Four Dimensions Strike Back: Fast Isogeny Computation via Special Automorphisms

Jianming Lin[1,2], Chang-An Zhao[1,3], and Yuhao Zheng[1]

[1] School of Mathematics, Sun Yat-sen University,
Guangzhou, 510275, China.
linjm76@mail.sysu.edu.cn
zhaochan3@mail.sysu.edu.cn
zhengyh57@mail2.sysu.edu.cn
[2] School of Mathematics, Jiaying University,
Meizhou, 514015, China.
[3] Guangdong Key Laboratory of Information Security Technology,
Guangzhou 510006, P.R. China.

Abstract. Four-dimensional isogeny representations enable fast signing in SQIsignHD (Eurocrypt 2024) and efficient key generation and encryption in FastPOKÉ-4D, but their computational cost remains a bottleneck for verification and decryption. We present an algorithm that exploits two automorphisms of the structured abelian fourfolds underlying these representations. The first preserves the half-chain kernels and induces an action of the extra automorphism of Weil type $(2, 2)$ on the intermediate varieties. With compatible symplectic markings, its action becomes a permutation of theta coordinates. This allows us to apply the two-point techniques of MIKE and propagate only two kernel representatives through the generic isogeny steps. The second automorphism exchanges the two half-chain kernels. We determine the resulting middle involution explicitly as a Hadamard transform followed by a coordinate permutation, allowing the complete representation to be evaluated using one constructed half-chain and its stored dual-evaluation data. We apply this algorithm to SQIsignHD verification and FastPOKÉ-4D decryption. We also describe a fast heuristic SQIsignHD variant (denoted as F-SQIsignHD$_H$) with direct smooth-isogeny commitment generation, alongside an instantiation using Qlapoty for key generation and commitment. We provide C implementations and compare the resulting protocols with their two-dimensional variants. Our experimental results demonstrate that, with the Broadwell implementation, F-SQIsignHD$_H$ gains overall speedups of 1.66×–1.92× compared with SQIsign-v3.0 in terms of the total running time of the signing phase and verification across NIST-I, -III, and -V.

Keywords: Isogeny-Based Cryptography Four-Dimensional Isogenies Automorphisms SQIsignHD FastPOKÉ-4D

1 Introduction

Isogeny-based cryptography offers post-quantum constructions with compact public keys, ciphertexts, and signatures. An early milestone was Supersingular Isogeny Diffie–Hellman (SIDH), introduced by Jao and De Feo in 2011 [17]. Its key encapsulation counterpart, SIKE [8], subsequently became a candidate in the NIST Post-Quantum Cryptography Standardization Process. The security of this approach was overturned by the key-recovery attacks initiated in 2022 by Castryck and Decru [9] and Maino *et al.* [21], and further developed by Robert [30]. These attacks broke SIDH and SIKE, but also introduced techniques that have substantially changed the design of isogeny-based cryptography.

In particular, the use of Kani's lemma [18] in the SIDH attacks revealed how an elliptic-curve isogeny of non-smooth degree can be evaluated through a smooth-degree isogeny in higher dimension. This higher-dimensional representation provides an alternative to explicitly constructing the underlying one-dimensional isogeny. It has since become a central tool in both digital signatures and public-key encryption. Much of this progress has relied on two-dimensional isogenies, whose efficient arithmetic makes them particularly attractive for implementation. Examples include SQIsign2D-West [3] and related SQIsign variants [28,33,27], as well as FESTA [6], QFESTA [26], and POKÉ [5]. Improvements to the quaternionic ideal-to-isogeny conversion, including Qlapoti [7] and Qlapoty [15], have further strengthened this approach.

Four-dimensional representations offer a different set of advantages. In SQIsign HD [12], the signer supplies the degree of a response isogeny and its action on a suitable torsion basis, while the verifier checks this response through a four-dimensional representation. This division of work simplifies signing and permits fast signing algorithms, at the cost of more expensive verification. In public-key encryption, FastPOKÉ-4D [20] uses one-dimensional isogeny evaluations to avoid constructing a four-dimensional representation during key generation. Combined with its parameter selection and use of the quadratic twist, this gives favorable key-generation and encryption costs, while leaving the four-dimensional computation to decryption. The four-dimensional POKÉ framework also has a distinctive property with respect to adaptive attacks. Moriya's generalization of the MOXZ attack [23,22] applies to several FESTA and POKÉ variants in the presence of a suitable checking oracle. For POKÉ-4D, however, its scalar masking matrix prevents the extraction step used by the currently known attack.

Despite these advantages, four-dimensional isogeny computation remains the main obstacle to the practical competitiveness of SQIsignHD verification and FastPOKÉ-4D decryption. Existing algorithms use level-2 theta coordinates to compute chains of $(2, 2, 2, 2)$-isogenies [10]. In the split-and-dual approach [30], a representation is obtained by constructing two half chains and evaluating the

dual of one of them. A direct implementation constructs these chains independently and propagates four kernel generators through their generic steps. Consequently, improving the four-dimensional arithmetic can directly reduce the cost of the two operations that limit these protocols.

In this work, we exploit the additional structure of the four-dimensional representations arising in these applications. Their kernels are stable under a polarization-preserving automorphism on the product $E_1^2 \times E_2^2$. With an appropriate torsion marking, the induced automorphism acts by a permutation of theta coordinates, allowing us to apply the two-point techniques developed for MIKE [2]. A second automorphism relates the kernels of the two half chains. We determine the corresponding middle involution explicitly, so that both halves can be evaluated from the data of a single constructed chain. The two optimizations address different sources of redundant work: propagation of kernel points within a chain, and independent construction of the second half chain.

1.1 Our Contributions

Our main contributions are summarized as follows.

– **An efficient algorithm for structured four-dimensional representations.** We consider the Kani representations on $A_0 = E_1^2 \times E_2^2$ associated with an odd-degree isogeny $\varphi : E_1 \to E_2$ and integers satisfying $a_1^2 + a_2^2 + \deg(\varphi) = 2^{2a}$. We show that the automorphism

$$\mathcal{J}(X_1, X_2, Y_1, Y_2) = (-X_2, X_1, -Y_2, Y_1), \qquad \mathcal{J}^2 = [-1],$$

preserves the half-chain kernels and descends to the intermediate principally polarized abelian fourfolds, which are of Weil type $(2, 2)$. We construct compatible symplectic markings for which $\mathcal{J}$ acts on theta coordinates by reversing the four-bit coordinate index. The four kernel directions therefore occur in two automorphism orbits. Applying the MIKE codomain formulas [2], our algorithm TwoPointIsogChain propagates only two representatives through each generic step; the other two directions are recovered by coordinate permutation.
To share the two half chains, we use the automorphism

$$\mathcal{R}(X_1, X_2, Y_1, Y_2) = (X_1, -X_2, -Y_1, Y_2),$$

which satisfies $\mathcal{R}F\mathcal{R} = \widetilde{F}$ and exchanges their kernels. We derive the factorization

$$F = \mathcal{R} \circ \widehat{h_+} \circ \eta \circ h_+,$$

and determine η from the induced action on the marked 4-torsion. In the prescribed theta coordinates, η is a Hadamard transform followed by a fixed coordinate permutation. Thus, the dual traversal reuses the theta-null and gluing data stored during the construction of h_+, without independently constructing the other half chain. Combining these steps gives FourDimRepresent. The reduction from four propagated kernel points to two applies to

the generic part of the chain; initial gluing, evaluation points, and the reverse traversal are accounted for separately.

– **Applications to SQIsignHD and FastPOKÉ-4D.** We apply FourDimRepresent to SQIsignHD verification and FastPOKÉ-4D decryption. We consider two instantiations for SQIsignHD. The first, denoted by F-SQIsignHD$_H$, is a fast heuristic variant in the parameter regime of Section 4.1. It retains FastDoublePath for key generation but replaces FastCommit by direct sampling of a smooth 3-power commitment isogeny and its kernel ideal. This removes the repeated construction and evaluation of paired 2- and 3-isogeny paths from commitment generation, while retaining the relevant distributional heuristics. The second follows the ideal-to-isogeny approach described by Dartois [11], using Qlapoty [15] for key generation and commitment and the parameter family used by SQIsign-v3.0 [1]. This instantiation is denoted as F-SQIsignHD by Dartois in [11, Section 3], following Section 5. For FastPOKÉ-4D, the new algorithm evaluates the pushforward isogeny needed to recover the shared secret during decryption, leaving key generation, encryption, and the scalar masking mechanism unchanged.
– **C implementations and performance comparisons.** We implement the optimized four-dimensional arithmetic in C and integrate it into both SQIsignHD instantiations and FastPOKÉ-4D. Our evaluation compares the resulting protocols with their two-dimensional variants. In particular, we compare the combined signing and verification time of each SQIsignHD instantiation with SQIsign-v3.0, and the overhead of decryption for the optimized FastPOKÉ-4D with POKÉ. Our experiments show that, with the Broadwell backend, F-SQIsignHD and F-SQIsignHD$_H$ reduce the combined cost of one signature generation and one verification by 28.9%–39.9% and 39.9%–48.0%, respectively, compared with SQIsign-v3.0 across NIST-I, -III, and -V. These reductions correspond to overall speedups of 1.41×–1.66× and 1.66×–1.92×, respectively. Our optimization substantially narrows the verification-performance gap between SQIsignHD and SQIsign-v3.0, allowing the signing gains to outweigh the remaining verification overhead. Besides, our proposed algorithm reduces the decryption cost of FastPOKÉ-4D, deriving a 3.79× speedup over the original implementation at NIST-I. This brings its decryption performance comparable to POKÉ, substantially improving the practical competitiveness of FastPOKÉ-4D.

We note that Ray and Barreto [29] have independently obtained fine results on this topic. Both works exploit an automorphism of order four for two-point propagation. Our work additionally uses a second automorphism that exchanges the two half-chain kernels and determines the resulting middle involution explicitly as a Hadamard transform followed by a coordinate permutation. This allows the full representation to be evaluated from a single constructed half-chain and its stored data for dual evaluation. We also apply the resulting algorithm to FastPOKÉ-4D decryption and study two SQIsignHD instantiations. Our results were obtained independently.

1.2 Organization of This Paper

Section 2 recalls the mathematical background and the SQIsignHD and FastPOKÉ-4D protocols. Section 3 develops the automorphism-based algorithms for two-point propagation and half-chain reuse. Section 4 applies these algorithms to the two SQIsignHD instantiations and to FastPOKÉ-4D decryption. Section 5 presents the implementations and performance comparisons. Finally, Section 6 discusses the conclusions and directions for further work.

2 Preliminaries

This section establishes the mathematical background required for this paper, covering the Deuring correspondence, four-dimensional Kani's lemma, theta coordinates and $(2,2,2,2)$-isogeny, the SQIsignHD scheme, together with the POKÉ-4D PKE protocol.

Notations. Let S denote a finite set; $x \xleftarrow{\$} S$ indicates that x is sampled uniformly at random from S. We denote by KDF a key-derivation function. Let p be a prime of cryptographic size, i.e., $p \geq 2^\lambda$, where λ is the security parameter. The neutral element of an elliptic curve E is denoted by 0_E.

An isogeny $\varphi : E_1 \to E_2$ defined over $\mathbb{F}_{p^2}$ is a non-constant algebraic group homomorphism from $E_1(\overline{\mathbb{F}_{p^2}})$ to $E_2(\overline{\mathbb{F}_{p^2}})$ given by rational functions over $\mathbb{F}_{p^2}$ [31, III.4]. The degree of φ, denoted $\deg(\varphi)$, is its degree as a rational map. Define $\ker(\varphi)$ to be the kernel of φ. For a separable isogeny, which we assume throughout, it holds that $\deg(\varphi) = \#\ker(\varphi)$. We refer to an isogeny of degree n simply as an n-isogeny. Every isogeny $\varphi : E_1 \to E_2$ admits a unique *dual isogeny* $\widehat{\varphi} : E_2 \to E_1$ satisfying $\widehat{\varphi} \circ \varphi = \varphi \circ \widehat{\varphi} = [\deg(\varphi)]$.

Let $\varphi : E_1 \to E_2$ and $\psi : E_1 \to E_3$ be isogenies of coprime degrees. The *pushforward* of ψ under φ, denoted by $\psi' = [\varphi]_*\psi$, is the unique isogeny $\psi' : E_2 \to E_4$ with kernel $\ker(\psi') = \varphi(\ker(\psi))$. Conversely, ψ is called the *pullback* of ψ' by φ, denoted $\psi = [\varphi]^*\psi'$.

For a positive integer n coprime to p, the n-torsion subgroup of E over $\overline{\mathbb{F}_{p^2}}$ is defined as

$$E[n] = \left\{ P \in E\left(\overline{\mathbb{F}_{p^2}}\right) \mid [n]P = 0_E \right\}.$$

It is a well-known fact that $E[n] \cong \mathbb{Z}_n \oplus \mathbb{Z}_n$. Throughout this paper, we work exclusively with supersingular elliptic curves defined over $\mathbb{F}_{p^2}$, with $p \equiv 3 \pmod 4$.

2.1 Quaternion Algebras and the Deuring Correspondence

A quaternion algebra over $\mathbb{Q}$ with the basis $1, i, j, k$ is defined as follows:

$$H(a,b) = \mathbb{Q} + i\mathbb{Q} + j\mathbb{Q} + k\mathbb{Q},$$

where $i^2 = a, j^2 = b$ and $k = ij = -ji$. The endomorphism ring $\mathrm{End}(E)$ of a supersingular curve E over $\mathbb{F}_{p^2}$ is isomorphic to a maximal order of $\mathcal{B}_{p,\infty}$, which is the quaternion algebra ramified at p and ∞. Therefore, we are interested in $\mathcal{B}_{p,\infty}$.

Specifically, it yields that $\mathcal{B}_{p,\infty} = H(-1,-p)$ when $p \equiv 3 \bmod 4$. For an element $\alpha = a_1 + a_2 i + a_3 j + a_4 k \in \mathcal{B}_{p,\infty}$, its conjugate is given as $\overline{\alpha} = a_1 - a_2 i - a_3 j - a_4 k$. Define $tr(\alpha) = \alpha + \overline{\alpha}$ and $n(\alpha) = \alpha\overline{\alpha}$ to be the *reduced trace* and *reduced norm* of α, respectively.

A fractional ideal $I = \alpha_1\mathbb{Z} + \alpha_2\mathbb{Z} + \alpha_3\mathbb{Z} + \alpha_4\mathbb{Z}$, where $\langle \alpha_1, \alpha_2, \alpha_3, \alpha_4 \rangle = \mathcal{B}_{p,\infty}$, is a $\mathbb{Z}$-lattice of rank 4. The *norm* of I is defined as $n(I) = \gcd(n(\alpha) : \alpha \in I)$. An order $\mathcal{O}$ is a subring of $\mathcal{B}_{p,\infty}$, whose elements are called integral as their reduced norm and trace are in $\mathbb{Z}$. A left order of I is $\mathcal{O}_L(I) = \{\alpha \in \mathcal{B}_{p,\infty} \mid \alpha I \subset I\}$. Similarly, we can define the right order of I as $\mathcal{O}_R(I) = \{\alpha \in \mathcal{B}_{p,\infty} \mid I\alpha \subset I\}$. An integral ideal is contained in its left order, and can be represented as $I = \mathcal{O}_L(I)\alpha + \mathcal{O}_L(I)n(I), \alpha \in \mathcal{O}_L(I)$, or $I = \beta\mathcal{O}_R(I) + n(I)\mathcal{O}_R(I), \beta \in \mathcal{O}_R(I)$.

Given two ideals I and J, if $\mathcal{O}_R(I) = \mathcal{O}_L(J)$, then we can define their *product* as $IJ = \{\alpha_1\alpha_2 \mid \alpha_1 \in I,\ \alpha_2 \in J\}$. Besides, it holds that $n(IJ) = n(I)n(J)$. If there is an ideal I^{-1} such that $II^{-1} = \mathcal{O}_L(I)$ and $I^{-1}I = \mathcal{O}_R(I)$, then I is called invertible and I^{-1} is the inversion of I. Denote by $\overline{I} = \{\overline{\alpha} \mid \alpha \in I\}$ the conjugate of I. Then it follows that $I\overline{I} = n(I)\mathcal{O}_L(I)$ and $\overline{I}I = n(I)\mathcal{O}_R(I)$ if I is an invertible ideal. Additionally, the left order of $\overline{I}$ and the right order of I satisfy $\mathcal{O}_L(\overline{I}) = \mathcal{O}_R(I)$. Consequently, the inverses of I and the element $\alpha \in I$ can be represented as follows:

$$I^{-1} = \frac{\overline{I}}{n(I)},\ \alpha^{-1} = \frac{\overline{\alpha}}{n(\alpha)}.$$

Suppose that I and J are two left $\mathcal{O}$-ideals. If there is an element $\beta \in \mathcal{B}_{p,\infty}^*$ such that $I = J\beta$, then I and J are equivalent, denoted as $I \sim J$, and $\beta\mathcal{O}_R(I) = \mathcal{O}_R(J)\beta$. Similarly, if $I \sim J$ are two right $\mathcal{O}$-ideals, then there exists an element $\beta \in \mathcal{B}_{p,\infty}^*$ such that $I = \beta J$, and it follows that $\mathcal{O}_L(I)\beta = \beta\mathcal{O}_L(J)$.

