SQIPrime: A dimension 2 variant of SQISignHD with non-smooth challenge isogenies

Max Duparc and Tako Boris Fouotsa

EPFL, Lausanne, Switzerland
tako.fouotsa@epfl.ch
maxduparc@gmail.com

Abstract. We introduce SQIPrime, a post-quantum digital signature scheme based on the Deuring correspondence and Kani's Lemma. Compared to its predecessors that are SQISign and especially SQISignHD, SQIPrime further expands the use of high dimensional isogenies, already in use in the verification in SQISignHD, to both key generation and commitment. In doing so, it no longer relies on smooth degree isogenies (of dimension 1). SQIPrime operates with a prime number of the form $p = 2^\alpha f - 1$, as opposed to SQISignHD that uses SIDH primes.

The most intriguing novelty in SQIPrime is the use of non-smooth degree isogenies as challenge isogeny. In fact, in the SQISign family identification scheme, the challenge isogeny is computed by the verifier, who is not well-equipped to compute an isogeny of large non-smooth degree. To overcome this obstacle, the verifier samples the kernel of the challenge isogeny and the task of computing this isogeny is accomplished by the prover. The response is modified in such a way that the verifier can check that his challenge isogeny was correctly computed by the prover, on top of verifying the usual response in the SQISign family.

We describe two variants of SQIPrime: SQIPrime4D which uses dimension 4 isogenies to represent the response isogeny, and SQIPrime2D which solely uses dimension 2 isogenies to represent the response isogeny and hence is more efficient compared to SQIPrime4D and to SQISignHD.

Keywords: Post-Quantum Cryptography · Supersingular Isogenies · SQISign · SQISignHD · Kani's Lemma · SQIPrime

1 Introduction

The interest of isogeny based signature schemes is that they provide compact post-quantum signatures. This property, which comes at the cost of a greater computational cost, motivated their research. Among the early propositions of isogeny based signature schemes [45,4,13, ...], was GPS [24] that specifically relied on Deuring correspondence [18]. Its ideas were expended and improved in 2020 by De Feo, Kohel, Leroux, Petit and Wesolowski to create the SQISign protocol in [15]. As of today, SQISign is the only isogeny based candidate at the NIST post-quantum cryptography standardization effort. In 2023, Dartois, Leroux, Robert and Wesolowski proposed SQISignHD [9], a variant of SQISign utilising Kani's Lemma [27] for verification. Both SQISign and SQISignHD are, as of today, the two most compact post-quantum signatures, of respective size 177B for SQISign and 109B for SQISignHD.

Kani's Lemma and high dimensional isogenies (originally used in [5,32,41] to prove that SIDH [26,14] was insecure by leveraging accessible images of torsion points) are used in SQISignHD to solve some drawbacks of SQISign as it can be used to represent isogenies of unsmooth degree, which significantly simplifies the signature part of SQISignHD, at the cost of a more complex verification. The emergence of SQISignHD is part of a broader trend in Isogeny Based Cryptography, consisting in leveraging the new capabilities enabled by Kani's Lemma, a trend that birthed many new cryptographic schemes such as SQISignHD [9], FESTA and QFESTA [2,35], IS-CUBE [34], SCALLOP-HD [6], DeuringVRF [31] or SILBE [19].

As mentioned above, the main input in SQISignHD is the use of high dimensional isogenies to represent the response. In SQISign, the secret key is an isogeny $\tau : E_0 \to E_A$, where E_0 has $j-$invariant 1728. The commitment is a curve E_1 obtained by computing an isogeny $\psi : E_0 \to E_1$, the challenge is an isogeny $\varphi : E_1 \to E_2$. The response is an isogeny $\sigma : E_A \to E_2$, which closes the diagram (see left-hand side of Figure 1). The isogeny σ is in fact a long smooth isogeny of degree roughly $p^{15/4}$, obtained through a more efficient variant [15,16] of the KLPT algorithm [28]. The usage of the KLPT algorithm and the fact that the degree of the response isogeny σ is roughly $p^{15/4}$ implies that one needs to use primes with as much accessible (defined over a small extension of $\mathbb{F}_p$) smooth torsion as possible. This is one of the biggest constraints in SQISign and was solved in SQIsignHD.

The attacks [5,32,41] on SIDH/SIKE (and any other isogeny-based protocol revealing images of smooth order torsion points such as [17,8,22, ...]) led to a new method for representing isogenies of generic degree [40]. In fact, an evaluation of an isogeny on torsion points of large (with respect to the degree of the isogeny) smooth order is a representation of this isogeny. In SQISignHD, from the knowledge of the endomorphism rings of the curves in play, the signer samples a relatively short (but non-smooth) response isogeny σ and evaluates it on torsion points of smooth order. This evaluation is then returned to the verifier as the response. Since this evaluation represents the isogeny, the verifier can efficiently check that the data received represents an isogeny $\sigma : E_1 \to E_2$. Note that here, the response goes from E_1 to E_2 while the challenge goes from E_A to E_2, this change is made for a more convenient implementation. This brings several relaxations, among which the change of the base prime p to an SIDH prime: $p = 2^a 3^b f - 1$. In SQISign, the most computationally involving part is transforming the ideal obtained from KLPT into an isogeny, this is done during the signing process. In SQISignHD, the signing is relatively easier since the KLPT algorithm is avoided, but the verification is computationally involving. In fact, in order to validate that the evaluation returned by the signer represents an isogeny $\sigma : E_1 \to E_2$, one needs to compute and evaluate an isogeny in higher dimension: 2, 4 or 8 in general. The smaller the dimension, the more efficient the computation and the evaluation are. In SQISignHD, the verification uses dimension 4 isogenies. There is a huge efficiency gap between dimension 4 isogenies and dimension 2 isogenies [29,9,10,42]. Hence, in the quest for better efficiency, it becomes natural to ask the following question:

Can one design a variant of SQISignHD that uses only dimension 1 and/or dimension 2 isogenies?

Contributions. In this paper, we answer the question above in the affirmative, by describing SQIPrime, a derivative of SQISignHD. To do so, we first extend the usage of Kani's Lemma to both key generation and commitment, by adapting the **RandIsogImages** algorithm from QFESTA [35]. Next, we modify the challenge isogeny generation in such a way that the verifier can use non-smooth degree isogenies, by sampling solely the kernel generator of this isogeny. The signer/prover can then use the techniques introduced by Leroux [31] to compute this challenge isogeny and include it in the response. As a consequence, we use primes of the form $p = 2^\alpha f - 1 = 2Nq + 1$ where q is the degree of the challenge. These changes induce numerous adaptations and optimizations throughout the protocol. In order to ease the digestion of the numerous changes, we propose two variants of SQIPrime: SQIPrime4D and SQIPrime2D.

In SQIPrime4D, we incorporate the most basic changes to SQISignHD, without necessarily aiming for a better efficiency. These changes include: the usage of an adaptation (**KaniDoublePath**, Section 3.1) of the **RandIsogImages** algorithm from QFESTA [35] for key generation and commitment, and the usage of a non-smooth degree isogeny for commitment. More precisely, let $\tau : E_0 \to E_A$, $\psi : E_0 \to E_1$, $\varphi : E_A \to E_2$ and $\sigma : E_1 \to E_2$ be the secret, commitment, challenge and response isogenies in SQISignHD. In SQIPrime4D, τ and ψ are generated using the **KaniDoublePath** algorithm. For the challenge, the verifier samples a uniformly random scalar $a \in \mathbb{Z}_q$ where q is the degree of the commitment isogeny. The scalar a defines a point $C = P + [a]Q$ where (P, Q) is a specified basis of $E_A[q]$. The signer/prover uses the techniques in the DeuringVRF [31] to translate C into its corresponding ideal I_φ, which is in fact the ideal corresponding to the challenge isogeny $\varphi : E_A \to E_2$. From here, he recovers the endomorphism ring of E_2, solves for a short isogeny $\sigma : E_2 \to E_1$ (note that this is the dual of the response in the original SQISignHD), and evaluates $\kappa = \sigma \circ \varphi$ on the 2^α-torsion points (this is illustrated in Figure 2). The evaluation of $\kappa = \sigma \circ \varphi$ is then returned to the verifier as the response. The verifier checks that the data he received represents an isogeny $\kappa : E_A \to E_1$ of degree qd whose kernel contains $C = P + [a]Q$ and, q and d are co-prime. This proves that κ factors through the challenge $\varphi : E_A \to E_2$ whose kernel was sampled by the verifier. The verification is performed using dimension 4 isogenies. In SQIPrime2D, we implement further adjustments in order to use only dimension 2 isogenies.

The main obstacle when representing isogenies in dimension 2 is the need of an auxiliary isogeny. To represent the isogeny $\kappa := \sigma \circ \varphi : E_A \to E_1$ of degree qd returned in SQIPrime4D in dimension 2, we need an auxiliary isogeny

$\delta : E_A \to E_\delta$ of degree $2^\alpha - qd$. Hence, the goal of all the changes we will operate from now on will be to enable an efficient computation of such an auxiliary isogeny. The main change consists in fixing the degree of the secret isogeny τ to q, the same degree as that of the challenge isogeny φ, and making sure that this degree is prime. Once this is done, we sample an endomorphism $\gamma \in \mathrm{End}(E_0)$ of degree $d(2^a - dq)$, and compose it with the secret isogeny $\tau : E_0 \to E_A$ to obtain an isogeny $\tau \circ \gamma : E_0 \to E_A$ of degree $dq(2^a - dq)$. This isogeny can be seen as the composition of two isogenies of degree dq and $2^a - dq$ respectively. We then use Kani's Lemma to recover the pushforward of the isogeny of degree $2^a - dq$ in such a way that its domain is E_A, and his codomain is some curve E_δ which is computed at the same time. This pushforward is used as the sought auxiliary isogeny, allowing us to have a variant SQIPrime2D which only uses dimension 2 isogenies. The SQIPrime2D identification scheme is illustrated in Figure 3.

The key generation in SQIPrime2D requires one dimension 2 isogeny computation and evaluation. The signing process requires two dimension 2 isogeny computations and evaluations, one for the commitment isogeny and another for generating the auxiliary isogeny. The verification requires one dimension 2 isogeny computation and evaluation, bringing it up to a total of three dimension 2 isogeny computations and evaluations for the signature and verification. Given the current efficiency gap between dimension 2 and dimension 4 isogenies, we expect SQIprime2D to be more efficient compared to SQISignHD. We are currently preparing a proof of concept implementation to support this claim.

In order to prove the security of SQIPrime4D and SQIPrime2D, we assume that the codomain of an isogeny computed using the **KaniDoublePath** algorithm is computationally indistinguishable from a random supersingular curve. Once this assumption is made, we reduce the security of SQIPrime4D and SQIPrime2D to the Supersingular Endomorphism problem in the RUCGDIO or RUCODIO+AIO models respectively, models that we introduce and which are translations of the RUDGIO model (introduced in the context of SQISignHD) into the context of SQIPrime4D and SQIPrime2D respectively.

Related work. While this work was under finalisation, we came aware of two other concurrent but independent projects that were trying to answer the same open question we answer in the paper. The first project is from Nakagawa and Onuki, named SQISign2D-East [36] and the second one is from Basso, Dartois, De Feo, Leroux, Maino, Pope, Robert and Wesolowski, named SQISign2D-West [1]. Interestingly, all three papers adopt different approaches to solving this problem.

- Our mechanism mainly relies on the primality of the challenge isogeny φ and on the fact that it has the same degree as our secret isogeny τ.
- The SQISign2D-East [36] mechanism uses Eichler modules [30, Definition 1.2.7] to sample endomorphisms over E_0 that can also be interpreted as endomorphisms over E_A. The auxiliary isogeny $\delta : E_A \to E_\delta$ is then generated using such endomorphisms on E_A.

- Finally, the SQISign2D-West [1] mechanism merges **RandIsogImages** with Clapoti [37] to design a new efficient algorithm to evaluate random ideals. This algorithm is then used to compute the auxiliary isogeny by sampling its ideal, composing it with the commitment and challenge ideals, evaluating the composition. Using the knowledge of the commitment and challenge isogenies, the auxiliary isogeny is retrieved.

We therefore wholeheartedly recommend the reader to delve into these two papers (after completing ours, naturally).

Outline. The remainder of this paper is organised as follows. In Section 2, we give a quick recall on isogenies and on the architecture of both SQISign [15,16] and SQISignHD [9], together with a reminder of the standard algorithms in Isogeny Based Cryptography that we use to define SQIPrime. In Section 3, we will introduce special tools that we will need to construct both SQIPrime4D and SQIPrime2D. In Section 4, we give the detailed construction of SQIPrime4D, together with an analysis of its security in Section 5. Similarly, we give the detailed specification of SQIPrime2D in Section 6, with its security analysis in Section 7. Finally, we discuss in Section 8 how to find adequate parameters for both SQIPrime4D and SQIPrime2D and have a word about their foreseen efficiency.

2 Background

Throughout this paper, we denote by λ the security parameter. Let p be a prime, $\mathbb{F}_p$ is the finite field of cardinality p and $\overline{\mathbb{F}_p}$ is its algebraic closure. Let E and E' be elliptic curves defined over $\overline{\mathbb{F}_p}$.

2.1 Isogenies, Deuring correspondence and Kani's Lemma

Below, we provide a concise overview of Isogenies, Deuring correspondence, and Kani's Lemma. For a more comprehensive exploration, we recommend referring to De Feo's notes [11] and Silverman's book [43] for a general understanding of elliptic curves and isogenies. For insights into the Deuring Correspondence, Leroux's thesis [30] is an excellent resource, while Robert's attack on SIDH [41,40] provides valuable details on Kani's Lemma.

Isogenies: An isogeny $\phi : E \to E'$ is a surjective projective rational map between $E(\overline{\mathbb{F}_p})$ and $E'(\overline{\mathbb{F}_p})$ that preserves the group structure. The degree of this rational map in its x-value defines the *degree* of the isogeny. Consequently, the degree of a composition of isogenies is the product of the degrees of each individual isogeny.

Isogenies are considered up to isomorphism, where two isogenies $\phi : E \to F$ and $\psi : E' \to F'$ are *isomorphic* if they are equal up to pre- and/or post-composition by isomorphisms (isogenies of degree 1). This implies that if E

and E' are isomorphic, they share the same j-invariant, and both notions are equivalent when considered in $\overline{\mathbb{F}_p}$.

