Improving the Efficiency of 4-Dimensional SQIsign

Dustin Ray[1] and Paulo L. Barreto[2]

[1] Anchorage Digital, Security Engineering, USA
[2] University of Washington Tacoma, School of Engineering and Technology, Tacoma, USA

Abstract. SQIsignHD verifies a dimension-4 isogeny where SQIsign verifies a dimension-2 one, which removes the auxiliary curve from the signature but makes verification much costlier. We show that the verifier's chain of dimension-4 isogenies respects an automorphism of order 4 that acts on its theta coordinates as one fixed permutation, for every input, honest or forged. After the gluing, two of the four kernel points can therefore be recovered from the other two by permuting coordinates, and the constants of each step are determined by ten coordinates instead of sixteen, which cuts the verifier's cost by 37 to 40 %. Along the way we derive the change of basis that the verifier applies at the gluing, which Dartois states with only a brief justification, prove that the verifier's image check remains sound with the signs that its implementation leaves free, and show that the verifier must validate the basis hints, which a forger controls. We implement the scheme at the round-3 SQIsign primes, adopted after Wesolowski's attack. At level I a signature takes 139 bytes, against 200 for the round-3 specification, and key and signature together take 223 bytes, the smallest level-I combination of any post-quantum signature we are aware of at parameters that stand after that attack; verification costs 4.3 times that of the dimension-2 scheme in the same code.

Keywords: SQIsign · SQIsignHD · isogeny-based cryptography · post-quantum signatures · signature size

1 Introduction

SQIsign is a signature scheme whose public key is an elliptic curve and whose signature describes a path of curves. Its first version had the smallest signatures of any post-quantum proposal and verified fast, but it signed in seconds, because its signer relied on the KLPT algorithm to find paths in the quaternion algebra of the curve [FKL+20].

The situation changed in 2022, when three groups broke the SIDH key exchange by recovering an isogeny between elliptic curves from its action on a few points, which they embedded, by Kani's lemma [Kan97], in an isogeny between higher-dimensional abelian varieties [CD23, MMP+23, Rob23]. The attack soon became a tool for SQIsign: the signer no longer needs a path of smooth degree, since any isogeny of known degree can be handed to the verifier through its action on points and rebuilt in higher dimension. SQIsignHD was the first to do so, in dimension 4 [DLRW24]. Its signer is fast and its signature small, while the cost moves to the verifier, which computes a chain of dimension-4 isogenies; its authors released a C signer and a SageMath verifier [DLRW25]. SQIsign2D-West then brought the response back to dimension 2 [BDF+24]: the signer computes one more isogeny, the auxiliary one, so that the verifier's chain runs on a product of two elliptic curves. Verification became fast again at the cost of sending the auxiliary curve, and this design has been SQIsign's submission to NIST since its second round [AAA+25, AAA+26].

E-mail: dustin.ray@anchorlabs.com (Dustin Ray), pbarreto@uw.edu (Paulo L. Barreto)

The parameters moved once more in 2026, when Wesolowski gave an algorithm for the supersingular isogeny problem in time and memory $p^{1/3+o(1)}$ [Wes26]. The round-3 submission of September 2026 answered by replacing the round-2 primes with larger ones, $3 \cdot 2^{324} - 1$, $27 \cdot 2^{500} - 1$ and $17 \cdot 2^{664} - 1$ at levels I, III and V [AAA+26]. The larger primes raised the level-I signature from 148 to 200 bytes [AAA+25, AAA+26]. Some carriers have a fixed byte budget for the signature and the signed data together: a comment on the reference implementation's issue tracker reports that barcodes on physical credentials, such as driver's licenses, can be limited to 250 to 300 bytes for the signature and all the other information, and that verification overhead in the millisecond range is acceptable for these uses [SQI26a].

Our contributions. We revisit SQIsignHD's dimension-4 response at these primes. Our research contributions concern the verifier's chain of dimension-4 isogenies.

- *Two-point verification.* After the gluing, the verifier can carry only two of the four kernel points through each half-chain, doubling and pushing them, and recover the other two by one fixed permutation of their theta coordinates. This comes from a symmetry of the chain: an automorphism of order 4 that the verifier's isogeny respects and that acts as the same permutation of the sixteen theta coordinates on every fourfold from the gluing on, for every input, honest or forged (Theorems 1 and 2). The same symmetry fixes the null points along the chain (Proposition 5), so the constants of each step are determined by ten coordinates instead of sixteen. Together these turn the symmetry into a concrete speedup: the two-point verifier takes 0.63, 0.62 and 0.60 times the cycles of the four-point one at levels I, III and V (Section 5 and Appendix A).

- *The change of basis at the gluing.* We prove that the change of basis the verifier applies before the dimension-4 gluing, given by closed formulas in [Dar24], expresses the structure that the rest of the chain needs (Section 6 and Proposition 6).

- *Soundness of the verifier.* We prove that SQIsignHD's image check stays sound with the signs that its library leaves free (Proposition 1), and show that the verifier must check the basis hints, which a forger controls: without that check, anyone can re-encode a valid signature, and crafted hints drive the chain outside the hypotheses of our proofs (Section 7.2). We also estimate how often a commitment lands in a class with no response of good degree: about once in 2^{44} at level I, if response lattices behave like random lattices (Section 7.1).