Deuring [14] established the connection between the arithmetic of elliptic curves and quaternion algebras over $\mathbb{Q}$. The Deuring correspondence characterizes the equivalence between an isogeny $\varphi : E_0 \to E_1$ and an integral ideal I_φ that links $\mathrm{End}(E_0) \cong \mathcal{O}_0$ and $\mathrm{End}(E_1) \cong \mathcal{O}_1$, i.e.,

$$\mathcal{O}_L(I_\varphi) = \mathcal{O}_0,\ \mathcal{O}_R(I_\varphi) = \mathcal{O}_1.$$

Table 1 presents the properties of the Deuring correspondence.

Example 1. Let $p \equiv 3 \bmod 4$. Then the supersingular curve $E_0/\mathbb{F}_{p^2} : y^2 = x^3 + x$ with j-invariant 1728 has a known endomorphism ring $\mathrm{End}(E_0)$, which is isomorphic to the maximal order $\mathcal{O}_0 = \mathbb{Z}\left\langle 1, i, \frac{i+j}{2}, \frac{1+k}{2} \right\rangle, i^2 = -1, j^2 = -p, k = ij = -ji$ of $\mathcal{B}_{p,\infty}$. Let $\varepsilon : \mathcal{O}_0 \to \mathrm{End}(E_0)$ denote this isomorphism. For each $\theta = a + bi + cj + dk \in \mathcal{O}_0$, it holds that $\varepsilon(\theta) = a + b\varepsilon(i) + c\varepsilon(j) + d\varepsilon(i)\varepsilon(j)$.

More precisely, i and j correspond to the automorphism $\varepsilon(i) : (x, y) \mapsto (-x, \sqrt{-1}y)$ and the p-power Frobenius endomorphism $\varepsilon(j) : (x, y) \mapsto (x^p, y^p)$, respectively. To simplify the notation, we replace $\varepsilon(\theta)$ with θ to directly represent an endomorphism in the remainder of this paper.

Supersingular j-invariants over $\mathbb{F}_{p^2}$ $j(E)$	Maximal orders in $\mathcal{B}_{p,\infty}$ $\mathcal{O} \cong \mathrm{End}(E)$ (up to isomorphism)
An isogeny $\varphi : E \to E_1$	I_φ: left $\mathcal{O}$-ideal and right $\mathcal{O}_1$-ideal
$\theta \in \mathrm{End}(E_0)$	Principal ideal $\mathcal{O}\theta$
$\deg(\varphi)$	Norm $n(I_\varphi)$
Dual $\widehat{\varphi}$	Conjugate $\overline{I_\varphi}$
Isogenies $\varphi : E \to E_1, \psi : E \to E_1$	Equivalent Ideals $I_\varphi \sim I_\psi$
Supersingular j-invariants over $\mathbb{F}_{p^2}$	$Cl(\mathcal{O})$
Composition $\tau \circ \rho : E \to E_1 \to E_2$	Product $I_{\tau\circ\rho} = I_\rho \cdot I_\tau$

Table 1. The Deuring correspondence [16, Table 1].

2.2 Four-Dimensional Kani's Lemma and Split-And-Dual Technique

In this subsection, we introduce Kani's Lemma [18] in dimension four [12, Section 2.3], which is utilized to develop high-dimensional isogeny-based cryptographic protocols, such as SQIsignHD. We also provide the split-and-dual method that efficiently computes the four-dimensional isogenies. The following theorem gives how to embed one-dimensional isogenies into a four-dimensional commutative diagram.

Theorem 1 (Kani's Lemma in dimension four [18,12]). *Let $N = a_1^2 + a_2^2 + q$, and let $\varphi : E_1 \to E_2$ be an isogeny of degree q such that $\gcd(q, a_1^2 + a_2^2) = 1$. Given the following isogeny diamond:*

$$\begin{array}{ccc} E_1^2 & \xrightarrow{\alpha_1} & E_1^2 \\ {\scriptstyle \Phi}\downarrow & & \downarrow{\scriptstyle \Phi} \\ E_2^2 & \xrightarrow[\alpha_2]{} & E_2^2 \end{array}$$

Fig. 1. Commutative diagram of isogenies between E_1^2 and E_2^2.

Define $\Phi := Diag(\varphi, \varphi) : E_1^2 \to E_2^2$, and $\alpha_i := \begin{pmatrix} a_1 & a_2 \\ -a_2 & a_1 \end{pmatrix} \in End(E_i^2)$ for $i \in \{1, 2\}$. Then the following isogeny

$$F := \begin{pmatrix} \alpha_1 & \widetilde{\Phi} \\ -\Phi & \widetilde{\alpha_2} \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & \widehat{\varphi} & 0 \\ -a_2 & a_1 & 0 & \widehat{\varphi} \\ -\varphi & 0 & a_1 & -a_2 \\ 0 & -\varphi & a_2 & a_1 \end{pmatrix} \in End(E_1^2 \times E_2^2)$$

is a four-dimensional (N, N, N, N)-isogeny with kernel

$$\ker(F) = \{(\widetilde{\alpha_1}(P), \Phi(P)) \mid P \in E_1^2[N]\}.$$

In practice, we can take $N = 2^{2a}$ for efficiency purposes, and the isogeny F can split into two $(2^a, 2^a, 2^a, 2^a)$-isogenies $h_+ : E_1^2 \times E_2^2 \to A$, $\widehat{h_-} : E_1^2 \times E_2^2 \to A'$ as well as a polarized isomorphism $\iota : A \to A'$ such that

$$\ker(h_+) = \ker(F) \cap (E_1^2 \times E_2^2)[2^a],$$
$$\ker(\widehat{h_-}) = \ker(\widetilde{F}) \cap (E_1^2 \times E_2^2)[2^a].$$

Then evaluating $Q \in E_1$ via φ can be executed as

$$(-[a_1]Q, [a_2]Q, \varphi(Q), 0_{E_2})^T = -h_- \circ \iota \circ h_+(Q, 0_{E_1}, 0_{E_2}, 0_{E_2})^T.$$

This is called the split-and-dual approach, which was first developed in [30, Section 6.4]. By using this technique, one can accomplish a $(2^{2a}, 2^{2a}, 2^{2a}, 2^{2a})$-isogeny computation with only 2^a-torsion points available.

2.3 Theta Model and (2,2,2,2)-Isogenies

In this subsection, we recall the definition and basic properties of theta coordinates used in $(2, 2, 2, 2)$-isogeny computations. We refer to [13, Section 2] and [10, Section 2] for a detailed treatment.

Let k be a field with $\mathrm{char}(k) \neq 2$, and let A be a g-dimensional abelian variety over k. Let $\mathcal{L}$ be a symmetric line bundle on A, and denote translation by P by T_P. Working over $\bar{k}$, define

$$K(\mathcal{L}) = \{P \in A(\bar{k}) \mid T_P^* \mathcal{L} \cong \mathcal{L}\}.$$

The set

$$\mathcal{G}(\mathcal{L}) = \left\{(x, \phi_x) \mid x \in K(\mathcal{L}),\ \phi_x : \mathcal{L} \xrightarrow{\sim} T_x^* \mathcal{L}\right\}$$

has a natural group structure and is called the *Mumford theta group*. It fits into the exact sequence

$$1 \longrightarrow \bar{k}^* \longrightarrow \mathcal{G}(\mathcal{L}) \longrightarrow K(\mathcal{L}) \longrightarrow 0.$$

Suppose that $K(\mathcal{L})$ admits a decomposition into complementary maximal isotropic subgroups,

$$K(\mathcal{L}) = K(\mathcal{L})_1 \oplus K(\mathcal{L})_2.$$

Let $\delta = (d_1, \ldots, d_g)$ be the corresponding sequence of elementary divisors, and write

$$K(\delta) = \bigoplus_{i=1}^{g} \mathbb{Z}/d_i\mathbb{Z}, \qquad H(\delta) = K(\delta) \oplus \mathrm{Hom}(K(\delta), \bar{k}^*).$$

A *theta structure of type* δ is an isomorphism

$$\Theta^{\mathcal{L}} : \mathcal{G}(\delta) \xrightarrow{\sim} \mathcal{G}(\mathcal{L})$$

that acts identically on the central subgroup $\bar{k}^*$, where $\mathcal{G}(\delta)$ is the Heisenberg group associated with δ. The space

$$V(\delta) = \{h : K(\delta) \to \bar{k}\}$$

carries the unique irreducible representation of $\mathcal{G}(\delta)$, up to isomorphism, on which $\bar{k}^*$ acts by scalar multiplication. Similarly, $\Gamma(A, \mathcal{L})$ carries the unique irreducible representation of $\mathcal{G}(\mathcal{L})$ with this central action. Thus, $\Theta^{\mathcal{L}}$ induces an equivariant isomorphism

$$\beta : V(\delta) \xrightarrow{\sim} \Gamma(A, \mathcal{L}),$$

unique up to a nonzero scalar. The images of the Kronecker delta basis $(\delta_i)_{i \in K(\delta)}$ under β form a basis $(\theta_i)_{i \in K(\delta)}$ of $\Gamma(A, \mathcal{L})$, whose evaluations give the *theta coordinates*. When $d_1 = \cdots = d_g = n$, these coordinates are said to have *level* n.

Throughout this paper, we use level-2 theta coordinates on abelian fourfolds. A point $P \in A(\bar{k})$ is represented by

$$\theta^A(P) = \big(\theta_i^A(P)\big)_{i \in (\mathbb{Z}/2\mathbb{Z})^4} \in \mathbb{P}^{15}.$$

The projective point

$$\mathbf{a}^A = \theta^A(0_A) = \big(\theta_i^A(0_A)\big)_{i \in (\mathbb{Z}/2\mathbb{Z})^4}$$

is called the *theta-null point*. Let $S_1, \ldots, S_4$ and $T_1, \ldots, T_4$ be the points of $A[2]$ corresponding to the standard basis of $K(2,2,2,2)$ and its dual basis in $\mathrm{Hom}(K(2,2,2,2), \bar{k}^*)$, respectively, under the induced isomorphism

$$\overline{\Theta}^{\mathcal{L}} : H(2,2,2,2) \xrightarrow{\sim} K(\mathcal{L}) = A[2].$$

The theta structure Θ^A is determined by a symplectic basis

$$(S_1', S_2', S_3', S_4', T_1', T_2', T_3', T_4')$$

of $A[4]$ lifting these points, with $[2]S_i' = S_i$ and $[2]T_i' = T_i$ [25, Remark 3, p. 319]. For $P \in A(\bar{k})$ and $T \in A[2]$ corresponding to $(s, \chi) \in H(2,2,2,2)$, translation by T acts on theta coordinates as

$$\big(\theta_i^A(P+T)\big)_i \sim \big(\chi(i)\theta_{i+s}^A(P)\big)_i,$$

where $\sim$ denotes equality up to a common nonzero scalar.

Hadamard transform and duplication formulas. The Hadamard transform and duplication formulas underlie $(2,2,2,2)$-isogeny evaluation. For a coordinate vector $x = (x_i)_{i \in (\mathbb{Z}/2\mathbb{Z})^4}$, define

$$\mathcal{H}(x)_i = \sum_{j \in (\mathbb{Z}/2\mathbb{Z})^4} (-1)^{i^{\mathsf{T}} j} x_j, \ \mathcal{S}(x) = (x_i^2)_{i \in (\mathbb{Z}/2\mathbb{Z})^4}.$$

The *dual theta coordinates* are defined by $\big(\widetilde{\theta}_i^A(P)\big)_i = \mathcal{H}\big((\theta_i^A(P))_i\big)$. Since $\mathcal{H}^2 = 16\,\mathrm{id}$, the Hadamard transform is a projective involution.

Let A be endowed with the type-$(2,2,2,2)$ theta structure induced by the symplectic basis $(S_1', \ldots, S_4', T_1', \ldots, T_4')$ of $A[4]$. Consider a $(2,2,2,2)$-isogeny $f : A \to B$ with $\ker(f) = K(\mathcal{L})_2 = \langle T_1, T_2, T_3, T_4 \rangle$. Write $\star$ for coordinatewise multiplication, $(x_i)_i \star (y_i)_i = (x_i y_i)_i$. With compatible theta structures, the

duplication identities [25,13,10] give

$$\left(\theta_i^A(P+Q)\right)_i \star \left(\theta_i^A(P-Q)\right)_i = \mathcal{H}\left(\left(\widetilde{\theta}_i^B(f(P))\right)_i \star \left(\widetilde{\theta}_i^B(f(Q))\right)_i\right), \quad (1)$$

$$\mathcal{H}\left(\left(\theta_i^A(\widetilde{f}(R))\right)_i \star \left(\theta_i^A(\widetilde{f}(S))\right)_i\right) = \left(\widetilde{\theta}_i^B(R+S)\right)_i \star \left(\widetilde{\theta}_i^B(R-S)\right)_i, \quad (2)$$

for $P, Q \in A(\bar{k})$ and $R, S \in B(\bar{k})$, where $\widetilde{f}$ denotes the polarized dual of f.

2.4 The SQIsignHD Signature Scheme

In this subsection, we introduce the digital signature scheme SQIsignHD, which uses four-dimensional isogenies in the verification phase. For simplicity, we concentrate on the fast variant of SQIsignHD (making use of 2- and 3- isogenies) called F-SQIsignHD, whose security is based on heuristic assumptions for computational distributions.

The public parameters fix a prime $p = c\,2^f 3^{f'} - 1$, a response bound 2^e, and a supersingular elliptic curve $E_0/\mathbb{F}_{p^2} : y^2 = x^3 + x$ with known endomorphism ring $\mathcal{O}_0$.

Key generation. FastKeyGen [12, Section 3.3] calls FastDoublePath [12, Algorithm 1] to sample two secret isogenies $\tau, \tau' \colon E_0 \to E_{\mathsf{pk}}$ with the same codomain. Their degrees divide $3^{2f'}$ and 2^{2f}, respectively. The public key is E_{pk}. The secret key consists of τ, τ' and their kernel ideals $I_\tau, I_{\tau'}$.

Signing. To sign a message m, the signer runs FastCommit [12, Remark 3.1] to sample an isogeny $\psi \colon E_0 \to E_{\mathsf{com}}$ with $\deg(\psi) \mid 3^{2f'}$ and its kernel ideal I_ψ. The Fiat–Shamir hash of the commitment E_{com} and the message m determines a cyclic subgroup $K \subset E_{\mathsf{pk}}$ of order $3^{f'}$, giving the challenge isogeny $\varphi \colon E_{\mathsf{pk}} \to E_{\mathsf{chl}} = E_{\mathsf{pk}}/K$. Using the secret key and the commitment, FastRespond [12, Algorithm 2] samples an ideal in the class corresponding to $\varphi \circ \tau \circ \widehat{\psi}$, where $\widehat{\psi}$ is the dual of ψ. The associated response isogeny $\sigma \colon E_{\mathsf{com}} \to E_{\mathsf{chl}}$ has degree q coprime to 3, with $2^{2a} - q$, $a \leq f$ prime and congruent to 1 modulo 4. For the canonical basis (P_1, P_2) of $E_{\mathsf{com}}[2^f]$, the signer computes $R_i = \sigma(P_i)$ and outputs the signature $(E_{\mathsf{com}}, q, R_1, R_2)$ in uncompressed form.

Verification. Given E_{pk}, m, and $(E_{\mathsf{com}}, q, R_1, R_2)$, the verifier recomputes φ and E_{chl}. It checks the conditions on q and that $R_1, R_2 \in E_{\mathsf{chl}}[2^f]$, recovers (P_1, P_2) from E_{com}, and writes $2^{2a} - q = a_1^2 + a_2^2$, $a_i \in \mathbb{Z}$. From these data, FastVerify [12, Algorithm 3] constructs a chain of four-dimensional $(2,2,2,2)$-isogenies on $E_{\mathsf{com}}^2 \times E_{\mathsf{chl}}^2$ using the split-and-dual method mentioned in Section 2.2. It accepts exactly when the chain represents an isogeny $E_{\mathsf{com}} \to E_{\mathsf{chl}}$ of degree q sending P_i to R_i for $i = 1, 2$.

SQIsignHD possesses a faster signing phase but a much more expensive verification compared with the SQIsign2D variants due to the costly four-dimensional isogeny computation.

2.5 The FastPOKÉ-4D Protocol

In this subsection, we present FastPOKÉ-4D [20], a variant of POKÉ PKE that uses four-dimensional isogenies in the decryption phase. The difference between FastPOKÉ-4D and POKÉ-4D is that the former exploits one-dimensional isogenies to avoid the expensive four-dimensional representation computation during key generation, thereby enabling faster key generation and encryption compared with the original POKÉ. In the following, we provide a concrete procedure for FastPOKÉ-4D.

