · SQIsign · the Deuring correspondence · Kani's theorem.

1 Introduction

Isogeny-based cryptography is a promising candidate for post-quantum cryptography. Many isogeny-based schemes use supersingular elliptic curves because the isogeny graph of supersingular elliptic curves has a more attractive structure than that of ordinary elliptic curves. Some of these schemes use the Deuring correspondence, which is a correspondence between supersingular elliptic curves and quaternion orders. SQIsign is a signature scheme proposed by De Feo, Kohel, Leroux, Petit and Wesolowski [16] that uses the Deuring correspondence. It was submitted to the additional digital signature candidates for the NIST post-quantum cryptography standardization process [8]. In this paper, we refer the NIST submission of SQIsign as the SQIsign to distinguish it from the name of the scheme. An advantage of the SQIsign is that it has short key sizes and signature sizes compared to other candidates. Its disadvantage is that the signing algorithm is slow. This mainly comes from the computation of an isogeny corresponding to a quaternion ideal via the Deuring correspondence. We call an algorithm to compute an isogeny corresponding to a quaternion ideal an *ideal-to-isogeny algorithm*.

Ideal-to-isogeny algorithms are crucial for isogeny-based cryptography. Before SQIsign was proposed, ideal-to-isogeny algorithms were appeared in the cryptoanalysis by Eisenträger, Hallgren, Lauter, Morrison and Petit [24] and the signature scheme by Galbraith, Petit and Silva [27]. Although, the ideal-to-isogeny algorithms in these works have a polynomial-time complexity, they are not efficient in practice because they require operations on extension fields. The first efficient ideal-to-isogeny algorithm was proposed in SQIsign [16]. This algorithm does not require operations on extension fields, but it requires that the characteristic of the base field is in a special form. Later, the restriction on the characteristic was relaxed by De Feo, Leroux, Longa and Wesolowski [18].

Another important mathematical tool for isogeny-based cryptography is Kani's reducibility theorem [31]. This theorem gives a connection between isogenies between elliptic curves and isogenies between abelian surfaces, in other words, a connection between isogenies of dimension 1 and 2. Castryck and Decru [7] and Maino, Martindale, Panny, Pope, and Wesolowski [36] used this theorem to attack SIDH, which is an isogeny-based key exchange protocol by Jao and De Feo [30]. Robert [42] extended these attacks to attacks using a connection between isogenies of dimension 2 (resp. 4) and 4 (resp. 8). Later, this theorem has been used to construct isogeny-based schemes, for example, a signature scheme by [14], a public-key encryption scheme by [4], a key encapsulation mechanism by [38], and an updatable public-key encryption scheme by [22].

Some of these schemes use the Deuring correspondence in addition to Kani's reducibility theorem. SQIsignHD [14] uses isogenies of dimension 4 or 8 to confirm the existence of an isogeny corresponding to a quaternion ideal in its verification algorithm. QFESTA [38] uses isogenies of dimension 2 and the Deuring correspondence to generate a random isogeny between elliptic curves of given degree. Ideal-to-isogeny algorithms using Kani's reducibility theorem have been proposed in verifiable random functions by [34] and in SILBE [22].

1.1 Our contributions

Motivated by these developments, this paper advances this line of research by proposing a novel ideal-to-isogeny algorithm using isogenies of dimension 2. Our contributions are as follows:

1. Proposing a novel ideal-to-isogeny algorithm IdealToIsogenyIQO, which uses isogenies of dimension 2 and an embedding of an imaginary quadratic order into the endomorphism ring of the domain elliptic curve (IQO stands for Imaginary Quadratic Order).
2. Applying IdealToIsogenyIQO to SQIsign and comparing the efficiency of the new algorithm with the existing one.

Our algorithm is based on a similar idea as the algorithm in SILBE, which uses isogenies of dimension 4. Compared to the algorithm in SILBE, our algorithm has two advantages. The first advantage is using more efficient isogenies of dimension 2 instead of isogenies of dimension 4. The second advantage is that our algorithm does not require that the degree of the output isogeny of dimension 1 is prime to the degree of isogenies of dimension 2. Thanks to these advantages, we can use only 2-isogenies and $(2, 2)$-isogenies to run our algorithm in practice if we choose the characteristic of the base field properly.

As an application of our algorithm, we propose a new algorithm for SQIsign. Our algorithm uses the characteristics of the form $2^f g - 1$ for a small odd integer g. The isogenies directly computed in our algorithm are only 2-isogenies and $(2, 2)$-isogenies. By using an efficient algorithm to compute $(2, 2)$-isogenies by Dartios, Maino, Pope, and Robert [15], we expect that the key generation and the signing in our algorithm are faster than those in the SQIsign. The verification in our algorithm is faster than that in the SQIsign because the number of the separations of the isogeny chain in the signature of our algorithm is smaller than that in the SQIsign. Note that our algorithm does not affect the security of the SQIsign, and the sizes of the keys and the signatures of our algorithm are almost the same as those of the SQIsign because we just replace the ideal-to-isogeny algorithm in the SQIsign.

1.2 Related works

As mentioned in [34, §6], the ideal-to-isogeny algorithm in [34] could be applied to SQIsign in a manner similar to our algorithm. This algorithm also uses isogenies of dimension 2, but takes a different approach from our algorithm. We discuss a comparison with this algorithm in Section 4.6.

SQIsignHD [14] is a variant of SQIsign, which uses isogenies of dimension 4 or 8. The key generation and signing algorithms in SQIsignHD are more efficient than those in SQIsign while the verification algorithm in SQIsignHD is slower than that in SQIsign. In terms of key and signature sizes, SQIsignHD has the same key sizes as SQIsign and smaller signature sizes than SQIsign. Furthermore, SQIsignHD relies on distinct assumptions from SQIsign for security.

Although we improve the efficiency of SQIsign, the key generation and signing algorithms in our algorithm are slower than those in SQIsignHD. Nonetheless, we contend that our approach remains valuable due to its fast verification process compared to these schemes. Furthermore, the exploration of diverse isogeny-based schemes based on distinct assumptions remains crucial.

At the same time as this work, other variants of SQIsign, SQIsign2D-West [3], SQIPrime [21], and SQIsign2D-East [39], have been proposed. These variants use isogenies of dimension 2 or 4 and offer different trade-offs between efficiency and security. We leave the comparison with these schemes as future work. In addition, a new ideal-to-isogeny algorithm using isogenies of dimension 2 was proposed in SQIsign2D-West. It could be applied to SQIsign similarly to our algorithm. We also leave the comparison with this algorithm as future work.

1.3 Organization

The rest of this paper is organized as follows. In Section 2, we give the technical background on this paper. In particular, Section 2.1 gives the mathematical background, Section 2.2 gives the existing ideal-to-isogeny algorithms, and Section 2.3 explains the outline of SQIsign. In Section 3, we propose a novel ideal-to-isogeny algorithm using isogenies of dimension 2. In Section 4, we apply our algorithm to SQIsign and compare the efficiency of the new algorithm with the existing one. Finally, we conclude this paper in Section 6.

2 Preliminaries

This section gives the technical background on this paper. Throughout this paper, we let p be a prime number of cryptographic size, i.e., p is at least about 2^{256}.

2.1 Mathematical background

In this subsection, we recall the mathematical background necessary for the rest of this paper.

Supersingular elliptic curves. Let E be an elliptic curve over a finite field of characteristic p. We denote the neutral element of E by O_E. For an integer n, the *n-torsion subgroup* of E is defined by $E[n] = \{P \in E \mid nP = O_E\}$. If the trace of the Frobenius endomorphism of E is congruent to 0 modulo p, then E is called *supersingular*. A supersingular elliptic curve over a field of characteristic p is isomorphic to a curve E defined over $\mathbb{F}_{p^2}$ such that the p^2-th power Frobenius endomorphism of E is the multiplication-by-$(-p)$ map. Then we have $E(\mathbb{F}_{p^2}) = E[p+1]$. This property is preserved under isogenies over $\mathbb{F}_{p^2}$, i.e., if there exists an isogeny $E \to E'$ defined over $\mathbb{F}_{p^2}$ then E' is also a supersingular elliptic curve such that $E'(\mathbb{F}_{p^2}) = E'[p+1]$. In the rest of this paper, we assume that all elliptic curves are supersingular and satisfy the above property.

Abelian surfaces. An elliptic curve is an abelian variety of dimension 1. The generalization of elliptic curves to dimension 2 is called an *abelian surface*. An abelian surfaces is *principally polarized* if it is isomorphic to its dual abelian surface. A principally polarized abelian surface is isomorphic to the Jacobian of a genus-2 hyperelliptic curve or the product of two elliptic curves.

Isogenies. An *isogeny* is a rational map between principally polarized abelian varieties which is a surjective group homomorphism and has finite kernel. The *degree* of an isogeny φ is its degree as a rational map and 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 $\varphi : A \to B$, the *dual isogeny* of φ is the isogeny $\hat{\varphi} : B \to A$ such that $\hat{\varphi} \circ \varphi$ is equal to the multiplication-by-$\deg \varphi$ map on A. Note that the dual isogeny uniquely exists.

Let ℓ be a positive integer. We say an isogeny φ between two elliptic curves is an *ℓ-isogeny* if $\ker \varphi$ is a cyclic group of order ℓ. We say an isogeny φ between two principally polarized abelian surfaces is an *(ℓ, ℓ)-isogeny* if the Weil pairing acts trivially on $\ker \varphi$ and the order of $\ker \varphi$ is ℓ^2.

Algorithms to compute isogenies An isogeny between elliptic curves can be computed by Vélu's formulas [44]. Let $\varphi : E \to E'$ be an ℓ-isogeny between elliptic curves. Vélu's formulas give an algorithm to compute E' with input E and a generator of $\ker \varphi$ in $O(\ell)$ operations on a field where the generator is defined. For an additional input $P \in E$, we can compute $\varphi(P)$ in $O(\ell)$ operations on a field where the generator and P are defined. These computational costs were improved to $\tilde{O}(\sqrt{\ell})$ by [5].

Algorithms to compute a $(2, 2)$-isogeny can be found in [43] and [28]. Recently, an efficient algorithm for a $(2, 2)$-isogeny using theta functions was given by [15]. An algorithm for a general degree was given by [11] and later improved by [35]. The computational cost of this algorithm is $O(\ell^2)$ operations on a field where a generator of the kernel is defined.

Let d be a positive integer prime to p having the prime factorization $d = \prod_i \ell_i$ and φ be a d-isogeny or (d, d)-isogeny. Then we can compute φ as the composition of ℓ_i-isogenies or (ℓ_i, ℓ_i)-isogenies. Therefore, if d is smooth and a generator of $\ker \varphi$ are defined over a $\mathbb{F}_{p^k}$ of $k \in \mathrm{poly}(\log p)$, then we can compute a d-isogeny in polynomial time in $\log p$.

Quaternion algebras. We denote by $\mathcal{B}_{p,\infty}$ the quaternion algebra over $\mathbb{Q}$ ramified at p and ∞. The quaternion algebra $\mathcal{B}_{p,\infty}$ has $\mathbb{Q}$-basis $\{1, \mathbf{i}, \mathbf{j}, \mathbf{k}\}$ such that $\mathbf{i}^2 = -q, \mathbf{j}^2 = -p, \mathbf{k} = \mathbf{ij} = -\mathbf{ji}$ for some positive integer q. If $p \equiv 3 \pmod 4$, then we let $q = 1$. If $p \equiv 5 \pmod 8$, then we let $q = 2$. Otherwise, we let q be the smallest prime number such that q is a quadratic non-residue modulo p. The *canonical involution* of $\alpha = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k} \in \mathcal{B}_{p,\infty}$ is defined by $\bar{\alpha} \coloneqq a - b\mathbf{i} - c\mathbf{j} - d\mathbf{k}$. The *trace* and the *norm* of α are defined by $\mathrm{tr}(\alpha) \coloneqq \alpha + \bar{\alpha}$ and $\mathrm{n}(\alpha) \coloneqq \alpha\bar{\alpha}$, respectively. An order in $\mathcal{B}_{p,\infty}$ is a subring of $\mathcal{B}_{p,\infty}$ that is a free $\mathbb{Z}$-module of rank 4. A maximal order in $\mathcal{B}_{p,\infty}$ is an order that is maximal with respect to inclusion. A *fractional ideal* of $\mathcal{B}_{p,\infty}$ is a $\mathbb{Z}$-submodule of $\mathcal{B}_{p,\infty}$ of rank 4. Let I be a fractional ideal of $\mathcal{B}_{p,\infty}$. The *canonical involution* of I is defined by $\bar{I} \coloneqq \{\bar{\alpha} \mid \alpha \in I\}$. The *left order* of I is defined by $\mathcal{O}_{\mathrm{L}}(I) \coloneqq \{x \in \mathcal{B}_{p,\infty} \mid xI \subset I\}$ and we define the *right order* $\mathcal{O}_{\mathrm{R}}(I)$ of I similarly. For an order $\mathcal{O}$ of $\mathcal{B}_{p,\infty}$, we say I is a *left (or right) $\mathcal{O}$-ideal* if I is a left (or right) ideal of $\mathcal{O}$ in the usual sense. If I is a left $\mathcal{O}$-ideal for a maximal order $\mathcal{O}$, then $\mathcal{O}_{\mathrm{L}}(I) = \mathcal{O}$ and $\mathcal{O}_{\mathrm{R}}(I)$ is a maximal order. If I is contained in an order of $\mathcal{B}_{p,\infty}$, the we define the *norm* of I by $\mathrm{n}(I) \coloneqq \gcd(\{\mathrm{n}(\alpha) \mid \alpha \in I\})$. For $\alpha \in I$, we define the *normalized norm* *of* α by $q_I(\alpha) \coloneqq \mathrm{n}(\alpha)/\mathrm{n}(I)$. By the definition of the norm of I, the normalized norm $q_I(\alpha)$ is an integer for all $\alpha \in I$.

The Deuring correspondence. Deuring [20] 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 E curve over $\mathbb{F}_{p^2}$ corresponds to a maximal order isomorphic to $\mathrm{End}(E)$.

