SQISign: Compact Post-Quantum Signatures from Quaternions and Isogenies

Luca De Feo[1,7,8][0000−0002−9321−0773], David Kohel[2], Antonin Leroux[3,7,8], Christophe Petit[4,9], and Benjamin Wesolowski[5,6]

[1] IBM Research
[2] Aix Marseille Univ, CNRS, Centrale Marseille, I2M, Marseille, France
[3] DGA
[4] University of Birmingham
[5] Univ. Bordeaux, CNRS, Bordeaux INP, IMB, UMR 5251, F-33400, Talence, France
[6] INRIA, IMB, UMR 5251, F-33400, Talence, France
[7] LIX, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris
[8] INRIA
[9] Université libre de Bruxelles

Abstract. We introduce a new signature scheme, *SQISign*, (for *Short Quaternion and Isogeny Signature*) from isogeny graphs of supersingular elliptic curves. The signature scheme is derived from a new one-round, high soundness, interactive identification protocol. Targeting the post-quantum NIST-1 level of security, our implementation results in signatures of 204 bytes, secret keys of 16 bytes and public keys of 64 bytes. In particular, the signature and public key sizes combined are an order of magnitude smaller than all other post-quantum signature schemes. On a modern workstation, our implementation in C takes 0.6s for key generation, 2.5s for signing, and 50ms for verification.
While the soundness of the identification protocol follows from classical assumptions, the zero-knowledge property relies on the second main contribution of this paper. We introduce a new algorithm to find an isogeny path connecting two given supersingular elliptic curves of known endomorphism rings. A previous algorithm to solve this problem, due to Kohel, Lauter, Petit and Tignol, systematically reveals paths from the input curves to a 'special' curve. This leakage would break the zero-knowledge property of the protocol. Our algorithm does not directly reveal such a path, and subject to a new computational assumption, we prove that the resulting identification protocol is zero-knowledge.

Keywords: Post-quantum · Signatures · Isogenies.

1 Introduction

Isogeny-based cryptography has existed since at least the work of Couveignes in 1997 [9] and has developed significantly in the last decade due to increasing interest in post-quantum cryptography. The CGL hash function of [6] and the SIDH key exchange proposed in [20] have put isogenies between supersingular elliptic curves at the center of attention. The security of these schemes relies on the hardness of finding a path in the ℓ-isogeny supersingular graph between two given vertices. This problem is believed to be hard for both classical and quantum computers. This assumption was studied by Kohel, Lauter, Petit and Tignol, who in [22] introduced a new algorithm (often called KLPT in the litterature) that solves the quaternion analog of the ℓ-isogeny path problem under the Deuring correspondence. This algorithm revealed its full potential in [17], leading to several reductions between computational problems related to isogenies between supersingular curves, most notably a heuristic security reduction between the ℓ-isogeny path problem and the endomorphism ring computation.

In parallel to these cryptanalytic efforts, isogeny-based cryptography has continued to develop with several new proposals. We can mention CSIDH [5], an efficient reinterpretation of Couveignes' idea using supersingular elliptic curves defined over $\mathbb{F}_p$. Another active area of research has been isogeny-based signature schemes, see for instance [33,19,12,14,3].

Galbraith, Petit and Silva's signature scheme [19] (also known as GPS) was the first constructive cryptographic application of the KLPT algorithm. However, their work remains mainly theoretical and, to this day, we are not aware of any implementation of their scheme. We follow in the footsteps of GPS by introducing a new signature scheme based on the quaternion ℓ-isogeny path problem. Indeed, GPS relies on the KLPT algorithm for so-called "special" maximal orders (the main focus of [22]), whereas our protocol requires a new variant of KLPT working for arbitrary maximal orders, which we introduce here.

The contributions of this paper can be summarized as follows:

- A new interactive identification protocol and the resulting signature scheme based on a generic algorithm for the quaternion ℓ-isogeny path problem.
- A new generic KLPT algorithm, suited for our signature scheme, which produces a smaller output than the existing algorithm of [22].
- A proof of the interpretation of Eichler orders and their class sets under the Deuring correspondence, and its application to the analysis of the output of our algorithm. This leads us to a natural security assumption from which we prove zero-knowledge of the identification scheme, and consequently unforgeability of the signature scheme.
- New algorithms for the efficient instantiation of the protocol, along with parameters targeting the NIST-1 level of post-quantum security, and a complete implementation of our signature scheme in C.

The remainder of this paper is organized as follows. Section 2 contains preliminaries on elliptic curves and quaternion algebras. Section 3 sketches our new protocols along with some proofs. Section 4 lays out the mathematical background on Eichler orders necessary for the rest of the paper. Section 5 gives a generic description of our new Generalized KLPT algorithm. Section 6 provides the generic variant used in our protocols. Section 7 studies the zero knowledge property of the identification scheme. Finally, Section 8 provides algorithms for efficient implementation of the schemes.

2 Preliminaries

A negligible function $f : \mathbb{Z}_{>0} \rightarrow \mathbb{R}_{>0}$ is a function whose growth is bounded by $O(x^{-n})$ for all $n > 0$. In the analysis of a probabilistic algorithm, we say that an event happens with *overwhelming probability* if its probability of failure is a negligible function of the length of the input. We say that a distinguishing problem is hard when any probabilistic polynomial-time distinguisher has a negligible advantage with respect to the length of the instance. Two distributions are computationally indistinguishable if their associated distinguishing problem is hard.

Throughout this work, p is a prime number and $\mathbb{F}_q$ a finite field of characteristic p. We are interested in supersingular elliptic curves over $\mathbb{F}_q = \mathbb{F}_{p^2}$, in an isogeny class such that the full endomorphism ring is defined over $\mathbb{F}_q$, and is isomorphic to a maximal order in a quaternion algebra. The extended version of this work [13] contains more background on elliptic curves and their endomorphism rings; other useful references are [10,29,21,31].

2.1 The Deuring Correspondence

In [15], Deuring made the link between the geometric world of elliptic curves and the arithmetic world of quaternion algebras over $\mathbb{Q}$ by showing that the endomorphism ring of a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$ is isomorphic to a maximal order in the quaternion algebra $\mathcal{B}_{p,\infty}$ ramified at p and infinity. This correspondence is in fact an equivalence of categories [21] between supersingular elliptic curves and left ideals for a maximal order $\mathcal{O}$ of $\mathcal{B}_{p,\infty}$, inducing a bijection between conjugacy classes of supersingular j-invariants and maximal orders (up to equivalence). Given a supersingular curve E_0, this lets us associate each pair (E_1, φ), where E_1 is another supersingular elliptic curve and $\varphi : E_0 \rightarrow E_1$ is an isogeny, to a left integral $\mathcal{O}_0$-ideal (with $\mathrm{End}(E_0) \simeq \mathcal{O}_0$), and every such ideal arises in this way. In this case $\mathrm{End}(E_1)$ is isomorphic to the right order of this ideal. The explicit correspondence between isogenies and ideals is given through kernel ideals as defined in [32]. Given I an integral left-$\mathcal{O}_0$-ideal we define the set $E_0[I] = \{P \in E_0(\overline{\mathbb{F}}_{p^2}) : \alpha(P) = 0 \text{ for all } \alpha \in I\}$ as the kernel of I. To I, we associate the isogeny φ_I of kernel $E_0[I]$ defined by $\varphi_I : E_0 \rightarrow E_0/E_0[I]$. Conversely given an isogeny φ, the corresponding kernel ideal is defined as $I_\varphi = \{\alpha \in \mathcal{O}_0 \ : \ \alpha(P) = 0 \text{ for all } P \in \ker(\varphi)\}$.

Remark 1. In the definitions above we identify $\alpha \in \mathcal{O}_0$ with the related endomorphism in $\mathrm{End}(E_0)$, implicitly assuming a fixed isomorphism between $\mathcal{O}_0$ and $\mathrm{End}(E_0)$. This is a simplification that we will reiterate throughout this paper to lighten notations. In fact, we will sometimes go further and also write α for the principal ideal $\mathcal{O}_0\alpha$. It is easily verified that this ideal corresponds to the kernel ideal I_α, and conversely any principal ideal corresponds to an endomorphism $\varphi_{\mathcal{O}_0\alpha}$.

We summarize the main properties of this correspondence in Table 1.

Supersingular j-invariants over $\mathbb{F}_{p^2}$ $j(E)$ (up to galois conjugacy)	Maximal orders in $\mathcal{B}_{p,\infty}$ $\mathcal{O} \cong \mathrm{End}(E)$ (up to isomorpshim)
(E_1, φ) with $\varphi : E \to E_1$	I_φ integral left $\mathcal{O}$-ideal and right $\mathcal{O}_1$-ideal
$\theta \in \mathrm{End}(E_0)$	Principal ideal $\mathcal{O}\theta$
$\deg(\varphi)$	$n(I_\varphi)$
$\hat{\varphi}$	$\overline{I_\varphi}$
$\varphi : E \to E_1, \psi : E \to E_1$	Equivalent Ideals $I_\varphi \sim I_\psi$
Supersingular j-invariants over $\mathbb{F}_{p^2}$	$\mathrm{Cl}(\mathcal{O})$
$\tau \circ \rho : E \to E_1 \to E_2$	$I_{\tau\circ\rho} = I_\rho \cdot I_\tau$

Table 1. The Deuring correspondence, a summary.

2.2 Algorithmic building blocks

In this section we introduce some sub-algorithms that will be used in the remaining of the paper. These algorithms are either classical or inherited from recent works [22,19] in the literature.

We will write $\mathsf{CRT}_{M,N}(x, y)$ for the Chinese Remainder algorithm, that takes $x \in \mathbb{Z}/M\mathbb{Z}$, $y \in \mathbb{Z}/N\mathbb{Z}$ and returns $z \in \mathbb{Z}/MN\mathbb{Z}$ with $z = x \bmod M$ and $z = y \bmod N$.

KLPT Algorithm A significant part of the present work is spent on providing a new generalization of the KLPT algorithm [22] (see Algorithm 3). This algorithm takes an integral ideal I as input and finds an equivalent ideal $J \sim I$ of given norm. For instance, the norm can be required to be ℓ^e for some $e \in \mathbb{N}$. In general, in the rest of this paper when an output of an algorithm is required to be a power of ℓ, we write $\ell^\bullet$.

We start by introducing a few notations taken from [22], before introducing several sub-algorithms that we will use. Finally we describe a short version of KLPT in Algorithm 1 built from these sub-algorithms.

An important notion introduced in [22] is that of *special extremal* orders, i.e., maximal orders $\mathcal{O}_0$ containing a suborder admitting an orthogonal decomposition $R + jR$ where $R = \mathbb{Z}[\omega] \subset \mathbb{Q}[i]$ is a quadratic order of minimal discriminant (or equivalently such that ω has smallest norm in $\mathcal{O}_0$). By orthogonal decomposition we mean that $R \subset (jR)^\perp$. The order $\mathcal{O}_0 = \mathbb{Z}\langle\sqrt{-1}, \sqrt{-p}\rangle$, corresponding to the elliptic curve of j-invariant 1728 when $p = 3 \bmod 4$, is one of the simplest examples of such special extremal orders, as it contains the suborder $\mathbb{Z}[\sqrt{-1}] + (\sqrt{-p})\mathbb{Z}[\sqrt{-1}]$. For the rest of this paper, we fix these notations for j, R, ω. The method of resolution resulting in Algorithm 1 is inspired by [22, Lemma 5]. We introduce here a reformulation of this lemma using notations that we will keep for the rest of this article.

Lemma 1. *For any integral ideal I, the map $\chi_I(\alpha) = I\overline{\alpha}/n(I)$ is a surjection from $I \setminus \{0\}$ to the set of ideals J equivalent to I. For $\alpha \neq \beta$, we have $\chi_I(\alpha) = \chi_I(\beta)$ if and only if $\alpha = \beta\delta$ where $\delta \in \mathcal{O}_R(I)^\times$.*

Proof. This map is well-defined as proved in [22]. We see that it is a surjection by identifying $\overline{I} \cdot J$ with a principal ideal $\mathcal{O}_R(I)\overline{\beta}$. Then, it is clear that $\beta \in I$ and $J = \chi_I(\beta)$. Finally, one can verify that $\mathcal{O}_R(I)\beta_1 = \mathcal{O}_R(I)\beta_2$ if and only if $\beta_1 = \delta\beta_2$ where $\delta \in \mathcal{O}_R(I)^\times$.

With $n(\chi_I(\alpha)) = n(\alpha)/n(I)$, we see that finding $J \sim I$ of given norm N is equivalent to finding some $\alpha \in I$ of norm $n(I)N$. This observation underlies the solution of [22] for Algorithm 1.

