SQIsignHD: New Dimensions in Cryptography

Pierrick Dartois[1,2][0009-0008-2808-9867], Antonin Leroux[3,4][0009-0002-3737-0075], Damien Robert[1,2][0000-0003-4378-4274] and Benjamin Wesolowski[5][0000-0003-1249-6077]

[1] Univ. Bordeaux, CNRS, INRIA, IMB, UMR 5251, F-33400 Talence, France
[2] INRIA, IMB, UMR 5251, F-33400, Talence, France
`{pierrick.dartois,damien.robert}@inria.fr`
[3] DGA-MI, Bruz, France,
[4] IRMAR - UMR 6625, Université de Rennes, France
`antonin.leroux@polytechnique.org`
[5] ENS de Lyon, CNRS, UMPA, UMR 5669, Lyon, France
`benjamin.wesolowski@ens-lyon.fr`

Abstract. We introduce SQIsignHD, a new post-quantum digital signature scheme inspired by SQIsign. SQIsignHD exploits the recent algorithmic breakthrough underlying the attack on SIDH, which allows to efficiently represent isogenies of arbitrary degrees as components of a higher dimensional isogeny. SQIsignHD overcomes the main drawbacks of SQIsign. First, it scales well to high security levels, since the public parameters for SQIsignHD are easy to generate: the characteristic of the underlying field needs only be of the form $2^f 3^{f'} - 1$. Second, the signing procedure is simpler and more efficient. Our signing procedure implemented in C runs in 28 ms, which is a significant improvement compared to SQISign. Third, the scheme is easier to analyse, allowing for a much more compelling security reduction. Finally, the signature sizes are even more compact than (the already record-breaking) SQIsign, with compressed signatures as small as 109 bytes for the post-quantum NIST-1 level of security. These advantages may come at the expense of the verification, which now requires the computation of an isogeny in dimension 4, a task whose optimised cost is still uncertain, as it has been the focus of very little attention. Our experimental `sagemath` implementation of the verification runs in around 600 ms, indicating the potential cryptographic interest of dimension 4 isogenies after optimisations and low level implementation.

Acknowledgements. We thank Luca De Feo for his advice all along this project and for suggesting the title of this paper. This project was supported by ANR grant CIAO (ANR-19-CE48-0008), PEPR PQ-TLS (the France 2030 program under grant agreement ANR-22-PETQ-0008 PQ-TLS) and the European Research Council under grant No. 101116169 (AGATHA CRYPTY).

1 Introduction

Isogeny-based cryptography has been a promising area of research in post-quantum cryptography since Couveignes, Rostovtsev and Stolbunov introduced the first key exchange using ordinary isogenies [Cou06; RS06]. Schemes from this family often distinguish themselves by their compactness, in particular with respect to key sizes. It is notably the case of the digital signature scheme SQIsign [DKLPW20; FLLW23], the most compact post-quantum signature scheme by a decent margin. However, efficiency has been a recurring challenge for isogeny-based schemes, and indeed, SQIsign is much slower than other post-quantum signatures.

In this paper, we introduce SQIsignHD, a new digital signature scheme derived from SQIsign. As in [GPS20], SQIsign uses the Deuring correspondence between supersingular elliptic curves and quaternion orders. This Deuring correspondence is a powerful tool to construct cryptosystems because it is one way: it is easy to turn an order into the corresponding elliptic curve, but the converse direction is the presumably hard *supersingular endomorphism ring problem* [EHLMP18; Wes22]. In SQIsign, the signer's public key is a supersingular elliptic curve, and a signature effectively proves that the signer knows the associated quaternion order. This requires algorithms to translate between orders (and ideals in these orders) and elliptic curves (and isogenies from these curves). This translation is costly, and crucially requires the ideals (or isogenies) to have smooth norms (or degrees). The original methods have been improved upon [FLLW23], but that remains the bottleneck of SQIsign. Another issue with SQIsign is its scalability to higher security levels. Indeed, to set public parameters, one needs to find a prime p such that p^2-1 has a very large smooth factor. Searching for such primes p becomes harder as the security level grows, and is still an active area of research [CMN21; BSC+22; Ahr23]. Besides, the security of SQIsign relies on the fact that signatures are computationally indistinguishable from random isogenies of fixed powersmooth degrees. There is no known formal proof of this *ad hoc* heuristic assumption.

The new scheme SQIsignHD follows a similar outline as SQIsign, but resolves its main drawbacks by fundamentally reforging the computational approach. The main ingredient is the ground-breaking technique that has recently led to the downfall of SIDH [CD23; MMPPW23; Rob23b]. Namely, these attacks use a lemma due to Kani [Kan97] combined with Zahrin's trick, which allows one to "embed" any isogeny into an isogeny of higher dimension. As remarked in [Rob22], this technique allows one to describe an isogeny by listing only the image of a few well-chosen points; from this description, one can efficiently evaluate the isogeny on any other point, regardless of the factorisation pattern of the underlying isogeny. This newly gained freedom on usable isogenies unlocks challenges in efficiency, security, and scalability.

Our contribution. We introduce the digital signature scheme SQIsignHD. It leverages recent algorithmic breakthroughs [CD23; MMPPW23; Rob23b] to overcome the main drawbacks of SQIsign. It has the following advantages:

- SQIsignHD scales well to high security levels. Indeed, while SQIsign requires a search for primes p with strong constraints, the primes used in SQIsignHD may be of the form $c2^f3^{f'} - 1$, where c is some (preferably small) cofactor. Such primes, already used in SIDH [JD11], are easy to find, and allow for efficient field arithmetic.
- The signing procedure of SQIsignHD is simpler and more efficient than SQIsign. Let us stress that no high dimensional isogeny needs to be computed when signing. Our proof-of-concept implementation, which still lacks many standard optimisations, is already about ten times faster than the fastest SQIsign implementation. This is discussed in further detail in Section 6.2.
- SQIsignHD is easier to analyse, allowing for a much more compelling security reduction to the supersingular endomorphism problem. Unlike in SQIsign, our proof of the zero-knowledge property in SQIsignHD relies on simple and plausible heuristic assumptions. In fact, we propose two variants of SQIsign, one of which is less efficient but benefits from a heuristic-free analysis. In both cases, the zero-knowledge property is based on a simulator which is given access to a non-standard oracle. We carefully discuss the impact of this oracle on the supersingular endomorphism problem.
- SQIsignHD signatures are even more compact than SQIsign, as they are only 6.5λ bits long, for λ bits of security. In particular, they are as small as 109 bytes for the NIST-1 security level. SQIsign already had the most compact signature and public keys combined of all post-quantum signature schemes, and SQIsignHD breaks this record.

These advantages may come at the expense of the verification, which now requires the computation of a chain of 2-isogenies in dimension 4 (or 8 in the less efficient variant). We provide an algorithm for the verification, and an experimental implementation in `sagemath` [The23a; 22]. An optimised low-level implementation is left for future work, hence the true cost of verification is still uncertain. The verification in SQIsign also requires the computation of a (longer!) chain of 2-isogenies, but only in dimension 1.

1.1 A modular overview of SQIsignHD

We introduce two distinct versions of SQIsignHD, optimised in different directions. FastSQIsignHD is optimised for speed, while RigorousSQIsignHD is optimised for the security proof. Note that the security proof applies to both: the difference lies in the proof being unconditional for RigorousSQIsignHD when given access to an oracle, but requiring additional heuristics for FastSQIsignHD (see Appendix D.2 and Section 5.2). Under the hood, FastSQIsignHD relies on isogenies of dimension 4, while RigorousSQIsignHD relies on isogenies of dimension 8. The reader may sense the parallel with the heuristic (dimension 4) and rigorous (dimension 8) variants of the algorithms of [Rob23b].

We present here the main algorithmic building blocks of the identification scheme undelying SQIsignHD to give a modular overview of the protocol. Those algorithms will be presented in detail in the course of the paper for FastSQIsignHD and in Appendix B for RigorousSQIsignHD. Unsurprisingly, the protocol shares a lot of similarities with SQIsign. The full signature scheme can be derived from there with the Fiat-Shamir transform [FS87] as in [DKLPW20, § 3.4] (see Appendix A.1 for details).

Public set-up. We choose a prime p and a supersingular elliptic curve $E_0/\mathbb{F}_{p^2}$ of known endomorphism ring $\mathcal{O}_0 \cong \mathrm{End}(E_0)$ such that E_0 has smooth torsion defined over a small extension of $\mathbb{F}_{p^2}$ (of degree 1 or 2). In practice, one may use the curve $E_0 : y^2 = x^3 + x$ (and $p \equiv 3 \mod 4$).

Key generation. The prover generates a random secret isogeny $\tau : E_0 \longrightarrow E_A$ of fixed smooth degree D_τ. Then, the prover publishes E_A. Knowing τ, only the prover can compute the endomorphism ring $\mathrm{End}(E_A)$. In the fast method FastKeyGen, the isogeny τ has degree $D_\tau = \Theta(p)$, which is heuristically sufficient to ensure that the distribution of E_A is computationally indistinguishable from uniform. In the alternate method RigorousKeyGen, the degree is chosen a bit larger to make the distribution of E_A statistically close to uniform.

Commitment. The prover generates a random isogeny $\psi : E_0 \longrightarrow E_1$ of smooth degree D_ψ and returns E_1 to the verifier (ψ being secret). The resulting distribution for E_1 is as close as possible to the uniform distribution in the supersingular isogeny graph. As in the key generation, we propose a fast procedure FastCommit(E_0) in Section 3.3 resulting in a distribution heuristically indistinguishable from uniform, and a slower variant RigorousCommit(E_0) in Appendix B.2 which guarantees statistical closeness to uniform.

Challenge. The verifier generates a random isogeny $\varphi : E_A \longrightarrow E_2$ of smooth degree D_φ sufficiently large for φ to have high entropy. Then, φ is sent to the prover. The Challenge procedure is described in Section 3.2. Unlike SQIsign, we chose to start the challenge from E_A instead of E_1 in order to optimize the response process.

Response. The prover generates an *efficient representation* of an isogeny $\sigma : E_1 \longrightarrow E_2$ of small degree $q \simeq \sqrt{p}$ in the sense of the following definition and returns it to the verifier.

Definition 1. Let $\mathscr{A}$ be an algorithm and $\varphi : E \longrightarrow E'$ be an isogeny defined over a finite field $\mathbb{F}_q$. An *efficient representation* of φ (with respect to $\mathscr{A}$) is some data $D \in \{0,1\}^*$ such that:

(i) D has polynomial size in $\log(\deg(\varphi))$ and $\log(q)$.
(ii) On input D and $P \in E(\mathbb{F}_{q^k})$, $\mathscr{A}$ returns $\varphi(P)$ in polynomial time in $k\log(q)$ and $\log(\deg(\varphi))$.

There always exists an efficient representation of a smooth degree isogeny. For instance, it can be written as a chain of small degree isogenies. Until the recent attacks on SIDH [CD23; MMPPW23; Rob23b], we did not know how to efficiently represent isogenies with non-smooth degrees without revealing the endomorphism ring of the domain. For that reason, the original version of SQIsign uses smooth degree isogenies for the signature. These smooth degree isogenies are found with a variant of the KLPT algorithm [KLPT14] and have very big degree $\simeq p^{15/4}$. This not only hurts efficiency, but also security: the isogeny σ is so carefully crafted that it is hard to simulate, and as a result, the zero-knowledge property of SQIsign is very *ad hoc*.

Now, the methods from [CD23; MMPPW23; Rob23b] give much more freedom on the isogenies that can be efficiently represented. This allows SQIsignHD to improve both efficiency (using isogenies σ of degree as low as $\simeq \sqrt{p}$), and security (the isogenies σ are now nicely distributed, hence simulatable).

The idea is to "embed" σ into an isogeny of higher dimension — and that only requires knowing the image of a few points through σ. As in the attacks against SIDH, such an isogeny can have dimension 2, 4 or 8. We shall see that dimension 2 has little interest compared to the original SQIsign protocol from an efficiency and security point of view. In SQIsignHD, we propose a response procedure FastRespond to represent σ in dimension 4, and an alternative procedure RigorousRespond based on an isogeny computation in dimension 8. The procedure FastRespond is fast, and its security analysis relies on reasonable heuristics. On the other hand, RigorousRespond is much slower (though still polynomial time), but allows for a rigorous analysis.

In either case, for efficiency reasons, the prover does not actually compute higher dimensional isogenies but only images of some points through σ (we explain how these points are evaluated in the course of the paper). Those points provide an efficient representation of σ (along with $\deg(\sigma)$) and this data is sent to the verifier who can then compute higher dimensional isogenies representing σ.

Verification. The verifier checks that the response returned by the prover (points of E_2) correctly represents an isogeny $\sigma : E_1 \longrightarrow E_2$. We propose two procedures FastVerify and RigorousVerify computing isogenies embedding σ in dimension 4 or 8. So far, isogeny computations in dimension 4 has been the subject of very little literature.

Nonetheless, our proof of concept implementation of dimension 4 isogenies in `sagemath` [The23a; 22] demonstrates the cryptographic feasibility of this phase. We expect an optimized implementation to be at worse twice as slow as the original SQIsign verification, and hopefully even closer than that. We refer to Appendix F for an estimate of the number of operations required for the verification.

Content. The rest of this paper is organized as follows. In Section 2, we present the core idea of our paper: how to embed signature/response isogenies in higher dimension with Kani's lemma. Section 3 introduces algorithms for key generation, commitment and challenge whereas Section 4 presents the response and verification phase for FastSQIsignHD. A security analysis of FastSQIsignHD identification protocol is conducted in Section 5. Finally, we discuss the expected

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

— Public
— Secret

Fig. 1. The SQIsign/SQIsignHD identification protocol.

performance of the digital signature scheme derived from FastSQIsignHD in Section 6.

Due to the lack of space, some preliminaries are given in Appendix A. A detailed presentation of RigorousSQIsignHD is given in Appendix B and Appendix C. We give a complete security analysis of this version of the protocol in Appendix D. Some proofs of our results are deferred to Appendix E. Finally Appendix F details the verification algorithm when using the theta model to compute isogenies in dimension 4 and 8, and in particular gives an algorithm to compute a 2^e-isogeny and the corresponding number of arithmetic operations.

2 Representing the response isogeny efficiently in higher dimension

In this section, we explore our main idea to improve SQIsign by embedding the signature isogeny inside an isogeny in higher dimension. We start by recalling how the signature is represented in the original SQIsign protocol in Section 2.1 and why this representation is slow to compute. Then, we introduce Kani's lemma and explain how to embed isogenies in higher dimension in Section 2.2. Finally, we apply this idea to provide another representation of the signature isogeny in SQIsign in Section 2.3.

2.1 State of the art isogeny representation: a slow signature process

With state of the art techniques prior to the attacks against SIDH, we could only efficiently represent isogenies of smooth degrees. That is why in the original versions of SQIsign [DKLPW20; FLLW23], the signature isogeny σ has degree a prime power ℓ^e and is represented as a chain of ℓ-isogenies.

To compute such a signature σ, the prover computes the ideal J associated to the isogeny path given by the secret key, commitment and challenge. They then apply a SigningKLPT algorithm to J, to return a random equivalent ideal $I \sim J$ of norm ℓ^e. Then, the prover converts I into an isogeny. This last computation is very costly because $\mathrm{nrd}(I) = \ell^e$ is close to $p^{15/4}$, while the accessible torsion points have much smaller order. The method introduced in [DKLPW20] (and later improved in [FLLW23]) requires to cut J into several pieces in order to compute σ as a chain of isogenies. This complicated mechanism is by far the bottleneck in the signing algorithm.

In order to avoid this costly ideal to isogeny translation in SQIsignHD, we shall no longer require σ to have smooth degree and embed it in an isogeny of dimension 4 or 8 having smooth degree. This embedding will provide an efficient representation, and is faster to compute than the one in the original SQIsign. We shall also explain why this improves security in Section 5.

2.2 Embedding isogenies in higher dimension with Kani's lemma

In this section, we explain in more detail this idea of embedding isogenies in higher dimension. For that, we need a few definitions first.

Definition 2 (d-isogeny)**.** Let $\alpha : (A, \lambda_A) \longrightarrow (B, \lambda_B)$ be an isogeny between principally polarized abelian varieties. We say that α is a *d-isogeny* if $\widehat{\alpha} \circ \lambda_B \circ \alpha = [d]\lambda_A$, where $\widehat{\alpha} : \widehat{B} \longrightarrow \widehat{A}$ is the dual isogeny of α.

Equivalently, α is a d-isogeny if $\tilde{\alpha} \circ \alpha = [d]_A$, where $\widetilde{\alpha} := \lambda_A^{-1} \circ \widehat{\alpha} \circ \lambda_B$ is the dual isogeny of α with respect to the principal polarisations λ_A and λ_B.

Definition 3 (Isogeny diamond)**.** Let $a, b \in \mathbb{N}^*$. An *(a, b)-isogeny diamond* is a commutative diagram of isogenies between principally polarized abelian varieties

$$\begin{tikzcd} A' \arrow[r, "\varphi'"] & B' \\ A \arrow[u, "\psi"] \arrow[r, "\varphi"] & B \arrow[u, "\psi'"'] \end{tikzcd}$$

where φ and φ' are a-isogenies and ψ and ψ' are b-isogenies.

Lemma 4 (Kani)**.** *We consider an (a, b)-isogeny diamond as above, with $d := a + b$ prime to the characteristic of the base field of abelian varieties. Then, the isogeny $F : A \times B' \longrightarrow B \times A'$ given in matrix notation by*

$$F := \begin{pmatrix} \varphi & \widetilde{\psi'} \\ -\psi & \widetilde{\varphi'} \end{pmatrix}$$

is a d-isogeny with $d = a + b$, for the product polarisations.

If a and b are coprime, the kernel of F is

$$\ker(F) = \{(\widetilde{\varphi}(x), \psi'(x)) \mid x \in B[d]\}.$$

This lemma has first been proved in [Kan97, Theorem 2.3]. We also give a proof in Appendix E.1.

Remark 2.1. The existence of $F : A \times B' \to B \times A'$, implies the existence of $\varphi : A \to B$. We can recover φ as $\pi \circ F \circ \iota$ where ι is the embedding morphism $x \in A \longmapsto (x, 0) \in A \times B'$ and π is the projection from $B \times A'$ to B. Hence, F is an efficient representation of φ.

2.3 Application of Kani's lemma to SQIsign

Let us now see how we propose to use Kani's Lemma (Lemma 4) in SQIsignHD.

Signing in dimension 4. The idea is to embed the signature $\sigma : E_1 \longrightarrow E_2$ in an isogeny of dimension 4. We consider the 2-dimensional q-isogeny $\Sigma := \mathrm{Diag}(\sigma, \sigma) : E_1^2 \longrightarrow E_2^2$, and for $a_1, a_2 \in \mathbb{Z}$ and $i \in \{1, 2\}$ the $(a_1^2 + a_2^2)$-isogeny

$$\alpha_i := \begin{pmatrix} a_1 & a_2 \\ -a_2 & a_1 \end{pmatrix} \in \mathrm{End}(E_i^2).$$

Then, we have an isogeny diamond

$$\begin{tikzcd}
E_2^2 \arrow[r, "\alpha_2"] & E_2^2 \\
E_1^2 \arrow[u, "\Sigma"] \arrow[r, "\alpha_1"] & E_1^2 \arrow[u, "\Sigma"']
\end{tikzcd}$$

yielding an N-isogeny (with $N := q + a_1^2 + a_2^2$):

$$F := \begin{pmatrix} \alpha_1 & \widetilde{\Sigma} \\ -\Sigma & \widetilde{\alpha}_2 \end{pmatrix} \in \mathrm{End}(E_1^2 \times E_2^2).$$

Notation 5. We shall denote $F(\sigma, a_1, a_2)$ when we want to specify the dependence of F on σ, a_1, a_2.

We choose the parameters q, a_1, a_2, so that $N = \ell^e$, with ℓ a small prime and $e \in \mathbb{N}^*$ big enough. Provided that q and ℓ are coprime, we know that

$$\ker(F) = \{(\widetilde{\alpha}_1(P), \Sigma(P)) \mid P \in E_1^2[\ell^e]\}, \tag{1}$$

by Lemma 4. Then, knowing $\ker(F)$ we can compute F as an ℓ-isogeny chain and obtain an efficient representation of σ, as explained in Remark 2.1.

It follows that our idea requires to compute $\ker(F)$, which becomes easy once we know how to evaluate σ on $E_1[\ell^e]$, by formula 1. The idea is to use the alternate isogeny path $\varphi \circ \tau \circ \widehat{\psi} : E_1 \to E_2$. Since the signature requires to compute the three isogenies φ, ψ, τ, it will not cost too much to use them in order to evaluate σ. There are several technicalities to make it work in practice (such as to making sure that this alternate path has degree prime to ℓ) but it is manageable (see Appendix A.5).

Computing such a representation for the signature is simpler than in the original SQIsign protocol. This shifts the main computation effort to the verification, where the actual isogeny in dimension 4 must be computed.

Parameters. Even though we no longer impose $q = \deg(\sigma)$ to be smooth, we still impose conditions on q to make it work. We shall need $\ell^e - q$ to be a prime congruent to 1 modulo 4 in order to decompose it easily as a sum of two squares $\ell^e - q = a_1^2 + a_2^2$ by Cornacchia's algorithm [Cor08]. This choice of q ensures its coprimality with ℓ, as required to compute $\ker(F)$. The exponent e is fixed to be as small as possible so that there always exists an isogeny $\sigma : E_1 \longrightarrow E_2$ of ℓ^e-*good* degree in the sense of the following definition. In practice, the smallest values for q are close to $\sqrt{p}$ (Section 4.2) so ℓ^e will be slightly bigger than $\sqrt{p}$.

Definition 6. We say that an integer q is ℓ^e-*good* when $\ell^e - q$ is a prime number congruent to 1 modulo 4.[6]

Remark 7 (The issue of the signature distribution)**.** Those restrictions on the degree q impact the distribution of signatures. The bound $\ell^e \simeq \sqrt{p}$ is also restrictive (see Theorem 42). For that reason, we need some plausible heuristic assumptions to prove the zero-knowledge property of our scheme. This can be fixed by going to dimension 8 as long as $q < \ell^e$ and $\ell^e = \Omega(p^2)$. This way, we shall obtain a uniform distribution of signatures and a provably zero-knowledge scheme which is the purpose of our scheme in dimension 8 that we present below.

Signing in dimension 8. By Lagrange's four square theorem [Lag70], if $q < \ell^e$, there always exists $a_1, \cdots, a_4 \in \mathbb{Z}$ such that $q + a_1^2 + \cdots + a_4^2 = \ell^e$. We can find such a decomposition in polynomial time in e with Rabin and Shallit's algorithm [Rab86] improved by Pollack and Treviño [PT18]. We then consider the endomorphisms

$$\alpha_i := \begin{pmatrix} a_1 & -a_2 & -a_3 & -a_4 \\ a_2 & a_1 & a_4 & -a_3 \\ a_3 & -a_4 & a_1 & a_2 \\ a_4 & a_3 & -a_2 & a_1 \end{pmatrix} \in \mathrm{End}(E_i^4),$$

for $i \in \{1, 2\}$, which are $(a_1^2 + \cdots + a_4^2)$-isogenies, and the q-isogeny $\Sigma := \mathrm{Diag}(\sigma, \cdots, \sigma) : E_1^4 \longrightarrow E_2^4$. As previously, by Kani's lemma, we have the ℓ^e-isogeny

$$F := \begin{pmatrix} \alpha_1 & \widetilde{\Sigma} \\ -\Sigma & \widetilde{\alpha}_2 \end{pmatrix} \in \mathrm{End}(E_1^4 \times E_2^4).$$

Similarly to dimension 4, we write $F(\sigma, a_1, \cdots, a_4)$ to highlight the dependence of F on $\sigma, a_1, \cdots, a_4$. To ensure the uniformity of the response, in dimension 8 we no longer restrict to the case q prime to ℓ. This means we might have to embed in dimension 8 a factor of σ of degree prime to ℓ instead of σ (see Appendix C for details). As in dimension 4, can compute $\ker(F)$ by evaluating σ on $E_1[\ell^e]$ and then compute F as an ℓ-isogeny chain. This way, we can represent any signature isogeny σ of degree $q < \ell^e$, with the implications on the security proof that we mentioned before. However, computing isogenies in dimension 8

---

[6] One could improve slightly the scheme by defining ℓ^e-good integers as integers q such that $\ell^e - q = sq'$, with s a smooth integer whose prime factors are all congruent to 1 modulo 4 and q' is a prime congruent to 1 modulo 4. Indeed, all we really need is that $\ell^e - q$ is easy to factor so Cornacchia's algorithm can be applied efficiently. This alternate definition would improve a bit the search for ℓ^e-good integer, but we went for the simplest definition.

is much more costly than in dimension 4 (though, still polynomial), so we do not recommend to use this representation and only propose it in the alternate version RigorousSQIsignHD.

More generally, the same techniques allow, given an ideal I representing an isogeny of degree q, to give an efficient representation of the isogeny σ associated to I by the Deuring correspondance, even when q is not smooth, see Appendices A.3 and A.4.

Why not signing in dimension 2? The cost of computing an isogeny grows exponentially with the dimension [LR12; LR15; LR23]. For that reason, finding an efficient representation in dimension 2 could be fruitful for SQIsignHD. On the other hand, the higher the dimension, the lesser the constraints on the isogeny σ. We have already seen that going from dimension 4 to 8 relaxes the constraints on $q = \deg(\sigma)$. Unsurprisingly, the constraints on σ are tighter in dimension 2. So far, under those constraints, we have failed to provide an efficient and secure version of SQIsignHD. We leave this question to future works.

3 Key generation, commitment and challenge

To evaluate σ on the ℓ^e-torsion, as required for the response computation, we apply the $\mathsf{EvalTorsion}_{\ell^f}$ procedure (Line 11) which uses the alternate path $\varphi \circ \tau \circ \widehat{\psi} : E_1 \longrightarrow E_2$ formed by the challenge φ, secret key τ and commitment isogeny ψ along with their ideals I_φ, I_τ and I_ψ. These ideals are also necessary to compute the ideal I_σ.

For the $\mathsf{EvalTorsion}_{\ell^f}$ procedure to work, the degrees of φ, τ and ψ must be prime to ℓ. The ideals I_ψ and I_τ can be generated directly along with ψ and τ. However, the computation of I_φ uses the procedure $\mathsf{IsogenyToIdeal}$ (Line 10) which requires a precomputation in the key generation phase. Namely, the prover will need to generate an alternate secret path $\tau' : E_0 \longrightarrow E_A$ of degree $D_{\tau'}$ prime to D_φ along with the secret key $\tau : E_0 \longrightarrow E_A$. This will be explained in section 3.3.

3.1 Accessible torsion and choice of the prime characteristic

The choice of p is usually made to provide enough accessible torsion for our isogeny computations. In FastSQIsignHD, we can choose $p = c\ell^f \ell'^{f'} - 1$ with $\ell \neq \ell'$ two primes, $c \in \mathbb{N}^*$ small and $\ell^f \simeq \ell'^{f'} \simeq \sqrt{p}$, as in SIDH [JD11]. In practice, $\ell = 2$ and $\ell' = 3$ are the best choice.

We then require $D_\tau = D_\psi = \ell'^{2f'}$, $D_\varphi = \ell'^{f'}$ and $D_{\tau'} = \ell^{2f}$[7]. This choice ensures that D_τ, D_ψ and D_φ are prime to ℓ and that $D_{\tau'}$ is prime to D_φ, as needed. We also have $D_\tau, D_\psi, D_{\tau'} = \Theta(p)$, which guarantees (at least heuristically) that the public key E_A and the commitment E_1 are computationally

---

[7] Actually, we will not have exactly $D_\tau = D_\psi = \ell'^{2f'}$ but D_τ and D_ψ will be divisors of $\ell'^{2f'}$ close to $\ell'^{2f'}$. It will be the same for $D_{\psi'}$ (see Algorithm 10). We assume equality to simplify the exposition.

indistinguishable from a uniformly random supersingular elliptic curve – which is essential to the security of FastSQIsignHD.

This choice of prime also provides enough accessible torsion to compute the ℓ^e-isogeny F representing the response σ in dimension 4, where $\ell^e > q := D_\sigma$. In fact, we even have much more than the minimum requirement since it will be enough to have $2f \geq e+4$ (so $\ell^f = \Omega(p^{1/4})$) as will be explained in Sections 4.3 and 4.4 and Remark 4.2. This freedom is welcome anyway because it allows us to take ℓ^e slightly bigger than $\sqrt{p}$ to make sure that we can always find an ideal I of ℓ^e-good norm $q < \ell^e$ (see Section 4.2).

We finally discuss the security requirements regarding the size of p. The best known classical key recovery attacks are the meet-in-the-middle algorithm or the general Delfs and Galbraith attack [DG16] in the supersingular isogeny graph which both have a complexity in $\tilde{O}(\sqrt{p})$. Using Grover's algorithm [Gro96], we reach a quantum complexity of $\tilde{O}(p^{1/4})$. Hence, to ensure a classical security level of λ bits and a quantum security level of $\lambda/2$ bits, we need to take $p = \Theta(2^{2\lambda})$, as in the original version of SQIsign [DKLPW20].

We give below some concrete values of primes for NIST levels 1, 3 and 5.

NIST security level	Security parameter λ (bits)	Prime p
NIST-I	128	$13 \cdot 2^{126} \cdot 3^{78} - 1$
NIST-III	192	$5 \cdot 2^{193} \cdot 3^{122} - 1$
NIST-V	256	$11 \cdot 2^{257} \cdot 3^{163} - 1$

3.2 Challenge generation

To ensure a soundness security level of λ bits, the challenge space needs to have size at least $2^\lambda \simeq \sqrt{p}$. We also need the challenge degree D_φ to be prime to ℓ to be able to push the points of order ℓ^f through ψ during the signing procedure. The challenge generation procedure $\mathsf{Challenge}_{D_\varphi}$ is the same in the fast and provably secure challenge generation procedure. It simply generates a random element $P \in E_A$ of order D_φ and computes φ of kernel $\langle P \rangle$. Only the degree D_φ changes; in FastSQIsignHD, we take $D_\varphi = \ell'^{f'}$.

3.3 Fast key generation and commitment

We now present FastDoublePath (Algorithm 1) the main algorithmic block for the key generation and commitment of FastSQIsignHD. The goal of this algorithm is to generate two isogeny paths $\phi, \phi' : E_0 \longrightarrow E$ of degree dividing $\ell^{2f} \simeq p$ and $\ell'^{2f'} \simeq p$ respectively, computing the kernel ideals I_ϕ and $I_{\phi'}$ along the way. This algorithm is directly applicable to the key generation procedure FastKeyGen where we need to generate a double path to be able to compute the challenge kernel ideal I_φ (using Algorithm 10 and an ℓ-isogeny path of degree prime to ℓ') in order to apply the $\mathsf{EvalTorsion}_{\ell^f}$ procedure (with the ℓ'-isogeny path of degree prime to ℓ).

For the commitment FastCommit, we only need the ℓ'-isogeny path $\psi = \phi'$ but the algorithm is essentially the same, except that we do not compute ϕ and

I_ϕ completely. This is the reason why we changed the side of the challenge: to save time in the commitment phase. Had we started the challenge φ from E_1 as in SQISign, we would have needed to compute a double isogeny path in the commitment phase. Instead, we precompute this double path during the key generation.

Note that generating isogenies of degree $\simeq p$ is essential for security reasons, in order to ensure that the codomain E is heuristically close to a random elliptic curve in the supersingular isogeny graph. To compute such long isogeny paths, however, we are limited by the accessible torsion in E_0 (we have access to the $\ell^f \ell'^{f'}$-torsion only). To circumvent this difficulty, we use pushforward isogenies, as defined in [DKLPW20, § 4.1].

Definition 8. Let $\rho : E \longrightarrow E_1$ and $\theta : E \longrightarrow E_2$ be two isogenies with coprime degree. The *pushforward* of ρ via θ, denoted by $\rho' := [\theta]_*\rho$ is an isogeny $E_2 \longrightarrow E_3$ satisfying $\ker(\rho') = \theta(\ker(\rho))$.

Remark 9. θ and ρ satisfy $[\theta]_*\rho \circ \theta = [\rho]_*\theta \circ \rho$. In particular, $[\theta]_*\rho$ and $[\rho]_*\theta$ have the same codomain. If I and J are the ideals associated to ρ and θ respectively via the Deuring correspondence, we denote by $[J]_*I$ the *pushforward ideal* associated to $[\theta]_*\rho$. By [DKLPW20, Lemma 3], the ideal $[J]_*I$ can be computed as follows: $[J]_*I = J^{-1} \cdot (I \cap J)$.

The algorithm. The idea is to construct the isogenies ϕ and ϕ' (of degree dividing ℓ^{2f} and $\ell'^{2f'}$ respectively) by finding an endomorphism γ of degree dividing $\ell^{2f}\ell'^{2f'}$, and factoring it as $\gamma = \hat{\phi}' \circ \phi$. Since $\ell^{2f}\ell'^{2f'} = \Theta(p^2) = \omega(p)$, we can easily find $\gamma \in \mathcal{O}_0$ non divisible by ℓ or ℓ', of norm $\mathrm{nrd}(\gamma) = \ell^{2g}\ell'^{2g'}$ with $g \leq f$ close to f and $g' \leq f'$ close to f', using [Ler22, Algorithm 4].

Since ℓ^{2f} (and $\ell'^{2f'}$) exceeds the available torsion, some "pushforward gymnastics" is required to compute the factorisation. We thus decompose $\varepsilon(\gamma) = \widehat{\rho_2} \circ \rho_1$ where ρ_1 and ρ_2 are isogenies $E_0 \longrightarrow E'$ of degree $\ell^g\ell'^{g'}$ and ε is an isomorphism $\mathcal{O}_0 \xrightarrow{\sim} \mathrm{End}(E_0)$. $\varepsilon(\gamma)$ being cyclic, according to the following lemma, ρ_1 and its associated kernel ideal K_1 are given by:

$$\ker(\rho_1) = \ker(\varepsilon(\gamma)) \cap E_0[\ell^g\ell'^{g'}] \quad \text{and} \quad K_1 = \mathcal{O}_0\gamma + \mathcal{O}_0\ell^g\ell'^{g'}.$$

Similarly, $\ker(\rho_2) = \ker(\widehat{\varepsilon(\gamma)}) \cap E_0[\ell^g\ell'^{g'}]$ and the associated kernel ideal is $K_2 = \mathcal{O}_0\overline{\gamma} + \mathcal{O}_0\ell^g\ell'^{g'}$.

Lemma 10. *Let $\rho : E \longrightarrow E'$ be a cyclic isogeny decomposed into $\rho = \theta \circ \rho_1$. Then we have:*

(i) $\ker(\rho_1) = \ker(\rho) \cap E[d_1]$ *with* $d_1 := \deg(\rho_1)$.

(ii) *If ρ is a cyclic endomorphism ($E = E'$), then the kernel ideal of ρ_1 is $K_1 = \mathcal{O}\rho + \mathcal{O}d_1$, where $\mathcal{O} := \mathrm{End}(E)$.*

Proof. Since $\rho = \theta \circ \rho_1$ and $\deg(\rho_1) = d_1$, we clearly have $\ker(\rho_1) \subseteq \ker(\rho) \cap E[d_1]$. Since ρ is cyclic, there exists a generator $P \in E$ of $\ker(\rho)$ of order $d := \deg(\rho)$ and we have $\ker(\rho) \cap E[d_1] = \langle [d/d_1]P \rangle$, where $[d/d_1]P$ has order d_1, so we conclude that the inclusion is an equality by cardinality, since ρ_1 is separable. (i) follows.

To prove (ii), we remark that $E[\mathcal{O}\rho + \mathcal{O}d_1] = E[\rho] \cap E[d_1] = \ker(\rho_1)$, where the last equality was proved in (i). Then, we conclude that $K_1 = \mathcal{O}\rho + \mathcal{O}d_1$ by injectivity of the Deuring correspondence between left $\mathcal{O}$-ideals and isogenies of domain E [Voi20, Proposition 42.2.16]. This completes the proof. $\square$

Then, we can decompose ρ_1 and ρ_2 into $\rho_1 = \widehat{\theta_1'} \circ \theta_1$ and $\rho_2 = \widehat{\theta_2} \circ \theta_2'$ where the θ_i are isogenies of degree ℓ^g and the θ_i' are isogenies of degree $\ell'^{g'}$ for $i \in \{1, 2\}$, as in the following diagram:

$$\begin{tikzcd}
& F_2 \arrow[d, "{[\theta_2]_*\theta_1'}"] & \\
E_0 \arrow[ur, "\theta_2'"] \arrow[dr, "\theta_1"'] & E & E' \arrow[ul, "\theta_2"'] \arrow[dl, "\theta_1'"] \\
& F_1 \arrow[u, "{[\theta_1']_*\theta_2}"'] &
\end{tikzcd}$$

The pushforward isogenies $[\theta_1']_*\theta_2$ and $[\theta_2]_*\theta_1'$ have the same codomain E and degree ℓ^g and $\ell'^{g'}$ respectively. Hence, $\phi := [\theta_1']_*\theta_2 \circ \theta_1$ and $\phi' := [\theta_2]_*\theta_1' \circ \theta_2'$ are isogenies $E_0 \longrightarrow E$ of desired degrees ℓ^{2g} and $\ell^{2g'}$ respectively. By Lemma 10, we can compute $\ker(\theta_1)$, $\ker(\theta_2')$, $\ker(\theta_1')$ and $\ker(\theta_2)$, and obtain the θ_i and θ_i' with Vélu's formulas [Vél71]. We then compute $\ker([\theta_1']_*\theta_2) = \theta_1'(\ker(\theta_2))$ and $\ker([\theta_2]_*\theta_1') = \theta_2(\ker(\theta_1'))$ and use Vélu's formulas again. We then easily get ϕ and ϕ'.