This paper builds on an earlier report on the sizes and costs of the compact format at these parameters [Ray26a]. New here are the two-point verification theorem and its proof, the change of basis at the gluing, the validation of the basis hints, and levels III and V.

These results are supported by engineering work that makes the dimension-4 response usable at the round-3 primes. The rule of SQIsignHD's library for the bound 2^e on the response degree is written in terms of the security parameter and admits no response at these primes; we restate it in terms of $\sqrt{p}$ and derive e and the torsion exponent r at all three levels (Section 7.1). We make explicit which entry of the transmitted matrix is dropped and how the verifier recovers it, and give a bit-packed layout, with which a level-I signature takes 139 bytes against 200 for the specification's format (Section 7.2). Finally, we implemented the scheme and both verifiers at all three levels in sqisign-rs, our Rust implementation of SQIsign, whose dimension-2 signer and verifier reproduce the reference's known-answer vectors byte for byte [Ray26b]; the SQIsignHD authors' verifier, re-parameterized to our level-I prime, accepts our signatures.

The remainder of this paper is organized as follows. In Section 2 we recall the theta-model formulas the verifier uses and fix the terminology, and in Section 3 we review

related work. Section 4 describes SQIsignHD's dimension-4 response at the round-3 primes, proves the soundness of its image check, and identifies the two issues that our main results address. Section 5 presents two-point verification, and Section 6 the change of basis at the gluing. Section 7 gives the parameters and the wire format of our implementation, and Section 8 reports our experimental results. We conclude in Section 9. Appendix A contains the proofs, Appendix B relates our implementation to the algorithm they concern, and Appendix C discusses further savings and an open question.

2 Preliminaries

2.1 Level-2 theta coordinates and 2-isogenies

The verifier computes chains of 2-isogenies between abelian varieties in the level-2 theta model, with the formulas of Dartois [Dar24], which we recall. Let A be a principally polarized abelian variety of dimension g over $\mathbb{F}_{p^2}$, with a level-2 theta structure. Its theta coordinates $\theta(x) = (\theta_i(x))_{0 \le i < 2^g}$ map the Kummer variety $A/\pm 1$ to $\mathbb{P}^{2^g-1}$; the index i stands for the element of $(\mathbb{Z}/2\mathbb{Z})^g$ given by its bits, and $\theta(0)$ is the theta null point. For vectors x, y of 2^g coordinates, write $x \star y$ for the coordinatewise product, x^{-1} for the coordinatewise inverse, $S(x) = x \star x$, and H for the Hadamard transform,

$$(Hx)_j = \sum_{i=0}^{2^g-1} (-1)^{i \cdot j} x_i,$$

where $i \cdot j$ is the dot product of the bit vectors; $H^2 = 2^g$. All the formulas below are projective: they hold up to a scalar.

Let $f \colon A \to B$ be an isogeny with kernel $K \cong (\mathbb{Z}/2\mathbb{Z})^g$, isotropic for the Weil pairing on $A[2]$, and give B the theta structure that f induces [Dar24, Section 4]. Let $U = H(\theta_B(0))$ be the dual theta null point of B. Images are then computed as

$$\theta_B(f(x)) = H\big(U^{-1} \star H(S(\theta_A(x)))\big), \tag{1}$$

when U has no zero coordinate [Dar24, Algorithm 1]. To find U, let $T_0, \ldots, T_{g-1} \in A[8]$ be points such that $4T_0, \ldots, 4T_{g-1}$ generate K, with T_l attached to bit l by the theta structure. Then, for all indices j and all l,

$$U_{j \oplus 2^l} \cdot H(S(\theta_A(T_l)))_j = U_j \cdot H(S(\theta_A(T_l)))_{j \oplus 2^l}, \tag{2}$$

which determines U, and with it $\theta_B(0) = H(U)$, along a spanning tree of the hypercube of indices when the coordinates involved do not vanish [Dar24, Algorithms 2–4 and Remark 17]. Finally, doubling on A is

$$\theta_A(2x) = \theta_A(0)^{-1} \star H\big(W^{-1} \star S(H(S(\theta_A(x))))\big), \qquad W = H(S(\theta_A(0))) \tag{3}$$

[Dar24, Algorithm 8]. A chain of n 2-isogenies is computed from points of order 2^{n+2} above its kernel: each step takes the 8-torsion points above the kernel of the next 2-isogeny by doubling, computes the codomain by (2), and pushes the remaining points by (1); a strategy decides which multiples to keep, as for isogenies of elliptic curves [DLRW24]. In dimension $g = 4$, every S and $\star$ costs 16 squarings or multiplications in $\mathbb{F}_{p^2}$, and every H costs 64 additions. Steps whose kernel meets a product of lower dimension, such as the gluing of Section 6, need modified formulas [Dar24, Section 4.3].