Fix a supersingular elliptic curve E_0 and an isomorphism $\iota : \mathcal{O}_0 \to \mathrm{End}(E_0)$. For a left $\mathcal{O}_0$-ideal I, we define the *I-torsion subgroup* of E_0 by $E_0[I] = \{P \in E_0 \mid \iota(\alpha)(P) = 0 \text{ for all } \alpha \in I\}$. If $\mathrm{n}(I)$ is not divisible by p, then $E_0[I]$ is a subgroup of E_0 of order $\mathrm{n}(I)$. In this case, we define an isogeny corresponding to I by an isogeny with kernel $E_0[I]$ and denote it by φ_I. Let E be the codomain of φ_I. Then $\mathrm{End}(E)$ is isomorphic to $\mathcal{O}_{\mathrm{R}}(I)$. In particular, an isomorphism is induced by

$$\mathcal{B}_{p,\infty} \to \mathrm{End}(E) \otimes_{\mathbb{Z}} \mathbb{Q};\ \alpha \mapsto \frac{1}{\mathrm{n}(I)} \varphi_I \circ \iota(\alpha) \circ \hat{\varphi}_I. \tag{1}$$

The canonical involution $\bar{I}$ of I corresponds to the dual isogeny $\hat{\varphi}_I$. Let J be a left $\mathcal{O}_{\mathrm{R}}(I)$-ideal and φ_J be an isogeny corresponding to J via the above isomorphism. Then the composition $\varphi_J \circ \varphi_I$ is an isogeny corresponding to IJ. If $\mathrm{n}(I)$ is prime to $\mathrm{n}(J)$, then $\ker \varphi_J = \varphi_I(E_0[IJ] \cap E_0[\mathrm{n}(J)])$. For more detailed discussion for the relation between ideals and isogenies, see [16, §4].

For left $\mathcal{O}_0$-ideals I_1 and I_2, the codomains of φ_{I_1} and φ_{I_2} are isomorphic if and only if there exists $\alpha \in \mathcal{B}_{p,\infty}$ such that $I_1 = I_2\alpha$. If this is the case, we say I_1 and I_2 are *equivalent* and denote it by $I_1 \sim I_2$.

Special extremal orders. Let R be the integer ring of $\mathbb{Q}(\mathbf{i})$. We say a maximal order $\mathcal{O}$ in $\mathcal{B}_{p,\infty}$ is *special extremal* if $\mathcal{O}$ contains $R + \mathbf{j}R$. In this paper, we mainly focus on the case $p \equiv 3 \pmod 4$. In this case, the maximal order $\mathcal{O}_0 \coloneqq \left\langle 1, \mathbf{i}, \frac{\mathbf{i}+\mathbf{j}}{2}, \frac{1+\mathbf{k}}{2} \right\rangle_{\mathbb{Z}}$ is a special extremal order and the supersingular elliptic curve with j-invariant 1728 corresponds to $\mathcal{O}_0$ via the Deuring correspondence. Let E_0 be the supersingular elliptic curve over $\mathbb{F}_{p^2}$ defined by $y^2 = x^3 + x$, which has j-invariant 1728. Then an isomorphism $\mathcal{O}_0 \to \mathrm{End}(E_0)$ is induced by $\mathbf{i} \mapsto ((x, y) \mapsto (-x, \sqrt{-1}y))$ and mapping $\mathbf{j}$ to the p-th power Frobenius endomorphism.

KLPT algorithms. An algorithm to transform an ideal to another equivalent ideal is given by [32], which is called the *KLPT algorithm.* Let $\mathcal{O}_0$ be a special extremal order in $\mathcal{B}_{p,\infty}$. The KLPT algorithm takes a left $\mathcal{O}_0$-ideal I and a smooth integer $n > p^{3.5}$ as input and outputs a left $\mathcal{O}_0$-ideal J such that $J \sim I$ and $\mathrm{n}(J) = n$. Its computational cost is bounded by a polynomial in $\log p$ under heuristic assumptions. Later, the bound $p^{3.5}$ was improved to p^3 by [41].

The KLPT algorithm was generalized to ideals of arbitrary maximal orders by [16]. We call this the *generalized KLPT algorithm.* Let $\mathcal{O}$ be a maximal order of $\mathcal{B}_{p,\infty}$ and I_0 be a left $\mathcal{O}_0$-ideal such that $\mathcal{O}_{\mathrm{R}}(I_0) = \mathcal{O}$. The generalized KLPT algorithm takes I_0, a left $\mathcal{O}$-ideal I, and a smooth integer $n > p^3 \mathrm{n}(I_0)^3$ as input and outputs a left $\mathcal{O}$-ideal J such that $J \sim I$ and $\mathrm{n}(J) = n$.

These algorithms are based on the fact that $I\bar{\alpha}/\mathrm{n}(I)$ is a left $\mathcal{O}_{\mathrm{L}}(I)$-ideal of norm $q_I(\alpha)$ for $\alpha \in I$ [32, Lemma 5]. Indeed, these algorithms find $\alpha \in I$ such that $q_I(\alpha) = n$ and output $I\bar{\alpha}/\mathrm{n}(I)$. We denote the ideal $I\bar{\alpha}/\mathrm{n}(I)$ by $\chi_I(\alpha)$. Then $\hat{\varphi}_{\chi_I(\alpha)} \circ \varphi_I$ is equal to α as endomorphisms up to post-composition of an automorphism of E_0. Note that the right order of $\chi_I(\alpha)$ is $\alpha\mathcal{O}_{\mathrm{R}}(I)\alpha^{-1}$, which is isomorphic to $\mathcal{O}_{\mathrm{R}}(I)$ but not equal to $\mathcal{O}_{\mathrm{R}}(I)$ in general.

Kani's reducibility theorem. Let d_1 and d_2 be positive integers prime to each other and p. Let $\varphi_1 : E_0 \to E_1$ be a d_1-isogeny and $\varphi_2 : E_0 \to E_2$ be a d_2-isogeny between elliptic curves over a field of characteristic p. Then we say an isogeny with kernel $\varphi_1(\ker \varphi_2)$ a *push-forward of* φ_2 *by* φ_1 and denote it by $\varphi_{1*}\varphi_2$. Since $\ker((\varphi_{1*}\varphi_2) \circ \varphi_1) = \langle \ker \varphi_1, \ker \varphi_2 \rangle = \ker((\varphi_{2*}\varphi_1) \circ \varphi_2)$, the codomains of $\varphi_{1*}\varphi_2$ and $\varphi_{2*}\varphi_1$ are isomorphic. Let F be the codomain of $\varphi_{1*}\varphi_2$. Then we can take $\varphi_{2*}\varphi_1$ so that the following diagram commutes:

$$\begin{array}{ccc} E_0 & \xrightarrow{\varphi_1} & E_1 \\ {\scriptstyle \varphi_2}\downarrow & & \downarrow{\scriptstyle \varphi_{1*}\varphi_2} \\ E_2 & \xrightarrow[\varphi_{2*}\varphi_1]{} & F. \end{array}$$

Kani [31] showed that this diagram induces an isogeny $E_1 \times E_2 \to E_0 \times F$. More precisely, we have the following theorem based on Kani's reducibility theorem [31, Theorem 2.3].

Theorem 1 ([36, Theorem 1]). *We use the same notation as above and let* $d = d_1 + d_2$. *Suppose that we take the push-forwards so that the above diagram is commutative. We define an isogeny*

$$\Phi = \begin{pmatrix} \hat{\varphi}_1 & \hat{\varphi}_2 \\ -\varphi_{1*}\varphi_2 & \varphi_{2*}\varphi_1 \end{pmatrix} : E_1 \times E_2 \to E_0 \times F,$$

i.e., $\Phi((P_1, P_2)) = (\hat{\varphi}_1(P_1) + \hat{\varphi}_2(P_2), -\varphi_{1*}\varphi_2(P_1) + \varphi_{2*}\varphi_1(P_2))$ *for* $P_1 \in E_1$ *and* $P_2 \in E_2$. *Then* Φ *is a* (d, d)*-isogeny with kernel* $\{(\varphi_1(P), \varphi_2(P)) \mid P \in E_0[d]\}$.

This theorem says that we can compute the images of any points under $\hat{\varphi}_1$ and $\hat{\varphi}_2$ by using the images of a basis of $E_0[d]$ under φ_1 and φ_2.

If F is not isomorphic to E_0, then an isogeny with the same kernel as Φ is of the form $\begin{pmatrix} \iota_0 & 0 \\ 0 & \iota \end{pmatrix} \circ \Phi$ or $\begin{pmatrix} 0 & \iota \\ \iota_0 & 0 \end{pmatrix} \circ \Phi$, where ι_0 and ι are automorphisms of E_0 and F, respectively. In this case, we can compute the images of any point in E_1 under $\hat{\varphi}_1$ by Algorithm 1. In the output of Algorithm 1, the automorphism ι_0 is not determined by the input and depends on the choice of Φ.

Algorithm 1: EvalByKani

Input: d, E_0, E_1, E_2 in Theorem 1, $\varphi_1(P), \varphi_2(P), \varphi_1(Q), \varphi_2(Q)$, where (P, Q) is a basis of $E_0[d]$, and a finite subset $S \subset E_1$.

Output: the image of S under $\iota_0 \circ \hat{\varphi}_1$, where ι_0 is an automorphism of E_0.

Let Φ be a (d, d)-isogeny with kernel $\langle(\varphi_1(P), \varphi_2(P)), (\varphi_1(Q), \varphi_2(Q))\rangle$;
Let pr_{E_0} be the projection to E_0 from the codomain of Φ;
return $\{\mathrm{pr}_{E_0} \circ \Phi((R, 0_{E_2})) \mid R \in S\}$;

2.2 Ideal-to-isogeny algorithms

In this subsection, we recall some of the existing algorithms for computing the codomain of an isogeny corresponding to a given ideal.

Let E_0 be a supersingular elliptic curve over $\mathbb{F}_{p^2}$ whose endomorphism ring is isomorphic to a special extremal order $\mathcal{O}_0$ in $\mathcal{B}_{p,\infty}$, and $\iota : \mathcal{O}_0 \to \mathrm{End}(E_0)$ be an isomorphism. Suppose that we can efficiency compute $\iota(\alpha)(P)$ for $\alpha \in \mathcal{O}_0$ and $P \in E_0(\mathbb{F}_{p^2})$. Given a left $\mathcal{O}_0$-ideal I, we consider an algorithm to compute the codomain of an isogeny corresponding to I. We call this algorithm an *ideal-to-isogeny algorithm*.

By using the KLPT algorithm, we can use an ideal J such that $J \sim I$ and $\mathrm{n}(J)$ is smooth and greater than p^3 instead of I. Since $\varphi_J = \mathrm{n}(J)$ is smooth, we can compute φ_J in polynomial time if $E[J]$ is defined over a field of polynomial size. However, $E[J]$ is in an exponential-size field in general. The following algorithms deal with this issue.

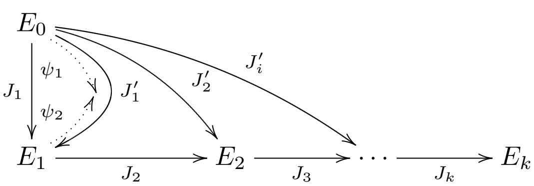
Algorithm for power-smooth norms An ideal-to-isogeny algorithm was first proposed by [27]. Their algorithm uses an ideal J such that $J \sim I$ and $\mathrm{n}(J)$ is power-smooth, i.e., any prime power dividing $\mathrm{n}(J)$ is small. Its computation requires operations on extension fields of $\mathbb{F}_{p^2}$, thus the algorithm is not efficient in practice.

Algorithm in the original SQIsign. An ideal-to-isogeny algorithm that works in $\mathbb{F}_{p^2}$ was proposed by [16]. This algorithm requires p to be a a special form while the algorithm of [27] does not. In particular, the algorithm by [16] requires that p^2-1 is divisible by ℓ^f and T, where ℓ is a small prime number, f is an integer and T is a smooth integer greater than $p^{1.5}$ and prime to ℓ. In this setting, we can compute ℓ^f-isogenies and T-isogenies over $\mathbb{F}_{p^2}$ by using the method in [12]. The basic idea of the algorithm is as follows:

1. Compute a left $\mathcal{O}_0$-ideal J such that $J \sim I$ and $\mathrm{n}(J)$ is a power of ℓ.
2. Divide J into the product $J_1 \cdots J_k$ such that $\mathrm{n}(J_i) = \ell$ (for simplicity, we assume $\mathrm{n}(J) = \ell^{kf}$ for an integer k).
3. Let $J'_0 = \mathcal{O}_0$ and $\alpha = 1$.
4. For $i = 1, \ldots, k$:
   (a) Compute $\varphi_{J_i} : E_{i-1} \to E_i$ by $\ker \varphi_{J_i} = \varphi_{J'_{i-1}}(E_0[J'_{i-1}\alpha J_i \alpha^{-1}] \cap E_0[\ell^f])$.

(b) Find $\alpha \in J_1 \cdots J_i$ such that $q_{J_1 \cdots J_i}(\alpha) = T^2$ by using the KLPT algorithm.
(c) Let $J'_i = \chi_{J_1 \cdots J_i}(\alpha)$.
(d) Compute isogenies ψ_1 with kernel $E_0[J'_i] \cap E_0[T]$ and ψ_2 with kernel $\varphi_{J_i} \circ \cdots \circ \varphi_{J_1}(E_0[\alpha] \cap E_0[T])$.
(e) Obtain $\varphi_{J'_i} = \hat{\psi}_2 \circ \psi_1$.

The following diagram illustrates the above algorithm:



Note that $\alpha J_i \alpha^{-1}$ in Step (4.a) is a left $\mathcal{O}_{\mathrm{R}}(J'_{i-1})$-ideal corresponding to J_i. We also note that Step (4.d) and (4.e) is based on the fact that $\alpha = \hat{\varphi}_{J'_i} \circ \varphi_{J_i} \circ \cdots \circ \varphi_{J_1}$ up to post-composition of an automorphism of E_0. The algorithm by [16] is a further elaboration of the above idea. See [16, §8.1] for more details.

IdealToIsogenyEichler. The restriction on p in the above algorithm was relaxed by [18]. In particular, the lower bound on T in the restriction was improved to $p^{1.25}$. They use an endomorphism of each E_i instead of the isogeny $\varphi_{J'_i}$. The rough idea of their algorithm is as follows:

We use the same notation as above and consider the computation of $\varphi_{J_{i+1}}$ from E_i and φ_{J_i}. Let $\mathcal{O}$ be an order isomorphic to $\mathrm{End}(E_i)$ and $\beta \in \mathcal{O}$ such that $J_{i+1} = \mathcal{O}\beta + \mathcal{O}\ell^f$. Then we search $\theta \in \mathcal{O}$ and coprime integers C, D such that $\mathrm{n}(\theta) = T^2$ and $\beta(C + D\theta) \in \bar{J}_i$. The latter condition means that $(C + D\theta)(P)$ generates $E_i[J_{i+1}]$ for a generator P of $\ker \hat{\varphi}_{J_i}$. From this, we can compute $\varphi_{J_{i+1}}$ by using Vélu's formulas.

Algorithm 3 in [18] shows how to find θ and C, D. This algorithm succeeds if $\mathrm{n}(\theta) > p^{2.5}$, thus we can take $T > p^{1.25}$.

We call this algorithm IdealToIsogenyEichler. The SQIsign uses this algorithm as an ideal-to-isogeny algorithm.