Remark 2. In what follows will often define a projective point $(C_0 : D_0) \in \mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$ for some prime N and then, by an abuse of notation, define an element $C_0 + \omega D_0$ inside our maximal order.

Below we list sub-algorithms introduced in [22] as part of KLPT; see [22,25,13] for detailed descriptions of each.

- EquivalentPrimeIdeal(I) Given a left $\mathcal{O}_0$-ideal I, find an equivalent left $\mathcal{O}_0$-ideal of prime norm.
- $\mathsf{RepresentInteger}_{\mathcal{O}_0}(M)$ Given $M \in \mathbb{N}$ with $M > p$, find $\gamma \in \mathcal{O}_0$ of norm M.
- $\mathsf{IdealModConstraint}(I, \gamma)$ Given an ideal I of norm N, and $\gamma \in \mathcal{O}_0$ of norm Nn, find $(C_0 : D_0) \in \mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$ such that $\mu_0 = j(C_0 + \omega D_0)$ verifies $\gamma\mu_0 \in I$.
- $\mathsf{StrongApproximation}_{\mathsf{F}}(N, C_0, D_0)$ Given a prime N and $C_0, D_0 \in \mathbb{Z}$, find $\mu = \lambda\mu_0 + N\mu_1 \in \mathcal{O}_0$ of norm dividing F, with $\mu_0 = j(C_0 + \omega D_0)$. We write $\mathsf{StrongApproximation}_{\ell^\bullet}$ when the expected norm is a power of ℓ.

Remark 3. For our scheme, we will need to turn KLPT into a deterministic algorithm. The sub-routine EquivalentPrimeIdeal can be made deterministic if we look for the ideal of smallest norm satisfying the desired condition. Since we are looking at lattices of dimension at most 4, finding an ordered set of smallest vectors can be done efficiently. StrongApproximation can also be made deterministic, as the method in [25] involves solving a closest vector problem in some lattice. The sub-routine IdealModConstraint is deterministic as was shown in [22]. For $\mathsf{RepresentInteger}_{\mathcal{O}_0}$, this is less natural as there are several solutions for a given input M. Still, if we want, we can find an ordering for the tuple (x, y, z, t) of coordinates over $\mathbb{Z}\langle\omega, j\rangle$ and search for the smallest solution with respect to that ordering.

With these sub-routines we are able to give a compact description of the KLPT algorithm. There are several versions of this algorithm depending on the norm sought for the output: we will write $\mathsf{KLPT}_{\ell^\bullet}$ when the algorithm produces an output of norm a power of ℓ; KLPT_T when the norm is a divisor of $T \in \mathbb{Z}$. The changes between the two variants are minimal; for simplicity, we describe only $\mathsf{KLPT}_{\ell^\bullet}$ in Algorithm 1.

Remark 4. A result of [19] shows that the outputs of EquivalentPrimeIdeal and KLPT only depend on the equivalence class of the input (in fact this is only true with a minor tweak to the original algorithm of [22]). Hence, we will sometimes abuse notations and use both as if they took inputs in $\mathrm{Cl}(\mathcal{O}_0)$.

Algorithm 1 $\mathsf{KLPT}_{\ell^\bullet}(I)$

Require: I a left $\mathcal{O}_0$-ideal.
Ensure: $J \sim I$ of norm ℓ^e.
1: Compute $L = \mathsf{EquivalentPrimeIdeal}(I)$, $L = \chi_I(\delta)$ for $\delta \in I$ with $N = n(L)$.
2: Compute $\gamma = \mathsf{RepresentInteger}_{\mathcal{O}_0}(N\ell^{e_0})$ for $e_0 \in \mathbb{N}$.
3: Compute $(C_0 : D_0) = \mathsf{IdealModConstraint}(L, \gamma)$.
4: Compute $\nu = \mathsf{StrongApproximation}_{\ell^\bullet}(N, C_0, D_0))$ and set $\beta = \gamma\nu$ and e such that $n(\beta) = N\ell^e$.
5: **return** $J = \chi_L(\beta)$.

3 New identification protocol and signature scheme

3.1 An identification protocol

Let λ be a security parameter. We start by describing an interactive identification protocol based on supersingular isogeny problems.

$\mathsf{setup} : \lambda \mapsto \mathsf{param}$ Pick a prime number p and a supersingular elliptic curve E_0 defined over $\mathbb{F}_p$ with known special extremal endomorphism ring $\mathcal{O}_0$. Select an odd smooth number D_c of λ bits and $D = 2^e$ where e is above the diameter of the supersingular 2-isogeny graph.

$\mathsf{keygen} : \mathsf{param} \mapsto (\mathsf{pk} = E_A, \mathsf{sk} = \tau)$ Pick a random isogeny walk $\tau : E_0 \to E_A$, leading to a random elliptic curve E_A. The public key is E_A, and the secret key is the isogeny τ.

To prove knowledge of the secret τ, the prover engages in the following Σ-protocol with the verifier.

Commitment The prover generates a random (secret) isogeny walk $\psi : E_0 \to E_1$, and sends E_1 to the verifier.

Challenge The verifier sends the description of a cyclic isogeny $\varphi : E_1 \to E_2$ of degree D_c to the prover.

Response From the isogeny $\varphi \circ \psi \circ \hat{\tau} : E_A \to E_2$, the prover constructs a new isogeny $\sigma : E_A \to E_2$ of degree D such that $\hat{\varphi} \circ \sigma$ is cyclic, and sends σ to the verifier.

Verification The verifier accepts if σ is an isogeny of degree D from E_A to E_2 and $\hat{\varphi} \circ \sigma$ is cyclic. They reject otherwise.

We summarize the protocol in Fig. 1. Completeness follows from the correctness of Algorithm 3, allowing a honest prover to construct $\sigma : E_A \to E_2$ such that $\hat{\varphi} \circ \sigma$ is cyclic. Soundness is analysed in Section 3.2, and follows from the difficulty of the Smooth Endomorphism Problem — a problem heuristically equivalent to the classic Endomorphism Ring Problem. Zero-knowledge is more difficult to prove, as we argue in Section 3.3, and we defer its analysis to Section 7.

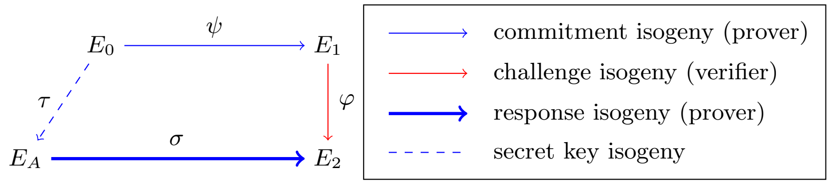

Fig. 1. A picture of the identification protocol

3.2 Soundness

Problem 1 (Supersingular Smooth Endomorphism Problem). Given a prime p and a supersingular elliptic curve E over $\mathbb{F}_{p^2}$, find a cyclic endomorphism of E of smooth degree.

Remark 5. Note that under heuristics similar to those used in [17], the above problem is equivalent to the Endomorphism Ring Problem (given $E/\mathbb{F}_{p^2}$, compute endomorphisms forming a $\mathbb{Z}$-basis of $\mathrm{End}(E)$).

Theorem 1 (Soundness). *If there is an adversary that breaks the soundness of the protocol with probability w and expected running time r for the public key E_A, then there is an algorithm for the Supersingular Smooth Endomorphism Problem on E_A with expected running time $O(r/(w-1/c))$, where c is the size of the challenge space.*

The theorem is a consequence of the following lemma.

Lemma 2. *Given two accepting conversations (E_1, φ, σ) and (E_1, φ', σ') where $\varphi \neq \varphi'$, the composition $\hat{\sigma}' \circ \varphi' \circ \hat{\varphi} \circ \sigma$ is a non-scalar endomorphism of E_A of smooth degree.*

Proof. By construction, $\hat{\sigma}' \circ \varphi' \circ \hat{\varphi} \circ \sigma$ is an endomorphism of E_A of degree $(DD_c)^2$. This shows that the degree is smooth. It remains to prove that it is not a scalar. Suppose by contradiction that $\hat{\sigma}' \circ \varphi' \circ \hat{\varphi} \circ \sigma = [DD_c]$. The compositions $\hat{\varphi} \circ \sigma$ and $\hat{\varphi}' \circ \sigma'$ are two cyclic isogenies from E_A to E_1 of same degree. Therefore $\hat{\sigma}' \circ \varphi'$ is the dual of $\hat{\varphi} \circ \sigma$. We deduce that $\hat{\varphi} \circ \sigma = \hat{\varphi}' \circ \sigma'$, a contradiction.

Proof of Theorem 1. The endomorphism $\hat{\sigma}' \circ \varphi' \circ \hat{\varphi} \circ \sigma$ in Lemma 2 corresponds to a (possibly backtracking) sequence of isogenies, and removing the backtracking subsequences, we obtain a solution to the Supersingular Smooth Endomorphism Problem of E_A. Therefore the protocol has *special soundness* for the relation R defined as

$$(E_A, \alpha) \in R \Longleftrightarrow \alpha \text{ is a cyclic smooth degree endomorphism of } E_A.$$

It is therefore a proof of knowledge for R with knowledge error $1/c$ — see for instance [11, Theorem 1]. In other words, an adversarial prover with success probability w and running time r can be turned into a knowledge extractor for R of expected running time $O(r/(w-1/c))$. □

3.3 Zero-knowledge: two insecure approaches

The sketch given in Section 3.1 is incomplete, as it does not specify a method to compute the response isogeny σ. Zero-knwoledge of the scheme clearly depends upon this method, and it turns out that the only known solutions so far are insecure. Indeed the trivial approach of setting $\sigma = \varphi \circ \psi \circ \hat{\tau}$ immediately reveals the secret, while using the algorithm from [22] instead (like in [19]) ends up revealing some path from E_A to E_0, which is equivalent to revealing τ thanks to the reductions in [17].

In Sections 5 and 6 we will introduce a new variant of the KLPT algorithm that conjecturally does not suffer from the same leakages. Then, we will prove zero-knowledge in Section 7, under a new conjecturally hard computational problem.

3.4 The signature scheme

The new signature scheme is simply a Fiat-Shamir transformation of the identification protocol introduced in Section 3.1. Following the construction of [6] extended in [28] for smooth degrees, if $D_c = \prod_{i=1}^{n} \ell_i^{e_i}$, we write $\mu(D_c) = \prod_{i=1}^{n} \ell_i^{e_i-1}(\ell_i + 1)$ and we define an arbitrary function $\Phi_{D_c}(E, s)$, mapping integers $s \in [1, \mu(D_c)]$ to non-backtracking sequences of isogenies of total degree D_c starting at E. Let $H : \{0,1\}^* \to [1, \mu(D_c)]$ be a cryptographically secure hash function.

The signature scheme is as follows.

$\mathsf{sign} : (\mathsf{sk}, m) \mapsto \Sigma$ Pick a random (secret) isogeny $\psi : E_0 \to E_1$. Let $s = H(j(E_1), m)$, and build the isogeny $\Phi_{D_c}(E_1, s) = \varphi : E_1 \to E_2$. From the knowledge of $\mathcal{O}_A$, and of the isogeny $\varphi \circ \psi : E_0 \to E_2$, construct an isogeny $\sigma : E_A \to E_2$ of degree D such that $\hat{\varphi} \circ \sigma$ is cyclic. The signature is the pair (E_1, σ).

$\mathsf{verify} : (\mathsf{pk}, m, \Sigma) \mapsto \mathsf{true}$ **or** false Parse Σ as (E_1, σ). From $s = H(j(E_1), m)$, recover the isogeny $\Phi_{D_c}(E_1, s) = \varphi : E_1 \to E_2$. Check that σ is an isogeny from E_A to E_2 and that $\hat{\varphi} \circ \sigma$ is cyclic.

Theorem 2. *The signature described above is secure against chosen-message attacks in the random oracle model assuming the hardness of Problems 1 and 2.*

4 Eichler orders and the Deuring correspondence

We recall here the notion of Eichler orders and we interpret them under the Deuring correspondence. As the results of this section are well known, we only state the main theorems without proof here; for a detailed treatment see the extended version of this work [13], or [16,26,31].

An *Eichler order* is the intersection of two maximal orders inside $\mathcal{B}_{p,\infty}$. In all this section we will consider the case of the Eichler order $\mathfrak{O} = \mathcal{O}_0 \cap \mathcal{O}$ where $\mathcal{O}_0$ and $\mathcal{O}$ are maximal orders connected through an ideal I of norm $n(I)$ such that

$I \not\subseteq n\mathcal{O}_L(I)$ for any $n > 1$. This setting corresponds to curves E_0, E connected by an isogeny φ_I of cyclic kernel and degree $n(I)$ with $\mathrm{End}(E_0) \cong \mathcal{O}_0$ and $\mathrm{End}(E) \cong \mathcal{O}$.