2.2 Terms

auxiliary isogeny

In dimension 2, a second isogeny the signer computes next to the response so that

together they define a $(2^{e_{\mathrm{rsp}}}, 2^{e_{\mathrm{rsp}}})$-isogeny the verifier can compute. Its target, the auxiliary curve, is sent in the signature. Dimension 4 has none.

challenge
: The isogeny from the public-key curve selected by hashing the message and the commitment; the signer must connect its target to the commitment.

commitment
: A fresh random curve the signer publishes first, so that the challenge cannot be chosen after the response.

degree
: The size of an isogeny's kernel, the number of points that map to zero. An isogeny of degree 2^e is a path of e steps of degree 2.

dimension (of the embedding)
: The number of elliptic curves whose product carries the verifier's chain: two in SQIsign2D-West and the round-3 submission, four in SQIsignHD and here. Each step in dimension g acts on 2^g coordinates.

endomorphism ring
: All isogenies from a curve to itself. It is the secret: the security of every scheme here rests on its being hard to compute for a random supersingular curve.

isogeny
: A map between elliptic curves that respects their group law. It is determined by its kernel; "walking" from one curve to another means composing isogenies of small degree.

Kani embedding
: Kani's lemma: an isogeny of degree q between elliptic curves, together with one of degree $N - q$, sits inside an N-isogeny between products of curves, of degree N^g in dimension g. In dimension 4 the second isogeny is replaced by an endomorphism of the same curves built from a sum of two squares, $N - q = a_1^2 + a_2^2$, so nothing extra is computed or sent.

KAT
: Known-answer test: the reference's published keys and signatures for fixed seeds, which an implementation must reproduce byte for byte.

megacycle
: One million processor cycles, read with `rdtsc`; the unit of every cost in this paper, as in the reference's own benchmark.

response
: The isogeny from the commitment curve to the challenge curve that the signer, and only the signer, can compute. In dimension 4 it is described by its degree q and its action on a few points.

supersingular curve
: An elliptic curve over $\mathbb{F}_{p^2}$ whose endomorphism ring is a maximal order in a quaternion algebra. All curves in this paper are supersingular.

theta model
: The coordinate system the verifier's chain runs in. A point of a dimension-g variety has 2^g theta coordinates: four in dimension 2, sixteen in dimension 4.

torsion
: The points of a curve whose order divides a given number. The 2^r-torsion is where the response's action is written down; 2^r divides $p+1$ on these primes, so those points are defined over $\mathbb{F}_{p^2}$.

3 Related work

SQIsignHD Our starting point is the construction of Dartois, Leroux, Robert and Wesolowski [DLRW24]: the response has degree $q < 2^e$ with $2^e - q$ a prime congruent to 1 modulo 4, and the verifier rebuilds it as a $(2^e, 2^e, 2^e, 2^e)$-isogeny on $E_{\mathrm{com}}^2 \times E_{\mathrm{chl}}^2$, computed as two half-chains that meet in the middle (their Section 4). Their library consists of a C signer and a SageMath verifier built on the Theta dim4 package [DLRW25, Dar25]; its parameters, $e = \lambda + \lceil \log_2 2\lambda \rceil$ and $r = \lceil e/2 \rceil + 2$, were set for primes of about 2λ bits, the size of the SQIsign primes before Wesolowski's attack [Wes26] (Section 7.1). The dimension-4 arithmetic, and the step formulas our verifier uses, are Dartois's [Dar24].

SQIsign The scheme we compare with is the round-3 submission, a dimension-2 scheme in the SQIsign2D-West design [BDF+24, AAA+26], whose reference code, the-sqisign [SQI26b], has a portable and an assembly-backed field arithmetic; its specification fixes the parameters of our dimension-2 baseline.

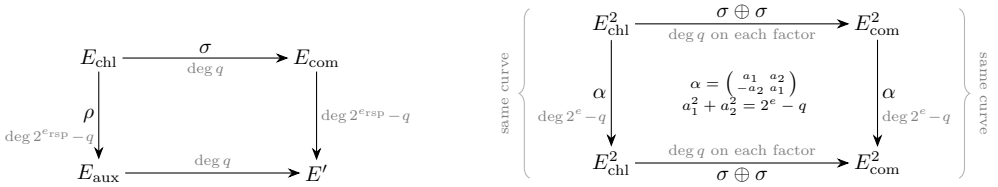
MIKE The closest relative of our verifier is MIKE [BDD+26], a non-interactive key exchange built on Robert's module action [Rob24], the other recent system whose main cost is a chain of dimension-4 2-isogenies in the theta model. Its fourfolds carry an automorphism $[i]$ of order 4 that the chain respects, which MIKE uses to compute each step from two points above the kernel instead of four [BDD+26, Appendix E.2]; our two-point verifier adapts this idea. What that idea does not give for the verifier, and what we prove, is that the verifier's chain has such an automorphism (Propositions 2 and 3), that it acts on the theta coordinates in which the verifier computes as one fixed permutation (Theorem 2), and that this holds for every input, honest or forged (Proposition 4 and Theorem 2); the verifier also checks the symmetry at the gluing and falls back to four points otherwise (Algorithm 2). MIKE's chain is moreover defined over $\mathbb{F}_p$ rather than $\mathbb{F}_{p^2}$, its domain being a Weil restriction, and at level I it runs 228 steps in 16.0 megacycles of constant-time C [BDD+26, Table 1]. MIKE also argues that the round-3 primes are conservative and proposes $\log_2 p = 317$, 484 and 651 instead of 325, 504 and 668 [BDD+26, Section 4.5]; we keep the specification's primes, so that our figures compare directly with the round-3 submission.