Algorithm in DeuringVRF. DeuringVRF is the family of verifiable random functions proposed by [34]. To construct this scheme, an ideal-to-isogeny algorithm was proposed. This algorithm can be seen as an extension of IdealToIsogenyEichler using isogenies of dimension 2. In this algorithm, the image of P under the endomorphism θ in the explanation above is computed by using a 2-dimensional isogeny $E_i \times E_i \to E_i \times E_i$. See Algorithm 7 in [34] for more details.

Algorithm in the key generation in SILBE. An ideal-to-isogeny algorithm using higher-dimension isogenies was proposed by [22], which is used in the key generation in an updatable public key encryption scheme SILBE. This algorithm is based on the similar idea as in the algorithm in the original SQIsign. To compute the auxiliary isogenies $\varphi_{J'_i}$, it uses an extension of Kani's reducibility theorem to dimension 4 by [42]. See [22, §3.1] for more details.

2.3 SQIsign

SQIsign is a signature scheme proposed by [16], which uses the generalized KLPT algorithm and an ideal-to-isogeny algorithm as building blocks.

Overview. Let $\mathcal{O}_0$ be a special extremal order and E_0 a supersingular elliptic curve over $\mathbb{F}_{p^2}$ whose endomorphism ring is isomorphic to $\mathcal{O}_0$. We consider the following zero-knowledge proof.

The public parameters are p, $\mathcal{O}_0$, and E_0. The protocol proves the knowledge of a secret left $\mathcal{O}_0$-ideal I_{sec} with a public key E_{pub}, which is the codomain of an isogeny φ_{sec} corresponding to I_{sec}, The protocol is as follows:

1. The prover computes an isogeny $\varphi_{\mathrm{com}} : E_0 \to E_{\mathrm{com}}$ and sends E_{com} as a commitment to the verifier.
2. The verifier computes an isogeny $\varphi_{\mathrm{chall}} : E_{\mathrm{com}} \to E_{\mathrm{chall}}$ and sends φ_{chall} and E_{chall} as a challenge to the prover.
3. The prover computes ideals I_{com} corresponding to φ_{com} and I_{chall} corresponding to φ_{chall}.
4. The prover applies the generalized KLPT algorithm to I_{sec} and $\bar{I}_{\mathrm{sec}} I_{\mathrm{com}} I_{\mathrm{chall}}$ and computes an ideal $I_{\mathrm{res}} \sim \bar{I}_{\mathrm{sec}} I_{\mathrm{com}} I_{\mathrm{chall}}$.
5. The prover computes φ_{res} corresponding to I_{res} and sends it to the verifier.
6. The verifier check that φ_{res} is an isogeny from E_{pub} to E_{chall} and the kernel of $\hat{\varphi}_{\mathrm{chall}} \circ \varphi_{\mathrm{res}}$ is cyclic.

The following diagram illustrates the above protocol, where the dotted arrows represent the isogenies kept secret by the prover:

$$\begin{array}{ccc} E_0 & \xrightarrow{\varphi_{\mathrm{com}}} & E_{\mathrm{com}} \\ {\scriptstyle \varphi_{\mathrm{sec}}}\downarrow & & \downarrow{\scriptstyle \varphi_{\mathrm{chall}}} \\ E_{\mathrm{pub}} & \xrightarrow[\varphi_{\mathrm{res}}]{} & E_{\mathrm{chall}} \end{array}$$

SQIsign is a signature scheme obtained by applying the Fiat-Shamir transform [25] to the above protocol.

In the following, we describe the each step of the SQIsign in more detail.

Parameter. As we mentioned in Section 2.2, the SQIsign uses IdealToIsogenyEichler as an ideal-to-isogeny algorithm. For efficiency, we use 2 as ℓ. Therefore, we use p such that $p^2 - 1$ is divisible by 2^f and T for an odd smooth integer T greater than $p^{1.25}$.

Let λ be a security parameter. To achieve a λ-bit security level, we require the following conditions:

- $p \approx 2^{2\lambda}$ (to address the attacks by [19] and [23]),
- $\deg \varphi_{\mathrm{com}} \approx 2^{2\lambda}$ (to address the meet-in-the-middle attack [30, §5.2]),
- $\deg \varphi_{\mathrm{chall}} \approx 2^{\lambda}$ (to ensure the challenge space of size 2^{λ}).

In the SQIsign, the prime p satisfies that T is divisible by 3^g such that $2^f 3^g \approx 2^{2\lambda}$ and $T/3^g$ is prime to 3 and greater than 2^{λ}. And we set $\deg \varphi_{\mathrm{com}} = T/3^g$ and $\deg \varphi_{\mathrm{chall}} = 2^f 3^g$.

Key generation. The key generation algorithm is as follows:

1. Sample a prime $D_{\mathrm{sec}} < p^{1/4}$ such that $D_{\mathrm{sec}} \equiv 3 \pmod 4$ uniformly at random.
2. Sample a left $\mathcal{O}_0$-ideal I_{sec} of norm D_{sec} uniformly at random.
3. Compute J be a left $\mathcal{O}_0$-ideal such that $J \sim I_{\mathrm{sec}}$ and $\mathrm{n}(J)$ is a power of 2 by the KLPT algorithm.
4. Compute the codomain E_{pub} of an isogeny φ_{sec} corresponding to I_{sec} by IdealToIsogenyEichler.
5. Output a public key E_{pub} and a secret key I_{sec}.

The reason that we take $D_{\mathrm{sec}} < p^{1/4}$ is to reduce the norm of the output of the generalized KLPT algorithm. The bound $p^{1/4}$ is the minimum to make the size of the secret key space larger than 2^{λ}.

Commitment. A commitment is computed by using Vélu's formulas. Taking a point K of order $T/3^g$ on E_0, we compute an isogeny φ_{com} with kernel $\langle K \rangle$. Then we output the codomain E_{com} of φ_{com} as a commitment.

Challenge. A challenge c is sampled from the integers in $[0, 2^f 3^g)$. Then the corresponding isogeny φ_{chall} is computed by using Vélu's formulas from the kernel $\langle P_{E_{\mathrm{com}}} + cQ_{E_{\mathrm{com}}} \rangle$, where $(P_{E_{\mathrm{com}}}, Q_{E_{\mathrm{com}}})$ is a deterministic basis of $E_{\mathrm{com}}[2^f 3^g]$.

Response. Let $\mathcal{O}$ be the right order of I_{sec}. Since $\mathrm{n}(I_{\mathrm{sec}}) < p^{1/4}$, we can find a left $\mathcal{O}$-ideal of norm approximately $p^{3.75}$ equivalent to a given left $\mathcal{O}_0$-ideal by the generalized KLPT algorithm. Let k be an integer such that $2^{kf} \approx p^{3.75}$. The degree of φ_{res} is set to 2^{kf}.

Given $E_{\mathrm{pub}}, E_{\mathrm{com}}, I_{\mathrm{sec}}, \varphi_{\mathrm{com}}, \varphi_{\mathrm{chall}}$, the corresponding response is computed as follows:

1. Compute the left $\mathcal{O}_0$-ideal I_{com} corresponding to φ_{com}.

2. Compute the left $\mathcal{O}_0$-ideal I'_{chall} corresponding to an isogeny with kernel $\hat{\varphi}_{\mathrm{com}}(P_{E_{\mathrm{com}}} + cQ_{E_{\mathrm{com}}})$.
3. Compute a left $\mathcal{O}_{\mathrm{R}}(I_{\mathrm{sec}})$-ideal I_{res} such that $I_{\mathrm{res}} \sim \bar{I}_{\mathrm{sec}}(I_{\mathrm{com}} \cap I'_{\mathrm{chall}})$ and $\mathrm{n}(I_{\mathrm{res}}) = 2^{kf}$ by the generalized KLPT algorithm.
4. Compute an isogeny $\varphi_{\mathrm{res}} = \varphi_{\mathrm{res},k} \circ \cdots \circ \varphi_{\mathrm{res},1}$ corresponding to I_{res} by $\mathsf{IdealToIsogenyEichler}$, where $\deg \varphi_{\mathrm{res},i} = 2^f$ for $i = 1, \ldots, k$.
5. Output the sequence of generators of $\ker \varphi_{\mathrm{res},1}, \ldots, \ker \varphi_{\mathrm{res},k}$ as a response.

Verification. The verification checks the following:

1. The codomain of the composition $\varphi_{\mathrm{res},k} \circ \cdots \circ \varphi_{\mathrm{res},1}$ is isomorphic to E_{chall}.
2. The kernel of $\hat{\varphi}_{\mathrm{chall}} \circ \varphi_{\mathrm{res},k} \circ \cdots \circ \varphi_{\mathrm{res},1}$ is cyclic.

Compression. To reduce the size of the response, a generator of the kernel of $\varphi_{\mathrm{res},i}$ is represented by coefficients of the linear combination of a deterministic basis of the 2^f-torsion subgroup of the domain. The detail is as follows: Let E_i be the domain of $\varphi_{\mathrm{res},i}$ and (P_{E_i}, Q_{E_i}) be a basis of $E_i[2^f]$ that is computed deterministically. Then $\ker \varphi_{\mathrm{res},i}$ is generated by a point of the form $aP_{E_i} + Q_{E_i}$ or $P_{E_i} + aQ_{E_i}$ for an integer $a \in [0, 2^f)$. Therefore, we can represent a generator of $\ker \varphi_{\mathrm{res},i}$ by the integer a and a bit indicating the form of the generator.

Since the SQIsign is the signature scheme obtained by applying the Fiat-Shamir transform to the above protocol, a signature of the SQIsign is a pair of a commitment and a response. In the SQIsign, a commitment is compressed by generators of the kernels of φ_{chall} and its dual isogeny and these generators are compressed by the above method. For details, see §3.4 and §3.5 in the document in [8].

3 New algorithms

In this section, we give new ideal-to-isogeny algorithms. Our algorithms are based on the same idea as in the algorithm in the original SQIsign and the algorithm in the key generation in SILBE explained in Section 2.2. In particular, we use isogenies of dimension 2 instead of T-isogenies or isogenies of dimension 4.

To use isogenies of dimension 2, we use an embedding of an imaginary quadratic order into the endomorphism ring of the domain curve. Unlike the algorithm in SILBE, our algorithms allow that the degrees of isogenies of dimension 1 and 2 have the same prime factors. This flexibility enables the use of 2-isogenies and $(2, 2)$-isogenies, thereby enhancing algorithmic efficiency.

3.1 Setting

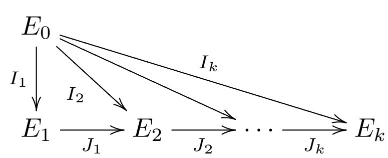
Let m_1 and m_2 be smooth integers such that $p = m_1 m_2 f - 1$ is a prime number for a small positive integer f. We assume that $m_2 > \sqrt{p}$. Note that we do not require m_1 and m_2 to be coprime. Let $\mathcal{O}_0$ be a special extremal order in

$\mathcal{B}_{p,\infty}$, R be the integer ring of $\mathbb{Q}(\mathbf{i})$ contained in $\mathcal{O}_0$, and E_0 be a supersingular elliptic curve over $\mathbb{F}_{p^2}$ whose endomorphism ring is isomorphic to $\mathcal{O}_0$. We fix an isomorphism $\iota : \mathcal{O}_0 \to \mathrm{End}(E_0)$ and identify $\mathcal{O}_0$ with $\mathrm{End}(E_0)$ by ι. We also fix a basis (P_0, Q_0) of $E_0[m_1 m_2]$. We assume that these parameters are implicitly given as input for the algorithms in this section.

Let I_1 be a left $\mathcal{O}_0$-ideal such that $\mathrm{n}(I_1)$ is prime to $m_1 m_2$, $\mathcal{O}$ be the right order of I_1, and E be the codomain of φ_{I_1}. Suppose that we know the images $\varphi_{I_1}(P_0)$ and $\varphi_{I_1}(Q_0)$. Given a left $\mathcal{O}$-ideal J, we consider an algorithm to compute the codomain of φ_J. Here, the correspondence between J and φ_J is given by the isomorphism induced by (1). By using generalized KLPT algorithm, we can assume that $\mathrm{n}(J)$ is a power of m_1.

3.2 Core of the algorithm

Decompose J into the product $J = J_1 \cdots J_k$ such that $\mathrm{n}(J_i) = m_1$. For each i, the J_i-torsion subgroup is rational over $\mathbb{F}_{p^2}$. Therefore, we can compute an isogeny corresponding to J_i by Vélu's formulas if we know a generator of its kernel. For J_1, we have $\ker \varphi_{J_1} = \varphi_{I_1}(E_0[I_1 J_1] \cap E_0[m_1])$ since $\varphi_{J_1} \circ \varphi_{I_0}$ corresponds to $I_1 J_1$ and $\mathrm{n}(I_1)$ is prime to m_1. Therefore, we can compute the codomain of φ_{J_1} by using $\varphi_{I_1}(P_0)$ and $\varphi_{I_1}(Q_0)$. Similarly, we can compute the codomain of φ_{J_i} for $i > 1$ if we have an ideal I_i such that $I_i \sim I_1 J_1 \cdots J_{i-1}$ and $\mathrm{n}(I_i)$ is prime to m_1, and the images of a basis of $E_0[m_1]$ under φ_{I_i}. The following diagram shows the relationship between $I_1, J_1, \ldots, J_k$ and the isogenies corresponding to them.

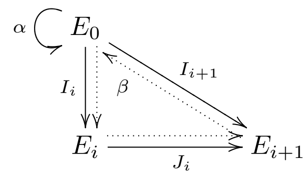


Therefore, our task is to compute the codomain E_i of φ_{J_i}, the ideal I_i, and the images of a basis of $E_0[m_1]$ under φ_{I_i} for $i > 1$. More precisely, our algorithm takes E_i, I_i, J_i, $\varphi_{I_i}(P_0)$, and $\varphi_{I_i}(Q_0)$ as input and computes E_{i+1}, I_{i+1}, $\varphi_{I_{i+1}}(P_0)$, and $\varphi_{I_{i+1}}(Q_0)$ as output.

To compute the images $\varphi_{I_i}(P_0)$ and $\varphi_{I_i}(Q_0)$, we use EvalByKani. The outline of our algorithm is as follows:

1. Compute E_{i+1}, $\varphi_{J_i} \circ \varphi_{I_i}(P_0)$, and $\varphi_{J_i} \circ \varphi_{I_i}(Q_0)$ by using Vélu's formulas with input E_i and a generator of $\ker \varphi_{J_i} = \varphi_{I_i}(E_0[I_i J_i])$.
2. Find $\beta \in I_i J_i$ and $\alpha \in R$ such that $q_{I_i J_i}(\beta)$ is prime to $m_1 m_2$ and $\mathrm{n}(\alpha) + q_{I_i J_i}(\beta) = m_2$. Let I_{i+1} be $\chi_{I_i J_i}(\beta)$.
3. Compute $\varphi_{I_{i+1}}(m_1 P_0)$ and $\varphi_{I_{i+1}}(m_1 Q_0)$ by $m_1 \varphi_{I_{i+1}} = \frac{1}{\mathrm{n}(I_i)} \varphi_{J_i} \circ \varphi_{I_i} \circ \hat{\beta}$.
4. Compute the image of a basis of $E_{i+1}[m_1 m_2]$ under $\hat{\varphi}_{I_{i+1}}$ by EvalByKani with input $m_2, E_0, E_{i+1} E_0, \varphi_{I_{i+1}}(m_1 P_0), \varphi_{I_{i+1}}(m_1 Q_0), \alpha(m_1 P_0), \alpha(m_1 Q_0)$.
5. Compute $\varphi_{I_{i+1}}(P_0)$ and $\varphi_{I_{i+1}}(Q_0)$ by solving a discrete logarithm problem in $E_0[m_1 m_2]$.