Let the public parameters consist of a prime $p = f2^aD - 1$ with $3^b \mid p - 1$, a supersingular curve $E_0/\mathbb{F}_{p^2} : y^2 = x^3 + x$ with known endomorphism ring, bases (P_0, Q_0) and (R_0, S_0) of $E_0[2^a]$ and $E_0[3^b]$, respectively, and a point X_0 of order D. The 3^b-torsion subgroup is represented using the quadratic twist. Fix a key-derivation function KDF whose output length equals the message length.

Key generation. During the key generation, $\mathsf{TwinPathGenIso}$ [20, Algorithm 5] produces a curve E_A, four-dimensional representation parameters (q, a_1, a_2) satisfying $2^{2a-4} - q = a_1^2 + a_2^2$, $\gcd(q, 6D) = 1$, and the images of the public torsion points under a secret q-isogeny $\varphi\colon E_0 \to E_A$. The receiver masks these images using independently sampled invertible scalars α_2, γ_3, and δ_D on the 2^a-, 3^b-, and D-torsion, respectively. The public key contains E_A and the masked images, while the secret key is $\mathsf{sk} = (q, a_1, a_2, \alpha_2, \delta_D)$.

Encryption. The Encryption constructs two compatible 3^b-isogenies $\psi\colon E_0 \to E_B$ and $\psi'\colon E_A \to E_{AB}$ from a randomly chosen kernel and its image under φ. The sender evaluates these isogenies on the corresponding public points and applies fresh masks ω_2 and η_D to the 2^a-torsion bases and the D-torsion points. The resulting shared point is $X_{AB} = [\eta_D]\psi'(X_A)$, from which the sender derives the message mask $\mathsf{KDF}(X_{AB})$. The ciphertext contains the resulting curves, masked torsion data, and $m \oplus \mathsf{KDF}(X_{AB})$.

Decryption. For the decryption, the receiver removes the mask α_2 to recover the action of the pushforward isogeny $\varphi' = [\psi]_*\varphi$ on a basis of $E_B[2^a]$. Together with (a_1, a_2), these images determine a four-dimensional representation of φ', computed through two half $(2^{a-2}, 2^{a-2}, 2^{a-2}, 2^{a-2})$-isogenies using the split-and-dual approach in Section 2.2. Evaluating this representation at the ciphertext point X_B recovers $\varphi'(X_B)$ and hence $X_{AB} = [\delta_D]\varphi'(X_B)$. The receiver then recovers the message using the same derived mask.

Algorithms 1–3 give the procedures for the key generation, encryption and decryption of FastPOKÉ-4D.

Correctness follows from $\varphi' \circ \psi = \psi' \circ \varphi$, where $\varphi' = [\psi]_*\varphi$. The decryption algorithm recovers $Y = \varphi'(X_B)$, and hence $[\delta_D]Y = [\eta_D]\psi'(X_A) = X_{AB}$. Compared with the traditional POKÉ-4D, FastPOKÉ-4D avoids time-consuming high-dimensional isogeny construction for determining the four-dimensional representation during the key generation. Nevertheless, the decryption phase still

Algorithm 1 Key Generation of FastPOKÉ-4D [20, Algorithm 6]

Input: E_0, $P_0, Q_0 \in E_0[2^a]$, $R_0, S_0 \in E_0[3^b]$, and $X_0 \in E_0[D]$.
Output: A secret/public key pair.

1: Compute the tuple $(E_A, q, a_1, a_2, \varphi(P_0), \varphi(Q_0), \varphi(R_0), \varphi(S_0), \varphi(X_0)) \leftarrow \mathsf{TwinPathGenIso}(E_0, P_0, Q_0, R_0, S_0, X_0)$. ▷ Using [20, Algorithm 5]
2: Sample a random $\alpha_2 \xleftarrow{\$} \mathbb{Z}_{2^a}^{\times}$.
3: Sample a random $\gamma_3 \xleftarrow{\$} \mathbb{Z}_{3^b}^{\times}$.
4: Sample a random $\delta_D \xleftarrow{\$} \mathbb{Z}_D^{\times}$.
5: Compute $P_A, Q_A \leftarrow [\alpha_2]\varphi(P_0), [\alpha_2]\varphi(Q_0)$.
6: Compute $R_A, S_A \leftarrow [\gamma_3]\varphi(R_0), [\gamma_3]\varphi(S_0)$.
7: Compute $X_A \leftarrow [\delta_D]\varphi(X_0)$.
8: **return** $\mathsf{sk} = (q, a_1, a_2, \alpha_2, \delta_D)$, $\mathsf{pk} = (E_A, P_A, Q_A, R_A, S_A, X_A)$.

Algorithm 2 Encryption of FastPOKÉ-4D [20, Algorithm 7]

Input: A message m, a public key $\mathsf{pk} = (E_A, P_A, Q_A, R_A, S_A, X_A)$
Output: A ciphertext ct.

1: Sample a random $r_3 \xleftarrow{\$} \mathbb{Z}_{3^b}$.
2: Compute the isogeny $\psi : E_0 \to E_B$ with kernel $\langle R_0 + [r_3]S_0 \rangle$.
3: Compute the isogeny $\psi' : E_A \to E_{AB}$ with kernel $\langle R_A + [r_3]S_A \rangle$.
4: Compute $\psi(P_0), \psi(Q_0)$ and $\psi(X_0)$.
5: Compute $\psi'(P_A), \psi'(Q_A)$ and $\psi'(X_A)$.
6: Sample a random $\omega_2 \xleftarrow{\$} \mathbb{Z}_{2^a}^{\times}$.
7: Compute $P_B, Q_B = [\omega_2]\psi(P_0), [1/\omega_2]\psi(Q_0)$.
8: Compute $P_{AB}, Q_{AB} = [\omega_2]\psi'(P_A), [1/\omega_2]\psi'(Q_A)$.
9: Sample a random $\eta_D \xleftarrow{\$} \mathbb{Z}_D^{\times}$.
10: Compute $X_B, X_{AB} = [\eta_D]\psi(X_0), [\eta_D]\psi'(X_A)$.
11: Compute $\mathsf{ct}' = \mathrm{KDF}(X_{AB}) \oplus m$.
12: **return** $\mathsf{ct} = ((E_B, P_B, Q_B, X_B), (E_{AB}, P_{AB}, Q_{AB}), \mathsf{ct}')$.

need to compute the expensive four-dimensional isogeny chain, which is much more expensive than the two-dimensional isogeny construction and limits the practicality of FastPOKÉ-4D.

3 Efficient Four-Dimensional Isogeny Computation Using Automorphisms

In this section, we establish systematic theory and a concrete new algorithm for efficiently computing and evaluating four-dimensional representations between specific four-dimensional abelian varieties as detailed in Section 2.2, which are employed in some isogeny-based cryptographic schemes, including SQIsignHD

Algorithm 3 Decryption of FastPOKÉ-4D [20, Algorithm 8]

Input: The secret key $\mathsf{sk} = (q, a_1, a_2, \alpha_2, \delta_D)$ with $2^{2a-4} - q = a_1^2 + a_2^2$, and the ciphertext $\mathsf{ct} = ((E_B, P_B, Q_B, X_B), (E_{AB}, P_{AB}, Q_{AB}), \mathsf{ct}')$.

Output: A message m'.

1: $P'_{AB} \leftarrow [1/\alpha_2]P_{AB},\ Q'_{AB} \leftarrow [1/\alpha_2]Q_{AB}$.
2: $K'_1 \leftarrow ([a_1]P_B, [a_2]P_B, P'_{AB}, 0_{E_{AB}}),\ K'_2 \leftarrow (-[a_2]P_B, [a_1]P_B, 0_{E_{AB}}, P'_{AB})$.
3: $K'_3 \leftarrow ([a_1]Q_B, [a_2]Q_B, Q'_{AB}, 0_{E_{AB}}), K'_4 \leftarrow (-[a_2]Q_B, [a_1]Q_B, 0_{E_{AB}}, Q'_{AB})$.
4: $K''_1 \leftarrow ([a_1]P_B, -[a_2]P_B, -P'_{AB}, 0_{E_{AB}}), K''_2 \leftarrow ([a_2]P_B, [a_1]P_B, 0_{E_{AB}}, -P'_{AB})$.
5: $K''_3 \leftarrow ([a_1]Q_B, -[a_2]Q_B, -Q'_{AB}, 0_{E_{AB}}), K''_4 \leftarrow ([a_2]Q_B, [a_1]Q_B, 0_{E_{AB}}, -Q'_{AB})$.
6: Compute $F_1 : E_B^2 \times E_{AB}^2 \to \mathcal{A}$ with kernel $\langle [4]K'_1, [4]K'_2, [4]K'_3, [4]K'_4 \rangle$.
7: Compute $\widetilde{F_2} : E_B^2 \times E_{AB}^2 \to \mathcal{A}$ with kernel $\langle [4]K''_1, [4]K''_2, [4]K''_3, [4]K''_4 \rangle$.
8: $(-[a_1]X_B, [a_2]X_B, \varphi'(X_B), 0_{E_{AB}}) \leftarrow -\widetilde{\widetilde{F_2}} \circ \iota \circ F_1(X_B, 0_{E_B}, 0_{E_{AB}}, 0_{E_{AB}})$.
▷ ι is the middle polarized involution
9: Compute $X_{AB} \leftarrow [\delta_D]\varphi'(X_B)$.
10: Compute $m' = \mathsf{KDF}(X_{AB}) \oplus \mathsf{ct}'$.
11: **return** m'.

and FastPOKÉ-4D. Inspired by MIKE [2], we prove that the four-dimensional PPAV used in SQIsignHD has an extra automorphism $\mathcal{J}$ and is of Weil type $(2, 2)$. By choosing appropriate symplectic bases, we can view the automorphism's action as a coordinate permutation, and we can evaluate the generic isogeny by updating only two kernel points. Furthermore, we use another automorphism $\mathcal{R}$ to determine the involution between the medium abelian varieties, thereby sharing the constructions of two half-isogeny chains, h_+ and $\widehat{h_-}$. Combining these two techniques, our method achieves a nearly $4\times$ speedup in computing the four-dimensional representation.

With the same notation as in Section 2.2, the goal of this section is to accelerate the following fundamental building block:

Four-Dimensional Representation Computation. Define $n = 2^a, n' = 4n = 2^{a+2}, a \geq 2$. Denote by P, Q a basis of $E_1[n']$, and $(P', Q') = (\varphi(P), \varphi(Q))$, where $\varphi : E_1 \to E_2$ has odd degree q and $a_1^2 + a_2^2 + q = n^2$. Then it holds that $\langle P', Q' \rangle = E_2[n']$. Let $A_0 = E_1^2 \times E_2^2$, $R \in E_1$, and $R' = (\widetilde{R}, 0_{E_1}, 0_{E_2}, 0_{E_2})$. Given A_0, P, Q, P', Q', R', compute two (n, n, n, n)-isogenies $h_+ : A_0 \to A$ and $\widehat{h_-} : A_0 \to A'$ with kernel

$$H_+ := \ker(h_+) = \ker(F) \cap A_0[n] = \{(\alpha_1^{\mathrm{T}}(\widetilde{P}), \Phi(\widetilde{P})) \mid \widetilde{P} \in E_1^2[n]\},$$
$$H_- := \ker(\widehat{h_-}) = \ker(\widetilde{F}) \cap A_0[n] = \{(\alpha_1(\widetilde{P}), -\Phi(\widetilde{P})) \mid \widetilde{P} \in E_1^2[n]\}$$

to derive $F = h_- \circ \iota \circ h_+$ and $F(R')$ $(\varphi(\widetilde{R}))$. It is worth noting that the polarized isomorphism ι is fast to compute.

The previous method is to compute separately two half (n, n, n, n)-isogeny chains h_+, $\widehat{h_-}$ to derive $h_+(R')$, and then evaluate ι at $h_+(R')$ and $h_- = \widehat{\widehat{h_-}}$ at

$\iota(h_+(R'))$ through the intermediate results derived from the construction of $\widehat{h_-}$. More precisely, each computation of the generic isogenies in the construction of h_+ and $\widehat{h_-}$ needs to update four points. In the following, we present the details of the improved approach for the four-dimensional isogeny computation.

3.1 Using the Specific Automorphisms

In this subsection, we employ the technique of extra automorphisms to halve the required number of updating points involved in the generic isogenies of h_+ and $\widehat{h_-}$. The technical framework is that

- Construct the automorphism $\mathcal{J} \in \mathrm{Aut}(A_0)$ and prove that the middle abelian variety A and A' is of Weil type $(2,2)$;
- Find a suitable symplectic basis such that the action of $\mathcal{J}$ on such a basis is a signed permutation inducing a theta-coordinate permutation, and thus the theta-null point in the generic isogenies can be determined by only two points.

The definition of Weil type $(2,2)$ is given as follows.

Definition 1. *[2, Definition 4.1] Let X denote a four-dimensional principally polarized abelian variety over $\overline{\mathbb{F}}_p$, with $p \neq 2$. Then X is of Weil type $(2,2)$ if*

- *The fourfold X admits an embedding $\tau : \mathbb{Z}[i] \hookrightarrow End(X)$, where $i^2 = -1$, satisfying $\tau(i)^\dagger = -\tau(i)$ for the Rosati involution of the principal polarization;*
- *For a suitable basis, the action of $\tau(i)$ on the tangent space T_0X at the origin of X is given by $\mathrm{diag}(\zeta, \zeta, -\zeta, -\zeta)$, where $\zeta \in \overline{\mathbb{F}}_p$ satisfies $\zeta^2 = -1$.*

In the following, we prove that the abelian fourfolds A and A' are of Weil type $(2,2)$, which is summarized in the following proposition.

Proposition 1. *With the notation above, let*

$$h_+ \colon A_0 \longrightarrow A, \quad \widehat{h_-} \colon A_0 \longrightarrow A'$$

be the two half (n,n,n,n)-isogenies. Then the abelian fourfolds A and A' are of Weil type $(2,2)$.

Proof. We prove the assertion for A, the argument for A' is analogous. Throughout this paper, we work over $\overline{\mathbb{F}}_p$, with characteristic $p \neq 2$. Define an automorphism of $A_0 = E_1^2 \times E_2^2$ by

$$\mathcal{J}(X_1, X_2, Y_1, Y_2) = (-X_2, X_1, -Y_2, Y_1), \quad X_i \in E_1, \quad Y_i \in E_2.$$

By construction, we have $\mathcal{J}^2 = [-1]$. Hence the map

$$\begin{aligned} \tau \colon \mathbb{Z}[i] &\longrightarrow \mathrm{End}(A_0), \\ r + si &\longmapsto [r] + [s]\mathcal{J} \end{aligned}$$

defines a ring homomorphism. To prove that τ is injective, suppose that $[r] + [s]\mathcal{J} = 0$. Then $\big([r] + [s]\mathcal{J}\big)\big([r] - [s]\mathcal{J}\big) = [r^2 + s^2] = 0$. Since multiplication by a nonzero integer is a nonzero isogeny of A_0, we obtain $r = s = 0$. Thus τ is an embedding, giving A_0 an action of $\mathbb{Z}[i]$.