Proposition 1. $\mathfrak{O} := \mathcal{O}_0 \cap \mathcal{O} = \mathcal{O}_L(I) \cap \mathcal{O}_R(I) = \mathbb{Z} + I$.

One goal of this section is to interpret the elements in $\mathfrak{O}$ under the Deuring correspondence.

The decomposition $\mathbb{Z}+I$ allows us to interpret the elements of $\mathfrak{O}$. In fact, we can separate elements in $\mathfrak{O}$ according to whether their norm is coprime to $n(I)$ or not. Given that $n(I)\mathbb{Z} \subset I$, it is easily verified that this partition can be written as $\mathfrak{O} = (I \cup \overline{I}) \bigcup (\mathbb{Z} \setminus n(I)\mathbb{Z} + I)$. It is well-known that $I = \mathrm{Hom}(E, E_0)\varphi_I$. Hence, the elements in I correspond to the endomorphisms $\psi \circ \varphi_I$ for any isogeny $\psi : E \to E_0$. The same analysis proves $\overline{I} = \mathrm{Hom}(E_0, E)\hat{\varphi}_I$. The elements of $\overline{I}$ correspond to the same endomorphisms as those of I, but decomposed as $\hat{\psi} \circ \hat{\varphi}_I$ in $\mathrm{End}(E)$.

4.1 Commutative Isogeny Diagrams

We define commutative diagrams of isogenies using the classical notations of *pushforward* and *pullback* maps. Let us take 3 curves E_0, E_1, E_2 and two separable isogenies $\varphi_1 : E_0 \to E_1$ and $\varphi_2 : E_0 \to E_2$ of coprime degrees, N_1 and N_2. Then, there is a fourth curve E_3 and two *pushforward isogenies* $[\varphi_1]_*\varphi_2$ and $[\varphi_2]_*\varphi_1$ going from E_1 and E_2 toward E_3, verifying $\deg([\varphi_1]_*\varphi_2) = N_2$ and $\deg([\varphi_2]_*\varphi_1) = N_1$.

The isogenies $[\varphi_2]_*\varphi_1$ and $[\varphi_1]_*\varphi_2$ are defined as the separable isogenies of respective kernels $\varphi_2\big(\ker(\varphi_1)\big)$ and $\varphi_1\big(\ker(\varphi_2)\big)$. We will sometimes refer to $[\varphi_2]_*\varphi_1$ as *the image of* φ_1 *through* φ_2. The two sides of the diagram can be seen as two decompositions of the same isogeny $\psi = [\varphi_2]_*\varphi_1 \circ \varphi_2 = [\varphi_1]_*\varphi_2 \circ \varphi_1$.

There is a dual notion of *pullback isogeny*: given $\varphi_1 : E_0 \to E_1$ and $\rho_2 : E_1 \to E_3$, of coprime degrees, we can define the pullback of ρ_2 by φ_1 as $[\varphi_1]^*\rho_2 = [\hat{\varphi}_1]_*\rho_2$. With this definition it is easy to see that $\varphi_2 = [\varphi_1]^*[\varphi_1]_*\varphi_2$.

For simplicity, when the isogenies have not been defined we will implicitly write $[I]_*J$ for the ideal $I_{[\varphi_J]_*\varphi_I}$ corresponding to the pushforward of φ_J by φ_I. The same holds for $[I]^*J$. With this convention, we extend the terms *pushforward* and *pullback* to ideals.

4.2 The endomorphism ring $\mathfrak{O}$

The next proposition states that the image through φ of the endomorphism corresponding to any element in $\mathfrak{O} \subset \mathcal{O}_0$ (which is neither in I nor in $\bar{I}$) is an endomorphism of E.

Proposition 2. *Let* $\beta \in \mathcal{O}_0$ *of norm coprime with* N*, then* $[\mathcal{O}_0\beta]_*I = I$ *if and only if* $\beta \in \mathfrak{O} \setminus (I \cup \overline{I})$*. In particular,* $[I]_*\mathcal{O}_0\beta$ *is a principal* $\mathcal{O}$*-ideal equal to* $\mathcal{O}\beta$*.*

Said otherwise, the endomorphisms in $\mathfrak{O} \setminus (I \cup \bar{I})$ leave φ_I stable. Equivalently, the endomorphisms of $\mathfrak{O}$ remain endomorphisms after being pushed forward by φ_I, and thus belong to both $\mathrm{End}(E_0)$ and $\mathrm{End}(E)$.

From Proposition 2, we deduce the following result which will underlie Algorithm 3; it is a reformulation using the map χ of Lemma 1.

Corollary 1. *Let J_1, J_2 be $\mathcal{O}_0$-ideals, with $J_1 \sim J_2$ and $\gcd(n(J_1)n(J_2), n(I)) = 1$. Suppose that $J_1 = \chi_{J_2}(\beta)$ with $\beta \in J_2 \cap \mathfrak{O}$. Then $[I]_* J_1 \sim [I]_* J_2$ and $[I]_* J_1 = \chi_{[I]_* J_2}(\beta)$.*

4.3 Ideal class sets of Eichler orders

In this section, we write again $\mathfrak{O} = \mathcal{O}_0 \cap \mathcal{O}$. We write I for the ideal connecting $\mathcal{O}_0$ and $\mathcal{O}$ and we assume in this section that its norm N is prime.

Class sets of ideals play an important role through the Deuring correspondence. When $\mathcal{O}$ is a maximal order we can put $\mathrm{Cl}(\mathcal{O})$ in bijection with the set of supersingular curves (see Table 1). This motivates studying Eichler orders, and indeed isogeny graphs were first constructed through class sets of quaternion orders by [27], and only later reinterpreted as isogeny graphs in [6]. Eichler [16] proved a formula for the class number $h(\mathfrak{O}) = |\,\mathrm{Cl}(\mathfrak{O})|$. When N is prime it gives

$$h(\mathfrak{O}) = \frac{(p+1)(N+1)}{12} + \varepsilon_{N,p}$$

where $\varepsilon_{N,p}$ is a small value depending on N and p modulo 12. This, combined with $h(\mathcal{O}_0) = p/12 + \varepsilon_p$, ($\varepsilon_p$ depends on the value $p \mod 12$) suggests that there is a $(N+1)$-to-1 correspondence between $\mathrm{Cl}(\mathfrak{O})$ and $\mathrm{Cl}(\mathcal{O}_0)$, which we are now going to exhibit.

Let us write $\mathcal{I}_N(\mathcal{O})$ for the set of left integral $\mathcal{O}$-ideals of norm coprime to N for any order $\mathcal{O}$. We start by showing a connection between $\mathcal{I}_N(\mathcal{O}_0)$ and $\mathcal{I}_N(\mathfrak{O})$.

Lemma 3. *The map*

$$\begin{aligned} \Psi : \mathcal{I}_N(\mathcal{O}_0) &\longrightarrow \mathcal{I}_N(\mathfrak{O}) \\ J &\longmapsto J \cap \mathfrak{O} \end{aligned}$$

is a well-defined bijection between the set of integral $\mathcal{O}_0$-ideals and $\mathfrak{O}$-ideals of norm coprime with N. Its inverse is given by : $\Psi^{-1} : \mathfrak{J} \mapsto \mathcal{O}_0 \mathfrak{J}$.

From the fact that any ideal class of $\mathrm{Cl}(\mathfrak{O})$ or $\mathrm{Cl}(\mathcal{O}_0)$ has a representative of norm coprime with N, we can easily identify the equivalence classes of $\mathcal{I}_N(\mathcal{O}_0)$ and $\mathcal{I}_N(\mathfrak{O})$ to the ones of $\mathcal{O}_0$ and $\mathfrak{O}$ respectively.

The bijection of Lemma 3 suggests defining the following equivalence relation $\sim_{\mathfrak{O}}$ on left $\mathcal{O}_0$-ideals of norm coprime with N. We say that $J \sim_{\mathfrak{O}} K$ if and only if $\Psi(J) \sim \Psi(K)$ as $\mathfrak{O}$-ideals (here $\sim$ is the classical equivalence relation between ideals having the same left order). The bijection Ψ transports the structure of $\sim$ to $\sim_{\mathfrak{O}}$ and this implies that we have defined an equivalence relation.

Definition 1. *We write* $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{O}_0)$ *for the set of equivalence classes of* $\mathcal{I}_N(\mathcal{O}_0)$ *under* $\sim_{\mathfrak{O}}$.

From the definition, we have that $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{O}_0)$ is in bijection with $\mathrm{Cl}(\mathfrak{O})$ through Ψ. In the next proposition we show that we can obtain an explicit correspondence between ideals of norm N and $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{O}_0)$ using pushforward ideals.

Proposition 3. $J \sim_{\mathfrak{O}} K$ *if and only if there exists* $\beta \in \mathfrak{O}$ *such that* $K = \chi_J(\beta)$ *and* $\beta^{-1}[K]_* I \beta = [J]_* I$.

An interesting question is how the new equivalence relation $\sim_{\mathfrak{O}}$ relates to the classical one $\sim$. In fact, $\sim_{\mathfrak{O}}$ is compatible with $\sim$ in the sense that $J \sim_{\mathfrak{O}} K$ implies $J \sim K$, as is easily verified from Corollary 1. This suggests partitioning $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{O}_0)$ in subsets indexed by the elements of $\mathrm{Cl}(\mathcal{O}_0)$. Hence, we write $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{O}_0) = \bigcup_{\mathcal{C} \in \mathrm{Cl}(\mathcal{O}_0)} \mathrm{Cl}_{\mathfrak{O}}(\mathcal{C})$ where $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{C})$ is the set of classes in $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{O}_0)$ contained in $\mathcal{C}$. The respective sizes of $\mathrm{Cl}(\mathcal{O}_0)$ and $\mathrm{Cl}(\mathfrak{O})$ suggest that the partition above provides an $(N+1)$-to-1 correspondence between $\mathrm{Cl}(\mathcal{O}_0)$ and $\mathrm{Cl}(\mathfrak{O})$. This correspondence only fails for a small number of classes, as the following proposition shows.

Proposition 4. *For* $\mathcal{C} \in \mathrm{Cl}(\mathcal{O}_0)$, *let us take* $L \in \mathcal{C}$ *and define* $\mathcal{O}_{\mathcal{C}} := \mathcal{O}_R(L)$. *If* $\mathcal{O}_{\mathcal{C}}^{\times} = \langle \pm 1 \rangle$, *then for any* $\gamma \in L \setminus N\mathcal{O}_0$ *and quadratic order* $S = \mathbb{Z}[\omega_s]$ *of discriminant* Δ_S *inside* $\mathcal{O}_0$ *in which* N *is inert, the map:*

$$\begin{aligned} \Theta : \mathbb{P}^1(\mathbb{Z}/N\mathbb{Z}) &\longrightarrow \mathrm{Cl}_{\mathfrak{O}}(\mathcal{C}) \\ (C : D) &\longmapsto \chi_L((C + \omega_s D)\gamma) \end{aligned}$$

is a bijection. In particular, $|\,\mathrm{Cl}_{\mathfrak{O}}(\mathcal{C})| = N + 1$.

5 New generalized KLPT algorithm

We introduce in this section a new algorithm to perform the computation of the response in our identification protocol. We aim at solving the issues raised in Section 3.3 with the original KLPT algorithm [22].

The existence of the suborder $\mathfrak{O} = Z\langle \omega, j \rangle = R + Rj$ introduced in Section 2.2 is what makes special extremal orders good candidates for applying the KLPT algorithm. Here, $R = \mathbb{Z}[\omega]$ is a quadratic order of small discriminant generated by ω, an element of small norm. The norm equation $f(x, y) = M$ over R has a good probability of being solvable for any M and as a consequence, solving norm equations over $\mathfrak{O}$ is easy.

To extend the KLPT algorithm to arbitrary orders, our approach is to find an appropriate Eichler suborder in which we know how to solve norm equations. More precisely, let us take $\mathcal{O}_0$ a special extremal order and $\mathcal{O}$ an arbitrary maximal order, our goal is to extend the KLPT algorithm to left $\mathcal{O}$-ideals. Then, the Eichler order $\mathfrak{O} = \mathcal{O} \cap \mathcal{O}_0$ is a suborder of $\mathcal{O}_0$, thus we can apply the techniques developed in [22] for special extremal orders.