Since $\varepsilon(\gamma) = \widehat{\rho_2} \circ \rho_1$ and $[\theta_1']_*\theta_2 \circ \theta_1' = [\theta_2]_*\theta_1' \circ \theta_2$, we get that $\varepsilon(\gamma) = \hat{\phi}' \circ \phi$. Lemma 10 implies that the ideals $J := \mathcal{O}_0\gamma + \mathcal{O}_0\ell^{2g}$ and $J' := \mathcal{O}_0\overline{\gamma} + \mathcal{O}_0\ell'^{2g'}$ are the respective kernel ideals of ϕ and ϕ'. Algorithm 1 follows.

Remark 3.1. The FastKeyGen procedure calls Algorithm 1 directly. For FastCommit, only ϕ' and J' are necessary, so we use a slightly modified version of Algorithm 1 where $\mathcal{H}_1$ (line 4), θ_1 (line 5), ϕ (line 7), and J (line 8) are not computed.

4 Response and verification

The goal of this section is to present a precise description of the algorithmic building blocks required by our new signature scheme in dimension 4. We refer to B.4 for details on the dimension 8 version.

Throughout this section, we assume that the prover has generated two secret key paths $\tau, \tau' : E_0 \longrightarrow E_A$ of respective degrees $D_\tau = \ell'^{2f'}$ and $D_{\tau'} = \ell^{2f}$ and a secret commitment path $\psi : E_0 \longrightarrow E_1$ of degree $D_\psi = \ell'^{2f'}$. We also assume the prover has access to the challenge $\varphi : E_A \longrightarrow E_2$ of degree $D_\varphi = \ell'^{f'}$.

Algorithm 1: $\mathsf{FastDoublePath}_{\ell^f,\ell'^{f'}}$

```
Data: A basis of $\mathcal{O}_0$ and an isomorphism $\varepsilon : \mathcal{O}_0 \xrightarrow{\sim} \mathrm{End}(E_0)$.
Result: Two cyclic isogenies $\phi : E_0 \longrightarrow E$ of degree dividing $\ell^{2f}$ and $\phi' : E_0 \longrightarrow E$ of degree dividing $\ell'^{2f'}$ and their respective kernel ideals $J$ and $J'$.
    Use [Ler22, Algorithm 4] to find $\gamma \in \mathcal{O}_0$ non divisible by $\ell$ and $\ell'$ of norm $\mathrm{nrd}(\gamma) = \ell^{2g}\ell'^{2g'}$ with $g \leq f$ close to $f$ and $g' \leq f'$ close to $f'$;
    Evaluate $\varepsilon(\gamma)$ and $\varepsilon(\overline{\gamma})$ on a basis of $E_0[\ell^g\ell'^{g'}]$ and solve discrete logarithm problems to compute $\mathcal{G}_1 := \ker(\varepsilon(\gamma)) \cap E_0[\ell^g\ell'^{g'}]$ and $\mathcal{G}_2 := \ker(\widehat{\varepsilon(\gamma)}) \cap E_0[\ell^g\ell'^{g'}]$;
    Compute $\rho_i : E_0 \longrightarrow E'$ of kernel $\mathcal{G}_i$ for $i = 1, 2$;
    Compute $\mathcal{H}_1 := \ker(\varepsilon(\gamma)) \cap E_0[\ell^g]$, $\mathcal{H}'_2 := \ker(\widehat{\varepsilon(\gamma)}) \cap E_0[\ell'^{g'}]$, $\mathcal{H}'_1 := \ker(\widehat{\rho_1}) \cap E'[\ell'^{g'}]$ and $\mathcal{H}_2 := \ker(\widehat{\rho_2}) \cap E'[\ell^g]$;
    Compute $\theta_i$ of kernel $\mathcal{H}_i$ and $\theta'_i$ of kernel $\mathcal{H}'_i$ for $i = 1, 2$;
    Compute $[\theta'_1]_*\theta_2$ and $[\theta_2]_*\theta'_1$ of kernels $\theta'_1(\ker(\theta_2))$ and $\theta_2(\ker(\theta'_1))$ respectively;
    Let $\phi := [\theta'_1]_*\theta_2 \circ \theta_1$ and $\phi' := [\theta_2]_*\theta'_1 \circ \theta'_2$;
    Let $J := \mathcal{O}_0\gamma + \mathcal{O}_0\ell^{2g}$ and $J' := \mathcal{O}_0\overline{\gamma} + \mathcal{O}_0\ell'^{2g'}$;
    Return $\phi, \phi', J, J'$;
```

4.1 Overview of the response computation

In this section, we present the algorithm $\mathsf{FastRespond}$ used to compute the response in the FastSQIsignHD identification protocol (in dimension 4) and its verification counterpart $\mathsf{FastVerify}$.

Those algorithms use the following sub-algorithms that will be introduced in this section (if not already):

- $\mathsf{IsogenyToIdeal}(\varphi, \tau', I_{\tau'})$ (presented in Appendix A.4) takes as input a basis of $\mathrm{End}(E_0)$ that we can evaluate on points, an isogeny $\varphi : E_A \longrightarrow E_2$ of degree D_φ, an isogeny $\tau' : E_0 \longrightarrow E_A$ of degree prime to D_φ, its ideal $I_{\tau'} \subset \mathcal{O}_0$ and returns the kernel ideal I_φ of φ.
- $\mathsf{RandomEquivalentIdeal}_{\ell^e}$ takes as input an $\mathcal{O}_0$-left ideal J and returns an equivalent ideal I that is uniformly random among ideals of norm $\leq \ell^e$.
- $\mathsf{EvalTorsion}_{\ell^f}$ (presented in A.5) evaluates a non-smooth degree isogeny on ℓ^f-torsion points knowing its kernel ideal and an alternate smooth degree path. Namely, it takes as input an ideal I connecting $\mathcal{O} \cong \mathrm{End}(E)$ and $\mathcal{O}' \cong \mathrm{End}(E')$, a basis (P_1, P_2) of $E[\ell^f]$, two isogenies $\rho_1 : E_0 \longrightarrow E$ and $\rho_2 : E_0 \longrightarrow E'$ of smooth degrees prime to ℓ, with their respective kernel ideals I_1 and I_2 and returns $(\phi_I(P_1), \phi_I(P_2))$, where $\phi_I : E \longrightarrow E'$ is the isogeny associated to I.
- $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}$ takes as input an ℓ^e-good integer q, integers a_1, a_2 such that $a_1^2 + a_2^2 + q = \ell^e$, a basis (P_1, P_2) of $E_1[\ell^f]$, $(\sigma(P_1), \sigma(P_2))$, where $\sigma : E_1 \longrightarrow E_2$ is a q-isogeny, and returns a chain of 4-dimensional ℓ-isogenies whose composition is $F(\sigma, a_1, a_2)$ as in Notation 5.

- $\mathsf{IsValid}_4$, with input $F, E_1, E_2, \ell^e, \ell^f$, checks if F is a valid output of $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}$ representing an isogeny $\sigma : E_1 \longrightarrow E_2$ in dimension 4.

The prover sends the image of two points P_1, P_2 forming a basis of $E_1[\ell^f]$ by σ and its degree q. The verifier can then use q to compute a_1, a_2 and compute $F(\sigma, a_1, a_2)$ with the $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}$ procedure. If the computation succeeds and is validated by the $\mathsf{IsValid}_4$ procedure, then the verification is complete. Algorithm 3 follows.

Remark 4.1 (On the ℓ^f-torsion basis). It is sufficient to send the data $(\sigma(P_1), \sigma(P_2), q)$ to the verifier as the basis (P_1, P_2) can be computed canonically knowing E_1 by classical compression techniques developed for SIDH [AJKKL16; ZSPDB18]. This decreases the communications size at a small computational cost. Later, with the compression/decompression algorithms (see Algorithms 6 and 7), we will see how to further compress this data.

Note that we use a basis of the ℓ^f-torsion with $2f \geq e + 4$ here because we might not have the ℓ^e-torsion accessible. We can still compute F with this partial information as explained in Section 4.3.

To respond, the prover starts by computing an ideal $I \sim \overline{I_\psi} \cdot I_\tau \cdot I_\varphi$ connecting $\mathcal{O}_1 \cong \mathrm{End}(E_1)$ to $\mathcal{O}_2 \cong \mathrm{End}(E_2)$ of ℓ^e-good norm q and prime to ℓ' with uniform distribution using $\mathsf{RandomEquivalentIdeal}_{\ell^e}$. The coprimality with ℓ' is justified by security reasons (see Section 5.1). Then, the prover generates the basis (P_1, P_2) of $E_1[\ell^f]$ canonically and evaluates σ on it with $\mathsf{EvalTorsion}_{\ell^f}$ using I (kernel ideal of σ) and the paths $\psi : E_0 \longrightarrow E_1$ and $\varphi \circ \tau : E_0 \longrightarrow E_2$ of degrees prime to ℓ.

As input of Algorithms 2 and 3, we denote by:

- $\mathsf{FastSetup}$, the public parameters of FastSQIsignHD, $p = c\ell^f \ell'^{f'} - 1$, ℓ, ℓ', f, f', the exponent e and the elliptic curve $E_0/\mathbb{F}_p$;
- $\mathsf{SecretKey}$, the isogenies $\tau, \tau' : E_0 \longrightarrow E_A$ of degrees $D_\tau = \ell'^{2f'}$ and $D_{\tau'} = \ell^{2f}$ respectively along with their kernel ideals I_τ and $I_{\tau'}$;
- $\mathsf{CommitData}$, the isogeny $\psi : E_0 \longrightarrow E_1$ of degree $D_\psi = \ell'^{2f}$ and its kernel ideal I_ψ;
- $\mathsf{ChallData}$, the isogeny $\varphi : E_A \longrightarrow E_2$ of degree $D_\varphi = \ell'^{f'}$.

4.2 Finding a uniformly random tight response ideal

In this section, we present the algorithm $\mathsf{RandomEquivalentIdeal}_{\ell^e}$ taking a left $\mathcal{O}_0$-ideal J as input and returning an ideal I which is uniformly random among the ideals $I \sim J$ of norm $q < \ell^e$. By [DKLPW20, Lemma 1], all the equivalent ideals $I \sim J$ are of the form $\chi_J(\alpha) := J\overline{\alpha}/\mathrm{nrd}(J)$ for some $\alpha \in J$ and α determines I up to multiplication by an element of $\mathcal{O}_0^\times$. Besides, the norm of $I = \chi_J(\alpha)$ is $q_J(\alpha) := \mathrm{nrd}(\alpha)/\mathrm{nrd}(J)$, so we need $q_J(\alpha) \leq \ell^e$.

Hence, to sample an ideal $I \sim J$ such that $\mathrm{nrd}(I) \leq \ell^e$ with uniform distribution is equivalent to sample $\alpha \in J \setminus \{0\}$ such that $q_J(\alpha) \leq \ell^e$ with uniform

Algorithm 2: FastRespond

```
Data: FastSetup, SecretKey, CommitData and ChallData.
Result: $(\sigma(P_1), \sigma(P_2), q)$, where $(P_1, P_2)$ is a canonically determined basis of
        $E_1[\ell^f]$ and $\sigma : E_1 \longrightarrow E_2$ is an isogeny of $\ell^e$-good degree $q$ prime to $\ell'$.
1 $I_\varphi \longleftarrow$ IsogenyToIdeal$(\varphi, \tau', I_{\tau'})$;
2 $J \longleftarrow \overline{I_\psi} \cdot I_\tau \cdot I_\varphi$;
3 $I \longleftarrow$ RandomEquivalentIdeal$_{\ell^e}(J)$ and $q \longleftarrow \mathrm{nrd}(I)$;
4 If $q$ is not $\ell^e$-good or $q \wedge \ell' \neq 1$, go back to line 3;
5 Compute the canonical basis $(P_1, P_2)$ of $E_1[\ell^f]$;
6 $(\sigma(P_1), \sigma(P_2)) \longleftarrow$ EvalTorsion$_{\ell^f}(I, P_1, P_2, \psi, \varphi \circ \tau, I_\psi, I_\tau \cdot I_\varphi)$;
7 Return $(\sigma(P_1), \sigma(P_2), q)$;
```

Algorithm 3: FastVerify

```
Data: FastSetup, $E_1, E_2$ and an output $R$ from FastRespond.
Result: Determines if $R$ is a valid response.
Try to parse $R := (R_1, R_2, q)$, where $R_1, R_2 \in E_2[\ell^f]$ and $q < \ell^e$ and return
  False if it fails;
If $q$ is not $\ell^e$-good or $q \wedge \ell' \neq 1$, return False;
Compute the canonical basis $(P_1, P_2)$ of $E_1[\ell^f]$;
Find $a_1, a_2 \in \mathbb{Z}$ such that $a_1^2 + a_2^2 = \ell^e - q$ using Cornacchia's algorithm
  [Cor08];
$F \longleftarrow$ RepresentIsogeny$_{4,\ell^e,\ell^f}(E_1, E_2, a_1, a_2, P_1, P_2, R_1, R_2)$;
if $F \neq$ False then
    Return IsValid$_{4,\ell^e,\ell^f,\ell'^{f'}}(F, E_1, E_2, a_1, a_2)$;
else
    Return False.
end
```

distribution. If we fix a basis of J, we can see q_J as a primitive positive definite integral quadratic form with four variables. By the following lemma, which is a simple generalization of [Wes22, Lemma 3.3], we can sample uniformly $\alpha \in J$ such that $q_J(\alpha) \leq \ell^e$. RandomEquivalentIdeal$_{\ell^e}$ calls this procedure to get $\alpha \in J$ uniform and rejects the result if $\alpha = 0$. Then the distribution of α is still uniform but in $J \setminus \{0\}$. The proofs of the two following lemmas can be found in Appendix E.2.

Lemma 11. *Let f be a primitive positive definite integral quadratic form in k variables and let $\rho > 0$. Then there exists an algorithm that samples uniformly random elements from the set*

$$\{x \in \mathbb{Z}^k \mid f(x) \leq \rho\}$$

in polynomial time in $\log(\rho)$ *and the length of f (namely, the maximal number of bits of the coefficients of f). This algorithm runs in exponential time in k.*

For $\mathsf{RandomEquivalentIdeal}_{\ell^e}(J)$ to terminate, we need to find $\alpha \in J \setminus \{0\}$ such that $q_J(\alpha) \leq \ell^e$. For such an α to exist, we need $\ell^e = \Omega(\sqrt{p})$ according to the following lemma (Lemma 12).

Lemma 12. *Let $\mathcal{O}$ be a maximal order and J be a left $\mathcal{O}$-ideal. Then there exists $\alpha \in J$ such that $q_J(\alpha) \leq 2\sqrt{2p}/\pi$.*

In the procedure $\mathsf{FastRespond}$, we reject the results of $\mathsf{RandomEquivalentIdeal}_{\ell^e}$ whose norm is not ℓ^e-good or divisible by ℓ'. If it terminates, this rejection sampling outputs ideals which are uniformly random among the targeted ones, as desired. However, we can only give a heuristic argument for the termination. Assuming that $q_J(\alpha)$ behaves like a random integer, we should expect to find a suitable $\alpha \in J$ with probability $O(1/\log(p))$. Hence, taking ℓ^e a few bits over $\sqrt{p}$ might be sufficient. For that reason, in our choice of parameters, we only have accessible ℓ^f-torsion with $\ell^f < \sqrt{p} < \ell^e$ (see Section 3.1). Proving formally that we can always find an ℓ^e-good value of $q_J(\alpha)$ would certainly require to increase ℓ^e by a lot. As [RT22] indicates, we should expect lower bounds close to $\ell^e = \omega(p^2)$, causing a huge efficiency loss.

4.3 Dividing the higher dimensional isogeny computation in two

As explained in Section 4.2, we do not necessarily have enough accessible torsion to compute the whole kernel of the higher dimensional representation of the response F. In this section, we explain in plain generality how to circumvent this difficulty. Hence, the following discussion applies to both dimension 4 and 8. Let us keep the notations of Section 2.2. Recall that we have the following isogeny

$$F := \begin{pmatrix} \varphi & \widetilde{\psi}' \\ -\psi & \widetilde{\varphi}' \end{pmatrix}, \quad \text{with} \quad \ker(F) = \{(\widetilde{\varphi}(x), \psi'(x)) \mid x \in B[d]\}.$$

To compute F, we need to evaluate $\widetilde{\varphi}$ and ψ' on $B[d]$, so we need to have accessible d-torsion. However, we assume that we only have d'-accessible torsion with $d'|d$.

The idea is to decompose $F = F_2 \circ F_1$ where $F_1 : \mathcal{A} := A \times B' \longrightarrow \mathcal{C}$ and $F_2 : \mathcal{C} \longrightarrow \mathcal{B} := B \times A'$ are respectively d_1 and d_2-isogenies such that $d_1, d_2 | d'$ and to use the following proposition (proved in Appendix E.3) to compute F_1 and $\widetilde{F_2}$ to infer F.

Proposition 13. *Suppose d prime to p so that F is separable. Then:*

(i) *We can always decompose $F = F_2 \circ F_1$, as above.*
(ii) $\ker(F_1) \subseteq \ker(F) \cap \mathcal{A}[d_1]$.
(iii) $\ker(\widetilde{F_2}) \subseteq \ker(\widetilde{F}) \cap \mathcal{B}[d_2] = F(\mathcal{A}[d]) \cap \mathcal{B}[d_2]$.
(iv) *When* $\ker(F)$ *has rank* $g := \dim(\mathcal{A})$*, those inclusions are equalities.*

In SQIsignHD, $d_1 = \ell^{e_1}$ and $d_2 = \ell^{e_2}$ with $e = e_1 + e_2$ and we have accessible ℓ^f-torsion such that $f \geq e_1, e_2$. Since $\ker(F)$ has maximal rank $g = 4$ (or 8), we have by point (iv) of the above proposition

$$\ker(F_1) = \ker(F)[\ell^{e_1}] = \{(\widetilde{\alpha_1}(P), \Sigma(P)) \mid P \in E_1^{g/2}[\ell^{e_1}]\}$$

and similarly, $\ker(\widetilde{F_2}) = \ker(\widetilde{F})[\ell^{e_2}] = \{(\alpha_1(P), -\Sigma(P)) \mid P \in E_1^{g/2}[\ell^{e_2}]\}$, with the notations of Section 2.3.

In Appendix F, we give an overview of the higher dimensional isogeny computation required in the procedures $\mathsf{RepresentIsogeny}_{g,\ell^e,\ell^f}$ of our SQIsignHD scheme. We provide a proof of concept `sagemath` implementation in dimension 4. Optimizing this implementation in a low level programming language is left for future works.

4.4 Computing the response isogeny representation

We finally give algorithms to compute the signature representation in dimension 4 using all the ideas presented in Section 4.3 and Appendix F. We refer to $\mathsf{KernelToIsogeny}_{g,\ell^e}(\mathscr{B}_0)$ as the algorithm computing an ℓ-isogeny chain in dimension g given a basis $\mathscr{B}_0$ of its kernel. We refer to Appendix F for more details on this algorithm.

In dimension 4, $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}$ (Algorithm 4) computes basis of $\ker(F_1)$ and $\ker(\widetilde{F_2})$ with $F := F_2 \circ F_1$, as in Section 4.3. Then, it calls $\mathsf{KernelToIsogeny}_{4,\ell^e}$ to obtain F_1 and $\widetilde{F_2}$ as isogeny chains. The ideas are the same in dimension 8.

Algorithm 4: $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}$

```
Data: $E_1, E_2$, $a_1, a_2 \in \mathbb{Z}$, a basis $(P_1, P_2)$ of $E_1[\ell^f]$ and $(\sigma(P_1), \sigma(P_2))$, where
      $\sigma : E_1 \longrightarrow E_2$ is a $q$-isogeny with $a_1^2 + a_2^2 + q = \ell^e$.
Result: An $\ell^{e_1}$-isogeny $F_1 : E_1^2 \times E_2^2 \longrightarrow \mathcal{C}$ and a $\ell^{e_2}$-isogeny $\widetilde{F_2} : E_1^2 \times E_2^2 \longrightarrow \mathcal{C}$
        such that $F(\sigma, a_1, a_2) = F_2 \circ F_1$, with $e_1, e_2 \leq f$ and $e_1 + e_2 = e$.
$e_2 \longleftarrow \lceil e/2 \rceil, e_1 \longleftarrow e - e_2$;
$Q_i \longleftarrow [\ell^{f-e_1}]P_i$, $R_i \longleftarrow [\ell^{f-e_1}]\sigma(P_i)$, $Q'_i \longleftarrow [\ell^{f-e_2}]P_i$, $R'_i \longleftarrow [\ell^{f-e_2}]\sigma(P_i)$
  for $i \in \{1, 2\}$;
$\mathscr{B}_0 \longleftarrow \big(([a_1]Q_i, [a_2]Q_i, R_i, 0)_{i\in\{1,2\}}, (-[a_2]Q_i, [a_1]Q_i, 0, R_i)_{i\in\{1,2\}}\big)$;
$\mathscr{C}_0 \longleftarrow (([a_1]Q'_i, -[a_2]Q'_i, -R'_i, 0)_{i\in\{1,2\}}, ([a_2]Q'_i, [a_1]Q'_i, 0, -R'_i)_{i\in\{1,2\}})$;
if $\mathscr{C}_0$ and $\mathscr{B}_0$ are valid kernels of $\ell^{e_1}$ and $\ell^{e_2}$-isogenies then
    $F_1 \longleftarrow \mathsf{KernelToIsogeny}_{4,\ell^{e_1}}(\mathscr{B}_0)$;
    $\widetilde{F_2} \longleftarrow \mathsf{KernelToIsogeny}_{4,\ell^{e_2}}(\mathscr{C}_0)$;
    Return $F_1$ and $\widetilde{F_2}$;
else
    Return False;
end
```

Proposition 14. *Algorithm 4 is correct. Namely, Algorithm 4 returns* $F_1, \tilde{F}_2$ *such that* $F_2 \circ F_1 = F(\sigma, a_1, a_2)$ *on entry* $a_1, a_2, P_1, P_2, \sigma(P_2)$, $\sigma(P_2)$, *where* $\sigma : E_1 \longrightarrow E_2$ *is a* q*-isogeny with* $a_1^2 + a_2^2 + q = \ell^e$.

Proof. See Appendix E.4. □

Remark 4.2. To make sure we have enough accessible torsion, we need $f \geq e_1, e_2$, so that $2f \geq e$. Actually, for $\mathsf{KernelToIsogeny}_{4,\ell^{e_i}}$ to work (with theta coordinates of level 2), we need $4\ell^{e_i}$-torsion points (see Appendix F.3). Then, when $\ell = 2$, we have $f \geq e_i + 2$, so $2f \geq e + 4$.

4.5 Verification

We describe the verification procedure $\mathsf{IsValid}_4$ taking as input the isogenies F_1 and $\widetilde{F_2}$ outputted by $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}$ and determining if they represent an isogeny $\sigma : E_1 \longrightarrow E_2$ of degree q. The idea is to check if F_1 and $\widetilde{F_2}$ have the same codomain (computed as principally polarized abelian varieties) and then evaluate $F_2 \circ F_1$ on some points to check that the degree is correct.

Algorithm 5: $\mathsf{IsValid}_{4,\ell^e,\ell^f\ell'^{f'}}$

```
Data: Elliptic curves $E_1, E_2$, integers $a_1, a_2 \in \mathbb{Z}$ and the output $(F_1, \widetilde{F_2})$ of
      $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}(E_1, E_2, a_1, a_2, *, *, *, *)$.
Result: Determines if $F_2 \circ F_1$ is an efficient repersentation of an isogeny $\sigma :
        E_1 \longrightarrow E_2$ of degree $q := \ell^e - a_1^2 - a_2^2$.
Let $(\mathcal{C}_1, \lambda_1)$ and $(\mathcal{C}_2, \lambda_2)$ be the respective codomains of $F_1$ and $\widetilde{F_2}$;
if $(\mathcal{C}_1, \lambda_1) \neq (\mathcal{C}_2, \lambda_2)$ then
    Return False;
else
    Find a point $Q \in E_1$ of order $\ell^f \ell'^{f'}$;
    Compute compute $F_2$ as the dual of $\widetilde{F_2}$ and $\underline{T} \longleftarrow F_2 \circ F_1(Q, 0, 0, 0)$;
    if $\underline{T} = ([a_1]Q, -[a_2]Q, *, 0)$ then
        Return True;
    else
        Return False;
    end
end
```

The following results (proved in Appendix E.5) ensure that our verification procedure is correct.

Proposition 15. *Algorithm 5 is correct. Namely, when given* $E_1, E_2, a_1, a_2, F_1, \tilde{F}_2$, *if Algorithm 5 returns* True, *then* $F_2 \circ F_1$ *is an efficient representation of an isogeny* $\sigma : E_1 \longrightarrow E_2$ *of degree* $q = \ell^e - a_1^2 - a_2^2$.

Corollary 16. *The verification procedure* FastVerify *(Algorithm 3) is correct. Namely, on input* (R_1, R_2, q)*,* FastVerify *returns* True *if and only if* (R_1, R_2, q) *defines an efficient representation of an isogeny* $\sigma : E_1 \longrightarrow E_2$ *of degree* q*, where* q *is* ℓ^e*-good and prime to* ℓ'*.*

In Appendix B.4, we provide algorithms $\mathsf{RepresentIsogeny}_{8,\ell^e,\ell^f}$ and $\mathsf{IsValid}_{8,\ell^e,\ell^f}$ in dimension 8 achieving similar correctness results.

5 Security analysis

In this section, we prove that the SQIsignHD identification protocol is secure, namely that it is complete, knowledge sound and honest-verifier zero knowledge. Recall that by [VV15, Theorem 7], it is sufficient to ensure that our signature scheme obtained by Fiat-Shamir transform is universally unforgeable under chosen message attacks in the random oracle model.

Completeness means that a honest execution of the protocol is always accepted by the verifier. This is true by Proposition 14 and by construction of IsValid. Knowledge soundness means that an attacker can only "guess" a response with very low probability. It is proven under the assumption that computing an endomorphism in a supersingular elliptic curve is hard, a well known difficult problem in isogeny based cryptography.

The honest-verifier zero-knowledge property implies that the response does not leak any information on the secret key τ. More precisely, we can simulate transcripts of the identification protocol without using the secret key with the same distribution as real transcripts. To construct such a simulator of SQIsignHD, we need access to an oracle evaluating isogenies of non-smooth degrees. In RigorousSQIsignHD, this oracle is very generic and we do not need any additional hypothesis to prove the zero-knowledge property (hence the name of this version). On the contrary, in FastSQIsignHD, the oracle definition is *ad hoc* and we need an additional heuristic assumption to prove the zero-knowledge property. However, it is very unlikely to build an attack on this assumption as we argue in Section 5.3 and both oracles do not undermine the knowledge soundess. As previously, this section mainly focuses on FastSQIsignHD and refer to Appendix D for a complete security analysis of RigorousSQIsignHD.

5.1 Knowledge soundness

The proof that FastSQIsignHD is knowledge sound is a straightforward special soundess argument identical to the original version of SQIsign [DKLPW20, Theorem 1]. Namely, we prove that given two transcripts with the same commitment but disctinct challenges, we can find an endomorphism in E_A. This *special soundness* property is sufficient to prove that SQIsignHD satisfies *knowledge soundness* [HL10, Theorem 6.3.2]. However, note that we have to require the prime ideal norm q to be not only ℓ^e-good but also prime to ℓ' in order to complete the proof.

Proposition 17. *Under the assumption that* $q = \deg(\sigma)$ *is always prime to* ℓ'*, the* FastSQIsignHD *identification protocol satisfies special soundness. Namely, given two transcripts* $(E_1, \varphi, R_1, R_2, q)$ *and* $(E_1, \varphi', R_1', R_2', q')$ *with the same commitment* E_1 *but different challenges* $\varphi \neq \varphi'$*, we can extract an efficient representation of a non-scalar endomorphism* $\alpha \in \mathrm{End}(E_A)$*.*

Proof. Let $(E_1, \varphi, R_1, R_2, q)$ and $(E_1, \varphi', R_1', R_2', q')$ be two FastSQIsignHD transcripts with the same commitment E_1 but different challenges $\varphi \neq \varphi'$. Then, by Corollary 16, (R_1, R_2, q) and (R_1', R_2', q') define efficient representations of isogenies $\sigma : E_1 \longrightarrow E_2$ and $\sigma' : E_1 \longrightarrow E_2'$ of degrees q and q' respectively which are ℓ^e-good and comprime with ℓ'. Knowing $(R_1, R_2) = (\sigma(P_1), \sigma(P_2))$, where (P_1, P_2) is a canonical basis of $E_1[\ell^f]$, we can also find $a_1, a_2 \in \mathbb{Z}$ such that ${a_1}^2 + {a_2}^2 + q = \ell^e$ and apply $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}$ to compute $F := F(\sigma, a_1, a_2)$ by Proposition 14. Then, F provides an efficient representation of $\widehat{\sigma}$.

Hence, we know an efficient representation of $\alpha := \widehat{\varphi'} \circ \sigma' \circ \widehat{\sigma} \circ \varphi \in \mathrm{End}(E_A)$. We now prove that α is not scalar. Indeed, if it was, we would have $\alpha = [\lambda]$ for some $\lambda \in \mathbb{Z}$ and $qq'\ell'^{2f'} = \lambda^2$ where $q := \deg(\sigma)$ and $q' := \deg(\sigma')$ are prime to ℓ'. Hence, $\lambda = \ell'^{f'}\lambda'$ with $\lambda' \in \mathbb{Z}$ prime to ℓ' ($\lambda'^2 = qq'$). It follows that $[q']\widehat{\sigma} \circ \varphi = [\lambda']\widehat{\sigma'} \circ \varphi'$. Since q, q' and λ' are prime to ℓ', we get that $\ker(\varphi) = \ker(\varphi')$ *i.e.* $\varphi = \varphi'$ up to post-composition by an automorphism. Contradiction. This completes the proof. □

For RigorousSQIsignHD, our special soundness argument does not apply because we have no guarantee on q in general. For that reason, we need to come back to the formal definition of knowledge soundess given in [HL10, Definition 6.3.1]. This analysis is conducted in Appendix D.1.

The previous proof of knowledge would be trivial if it was easy to find an endomorphism. Fortunately, this is a well-known hard problem in isogeny-based cryptography.

Problem 18 (Supersingular Endomorphism Problem)**.** Given a supersingular elliptic curve $E/\mathbb{F}_{p^2}$, find an efficient representation of a non-scalar endomorphism $\alpha \in \mathrm{End}(E)$.

This problem is very similar to [DKLPW20, Problem 1], except that we do not require the endomorphism to have smooth degree. This does not seem to make the problem easier since the endomorphisms solution to this can be evaluated (which was the reason why smoothness was imposed in the first place). The supersingular endomorphism ring problem (Problem 19) reduces to Problem 18. Problem 19 is notoriously hard and it has been proven it is equivalent to path finding in the supersingular ℓ-isogeny graph [Wes22]. The heuristic reduction from Problem 19 to 18 is given by [EHLMP18, Algorithm 8]. Basically, if we have an oracle finding endomorphisms of E, we call this oracle until we have found enough endomorphisms to generate $\mathrm{End}(E)$.

Problem 19 (Supersingular Endomorphism Ring Problem)**.** Given a supersingular elliptic curve $E/\mathbb{F}_{p^2}$, find four endomorphisms of E (that we can evaluate) forming a $\mathbb{Z}$-basis of $\mathrm{End}(E)$.

5.2 Heuristic zero-knowledge property

The proof of the zero-knowledge property of SQIsignHD uses an oracle generating isogenies of non-smooth degree. To our knowledge, there is no efficient algorithm implementing such an oracle. Nonetheless, it is believed that access to such an oracle does not affect the hardness of the underlying problem (the endomorphism ring problem, see Section 5.3). In RigorousSQIsignHD, the definition of such an oracle is very natural. In FastSQIsignHD, we add (mild) conditions on the degree to account for the computational constraints imposed by the method in dimension 4. These degree constraints are the main reason why the signatures are represented in dimension 8 instead of 4 in RigorousSQIsignHD. Our proof is limited to FastSQIsignHD in this section and we refer to Appendix D.2 for an analysis of RigorousSQIsignHD.

Definition 20. A *random uniform good degree isogeny oracle* (RUGDIO) is an oracle taking as input a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$ and returning an efficient representation of a random isogeny $\sigma : E \longrightarrow E'$ of ℓ^e-good degree prime to ℓ', such that:

(i) The distribution of E' is uniform in the supersingular isogeny graph.
(ii) The conditional distribution of σ given E' is uniform among isogenies $E \longrightarrow E'$ of ℓ^e-good degree prime to ℓ'.

In addition to the constraint on the degree of the RUGDIO output, we add constraints on the distributions of isogenies. Those constraints are necessary to construct a simulator of FastSQIsignHD. We already justified that these constraints can be mathematically satisfied, namely that for all supersingular elliptic curves E and E', there exists $\sigma : E \longrightarrow E'$ of ℓ^e-good norm. As explained in Section 4.2, taking ℓ^e slightly bigger than $\sqrt{p}$ by a few bits is heuristically sufficient. Note that to prove the zero-knowledge property, we not only need access to a RUGDIO, but also to make a heuristic assumption on the distribution of the commitment E_1. This assumption is no longer necessary in RigorousSQIsignHD.

Theorem 21. *Assume that the commitment E_1 is computationally indistinguishable from an elliptic curve chosen uniformly at random in the supersingular isogeny graph. Then, the* FastSQIsignHD *identification protocol is computationally honest-verifier zero knowledge in the RUGDIO model.*

In other words, under this assumption, there exists a random polynomial time simulator $\mathcal{S}$ with access to a RUGDIO that simulates transcripts (E_1, φ, R) with a computationally indistinguishable distribution from the transcripts of the FastSQIsignHD *identification protocol.*

Proof. First, we explain how to construct the simulator $\mathcal{S}$. The simulator starts by generating a challenge $\varphi' : E_A \longrightarrow E_2'$. Then, it applies the RUGDIO on entry E_2' to get an efficient representation of a dual response isogeny $\widehat{\sigma'} : E_2' \longrightarrow E_1'$. We can use this efficient representation to evaluate $\widehat{\sigma'}$ on $E_2'[\ell^f]$ and obtain its degree in polynomial time in $\log(p)$[8]. Then, as explained in the proof of Proposition 17,

---

[8] We can compute the norm of σ' on $E_2'[m]$ (which is $\deg(\sigma') \mod m$) for a bunch of small primes m and apply the Chinese remainder theroem.

we can compute a dimension 4 isogeny representation of $\widehat{\sigma'}$, which is also to an efficient representation of σ'. Hence, we can compute $R' := (\sigma'(P_1), \sigma'(P_2), q')$ in polynomial time, where (P_1, P_2) is a canonical basis of $E_1'[\ell^f]$ and $q' := \deg(\sigma')$.

We now prove that the transcripts (E_1', φ', R') of $\mathcal{S}$ are statistically indistinguishable from the transcripts (E_1, φ, R) of the FastSQIsignHD identification protocol. By construction, φ and φ' have the same distribution. Given E_2', by the definition of the RUGDIO, E_1' is uniformly random in the supersingular isogeny graph. Besides, E_1 is statistically close to uniformly random as well by assumption.

Finally, conditionally to E_1' and E_2', $\widehat{\sigma'}$ (represented by R') is uniformly random among the isogenies $E_2' \longrightarrow E_1'$ of ℓ^e-good degree prime to ℓ' by the definition of the RUGDIO. The dual map $\phi \longmapsto \widehat{\phi}$ being a bijection preserving the degree, conditionally to E_1' and E_2', σ' is also uniformly random among the isogenies $E_2' \longrightarrow E_1'$ of ℓ^e-good degree prime to ℓ'. By construction (see Section 4.2), conditionally to E_1 and E_2, σ has the same distribution. This completes the proof. ☐

It remains to justify that the commitment E_1 is computationally indistinguishable from an elliptic curve chosen uniformly at random in the supersingular isogeny graph. While RigorousCommit satisfies statistical indistinguishability, the variant FastCommit relies on heuristics. Consider the distributions on E_1 induced by the following procedures

1. Return the output E_1 of FastCommit.
2. Generate a uniformly random cyclic endomorphism γ of E_0 of degree $\ell^{2f}\ell'^{2f'}$. Factor it as $\gamma = \hat{\phi}' \circ \phi$ with $\deg(\phi) = \ell^{2f}$. Return the codomain E_1 of ϕ.
3. Generate a uniformly random cyclic isogeny ϕ from E_0 of degree ℓ^{2f}. Let E_1 be its codomain; let m be the number of cyclic isogenies $\phi' : E_0 \to E_1$ of degree $\ell'^{2f'}$. Return E_1 with probability m/M (for some fixed upper bound M on m, for instance $M = (\ell' + 1)\ell'^{2f'-1}$), otherwise resample.
4. Generate a uniformly random cyclic isogeny ϕ from E_0 of degree ℓ^{2f}; return its codomain E_1.
5. Return a uniformly random elliptic curve E_1.

We argue that each distribution from the list is somewhat close to the next. The difference between 1 and 2 is that in FastCommit, the endomorphism γ is not truly uniform: they follow a distribution biased by the fact that some intermediate result should be easy to factor. Since this property appears somewhat decorrelated from the final distribution of γ it seems plausible to argue that the distribution of γ in 1 is close to the one in 2. The distributions 2 and 3 are actually identical: distribution 3 simulated distribution 2 by rejection sampling. The difference between 3 and 4 is that m is not necessarily a (positive) constant; it is however heuristically expected to be almost a constant: there are about $(\ell' - 1)\ell'^{2f'-1}$ possible paths, and about $p/12$ vertices, so we expect about $m \approx 12(\ell' - 1)\ell'^{2f'-1}/p$ distinct paths to any fixed vertex. The difference between 4 and 5 is similar, but reasoning about ℓ-paths instead of ℓ'-paths.