We next verify that $\mathcal{J}$ preserves the product principal polarization

$$\lambda_0 = \lambda_{E_1} \times \lambda_{E_1} \times \lambda_{E_2} \times \lambda_{E_2},$$

where $\lambda_{E_i} \colon E_i \to E_i^\vee$ is the canonical principal polarization of E_i. Relative to the product decomposition of A_0, the automorphism $\mathcal{J}$ is represented by

$$J = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{3}$$

Each of its two diagonal blocks consists of integer endomorphisms acting on a pair of identical elliptic factors. On these blocks, the Rosati involution associated with λ_0 (denoted as $\dagger_0$) is therefore given by transposition. Consequently, from Eq. (3) it follows that

$$J^{\dagger_0} = J^{\mathsf{T}} = -J, \quad J^{\dagger_0} J = -J^2 = I_4.$$

Equivalently, it yields that $\mathcal{J}^* \lambda_0 = \mathcal{J}^\vee \circ \lambda_0 \circ \mathcal{J} = \lambda_0$. We now show that $\mathcal{J}$ descends to an automorphism of the intermediate fourfold $A = A_0/H_+$. Suppose first that $\mathcal{J}(H_+) = H_+$. We may then define

$$\mathcal{J}_{H_+}\big(h_+(Z)\big) := h_+\big(\mathcal{J}(Z)\big), \quad Z \in A_0.$$

Indeed, if $h_+(Z) = h_+(Z')$, then $Z - Z' \in H_+$, and stability under $\mathcal{J}$ gives

$$\mathcal{J}(Z - Z') \in H_+.$$

It follows that $h_+\big(\mathcal{J}(Z)\big) = h_+\big(\mathcal{J}(Z')\big)$. By the universal property of the quotient, this construction defines a unique endomorphism $\mathcal{J}_{H_+}$ of A satisfying

$$\mathcal{J}_{H_+} \circ h_+ = h_+ \circ \mathcal{J}. \tag{4}$$

Moreover, we have

$$\mathcal{J}_{H_+}^2 \circ h_+ = \mathcal{J}_{H_+} \circ h_+ \circ \mathcal{J} = h_+ \circ \mathcal{J}^2 = [-1] \circ h_+.$$

Since h_+ is surjective, we obtain $\mathcal{J}_{H_+}^2 = [-1]$. In particular, $\mathcal{J}_{H_+}$ is an automorphism on A, with inverse $-\mathcal{J}_{H_+}$. We still need to verify the required stability of H_+. Set

$$U_+ = \big([a_1]P, [a_2]P, P', 0_{E_2}\big), \quad V_+ = \big([a_1]Q, [a_2]Q, Q', 0_{E_2}\big).$$

By the definition of H_+, it holds that

$$H_+ = \langle [4]U_+, [4]\mathcal{J}U_+, [4]V_+, [4]\mathcal{J}V_+ \rangle.$$

Using $\mathcal{J}^2 = [-1]$, we find

$$\begin{aligned}\mathcal{J}(H_+) &= \langle [4]\mathcal{J}U_+, [4]\mathcal{J}^2U_+, [4]\mathcal{J}V_+, [4]\mathcal{J}^2V_+\rangle \\ &= \langle [4]\mathcal{J}U_+, -[4]U_+, [4]\mathcal{J}V_+, -[4]V_+\rangle \\ &= H_+.\end{aligned}$$

Thus the preceding construction applies. We next prove that $\mathcal{J}_{H_+}$ preserves the principal polarization λ_+ on A. By construction,

$$h_+^*\lambda_+ = h_+^\vee \circ \lambda_+ \circ h_+ = h_+^\vee \circ (h_+^\dagger)^\vee \circ \lambda_0 = [n]_{A_0^\vee}\lambda_0.$$

Together with Eq. (4), this gives

$$\begin{aligned}h_+^*\big(\mathcal{J}_{H_+}^*\lambda_+\big) &= (\mathcal{J}_{H_+} \circ h_+)^*\lambda_+ \\ &= (h_+ \circ \mathcal{J})^*\lambda_+ \\ &= \mathcal{J}^*(h_+^*\lambda_+) \\ &= \mathcal{J}^*([n]_{A_0^\vee}\lambda_0) \\ &= [n]_{A_0^\vee}\lambda_0 = h_+^*\lambda_+.\end{aligned}$$

The pullback map on polarization homomorphisms is injective: if a homomorphism $\mu\colon A \to A^\vee$ satisfies $h_+^\vee \mu h_+ = 0$, then $\mu = 0$, since h_+ and $h_+^\vee$ are invertible in the corresponding rational homomorphism spaces. We therefore obtain $\mathcal{J}_{H_+}^*\lambda_+ = \mathcal{J}_{H_+}^\vee \circ \lambda_+ \circ \mathcal{J}_{H_+} = \lambda_+$. Writing $\dagger$ for the Rosati involution associated with λ_+, we conclude that

$$\mathcal{J}_{H_+}^\dagger \mathcal{J}_{H_+} = \lambda_+^{-1} \circ \mathcal{J}_{H_+}^\vee \circ \lambda_+ \circ \mathcal{J}_{H_+} = \mathrm{id}_A$$

Since $\mathcal{J}_{H_+}^2 = [-1]$, it follows that $\mathcal{J}_{H_+}^\dagger = \mathcal{J}_{H_+}^{-1} = -\mathcal{J}_{H_+}$. The relation $\mathcal{J}_{H_+}^2 = [-1]$ defines a ring homomorphism

$$\begin{aligned}\tau_+\colon \mathbb{Z}[i] &\longrightarrow \mathrm{End}(A), \\ r + si &\longmapsto [r] + [s]\mathcal{J}_{H_+}.\end{aligned}$$

The same argument used for τ proves that τ_+ is injective. Moreover, its extension to $\mathbb{Q}(i)$ satisfies $\tau_+(\alpha)^\dagger = \tau_+(\overline{\alpha}),\ \alpha \in \mathbb{Q}(i)$. Thus the induced $\mathbb{Z}[i]$-action on A is compatible with the principal polarization.

Finally, we compute the action on the tangent space. We have

$$T_0A_0 = T_0E_1 \oplus T_0E_1 \oplus T_0E_2 \oplus T_0E_2.$$

Choosing the same tangent-space basis on the two copies of each elliptic factor, the differential $d\mathcal{J}$ is represented by the matrix J in Eq. (3). Choose $\zeta \in \overline{\mathbb{F}}_p$ such that $\zeta^2 = -1$. Then

$$\chi_{d\mathcal{J}}(T) = (T^2+1)^2 = (T-\zeta)^2(T+\zeta)^2.$$

Since $p \neq 2$ and $(d\mathcal{J})^2 = -I_4$, the operator $d\mathcal{J}$ is diagonalizable. Hence

$$\dim_{\overline{\mathbb{F}}_p} \ker(d\mathcal{J} - \zeta I_4) = \dim_{\overline{\mathbb{F}}_p} \ker(d\mathcal{J} + \zeta I_4) = 2.$$

As $\gcd(\deg(h_+), p) = 1$, the isogeny h_+ is separable, and its differential

$$dh_+ \colon T_0 A_0 \longrightarrow T_0 A$$

is an isomorphism. Differentiating Eq. (4) yields $d\mathcal{J}_{H_+} \circ dh_+ = dh_+ \circ d\mathcal{J}$, and therefore

$$d\mathcal{J}_{H_+} = dh_+ \circ d\mathcal{J} \circ (dh_+)^{-1}.$$

Thus $d\mathcal{J}_{H_+}$ is conjugate to $d\mathcal{J}$. In particular, $\chi_{d\mathcal{J}_{H_+}}(T) = (T-\zeta)^2(T+\zeta)^2$, and, in a suitable basis of $T_0 A$, we have

$$d\mathcal{J}_{H_+} = \mathrm{diag}(\zeta, \zeta, -\zeta, -\zeta).$$

The two eigenspaces therefore both have dimension two. Together with the polarization compatibility established above, this proves that A is of Weil type $(2,2)$. Applying the same argument to $\widehat{h_-}$ proves the assertion for A'. □

We now construct an explicit symplectic basis such that the automorphism $\mathcal{J}$ acts on this basis as a signed permutation. Let $c = a_1^2 + a_2^2$. Define $r_q \equiv q^{-1} \bmod n'$ and $r_c \equiv c^{-1} \bmod n'$. Since $a \geq 2$, we have $n' = 4n \mid n^2 = c + q$, and it follows that $r_c \equiv -r_q \bmod n'$. Similar to the above analysis, we detail the case of h_+. Define

$$L_{U,+} = (0_{E_1}, 0_{E_1}, [-r_q]Q', 0_{E_2}),\ L_{V,+} = ([r_c a_1]P, [r_c a_2]P, 0_{E_2}, 0_{E_2}),$$

Then we take the following basis

$$\begin{aligned}(T_1, T_2, T_3, T_4) &= (-\mathcal{J}U_+, \mathcal{J}V_+, V_+, U_+),\\ (S_1, S_2, S_3, S_4) &= (-\mathcal{J}L_{U,+}, \mathcal{J}L_{V,+}, L_{V,+}, L_{U,+}).\end{aligned} \tag{5}$$

It can be verified that $e_{n'}(S_i, S_j) = e_{n'}(T_i, T_j) = 1,\ e_{n'}(S_i, T_j) = \zeta^{\delta_{ij}},\ \zeta = e_{n'}(P, Q) \in \mu_{n'}$, where $e_{n'}(\cdot,\cdot)$ denotes the product Weil pairing on $A_0[n']$, ζ is a primitive n'-th root of unity, and $\delta_{ij} = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases}$.

Hence, $\mathcal{B} = (S_1, S_2, S_3, S_4, T_1, T_2, T_3, T_4)$ gives an n'-symplectic basis, and $H_+ = [4]\langle T_1, T_2, T_3, T_4\rangle$. The action of $\mathcal{J}$ on this basis is given by

$$[\mathcal{J}]_{\mathcal{B}} = \mathrm{diag}(C, C),\ C = \begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

Set $S_j^{(4)} = [n]S_j,\ T_j^{(4)} = [n]T_j,\ 1 \leq j \leq 4$, then these points form a symplectic basis of $A_0[4]$, with respect to which $\mathcal{J}$ still has matrix $\mathrm{diag}(C, C)$. Let $\theta_{\mathbf{u}}$ denote the associated level-2 theta sections, whose theta structure is determined by $\mathcal{B}$, indexed by $\mathbf{u} = (u_1, u_2, u_3, u_4) \in (\mathbb{Z}/2\mathbb{Z})^4,\ u_j \in \{0, 1\}$.

Their evaluations give the projective theta coordinates $\theta(Z) = \big(\theta_{\mathbf{u}}(Z)\big)_{\mathbf{u} \in (\mathbb{Z}/2\mathbb{Z})^4}$. Write $\mathbf{0} = (0,0,0,0)$, and let $\mathbf{e}_j$ be the j-th standard basis vector of $(\mathbb{Z}/2\mathbb{Z})^4$. For

$U \in A_0[4]$, let g_U denote the symmetric theta-group lift determined by U above translation by $[2]U$. We use the notation of [2, Corollary G.3], under which

$$g_{T_j^{(4)}}\theta_{\mathbf{u}} = (-1)^{u_j}\theta_{\mathbf{u}},\ g_{S_j^{(4)}}\theta_{\mathbf{u}} = \theta_{\mathbf{u}+\mathbf{e}_j}.$$

In particular, $\theta_{\mathbf{0}}$ spans the common invariant subspace of the $g_{T_j^{(4)}}$, and $\theta_{\mathbf{u}} = g_{S_1^{(4)}}^{u_1} g_{S_2^{(4)}}^{u_2} g_{S_3^{(4)}}^{u_3} g_{S_4^{(4)}}^{u_4}\theta_{\mathbf{0}}$.

The signs in C introduce no additional factors on these lifts. Indeed, [2, Remark G.5] gives $g_{U+W} = e_2([2]U, W)\, g_U,\ W \in A_0[2]$. Taking $W = [2]U$ and using the alternating property of the Weil pairing, we obtain $g_{-U} = e_2([2]U, [2]U)\, g_U = g_U$. Let J^* denote the pullback action of $\mathcal{J}$ on theta sections, defined up to a common nonzero scalar. Using the Functoriality and $\mathcal{J}^{-1} = -\mathcal{J}$ yields

$$J^* g_{S_j^{(4)}} (J^*)^{-1} = g_{S_{5-j}^{(4)}},\ J^* g_{T_j^{(4)}} (J^*)^{-1} = g_{T_{5-j}^{(4)}}.$$

Hence J^* preserves the line spanned by $\theta_{\mathbf{0}}$. Writing $J^*\theta_{\mathbf{0}} = \kappa\theta_{\mathbf{0}}$, with $\kappa \neq 0$, we find

$$J^*\theta_{(u_1,u_2,u_3,u_4)} = \kappa\, g_{S_4^{(4)}}^{u_1} g_{S_3^{(4)}}^{u_2} g_{S_2^{(4)}}^{u_3} g_{S_1^{(4)}}^{u_4}\theta_{\mathbf{0}} = \kappa\,\theta_{(u_4,u_3,u_2,u_1)},$$

where the last equality uses the commutativity of the action $g_{S_j^{(4)}}$. Define the index permutation $\rho(u_1, u_2, u_3, u_4) = (u_4, u_3, u_2, u_1)$. Since κ is independent of $\mathbf{u}$, the evaluation gives

$$\big(\theta_{\mathbf{u}}(\mathcal{J}Z)\big)_{\mathbf{u}\in(\mathbb{Z}/2\mathbb{Z})^4} = \big(\theta_{\rho(\mathbf{u})}(Z)\big)_{\mathbf{u}\in(\mathbb{Z}/2\mathbb{Z})^4} \quad \text{in } \mathbb{P}^{15}.$$

Under the little-endian identification $(u_1, u_2, u_3, u_4) \longleftrightarrow u_1 + 2u_2 + 4u_3 + 8u_4$, the induced permutation of $\{0, \ldots, 15\}$ is $(1\ 8)(2\ 4)(3\ 12)(5\ 10)(7\ 14)(11\ 13)$, with fixed indices $0, 6, 9, 15$. This agrees with [2, Corollary G.3, Eq. (22)]. For the corresponding theta-null point $\theta(0) = (a_0 : \cdots : a_{15})$, when $a_0 \neq 0$, the projective permutation identity implies that $a_1 = a_8,\ a_2 = a_4,\ a_3 = a_{12},\ a_5 = a_{10},\ a_7 = a_{14},\ a_{11} = a_{13}$. Hence, we can use [2, Algorithm 4] to accelerate the generic isogenies involved in computing h_+ and $\widehat{h_-}$, by updating only two kernel points. For simplicity, we only present the detailed procedure for the computation of h_+.

Denote $h_j = f_j \circ \cdots \circ f_1 \colon A_0 \to A_j,\ 1 \leq j \leq a$, with $h_0 = \mathrm{id}_{A_0}$, $h_a = h_+$ and $A_a = A$. At each step, we use the theta structure induced by the symplectic 4-basis

$$\mathcal{B}_j^{(4)} = \big([n]h_j(S_1), \ldots, [n]h_j(S_4), [2^{a-j}]h_j(T_1), \ldots, [2^{a-j}]h_j(T_4)\big)$$

of $A_j[4]$. These theta structures are compatible with the isogeny chain. Since $\mathcal{J}_j \circ h_j = h_j \circ \mathcal{J}$, the descended automorphism $\mathcal{J}_j$ by h_j acts on $\mathcal{B}_j^{(4)}$ by $\mathrm{diag}(C, C)$.

The initial isogenies are computed using the method of [10, Algorithm 13]. Assume that a_1 is odd and a_2 is even, and set $m = \min\{v_2(a_2), a\}$, $m' = \min\{m+1, a\}$, with $v_2(0) = +\infty$, and write $e = a - m'$. After regrouping the factors as $(E_1 \times E_2)^2$, the first m steps are diagonal four-dimensional isogenies obtained from a two-dimensional isogeny chain $\Phi_m \colon E_1 \times E_2 \to D_m$ with

$$\ker(\Phi_m) = \langle p_{13}([n'/2^m]T_4), p_{13}([n'/2^m]T_3)\rangle\,,$$

where $p_{13} : (X_1, X_2, Y_1, Y_2) \to (X_1, Y_1),\ X_i \in E_1,\ Y_i \in E_2,\ i = 1, 2$ denotes a projection. More precisely, the evaluation of this diagonal isogeny is executed as $h_m(X_1, X_2, Y_1, Y_2) = (\Phi_m(X_1, Y_1), \Phi_m(X_2, Y_2)),\ X_i \in E_1,\ Y_i \in E_2,\ i = 1, 2$.

If $m = a$, then h_m already gives h_+. Otherwise, the next step is the four-dimensional gluing isogeny $f_{m+1} : A_m = D_m^2 \to A_{m+1}$ with $\ker(f_{m+1}) = \langle [4]W_1, [4]W_2, [4]W_3, [4]W_4 \rangle$, where $W_i = [2^{a-m-1}]h_m(T_i)$ for $1 \le i \le 4$. We compute its codomain using [10, Algorithm 4 and Remark 21], including the required sums of eight-torsion points, and evaluate it using [10, Algorithm 5], including the required translated points. The changes of theta coordinates are computed as in [10, Theorem 12], using the basis $\mathcal{B}$ to obtain $\mathcal{B}^{(4)}_{m'}$. We denote the above computation by $\mathsf{ComputeGluing}(A_0, \mathcal{B}, a_1, a_2, q, a, m, R')$. It returns a representation $\mathcal{C}_{m'}$ for $\widehat{h_{m'}}$, including the intermediate results (such as the theta-null inverses for $0 \le j < m'$ where defined, the inverse of the change-of-basis matrix) needed for the dual gluing evaluations, as well as the theta coordinates $\mathbf{x}_{m'} = \theta^{A_{m'}}(h_{m'}(U_+))$, $\mathbf{y}_{m'} = \theta^{A_{m'}}(h_{m'}(V_+))$, $\mathbf{a}_{m'} = \theta^{A_{m'}}(0)$ and $\mathbf{z}_{m'} = \theta^{A_{m'}}(h_{m'}(R'))$.

The remaining generic steps are computed from the two representatives $\mathbf{x}_{m'},\ \mathbf{y}_{m'}$ using $\mathsf{WeilCodomain}$ from [2, Algorithm 4]. By the duplication formula (see Eq. (1)), the generic isogenies are evaluated by

$$\mathsf{Eval}(\mathbf{x}, \mathbf{d}) = \mathcal{H}(\mathbf{d} \star (\mathcal{H} \circ \mathcal{S}(\mathbf{x}))),$$

with the inverse dual theta null point $\mathbf{d}$ on the codomain as in [2, Appendix E.1]. Denote by $\mathsf{xDBL}(\mathbf{x}, s; \mathbf{a}_j)$ the algorithm of performing s successive doublings on the theta coordinates $\mathbf{x} = \theta^{A_j}(\widetilde{X})$ of $\widetilde{X} \in A_j$, using the theta-null point $\mathbf{a}_j$, for $0 \le j \le a$ using [10, Algorithm 8].