5.1 The generic algorithm

We now use our observations of Section 4 to design a new GeneralizedKLPT algorithm. As already mentioned, there are several possible variants of this algorithm depending on the kind of norm we need to obtain. For simplicity, we present the case $\ell^\bullet$ where we look for an equivalent ideal of norm ℓ^e. Any other variant is easily derived from this.

For the rest of this paper, let $\mathcal{O}_0$ and $\mathcal{O}$ be two maximal orders, with $\mathcal{O}_0$ being special extremal. These maximal orders are respectively isomorphic to the endomorphism rings of two supersingular curves E_0 and E. From now on, we write I_τ (instead of I in the previous section) for the ideal connecting $\mathcal{O}_0$ with $\mathcal{O}$, and we denote its norm by N_τ. This notation is motivated by the fact that, in the signature context, I_τ will be the ideal corresponding to the secret isogeny τ of degree N_τ. Up to replacing $\mathcal{O}$ with an isomorphic representative, we can assume that N_τ is prime and inert in R (we explain in Section 6.2, the reasons behind this last condition). We consider the Eichler order $\mathfrak{O} = \mathcal{O} \cap \mathcal{O}_0$ of level N_τ.

Let I be a left integral $\mathcal{O}$-ideal, given as input. Our purpose is to find $e \in \mathbb{N}$ and $J \sim I$ of norm ℓ^e upon input I. As a consequence of Lemma 1, this problem is equivalent to finding $\beta \in I$ of norm $n(I)\ell^e$ and setting $J = \chi_I(\beta)$. From Corollary 1, we see that if $\beta \in I \cap \mathfrak{O}$ we have $[I_\tau]^* J = \chi_{[I_\tau]^* I}(\beta)$. In particular, $\beta \in \mathfrak{O} \cap [I_\tau]^* I$ and so we can search for β inside $([I_\tau]^* I) \cap \mathfrak{O}$ instead. The ideal $K' := [I_\tau]^* I$ is a left $\mathcal{O}_0$-ideal and this is a situation close to $\mathsf{KLPT}_{\ell^\bullet}$. The fact that we look for a solution inside $K' \cap \mathfrak{O}$ instead of just K' will add an additional constraint. Proposition 1 allows us to write $\mathfrak{O} = \mathbb{Z} + I_\tau$, and intuitively this decomposition tells us that the algorithm for integral ideals used in [22] will be applicable to Eichler orders with small changes.

This suggests the method detailed in Algorithm 2, which can be seen as an adaptation of the $\mathsf{KLPT}_{\ell^\bullet}$ algorithm (Algorithm 1), replacing the input I by $I \cap \mathfrak{O}$. In $\mathsf{KLPT}_{\ell^\bullet}$ we satisfy the constraint that the desired element is in I using the sub-algorithm IdealModConstraint. We proceed similarly in Step 4 to ensure that the solution is in $\mathfrak{O}$ as well. Combining the two constraints ensures that the solution is in their intersection. An algorithm to perform Step 4 will be described in Section 6.2; its description is not needed to convey the principle of Algorithm 2. We omit the extension of StrongApproximation to the case where N is not prime; the interested reader will find it in the extended paper [13].

Lemma 4. *Algorithm 2 is correct and returns $J \sim I$ of norm ℓ^e.*

Proof. We assume here that the algorithm terminates without failure and do not consider its complexity for now. First, Lemma 1 and the conservation of the norm through pushforward ideals shows that J has norm ℓ^e. Then Corollary 1 applied to $\chi_L(\beta) = \chi_{K'}\left(\frac{\overline{\beta\delta}}{n(L)}\right)$ implies that $[I_\tau]_* \chi_L(\beta) \sim [I_\tau]_* K$ since $\beta\delta \in \mathfrak{O}$. This proves $J \sim I$.

Remark 6. As pointed out in Remark 3, KLPT is essentially deterministic when one looks for the smallest possible solution with this method. Given that the

Algorithm 2 $\mathsf{GeneralizedKLPT}_{\ell^\bullet}(I, I_\tau)$

Require: I, a left $\mathcal{O}$-ideal, and I_τ, a left $\mathcal{O}_0$-ideal and right $\mathcal{O}$-ideal of norm N_τ.
Ensure: $J \sim I$ of norm ℓ^e.
1: Compute $K' = [I_\tau]^* I$ and set $L = \mathsf{EquivalentPrimeIdeal}(K')$, $L = \chi_{K'}(\delta)$ for $\delta \in K'$ with $N = n(L)$.
2: Compute $\gamma = \mathsf{RepresentInteger}_{\mathcal{O}_0}(N\ell^{e_0})$.
3: Compute $(C_0 : D_0) = \mathsf{IdealModConstraint}(L, \gamma)$.
4: Find $(C_1 : D_1) \in \mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z})$ such that $\gamma j(C_1 + \omega D_1)\delta \in \mathbb{Z} + I_\tau$.
5: Compute $C = \mathsf{CRT}_{N_\tau,N}(C_0, C_1)$ and $D = \mathsf{CRT}_{N_\tau,N}(D_0, D_1)$.
6: Compute $\mu = \mathsf{StrongApproximation}_{\ell^\bullet}(NN_\tau, C, D)$ of norm ℓ^{e_1}
7: Set $\beta = \gamma\mu$ and $e = e_0 + e_1$ such that $n(\beta) = N\ell^e$.
8: **return** $J = [I_\tau]_* \chi_L(\beta)$.

only major difference in Algorithm 2 is the additional Step 4 (for which there is only one solution as we will see in Section 6.2) it is not difficult to argue that Algorithm 2 can be made deterministic.

5.2 On the length of the solution

The length of the output of Algorithm 2 can be derived from the one of $\mathsf{KLPT}_{\ell^\bullet}$. Indeed, in terms of norm, the only real difference is the fact that the StrongApproximation is performed on NN_τ instead of just N. From the analysis provided in [22] and [25], we see that this implies $e = e_0 + e_1 \sim \frac{9}{2}\log_\ell(p)$ (instead of $e \sim 3\log_\ell(p)$ for $\mathsf{KLPT}_{\ell^\bullet}$). This estimate is obtained by considering the plausible approximation $N_\tau \sim \sqrt{p}$. We will argue in Section 7.1 that it might be acceptable to consider cases where N_τ is significantly smaller than this average estimate. This allows us to decrease the size of the solution. We give in Section 6.3 a more proper statement for the approximations introduced above.

In our signature scheme, we will use a variant of Algorithm 2, called SigningKLPT, suited for our application. The purpose of Section 6 is to detail this algorithm and to fill in the gaps left in the description of Algorithm 2.

6 Application to the signature scheme: the SigningKLPT algorithm

In this section, we describe the SigningKLPT procedure used in our signature scheme. This procedure, described in Algorithm 3, is a variant of Algorithm 2. Most of its building blocks are common to Algorithm 1 and were introduced in [22]. The rest of this section fills in the remaining gaps as follows. In Section 6.1, we introduce the EquivalentRandomEichlerIdeal used in Step 1. In Section 6.2, we describe the EichlerModConstraint algorithm to perform Step 5 of Algorithm 3 (or Step 4 in Algorithm 2). The parameter e is fixed (and it only depends on p). To ensure this, we will need to adapt the exponent e_0 and e_1 to the values $N = n(L)$ and N_τ. That is why we will write $e_0(N)$. In Section 6.3 we justify that this is possible. We establish the termination, correctness and complexity of our algorithm in Section 6.4.

Algorithm 3 $\mathsf{SigningKLPT}(I, I_\tau)$

Require: I_τ a left $\mathcal{O}_0$-ideal and right $\mathcal{O}$-ideal of norm N_τ, and I, a left O-ideal.
Ensure: $J \sim I$ of norm ℓ^e, where e is fixed.
1: Compute $K = \mathsf{EquivalentRandomEichlerIdeal}(I, N_\tau)$
2: Compute $K' = [I_\tau]^* K$ and set $L = \mathsf{EquivalentPrimeIdeal}(K')$, $L = \chi_{K'}(\delta)$ for $\delta \in K'$ with $N = n(L)$. Set $e_0 = e_0(N)$ and $e_1 = e - e_0$.
3: Compute $\gamma = \mathsf{RepresentInteger}_{\mathcal{O}_0}(N\ell^{e_0})$.
4: Compute $(C_0 : D_0) = \mathsf{IdealModConstraint}(L, \gamma)$.
5: Compute $(C_1 : D_1) = \mathsf{EichlerModConstraint}(\mathbb{Z} + I_\tau, \gamma, \delta)$.
6: Compute $C = \mathsf{CRT}_{N_\tau, N}(C_0, C_1)$ and $D = \mathsf{CRT}_{N_\tau, N}(D_0, D_1)$. If $\ell^e p(C^2 + D^2)$ is not a quadratic residue, go back to Step 3.
7: Compute $\mu = \mathsf{StrongApproximation}_{\ell^\bullet}(NN_\tau, C, D)$ of norm ℓ^{e_1}
8: Set $\beta = \gamma\mu$.
9: **return** $J = [I_\tau]_* \chi_L(\beta)$.

6.1 The randomization procedure

The purpose of Step 1 is to perform a randomization step which we will use to argue the security of our signature. This addition has two interesting consequences for us. First, the output of Algorithm 3 only depends on the equivalence class of the input I. Second, it randomizes the execution as otherwise the algorithm would be essentially deterministic as noted in Remark 6.

The $\mathsf{EquivalentRandomEichlerIdeal}$ algorithm receives an ideal I as input and returns an equivalent random ideal. In this context equivalent random ideal means that if we write $\mathcal{C}$ the class of I in $\mathrm{Cl}(\mathcal{O})$, we want an output ideal equivalent to I and lying in a uniformly random class of $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{C})$ (see Definition 1). This condition might seem a bit arbitrary at first; however Proposition 5 will justify that this is exactly the kind of randomness we need.

To reach this goal, we use the classical technique of finding some well-chosen $\beta \in I$ and output $\chi_I(\beta)$. The method to choose the β is inspired by the results of Section 4.3. The idea is to use the bijection from Proposition 4 in order to sample a class uniformly. Note that Proposition 4 does not hold for some special cases of maximal orders $\mathcal{O}$, but we may assume that this is not the case here (in the worst case there are two such types of maximal orders among $O(p)$ possibilities).

We start by showing that Algorithm 4 terminates and that the output distribution is correct.

Lemma 5. *Algorithm 4 terminates in polynomial time and outputs an ideal equivalent to I and uniformly distributed among the $N_\tau + 1$ possible classes of $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{O})$.*

Algorithm 4 EquivalentRandomEichlerIdeal(I, N_τ)

Require: I a left $\mathcal{O}$-ideal.
Ensure: $K \sim I$ of norm coprime with N_τ.
1: Sample a random element ω_S in $\mathcal{O}$ until N_τ is inert in $\mathbb{Z}[\omega_S]$.
2: Sample γ a random element in I such that $n(\gamma)/n(I)$ is coprime with N_τ.
3: Select a random class $(C : D) \in \mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z})$.
4: Set $\beta = (C + \omega_S D)\gamma$.
5: **return** $K = \chi_I(\beta)$

Proof. We can find in $O(\log(p))$ attempts a quadratic suborder $\mathbb{Z}[\omega_S] \subset \mathcal{O}$ in which N_τ is inert. Then, it is clear that taking a random element in I will verify that $n(\gamma)/n(I)$ is coprime with N_τ with overwhelming probability. Thus, the algorithm terminates in polynomial time.

The algorithm concretely instantiates the map Θ from Proposition 4. This map is bijective and we choose $(C : D)$ uniformly at random inside $\mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z})$ so the output is uniformly distributed.

Consequently, the output of EquivalentRandomEichlerIdeal only depends on the class (inside $\mathrm{Cl}(\mathcal{O})$) of the ideal in input. The call to EquivalentRandomEichlerIdeal in Step 1 of Algorithm 3 thus implies the following lemma that will prove useful in Section 7.

Lemma 6. *For any I_τ, the output distributions of* SigningKLPT(I, I_τ) *and* SigningKLPT(J, I_τ) *are the same for any $I \sim J$. Said otherwise, for fixed I_τ, the output distribution of Algorithm 3 only depends on the equivalence class of the ideal I in input.*

Next, we describe how the distribution of L (as defined in Step 2 of Algorithm 3) is determined by the output distribution of EquivalentRandomEichlerIdeal. This is what motivates the current formulation of Algorithm 4.