Note that the differences at some of these steps are statistically significant. We only argue that they are not computationally detectable, at least when the endomorphism rings are not known.

5.3 On hardness of the supersingular endomorphism problem with access to an auxiliary oracle

The FastSQIsignHD identification protocol is sound assuming the hardness of the supersingular endomorphism problem 18, and zero-knowledge with respect to a simulator that has access to a RUGDIO (or a RADIO for RigorousSQIsignHD, as defined in Appendix D.2). For the resulting signature scheme to be secure, one therefore needs to assume that the supersingular endomorphism problem remains hard even when given access to a RUGDIO.

While it currently seems out of reach to prove that the supersingular endomorphism problem is equivalent to the variant with RUGDIO access, let us argue that the RUGDIO indeed does not help. We focus the following discussion on the RUGDIO, but the same arguments apply to the RADIO despite the slightly different distribution.

The RUGDIO allows to generate random isogenies with a chosen domain E. Note that this task is already known to be easy, with isogenies of smooth degree. The RUGDIO only lifts this smoothness restriction and replaces it with other restrictions (ℓ^e-good and prime to ℓ'): it allows to generate random isogenies whose degrees have large prime factors. It does not allow to reach more target curves, nor does it give more control on which specific target to hit: the target curve is uniformly distributed in the supersingular graph, which was already possible with smooth isogenies.

Smoothness of random isogenies has never been an inconvenience in finding endomorphisms. In fact, the current fastest algorithms for this problem only require very smooth degree isogenies, typically a power of 2. The reason is the following: the purpose of constructing a random isogeny from a fixed source is to reach a random target. As very smooth isogenies (even 2-smooth) are sufficient for optimal randomisation, there is no incentive to involve much larger prime factors. More specifically, the best known strategies to solve the supersingular endomorphism problem [DG16] have classical time complexity $\widetilde{O}(\sqrt{p})$ (and quantum time complexity $\widetilde{O}(p^{1/4})$ with a Grover argument [Gro96]) and essentially perform a meet-in-the-middle search in the supersingular isogeny graph. Access to a RUGDIO would allow to use isogenies of a different shape in the search, but would not speed it up, as the probability to find isogenies with matching codomains stays the same. Another illustration that having access to non-smooth degree isogenies does not help is the fact that the discovery of the $\sqrt{\text{élu}}$ algorithm [BDLS20] (which dramatically improved the complexity of computing prime degree isogenies) did not affect the state-of-the-art of the supersingular endomorphism problem.

The above arguments support that random isogenies of non-smooth degrees are not more helpful than random isogenies of smooth degrees. Now, one may be concerned that the encoding of the output of the RUGDIO may leak more information than it should. Non-smooth degree isogenies are represented as a component of a higher dimensional isogeny (Section 2.2). This representation is universal, in the sense that any efficient representation of an isogeny can be efficiently rewritten in this form. In particular, this encoding contains no more information than any other efficient representation of the same isogeny.

6 The SQIsignHD digital signature scheme

The SQIsignHD identification protocol that we presented yields a digital signature scheme via the Fiat-Shamir transform. The security of the transform of both versions FastSQIsignHD and RigorousSQIsignHD follows from the analysis conducted in Section 5 and Appendix D, so the digital signature is also secure under the same computational assumptions. Namely, we have seen it is universally unforgeable under chosen message attacks in the random oracle and RADIO or RUGDIO model, assuming the hardness of the endomorphism ring problem. In this section, we present the performance of the signature scheme obtained from FastSQIsignHD.

6.1 Compactness

As explained before, the signature is made of the data $(E_1, q, \sigma(P_1), \sigma(P_2))$, with $q < \ell^e$, $\sigma : E_1 \longrightarrow E_2$ a q-isogeny and (P_1, P_2) a basis of $E_1[\ell^f]$ determined canonically.

E_1 can be entirely determined by its j-invariant $j(E_1) \in \mathbb{F}_{p^2}$. Since any element of $\mathbb{F}_{p^2}$ can be represented by 2 integers in $[\![0\ ;\ p-1]\!]$, storing $j(E_1)$ takes approximately $2\log_2(p) \simeq 4\lambda$ bits, given that $p = \Theta(2^{2\lambda})$ (where λ is the security level). Similarly, $q < \ell^e \simeq \sqrt{p}$, so q is an integer of $1/2\log_2(p) \simeq \lambda$ bits.

The points $\sigma(P_1)$ and $\sigma(P_2)$ need not be represented explicitely with coordinates in $\mathbb{F}_{p^2}$. They can be compressed. Indeed, if we generate a canonical basis (Q_1, Q_2) of $E_2[\ell^f]$, then we may write $\sigma(P_1) = a_1Q_1 + b_1Q_2$ and $\sigma(P_2) = a_2Q_1 + b_2Q_2$ with $a_1, b_1, a_2, b_2 \in \mathbb{Z}/\ell^f\mathbb{Z}$. Storing the scalars a_1, b_1, a_2, b_2 requires $4f$ bits (assuming $\ell = 2$, which will be the case in practice).

Actually, we can gain f bits by omitting one of the scalars a_1, b_1, a_2, b_2 if we use the Weil pairing. Indeed, we have

$$\begin{aligned} e_{\ell^f}(\sigma(P_1), \sigma(P_2)) =& e_{\ell^f}(P_1, P_2)^q. \\ e_{\ell^f}(\sigma(P_1), \sigma(P_2)) =& e_{\ell^f}(a_1Q_1 + b_1Q_2, a_2Q_1 + b_2Q_2) = e_{\ell^f}(Q_1, Q_2)^{a_1b_2 - b_1a_2}. \end{aligned}$$

Since (P_1, P_2) and (Q_1, Q_2) are basis of $E_1[\ell^f]$ and $E_2[\ell^f]$ respectively, $e_{\ell^f}(P_1, P_2)$ and $e_{\ell^f}(Q_1, Q_2)$ are both primitive ℓ^f-th root of unity. Hence, we may find $k \in (\mathbb{Z}/\ell^f\mathbb{Z})^\times$ such that $e_{\ell^f}(P_1, P_2) = e_{\ell^f}(Q_1, Q_2)^k$, and we must have

$$a_1b_2 - b_1a_2 \equiv kq \mod \ell^f \qquad (2)$$

Remark 6.1. Since $\ell^f | p-1$, the ℓ^f-th Weil pairing takes values in $\mathbb{F}_p^*$, so we find k easily by solving a discrete logarithm problem in a subgroup of order ℓ^f of $\mathbb{F}_p^*$ by Pohlig-Hellman [PH78] techniques (which apply since $p-1$ is smooth).

Since q is prime to ℓ, $\sigma(P_1)$ have order ℓ^f so either a_1 or b_1 is invertible modulo ℓ^f. If a_1 is invertible, we can recover b_2 from the other scalars using equation 2 and we can recover a_2 otherwise. Hence we only need 3 scalars among 4.

We can make the representation of $\sigma(P_1)$ and $\sigma(P_2)$ even more compact. Indeed, by Remark 4.2 the ℓ^e-isogeny F representing σ can be computed as long as $2f \geq e+4$. But in FastSQIsignHD, $f \simeq e \simeq \lambda$ so we may use points of ℓ^{f_1}-torsion with $f_1 := \lceil e/2 \rceil + 3$ instead of points of ℓ^f-torsion. This reduces the storage cost of $\sigma(P_1)$ and $\sigma(P_2)$ from $3f \simeq 3\lambda$ to $3f_1 \simeq 3/2\lambda$.

On the whole, we can represent the signatures with $s = 13/2\lambda + O(\log(\lambda))$ bits if we use the compression and decompression algorithms given by Algorithms 6 and 7, breaking the previous record of SQIsign. Indeed, in SQIsign, the kernels of the signature isogeny σ (of degree $p^{15/4}$) and of the dual of the challenge (of degree $\sqrt{p}$) need to be transmitted so we get a signature of size $s = 17/2\lambda + O(\log(\lambda))$ at least.

Example 22. For NIST-I security level ($\lambda = 128$ bits), we can choose the parameters $p = 13 \cdot 2^{126} 3^{78} - 1$, $e = 142$ and $f_1 = 73$. The total signature size in SQIsignHD is $2\lceil \log_2(p) \rceil + e + 3f_1 + 1 = 870$ bits or 109 bytes. For the same security level, SQIsign signatures took 177 bytes in the NIST implementation [The23b].

Remark 23. We can use the same techniques in dimension 8 but we output signatures of size $s = 14\lambda + O(\log(\lambda))$ instead of $13/2\lambda + O(\log(\lambda))$ since e is bigger ($\ell^e = \Theta(p^2)$). Details may be found in Appendix C.3.

6.2 Time efficiency

Low signing time (and key generation time). In the latest version of SQIsign [FLLW23], the signature time was dominated by the computation of 30 T-isogenies with $T \simeq p^{5/4}$. Each T-isogeny is rather slow as T typically has prime factors as large as a few thousands. In contrast, in FastSQIsignHD, signing only requires a handful of ℓ^f and $\ell'^{f'}$-isogenies, where typically $\ell = 2$ and $\ell' = 3$. Computing these isogenies is several orders of magnitude faster than a SQIsign [FLLW23] signature.

We have implemented FastSQIsignHD in C, based on the implementation of SQIsign [The23b]. The signature takes an average time of 28 ms at the 128 bits security level on an Intel(R) Core(TM) i5-1335U 4600MHz CPU (average over 1000 signature computations). Key generation takes 70 ms on average on the same CPU. While signature is already close to ten times faster than the fastest implementations of SQIsign [FLLW23; LWXZ23], we refrain from reporting a detailed clock-cycle comparison, as the bottleneck of our implementation has shifted from isogeny computations to a variety of steps which have not been

```
Algorithm 6: Compression
  Data: $E_1, E_2, q, P_1, P_2, \sigma(P_1), \sigma(P_2)$, where $q < \ell^e$, $\sigma : E_1 \longrightarrow E_2$ a $q$-isogeny
        and $(P_1, P_2)$ a basis of $E_1[\ell^{f_1}]$ determined canonically (with $f_1 := \lceil e/2 \rceil + 3$).
  Result: A word of length $2\lceil \log_2(p) \rceil + e + 3f_1$ bits (assuming $\ell = 2$).
  Compute $j(E_1) \in \mathbb{F}_{p^2}$;
  Let $\zeta$ be a canonical generator of $\mathbb{F}_{p^2}$. Write $\zeta := n_1 + n_2\zeta$ where $n_1, n_2 \in \mathbb{F}_p$
    are represented by integers in $[\![0 \ ; \ p-1]\!]$ of length $\lceil \log_2(p) \rceil$ bits each;
  Compute the canonical basis $(Q_1, Q_2)$ of $E_2[\ell^{f_1}]$;
  Find $k \in (\mathbb{Z}/\ell^{f_1}\mathbb{Z})^\times$ such that $e_{\ell^{f_1}}(P_1, P_2) = e_{\ell^{f_1}}(Q_1, Q_2)^k$;
  Find $a_1, b_1, a_2, b_2 \in \mathbb{Z}/\ell^{f_1}\mathbb{Z}$ such that $\sigma(P_1) = a_1 Q_1 + b_1 Q_2$ and
    $\sigma(P_2) = a_2 Q_1 + b_2 Q_2$;
  if $\ell \nmid a_1$ then
      Return $\|n_1\|n_2\|q\|a_1\|b_1\|b_2\|$;
  else
      Return $\|n_1\|n_2\|q\|a_1\|b_1\|a_2\|$;
  end
```

optimised (as they were negligible in former SQIsign implementations). Most notably, about 29% of the FastSQIsignHD signing time is spent computing two discrete logarithms; a 4.8× speedup is reported in [LWXZ23] using Tate pairings for comparable discrete logarithm computations. Another 20% is spent solving a quaternion norm equation [Ler22, Algorithm 4], a step that has not been the focus of much attention. In addition, contrary to former implementations of SQIsign, our implementation is purely in C, with no assembly-optimised field arithmetic.

Providing a completely optimized implementation is left for future works, yet this first implementation is already compelling.

Verification time. This efficiency gain in the signature is made at the expense of the verification time where a 4-dimensional ℓ^e-isogeny has to be computed. Of course ℓ-isogenies in dimension 4 are expected to be slower to compute than in dimension 1. Nonetheless, we only have to compute two chains of ℓ-isogenies of length $1/4 \log_\ell(p)$, whereas the verifier had to compute an ℓ-isogeny chain of size $15/4 \log_\ell(p)$ in the last version of SQIsign [FLLW23]. Furthermore, our choice of parameters allows for more efficient field arithmetic.

Our experimental `sagemath` implementation provides an upper bound on the verification time: for 128 bits of security the verification takes around 600 ms on the same CPU as above. We expect this time to be significantly reduced in the future: new optimisations remain to be implemented, and more importantly a low level implementation should lead to a significant gain. Currently, the time spent on verification is as follows: around 60 ms for the challenge computation[9],

---

[9] Which essentially consists in computing a chain of 3-isogenies in dimension 1. The signature also needs to compute similar isogenies, and in this case the low level C

Algorithm 7: Decompression

```
Data: A word $w$ of length $2\lceil\log_2(p)\rceil + e + 3f_1$ bits ($\ell := 2, f_1 := \lceil e/2\rceil + 3$),
      the public key $E_A$ and the message $m$, hash functions $\Phi$ and $H$ used to
      generate the challenge after Fiat-Shamir transform (see Appendix A.1).
Result: $E_1, E_2, q, P_1, P_2, \sigma(P_1), \sigma(P_2)$, where $q < \ell^e$, $\sigma : E_1 \longrightarrow E_2$ is a $q$-
        isogeny and $(P_1, P_2)$ is a basis of $E_1[\ell^{f_1}]$ determined canonically.
Parse $\|n_1\|n_2\|q\|a_1\|b_1\|c_2\| \longleftarrow w$;
Set $j \longleftarrow n_1 + n_2\zeta$, where $\zeta$ is the canonical generator of $\mathbb{F}_{p^2}$;
Compute $E_1$ of $j$-invariant $j(E_1) = j$;
Recover the commitment $\varphi \longleftarrow \Phi(E_A, H(E_1, m))$. Let $E_2$ be the codomain of
  $\varphi$;
Compute the canonical basis $(P_1, P_2)$ of $E_1[\ell^{f_1}]$ and the canonical basis
  $(Q_1, Q_2)$ of $E_2[\ell^{f_1}]$;
Find $k \in (\mathbb{Z}/\ell^{f_1}\mathbb{Z})^\times$ such that $e_{\ell^{f_1}}(P_1, P_2) = e_{\ell^{f_1}}(Q_1, Q_2)^k$;
if $\ell \nmid a_1$ then
    $a_2 \longleftarrow c_2$;
    Find $b_2 \in \mathbb{Z}/\ell^{f_1}\mathbb{Z}$ such that $a_1 b_2 - b_1 a_2 \equiv kq \mod \ell^{f_1}$;
else
    $b_2 \longleftarrow c_2$;
    Find $a_2 \in \mathbb{Z}/\ell^{f_1}\mathbb{Z}$ such that $a_1 b_2 - b_1 a_2 \equiv kq \mod \ell^{f_1}$;
end
Return $E_1, E_2, q, P_1, P_2, a_1Q_1 + b_1Q_2, a_2Q_1 + b_2Q_2$;
```

510 ms for the two dimension 4 2^{71}-isogenies giving F, and 30 ms for the image of a point through F.

---

implementation only takes 1-2 ms, which shows the potential of improvements of writing a low level implementation of the verification.

References

[Ahr23] K. Ahrens. *Sieving for large twin smooth integers using single solutions to Prouhet-Tarry-Escott.* Cryptology ePrint Archive, Paper 2023/219. 2023. URL: https://eprint.iacr.org/2023/219.

[AJKKL16] R. Azarderakhsh, D. Jao, K. Kalach, B. Koziel, and C. Leonardi. "Key compression for isogeny-based cryptosystems". In: *Proceedings of the 3rd ACM International Workshop on ASIA Public-Key Cryptography.* Xi'an, China: ACM, 2016, pp. 1–10.

[BCC+23] A. Basso, G. Codogni, D. Connolly, L. De Feo, T. B. Fouotsa, G. M. Lido, T. Morrison, L. Panny, S. Patranabis, and B. Wesolowski. "Supersingular Curves You Can Trust". In: *Advances in Cryptology – EUROCRYPT 2023.* Ed. by C. Hazay and M. Stam. Cham: Springer Nature Switzerland, 2023, pp. 405–437. ISBN: 978-3-031-30617-4.

[BDLS20] D. J. Bernstein, L. De Feo, A. Leroux, and B. Smith. "Faster computation of isogenies of large prime degree". In: *Open Book Series, Proceedings of the Fourteenth Algorithmic Number Theory Symposium – ANTS XIV* 4.1 (2020), pp. 39–55.

[BST+17] M. Bhargava, A. Shankar, T. Taniguchi, F. Thorne, J. Tsimerman, and Y. Zhao. "Bounds on 2-torsion in class groups of number fields and integral points on elliptic curves". In: *Journal of the American Mathematical Society* 33 (Jan. 2017). DOI: 10.1090/jams/945.

[BL04] C. Birkenhake and H. Lange. *Complex abelian varieties.* Vol. 302. Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]. Berlin: Springer-Verlag, 2004, pp. xii+635. ISBN: 3-540-20488-1.

[BSC+22] G. Bruno, M. C.-R. Santos, C. Costello, J. K. Eriksen, M. Naehrig, M. Meyer, and B. Sterner. *Cryptographic Smooth Neighbors.* Cryptology ePrint Archive, Paper 2022/1439. 2022. URL: https://eprint.iacr.org/2022/1439.

[CD23] W. Castryck and T. Decru. "An Efficient Key Recovery Attack On SIDH". In: *Advances in Cryptology – EUROCRYPT 2023: 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part V.* Lyon, France: Springer-Verlag, 2023, pp. 423–447. ISBN: 978-3-031-30588-7. URL: https://doi.org/10.1007/978-3-031-30589-4_15.

[CLG09] D. X. Charles, K. E. Lauter, and E. Z. Goren. "Cryptographic Hash Functions from Expander Graphs". In: *Journal of Cryptology* 22.1 (2009), pp. 93–113. DOI: 10.1007/s00145-007-9002-x.

[Cor08] G. Cornacchia. "Su di un metodo per la risoluzione in numeri interi dell'equazione $\sum_{h=0}^{n} C_h x^{n-h} y^h = P$". In: *Giornale di matematiche di Battaglini* 46 (1908), pp. 33–90.

[CR15] R. Cosset and D. Robert. “An algorithm for computing (ℓ,ℓ)-isogenies in polynomial time on Jacobians of hyperelliptic curves of genus 2”. In: *Mathematics of Computation* 84.294 (Nov. 2015), pp. 1953–1975. DOI: `10.1090/S0025-5718-2014-02899-8`.

[CMN21] C. Costello, M. Meyer, and M. Naehrig. “Sieving for Twin Smooth Integers with Solutions to the Prouhet-Tarry-Escott Problem”. In: *Advances in Cryptology – EUROCRYPT 2021*. Ed. by A. Canteaut and F.-X. Standaert. Cham: Springer International Publishing, 2021, pp. 272–301. ISBN: 978-3-030-77870-5.

[Cou06] J.-M. Couveignes. *Hard Homogeneous Spaces*. Cryptology ePrint Archive, Report 2006/291. 2006. URL: `https://eprint.iacr.org/2006/291`.

[DDF+21] L. De Feo, C. Delpech de Saint Guilhem, T. B. Fouotsa, P. Kutas, A. Leroux, C. Petit, J. Silva, and B. Wesolowski. “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*. Singapore, Singapore: Springer-Verlag, 2021, pp. 249–278. ISBN: 978-3-030-92067-8. DOI: `10.1007/978-3-030-92068-5_9`.

[DKLPW20] L. De Feo, D. Kohel, A. Leroux, C. Petit, and B. Wesolowski. “SQIsign: Compact Post-quantum Signatures from Quaternions and Isogenies”. In: *Advances in Cryptology – ASIACRYPT 2020*. Ed. by S. Moriai and H. Wang. Cham: Springer International Publishing, 2020, pp. 64–93. ISBN: 978-3-030-64837-4.

[DG16] C. Delfs and S. D. Galbraith. “Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$”. In: *Designs, Codes and Cryptography* 78.2 (2016), pp. 425–440. DOI: `10.1007/s10623-014-0010-1`.

[Deu41] M. Deuring. “Die Typen der Multiplikatorenringe elliptischer Funktionenkörper”. In: *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg* 14 (1941), pp. 197–272.

[EHLMP18] K. Eisenträger, S. Hallgren, K. Lauter, T. Morrison, and C. Petit. “Supersingular Isogeny Graphs and Endomorphism Rings: Reductions and Solutions”. In: *Advances in Cryptology – EUROCRYPT 2018*. Ed. by J. B. Nielsen and V. Rijmen. Cham: Springer International Publishing, 2018, pp. 329–368. ISBN: 978-3-319-78372-7.

[FLLW23] L. D. Feo, A. Leroux, P. Longa, and B. Wesolowski. “New Algorithms for the Deuring Correspondence - Towards Practical and Secure SQISign Signatures”. In: *Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part V*. Ed. by C. Hazay and M. Stam. Vol. 14008. Lecture Notes in Computer Science.

Springer, 2023, pp. 659–690. URL: https://doi.org/10.1007/978-3-031-30589-4%5C_23.

[FS87] A. Fiat and A. Shamir. "How To Prove Yourself: Practical Solutions to Identification and Signature Problems". In: *Advances in Cryptology — CRYPTO' 86*. Ed. by A. M. Odlyzko. Berlin, Heidelberg: Springer Berlin Heidelberg, 1987, pp. 186–194. ISBN: 978-3-540-47721-1.

[FKMT22] T. B. Fouotsa, P. Kutas, S.-P. Merz, and Y. B. Ti. "On the isogeny problem with torsion point information". In: *PKC*. Springer. 2022, pp. 142–161.

[GPS20] S. D. Galbraith, C. Petit, and J. Silva. "Identification Protocols and Signature Schemes Based on Supersingular Isogeny Problems". In: *J. Cryptol.* 33.1 (Jan. 2020), pp. 130–175. ISSN: 0933-2790. URL: https://doi.org/10.1007/s00145-019-09316-0.

[Gro96] L. K. Grover. "A Fast Quantum Mechanical Algorithm for Database Search". In: *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*. STOC '96. Philadelphia, Pennsylvania, USA: Association for Computing Machinery, 1996, pp. 212–219. ISBN: 0897917855. DOI: 10.1145/237814.237866.

[HW08] G. H. Hardy and E. M. Wright. *An introduction to the theory of numbers*. 6th. Oxford University Press, 1938 (2008), p. 622. ISBN: 978-0-19-921986-5.

[HL10] C. Hazay and Y. Lindell. *Efficient Secure Two-Party Protocols: Techniques and Constructions*. 1st. Berlin, Heidelberg: Springer-Verlag, 2010. ISBN: 3642143024.

[Igu72] J.-i. Igusa. *Theta functions*. Die Grundlehren der mathematischen Wissenschaften, Band 194. New York: Springer-Verlag, 1972, pp. x+232.

[JD11] D. Jao and L. De Feo. "Towards Quantum-Resistant Cryptosystems from Supersingular Elliptic Curve Isogenies". In: *Post-Quantum Cryptography*. Ed. by B.-Y. Yang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 19–34. ISBN: 978-3-642-25405-5.

[Kan97] E. Kani. "The number of curves of genus two with elliptic differentials". In: *Journal für die reine und angewandte Mathematik* 1997.485 (1997), pp. 93–122. DOI: 10.1515/crll.1997.485.93.

[Kan14] E. Kani. "The moduli spaces of Jacobians isomorphic to a product of two elliptic curves". In: *Journal of Number Theory* 139 (June 2014), pp. 138–174. DOI: 10.1016/j.jnt.2013.12.006.

[KLPT14] D. Kohel, K. Lauter, C. Petit, and J.-P. Tignol. "On the quaternion -isogeny path problem". In: *LMS Journal of Computation and Mathematics* 17 (June 2014). DOI: 10.1112/S1461157014000151.

[Lag70] J. L. de Lagrange. “Démonstration d’un théoreme d’arithmétique”. In: *Nouveau Mémoire de l’Académie Royale des Sciences de Berlin* (1770), pp. 123–133.

[Ler22] A. Leroux. *Quaternion algebras and isogeny-based cryptography.* 2022. URL: http://www.lix.polytechnique.fr/Labo/Antonin.LEROUX/manuscrit_these.pdf.

[Ler23] A. Leroux. “A new isogeny representation and applications to cryptography”. In: *Advances in Cryptology–ASIACRYPT 2022: 28th International Conference on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, December 5–9, 2022, Proceedings, Part II.* Springer. 2023, pp. 3–35.

[LWXZ23] K. Lin, W. Wang, Z. Xu, and C.-A. Zhao. *A Faster Software Implementation of SQISign.* Cryptology ePrint Archive, Paper 2023/753. 2023. URL: https://eprint.iacr.org/2023/753.

[Lon23] P. Longa. “Efficient Algorithms for Large Prime Characteristic Fields and Their Application to Bilinear Pairings”. In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2023.3 (June 2023), pp. 445–472. DOI: 10.46586/tches.v2023.i3.445-472. URL: https://tches.iacr.org/index.php/TCHES/article/view/10970.

[LR12] D. Lubicz and D. Robert. “Computing isogenies between abelian varieties”. In: *Compositio Mathematica* 148.5 (Sept. 2012), pp. 1483–1515. DOI: 10.1112/S0010437X12000243.

[LR15] D. Lubicz and D. Robert. “Computing separable isogenies in quasi-optimal time”. In: *LMS Journal of Computation and Mathematics* 18.1 (2015), pp. 198–216. DOI: 10.1112/S146115701400045X.

[LR23] D. Lubicz and D. Robert. “Fast change of level and applications to isogenies”. In: vol. 9. 1. Springer, 2023. DOI: 10.1007/s40993-022-00407-9.

[MMPPW23] L. Maino, C. Martindale, L. Panny, G. Pope, and B. Wesolowski. “A Direct Key Recovery Attack on SIDH”. In: Springer-Verlag, 2023.

[Mic16] D. Micciancio. *Minkowski’s theorem.* In *CSE 206A: Lattice Algorithms and Applications.* 2016. URL: https://cseweb.ucsd.edu/classes/wi16/cse206A-a/lec2.pdf.

[Mil86] J. S. Milne. “Abelian Varieties”. In: *Arithmetic Geometry.* New York, NY: Springer New York, 1986, pp. 103–150. ISBN: 978-1-4613-8655-1. DOI: 10.1007/978-1-4613-8655-1_5.

[Mum66a] D. Mumford. “On the equations defining abelian varieties 1”. In: *Inventiones mathematicae* 1.4 (1966), pp. 287–354. DOI: 10.1007/BF01389737.

[Mum66b] D. Mumford. “On the equations defining abelian varieties. I”. In: *Invent. Math.* 1 (1966), pp. 287–354.

[Mum67a] D. Mumford. "On the equations defining abelian varieties. II". In: *Invent. Math.* 3 (1967), pp. 75–135.

[Mum67b] D. Mumford. "On the equations defining abelian varieties. III". In: *Invent. Math.* 3 (1967), pp. 215–244.

[Mum74] D. Mumford. *Abelian varieties.* Second Edition. Tata Institute of fundamental research studies in mathematics. London: Oxford University Press, 1974, pp. x+279.

[Mum84] D. Mumford. *Tata lectures on theta II.* Vol. 43. Progress in Mathematics. Jacobian theta functions and differential equations, With the collaboration of C. Musili, M. Nori, E. Previato, M. Stillman and H. Umemura. Boston, MA: Birkhäuser Boston Inc., 1984, pp. xiv+272. ISBN: 0-8176-3110-0.

[22] *PARI/GP version* `2.13.4`. available from `http://pari.math.u-bordeaux.fr/`. The PARI Group. Univ. Bordeaux, 2022.

[Piz80] A. Pizer. "An algorithm for computing modular forms on $\Gamma_0(N)$". In: *Journal of Algebra* 64 (June 1980), pp. 340–340. DOI: `0.1016/0021-8693(80)90151-9`.

[PH78] S. C. Pohlig and M. E. Hellman. "An improved algorithm for computing logarithms over GF(p) and its cryptographic significance". In: *IEEE Transactions on information theory* 24.1 (Jan. 1978), pp. 106–110.

[PT18] P. Pollack and E. Treviño. "Finding the Four Squares in Lagrange's Theorem". In: *Integers* 18A (2018), A15.

[Rab86] J. O. Rabin M. O.; Shallit. "Randomized Algorithms in Number Theory". In: *Communications on Pure and Applied Mathematic* 39.S1 (1986), S239–S256. DOI: `10.1002/cpa.3160390713`.

[Rob10] D. Robert. "Theta functions and cryptographic applications". PhD thesis. Université Henri-Poincarré, Nancy 1, France, July 2010. URL: `http://www.normalesup.org/~robert/pro/publications/academic/phd.pdf`.

[Rob21] D. Robert. "Efficient algorithms for abelian varieties and their moduli spaces". HDR thesis. Université Bordeaux, June 2021. URL: `http://www.normalesup.org/~robert/pro/publications/academic/hdr.pdf`.

[Rob22] D. Robert. *Evaluating isogenies in polylogarithmic time.* Cryptology ePrint Archive, Paper 2022/1068. 2022. URL: `https://eprint.iacr.org/2022/1068`.

[Rob23a] D. Robert. "A note on optimising 2^n-isogenies in higher dimension". June 2023. In preparation.

[Rob23b] D. Robert. "Breaking SIDH in Polynomial Time". In: *Advances in Cryptology – EUROCRYPT 2023.* Ed. by C. Hazay and M. Stam. Cham: Springer Nature Switzerland, 2023, pp. 472–503. ISBN: 978-3-031-30589-4.

[RS06] A. Rostovtsev and A. Stolbunov. *Public-Key Cryptosystem Based On Isogenies.* Cryptology ePrint Archive, Report 2006/145. 2006. URL: https://eprint.iacr.org/2006/145.

[RT22] J. Rouse and K. Thompson. *Quaternary quadratic forms with prime discriminant.* 2022. arXiv: 2206.00412 [math.NT].

[The23a] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 10.0).* 2023. URL: https://www.sagemath.org.

[Sil09] J. H. Silverman. *The Arithmetic of Elliptic Curves.* Springer, 2009, p. 522.

[The23b] The SQIsign team. *SQIsign.* 2023. URL: https://www.sqisign.org.

[Tes99] E. Teske. "The Pohlig-Hellman Method Generalized for Group Structure Computation". In: *Journal of Symbolic Computation* 11 (1999), pp. 1–14.

[Vél71] J. Vélu. "Isogénies entre courbes elliptiques". In: *Comptes-rendus de l'Académie des Sciences* 273 (July 1971). Available at https://gallica.bnf.fr, pp. 238–241.

[VV15] D. Venturi and A. Villani. *Zero-Knowledge Proofs and Applications.* May 2015. URL: http://danieleventuri.altervista.org/files/zero-knowledge.pdf.

[Voi20] J. Voight. *Quaternion algebras.* v.0.9.23. Aug. 2020. URL: https://math.dartmouth.edu/~jvoight/quat.html.

[Wes22] B. Wesolowski. "The supersingular isogeny path and endomorphism ring problems are equivalent". In: *FOCS 2021 - 62nd Annual IEEE Symposium on Foundations of Computer Science.* Denver, Colorado, United States, Feb. 2022. URL: https://hal.archives-ouvertes.fr/hal-03340899.

[ZSPDB18] G. Zanon, M. A. Simplicio, G. Pereira, J. Doliskani, and P. L. Barreto. "Faster isogeny-based compressed key agreement". In: *Post-Quantum Cryptography.* Springer International Publishing, 2018, pp. 248–268.

A Preliminaries

A.1 Constructing a signature with the Fiat-Shamir transform

In this section, we explain how to transform our SQIsignHD identification protocol into a signature scheme using the Fiat-Shamir transform [FS87]. The method is analogous to the original SQIsign protocol. The following procedure applies to both FastSQIsignHD and RigorousSQIsignHD.

Decomposing the degree of the challenge into primes $D_\varphi := \prod_{i=1}^r \ell_i^{e_i}$ and setting $\mu(D_\varphi) := \prod_{i=1}^r \ell_i^{e_i-1}(\ell_i + 1)$, we define a secure hash function in the supersingular $\{\ell_1, \cdots, \ell_r\}$-isogeny graph mapping a supersingular elliptic curve E and an integer $s \in [\![1 \; ; \; \mu(D_\varphi)]\!]$ to a cyclic D_φ-isogeny $\Phi(E, s)$ with domain E. Such a hash function has been constructed in [DDF+21, § 3.1], which is a generalization of [CLG09]. We also use another secure hash function $H : \{0,1\}^* \longrightarrow [\![1 \; ; \; \mu(D_\varphi)]\!]$.

Signature. To sign a message m with a secret key $\tau : E_0 \longrightarrow E_A$, generate a random commitment $\psi : E_0 \longrightarrow E_1$, let $s := H(j(E_1), m)$ and $\varphi := \Phi(E_A, s) : E_A \longrightarrow E_2$. From the knowledge of τ, φ and ψ, construct an efficient representation $R = (\sigma(P_1), \sigma(P_2), q)$ given by the image of torsion points by a response isogeny $\sigma : E_1 \longrightarrow E_2$ and return (E_1, R) as a signature.

Verification. A verifier receiving a signature (E_1, R) associated to the message m and public key E_A computes $s = H(j(E_1), m)$ and then $\varphi = \Phi(E_A, s) : E_A \longrightarrow E_2$. The verifier finally checks that R represents correctly an isogeny $\sigma : E_1 \longrightarrow E_2$ by computing a higher dimensional isogeny, as explained previously.

Once it is established that the SQIsignHD identification protocol is complete, sound, and honest verifier zero-knowledge, and assuming the hardness of the endomorphism ring problem, we obtain a universal unforgeable signature against chosen message attacks in the random oracle model [VV15, Theorem 7].

A.2 Abelian varieties and their isogenies

An abelian variety A over a field k is a connected projective k-variety with an algebraic group law (which is then automatically abelian by rigidity). Abelian varieties are generalizations of elliptic curves in any dimension. In particular, elliptic curves are abelian varieties of dimension 1 and products of elliptic curves are abelian varieties.

A morphism of abelian varieties is an algebraic map $\varphi : A \to B$ which is a group homomorphism; by rigidity it suffices to check that $\varphi(0_A) = 0_B$. It is an *isogeny* if it is surjective and has finite kernel. The degree of an isogeny is its degree as a rational map. If φ is a *separable* isogeny, then $\deg(\varphi) = \# \ker(\varphi)$. If $\deg(\varphi)$ is prime to the characteristic p of the base field k, then φ is automatically separable. As in elliptic curves, the multiplication by n map in an abelian variety $[n]_A : A \longrightarrow A$ is an isogeny of degree n^{2g} with $g := \dim(A)$ and its kernel $A[n]$ is isomorphic to $(\mathbb{Z}/n\mathbb{Z})^{2g}$ when n is prime to p.

To any abelian variety A, we associate its *dual abelian variety* $\widehat{A}$, which has the same dimension. The dual defines a contravariant functor: any isogeny $\varphi : A \longrightarrow B$ induces a *dual isogeny* $\widehat{\varphi} : \widehat{B} \longrightarrow \widehat{A}$ which has the same degree. A *polarization* is an isogeny $\lambda : A \longrightarrow \widehat{A}$ induced by an ample line bundle. A polarized abelian variety (A, λ) is *principally polarized* if λ is an isomorphism.

If $n \in \mathbb{N}^*$ is prime to p, then we have a non-degenerate pairing $e_n : A[n] \times \widehat{A}[n] \longrightarrow \overline{k}^*$ called the *Weil pairing*. Given a polarization $\lambda : A \longrightarrow \widehat{A}$, the Weil pairing yields a non-degenerate antisymmetric pairing $e_n^\lambda : A[n] \times A[n] \longrightarrow \overline{k}^*$.

Morally, a polarization can be seen as "a way to represent an abelian variety". Indeed, in plain generality, we do not have nice analogues for the Weierstrass model for elliptic curves, but every abelian variety can be described by a theta model [Mum66b; Mum67a; Mum67b]. We refer to the notes of Milne [Mil86] or the book of Mumford [Mum74] for a complete introduction to abelian varieties.

A.3 The Deuring correspondence

Quaternions, orders and ideals. Let $\mathcal{B}_{p,\infty}$ be the quaternion algebra over $\mathbb{Q}$ ramifying at p and ∞. By [Piz80, Proposition 5.1], there exists a $\mathbb{Q}$-basis $(1, i, j, k)$ of $\mathcal{B}_{p,\infty}$ with $j^2 = -p$, $k = ij = -ji$ and $i^2 = -1, -2, -q$, when $p \equiv 3 \mod 4$, $p \equiv 5 \mod 8$ and $p \equiv 1 \mod 8$ respectively, q being a prime such that $(-p/q) = 1$. $\mathcal{B}_{p,\infty}$ has a *conjugation* $\alpha := x+iy+jz+kt \longmapsto \overline{\alpha} := x-iy-jz-kt$. For all $\alpha \in \mathcal{B}_{p,\infty}$, we define the *reduced norm* $\operatorname{nrd}(\alpha) := \alpha\overline{\alpha}$ and *trace* $\operatorname{Tr}(\alpha) := \alpha + \overline{\alpha}$.