The following diagram shows the relationship between the ideals and the isogenies in the above algorithm.



We give supplementary explanations for the above steps except for Step 1.

Step 2. As discussed in [32, §3.1], we can find elements in I_iJ_i whose normalized norms are appropriately $\sqrt{p}$ by using lattice enumeration (e.g., see [10, Algorithm 2.7.5]) for many I_iJ_i. However, there exist exceptional cases that we will discuss later. Once we find many elements in I_iJ_i whose normalized norms are appropriately $\sqrt{p}$, we can find β such that $m_2 - q_{I_iJ_i}(\beta)$ is a prime number splitting in R. Then we can find α by Cornacchia's algorithm [10, Algorithm 1.5.2, 1.5.3]. This method is also used in SQIsignHD and SILBE for the case $R = \mathbb{Z}[\sqrt{-1}]$, i.e., $m_2 - q_{I_iJ_i}(\beta)$ is the sum of two squares. However, these schemes do not use an endomorphism of E_0, instead they use an endomorphism of the abelian surface E_0^2.

Step 3. Since $\beta = \hat{\varphi}_{I_{i+1}} \circ \varphi_{J_i} \circ \varphi_{I_i}$, we have $\varphi_{J_i} \circ \varphi_{I_i} \circ \hat{\beta} = m_1 \mathrm{n}(I_i)\varphi_{I_{i+1}}$. Therefore, we can compute $\varphi_{I_{i+1}}(m_1P_0)$ by $\frac{1}{\mathrm{n}(I_i)}\varphi_{J_i} \circ \varphi_{I_i} \circ \hat{\beta}(P_0)$ and $\varphi_{I_{i+1}}(m_1Q_0)$ similarly, where $\frac{1}{\mathrm{n}(I_i)}$ is the inverse of $\mathrm{n}(I_i)$ modulo m_2.

Step 4. We need to care about the fact that the output of EvalByKani could not be $\hat{\varphi}_{I_{i+1}}$ but $\iota_0 \circ \hat{\varphi}_{I_{i+1}}$ for some automorphism ι_0 of E_0. If the automorphism groups of E_0 and E_{i+1} are $\{\pm 1\}$, then this does not cause any problem. This is because $-\varphi_{I_{i+1}}$ is also an isogeny corresponding to I_{i+1}. However, if ι_0 is an isomorphism not in $\{\pm 1\}$, then the dual isogeny of $\iota_0 \circ \hat{\varphi}_{I_{i+1}}$ does not correspond to I_{i+1}. This occurs when $j(E_0) = 0$ or 1728. Therefore, we need to fix the post-composition by ι_0. To do this, we evaluate $\varphi_{J_i} \circ \varphi_{I_i}(P_0)$ in addition to a basis of $E_{i+1}[m_1m_2]$ by EvalByKani. By comparing the output with $\beta(P_0)$, we can determine ι_0.

We also note that we want the codomain of the (m_2, m_2)-isogeny in EvalByKani in Step 4 to not be isomorphic to $E_0 \times E_0$. As discussed in [38, §2.4], this only occurs with a negligible probability for a cryptographic size of p. In addition, we can prove the following lemma.

Lemma 1. *We use the notation in the above setting. Suppose that $E_0 \not\cong E_{i+1}$ and any $\gamma \in \mathrm{End}(E_0)$ whose norm is less than m_2 is in R. Then the codomain of the (m_2, m_2)-isogeny in Step 4 is not isomorphic to $E_0 \times E_0$.*

Proof. Suppose that the codomain of the (m_2, m_2)-isogeny is isomorphic to $E_0 \times E_0$. Then we have the following commutative diagram:

$$\begin{CD} E_0 @>{\varphi_{I_{i+1}}}>> E_{i_1} \\ @V{\alpha}VV @VVV \\ E_0 @>>{\gamma}> E_0, \end{CD}$$

where $\mathrm{n}(\gamma) = \mathrm{n}(I_{i+1})$ and the right vertical arrow is an isogeny of degree $\mathrm{n}(\alpha)$. Since α commutes with γ and $\mathrm{n}(\alpha)$ is prime to $\mathrm{n}(I_{i+1})$ we have $\varphi_{I_{i+1}} = \gamma$ up to post-composition of an isomorphism. This contradicts $E_0 \not\cong E_{i+1}$. □

The latter assumption in the lemma is satisfied if we take $m_2 \approx \sqrt{p}$ because an element in $\mathcal{O}_0 \setminus R$ has a norm approximately greater than p. If the first assumption is not satisfied, we can compute the image under $\varphi_{I_{i+1}}$ directly since this is an endomorphism of E_0.

Step 5. Let (P_{i+1}, Q_{i+1}) be a basis of $E_{i+1}[m_1 m_2]$. By Step 4, we know $\hat{\varphi}_{I_i}(P_{i+1})$ and $\hat{\varphi}_{I_i}(Q_{i+1})$. Solving the discrete logarithm problems for these points with base P_0 and Q_0, we have integers a, b, c, d such that

$$\hat{\varphi}_{I_i} \begin{pmatrix} P_{i+1} \\ Q_{i+1} \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} P_0 \\ Q_0 \end{pmatrix}.$$

Acting $\varphi_{I_{i+1}}$ and multiplying the inverse of $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ modulo $m_1 m_2$ to the above equation, we have

$$\mathrm{n}(I_{i+1}) \begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} \begin{pmatrix} P_{i+1} \\ Q_{i+1} \end{pmatrix} = \begin{pmatrix} \varphi_{I_{i+1}}(P_0) \\ \varphi_{I_{i+1}}(Q_0) \end{pmatrix}.$$

3.3 Exceptional cases

In this subsection, we discuss exceptional cases in Step 2 in the previous subsection. In particular, we consider what kind of ideals $I_i J_i$ fail to find β and α, and how to avoid them. To ease the notation, we denote $I_i J_i$ by I and E_{i+1} by E.

Exceptional ideals. Step 2 fails if the normalized norm of the smallest element in I is much smaller than $\sqrt{p}$ and not prime to $m_1 m_2$. Consider this case and let $(\beta_1, \beta_2, \beta_3, \beta_4)$ be the Minkowski-reduced basis of I. Then we have (see [32, §3.1])

$$p^2 \leq q_I(\beta_1) q_I(\beta_2) q_I(\beta_3) q_I(\beta_4) \leq 4p^2.$$

Since $q_I(\beta_1) \ll \sqrt{p}$ and $\mathrm{n}(\mathbf{i})$ is small, we have $\beta_2 = \gamma\beta_1$, where γ is the smallest element in $R \setminus \mathbb{Z}$. Therefore, the elements in I whose normalized norms is smaller than $\sqrt{p}$ are of the form $\delta\beta_1$ for $\delta \in R$. The normalized norms of these elements are divisible by $q_I(\beta_1)$, so not prime to m_1m_2.

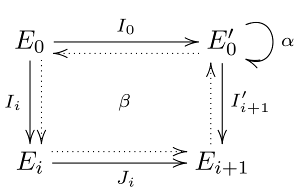
In terms of elliptic curves, the above exceptional case occurs when there exists an isogeny $E_0 \to E$ whose degree is much smaller than $\sqrt{p}$ and not prime to m_1m_2. Let n be a positive integer smaller than $\sqrt{p}$. Then the number of isogenies from E_0 whose degree is smaller than n is approximately linear in n^2. Therefore, we can assume that the probability that I has an element whose normalized norm is smaller than n is approximately n^2/p. This probability is small but not negligible in practice. Especially, it occurs with a high probability in the case that J is a left $\mathcal{O}_0$-ideal, i.e., $I_1 = \mathcal{O}_0$ and $E_1 = E_0$. In this case, there exists the isogeny $\varphi_{J_1} : E_0 \to E_2$ of degree $m_1 < \sqrt{p}$. Therefore, we need to avoid the exceptional cases.

Avoiding exceptional cases. To avoid the exceptional cases, we use other supersingular elliptic curves whose endomorphism rings contain an imaginary quadratic order with a small discriminant. A similar idea is used in the SQIsign. See §2.5.2 in the document in [8] for the details.

To explain our method, we define the following term.

Definition 1 (connecting tuple). *For a positive integer N, an $(E_0; P_0, Q_0)$-connecting tuple is a tuple $(E_0', I_0, \mathfrak{O}, P_0', Q_0', \sigma(P_0'), \sigma(Q_0'))$, where E_0' is a supersingular elliptic curve over $\mathbb{F}_{p^2}$, I_0 is a left $\mathcal{O}_0$-ideal such that $\mathcal{O}_{\mathrm{R}}(I_0) \cong \mathrm{End}(E_0')$ and $\mathrm{n}(I_0)$ is prime to the order of P_0 and Q_0, $\mathfrak{O}$ is an imaginary quadratic order contained in $\mathrm{End}(E_0')$, P_0' and Q_0' are the images of P_0 and Q_0 under φ_{I_0}, and σ is an element in $\mathfrak{O}$ such that $\mathfrak{O} = \mathbb{Z}[\sigma]$.*

Suppose that we are given an $(E_0; P_0, Q_0)$-connecting tuple $(E_0', I_0, \mathfrak{O}, P_0', Q_0', \sigma(P_0'), \sigma(Q_0'))$. Instead of computing an isogeny between E_0 and E_{i+1}, we compute an isogeny between E_0' and E_{i+1}. The following diagram shows the relationship between ideals and isogenies in this case.



In summary, we execute the following steps instead of Step 2-5 in the previous subsection:

2' Find $\beta \in \bar{I}_0 I_i J_i$ and $\alpha \in \mathfrak{O}$ such that $q_{\bar{I}_0 I_i J_i}(\beta)$ is prime to m_1m_2 and $\mathrm{n}(\alpha) + q_{\bar{I}_0 I_i J_i}(\beta) = m_2$. Let I_{i+1}' be $\chi_{\bar{I}_0 I_i J_i}(\beta)$.
3' Compute $\varphi_{I_{i+1}'}(m_1P_0')$ and $\varphi_{I_{i+1}'}(m_1Q_0')$ by $m_1\varphi_{I_{i+1}'} = \frac{1}{\mathrm{n}(I_i)\mathrm{n}(I_0)}\varphi_{J_i} \circ \varphi_{I_i} \circ \hat{\varphi}_{I_0} \circ \hat{\beta}$.
4' Compute the image of a basis of $E_{i+1}[m_1m_2]$ under $\hat{\varphi}_{I_{i+1}}$ by EvalByKani with input $m_2, E_0', E_{i+1}E_0', \varphi_{I_{i+1}'}(m_1P_0'), \varphi_{I_{i+1}'}(m_1Q_0'), \alpha(m_1P_0'), \alpha(m_1Q_0')$.

5' Compute $\varphi_{I'_{i+1}}(P'_0)$ and $\varphi_{I'_{i+1}}(Q'_0)$ by solving a discrete logarithm problem in $E'_0[m_1m_2]$.

Consequently, we obtain a left $\mathcal{O}_0$-ideal $I_0I'_{i+1}$ whose norm is prime to m_1m_2 and the images of P_0 and Q_0 under $\varphi_{I_0I'_{i+1}} = \varphi_{I'_{i+1}} \circ \varphi_{I_0}$.

3.4 Explicit algorithms

In this subsection, we give explicit ideal-to-isogeny algorithms. The first is an algorithm to compute an isogeny corresponding to a left $\mathcal{O}_0$-ideal of norm m_1, which is explained in the previous subsections. The second is an algorithm to compute an isogeny from an ideal by using the first algorithm repeatedly.

Let S_{ct} be a finite ordered set of $(E_0; P_0, Q_0)$-connecting tuples with different elliptic curves. We set the first entry of S_{ct} to be the trivial connecting tuple $(E_0, \mathcal{O}_0, R, P_0, Q_0, \sigma(P_0), \sigma(Q_0))$, where σ is a generator of R. We assume that S_{ct} is implicitly given as input for the algorithms in this section. A method to compute connecting tuples is given in Appendix A.

The first algorithm is given in Algorithm 2. We name this algorithm ShortIdealToIsogenyIQO. The second algorithm, IdealToIsogenyIQO, is given in Algorithm 3. This algorithm computes an isogeny corresponding to an ideal of norm m_1^k.

4 New algorithm for SQIsign

As an application of our algorithms, we propose a new algorithm for SQIsign. Our algorithm for SQIsign uses IdealToIsogenyIQO instead of IdealToIsogenyEichler. This change replaces the computation of isogenies of higher degrees between elliptic curves with the computation of 2-isogenies and $(2,2)$-isogenies and is expected to reduce the computational cost of SQIsign, but not affect its security and the size of public keys and signatures.

4.1 Setting

For the efficiency of our algorithm, we use powers of 2 as m_1 and m_2 and the theta algorithm by [15] for $(2,2)$-isogenies. The theta algorithm requires points of order 8 instead of 2 to compute $(2,2)$-isogenies. Therefore, we use p of the form $2^{f_1+f_2+2}g-1$ for small odd integer$g$ and compute $(2^{f_2}, 2^{f_2})$-isogenies from points of order 2^{f_2+2}.

In the following, we use elliptic curves over $\mathbb{F}_{p^2}$ which are uniquely chosen from their isomorphic classes, e.g., "normalized Montgomery curves" obtained by [8, Algorithm 1]. For such a curve E, we fix a basis of $E[2^{f_1+f_2+2}]$ and denote it by (P_E, Q_E).

Let E_0 be a supersingular elliptic curve over $\mathbb{F}_{p^2}$ of j-invariant 1728 and $\iota : \mathcal{O}_0 \coloneqq \left\langle 1, \mathbf{i}, \frac{\mathbf{i}+\mathbf{j}}{2}, \frac{1+\mathbf{k}}{2} \right\rangle_{\mathbb{Z}} \to \mathrm{End}(E_0)$ be an isomorphism. Let λ be a security parameter. For λ-bit security, our system parameters are $p \approx 2^{2\lambda}$ of the above form, E_0, ι, and a finite ordered set S_{ct} of $(E_0; P_{E_0}, Q_{E_0})$-connecting tuples with the first entry being the trivial connecting tuple $(E_0, \mathcal{O}_0, \mathbb{Z}[\mathbf{i}], P_{E_0}, Q_{E_0}, \mathbf{i}(P_{E_0}), \mathbf{i}(Q_{E_0}))$.