The strategy of the isogeny chain follows the traversal of [10, Algorithm 20], with each four-point basis replaced by its two representatives. At the i-th iteration, an entry of depth d represents the points $[2^d]h_j(U_+)$ and $[2^d]h_j(V_+)$. At depth $d = a - j - 1$, these give the eight-torsion points

$$T_3^{(8)} = [2^d]h_j(V_+),\ T_4^{(8)} = [2^d]h_j(U_+)$$

when $j = m' + i - 1,\ d = e - i = a - j - 1$. Thus the inputs to $\mathsf{WeilCodomain}$ are $T_3^{(8)}$ and $T_4^{(8)}$. Their images under the automorphism $\mathcal{J}_j$ yield the other two directions $T_1^{(8)} = -\mathcal{J}_j T_4^{(8)},\ T_2^{(8)} = \mathcal{J}_j T_3^{(8)}$. Algorithm 4 describes the improved procedure for computing and evaluating the half chain h_+.

Remark 1. For the original algorithm of computing h_+, it is unnecessary to store the intermediate results for evaluating $\widehat{h_+}$, since we finally need to evaluate $h_- = \widehat{\widehat{h_-}}$ and not $\widehat{h_+}$. Thanks to the results in Section 3.2, h_- can be represented by $\widehat{h_+}$ composed with a middle involution and a special automorphism, and thus the computations of the two half chains can be shared. Consequently, we store the representation of $\widehat{h_+}$ prepared for the whole four-dimensional representation evaluation.

Algorithm 4 $\mathsf{TwoPointIsogChain}(E_1, E_2, a, a_1, a_2, q, P, Q, P', Q', R')$

Input: Supersingular curves E_1, E_2; integers $a \geq 2$, a_1 odd, a_2 even, and q odd satisfying $a_1^2 + a_2^2 + q = 2^{2a}$; a basis (P, Q) of $E_1[2^{a+2}]$ and its images (P', Q') under a q-isogeny $\varphi : E_1 \to E_2$; a point $R' = (\widetilde{R}, 0_{E_1}, 0_{E_2}, 0_{E_2})$ need to be evaluated, with $\widetilde{R} \in E_1$.

Output: Theta coordinates $\mathbf{a} = \theta^A(0)$, $\mathbf{z} = \theta^A(h_+(R'))$, and $\mathcal{C}$ containing the intermediate results for evaluating $\widehat{h_+}$, or $\perp$ if the algorithm is failure.

$n \leftarrow 2^a$, $n' \leftarrow 4n$, $c \leftarrow a_1^2 + a_2^2$, $r_q \leftarrow q^{-1} \bmod n'$, $r_c \leftarrow c^{-1} \bmod n'$
$U_+ \leftarrow ([a_1]P, [a_2]P, P', 0_{E_2})$, $V_+ \leftarrow ([a_1]Q, [a_2]Q, Q', 0_{E_2})$
$L_{U,+} \leftarrow (0_{E_1}, 0_{E_1}, [-r_q]Q', 0_{E_2})$, $L_{V,+} \leftarrow ([r_c a_1]P, [r_c a_2]P, 0_{E_2}, 0_{E_2})$
$\mathcal{B} \leftarrow (-\mathcal{J}L_{U,+}, \mathcal{J}L_{V,+}, L_{V,+}, L_{U,+}, -\mathcal{J}U_+, \mathcal{J}V_+, V_+, U_+)$
$m \leftarrow \min\{v_2(a_2), a\}$, $m' \leftarrow \min\{m+1, a\}$, $e \leftarrow a - m'$
$(\mathcal{C}, \mathbf{a}_{m'}, \mathbf{x}_{m'}, \mathbf{y}_{m'}, \mathbf{z}_{m'}) \leftarrow \mathsf{ComputeGluing}(A_0, \mathcal{B}, a_1, a_2, q, a, m, R')$
if ComputeGluing returns $\perp$ **then return** $\perp$
if $e = 0$ **then return** $(\mathbf{a}_{m'}, \mathbf{z}_{m'}, \mathcal{C})$
Compute a strategy $\Sigma = (s_1, \ldots, s_{e-1})$ for a chain of length e with two kernel points updated
$\ell \leftarrow 1$, $\mathcal{L} \leftarrow [(\mathbf{x}_{m'}, \mathbf{y}_{m'}, 0)]$
for $i = 1, \ldots, e$ **do**
  $j \leftarrow m' + i - 1$; $(\mathbf{x}, \mathbf{y}, d) \leftarrow \mathsf{Top}(\mathcal{L})$
  if $\prod_{k=0}^{15} (\mathbf{a}_j)_k \cdot \big(\mathcal{H} \circ \mathcal{S}(\mathbf{a}_j)\big)_k = 0$ **then return** $\perp$
  while $d < e - i$ **do**
    $s \leftarrow s_\ell$, $\ell \leftarrow \ell + 1$, $d \leftarrow d + s$, $\mathbf{x} \leftarrow \mathsf{xDBL}(\mathbf{x}, s; \mathbf{a}_j)$, $\mathbf{y} \leftarrow \mathsf{xDBL}(\mathbf{y}, s; \mathbf{a}_j)$
    $\mathsf{Push}(\mathcal{L}, (\mathbf{x}, \mathbf{y}, d))$
  end while
  if $\prod_{k=0}^{15} \big(\mathcal{H} \circ \mathcal{S}(\mathbf{x})\big)_k \cdot \big(\mathcal{H} \circ \mathcal{S}(\mathbf{y})\big)_k = 0$ **then return** $\perp$
  $\mathbf{d}_{j+1} \leftarrow \mathsf{WeilCodomain}(\mathbf{y}, \mathbf{x})$ ▷ [2, Algorithm 4]
  if $\prod_{k=0}^{15} (\mathbf{d}_{j+1})_k = 0$ **then return** $\perp$
  $\mathbf{c}_{j+1} \leftarrow \mathbf{a}_j^{-1}$, $\mathcal{C} \leftarrow \mathcal{C}.\mathsf{append}(\mathbf{c}_{j+1})$
  $\mathbf{a}_{j+1} \leftarrow \mathsf{Eval}(\mathbf{a}_j, \mathbf{d}_{j+1})$, $\mathbf{z}_{j+1} \leftarrow \mathsf{Eval}(\mathbf{z}_j, \mathbf{d}_{j+1})$, $\mathsf{Pop}(\mathcal{L})$
  for all entries $(\mathbf{x}', \mathbf{y}', d')$ in $\mathcal{L}$ **do**
    Replace this entry by $\big(\mathsf{Eval}(\mathbf{x}', \mathbf{d}_{j+1}), \mathsf{Eval}(\mathbf{y}', \mathbf{d}_{j+1}), d'\big)$
  end for
end for
$\mathbf{a} \leftarrow \mathbf{a}_a$, $\mathbf{z} \leftarrow \mathbf{z}_a$
return $(\mathbf{a}, \mathbf{z}, \mathcal{C})$

3.2 Determining the Middle Involution via Automorphisms

In this subsection, we further optimize the computation of four-dimensional isogeny by sharing the computation of two half chains h_+ and $\widehat{h_-}$ motivated by [24, Appendix C.5] in the two-dimensional case. The core idea is to represent

h_- by $\widehat{h_+}$ together with a middle through a specific automorphism $\mathcal{R}$. In the following, we investigate how to determine this involution.

Consider an automorphism $\mathcal{R} \in \mathrm{End}(A_0)$:

$$\mathcal{R}(X_1, X_2, Y_1, Y_2) = (X_1, -X_2, -Y_1, Y_2).$$

Then $\mathcal{R}$ can be represented by the matrix $R = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$, and thus it satisfies $R^2 = I$ and $R^{\mathsf{T}} = R$. Besides, we can deduce that $\mathcal{R} \circ F \circ \mathcal{R} = \widetilde{F}$. Simultaneously, it holds that $\mathcal{R} \circ \mathcal{J} = -\mathcal{J} \circ \mathcal{R}$.

Based on the analysis in Section 3.1, we have $H_+ = \ker(F) \cap A_0[n]$ and $H_- = \ker(\widetilde{F}) \cap A_0[n]$. It follows from $\mathcal{R} \circ F \circ \mathcal{R} = \widetilde{F}$ that $\mathcal{R}(H_+) = H_-$. The split-and-dual approach gives the involution $\iota : A \to A'$ such that $F = h_- \circ \iota \circ h_+$. By $\mathcal{R}(H_+) = H_-$ it can be deduced that there exists an involution $r : A \to A'$ such that $r \circ h_+ = \widehat{h_-} \circ \mathcal{R}$. Hence, we have

$$h_- = \mathcal{R} \circ \widehat{h_+} \circ r^{-1},$$

which implies $F = \mathcal{R} \circ \widehat{h_+} \circ \eta \circ h_+$, where $\eta = r^{-1} \circ \iota$. Since $\widehat{h_+}$ can be evaluated according to the shared intermediate results derived from computing h_+, we can reduce a half-chain computation if we can determine the efficiently computable involution η. The following proposition illustrates the form of η.

Proposition 2. *With the notation above, let $F = \mathcal{R} \circ \widehat{h_+} \circ \eta \circ h_+$, where $\mathcal{R} \in End(A_0)$ is an automorphism $\mathcal{R} : \mathcal{R}(X_1, X_2, Y_1, Y_2) = (X_1, -X_2, -Y_1, Y_2)$. Assume that the theta structure on A_0 for the four-dimensional (n, n, n, n)-isogeny is determined by the 4-symplectic basis $\mathcal{B}^{(4)} = [n](S_1, S_2, S_3, S_4, T_1, T_2, T_3, T_4)$, where $S_i, T_i,\ i = 1, \cdots, 4$ are denoted in Eq. (5). Then the action of η on the theta coordinate of $h_+(R')$, is given as follows:*

$$(N_\eta(\theta^A(h_+(R'))))_I = \left(\mathcal{H}(\theta^A(h_+(R')))\right)_{M_2 \cdot I},$$

where $I \in (\mathbb{Z}/2\mathbb{Z})^4$, $M_2 = \begin{pmatrix} 0 & -r_q a_1 & -r_q a_2 & 0 \\ r_q a_1 & 0 & 0 & -r_q a_2 \\ r_q a_2 & 0 & 0 & r_q a_1 \\ 0 & r_q a_2 & -r_q a_1 & 0 \end{pmatrix} \pmod 2$, and $r_q \equiv q^{-1}$ mod n'.

Proof. Define $\Omega = \begin{pmatrix} 0 & I_4 \\ -I_4 & 0 \end{pmatrix}$, and let $S = (S_1, S_2, S_3, S_4),\ T = (T_1, T_2, T_3, T_4)$. Then, under the following basis

$$\begin{aligned}
P_1 &= (P, 0_{E_1}, 0_{E_2}, 0_{E_2}),\ Q_1 = (Q, 0_{E_1}, 0_{E_2}, 0_{E_2}), \\
P_2 &= (0_{E_1}, P, 0_{E_2}, 0_{E_2}),\ Q_2 = (0_{E_1}, Q, 0_{E_2}, 0_{E_2}), \\
P_3 &= (0_{E_1}, 0_{E_1}, P', 0_{E_2}),\ Q_3 = (0_{E_1}, 0_{E_1}, [r_q]Q', 0_{E_2}), \\
P_4 &= (0_{E_1}, 0_{E_1}, 0_{E_2}, P'),\ Q_4 = (0_{E_1}, 0_{E_1}, 0_{E_2}, [r_q]Q'),
\end{aligned}$$

each S_i can be represented as $S_i = \sum_{k=1}^{4}[x_k^{(i)}]P_k + \sum_{k=1}^{4}[y_k^{(i)}]Q_k$. The same argument applies to $T_i,\ i = 1, \cdots, 4$. Arranging the coordinates of the S_i (resp. T_i) with respect to this basis as column vectors, we represent S (resp. T) by an 8×4 matrix S (resp. T). By calculation, it can be verified that

$$\mathrm{T}^{\mathsf{T}}\Omega\mathrm{T} = \mathrm{S}^{\mathsf{T}}\Omega\mathrm{S} = 0,\ \mathrm{T}^{\mathsf{T}}\Omega\mathrm{S} = -I_4,\ \mathrm{S}^{\mathsf{T}}\Omega\mathrm{T} = I_4. \tag{6}$$

Define $T_- := F \cdot S = (F(S_1), F(S_2), F(S_3), F(S_4))$. Extend T_- to a symplectic basis $\mathcal{B}_- = (T_-, S_-)$. Similarly, we can also represent T_- and S_- as matrices T_- and S_- with respect to the above basis. It follows from $\widetilde{F}F = [n^2] = 0 \mid_{A_0[n']}$ and $\ker(F \mid_{A_0[n']}) = \langle T\rangle$ that T_- generates $\ker\left(\widetilde{F} \mid_{A_0[n']}\right)$, $\langle [4]T_-\rangle = H_-$ and $\widetilde{F}S_- \in \langle T\rangle$. Suppose that $\widetilde{F}S_- = TD$, where $D \in M_4(\mathbb{Z}/n'\mathbb{Z})$. By the adjointness of F and $\widetilde{F}$, their matrices satisfy $\Omega F^\dagger = F^{\mathsf{T}}\Omega$. Thus

$$\mathrm{S}^{\mathsf{T}}\Omega F^\dagger\mathrm{S}_- = \mathrm{S}^{\mathsf{T}}F^{\mathsf{T}}\Omega\mathrm{S}_- = (F\mathrm{S})^{\mathsf{T}}\Omega\mathrm{S}_- = \mathrm{T}_-^{\mathsf{T}}\Omega\mathrm{S}_- = -I_4$$

by Eq. (6). On the other hand, it yields that $\mathrm{S}^{\mathsf{T}}\Omega F^\dagger\mathrm{S}_- = \mathrm{S}^{\mathsf{T}}\Omega\mathrm{T}D = I_4 D$, which implies $D = -I_4$ and thus $\widetilde{F}S_- = -T$.

Denote by $\mathcal{B}_A = (h_+T, [n]h_+S)$ and $\mathcal{B}_{A'} = (\widehat{h_-}T_-, [n]\widehat{h_-}S_-)$ the 4-symplectic basis on A and A', respectively. From $\widehat{h_-} \circ F = [n]\iota \circ h_+$ and $FS = T_-$ it follows that $\iota([n]h_+S) = \widehat{h_-}T_-$. Besides, by $h_+ \circ \widetilde{F} = [n]\iota^{-1} \circ \widehat{h_-}$ and $\widetilde{F}S_- = -T$ we have $\iota(h_+T) = -[n]\widehat{h_-}S_-$. Hence, the action of ι from $\mathcal{B}_A$ to $\mathcal{B}_{A'}$ can be regarded as the change-of-basis matrix

$$[\iota] = \begin{pmatrix} 0 & I_4 \\ -I_4 & 0 \end{pmatrix}.$$

In the following, we determine the transition matrix associated with r^{-1} from $\mathcal{B}_{A'}$ to $\mathcal{B}_A$.

Based on the above analysis, we deduce that $\mathcal{R}T_- = \mathcal{R} \circ FS = \widetilde{F} \circ \mathcal{R}^{-1}S$. From $F\widetilde{F} = 0 \mid_{A_0[n']}$ it follows that $\mathcal{R}T_- \in \langle T\rangle$ and thus there is a matrix $M \in M_4(\mathbb{Z}/n'\mathbb{Z})$ such that $\mathcal{R}T_- = TM$, *i.e.*, $RF\mathrm{S} = \mathrm{T}M$. Multiplying both sides by $\mathrm{S}^{\mathsf{T}}\Omega$ yields $M = \mathrm{S}^{\mathsf{T}}\Omega RF\mathrm{S} \pmod{n'}$.