A *fractional ideal* $I \subset \mathcal{B}_{p,\infty}$ is a $\mathbb{Z}$-lattice of rank 4. We also define the *reduced norm* of I as $\operatorname{nrd}(I) := \gcd\{\operatorname{nrd}(\alpha) \mid \alpha \in I\}$ and the conjugation $\overline{I} := \{\overline{\alpha} \mid \alpha \in I\}$. If $I \subset J$ are two fractional ideals, then $[J : I] = \operatorname{nrd}(I)^2/\operatorname{nrd}(J)^2$. If $(\alpha_1, \cdots, \alpha_4)$ is a basis of I, then $|\det(\operatorname{Tr}(\alpha_i\overline{\alpha_j}))_{1\leq i,j\leq 4}|^{1/2}$ does not depend on the basis. This invariant is called the the *reduced discriminant* of I and denoted by $\operatorname{discrd}(I)$.

An *order* of $\mathcal{O} \subset \mathcal{B}_{p,\infty}$ is a fractional ideal which is stable by multiplication and contains 1. We say it is *maximal* if it is maximal for the inclusion. If I is a fractional ideal, we define its *left order* $O_L(I) := \{\alpha \in \mathcal{B}_{p,\infty} \mid \alpha I \subseteq I\}$ and its *right order* $O_R(I) := \{\alpha \in \mathcal{B}_{p,\infty} \mid I\alpha \subseteq I\}$. We say that I is a left (respectively right) $\mathcal{O}$-ideal when $\mathcal{O} \subseteq O_L(I)$ (respectively $\mathcal{O} \subseteq O_R(I)$). We also say that I *connects* $\mathcal{O}$ and $\mathcal{O}'$ when $\mathcal{O} = O_L(I)$ and $\mathcal{O}' = O_R(I)$. I is *integral* if $I \subseteq O_L(I)$. In this case, $I \subseteq O_R(I)$ and both $O_L(I)$ and $O_R(I)$ are maximal. **In this paper, we only consider integral ideals and simply refer to them as ideals.** Two fractional ideals $I \sim J$ are equivalent if there exists $\beta \in B_{p,\infty}^*$ such that $J = I\beta$. In that case, $O_L(I) = O_L(J)$ and $O_R(I) = \beta O_R(J)\beta^{-1}$.

The Deuring correspondence. The Deuring correspondence due to Max Deuring [Deu41] draws a parallel between the world of quaternions and the world of supersingular elliptic curves. Indeed, if $E/\mathbb{F}_{p^2}$ is a supersingular elliptic curve, then its endomorphism ring $\operatorname{End}(E)$ is isomorphic to a maximal order $\mathcal{O} \subset \mathcal{B}_{p,\infty}$.

Example 24. If $p \equiv 3 \mod 4$, the elliptic curve $E_0 : y^2 = x^3 + x$ defined over $\mathbb{F}_p$ is supersingular and has a very explicit endomorphism ring $\mathrm{End}(E_0)$ isomorphic to $\mathcal{O}_0 := \langle 1, i, (i+j)/2, (1+k)/2 \rangle$, where j corresponds to the Frobenius endomorphism $(x, y) \in E_0 \longmapsto (x^p, y^p) \in E_0$ and i corresponds to the automorphism $(x, y) \in E_0 \longmapsto (-x, \zeta y) \in E_0$ (with $\zeta \in \mathbb{F}_{p^2}$ such that $\zeta^2 = -1$).

This is one of the very few examples where $\mathrm{End}(E_0)$ can be easily and explicitly computed. Computing $\mathrm{End}(E)$ is difficult in general.

Let E be a supersingular elliptic curve and $\mathcal{O} := \mathrm{End}(E)$. An isogeny $\phi : E \longrightarrow E'$ has a *kernel ideal* $I_\phi := \{\alpha \in \mathcal{O} \mid \forall P \in \ker(\phi), \quad \alpha(P) = 0\}$, which is a left $\mathcal{O}$-ideal of norm $\mathrm{nrd}(I_\phi) = \deg(\phi)$. Conversely, any left $\mathcal{O}$-ideal I defines an isogeny $\phi_I : E \longrightarrow E_I$ of kernel $E[I] := \{P \in E \mid \forall \alpha \in I, \quad \alpha(P) = 0\}$ and degree $\deg(\phi_I) = \mathrm{nrd}(I)$. We have $I_{\phi_I} = I$ and $\phi_{I_\phi} = \phi$ so this correspondence is one to one.

The Deuring correspondence between ideals and isogenies satisfies the following properties: two equivalent ideals $I \sim J$ have isomorphic codomains $E_I \simeq E_J$, the endomorphism ring of the codomain E_I is $\mathrm{End}(E_I) \cong O_R(I)$, the conjugate $\overline{I}$ corresponds to the dual isogeny $\widehat{\phi_I}$, the kernel ideal of the composite $\phi \circ \psi$ isogeny is $I_{\phi \circ \psi} = I_\psi \cdot I_\phi$, a principal ideal corresponds to an endomorphism. For a thorough presentation of quaternions and the Deuring correspondence, we recommend the book of Voight [Voi20].

Accessible torsion to make the Deuring correspondence effective. Making the Deuring correspondence effective means computing the isogeny $\phi_I : E \longrightarrow E_I$ associated to an ideal I and conversely, computing the kernel ideal I_ϕ of a known isogeny $\phi : E \longrightarrow E'$ when $\mathrm{End}(E)$ is known. Until recently, this could be done in polynomial time only when the ideal norm/the degree is smooth and the necessary $\mathrm{nrd}(I)$-torsion to do these computations is "accessible" in the following sense.

Definition 25. Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve and $T := \prod_{i=1}^r \ell_i^{e_i}$ be an integer, where the ℓ_i are distinct primes. We say that E has *accessible* T-torsion if $E[\ell_i^{e_i}]$ is defined over an extension of $\mathbb{F}_p$ of degree polynomial in $\log(p)$ for all $i \in [\![1 \ ; \ r]\!]$.

Usually, in cryptographic protocols we choose p to ensure the T-torsion is defined over $\mathbb{F}_{p^2}$ or $\mathbb{F}_{p^4}$ to optimize T-isogeny computations when T is smooth. In general, if T is B-powersmooth with B polynomial in $\log(p)$, the T-torsion is always accessible and we can compute T-isogenies in polynomial time as a product of low degree isogenies as in [EHLMP18, Proposition 4]. Until recently, those were the only way to compute isogenies and make the Deuring correspondence effective. In this work, we propose a method to compute isogenies of non-smooth degree (see Section 2). We specialize it for SQIsignHD but this could be easily generalized.

A.4 Algorithms for effective Deuring correspondence

In this section, we recall already known polynomial time algorithms making the Deuring correspondence effective. Those algorithms are used as ingredients of SQIsignHD. They were mainly introduced for previous versions of SQIsign [DKLPW20; FLLW23].

Pushing the endomorphism ring through an isogeny. In this paragraph, we introduce an algorithm to compute a basis of $\mathrm{End}(E)$ that we can easily evaluate when we know an isogeny $E_0 \longrightarrow E$ and a basis of $\mathrm{End}(E_0)$ (in practice, E_0 is the elliptic curve of Example 24).

Definition 26. Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve and $\mathcal{O}$ be a maximal quaternion order in $\mathcal{B}_{p,\infty}$ isomorphic to $\mathrm{End}(E)$. An *eval-basis* of $\mathrm{End}(E)$ is the data of a basis $(\alpha_1, \cdots, \alpha_4)$ of $\mathcal{O}$ and an isomorphism $\varepsilon : \mathcal{O} \xrightarrow{\sim} \mathrm{End}(E)$ such that the $\varepsilon(\alpha_i)$ can be evaluated at any point of E in polynomial time in $\log(p)$. We say it is an N*-eval-basis* if we can only evaluate the $\varepsilon(\alpha_i)$ on points of order prime to N.

For any left-ideal $I \subseteq \mathcal{O}$, we define an $(N$-)eval-basis of I in a similar way. Such a basis can be obtained from an $(N$-)eval-basis of $\mathrm{End}(E)$.

Assume that we know an eval-basis $((\alpha_1, \cdots, \alpha_4), \varepsilon)$ of $\mathrm{End}(E_0)$. Let $\psi : E_0 \longrightarrow E_1$ be an isogeny of degree N with an efficient representation. Here, we explain how to use ψ to compute an N-eval-basis of $\mathrm{End}(E_1)$.

By [Voi20, Lemma 42.2.9], the map

$$\begin{aligned} \iota : \mathrm{End}(E_1) &\longrightarrow \mathcal{B}_{p,\infty} \\ \phi &\longmapsto \frac{1}{N}\varepsilon^{-1}(\widehat{\psi} \circ \phi \circ \psi) \end{aligned}$$

induces an isomorphism $\mathrm{End}(E_1) \xrightarrow{\sim} \mathcal{O}_1 := O_R(I_\psi)$. Assuming that we know I_ψ, we can obtain a $\mathbb{Z}$-basis $(\beta_1, \cdots, \beta_4)$ of $\mathcal{O}_1$ via the formula $\mathcal{O}_1 = 1/N\overline{I_\psi}I_\psi$ [Voi20, Proposition 16.6.15]. Then, ι^{-1} induces an isomorphism $\mathcal{O}_1 \xrightarrow{\sim} \mathrm{End}(E_1)$.

We now explain how to use ι^{-1} to evaluate the β_i. Since $N\mathcal{O}_1 = \overline{I_\psi}I_\psi \subseteq \mathcal{O}_0$, we can write

$$\beta_i := \frac{1}{N}\sum_{j=1} c_{i,j}\alpha_j,$$

with $c_{i,j} \in \mathbb{Z}$, for all $i \in \llbracket 1 \ ; \ 4 \rrbracket$. We then have:

$$\iota^{-1}(\beta_i) = \frac{1}{N^2}\sum_{j=1}^{4} c_{i,j}\psi \circ \varepsilon(\alpha_j) \circ \widehat{\psi},$$

so we can indeed easily evaluate the β_i on any point of E_1 of order prime to N.

```
Algorithm 8: PushEndRing [EHLMP18, Algorithm 4]
  Data: An eval-basis $((\alpha_1, \cdots, \alpha_4), \varepsilon)$ of $\mathrm{End}(E_0)$ and an isogeny $\psi : E_0 \longrightarrow E_1$
        of degree $N$ with known kernel ideal $I_\psi$.
  Result: An $N$-eval-basis $(\beta_1, \cdots, \beta_4)$ of $\mathrm{End}(E_1)$.
  Compute a $\mathbb{Z}$-basis $(\beta_1, \cdots, \beta_4)$ of $\mathcal{O}_1 := 1/N\overline{I_\psi}I_\psi$;
  Write $\beta_i := 1/N \sum_{j=1} c_{i,j}\alpha_j$, with $c_{i,j} \in \mathbb{Z}$, for all $i = 1, \cdots, 4$;
  Let $\varepsilon' : \mathcal{O}_1 \xrightarrow{\sim} \mathrm{End}(E_1), \beta_i \longmapsto 1/N^2 \sum_{j=1}^4 c_{i,j}\psi \circ \varepsilon(\alpha_j) \circ \widehat{\psi}$;
  Return $(\beta_1, \cdots, \beta_4)$ and $\varepsilon'$;
```

```
Algorithm 9: KernelToIdeal$_D$
  Data: A point $P \in E_1$ of smooth order $D$ and an eval-basis $((\beta_1, \cdots, \beta_4), \varepsilon')$
        of $\mathrm{End}(E_1) \cong \mathcal{O}_1$ that can be evaluated on $E_1[D]$.
  Result: The ideal $I(\langle P \rangle) \subseteq \mathcal{O}_1$ associated to the isogeny of kernel $\langle P \rangle$.
  1 Compute $Q_i := \varepsilon'(\beta_i)(P)$ for $i = 1, \cdots, 4$;
  2 Find $i, j$ such that $(Q_i, Q_j)$ is a basis of $E_1[D]$;
  3 For $k \neq i, j$, find $a, b \in \mathbb{Z}/D\mathbb{Z}$ such that $Q_k = aQ_i + bQ_j$ (discrete logarithm
    problem);
  4 Let $\gamma := \beta_k - a\beta_i - b\beta_j$;
  5 Return $\mathcal{O}_1\gamma + \mathcal{O}_1 D$;
```

From isogenies to ideals. Let $\varphi : E_1 \longrightarrow E_2$ be a cyclic D-isogeny (with D smooth) represented by a generator of its kernel P. We give an algorithm KernelToIdeal$_D$ (Algorithm 9) due to Leroux [Ler22, Algorithm 20] to compute the ideal associated to a cyclic group $\langle P \rangle$ of a supersingular elliptic curve E_1 when we know an eval-basis of $\mathrm{End}(E_1)$ that can be evaluated on $E_1[D]$. This algorithm is a variant of the algorithm introduced by Galbraith, Petit and Silva [GPS20, Algorithm 2] with the same purpose.

Remark 27. Since D is smooth, the discrete logarithm problem in line 3 of Algorithm 9 is easy to solve with Pohlig-Hellman methods generalized by Teske to multiple discrete logarithms [PH78; Tes99].

Lemma 28. *[Ler22, Lemma 4.22] Algorithm 9 terminates and is correct.*

Kernel ideal computation. The two preceding algorithms immediately yield an algorithm IsogenyToIdeal (Algorithm 10) computing the kernel ideal of $\varphi : E_1 \longrightarrow E_2$ of smooth degree D_φ when given an isogeny $\psi : E_0 \longrightarrow E_1$ of smooth degree D_ψ prime to D_φ with known kernel ideal I_ψ and an eval-basis of $\mathrm{End}(E_0)$.

The KLPT algorithm. The KLPT algorithm was first introduced in [KLPT14] for extremal orders (such as $\mathcal{O}_0$ in Example 24) and then generalized to other orders and improved in [DKLPW20], [FLLW23] and [Ler22]. In the following, we refer to the version of KLPT introduced in [Ler22, Algorithm 7]. Given a left

```
Algorithm 10: IsogenyToIdeal
  Data: An isogeny $\varphi : E_1 \longrightarrow E_2$ of smooth degree $D_\varphi$, another isogeny $\psi$ :
        $E_0 \longrightarrow E_1$ of smooth degree $D_\psi$ coprime with $D_\varphi$ along with its kernel
        ideal $I_\psi$ and an eval-basis $\mathcal{B}_0$ of $\mathrm{End}(E_0)$.
  Result: The kernel ideal $I_\varphi$ associated to $\varphi$.
  1 Compute a $D_\psi$-eval-basis $\mathcal{B}_1 := \mathsf{PushEndRing}(\mathcal{B}_0, \psi, I_\psi)$ of $\mathrm{End}(E_1)$;
  2 Let $P$ be a generator of $\ker(\varphi)$;
  3 $I_\varphi \longleftarrow \mathsf{KernelToIdeal}_{D_\varphi}(P, \mathcal{B}_1)$;
  4 Return $I_\varphi$;
```

$\mathcal{O}_0$-ideal I and a (smooth) integer $N = \Omega(p^3)$, $\mathsf{KLPT}_N(I)$ returns an equivalent ideal $J \sim I$ of norm dividing N.

This algorithm is generally used when we want to compute an isogeny path $E_0 \longrightarrow E$. Given an ideal I connecting $\mathcal{O}_0 \cong \mathrm{End}(E_0)$ to $\mathcal{O} \cong \mathrm{End}(E)$, which may not have smooth norm, we find $J \sim I$ with KLPT of smooth norm N, so that the isogeny associated to J, $\phi_J : E_0 \longrightarrow E$ can be computed.

From ideal to isogenies. Given a maximal order $\mathcal{O}$ and a left $\mathcal{O}$-ideal J, we have a straightforward way to compute the associated isogeny $\phi_J : E \longrightarrow E'$. We evaluate a basis of J on $E[\mathrm{nrd}(J)]$ to compute $E[J] = \ker(\phi_J)$ and then compute ϕ_J as a chain of small degree isogenies (assuming $\mathrm{nrd}(J)$ is smooth) using Vélu's formulas [Vél71]. Unfortunately, we might not have accessible $\mathrm{nrd}(J)$-torsion, especially if J is obtained from KLPT ($\mathrm{nrd}(J) = \Theta(p^3)$ when $\mathcal{O} = \mathcal{O}_0$ and even bigger otherwise).

In [DKLPW20, Algorithm 7], the authors introduced the SpecialIdealToIsogeny algorithm to perform this computation with half of the torsion when $\mathcal{O} = \mathcal{O}_0$ and when we know an alternate isogeny path. Assume we have accessible T-torsion with $T = \Omega(p^{3/2})$. SpecialIdealToIsogeny takes as input two left-ideals $J, I \subseteq \mathcal{O}_0$ of coprime norm such that $I \sim J$ and $\mathrm{nrd}(J) | T^2$ along with the isogeny $\phi_I : E_0 \longrightarrow E$ associated to I. It returns the isogeny $\phi_J : E_0 \longrightarrow E$ associated to J.

A.5 A non-smooth isogeny torsion evaluation algorithm

We present $\mathsf{EvalTorsion}_D$ that evaluates a non-smooth degree isogeny on D-torsion points (with D smooth) knowing its kernel ideal and an alternate smooth degree path. This algorithm is greatly inspired from [FKMT22, Algorithm1] or [Ler23, Algorithm 2]. Let I be an ideal connecting $\mathcal{O} \cong \mathrm{End}(E)$ and $\mathcal{O}' \cong \mathrm{End}(E')$ of non-smooth norm q, (P_1, P_2) be a basis of $E[D]$, $\rho_1 : E_0 \longrightarrow E$ and $\rho_2 : E_0 \longrightarrow E'$, be two isogenies of respective degrees d_1, d_2 prime to D, and respective kernel ideals I_1 and I_2. We want to compute $(\phi_I(P_1), \phi_I(P_2))$, where $\phi_I : E \longrightarrow E'$ is the isogeny associated to I.

Let us consider the endomorphism $\gamma := \widehat{\rho}_2 \circ \phi_I \circ \rho_1$ of E_0. From that definition of γ comes the equality

$$[d_1 d_2]\phi_I = \rho_2 \circ \gamma \circ \widehat{\rho}_1.$$

Algorithm 11: $\mathsf{EvalTorsion}_D$

```
Data: A basis $(P_1, P_2)$ of $E[D]$, an ideal $I$ connecting $\mathcal{O} \cong \mathrm{End}(E)$ and $\mathcal{O}' := \mathrm{End}(E')$, two isogenies $\rho_1 : E_0 \longrightarrow E$ and $\rho_2 : E_0 \longrightarrow E'$ of respective degrees $d_1$ and $d_2$ prime to $D$ and their respective kernel ideals $I_1$ and $I_2$.
Result: $(\phi_I(P_1), \phi_I(P_2))$, where $\phi_I : E \longrightarrow E'$ is the isogeny associated to $I$.
Find $\gamma \in \mathcal{O}_0$ such that $\mathcal{O}_0\gamma = I_1 \cdot I \cdot \overline{I_2}$;
$R_i \longleftarrow \rho_2 \circ \gamma \circ \widehat{\rho_1}(P_i)$ for $i \in \{1, 2\}$;
Compute $\lambda$, an inverse of $d_1 d_2$ modulo $D$;
Return $([\lambda]R_1, [\lambda]R_2)$;
```

Since D is prime to d_1 and d_2, the scalar $d_1 d_2$ can be inverted modulo D and we see that it suffices to evaluate $\gamma, \widehat{\rho_1}, \rho_2$ on the D-torsion of their respective domains.

The curve E_0 is chosen to have a known endomorphism ring so we can easily evaluate γ at any point from a basis of endomorphisms if we know the principal ideal $\mathcal{O}_0\gamma$. This ideal can be computed from the ideals $I_1, I_2 \subset \mathcal{O}_0$ and $I \subset \mathcal{O}$ associated to ρ_1, ρ_2 and ϕ_I respectively, with the formula $I_1 \cdot I \cdot \overline{I_2} = \mathcal{O}_0\gamma$. The $\mathsf{EvalTorsion}_D$ algorithm summarizes the procedure described above.

B The description of RigorousSQIsignHD

In this section, we describe the details of the RigorousSQIsignHD protocols and parameters. Recall that unlike FastSQIsignHD, this algorithm relies on a verification procedure in dimension 8 and is optimised for security at the expense of efficiency. This scheme is theoretical and not meant to be implemented.

B.1 In RigorousSQIsignHD, more torsion is needed than in FastSQIsignHD.

As in FastSQIsignHD, in RigorousSQIsignHD we still require $p = \Omega(2^{2\lambda})$ to achieve a classical security level of λ bits (and a quantum security level of $\lambda/2$ bits). The degrees D_τ, D_ψ and D_φ still need to be coprime with ℓ for the response isogeny evaluation with $\mathsf{EvalTorsion}_{\ell^f}$, while $D_{\tau'}$ needs to be coprime with D_φ for the computation of I_φ. In addition, we need E_1 and E_A to be rigorously (and not only heuristically) computationally indistinguishable from a uniformly random supersingular elliptic curve. For that reason, we shall have $D_\tau, D_\psi = \Theta(p^3)$ and $D_{\tau'} = \ell^h = \Theta(p^2)$ (as will be explained in Appendix B.2).

For the computation of the secret key τ and commitment isogeny ψ and challenge φ, we need accessible T-torsion such that $D_\tau, D_\psi | T^2$ with T prime to ℓ and $T = \Omega(p^{3/2})$, as in the original SQIsign protocol [DKLPW20]. We also require $D_\varphi | T$ to compute the challenge whose degree is $D_\varphi = O(p^3)$ (as will be explained in Appendix D.1). Hence, we require $T \simeq p^3$.

As previously, we can compute the ℓ^e-isogeny F representing the response σ in dimension 8 as long as we have accessible ℓ^f-torsion with $2f \geq e+4$. However, as will be explained in Appendix D.2, to prove the zero-knowledge property, we need $\ell^e = \Theta(p^2)$ instead of $\Theta(\sqrt{p})$, so we can have $\ell^f = \Theta(p)$. Hence p should be of the form $p = c\ell^f - 1$ with c as small as possible.

Then, the accessible T-torsion will be defined over small field extensions of $\mathbb{F}_{p^2}$. For instance, T could be the product of the smallest successive primes (ℓ excluded) such that $T \simeq p^3$. All the primes we need will be $O(\log(p))$, so the T-torsion will be defined on extensions of degree $O(\log(p))$ of $\mathbb{F}_{p^2}$ and all isogeny computations will be polynomial in $\log(p)$.

This ensures in particular that RigorousSQIsignHD complexity scales polynomially with the security parameter λ (as FastSQIsignHD), which is an interesting property in comparison with SQIsign. Indeed, in SQIsign [FLLW23] the authors imposed $\ell^f | p-1$ and $T | p^2 - 1$ (with $T \simeq p^{5/4}$), so that the whole T-torsion is defined over $\mathbb{F}_{p^4}$ (and x-coordinates are defined over $\mathbb{F}_{p^2}$). Finding such primes has been an open research question since the introduction of SQIsign [CMN21; BSC+22; Ahr23]. It is still unclear if we can still find T sufficiently smooth as the security level grows. Since computing a prime degree isogeny is exponential in the degree, the SQIsign protocol might not be polynomial in $\log(p)$.

B.2 Provably secure key generation and commitment

To prove security, the distribution of the public key E_A and the commitment E_1 need to be close to the uniform distribution in the supersingular isogeny graph. Heuristically, we expect to get an elliptic curve with a distribution somewhat close to uniform after a random isogeny walk of length $\Theta(p)$. However, the fact that we compute two paths simultaneously in FastCommit and FastKeyGen might alter the distribution of the resulting elliptic curve. While we do not expect the induced bias to be computationally relevant, it prevents a rigorous analysis. This is the reason why we now propose a different procedure for RigorousKeyGen and RigorousCommit.

Starting from E_0, we generate a random ℓ-isogeny walk $\phi : E_0 \longrightarrow E$ long enough to make E uniformly random and compute its kernel ideal I_ϕ as well as an alternate path $\phi' : E_0 \longrightarrow E$ of degree dividing T^2. This algorithm is very similar to what is done in the first version of SQIsign signing algorithm [DKLPW20]. The procedure we shall describe is costly (though polynomial) and this is one of the reasons why we do not expect the efficiency of RigorousSQIsignHD to compare favourably to the original SQIsign protocol.

A long enough supersingular ℓ-isogeny walk.

Proposition 29. *Let $\phi : E_0 \longrightarrow E$ be an ℓ^h-isogeny obtained from a non-backtracking random ℓ-isogeny walk. Then, for all $\varepsilon \in]0, 2]$, the distribution of E has statistical distance $\tilde{O}(p^{-\varepsilon/2})$ to the uniform distribution in the supersingular isogeny graph, provided that $h \geq (1+\varepsilon)\log_\ell(p)$.*

Proof. Let $\mathrm{SS}(p)$ be the set of supersingular elliptic curves over $\mathbb{F}_{p^2}$ (up to isomorphism) and S be the probability distribution on $\mathrm{SS}(p)$ given by $S(E) := K^{-1}/\#\operatorname{Aut}(E)$ for all $E \in \mathrm{SS}(p)$, with $K := \sum_{E\in \mathrm{SS}(p)} 1/\#\operatorname{Aut}(E)$. Let δ_0 be the Dirac distribution on E_0 and $\delta_0^{(h)}$ the distribution obtained from δ_0 after a non-backtracking ℓ-isogeny walk of length h. By [BCC+23, Theorem 11], the statistical distance between S and $\delta_0^{(h)}$ satisfies

$$d_{TV}\left(S, \delta_0^{(h)}\right) := \frac{1}{2}\sum_{E\in \mathrm{SS}(p)} \left|S(E) - \delta_0^{(h)}(E)\right| \leq \frac{\sqrt{6K}}{2}\frac{(\ell+1)(h+1)-2}{(\ell+1)\sqrt{\ell^h}}.$$

By the Eichler's mass formula [Voi20, p. 42.3.8], we know that $K = (p-1)/24$. Then, when $h \geq (1+\varepsilon)\log_\ell(p)$, we get that $d_{TV}(S, \delta_0^{(h)}) = O(\log(p)p^{-\varepsilon/2}) = \tilde{O}(p^{-\varepsilon/2})$.

By Lemma 30, we have $d_{TV}(U,S) = O(p^{-1})$, so we finally get, by triangular inequality, that $d_{TV}(U, \delta_0^{(h)}) = \tilde{O}(p^{-\varepsilon/2})$. □

Lemma 30. *The statistical distance between S and U is $d_{TV}(S,U) = O(p^{-1})$.*

Proof. By [Sil09][Theorem III.10.1], we have $\#\operatorname{Aut}(E) = 2$ for all $E \in \mathrm{SS}(p)$ such that $j(E) \neq 0, 1728$, and $\#\operatorname{Aut}(E) \in \{4,6\}$ otherwise and by [Sil09][Theorem V.4.1], there exists $C_p \in \mathbb{Z}$ small such that $\#\mathrm{SS}(p) = 2K + C_p$. Hence, we have

$$\begin{aligned} d_{TV}(U,S) &= \frac{1}{2}\sum_{E\in \mathrm{SS}(p)} \left|\frac{1}{\#\mathrm{SS}(p)} - \frac{1}{K\#\operatorname{Aut}(E)}\right| \\ &= \frac{1}{2}\sum_{\substack{E\in \mathrm{SS}(p)\\ j(E)\neq 0,1728}} \left|\frac{1}{2K+C_p} - \frac{1}{2K}\right| + O(p^{-1}) \\ &= \frac{1}{2}\frac{C_p}{2K(2K+C_p)}(\#SS(p) + O(1)) + O(p^{-1}) \\ &= \frac{C_p}{4K} + O(p^{-1}) = O(p^{-1}). \end{aligned}$$

□

By Proposition 29 (with $\varepsilon = 1$), an adversary needs time $\tilde{O}(\sqrt{p})$ to distinguish a supersingular elliptic curve obtained from E_0 after a non-backtracking ℓ-isogeny path of degree $\Theta(p^2)$ from a uniformly random elliptic curve in the supersingular isogeny graph. Since $p = \Theta(2^\lambda)$, this is sufficient to ensure a security level of λ bits.

Computing the kernel ideal and an alternate path. Assume that we have generated a random ℓ-isogeny walk $\phi : E_0 \longrightarrow E$ of degree $\ell^h = \Theta(p^2)$.

Algorithm 12: $\mathsf{RigorousDoublePath}_{\ell^f,T}$

```
Data: A supersingular elliptic curve $E_0$ with accessible $T$-torsion and $\ell^f$-
      torsion, an eval-basis $\mathcal{B}_0$ of $\mathrm{End}(E_0)$ in the sense of Definition 26.
Result: A random $\ell$-isogeny walk $\phi : E_0 \longrightarrow E$ of degree $\ell^h = \Theta(p^2)$ such that
        the distribution of $E$ has statistical distance $\tilde{O}(p^{-1/2})$ to the uniform,
        an isogeny $\phi' : E_0 \longrightarrow E$ of degree dividing $T^2$ and their respective
        kernel ideals $I_\phi$ and $I_{\phi'}$.
$h \longleftarrow \lceil 2\log_\ell(p) \rceil$;
Perform a random non-backtracking $\ell$-isogeny walk $\phi : E_0 \longrightarrow E$ of degree $\ell^h$;
Decompose $\phi$ in a sequence of isogenies $\phi_i : E_i \longrightarrow E_{i+1}$ $(1 \leq i \leq r)$ of degree
  dividing $\ell^f$;
Let $P_i$ be a generator of $\ker(\phi_i)$ for all $i \in [\![1\ ;\ r]\!]$;
$J_0 \longleftarrow \mathcal{O}_0$;
for $i := 0$ to $r - 1$ do
    $I_i \longleftarrow \mathsf{KernelToIdeal}_{\ell^f}(\mathcal{B}_i, P_i)$;
    $J_{i+1} \longleftarrow \mathsf{KLPT}_{T^2}(J_i \cdot I_i)$;
    $\phi'_{i+1} \longleftarrow \mathsf{SpecialIdealToIsogeny}(J_{i+1}, I_0 \cdots I_i, \phi_i \circ \cdots \circ \phi_0)$;
    Compute $\mathcal{B}_{i+1} := \mathsf{PushEndRing}(\mathcal{B}_0, \phi'_{i+1}, J_{i+1})$, a $T$-eval-basis of
      $\mathrm{End}(E_{i+1})$;
end
$\phi' \longleftarrow \phi'_r, I_\phi \longleftarrow I_0 \cdots I_{r-1}, I_{\phi'} \longleftarrow J_r$;
Return $\phi, \phi', I_\phi, I_{\phi'}$;
```

To compute the kernel ideal I_ϕ, we cannot use $\mathsf{KernelToIdeal}_{\ell^h}$ (Algorithm 9) because we would need accessible ℓ^h-torsion, which is impossible since $\ell^h \gg p$.

Instead, as in [DKLPW20], we divide ϕ into a sequence of isogenies $\phi_i : E_i \longrightarrow E_{i+1}$ $(1 \leq i \leq r)$ of degree dividing ℓ^f and compute their associated kernel ideals I_i successively. For the computation of I_i, we need to compute an alternate isogeny $\phi'_i : E_0 \longrightarrow E_i$ of degree dividing T^2 obtained at step $i-1$. Hence, the alternate path $\phi' := \phi_r : E_0 \longrightarrow E_r = E$ will be a convenient by-product of our ideal computation. The algorithm we propose (see Algorithm 12) uses the sub-algorithms KLPT_{T^2} and $\mathsf{SpecialIdealToIsogeny}$ as black boxes (see Appendix A.4).

Application to key generation and commitment. For the key generation phase $\mathsf{RigorousKeyGen}$, we simply call $\mathsf{RigorousDoublePath}_{\ell^f,T}$ to output two isogenies $\tau', \tau : E_0 \longrightarrow E_A$ with codomain E_A statistically close to uniform and respective degrees ℓ^h and dividing T^2, along with their kernel ideals $I_{\tau'}$ and I_τ.

For the commitment $\mathsf{RigorousCommit}$, we only need an isogeny $\psi : E_0 \longrightarrow E_1$ of degree T^2 (prime to ℓ) with codomain E_1 statistically close to uniform. $\mathsf{RigorousDoublePath}_{\ell^f,T}$ will output $\phi, \phi' : E_0 \longrightarrow E_1$ of respective degrees ℓ^h and $D_{\phi'} | T^2$, along with $I_\phi, I_{\phi'}$. The data (ϕ, I_ϕ) will not be used so we only compute ϕ to ensure the randomness of E_1 and I_ϕ as an intermediary tool to obtain $(\psi', I_{\psi'}) := (\phi', I_{\phi'})$.

B.3 The challenge generation

In RigorousSQIsignHD, the degree of the challenge isogeny D_φ will be a divisor of T of size $D_\varphi \simeq p^3$. More details on the choice of D_φ in RigorousSQIsignHD will be given in the security analysis (see Appendix D.1).

B.4 Response and verification in RigorousSQIsignHD

In this section, we cover the algorithmic blocks introduced in Section 4.1 for FastSQIsignHD. Some of the algorithms are common with the dimension 4 and some are specific to the dimension 8. As in Section 4.1, we start with an overview of the algorithmic building blocks of $\mathsf{RigorousRespond}$ and $\mathsf{RigorousVerify}$ and then present them in detail.

Those algorithms use the following sub-algorithms:

- $\mathsf{IsogenyToIdeal}(\varphi, \psi, I_\psi)$ (already presented in Appendix A.4).
- $\mathsf{RandomEquivalentIdeal}_{\ell^e}$ (already presented in Section 4.2).
- $\mathsf{EvalTorsion}_{\ell^f}$ (already presented in Appendix A.5).
- $\mathsf{RepresentIsogeny}_{8,\ell^e,\ell^f}$ takes as input an integer $q < \ell^e$ coprime with ℓ, integers $a_1, \cdots, a_4$ such that $a_1^2 + \cdots + a_4^2 + q = \ell^e$, a basis (P_1, P_2) of $E_1[\ell^f]$, $(\sigma(P_1), \sigma(P_2))$, where $\sigma : E_1 \longrightarrow E_2$ is a q-isogeny, and returns a chain of 8-dimensional ℓ-isogenies whose composition is $F(\sigma, a_1, \cdots, a_4)$.
- $\mathsf{IsValid}_8$, with input $F, E_1, E_2, \ell^e, \ell^f$, checks if F is a valid output of $\mathsf{RepresentIsogeny}_{8,\ell^e,\ell^f}$ representing an isogeny $\sigma : E_1 \longrightarrow E_2$ in dimension 8.

The response and verification procedures follow the same ideas introduced for FastSQIsignHD. To respond, the prover computes a random ideal $I \sim \overline{I_\psi} \cdot I_\tau \cdot I_\varphi$ of norm $q \leq \ell^e$ and evaluates the associated isogeny $\sigma : E_1 \longrightarrow E_2$ on a canonical basis (P_1, P_2) of $E_1[\ell^f]$. Then, they send $(q, \sigma(P_1), \sigma(P_2))$ to the verifier. The latter, after receiving $(q, \sigma(P_1), \sigma(P_2))$ computes integers $a_1, \cdots, a_4$ such that $a_1^2 + \cdots + a_4^2 + q = \ell^e$, computes $F(\sigma, a_1, \cdots, a_4)$ and verifies the result is correct with $\mathsf{IsValid}_8$. Algorithms 13 and 14 follow.

Remark B.1 (Case $\gcd(q, \ell) \neq 1$*).* For security reasons (see Remark 7), $q = \mathrm{nrd}(I) = \deg(\sigma)$ can take any value $\leq \ell^e$, including multiples of ℓ. In that case, we cannot evaluate σ on $E_1[\ell^f]$ with $\mathsf{EvalTorsion}_{\ell^f}$ directly. To know how to treat this case, we refer to Appendix C. Our solution involves factoring σ by ℓ-isogenies and applying $\mathsf{RepresentIsogeny}_{8,\ell^e,\ell^f}$ to a factor of σ of degree prime to ℓ. In the following, we assume q prime to ℓ for simplicity.

As input of Algorithms 13 and 14, we denote by:

- $\mathsf{RigorousSetup}$, the public parameters of RigorousSQIsignHD, p, ℓ and f such that $\ell^f | p - 1$, e, T powersmooth (accessible torsion) and E_0;
- $\mathsf{SecretKey}$, the isogenies $\tau, \tau' : E_0 \longrightarrow E_A$ of degrees $D_\tau | T^2$ and $D_{\tau'} = \ell^h$ respectively along with their kernel ideals I_τ and $I_{\tau'}$;
- $\mathsf{CommitData}$, the isogeny $\psi : E_0 \longrightarrow E_1$ of degree $D_\psi | T^2$ and its kernel ideal I_ψ;
- $\mathsf{ChallData}$, the isogeny $\varphi : E_A \longrightarrow E_2$ of degree $D_\varphi | T$.