Proposition 5. *The set $\mathcal{G}_I = \{L, L = \mathsf{EquivalentPrimeIdeal}([I_\tau]^*K) \text{ for } K \sim I\}$ has size at most $N_\tau + 1$ and for every $L \in \mathcal{G}_I$ there exists an output $K =$ EquivalentRandomEichlerIdeal(I) such that $L =$ EquivalentPrimeIdeal$([I_\tau]^*K)$. When $\#\mathcal{G}_I = N_\tau + 1$, the ideal L is uniformly distributed inside this set.*

Proof. As we mentioned already, there are exactly $N_\tau + 1$ classes for $K \sim I$ in $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{O})$. By Corollary 1[10], the class of K in $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{O})$ uniquely determines the class of $[I_\tau]^*K$ in $\mathrm{Cl}(\mathcal{O}_0)$. As noted in Section 2.2, the output of EquivalentPrimeIdeal is well-defined and deterministic on $\mathrm{Cl}(\mathcal{O}_0)$. The result is proved if we combine the above remark with Lemma 5.

[10] Corollary 1 uses pushforwards rather than pullbacks, but we obtain the desired result by replacing I with $\overline{I}$.

6.2 Eichler modular constraint

Step 5 in Algorithm 3 (or Step 4 of Algorithm 2) is essential to find a solution that lies in $\mathfrak{O} = \mathcal{O} \cap \mathcal{O}_0$. More precisely for given γ, δ of norm coprime with N_τ we need to find $\mu_1 \in jR$ such that $\gamma\mu_1\delta \in \mathfrak{O}$. In fact, this can be done for any γ, δ of norm coprime with N_τ. This is stated and proved in Proposition 6 below, following a reasoning similar to the one used in [22] for IdealModConstraint.

The method of resolution is also strongly inspired by IdealModConstraint. Namely, we use an explicit isomorphism $\mathcal{O}_0/N_\tau\mathcal{O}_0 \cong \mathbb{M}_2(\mathbb{Z}/N_\tau\mathbb{Z})$ and a correspondence between the set of proper nonzero left ideals in $\mathbb{M}_2(\mathbb{Z}/N_\tau\mathbb{Z})$ and $\mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z})$ to translate the condition $\gamma\mu_1\delta \in \mathbb{Z} + I_\tau$ as a system of linear equations $\bmod N_\tau$. We write $\mathsf{EichlerModConstraint}(\mathfrak{O}, \gamma, \delta)$ for this. It outputs $(C_1 : D_1) \in \mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z})$ such that $\gamma j(C_1 + \omega D_1)\delta \in \mathfrak{O}$.

We remind the reader that we consider N_τ inert in R (where R is defined, like in Section 2.2, as the quadratic suborder of minimal discriminant inside $\mathcal{O}_0$). If N_τ is split, the method is very likely to work as well but there may be some cases where it fails. Since the constraint that N_τ is inert in R is quite easy to satisfy (see Section 8.3) we may assume that it holds.

Proposition 6. *The sub-routine* EichlerModConstraint *on any input* $\mathfrak{O}, \gamma, \delta$ *returns* $(C_1 : D_1) \in \mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z})$ *such that* $\gamma\mu\delta \in \mathfrak{O}$ *with* $\mu = (C_1 + \omega D_1)j$.

Proof. In Algorithm 3, we want to find μ such that $\beta = \gamma\mu$ verifies $\beta\delta \in \mathfrak{O}$ to ensure that $[I_\tau]_*\chi_L(\beta) \sim I$. In Section 4.3, we showed that this was equivalent to $\chi_L(\beta)$ lying in the correct equivalence class of $\mathrm{Cl}(\mathfrak{O})$. To prove that a solution can always be found it suffices to show that the map $\Theta' : \mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z}) \to \mathrm{Cl}(\mathfrak{O})$ sending $(C : D)$ to $\gamma(C + \omega D)$ is surjective. In fact, this map is almost the one from Proposition 4 and is bijective (thus surjective) for the same reasons.

Hence we see that there always exists a solution μ such that $\chi_L(\gamma\mu)$ lies in the correct class in $\mathrm{Cl}_\mathfrak{O}(\mathcal{O}_0) \equiv \mathrm{Cl}(\mathfrak{O})$ and this proves the result.

We deduce a useful corollary, which shows that EichlerModConstraint is independent of the choice of δ.

Corollary 2. *Taking* δ, δ' *as above, for any given* $\gamma \in \mathcal{O}_0$ *of norm coprime with* N_τ, $\mathsf{EichlerModConstraint}(\mathfrak{O}, \gamma, \delta) = \mathsf{EichlerModConstraint}(\mathfrak{O}, \gamma, \delta')$.

Proof. In the proof of Proposition 6, we showed that the map $(C_1 : D_1) \to \gamma j(C_1 + \omega D_1)$ is injective for any γ of norm coprime with N_τ. This justifies that there is only one solution in $\mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z})$ giving a β lying in the correct class inside $L/\sim_\mathfrak{O}$ (and thus with $\chi_L(\beta)$ in the correct class of $\mathrm{Cl}_\mathfrak{O}(\mathcal{O}_0)$). Hence, $\mathsf{EichlerModConstraint}(\mathfrak{O}, \gamma, \delta)$ and $\mathsf{EichlerModConstraint}(\mathfrak{O}, \gamma, \delta')$ are both equal to this unique solution.

6.3 Suitable values for e_0 and e_1

For security (specifically zero-knowledge) it is important that our output has fixed norm so that the size of the output does not reveal any information on the input. In this section, we justify that it is possible to find a parameter e such that finding an output of exact size ℓ^e is possible for almost every input. The exponent e is the sum of two exponents $e_0(N)$ and $e_1(N, N_\tau)$ whose individual values depend on N and N_τ but whose sum can be fixed. In fact, we will pick e following the approximations of [22] presented in Section 5.2 as they appear to be quite tight in practice. To simplify notations we write log instead of $\log_\ell$ in the rest of this section. Let us refine the statements of Section 5.2. For KLPT, the most important parameter is the size of N. We state in Lemma 7 that N cannot be a lot bigger than $\sqrt{p}$. This result holds under an assumption on the norms of elements in a Minkowski basis of an integral ideal, and heuristic assumptions on the distribution of primes represented by some quadratic forms (see [22]). We stress that this approximation is quite tight in practice as illustrated in the experimental results of [22] and it seems to hold by taking $\varepsilon = \log\log(p)$.

Lemma 7. *There exists* $\varepsilon = O(\log\log(p))$ *such that for a random class* $\mathcal{C} \in \mathrm{Cl}(\mathcal{O}_0)$*, the norm* N *of* $\mathsf{EquivalentPrimeIdeal}(\mathcal{C})$ *verifies* $\log(N) < \log(p)/2 + \varepsilon$ *with overwhelming probability.*

This approximation is valid for both N and N_τ, and we will assume that it holds for both values for the rest of this section. As we will not be able to provide a tight lower bound on $\log(N), \log(N_\tau)$, we need to adjust the exponents e_0 and e_1 and that is why we write $e_0(N)$ and $e_1(N, N_\tau)$ for the lower bounds of Lemmas 8 and 9. We recall our assumption that the failure probability in the quadratic residuosity condition of Steps 6 is $3/4$ on average for a given γ and δ.

In Lemmas 8 and 9, we assume that we are in an execution of Algorithm 3 that led to an ideal L of norm N. We keep the notation ε from Lemma 7.

Lemma 8. *For any* $\kappa \in \mathbb{N}$*, there exists* $\eta_0 = O(\log\log(p) + \log(\kappa))$ *such that for any* $e_0 \geq e_0(N) = \log(p) - \log(N) + \varepsilon + \eta_0$*, the probability that there exists a solution* $\gamma = \mathsf{RepresentInteger}_{\mathcal{O}_0}(N\ell^{e_0})$ *that will lead to a correct execution of Algorithm 3 is higher than* $1 - 2^{-\kappa}$*.*

Remark 7. We note that taking $\kappa \sim \log(p)$ ensures that the success probability in Lemma 8 is overwhelming. In the case of (very unlikely) failure where one of the assumptions above does not hold, we simply abort and start the computation again.

We conclude this section by evaluating the size of the exponent e_1 in the output of StrongApproximation. The algorithm for $\mathsf{StrongApproximation}(N, \cdot)$ in [25] computes close vectors in some lattice of discriminant $\tilde{O}(N^3 p)$.

Lemma 9. *There exists* $\eta_1 = O(\log\log(p))$ *such that if* $e_1 \geq e_1(N, N_\tau) \log p + 3\log(N) + 3\log(N_\tau) + \eta_1$*, Step 7 of Algorithm 3 succeeds in finding a solution* μ *of norm* ℓ^{e_1} *with overwhelming probability.*

6.4 Termination, correctness and complexity

We are now ready to state the following proposition. As noted in Remark 7, we take $\kappa \sim \log(p)$ for Lemma 8.

Proposition 7. *Algorithm 3 terminates in heuristic probabilistic polynomial time. It returns an ideal $J \sim I$ of fixed norm ℓ^e for any input I with overwhelming probability if $e \geq 9/2\log(p) + 6\varepsilon + \eta_0 + \eta_1$ where $\varepsilon, \eta_0, \eta_1$ are defined as in Lemmas 7 to 9.*

Proof. The proof of correctness follows almost directly from Lemma 4, replacing I by an equivalent K. Since the correctness of Algorithm 2 holds for any input and $K \sim I$, we see that Algorithm 3 is correct. Combining Lemmas 8 and 9 we see that we need to pick e_0, e_1 above the bounds $e_0(N), e_1(N, N_\tau)$ for the computation to succeed with overwhelming probability. We obtain $e_0 + e_1 \geq 2\log(p) + 2\log(N) + 3\log(N_\tau) + \eta_0 + \eta_1 + \epsilon$. Taking the upper bound of Lemma 7 for both N and N_τ we obtain $e \geq 9/2\log(p) + 6\varepsilon + \eta_0 + \eta_1$. Given that the probability of failure is $3/4$, the number of different values γ that we need to choose before finding a fitting choice is logarithmic in p. This proves termination. The complexity statement follows directly from the heuristic polynomial-time complexities argued in [22]. From the description in Section 6.2, it is clear that the complexity of EichlerModConstraint is the same as IdealModConstraint and it is also polynomial in $\log(p)$.

7 Zero-Knowledge

In Section 3 we left open the question of proving zero-knowledge of the identification scheme, and consequently unforgeability of the signature scheme. Unlike other identification schemes based on isogenies [12,3], SQISign does not achieve perfect zero-knowledge, but necessitates an *ad hoc* computational assumption instead. As usual, we need to prove that there exists a simulator that outputs transcripts indistinguishable from real interactions between prover and verifier, and it is easy to see that this boils down to proving that the distribution of the response isogenies σ for a given secret τ can be simulated without knowledge of τ. Of course, the distribution of σ depends on the variant of KLPT employed, and we already argued in Section 3.3 that the variants known prior to this work provide no security at all. In this section we shall state the security assumption and sketch the associated security reduction for algorithm SigningKLPT. Due to space constraints all proofs are omitted here; they can be found in [13].

7.1 On the distribution of signatures

We want to understand the distribution of the isogenies σ obtained from $J = \mathsf{SigningKLPT}(I, I_\tau)$ for some secret τ. It turns out any such σ is the image under τ of some other isogeny ι, whose properties are precisely stated in the following lemma.

Lemma 10. *Let $L \subset \mathcal{O}$ and $\beta \in L$ be as in steps 2, 8 respectively of Algorithm 3. The isogeny σ corresponding to the output J of Algorithm 3 is equal to $\sigma = [\tau]_*\iota$, where ι is an isogeny of degree ℓ^e verifying $\beta = \hat{\iota} \circ \varphi_L$.*

We will argue that there exists a set $\mathcal{P}_{N_\tau}$, depending only on the degree N_τ, such that $\iota \in \mathcal{P}_{N_\tau}$ if and only if $\sigma = [\tau]_*\iota$ for some output σ of Algorithm 3. $L \subset \mathcal{O}$ being defined as in Lemma 10, it is clear that the codomain of ι is determined by the class of L in $\mathrm{Cl}(\mathcal{O}_0)$. Suppose we have chosen a class for L among the $N_\tau + 1$ candidates, we want to determine how the rest of the computation follows from this initial choice. During Step 3 we compute a value γ, and it is clear that $N = n(L)$ uniquely determines the distribution of outputs for $\mathsf{RepresentInteger}_{\mathcal{O}_0}(N\ell^{e_0(N)})$. Then, the projective pair $(C_0 : D_0)$ only depends on L and γ. We have proved in Corollary 2 that the projective pair $(C_1 : D_1)$ did not depend on the actual value of δ, so it is also uniquely determined by the choice of class for K (and thus of L) and γ. The rest of the computation is deterministic from there (up to failures that imply picking another γ). We are now ready to characterize the set of all possible outputs of our algorithm $\mathsf{SigningKLPT}$.