For every isogeny $\phi : E \to E'$, there exists a unique *dual isogeny* $\widehat{\phi} : E' \to E$ such that $\phi \circ \widehat{\phi} = [\deg(\phi)]$ and $\widehat{\phi} \circ \phi = [\deg(\phi)]$ on the respective curves, where $[n]$ denotes the scalar multiplication by n.

Given a natural number n, the *n-torsion group* of E, denoted by $E[n]$, is the kernel $\ker([n])$ of the scalar multiplication by n. It holds that $E[n] \cong \mathbb{Z}_n^2$ for n co-prime to p.

An isogeny $\phi : E \to E'$ is said to be *separable* if $\deg(\phi) = |\ker(\phi)|$. According to the intuition provided by the fundamental theorem of isomorphism, any separable isogeny is defined up to isomorphism by its kernel. This means that $\phi : E \to E'$ and $\psi : E \to E/\ker(\phi)$ are isomorphic. Additionally, for any isogeny $\phi : E \to E'$, it holds that $\ker(\phi) \subset E[\deg(\phi)]$.

The characterization of isogenies by their kernels allows us to define the notion of *pushforwards.* Let $\phi : E \to F$ and $\psi : E \to F'$ be two isogenies of co-prime degree. The *pushforward* of ψ by ϕ is the isogeny $\phi_* \psi : F \to E'$ whose kernel is given by $\ker(\phi_* \psi) = \phi\big(\ker(\psi)\big)$.

Deuring Correspondence: An endomorphism of E is an isogeny $\phi : E \to E$. Among isogenies, endomorphisms have important additional properties. First, $\mathrm{End}(E)$, the set of all endomorphisms of E is an integral ring of characteristic zero, under addition and composition. An elliptic curve E defined over $\overline{\mathbb{F}_p}$ is said to be *ordinary* if $\mathrm{End}(E)$ is isomorphic to an order of an imaginary quadratic field $\mathbb{Q}(\sqrt{-d})$. Otherwise, E is said to be *supersingular* and $\mathrm{End}(E)$ is isomorphic to a maximal order of the quaternion algebra $\mathbf{B}_{p,\infty}$ ramified exactly at p and ∞. An order $\mathcal{O}$ of $\mathbf{B}_{p,\infty}$ is a subring such that $\mathcal{O} \otimes_{\mathbb{Z}} \mathbb{Q} = \mathbf{B}_{p,\infty}$ with $\mathbf{B}_{p,\infty}$ of the form $\mathbb{Q} + \mathbb{Q}\mathbf{i} + \mathbb{Q}\mathbf{j} + \mathbb{Q}\mathbf{ij}$ such that $\mathbf{j}^2 = -p$, $\mathbf{i}^2$ depending on p and $\mathbf{ij} = -\mathbf{ji}$. An important example is the curve $E_0 : y^2 = x^3 + x$ whose j-invariant is 1728. If $p = 3 \mod 4$, then it is supersingular and its endomorphism ring correspond to the maximal order $\mathcal{O}_0 = \mathbb{Z} + \mathbf{i}\mathbb{Z} + \frac{\mathbf{i}+\mathbf{j}}{2}\mathbb{Z} + \frac{1+\mathbf{ij}}{2}\mathbb{Z}$ with $\mathbf{i} : (x, y) \to (-x, \sqrt{-1}y)$ and $\mathbf{j} = \pi$ the Frobenius endomorphism.

Supersingularity is a crucial property, as it is preserved by isogenies. Furthermore, all supersingular curves are defined (up to isomorphism) over $\mathbb{F}_{p^2}$ and are isogenous to each other. Supersingular curves and their isogenies can be represented as unoriented graphs known as supersingular isogeny graphs, denoted $\mathcal{G}_p^\ell$, with edges representing isogenies of prime degree ℓ up to isomorphism. These graphs, $\mathcal{G}_p^\ell$, are $(\ell + 1)$-regular and are in fact Ramanujan [38].

Deuring proved in [18] that there is an equivalence between supersingular curves and maximal orders of the quaternion algebra $\mathbf{B}_{p,\infty}$. Specifically, an isogeny ϕ between two curves E_0 and E_1, with $\mathrm{End}(E_0) \cong \mathcal{O}_0$ and $\mathrm{End}(E_1) \cong \mathcal{O}_1$, can be represented as an integral ideal I connecting $\mathcal{O}_0$ and $\mathcal{O}_1$. Integral ideals are fractional ideals such that $I \subseteq \mathcal{O}_L(I)$, where $\mathcal{O}_L(I) = \{\alpha \in \mathbf{B}_{p,\infty} \mid \alpha I \subseteq I\}$. Similarly, there exists $\mathcal{O}_R(I) = \{\alpha \in \mathbf{B}_{p,\infty} \mid I\alpha \subseteq I\}$. All ideals can be viewed as $\big(\mathcal{O}_L(I), \mathcal{O}_R(I)\big)$-ideals, with both $\mathcal{O}_L(I)$ and $\mathcal{O}_R(I)$ being maximal orders whenever I is integral. The *norm* of an ideal is defined as $n(I) = \gcd\big(\{n(\alpha) \mid \alpha \in I\}\big)$.

Let $\phi : E \to E'$ be an isogeny between two supersingular curves. Let $\mathcal{O}_E$ and $\mathcal{O}_{E'}$ be the maximal orders of $\mathbf{B}_{p,\infty}$ corresponding to $\mathrm{End}(E)$ and $\mathrm{End}(E')$. The *kernel ideal* of ϕ is defined as $I_\phi = \big\{\alpha \in \mathcal{O}_E \big| \ \alpha(\ker(\phi)) = 0\big\}$. Conversely, given I an $(\mathcal{O}_E, \mathcal{O}_{E'})$-ideal, it induces an isogeny $\phi_I : E \to F$ given by $\ker(\phi_I) = E[I] = \big\{P \in E \ \big| \ \alpha(P) = 0 \ \forall \alpha \in I\big\}$. The Deuring correspondence relates those different notions as follows:

supersingular j-invariants over $\mathbb{F}_{p^2}$	**maximal orders in $\mathbf{B}_{p,\infty}$**
$j(E)$	$\mathcal{O}_E$
$\phi \circ \psi$	$I_\psi I_\phi$
$\deg(\phi)$	$n(I_\phi)$
$\widehat{\phi}$	$\overline{I_\phi}$
$\psi_*\phi$	$[I_\psi]_* I_\phi = \frac{1}{n(I_\psi)}\overline{I_\psi}(I_\psi \cap I_\phi)$
$\gamma \in \mathrm{End}(E)$	$\mathcal{O}_E\gamma$

Kani's Lemma: Lastly, an important recent concept in Isogeny-Based Cryptography is Kani's Lemma [27], particularly its application in breaking SIDH as proposed in [5,32,41]. These works used Kani's Lemma to embed isogenies between elliptic curves into higher-dimensional isogenies. In this paper, we focus exclusively on principally polarized abelian varieties, omitting the detailed notion of polarization. For readers interested in the topic of polarization, we recommend Milne's book [33].

The only exception in our discussion is the notation for the dual of a high-dimensional isogeny ϕ, which we denote as $\tilde{\phi}$, referring to its polarized dual. Below, we provide Kani's Lemma as defined in [41, Lemma 3.2].

Lemma 1. *Let $f : A \to B$, $g : A \to A'$, $f' : A' \to B'$ and $g' : B \to B'$, be polarized separable isogenies such that $g' \circ f = f' \circ g$, with $\deg(f) = \deg(f')$ and $\deg(g) = \deg(g')$.*

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, "g"'] & B \arrow[d, "g'"] \arrow[dl, "g \circ \widetilde{f}"'] \\
A' \arrow[r, "f'"'] & B'
\end{tikzcd}$$

Then, the map $F : B \times A' \to A \times B'$ given by the matrix $\begin{pmatrix} \tilde{f} & -\tilde{g} \\ g' & f' \end{pmatrix}$ is a polarised separable isogeny with $\deg(F) = \deg(f) + \deg(g) = D$, $\ker(F) = \big\{\big(f(P), -g(P)\big) \big| \ P \in A[D]\big\}$ and $\ker(\widetilde{F}) = \big\{\big(-\tilde{g}(P), f'(P)\big) \big| \ P \in A'[D]\big\}$.

An important observation is that, given $\deg(F) = d_1 d_2$, we can then write $F = F_2 \circ F_1$ with $\deg(F_1) = d_1$ and $\deg(F_2) = d_2$ such that

$$\begin{tikzcd}
 & V & \\
B \times A' \arrow[ur, "F_1"] \arrow[rr, "F"] & & A \times B' \arrow[ul, "\widetilde{F_2}"']
\end{tikzcd}$$

$$\ker(F_1) = \Big\{ \big(f(P), g(P)\big) \Big| \; P \in A[d_1] \Big\} \;\; \& \;\; \ker(\widetilde{F_2}) = \Big\{ \big(\tilde{f}(P), g'(P)\big) \big| \; P \in B[d_2] \Big\}$$

Lastly, provided that $\deg(f)$ and $\deg(g)$ are co-prime, we can also define the kernel of F as $\ker(F) = \Big\{ \big([\deg(f)]P, g \circ \widetilde{f}(P)\big) \big| \; P \in B[D] \Big\}$. This property can be used to split a composition of isogeny and will be utilised throughout this paper.

2.2 Standard Algorithms

SQIPrime, even more profundly than SQISign and SQISignHD, heavenly relies on the different efficient representations [9, Definition 1] of isogenies and more specifically the kernel, ideal and high dimensional representations. To do so, it uses the following standard algorithms in Isogeny Based Cryptography:

- **KernelToIsogeny**: Takes as input E a supersingular curve and $K \in E[d]$ and return ϕ the isogeny of degree d whose kernel is generated by K together with E', its codomain. To do so, it uses Vélu's Formulas [44] and factorises ϕ as a composition of prime degree isogenies. To be efficient, it needs for d to be smooth.[1]
- **CanonicalTorsionBasis**: Takes as input E a supersingular curve and N an integer such that $N|(p^2-1)$ and return $\langle P, Q\rangle = E[N]$. To do so, it simply samples points at random in $E(\mathbb{F}_{p^2})$ or its quadratic twist and multiplies it by the right cofactor. To ensure that this method is deterministic, the sampling is performed deterministically.
- **PushEndRing** [9, Algorithm 8]: Takes as input $\mathfrak{O}_E$ an evaluation basis of $\mathrm{End}(E)$, $\varphi : E \to F$ an isogeny of degree d that is efficiently computable together with its ideal I_φ. It outputs $\mathfrak{O}_F$ a d-evaluation basis of $\mathrm{End}(F)$. An evaluation basis [9, Definition A.4.1] consist in an isomorphism between the endomorphism ring and a maximal order such that every element of the basis is efficiently computable [9, Definition 1.1.1].
- **KernelToIdeal** [9, Algorithm 9]: Takes as input $\mathfrak{O}_E$ a N-evaluation basis of $\mathrm{End}(E)$ and K a generator of the kernel of an isogeny ϕ of smooth degree d co-prime to N and return I_ϕ.
- **FullRepresentInteger** [30, Algorithm 4]: Takes as input a number $N > p$ and return $\gamma \in \mathcal{O}_0$ an endomorphism of E_0 such that $\gamma\overline{\gamma} = N$. To do so, it uses a modification of the **Cornacchia** algorithm[2], named the **Cornac-**

[1] Note that this algorithm, as presented here, is not optimal. Among the important improvements on those computations, see [14] and [3].

[2] Defined in [7], the **Cornacchia** algorithm solves efficiently equations of the form $x^2 + qy^2 = N$ with $x, y \in \mathbb{Z}$ provided that we know the factorisation of N.

chiaExtended [30, algorithm 1] that does not require knowledge of the factorization of N but at the cost of some bias over the distributions of its answers.

- **EvalTorsion** [9, Algorithm 11]: It takes as input $\mathfrak{O}_F$ an evaluation basis of $\mathrm{End}(F)$, $\rho_1 : F \to E$ of degree $d_1, \rho_2 : F \to E'$ of degree d_2, both efficiently computable isogenies together with their respective ideals I_1 and I_2. It also takes as input J an $(\mathcal{O}_E, \mathcal{O}_{E'})$-ideal of norm N co-prime to d_1 and d_2. It outputs $\phi_J(P)$, with P any point whose order is co-prime to $d_1 d_2$.
- **RandomEquivalentIdeal** [30, Algorithm 6]: It takes as input a $(\mathcal{O}_E, \mathcal{O}_F)$-ideal I and returns J another $(\mathcal{O}_E, \mathcal{O}_F)$-ideal such that $n(J)$ is a "small" prime, meaning that $n(J) \simeq \sqrt{p}$ with extremely high probability, as shown in [30, Lemma 3.2.3 & Lemma 3.2.4].
- **HDKernelToIsogeny**: This is an high dimensional equivalent to **KernelToIsogeny**. Depending on the dimension, it can be based upon Theta series [39,10,9], or over Kummer surfaces [42].

2.3 SQISign and SQISignHD

We previously introduced SQISign and SQISignHD as signature schemes, but a more accurate characterization would be to regard them as identification schemes at their core, based on Σ-protocols [25, Chapter 4]. These schemes are subsequently transformed into digital signature schemes using the Fiat-Samir transform [20], rendering them Universally Unforgeable under Chosen Message Attacks (UU-CMA) secure in the Random Oracle Model (ROM). Both SQISign and SQISignHD are Σ protocol-based identification schemes built upon the Deuring correspondence, hence the acronym SQIS for Short Quaternion Identification Scheme. Both protocols rely on the *one endomorphism problem*, a central challenge believed to be hard.

Problem 1. Let E be any supersingular curve defined over $\mathbb{F}_{p^2}$, find a nontrivial ($\alpha \notin \mathbb{Z}$) endomorphism of E.