Algorithm 13: RigorousRespond

```
Data: RigorousSetup, SecretKey, CommitData and ChallData.
Result: $(\sigma(P_1), \sigma(P_2), q)$, where $(P_1, P_2)$ is a canonically determined basis of
        $E_1[\ell^f]$ and $\sigma : E_1 \longrightarrow E_2$ is a $q$-isogeny with $q < \ell^e$.
$I_\varphi \longleftarrow$ IsogenyToIdeal$(\varphi, \tau', I_{\tau'})$;
$J \longleftarrow \overline{I_\psi} \cdot I_\tau \cdot I_\varphi$;
$I \longleftarrow$ RandomEquivalentIdeal$_{\ell^e}(J)$ and $q \longleftarrow \mathrm{nrd}(I)$;
Compute the canonical basis $(P_1, P_2)$ of $E_1[\ell^f]$;
$(\sigma(P_1), \sigma(P_2)) \longleftarrow$ EvalTorsion$_{\ell^f}(I, P_1, P_2, \psi, \varphi \circ \tau, I_\psi, I_\tau \cdot I_\varphi)$;
Return $(\sigma(P_1), \sigma(P_2), q)$;
```

Algorithm 14: RigorousVerify

```
Data: RigorousSetup, $E_1, E_2$ and an output $R$ from RigorousRespond.
Result: Determines if $R$ is a valid response.
Try to parse $R := (R_1, R_2, q)$, where $R_1, R_2 \in E_2[\ell^f]$ and $q < \ell^e$ and return
  False if it fails;
Compute the canonical basis $(P_1, P_2)$ of $E_1[\ell^f]$;
Find $a_1, \cdots, a_4 \in \mathbb{Z}$ such that $a_1^2 + \cdots + a_4^2 + q = \ell^e$ using Pollack and
  Treviño's algorithm [PT18];
$F \longleftarrow$ RepresentIsogeny$_{8,\ell^e,\ell^f}(E_1, E_2, a_1, \cdots, a_4, P_1, P_2, R_1, R_2)$;
if $F \neq$ False then
    Return IsValid$_{8,\ell^e,T}(F, E_1, E_2, a_1, \cdots, a_4)$;
else
    Return False;
end
```

Computing the response isogeny representation in dimension 8. We introduce RepresentIsogeny$_{8,\ell^e,\ell^f}$ that takes as input an integer $q < \ell^e$ prime to ℓ, integers $a_1, \cdots, a_4$ such that $a_1^2 + \cdots + a_4^2 + q = \ell^e$, a basis (P_1, P_2) of $E_1[\ell^f]$, $(\sigma(P_1), \sigma(P_2))$, where $\sigma : E_1 \longrightarrow E_2$ is a q-isogeny, and returns a chain of 8-dimensional ℓ-isogenies whose composition is $F(\sigma, a_1, \cdots, a_4)$.

The algorithm RepresentIsogeny$_{8,\ell^e,\ell^f}$ (Algorithm 15) only works when q and ℓ are coprime. We explain in Appendix C how to treat the general case. The ingredients of Algorithm 15 are very similar to those of Algorithm 4. As explained in Section 4.3, we divide the computation in two $F = F_2 \circ F_1$ due to missing accessible ℓ^e-torsion. We start by computing a basis of $\ker(F_1)$ and $\ker(\widetilde{F_2})$ and compute F_1 and $\widetilde{F_2}$ as ℓ-isogeny chains in the Θ-model using KernelToIsogeny$_8$ algorithms (presented in Appendix F). So far, dimension 8 isogenies have not been implemented and we do not provide any implementation (unlike in dimension 4). Such an implementation would probably not be practical for cryptography.

Verification in dimension 8. We describe the verification procedure IsValid$_8$ (Algorithm 16) taking as input the isogenies F_1 and $\widetilde{F_2}$ outputted by RepresentIsogeny$_{8,\ell^e,\ell^f}$ and determining if they represent an isogeny $E_1 \longrightarrow E_2$

Algorithm 15: $\mathsf{RepresentIsogeny}_{8,\ell^e,\ell^f}$

```
Data: $E_1, E_2, a_1, a_2, a_3, a_4 \in \mathbb{Z}$, a basis $(P_1, P_2)$ of $E_1[\ell^f]$, $(\sigma(P_1), \sigma(P_2))$ where
      $\sigma : E_1 \longrightarrow E_2$ is a $q$-isogeny with $a_1^2 + a_2^2 + a_3^2 + a_4^2 + q = \ell^e$.
Result: An $\ell^{e_1}$-isogeny $F_1 : E_1^4 \times E_2^4 \longrightarrow \mathcal{C}$ and a $\ell^{e_2}$-isogeny $\widetilde{F_2} : E_1^4 \times E_2^4 \longrightarrow \mathcal{C}$
        such that $F(\sigma, a_1, a_2, a_3, a_4) = F_2 \circ F_1$, with $e_1, e_2 \leq f$ and $e_1 + e_2 = e$.
$e_2 \longleftarrow \lceil e/2 \rceil, e_1 \longleftarrow e - e_2$;
Let $\alpha_1 := \begin{pmatrix} a_1 & -a_2 & -a_3 & -a_4 \\ a_2 & a_1 & a_4 & -a_3 \\ a_3 & -a_4 & a_1 & a_2 \\ a_4 & a_3 & -a_2 & a_1 \end{pmatrix} \in \mathrm{End}(E_1^2)$ and $\Sigma := \mathrm{Diag}(\sigma, \sigma, \sigma, \sigma)$;
For all $i \in \{1, 2\}$ and $j \in \{1, 2, 3, 4\}$, let $\underline{P}_{i,j} \in E_1^4$ be the tuple with $P_i$ in
  position $j$ and 0 elsewhere;
$\mathscr{B}_0 \longleftarrow ([\ell^{f-e_1}]\widetilde{\alpha_1}(\underline{P}_{i,j}), [\ell^{f-e_1}]\Sigma(\underline{P}_{i,j}))_{\substack{1 \leq i \leq 2 \\ 1 \leq j \leq 4}}$;
$\mathscr{C}_0 \longleftarrow ([\ell^{f-e_2}]\alpha_1(\underline{P}_{i,j}), -[\ell^{f-e_2}]\Sigma(\underline{P}_{i,j}))_{\substack{1 \leq i \leq 2 \\ 1 \leq j \leq 4}}$;
if $\mathscr{C}_0$ and $\mathscr{B}_0$ are valid kernels of $2^{e_1}$ and $2^{e_2}$-isogenies then
    $F_1 \longleftarrow \mathsf{KernelToIsogeny}_{8,\ell^{e_1}}(\mathscr{B}_0)$;
    $\widetilde{F_2} \longleftarrow \mathsf{KernelToIsogeny}_{8,\ell^{e_2}}(\mathscr{C}_0)$;
    Return $F_1$ and $\widetilde{F_2}$;
else
    Return False;
end
```

of degree q. As in dimension 4, the idea is to check if F_1 and $\tilde{F}_2$ have the same codomain (computed as principally polarized abelian varieties) and then evaluate $F_2 \circ F_1$ on some points.

As in dimension 4, we can prove that this verification procedure is correct. We refer to Appendix E.5 for a proof in dimension 4. The proof would be very similar in dimension 8.

Proposition 31. *Algorithm 16 is correct. Namely, when given* $E_1, E_2, a_1, a_2, a_3, a_4, F_1, \widetilde{F_2}$*, if Algorithm 16 returns* True*, then* $F_2 \circ F_1$ *is an efficient representation of an isogeny* $\sigma : E_1 \longrightarrow E_2$ *of degree* $q = \ell^e - a_1^2 - a_2^2 - a_3^2 - a_4^2$.

Corollary 32. *The verification procedure* RigorousVerify *(Algorithm 14) is correct. Namely, on input* (R_1, R_2, q)*,* RigorousVerify *returns* True *if and only if* (R_1, R_2, q) *defines an efficient representation of an isogeny* $\sigma : E_1 \longrightarrow E_2$ *of degree* $q \leq \ell^e$.

C Response and verification in RigorousSQIsignHD when q is not prime to ℓ

As explained in Section 2.3, we have no guarantee that q is prime to ℓ in dimension 8, and in that case we can no longer use directly the simple formula

```
Algorithm 16: $\mathsf{IsValid}_{8,\ell^e,T}$
Data: Elliptic curves $E_1, E_2$, integers $a_1, a_2, a_3, a_4 \in \mathbb{Z}$ and the output $(F_1, \widetilde{F_2})$
      of $\mathsf{RepresentIsogeny}_{8,\ell^e,\ell^f}(E_1, E_2, a_1, a_2, a_3, a_4, *, *, *, *)$.
Result: Determines if $F_2 \circ F_1$ is an efficient representation of an isogeny $\sigma$ :
      $E_1 \longrightarrow E_2$ of degree $q := \ell^e - a_1^2 - a_2^2 - a_3^2 - a_4^2$.
Let $(\mathcal{C}_1, \lambda_1)$ and $(\mathcal{C}_2, \lambda_2)$ be the respective codomains of $F_1$ and $\widetilde{F_2}$;
if $(\mathcal{C}_1, \lambda_1) \neq (\mathcal{C}_2, \lambda_2)$ then
    Return False;
else
    Write $T := \prod_{i=1}^r \ell_i^{\beta_i}$ (accessible torsion);
    Generate a basis $(U_{i,1}, U_{i,2})$ of $E_1[\ell_i^{\beta_i}]$ for all $i \in [\![1\ ;\ r]\!]$;
    Let $\alpha_1 \in \mathrm{End}(E_1^2)$ be as in line 2 of Algorithm 15;
    Compute $\underline{T}_{i,j} \longleftarrow F_2 \circ F_1(U_{i,j}, 0, \cdots, 0)$ for all $i \in [\![1\ ;\ r]\!]$ and $j \in \{1, 2\}$;
    if $\underline{T}_{i,j} = (\alpha_1(U_{i,j}, 0, 0, 0), *, 0, 0, 0)$ for all $(i, j) \in [\![1\ ;\ r]\!] \times \{1, 2, 3, 4\}$ then
        Return True;
    else
        Return False;
    end
end
```

for $\ker(F)$ and the optimisations of Section 4.3 to compute the 8-dimensional isogeny F embedding the response σ. To be able to use the techniques we developed, we factor σ into $\sigma := \widehat{\sigma_2} \circ \sigma' \circ \sigma_1$, where σ_1 and σ_2 both have degree dividing ℓ^f and σ' has degree prime to ℓ. We then represent σ' with the techniques we presented earlier.

C.1 Finding the ℓ-isogeny factors in the response

In this section, we explain how to factor $\sigma : E_1 \longrightarrow E_2$ by ℓ-isogenies, when the only thing we know is its kernel ideal I. We not only need to find the factors $\sigma_1 : E_1 \longrightarrow E_1'$ and $\sigma_2 : E_2 \longrightarrow E_2'$ but also alternate paths $\theta_1 : E_0 \longrightarrow E_1'$ and $\theta_2 : E_0 \longrightarrow E_2'$ of norm prime to ℓ to be able to evaluate $\sigma' : E_1' \longrightarrow E_2'$ with $\mathsf{EvalTorsion}_{\ell^f}$.

$$\begin{tikzcd}
E_1 \arrow[r, "\sigma_1"] & E_1' \arrow[rr, dashed, "\sigma'"] & & E_2' & E_2 \arrow[l, "\sigma_2"'] \\
E_0 \arrow[u, bend left, "\psi"] \arrow[u, bend right, "\psi'"'] \arrow[ur, "\theta_1"'] \arrow[urrr, "\theta_2"] \arrow[urrrr, "\rho_2'"'] \arrow[rrrr, bend left, "\tau"] \arrow[rrrr, bend right, "\tau'"] & & & & E_A \arrow[u, "\varphi"]
\end{tikzcd}$$

We start by factoring I to find the kernel ideals J_1 and J_2 of σ_1 and σ_2 of norm at most ℓ^f. Let us write $I := \ell^b m \cdot J$ where m is prime to ℓ and J a left $\mathcal{O}_1$-ideal without integer factor. Let us write $\mathrm{nrd}(J) := \ell^c m'$, with m' prime to

ℓ. Then $\mathrm{nrd}(I) = \ell^{2b+c}m^2m' < \ell^e$ so $2b+c \leq e \leq 2f$. We may write $b := b_1 + b_2$ and $c := c_1 + c_2$ with $2b_1 + c_1, 2b_2 + c_2 \leq f$. Then $J_1 := (mJ + \mathcal{O}_1\ell^{c_1})\ell^{b_1}$ and $J_2 := (m\overline{J} + \mathcal{O}_2\ell^{c_2})\ell^{b_2}$ have norms $\ell^{2b_1+c_1}|\ell^f$ and $\ell^{2b_2+c_2}|\ell^f$ respectively by the following lemma. Furthermore, we have $I := J_1I'\overline{J_2}$, where I' is a quaternion ideal of norm prime to ℓ by construction.

Lemma 33. *Let I be a left ideal of a maximal order $\mathcal{O} \subset \mathcal{B}_{p,\infty}$ of norm prime to p. Let us write $I := m \cdot J$ with $m \in \mathbb{N}^*$ and I' a left $\mathcal{O}$-ideal such that $J \not\subset n\mathcal{O}$ for all $n \in \mathbb{Z}$. Let $d \in \mathbb{N}^*$ prime to m and $K := I + d\mathcal{O}$. Then $\mathrm{nrd}(K) = d \wedge \mathrm{nrd}(J)$.*

Proof. Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve of endomorphism ring isomorphic to $\mathcal{O}$. Then:

$$\begin{aligned} E[K] &= E[mJ + d\mathcal{O}] = E[mJ] \cap E[d] = \{P \in E \mid \forall \alpha \in J, [m]\alpha(P) = 0\} \cap E[d] \\ &= \{P \in E \mid [m]P \in E[J]\} \cap E[d] = [m]^{-1}(E[J]) \cap E[d] \end{aligned}$$

Since J is not divisible by any integer, $E[J]$ is cyclic so we may consider a generator $P \in E$ of $E[J]$. Let $N' := \mathrm{nrd}(J)$ and $d' := d \wedge \mathrm{nrd}(J)$. Let $Q_0 := [N'/d']P$. Then, $[d]Q_0 = [d/d'][N']P = 0$ and $[m]Q_0 \in \langle P \rangle$ by construction, so that $Q_0 \in E[K]$. Conversely, let $Q \in E[K]$. Then $[m]Q = [k]P$ for some $k \in \mathbb{Z}$ and $[d]Q = 0$. In particular $[kd]P = [md]Q = 0$. Then, $N'|kd$ since P has order N', so that $N'/d'|k$, so we may write $k = k'N'/d'$ with $k' \in \mathbb{Z}$, so that $[m]Q = [k'N'/d']P = [k']Q_0$. Since m and d are coprime, there exists $u, v \in \mathbb{Z}$ such that $mu + dv = 1$ and we then have $Q = [mu + dv]Q = [um]Q = [uk']Q_0$. Hence, $E[K] = \langle Q_0 \rangle$ and finally

$$\mathrm{nrd}(K) = \#E[K] = \#\langle Q_0 \rangle = \#\langle [N'/d']P \rangle = d' = d \wedge \mathrm{nrd}(J).$$

□

Knowing J_1 and J_2, we can then compute their associated isogenies σ_1 and σ_2. Since J_1 and J_2 have norm dividing ℓ^f, $E_1[J_1]$ and $E_2[J_2]$ are contained in the accessible ℓ^f-torsion. So we only have to evaluate a basis of J_1 and J_2 on the ℓ^f-torsion and solve discrete logarithms in groups of exponent ℓ^f to compute $E_1[J_1]$ and $E_2[J_2]$. We can then apply Vélu's formulas [Vél71] to compute σ_1 and σ_2. To obtain a basis of J_1 that we can evaluate on the ℓ^f-torsion, we compute a T-eval-basis of J_1 in the sense of Definition 26 by expressing the basis of J_1 that we already know as integer linear combinations of a T-eval-basis $\mathcal{B}_1 := \mathsf{PushEndRing}(\psi, I_\psi)$ of $\mathrm{End}(E_1)$ obtained via Algorithm 8. The same principle applies to J_2. Let $\rho_2 := \varphi \circ \tau$ and $I_2 := I_\tau \cdot I_\varphi$ its kernel ideal. Then, we can obtain a T-eval-basis $\mathcal{B}_2 := \mathsf{PushEndRing}(\rho_2, I_2)$ yielding a T-eval-basis of J_2.

Now we explain how to find alternate paths $\theta_1 : E_0 \longrightarrow E_1'$ and $\theta_2 : E_0 \longrightarrow E_2'$ of degree prime to ℓ. First, we find left $\mathcal{O}_0$-ideals $K_1 \sim I_\psi \cdot J_1$ and $K_2 \sim I_2 \cdot J_2$ of powersmooth norm prime to ℓ using the KLPT algorithm [KLPT14]. To translate K_1 and K_2 into isogenies θ_1 and θ_2, we could use the paths $\sigma_1 \circ \tau$ and $\sigma_2 \circ \rho_2$ (where $\rho_2 = \varphi \circ \psi$) and apply $\mathsf{SpecialIdealToIsogeny}$ (presented in

```
Algorithm 17: $\mathsf{FactorIsogeny}_{\ell^f,T}$
Data: A quaternion ideal $I$ of norm $< \ell^e$ connecting $\mathcal{O}_1 \cong \mathrm{End}(E_1)$ and $\mathcal{O}_2 \cong \mathrm{End}(E_2)$, two isogenies $\psi, \psi' : E_0 \longrightarrow E_1$ of degrees dividing $T^2$ and a power of $\ell$ respectively, $\rho_2 : E_0 \longrightarrow E_2$ of degree dividing a power of $T$ and $I_\psi, I_{\psi'}, I_2$ their respective kernel ideals.
Result: Two left-ideals $J_1 \subseteq \mathcal{O}_1$ and $J_2 \subseteq \mathcal{O}_2$ whose norms divide $\ell^f$ such that $I := J_1 I' \overline{J_2}$, with $I'$ of norm prime to $\ell$, two ideals $K_1 \sim I_\psi \cdot J_1$ and $K_2 \sim I_2 \cdot J_2$ of norms dividing $T^2$ along with isogenies $\sigma_1 : E_1 \longrightarrow E_1'$, $\sigma_2 : E_2 \longrightarrow E_2'$, $\theta_1 : E_0 \longrightarrow E_1'$ and $\theta_2 : E_0 \longrightarrow E_2'$ respectively associated to $J_1, J_2, K_1$ and $K_2$.
1  Factor $I := \ell^b m J$ with $\ell \wedge m = 1$ and $I'$ without integer factors and factor $\mathrm{nrd}(J) := \ell^c m'$ with $\ell \wedge m' = 1$;
2  Let $b := b_1 + b_2$ and $c := c_1 + c_2$ with $2b_i + c_i \leq f$ for $i \in \{1, 2\}$;
3  $J_1 \longleftarrow (mJ + \mathcal{O}_1 \ell^{c_1})\ell^{b_1}$, $J_2 \longleftarrow (m\overline{J} + \mathcal{O}_2 \ell^{c_2})\ell^{b_2}$;
4  $I' \longleftarrow J_1^{-1} I \overline{J_2}^{-1}$;
5  Compute two $T$-eval-basis $\mathcal{B}_1 := \mathsf{PushEndRing}(\psi, I_\psi)$ and $\mathcal{B}_2 := \mathsf{PushEndRing}(\rho_2, I_2)$;
6  Infer $T$-eval-basis $\mathcal{C}_i$ of $J_i$ from $\mathcal{B}_i$ for $i \in \{1, 2\}$;
7  Evaluate $\mathcal{C}_i$ on a basis of $E_i[\ell^f]$ to compute $G_i := E_i[J_i]$ for $i \in \{1, 2\}$;
8  Compute $\sigma_i : E_i \longrightarrow E_i'$ of kernel $G_i$ for $i \in \{1, 2\}$;
9  $K_1 \longleftarrow \mathsf{KLPT}_{T^2}(I_\psi \cdot J_1)$, $K_2 \longleftarrow \mathsf{KLPT}_{T^2}(I_2 \cdot J_2)$;
10 $\theta_1 \longleftarrow \mathsf{SpecialIdealToIsogeny}(K_1, I_{\psi'} \cdot J_1, \sigma_1 \circ \psi')$ ([DKLPW20, Algorithm 7] presented in Appendix A.4);
11 $I_2' \longleftarrow \mathsf{KLPT}_{\ell^h}(I_2)$;
12 Compute $\rho_2'$ of kernel ideal $I_2'$ using [DKLPW20, Algorithm 9];
13 $\theta_2 \longleftarrow \mathsf{SpecialIdealToIsogeny}(K_2, I_2' \cdot J_2, \sigma_2 \circ \rho_2')$;
14 Return $J_1, J_2, I', K_1, K_2, \sigma_1, \sigma_2, \theta_1, \theta_2$;
```

Appendix A.4) but $\mathrm{nrd}(K_1)$ and $\mathrm{nrd}(K_2)$ would need to be prime to T. We could use powersmooth torsion coprime with T and ℓ and still compute θ_1 and θ_2 in polynomial time but this would not be optimal. Instead, we propose to seek K_1 and K_2 of norm dividing $T^2 \simeq p^3$ and to use paths $\sigma_1 \circ \psi'$ and $\sigma_2 \circ \rho_2'$ in SpecialIdealToIsogeny, where $\psi' : E_0 \longrightarrow E_1$ and $\rho_2' : E_0 \longrightarrow E_2$ are isogenies of degree a power of ℓ.

The input ψ' is obtained as a by-product of the challenge generation. We can simply run Algorithm 12 completely to obtain $\psi, \psi' : E_0 \longrightarrow E_1$ at the same time.

To find ρ_2', we apply KLPT to the kernel ideal $I_2 := I_\tau \cdot I_\varphi$ of $\rho_2 := \varphi \circ \tau$, to find $I_2' \sim I_2$ of norm $\ell^h \simeq p^3$. We can then translate the ideal I_2' into its associated isogeny ρ_2' via the effective Deuring correspondence algorithm introduced in the original SQIsign paper [DKLPW20, Algorithm 9]. We summarize all the computations to factor σ in the $\mathsf{FactorIsogeny}_{\ell^f,T}$ algorithm (Algorithm 17).

C.2 Adaptations of the response and verification when q is not prime to ℓ

Keeping the notations of the previous section, assume we have factored $\sigma := \widehat{\sigma_2} \circ \sigma' \circ \sigma_1$. Then, we can embed σ' in an isogeny F of dimension 8 using the same techniques presented earlier since $q' := \deg(\sigma')$ has degree prime to ℓ. To proceed, we evaluate σ' on a canonically generated basis (P_1', P_2') of $E_1'[\ell^f]$ using the isogeny paths $\theta_1 : E_0 \longrightarrow E_1'$ and $\theta_2 : E_0 \longrightarrow E_2'$ of degree dividing T^2 to apply $\mathsf{EvalTorsion}_{\ell^f}$ (Algorithm 11). Once all these computations are done, the prover simply sends $(\sigma_1, \sigma_2, \sigma'(P_1'), \sigma'(P_2'), q')$ to the verifier. The complete $\mathsf{RigorousRespond}$ procedure follows (Algorithm 18).

The complete verification procedure $\mathsf{RigorousVerify}$ (Algorithm 19) is very similar to the original one. Indeed, representing σ' and representing $\sigma = \widehat{\sigma_2} \circ \sigma' \circ \sigma_1$ is equivalent when σ_1 and σ_2 are known.

As inputs of Algorithms 18 and 19, recall the following notations:

- $\mathsf{RigorousSetup}$, the public parameters of RigorousSQIsignHD, p, ℓ and f such that $\ell^f | p-1$, e, T powersmooth (accessible torsion) and E_0;
- $\mathsf{SecretKey}$, the isogenies $\tau, \tau' : E_0 \longrightarrow E_A$ of degrees $D_\tau | T^2$ and $D_{\tau'} = \ell^h$ respectively along with their kernel ideals I_τ and $I_{\tau'}$;
- $\mathsf{CommitData}$, two isogenies $\psi, \psi' : E_0 \longrightarrow E_1$ of degree $D_\psi | T^2$ and $D_{\psi'} = \ell^h$ respectively along with their kernel ideals I_ψ and $I_{\psi'}$;
- $\mathsf{ChallData}$, the isogeny $\varphi : E_A \longrightarrow E_2$ of degree $D_\varphi | T$.

Algorithm 18: $\mathsf{RigorousRespond}$

```
Data: $\mathsf{RigorousSetup}$, $\mathsf{SecretKey}$, $\mathsf{CommitData}$, $\mathsf{ChallData}$.
Result: $(\sigma_1, \sigma_2, \sigma'(P_1'), \sigma'(P_2'), q')$, where $\sigma_1 : E_1 \longrightarrow E_1'$ and $\sigma_2 : E_2 \longrightarrow E_2'$
        are isogenies of degree dividing $\ell^f$, $\sigma' : E_1' \longrightarrow E_2'$ is an isogeny of
        degree $q' < \ell^e$ prime to $\ell$ and $(P_1', P_2')$ is a canonically determined
        basis of $E_1'[\ell^f]$.
$I_\varphi \longleftarrow \mathsf{IsogenyToIdeal}(\varphi, \tau', I_{\tau'})$;
$J \longleftarrow \overline{I_\psi} \cdot I_\tau \cdot I_\varphi$;
$I \longleftarrow \mathsf{RandomEquivalentIdeal}_{\ell^e}(J)$ and $q \longleftarrow \mathrm{nrd}(I)$;
$J_1, I', J_2, K_1, K_2, \sigma_1, \sigma_2, \theta_1, \theta_2 \longleftarrow$
    $\mathsf{FactorIsogeny}_{\ell^f, T}(I, \psi, \psi', \varphi \circ \tau, I_\psi, I_{\psi'}, I_\tau \cdot I_\varphi)$;
$q' \longleftarrow q / \mathrm{nrd}(J_1)\,\mathrm{nrd}(J_2)$;
Compute the canonical basis $(P_1', P_2')$ of $E_1'[\ell^f]$;
$(\sigma'(P_1'), \sigma'(P_2')) \longleftarrow \mathsf{EvalTorsion}_{\ell^f}(I', P_1', P_2', \theta_1, \theta_2, K_1, K_2)$;
Return $(\sigma_1, \sigma_2, \sigma'(P_1'), \sigma'(P_2'), q')$;
```

C.3 Impact on compactness in dimension 8

When q is not prime to ℓ, the factors σ_1 et σ_2 are transmitted in the signature in addition to the data $(E_1, \sigma'(P_1'), \sigma'(P_2'), q')$. This is apparently more information

Algorithm 19: RigorousVerify

```
Data: RigorousSetup, ChallData, an output from RigorousRespond,
      $(\sigma_1, \sigma_2, R_1', R_2', q')$, where $\sigma_1 : E_1 \longrightarrow E_1'$ and $\sigma_2 : E_2 \longrightarrow E_2'$ are
      isogenies of degree dividing $\ell^f$, $R_1', R_2' \in E_2'[\ell^f]$ and $q' \in \mathbb{N}^*$.
Result: 1 if $(\sigma_1, \sigma_2, R_1', R_2', q')$ is a valid response and 0 otherwise.
if $q' \deg(\sigma_1) \deg(\sigma_2) > \ell^e$ then
    Return 0;
end
Compute the canonical basis $(P_1', P_2')$ of $E_1'[\ell^f]$;
Find $a_1, \cdots, a_4 \in \mathbb{Z}$ such that $a_1^2 + \cdots + a_4^2 + q = \ell^e$ using Pollack and
  Treviño's algorithm [PT18];
$F' \longleftarrow \mathsf{RepresentIsogeny}_{8,\ell^e}(q', a_1, \cdots, a_4, P_1', P_2', R_1', R_2')$;
Return $\mathsf{IsValid}_8(F', E_1', E_2', a_1, \cdots, a_4)$;
```

than in the case $q \wedge \ell = 1$. However, we can optimize the communications to avoid any compactness loss.

We may write $\deg(\sigma_1) := \ell^{f_1}$ and $\deg(\sigma_2) := \ell^{f_2}$, with $f_1, f_2 \leq f$, so that $q' = q/\ell^{f_1+f_2} < \ell^{e'}$, where $e' := e - f_1 - f_2$. Hence, we can represent σ' by an $\ell^{e'}$-isogeny F' in dimension 8. By Remark 4.2, we only need to evaluate the ℓ^{f_3}-torsion by σ', where $2f_3 \geq e' + 4$. Hence the points P_1' and P_2' may form a basis of $E_1'[\ell^{f_3}]$ instead of $E_1'[\ell^f]$ and we can represent $\sigma'(P_1')$ and $\sigma'(P_1')$ with $3f_3$ bits by the techniques of Section 6.1 (assuming $\ell = 2$).

To represent σ_1, we may factor $\sigma_1 := [\ell^{b_1}] \circ \sigma_1'$, where $2b_1 \leq f_1$ and $\sigma_1' : E_1 \longrightarrow E_1'$ a cyclic $\ell^{f_1'}$-isogeny with $f_1' := f_1 - 2b_1$. So we may represent σ_1 by the integer b_1 and $\ker(\sigma_1') \subset E_1[\ell^{f_1'}]$. Let (Q_1, Q_2) be a canonical basis of $E_1[\ell^{f_1'}]$. Then, $\ker(\sigma_1')$ is generated by either one of the points $Q_1 + kQ_2$ with $0 \leq k \leq \ell^{f_1'} - 1$ or one of the points $\ell k' Q_1 + Q_2$ with $0 \leq k' \leq \ell^{f_1'-1} - 1$. Hence, $\ker(\sigma_1')$ can be represented by $f_1' + 1$ bits (one bit to tell which form takes $\ker(\sigma_1')$ and f_1' bits for k or k'). Since the number of bits to represent b_1 is very small ($O(\log(b_1))$), we may represent σ_1 by at most $f_1 + 1$ bits, and similarly, we may represent σ_2 by at most $f_2 + 1$ bits.

As in Section 6.1, we represent $q' < \ell^{e'}$ with e' bits and E_1 with 4λ bits (where λ is the security level, satisfying $p \simeq 2^\lambda$). Hence, the signature size is

$$\begin{aligned} 3f_3 + f_1 + f_2 + 2 + 4\lambda + e' &= \frac{5}{2}(e - f_1 - f_2) + f_1 + f_2 + 4\lambda + O(\log(\lambda)) \\ &< \frac{5}{2}e + 4\lambda + O(\log(\lambda)) = 14\lambda + O(\log(\lambda)) \end{aligned}$$

in bits, so we do not suffer any communication loss compared to the case $q \wedge \ell = 1$ (as the inequality indicates). However, since $e \simeq 4\lambda$ ($\ell^e = \Theta(p^2)$), RigorousSQIsignHD signatures are much less compact than FastSQIsignHD signatures (14λ instead of $13/2\lambda$ bits).

D Security analysis of RigorousSQIsignHD

To prove the security of RigorousSQIsignHD, we proceed as in Section 5, namely, we prove completeness, knowledge soundness and zero-knowledge properties. Completeness is a consequence of Corollary 32.

In Appendix D.1, we prove the knowledge soundness of RigorousSQIsignHD. Our previous straightforward special soundness argument used in Section 5.1 does not apply anymore because we have no guarantee on $q = \deg(\sigma)$ in general. For that reason, we need to come back to the formal definition of knowledge soundess given in [HL10, Definition 6.3.1].

In Appendix D.2, we prove the zero-knowledge property provided the simulator can access an oracle which is more natural than the *ad-hoc* one introduced in Section 5.2 and without any additional heuristic assumption. RigorousSQIsignHD was designed to prove rigorously this zero-knowledge property.

D.1 Knowledge soundness of RigorousSQIsignHD

We recall the formal definition of knowledge soundness given in [HL10, Definition 6.3.1].

Definition 34. A protocol (P, V) between a prover and a verifier is a *proof of knowledge* for a relation $\mathcal{R} \subset X \times W$ with knowledge error κ if it satisfies the following properties:

Completeness: If P interacts with V as input $x \in X$ and private input $w \in W$ with $(x, w) \in \mathcal{R}$, then V always accepts.

Knowledge soundness: There exists a *knowledge extractor* K such that for every interactive prover P^* and every $x \in X$, K satisfies the following condition. Let $\varepsilon(x)$ be the success probability of P^* on input x (the probability that V accepts on input x). If $\varepsilon(x) > \kappa(x)$, then upon input x and oracle access to P^*, K outputs a witness $w \in W$ such that $(x, w) \in \mathcal{R}$ within an expected number of steps $O(1/(\varepsilon(x) - \kappa(x)))$.

Definition 35. A 3-round protocol (commitment, challenge, response) (P, V) satisfies *special soundness* for a relation $\mathcal{R} \subset X \times W$ if given $x \in X$ and two accepting transcripts $(a, c, r), (a, c', r')$ for $x \in X$ with the same commitment a and distinct challenges $c \neq c'$, one can extract a witness $w \in W$ such that $(x, w) \in \mathcal{R}$ in polynomial time.

Theorem 36. *[HL10, Theorem 6.3.2] A complete 3-round protocol satisfying special soundness for a relation $\mathcal{R}$ with challenge space $\mathcal{C}$ is a proof of knowledge with knowledge error $1/\#\mathcal{C}$.*

In Proposition 17, we proved that FastSQIsignHD satisfies special soundness for the relation:

$$\mathcal{R} := \{(E_A, \alpha) \mid \alpha \in \mathrm{End}(E_A) \text{ non-scalar}\}.$$

Since the challenge space has size $\mu(\ell'^{f'}) = \ell'^{f'-1}(\ell'+1) = \Omega(p^{1/2})$, we get by Theorem 36 that FastSQIsignHD is a proof of knowledge for $\mathcal{R}$ with knowledge error $O(p^{-1/2})$.

Unfortunately, the special soundness argument no longer holds in RigorousSQIsignHD because we can no longer impose conditions on q (except $q < \ell^e$), and especially we cannot impose q to be prime to D_φ. However, choosing D_φ big enough will ensure that the endomorphism α is non-scalar with overwhelming probability, since $\varphi' \circ \widehat{\varphi}$ has a big cyclic factor with overwhelming probability. We first introduce a useful lemma to prove this result.

Lemma 37. *Let $\phi : E_1 \longrightarrow E_2$ and $\phi' : E_1 \longrightarrow E_2'$ be two cyclic isogenies. Let $\phi_0 : E_1 \longrightarrow E_3$ be the cyclic separable isogeny such that* $\ker(\phi_0) = \ker(\phi) \cap \ker(\phi')$*. Then, there exists two cyclic isogenies $\phi_1 : E_3 \longrightarrow E_2$ and $\phi_1' : E_3 \longrightarrow E_2'$ such that $\phi = \phi_1 \circ \phi_0$, $\phi' = \phi_1' \circ \phi_0$ and $\phi_1' \circ \widehat{\phi_1}$ is cyclic. ϕ_0 will be called the* greatest common factor *of ϕ and ϕ'.*

Proof. We can always factor $\phi = \phi_1 \circ \phi_0$, $\phi' = \phi_1' \circ \phi_0$, with $\phi_0 : E_1 \longrightarrow E_3$ such that $\ker(\phi_0) = \ker(\phi) \cap \ker(\phi')$.

First, we prove that $\ker(\phi_1) \cap \ker(\phi_1') = \{0\}$. If it was not the case, we could find a cyclic isogeny $\psi : E_2 \longrightarrow E_4$ with non-trivial kernel such that ϕ_1 and ϕ_1' factor through ψ, and both ϕ and ϕ' would factor through $\psi \circ \phi_0$, so we would have $\ker(\psi \circ \phi_0) \subseteq \ker(\phi) \cap \ker(\phi')$. But $\ker(\psi \circ \phi_0) = \phi_0^{-1}(\ker(\psi)) \supsetneq \ker(\phi_0)$ since $\ker(\psi)$ is non-trivial. Contradiction.

Now, we prove that $\phi_1' \circ \widehat{\phi_1}$ is cyclic. Actually, it suffices to prove it when $\deg(\phi_1')$ and $\deg(\phi_1)$ are powers of the same prime ℓ. Indeed, if not, we can decompose $\phi_1 = \psi_2 \circ \psi_1$ and $\phi_1' = \psi_2' \circ \psi_1'$ with $\deg(\psi_1)\deg(\psi_1')$ coprime with $\deg(\psi_2)\deg(\psi_2')$. Then, we may write $\phi_1' \circ \widehat{\phi_1} = \psi_2' \circ \psi_3 \circ \widehat{\psi_2}$, with $\psi_3 = \psi_1' \circ \widehat{\psi_1}$. Using pushforward isogenies (as defined in Definition 8), we may write $[\psi_2]_*\psi_3 \circ \psi_2 = [\psi_3]_*\psi_2 \circ \psi_3$, so that $\psi_3 \circ \widehat{\psi_2} = \widehat{[\psi_3]_*\psi_2} \circ [\psi_2]_*\psi_3$ and

$$\phi_1' \circ \widehat{\phi_1} = \psi_2' \circ \widehat{[\psi_3]_*\psi_2} \circ [\psi_2]_*\psi_3.$$

If we assume that ψ_3 is cyclic then $[\psi_2]_*\psi_3$ is cyclic. Besides, ψ_2' and $[\psi_3]_*\psi_2$ are cyclic and $\ker(\psi_2') \cap \ker([\psi_3]_*\psi_2) = \{0\}$. Indeed, if $P \in \ker(\psi_2') \cap \ker([\psi_3]_*\psi_2)$, then $P \in \ker([\psi_3]_*\psi_2) = \psi_3(\ker(\psi_2))$ so we may write $P = \psi_3(Q)$ with $Q \in \ker(\psi_2)$. Let $R := \widehat{\psi_1}(Q)$. Then, $\phi_1(R) = \psi_2 \circ \psi_1 \circ \widehat{\psi_1}(Q) = [\deg(\psi_1)]\psi_2(Q) = 0$ and $\phi_1'(R) = \psi_2'(P) = 0$, so $R \in \ker(\phi_1) \cap \ker(\phi_1') = \{0\}$ and $R = 0$, so $P = \psi_1'(R) = 0$.

Since the product of cyclic isogenies of coprime degrees is cyclic, it follows by induction that we only have to prove that $\ker(\phi_1) \cap \ker(\phi_1') = \{0\}$ implies the cyclicity of $\phi_1' \circ \widehat{\phi_1}$ when $\deg(\phi_1')$ and $\deg(\phi_1)$ are powers of the same prime ℓ. We assume that in the following.

We proceed by induction on $\deg(\phi_1')$. When $\deg(\phi_1') = 1$ it follows from the fact that the dual of a cyclic isogeny is cyclic. Now, we assume the result holds when $\deg(\phi_1') = \ell^n$ with $n \in \mathbb{N}$ and prove it holds when $\deg(\phi_1') = \ell^{n+1}$. We may factor $\phi_1' := \phi_2 \circ \phi_2'$ with $\deg(\phi_2) = \ell$ and $\deg(\phi_2') = \ell^n$. By assumption,

$\phi_3 := \phi_2' \circ \widehat{\phi_1}$ is cyclic so we only have to prove that $\phi_2 \circ \phi_3$ is cyclic, *i.e.* that $\ker(\phi_2 \circ \phi_3) = \phi_3^{-1}(\ker(\phi_2))$ is cyclic.