Algorithm 2: ShortIdealToIsogenyIQO

Input: a supersingular elliptic curve E, a left $\mathcal{O}_0$-ideal I whose norm is prime to m_1m_2, a left $\mathcal{O}_{\mathrm{R}}(I)$-ideal J of norm m_1, and $\varphi_I(P_0), \varphi_I(Q_0)$.

Output: The codomain of the isogeny φ_J, $\beta \in IJ$ such that $q_{IJ}(\beta)$ is prime to m_1m_2, and $\varphi_{\chi_{IJ}(\beta)}(P_0), \varphi_{\chi_{IJ}(\beta)}(Q_0)$.

Let K be a generator of $\varphi_I(E_0[m] \cap E_0[IJ])$;
Compute an isogeny $\varphi_J : E \to E'$ with kernel $\langle K \rangle$;
Let S be $\{P', Q'\}$, where P', Q' a basis of $E'[m_1m_2]$;
for $(E'_0, I_0, \mathfrak{O}, P'_0, Q'_0, \sigma(P'_0), \sigma(Q'_0)) \in S_{\mathrm{ct}}$ **do**
  Search $\alpha \in \mathfrak{O}$ and $\beta \in \bar{I}_0IJ$ such that $q_{\bar{I}_0IJ}(\beta)$ is prime to m_1m_2 and $\mathrm{n}(\alpha) + q_{\bar{I}_0IJ}(\beta) = m_2$;
  if *α and β are found* **then**
    Let $I' = \chi_{\bar{I}_0IJ}(\beta)$;
    Let $P_1 = \varphi_{I'}(m_1P'_0)$ and $Q_1 = \varphi_{I'}(m_1Q'_0)$;
    Let $P_2 = \alpha(m_1P'_0)$ and $Q_2 = \alpha(m_1Q'_0)$;
    if $j(E'_0) = 0$ *or* 1728 **then**
      Append $\varphi_J \circ \varphi_I \circ \hat{\varphi}_{I_0}(P'_0)$ to S;
      Let $P'' = \beta(P'_0)$;
    break;
Let $S' = \mathsf{EvalByKani}(m_2, E'_0, E', E'_0, P_1, Q_1, P_2, Q_2, S)$;
if $\#S = 3$ **then**
  Let P''' be the third element of S';
  Compute $\iota_0 \in \mathrm{Aut}(E'_0)$ such that $\iota_0(P''') = P''$;
  Let $S' = \iota_0(S')$;
Let P, Q be the first and second elements of S';
Find $a, b, c, d \in \mathbb{Z}$ such that $P = aP'_0 + bQ'_0$ and $Q = cP'_0 + dQ'_0$;
Let M be the inverse of $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ modulo m_1m_2;
Let $\begin{pmatrix} P \\ Q \end{pmatrix} = \mathrm{n}(I')M \begin{pmatrix} P' \\ Q' \end{pmatrix}$;
return E', β, P, Q;

4.2 Key generation

Our key generation algorithm is almost the same as the SQIsign. In particular, it is as follows:

1-3. The same as the key generation in Section 2.3.
4. Compute the codomain E_{pub} of $\varphi_{I_{\mathrm{sec}}}$ and the image of P_{E_0} and Q_{E_0} under $\varphi_{I_{\mathrm{sec}}}$ by IdealToIsogenyIQO.
5. Output a public key is E_{pub} and a secret key is $(I_{\mathrm{sec}}, \varphi_{I_{\mathrm{sec}}}(P_{E_0}), \varphi_{I_{\mathrm{sec}}}(Q_{E_0}))$.

The product of the ideals in the input of the final call of ShortIdealToIsogenyIQO in IdealToIsogenyIQO in Step 4 above is equivalent to I_{sec}. Therefore, there exists an element β in this product whose normalized norm is prime to D_{sec} and we can compute β by the quaternions obtained by the calls of ShortIdealToIsogenyIQO

Algorithm 3: IdealToIsogenyIQO

Input: a supersingular elliptic curve E, a left $\mathcal{O}_0$-ideal I whose norm is prime to $m_1 m_2$, a left $\mathcal{O}_{\mathrm{R}}(I)$-ideal J of norm m_1^k, and $\varphi_I(P_0), \varphi_I(Q_0)$.

Output: The codomain of the isogeny φ_J, a left $\mathcal{O}_0$-ideal I' such that $I' \sim IJ$ and $\mathrm{n}(I')$ is prime to $m_1 m_2$, and $\varphi_{I'}(P_0), \varphi_{I'}(Q_0)$.

```
Let P = φ_I(P_0), Q = φ_I(Q_0);
Let J_1, ..., J_k be ideals such that n(J_i) = m_1 and J = J_1 ··· J_k;
for i = 1, ..., k do
    Let E, β, P, Q = ShortIdealToIsogenyIQO(E, I, J_i, P, Q);
    Let I = χ_{IJ_i}(β);
    Let J_j = β J_j β^{-1} for j = i + 1, ..., k;
return E, I, P, Q;
```

other than the final call. Since $D_{\mathrm{sec}} < p^{1/4} \ll \sqrt{p} < 2^{f_2}$, we can find an odd integer m and $\alpha \in \mathbb{Z}[\mathbf{i}]$ such that $m^2 D_{\mathrm{sec}} + \mathrm{n}(\alpha) = 2^{f_2}$. By using $m\beta$ in the final call of ShortIdealToIsogenyIQO, we obtain the images of P_{E_0} and Q_{E_0} under $m\varphi_{I_{\mathrm{sec}}}$. By dividing the images by m modulo $2^{f_1+f_2+2}$, we obtain the images of P_{E_0} and Q_{E_0} under $\varphi_{I_{\mathrm{sec}}}$.

4.3 Commitment

To pull back a challenge isogeny to E_0, the degree of a commitment isogeny must be prime to the degree of the challenge isogeny. Since the degree of a challenge isogeny must be a power of 2 in our setting, the degree of a commitment isogeny must be odd. To achieve this, we use IdealToIsogenyIQO for commitments. Our commitment algorithm is as follows:

1. Sample a left $\mathcal{O}_0$-ideal J_{com} of norm $2^{2\lambda}$ uniformly at random.
2. Compute the codomain E_{com} of $\varphi_{J_{\mathrm{com}}}$, a left $\mathcal{O}_0$-ideal I_{com} such that $I_{\mathrm{com}} \sim J_{\mathrm{com}}$ and $\mathrm{n}(I_{\mathrm{com}})$ is odd, and the image of P_{E_0} and Q_{E_0} under $\varphi_{I_{\mathrm{com}}}$ by IdealToIsogenyIQO.
3. Output a commitment E_{com} and commitment key $(I_{\mathrm{com}}, \varphi_{I_{\mathrm{com}}}(P_{E_0}), \varphi_{I_{\mathrm{com}}}(Q_{E_0}))$.

4.4 Challenge

Our challenge algorithm is almost the same as the SQIsign, but we use only 2-isogenies. Consequently, a challenge is an integer c in $[0, 2^{\lambda})$. This challenge corresponds to an isogeny with kernel $\langle P_{E_{\mathrm{com}}} + cQ_{E_{\mathrm{com}}} \rangle$.

4.5 Response and verification

In the SQIsign, a response is computed by dividing into 2^f-isogenies and represented by the sequence of their kernels. In our algorithm, a response is computed by dividing into 2^{f_1}-isogenies and represented by the sequence of the kernels of

Algorithm 4: Response

Input: a public key E_{pub}, a secret key $(I_{\text{sec}}, \varphi_{I_{\text{sec}}}(P_{E_0}), \varphi_{I_{\text{sec}}}(Q_{E_0}))$, a commitment E_{com}, a commitment key $(I_{\text{com}}, \varphi_{I_{\text{com}}}(P_{E_0}), \varphi_{I_{\text{com}}}(Q_{E_0}))$, and a challenge c.

Output: A sequence of the kernels of isogenies of degree 2^{2f_1}.

```
Compute the left O_0-ideal I'_chall corresponding to an isogeny with kernel
  ⟨φ̂_{I_com}(P_{E_com} + cQ_{E_com})⟩;
Let I_res = Ī_sec(I_com ∩ I'_chall);
Compute a left O_R(I_sec)-ideal J_res such that J_res ~ I_res and n(J_res) = 2^{2kf_1};
Let J_1, ..., J_k be ideals such that n(J_i) = 2^{f_1} and J_res = J_1 ··· J_k;
Let E = E_pub, I = I_sec, P = φ_{I_sec}(P_{E_0}), Q = φ_{I_sec}(Q_{E_0});
Let S_res = ∅;
for i = 1, ..., 2k − 2 do
    if i is odd then
        Append a generator of φ_I(E_0[2^{2f_1}] ∩ E_0[I J_i J_{i+1}]) to S_res;
    Let E, β, P, Q = ShortIdealToIsogenyIQO(E, I, J_i, P, Q);
    Let I = χ_{I J_i}(β);
    Let J_j = β J_j β^{-1} for j = i + 1, ..., 2k;
Append a generator of φ_I(E_0[2^{2f_1}] ∩ E_0[I J_{2k−1} J_{2k}]) to S_res;
return S_res;
```

the compositions of consecutive two 2^{f_1}-isogenies. More precisely, for a chain of 2^{f_1}-isogenies $\varphi_k \circ \varphi_{k-1} \circ \cdots \circ \varphi_1$, our response is represented by the sequence of the kernels of $\varphi_{2i} \circ \varphi_{2i-1}$ for $i = 1, \dots, k/2$. This is possible since $E[2^{2f_1}]$ is $\mathbb{F}_{p^2}$-rational for elliptic curves E appearing in the response. This reduces the number of computations of torsion bases in verification and improves the efficiency of verification. In addition, we can compute the final two isogenies φ_k and φ_{k-1} without ShortIdealToIsogenyIQO since we do not need to compute the image of any point under the isogenies in the response.

Our response algorithm is given in Algorithm 4. Here we assume that the norm of the ideal obtained by the generalized KLPT algorithm is 2^{2kf_1} for some integer k. Our verification algorithm is the same as the SQIsign.

4.6 Other methods

Algorihtm in QFESTA. We can use RandIsogImages by [38] instead of IdealToIsogenyIQO in the key generation and commitment algorithms. RandIsogImages takes an integer d and points in E_0 and outputs the codomain and the images of the points under a random d-isogeny. This algorithm is more efficient than IdealToIsogenyIQO because it requires only one $(2^{f_1+f_2+2}, 2^{f_1+f_2+2})$-isogeny while IdealToIsogenyIQO requires several $(2^{f_2}, 2^{f_2})$-isogenies (see Table 1 in the next section). However, the distribution of the codomain in the output of RandIsogImages is not guaranteed to be uniform in the codomains of the d-isogenies from E_0.

In summary, this alternative method offers a trade-off between efficiency and security. This is the same situation as in an alternative method for the key generation in SQIsign [17, Appendix D]. The security of this alternative method is discussed in [40]. We leave the detailed comparison between these methods in our algorithm for future work.

Algorithm in DeuringVRF. As mentioned in [34, §6], the ideal-to-isogeny algorithm in DeuringVRF could be applied to SQIsign. Indeed, we can replace ShortIdealToIsogenyIQO with the ideal-to-isogeny algorithm in DeuringVRF (Algorithm 7 in [34]) in the SQIsign algorithms explained in this section. In each call of the ideal-to-isogeny algorithm in DeuringVRF, we can computed an isogeny corresponding to an ideal of norm approximately $p^{1/2}$. This is almost the same as IdealToIsogenyIQO. On the other hand, the degree of the isogeny of dimension 2 in the ideal-to-isogeny algorithm in DeuringVRF is slightly larger than that in ShortIdealToIsogenyIQO. Therefore, simple replacement of ShortIdealToIsogeny-IQO with the ideal-to-isogeny algorithm in DeuringVRF is not expected to improve the performance of SQIsign.

5 Concrete parameters and efficiency

In this section, we propose concrete parameters for our algorithm for SQIsign and discuss the efficiency of our algorithm compared with the SQIsign.

5.1 Proposed parameters

For the NIST security level 1, 3, and 5, we proposed the following parameters:

- For the security level 1: $p = 2^{247} \cdot 79 - 1,\ f_1 = 106,\ f_2 = 139.$
- For the security level 3: $p = 2^{370} \cdot 231 - 1,\ f_1 = 156,\ f_2 = 212.$
- For the security level 5: $p = 2^{492} \cdot 539 - 1,\ f_1 = 216,\ f_2 = 274.$

These primes have almost the same size as the primes in the SQIsign corresponding to the same security levels.

5.2 Efficiency

The main difference between our algorithm and the SQIsign is that our algorithm uses IdealToIsogenyIQO instead of IdealToIsogenyEichler. Most of the computation time in IdealToIsogenyIQO is spent on the computation of $(2^{f_2}, 2^{f_2})$-isogenies. On the other hand, most of the computation time in IdealToIsogenyEichler is spent on the computation of T-isogenies. Therefore, we count the numbers of these isogenies in our algorithm and the SQIsign.

In the following, we focus on the NIST security level 1 and claim that our algorithm is more efficient than the SQIsign. At higher security levels, the advantage of our algorithm becomes more significant because the smoothness of T decreases as the security level increases.

	SQIsign # of T-isogenies	Ours # of $(2^{f_2}, 2^{f_2})$-isogenies
Key generation	16	7
Commitment	one $T/3^g$-isogeny	3
Response	28	8

Table 1. The numbers of T-isogenies and $(2^f, 2^f)$-isogenies in the key generation, commitment, and response in the SQIsign and our algorithm.

At the current implementation of the SQIsign for the NIST security level 1, the following parameters are used: $f = 75$, the norm of the output of the KLPT algorithm in the key generation is 2^{675}, and $\mathrm{n}(I_{\mathrm{rep}}) = 2^{1050}$. Since IdealToIsogenyEichler requires two T-isogenies for each 2^f-isogeny, the number of T-isogenies in the key generation is 16, and that in the response is 28. Note that the first 2^f-isogeny in the key generation does not require any T-isogenies. In addition, the commitment algorithm requires one $T/3^g$-isogeny.

On the other hand, our algorithm uses 2^{742} for the norm of the output of the KLPT algorithm in the key generation (we use $742 = 7 \cdot f_1$ for simplicity) and the same norm for the response. Consequently, our algorithm requires 7 $(2^{f_2}, 2^{f_2})$-isogenies in the key generation, 3 $(= \lceil 256/f_1 \rceil)$ $(2^{f_2}, 2^{f_2})$-isogenies in the commitment, and 8 $(= \lceil 1050/f_1 \rceil - 2)$ $(2^{f_2}, 2^{f_2})$-isogenies in the response.

Table 1 shows the numbers of isogenies in the key generation, commitment, and response in our algorithm and the SQIsign.