According to the choice for $S = (S_1, S_2, S_3, S_4)$, we obtain

$$M = \begin{pmatrix} 0 & -r_q a_1 & -r_q a_2 & 0 \\ r_q a_1 & 0 & 0 & -r_q a_2 \\ r_q a_2 & 0 & 0 & r_q a_1 \\ 0 & r_q a_2 & -r_q a_1 & 0 \end{pmatrix} \pmod{n'}.$$

Since both (T, S) and (T_-, S_-) are (symplectic) bases on A_0, it holds that

$$\mathcal{R}(T_-, S_-) = (T, S)\begin{pmatrix} A_1 & B_1 \\ C_1 & D_1 \end{pmatrix} = \left(TA_1 + SC_1, TB_1 + SD_1\right),$$

where $A_1, B_1, C_1, D_1 \in M_4(\mathbb{Z}/n'\mathbb{Z})$. By $\mathcal{R}T_- = TM$ we have $A_1 = M,\ C_1 = 0$. Besides, one can deduce that $(R\mathrm{T}_-)^{\mathsf{T}}\Omega(R\mathrm{S}_-) = \mathrm{T}_-^{\mathsf{T}}\Omega\mathrm{S}_- = -I_4$. Consequently,

by Eq. (6) we derive

$$\begin{aligned}(R\mathrm{T}_-)^{\mathsf{T}}\Omega(R\mathrm{S}_-) &= (\mathrm{T}M)^{\mathsf{T}}\Omega(\mathrm{T}B_1 + \mathrm{S}D_1)\\ &= M^{\mathsf{T}}(\mathrm{T}^{\mathsf{T}}\Omega\mathrm{T})B_1 + M^{\mathsf{T}}(\mathrm{T}^{\mathsf{T}}\Omega\mathrm{S})D_1\\ &= -M^{\mathsf{T}}D_1,\end{aligned}$$

which implies $D_1 = (M^{\mathsf{T}})^{-1}$. Additionally, by $r^{-1}\circ\widehat{h_-} = h_+\circ\mathcal{R}$ and $\mathcal{R}(T_-, S_-) = (T, S)\begin{pmatrix} M & B_1 \\ 0 & (M^{\mathsf{T}})^{-1}\end{pmatrix}$ it follows that

$$\begin{aligned}r^{-1}(\widehat{h_-}T_-, [n]\widehat{h_-}S_-) &= (h_+\circ\mathcal{R}T_-, [n]h_+\circ\mathcal{R}S_-)\\ &= \left(h_+TM, [n]h_+\left(TB_1 + S(M^{\mathsf{T}})^{-1}\right)\right)\\ &= (h_+T, [n]h_+S)\begin{pmatrix} M_4 & 0 \\ 0 & (M_4^{\mathsf{T}})^{-1}\end{pmatrix},\end{aligned}$$

where $M_4 \equiv M \bmod 4$. The last equality holds because $[n]h_+TB_1 = 0_A$. Therefore, we obtain the action of η on $\mathcal{B}_A$ as follows:

$$M_\eta = [r^{-1}][\iota] = \begin{pmatrix} M_4 & 0 \\ 0 & (M_4^{\mathsf{T}})^{-1}\end{pmatrix}\begin{pmatrix} 0 & I_4 \\ -I_4 & 0\end{pmatrix} = \begin{pmatrix} 0 & M_4 \\ -(M_4^{\mathsf{T}})^{-1} & 0\end{pmatrix}.$$

The explicit expression for M gives

$$M^{\mathsf{T}} = -M, \quad M^2 = -r_q^2(a_1^2 + a_2^2)I_4 \equiv r_q I_4 \pmod{n'}.$$

Since exactly one of a_1, a_2 is odd. Hence $q \equiv r_q \equiv -1 \pmod 4$, and therefore

$$M_4^2 = -I_4, \quad (M_4^{\mathsf{T}})^{-1} = M_4, \quad M_2^{\mathsf{T}} = M_2 = M_2^{-1}.$$

We now pass to theta coordinates. In the induced symmetric level-2 theta structure, the first and second halves of $\mathcal{B}_A$ correspond to the sign and translation subgroups, respectively. The theta transformation associated with $\begin{pmatrix} 0 & I_4 \\ -I_4 & 0\end{pmatrix}$ is the Hadamard transform $\mathcal{H}((z_J)_J)_I = \sum_{J\in(\mathbb{Z}/2\mathbb{Z})^4}(-1)^{I^{\mathsf{T}}J}z_J$. The block-diagonal factor $\begin{pmatrix} M_4 & 0 \\ 0 & (M_4^{\mathsf{T}})^{-1}\end{pmatrix}$ preserves the two isotropic summands of the 4-torsion marking. Its theta transformation is consequently the permutation $(z_I)_I \longmapsto (z_{M_2^{\mathsf{T}}I})_I$. Here the symmetric lifts are those determined by the prescribed 4-torsion marking, so no coordinate-dependent multiplier occurs.

Applying these two transformations in order and using $M_2^{\mathsf{T}} = M_2$, we obtain, after choosing a common scalar normalization for N_η,

$$\begin{aligned}\left(N_\eta\left(\theta^A(h_+(R'))\right)\right)_I &= \sum_{J\in(\mathbb{Z}/2\mathbb{Z})^4}(-1)^{(M_2^{\mathsf{T}}I)^{\mathsf{T}}J}\theta^A(h_+(R'))_J\\ &= \left(\mathcal{H}\left(\theta^A(h_+(R'))\right)\right)_{M_2 I}.\end{aligned}$$

This ends the proof of this proposition. □

Combining Algorithm 4 with Proposition 2, we evaluate $F = \mathcal{R} \circ \widehat{h_+} \circ \eta \circ h_+$ at R'. The intermediate results derived from $\mathsf{TwoPointIsogChain}$ are written as $\mathcal{C} = (\mathcal{C}_{m'}, \mathbf{c}_{m'+1}, \ldots, \mathbf{c}_a)$, $\mathbf{c}_j = \mathbf{a}_{j-1}^{-1} = \theta^{A_{j-1}}(0)^{-1}$, where $\mathcal{C}_{m'}$ contains the data for evaluating $\widehat{h_{m'}}$, including the inverse change-of-basis matrices and the results needed for recovering the elliptic coordinates. For each generic isogeny step $f_j : A_{j-1} \to A_j$, the dual $\widehat{f_j}$ is evaluated using [10, Section 4.4, Lemma 22]. We denote this evaluation by

$$\mathsf{EvalDual}(\mathbf{z}, \mathbf{c}_j) = \mathbf{c}_j \star (\mathcal{H} \circ \mathcal{S} \circ \mathcal{H})(\mathbf{z}).$$

For the initial steps, we denote by $\mathsf{EvalDualGluing}(\mathbf{z}, \mathcal{C}_{m'})$ the reverse evaluation and coordinate recovery of [10, Algorithm 14], applied to the stored shared data $\mathcal{C}_{m'}$. This procedure evaluates the dual four-dimensional gluing, when present, and the two dual two-dimensional chain, using the stored inverse change-of-basis matrices. The elliptic coordinates are then recovered using [10, Algorithms 16 and 17] and returned in the factor order of $E_1^2 \times E_2^2$. Since a_1 is odd and a_2 is even in Algorithm 4, we have

$$M_2(i_1, i_2, i_3, i_4)^\mathsf{T} = (i_2, i_1, i_4, i_3)^\mathsf{T}.$$

Thus evaluating the middle involution $\eta = r^{-1} \circ \iota$ requires one Hadamard transform followed by this coordinate permutation. The whole procedure of our efficient method for computing the four-dimensional representation and evaluating $\varphi(\widetilde{R})$ (the third component of $F(R')$), is presented in Algorithm 5.

4 Applications to SQIsignHD and FastPOKÉ-4D

This section gives the application of our approach to computing and evaluating four-dimensional representations on two common cryptographic schemes: SQIsignHD [12] and FastPOKÉ-4D [20], the four-dimensional versions of SQIsign and POKÉ, respectively. By utilizing the improved algorithm for four-dimensional representation computation, the verification of SQIsignHD and the decryption of FastPOKÉ-4D can be significantly accelerated, strengthening the competitiveness of these two cryptographic protocols compared with their corresponding two-dimensional variants. Furthermore, combining with the scaling of parameters due to the new endomorphism attack [32], we can develop an efficient variant of SQIsignHD with a significantly fast signing phase together with a highly optimized verification.

4.1 A Fast Heuristic Version of SQIsignHD

In this subsection, we present a fast version of SQIsignHD that exploits Algorithm 5. First, we introduce the corresponding parameters. Let $p = c\,2^f 3^{f'} - 1$ such that $f \approx \lambda$, $3^{f'} \approx 2^{2\lambda}$, where λ is the security parameter. Thus, we have $\log(p) \approx 3\lambda$, which ensures the λ-bit security under the new attack [32]. Denote by 2^e the response bound satisfying $\deg(\varphi_{\mathsf{rsp}}) < 2^e$, and $E_0/\mathbb{F}_{p^2} : y^2 = x^3 + x$ a supersingular curve.

Algorithm 5 FourDimRepresent$(E_1, E_2, a, a_1, a_2, q, P, Q, P', Q', R')$

Input: Supersingular curves E_1, E_2; integers $a \geq 2$, a_1 odd, a_2 even, and q odd satisfying $a_1^2 + a_2^2 + q = 2^{2a}$; a basis (P, Q) of $E_1[2^{a+2}]$ and its images (P', Q') under a q-isogeny $\varphi : E_1 \to E_2$; a point $R' = (\widetilde{R}, 0_{E_1}, 0_{E_2}, 0_{E_2})$ need to be evaluated, with $\widetilde{R} \in E_1$.

Output: The Montgomery $(x : z)$-coordinates of $\varphi(\widetilde{R})$, or $\perp$ if this algorithm is failure.

1: $(\mathbf{a}, \mathbf{z}, \mathcal{C}) \leftarrow$ TwoPointIsogChain$(E_1, E_2, a, a_1, a_2, q, P, Q, P', Q', R')$
▷ Using Algorithm 4
2: **if** TwoPointIsogChain returns $\perp$ **then return** $\perp$
3: $m \leftarrow \min\{v_2(a_2), a\}$, $m' \leftarrow \min\{m+1, a\}$
4: Represent $\mathcal{C}$ as $(\mathcal{C}_{m'}, \mathbf{c}_{m'+1}, \ldots, \mathbf{c}_a)$
5: $\mathbf{w} \leftarrow \mathcal{H}(\mathbf{z})$
6: **for all** $I \in (\mathbb{Z}/2\mathbb{Z})^4$ **do**
7: $\quad \mathbf{z}_I \leftarrow \mathbf{w}_{M_2 I}$
8: **end for**
9: **for** $j = a, a-1, \ldots, m'+1$ **do**
10: $\quad \mathbf{z} \leftarrow$ EvalDual$(\mathbf{z}, \mathbf{c}_j)$
11: **end for**
12: $\big((x_1 : z_1), \ldots, (x_4 : z_4)\big) \leftarrow$ EvalDualGluing$(\mathbf{z}, \mathcal{C}_{m'})$ ▷ [10, Algorithms 14, 16 and 17]
13: **if** EvalDualGluing returns $\perp$ **then return** $\perp$
14: **return** $(x_3 : z_3)$

Key generation. The key generation phase is the same as the original SQIsignHD. One uses FastDoublePath to sample two secret isogenies $\tau, \tau' \colon E_0 \to E_{\mathsf{pk}}$ such that $\deg(\tau) \mid 3^{f'}$ and $\deg(\tau') \mid 2^{2f}$. It is worth noting that we use only $3^{f'/2}$-torsion to construct the isogeny τ. To satisfy the heuristic assumption on the distribution of elliptic curves in [12], the bit-lengths of $\deg(\tau)$ and $\deg(\tau')$ should be approximately 2λ. Then the public and secret keys are $\mathsf{pk} = E_{\mathsf{pk}}$ and $\mathsf{sk} = (\tau, \tau', I_\tau, I_{\tau'})$, respectively.

Signing. The difference compared with the original SQIsignHD is that we need not execute FastCommit during the signing phase. The signer directly sample an isogeny $\psi \colon E_0 \to E_{\mathsf{com}}$ with $\deg(\psi) \mid 3^{f'}$ and computed its kernel ideal I_ψ. The hash of E_{com} and the message m determines the challenge isogeny $\varphi \colon E_{\mathsf{pk}} \to E_{\mathsf{chl}}$ of degree $3^{f'/2}$ as well as the corresponding ideal I_φ. The remaining steps are the same as the original SQIsignHD. Sampling an ideal I_σ equivalent to $I = \overline{I_\psi} \cdot I_\tau \cdot I_\varphi$ such that the associated response isogeny $\sigma \colon E_{\mathsf{com}} \to E_{\mathsf{chl}}$ has degree q coprime to 3, with $2^{2a} - q$ prime and congruent to 1 modulo 4. For the basis (P_1, P_2) of $E_{\mathsf{com}}[2^f]$, the signer computes $R_i = \sigma(P_i)$ and outputs the signature $\mathsf{sig} = (E_{\mathsf{com}}, q, R_1, R_2)$. The bit-length of ψ is set to around 2λ to resist the meet-in-the-middle (MITM) attack. Additionally, since $q \approx 2^{2a} \approx \sqrt{p} \approx 2^{3\lambda/2}$, it yields that $a \approx 3\lambda/4$ is strictly smaller than f.

Verification. Given E_{pk}, m, and $\mathsf{sig} = (E_{\mathsf{com}}, q, R_1, R_2)$, the verifier recomputes φ and E_{chl}, checking the conditions on q and that $R_1, R_2 \in E_{\mathsf{chl}}[2^f]$, and recovering (P_1, P_2) from E_{com}. Then he constructs a chain of four-dimensional $(2,2,2,2)$-isogenies on $E_{\mathsf{com}}^2 \times E_{\mathsf{chl}}^2$ calling Algorithm 5 (FourDimRepresent) with input

$$\left(E_{\mathsf{com}}, E_{\mathsf{chl}}, a, a_1, a_2, q, [2^{f-a-2}]P_1, [2^{f-a-2}]P_2, [2^{f-a-2}]R_1, [2^{f-a-2}]R_2, P_i'\right),$$

where $P_i' = (P_i, 0_{E_{\mathsf{com}}}, 0_{E_{\mathsf{chl}}}, 0_{E_{\mathsf{chl}}})$, $i = 1, 2$. It accepts when the chain represents an isogeny $\sigma' : E_{\mathsf{com}} \to E_{\mathsf{chl}}$ of degree q sending P_i to R_i. The detailed analysis for the verification is presented in Appendix A.

Based on the above analysis, the main fundamental building block within the signing phase is the computation of the 3-power isogeny ψ. Compared with the previous version of SQIsignHD, it is unnecessary for us to repeatedly construct and evaluate $2,3$-isogeny chains, and the procedure becomes significantly simpler. This leads to an extremely fast signing phase.

Additionally, using our method to compute the four-dimensional representation can fill the gap between special four-dimensional isogenies in SQIsignHD and the general two-dimensional isogenies in SQIsign2D variants, resulting in considerably more efficient verification. In Section 5, our experimental results show that the total running time for signing and verification in the optimized SQIsignHD is lower than that of the NIST candidate SQIsign-v3.0 [1]. As a trade-off, the public-key or signature size will be larger due to the bigger parameter p of around 3λ bits.

Remark 2. One can also construct another variant of SQIsignHD by sampling the ideals I_{sk} and I_{com} via IdealToIsogeny (using Qlapoti [7] as well as its variants Qlapoti+ [19], Qlapoty [15]), as mentioned in [11, Section 3]. Under this circumstance, the parameter $p = c2^f - 1$ is of the same form as SQIsign-v3.0. The verification phase can also be integrated with our improved method (Algorithm 5). This version can eliminate the heuristic assumption on the distribution of elliptic curves during the key generation and signing phases and strengthen the security. We also implement this variant, and the corresponding experimental results are presented in Section 5.

4.2 Application to FastPOKÉ-4D

In this subsection, we integrate FastPOKÉ-4D [20] with our optimized approach for computing the four-dimensional representation to accelerate the decryption phase. In the following, we present the explicit procedure for the decryption of FastPOKÉ-4D.

During the decryption, the receiver first recovers the action of the pushforward isogeny $\varphi' = [\psi]_*\varphi$ on a basis of $E_B[2^a]$ by multiplying $[\alpha_2^{-1}]$ on this basis, where α_2 is the mask scalar. The images of this basis as well as a_1, a_2 determine a four-dimensional representation of φ'. Then he can employ Algorithm 5 to compute and evaluate this representation at the ciphertext point X_B to recover $\varphi'(X_B)$ and obtain the shared secret $X_{AB} = [\delta_D]\varphi'(X_B)$. Algorithm 6

gives the procedures for the decryption of FastPOKÉ-4D combined with our new approach.

Algorithm 6 Improved Decryption of FastPOKÉ-4D

Input: The secret key $\mathsf{sk} = (q, a_1, a_2, \alpha_2, \delta_D)$ with $2^{2a-4} - q = a_1^2 + a_2^2$, and the ciphertext $\mathsf{ct} = ((E_B, P_B, Q_B, X_B), (E_{AB}, P_{AB}, Q_{AB}), \mathsf{ct}')$.

Output: A message m'.

1: $P'_{AB} \leftarrow [1/\alpha_2]P_{AB},\ Q'_{AB} \leftarrow [1/\alpha_2]Q_{AB}$.
2: $X'_B \leftarrow (X_B, 0_{E_B}, 0_{E_{AB}}, 0_{E_{AB}})$.
3: $X'_{AB} \leftarrow \mathsf{FourDimRepresent}(E_B, E_{AB}, a-2, a_1, a_2, q, P_B, Q_B, P'_{AB}, Q'_{AB}, X'_B)$
4: Compute $X_{AB} \leftarrow [\delta_D]X'_{AB}$.
5: Compute $m' = \mathsf{KDF}(X_{AB}) \oplus \mathsf{ct}'$.
6: **return** m'.