Let Q be a generator of $\ker(\phi_2)$, P be a generator of $\ker(\phi_3)$ and $P' \in E_2$ such that $Q = \phi_3(P')$. Then

$$\ker(\phi_2 \circ \phi_3) = \phi_3^{-1}(\ker(\phi_2)) = \langle P, P' \rangle.$$

To conclude, it suffices to prove that $P \in \langle P' \rangle$. We have $P' \in \ker(\phi_2 \circ \phi_3) \subset E_2[\ell^{m+1}]$, with $\deg(\phi_3) := \ell^m$ and $[\ell^m]P' = \widehat{\phi_3} \circ \phi_3(P') = \widehat{\phi_3}(Q)$ and $\widehat{\phi_3}(Q) \neq 0$. Indeed, if $\widehat{\phi_3}(Q) = 0$, we may write $S := \widehat{\phi_2'}(Q)$, so that $\phi_1(S) = \widehat{\phi_3}(Q) = 0$ and $\phi_1'(S) = \phi_2 \circ \phi_2' \circ \widehat{\phi_2'}(Q) = [\ell^n]\phi_2(Q) = 0$, so that $S \in \ker(\phi_1) \cap \ker(\phi_1') = \{0\}$ and $\widehat{\phi_2'}(Q) = 0$. Hence, ϕ_2' factors through ϕ_2 and ϕ_1' factors through $[\ell] = \phi_2 \circ \widehat{\phi_2}$ and is not cyclic. Contradiction. Hence, $\widehat{\phi_3}(Q) \neq 0$ and P' has order ℓ^{m+1}.

Let $R \in E_2[\ell^m]$ such that $([\ell]P', R)$ is a basis of $E_2[\ell^m]$. Then, we may write $P := [a\ell]P' + [b]R$ for some $a, b \in \mathbb{Z}$ since $P \in \ker(\phi_3) \subset E_2[\ell^m]$. Since $Q \in \ker(\phi_2)$ has order ℓ, we get that

$$0 = \phi_3(P) = [a\ell]Q + [b]\phi_3(R) = [b]\phi_3(R),$$

and that $\phi_3(R)$ generates $\phi_3(E_2[\ell^m]) = \ker(\widehat{\phi_3})$, which is cyclic so it has order ℓ^m. It follows that $b \equiv 0 \mod \ell^m$, so that $P = [a\ell]P' \in \langle P' \rangle$. This completes the proof. □

Lemma 38. *Let* $(E_1, \varphi, R_1, R_2, q)$ *and* $(E_1, \varphi', R_1', R_2', q')$ *be two* Rigorous-SQIsignHD *transcripts with the same commitment* E_1. *If the greatest common factor of* φ *and* φ' *has degree* $< D_\varphi/\ell^e$, *then we can infer an efficient representation of a non-scalar endomorphism* $\alpha \in \mathrm{End}(E_A)$ *from these transcripts. In this case we say that* φ *and* φ' *are* relatively good *and* relatively bad *if this is not satisfied.*

Proof. Let σ and σ' be respectively the isogenies defined on E_1 represented by (R_1, R_2, q) and (R_1', R_2', q') and $\alpha := \widehat{\varphi'} \circ \sigma' \circ \widehat{\sigma} \circ \varphi \in \mathrm{End}(E_A)$. As explained in the proof of Proposition 17, we can compute an efficient representation of α in polynomial time in $\log(p)$. Assume that α is a scalar endomorphism: $\alpha = [\lambda]_{E_A}$ for some $\lambda \in \mathbb{Z}$. Then $[\lambda]_{E_2} = \varphi \circ \widehat{\varphi'} \circ \sigma' \circ \widehat{\sigma}$. Let us write $\varphi := \varphi_1 \circ \varphi_0$ and $\varphi := \varphi_1' \circ \varphi_0$, where φ_0 is the greatest common factor of φ and φ'. Then $\phi := \varphi_1 \circ \widehat{\varphi_1'}$ is cyclic by Lemma 37 and we have $\varphi \circ \widehat{\varphi'} := [D]\phi$ with $D := \deg(\varphi_0)$. We can also write $\sigma \circ \widehat{\sigma'} = [D']\phi'$ where ϕ' is a cyclic isogeny $E_2' \longrightarrow E_2$. It follows that $[\lambda/DD']_{E_2'} = \phi \circ \widehat{\phi'}$. Hence, by Lemma 37, the greatest common factor of ϕ and ϕ' must be equal to both ϕ and ϕ' so $\phi = \phi'$. Hence, $\sigma \circ \widehat{\sigma'}$ factors through ϕ. But $\deg(\sigma \circ \widehat{\sigma'}) = qq' \leq \ell^{2e}$ and $\deg(\phi) = D_\varphi^2/D^2$ with $D < D_\varphi/\ell^e$ so $\deg(\phi) > \ell^{2e}$. Contradiction. □

Now, we prove that the probability to generate *relatively good* challenges is overwhelming. This will be the last essential ingredient to our knowledge soundness proof.

Lemma 39. *Fix a challenge* $\varphi : E_A \longrightarrow E_2$ *and let us write* $D_\varphi := \prod_{i=1}^r \ell_i^{e_i}$, *where* $\ell_1 \leq \cdots \leq \ell_r$ *are distinct ordered primes and* $e_1, \cdots, e_r \in \mathbb{N}^*$. *Then, the number of challenges* $\varphi' : E_A \longrightarrow E_2'$ *relatively bad to* φ *is*

$$O\left(\frac{\ell^e \mu(D_\varphi)}{D_\varphi^{1-\log(2)/\log\log(D_\varphi)}}\right),$$

with $\mu(D_\varphi) := \prod_{i=1}^r \ell_i^{e_i-1}(\ell_i+1)$.

Proof. φ and φ' relatively bad if their greatest common factor has degree $D \geq D_\varphi/\ell^e$. If we fix such a $D|D_\varphi$, then choosing φ' is choosing a cyclic isogeny of degree D_φ/D so there are $\mu(D_\varphi/D)$ possibilities. It follows that the number of challenges φ' relatively bad to φ is

$$\begin{aligned} N &\leq \sum_{\substack{D|D_\varphi \\ D>D_\varphi/\ell^e}} \mu\left(\frac{D_\varphi}{D}\right) = \mu(D_\varphi) \sum_{\substack{D|D_\varphi \\ D>D_\varphi/\ell^e}} \frac{1}{\mu(D)} \leq \mu(D_\varphi) \sum_{\substack{D|D_\varphi \\ D>D_\varphi/\ell^e}} \frac{1}{D} \\ &\leq \frac{\ell^e \mu(D_\varphi)}{D_\varphi} \#\{D \in \mathbb{N}^* \mid D|D_\varphi \text{ and } D > D_\varphi/\ell^e\} \\ &\leq \frac{\ell^e \mu(D_\varphi)}{D_\varphi} \#\{D \in \mathbb{N}^* \mid D|D_\varphi\} = \frac{\ell^e \mu(D_\varphi) d(D_\varphi)}{D_\varphi}, \end{aligned}$$

where $d(D_\varphi)$ is the number of divisors of $d(D_\varphi)$. By [HW08, § 18.1, Theorem 317], we know that $d(D_\varphi) = O\left(D_\varphi^{\log(2)/\log\log(D_\varphi)}\right)$. The result follows. □

Since $\ell^e = O(p^2)$, we choose $D_\varphi|T$ such that $D_\varphi \simeq p^{5/(2-2\log(2)/\log\log(p^{5/2}))}$, so that the proportion of challenges relatively bad to a given challenge is $O(p^{-1/2})$. This means in practice $D_\varphi \simeq p^{2.82}$ when p has size 256 bits (to achieve $\lambda = 128$ bits of classical security). Hence the choice $T \simeq p^3$. Then, under this condition, we can adapt the proof of [HL10, Theorem 6.3.2] to prove knowledge soundness of RigorousSQIsignHD.

Proposition 40. *Assume that* $D_\varphi > p^{5/(2-2\log(2)/\log\log(p^{5/2}))}$. *Then the* RigorousSQIsignHD *identification protocol is a* proof of knowledge *for the relation* $\mathcal{R}$ *of Proposition 17 with knowledge error* $O(p^{-1/2})$.

Proof. As required by Definition 34, we construct a knowledge extractor K. Let P^* be a prover with success probability ε. Then K is constructed as follows (as in [HL10, Theorem 6.3.2]). Fix E_A a supersingular elliptic curve (as public key). Then K executes the following algorithm:

1. Sample a seed $s \overset{\$}{\longleftarrow} \{0,1\}^*$ fixing the randomness of P^*, sample a challenge φ and run $P^*(E_A, s, \varphi)$ repeatedly until the transcript $(E_1, \varphi, R_1, R_2, q)$ outputted by P^* is accepted by the verifier and save s.
2. Sample another challenge φ' and run $P^*(E_A, s, \varphi')$ with the same seed s as in step 1 (fixing the commitment value E_1) to obtain a new transcript $(E_1, \varphi', R_1', R_2', q')$ and repeat until we can extract a witness $\alpha \in \mathrm{End}(E_A)$ non-scalar from $(E_1, \varphi, R_1, R_2, q)$ and $(E_1, \varphi', R_1', R_2', q')$.

3. Break step 2 after k iterations (to be determined) or return α.

This algorithm may fail so K may execute this algorithm multiple times. We determine k to optimize the running time and the probability of failure of this algorithm. To do this, we specify how we can extract a witness in step 2. As in the previous knowledge soundness proof, (R_1, R_2, q) and (R_1', R_2', q') respectively provide an efficient representation of a q-isogeny $\sigma : E_1 \longrightarrow E_2$ and a q'-isogeny $\widehat{\sigma'} : E_2 \longrightarrow E_1$, so we have an efficient representation of $\alpha := \widehat{\varphi'} \circ \sigma' \circ \widehat{\sigma} \circ \varphi \in \mathrm{End}(E_A)$. If φ and φ' are relatively good, then α is non-scalar by Lemma 38 and we have won.

By Lemma 39, since $D_\varphi > p^{5/2(1-\log(2)/\log\log(p^{5/2}))}$ by assumption, the number of challenges φ' relatively bad to φ is bounded by $C\mu(D_\varphi)/\sqrt{p}$ for some constant $C > 0$.

Now consider the matrix H whose rows are indexed by seeds s for P^*, whose columns are indexed by challenges φ and such that $H(s, \varphi)$ is the the result 0 or 1 returned by the verifier when P^* is run with E_A, s and φ. By assumption, the proportion of 1 in H is ε. A row with a proportion of 1 bigger than $\varepsilon/2$ is called a *heavy row*. Let R be the number of rows in H (*i.e.* the number of possible seeds for P^*). Let R' be the number of non-heavy rows. Then, the number of 1 located in a heavy rows is bigger than:

$$R\varepsilon\mu(D_\varphi) - R'\frac{\varepsilon}{2}\mu(D_\varphi) \geq R\varepsilon\mu(D_\varphi) - R\frac{\varepsilon}{2}\mu(D_\varphi) = R\frac{\varepsilon}{2}\mu(D_\varphi)$$

so at least half of the 1 are in heavy rows and the probability to fall in a heavy row at step 1 of the algorithm is $\geq 1/2$. Let φ be the challenge found at step 1. Now, at step 2, we are in the same row as in step 1 (since we fixed s). Assuming we are in a heavy row, the probability to find φ' that is not relatively bad to φ and such that $H(s, \varphi') = 1$ is

$$P \geq \frac{\varepsilon/2\mu(D_\varphi) - \mu(D_\varphi)C/\sqrt{p}}{\mu(D_\varphi)} = \frac{\varepsilon}{2} - \frac{C}{\sqrt{p}}.$$

In the following, we assume that $\varepsilon > 2C/\sqrt{p}$, so that $P > 0$. Then, the expected number of tries t to succeed in step 2 is:

$$\mathbb{E}(t) = \frac{1}{P} \leq \frac{2}{\varepsilon - 2C/\sqrt{p}}.$$

Now we choose the time limit k accordingly. By Markov's inequality, the probability that step 2 terminates within k tries is

$$\mathbb{P}(t < k) = 1 - \mathbb{P}(t \geq k) \geq 1 - \frac{\mathbb{E}(t)}{k} \geq 1 - \frac{2}{k(\varepsilon - 2C/\sqrt{p})}$$

We choose $k := \lceil 4/(\varepsilon - 2C/\sqrt{p}) \rceil$, so that $\mathbb{P}(t < k) \geq 1/2$. This probability is conditional to the fact that we fall into a heavy row, which has probability $\geq 1/2$ as we saw. Hence, the probability that the algorithm succeeds is $\geq 1/2 \times 1/2 = 1/4$ so K expects to repeat it 4 times to find a witness.

Now we estimate the running time of the algorithm. Step 1 is expected to terminate after $1/\varepsilon$ iterations and step 2 after $k = \lceil 4/(\varepsilon - 2C/\sqrt{p}) \rceil$ iterations, so the total time complexity is

$$O\left(\frac{1}{\varepsilon} + \frac{4}{\varepsilon - 2C/\sqrt{p}}\right) = O\left(\frac{5}{\varepsilon - 2C/\sqrt{p}}\right).$$

RigorousSQIsignHD being complete, we conclude that it is a proof of knowledge for $\mathcal{R}$ with knowledge error $\kappa := 2C/\sqrt{p} = O(p^{-1/2})$. □

D.2 Rigorous zero-knowledge property

To prove the zero-knowledge property of RigorousSQIsignHD, we introduce an oracle generating uniformly random isogenies of bounded degree without any other constraint on the degree or assumption on the distribution of the codomain, unlike the RUGDIO introduced for FastSQIsignHD in Section 5.2. Hence this oracle is more natural than the previous one. However, as previously, to our knowledge, there is no efficient algorithm implementing such an oracle and it is believed that access to such an oracle does not affect the hardness of the underlying problem (the endomorphism ring problem, see Section 5.3).

Definition 41. A *random any degree isogeny oracle* (RADIO) is an oracle taking as input a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$ and returning an efficient representation of an isogeny $\sigma : E \longrightarrow E'$, which is uniformly random among the isogenies of degree $q < \ell^e$ with domain E.

To prove zero-knowledge, not only the distribution of the isogeny should be uniform, but also the distribution of its codomain. Unfortunately, the RADIO does not determine the distribution of codomains. The following theorem (proved in Appendix E.6) solves this issue when ℓ^e is big enough. We need $\ell^e = \Theta(p^2)$ to ensure a statistical distance of $O(p^{-1/2})$ to the uniform distribution, hence a security level of λ (since $p = \Theta(2^{2\lambda})$).

Theorem 42. *Let $\varepsilon \in]0,2]$. Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve and π be the probability distribution of codomains E' (up to $\overline{\mathbb{F}_p}$-isomorphism) of isogenies $\sigma : E \longrightarrow E'$ chosen uniformly at random among isogenies of degree $\deg(\sigma) \leq p^{1+\varepsilon}$. Let U be the uniform distribution of supersingular elliptic curves over $\mathbb{F}_{p^2}$ (up to $\overline{\mathbb{F}_p}$-isomorphism). Then, the statistical distance d_{TV} between the two distributions satisfies $d_{TV}(U, \pi) = O(p^{-\varepsilon/2})$.*

Theorem 43. *The* RigorousSQIsignHD *protocol is statistically honest-verifier zero knowledge in the RADIO model (provided $\ell^e = \Theta(p^2)$). In other words, there exists a random polynomial time simulator $\mathcal{S}$ with black-box access to a RADIO that simulates transcripts (E_1, φ, R) with a statistically indistinguishable distribution from the transcripts of the* RigorousSQIsignHD *identification protocol.*

Proof. First, we explain how to construct the simulator $\mathcal{S}$. As for Fast-SQIsignHD, the simulator starts by generating a challenge $\varphi' : E_A \longrightarrow E_2'$ and applies the RADIO on entry E_2' to get an efficient representation of a dual response isogeny $\widehat{\sigma'} : E_2' \longrightarrow E_1'$. As explained in the proof of Theorem 21, the simulator can use the efficient representation of $\widehat{\sigma'}$ to compute $R' := (q', \sigma'(P_1), \sigma'(P_2))$ in polynomial time, where (P_1, P_2) is a canonical basis of $E_1'[\ell^f]$. $\mathcal{S}$ finally outputs (E_1', φ', R').

We now prove that the transcripts (E_1', φ', R') of $\mathcal{S}$ are statistically indistinguishable from the transcripts (E_1, φ, R) of the RigorousSQIsignHD identification protocol. By construction, φ and φ' have the same distribution. Given E_2', by the definition of the RADIO, $\widehat{\sigma'}$ is uniform among isogenies of degree $q' \leq \ell^e$ with domain E_2', so E_1' is statistically close to uniformly random in the supersingular isogeny graph by Theorem 42. Besides, E_1 is statistically close to uniformly random as well, by Proposition 29.

Finally, conditionally to E_1' and E_2', $\widehat{\sigma'}$ (represented by R') is uniformly random among the isogenies $E_2' \longrightarrow E_1'$ of degree $q' < \ell^e$ by the definition of the RADIO. The dual map $\phi \longmapsto \widehat{\phi}$ being a bijection preserving the degree, conditionally to E_1' and E_2', σ' is also uniformly random among the isogenies $E_2' \longrightarrow E_1'$ of degree $q' < \ell^e$. By construction (see Section 4.2), conditionally to E_1 and E_2, σ has the same distribution. This completes the proof. ☐

E Omitted proofs

E.1 Kani's lemma (Lemma 4)

Lemma 4 (Kani)**.** *Consider the following (a, b)-isogeny diamond*

$$\begin{tikzcd}
A' \arrow[r, "\varphi'"] & B' \\
A \arrow[u, "\psi"] \arrow[r, "\varphi"] & B \arrow[u, "\psi'"']
\end{tikzcd}$$

with $d := a + b$ prime to the characteristic of the base field of abelian varieties. Then, the isogeny $F : A \times B' \longrightarrow B \times A'$ given in matrix notation by

$$F := \begin{pmatrix} \varphi & \widetilde{\psi}' \\ -\psi & \widetilde{\varphi}' \end{pmatrix}$$

is a d-isogeny with $d = a + b$.

If a and b are coprime, the kernel of F is

$$\ker(F) = \{(\widetilde{\varphi}(x), \psi'(x)) \mid x \in B[d]\}.$$

Proof. Here we use, without proof, the main properties of the dual abelian variety with respect to polarisations (see for instance [Kan14, § 11]). It is a classical result that the dual of a matrix for the product polarisations is the transpose of the matrix obtained after dualizing the coefficients and that dualization is an involutive operation, so that

$$\tilde{F} = \begin{pmatrix} \tilde{\varphi} & -\widetilde{\psi} \\ \psi' & \varphi' \end{pmatrix}.$$

Hence

$$\tilde{F}F = \begin{pmatrix} \widetilde{\varphi}\varphi + \widetilde{\psi}\psi & \widetilde{\varphi}\widetilde{\psi'} - \widetilde{\psi}\widetilde{\varphi'} \\ \psi'\varphi - \varphi'\psi & \psi'\widetilde{\psi'} + \varphi'\widetilde{\varphi'} \end{pmatrix}$$

with $\widetilde{\varphi}\varphi + \widetilde{\psi}\psi = [a]_A + [b]_A = [d]_A$ since φ is an a-isogeny and ψ is a b-isogeny. Similarly, we get $\widetilde{\psi'}\psi' + \widetilde{\varphi'}\varphi' = [d]_{A'}$, so that $\psi'\widetilde{\psi'} + \varphi'\widetilde{\varphi'} = [d]_{B'}$ after dualizing (the dual being anti-commutative and the dual of an integer being an integer). Clearly, $\psi'\varphi - \varphi'\psi = 0$ since we have an isogeny diamond and we obtain $\widetilde{\varphi}\widetilde{\psi'} - \widetilde{\psi}\widetilde{\varphi'} = 0$ by dualizing the preceding equality. Hence, $\tilde{F}F = [d]_{A\times B'}$ so F is a d-isogeny.

If $x \in B[d]$, we have

$$\begin{aligned} F(\widetilde{\varphi}(x), \psi'(x)) &= (\varphi \circ \widetilde{\varphi}(x) + \widetilde{\psi'} \circ \psi'(x), -\psi \circ \widetilde{\varphi}(x) + \widetilde{\varphi'} \circ \psi'(x)) \\ &= ([a]x + [b]x, 0) = ([d]x, 0) = (0, 0) \end{aligned}$$

where we used the fact that $\psi \circ \widetilde{\varphi} = \widetilde{\varphi'} \circ \psi'$. Indeed, $\psi'\varphi = \varphi'\psi$, which implies that $[a]\psi \circ \widetilde{\varphi} = [a]\widetilde{\varphi'} \circ \psi'$ after multiplying on the right by $\widetilde{\varphi}$ and on the left by $\widetilde{\varphi'}$, so that $\psi \circ \widetilde{\varphi} = \widetilde{\varphi'} \circ \psi'$ since $[a]_{A'}$ is an isogeny (it has finite kernel).

It follows that $\ker(F)$ contains the subgroub:

$$S := \{(\widetilde{\varphi}(x), \psi'(x)) \mid x \in B[d]\}.$$

Since $\widetilde{\varphi}$ and ψ' are a and b-isogenies respectively, we have $\ker(\widetilde{\varphi}) \subseteq B[a]$ and $\ker(\psi') \subseteq B[b]$. It follows that $\ker(\widetilde{\varphi}) \cap \ker(\psi') = \{0\}$, when a and b are coprime, so that $x \in B \longmapsto (\widetilde{\varphi}(x), \psi'(x))$ is injective and $\#S = \#B[d]$. Since d is prime to the characteristic of the field, we have $\#B[d] = d^{2g}$ with $g := \dim(B)$. A being isogenous to B and B', we also have $g = \dim(A) = \dim(B')$ and $\deg(F) = d^{\dim(A\times B')} = d^{2g}$ since F is a d-isogeny, so F is separable and $\#\ker(F) = \deg(F) = d^{2g}$. Hence, $\ker(F) = S$ when a and b are coprime and the proof is complete. □

Actually, we can also prove a weak converse of Kani's lemma.

Lemma 44. *Let $F : A \times B' \longrightarrow B \times A'$ be a d-isogeny (for the product principal polarizations), where d is prime to the characteristic of the base field. Write F as a matrix:*

$$F := \begin{pmatrix} \varphi & \widetilde{\psi'} \\ -\psi & \widetilde{\varphi'} \end{pmatrix}$$

and suppose φ is an a-isogeny. Then $\varphi, \varphi', \psi, \psi'$ form the following (a, b)-isogeny diamond with $a + b = d$:

$$\begin{tikzcd}
A' \arrow[r, "\varphi'"] & B' \\
A \arrow[u, "\psi"] \arrow[r, "\varphi"] & B \arrow[u, "\psi'"']
\end{tikzcd}$$

Proof. Since F is a d-isogeny, we have $\widetilde{F} \circ F = [d]$, so we get that $\psi' \circ \varphi = \varphi' \circ \psi$, $\widetilde{\varphi} \circ \varphi + \widetilde{\psi} \circ \psi = [d]$ and $\psi' \circ \widetilde{\psi'} + \varphi' \circ \widetilde{\varphi'} = [d]$. Since, φ is an a-isogeny, we have $\widetilde{\varphi} \circ \varphi = [a]$, so $\widetilde{\psi} \circ \psi = [b]$ and ψ is a b-isogeny.

We also have $F \circ \widetilde{F} = [d]$, so that $\varphi \circ \widetilde{\varphi} + \psi' \circ \widetilde{\psi'} = [d]$ and we get that ψ' is a b-isogeny. Then, the equality $\psi' \circ \widetilde{\psi'} + \varphi' \circ \widetilde{\varphi'} = [d]$ ensures that φ' is an a-isogeny. This completes the proof. ☐

Remark 45. We obtain the same result if we suppose that any of the isogenies φ', ψ and ψ' is an a or b-isogeny.

E.2 Finding a uniformly random tight response ideal (Lemmas 11 and 12)

Lemma 11. *Let f be a primitive positive definite integral quadratic form in k variables and let $\rho > 0$. Then there exists an algorithm that samples uniformly random elements from the set*

$$\{x \in \mathbb{Z}^k \mid f(x) \leq \rho\}$$

in polynomial time in $\log(\rho)$ and the length of f (namely, the maximal number of bits of the coefficients of f). This algorithm runs in exponential time in k.

Proof. By Cholesky decomposition theorem, there exists a matrix $B \in GL_k(\mathbb{R})$ such that $f(x) = \|Bx\|^2$ for all $x \in \mathbb{R}^k$, $\|.\|$ being the Euclidean norm. Let $\Lambda := \Lambda(B)$ be the lattice generated by the columns of B. We want to sample in $B(0, \sqrt{\rho}) \cap \Lambda$ with uniform distribution. Let $(b_1, \cdots, b_k)$ be an LLL-reduced basis of Λ. Let $\nu := \sqrt{k}\|b_k\|/2$ and consider the following sampling algorithm:

1. Sample $v \in B(0, \sqrt{\rho} + \nu)$ uniformly at random.
2. Find a solution $\lambda(v) \in \Lambda$ to the closest vector problem for v.
3. If $\lambda(v) \in B(0, \sqrt{\rho})$, return $\lambda(v)$; else restart.

We prove that the output $\lambda(v)$ has uniform distribution in $B(0, \sqrt{\rho}) \cap \Lambda$. Let $\mathcal{V} := \{v \in \mathbb{R}^k \mid \|v\| = \min_{\lambda \in \Lambda} \|v + \lambda\|\}$ be the Voronoi cell at the origin. Then, the closest vector $\lambda(v)$ satisfies $v \in \mathcal{V} + \lambda(v)$ and $\lambda(v)$ is unique when v is not at the border of a Voronoi cell, so it is unique with probability 1. Hence, for all $u \in B(0, \sqrt{\rho}) \cap \Lambda$,

$$\mathbb{P}(\lambda(v) = u) = \frac{\mathrm{Vol}((\mathcal{V} + u) \cap B(0, \sqrt{\rho} + \nu))}{\mathrm{Vol}(B(0, \sqrt{\rho} + \nu))}.$$

Let $\mu := \inf\{r > 0 \mid \forall v \in \mathbb{R}^k,\ \exists \lambda \in \Lambda, \|x - \lambda\| \leq r\}$ be the covering radius of Λ. Then $\mathcal{V} \subseteq B(0, \mu)$ and $\mu \leq \sqrt{k}\lambda_k/2$ where λ_k is the last minimum of Λ (this is a classical result, see for instance [Mic16, Exercise 11]), so that $\mu \leq \nu$. It follows that $\mathcal{V} + u \subseteq B(0, \sqrt{\rho} + \nu)$ for all $u \in B(0, \sqrt{\rho}) \cap \Lambda$. Hence

$$\mathbb{P}(\lambda(v) = u) = \frac{\mathrm{Vol}(\mathcal{V} + u)}{\mathrm{Vol}(B(0, \sqrt{\rho} + \nu))} = \frac{\mathrm{Vol}(\mathcal{V})}{\mathrm{Vol}(B(0, \sqrt{\rho} + \nu))}.$$

This quantity does not depend on u so $\lambda(v)$ has uniform distribution.

We finally check that the algorithm terminates after an expected polynomial number of steps in $\log(\rho)$ and the length of f. Indeed, when v is uniform in $B(0, \sqrt{\rho} + \nu)$, we have

$$\mathbb{P}(\|\lambda(v)\| \leq \sqrt{\rho}) \geq \mathbb{P}(\lambda(v) = 0) \geq \frac{\mathrm{Vol}(B(0, \lambda_1/2))}{\mathrm{Vol}(B(0, \sqrt{\rho} + \nu))} = \left(\frac{\lambda_1}{2\sqrt{\rho} + \sqrt{k}\|b_k\|}\right)^k,$$

where λ_1 is the first minimum of Λ. Since f is integral, we have $\|b_k\| \geq \cdots \geq \|b_1\| \geq \lambda_1 \geq 1$ and since $(b_1, \cdots, b_k)$ is LLL-reduced, we have:

$$\|b_1\| \cdots \|b_k\| \leq 2^{k(k-1)/4}\, \mathrm{Covol}(\Lambda) = 2^{k(k-1)/4}\, \mathrm{disc}(f),$$

so $\|b_k\| \leq 2^{k(k-1)/4}\, \mathrm{disc}(f)$. Hence, the algorithm terminates after an expected number of steps $O(1/\log \mathbb{P}(\|\lambda(v)\| \leq \sqrt{\rho}))$, a quantity that is polynomial in $\log(\rho)$, $\log(\mathrm{disc}(f))$ (itself polynomial in the length of f) and k. We conclude that the algorithm has the desired complexity, since the LLL algorithm has polynomial time complexity and the closest vector problem (used in step 2 of the algorithm) can be solved in the desired time complexity. □

Lemma 12. *Let $\mathcal{O}$ be a maximal order and J be a left $\mathcal{O}$-ideal. Then there exists $\alpha \in J$ such that $q_J(\alpha) \leq 2\sqrt{2p}/\pi$.*

Proof. Consider the canonical embedding $\iota : \mathcal{B}_{p,\infty} \hookrightarrow \mathbb{R}^4$:

$$1 \longmapsto (1,0,0,0), i \longmapsto (0, \sqrt{q_0}, 0, 0), j \longmapsto (0, 0, \sqrt{p}, 0), k \longmapsto (0, 0, 0, \sqrt{q_0 p}),$$

where $q_0 := \mathrm{nrd}(i)$. ι is an isometry in the following sense $\|\iota(\alpha)\|^2 = \mathrm{nrd}(\alpha)$ for all $\alpha \in \mathcal{B}_{p,\infty}$, where $\|.\|$ is the Euclidean norm of $\mathbb{R}^4$. By Minkowski's second theorem, the successive minima of the lattice $\iota(J)$ satisfy

$$\lambda_1 \cdots \lambda_4 \leq 2^4 \frac{\mathrm{Covol}(\iota(J))}{\mathrm{Vol}(B(0,1))} = \frac{32}{\pi^2} \mathrm{Covol}(\iota(J)). \tag{3}$$

But we have $\mathrm{Covol}(\iota(\mathcal{O})) = \mathrm{discrd}(\mathcal{O})/4$. Indeed, if $(\alpha_1, \cdots, \alpha_4)$ is a basis of $\mathcal{O}$, we get that

$$\begin{aligned}\mathrm{Covol}(\iota(\mathcal{O})) &= \sqrt{|\det(\langle \iota(\alpha_i), \iota(\alpha_j)\rangle)_{1\leq i,k\leq 4}|}\\ &= \sqrt{\left|\det\left(\frac{1}{2}(\mathrm{nrd}(\alpha_i+\alpha_j)-\mathrm{nrd}(\alpha_i)-\mathrm{nrd}(\alpha_j))\right)_{1\leq i,k\leq 4}\right|}\\ &= \sqrt{\left|\det\left(\frac{1}{2}\mathrm{Tr}(\alpha_i\overline{\alpha_j})\right)_{1\leq i,k\leq 4}\right|} = \frac{1}{4}\mathrm{discrd}(\mathcal{O})\end{aligned}$$

Besides, by [Voi20, Theorem 15.5.5], $\mathrm{discrd}(\mathcal{O}) = \mathrm{disc}(\mathcal{B}_{p,\infty}) = p$ since $\mathcal{O}$ is a maximal order. We then have

$$\mathrm{Covol}(\iota(J)) = [\mathcal{O}:J]\,\mathrm{Covol}(\iota(\mathcal{O})) = [\mathcal{O}:J]\,\mathrm{discrd}(\mathcal{O})/4 = \mathrm{nrd}(J)^2 p/4$$

It follows that the minimal value of q_J is

$$q_J(\alpha) = \lambda_1^2/\,\mathrm{nrd}(J) \leq \frac{2\sqrt{2p}}{\pi}.$$

□

E.3 Dividing the higher dimensional isogeny in two (Proposition 13)

Proposition 13. *Let $d := d_1d_2$ prime to p and $F : \mathcal{A} \longrightarrow \mathcal{B}$ be a d-isogeny between abelian varieties defined over $\overline{\mathbb{F}_p}$. Then:*

- **(i)** *We can always decompose $F = F_2 \circ F_1$, where F_1 is a d_1-isogeny and F_2 is a d_2-isogeny.*
- **(ii)** $\ker(F_1) \subseteq \ker(F) \cap \mathcal{A}[d_1]$.
- **(iii)** $\ker(\widetilde{F_2}) \subseteq \ker(\widetilde{F}) \cap \mathcal{B}[d_2] = F(\mathcal{A}[d]) \cap \mathcal{B}[d_2]$.
- **(iv)** *When $\ker(F)$ has rank $g := \dim(\mathcal{A})$, those inclusions are equalities.*

In order to prove the proposition, we need two intermediary results.

Lemma 46. *If $F : (\mathcal{A}, \lambda_{\mathcal{A}}) \longrightarrow (\mathcal{B}, \lambda_{\mathcal{B}})$ is a d-isogeny between principally polarized abelian varieties, then $\ker(F)$ is a maximal isotropic subgroup of $\mathcal{A}[d]$ (for the d-th Weil pairing).*

Proof. The inclusion $\ker(F) \subseteq \mathcal{A}[d]$ immediately follows from $\widetilde{F} \circ F = [d]$. Now we prove that $\ker(F)$ is isotropic. Let $x, y \in \ker(F)$. Since $\widetilde{F}$ is surjective, there exists $y' \in \mathcal{B}$ such that $y = \widetilde{F}(y')$. Let $\lambda_{\mathcal{A}}$ and $\lambda_{\mathcal{B}}$ be the principal polarisations on $\mathcal{A}$ and $\mathcal{B}$ respectively, $e_d^{\lambda_{\mathcal{A}}}$ and $e_d^{\lambda_{\mathcal{B}}}$ the associated Weil pairings. Then

$$e_d^{\lambda_{\mathcal{A}}}(x,y) = e_d(x, \lambda_{\mathcal{A}} \circ \widetilde{F}(y')) = e_d(x, \widehat{F} \circ \lambda_{\mathcal{B}}(y')) = e_d(F(x), \lambda_{\mathcal{B}}(y')) = 1.$$

Then $\ker(F)$ is isotropic. Since F is a d-isogeny, it has degree d^g with $g := \dim(\mathcal{A})$, and $\#\ker(F) = d^g$ since F is separable. So $\ker(F)$ is maximal isotropic.

□

Lemma 47. *let* $(\mathcal{A}, \lambda)$ *be a principally polarized abelian variety. If* $K \subset \mathcal{A}[d]$ *is isotropic, then the polarization* $[d]\lambda_{\mathcal{L}}$ *on* $\mathcal{A}$ *descends to a principal polarization on* $\mathcal{B} := \mathcal{A}/K$*. More precisely, there exists a principal polarization* λ' *on* $\mathcal{B}$ *such that* $[d]\lambda = \widehat{\pi} \circ \lambda' \circ \pi$*, where* $\pi : \mathcal{A} \longrightarrow \mathcal{B} = \mathcal{A}/K$ *is the canonical projection.*

Proof. We have $\ker([d]\lambda) = [d]^{-1}(\ker(\lambda)) = \mathcal{A}[d]$ since $\deg(\lambda) = 1$. Since $K \subset \mathcal{A}[d]$ is isotropic, the result follows from [Mil86, Proposition 6.8]. ☐

Proof. (of Proposition 13) **(i)** We prove that we have a decomposition of $F : \mathcal{A} \longrightarrow \mathcal{B}$ of the form

$$(\mathcal{A}, \lambda_{\mathcal{A}}) \xrightarrow{F_1} (\mathcal{C}, \lambda_{\mathcal{C}}) \xrightarrow{F_2} (\mathcal{B}, \lambda_{\mathcal{B}}),$$

where the intermediary abelian variety $\mathcal{C}$ is principally polarized and F_1 and F_2 are respectively d_1 and d_2-isogenies, with $d = d_1 d_2$. By induction, it suffices to prove this result when $d_1 = \ell$ is a prime.

Since $\#\ker(F) = d^g$, the ℓ-Sylow subgroup of $\ker(F)$ has cardinality $\ell^{g v_\ell}$, where v_ℓ is the ℓ-adic valuation of d. We may also write the ℓ-Sylow subgroup as follows:

$$G_\ell \cong \prod_{i=1}^{r} (\mathbb{Z}/\ell^{\alpha_i}\mathbb{Z}),$$

where the α_i are positive integers. Since $G_\ell \subseteq \ker(F) \subseteq \mathcal{A}[d]$, the α_i must all be $\leq v_\ell$. It follows that

$$g v_\ell = \log_\ell \#G_\ell = \sum_{i=1}^{r} \alpha_i \leq r v_\ell,$$

so that $r \geq g$ and $\ker(F)[\ell] = G_\ell[\ell] \cong (\mathbb{Z}/\ell\mathbb{Z})^r$, which implies that $\ker(F)[\ell]$ has rank $\geq g$. Hence, it contains a subgroup $K \subset \mathcal{A}[\ell]$ of rank g, which is isotropic in $\mathcal{A}[\ell]$ since $\ker(F) \supset K$ is isotropic in $\mathcal{A}[d]$ by Lemma 46.

By [Mum74, Theorem 4, p. 73], F factors through an isogeny F_1 of kernel K so we can indeed write $F = F_2 \circ F_1$. Since K is isotropic, by Lemma 47, the codomain $\mathcal{C}$ of F_1 admits a principal polarization $\lambda_{\mathcal{C}}$ such that $\widehat{F_1} \circ \lambda_{\mathcal{C}} \circ F_1 = [\ell]\lambda_{\mathcal{A}}$ *i.e.* $\widetilde{F_1} \circ F_1 = [\ell]$. So F_1 is an ℓ-isogeny.