We estimate the costs of a T-isogeny in the SQIsign and a $(2^{f_2}, 2^{f_2})$-isogeny in our algorithm. Our estimation is based on the number of operations in $\mathbb{F}_{p^2}$. We use the number of multiplication (including squaring) in $\mathbb{F}_p$ as the measure of the cost of an operation. Appendix B gives the details of the estimation. We counted the number of multiplications by Python code.[3] In conclusion, we claim that the cost of a T-isogeny in the SQIsign is at least $141,987$ $\mathbb{F}_p$-multiplications, and the cost of a $(2^{f_2}, 2^{f_2})$-isogeny in our algorithm is approximately $147,951 \mathbb{F}_p$-multiplications.

Together with the numbers of isogenies in Table 1, we conclude that our algorithm is at least twice as fast as the SQIsign in the key generation and the signing.

Our algorithms is also more efficient in the verification than the SQIsign because the numbers of separations in the response isogeny are reduced. In the NIST security level 1, the verification in the SQIsign computes 14 2^{75}-isogenies, while our algorithm computes 4 2^{212}-isogenies and one 2^{202}-isogeny. This reduce the number of the computations of deterministic torsion bases in the verification.

[3] The code is available at https://github.com/hiroshi-onuki/SQIsignIQO.jl/blob/main/measure_costs/measure_c

5.3 Implementation

We implemented our SQIsign by Julia language [6] with its computer algebra package Nemo [26]. Our code is available at

https://github.com/hiroshi-onuki/SQIsignIQO.jl.

Table 2 shows the computational times of key generation, signing, and verification in our implementation. The computational times are measured on a computer with an Intel Core i7-10700K CPU@3.70GHz without Turbo Boost. The values are the averages of 100 runs.

	Key gen.	Sign	Verify
Level 1	1.88	3.41	0.24
Level 3	2.81	6.15	0.32
Level 5	4.73	8.84	0.50

Table 2. The computational times of key generation, signing, and verification in our implementation (sec.). The values are the averages of 100 runs.

For reference, we executed the benchmarking suite in the reference implementation of the SQISIGN on the same computer. Table 3 shows the results. The computational times are given in 10^6 cycles.

	Key gen.	Sign	Verify
Level 1	4,813	8,103	202
Level 3	34,561	63,656	1,178
Level 5	134,317	243,076	3,476

Table 3. The computational times of key generation, signing, and verification in the reference implementation of the SQISIGN (10^6 cycles.). The values are the average of 100 runs.

In the NIST security level 1, our implementation is not faster than the reference implementation of the SQISIGN. However, we believe that our algorithm outperforms the SQISIGN if we implement our algorithm in a lower-level language such as C or Rust. We leave such an implementation for future work.

In the NIST security level 3 and 5, our implementation is faster than the reference implementation of the SQISIGN. The advantage of our algorithm becomes more significant at higher security levels as we mentioned in Section 5.2.

6 Conclusion and future work

In this paper, we proposed a new ideal-to-isogeny algorithm using Kani's reducibility theorem and embeddings of imaginary quadratic orders into the endomorphism rings of supersingular elliptic curves. Our algorithm works in the operations in $\mathbb{F}_{p^2}$ if we use the characteristic p such that $p+1$ has a smooth divisor greater than $\sqrt{p}$. Especially, our algorithm is efficient if we use p of the form $2^f g - 1$ for small odd integer g.

As an application of our algorithm, we proposed a new algorithm for SQIsign. Our estimation shows that our algorithm is at least twice as fast as the SQIsign in the key generation and the signing at the NIST security level 1. Our algorithm is also more efficient in the verification because the numbers of separations in the response isogeny are reduced. Notably, at higher security levels, the benefits of our algorithm become more pronounced. This assertion is substantiated by our implementation results.

Implementing our algorithm in a lower-level language such as C or Rust and comparing the efficiency with the SQIsign are left for future work. Further improvements in the efficiency of our algorithm are also left for future work. For example, using a smooth factor of $p-1$ in the degree of the response Isogeny in addition to 2^{f_1} may improve the efficiency of our algorithm since this could reduce the number of separations in the response isogeny.

Acknowledgments. We would like to thank Antonin Leroux for informing us of his ideal-to-isogeny algorithm. We also thank Andrea Basso, Luca De Feo, Pierrick Dartois, Antonin Leroux, Luciano Maino, and Benjamin Wesolowski for sharing their work on SQIsign2D-West, and Max Duparc and Tako Boris Fouotsa for sharing their work on SQIPrime with us prior to publication.

References

1. Adj, G., Chi-Domínguez, J.J., Rodríguez-Henríquez, F.: Karatsuba-based square-root Vélu's formulas applied to two isogeny-based protocols. Journal of Cryptographic Engineering **13**(1), 89–106 (Apr 2023). https://doi.org/10.1007/s13389-022-00293-y, https://doi.org/10.1007/s13389-022-00293-y
2. Adj, G., Rodríguez-Henríquez, F.: Square root computation over even extension fields. IEEE Transactions on Computers **63**(11), 2829–2841 (2014). https://doi.org/10.1109/TC.2013.145
3. Basso, A., Feo, L.D., Dartois, P., Leroux, A., Maino, L., Pope, G., Robert, D., Wesolowski, B.: SQIsign2D-West: The fast, the small, and the safer. Cryptology ePrint Archive, Paper 2024/760 (2024), https://eprint.iacr.org/2024/760, https://eprint.iacr.org/2024/760
4. Basso, A., Maino, L., Pope, G.: FESTA: Fast encryption from supersingular torsion attacks. In: Guo, J., Steinfeld, R. (eds.) ASIACRYPT 2023, Part VII. LNCS, vol. 14444, pp. 98–126. Springer, Heidelberg (Dec 2023). https://doi.org/10.1007/978-981-99-8739-9_4
5. Bernstein, D.J., De Feo, L., Leroux, A., Smith, B.: Faster computation of isogenies of large prime degree. In: Galbraith, S. (ed.) ANTS-XIV - 14th Algorithmic Number Theory Symposium. Proceedings of the Fourteenth Algorithmic Number Theory Symposium (ANTS-XIV), vol. 4, pp. 39–55. Mathematical Sciences Publishers, Auckland, New Zealand (2020). https://doi.org/10.2140/obs.2020.4.39, https://hal.inria.fr/hal-02514201

6. Bezanson, J., Edelman, A., Karpinski, S., Shah, V.B.: Julia: A fresh approach to numerical computing. SIAM Review **59**(1), 65–98 (2017). https://doi.org/10.1137/141000671, https://epubs.siam.org/doi/10.1137/141000671
7. Castryck, W., Decru, T.: An efficient key recovery attack on SIDH. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 423–447. Springer, Heidelberg (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_15
8. Chavez-Saab, J., Santos, M.C., De Feo, L., Eriksen, J.K., Hess, B., Kohel, D., Leroux, A., Longa, P., Meyer, M., Panny, L., Patranabis, S., Petit, C., Rodríguez Henríquez, F., Schaeffler, S., Wesolowski, B.: SQIsign. Tech. rep., National Institute of Standards and Technology (2023), available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures
9. Chi-Domínguez, J.J., Rodríguez-Henríquez, F.: Optimal strategies for CSIDH (2022). https://doi.org/10.3934/amc.2020116, https://www.aimsciences.org/article/id/f80e3ff5-2e53-499b-b72e-4eed139dc4b8
10. Cohen, H.: A Course in Computational Algebraic Number Theory, Graduate Texts in Mathematics, vol. 138. Springer Berlin, Heidelberg (2010)
11. Cosset, R., Robert, D.: Computing (l, l)-isogenies in polynomial time on Jacobians of genus 2 curves. Mathematics of Computation **84**(294), 1953–1975 (2015)
12. Costello, C.: B-SIDH: Supersingular isogeny Diffie-Hellman using twisted torsion. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020, Part II. LNCS, vol. 12492, pp. 440–463. Springer, Heidelberg (Dec 2020). https://doi.org/10.1007/978-3-030-64834-3_15
13. Costello, C., Hisil, H.: A simple and compact algorithm for SIDH with arbitrary degree isogenies. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017, Part II. LNCS, vol. 10625, pp. 303–329. Springer, Heidelberg (Dec 2017). https://doi.org/10.1007/978-3-319-70697-9_11
14. Dartois, P., Leroux, A., Robert, D., Wesolowski, B.: SQISignHD: New dimensions in cryptography. Cryptology ePrint Archive, Paper 2023/436 (2023), https://eprint.iacr.org/2023/436, https://eprint.iacr.org/2023/436
15. Dartois, P., Maino, L., Pope, G., Robert, D.: An algorithmic approach to $(2, 2)$-isogenies in the theta model and applications to isogeny-based cryptography. Cryptology ePrint Archive, Paper 2023/1747 (2023), https://eprint.iacr.org/2023/1747, https://eprint.iacr.org/2023/1747
16. De Feo, L., Kohel, D., Leroux, A., Petit, C., Wesolowski, B.: SQISign: Compact post-quantum signatures from quaternions and isogenies. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020, Part I. LNCS, vol. 12491, pp. 64–93. Springer, Heidelberg (Dec 2020). https://doi.org/10.1007/978-3-030-64837-4_3
17. De Feo, L., Kohel, D., Leroux, A., Petit, C., Wesolowski, B.: SQISign: compact post-quantum signatures from quaternions and isogenies. Cryptology ePrint Archive, Report 2020/1240 (2020), https://eprint.iacr.org/2020/1240
18. De Feo, L., Leroux, A., Longa, P., Wesolowski, B.: New algorithms for the deuring correspondence - towards practical and secure SQISign signatures. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 659–690. Springer, Heidelberg (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_23
19. Delfs, C., Galbraith, S.D.: Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$. DCC **78**(2), 425–440 (2016). https://doi.org/10.1007/s10623-014-0010-1

20. Deuring, M.: Die typen der multiplikatorenringe elliptischer funktionenkörper. Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg **14**, 197–272 (1941)
21. Duparc, M., Fouotsa, T.B.: SQIPrime: A dimension 2 variant of SQISignHD with non-smooth challenge isogenies. Cryptology ePrint Archive, Paper 2024/*** (2024), https://eprint.iacr.org/2024/***, https://eprint.iacr.org/2024/***
22. Duparc, M., Fouotsa, T.B., Vaudenay, S.: SILBE: an updatable public key encryption scheme from lollipop attacks. Cryptology ePrint Archive, Paper 2024/400 (2024), https://eprint.iacr.org/2024/400, https://eprint.iacr.org/2024/400
23. Eisentraeger, K., Hallgren, S., Leonardi, C., Morrison, T., Park, J.: Computing endomorphism rings of supersingular elliptic curves and connections to pathfinding in isogeny graphs. In: Galbraith, S. (ed.) ANTS-XIV - 14th Algorithmic Number Theory Symposium. Proceedings of the Fourteenth Algorithmic Number Theory Symposium (ANTS-XIV), vol. 4, pp. 215–232. Mathematical Sciences Publishers, Auckland, New Zealand (Jun 2020). https://doi.org/10.2140/obs.2020.4.215
24. Eisenträger, K., Hallgren, S., Lauter, K.E., Morrison, T., Petit, C.: Supersingular isogeny graphs and endomorphism rings: Reductions and solutions. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part III. LNCS, vol. 10822, pp. 329–368. Springer, Heidelberg (Apr / May 2018). https://doi.org/10.1007/978-3-319-78372-7_11
25. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO'86. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (Aug 1987). https://doi.org/10.1007/3-540-47721-7_12
26. Fieker, C., Hart, W., Hofmann, T., Johansson, F.: 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. pp. 157–164. ISSAC '17, ACM, New York, NY, USA (2017). https://doi.org/10.1145/3087604.3087611, http://doi.acm.org/10.1145/3087604.3087611
27. Galbraith, S.D., Petit, C., Silva, J.: Identification protocols and signature schemes based on supersingular isogeny problems. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017, Part I. LNCS, vol. 10624, pp. 3–33. Springer, Heidelberg (Dec 2017). https://doi.org/10.1007/978-3-319-70694-8_1
28. Howe, E.W., Leprévost, F., Poonen, B.: Large torsion subgroups of split Jacobians of curves of genus two or three. Forum Mathematicum **12**(3), 315–364 (2000)
29. Jao, D., Azarderakhsh, R., Campagna, M., Costello, C., De Feo, L., Hess, B., Jalali, A., Koziel, B., LaMacchia, B., Longa, P., Naehrig, M., Renes, J., Soukharev, V., Urbanik, D., Pereira, G., Karabina, K., Hutchinson, A.: SIKE. Tech. rep., National Institute of Standards and Technology (2022), available at https://csrc.nist.gov/Projects/post-quantum-cryptography/round-4-submissions
30. Jao, D., De Feo, L.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In: Yang, B.Y. (ed.) Post-Quantum Cryptography - 4th International Workshop, PQCrypto 2011. pp. 19–34. Springer, Heidelberg (Nov / Dec 2011). https://doi.org/10.1007/978-3-642-25405-5_2
31. Kani, E.: The number of curves of genus two with elliptic differentials. Journal für die reine und angewandte Mathematik **485**, 93–122 (1997). https://doi.org/doi:10.1515/crll.1997.485.93

32. Kohel, D., Lauter, K., Petit, C., Tignol, J.P.: On the quaternion ℓ-isogeny path problem. LMS Journal of Computation and Mathematics **17**(A), 418–432 (2014). https://doi.org/10.1112/S1461157014000151, https://hal.archives-ouvertes.fr/hal-01257092
33. Lang, S.: Elliptic Functions. Graduate texts in mathematics, Springer, 2nd edn. (1987), https://books.google.co.jp/books?id=IxRZAAAAYAAJ
34. Leroux, A.: Verifiable random function from the Deuring correspondence and higher dimensional isogenies. Cryptology ePrint Archive, Paper 2023/1251 (2023), https://eprint.iacr.org/2023/1251, https://eprint.iacr.org/2023/1251
35. Lubicz, D., Robert, D.: Computing isogenies between abelian varieties. Compositio Mathematica **148**(5), 1483–1515 (2012). https://doi.org/10.1112/S0010437X12000243
36. Maino, L., Martindale, C., Panny, L., Pope, G., Wesolowski, B.: A direct key recovery attack on SIDH. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 448–471. Springer, Heidelberg (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_16
37. Meyer, M., Reith, S.: A faster way to the CSIDH. In: Chakraborty, D., Iwata, T. (eds.) INDOCRYPT 2018. LNCS, vol. 11356, pp. 137–152. Springer, Heidelberg (Dec 2018). https://doi.org/10.1007/978-3-030-05378-9_8
38. Nakagawa, K., Onuki, H.: QFESTA: Efficient algorithms and parameters for FESTA using quaternion algebras. Cryptology ePrint Archive, Paper 2023/1468 (2023), https://eprint.iacr.org/2023/1468, https://eprint.iacr.org/2023/1468
39. Nakagawa, K., Onuki, H.: SQIsign2D-East: A new signature scheme using 2-dimensional isogenies. Cryptology ePrint Archive, Paper 2024/*** (2024), https://eprint.iacr.org/2024/***, https://eprint.iacr.org/2024/***
40. Onuki, H.: On the key generation in SQISign (2023). https://doi.org/10.4064/bc126-6
41. Petit, C., Smith, S.: An improvement to the quaternion analogue of the l-isogeny path problem (2018), conference talk at MathCrypt 2018
42. Robert, D.: Breaking SIDH in polynomial time. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 472–503. Springer, Heidelberg (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_17
43. Smith, B.A.: Explicit endomorphisms and correspondences. PhD thesis, University of Sydney (2005)
44. Vélu, J.: Isogénies entre courbes elliptiques. Comptes-Rendues de l'Académie des Sciences **273**, 238–241 (1971)
45. Voight, J.: Quaternion Algebras. Graduate Texts in Mathematics, Springer Cham (2022). https://doi.org/10.1007/978-3-030-56694-4

A Computing connecting tuples

Let $\mathcal{O}_0, R, m_1, m_2, E_0, P_0, Q_0$ be as in Section 3. In this section, we explain how to compute $(E_0; P_0, Q_0)$-connecting tuples. In particular, we compute a tuple

$$(E_0', I_0, \mathfrak{O}, P_0', Q_0', \sigma(P_0'), \sigma(Q_0')),$$

where E_0' is a supersingular elliptic curve over $\mathbb{F}_{p^2}$, I_0 is a left $\mathcal{O}_0$-ideal such that $\mathcal{O}_{\mathrm{R}}(I_0) \cong \mathrm{End}(E_0')$ and $\mathrm{n}(I_0)$ is prime to m_1m_2, $\mathfrak{O}$ is an imaginary quadratic order in $\mathrm{End}(E_0')$, P_0' and Q_0' are the images of P_0 and Q_0 under φ_{I_0}, and σ is an element in $\mathfrak{O}$ such that $\mathfrak{O} = \mathbb{Z}[\sigma]$.