Let us take the value $e_0(N)$ and $e_1(N, N_\tau)$ as defined in Section 6.3 for Algorithm 3. For a given L of norm N, we consider $\mathcal{U}_{L,N_\tau}$ as the set of all isogenies ι computed as in Lemma 10 from elements $\beta = \gamma\mu \in L$ where γ is a random output of $\mathsf{RepresentInteger}_{\mathcal{O}_0}(N\ell^{e_0(N)})$ and $\mu = (C + \omega D)j$ where $p(C^2 + D^2)\ell^{e_1(N,N_\tau)}$ is a quadratic residue $\mathrm{mod} NN_\tau$ and is defined as $C = \mathsf{CRT}_{N,N_\tau}(C_0, C_1)$, $D = \mathsf{CRT}_{N,N_\tau}(D_0, D_1)$ where $(C_0 : D_0) = \mathsf{IdealModConstraint}(L, \gamma)$ and $(C_1 : D_1)$ is a random element of $\mathbb{P}^1(\mathbb{Z}/N_\tau\mathbb{Z})$. For an equivalence class $\mathcal{C}$ in $\mathrm{Cl}(\mathcal{O}_0)$ we write $\mathcal{U}_{\mathcal{C},N_\tau}$ for $\mathcal{U}_{L,N_\tau}$ where $L = \mathsf{EquivalentPrimeIdeal}(\mathcal{C})$.

Definition 2. $\mathcal{P}_{N_\tau} = \bigcup_{\mathcal{C}\in \mathrm{Cl}(\mathcal{O}_0)} \mathcal{U}_{\mathcal{C},N_\tau}$

Proposition 8. *The set $\mathcal{P}_{N_\tau}$ from Definition 2 can be computed from the sole knowledge of N_τ. The set $\{J, J = [I_\tau]_* I_\iota, \iota \in \mathcal{P}_{N_\tau}\}$ is exactly the set of outputs* $\mathsf{SigningKLPT}(I, I_\tau)$ *for I ranging over all the non-trivial classes in* $\mathrm{Cl}(\mathcal{O})$.

7.2 Hardness Assumption for Zero-Knowledge

We are now ready to formulate a computational assumption which zero-knowledge reduces to. For $D \in \mathbb{N}$ and a supersingular curve E, we define $\mathtt{Iso}_{D,j(E)}$ as the set of cyclic isogenies of degree D, whose domain is a curve inside the isomorphism class of E. When $\mathcal{P}$ is a subset of $\mathtt{Iso}_{D,j(E)}$ and $\tau : E \to E'$ is an isogeny with $\gcd(\deg \tau, D) = 1$, we write $[\tau]_* \mathcal{P}$ for the subset $\{[\tau]_* \phi \mid \phi \in \mathcal{P}\}$ of $\mathtt{Iso}_{D,j(E')}$. Finally, we denote by $\mathcal{K}$ a probability distribution on the set of cyclic isogenies whose domain is E_0, representing the distribution of SQISign private keys.

Problem 2. Let p be a prime, and D a smooth integer. Let $\tau : E_0 \to E_A$ be a random isogeny drawn from $\mathcal{K}$, and let N_τ be its degree. Let $\mathcal{P}_{N_\tau} \subset \mathtt{Iso}_{D,j_0}$ as in Definition 2, and let O_τ be an oracle sampling random elements in $[\tau]_*\mathcal{P}_{N_\tau}$. Let $\sigma : E_A \to \star$ of degree D where either

1. σ is uniformly random in $\mathtt{Iso}_{D,j(E_A)}$;
2. σ is uniformly random in $[\tau]_* \mathcal{P}_{N_\tau}$.

The problem is, given $p, D, \mathcal{K}, E_A, \sigma$, to distinguish between the two cases with a polynomial number of queries to O_τ.

We assume that Problem 2 cannot be solved with non-negligible advantage by any polynomial time adversary. In [13] we briefly discuss several potential attack strategies; however, given current knowledge, no strategy seems to be better than a direct key recovery, computing τ from the knowledge of E_A only.

In order to state the security reduction, we also need some additional heuristic assumptions which are plausibly true.

Assumption 1 *Under the heuristic assumptions used in Section 6.3, we can fix a given degree $D = \ell^e$ with e depending only on p, such that Algorithm 3 succeeds in finding an output of norm D for any input with overwhelming probability.*

Assumption 2 *The distribution of classes obtained by taking the classes of the ideals I_ι corresponding to $\iota \in \mathcal{P}_{N_\tau}$ is statistically close to the uniform distribution on $\mathrm{Cl}_{\mathfrak{O}}(\mathcal{O}_0)$.*

We can finally state the main result of this section.

Proposition 9. *Let E_A be a SQISign public key. When SQISign is instantiated with Algorithm 3, distinguishing between the distribution $\mathcal{D}(E_A)$ of isogenies σ output by SQISign, and the uniform distribution of D-isogenies starting from E_A, reduces to Problem 2, under the heuristic assumptions listed above.*

8 Efficiency

In this section, we describe a concrete instantiation of our scheme. This includes a precise description of the protocols outlined in Section 3.1, along with all the missing sub-algorithms, concrete parameters and various ideas to improve the overall efficiency. The resulting signature reaches 128-bit of classical security and the post-quantum NIST level 1 and is very compact as highlighted in Table 2. We also provide a proof-of-concept implementation of the protocol.

The algorithm SigningKLPT was extensively studied in Sections 5 and 6, and we will see in Section 8.6 that it is reasonably efficient. The efficiency bottleneck of our signature scheme turns out to be the translation of the input and output ideals of Algorithm 3 from and to isogenies. Specifically, we seek to define two families of algorithms:

- IdealToIsogeny: Given a left $\mathcal{O}$-ideal I of smooth norm D, compute the corresponding isogeny φ_I as a sequence of prime-degree isogenies.
- IsogenyToIdeal: Given an isogeny from E of smooth degree D, compute the corresponding left $\mathcal{O}$-ideal.

Algorithms for these tasks in the case where $\mathcal{O}$ and E are special extremal were already introduced in [19]. They are very general, but not really efficient, owing to their use of D-torsion points defined in algebraic extensions of $\mathbb{F}_{p^2}$. A classical solution would be to choose a special prime p such that the D-torsion is $\mathbb{F}_{p^2}$-rational. However in our case D is a power of 2 and, following the estimates of Section 5.2, we need $D \approx p^{9/2}$ (or at best $D \approx p^{15/4}$ using the idea of Section 8.3). With these requirements finding such a prime is not feasible, we thus devise new solutions to the two problems.

This section is organized as follows. We first present our version of IdealToIsogeny in Section 8.1. We then introduce a set of concrete parameters in Section 8.2, and we analyze two possible key spaces in Section 8.3. Following up, we give a detailed description of our identification scheme in Section 8.4. Size and time performances of the resulting signature scheme are presented in Section 8.6.

8.1 Translating ideals to isogenies

Let I be a left $\mathcal{O}_0$-ideal of smooth norm D where $\mathcal{O}_0$ is a special extremal maximal order, and let E_0 be a curve such that $\mathcal{O}_0$ is isomorphic to $\mathrm{End}(E_0)$. In this section we assume that we know an explicit representation of $\mathcal{O}_0$, meaning that we know an explicit isomorphism between $\mathrm{End}(E_0)$ and $\mathcal{O}_0$, allowing us to efficiently evaluate endomorphisms of E_0. We want to find the isogeny φ_I of degree D and domain E_0 corresponding to I. We will describe φ_I as the composition of several prime degree isogenies represented by their kernels. Most of the ideas presented in this section are adaptations of algorithms introduced in [19,17]; below we first recall these algorithms then describe our improvements.

Algorithm in [17] As each primary factor of D can be treated separately let us for simplicity assume that $D = \ell^e$. The idea is to divide φ_I into g isogenies of smaller degrees ℓ^f where the ℓ^f-torsion is defined over a reasonably small field extension. Following [17], to write $\varphi_I = \varphi_g \circ \dots \varphi_2 \circ \varphi_1$ under the ideal filtration $I = I_1 \cdot I_2 \cdots I_g$, we need an explicit representation of $\mathcal{O}_i = \mathcal{O}_R(I_i)$ in order to compute the action of $\mathrm{End}(E_i)$ on $E_i[\ell^f]$, where E_i is the codomain of φ_i. A formula is introduced in [17] providing such a representation from an ideal connecting $\mathcal{O}_i$ to $\mathcal{O}_0$ (equivalently an isogeny connecting E_i with E_0). However this formula involves division by the norm N_i of this ideal. In particular if e_i is the ℓ-adic valuation of N_i, we would need to compute the ℓ^{f+e_i}-torsion points. It thus appears that having N_i coprime to ℓ is essential for efficiency. We will therefore not be able to use $I_1 \cdots I_i$ as the connecting ideal, but we will instead use an equivalent ideal J_i of coprime degree. Fortunately, this can be found with KLPT. This idea underlies all the algorithms introduced in this section.

The discussion above motivates the introduction of a smooth integer T representing the torsion coprime with ℓ that is *accessible* (i.e., defined over small extensions of $\mathbb{F}_{p^2}$), we refer to Section 8.2 for concrete parameters illustrating what we mean by "accessible" and "small". Ideally, we would like to have J_i of norm dividing T (obtained by execution of the variant KLPT_T) so that the translations into the corresponding isogenies are efficient. However, once again we are hindered by the size of KLPT's outputs, which have norm around p^3. We now describe two tricks to reduce the torsion requirements.

Computing half of the isogeny from the image curve Let us assume that our ideal corresponds to $\psi : E_1 \to E_2$ where ψ has degree $D_1 D_2$ (with D_1 and D_2 not necessarily coprime). Instead of trying to express ψ from E_1 and using the $E_1[D_1 D_2]$ torsion, we can try and split ψ as $\hat{\psi}_2 \circ \psi_1$ where $\deg \psi_i = D_i$, $i = 1, 2$. We compute ψ_1 from $E_1[D_1]$ and ψ_2 from $E_2[D_2]$. We apply this idea in Algorithm 5 to translate an ideal of norm dividing T^2 (instead of T previously) to the corresponding isogeny. This means we now only need $T \sim p^{\frac{3}{2}}$ instead of $T \sim p^3$. We will see in Section 8.2 that this is indeed possible.

Meet-in-the-middle Let us now assume that $D = D_1 D_2 D'$, where D' is a reasonably small integer (in our application, D, D_1, D_2, D' are all ℓ-powers). We can write an isogeny ψ of degree D as $\hat{\psi}_2 \circ \theta \circ \psi_1$ where $\deg \psi_1 = D_1$, $\deg \theta = D'$ and $\deg \psi_2 = D_2$. The two isogenies $\psi_1, \hat{\psi}_2$ can be computed using $E_1[D_1]$ and $E_2[D_2]$ as before. Writing E_3 and E_4 for their codomains we know that there is $\theta : E_3 \to E_4$ of degree D'. If D' is small and smooth, a meet-in-the-middle search allows us to recover θ efficiently. This idea, combined with that of Section 8.1, underlies Algorithm 6 $\mathsf{IdealToIsogeny}_{\ell^{2f+\Delta}}$, that is illustrated in Fig. 2. In our implementation, this trick decreases the number of T-isogeny computations, which currently are the efficiency bottleneck.

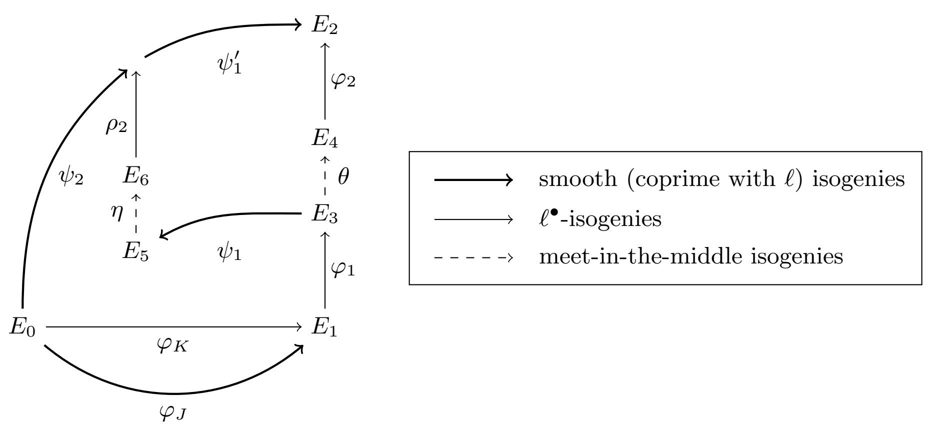

Fig. 2. Graphical representation of the ideal to isogeny translation of Algorithm 6

Ideal to isogeny: our optimized solution We are now ready to present the algorithm $\mathsf{IdealToIsogeny}_{\ell^\bullet}$ used in our implementation. The algorithm translates an $\mathcal{O}$-ideal in the corresponding isogeny for any maximal order $\mathcal{O}$. It requires K a left $\mathcal{O}_0$-ideal and right $\mathcal{O}$-ideal of degree $\ell^\bullet$ along with the corresponding isogeny $\varphi_K : E_0 \to E$ where $\mathcal{O} \cong \mathrm{End}(E)$. As before we write ℓ^f for the accessible

$\ell^\bullet$-torsion and T for the accessible smooth torsion coprime to ℓ. We write Δ for a meet-in-the-middle parameter $\ell^\Delta = D'$ (see Section 8.1). The algorithm uses the following subroutines.