4 SQIsignHD at the round-3 primes

This section describes what SQIsignHD's dimension-4 signature sends (Section 4.1) and examines its verifier (Section 4.2), whose image check we prove sound, and which raises the two issues that Sections 5 and 6 address: the cost of its chain, and the justification of the structure in which the chain is computed after the gluing. The parameters at the round-3 primes and the wire format, on which our implementation and measurements rest, are given in Section 7.

4.1 What is sent

In dimension 2 the verifier cannot run the response σ of degree q alone: Kani's embedding needs a second isogeny of degree $2^{e_{\mathrm{rsp}}} - q$ from the challenge curve, where e_{rsp} is the



(a) dimension 2: ρ is an isogeny to a new curve; the signer computes it and E_{aux} is sent

(b) dimension 4: the auxiliary α is a pair of integers computed by the verifier from q; there is no auxiliary curve, so none is sent

Figure 1: The two ways a verifier rebuilds the response; both are Kani squares. (a) In dimension 2 the response σ of degree q is completed by an auxiliary isogeny ρ of degree $2^{e_{\mathrm{rsp}}} - q$ to a new curve; the signer computes ρ and the signature carries E_{aux}. (b) In dimension 4 the same square sits on the doubled curves and the complement is the endomorphism $\alpha = \left(\begin{smallmatrix} a_1 & a_2 \\ -a_2 & a_1 \end{smallmatrix}\right)$ with $a_1^2 + a_2^2 = 2^e - q$, which the verifier computes from q; it acts on E^2_{chl} and on E^2_{com} alike, so each column of the square is one curve and there is no auxiliary curve to send. The signature carries E_{com}, q and the action of σ on the 2^r-torsion.

specification's response exponent: the auxiliary isogeny ρ, which the signer computes and whose target E_{aux} it sends (Figure 1, left). In dimension 4 the second isogeny is replaced by two integers with $a_1^2 + a_2^2 = 2^e - q$, which exist when $2^e - q$ is a prime congruent to 1 modulo 4 and which the verifier finds from q alone by Cornacchia's algorithm, so that the chain runs on $E^2_{\mathrm{com}} \times E^2_{\mathrm{chl}}$ and involves no other curve (Figure 1, right).

The signature therefore carries only the commitment curve, the degree q, and the action of σ on a basis of the 2^r-torsion: a 2×2 matrix modulo 2^r, of which three entries are sent and the fourth follows from the Weil pairing, $ad - bc \equiv kq \pmod{2^r}$ with k fixed by the two bases [DLRW24, Section 6.1]. At level I, with every field rounded up to whole bytes, that is $2 \cdot 41 = 82$ bytes of curve, 22 bytes of q, three 12-byte scalars and two one-byte basis hints: 142 bytes, against 200 for the specification's format, which carries the auxiliary curve, a matrix modulo 2^{198}, the λ-bit challenge and two hints [AAA+26, Chapter 6]. Each format sends one curve: the specification sends E_{aux} and recovers E_{com} from its chain, whereas the dimension-4 verifier needs E_{com} before its chain starts, so the 82 bytes of the auxiliary curve are replaced by the 82 bytes of the commitment curve. The saving lies elsewhere: the specification's four matrix entries take 100 bytes and its challenge 16, against 36 bytes for three entries modulo 2^r and 22 for q, for 58 bytes in all. The smaller modulus is a consequence of the two half-chains meeting in the middle, which need the action of σ on the $2^{\lceil e/2 \rceil + 2}$-torsion only. Packing the fields at bit granularity, as our format does (Section 7.2), saves only 3 more, for 139 bytes. The challenge is not sent: the verifier recomputes it by hashing the public key, the commitment curve and the message, whereas the specification sends it and checks it against the hash of the commitment curve that its chain recovers.

4.2 The verifier

The verifier decodes the key and the signature, recomputes the challenge, and checks that the basis hints are the canonical ones of their curves. From the hints it builds a basis of the 2^f-torsion of the public-key curve, where 2^f is the largest power of 2 dividing $p+1$, as in the specification, and a basis (R_0, S_0) of $E_{\mathrm{com}}[2^f]$. No basis of $E_{\mathrm{chl}}[2^f]$ is built: the challenge isogeny, of degree 2^λ, maps the multiples by $2^{f-\lambda-r}$ of the public-key basis to a basis (P, Q) of $E_{\mathrm{chl}}[2^r]$, and $(R, S) = ([2^{f-r}]R_0, [2^{f-r}]S_0)$ is the basis of $E_{\mathrm{com}}[2^r]$; the action is written in these two bases. The verifier checks, in this order, that $e(P, Q)$ has exact order 2^r, that