A.1 Algorithm for computing connecting tuples

The outline of the computation of an $(\mathcal{O}_0; P_0, Q_0)$-connecting tuple is as follows:

1. Take a small square-free integer d such that p does not split in $\mathbb{Q}(\sqrt{-d})$ and let $\mathfrak{O}$ be $\mathbb{Z}[\sqrt{-d}]$.
2. Take a prime $N > p$ such that there exists $\sigma_0 \in \mathcal{O}_0$ such that $\sigma_0^2 = -N^2 d$.
3. Let $I_0 = \mathcal{O}_0\sigma_0 + \mathcal{O}_0 N$.
4. Compute the codomain E_0' of φ_{I_0}, $P_0' = \varphi_{I_0}(P_0)$ and $Q_0' = \varphi_{I_0}(Q_0)$ by a variant of IdealToIsogenyIQO.
5. Compute $\sigma(P_0')$ and $\sigma(Q_0')$ by $\sigma = \frac{1}{N^2}\varphi_{I_0} \circ \sigma_0 \circ \hat{\varphi}_{I_0}$.

In the following, we explain the detail of each step.

Step 1. From the condition on d, there exists a supersingular elliptic curve over $\mathbb{F}_{p^2}$ whose endomorphism ring containing a subring isomorphic to $\mathbb{Z}[\sqrt{-d}]$ (see [33, Theorem 12 in Chapter 13]). The following steps computes such a curve E_0' and the connection between E_0 and E_0'.

Step 2. Since $N > p$, there exists an N-isogeny from E_0 to E_0' with high probability. Suppose such an isogeny φ exists. Let σ be an endomorphism on E_0' corresponding to $\sqrt{-d}$. Then $\sigma_0 := \hat{\varphi} \circ \sigma \circ \varphi$ is an endomorphism on E_0 corresponding to $N\sqrt{-d}$, i.e., $\sigma_0^2 = -N^2 d$.

A method to find such σ_0 is as follows:

(i) Let $\bar{a}$ be a square root of $N^2 d/\mathrm{n}(\mathbf{i})$ modulo p (such $\bar{a}$ always exists from the condition on d) and a be a lift of $\bar{a}$ to $\mathbb{Z}$ such that $0 \le a < p$.
(ii) Find $\tau \in R$ such that $\mathrm{n}(\tau) = (N^2 d - a^2 \mathrm{n}(\mathbf{i}))/p$ by Cornacchia's algorithm.
(iii) Let $\sigma_0 = a\mathbf{i} + \mathbf{j}\tau$.

Since $\mathrm{tr}(\sigma_0) = 0$ and $\mathrm{n}(\sigma_0) = N^2 d$, we have $\sigma_0^2 = -N^2 d$. The Step (ii) can fail. In this case, we replace N by another prime and retry.

Step 3. Let $\sigma = \frac{1}{N}\sigma_0$. Since $I_0\sigma \subset I_0$, we have $\sigma \in \mathcal{O}_{\mathrm{R}}(I_0)$. Therefore, the endomorphism ring of the codomain of φ_{I_0} contains a subring isomorphic to $\mathbb{Z}[\sqrt{-d}]$. It holds that $\mathrm{n}(I_0) = N$, which we will prove in the next subsection.

Step 4. As we mentioned in Section 3.3, the first ShortIdealToIsogenyIQO in the computation of φ_{I_0} may fail if we do not use connecting tuples.

To avoid this problem, we use the other factors of $p^2 - 1$. Let m_3 be a smooth factor of $p^2 - 1$ such that $m_1 m_3 > \sqrt{p}$ and m_3 is prime to $m_1 m_2$. Then we compute m_3-isogenies from E_0 efficiency by using the method in [12]. In particular, we use J such that $J \sim I_0$ and $\mathrm{n}(J) = m_3 m_1^k$ and decompose J into $J_1 \dots J_k$ such that $\mathrm{n}(J_1) = m_1 m_3$ and $\mathrm{n}(J_i) = m_1$ for $i = 2, \dots, k$. Based on this decomposition, we compute the codomain of φ_J and the images of P_0 and Q_0 under φ_I for a left $\mathcal{O}_0$-ideal I such that $I \sim J$ and $\mathrm{n}(I)$ is prime to $m_1 m_2$ by using IdealToIsogenyIQO. Note that IdealToIsogenyIQO may fail even in this case. In this case, we return to Step 2 and retry.

We can compute $\varphi_{I_0}(P_0)$ and $\varphi_{I_0}(Q_0)$ by using $\alpha \in I_0$ such that $\chi_{I_0}(\alpha) = I$, where α is obtained by the KLPT algorithm transforming I_0 to J and the outputs of ShortIdealToIsogenyIQO in the computation of φ_J. Since $\alpha = \hat{\varphi}_I \circ \varphi_{I_0}$, we have $\varphi_{I_0} = \frac{1}{\deg \varphi_I} \varphi_I \circ \alpha$.

Step 5. The isomorphism induced by φ_{I_0} maps σ to $\frac{1}{N}\varphi_{I_0} \circ \frac{1}{N}\sigma_0 \circ \hat{\varphi}_{I_0}$. Therefore, we have

$$\sigma(P_0') = \sigma \circ \varphi_{I_0}(P_0) = \frac{1}{N}\varphi_{I_0} \circ \sigma_0(P_0).$$

The same holds for $\sigma(Q_0')$.

A.2 Proof of $\mathrm{n}(I_0) = N$

We prove the following proposition, which we used in the above explanation.

Proposition 1. *The ideal I_0 in the Step 3 in the previous subsection satisfies* $\mathrm{n}(I_0) = N$.

First, we recall basic facts on quaternion algebras. We refer to [45, Chapter 15] for the details. For a fractional ideal I of $\mathcal{B}_{p,\infty}$, the *discriminant of I* is defined by

$$\mathrm{disc}(I) = \det((\mathrm{tr}(b_i b_j))_{i,j=1,\dots,4}),$$

where $b_1, \dots, b_4$ is a $\mathbb{Z}$-basis of I. For fractional ideals I, J of $\mathcal{B}_{p,\infty}$ such that $I \subset J$, we have $\mathrm{disc}(I) = [J : I]^2 \mathrm{disc}(J)$. Let $\mathcal{O}$ be a maximal order of $\mathcal{B}_{p,\infty}$ and I be a left $\mathcal{O}$-ideal. Then $\mathrm{disc}(\mathcal{O}) = p^2$ and $\mathrm{disc}(I) = \mathrm{n}(I)^4 p^2$.

Next, we prove the following lemmas used in the proof of Proposition 1.

Lemma 2. *Let $\mathcal{O}_0$ be a special extremal order in $\mathcal{B}_{p,\infty}$ and $N > p$ be a prime. Let α be an element in $\mathcal{B}_{p,\infty}$ of the form $\alpha = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$ for $a, c, d \in \mathbb{Q}$ and $b \in \frac{1}{N}\mathbb{Z} \setminus \mathbb{Z}$. Then $\alpha \notin \mathcal{O}_0$.*

Proof. Since $\mathcal{O}_0$ is a special extremal order, $\mathcal{O}_0$ contains a sub-lattice $L \coloneqq \mathbb{Z} + \mathbb{Z}\mathbf{i} + \mathbb{Z}\mathbf{j} + \mathbb{Z}\mathbf{k}$. An easy computation shows that $\mathrm{disc}(L) = (4\mathrm{n}(\mathbf{i}))^2$. Therefore, we have $[\mathcal{O}_0 : L] = 4\mathrm{n}(\mathbf{i})$. Since $\mathrm{n}(\mathbf{i})$ is 1, 2, or the smallest prime that is a quadratic non-residue modulo p, we have $\mathrm{n}(\mathbf{i}) < p < N$. Therefore, $4\mathrm{n}(\mathbf{i})\alpha \notin L$. This means that $\alpha \notin \mathcal{O}_0$. □

Lemma 3. *Let $\mathcal{O}_0$ be a special extremal order in $\mathcal{B}_{p,\infty}$ and N be a prime. Let α be an element in $\mathcal{O}_0 \setminus N\mathcal{O}_0$ of norm divisible by N. Then the norm of a left $\mathcal{O}_0$-ideal $I_0 = \mathcal{O}_0\alpha + \mathcal{O}_0 N$ is N.*

Proof. By definition of the norm of an ideal, we have $\mathrm{n}(I_0) \mid N^2$. Let $\beta \in I_0$. Then $\beta = \alpha\beta_1 + N\beta_2$ for some $\beta_1, \beta_2 \in \mathcal{O}_0$. Therefore, $\mathrm{n}(\beta) \equiv \mathrm{n}(\alpha)\mathrm{n}(\beta_1) \equiv 0 \pmod{N}$. This implies that $N \mid \mathrm{n}(\beta)$.

Therefore, $\mathrm{n}(I_0) = N$ or N^2. The inclusion $\mathcal{O}_0 N \subset I_0$ and $\mathrm{n}(\mathcal{O}_0 N) = N^2$ imply that if $\mathrm{n}(I_0) = N^2$, then $I_0 = \mathcal{O}_0 N$. This contradicts the assumption that $\alpha \notin N\mathcal{O}_0$. Therefore, we have $\mathrm{n}(I_0) = N$. □

Proof (Proof of Proposition 1). Since $\sigma_0 = a\mathbf{i} + b\mathbf{j} + c\mathbf{k} + d\mathbf{ij}$ with $0 \le a < p$, we have $\sigma_0 \notin N\mathcal{O}_0$ by Lemma 2. Therefore, we have $\mathrm{n}(I_0) = N$ by Lemma 3. □

B The costs of a T-isogeny and a $(2^{f_2}, 2^{f_2})$-isogeny

In this section, we estimate the costs of a T-isogeny in the SQIsign and a $(2^{f_2}, 2^{f_2})$-isogeny our algorithm for the NIST security level 1. In the following, we show that the cost of a T-isogeny in the SQIsign is at least almost the same as the cost of a $(2^{f_2}, 2^{f_2})$-isogeny in our algorithm.

We use the characteristic p^{I}_{1973} in the document [8] for the SQIsign and $p = 2^{247} \cdot 79 - 1$ for our algorithm. Since the size of these parameters is almost the same, we assume that the costs of the operations in $\mathbb{F}_{p^{\mathrm{I}}_{1973}}$ and $\mathbb{F}_p$ are the same.

We use the number of multiplications (including squarings) in $\mathbb{F}_p$ as the measure of costs and ignore the cost of additions. We denote this cost by $\mathbf{M}_{\mathbb{F}_p}$, the cost of a multiplication in $\mathbb{F}_{p^2}$ by $\mathbf{M}$, the cost of a squaring in $\mathbb{F}_{p^2}$ by $\mathbf{S}$, and the cost of an inversion in $\mathbb{F}_{p^2}$ by $\mathbf{I}$. By the standard method to compute multiplications and squarings in $\mathbb{F}_{p^2}$, it holds that $\mathbf{M} = 3\mathbf{M}_{\mathbb{F}_p}$, $\mathbf{S} = 2\mathbf{M}_{\mathbb{F}_p}$ and $\mathbf{I}$ is the sum of $\mathbf{M}+2\mathbf{S}$ and the cost of an inversion in $\mathbb{F}_p$. We assume that an inversion in $\mathbb{F}_p$ is computed in constant time, thus we assume $\mathbf{I} = \mathbf{M}+2\mathbf{S}+1.5\log_2(p)\mathbf{M}_{\mathbb{F}_p}$.

Based on the above assumptions, we show that the cost of a T-isogeny in the SQIsign is at least $141,987\mathbf{M}_{\mathbb{F}_p}$, and the cost of a $(2^{f_2}, 2^{f_2})$-isogeny in our algorithm is appropriately $147,951\mathbf{M}_{\mathbb{F}_p}$.

B.1 The cost of a T-isogeny in the SQIsign

As in the implementation of the SQIsign, we assume that the computation is done in the x-coordinate of Montgomery curves.

Let T^+ and T^- be integers such that $T = T^+T^-$ and $T^+ \mid p+1$ and $T^- \mid p-1$. Consider the computation of a T-isogeny from a supersingular elliptic curve E. To use points in $E[T^-]$, we use the quadratic twist $E^{(t)}$ of E. In this setting, we have $E[T^+] \subset E(\mathbb{F}_{p^2})$ and $E^{(t)}[T^-] \subset E^{(t)}(\mathbb{F}_{p^2})$.

A T-isogeny is computed by dividing into T^+-isogenies and T^--isogenies. Algorithm 5 shows the computation of a T-isogeny. In the algorithm, we denote the x-coordinate of a point P by $x(P)$.

Algorithm 5: Computing a T-isogeny in the SQIsign

Input: A supersingular elliptic curve E, $x(K^+)$ for $K^+ \in E$ of order T^+, $x(K^-)$ for $K^- \in E^{(t)}$ of order T^-, and $x(P), x(Q), x(P-Q)$ for a basis (P, Q) of $E[2^f]$.