The main idea behind SQISign and SQISignHD is to prove the knowledge of the endomorphism ring $\mathrm{End}(E_A)$ for E_A a supersingular curve. To do so, the idea is to use the fact that knowing $\mathrm{End}(E_A)$ enables the prover to find a connecting isogeny between E_A and any other curve E_2, provided that he also knows $\mathrm{End}(E_2)$. The idea is then to let E_2 be chosen as a challenge by the verifier in such a way that the prover can retrieve $\mathrm{End}(E_2)$ and respond the connecting isogeny that can be easily verified. The main difference between SQISign and SQISignHD consist in how this connecting isogeny is computed and represented. The respective architecture of SQISign and SQISignHD are given in Figure 1.

SQISign: To construct σ the connecting isogeny, SQISign uses a variant of the **KLPT** [28] named the **SigningKLPT** [15, Algorithm 5]. The ideal I_σ it retrieves is smooth, as its norm is a large power of 2 of size $O(p^{15/4})$. To be efficiently computed, σ is represented as a composition of isogenies with rational

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "\psi"] \arrow[d, dashed, "\tau"'] & E_1 \arrow[d, "\varphi"] \\
E_A \arrow[r, "\sigma"'] & E_2
\end{tikzcd}
\qquad
\begin{tikzcd}
E_0 \arrow[r, dashed, "\psi"] \arrow[d, dashed, bend right, "\tau'"'] \arrow[d, dashed, bend left, "\tau"] & E_1 \arrow[d, "\sigma"] \\
E_A \arrow[r, "\varphi"'] & E_2
\end{tikzcd}$$

Fig. 1. Diagrams of SQISign (left) and SQISignHD (right). The prover is in blue and the verifier is in red. Dashed isogenies are secrets.

kernel generator. Transcribing I_σ to these kernels is done efficiently using **IdealToIsogeny** [16, Algo. 7] by setting the prime p of SQISign to be such that $2^\ell T | p^2 - 1$ with $T \simeq p^{5/4}$ and T smooth. Finding such primes is *difficult* and T often has prime factors in the order of 10^3. Those big factors significantly slow the signing procedure, as several T isogenies have to be computed throughout **IdealToIsogeny**. On the other hand, the verification of SQISign is very efficient, as it essentially consists in computing a sequence of isogenies of degree 2^ℓ from their kernels. SQISign is performed as such:

- KeyGen: Compute $\tau : E_0 \to E_A$ together with its corresponding ideal I_τ. E_A is the public key, while τ is the secret key. E_A is the domain of the response isogeny.
- Commit: The prover computes $\psi : E_0 \to E_1$ together with its corresponding ideal I_ψ. It gives ψ to the verifier.
- Challenge: The verifier then computes a challenge isogeny $\varphi : E_1 \to E_2$ and sends it to the prover. E_2 is the codomain for the answer isogeny.
- Response: Using its knowledge of ψ, the prover uses **KernelToIdeal** to compute I_φ. Then, using the **SigningKLPT** and **IdealToIsogeny**, the prover constructs an isogeny $\sigma : E_2 \to E_1$ different from $\varphi \circ \psi \circ \widehat{\tau}$ and gives σ as a response to the verifier.
- Verify: The verifier then checks that the received isogeny is valid using **KernelToIsogeny**.

SQISignHD: On the other hand, SQISignHD uses the **RandomEquivalentIdeal** to compute σ. The response isogeny is therefore short $\widetilde{O}(\sqrt{p})$ but not smooth. It is then given to the verifier using high dimension representation [40]. This shift to high dimension isogenies considerably speeds up the signature part of SQISignHD but shifts most of the expensive computation to the verification that has to use Kani's Lemma in dimension 4. To be efficient, SQISignHD uses "SIDH-like" prime, that are easy to find. SQISignHD is thus performed as such:

- KeyGen: Compute $\tau, \tau' : E_0 \to E_A$ using **DoublePath** [9, Section 3.3] together with its corresponding ideal I_τ. E_A is the public key, while τ is the secret key.

– Commit: The prover computes an isogeny $\psi : E_0 \to E_1$ of odd degree with **DoublePath** together with its ideal I_ψ and shares E_1. This curve is the domain of the response.
– Challenge: The verifier computes a challenge isogeny $\varphi : E_A \to E_2$ and sends it to the prover. E_2 is the codomain for the answer isogeny.
– Response: Using **RandomEquivalentIdeal**, the prover constructs an isogeny $\sigma : E_1 \to E_2$ different from $\varphi \circ \tau \circ \hat{\psi}$, evaluate it using τ', ψ' and **EvalTorsion** and gives this evaluation of σ as a response to the verifier.
– Verify: The verifier then checks that the received isogeny is valid using Kani's Lemma in dimension 4.

3 Introduced Techniques

Before jumping into SQIPrime, we detail two new techniques that we will use to construct our variant of SQISignHD.

1. The first tool is called **KaniDoublePath**, a variant of **DoublePath** [9, Section 3.3] that uses Kani's Lemma to sample two (eventually non-smooth) isogenies between E_0 and E_A of co-prime degrees. This algorithm is a modification of the **RandIsogImages** [35, Algorithm 2], as it additionally computes the corresponding ideals of these isogenies. We also detail **ExtKaniDoublePath**, a variant of the former that relies on endomorphisms of greater norm.
2. The second is a method to compute, given K a generator of the kernel of an isogeny, the corresponding ideal even when the degree of this isogeny is non-smooth. This method is an adaptation of the work of Leroux [31] and it allows us to use large non-smooth degree isogenies as challenge isogeny in SQIPrime.

3.1 KaniDoublePath

The main idea behind **KaniDoublePath** is, likewise to the **DoublePath** algorithm, to construct two isogenies of co-prime degree between E_0 and another supersingular curve E. The main interest of **KaniDoublePath** lies in the fact that those isogenies are not necessary smooth.

To perform the **KaniDoublePath**, we first use **FullRepresentInteger** to find an endomorphism $\gamma \in \mathrm{End}(E_0)$ such that $\deg(\gamma) = \ell(N - \ell)$ with ℓ, N co-prime and N smooth. We can decompose γ as $\gamma = \rho \circ \tau$ with $\deg \tau = \ell$ and $\deg \rho = N - \ell$. Using Kani's Lemma, we compute the dimension 2 isogeny F given by the following diagram and kernel:

$$\begin{tikzcd}
E \arrow[r, "\hat{\tau}"] \arrow[d, "\rho"'] & E_0 \arrow[d, "\hat{\tau}_*\rho"] \arrow[dl, "\gamma"'] \\
E_0 \arrow[r, "\rho_*\hat{\tau}"'] & E'
\end{tikzcd}$$

$$\ker(F) = \left\{ \big([\ell](P), \gamma(P)\big) \,\middle|\, P \in E_0[N] \right\}$$

We can therefore evaluate both τ and $\hat{\rho}$ at any points of E_0. Additionally, we also retrieve I_τ and I_ρ the ideal corresponding to τ and ρ as follows:

$$I_\tau = \mathcal{O}_0\overline{\gamma} + \mathcal{O}_0\ell, \qquad I_\rho = \frac{\overline{I_\tau}\mathcal{O}_0\gamma}{\ell}.$$

The full process is summarised in Algorithm 1. In practice, if $N-\ell$ is not way larger than p, it may happen that for some curve E which is ℓ-isogenous to E_0, there exists no isogeny of degree $N-\ell$ between E_0 and E. We describe **ExtKaniDoublePath**, a variation of **KaniDoublePath** in which the degree of the byproduct isogeny ρ is larger, hence increasing the chances that there exist such an isogeny between any curve E which is ℓ-isogenous to E_0.

Algorithm 1 KaniDoublePath

Input: $\mathfrak{O}_0$ an evaluation basis of $\mathrm{End}(E_0)$ with $\langle P, Q\rangle$ a basis of $E_0[N]$ and ℓ such that $\gcd(\ell, N) = 1$ and $\ell(N-\ell) > p$ with N smooth

Output: τ,$\widehat{\rho} : E_0 \to E$ isogenies of respective degree ℓ and $N-\ell$ given as dimension 2 isogenies, together with I_τ and $I_{\widehat{\rho}}$ their ideals.

```
$\gamma \leftarrow$ FullRepresentInteger$(\mathfrak{O}_0, \ell(N-\ell))$
$\mathsf{B} \leftarrow \left\{\big([\ell]P, \gamma(P)\big), \big([\ell]Q, \gamma(Q)\big)\right\}$
$F \leftarrow$ HDKernelToIsogeny$(E_0^2, \mathsf{B})$
$I_\tau \leftarrow \mathcal{O}_0\overline{\gamma} + \mathcal{O}_0\ell$
$I_{\hat{\rho}} \leftarrow \frac{1}{\ell}\mathcal{O}_0\overline{\gamma}I_\tau$
return $F, I_\tau, I_{\hat{\rho}}$                ▷ $\tau(-) = F(-,0)_1$ and $\hat{\rho}(-) = -F(0,-)_1$
```

The concept behind **ExtKaniDoublePath** closely resembles that of **KaniDoublePath**, albeit with a slight variation. Instead of operating with $\gamma \in \mathrm{End}(E_0)$ of norm $\ell(N-\ell)$, **ExtKaniDoublePath** involves working with $\gamma \in \mathrm{End}(E_0)$ of norm $\ell(N'-\ell)\big(N - \ell(N'-\ell)\big)$, where N and N' are smooth. Consequently, we have $\deg(\rho) = (N'-\ell)\big(N-\ell(N'-\ell)\big)$. Both τ and $\widehat{\rho}$ are computed by applying Kani's Lemma twice:

1. Initially, we decompose γ into $\gamma = \rho_1 \circ \rho_2 \circ \tau$ where ρ_1 has degree $N - \ell(N'-\ell)$ and $\rho_2 \circ \tau$ has degree $\ell(N'-\ell)$, and we assess $\rho_2 \circ \tau$ over $E_0[N']$.
2. Subsequently, we further break down $\rho_2 \circ \tau$ of degree $\ell(N'-\ell)$ into τ and $\widehat{\rho_2}$ of degree ℓ and $N'-\ell$ respectively, allowing for the computation of $\widehat{\rho}$ as a composition of $\widehat{\rho_1}$ and $\widehat{\rho_2}$.

You may find below the commutative diagram of the **ExtKaniDoublePath**. The first use of Kani's Lemma is in blue and the second is in red.

$$\begin{tikzcd}
 & E_0 \arrow[rr, "{[\widehat{\rho_2\circ\tau}]_*\rho_1}"] \arrow[d, "\rho_2\circ\tau"] \arrow[dr, "\widehat{\tau}_*\rho_2"] \arrow[ddrr, bend left, "\gamma"] & & E_1' \\
E \arrow[ur, "\widehat{\tau}"] \arrow[dr, "\rho_2"'] & & E' & \\
 & E_1 \arrow[ur, "{[\rho_2]_*\widehat{\tau}}"] \arrow[rr, "\rho_1"'] & & E_0 \arrow[uu, "{[\rho_1]_*\widehat{\rho_2\circ\tau}}"']
\end{tikzcd}$$

Algorithm 2 ExtKaniDoublePath

Input: $\mathfrak{O}_0$ an evaluation basis of $\mathrm{End}(E_0)$ with $\langle P, Q\rangle$ a basis of $E_0[N]$, $\langle P', Q'\rangle$ a basis of $E_0[N']$ and ℓ such that $\gcd(\ell, N) = \gcd(\ell, N') = 1$ and $\ell(N'-\ell)(N-\ell(N'-\ell)) > p$ with N, N' smooth

Output: τ,$\widehat{\rho} : E_0 \to E$ isogenies of respective degree ℓ and $(N'-\ell)(N-\ell(N'-\ell))$ given as dimension 2 isogenies, together with I_τ and $I_{\widehat{\rho}}$ their ideals.

```
$\gamma \leftarrow$ FullRepresentInteger$(\mathfrak{O}_0, \ell(N'-\ell)(N-\ell(N'-\ell))$
$\mathsf{B}_1 \leftarrow \{([\ell(N'-\ell)]P, \gamma(P)), ([\ell(N'-\ell)]Q, \gamma(Q))\}$
$F_1 \leftarrow$ HDKernelToIsogeny$(E_0^2, \mathsf{B}_1)$        ▷ $\tau \circ \rho_2(-) = F_1(-, 0)_1$
Find $E_1$ =codomain$(\hat{\rho}_1)$.
$\mathsf{B}_2 \leftarrow \{([N'-\ell]P', \tau\circ\rho_2(P')), ([N'-\ell]Q', \tau\circ\rho_2(Q'))\}$
$F_2 \leftarrow$ HDKernelToIsogeny$(E_0 \times E_1, \mathsf{B}_2)$
$I_\tau \leftarrow \mathcal{O}_0\overline{\gamma} + \mathcal{O}_0\ell$
$I_{\hat{\rho}} \leftarrow \frac{1}{\ell}\mathcal{O}_0\overline{\gamma}I_\tau$
return $F, I_\tau, I_{\hat{\rho}}$        ▷ $\tau(-) = -F_2(0,-)_1$ and $\hat{\rho}(-) = F_2(-,0)_1 \circ -F_1(0,-)_1$
```

We will rely on the following assumptions when discussing the security of SQIPrime.

Assumption 1. *The distribution of E the codomain of τ and $\hat{\rho}$, outputted by **KaniDoublePath** (N, P, Q, ℓ) with ℓ a random prime smaller than $\sqrt{p}$ is computationally undistinguishable from the distribution of E sampled randomly among all supersingular curves.*

Assumption 2. *The distribution of $\tau : E_0 \to E$ an isogeny given by outputted by **ExtKaniDoublePath** $(N, P, Q, N', P', Q', \ell)$ with ℓ a random prime of size smaller than $\sqrt{p}$ is computationally undistinguishable from the distribution of τ sampled randomly among isogeny of degree ℓ and of domain E_0.*

3.2 KernelToIdeal for generic degree isogenies

Looking at the details of **KernelToIdeal** [9, Algorithm 9], we see that it makes extensive usage of discrete logarithms over $E[d]$, with d being the degree of the isogeny for which the representing ideal is being computed. To be efficient, this method requires d being smooth. We therefore need another method for isogenies of generic degree. The idea proposed by Leroux in [31] is to use the knowledge of the endomorphism ring of E to construct a *precomputed basis* of $E[d]$.