the transmitted a and b are not both even, and that $e(R,S)$ has exact order 2^r, which it does by computing the integer k with $e(R,S) = e(P,Q)^k$ and requiring that it exist and be odd. It then completes the dropped scalar by the rule of Section 7.2, repeats the goodness test on q, writes $2^e - q = a_1^2 + a_2^2$ by Cornacchia's algorithm, and builds the kernel of the dimension-4 isogeny on $E_{\mathrm{com}}^2 \times E_{\mathrm{chl}}^2$ from a_1, a_2 and the transmitted action [DLRW24, Lemma 4 and Section 4.4]. Finally, it computes the two half-chains of $\lceil e/2 \rceil$ steps from both ends, and accepts when two conditions hold: the two middle codomains have the same theta null point, projectively (our code compares the null point of the first with the Hadamard transform of that of the second, since it evaluates the second half-chain through its dual), and the chain maps $(R_0, 0, 0, 0)$ to $(\pm[a_1]R_0, \pm[a_2]R_0, *, 0)$, the first two components being compared by their x-coordinates. These are the two conditions of SQIsignHD's Section 4.5 in the form in which its library checks them. Their Algorithm 5 fixes the signs, $([a_1]R_0, -[a_2]R_0, *, 0)$, which a chain computed on Kummer varieties does not determine. The signs are inessential: Proposition 1 below shows that their Proposition 15 holds for the condition with free signs, since its proof uses only the degrees a_1^2 and a_2^2. The component marked $*$, on which the transmitted action acts, need not be compared either. A signature thus fails on any decoding error, on a curve coefficient that is not that of a curve, on a hint that is not canonical, on a pairing that does not have exact order 2^r, on transmitted a and b that are both even, on a degree that is not good, on a failure of Cornacchia's algorithm or a value $m > 40$ (Appendix B), on a step that would divide by a vanishing value, as when the chain lands on a product of curves where it should not, or on either condition.

Proposition 1 (Soundness of the image check)**.** *Suppose that the verifier of Algorithm 1 accepts, and let $F = \widehat{\Phi}_2 \circ \Phi_1$ be the endomorphism of $A = E_{\mathrm{com}}^2 \times E_{\mathrm{chl}}^2$ defined by its two half-chains Φ_1 and Φ_2 (Definition 2). Then R_0 has order 2^f, the first column of F, in the matrix notation of Section 5.1, is $(\epsilon_1 a_1, \epsilon_2 a_2, \psi, 0)$ with $\epsilon_1, \epsilon_2 \in \{\pm 1\}$, and $\psi\colon E_{\mathrm{com}} \to E_{\mathrm{chl}}$ is an isogeny of degree q, which the chain evaluates as the third component of $x \mapsto F(x, 0, 0, 0)$.*

Most of these choices are SQIsignHD's: the response as an ideal of good norm and the goodness test; the half-chains of $\lceil e/2 \rceil$ steps from both ends, with $r = \lceil e/2 \rceil + 2$; the action modulo 2^r with one scalar recovered from the pairing; the two acceptance conditions; and the library's basis-from-hint convention, which we keep so that its verifier checks our signatures unmodified. Ours are the move to the round-3 primes, with e and its margin; the round-3 commitment and λ-bit challenge, recomputed rather than sent; the checks on the hints and on the orders of the pairings; the bit-packed wire format, with two hint bytes in place of the library's packed bits; and the signer's sampler, which draws lattice vectors of the response ideal and rejects on goodness.

Where the verifier's time goes. The verifier pays for the smaller signature. Its chain of dimension-4 isogeny steps, 87 from each end at level I, works on sixteen theta coordinates instead of the four of a dimension-2 step, and costs 6.8 times the dimension-2 verifier of the same crate at the same prime (76.5 against 11.2 megacycles), and 7.0 times the reference's assembly build (Section 8). Almost all of this time is spent in the steps after the gluing. A profile of the verifier at level I attributes 60 % of it to doubling the four generators of each step's kernel and pushing them, with the other saved points, through the chain, and 15 % to computing each step's codomain and its constants from those four points; the dimension-2 steps, the gluing and the final checks take the rest. A substantial saving must therefore reduce the work spent on kernel points, at every step.

The structure at the gluing. The step formulas compute in theta coordinates, and each step needs the codomain's theta structure to be the one induced by the images of

Algorithm 1 Verification of a dimension-4 signature.

Require: public key, message μ, signature $(A_{\text{com}}, q, a, b, c \text{ or } d, h_R, h_S)$
Ensure: accept or reject

1: decode the key and the signature, rejecting malformed input
2: $E_{\text{com}} \leftarrow$ the curve with coefficient A_{com}; reject unless the hints of the public-key curve and of E_{com} are canonical
3: $E_{\text{chl}} \leftarrow$ the challenge curve, recomputed from the public key, E_{com} and μ; $(P, Q) \leftarrow$ the basis of $E_{\text{chl}}[2^r]$ that is the image, under the challenge isogeny, of the hinted basis of the public-key curve; reject unless $e(P, Q)$ has exact order 2^r
4: reject if a and b are both even
5: $(R, S) \leftarrow$ the basis of $E_{\text{com}}[2^r]$ from the hinted basis (R_0, S_0) of $E_{\text{com}}[2^f]$; reject unless $e(R, S)$ has exact order 2^r
6: complete $M = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ by the rule of Section 7.2
7: **if** $q \geq 2^e$ or $2^e - q$ is not a probable prime congruent to 1 modulo 4 **then**
8: reject
9: **end if**
10: $(a_1, a_2) \leftarrow$ Cornacchia$(2^e - q)$, a_1 odd; $m \leftarrow v_2(a_2)$
11: the starting matrices M_1, M_2 and the kernel generators from a_1, a_2 and M (Lemma 9)
12: **for** $h \in \{1, 2\}$ **do**
13: compute the half-chain Φ_h: m steps in dimension 2, the gluing, and the plain steps up to step e_h (Section 2.1)
14: **end for**
15: **return** accept if the two middle codomains have the same theta null point, projectively, and the chain maps $(R_0, 0, 0, 0)$ to $(\pm[a_1]R_0, \pm[a_2]R_0, *, 0)$, else reject