Output: The codomain of a T-isogeny with kernel $\langle K^+, K^- \rangle$ and the x-coordinates of the images of $P, Q, P-Q$ under the isogeny.

1 Let $k = T^+$;
2 Let $x(R) = x(P)$ and $x(S) = x(Q)$;
3 **for** *prime-power diviors* ℓ^e *of* T^+ **do**
4   Let $k = k/\ell^e$;
5   Let $x(K) = x(kK^+)$;
6   Compute the codomain of an ℓ^e-isogeny φ with kernel $\langle K \rangle$ and the images $x(\varphi(K^+)), x(\varphi(K^-)), x(\varphi(R)), x(\varphi(S))$, and $x(\varphi(R-S))$;
7 Let $k = T^-$;
8 **for** *prime-power diviors* ℓ^e *of* T^- **do**
9   Let $k = k/\ell^e$;
10   Let $x(K) = x(kK^-)$;
11   Compute the codomain of an ℓ^e-isogeny φ with kernel $\langle K \rangle$ and the images $x(\varphi(K^-)), x(\varphi(R)), x(\varphi(S))$, and $x(\varphi(R-S))$;
12 **return** *the codomain,* $x(\varphi(R))$*,* $\varphi(S)$*, and* $\varphi(R-S)$;

First, we consider the cost of the multiplications by k in line 5 and 10 in Algorithm 5. For minimizing this cost, we suppose that the for loops in the algorithm are executed in the descending order in the prime-power divisors of T^+ and T^-. Let $T^+ = \prod_{i=1}^{m_+} \ell_{+,i}^{e_{+,i}}$ and $T^- = \prod_{i=1}^{m_-} \ell_{-,i}^{e_{-,i}}$ be the prime factorizations of T^+ and T^- in the descending order. By using the Montgomery ladder, the scalar multiplication by an integer n is computed by $\lfloor \log_2(b) \rfloor$ calls of xDBLADD (Algorithm 5 in [29]). Since the cost of xDBLADD is $7\mathbf{M} + 4\mathbf{S}$, the cost of the scalar multiplications is

$$\left(\sum_{i=1}^{m_+} \left\lfloor \sum_{j=i+1}^{m_+} \log_2(\ell_{+,j}^{e_{+,j}}) \right\rfloor + \sum_{i=1}^{m,-} \left\lfloor \sum_{j=i+1}^{m_-} \log_2(\ell_{-,j}^{e_{-,j}}) \right\rfloor \right) (7\mathbf{M} + 4\mathbf{S}).$$

Next, we consider the cost of the isogenies. The SQIsign uses an algorithm by [13,37] for isogenies of small degrees and an algorithm by [5] for isogenies of large degrees. These algorithms consist of the following three parts:

1. KPS, which computes (a part of) the kernel of the isogeny from a generator of the kernel in the input.
2. CODOM, which computes the codomain of the isogeny from the kernel.
3. PEVAL, which computes the image of a point under the isogeny from the point and the kernel.

Since our purpose is to estimate a lower bound of the costs, we ignore the cost to compute a generator of the kernel from a point of order ℓ^e and the cost to compute $x(\varphi(K^+))$ and $x(\varphi(K^-))$ for simplicity. In addition, we ignore the cost of CODOM because the codomain of φ can be computed by $x(\varphi(R)), x(\varphi(S))$, and $x(\varphi(R-S))$ by using Algorithm 10 in [29]. As a result, we estimate a lower bound of the costs of the isogenies by the the cost of one KPS and three PEVALs.

The cost of the algorithm for small degrees can be found in Table 1 in [9] for example. The cost of KPS is $2(\ell-3)\mathbf{M}+(\ell-3)\mathbf{S}$ and the cost of PEVAL is $2(\ell-1)\mathbf{M}+2\mathbf{S}$ for an ℓ-isogeny in this algorithm. We denote the total cost of computing an ℓ-isogeny by this algorithm by $c_{\text{small}}(\ell)$. In particular,

$$c_{\text{small}}(\ell) = 2(\ell-3)\mathbf{M} + (\ell-3)\mathbf{S} + 3(2(\ell-1)\mathbf{M} + 2\mathbf{S}).$$

The cost of the algorithm for large degrees is analyzed in [1]. Their experiment shows that their estimation is 20-30% lower than the actual cost. Therefore, their estimation matches our purpose. Their estimation is given in the number of multiplications that does not distinguish multiplications and squarings. We conservatively assume that all multiplications are squarings. As a result, the cost of computing an ℓ-isogeny by the algorithm for large degrees is as follows: Let b be $\lfloor\sqrt{\ell-1}/2\rfloor$. The cost of KPS is $24b\mathbf{S}$. The cost of PEVAL is at least

$$(2\left(2b^{\log_2(3)} + b\log_2(b) - \frac{5}{3}b + \frac{5}{6}\right) + 16b + 3b^{\log_2(2)} - 2)\mathbf{S}.$$

See §4.3 and Appendix A.3 in [1] for the details. We denote the total cost of computing an ℓ-isogeny by this algorithm by $c_{\text{large}}(\ell)$. In particular,

$$c_{\text{large}}(\ell) = \left(24b + 3\left(2\left(2b^{\log_2(3)} + b\log_2(b) - \frac{5}{3}b + \frac{5}{6}\right) + 16b + 3b^{\log_2(2)} - 2\right)\right)\mathbf{S}.$$

In summary, the cost of the isogenies in Algorithm 5 is bounded below by

$$\sum_{\ell^e \mid T} e \min\{c_{\text{small}}(\ell), c_{\text{large}}(\ell)\},$$

where ℓ^e runs over the prime-power divisors of T.

By summing up the costs of the multiplications and the isogenies, we obtain a lower bound of the total cost of the T-isogenies in the SQIsign. The cost is at least $141,987\mathbf{M}_{\mathbb{F}_p}$.

B.2 The cost of a $(2^{f_2}, 2^{f_2})$-isogeny in our algorithm

We compute $(2^{f_2}, 2^{f_2})$-isogenies by the algorithm using the theta algorithm by [15].

Algorithm 6 shows the computation of a $(2^{f_2}, 2^{f_2})$-isogeny. In this algorithm, we compute the images of four points in E under the isogeny. This is because we need to compute the image of the deterministic basis of $E[2^{f_1+f_2+2}]$ and a point in E for checking the post-composition of an automorphism (see Algorithm 2).

Algorithm 6: Computing a $(2, 2)$-isogeny in our algorithm

Input: The product of two supersingular elliptic curves $E \times E'$, $x(K_1), x(K_2), x(K_1 - K_2)$, $x(K'_1), x(K'_2), x(K'_1 - K'_2)$ such that the orders of the points are 2^{f_2+2} and the Weil pairing acts trivially on $\langle (4K_1, 4K_2), (4K'_1, 4K'_2) \rangle$, and $x(P_E), x(Q_E), x(P_E - Q_E)$, $x(R)$ for $R \in E$.

Output: The codomain of a $(2^{f_2}, 2^{f_2})$-isogeny with kernel $\langle (4K_1, 4K_2), (4K'_1, 4K'_2) \rangle$, and the images of $(P_E, 0_{E'}), (Q_E, 0_{E'}), (P_E - Q_E, 0_{E'}), (R, 0_{E'})$ under the isogeny.

1 Let $T_1 = 2^{f_2} K_1$, $T_2 = 2^{f_2} K_2$, $T'_1 = 2^{f_2} K'_1$, and $T'_2 = 2^{f_2} K'_2$;
2 Compute the x-coordinates of $K_1 + T1 = (2^{f_2} + 1) K_1$ and $K'_1 + T'_1 = (2^{f_2} + 1) K'_1$ by using the Montgomery ladder;
3 Compute the x-coordinates of $K_2 + T_1 = 2^{f_2} K_1 + K_2$ and $K'_2 + T'_1 = 2^{f_2} K'_1 + K'_2$ by using Algorithm 8 in [29];
4 Compute the x-coordinates of $P_E + T_1, Q_E + T_1, P_E - Q_E + T_1, R + T_1$ by solving quadratic equations;
5 Compute the codomain of a $(2, 2)$-isogeny Φ with kernel $\langle (4T_1, 4T_2), (4T'_1, 4T'_2) \rangle$ by Algorithm 7 in [15];
6 Compute the images of of $(K_1, K_2), (K'_1, K'_2), (P_E, 0_{E'}), (Q_E, 0_{E'}), (P_E - Q_E, 0_{E'}), (R, 0_{E'})$ under Φ by Algorithm 8 in [29];
7 Compute the codomain of a $(2^{f_2-1}, 2^{f_2-1})$-isogeny Ψ with kernel $\langle 4\Phi((K_1, K_2)), 4\Phi((K'_1, K'_2)) \rangle$ by Algorithm 5 in [15];
8 Compute the images of $\Phi((P_E, 0_{E'})), \Phi((Q_E, 0_{E'})), \Phi((P_E - Q_E, 0_{E'})), \Phi((R, 0_{E'}))$ under Ψ by Algorithm 6 in [29];
9 **return** *the codomain,* $\Psi \circ \Phi((P_E, 0)), \Psi \circ \Phi((Q_E, 0)), \Psi \circ \Phi((P_E - Q_E, 0)), \Psi \circ \Phi((R, 0))$;

The first line is done by the Montgomery doubling (Algorithm 3 in [29]) and its cost is

$$4(f_2 - 1)(4\mathbf{M} + 2\mathbf{S}). \tag{2}$$

The computation of the first $(2, 2)$-isogeny, which is an isogeny from the product of two elliptic curves to the jacobian of a genus 2 curve, is done by Algorithm 7 and 8 in [15]. Algorithm 7 computes the codomain of the isogeny and Algorithm 8 computes the images of the points under the isogeny. We call these algorithms GluingCodom and GluingImage, respectively.

As in Algorithm 6, let $T_1, T_2 \in E$ and $T'_1, T'_2 \in E'$ be points of order 4 such that the kernel of the first isogeny Φ is generated by $(2T_1, 2T_2)$ and $(2T'_1, 2T'_2)$. To compute the images of $(P, P') \in E \times E'$ under Φ, GluingImage requires the x-coordinates of $P + T_1$ and $P' + T'_1$ in addition to the x-coordinates of the generators of the kernel and P, P'. Lines 2-4 in Algorithm 6 computes these x-coordinates.

Algorithm 7: Computing the x-coordinate of $P_1 + P_2$ or $P_1 - P_2$

Input: The projective Montgomery coefficient $(A : C)$ of a Montgomery curve E, and x-coordinates $(X_1 : Z_1)$ and $(X_2 : Z_2)$ of $P_1, P_2 \in E$.
Output: The projective x-coordinate of $P_1 + P_2$ or $P_1 - P_2$.

Let $X_{12} = X_1X_2$, $Z_{12} = Z_1Z_2$, $W_{12} = X_1Z_2$, and $W_{21} = X_2Z_1$;
Let $a = (W_{12} - W_{21})^2C$;
Let $b = (X_{12} + Z_{12})(W_{12} + W_{21})C + (A + A)X_{12}Z_{12}$;
Let $c = (X_{12} - Z_{12})^2C$;
Let $D = \sqrt{b^2 - ac}$;
return $(b + D : a)$;

In line 2, we compute $K_1 + T_1$ and $K_1' + T_1'$ by using the Montgomery ladder. As in the previous subsection, the cost of this computation is

$$2\lfloor \log_2(2^{f_2} + 1)\rfloor(7\mathbf{M} + 4\mathbf{S}). \tag{3}$$

In line 3, we compute $K_2 + T_1$ and $K_2' + T_1'$ by the Three point ladder (Algorithm 8 in [29]). For this, we need $x(K_1 - K_2)$ and $x(K_1' - K_2')$ in the input. Its cost is the same as the cost of the Montgomery ladder. Therefore, the cost of this computation is

$$2\lfloor \log_2(2^{f_2})\rfloor(7\mathbf{M} + 4\mathbf{S}). \tag{4}$$

In line 4, we solve quadratic equations to compute the x-coordinates we need. Let A be the Montgomery coefficient of E, and P_1, P_2 be points in E. Let $x_1 = x(P_1)$ and $x_2 = x(P_2)$. Then the quadratic equation in X

$$(x_1 - x_2)^2X^2 - 2((x_1x_2 + 1)(x_1 + x_2) + 2Ax_1x_2)X + (x_1x_2 - 1)^2 = 0$$

has solutions $x(P_1 + P_2)$ and $x(P_1 - P_2)$. Note that we do not need to distinguish $P_1 + P_2$ and $P_1 - P_2$ because the rest of the computation done on Kummer surfaces. This equation is solved by Algorithm 7 and its cost is the sum of $11\mathbf{M} + 2\mathbf{S}$ and the cost of computing a square root in $\mathbb{F}_{p^2}$. A square root in $\mathbb{F}_{p^2}$ is computed by Algorithm 9 in [2] and its cost is $4\mathbf{M} + 3\log_2(p)\mathbf{M}_{\mathbb{F}_p}$, where we assume that the cost of computing a square root in $\mathbb{F}_p$ is $1.5\log_2(p)\mathbf{M}_{\mathbb{F}_p}$. In summary, the cost in line 4 is

$$4(11\mathbf{M} + 2\mathbf{S} + 4\mathbf{M} + 3\log_2(p)\mathbf{M}_{\mathbb{F}_p}). \tag{5}$$

The costs of GluingCodom and GluingImage are given by [15] and these are

$$13\mathbf{M} + 8\mathbf{S} + 1\mathbf{I} \text{ and } 6(5\mathbf{M} + 8\mathbf{S} + 1\mathbf{I}). \tag{6}$$

The $(2^{f_2-1}, 2^{f_2-1})$-isogeny Ψ in lines 7 and 8 is computed by using the strategy technique in [30]. The isogeny Ψ is computed by the composition of $f_2 - 1$ $(2, 2)$-isogenies, and the strategy technique gives an optimal method to compute generators of the kernels of these $(2,2)$-isogenies. This optimal method and its cost can be computed by Algorithm 60 in [29]. Note that this algorithm does not output the cost as it is, but the cost is computed in the algorithm and we can extract it. We denote the cost computed by this algorithm with input n, p, q by $c_{\mathrm{st}}(n, p, q)$. By using this algorithm, the cost in lines 7 and 8 is

$$2c_{\mathrm{st}}(f_2 - 2, 6\mathbf{M} + 8\mathbf{S}, 3\mathbf{M} + 4\mathbf{S}) + (f_2 - 1)(9\mathbf{M} + 8\mathbf{S} + 1\mathbf{I} + 4(3\mathbf{M} + 4\mathbf{S})).$$

Summing up this cost and the costs in (2)-(6) we obtain the total cost of the $(2^{f_2}, 2^{f_2})$-isogenies in our algorithm. The cost is approximately $147,951\mathbf{M}_{\mathbb{F}_p}$.