Definition 1. *Let E be any supersingular curve. The tuple (P, Q, ι, I_P) is a* ***precomputed basis*** *of $E[d]$ if the following conditions are satisfied:*

- *$P, Q \in E$ form a basis of $E[d]$.*
- *$\iota \in End(E)$ and $\iota(P) = Q$.*
- *I_P is the ideal corresponding to the isogeny of kernel $\langle P \rangle$.*

The knowledge of $\mathfrak{O}_E$ and of an evaluation basis of $\mathrm{End}(E)$ enables us to efficiently construct a precomputed basis using the **FindPrecomputedBasis** algorithm (Algorithm 3), proposed in [31]. Using a precomputed basis, we can compute ideals from a kernel generator $K \in E[d]$ using the following lemma.

Algorithm 3 FindPrecomputedBasis

```
Input: $\mathfrak{O}_E = \big(\{b_i\}_{i=1}^4, \delta\big)$ an evaluation basis of $E$ with $d$ an integer
Output: $(P, Q, \iota, I_P)$ a precomputed basis of $E[d]$.
Sample $R \in_{\$} E[d]$
Sample $\alpha \in_{\$} \mathcal{O}_E$ such that gcd $\big(n(\alpha), d^2\big) = d$
if $\delta^{-1}(\alpha)(R) = 0$ do try with new $R$.
$P \leftarrow \delta^{-1}(\alpha)(R)$
$I_P \leftarrow \mathcal{O}_E\overline{\alpha} + \mathcal{O}_E d$
Sample $\iota \in_{\$} \mathcal{O}_E$ such that gcd$(n(\iota), d) = 1$
if $e_d(P, \delta^{-1}(\iota)(P)) \overset{?}{=} 1$ do sample new $\iota$.        ▷ Ensures they are not colinear
return $P, \delta^{-1}(\iota)(P), \delta^{-1}(\iota), I_P$
```

Lemma 2. *Let (P, Q, ι, I_P) be a precomputed basis of $E[d]$ and let $K = [a]P + [b]Q$ be a point in $E[d]$. Then the representing ideal of the isogeny $\phi_K : E \to E/\langle K \rangle$ is given by*

$$I_K = [a + b\delta(\iota)]_* I_P, \qquad \textit{where } \delta : End(E) \cong \mathcal{O}_E.$$

Proof. This comes from the fact that $\langle K \rangle = \langle [a]P + [b]Q \rangle = \langle [a]P + [b]\iota(P) \rangle = [a + b\iota]\langle P \rangle$, meaning that $\phi_K = [a + b\delta(\iota)]_* \phi_P$. We then get the desired result through the Deuring correspondence. □

We can thus compute the ideals corresponding to a kernel of generic order. Nevertheless, the method that we presented here requires knowing $\mathfrak{O}_E$. Most of the time, the curve E is obtained by computing an isogeny $\phi : E_0 \to E$. With the knowledge of $\mathfrak{O}_0$ and $\phi : E_0 \to E$, one can recover $\mathfrak{O}_E$, and hence determine a precomputed basis of $E[d]$ using the **FindPrecomputedBasis** algorithm. Even though this is already efficient, in Corollary 1, we describe a faster and more convenient method to translate a kernel generator $K \in E[d]$ into an ideal knowing a precomputed basis of $E_0[d]$, $\phi : E_0 \to E$ of degree co-prime to d and its corresponding ideal I_ϕ.

Corollary 1. *Let (P, Q, ι, I_P) be a precomputed basis of $E_0[d]$ and let $\phi : E_0 \to E$ be an isogeny of degree q with corresponding ideal I_ϕ such that d and q are co-prime. Let $S, T \in E$ be the respective images of P and Q by ϕ and let $K = [a]S + [b]T$ be a point in $E[d]$. Then,*

$$I_K = \big[(a + b\delta(\iota))I_\phi\big]_* I_P$$

Proof. Similarly to Lemma 2, we have that

$$\begin{aligned}
\langle K\rangle &= [q]\langle K\rangle \\
&= \phi\hat{\phi}\,\langle [a]S + [b]T\rangle \\
&= \phi\langle [a]\widehat{\phi}(S) + [b]\widehat{\phi}(T)\rangle \\
&= \phi\langle [aq]P + [qd]Q\rangle \\
&= \phi\langle [a]P + [b]Q\rangle \\
&= \phi\langle [a]P + [b]\iota(P)\rangle \\
&= \phi \circ [a + b\iota]\langle P\rangle
\end{aligned}$$

$$\text{i.e. } \phi_K = [\phi \circ (a + b\iota)]_* \phi_P \text{ and thus } I_K = [(a + b\delta(\iota))I_\phi]_* I_P$$

□

It's worth noting that [31] proposes using ϕ to directly generate a precomputed basis over E. Specifically, if (P, Q, ι, I_P) represents a precomputed basis over $E_0[d]$, then $\big(\phi(P), [\deg(\phi)]\phi(Q), \theta, [I_\phi]_* I_P\big)$ constitutes a precomputed basis of $E[d]$ with $\theta = \phi \circ \iota \circ \widehat{\phi}$. The significant advantage of Corollary 1 lies in its exclusive use of endomorphisms over E_0 rather than over E. This characteristic aligns more closely with our requirements in SQIPrime, making it better suited for our purposes.

4 SQIPrime4D: SQIPrime in dimension 4

Now that we are familiar with the architecture behind SQISign and SQISignHD, and that we have introduced and explained our new techniques, we can construct SQIPrime. As previously stated in the introduction, SQIPrime4D further expands the use of Kani's Lemma to both KeyGen and Commit. Moreover, the challenge isogeny has non-smooth degree. Only the kernel of the challenge isogeny is sampled by the verifier. The challenge isogeny $\varphi : E_A \to E_2$ is computed by the prover, who then appends the usual response isogeny $\sigma : E_2 \to E_1$ to it to get $\kappa := \sigma \circ \varphi : E_A \to E_1$. The high dimensional representation of κ is returned to the verifier. Figure 2 isllustrates the architecture of SQIPrime4D.

The public parameters of SQIPrime4D are defined as:

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "\psi"] \arrow[d, dashed, "\tau"'] & E_1 \\
E_A \arrow[ru, "\kappa"] \arrow[r, "\varphi"'] & E_2 \arrow[u, dashed, "\sigma"']
\end{tikzcd}$$

Fig. 2. Diagram of SQIPrime4D, prover in blue and verifier in red. Dashed isogenies are not shared.

- p a prime number of the form $2^\alpha f - 1 \simeq 2^{2\lambda}$ and such that $p = 2Nq + 1$, with $q \simeq 2^\lambda$. We discuss in Section 8 how to efficiently compute such primes.
- P_0, Q_0 a basis of $E_0[2^\alpha]$.
- $(P, Q, \iota, I_{[N]P})$ which is almost a precomputed basis over $E_0[Nq]$. (It is if we use I_P instead of $I_{[N]P}$ but this ideal is more adapted to SQIPrime.
- β an integer such that $2^\beta - q\sqrt{p}\log(p) \geq 0$.

They are constructed using the Setup algorithm described in Algorithm 4.

Algorithm 4 SQIPrime4D.Setup

```
Input: $1^\lambda$
Output: $\mathsf{pp} = \big(p, \alpha, q, N, (P_0, Q_0), (P, Q, \iota, I_{[N]P}), \beta\big)$
Take $p$ a prime of the form $p = 2^\alpha f - 1 \simeq 2^{2\lambda}$ such that $p - 1 = 2Nq$ with $q \simeq 2^\lambda$
    prime and $N$ co-prime to $q$.
$P_0, Q_0 \leftarrow$ CanonicalTorsionBasis$(E_0, 2^\alpha)$
$(P, Q, \iota, I_P) \leftarrow$ FindprecomputedBasis$(\mathfrak{O}_0, qN)$
Compute $I_{[N]P} = I_P + \mathcal{O}_0 q$
$\beta \leftarrow \lceil \log_2(p)/2 + \log_2(q) + \log_2\log_2(p) \rceil$
$\mathsf{pp} \leftarrow \big(p, (P_0, Q_0), (P, Q, \iota, I_{[N]P}), \beta\big)$
return pp
```

At a high level, the subroutines of SQIPrime4D are as follows.

- KeyGen: Compute $\tau : E_0 \to E_A$ together with its corresponding ideal I_τ using **KaniDoublePath**. Additionally, compute a matrix $\mathbf{M}$ and use it to mask the image through τ of a precomputed basis of degree qN, with $q \simeq 2^\lambda$.The curve E_A and the masked basis form the public key, while τ, I_τ and the matrix $\mathbf{M}$ form the secret key.
- Commit: The prover computes an isogeny $\psi : E_0 \to E_1$ with **KaniDoublePath** together with its ideal I_ψ and shares E_1.

- Challenge: The verifier samples a random scalar $a \in \mathbb{Z}_q$ and returns it to the prover. This scalar defines a point $C_a = P + [a]Q$ where P, Q is a specified basis of $E_A[q]$.
- Response: Using the precomputed basis over E_0 and its knowledge of I_τ, the prover retrieves I_φ, the ideal corresponding to the challenge isogeny $\varphi : E_A \to E_2$ whose kernel is given by $\ker(\varphi) = \langle C_a \rangle$. Using **RandomEquivalentIdeal**, he computes a short $(\mathcal{O}_2, \mathcal{O}_1)$-ideal I_σ corresponding to an isogeny $\sigma : E_2 \to E_1$, and constructs $\kappa = \sigma \circ \varphi$, evaluates it using **EvalTorsion** and sends this evaluation of κ as a response to the verifier.
- Verify: The verifier receives κ and checks using Kani's Lemma that it is valid by verifying that it is an isogeny from E_A to E_1 and that $\kappa(C_a) = 0$.

4.1 Key Generation and Commitment

Both key generation and commitment consist essentially in using **KaniDoublePath**. We take a random prime ℓ smaller than $\sqrt{p}$ and use the **KaniDoublePath** with an endomorphism of norm $\ell(2^\alpha - \ell)$ to retrieve τ in the case of **SQIPrime.KeyGen** (Algorithm 5) and ψ in **SQIPrime.Commit**. (Algorithm 6). The only significant differences between the key and commitment generation is that during the key generation, we additionally compute a masked basis of $E_A[Nq]$. To do so, we compute the image of (P, Q) through the isogeny τ and use a random matrix $\mathbf{M} \in \mathrm{GL}_2(Nq)$ to mask the torsion points. Note that this masking makes of R, S a random basis of $E_A[Nq]$.

Algorithm 5 SQIPrime4D.KeyGen

```
Input: pp = $\big(p, \alpha, q, N, (P_0, Q_0), (P, Q, \iota, I_{[N]P}), \beta\big)$
Output: sk = $\big(\mathsf{F_A}, I_\tau, \mathbf{M}\big)$, pk = $\big(E_A, (R, S)\big)$
1: Sample $\ell_A$ a random prime smaller than $\sqrt{p}$ such that $\ell_A \neq q$.
2: $\mathsf{F_A}, I_\tau, * \leftarrow$ KaniDoublePath$(2^\alpha, P_0, Q_0, \ell_A)$
3: Compute $E_A$.
4: Sample $\mathbf{M} \in_\$ \mathrm{GL}_2(Nq)$
5: $\binom{R}{S} \leftarrow \mathbf{M}\binom{\tau(P)}{\tau(Q)}$        ▷ $\tau(-) = F_A(-, 0)_1$
6: return $(\mathsf{F_A}, I_\tau, \mathbf{M})$, $\big(E_A, (R, S)\big)$
```

Algorithm 6 SQIPrime4D.Commit

```
Input: pp = $\big(p, \alpha, q, N, (P_0, Q_0), (P, Q, \iota, I_{[N]P}), \beta\big)$
Output: sec = $\big(\mathsf{F_1}, I_\psi\big)$, pub = $(E_1)$
1: Take $\ell_1$ a random prime smaller than $\sqrt{p}$ such that $\ell_1 \neq q$.
2: $\mathsf{F_1}, I_\psi, * \leftarrow$ KaniDoublePath$(2^\alpha, P_0, Q_0, \ell_1)$
3: Compute $E_1$
4: return $\big(\mathsf{F_1}, I_\psi\big)$, $\big(E_1\big)$        ▷ $\psi(-) = F_1(-, 0)_1$
```

4.2 Challenge and Response

Challenge As touched earlier, our challenge is significantly different from the challenge of SQISign and SQISignHD, as the evaluation of the challenge isogeny has been moved from the verifier to the prover. This adjustment is necessary since the verifier lacks an efficient mean to evaluate this isogeny, as it only has access to the kernel representation of φ, whose degree is not smooth. In idea, the prover uses the ideal representation to construct a high dimension representation of φ that is then sent to the verifier together with the high dimension representation of the answer isogeny σ. Thus, instead of providing an isogeny of smooth degree, the challenger simply sends a challenge point $C_a \in E_A[q]$. This point is given as $a \in \mathbb{Z}_q$ such that $C_a = [N](R + [a]S)$ where R, S is the basis of $E_A[Nq]$ included in the public key. This point is the generator of the kernel of $\varphi : E_A \to E/\langle C_a \rangle = E_2$. We have $q \simeq 2^\lambda$ possible challenge isogenies.

Response In line with SQISignHD, our objective is to compute an isogeny $\sigma : E_2 \to E_1$. However, the verifier lacks knowledge of E_2. An initial idea might be to provide the verifier with an HD representation of φ, allowing him to check that the kernels match. However, this approach requires knowledge of a map between E_0 and E_2 (or E_A and E_2), which is challenging to construct.[3] Instead of sending σ and φ separately, the idea is to send $\kappa = \sigma \circ \varphi$ and use Kani's Lemma over κ to prove that κ factors through φ, utilising the fact that $\ker(\kappa) \cap E_A[q] = \ker(\varphi)$.