It is worth noting that the output of FourDimRepresent in Line 3 of Algorithm 6 is $(x_{X'_{AB}} : z_{X'_{AB}})$. Since we only use the $(x : z)$-coordinates in practice, the coordinates $(x_{X'_{AB}} : z_{X'_{AB}})$ are replaced by X'_{AB} for simplicity. By leveraging our method, we can significantly speed up the decryption phase. The implementation results in Section 5 show that optimized FastPOKÉ-4D decryption is nearly as efficient as the original POKÉ. This further strengthens FastPOKÉ-4D's competitiveness, as it offers faster key generation and encryption and resists MOXZ adaptive attacks [23] compared with POKÉ.

Remark 3. Our new method for efficiently computing four-dimensional representations also possesses the potential to accelerate other four-dimensional-based cryptographic schemes, such as the UPKE (Updatable public-key encryption) based on the four-dimensional version of FESTA [4] (Asiacrypt 2026).

5 Experimental Results

In this section, we present the experimental results for the SQIsignHD and FastPOKÉ-4D integrated with our optimized techniques for the four-dimensional representation computation. The efficiency comparison between the improved SQIsignHD/FastPOKÉ-4D and their original versions is also provided.

All benchmarks were run on an Intel Core i9-12900K processor with Hyper-Threading and Turbo Boost disabled. The implementations were compiled with GCC 13.3.0 using the Broadwell configuration in optimized release mode. All measurements were taken with the benchmark process pinned to a single CPU core.

5.1 The Implementation Results for the Optimized SQIsignHD

In this subsection, we provide the implementation results of two versions of SQIsignHD, including the fast heuristic SQIsignHD (denoted as F-SQIsignHD$_H$)

together with the SQIsignHD variant using IdealToIsogeny in the key generation and commitment (denoted as F-SQIsignHD in [11, Section 3]).

For F-SQIsignHD$_H$, we construct the parameters satisfying the conditions in Section 4.1 at NIST-I, -III, and -V security levels as follows:

$$\begin{aligned}
&\text{NIST-I}: \ p_{383} = 127 \cdot 2^{127} \cdot 3^{157} - 1, \\
&\text{NIST-III}: \ p_{575} = 13 \cdot 2^{190} \cdot 3^{240} - 1, \\
&\text{NIST-V}: \ p_{767} = 397 \cdot 2^{254} \cdot 3^{318} - 1.
\end{aligned}$$

The computation of the $2,3$-isogeny chain and FastDoublePath during the key generation and signing phases comes from the implementation of the original version of SQIsignHD [12].

As for F-SQIsignHD, we use the same parameters as SQIsign-v3.0 [1] at NIST-I, -III, and -V as:

$$\begin{aligned}
&\text{NIST-I}: \ p_{324} = 3 \cdot 2^{324} - 1, \\
&\text{NIST-III}: \ p_{500} = 27 \cdot 2^{500} - 1, \\
&\text{NIST-V}: \ p_{664} = 17 \cdot 2^{664} - 1.
\end{aligned}$$

For the key generation and commitment phases of F-SQIsignHD, we have integrated qlapoty [15] into IdealToIsogeny as SQIsign-v3.0 did. The finite field arithmetic is derived from the implementation of SQIsign-v3.0. The comparison for the C and Broadwell implementations of F-SQIsignHD$_H$, F-SQIsignHD, and SQIsign-v3.0 is presented in Table 2.

It follows from Table 2 that with the Broadwell backend, F-SQIsignHD and F-SQIsignHD$_H$ incorporating our method achieve signing speedups of 2.11×–2.44× and 5.29×–6.63×, respectively, over SQIsign-v3.0 across NIST-I, -III, and -V. Although their verification speeds are 0.38×–0.42× and 0.21×–0.25× that of the baseline, this gap has been significantly filled due to our optimization, and thus signing gains outweigh the additional verification cost. Consequently, the total time for one signature generation and one verification is reduced by 28.9%–39.9% and 39.9%–48.0%, respectively, corresponding to overall speedups of 1.41×–1.66× and 1.66×–1.92×.

5.2 The Experimental Results for FastPOKÉ-4D

In this subsection, we present the implementation results for FastPOKÉ-4D, which has been integrated with our optimized approach for the four-dimensional representation computation. The parameter of FastPOKÉ-4D at NIST-I security level is presented as follows:

Table 2. Efficiency comparison of signing and verification for SQIsign-v3.0, F-SQIsignHD, and F-SQIsignHD$_H$ at NIST-I, -III, and -V security levels. The latter two schemes incorporate our proposed method for computing four-dimensional representations. Cycle counts are arithmetic means over 1000 executions, expressed in Mcycles (1 Mcycle = 10^6 CPU cycles). Ratio denotes the SQIsign-v3.0 cycle count divided by that of the corresponding scheme under the same backend and security level.

Backend	NIST	Scheme	Signing		Verification	
			Mcycles	Ratio	Mcycles	Ratio
C	I	SQIsign-v3.0 [1]	124.57	×1.00	17.07	×1.00
C	I	F-SQIsignHD	60.73	×2.05	47.58	×0.36
C	I	F-SQIsignHD$_H$	27.67	×4.50	77.72	×0.22
C	III	SQIsign-v3.0 [1]	455.58	×1.00	50.63	×1.00
C	III	F-SQIsignHD	202.10	×2.25	130.49	×0.39
C	III	F-SQIsignHD$_H$	91.23	×4.99	265.57	×0.19
C	V	SQIsign-v3.0 [1]	806.37	×1.00	96.76	×1.00
C	V	F-SQIsignHD	334.47	×2.41	238.72	×0.41
C	V	F-SQIsignHD$_H$	188.93	×4.27	544.35	×0.18
Broadwell	I	SQIsign-v3.0 [1]	100.06	×1.00	12.27	×1.00
Broadwell	I	F-SQIsignHD	47.52	×2.11	32.32	×0.38
Broadwell	I	F-SQIsignHD$_H$	18.92	×5.29	48.62	×0.25
Broadwell	III	SQIsign-v3.0 [1]	358.06	×1.00	31.82	×1.00
Broadwell	III	F-SQIsignHD	155.26	×2.31	78.93	×0.40
Broadwell	III	F-SQIsignHD$_H$	53.97	×6.63	148.86	×0.21
Broadwell	V	SQIsign-v3.0 [1]	704.90	×1.00	78.30	×1.00
Broadwell	V	F-SQIsignHD	289.04	×2.44	185.68	×0.42
Broadwell	V	F-SQIsignHD$_H$	120.42	×5.85	337.06	×0.23

NIST-I

$$\begin{aligned} p &= 2^{126} \cdot 251 \cdot 337 \cdot 349837 \cdot D \cdot \\ &\quad 43176887214720337703096707961986794523309266337 - 1 \\ &= 2 \cdot 3^{158} \cdot 5 \cdot 11 \cdot 41953 \cdot 151273 \cdot 322633 \cdot 2210318753 \cdot \\ &\quad 14248171259089 + 1, \\ \log p &= 383, \\ D &= 158415637294761427561. \end{aligned}$$

Table 3 presents the running times of the original FastPOKÉ-4D, FastPOKÉ-4D that is integrated with our approach for computing the four-dimensional

representation, POKÉ, and PIKE. The same implementation framework and benchmarking procedure were used for all three schemes.

Table 3. Performance comparison for the NIST-I parameter sets. Cycle counts are arithmetic means over 10^4 executions, reported in millions of CPU cycles (MCycles).

Scheme	KeyGen	Encrypt	Decrypt
POKÉ	237.65	56.69	82.57
PIKE	265.03	50.23	45.23
FastPOKÉ-4D	86.62	30.02	316.45
FastPOKÉ-4D + Algorithm 6	85.90	29.65	83.55

At NIST-I, incorporating our algorithm into FastPOKÉ-4D reduces the decryption cost from 316.45 to 83.55 Mcycles, yielding a 3.79× speedup over the original implementation. Decryption is thus comparable to that of POKÉ, with only a 1.2% overhead relative to its 82.57 Mcycles. This improvement substantially strengthens the practical competitiveness of FastPOKÉ-4D, bringing its decryption performance close to that of POKÉ while retaining resistance to generalized MOXZ adaptive attacks [23,22].

6 Conclusion

We have presented an algorithm for the structured four-dimensional isogeny representations used in SQIsignHD and FastPOKÉ-4D. The algorithm combines two reductions in redundant computation. The action of a specific automorphism pairs the four kernel directions and, in compatible theta coordinates, allows the generic chain computation to propagate only two representatives. A second automorphism relates the two half-chain kernels and determines an explicit middle involution. The complete representation can therefore be evaluated by constructing one half-chain and reusing its intermediate data for the dual isogeny construction.

These techniques apply directly to SQIsignHD verification and FastPOKÉ-4D decryption. They also complement changes to the signing procedure: our heuristic SQIsignHD variant uses direct smooth-isogeny commitment generation, whereas the ideal-to-isogeny instantiation uses Qlapoty for key generation and commitment. The arithmetic optimization applies to both, independently of this choice. Our C implementations show that our proposed measure fills the remaining gap between these four-dimensional constructions and their two-dimensional counterparts, thereby strengthening the competitiveness of the four-dimensional-based cryptographic schemes.

References

1. Aardal, M.A., Adj, G., Aranha, D.F., Basso, A., Borin, G., Canales Martínez, I.A., Chávez-Saab, J., Corte-Real Santos, M., Dartois, P., De Feo, L., Duparc, M., Espitau, T., Eriksen, J.K., Fouotsa, T.B., Gazzoni Filho, D.L., Hess, B., Invernizzi, R., Kohel, D., Leroux, A., Longa, P., Maino, L., Meyer, M., Mula, M., Nakagawa, K., Onuki, H., Panny, L., Patranabis, S., Petit, C., Pope, G., Reijnders, K., Robert, D., Rodríguez-Henríquez, F., Schaeffler, S., Vercauteren, F., Wallet, A., Wesolowski, B., van Woerden, W.: SQIsign. Tech. rep., National Institute of Standards and Technology (2026), https://sqisign.org
2. Basso, A., Dartois, P., Duparc, M., Eriksen, J.K., Kunzweiler, S., Meyer, M., Pope, G., Reijnders, K., Robert, D., Rueger, R., Schaeffler, S.: MIKE: a fast and compact post-quantum NIKE. Cryptology ePrint Archive, Paper 2026/2377 (2026), https://eprint.iacr.org/2026/2377
3. Basso, A., Dartois, P., Feo, L.D., Leroux, A., Maino, L., Pope, G., Robert, D., Wesolowski, B.: SQIsign2D-West - The Fast, the Small, and the Safer. In: Chung, K., Sasaki, Y. (eds.) Advances in Cryptology - ASIACRYPT 2024 - 30th International Conference on the Theory and Application of Cryptology and Information Security, Kolkata, India, December 9-13, 2024, Proceedings, Part III. pp. 339–370. Lecture Notes in Computer Science, Springer (2024)
4. Basso, A., Fouotsa, T.B., Kouider, F., Kutas, P., Maino, L., Marco, L.: Updatable public-key encryption from FESTA. Cryptology ePrint Archive, Paper 2026/1014 (2026), https://eprint.iacr.org/2026/1014
5. Basso, A., Maino, L.: POKÉ: A Compact and Efficient PKE from Higher-Dimensional Isogenies. In: Fehr, S., Fouque, P. (eds.) Advances in Cryptology - EUROCRYPT 2025 - 44th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Madrid, Spain, May 4-8, 2025, Proceedings, Part II. pp. 94–123. Lecture Notes in Computer Science, Springer (2025)
6. Basso, A., Maino, L., Pope, G.: FESTA: Fast Encryption from Supersingular Torsion Attacks. In: Guo, J., Steinfeld, R. (eds.) Advances in Cryptology - ASIACRYPT 2023 - 29th International Conference on the Theory and Application of Cryptology and Information Security, Guangzhou, China, December 4-8, 2023, Proceedings, Part VII. pp. 98–126. Lecture Notes in Computer Science, Springer (2023)
7. Borin, G., Corte-Real Santos, M., Eriksen, J.K., Invernizzi, R., Mula, M., Schaeffler, S., Vercauteren, F.: Qlapoti: Simple and Efficient Translation of Quaternion Ideals to Isogenies. In: Hanaoka, G., Yang, B.Y. (eds.) Advances in Cryptology – ASIACRYPT 2025. pp. 174–205. Springer Nature Singapore, Singapore (2026)
8. Campagna, M., Costello, C., Hess, B., Jalali, A., Koziel, B., LaMacchia, B., Longa, P., Naehrig, M., Renes, J., Urbanik, D., et al.: Supersingular Isogeny Key Encapsulation (2019)
9. Castryck, W., Decru, T.: An Efficient Key Recovery Attack on SIDH. In: Hazay, C., Stam, M. (eds.) 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. pp. 423–447. Lecture Notes in Computer Science, Springer (2023)
10. Dartois, P.: Fast computation of 2-isogenies in dimension 4 and cryptographic applications. Journal of Algebra **683**, 449–514 (2025). https://doi.org/https://doi.org/10.1016/j.jalgebra.2025.06.033, https://www.sciencedirect.com/science/article/pii/S0021869325003771
11. Dartois, P.: Fast computation of higher dimensional isogenies for cryptographic applications. Theses, Université de Bordeaux (Jul 2025)
12. Dartois, P., Leroux, A., Robert, D., Wesolowski, B.: SQIsignHD: New Dimensions in Cryptography. In: Joye, M., Leander, G. (eds.) Advances in Cryptology - EUROCRYPT 2024 - 43rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zurich, Switzerland, May 26-30, 2024, Proceedings, Part I. Lecture Notes in Computer Science, vol. 14651, pp. 3–32. Springer (2024)
13. Dartois, P., Maino, L., Pope, G., Robert, D.: An Algorithmic Approach to (2, 2)-Isogenies in the Theta Model and Applications to Isogeny-Based Cryptography. In: Chung, K., Sasaki, Y. (eds.) Advances in Cryptology - ASIACRYPT 2024 - 30th International Conference on the Theory and Application of Cryptology and Information Security, Kolkata, India, December 9-13, 2024, Proceedings, Part III. Lecture Notes in Computer Science, vol. 15486, pp. 304–338. Springer (2024)
14. Deuring, M.: Die Typen der Multiplikatorenringe elliptischer Funktionenkörper. Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg **14**, 197–272 (1941)
15. Duparc, M., Leroux, A., Schaeffler, S.: Qlapoty: Improved analysis and efficiency for quaternionic ideal to isogeny transformation. Cryptology ePrint Archive, Paper 2026/1700 (2026), https://eprint.iacr.org/2026/1700
16. Feo, L.D., Kohel, D., Leroux, A., Petit, C., Wesolowski, B.: SQISign: Compact Post-quantum Signatures from Quaternions and Isogenies. In: Moriai, S., Wang, H. (eds.) Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12491, pp. 64–93. Springer (2020)
17. Jao, D., Feo, L.D.: Towards Quantum-Resistant Cryptosystems from Supersingular Elliptic Curve Isogenies. In: Yang, B. (ed.) Post-Quantum Cryptography - 4th International Workshop, PQCrypto 2011, Taipei, Taiwan, November 29 - December 2, 2011. Proceedings. pp. 19–34. Lecture Notes in Computer Science, Springer (2011)
18. Kani, E.: The Number of Curves of Genus Two with Elliptic Differentials. Journal für die reine und angewandte Mathematik **1997**(485), 93–122 (1997)
19. Lai, Y.F.: Qlapoti+ and more: Optimizing isogeny-based signatures. Cryptology ePrint Archive, Paper 2026/1640 (2026), https://eprint.iacr.org/2026/1640

20. Lin, J., Wang, S., Zheng, Y., Lin, K., Zhao, C.A.: FastPOKÉ-4D: Efficient and Secure Public-Key Encryption via Four-Dimensional Isogenies (2026), preprint
21. Maino, L., Martindale, C., Panny, L., Pope, G., Wesolowski, B.: A Direct Key Recovery Attack on SIDH. In: Hazay, C., Stam, M. (eds.) 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. pp. 448–471. Lecture Notes in Computer Science, Springer (2023)
22. Moriya, T.: Adaptive Attacks on FESTA Variants with Masked-Degree Isogenies. Cryptology ePrint Archive, Paper 2026/1240 (2026)
23. Moriya, T., Onuki, H., Xu, M., Zhou, G.: Adaptive Attacks Against FESTA Without Input Validation or Constant-Time Implementation. In: Saarinen, M.O., Smith-Tone, D. (eds.) Post-Quantum Cryptography - 15th International Workshop, PQCrypto 2024, Oxford, UK, June 12-14, 2024, Proceedings, Part II. pp. 3–19. Lecture Notes in Computer Science, Springer (2024)
24. Moriya, T., Stopar, M.: LIT-SiGamal: An Efficient Isogeny-Based PKE Based on a LIT Diagram. In: Hanaoka, G., Yang, B. (eds.) Advances in Cryptology - ASIACRYPT 2025 - 31st International Conference on the Theory and Application of Cryptology and Information Security, Melbourne, VIC, Australia, December 8-12, 2025, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 16248, pp. 275–306. Springer (2025)
25. Mumford, D.: On the Equations Defining Abelian Varieties. I. Invent. Math. **1**, 287–354 (1966)
26. Nakagawa, K., Onuki, H.: QFESTA: Efficient Algorithms and Parameters for FESTA Using Quaternion Algebras. In: Reyzin, L., Stebila, D. (eds.) Advances in Cryptology - CRYPTO 2024 - 44th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2024, Proceedings, Part V. pp. 75–106. Lecture Notes in Computer Science, Springer (2024)
27. Nakagawa, K., Onuki, H.: SQIsign2DPush: Faster Signature Scheme Using 2-Dimensional Isogenies. In: Daemen, J., Thomé, E. (eds.) Advances in Cryptology - EUROCRYPT 2026 - 45th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Rome, Italy, May 10-14, 2026, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 16544, pp. 511–540. Springer (2026)
28. Nakagawa, K., Onuki, H., Castryck, W., Chen, M., Invernizzi, R., Lorenzon, G., Vercauteren, F.: SQIsign2D-East: A New Signature Scheme Using 2-Dimensional Isogenies. In: Chung, K., Sasaki, Y. (eds.) Advances in Cryptology - ASIACRYPT 2024 - 30th International Conference on the Theory and Application of Cryptology and Information Security, Kolkata, India, December 9-13, 2024, Proceedings, Part III. pp. 272–303. Lecture Notes in Computer Science, Springer (2024)
29. Ray, D., Barreto, P.L.: Improving the efficiency of 4-dimensional SQIsign. Cryptology ePrint Archive, Paper 2026/2906 (2026), https://eprint.iacr.org/2026/2906