We also have

$$[d]_{\mathcal{B}} = F \circ \widetilde{F} = F_2 \circ F_1 \circ \widetilde{F_1} \circ \widetilde{F_2} = F_2 \circ [\ell]_{\mathcal{C}} \circ \widetilde{F_2} = F_2 \circ \widetilde{F_2} \circ [\ell]_{\mathcal{B}},$$

so $F_2 \circ \widetilde{F_2} = [d/\ell]_{\mathcal{B}}$ since $[\ell]_{\mathcal{B}}$ is surjective and F_2 is a $d_2 = d/\ell$-isogeny. To prove the result in the general case, we can proceed by induction on the degree d (and factor F_2).

(ii) We always have $\ker(F_1) \subseteq \ker(F) \cap \mathcal{A}[d_1]$ since $F = F_2 \circ F_1$ and F_1 is a d_1-isogeny.

(iii) Similarly, we get that $\ker(\widetilde{F_2}) \subseteq \ker(\widetilde{F}) \cap \mathcal{B}[d_2]$. Besides, $\widetilde{F} \circ F = [d]_{\mathcal{A}}$ so $F(\mathcal{A}[d]) \subseteq \ker(\widetilde{F})$. Since $\ker(F) \subseteq \mathcal{A}[d]$, we have an isomorphism $F(\mathcal{A}[d]) \cong \mathcal{A}[d]/\ker(F)$. Furthermore, d being prime to p, $[d]$, F and $\widetilde{F}$ are separable, so that

$\#\mathcal{A}[d] = d^{2g}$ and $\#\ker(\widetilde{F}) = \#\ker(F) = d^g$. Hence, $\#F(\mathcal{A}[d]) = d^g = \#\ker(\widetilde{F})$ and $\ker(\widetilde{F}) = F(\mathcal{A}[d])$. Point (iii) follows.

(iv) Suppose $\ker(F)$ has rank g. Let $(x_1, \cdots, x_g)$ be a basis of generators of $\ker(F)$. Then, the x_i all have order d and are linearly independent over $\mathbb{Z}/d\mathbb{Z}$. Then, $\ker(F) \cap \mathcal{A}[d_1] = ([d_2]x_1, \cdots, [d_2]x_g)$ and the $[d_2]x_i$ all have order d_1 and are linearly independent over $\mathbb{Z}/d_1\mathbb{Z}$, so that $\#\ker(F) \cap \mathcal{A}[d_1] = d_1^g = \#\ker(F_1)$. The equality $\ker(F_1) = \ker(F) \cap \mathcal{A}[d_1]$ follows by (ii).

To prove that $\ker(\widetilde{F_2}) = F(\mathcal{A}[d]) \cap \mathcal{B}[d_2]$, it suffices to prove that $F(\mathcal{A}[d]) = \ker(\widetilde{F})$ has rank g and the preceding reasoning will apply. We have $F(\mathcal{A}[d]) \cong \mathcal{A}[d]/\ker(F)$, so it suffices to prove that there is a subgroup $G \subset \mathcal{A}[d]$ of rank g such that $\ker(F) \oplus G = \mathcal{A}[d]$, because, we will have $F(\mathcal{A}[d]) \cong G$. We consider the d-th Weil pairing $e_d^{\lambda_\mathcal{A}} : \mathcal{A}[d]^2 \longrightarrow \mu_d(\overline{\mathbb{F}_p})$, where $\mu_d(\overline{\mathbb{F}_p})$ is the group of d-th roots of unity in $\overline{\mathbb{F}_p}$, and the group homomorphism

$$\begin{aligned} \Phi : \mathcal{A}[d] &\longrightarrow \mu_d(\overline{\mathbb{F}_p})^g \\ y &\longmapsto (e_d^{\lambda_\mathcal{A}}(x_i, y))_{1 \leq i \leq g}. \end{aligned}$$

Since $\ker(F)$ is maximal isotropic, we have $\ker(\Phi) = \ker(F)$. It follows that $\#\operatorname{im}(\Phi) = \#\mathcal{A}[d]/\#\ker(\Phi) = d^{2g}/d^g = d^g$, so that $\operatorname{im}(\Phi) = \mu_d(\overline{\mathbb{F}_p})^g$ *i.e.* Φ is surjective. Let $\zeta \in \mu_d(\overline{\mathbb{F}_p})$ be a primitive d-th root of unity. Then, for all $j \in [\![1 \; ; \; g]\!]$ there exists $y_j \in \mathcal{A}[d]$ such that $e_d^{\lambda_\mathcal{A}}(x_i, y_j) = \zeta^{\delta_{i,j}}$ for all $i \in [\![1 \; ; \; g]\!]$, where $\delta_{i,j}$ is the Kronecker delta. It follows that the y_i all have order d (since $e_d^{\lambda_\mathcal{A}}(x_i, y_i) = \zeta$ has order d), are linearly independent over $\mathbb{Z}/d\mathbb{Z}$ and linearly independent of the x_i. We can then take $G := \langle y_1, \cdots, y_g \rangle$. This completes the proof.

□

E.4 Correctness of $\mathsf{RepresentIsogeny}_{g,\ell^e,\ell^f}$ (Proposition 14)

Proposition 14. *Algorithms 4 and 15 are correct. Namely, Algorithm 4 returns* $F_1, \tilde{F}_2$ *such that* $F_2 \circ F_1 = F(\sigma, a_1, a_2)$ *on entry* $a_1, a_2, P_1, P_2, \sigma(P_2), \sigma(P_2)$*, where* $\sigma : E_1 \longrightarrow E_2$ *is a* q*-isogeny with* $a_1^2 + a_2^2 + q = \ell^e$ *(and similarly for Algorithm 15).*

Proof. We assume that Algorithm 4 received $a_1, a_2, P_1, P_2, \sigma(P_2), \sigma(P_2)$ on entry. Let us write $F^* := F(\sigma, a_1, a_2)$ and decompose $F^* := F_2^* \circ F_1^*$, where F_1^* is an ℓ^{e_1}-isogeny of kernel $\ker(F^*)[\ell^{e_1}]$ and $\widetilde{F_2^*}$ is an ℓ^{e_2}-isogeny of kernel $F^*(E_1^2 \times E_2^2[\ell^{e_2}])$. Then, by construction the outputs F_1 and $\widetilde{F_2}$ of Algorithm 4 satisfy $\ker(F_1) = \ker(F_1^*)$, $\ker(\widetilde{F_2}) = \ker(\widetilde{F_2^*})$, $\deg(F_1) = \deg(F_1^*)$ and $\deg(\widetilde{F_2}) = \deg(\widetilde{F_2^*})$. Hence, by [Mum74, Theorem 4, p.73] if we denote $(\mathcal{C}, \lambda_\mathcal{C})$ the common codomain of F_1 and $\widetilde{F_2}$ and $(\mathcal{C}^*, \lambda_{\mathcal{C}^*})$ the common codomain of F_1^* and $\widetilde{F_2^*}$, then we get that $F_1 = \lambda \circ F_1^*$ and $\widetilde{F_2} = \mu \circ \widetilde{F_2^*}$, where λ and μ are isomorphisms $(\mathcal{C}^*, \lambda_{\mathcal{C}^*}) \xrightarrow{\sim} (\mathcal{C}, \lambda_\mathcal{C})$.

To conclude, we need to be more specific about the way F_1 and $\widetilde{F_2}$ are computed explicitly with the theta-model of level 2 (see Appendix F.3). To ensure F_1

and $\widetilde{F_2}$ define the same level-structure on $\mathcal{C}[4]$, we use Algorithm 20 outputting two symplectic basis of $E_1^2 \times E_2^2[4\ell^{\max(e_1,e_2)}]$ satisfying the compatibility equation of Corollary 58. The output of this algorithm only depends on the input $a_1, a_2, P_1, P_2, \sigma(P_2), \sigma(P_2)$, so it would be the same if we applied it for F_1^* and $\widetilde{F_2^*}$. It follows that $(\mathcal{C}, \lambda_{\mathcal{C}}) = (\mathcal{C}^*, \lambda_{\mathcal{C}^*})$ and that F_1 and F_1^* coincide on $F_1^{-1}(\mathcal{C}[4])$, so that λ and $\mathrm{id}_{\mathcal{C}}$ coincide on $\mathcal{C}[4]$, so that $\lambda = \mathrm{id}_{\mathcal{C}}$ by [Mil86, Proposition 17.5.(b)]. Similarly, $\widetilde{F_2}$ and $\widetilde{F_2^*}$ coincide on $\widetilde{F_2}^{-1}(\mathcal{C}[4])$, so that $\mu = \mathrm{id}_{\mathcal{C}}$. It follows that $F_2 \circ F_1 = F^* = F(\sigma, a_1, a_2)$, so Algorithm 4 is correct. The same proof applies to Algorithm 15. □

E.5 Verification (Proposition 15 and Corollary 16)

Proposition 15. *Algorithms 5 and 16 are correct. Namely, when given* $E_1, E_2, a_1, a_2, F_1, \tilde{F}_2$, *if Algorithm 5 returns* True, *then* $F_2 \circ F_1$ *is an efficient representation of an isogeny* $\sigma : E_1 \longrightarrow E_2$ *of degree* $q = \ell^e - a_1^2 - a_2^2$ *(and similarly for Algorithm 16).*

Proof. Assume that Algorithm 5 returns True on input $E_1, E_2, a_1, a_2, F_1, \widetilde{F_2}$. Then, $F := F_2 \circ F_1$ is well defined and is an ℓ^e-isogeny. We may write $F := (f_{i,j})_{1 \leq i,j \leq 4} \in \mathrm{End}(E_1^2 \times E_2^2)$. Then, since $\widetilde{F} \circ F = [\ell^e]$, we get that

$$\forall 1 \leq j \leq 4, \quad \sum_{i=1}^{4} \deg(f_{i,j}) = \ell^e, \tag{4}$$

so the $f_{i,j}$ have degree $\leq \ell^e$. By assumption, there exists $Q \in E_1$ of order $\ell^f \ell'^{f'}$ such that $f_{1,1}(Q) = [a_1]Q$, $f_{2,1}(Q) = -[a_2]Q$ and $f_{4,1}(Q) = 0$. Besides, by Cauchy Schwarz inequality

$$\deg(f_{1,1} - [a_1]) \leq \left(\sqrt{\deg(f_{1,1})} + \sqrt{\deg([a_1])}\right)^2 \leq 4\ell^e < \ell^f \ell'^{f'}.$$

It follows that $f_{1,1} = [a_1]$. We similarly obtain that $f_{2,1} = -[a_2]$ and $f_{4,1} = 0$. Hence, by 4, we get $\deg(f_{3,1}) = \ell^e - a_1^2 - a_2^2 = q$. But $\sigma := f_{3,1}$ is an isogeny $E_1 \longrightarrow E_2$. This proves the correctness of Algorithm 5.

We use a similar argument to prove the correctness of Algorithm 16. We write $F := (f_{i,j})_{1 \leq i,j \leq 8}$. Then, the evaluation of the T-torsion points ensures that $f_{i,1} = [a_i]$ for all $i \in [\![1 \; ; \; 4]\!]$ and $f_{i,1}$ for all $i \in \{6, 7, 8\}$ since $T > 4\ell^e$ ($T \simeq p^3$ and $\ell^e \simeq p^2$). Using an analogue of 4, we conclude that $\sigma := f_{5,1} : E_1 \longrightarrow E_2$ has degree $q = \ell^e - a_1^2 - a_2^2 - a_3^2 - a_4^2$. This completes the proof. □

Corollary 16. *The verification procedures* FastVerify *and* RigorousVerify *(Algorithms 3 and 14) are correct. Namely, on input* (R_1, R_2, q), FastVerify *(respectively* RigorousVerify*) returns* True *if and only if* (R_1, R_2, q) *defines an efficient representation of an isogeny* $\sigma : E_1 \longrightarrow E_2$ *of degree* q, *where* q *is* ℓ^e*-good and prime to* ℓ' *(respectively* $q < \ell^e$*).*

Proof. If $\mathsf{FastVerify}(R_1, R_2, q)$ returns True, it means that q is ℓ^e-good and prime to ℓ, that $\mathsf{FastVerify}$ found $a_1, a_2 \in \mathbb{Z}$ such that $a_1^2 + a_2^2 + q = \ell^e$ and that $\mathsf{IsValid}_{4,\ell^e,\ell^f\ell'^{f'}}(E_1, E_2, a_1, a_2, *)$ returned True. In particular, by Proposition 15, it means that $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}(E_1, E_2, a_1, a_2, P_1, P_2, R_1, R_2)$ returned an efficient representation of a q-isogeny $\sigma : E_1 \longrightarrow E_2$. Hence, the knowledge of (R_1, R_2, q), Cornacchia's algorithm to find a_1, a_2 and $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}$ (running in polynomial time in $\log(p)$) define an efficient representation of σ.

Conversely, if (R_1, R_2, q) defines an efficient representation of a q-isogeny $\sigma : E_1 \longrightarrow E_2$, where q is ℓ^e-good and prime to ℓ', then we can evaluate σ on the canonical basis (P_1, P_2) of $E_1[\ell^f]$ in polynomial time in $\log(p)$ and obtain $(R_1, R_2) := (\sigma(P_1), \sigma(P_2))$. On input (R_1, R_2, q), $\mathsf{FastVerify}$ will find $a_1, a_2 \in \mathbb{Z}$ such that $a_1^2+a_2^2+q = \ell^e$ and apply $\mathsf{RepresentIsogeny}_{4,\ell^e,\ell^f}(E_1, E_2, a_1, a_2, P_1, P_2, R_1, R_2)$, which will return $F_1, \widetilde{F_2}$ such that $F_2 \circ F_1 = F(\sigma, a_1, a_2)$ by Proposition 14. Hence, $\mathsf{IsValid}_{4,\ell^e,\ell^f\ell'^{f'}}$ will return True by construction, and so will $\mathsf{FastVerify}$.

The same proof applies to $\mathsf{RigorousVerify}$. □

E.6 On the codomain distribution of random isogenies with bounded degree (Theorem 42)

The goal of this section is to prove a bound on the statistical distance between codomains of random isogenies with bounded degrees and the uniform distribution on the supersingular isogeny graph. Similar results have been proved on fixed degree smooth isogeny walks [GPS20, Theorem 1] and non-bactracking ℓ-isogeny walks [BCC+23, Theorem 11]. We generalize these results to the case of non-fixed degree. Heuristically, we should expect to be as close to the uniform distribution lower degree bounds than in the fixed degree case, but this is not the case. In particular, as for non-bactracking ℓ-isogeny walks, we need to allow isogenies of degree $p^{1+\varepsilon}$ to reach a statistical distance of $O(p^{-\varepsilon/2})$ to the uniform distribution. However, we provide an elementary proof that does not require to study adjacency matrices of the supersingular isogeny graph and modular forms (unlike [BCC+23]). The main ingredients are the Deuring correspondence and a count of small quaternion ideal vectors (Corollary 50). We start by proving a classical bound on the last minimum of quaternion ideals claimed in [KLPT14, § 3.1] but never proved so far.

Lemma 48. *Let $\mathcal{O} \subset \mathcal{B}_{p,\infty}$ be a quaternion order and I be a left ideal of $\mathcal{O}$. Let $(\alpha_1, \cdots, \alpha_4)$ be a Minkowski reduced basis of I for the quadratic form $q_I : \alpha \in I \longmapsto \mathrm{nrd}(\alpha)/\mathrm{nrd}(I)$, so that $q_I(\alpha_i) \leq q_I(\alpha_{i+1})$ for all $i \in \{1, 2, 3\}$. Then*

$$q_I(\alpha_4) \leq \frac{8p}{\pi^2}.$$

Proof. As we saw in the proof of Lemma 12, we have by Minkowski's second theorem (Equation 3):

$$\prod_{i=1}^{4} q_I(\alpha_i) \leq \frac{64p^2}{\pi^4}.$$

This inequality is not sufficient to conclude (we only get $q_I(\alpha_4) = O(p^2)$ instead of $O(p)$). To complete the proof, we follow [BST+17, Theorem 3.1].

Let $(\beta_1, \cdots, \beta_4)$ be a Minkowski reduced basis of $\mathcal{O}$. As the α_i, the β_i satisfy

$$\prod_{i=1}^{4} \mathrm{nrd}(\beta_i) \leq \frac{64p^2}{\pi^4}.$$

Let $A := (a_{i,j})_{1 \leq i,j \leq 4} \in M_4(\mathbb{Z})$, where for all $1 \leq i, j \leq 4$, $a_{i,j}$ is the coefficient of α_4 in the decomposition of $\beta_i \alpha_j$ in the basis $(\alpha_1, \cdots, \alpha_4)$ (this is an integer since $\mathcal{O}I \subseteq I$). Then A is invertible. Indeed, if $x \in \mathbb{Z}^4$ satisfies $Ax = 0$ *i.e.* $\sum_{j=1}^{4} a_{i,j} x_j = 0$ for all $i \in [\![1 \; ; \; 4]\!]$, then $\alpha := \sum_{j=1}^{4} x_j \alpha_j$ satisfies $\mathcal{O}\alpha \subseteq \langle \alpha_1, \alpha_2, \alpha_3 \rangle$. But $\mathcal{O}\alpha$ has rank 4 whenever $\alpha \neq 0$, so $\alpha = 0$ and $x = 0$.

A being invertible, there exists a permutation $\sigma \in \mathfrak{S}_4$ such that $a_{i,\sigma(i)} \neq 0$ for all $i \in [\![1 \; ; \; 4]\!]$. It follows that for all $i \in [\![1 \; ; \; 4]\!]$, $\beta_i \alpha_{\sigma(i)}$ completes $\langle \alpha_1, \alpha_2, \alpha_3 \rangle$ as a full-rank sublattice of I, so that $\mathrm{nrd}(\alpha_4) \leq \mathrm{nrd}(\beta_i \alpha_{\sigma(i)})$ *i.e.* $q_I(\alpha_4) \leq \mathrm{nrd}(\beta_i) q_I(\alpha_{\sigma(i)})$, since $(\alpha_1, \cdots, \alpha_4)$ is Minkowski reduced. It follows that

$$q_I(\alpha_4)^4 \leq \prod_{i=1}^{4} (\mathrm{nrd}(\beta_i) q_I(\alpha_{\sigma(i)})) = \prod_{i=1}^{4} \mathrm{nrd}(\beta_i) \prod_{i=1}^{4} q_I(\alpha_i) \leq \left(\frac{64p^2}{\pi^4} \right)^2.$$

The result follows. □

Now, we introduce a generalization of [Wes22, Lemma 3.2] in every dimension, counting the elements of bounded norm in a lattice.

Lemma 49. *Let $\Lambda \subseteq \mathbb{Z}^k$ be a full-rank lattice of last minimum λ_k and $\rho > \sqrt{k}/2\lambda_k$. Then*

$$\frac{\pi^{k/2} \left(\rho - \frac{\sqrt{k}\lambda_k}{2} \right)^k}{\Gamma\left(\frac{k}{2} + 1\right) \mathrm{Covol}(\Lambda)} \leq \# \Lambda \cap B(0, \rho) \leq \frac{\pi^{k/2} \left(\rho + \frac{\sqrt{k}\lambda_k}{2} \right)^k}{\Gamma\left(\frac{k}{2} + 1\right) \mathrm{Covol}(\Lambda)},$$

where $B(0, \rho)$ is the ball of center 0 and radius ρ for the Euclidean norm and Γ is Euler's gamma function.

Proof. Let $\mathcal{V} := \{v \in \mathbb{R}^k \mid \|v\| = \min_{\lambda \in \Lambda} \|v + \lambda\|\}$ be the Voronoi cell at the origin of Λ and $\mu := \sup_{v \in \mathcal{V}} \|v\|$ be the covering radius of Λ. Then, we have

$$B(0, \rho - \mu) \subseteq \bigsqcup_{\lambda \in \Lambda \cap B(0, \rho)} (\lambda + \mathcal{V}) \subseteq B(0, \rho + \mu),$$

so that

$$\mathrm{Vol}(B(0, \rho - \mu)) \leq (\# \Lambda \cap B(0, \rho)) \cdot \mathrm{Vol}(\mathcal{V}) \leq \mathrm{Vol}(B(0, \rho + \mu)).$$

Since $\mathrm{Vol}(\mathcal{V}) = \mathrm{Covol}(\Lambda)$, $\mathrm{Vol}(B(0, \rho \pm \mu)) = \pi^{k/2} (\rho \pm \mu)^k / \Gamma(k/2 + 1)$ and $\mu \leq \sqrt{k}\lambda_k/2$, the result follows. □

Corollary 50. *Let $\mathcal{O} \subset \mathcal{B}_{p,\infty}$ be a maximal order and I be an integral left $\mathcal{O}$-ideal. Then, for all $\varepsilon > 0$ the number of ideals of norm $\leq p^{1+\varepsilon}$ that are right-equivalent to I is*

$$N_{p^{1+\varepsilon}}([I]) := \#\{J \sim I \mid \operatorname{nrd}(J) \leq p^{1+\varepsilon}\} = \frac{2\pi^2}{\#\mathcal{O}_R(I)^\times} p^{1+2\varepsilon}(1 + O(p^{-\varepsilon/2})).$$

Proof. By [DKLPW20, Lemma 1], an ideal J is right-equivalent to I if and only if it is of the form $J := I\overline{\alpha}/\operatorname{nrd}(I)$ for some $\alpha \in I$. Besides, α is uniquely determined by J up to multiplication on the right by an element of $\mathcal{O}_R(I)^\times$ and we have $\operatorname{nrd}(J) = \operatorname{nrd}(\alpha)/\operatorname{nrd}(I) = q_I(\alpha)$. It follows that

$$N_{p^{1+\varepsilon}}([I]) := \#\{J \sim I \mid \operatorname{nrd}(J) \leq p^{1+\varepsilon}\} = \frac{1}{\#\mathcal{O}_R(I)^\times} \#\{\alpha \in I \mid q_I(\alpha) \leq p^{1+\varepsilon}\}.$$

As in the proof of Lemma 12, let $\iota : \mathcal{B}_{p,\infty} \hookrightarrow \mathbb{R}^4$ be the canonical embedding, such that $\|\iota(\alpha)\|^2 = \operatorname{nrd}(\alpha)$ for all $\alpha \in \mathcal{B}_{p,\infty}$, where $\|.\|$ is the Euclidean norm on $\mathbb{R}^4$. Consider the lattice $\Lambda := \iota(I)$. We then have

$$N_{p^{1+\varepsilon}}([I]) = \frac{1}{\#\mathcal{O}_R(I)^\times} \#\Lambda \cap B\left(0, p^{(1+\varepsilon)/2}\sqrt{\operatorname{nrd}(I)}\right)$$

By Lemmas 48 and 49, we get

$$\#\Lambda \cap B\left(0, p^{(1+\varepsilon)/2}\sqrt{\operatorname{nrd}(I)}\right) \leq \frac{\pi^2\left(p^{(1+\varepsilon)/2}\sqrt{\operatorname{nrd}(I)} + \frac{2\sqrt{2p\operatorname{nrd}(I)}}{\pi}\right)^4}{2\operatorname{Covol}(\Lambda)},$$

with $\operatorname{Covol}(\Lambda) = p/4\operatorname{nrd}(I)^2$, as we saw in the proof of Lemma 12. It follows, that the right term of the inequality is $2\pi^2 p^{1+2\varepsilon}(1 + O(p^{-\varepsilon/2}))$. Applying the lower bound of Lemma 49, we also get that

$$\#\Lambda \cap B\left(0, p^{(1+\varepsilon)/2}\sqrt{\operatorname{nrd}(I)}\right) \geq 2\pi^2 p^{1+2\varepsilon}(1 + O(p^{-\varepsilon/2})).$$

The result follows. □

We denote by $\operatorname{SS}(p)$ the set of supersingular elliptic curves over $\mathbb{F}_{p^2}$ (up to $\overline{\mathbb{F}_p}$-isomorphism) and S the probability distribution on $\operatorname{SS}(p)$ given by $S(E) := 1/(K\#\operatorname{Aut}(E))$ for all $E \in \operatorname{SS}(p)$, with $K := \sum_{E\in\operatorname{SS}(p)} 1/\#\operatorname{Aut}(E) = (p-1)/24$ by Eichler mass formula [Voi20, Theorem 25.1.1]. Let U be the uniform distribution on $\operatorname{SS}(p)$. Recall that by Lemma 30, the statistical distance between S and U is $d_{TV}(S, U) = O(p^{-1})$. We can now finally prove our main result.

Theorem 42. *Let $\varepsilon \in]0, 2]$. Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve and π be the probability distribution of codomains E' (up to $\overline{\mathbb{F}_p}$-isomorphism) of isogenies $\sigma : E \longrightarrow E'$ chosen uniformly at random among isogenies of degree $\deg(\sigma) \leq p^{1+\varepsilon}$. Then $d_{TV}(U, \pi) = O(p^{-\varepsilon/2})$.*

Proof. By the Deuring correspondence, it suffices to prove that given a maximal order $\mathcal{O} \subset \mathcal{B}_{p,\infty}$ and $\mathrm{Cl}(\mathcal{O})$ the set of right-equivalence classes of left-ideals of $\mathcal{O}$, the distribution π' of the ideal classes $[I] \in \mathrm{Cl}(\mathcal{O})$ when I is sampled uniformly at random among ideals of norm $\leq p^{1+\varepsilon}$ (which is the quaternion analogue of π) is at statistical distance $O(p^{-\varepsilon/2})$ from the uniform distribution U' on $\mathrm{Cl}(\mathcal{O})$. We also denote by S' the quaternion analogue of S, namely the distribution on $\mathrm{Cl}(\mathcal{O})$ given by $S([I]) := 1/(K\#O_R(I)^\times)$ for all $[I] \in \mathrm{Cl}(\mathcal{O})$, where $K := \sum_{[I]\in\mathrm{Cl}(\mathcal{O})} 1/\#O_R(I)^\times = (p-1)/24$ is the Eichler mass. By Lemma 30, $d_{TV}(U', S') = O(p^{-1})$, so it suffices to prove that $d_{TV}(S', \pi') = O(p^{-\varepsilon/2})$.

By Corollary 50, the number of left $\mathcal{O}$-ideals of norm $\leq p^{1+\varepsilon}$ is

$$N_{p^{1+\varepsilon}} = \sum_{[I]\in\mathrm{Cl}(\mathcal{O})} N_{p^{1+\varepsilon}}([I]) = 2\pi^2 K p^{1+2\varepsilon}(1 + O(p^{-\varepsilon/2})),$$

so the distribution π' is given by

$$\forall [I] \in \mathrm{Cl}(\mathcal{O}), \quad \pi'([I]) = \frac{N_{p^{1+\varepsilon}}([I])}{N_{p^{1+\varepsilon}}} = \frac{1}{K}(1 + O(p^{-\varepsilon/2})).$$

It follows immediately that

$$d_{TV}(S', \pi') = \frac{1}{2} \sum_{[I]\in\mathrm{Cl}(\mathcal{O})} |S'([I]) - \pi'([I])| = \frac{\#Cl(\mathcal{O})}{K} O(p^{-\varepsilon/2}) = O(p^{-\varepsilon/2}).$$

The result follows. □

F Higher dimensional isogenies and the theta model

In this section, we explain how to compute an ℓ^e-isogeny between abelian varieties in plain generality and then apply it to our specific problem. In subsection F.1, we give a high level description of an ℓ^e-isogeny computation as a chain of ℓ-isogenies. In the remaining subsections, we introduce the Θ-model to compute ℓ-isogenies when $\ell = 2$.

Indeed, unlike in dimension 2 or 3, we cannot use Jacobians to compute isogenies in dimension 4 or 8. However, there already exist algorithms to compute ℓ-isogenies in any dimension g with the Θ-model [LR12; LR15; LR23] in time $O(\ell^g)$. To minimize the complexity, the best strategy would be to take $\ell = 2$ and to use Θ-coordinates of level $n = 2$. However, existing algorithms only work when ℓ and n are coprime. We propose an algorithm to compute 2-isogenies in level $n = 2$ in Appendix F.4 and how we can use it in our specific problem (computing two 2-isogeny chains F_1 and $\widetilde{F_2}$ with the same codomain).

F.1 Computing an ℓ-isogeny chain.

Let $F : (\mathcal{A}, \lambda_\mathcal{A}) \longrightarrow (\mathcal{B}, \lambda_\mathcal{B})$ be an ℓ^e-isogeny between principally polarized abelian varieties and let K be its kernel. Assume that K has rank g. Then, we can decompose F as an ℓ-isogeny chain as in dimension 1.

Lemma 51. *We can write F as a product of ℓ-isogenies $F = F_e \circ \cdots \circ F_1$ between principally polarized abelian varieties $F_i : \mathcal{A}_{i-1} \longrightarrow \mathcal{A}_i$ (for $i \in [\![1 \ ; \ e]\!]$, with $\mathcal{A}_0 := \mathcal{A}$ and $\mathcal{A}_e := \mathcal{B}$).*

Let $K_0 := K = \ker(F)$ and $K_i := F_i(K_{i-1})$ for all $i \in [\![1 \ ; \ e]\!]$. Then, we have $\ker(F_i) = [\ell^{e-i}]K_{i-1}$ for all $i \in [\![1 \ ; \ e]\!]$.

Proof. We prove by induction on $i \in [\![0 \ ; \ e]\!]$ that we can write $F = G_i \circ F_i \circ \cdots \circ F_1$ where the $F_j : \mathcal{A}_{j-1} \longrightarrow \mathcal{A}_j$ for $j \in [\![1 \ ; \ i]\!]$ are ℓ-isogenies between principally polarized abelian varieties (PPAV) of kernel $\ker(F_j) = [\ell^{e-j}]K_{j-1}$ and G_i is an ℓ^{e-i}-isogeny between PPAV of kernel $\ker(G_i) = K_i$. For $i = 0$, the result is trivial. Let us assume the result at rank $i \in [\![0 \ ; \ e-1]\!]$. Then we simply apply point (i) of Proposition 13 to G_i to write $G_i := G_{i+1} \circ F_{i+1}$, where F_{i+1} and G_{i+1} are respectively ℓ and $\ell^{e-(i+1)}$-isogenies between PPAV. By point (iv) of Proposition 13, we also have $\ker(F_{i+1}) = K_i \cap \mathcal{A}_i[\ell]$ since K_i has rank g (K also having rank g). Since $K_i \subset \mathcal{A}_i[\ell^{e-i}]$ is maximal isotropic of rank g, we may write $K_i = \langle x_1, \cdots, x_g \rangle$ where all the $x_i \in \mathcal{A}_i[\ell^{e-i}]$ have order ℓ^{e-i}, so that

$$\ker(F_{i+1}) = K_i \cap \mathcal{A}_i[\ell] = \langle [\ell^{e-(i+1)}]x_1, \cdots, [\ell^{e-(i+1)}]x_g \rangle = [\ell^{e-(i+1)}]K_i.$$

We also have $\ker(G_{i+1}) = F_{i+1}(\ker(G_i)) = K_{i+1}$. This completes the proof. □

The above lemma leads to similar isogeny computation algorithms to the dimension 1 case. Assume that we know a basis $\mathscr{B}_0$ of $K = \ker(F)$ and let $\mathscr{B}_i := F_i(\mathscr{B}_{i-1})$, which is a basis of K_i for all $i \in [\![1 \ ; \ e]\!]$. Consider the graph whose vertices are the $[\ell^j]\mathscr{B}_i$ for $0 \leq i \leq e-1$, $0 \leq j \leq e-1-i$ and edges are of two kind:

- multiplication by ℓ $[\ell^j]\mathscr{B}_i \longrightarrow [\ell^{j+1}]\mathscr{B}_i$ (left edges);
- ℓ-isogeny computation $[\ell^j]\mathscr{B}_i \longrightarrow [\ell^j]\mathscr{B}_{i+1}$ (right edges).

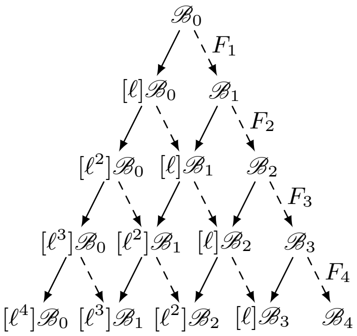

Fig. 2. Computational structure of the ℓ^e isogeny F with $e = 5$.

This graph represents the computational structure of F. To compute F, we need to compute the kernel basis $[\ell^{e-1}]\mathscr{B}_0, [\ell^{e-2}]\mathscr{B}_1 \cdots, \mathscr{B}_{e-1}$ representing the ℓ-isogenies in the chain, *i.e.* the bottom line in Figure 2. This graph is computed as follows: to go down right $[\ell^j]\mathscr{B}_i \longrightarrow [\ell^j]\mathscr{B}_{i+1}$, we need to have reached the bottom vertex $[\ell^{e-1-i}]\mathscr{B}_i$ beforehand. Of course there are naive algorithms where we compute every point of the graph but they are quadratic in e and far from optimal. There also exist divide and conquer strategies that require only $O(e\log(e))$ multiplications by ℓ and ℓ-isogeny evaluations (see [JD11, § 4.2.2] for details). We can even optimize such a strategy to minimize the global cost depending on the relative cost of scalar multiplications by ℓ and ℓ-isogeny evaluations.

F.2 Theta coordinates

For simplicity, even through we work over a finite field, we will describe our algorithm using analytic theta functions. The algebraic theory of Mumford [Mum66b] can be used to show that our algorithms are still valid over an arbitrary field of odd characteristic.

Let $A = \mathbb{C}^g/(\mathbb{Z}^g + \varOmega_A\mathbb{Z}^g)$ be an abelian variety with $\varOmega = \varOmega_A$ in the Siegel space corresponding to a principal polarisation $\mathcal{L} = \mathcal{L}_A$ on A. Let $\pi_A : \mathbb{C}^g \to A$ be the projection.

Recall that the analytic theta functions with characteristic $a, b \in \mathbb{Q}^g$ are given by

$$\theta\left[\begin{smallmatrix} a\\ b\end{smallmatrix}\right](z,\varOmega) = \sum_{n\in\mathbb{Z}^g} e^{\pi i\,{}^t(n+a)\varOmega(n+a)+2\pi i\,{}^t(n+a)(z+b)}.$$

A basis of level 2 theta functions is given by $\theta_i^A(P) = \theta\left[\begin{smallmatrix} 0\\ i/2\end{smallmatrix}\right](z_P, \varOmega/2), i \in (\mathbb{Z}/2\mathbb{Z})^g$ where $z_P \in \mathbb{C}^g$ represents $P \in A$: $P = \pi_A(z_P)$. Here we use the following abuse of notations: if $i \in (\mathbb{Z}/2\mathbb{Z})^g$, we denote by i any lift to $\mathbb{Z}^g$. Reciprocally if $i \in \mathbb{Z}^g$, we also denote by i its reduction to $(\mathbb{Z}/2\mathbb{Z})^g$.

The analytic theta functions depend on the period matrix $\varOmega_A$. Algebraically they are defined by a symmetric theta structure $\varTheta_A$ of level 2. We will denote our theta functions by $\theta_i^{\varTheta_A}$ when we want to make this dependence explicit.

We will also make use of the "dual" basis $\theta'^A_\chi(P) = \theta\left[\begin{smallmatrix} \chi/2\\ 0\end{smallmatrix}\right](2z_P, 2\varOmega), \chi \in (\hat{\mathbb{Z}}/2\hat{\mathbb{Z}})^g$, where we identify $(\hat{\mathbb{Z}}/2\hat{\mathbb{Z}})^g$ with $(\mathbb{Z}/2\mathbb{Z})^g$ via the inner product. Going to the dual level 2 coordinates corresponds analytically to the action of the symplectic matrix $S = \left(\begin{smallmatrix} 0 & 1\\ -1 & 0\end{smallmatrix}\right)$ on the period matrix $\varOmega_A$. Explicitly on theta coordinates the modular transform is given via the Hadamard transformation $\theta'_\chi = \sum_i \chi(i)\theta_i$, and reciprocally $2^g\theta_i = \sum_\chi \chi(i)\theta'_\chi$. We let H be the Hadamard matrix in dimension 2^g, by the formula above it allows to pass back and forth between the theta coordinates of level 2 and their duals.

F.3 Matching theta structures

Let us recapitulate the isogeny we need to compute in the verification step of SQIsignHD: we have a d-isogeny $F : A \to B$ where $(A, \mathcal{L})$ and $(B, \mathcal{M})$ are given by products of elliptic curves with their product polarisations. We split the isogeny in two as in Section 4.3: $F = F_2 \circ F_1$ with $F_1 : A \to C$ a d_1-isogeny and $F_2 : C \to B$ an d_2-isogeny, and we assume that we are given the kernel of F_1 in $A[d_1]$ and $\tilde{F}_2$ in $B[d_2]$.

The theta isogeny algorithm to compute F_1 requires (if d_1 is odd):

- A symmetric level 2 theta structure Θ_A on $(A, \mathcal{L})$. This level structure determines a symplectic basis of $A[2]$ and is in turn determined by a symplectic basis of $A[4]$. This symmetric level structure will be represented (up to twists) by the theta constant $(\theta_i^{\Theta_A}(0))_{i \in (\mathbb{Z}/2\mathbb{Z})^g}$ and gives a basis of sections $(\theta_i^{\Theta_A})_{i \in (\mathbb{Z}/2\mathbb{Z})^g}$ of $\mathcal{L}^2$. In particular if the 4-torsion is rational, the level 2 theta model will be rational (this is a sufficient but not necessary condition).
- Generators $P_1, \dots, P_g$ of the kernel $K = \operatorname{Ker} F_1$ in theta coordinates $\theta_i^{\Theta_A}(P_j)$, with the basis of theta coordinates induced by the symmetric theta structure fixed above.