First, one adapts Corollary 1 to compute $I_{C_a} = I_\varphi$. Upon receiving the challenge $\mathsf{Chal} = a$, the prover finds $b, c \in \mathbb{Z}_q$ such that $C_a = [N]\big([b]\tau(P) + [c]\tau(Q)\big)$. These scalars are given by $\binom{b}{c} = \mathbf{M}^{-1}\binom{1}{a}$.[4] One then recovers I_{C_a} as

$$I_{C_a} = \big[\big(b + c\delta(\iota)\big)I_\tau\big]_* I_{[N]P}$$

He then computes the $(\mathcal{O}_2, \mathcal{O}_1)$-ideal $\overline{I_{C_a} I_\tau} I_\varphi$ and finds an equivalent short $(\mathcal{O}_2, \mathcal{O}_1)$-ideal J using **RandomEquivalentIdeal**. The ideal J corresponds to an isogeny $\sigma : E_2 \to E_1$ of degree d that closes the diagram in Figure 2, with d such that $2^\beta - qd$ can be written as the sum of two square. One sufficient condition is to ask for $2^\beta - qd = 1 \mod 4$ and to be prime. Following the discussion in [9, Section 4.2] and by using the sampling method proposed in [9, Section E.2], we expect to find a valid J after sampling $O(1/\lambda)$ times.

The final response is composed of the evaluation of the isogeny $\kappa = \sigma \circ \varphi$ on $E_A[2^\alpha]$ and on the point $C_2 = [a]R - S$, together with the degree d of σ. To do so, one generates a basis of $E_A[2^\alpha]$ using **CanonicalTorsionBasis**, one uses **EvalTorsion** to evaluate κ on the generated basis and C_2. The point $\kappa(C_2)$ is used to ensure the soundness of our verification. It is important to note that C_2 is such that $\langle C_a, [N]C_2 \rangle = E_A[q]$.

[3] We could use the KLPT algorithm followed by the **IdealToKernel** algorithm, but avoiding these algorithms was a primary motivation behind the development of SQISignHD.

[4] Using $\binom{b}{c} = \det(\mathbf{M})\mathbf{M}^{-1}\binom{1}{a}$ is also valid and simplifies the computations.

Algorithm 7 SQIPrime4D.Response

```
Input: $\mathsf{pp} = \big(p, \alpha, q, N(P_0, Q_0), (P, Q, \iota, I_{[N]P}), \beta\big)$, $\mathsf{sk} = (\mathsf{F_A}, I_\tau, \mathbf{M})$, $\mathsf{sec} = (\mathsf{F}_1, I_\varphi), \mathsf{chal} = a$.
Output: $\mathsf{res} = (T, U, V, d)$ with $T, U \in E_1[2^\alpha]$, $V \in E_1[Nq]$ and $d$ the degree of $\sigma$.
1: $\binom{b}{c} \leftarrow \det(\mathbf{M})\mathbf{M}^{-1}\binom{1}{a}$
2: $I_{C_a} \leftarrow [(b + c\iota)I_\tau]_* I_{[N]P}$
3: $J \leftarrow$ RandomEquivalentIdeal$(\overline{I_{C_a} I_\tau} I_\psi)$          $d \leftarrow n(J)$
4: check if $2^\beta - dq = 1 \mod 4$ and is prime. If not, go back to line 3.
5: $X, Y \leftarrow$ CanonicalTorsionBasis$(E_A, 2^\alpha)$
6: $C_2 \leftarrow [a]R - S$
7: Define $\tau = (F_A(-, 0))_1$ and $\psi = (F_1(-, 0))_1$.
8: $T, U, V \leftarrow$ EvalTorsion$\Big(\mathfrak{O}_0, \tau, I_\tau, \psi, I_\psi, I_{C_a} J, qd, \{X, Y, C_2\}\Big)$
9: return $\mathsf{res} = (S, T, U, d)$
```

4.3 Verification

Upon receiving T, U, V, d, we want to verify that the following statement holds:

> The torsion points we received define a high dimensional representation of an isogeny $\kappa : E_A \to E_1$ of degree dq such that the isogeny κ factors through φ, meaning that $\ker(\kappa)[q] = \langle C_a \rangle$.

To perform this verification efficiently, we will use Kani's Lemma with the following diagram:

$$\begin{tikzcd}
E_A^2 \arrow[r, "\Sigma"] \arrow[d, "\gamma"'] & E_1^2 \arrow[d, "\gamma"] \\
E_A^2 \arrow[r, "\Sigma"] & E_1^2
\end{tikzcd}$$

where $\gamma := \begin{pmatrix} a_1 & -a_2 \\ a_2 & a_1 \end{pmatrix}$ such that $\deg(\gamma) = \sum_{i=1}^{2} a_i^2$; $\Sigma := \operatorname{diag}(\kappa, \kappa)$. If the parameters allow us to always have enough torsion, that is we always have $dq < 2^\alpha$ or equivalently $\beta = \alpha$, then F can be computed on one go and its kernel is given by $\ker(F) = \big\{\big(\Sigma(P), -\gamma(P)\big) \big|\ P \in E_A^2[2^\beta]\big\}$. If the parameters do not allow this, then we split the isogeny F into two isogenies F_1 and F_2 where $F = F_2 \circ F_1$ and $\deg F_i = 2^{\beta_i}$ $(\beta_1 + \beta_2 = \beta)$, $\ker(F_1) = \big\{\big(\Sigma(P), -\gamma(P)\big) \big|\ P \in E_A^2[2^{\beta_1}]\big\}$ and $\ker(\widetilde{F_2}) = \big\{\big(-\tilde{\gamma}(P), \Sigma(P)\big) \big|\ P \in E_A^2[2^{\beta_2}]\big\}$, similarly to SQISignHD[5]. We then use the following property:

Let $X \in E_A$ be a point of odd order, then:

[5] A slight change in the prime used in SQISignHD was suggested in [21] in order to avoid splitting the high dimensional isogeny, in the hope for a better efficiency, but we are not aware of any implementation of this variant.

$$F\begin{pmatrix}0\\0\\X\\0\end{pmatrix}=\begin{pmatrix}[a_1]X\\[-a_2]X\\Y\\0\end{pmatrix}\iff[2^{\beta_2}]F_1\begin{pmatrix}0\\0\\X\\0\end{pmatrix}=\widetilde{F_2}\begin{pmatrix}[a_1]X\\[-a_2]X\\Y\\0\end{pmatrix}$$

We use this equivalence on the two points C_a and C_2 of respective order q and Nq.

Algorithm 8 SQIPrime4D.Verify

```
Input: $\mathsf{pp} = \big(p, (P_0, Q_0), (P, Q, \iota, I_{[N]P}), \beta\big), \mathsf{pk} = (E_A, R, S), \mathsf{com} = E_1, \mathsf{chal} = a, \mathsf{res} = (T, U, V, d)$
Output: 0 or 1
if one of the points $S, T, U$ is not in $E_1$ do return 0
$\beta_1 \leftarrow \lfloor \frac{\beta}{2} \rfloor, \beta_2 \leftarrow \lceil \frac{\beta}{2} \rceil$, $k_1 \leftarrow 2^{\alpha-\beta_1}$, $k_2 \leftarrow 2^{\alpha-\beta_2}$
$(a_1, a_2) \leftarrow$ Cornacchia$(2^\beta - qd)$
Compute $\gamma$ and $\tilde{\gamma}$
Compute $\{P_{i,j}\}_{0\leqslant i,j\leqslant 2,2}$ a basis of $E_A^2[2^\alpha]$    ▷ Using CanonicalTorsionBasis
$\mathsf{B}_1 \leftarrow \big\{\big([k_1]\Sigma(P_{i,j}), [-k_1]\gamma(P_{i,j})\big)\big\}_{0\leqslant i,j\leqslant 2,2}$    ▷ $\Sigma(P_{i,j})$ computed using $T, U$
$\mathsf{B}_2 \leftarrow \big\{\big([-k_2]\tilde{\gamma}(P_{i,j}), [k_2]\Sigma(P_{i,j})\big)\big\}_{0\leqslant i,j\leqslant 2,2}$
$F_1 \leftarrow$ HDKernelToIsogeny$(\mathsf{B}_1)$
$\tilde{F_2} \leftarrow$ HDKernelToIsogeny$(\mathsf{B}_2)$
if codomain$(F_1) \neq$ codomain$(\tilde{F_2})$ do return 0    ▷ Do as [9, Section F.3]
$C_a \leftarrow [N](R + [a]S)$, $C_2 \leftarrow ([a]R - S)$
$b_1 \leftarrow [2^{\beta_2}]F_1(0, 0, C_a, 0) \stackrel{?}{=} \tilde{F_2}([a_1]C_a, [-a_2]C_a, 0, 0)$
$b_2 \leftarrow [2^{\beta_2}]F_1(0, 0, C_2, 0) \stackrel{?}{=} \tilde{F_2}([a_1]C_2, [-a_2]C_2, V, 0)$ and $[N]U \neq 0$.
return $b_1 \wedge b_2$
```

Proposition 1. *Let* $\mathsf{pp}, \mathsf{pk}, \mathsf{com}, \mathsf{chal}$ *be a valid public key, commitment, and challenge of SQIPrime4D and let* P, Q *be the canonical basis of* $E_A[2^\alpha]$*. Let* $\overline{\mathsf{Res}}$ *be a potential response.*

SQIPrime4D. Verify$(\mathsf{pp}, \mathsf{pk}, \mathsf{com}, \mathsf{chal}, \overline{\mathsf{Res}}) = 1$ *implies that* $\overline{\mathsf{Res}} = (\overline{T}, \overline{U}, \overline{V}, \overline{d})$ *is such that:*

- $(P, Q, \overline{T}, \overline{U})$ *is a high dimension representation of an isogeny* $\kappa : E_A \to E_1$ *of degree* $q\overline{d}$.
- $\ker(\kappa) \cap E[q] = \langle C_a \rangle$.

Proof. Our proof takes inspiration from [9, Section E.5]. Indeed, if we assume that **SQIPrime.Verify**$(\mathsf{pp}, \mathsf{pk}, \mathsf{pub}, \mathsf{chal}, \overline{\mathsf{Res}}) = 1$. Then, this means that $\overline{T}, \overline{U}, \overline{V}$ are in E_1, that $[N]\overline{U} \neq 0$, that $\overline{F_1}$ and $\overline{F_2}$ are well-defined, have the same codomain and that the following equalities hold.

$$[2^{\beta_2}]\overline{F_1}(0,0,C_a,0) = \widetilde{\overline{F_2}}([a_1]C_a, [-a_2]C_2, 0, 0) \implies \overline{F}(0,0,C_a,0) = ([a_1]C_a, [-a_2]C_2, 0, 0)$$

$$[2^{\beta_2}]\overline{F_1}(0,0,C_2,0) = \widetilde{\overline{F_2}}([a_1]C_a, [-a_2]C_2, \overline{V}, 0) \implies \overline{F}(0,0,C_2,0) = ([a_1]C_a, [-a_2]C_2, \overline{V}, 0)$$

From the isogeny $\overline{F}$, using ι_i and ρ_j the standard injections/restrictions of product spaces, we can construct 16 elliptic curve isogenies $\overline{F}_{i,j} = \rho_i \circ \overline{F} \circ \iota_j$ with $1 \leq i, j \leq 4$ such that for all $j = 1, \cdots, 4$:

$$\sum_{i=1}^{4} \deg(\overline{F}_{i,j}) = \deg(\overline{F}) = 2^{\beta}$$

We focus on the case when $j = 3$. We want to demonstrate that for $i = 1, 2,$ and 4, $F_{i,3} = [b_i]$, with b_i being a_1, $-a_2$, and 0, respectively. To achieve this, we utilize the Cauchy interpolation theorem. By applying the triangular inequality, we have:

$$\text{For } i = 1, 2, 4, \ \deg(\overline{F}_{i,3} - [b_i]) \leq 4 \cdot 2^{\beta} \approx 2^{2\lambda + 2\log(\lambda) + 2}.$$

We know that $\overline{F}_{i,3} = [b_i]$ for all points generated by $\langle C_a, C_2 \rangle$, i.e., for $Nq^2 \approx 2^{3\lambda}$ points. Thus, $\overline{F}_{1,3} = [a_1]$, $\overline{F}_{2,3} = [-a_2]$, and $\overline{F}_{4,3} = 0$. Using the previous equality, we can deduce that $\overline{F}_{3,3}$ is an isogeny of degree qd between E_A and E_1.

Furthermore, we have that $\overline{F}_{3,3}(C_a) = 0$ and $\overline{F}_{3,3}([N]C_2) \neq 0$, indicating that $\ker(\overline{F}_{3,3}) \cap E_A[q] = \langle C_a \rangle$, thereby proving our assertion. □

5 Security analysis of SQIPrime4D

We now prove that the SQIPrime4D identification protocol described in the Section 4 is a Σ-protocol. To do so, we have to show that SQIPrime4D has special soundness and is Honest Verifier Zero Knowledge (HVZK). Once both these points proven, applying the Fiat-Shamir transform [20] over SQIPrime4D will result in a digital signature scheme that is UU-CMA in the ROM. The extractor is constructed as follows.

Proposition 2. *Let $(E_1, \mathsf{chal}_1, T_1, U_1, V_1, d_1)$ and $(E_1, \mathsf{chal}_2, T_2, U_2, V_2, d_2)$ be 2 transcripts with identical commitment E_1 and $\mathsf{chal}_1 \neq \mathsf{chal}_2$. There exists an extractor $\mathcal{E}$ that, given both transcript, can efficiently solve the one endomorphism problem (Problem 1) over E_A, i.e. find $\gamma_A \in End(E_A)$ a non-trivial endomorphism.*

Proof. Our proof is very similar to [9, Proposition 17]. We can use T_1, U_1 to compute a high dimension representation of $\kappa_1 = \sigma_1 \circ \varphi_1$ and T_2, U_2 to compute a high dimension representation of $\widehat{\kappa_2} = \widehat{\sigma_2 \circ \varphi_2}$. Then, $\alpha = \widehat{\kappa_2} \circ \kappa_1 \in \mathrm{End}(E_A)$ is non-scalar, as otherwise, we have that $\alpha = [\chi]$ such that $\chi^2 = q^2 d_1 d_2$ and thus $\chi = q\chi'$. This would induce that $[d_2]\kappa_1 = [\chi']\kappa_2$ and thus induces that $\varphi_1 = \varphi_2$ i.e., that $\mathsf{chal}_1 = \mathsf{chal}_2$, which is a contradiction. □

The extractor ensures us that SQIPrime4D has special soundness. Similarly to [9, Section 5.2], we construct the simulator under the assumption that we have access to the following oracle.