a particular symplectic basis (Definition 2). Before the dimension-4 gluing, the verifier changes coordinates on the product of the two dimension-2 codomains by closed formulas from Dartois's work [Dar24, Lemmas 30 and 31], which he states with a brief justification and, for the second half-chain, from a dimension-2 basis given only modulo 4 [Dar24, Appendix C.1]. The correctness of every later step rests on these formulas, and so does any argument about the coordinates in which the chain is computed, such as the one Section 5 needs.

5 Two-point verification

Section 4.2 identified two issues: most of the verifier's time goes to the four kernel generators that each step doubles and pushes and from which it computes its codomain, and the structure in which the chain is computed after the gluing rests on closed formulas given without a full derivation. This section addresses the first, and Section 6 the second. It shows that half of the work on kernel points can be avoided, by exploiting a symmetry of the chain that MIKE uses in a different setting [BDD+26, Appendix E.2].

On E^2, the automorphism $J(P, Q) = (Q, -P)$ satisfies $J^2 = -1$, and both the endomorphism $\alpha = a_1 + a_2 J$ of the Kani matrix and $\sigma \oplus \sigma$ commute with it. Hence $D = \text{diag}(J, J)$ on $E_{\text{com}}^2 \times E_{\text{chl}}^2$ commutes with the verifier's isogeny, descends to an automorphism $D_{h,k}$ of every fourfold of both half-chains (Proposition 3), and maps the four kernel generators onto one another in pairs, $Dg_0 = -g_2$ and $Dg_1 = -g_3$ (Lemma 11). What remains is to know how $D_{h,k}$ acts on the theta coordinates in which the verifier computes. Theorem 2 answers this: for every input, honest or not, and at every step from the gluing on, $D_{h,k}$ acts on these coordinates as one fixed permutation π of the sixteen coordinates, which exchanges index bits 0 and 2 and bits 1 and 3 (Theorem 2). The proof combines the transport of theta structures along isomorphisms (Lemma 5), the explicit starting matrices of the

verifier (Lemmas 8–11), and the change of basis at the gluing (Proposition 6). Sections 5.1 to 5.6 develop this argument, Section 5.7 gives the resulting algorithm, and Appendix A contains the proofs.

5.1 Notation

Throughout, E denotes a supersingular elliptic curve over $\mathbb{F}_{p^2}$ and $E^2 = E \times E$. Homomorphisms between squares of curves are written as 2×2 matrices whose entries are homomorphisms between the curves, acting on column vectors:

$$\begin{pmatrix} u & v \\ w & z \end{pmatrix} \begin{pmatrix} P \\ Q \end{pmatrix} = \begin{pmatrix} u(P) + v(Q) \\ w(P) + z(Q) \end{pmatrix}.$$

An integer n stands for the multiplication map $[n]$. For a homomorphism ψ we write $\widehat{\psi}$ for its dual. With respect to the product polarizations, the dual of a matrix is the transpose of the matrix of duals; in particular the dual of an integer matrix is its transpose. We write E_{chl} for the challenge curve, E_{com} for the commitment curve, $\sigma\colon E_{\mathrm{chl}} \to E_{\mathrm{com}}$ for the response, of odd degree q, and a_1, a_2 for integers with $a_1^2 + a_2^2 = 2^e - q$, as in Section 4. We orient σ from E_{chl} to E_{com}, the reverse of the usual presentation of the response as a map from the commitment curve to the challenge curve; the transmitted action is then that of its dual $\widehat{\sigma}$, and nothing below depends on this choice. Finally

$$A_1 = E_{\mathrm{chl}}^2, \qquad A_2 = E_{\mathrm{com}}^2, \qquad A = A_1 \times A_2,$$

so that A is the abelian fourfold on which the verifier's chain starts.

5.2 The automorphism J of E^2

We begin on a single square E^2.

Definition 1. On any square E^2, let

$$J = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \qquad \text{that is,} \qquad J(P, Q) = (Q, -P),$$

and on $A = A_1 \times A_2$ let $D = \mathrm{diag}(J, J)$ be J acting on each square separately.

Lemma 1.

(i) $J^2 = -1$, *so* J *has order 4.*

(ii) The endomorphism $\alpha = \left(\begin{smallmatrix} a_1 & a_2 \\ -a_2 & a_1 \end{smallmatrix}\right)$ *of Figure 1 equals* $a_1 + a_2 J$, *and its dual is* $\widehat{\alpha} = \alpha^{\mathsf{T}} = a_1 - a_2 J$.

(iii) $\widehat{\alpha}\,\alpha = \alpha\,\widehat{\alpha} = a_1^2 + a_2^2 = 2^e - q$.