- $\mathsf{SpecialIdealToIsogeny}(J, I, \varphi_I)$: described in Algorithm 5, it takes I, J two left $\mathcal{O}_0$-ideals of norm $n(I) = \ell^\bullet$ and $n(J)$ dividing T^2 along with the isogeny $\varphi_I : E_0 \to E$ and outputs φ_J.
- $\mathsf{IdealToIsogeny}_{\ell^{2f+\Delta}}(I, J, K, \varphi_J, \varphi_K)$: described in Algorithm 6, it takes I a left $\mathcal{O}_0$-ideal of norm dividing $T^2\ell^{2f+\Delta}$, J containing I of norm dividing T^2 and $K \sim J$ of norm $\ell^\bullet$ along with φ_J, φ_K and outputs φ of degree $\ell^{2f+\Delta}$ such that $\varphi_I = \varphi \circ \varphi_J$.

The algorithm $\mathsf{IdealToIsogeny}_{\ell^\bullet}(I, K, \varphi_K)$ is described in Algorithm 7. Note that we do not provide any proof of correctness and termination for Algorithms 5 to 7. This is because these algorithms already existed in essence in [17,19] and were only improved with the ideas of Section 8.1 and Section 8.1 for efficiency.

Algorithm 5 $\mathsf{SpecialIdealToIsogeny}(J, I, \varphi_I)$

Require: Two equivalent left ideals I, J of $\mathcal{O}_0$, with J of norm dividing T^2 and I of norm $\ell^\bullet$, and the corresponding isogeny $\varphi_I : E_0 \to E$.
Ensure: φ_J.
1: $H_1 \leftarrow J + T\mathcal{O}_0$.
2: Let $\alpha \in I$ such that $J = \chi_I(\alpha)$.
3: $H_2 \leftarrow \langle \alpha, (n(J)/n(H_1)) \rangle$.
4: $\varphi_{H_i} \leftarrow \mathsf{IdealToIsogeny}_T(H_i) : E_0 \to E_i$.
5: Let $\psi : E \to E/\varphi_I(\ker \varphi_{H_2}) = E_1$.
6: **return** $\hat{\psi} \circ \varphi_{H_1}$.

8.2 Choosing the parameters

We discuss now the choice of the parameters and most importantly the prime p that we will use. As mentioned above, we need a prime p such that the $T\ell^f$-torsion is accessible for $T \simeq p^{3/2}$ and f is as big as possible. Recall that by "accessible" we generally mean that the full $T\ell^f$-torsion subgroup is defined over a small extension of $\mathbb{F}_{p^2}$. We can strengthen this by asking that $T\ell^f \mid (p^2 - 1)$, which implies that the full $T\ell^f$-torsion is generated by four points with x-coordinates in $\mathbb{F}_{p^2}$, or equivalently by two $\mathbb{F}_{p^2}$-rational points on the curve with Frobenius trace $-2p$ and two other $\mathbb{F}_{p^2}$-rational points on its twist. Similar primes were recently considered for use in B-SIDH [7], an adaptation of SIDH with smaller (uncompressed) public keys.

Algorithm 6 $\mathsf{IdealToIsogeny}_{\ell^{2f+\Delta}}(I, J, K, \varphi_J, \varphi_K)$

Require: I a left $\mathcal{O}_0$-ideal of norm dividing $T^2\ell^{2f+\Delta}$, an $\mathcal{O}_0$-ideal in J containing I of norm dividing T^2, and an ideal $K \sim J$ of norm a power of ℓ, as well as φ_J and φ_K.

Ensure: $\varphi = \varphi_2 \circ \theta \circ \varphi_1 : E_1 \to E_2$ of degree $\ell^{2f+\Delta}$ such that $\varphi_I = \varphi \circ \varphi_J$, $L \sim I$ of norm dividing T^2 and φ_L.

0: Write $\varphi_J, \varphi_K : E_0 \to E_1$.
1: Let $I_1 = I + \ell^f \mathcal{O}_0$.
2: Let $\varphi_1' = \mathsf{IdealToIsogeny}_{\ell^f}(I_1)$.
3: Let $\varphi_1 = [\varphi_J]_* \varphi_1' : E_1 \to E_3$.
4: Let $L = \mathsf{KLPT}_T(I)$.
5: Let $\alpha \in K$ such that $J = \chi_K(\alpha)$.
6: Let $\beta \in I$ such that $L = \chi_I(\beta)$.
7: Let $\gamma = \beta\alpha/n(J)$. We have $\gamma \in K$, $\bar{\gamma} \in L$, and $n(\gamma) = T^2\ell^{2f+\Delta}n(K)$.
8: Let $H_1 = \langle \gamma, n(K)\ell^f T\rangle$. We have $\varphi_{H_1} = \psi_1 \circ \varphi_1 \circ \varphi_K : E_0 \to E_5$, where ψ_1 has degree T.
9: Let $H_2 = \langle \overline{\gamma}, \ell^f T\rangle$. We have $\varphi_{H_2} = \rho_2 \circ \psi_2 : E_0 \to E_6$, where ψ_2 has degree T and φ_2 has degree ℓ^f.
10: Find $\eta : E_5 \to E_6$ of degree ℓ^Δ with meet-in-the-middle.
11: Let $\varphi_2 \circ \theta = [\hat{\psi}_1]_* \hat{\rho}_2 \circ \eta : E_3 \to E_2$ and $\psi_1' = [\hat{\varphi}_2 \circ \eta]_* \hat{\psi}_1$
12: **return** $\varphi = \varphi_2\theta \circ \varphi_1$, L and $\psi_1' \circ \psi_2$.

Algorithm 7 $\mathsf{IdealToIsogeny}_{\ell^\bullet}(I, K, \varphi_K)$

Require: A left $\mathcal{O}$-ideal I of norm a power of ℓ, K a left $\mathcal{O}_0$-ideal and right $\mathcal{O}$-ideal of norm $\ell^\bullet$, the corresponding φ_K.

Ensure: φ_I.

1: Write $I = I_n \subset \cdots \subset I_1 \subset I_0 = \mathcal{O}$ where $n(I_i)/n(I_{i-1}) \leq \ell^{2f+\Delta}$.
2: $J \leftarrow \mathsf{KLPT}_T(K)$.
3: $\varphi_J \leftarrow \mathsf{SpecialIdealToIsogeny}(J, K, \varphi_K)$.
4: **for** $i = 1, \ldots, n$ **do**
5: $\quad \varphi_i, J, \varphi_J \leftarrow \mathsf{IdealToIsogeny}_{\ell^{2f+\Delta}}(J \cdot I_i, J, K, \varphi_J, \varphi_K)$.
6: $\quad K \leftarrow K \cdot I_i$.
7: $\quad \varphi_K \leftarrow \varphi_i \circ \varphi_K$.
8: **end for**
9: **return** $\varphi_n \circ \cdots \circ \varphi_1$.

For λ bits of classical security, we need a prime of 2λ bits. In the implementation described in Section 8.6, we used the 256-bits prime p such that

$$\begin{aligned} p+1 &= 2^{33} \cdot 5^{21} \cdot 7^{2} \cdot 11 \cdot 31 \cdot 83 \cdot 107 \cdot 137 \cdot 751 \cdot 827 \cdot 3691 \cdot 4019 \cdot 6983 \\ &\quad \cdot 517434778561 \cdot 26602537156291\,, \\ p-1 &= 2 \cdot 3^{53} \cdot 43 \cdot 103 \cdot 109 \cdot 199 \cdot 227 \cdot 419 \cdot 491 \cdot 569 \cdot 631 \cdot 677 \cdot 857 \cdot 859 \\ &\quad \cdot 883 \cdot 1019 \cdot 2713 \cdot 4283\,. \end{aligned}$$

This prime verifies that p^2-1 is a multiple of $2^{33}T$ where T is a 395-bit 2^{13}-smooth number. We give more details on the search for such primes in [13].

Algorithm 7 requires numerous evaluations of T-isogenies, and this will prove to be the bottleneck of our scheme. The recent work of [2] provided a square root speedup to compute and evaluate an isogeny of degree d. Their method appears to be faster than the naive method for $d \geq 100$ approximately and our scheme's implementation also benefits from this improvement.

8.3 Defining the key space

For statistical security, the secret isogeny should be of degree sufficiently large, so to ensure a nearly uniform distribution of the public key E_A in the set of supersingular curves. However, a larger degree results in a bigger output for Algorithm 3, hence poorer performance. In this section we discuss an alternative key sampling method which trades off statistical security for efficiency. The key idea is to sample the degree of the secret isogeny as a secret big prime (instead of a public smooth number). Choosing the degree not smooth thwarts meet-in-the-middle attacks, while keeping it secret enlarges the search space. Together, these two facts allow us to pick a degree N_τ of size $\log(N_\tau) = \lambda/2$ for λ bits of security. The key sampling method is described in Section 8.4. A more detailed security analysis can be found in the longer version [13].

This improvement produces a shorter and more efficient signature for the same level of security, as it reduces the output size of Algorithm 3 from $\frac{9}{2}\log_\ell(p)$ to $\frac{15}{4}\log_\ell(p)$. We use it for the implementations presented in Section 8.6.

8.4 The concrete protocol

Now that we have all the preliminary algorithms, we can provide a concrete description of our identification scheme. Let us assume that we have found a prime p as described above in Section 8.2. We recall that $T \approx p^{3/2}$ is the smooth torsion defined over $\mathbb{F}_{p^2}$ for supersingular elliptic curves. For the challenge and the commitment we divide T as $D_c \cdot T'$ where D_c is a λ-bit integer and T' a 2λ-bit integer. In the protocol presented below we decided to use $D = \ell^\bullet$.

Building τ (keygen) We use the efficiency improvement from Section 8.3 hence fix $B_\tau = \frac{1}{2}\lambda$. The degree N_τ is a prime number inert in R and smaller than B_τ, chosen uniformly at random among such numbers.

Since N_τ is a large prime number, we never compute concretely the isogeny τ as this would be too inefficient. Instead we use the corresponding ideal I_τ. This is enough to apply $\mathsf{SigningKLPT}$ but it does not give us the public key E_A. For this, we compute another isogeny $\tau' : E_0 \to E_A$ of degree $\ell^\bullet$. This can be done with KLPT. We briefly summarize the description above for keygen:

1. Select a prime $N_\tau \leq B_\tau$ that is inert in R uniformly at random.
2. Select a left $\mathcal{O}_0$-ideal I_τ of norm N_τ, uniformly at random among the $N_\tau + 1$ possibilities.
3. Compute $J_\tau = \mathsf{KLPT}_{\ell^\bullet}(I_\tau)$
4. Compute $\tau' = \mathsf{IdealToIsogeny}_{\ell^\bullet}(J_\tau, \mathcal{O}_0, [1]_{E_0})$ and set $\mathsf{pk} = E_A$ the codomain of τ'.

Building ψ (commitment) There are several options for building the commitment (and incidentally the challenge); we present the most efficient option here. We note that for security reasons, ψ must be as hard to recover as the secret. This suggests taking a smooth isogeny of degree about p (here we do not gain anything by using the same idea as in Section 8.3). Given the factorization $T = D_c \cdot T'$, we choose ψ as a random isogeny of degree T' from E_0. With this choice, computing the isogeny and converting it to an ideal is efficient. Let $I_\psi := \mathsf{IsogenyToIdeal}_{T'}(\psi)$.