Definition 2. *The* Random Uniformly Constrained Good Degree Isogeny Oracle *(RUCGDIO) is an oracle that takes as input a supersingular curve E together with $P \in E[q]$ and that returns an efficient representation of $\kappa : E \to E'$ of degree $q\ell$ with ℓ prime such that:*

- *E' is uniformly distributed over all supersingular curves.*
- *κ is uniformly distributed among all isogenies between E and E' such that $P \in \ker(\kappa)$ and such that $2^\beta - q\ell$ is a prime equal to $1 \mod 4$.*

Proposition 3. *Given* pp, pk *and* chal*, there exists a simulator $\mathcal{S}$ with access to a RUCGDIO that simulates transcripts with a distribution that is computationally indistinguishable from the distribution of transcripts of SQIPrime, conditioned to* chal*.*

Proof. Given $a \in \mathbb{Z}_q$, we compute $C_a = [N](R + [a]S)$. Calling RUCGDIO over E_A and C_a, we retrieve an efficient representation of $\kappa : E_A \to E_1$ and use this representation to compute the points $A = \kappa(X), B = \kappa(Y)$, and $Z = \kappa([b]R - [a]S)$ with X, Y the canonical basis over $E_A[2^\alpha]$.

We then simply return the following transcript $(E_1, a, A, B, Z, \deg(\kappa)/q)$.

This transcript is computationally indistinguishable from a genuine transcript, as:

- Following Assumption 1, we have that a genuine E_1 or one given by RUDGIO are computationally indistinguishable.
- Following [30, Lemma 3.2.4], a genuine κ or one given by RUDGIO are computationally indistinguishable, and so does $A, B, Z, \deg(\kappa)/q$.

□

We now make the following assumption.

Assumption 3. *The one endomorphism problem (Problem 1) remains hard even when given access to RUCGDIO.*

Indeed, by definition, RUCGDIO, when given an input P, generates a random isogeny that factors ϕ_P and that is of good degree. If P is of smooth order, then RUCGDIO is in fact equivalent to the RUGDIO oracle [9, Definition 5.2.1]. Thus, the arguments of [9, Section 5.3] also applies to RUCGDIO. It is therefore reasonable to assume that RUCGDIO does not help to break the endomorphism ring problem.

6 SQIPrime2D: SQIPrime in dimension 2

In this section, we describe a version of SQIPrime which uses only dimension 2 isogenies. Using dimension two isogenies allows for a more efficient scheme, which is this time more efficient than SQISignHD itself.

6.1 High level description

Recall the diagram for SQIPrime4D in Figure 2. In order to represent $\kappa = \sigma \circ \varphi$ using Kani's Lemma in dimension 2, we need to compute and evaluate an auxiliary isogeny $\delta : E_A \to E_\delta$ of degree $2^\alpha - dq$. Since the prover knows the endomorphism ring of E_A, he could in fact compute such an isogeny by using the KLPT algorithm, but this is not an admissible way to do it since we want to avoid using the costly KLPT algorithm.

Instead, we will use the **KaniDoublePath** algorithm together with several other techniques to generate the auxiliary isogeny of degree $2^\alpha - dq$. To achieve this goal, we will operate the following change to SQIPrime4D:

the secret isogeny τ will now be of fixed[6] *degree q, which is also the degree of the challenge isogeny φ.*

With that change in mind, we now sketch how one generates an auxiliary isogeny $\delta : E_A \to E_\delta$ of degree $2^\alpha - dq$. Firstly, one samples an endomorphism $\gamma \in \mathrm{End}(E_0)$ of degree $d(2^\alpha - dq)$, and one evaluates it on the 2^α-torsion. Next, one evaluates $\tau \circ \widehat{\gamma}$ on the 2^α-torsion basis $\{P_0, Q_0\}$ of E_0. Write $\gamma = \gamma_2 \circ \gamma_1$ where γ_1 and γ_2 have degree d and $2^\alpha - dq$ respectively, and let E_0' be the codomain of γ_1. Let $\delta : E_A \to E_\delta$ be the pushforward of γ_2 through $\tau \circ \widehat{\gamma_1}$. Then E_0, E_0', E_A and E_δ are the vertices of an SIDH square where the degrees are dq and $2^\alpha - dq$. One can hence apply Kani's Lemma to compute the isogeny $\delta : E_A \to E_\delta$ and evaluate it on the 2^α-torsion points. This is illustrated in Figure 3.

$$\begin{tikzcd}
E_0 \arrow[dd, dashed, "\eta"'] & E_0' \arrow[l, dashed, "\gamma_2"'] & \\
& E_0 \arrow[ul, dashed, "\gamma"] \arrow[u, dashed, "\gamma_1"'] \arrow[r, dashed, "\psi"] \arrow[d, dashed, bend right, "\widehat{\rho}"'] \arrow[d, dashed, bend left, "\tau"] & E_1 \\
E_\delta & E_A \arrow[l, "\delta"'] \arrow[ur, "\kappa"] \arrow[r, "\varphi"] & E_2 \arrow[u, dashed, "\sigma"']
\end{tikzcd}$$

Fig. 3. Diagram of SQIPrime2D, prover in blue and verifier in red. Dashed isogenies are not shared.

For SQIPrime2D, the public parameters are defined as follows:

- The base prime p is of the form $p = 2^\alpha f - 1 = 2Nq + 1 \simeq 2^{2\lambda}$, with $q \simeq 2^\lambda$ prime, such that: $\alpha \geq \lceil \frac{\log_2(p)}{2} + \log_2(q) \rceil + 1$ and $\alpha \geq \lceil 2 \log_2(q) \rceil$.
- P_0, Q_0 is a basis of $E_0[2^\alpha]$.

[6] This already implies that the key recovery problem in SQIPrime4D and SQIPrime2D are different, since the degree of the secret isogeny in SQIPrime4D is random and is not public.

- (P, Q, ι, I_P) is a precomputed basis of $E_0[q]$.

The computation of the commitment isogeny in SQIPrime2D is identical to that of the secret isogeny in SQIPrime4D, but the key generation, the response and the verification algorithms are modified.

6.2 SQIPrime2D Key Generation algorithm

For the secret isogeny, which is of prime degree q, we use the usual **KaniDoublePath**. Since it may happen that there are relatively few curves q-isogenous to E_0 but which are not $(2^\alpha - q)$-isogeny to E_0, a more conservative option is to use the **ExtKaniDoublePath** algorithm, which is more costly.

In SQIPrime2D, the points R and S are no longer the masked images of points P and Q by τ (as in SQIPrime4D). Instead, they are the masked images by $\widehat{\rho}$ of the points P and Q, where ρ is the second isogeny computed using **KaniDoublePath** or **ExtKaniDoublePath**. This change is necessary since $\deg(\tau) = q$, which is also the order of the points P and Q. We thus have that $\binom{R}{S} = \mathbf{M}\widehat{\rho}\binom{P}{Q}$. This time, one also includes $I_{\widehat{\rho}}$ in the secret key since it is needed when translating the kernel of the non-smooth challenge isogeny into an ideal.

With respect to the current state of the art [5,32,41,12] when it comes to the supersingular isogeny problem with torsion point information, there is no known algorithm that exploits the images of torsion points of non-smooth order to weaken the supersingular isogeny problem. All known attacks require the torsion point images to have smooth order. This means that the mask $\mathbf{M}$ is not really necessary since q is prime. We hence omit this masking. In case of an eventual breakthrough in the computation of isogenies with non-smooth torsion point information, then restoring the mask $\mathbf{M}$ would thwart such an attack.

Algorithm 9 SQIPrime2D.KeyGen

```
Input: pp = $\big(p, \alpha, q, N, (P_0, Q_0), (P, Q, \iota, I_P)\big)$
Output: sk = $\big(\mathsf{F_A}, I_\tau, I_{\widehat{\rho}}, \mathbf{M}\big)$, pk = $\big(E_A, (R, S)\big)$
$\mathsf{F_A}, I_\tau, I_{\widehat{\rho}} \leftarrow$ KaniDoublePath$(2^\alpha, P_0, Q_0, q)$
Compute $E_A$.
$\binom{R}{S} \leftarrow \widehat{\rho}\binom{P}{Q}$        ▷ $\widehat{\rho}(-) = F_A(0, -)_1$
return $(\mathsf{F_A}, I_\tau, I_{\widehat{\rho}})$, $\big(E_A, (R, S)\big)$
```

6.3 SQIPrime2D Response algorithm

Upon receiving $\mathsf{Chal} = a \in \mathbb{Z}_q$ from the verifier, the prover computes $C_a = R + [a]S$. the prover then calculates I_{C_a}, defined as

$$I_{C_a} = \big[\big(1 + a\delta(\iota)\big)I_{\widehat{\rho}}\big]_* I_P$$

Next, the prover computes the $(\mathcal{O}_2, \mathcal{O}_1)$-ideal $\overline{I_{C_a} I_\tau} I_\psi$ and locates another small $(\mathcal{O}_2, \mathcal{O}_1)$-ideal J using the **RandomEquivalentIdeal** algorithm. Following [9, Lemma 12], we are assured of the existence of such an ideal with a norm smaller than $\sqrt{p}$. Additionally, we require that $n(J)$ is odd. Notably, this condition is considerably less restrictive than that of SQIPrime4D, as approximately half of all potential isogenies remain valid, compared to only $1/\log(p)$ in the case of SQIPrime4D. Therefore, we have a high heuristic probability of finding our desired J with an odd norm d smaller than $2\sqrt{p}$, thereby yielding the corresponding isogeny $\sigma : E_2 \to E_1$.

With knowledge of d, the objective now shifts to constructing an auxiliary isogeny $\delta : E_A \to E_\delta$ of degree $2^\alpha - qd$. This specific mechanism lies at the heart of SQIPrime2D and underscores the necessity for the secret isogeny τ to be of degree q. The approach involves sampling $\gamma \in \mathrm{End}(E_0)$, an endomorphism of degree $d(2^\alpha - qd)$. This is done using **FullRepresentInteger**. Next, we compute $\binom{V}{W} = \tau \circ \hat{\gamma}\binom{P_0}{Q_0}$. Given that $\deg(\tau \circ \widehat{\gamma}) = dq(2^\alpha - qd)$, we find ourselves in the following scenario:

$$\begin{tikzcd}
E_0 \arrow[dd, "\eta"'] \arrow[dr, "\widehat{\gamma}"'] & E_0' \arrow[l, "\gamma_2"'] \arrow[d, "\widehat{\gamma_1}"] \\
& E_0 \arrow[d, "\tau"] \\
E_\delta & E_1 \arrow[l, "\delta"']
\end{tikzcd}$$

where $\gamma = \gamma_2 \circ \gamma_1$, $\deg(\gamma_1) = d$ and $\deg(\gamma_2) = (2^\alpha - qd)$. By applying Kani's Lemma, we construct F a dimension 2 isogeny defined as such

$$F : E_0 \times E_A \to E_0' \times E_\delta \qquad F := \begin{pmatrix} \widehat{\gamma_2} & -\gamma_1 \circ \widehat{\tau} \\ [\gamma_2]_*(\tau \circ \widehat{\gamma_1}) & [\tau \circ \widehat{\gamma_1}]_*\gamma_2 \end{pmatrix}$$

$$\ker(F) = \left\{ \left([-qd]P, \tau \circ \widehat{\gamma}(P)\right) \middle| \; P \in E_0[2^\alpha] \right\}$$

We thus have constructed an efficient representation of our desired $\delta = [\tau \circ \widehat{\gamma_1}]_*\gamma_2$.

The response to our challenge is to give the evaluation T, U of $\delta \circ \widehat{\kappa} = \delta \circ \widehat{\varphi} \circ \widehat{\sigma}$ over a basis of $E_1[2^\alpha]$ to the verifier. Additionally, we share the image $V = \delta(C_a)$ of C_a through δ. To do the evaluation, we call **CanonicalTorsionBasis** over E_1 to deterministically find a basis X, Y of $E_1[2^\alpha]$, evaluate $\widehat{\kappa}$ on X and Y using the **EvalTorsion** and compute δ on these images using the dimension two isogeny F. Finally, we multiply the final points by $(qd)^{-1} \mod 2^\alpha$. The prover then sends these three points together with the curve E_δ.

6.4 SQIPrime2D Verification algorithm

The key difference between verification in SQIPrime4D and SQIPrime2D is that the verifier receives a dimension 2 representation of $\widehat{\kappa}$ rather than κ. Efficiently evaluating κ is thus more technical and requires a novel approach.

Algorithm 10 SQIPrime2D.Response

```
Input: pp = $\big(p, \alpha, q, N, (P_0, Q_0), (P, Q, \iota, I_P)\big)$, sk = $(\mathsf{F}_\mathsf{A}, I_\tau, I_{\widehat{\rho}}, \mathbf{M})$, sec = $(\mathsf{F}_1, I_\psi)$, chal = $a$.
Output: res = $(E_\delta, T, U, V)$ with $T, U \in E_\delta[2^\alpha]$.
$I_{C_a} \leftarrow [(1 + a\iota)I_{\widehat{\rho}}]_* I_P$
$J \leftarrow$ RandomEquivalentIdeal($\overline{I_{C_a} I_\tau} I_\psi$)        $d \leftarrow n(J)$    ▷ resample if d even
$X, Y \leftarrow$ CanonicalTorsionBasis($E_1, 2^\alpha$)
$\gamma \leftarrow$ FullRepresentInteger($\mathfrak{O}_0, d(2^\alpha - dq)$)
$\binom{V}{W} = \tau \circ \widehat{\gamma}\binom{P_0}{Q_0}$
$\mathsf{B} \leftarrow \{([-dq]P_0, V), ([-dq]Q_0, W)\}$
$F \leftarrow$ HDKernelToIsogeny($E_0 \times E_1, \mathsf{B}$)
Define $\tau = \mathsf{F}_\mathsf{A}(-, 0)_1$ and $\psi = \mathsf{F}_1(-, 0)_1$.
$T_1, U_1 \leftarrow$ EvalTorsion$\Big(\mathfrak{O}_0, \tau, I_\tau, \psi, I_\psi, I_{C_1} J, qd, \{X, Y\}\Big)$
$\binom{T}{U} = [(qd)^{-1}]\delta\binom{T_1}{U_1}$                    ▷ $\delta(-) = F(0, -)_2$
$V = \delta(R + [a]S)$
Find $E_\delta$ =codomain($\delta$)
return res = $(E_\delta, T, U, V)$
```

Upon receipt of T, U, and V, the verifier deterministically computes the basis $\langle X, Y\rangle = E_1[2^\alpha]$. Following this, the verifier uses X, Y, T, and U to compute a basis for the kernel of the isogeny F, as derived from Kani's Lemma over the following diagram.

$$\begin{tikzcd}
E_A \arrow[r, "\kappa"] \arrow[d, "\delta"'] & E_1 \arrow[d, "\kappa_*\delta"] \arrow[dl, "\delta \circ \widehat{\kappa}"'] \\
E_\delta \arrow[r, "\delta_*\kappa"'] & E_\bullet
\end{tikzcd}$$

$$F : E_1 \times E_\delta \to E_A \times E'_\delta \text{ is defined as } \begin{pmatrix} \widehat{\kappa} & -\widehat{\delta} \\ \kappa_*\delta & \delta_*\kappa \end{pmatrix}$$

$$\ker(F) = \big\langle \big([qd]X, \delta \circ \widehat{\kappa}(X)\big), \big([qd]Y, \delta \circ \widehat{\kappa}(Y)\big)\big\rangle = \big\langle \big(X, T\big), \big(Y, U\big)\big\rangle$$

Using F, he computes the point $F\binom{0}{V} = \binom{-\widehat{\delta}(V)}{\delta_*\kappa(V)}$ and checks that:

1. $\delta_*\kappa(V) = 0$.
2. $\widehat{\delta}(V) = [2^\alpha - qd](R + [a]S) = [2^\alpha](R + [a]S)$.