In SQIsignHD, A will be equal to a product of g elliptic curves $A = E_1 \times \cdots \times E_g$ and the points of the kernel K is described in terms of tuples of Weierstrass coordinates. We first need to explain how to convert these points to theta coordinates. We fix a symplectic basis (e_i, f_i) on each $E_i[4]$, this induces a product symplectic basis on $A[4]$, hence a product theta structure. There are well known formula to convert from Weierstrass coordinates on E_i to theta coordinates [Mum84]. We can then compute the theta coordinates on A as follows:

Lemma 52. *Let $\mathcal{L} = \mathcal{L}_1 \star \mathcal{L}_2 \cdots \star \mathcal{L}_g$ be a product polarisation on $A = E_1 \times \cdot \times E_g$. Endow $(A, \mathcal{L})$ with a product theta structure Θ_A of each theta structure Θ_{E_i} on E_i. If $P = (P_1, \dots, P_g) \in A$, then for $i = (i_1, \dots, i_g) \in (\mathbb{Z}/2\mathbb{Z})^g$, $\theta_i^{\Theta_A}(P) = \prod_{j=1}^g \theta_{i_j}^{\Theta_{E_j}}(P_j)$.*

Proof. This follows from

$$\theta \left[\begin{smallmatrix} a_1, a_2 \\ b_1, b_2 \end{smallmatrix}\right] ((z_1, z_2), \begin{pmatrix} \Omega_1 & 0 \\ 0 & \Omega_2 \end{pmatrix}) = \theta \left[\begin{smallmatrix} a_1 \\ b_1 \end{smallmatrix}\right] (z_1, \Omega_1) \theta \left[\begin{smallmatrix} a_2 \\ b_2 \end{smallmatrix}\right] (z_2, \Omega_2).$$

□

Hence if we have a point $R \in A$ given in product Weierstrass coordinates $R = (R_j) = ((x_j, y_j))$, we can convert each R_i from Weierstrass coordinates to level 2 theta coordinates $\theta_i^{\Theta_{E_j}}(R_j)$ then apply Lemma 52 to get the theta coordinates of R with respect to the product theta structure.

We can now apply the isogeny theorem from [CR15; LR23] with $N = d_1, d_2$:

Theorem 53. *Let $(A, \mathcal{L})$ be a principally polarised abelian variety with a symmetric theta structure Θ_A of level 2, induced by a symplectic basis $\mathcal{B}_A = (x_1, \dots, x_g, y_1, \dots, y_g)$ of $A[4]$ with respect to ζ, a primitive fourth root of unity.*

Let $(P_1, \dots, P_g)$ be a basis of the kernel of a maximal isotropic subgroup of $A[N]$ of rank g, given in theta coordinates, where N is an odd integer.

Let P be a point of A given in theta coordinates. Let $F : A \to B = A/K$ the induced isogeny. Then there is a unique descent of $\mathcal{L}^N$ to a polarisation $\mathcal{M}$ on B, and a symmetric theta structure Θ_B on $\mathcal{M}$ induced by the symplectic basis $F(\mathcal{B}_A) = (\frac{1}{N}F(x_1), \dots, \frac{1}{N}F(x_g), F(y_1), \dots, F(y_g))$ with respect to ζ of $B[4]$.

Furthermore, the theta null point of B and the theta coordinates of $F(P)$ can be computed in $O(N^g)$ arithmetic operations over the base field.

Proof. This is a special case of Theorem 56 proved below. ☐

We can use Theorem 53 to compute an ℓ^e-isogeny by splitting it into a product of ℓ-isogenies. In SQIsignHD we specifically want to handle the case $\ell = 2$. We will give an algorithm in Appendix F.4.

We have another difficulty to solve first. In SQIsignHD we glue two isogenies together $F_1 : A \to C$ and $\tilde{F}_2 : B \to C$. These isogenies are compatible with the product polarisation on A and B, so the codomain C is endowed with the same polarisation in both cases. However, when using Theorem 53 to compute C and its polarisation, it needs not be endowed with the same level 2 symmetric theta structure Θ_C for the two isogenies.

Let $\mathcal{B}_1$ be a symplectic basis of $C[4]$ giving the symmetric theta structure on C induced by F_1, and $\mathcal{B}_2$ be the one induced by $\tilde{F}_2$. Then there is a symplectic matrix $M \in \mathrm{Sp}_{2g}(\mathbb{Z}/4\mathbb{Z})$ such that $M\mathcal{B}_1 = \mathcal{B}_2$. We can use the theta transformation formula [BL04, §8.6] for M to convert the theta null point expressed in terms of $\mathcal{B}_1$ to the one expressed in terms of $\mathcal{B}_2$: $\theta_i^{\mathcal{B}_2}(0) = \theta_i^{M\mathcal{B}_1}(0)$. Once we have endowed C with the same theta structures, checking that they are indeed the same simply amounts to testing for equality of the theta null points seen as projective coordinates.

So one way to test that F_1 and $\tilde{F}_2$ indeed have the same polarised codomain C is to apply Theorem 53 twice and then to act by all matrices in $M \in \mathrm{Sp}_{2g}(\mathbb{Z}/4\mathbb{Z})$ on the theta null point induced by F_1 until we find an equality of projective theta null points with the theta null point induced by $\tilde{F}_2$. This costs $O(1)$ but in practice is too expensive. We will instead explain how to compute the correct correcting matrix M directly.

Remark 54. Many symplectic basis $\mathcal{B}_C$ of $C[4]$ will give the same symmetric theta structure Θ_C of level 2 on C (hence the same theta null point), indeed the theta null point only determines a symplectic basis of $C[2]$. Rather than working with the 4-torsion we could work only with the 2-torsion and take $M \in \mathrm{Sp}_{2g}(\mathbb{Z}/2\mathbb{Z})$; this does not completely determines all the symmetric theta structures of level 2 but it is easy to test all 2^{2g} possibilities.

Proposition 55. *Let $F_1 : A \to C$ be a d_1-isogeny, $\tilde{F}_2 : B \to C$ be a d_2-isogeny, $F = F_2 \circ F_1 : A \to B$, with d_1 and d_2 prime to 2. Let $\mathcal{B}_A$ a symplectic basis of $A[4]$, $\mathcal{B}_B$ a symplectic basis of $B[4]$. Let $\mathcal{B}'_B$ be the symplectic basis on B induced by F. Let $M' = \begin{pmatrix} \alpha' & \beta' \\ \gamma' & \delta' \end{pmatrix}$ be the symplectic matrix such that $M'\mathcal{B}'_B = \mathcal{B}_B$. Let $\mathcal{B}_1$ be the symplectic basis of $C[4]$ induced by F_1 and $\mathcal{B}_2$ be the symplectic basis of $C[4]$ induced by $\tilde{F}_2$. Then $\mathcal{B}_2 = M\mathcal{B}_1$, with $M = \begin{pmatrix} \alpha'/d_2 & \beta' \\ \gamma' & d_2\gamma' \end{pmatrix}$.*

Proof. Define $\gamma_x = \begin{pmatrix} x & 0 \\ 0 & 1 \end{pmatrix}$. By Theorem 53, we have $\mathcal{B}_1 = \gamma_{1/d_1} \cdot F_1 \mathcal{B}_A$, $\mathcal{B}_2 = \gamma_{1/d_2} \cdot \tilde{F}_2 \mathcal{B}_B$, $\mathcal{B}'_B = \gamma_{1/d} \cdot F \mathcal{B}_A$. If $M\mathcal{B}_1 = \mathcal{B}_2$, we get that $M = d_2 \gamma_{1/d_2} M' \gamma_{1/d} \gamma_{1/d_1}^{-1}$. □

So if we compute the image of F on $A[4]$, we can recover the correct matrix M. In SQIsignHD, F is built from $\sigma : E_1 \longrightarrow E_2$ and scalars, so for the verification it suffices to give the action of σ on the 4-torsion. If it is not provided, we can just guess it and try the corresponding symplectic matrix, this greatly reduces the number of symplectic matrices to try to only a few choices.

We now explain how to handle the case of a d-isogeny F where d is not prime to 2. A difficulty is that if we only start with a symplectic basis $\mathcal{B}_A$ of $A[4]$, then since the kernel of F_1 may contain points of 4-torsion, F_1 does not induces a canonical symplectic basis of $C[4]$ anymore. So the algorithm needs to start with more data.

Theorem 56. *Let $(A, \mathcal{L})$ be a principally polarised abelian variety, with $\mathcal{L}$ a symmetric ample line bundle. Let N be an integer, $\mathcal{B}'_A = (x'_1, \ldots, x'_g, y'_1, \ldots, y'_g)$ be a symplectic basis of $A[4N]$ with respect to a primitive $4N$-root of unity ζ, and $\mathcal{B}_A = N\mathcal{B}'_A$ the induced symplectic basis of $A[4]$. It induces a symmetric level 2 theta structure Θ_A on A.*

Let K be the maximal isotropic kernel generated by the points $P_j = 4x'_j$. Assume that we are given the theta coordinates of level 2 of the x'_j, and the theta coordinates of a point P in A.

Let $F : A \to B = A/K$ the induced isogeny. Then there is a unique descent of $\mathcal{L}^N$ to a symmetric line bundle $\mathcal{M}$ on B, and a symmetric theta structure on $\mathcal{M}$ induced by the symplectic basis

$$(F(x'_1), \ldots, F(x'_g), NF(y'_1), \ldots, NF(y'_g))$$

with respect to ζ^N of $B[4]$.

Furthermore, the theta null points of B and the theta coordinates of $F(P)$ can be computed in $O(N^g)$ arithmetic operations over the base field.

Proof. Since we are given the points (x'_i, y'_i) in level 2 theta coordinates, we can use the algorithms of [LR12; CR15; LR23] to construct a symmetric theta structure of level $2N$ on the theta group $G(\mathcal{L}^{2N})$. However, the references above assume for simplicity that the degree of the isogenies is prime to the level m of the symmetric theta structure we start with. Here $m = 2$ and N is no longer assumed to be odd. So we need the general case, which is described in [Rob10, Chapters 6 and 7], [Rob21, § 2.10, Remarks 2.10.3 and 2.10.7]. Once we are in level $2N$, we can apply Mumford's isogeny theorem [Mum66a, Theorem 4 p.302–303] to obtain a symmetric theta structure Θ_B of level 2 on B and the equations of the isogeny. The algorithm takes time $O(N^g)$. It remains to show that the theta structure we obtain on B is the one induced by the points $(F(x'_1), \ldots, F(x'_g), NF(y'_1), \ldots, NF(y'_g))$.

Let $Z(K)$ be the centralizer of K in the theta group $G(\mathcal{L}^{2N})$. The theta structure Θ_B is induced by the canonical map $\alpha_f : Z(K)/\tilde{K} \to G(\mathcal{M}^2)$ from [Mum66a, Equation (2) p.302], where $\tilde{K}$ is the canonical lift of K into the theta group $G(\mathcal{L}^{2N})$ induced by our theta structure of level $2N$. The points

$$(2F(x_1'), \dots, 2F(x_g'), 2NF(y_1'), \dots, 2NF(y_g'))$$

form a symplectic basis of $B[2]$. The theta structure Θ_B on $G(\mathcal{M}^2)$ is determined by the symmetric lifts of this basis into $G(\mathcal{M}^2)$. By definition of the induced theta structure, if T is in this basis, the symmetric lift $g_T \in G(\mathcal{M}^2)$ above T induced by Θ_B is given by the image by α_f of the symmetric lift $g_{T'} \in G(\mathcal{L}^{2N})$ induced by the theta structure of level $2N$ on A for any point T' such that $F(T') = T$. For $T = 2F(x_j')$, we can take $T' = 2x_j'$. Since the theta structure on $G(\mathcal{L}^{2N})$ is determined by the basis (x_j', y_j'), the lift $g_{T'}$ is determined as follow: let $g_{x_j'} \in G(\mathcal{L}^{4N})$ be any of the two symmetric lift of x_j', and define $g_{T'} = \eta_2(g_{x_j'})$ where η_2 is defined in [Mum66a, § 2, p.310]; this does not depend on the choice of $g_{x_j'}$. Since α_f sends symmetric elements into symmetric elements and commutes with η_2, we get that $g_T = \eta_2(g_{F(x_j')})$ where $g_{F(x_j')}$ is any of the two symmetric element above $F(x_j')$ in $G(\mathcal{M}^4)$. Likewise, when $T = 2NF(y_j')$, we check that $g_T = \eta_2(g_{NF(y_j')})$, for $g_{NF(y_j')} \in G(\mathcal{M}^4)$ one of the two symmetric elements above $NF(y_j')$. Hence the descent of the symmetric theta structure of level $2N$ on $G(\mathcal{L}^{2N})$ induced by the basis $(x_1', \dots, x_g', y_1', \dots, y_g')$ to a symmetric theta structure of level 2 on $G(\mathcal{M}^2)$ is indeed the one induced by $(F(x_1'), \dots, F(x_g'), NF(y_1'), \dots, NF(y_g'))$. □

Notice that $NF(y_j') = F(y_j)$, so Theorem 56 only needs the points (x_j', y_j) as input. If N is odd, Theorem 53 is a special case of Theorem 56: by the CRT, from a symplectic basis (x_j, y_j) of $A[4]$ and a basis P_j of K, there is a unique $x_j' \in A[4N]$ which induces both x_j and P_j: $x_j = Nx_j'$, $P_j = 4x_j'$.

However when N is not prime to 2, we cannot start with any symplectic basis (x_j, y_j) of $A[4]$, it has to be compatible with our kernel K, in the sense that there should exists x_j' a basis of a maximal isotropic subgroup of $A[4N]$ which induces both x_j and P_j. In the situation of SQIsignHD, where we convert our points given by tuple of Weierstrass coordinates into theta coordinates given by a product theta structure, the resulting product symplectic basis of $A[4]$ will not be compatible with our kernels in general. So to get an input suitable for Theorem 56, we first start with the basis (P_j) of K (in tuple of Weierstrass coordinates), fix points x_j' above each P_j such that $P_j = 4x_j'$ and the x_j' generate an isotropic subgroup of $A[4N]$. Then we let $x_j = Nx_j'$, and fix a symplectic complement y_j of the x_j. We compute the symplectic matrix M that changes the product symplectic basis into the (x_j, y_j) and act by this matrix to get the theta coordinates in terms of our new basis (x_j, y_j).

We can adapt Proposition 55 to the general case:

Proposition 57. *Let $F_1 : A \to C$ be a d_1-isogeny, $\tilde{F}_2 : B \to C$ a d_2-isogeny and $F = F_2 \circ F_1$ an d-isogeny, where $d = d_1 d_2$. Let d' be a common multiple of d_1 and d_2, and write $d' = c_1 d_1 = c_2 d_2$.*

Let $(P_1, \dots, P_g)$ *be a basis of the kernel of* F_1*, and* $(x'_1, \dots, x'_g, y'_1, \dots, y'_g)$ *a symplectic basis of* $A[4d']$ *with respect to* ζ*, a primitive* $4d'$*-root of unity, and such that* $P_i = 4c_1 x'_i$*.*

Let $(Q_1, \dots, Q_g)$ *be a basis of the kernel of* $\tilde{F}_2$*, and* $(u'_1, \dots, u'_g, v'_1, \dots, v'_g)$ *a symplectic basis of* $B[4d']$ *with respect to* ζ*, such that* $Q_i = 4c_2 v'_i$*.*

Then the induced theta structure on C *induced by* F_1 *and* $\tilde{F}_2$ *via Theorem 56 is the same if*

$$\tilde{F}_2(c_2 v'_i) = F_1(d' y'_i) \text{ and } F_1(c_1 x'_i) = \tilde{F}_2(d' u'_i) \tag{5}$$

Proof. The symplectic theta structure on C induced by F_1 is given by $(F_1(c_1 x'_i), F_1(d' y'_i))$, and the one induced by $\tilde{F}_2$ is given by $(\tilde{F}_2(d' u'_i), \tilde{F}_2(c_2 v'_i))$. □

We note that in the isogeny algorithm for F_1 we only need the points $c_1 x'_i, d' y'_i$ and for $\tilde{F}_2$ we only need the points $d' u'_i, c_2 v'_i$.

Corollary 58. *Let* $F = F_2 \circ F_1$*. To get the same theta structure on* C*, it suffices to choose* x'_i, y'_i, u'_i, v'_i *such that* $F(c_2 y'_i) = c_2 v'_i$ *and* $\tilde{F}(c_1 u'_i) = c_1 x'_i$*.*

An algorithm to construct suitable x'_i, y'_i, u'_i, v'_i is as follows. Take y''_i a basis of a symplectic complement of $\operatorname{Ker} F_1$ in $A[d_1]$, $y'_i \in A[d']$ isotropic such that $4c_1 y'_i = y''_i$, and let $v'_i = F(y'_i)$. Then $Q_i = 4c_2 v'_i$ is a basis of $\operatorname{Ker} \widetilde{F}_2$. We let u'_i be a symplectic complement of v'_i in $B[d']$, and we let $x'_i = \tilde{F}(u'_i)$. Let $P_i = 4c_1 x'_i$, they form a basis of $\operatorname{Ker} F_1$.

Corollary 58 explain why we need $f \geq e_2 + 2$ in Remark 4.2 when $\ell = 2$ (an alternative if we are only given the action of σ on $E_1[\ell^{e_2}]$ would be to just guess it on $E_1[\ell^{e_2+2}]$).

Algorithm 20: Finding two compatible basis.

```
Data: ker($F_1$), ker($\widetilde{F_2}$), an algorithm to evaluate $F = F_1 \circ F_2$ on $A[d']$ and an
      algorithm to evaluate $\widetilde{F}$ on $B[d']$.
Result: Two symplectic basis $(x'_1, \cdots, x'_g, y'_1, \cdots, y'_g)$ of $A[4d']$ and
        $(u'_1, \cdots, u'_g, v'_1, \cdots, v'_g)$ of $B[4d']$ such that $\ker(F_1) = \langle 4c_1 x'_i \rangle_i$,
        $\ker(\widetilde{F_2}) = \langle 4c_2 v'_i \rangle_i$, $F(c_2 y'_i) = c_2 v'_i$ and $\widetilde{F}(c_1 u'_i) = c_1 x'_i$ for all
        $i \in [\![1 ; g]\!]$.
Find $(y''_1, \cdots, y''_g)$ a basis of a symplectic complement of $\mathrm{Ker}(F_1)$ in $A[d_1]$;
Find $(y'_1, \cdots, y'_g)$ forming an isotropic subgroup of $A[d']$ such that $4c_1 y'_i = y''_i$
  for all $i \in [\![1 ; g]\!]$;
$v'_i \longleftarrow F(y'_i)$ for all $i \in [\![1 ; g]\!]$;
Let $(u'_1, \cdots, u'_g)$ be a symplectic complement of the $v'_i$ in $B[d']$;
$x'_i \longleftarrow \widetilde{F}(u'_i)$ for all $i \in [\![1 ; g]\!]$;
Return $(x'_1, \cdots, x'_g, y'_1, \cdots, y'_g)$ and $(u'_1, \cdots, u'_g, v'_1, \cdots, v'_g)$;
```

F.4 Computing 2^e-isogenies

We need to compute an N-isogeny, with $N = d_1$ or $N = d_2$ with the notations from Appendix F.3. The isogeny algorithms described in [CR15; LR23] assume that the degree is prime to 2 for simplicity. For SQIsignHD, we want to take $\ell = 2$ (so that $N = 2^e$) for efficiency. The general case of N even is described in [Rob10; Rob21]. In this section we focus on the case $N = 2^e$ and detail how the general algorithm can be used to compute 2-isogenies. Handling 2-isogenies is actually easier because we can use the duplication formulae directly.

With the notations of Theorem 56, we assume that we are given the $4N$-torsion points $(x'_1, \dots, x'_g)$ given in theta coordinates by the symmetric theta structure of level 2 induced by a symplectic basis $(x_1, \dots, x_g, y_1 \dots, y_g)$ of $A[4]$ with $x_j = N x'_j$. Let $P_j = 4x'_j$, K the subgroup generated by the P_j, $F : A \to B = A/K$ the corresponding isogeny.

We then write $N = 2N'$, and let $T_j = N'P_j$, the kernel of a 2-isogeny f through which F factorizes. We remark also that $T''_j = N'x'_j$ is a point of 8-torsion such that $x_j = 2T''_j$. We will explain how to compute the isogenous theta null point of the codomain of f, and how to push points through f. The points $f(x'_j)$ will then be points of $4N'$ torsion, and we iterate.

So from now on we let f be a 2-isogeny $A \to B$, $K = \langle T_1, \dots, T_g \rangle$ be the kernel of f, $(T''_1, \dots, T''_g)$ be points in $A[8]$ such that $T_j = 4T''_j, j = 1, \dots, g$. We assume that the symmetric theta structure Θ_A on A is induced by a symplectic basis $(T'_1, \dots, T'_g, U'_1, \dots U'_g)$ where $T'_j = 2T''_j$ and that we are given the coordinates of the T''_j, $j \in \{1, \dots, g\}$. If $i \in (\mathbb{Z}/2\mathbb{Z})^g$, we let $T''_i = \sum_{j=1}^g i_j T''_j$. For simplicity, we will even assume that we are given the theta coordinates of all T''_i, $i \in (\mathbb{Z}/2\mathbb{Z})^g$, in particular the points $T_i, i \in (\mathbb{Z}/2\mathbb{Z})^g$ span the full kernel K.

The 2-isogeny formula will be derived from the duplication formula [Igu72, Theorem 2 p. 139, p. 141]:

$$\theta \left[\begin{smallmatrix} a_1 \\ b_1 \end{smallmatrix}\right] (z_1, \Omega) \theta \left[\begin{smallmatrix} a_2 \\ b_2 \end{smallmatrix}\right] (z_2, \Omega) = \sum_{t \in \frac{1}{2}\mathbb{Z}^g/\mathbb{Z}^g} \theta \left[\begin{smallmatrix} \frac{a_1+a_2}{2}+t \\ b_1+b_2 \end{smallmatrix}\right] (z_1 + z_2, 2\Omega) \cdot \theta \left[\begin{smallmatrix} \frac{a_1-a_2}{2}+t \\ b_1-b_2 \end{smallmatrix}\right] (z_1 - z_2, 2\Omega) \quad (6)$$

$$2^g \theta \left[\begin{smallmatrix} a_1 \\ b_1 \end{smallmatrix}\right] (z_1, \Omega) \theta \left[\begin{smallmatrix} a_2 \\ b_2 \end{smallmatrix}\right] (z_2, \Omega) = \sum_{t \in \frac{1}{2}\mathbb{Z}^g/\mathbb{Z}^g} e^{-2\pi i (a_1 | 2t)} \theta \left[\begin{smallmatrix} a_1+a_2 \\ \frac{b_1+b_2}{2}+t \end{smallmatrix}\right] (\frac{z_1 + z_2}{2}, \frac{\Omega}{2}) \cdot \theta \left[\begin{smallmatrix} a_1-a_2 \\ \frac{b_1-b_2}{2}+t \end{smallmatrix}\right] (\frac{z_1 - z_2}{2}, \frac{\Omega}{2}). \quad (7)$$

We will derive algebraic formula for the analytic isogeny $\phi : A = \mathbb{C}^g/(\Omega_A \mathbb{Z}^g + \mathbb{Z}^g) \to B = \mathbb{C}^g/(\Omega_B \mathbb{Z}^g + \mathbb{Z}^g), z \mapsto 2z$ where $\Omega_B = 2\Omega_A$. We will then explain how to use these formula to compute our algebraic isogeny f.

Recall from Appendix F.2 that the Hadamard matrix H allows to convert from the theta coordinates θ_i^A to the dual theta coordinates ${\theta'}_\chi^A$. Given two points P_1, P_2 given by theta coordinates $\theta_i(P_j)$, we also let $(\theta_i(P_1)) \star (\theta_i(P_2)) = (\theta_i(P_1) \cdot \theta_i(P_2))_{i \in (\mathbb{Z}/2\mathbb{Z})^g}$.

Proposition 59. *Let P be a point on A. Then the theta coordinates of the points $\phi(P) \in B$, where $\phi : A \to B$ is the isogeny defined above, are given by:*

$$(\theta_i^B(\phi(P))) \star (\theta_i^B(0))_{i \in (\mathbb{Z}/2\mathbb{Z})^g} = H \cdot \left(({\theta'}_\chi^A(P))_{\chi \in (\hat{\mathbb{Z}}/2\hat{\mathbb{Z}})^g} \star ({\theta'}_\chi^A(P))_{\chi \in (\hat{\mathbb{Z}}/2\hat{\mathbb{Z}})^g} \right).$$

Proof. Using the duplication formula, we obtain (with $\Omega = \Omega_A$):

$$\theta \left[\begin{smallmatrix} 0 \\ i \end{smallmatrix}\right] (2z, \Omega) \theta \left[\begin{smallmatrix} 0 \\ i \end{smallmatrix}\right] (0, \Omega) = \sum_{t \in \frac{1}{2}\mathbb{Z}^g/\mathbb{Z}^g} \theta \left[\begin{smallmatrix} t \\ 2i \end{smallmatrix}\right] (2z, 2\Omega) \theta \left[\begin{smallmatrix} t \\ 0 \end{smallmatrix}\right] (2z, 2\Omega)$$

This means that $(\theta_i^B(\phi(P))\theta_i^B(0))_{i \in (\mathbb{Z}/2\mathbb{Z})^g} = H \cdot \left({\theta'}_\chi^A(P)^2 \right)_{\chi \in (\hat{\mathbb{Z}}/2\hat{\mathbb{Z}})^g}$. □

So the image of a point P by a 2-isogeny ϕ is simple to compute, provided we know the theta null point $\theta_i^B(0) = \theta \left[\begin{smallmatrix} 0 \\ i/2 \end{smallmatrix}\right] (0, 2\Omega)$ of B: start with the theta coordinates $\theta_i^A(P)$ of P, apply the Hadamard transform to get the dual coordinates ${\theta'}_\chi^A(P)$, square these coordinates, and apply the Hadamard transform again to obtain $\theta_i^B(\phi(P))\theta_i^B(0)$. It now only remains to divide by the coordinates $\theta_i^B(0)$ given by the theta null point of B.

Corollary 60. *Assume that we are given the theta coordinates $\theta_i^A(P)$ of $P \in A$ and of the theta null point $\theta_i^B(0)$ of B. After a precomputation of 2^g inversions to invert the coordinates of the theta null point of B, the theta coordinates of $\phi(P)$ can be computed in 2 Hadamard transforms, 2^g squares and 2^g multiplications.*

Furthermore, given the theta null points of A and B one can check (up to signs) that A and B are indeed 2-isogenous (with compatible theta structure) using 2^{g+1} squares and 2 Hadamard transforms.

Proof. The first statement follows from the Proposition. For the second statement, if A and B are 2-isogenous, then $\theta_i^B(0)^2 = H \cdot {\theta'}_\chi^A(0)^2$, which determines the $\theta_i^B(0)$ up to a sign. □

In particular the proof of Corollary 60 shows that we can easily compute the square of the coordinates of the theta null point of B. In practice in our complexity estimates, we will often neglect the Hadamard transforms since they just amount to some additions and subtractions. It remains to compute the correct square roots.

Proposition 61. *Let $i \in \mathbb{Z}^g$ and $z_i' \in \mathbb{C}^g$ be the analytic theta point given by the affine coordinates $\theta_t^A(z_i') = \theta \left[\begin{smallmatrix} 0 \\ i/4+t/2 \end{smallmatrix}\right] (0, \Omega_A/2)$. Then $T_i' = \pi_A(z_i')$ is a point in $A[4]$. The points $T_i = 2T_i'$ generate the kernel corresponding to the kernel $\frac{1}{2}\mathbb{Z}^g/\mathbb{Z}^g$ of the isogeny ϕ.*

We have up to a constant (not depending on i):

$$\theta_i^B(0) = \sum_t \theta_t^A(z_i')^2. \tag{8}$$

Proof. Using the duplication formula again, we obtain for $i \in \mathbb{Z}^g$:

$$2^g \theta \left[\begin{smallmatrix} 0 \\ i/2 \end{smallmatrix}\right] (0, \Omega) \theta \left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right] (0, \Omega) = \sum_t \theta \left[\begin{smallmatrix} 0 \\ i/4+t/2 \end{smallmatrix}\right] (0, \Omega/2)^2.$$

Up to the projective factor $2^g \theta \left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right] (0, \Omega)$, we recover Eq. (8). We also check that although z'_i depends on the choice of $i \in \mathbb{Z}^g$, the term on the left only depends on the reduction of i in $(\mathbb{Z}/2\mathbb{Z})^g$. □

In the analytic setting, when representing the points z'_i and T'_i by theta coordinates, they actually are represented by the same coordinates, but z'_i is represented by affine coordinates while T'_i is represented by projective coordinates. So the projection π_A amount to sending the affine point $(\theta_i(z'_i)) \in \mathbb{A}^{2^g}$ to the projective point in $\mathbb{P}^{2^g-1}$.

We go back to the algebraic setting: we let $f : A \to B$ be a 2-isogeny with kernel generated by the points (T_j), $j = 1, \dots, g$ and we assume that we are given isotropic points T''_j such that $T_j = 4T''_j$, and the T''_j are expressed as theta coordinates with respect to the theta structure induced by a symplectic basis $(T'_1, \dots, T'_g, U'_1, \dots U'_g)$ where $T'_j = 2T''_j$. We need to compute the theta null point on B for our algebraic isogeny f like we did for our analytic isogeny ϕ above in Proposition 61. Then we can apply Proposition 59 to compute the image by f of any point P.

Notice that Eq. (8), because of the sum, only makes sense for points in affine coordinates. So we cannot apply it directly in an algebraic algorithm, because we only have the 4-torsion points $T'_i, i \in (\mathbb{Z}/2\mathbb{Z})^g$ in projective coordinates. We need to lift the projective points T'_i into an affine point $\tilde{T'_i}$ such that Eq. (8) make sense. We follow the terminology of [LR12, § 3] and speak about affine lifts. Riemann relations give a well defined doubling and differential addition law on affine lifts, hence a scalar multiplication.

Since we have the theta null point of level 2 on A, induced by the symplectic basis $(T'_1, \dots, T'_g, U'_1, \dots U'_g)$ of $A[4]$, the induced theta structure gives a canonical affine lift $\tilde{T}_i$ of the T_i, for all $i \in (\mathbb{Z}/2\mathbb{Z})^g$, and a canonical affine translation $\tilde{P} \mapsto \tilde{P} + \tilde{T}_i$ for all affine lifts $\tilde{P}$ of a point $P \in A$.

We take an arbitrary affine lift $\tilde{T''_i}$ of T''_i, which we then normalize via the equation

$$-3\tilde{T''_i} = \tilde{T''_i} + \tilde{T}_i, \tag{9}$$

where the translation on the right is the canonical one induced by the theta structure. This rigidifies our choice of affine lift up to a root of unity μ of order $3^2 - 1^2 = 8$. Then $2\tilde{T''_i}$ is rigidified up to the action of $\mu^{2^2} = \pm 1$, and so if we let $\tilde{T'_i} = 2\tilde{T''_i}$, its coordinates are fully determined up to a sign. We can now apply Eq. (8), with $z'_i = \tilde{T'_i}$:

$$\theta^B_i(0) = \sum_{t \in (\mathbb{Z}/2\mathbb{Z})^g} \theta^A_t(\tilde{T'_i})^2 = \sum_{t \in (\mathbb{Z}/2\mathbb{Z})^g} \theta^A_t(2\tilde{T''_i})^2. \tag{10}$$

Because this equation only involves the squares of the coordinates of $\tilde{T'_i}$, our remaining sign ambiguity does not matter.

Remark 62. If we only had the points T'_i but not the T''_i, we could rigidify the choice of $\tilde{T}'_i$ via the equation $2\tilde{T}'_i = \tilde{T}_i$. This equation rigidifies the lift up to a root of unity μ of order $2^2 = 4$, so it remains a sign ambiguity in Eq. (10). However it is enough to do this rigidification for i going through $e_1, \dots, e_g, e_1 + e_2, \dots, e_j + e_k, \dots, e_{g-1} + e_g$ where e_j is a a basis of $(\mathbb{Z}/2\mathbb{Z})^g$. The remaining choices of $\tilde{T}'_i$ are then fully determined from the Riemann relations, in particular from three way additions and differential additions. Thus we only have $g(g+1)/2$ signs ambiguity rather than 2^g, and one can prove that all signs are actually valid [Rob10, Proposition 6.3.5]: they each correspond from a different choice of a 4-symplectic basis above our fixed 2-symplectic basis $(f(T'_j), f(U_j))$.

Nevertheless, when computing an 2^e-isogeny, we need to be careful that this choice of 4-symplectic basis is compatible with our next kernel. Also in the end for the equality testing of Appendix F.3, we need to be sure to have chosen the correct theta structure. That's why we have to assume that we are given the T''_i, not only the T'_i, this allows us to fully rigidify the theta structure on B.

Proposition 63. *Let T''_j, $j = 1, \dots, g$ be points of 8-torsion on A which generates an isotropic subgroup. Let K be the kernel generate by the $4T''_j$. Assume that we are given the theta coordinates of the T''_j via the theta structure induced by a symplectic basis $(2T'_j, U'_j)$. Let $(\theta_i^B(0))_{i \in (\mathbb{Z}/2\mathbb{Z})^g}$ be the projective theta null point of $B = A/K$ given by Theorem 56.*

Then if $i \in (\mathbb{Z}/2\mathbb{Z})^g$, the value $\theta_i^B(0)$ (up to a constant which does not depend on i) can be computed using Eq. (10) where $\tilde{T}''_i$ is an affine lift of T_i normalised using Eq. (9). This requires tripling an affine lift of $T''_i = \sum_{j=1}^g i_j T''_j$, a division and multiplication, and 2^g squares. If the tripling is computed via a doubling followed by a differential addition, it can be done in 2^{g+2} multiplications, 2^{g+1} doubling, and 2^g divisions.

The total cost to compute the theta null point of B is then $(2^g - 1)(2^{g+2} + 1)$ multiplications, $2^g(2^{g+1} - 1)$ squares, $(2^g - 1)(2^g + 1)$ divisions, and 2^{g+1} inversions, that is $2^g(7 \cdot 2^g - 2) - 2$ arithmetic operations.

Proof. We use Eq. (10) to compute $\theta_i^B(0)$. We take an arbitrary lift $\tilde{T}''_i$ and compute $3\tilde{T}''_i$. We use Eq. (9) to compute the correct normalisation, this costs one division. We then plug in Eq. (10), this costs 2^g squares, and one multiplication by our normalisation factor.

But we remark that if $3\tilde{T}''_i$ is computed through a doubling $2\tilde{T}''_i$ followed by a differential addition $2\tilde{T}''_i + \tilde{T}''_i$, then the squares of the theta coordinates of $2\tilde{T}''_i$ are already computed.

The cost of the doubling and differential addition is described in [Rob10, Table 4.1]. The computation of $\theta_i^B(0)$ for $i \neq 0$ then costs $4 \cdot 2^g + 1$ multiplications, $2 \cdot 2^g$ squares and $2^g + 1$ divisions.

This costs assume the precomputation of some constants depending only on A (more precisely its theta null point), which takes 2^g squares and 2^{g+1} inverses to compute once and for all. Taking this precomputation into account we get our final complexity. □

For computing 2^e-isogenies decomposed as e 2-isogenies, we start with g points of 2^{e+2} torsions x'_j, and even with the 2^g points x'_i for $i \in (\mathbb{Z}/2\mathbb{Z})^g$. We compute the 2^g points of each kernel using Fig. 2. For each 2-isogeny we need to apply Proposition 63 to compute the isogenous theta null point, and also apply Proposition 59 2^g times to push the points through each isogeny, each image costing 2^{g+1} arithmetic operations by Corollary 60.

In this section we only described how the general isogeny formula adapt to the case $\ell = 2$, and omitted some specific optimisations available only for 2^n-isogenies. Our implementation of the verification in dimension 4 use these further optimisations, which are described in [Rob23a]. Also to simplify the implementation of the verification, we require our response σ to be of degree q such that $\ell^e - q$ is a prime congruent to 5 modulo 8 (rather than just 1 modulo 4).

A profiling of the high level implementation of the FastSQIsignHD verification indicates that to compute a 2^{142}-isogeny in dimension 4, costs around $2.7 \cdot 10^5$ multiplications, $1.3 \cdot 10^5$ squarings, 500 inversions and $6.5 \cdot 10^5$ additions. Using the estimates provided by [Lon23], this amounts to $70M$ CPU clock cycles. In comparison, the SQIsign NIST-I verification [The23b] costs $37M$ CPU clock cycles with the assembly optimizations of [Lon23]. Even though this is a rough estimate, such a comparison sounds promising for FastSQIsignHD: we expect a low level implementation of the current implementation to have a verification time roughly twice as slow as SQISign. Given that there are many optimisations we have yet to implement even in the high level version, we hope that we will be able to reach a similar verification time as in SQISign. Of course, only an actual implementation will give a concrete optimised verification time.

F.5 A verification time vs compactness trade-off.

To speed up the verification time, the signer (or any verifier) can expand the compact signature by outputting all e intermediates theta constants computed in the chain of ℓ-isogenies computed during the verification. In dimension g, a theta constant over $\mathbb{F}_{p^2}$ takes $2^g \log(p^2)$ bits. The chain can be verified using Corollary 60.

When $g = 4$, $\lambda = 128$, $\ell = 2$, $e = 128$ and p has 256 bits, storing each 128 theta constants then takes $128 \cdot 2^4 \cdot 512$ bits, that is 131kB. This is a much larger output than the $13/2\lambda \approx 832$ bits of the compressed signature, but by Corollary 60 the verification then takes only $e \cdot 2^{g+1} = 4096$ squares over $\mathbb{F}_{p^2}$ and $2e = 256$ Hadamard transforms (and a final linear change of variable to glue the theta structures at the end), so will be much faster than via the compact signature.

This allows for a verification time vs compactness trade-off. We remark that expanding the compact isogeny to allow for fast verification time can be done by anyone.