(iv) $\widehat{J} = J^{\mathsf{T}} = -J = J^{-1}$, *so* J *preserves the product polarization of* E^2*: it is an automorphism of* E^2 *as a principally polarized abelian surface.*

5.3 The verifier's isogeny commutes with D

We now show that the verifier's isogeny commutes with $D = \mathrm{diag}(J, J)$. Write $\Sigma = \mathrm{diag}(\sigma, \sigma)\colon A_1 \to A_2$, the map $\sigma \oplus \sigma$ of Figure 1, so that $\widehat{\Sigma} = \mathrm{diag}(\widehat{\sigma}, \widehat{\sigma})$. Consider the Kani endomorphism [DLRW24, Section 4]

$$F = \begin{pmatrix} \alpha & \widehat{\Sigma} \\ -\Sigma & \widehat{\alpha} \end{pmatrix} : A_1 \times A_2 \longrightarrow A_1 \times A_2, \qquad F(x_1, x_2) = \big(\alpha(x_1) + \widehat{\Sigma}(x_2),\ -\Sigma(x_1) + \widehat{\alpha}(x_2)\big),$$

where the α in the top-left corner acts on A_1 and the $\widehat{\alpha}$ in the bottom-right corner on A_2. Its dual is

$$\widehat{F} = \begin{pmatrix} \widehat{\alpha} & -\widehat{\Sigma} \\ \Sigma & \alpha \end{pmatrix}.$$

The verifier's dimension-4 isogeny is this endomorphism with the roles of the two curves exchanged: $(E_{\mathrm{chl}}, E_{\mathrm{com}}, \sigma)$ replaced by $(E_{\mathrm{com}}, E_{\mathrm{chl}}, \widehat{\sigma})$, so that it acts on $E_{\mathrm{com}}^2 \times E_{\mathrm{chl}}^2$ (Lemma 9 gives it explicitly). Nothing below depends on which curve is which, so every statement about F and $\widehat{F}$ applies to it.

Lemma 2. *$\widehat{F}F = F\widehat{F} = 2^e$. Hence F is an isogeny of degree 2^{4e}, and $\ker F = \widehat{F}\big(A[2^e]\big)$.*

Proposition 2. *$F \circ D = D \circ F$. In particular $\ker F$ is stable under D.*

5.4 Consequences along the chain

The verifier computes F as a chain of e isogenies of degree 2. Write $K = \ker F$ and, for $0 \le k \le e$, let $F_k \colon A \to A_k = A/[2^{e-k}]K$ be the partial chain after k steps, so that $F_e = F$ up to isomorphism. The two half-chains that meet in the middle are partial chains of this kind, one read from each end.

Proposition 3. *For every k, there is an automorphism D_k of A_k with $D_k \circ F_k = F_k \circ D$. It satisfies $D_k^2 = -1$ and $\widehat{D_k} = -D_k$, and it preserves the principal polarization of A_k.*

Proposition 4. *For any matrix the verifier receives, honest or not, the kernel it builds is stable under D. Whenever that kernel defines a chain, that is, when it is maximal isotropic and the computation succeeds, the conclusion of Proposition 3 holds for that chain.*

5.5 The verifier's theta coordinates

Informally, a level-2 theta structure on a fourfold is a choice of coordinates: the sixteen functions θ_a, determined up to a common scalar by a basis of the 4-torsion that is compatible with the Weil pairing. Changing the basis changes the coordinates by an explicit linear map (Lemma 4), and an automorphism that permutes the basis permutes the coordinates (Lemma 5). The verifier never chooses these bases freely: each step of its chain hands a basis to the next (the index convention below), and this is what lets us follow D from one fourfold to the next.

This subsection proves how D acts in level-2 theta coordinates once it acts correctly on the basis inducing them; Section 5.6 proves that it does, on the theta structures of the verifier's algorithm. Throughout, π is the permutation of $\{0, \dots, 15\}$ that exchanges bits 0 and 2 and bits 1 and 3 of the index $a = i_0 + 2i_1 + 4i_2 + 8i_3$; it is an involution. We identify a with its multi-index $\mathbf{i} = (i_0, i_1, i_2, i_3) \in \{0,1\}^4$, write $\theta_{\mathbf{v}}$ for $\theta_{\mathbf{v} \bmod 2}$ when $\mathbf{v}$ is an integer vector, and $\langle \mathbf{i} \mid \mathbf{j} \rangle = \sum_l i_l j_l$.

Theta structures and bases Let A' be a principally polarized abelian fourfold over an algebraically closed field $\Bbbk$ of odd characteristic (the varieties of the chain are defined over $\mathbb{F}_{p^2}$, and we work over its algebraic closure, as does [Dar24, Section 2]), let $\mathcal{L} = \mathcal{L}_0^2$ with $\mathcal{L}_0$ a symmetric line bundle representing the principal polarization, so that $\mathcal{L}$ is totally symmetric of type $(2,2,2,2)$, and let ζ be a primitive fourth root of unity. By [Dar24, Theorem 5], a symmetric level-2 theta structure for $\mathcal{L}$ is determined by a symplectic isomorphism from $(\mathbb{Z}/4)^4 \times (\mathbb{Z}/4)^4$ to $A'[4]$, that is, by the image $B = (s_0, \dots, s_3; t_0, \dots, t_3)$ of the ζ-canonical basis $(e_0, \dots, e_3; \chi_0, \dots, \chi_3)$, a ζ-symplectic basis of $A'[4]$. We call it the structure *induced by* B and write θ_a^B for its coordinates. Bit l of the index is the l-th coordinate of $(\mathbb{Z}/2)^4$ [Dar24, Section 3.2], so it belongs to the pair (s_l, t_l). For a tuple $x = (x_0, \dots, x_3)$ of points and a 4×4 integer matrix U we write xU for the tuple $\big(\sum_l U_{lj} x_l\big)_j$, as for coefficient