Building φ (challenge) The previous choice of commitment generation was motivated by the fact that we want an efficient way to translate the challenge into its corresponding ideal. For λ-bit soundness security we need a challenge space of size $2^\lambda = O(\sqrt{p})$, so the challenge isogeny needs to be of degree $O(\sqrt{p})$. Let $\varphi : E_1 \to E_2$ be a random cyclic isogeny of degree D_c. Since the $T = T'D_c$-torsion is accessible, computing the corresponding ideal will be efficient for the prover.

Building σ (response) The response is computed as follows:

1. Compute $I_\varphi = [I_\psi]_* \big(\mathsf{IsogenyToIdeal}_{D_c}([\psi]^*\varphi)\big)$.
2. Set $I = \overline{I_\tau} \cdot I_\psi \cdot I_\varphi$ and compute $J = \mathsf{SigningKLPT}(I, I_\tau)$.
3. Compute $\sigma = \mathsf{IdealToIsogeny}_{\ell^\bullet}(J, J_\tau, \tau')$.

8.5 Response and verification

In this section we discuss the verification part of the protocol. We remind the reader that upon receiving σ, the verifier needs to check that it is an isogeny of degree D between E_A and E_2 such that the composition with the challenge φ is cyclic (this last part is trivial when D and D_c are coprime). All this can be done by computing the chain of isogenies associated with σ. We decompose σ of degree $D = \ell^e$ as $\sigma_g \circ \cdots \circ \sigma_1$ where each of the σ_j has degree at most ℓ^f ($f = 33$ in our case). The main problem is to find a compact and efficient representation of σ that can be sent to the verifier. A wide array of solutions already exist in the literature for SIDH/SIKE [34,23,1,8,24] most of which can be applied to our setting. In the longer version [13], we describe two compress, decompress algorithms well-suited to our application.

8.6 The concrete instantiation

We discuss below the performance features of our implementation.

Signature size and comparison with existing schemes For λ bit of classical security, we take a prime $p \approx 2^{2\lambda}$. The public key is the j-invariant of the curve E_A and it is of size $2\log_2(p) = 4\lambda$. The secret can be seen as a pair N_τ, I_τ. The integer N_τ is a $\log(p)/4$-bit prime, and we can represent I_τ as a number in $[1, N_\tau + 1]$, so another $\log(p)/4$-bit integer. In total the secret key has size λ. The signature is made of E_1 and σ, where σ is compressed as described in Section 8.5. As argued there, we can either use a full compression of exactly e bits, or allow for a few additional bits to accelerate the verification time. With the second method the size is $e + 4(\lceil e/f \rceil - 1)$. We recall that, using for keys as in 8.3, $e = 15/4\log(p) + O(\log(\lambda))$. Representing the commitment curve E_1 requires $2\log_2(p) = 4\lambda$ additional bits. We summarize these values in Table 2 when $\lambda = 128$, for our concrete instantiation we have $\log_2(p) = 256$, $f = 33$ and $e = 1000$.

Secret Key (bytes)	Public Key (bytes)	Signature (bytes)
16	64	204

Table 2. Size of SQISign keys and signature for the NIST-1 level of security.

These sizes make SQISign the most compact post-quantum digital signature targeting NIST-1 level of security, in terms of combined public key and signature size. With respect to round 2 candidates, it is more than 5 times more compact than Falcon [18] in terms of combined size, and only trails GeMSS [4] in terms of signature size. Signatures are more compact than RSA, and about three times larger than ECDSA, for a comparable level of classical security.

Performance We implemented SQISign in C, on top of the `libpari` library of PARI/GP 2.11.4 [30], and a port of the isogeny evaluation code published in [2]. Our code is available at https://github.com/SQISign/sqisign. We ran experiments on a 3.40GHz Intel Core i7-6700 (Skylake) CPU with Turbo Boost disabled. The code was compiled using `clang-6.0 -O3 -Os -march=native -mtune=native -Wall -Wextra -std=gnu99 -pedantic`.

The results are summarized in Table 3. We empirically chose the parameter $\Delta = 14$. For key generation we generated 100 random keys. For signature we generated 10 random keys and signed 10 random messages under each key. For verification we generated 5 random keys, we signed 5 random messages under each key, and we ran verification 10 times. We stress that we did not attempt at producing a constant-time implementation, which appears to be an intensive task owing to the complexity of the algorithms involved.

		Keygen	Sign	Verify
Mcycles	1st quartile	1,922	7,687	140
	median	1,959	7,767	142
	3rd quartile	2,000	7,909	148
ms	1st quartile	564	2,256	41
	median	575	2,279	42
	3rd quartile	587	2,321	43

Table 3. Performance of SQISign in millions of cycles and in milliseconds. Statistics over 100 runs for key generation and signature, and over 250 runs for verification.

9 Conclusion

We introduced a new signature scheme along with a concrete instantiation and implementation. Our implementation proves that our signature is quite efficient compared to other isogeny-based candidates. The associated identification scheme is sound under classical isogeny assumptions, while its zero-knowledge relies on hardness of a new *ad hoc* problem. We briefly justified that this new problem bears some resemblance with existing hard problems, lending some credibility to its conjectured hardness.

More work on understanding the output distribution of our generalized KLPT algorithm is needed to gain confidence in the security of SQISign. It would be interesting, for example, to reduce the zero-knowledge property to more classical assumptions. Such a result would probably come at a cost in terms of efficiency as this would mean using a different generalization of KLPT. Indeed, from our analysis in Section 7 it appears unlikely to prove security under classical assumptions with the current algorithm.

The second direction for improvement is efficiency. The scheme is complex and there is a lot of potential for optimizations. A search for better parameters could allow one to obtain a more efficient signature, and algorithmic progress in any aspect of isogeny computations and evaluations would probably impact the performance. The main bottleneck remains the translation from ideals to isogenies, new techniques for which could greatly benefit our protocol. For instance, finding a more direct algorithm that does not rely as heavily on rational torsion points could yield a more efficient translation. Finally, any improvement to KLPT producing ideals of smaller norm in reasonable time would improve every single step of the translation, thus greatly reducing the signature time.

References

1. Azarderakhsh, R., Jao, D., Kalach, K., Koziel, B., Leonardi, C.: Key compression for isogeny-based cryptosystems. In: Proceedings of the 3rd ACM International Workshop on ASIA Public-Key Cryptography. pp. 1–10. ACM (2016)
2. Bernstein, D.J., De Feo, L., Leroux, A., Smith, B.: Faster computation of isogenies of large prime degree. ANTS XIV (2020)
3. Beullens, W., Kleinjung, T., Vercauteren, F.: Csi-fish: Efficient isogeny based signatures through class group computations. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 227–247. Springer (2019)
4. Casanova, A., Faugere, J.C., Macario-Rat, G., Patarin, J., Perret, L., Ryckeghem, J.: GeMSS: a great multivariate short signature. NIST Post-Quantum Cryptography Standardization (2019), https://www-polsys.lip6.fr/Links/NIST/GeMSS.html
5. Castryck, W., Lange, T., Martindale, C., Panny, L., Renes, J.: CSIDH: an efficient post-quantum commutative group action. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 395–427. Springer (2018)
6. Charles, D.X., Lauter, K.E., Goren, E.Z.: Cryptographic hash functions from expander graphs. Journal of Cryptology **22**(1), 93–113 (Jan 2009). https://doi.org/10.1007/s00145-007-9002-x
7. Costello, C.: B-SIDH: supersingular isogeny Diffie-Hellman using twisted torsion (2019), https://ia.cr/2019/1145
8. Costello, C., Jao, D., Longa, P., Naehrig, M., Renes, J., Urbanik, D.: Efficient Compression of SIDH Public Keys, pp. 679–706. Springer International Publishing (2017). https://doi.org/10.1007/978-3-319-56620-7_24
9. Couveignes, J.M.: Hard homogeneous spaces. IACR Cryptology ePrint Archive **2006**, 291 (2006)
10. Cox, D.: Primes of the Form $x^2 + ny^2$, chap. 1, From Fermat to Gauss, pp. 7–85. John Wiley & Sons, Ltd (2013). https://doi.org/10.1002/9781118400722.ch1
11. Damgård: On Σ protocols. http://www.cs.au.dk/%7eivan/Sigma.pdf (2010)
12. De Feo, L., Galbraith, S.D.: Seasign: Compact isogeny signatures from class group actions. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 759–789. Springer (2019)
13. De Feo, L., Kohel, D., Leroux, A., Petit, C., Wesolowski, B.: SQISign: compact post-quantum signatures from quaternions and isogenies. Cryptology ePrint Archive, Report 2020/??? (2020), https://eprint.iacr.org/2020/???
14. Decru, T., Panny, L., Vercauteren, F.: Faster seasign signatures through improved rejection sampling. In: International Conference on Post-Quantum Cryptography. pp. 271–285. Springer (2019)
15. Deuring, M.: Die Typen der Multiplikatorenringe elliptischer Funktionenkörper. Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg **14**(1), 197–272 (Dec 1941)
16. Eichler, M.: Über die Idealklassenzahl total definiter Quaternionenalgebren. Mathematische Zeitschrift **43**(1), 102–109 (1938)
17. Eisenträger, K., Hallgren, S., Lauter, K., Morrison, T., Petit, C.: Supersingular isogeny graphs and endomorphism rings: Reductions and solutions. In: Nielsen, J.B., Rijmen, V. (eds.) Advances in Cryptology – EUROCRYPT 2018. pp. 329–368. Springer International Publishing, Cham (2018)

18. Fouque, P.A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Prest, T., Ricosset, T., Seiler, G., Whyte, W., Zhang, Z.: Falcon: Fast-Fourier lattice-based compact signatures over NTRU. NIST Post-Quantum Cryptography Standardization (2019), https://falcon-sign.info/
19. Galbraith, S.D., Petit, C., Silva, J.: Identification protocols and signature schemes based on supersingular isogeny problems. In: ASIACRYPT (2017)
20. Jao, D., De Feo, L.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In: Yang, B. (ed.) International Workshop on Post-Quantum Cryptography – PQCrypto 2011. pp. 19–34 (2011)
21. Kohel, D.: Endomorphism rings of elliptic curves over finite fields. Ph.D. thesis, University of California, Berkeley (1996)
22. Kohel, D., Lauter, K.E., Petit, C., Tignol, J.P.: On the quaternion ℓ-isogeny path problem. IACR Cryptology ePrint Archive **2014**, 505 (2014)
23. Naehrig, M., Renes, J.: Dual isogenies and their application to public-key compression for isogeny-based cryptography. In: Galbraith, S.D., Moriai, S. (eds.) Advances in Cryptology – ASIACRYPT 2019. pp. 243–272. Springer International Publishing, Cham (2019)
24. Pereira, G.C.C.F., Doliskani, J., Jao, D.: x-only point addition formula and faster torsion basis generation in compressed sike. Cryptology ePrint Archive, Report 2020/431 (2020), https://eprint.iacr.org/2020/431
25. Petit, C., Smith, S.: An improvement to the quaternion analogue of the l-isogeny path problem (2018), conference talk at MathCrypt
26. Pizer, A.: An algorithm for computing modular forms on $\gamma_0(n)$. Journal of Algebra - J ALGEBRA **64**, 340–390 (06 1980). https://doi.org/10.1016/0021-8693(80)90151-9
27. Pizer, A.K.: Ramanujan graphs and Hecke operators. Bulletin of the American Mathematical Society **23**(1), 127–137 (1990)
28. de Saint Guilhem, C.D., Kutas, P., Petit, C., Silva, J.: Séta: Supersingular encryption from torsion attacks. Tech. rep., Cryptology ePrint Archive, Report 2019/1291, 2019. https://eprint. iacr. org . . . (2019)
29. Silverman, J.H.: The Arithmetic of Elliptic Curves, Gradute Texts in Mathematics, vol. 106. Springer-Verlag (1986)
30. The PARI Group, Université de Bordeaux: PARI/GP version 2.11.4 (2020), available from http://pari.math.u-bordeaux.fr/
31. Voight, J.: Quaternion Algebras. Springer Graduate Texts in Mathematics series (2018)
32. Waterhouse, W.C.: Abelian varieties over finite fields. Annales scientifiques de l'École Normale Supérieure **2**(4), 521–560 (1969)
33. Yoo, Y., Azarderakhsh, R., Jalali, A., Jao, D., Soukharev, V.: A post-quantum digital signature scheme based on supersingular isogenies. In: International Conference on Financial Cryptography and Data Security. pp. 163–181. Springer (2017)
34. Zanon, G.H.M., Simplicio, M.A., Pereira, G.C.C.F., Doliskani, J., Barreto, P.S.L.M.: Faster isogeny-based compressed key agreement. In: Lange, T., Steinwandt, R. (eds.) Post-Quantum Cryptography. pp. 248–268. Springer International Publishing (2018)