30. Robert, D.: Breaking SIDH in Polynomial Time. In: Hazay, C., Stam, M. (eds.) 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. pp. 472–503. Lecture Notes in Computer Science, Springer (2023)
31. Silverman, J.H.: The Arithmetic of Elliptic Curves, vol. 106. Springer (2009)
32. Wesolowski, B.: The Supersingular Isogeny Problem in Time and Memory $p^{1/3+o(1)}$. Cryptology ePrint Archive, Paper 2026/1486 (2026)
33. Xu, Z., Lin, K., Zhao, C., Ouyang, Y.: SQIsign2D^2: New SQIsign2D Variant by Leveraging Power Smooth Isogenies in Dimension One. In: Hanaoka, G., Yang, B. (eds.) Advances in Cryptology - ASIACRYPT 2025 - 31st International Conference on the Theory and Application of Cryptology and Information Security, Melbourne, VIC, Australia, December 8-12, 2025, Proceedings, Part IV. pp. 341–371. Lecture Notes in Computer Science, Springer (2025)

A Verification with a Single Half Chain for F-SQIsignHD

We use the notation of Section 3, with $E_1 = E_{\mathsf{com}}$ and $E_2 = E_{\mathsf{chl}}$ in Section 4.1. The purpose of this appendix is to establish the soundness of the verification procedure of our improved SQIsignHD for arbitrary received torsion data. We write $\sim$ for projective equality of theta coordinates.

A.1 Input Validation

In addition to the challenge and degree checks of Section 4.1, the verifier checks that the curve and point encodings are valid, that both curves are nonsingular, and that

$$a \geq 2, \qquad f \geq a+2, \qquad a_1^2 + a_2^2 + q = n^2, \qquad q > 0, \qquad \gcd(q, 2p) = 1, \tag{7}$$

where a_1 is odd and a_2 is even. It also checks

$$[2^f]P_i = 0_{E_1}, \qquad [2^f]R_i = 0_{E_2} \ (i = 1, 2), \tag{8}$$

$$\operatorname{ord}\big(e_{2^f}(P_1, P_2)\big) = 2^f, \qquad e_{2^f}(R_1, R_2) = e_{2^f}(P_1, P_2)^q. \tag{9}$$

The order condition in Eq. (9) is equivalent to $e_{2^f}(P_1, P_2)^{2^{f-1}} \neq 1$. Since q is odd, both pairs are bases. They determine an isomorphism

$$\sigma_{\mathrm{tor}} : E_1[2^f] \longrightarrow E_2[2^f], \qquad \sigma_{\mathrm{tor}}(P_i) = R_i.$$

The point encodings fix the relative signs of R_1 and R_2, and therefore determine $\sigma_{\mathrm{tor}}(P_1 + P_2)$.

The input to Algorithm 5 is obtained from

$$\begin{aligned} (P, Q) &= [2^{f-a-2}](P_1, P_2), \\ (P', Q') &= [2^{f-a-2}](R_1, R_2). \end{aligned} \tag{10}$$

Thus P, Q, P', Q' have order $n' = 4n$.

On n'-torsion, put

$$\Phi_{\mathrm{tor}} = \mathrm{Diag}(\sigma_{\mathrm{tor}}, \sigma_{\mathrm{tor}}), \qquad \widetilde{\Phi}_{\mathrm{tor}} = [q]\Phi_{\mathrm{tor}}^{-1}.$$

Let F_{tor} be the matrix in Theorem 1 with Φ and $\widetilde{\Phi}$ replaced by Φ_{tor} and $\widetilde{\Phi}_{\mathrm{tor}}$, respectively, and let $\widetilde{F}_{\mathrm{tor}}$ be its adjoint with respect to the product Weil pairing. Eq. (9) and block multiplication give

$$\widetilde{F}_{\mathrm{tor}} F_{\mathrm{tor}} = F_{\mathrm{tor}} \widetilde{F}_{\mathrm{tor}} = [n^2] = 0, \qquad \mathcal{R} F_{\mathrm{tor}} \mathcal{R} = \widetilde{F}_{\mathrm{tor}}. \tag{11}$$

Define S, T by Eq. (5), using the received points and the inverses r_q, r_c modulo n'.

Lemma 1. *Under the conditions above, S, T satisfy Eq. (6). The subgroups*

$$H_+ = [4]\langle T_1, T_2, T_3, T_4 \rangle, \qquad H_- = \mathcal{R}(H_+)$$

are maximal isotropic in $A_0[n]$. Their polarized quotients $h_+ : A_0 \to A$ and $\widehat{h_-} : A_0 \to A'$ admit a polarized isomorphism $r : A \to A'$ satisfying

$$r \circ h_+ = \widehat{h_-} \circ \mathcal{R}.$$

For the matrix M in Proposition 2, the tuples

$$T_- = F_{\mathrm{tor}} S = \mathcal{R} T M, \qquad S_- = \mathcal{R} S M^{-T} \tag{12}$$

satisfy $S_-^T \Omega T_- = I_4$ and $\widetilde{F}_{\mathrm{tor}} S_- = -T$.

Proof. The symplectic identities in Eq. (6) follow from $e_{n'}(P', Q') = e_{n'}(P, Q)^q$ and $a_1^2 + a_2^2 + q = n^2 \equiv 0 \pmod{n'}$. The graph description of H_+ in Section 3.1, with Φ replaced by Φ_{tor}, is injectively parametrized by $E_1[n]^2$. The product Weil pairing on this graph is the source pairing raised to the power $a_1^2 + a_2^2 + q = n^2$. Hence H_+ is isotropic and has order n^4, so it is maximal isotropic in $A_0[n]$. Since $\mathcal{R}$ preserves the product polarization, the same holds for H_-. The quotient theorem gives the polarized quotients, and the identity $\mathcal{R}(H_+) = H_-$ induces the isomorphism r.

The definitions give $F_{\mathrm{tor}} T = 0$ and $\mathcal{R} F_{\mathrm{tor}} S = TM$. The matrix M satisfies $M^T = -M$ and $M^2 = r_q I_4$, as in Proposition 2. It is therefore invertible, and Eq. (6) gives

$$S_-^T \Omega T_- = M^{-1} M = I_4.$$

The two tuples are isotropic by the same identities. Moreover,

$$\widetilde{F}_{\mathrm{tor}} S_- = \mathcal{R} F_{\mathrm{tor}} S M^{-T} = TMM^{-T} = -T.$$

It follows that $H_- = [4]\langle T_- \rangle$.

A.2 The Middle Isomorphism

Let $\mathbf{a}^A$ be the theta-null computed by Algorithm 4, and equip A' with the marking $([n]\widehat{h_-} S_-, \widehat{h_-} T_-)$. In the markings $\mathcal{B}_A, \mathcal{B}_{A'}$ of Proposition 2, Eq. (12)

gives the matrix $\mathrm{Diag}(M_4, M_4^{-T})$ for r^{-1}. Its action on level-2 theta coordinates is the permutation indexed by M_2. Since $M_2^T = M_2 = M_2^{-1}$, this permutation commutes with the Hadamard transform $\mathcal{H}$. Consequently, the theta-null points in the chain markings satisfy

$$\mathbf{a}^{A'} \sim (\mathbf{a}^{A}_{M_2 I})_I, \qquad \mathbf{a}^{A'} \sim \mathcal{H}\mathbf{a}^{A} \quad \Longleftrightarrow \quad N_\eta \mathbf{a}^{A} \sim \mathbf{a}^{A}. \tag{13}$$

Thus the middle comparison can be performed on the single quotient A.

We use the following level-2 theta-null identification assumption, in the form considered in [2, Remark H.1].

Assumption 2 (Level-2 theta-null identification) *Let X and Y be principally polarized abelian fourfolds over $\overline{\mathbb{F}}_p$, equipped with symmetric level-2 theta structures. If their theta-null points are projectively equal, then there exists an isomorphism of principally polarized abelian varieties $f : X \to Y$ that identifies the two theta structures.*

The assumption is applied to the theta structures induced by the markings $\mathcal{B}_A$ and $\mathcal{B}_{A'}$ above. By Eq. (13), acceptance of the middle comparison therefore gives a polarized isomorphism $\iota : A \to A'$ compatible with these markings. Combining ι with the isomorphism r of Lemma 1, we obtain the polarized automorphism

$$\eta = r^{-1} \circ \iota : A \longrightarrow A.$$

The calculation in Proposition 2 then gives

$$\theta^A \circ \eta = N_\eta \theta^A, \qquad (N_\eta z)_I = \big(\mathcal{H}(z)\big)_{M_2 I}. \tag{14}$$

In particular, the middle operation used by Algorithm 5 is induced by a polarized automorphism of A.

A.3 Soundness and Completeness

For an input passing the checks of Section A.1, the verifier constructs h_+ by Algorithm 4 and stores the data $\mathcal{C}$ for the dual evaluation. It checks the middle condition in Eq. (13) and evaluates

$$G = \mathcal{R} \circ \widehat{h_+} \circ \eta \circ h_+$$

at $(X, 0_{E_1}, 0_{E_2}, 0_{E_2})$ for $X = P_1, P_2, P_1 + P_2$. Write the four elliptic components as

$$G(X, 0_{E_1}, 0_{E_2}, 0_{E_2}) = (g_1(X), g_2(X), g_3(X), g_4(X)).$$

The verifier accepts if and only if all preceding checks succeed and, for each of the three points X,

$$\begin{aligned}
(x(g_1(X)) : z(g_1(X))) &= (x([a_1]X) : z([a_1]X)),\\
(x(g_2(X)) : z(g_2(X))) &= (x([a_2]X) : z([a_2]X)),\\
(x(g_3(X)) : z(g_3(X))) &= (x(\sigma_{\mathrm{tor}}(X)) : z(\sigma_{\mathrm{tor}}(X))),\\
(x(g_4(X)) : z(g_4(X))) &= (1 : 0).
\end{aligned} \tag{15}$$

Here $(x(X) : z(X))$ denotes the projective Montgomery coordinate, with $(x(0) : z(0)) = (1 : 0)$, and $\sigma_{\mathrm{tor}}(P_1 + P_2) = R_1 + R_2$. Projective comparisons are performed by cross multiplication; zero coordinate vectors and undefined evaluations are rejected.

Theorem 3 (Soundness). *Under Assumption 2, every accepted input determines a separable isogeny $\sigma : E_1 \to E_2$ of degree q such that*

$$\deg(\sigma) = q, \qquad \sigma|_{E_1[2^f]} = \sigma_{\mathrm{tor}}. \tag{16}$$

If $E_1, E_2, P_1, P_2, R_1, R_2$ are defined over $\mathbb{F}_{p^2}$, then σ is defined over $\mathbb{F}_{p^2}$.

Proof. By Lemma 1 and Assumption 2, the middle map η is a polarized automorphism. Hence G is a homomorphism of principally polarized abelian varieties and

$$h_+^\dagger h_+ = [n]_{A_0}, \qquad h_+ \widehat{h_+} = [n]_A.$$

Since $\mathcal{R}$ and η preserve the polarizations, we obtain

$$\begin{aligned} G^\dagger G &= h_+^\dagger \eta^\dagger \widehat{h_+}^\dagger \mathcal{R}^\dagger \mathcal{R} \widehat{h_+} \eta h_+ \\ &= h_+^\dagger \eta^\dagger [n]_A \eta h_+ = [n^2]_{A_0}. \end{aligned} \tag{17}$$

Write

$$G(X, 0_{E_1}, 0_{E_2}, 0_{E_2}) = (g_1(X), g_2(X), g_3(X), g_4(X)),$$

where $g_1, g_2 \in \mathrm{End}(E_1)$ and $g_3, g_4 \in \mathrm{Hom}(E_1, E_2)$. The first diagonal entry of Eq. (17) gives

$$\deg(g_1) + \deg(g_2) + \deg(g_3) + \deg(g_4) = n^2, \tag{18}$$

where the degree of the zero homomorphism is taken to be zero.

We first consider the signs in the projective comparisons. Suppose that u, v are homomorphisms from E_1 to an elliptic curve and that $v(P_1), v(P_2)$ are independent points of the same order d. If $u(X) = \pm v(X)$ for $X = P_1, P_2, P_1 + P_2$, additivity gives

$$(\epsilon_1 - \epsilon_{12})v(P_1) + (\epsilon_2 - \epsilon_{12})v(P_2) = 0$$

for some $\epsilon_1, \epsilon_2, \epsilon_{12} \in \{\pm 1\}$. If $d > 2$, independence implies that all three signs coincide. If $d \le 2$, negation acts trivially on the image of v. In either case, there is a single sign ϵ such that $u = \epsilon v$ on $E_1[2^f]$.

Applying this observation to the first two rows of Eq. (15), we find signs $\epsilon_1, \epsilon_2 \in \{\pm 1\}$ such that $g_i - [\epsilon_i a_i]$ vanishes on $E_1[2^f]$ for $i = 1, 2$. The degree form on elliptic-curve homomorphisms satisfies

$$\deg\big(g_i - [\epsilon_i a_i]\big) \le \big(\sqrt{\deg(g_i)} + |a_i|\big)^2 < 4n^2 < 2^{2f}. \tag{19}$$

A nonzero elliptic-curve homomorphism that kills $E_1[2^f]$ has degree at least 2^{2f}. It follows that

$$g_1 = [\epsilon_1 a_1], \qquad g_2 = [\epsilon_2 a_2].$$

This also covers $a_2 = 0$. The last row of Eq. (15) shows that g_4 vanishes on P_1 and P_2. Since $\deg(g_4) \le n^2 < 2^{2f}$, we have $g_4 = 0$.

Substituting these identities into Eq. (18) gives

$$\deg(g_3) = n^2 - a_1^2 - a_2^2 = q.$$

The third row of Eq. (15) and the same sign argument imply $g_3|_{E_1[2^f]} = \epsilon\sigma_{\mathrm{tor}}$ for a single $\epsilon \in \{\pm 1\}$. Taking $\sigma = \epsilon g_3$ proves Eq. (16). Since $p \nmid q$, the isogeny σ is separable.

Finally, suppose that the input is defined over $\mathbb{F}_{p^2}$. Then the Frobenius difference $\sigma^{(p^2)} - \sigma$ kills $E_1[2^f]$ and has degree at most $4q < 2^{2f}$. It must therefore be zero, which proves that σ is defined over $\mathbb{F}_{p^2}$.

Proposition 3 (Completeness). *Suppose that the input satisfies the conditions of Section A.1 and that a separable isogeny $\sigma : E_1 \to E_2$ of degree q induces σ_{tor}. If the isogeny-chain and evaluation algorithms are defined on the resulting data, then the verifier accepts.*

Proof. For an honest input, Theorem 1 gives the polarized isogeny F and its split-and-dual decomposition $F = h_- \circ \iota \circ h_+$. By Lemma 1, the isomorphism $r : A \to A'$ satisfies $r \circ h_+ = \widehat{h_-} \circ \mathcal{R}$. Consequently,

$$F = \mathcal{R} \circ \widehat{h_+} \circ (r^{-1} \circ \iota) \circ h_+.$$

The middle isomorphism $r^{-1} \circ \iota$ is polarized and, by Proposition 2, its action in the prescribed theta coordinates is N_η. Thus the middle comparison in Eq. (13) holds. Moreover,

$$F(X, 0_{E_1}, 0_{E_2}, 0_{E_2}) = ([a_1]X, [-a_2]X, -\sigma(X), 0_{E_2}),$$

so all conditions in Eq. (15) hold. The verifier therefore accepts whenever the required computations are defined.