The following proposition shows us that our verification is correct.

Proposition 4. *Let* pp, pk, com, chal *be the public parameters, a valid public key, a commitment, and a challenge in SQIPrime2D and let X, Y be the canonical basis of $E_1[2^\alpha]$. Let* $\overline{\mathsf{Res}}$ *be any possible response to the above.*

Algorithm 11 SQIPrime2D.Verify

```
Input: pp = $\big(p, (P_0, Q_0), (P, Q, \iota, I_P)\big)$, pk = $(E_A, R, S)$, com = $(E_1)$, chal = $a$, res = $(E_\delta, T, U, V)$
Output: 0 or 1
Check $T, U, V \in E_\delta$.
$X, Y \leftarrow$ CanonicalTorsionBasis$(E_1, 2^\alpha)$
$\mathsf{B} \leftarrow \big\{\big(X, T\big), \big(Y, U\big)\big\}$
$F \leftarrow$ HDKernelToIsogeny$(E_1 \times E_\delta, \mathsf{B})$        ▷ If not well defined, return 0
if codomain $\widehat{\kappa} \neq E_A$ do return 0
$\binom{Z_1}{Z_2} \leftarrow F\binom{0}{V} = \binom{-\widehat{\delta}(V)}{\delta_* \kappa(V)}$
$b_1 \leftarrow Z_1 \overset{?}{=} [2^\alpha](R + [a]S)$
$b_2 \leftarrow Z_2 \overset{?}{=} 0$
return $b_1 \wedge b_2$
```

Then ***SQIPrime2D. Verify***$(\mathsf{pp}, \mathsf{pk}, \mathsf{com}, \mathsf{chal}, \overline{\mathsf{Res}}) = 1 \iff \overline{\mathsf{Res}} = (\overline{E_\delta}, \overline{T}, \overline{U}, \overline{V})$ *is such that* $(X, Y, \overline{T}, \overline{U})$ *is a dim 2 representation of an isogeny* $\widehat{\overline{\kappa}} : E_1 \to E_A$ *of degree* $d\overline{q} < 2^\alpha$ *and such that* $\overline{\kappa}$ *factors through* φ*, the isogeny corresponding to the challenge* chal*.*

Proof. Given $\overline{E_\delta}, \overline{T}, \overline{U}, \overline{V}$, the fact that $\overline{F}$, the $(2^\alpha, 2^\alpha)$ isogeny whose kernel is generated by $\{(X, \overline{T}), (Y, \overline{U})\}$ is well-defined, we know that $\deg(\overline{F}_{1,1}) = \deg(\overline{F}_{2,2})$, $\deg(\overline{F}_{1,2}) = \deg(\overline{F}_{2,1})$ and that $\deg(\overline{F}_{1,1}) + \deg(\overline{F}_{1,2}) = 2^\alpha$. Note that this induces that both degree cannot share an odd divisor.

Thus, as $\overline{F}_{2,2}(\overline{V}) = 0$, we know that q divides $\deg(\overline{F}_{2,2})$, meaning that it cannot divide $\deg(\overline{F}_{1,2})$. We have that $\overline{F}_{1,2}(\overline{V}) = [2^\alpha](R + [a]S)$ implies that $[2^\alpha]\widehat{\overline{F}_{1,2}}(R + [a]S) = [\deg(\overline{F}_{1,2})]\overline{V}$. As q and $2^\alpha \deg(\overline{F}_{1,2})$ are coprime, we have that this implies that $\overline{F}_{2,2} \circ \widehat{\overline{F}_{1,2}}(R + [a]S) = 0 = \overline{F}_{2,1} \circ \widehat{\overline{F}_{1,1}}(R + [a]S)$. As $\deg(\overline{F}_{1,2}) = \deg(\overline{F}_{2,1})$ is not divisible by q, then $\widehat{\overline{F}_{1,1}}(R + [a]S) = 0$. We therefore have that $\widehat{\overline{F}_{1,1}} : E_A \to E_1$ is of degree $d\overline{q} < 2^\alpha$ and it factors through the isogeny φ corresponding to the challenge chal, proving our soundness. □

7 Security analysis of SQIPrime2D

Similarly to SQIPrime4D, we have to show that SQIPrime2D defines a Σ protocol. We thus have to prove that we have special soundness and are Honest Verifier Zero Knowledge.

The proof of the special soundness of SQIPrime2D is almost identical to the proof of Proposition 2. It is therefore omitted. Regarding HVZK, there are several differences between SQIPrime4D and SQIPrime2D:

1. We have access to an auxiliary isogeny $\delta : E_A \to E_\delta$
2. Our isogeny κ is of degree qd with d no longer required to be such that $2^\beta - qd$ is prime and equal to $1 \mod 4$.

We therefore need to define our HVZK under new oracles, defined as such.

Definition 3. *The* Random Uniform Constrained Odd Degree Isogeny Oracle *(RUCODIO) is an oracle that takes as input a supersingular curve E together with $P \in E[q]$ and returns an efficient representation of an isogeny $\kappa : E \to E'$ of degree $q\ell$ such that:*

- *E' is uniformly distributed.*
- *κ is uniformly distributed among all isogenies between E and E' such that:*
  - *ℓ is odd and such that $q\ell \leq 2^\alpha$.*
  - *κ is such that $\kappa(P) = 0$.*

Definition 4. *The* Auxiliary Isogeny Oracle *(AIO) is an oracle that takes as input a supersingular curve E together with an odd integer $\ell < 2^\alpha/q$ and returns an efficient representation of an isogeny $\delta : E \to E''$ of degree $2^\alpha - q\ell$ such that it has the same distribution as the auxiliary isogeny computed in Algorithm 11.*

Using RUCODIO and AIO, we can now prove our HVZK.

Proposition 5. *Given* pp, pk *and* chal*, then there exists a simulator $\mathcal{S}$ with access to a RUCODIO and AIO that simulates transcripts with a distribution that is computationally indistinguishable from the distribution of transcripts of SQIPrime2D, conditioned to* chal.

Proof. Given E_A, we sample $a \in \mathbb{Z}_q$ and construct $C = R + [a]S$ call RUCODIO over E_A and C, we retrieve an efficient representation of $\kappa : E_A \to E_1$. We compute $\ell = \deg(\kappa)/q$ and call AIO over E_A and d to retrieve $\delta : E_A \to E_\delta$. We use this representation to compute the points $T = \delta \circ \widehat{\kappa}(X), U = \delta \circ \widehat{\kappa}(Y)$ and $V = \delta(C)$ with X, Y the canonical basis over $E_1[2^\alpha]$. We then simply return the following transcript $(E_1, a, E_\delta, T, U, V)$.

This transcript is computationally indistinguishable from a genuine transcript, as:

- A genuine E_1 or one given by RUCODIO are computationally indistinguishable, following Assumption 1.
- Due to Definition 4, E_δ has the same distribution as the isogeny computed during SQIPrime2D response. This also applies to the point V.
- Following [30, Lemma 3.2.4], a genuine κ or one given by RUCODIO are computationally indistinguishable, and so does T, U. □

We now make the following assumption.

Assumption 4. *The one endomorphism problem (Problem 1) remains hard even when given access to RUCODIO and AIO.*

Thus, we have that, under our assumptions, SQIPrime2D is a Σ-protocol. Note that since the degree q of the secret isogeny $\tau : E_0 \to E_A$ is fixed, and that the images of some non-smooth order torsion points through the byproduct isogeny $\widehat{\rho} : E_0 \to E_A$ are given, then the direct key recovery problem in SQIPrime2D is the following.

Problem 2. Let $\phi_1, \phi_2 : E_0 \to E'$ be two supersingular isogenies of degree q and $2^\alpha - q$ respectively, where $q \simeq 2^\lambda$ is a prime. Given E_0, E' and $\phi_2(E_0[q])$, retrieve ϕ_1 or ϕ_2.

When a mask is used to hide the images of the torsion points through ϕ_2 (which plays the role of $\widehat{\rho}$), then there is no torsion point information in the problem, which means that in the problem above, $\phi_2(E[q])$ is not provided. In practice, as previously touched in Section 6.2, all currently known algorithms to solve Problem 2 do not exploit the torsion points $\phi_2(E[q])$ at all, since they have non-smooth order. It is an open problem to know whether they are of any importance when solving the (fixed degree) supersingular isogeny problem. To the best of our knowledge, brute-force remains the best method to solve Problem 2.

8 Parameters & Efficiency

As discussed in Section 4 and Section 6, the public parameters in both versions of SQIPrime differ significantly from those used in SQISign [15,16] and SQISignHD [9], particularly concerning their base prime numbers. This section provides a detailed explanation on how to compute suitable baseline "SQIPrime-friendly" primes.

8.1 Finding "SQIPrime4D-friendly" primes

We can view "SQIPrime4D-friendly" primes as a combination of the "SIDH primes" used in SQISignHD and the stringent requirements on both $p + 1$ and $p - 1$ seen in SQISign primes. However, in SQIPrime4D, the only condition is that $p - 1$ needs to have a factor of size $O(2^\lambda)$.

Finding "SQIPrime4D-friendly" primes is actually easier than finding "SQISign-friendly" primes. These primes can in fact be found by brute-force searching over the cofactor f.

We give here below good candidates we found using this method:

- $\lambda = 128$:

$$\begin{aligned} p + 1 &= 2^{241} \cdot 33967 \simeq 2^{256.05} \\ p - 1 &= 2 \cdot 3^2 \cdot 17^2 \cdot 191 \cdot 193 \cdot 573197 \cdot 16874350656700045993471453 \cdot q \\ q &= 647133889352330391744288229376113975777 \simeq 2^{128.92} \end{aligned}$$

$$\begin{aligned} p + 1 &= 2^{240} \cdot 167 \cdot 397 \simeq 2^{256.01} \\ p - 1 &= 2 \cdot 3 \cdot 7 \cdot 11 \cdot 41 \cdot 5683514583831199 \cdot 500402127095125861 \cdot q \\ q &= 2174422729538275144428922863792468335219 \simeq 2^{130.67} \end{aligned}$$

- $\lambda = 192$:

$$\begin{aligned} p+1 &= 2^{368} \cdot 239 \cdot 277 \simeq 2^{384.01} \\ p-1 &= 2 \cdot 17189 \ \cdot 90887498813318264275 7539 \cdot 10823140854729895254 9129134411 \cdot q \\ q &= 3 \cdot 7 \cdot 4803463386334137403 \cdot 116682096886878909945888202135243873061 \\ &\simeq 2^{192.9} \end{aligned}$$

- $\lambda = 256$:

$$\begin{aligned} p+1 &= 2^{497} \cdot 5^2 \cdot 479 \simeq 2^{512.13} \\ p-1 &= 2 \cdot 2663 \cdot 63377 \cdot 34213446044029994491 \cdot 887410852538335907427659343850950255860606 4045631 \cdot q \\ q &= 97 \cdot 147869462015622684206054234380684709202350141554573643051528098693560900067 7 \simeq 2^{256.3} \end{aligned}$$

8.2 Finding "SQIPrime2D-friendly" primes

Similarly, finding *good* "SQIPrime2D-friendly" primes is also computationally involved. This essentially comes from the fact that we need to have enough 2-torsion points to compute the verification isogeny, meaning that we require for $\alpha \geq \lambda + \log_2(p)/2 \implies \alpha \geq 2\lambda + \log_2(f)$, which implies that p is around $2\lambda + 2\log_2(f)$ bit long. Furthermore, if we take $p = 2^\alpha f - 1$ to be a prime, then the probability that a random prime q divides $p-1$ is roughly $1/q$. Given that there are approximately $2^\lambda(2^t - 1)/\lambda$ distinct primes in the interval $[2^\lambda, 2^{\lambda+t}]$, the probability that there exists a prime q in $[2^\lambda, 2^{\lambda+t}]$ that divides $p-1$ is heuristically given by:

$$\begin{aligned} \mathbb{P}\left[\exists q \in \left[2^\lambda, 2^{\lambda+t}\right] \text{ such that } q \middle| (p-1)\right] &\geq \sum_{q \geq 2^\lambda}^{2^{\lambda+t}} \mathbb{P}\left[q|(p-1)\right] \\ &\simeq \sum_{q \geq 2^\lambda}^{2^{\lambda+t}} \frac{1}{q} \geq \sum_{i=1}^{t} \sum_{q \geq 2^{\lambda+i-1}}^{2^{\lambda+i}} \frac{1}{2^{\lambda+i}} \\ &\simeq \sum_{i=1}^{t} \frac{2^{\lambda+i}}{(\lambda+i)} \frac{1}{2^{\lambda+i}} \simeq \sum_{i=1}^{t} \frac{1}{\lambda+i} \geq \frac{t}{\lambda+t}. \end{aligned}$$

Following this computation, the probability that, for a given f, $p = 2^\alpha f - 1$ is prime *and* $p-1$ has a factor close to λ-bit long is about $O(1/\lambda^2)$.This induces that the expected size of f is around $2\log_2(\lambda)$, meaning that a "SQIPrime2D-friendly" primes for the security level λ is of expected size $2\lambda + 4\log_2(\lambda)$ bits.

These additional $4\log_2(\lambda)$ bits present a challenge. For $\lambda = 128$, this results in an overhead of approximately 28 bits, which translates to an 11% increase in the size of the base prime p.[7]

For $\lambda = 128$, the first "SQIPrime2D-friendly" prime we identified is denoted as p_{130}.

$$\begin{aligned} p_{130} + 1 &= 2^{273} \cdot 19^2 \simeq 2^{281.50} \\ p_{130} - 1 &= 2 \cdot 3 \cdot 5 \cdot 59 \cdot 191 \cdot 2797 \cdot 16585601 \cdot 20157471998472338092840795930 7 \cdot q_{130} \\ q_{130} &= 173312401330203632071817182256347704766 7 \simeq 2^{130.35} \end{aligned}$$

To find a smaller p, it is tempting to ask for q to be non-prime, as we did for SQIPrime4D, but this is not possible as our security would be downgraded by an unsmooth generalisation of the Galbraith meet-in-the-middle attack [23]. However, to maintain efficiency, we may tolerate a slight reduction in the bit length of q. We suggest the following prime.

$$\begin{aligned} p_{117} + 1 &= 2^{247} \cdot 79 \simeq 2^{253.34} \\ p_{117} - 1 &= 2 \cdot 3 \cdot 5 \cdot 2903 \cdot 1924673583633629 \cdot 634009940699607211039 \cdot q_{117} \\ q_{117} &= 168118140144706967996895604212334429 \simeq 2^{117.01} \end{aligned}$$

Searching for "SQIPrime2D-friendly" primes corresponding to security level $\lambda = 192$ and 256 is computationally heavy. This essentially comes from the fact that it requires to factor several numbers of about 384 and 512 bits. We will provide primes for these security levels in a follow-up version of this paper.

8.3 Compactness of SQIPrime

Similarly to SQISign and SQISignHD, both version of SQIPrime are made into digital signature schemes via the Fiat-Shamir transform [20]. Thoses digital schemes are universally unforgeable under chosen message attacks (UU-CMA) in the random oracle and RUCGDIO or RUCODIO+AIO model, assuming the hardness of the one endomorphism problem.

Signature size In the case of SQIPrime2D, the signature takes the form $\mathsf{sign} = \big(E_1, E_\delta, T, U, V\big)$. It is therefore of size This signature can be slightly compressed using methods akin to those outlined in [9, Section 6.1]. The crux of this compression lies in representing T and U by $a_1, a_2, a_3, \in \mathbb{Z}_{2^\alpha}$ corresponding to their coordinates in a deterministic basis of $E_\delta[2^\alpha]$, with the final coordinate a_4 derived using pairings and discrete logs and using d an integer of λ bits.

Employing this compression method, each component of a SQIPrime2D signature exhibits the following sizes:

[7] It is important to note that this overhead scales logarithmically with λ. As λ doubles, the overhead only increases by 4 bits, meaning that its relative cost decreases at higher security levels.

- E_1 and E_δ are represented by its j-invariant in $\mathbb{F}_{p^2}$, hence of size $8\lambda + O(\log \lambda)$.
- T and U are each represented by three integers of size α plus d of size $\log(p)/2$, totaling $7\lambda + O(\log \lambda)$ bits.
- Finally, because q is non-smooth, we can not compress V, meaning that they are represented as a point in $\mathbb{F}_{p^2}$, hence of size $4\lambda + O(\log \lambda)$.

Summing these sizes, a SQIPrime2D signature is $19\lambda + O(\log \lambda)$ bits long. Consequently, it is larger than the signature of SQISignHD, which was $13/2\lambda + O(\log \lambda)$ bits, and also larger than SQISign, which is at least $17/2\lambda + O(\log \lambda)$ bits. Nevertheless, it remains a highly compact post-quantum signature scheme.

It is noteworthy that similar compression techniques can be applied to the SQIPrime4D signature, which is of the form (E_1, T, U, V, d), resulting in a signature size of $12\lambda + O(\log \lambda)$ bits. This difference of 7λ bits comes from the fact that in SQIPrime2D, we have to share E_δ and because in SQIPrime4D, we split the verification in 2 dimension 4 isogenies, therefore only requiring 2^β-torsion points, as opposed to 2^α in SQIprime2D.

Scheme	pk	signature	signature (compressed)
SQISign	64	322	177
SQISignHD	64	208	109
SQIPrime4D	192	272	240
SQIPrime2D (p_{117})	191	320	299

Table 1. Size (in bytes) comparison between the different SQI-protocols for public keys and signatures in both normal and compressed form.

8.4 SQIPrime efficiency

The next phase for SQIPrime involves developing an efficient implementation of SQIPrime2D. Building upon the advancements made in [10], as well as leveraging the efficient implementations of SQISign [15,16] and SQISignHD [9], we anticipate that SQIPrime2D will demonstrate very competitive performance.

We expect that SQIPrime2D will offer key generation and signature times akin (maybe slightly slower for the signature) to those of SQISignHD, while maintaining verification times consistent with the initial versions of SQISign. This intuition follows the number of $(2,2)$ isogenies required to perform SQIPrime2D, as detailed in Table 2. We intend to provide more detailed performance metrics once our implementation is finalized.

References

1. Basso, A., de Feo, L., Dartois, P., Leroux, A., Maino, L., Pope, G., Robert, D., Wesolowski, B.: SQIsign2D-West: the Fast, the Small, and the Safer. Personal communication (May 2024)

Scheme ($\lambda = 128$)		2	3	(2,2)	(2,2,2,2)
SQISignHD	KeyGen	378	234	-	-
	Sign	252	312	-	-
	Verif	-	78	-	142
SQIPrime4D	KeyGen	-	-	241	-
	Sign	-	-	241	-
	Verif	-	-	-	263
SQIPrime2D (p_{130})	KeyGen	-	-	273	-
	Sign	-	-	546	-
	Verif	-	-	273	-
SQIPrime2D (p_{117})	KeyGen	-	-	247	-
	Sign	-	-	494	-
	Verif	-	-	247	-

Table 2. Number and types of isogenies needed to perform SQISignHD, SQIPrime4D and SQIPrime2D (numbers in parentheses are for the alternative KeyGen).

2. Basso, A., Maino, L., Pope, G.: FESTA: Fast Encryption from a Supersingular Torsion Attacks. In: Guo, J., Steinfeld, R. (eds.) Advances in Cryptology – ASIACRYPT 2023. pp. 98–126. Springer Nature Singapore, Singapore (2023)
3. Bernstein, D.J., De Feo, L., Leroux, A., Smith, B.: Faster computation of isogenies of large prime degree. Open Book Series **4**(1), 39–55 (2020)
4. Beullens, W., Kleinjung, T., Vercauteren, F.: CSI-FiSh: Efficient Isogeny Based Signatures Through Class Group Computations **11921**, 227–247 (2019). https://doi.org/10.1007/978-3-030-34578-5_9
5. Castryck, W., Decru, T.: An efficient key recovery attack on SIDH. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 423–447. Springer (2023)
6. Chen, M., Leroux, A.: SCALLOP-HD: group action from 2-dimensional isogenies. Cryptology ePrint Archive, Paper 2023/1488 (2023), https://eprint.iacr.org/2023/1488, https://eprint.iacr.org/2023/1488
7. Cornacchia, G.: Su di un metodo per la risoluzione in numeri interi dell'equazione... (1907)
8. Costello, C.: B-sidh: supersingular isogeny diffie-hellman using twisted torsion. In: Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part II 26. pp. 440–463. Springer (2020)
9. 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
10. 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
11. De Feo, L.: Mathematics of isogeny based cryptography. arXiv preprint arXiv:1711.04062 (2017)
12. De Feo, L., Fouotsa, T.B., Panny, L.: Isogeny problems with level structure. In: Joye, M., Leander, G. (eds.) Advances in Cryptology – EUROCRYPT 2024. pp. 181–204. Springer Nature Switzerland, Cham (2024)

13. De Feo, L., Galbraith, S.D.: SeaSign: compact isogeny signatures from class group actions. In: Advances in Cryptology–EUROCRYPT 2019: 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19–23, 2019, Proceedings, Part III 38. pp. 759–789. Springer (2019)
14. De Feo, L., Jao, D., Plût, J.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. Journal of Mathematical Cryptology **8**(3), 209–247 (2014)
15. De Feo, L., Kohel, D., Leroux, A., Petit, C., Wesolowski, B.: SQISign: compact post-quantum signatures from quaternions and isogenies. In: Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part I 26. pp. 64–93. Springer (2020)
16. De Feo, L., Leroux, A., Longa, P., Wesolowski, B.: New algorithms for the deuring correspondence: towards practical and secure SQISign signatures. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 659–690. Springer (2023)
17. De Feo, L., Delpech de Saint Guilhem, C., Fouotsa, T.B., Kutas, P., Leroux, A., Petit, C., Silva, J., Wesolowski, B.: SÉTA: Supersingular encryption from torsion attacks. In: Advances in Cryptology–ASIACRYPT 2021: 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6–10, 2021, Proceedings, Part IV 27. pp. 249–278. Springer (2021)
18. Deuring, M.: Die typen der multiplikatorenringe elliptischer funktionenkörper. In: Abhandlungen aus dem mathematischen Seminar der Universität Hamburg. vol. 14, pp. 197–272. Springer Berlin/Heidelberg (1941)
19. 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
20. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Conference on the theory and application of cryptographic techniques. pp. 186–194. Springer (1986)
21. Fouotsa, T.B.: A note on the prime in SQISignHD. Online (2024), https://github.com/BorisFouotsa/BorisFouotsa.github.io/blob/main/files/A_note_on_the_prime_in_SQISignHD.pdf
22. Fouotsa, T.B., Petit, C.: Sheals and heals: Isogeny-based pkes from a key validation method for sidh. In: Tibouchi, M., Wang, H. (eds.) Advances in Cryptology – ASIACRYPT 2021. pp. 279–307. Springer International Publishing, Cham (2021)
23. Galbraith, S.D.: Constructing isogenies between elliptic curves over finite fields. LMS Journal of Computation and Mathematics **2**, 118–138 (1999)
24. Galbraith, S.D., Petit, C., Silva, J.: Identification protocols and signature schemes based on supersingular isogeny problems. Cryptology ePrint Archive, Paper 2016/1154 (2016), https://eprint.iacr.org/2016/1154, https://eprint.iacr.org/2016/1154
25. Goldreich, O.: Foundations of cryptography: volume 2, basic applications. Cambridge university press (2009)
26. Jao, D., De Feo, L.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In: Post-Quantum Cryptography: 4th International Workshop, PQCrypto 2011, Taipei, Taiwan, November 29–December 2, 2011. Proceedings 4. pp. 19–34. Springer (2011)

27. Kani, E.: The number of curves of genus two with elliptic differentials. Walter de Gruyter, Berlin/New York Berlin, New York (1997)
28. Kohel, D., Lauter, K., Petit, C., Tignol, J.P.: On the quaternion-isogeny path problem. LMS Journal of Computation and Mathematics **17**(A), 418–432 (2014)
29. Kunzweiler, S.: Efficient computation of $(2^n, 2^n)$-isogenies. Cryptology ePrint Archive, Paper 2022/990 (2022), https://eprint.iacr.org/2022/990, https://eprint.iacr.org/2022/990
30. Leroux, A.: Quaternion Algebra and Isogeny-Based Cryptography. Ph.D. thesis, Ecole doctorale de l'Institut Polytechnique de Paris (2022)
31. 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
32. Maino, L., Martindale, C., Panny, L., Pope, G., Wesolowski, B.: A direct key recovery attack on SIDH. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 448–471. Springer (2023)
33. Milne, J.S.: Abelian varieties. Arithmetic Geometry pp. 103–150 (1986)
34. Moriya, T.: IS-CUBE: An isogeny-based compact KEM using a boxed SIDH diagram. Cryptology ePrint Archive, Paper 2023/1506 (2023), https://eprint.iacr.org/2023/1506, https://eprint.iacr.org/2023/1506
35. 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
36. Nakagawa, K., Onuki, H.: SQIsign2D-East: A New Signature Scheme Using 2-dimensional Isogenies. Personal communication (May 2024)
37. Page, A., Robert, D.: Introducing clapoti(s): Evaluating the isogeny class group action in polynomial time. Cryptology ePrint Archive, Paper 2023/1766 (2023), https://eprint.iacr.org/2023/1766, https://eprint.iacr.org/2023/1766
38. Pizer, A.K.: Ramanujan graphs and hecke operators. Bulletin of the American Mathematical Society **23**(1), 127–137 (1990)
39. Robert, D.: Fonctions thêta et applications à la cryptographie. Ph.D. thesis, Université Henri Poincaré-Nancy I (2010)
40. Robert, D.: Evaluating isogenies in polylogarithmic time. Cryptology ePrint Archive, Paper 2022/1068 (2022), https://eprint.iacr.org/2022/1068, https://eprint.iacr.org/2022/1068
41. Robert, D.: Breaking SIDH in polynomial time. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 472–503. Springer (2023)
42. Santos, M.C.R., Costello, C., Smith, B.: Efficient (3,3)-isogenies on fast Kummer surfaces. Cryptology ePrint Archive, Paper 2024/144 (2024), https://eprint.iacr.org/2024/144, https://eprint.iacr.org/2024/144
43. Silverman, J.H.: The arithmetic of elliptic curves, vol. 106. Springer (2009)
44. Vélu, J.: Isogénies entre courbes elliptiques. Comptes-Rendus de l'Académie des Sciences **273**, 238–241 (1971)
45. Yoo, Y., Azarderakhsh, R., Jalali, A., Jao, D., Soukharev, V.: A post-quantum digital signature scheme based on supersingular isogenies. In: Financial Cryptography and Data Security: 21st International Conference, FC 2017, Sliema, Malta, April 3-7, 2017, Revised Selected Papers 21. pp. 163–181. Springer (2017)