vectors, and for $M = \left(\begin{smallmatrix} U & V \\ W & U' \end{smallmatrix}\right)$ with 4×4 blocks, $BM = (sU + tW; sV + tU')$: the columns of M give the new basis in terms of the old.

Index convention In a chain the half t of the basis inducing the structure of each fourfold lies above the kernel of the next step [Dar24, Section 4.1], and positions are kept from step to step: if the structure of A_k is induced by $(s;t)$ and the step f uses 8-torsion points T'' with $2T'' = t$, the structure of its codomain is induced by $(f(s); f(T''))$, position by position [DLRW24, Theorem 56], [Dar24, Section 4.2]. On every fourfold of a half-chain, index bit l is therefore labeled by the l-th kernel generator g_l, which the verifier takes from column $4 + l$ of its starting matrix, in column order (Lemma 7). Remark 1 records that our crate follows this convention.

Lemma 3 (The product)**.** *Give both copies of a curve E the same level-2 theta structure, so that $\theta_{(i,j)}(P,Q) = \theta_i(P)\,\theta_j(Q)$ on E^2. Then $\theta_{(i,j)}(J(P,Q)) = \theta_{(j,i)}(P,Q)$: J exchanges the two index bits of the square, with no scalars.*

The next lemma is Dartois's change-of-coordinates theorem [Dar24, Theorem 12] in our notation, with his blocks $\left(\begin{smallmatrix} A & C \\ B & D \end{smallmatrix}\right)$ written $\left(\begin{smallmatrix} U & V \\ W & U' \end{smallmatrix}\right)$, followed by the special case $V = 0$, which is the one we use.

Lemma 4 (Change of structure)**.** *Let A', $\mathcal{L}$ and ζ be as above, B a ζ-symplectic basis of $A'[4]$ and $M = \left(\begin{smallmatrix} U & V \\ W & U' \end{smallmatrix}\right) \in \mathrm{Sp}_8(\mathbb{Z}/4)$. Then BM is a ζ-symplectic basis, and there is $\lambda \in \Bbbk^*$ such that, for all $\mathbf{i}$,*

$$\theta^{BM}_{\mathbf{i}} = \lambda \sum_{\mathbf{j} \in \{0,1\}^4} \zeta^{\langle \mathbf{i} | \mathbf{j} \rangle - \langle U\mathbf{i} + V\mathbf{j} + 2\mathbf{i}_0 \,|\, W\mathbf{i} + U'\mathbf{j} \rangle}\, \theta^{B}_{U\mathbf{i} + V\mathbf{j} + \mathbf{i}_0}$$

as an identity of sections of $\mathcal{L}$, for every $\mathbf{i}_0$ such that the section

$$T_{\mathbf{i}_0} = \sum_{\mathbf{j}} \zeta^{-\langle V\mathbf{j} + 2\mathbf{i}_0 | U'\mathbf{j} \rangle} \theta^{B}_{\mathbf{i}_0 + V\mathbf{j}}$$

is nonzero; any two such $\mathbf{i}_0$ give the same coordinates up to a scalar. If $V = 0$, then:

(i) $U^{\mathsf{T}}U' \equiv 1 \pmod 4$*, so $U' = U^{-\mathsf{T}}$ modulo 4;*

(ii) $\mathbf{i}_0 = 0$ *is admissible and is the only admissible index, whatever the theta null point;*

(iii) $\theta^{BM}_{\mathbf{i}} = \lambda\, \zeta^{-\langle U\mathbf{i} | W\mathbf{i} \rangle}\, \theta^{B}_{U\mathbf{i}}$*; in particular* $\theta^{BM}_{\mathbf{i}} = \lambda\, \theta^{B}_{U\mathbf{i}}$ *when also $W = 0$.*

Lemma 5 (Transport)**.** *Let A', $\mathcal{L}$, ζ and B be as in Lemma 4, and let φ be an automorphism of A' preserving its principal polarization. Then $\varphi(B)$ is ζ-symplectic, and there is $c \in \Bbbk^*$ such that $\theta^{\varphi(B)}_a(\varphi(z)) = c\,\theta^{B}_a(z)$ for all $z \in A'$ and all a. Consequently, if $\varphi(B) = BM$ with $M = \mathrm{diag}(U, U')$, there is $\mu \in \Bbbk^*$ such that*

$$\theta^{B}_a(\varphi(z)) = \mu\, \theta^{B}_{U^{-1}a}(z) \qquad \text{for all } z \text{ and } a:$$

φ acts on the coordinates of the structure induced by B through the inverse of the matrix by which it acts on B.

Lemma 6 (One step)**.** *Let $f\colon A_k \to A_{k+1}$ be a step of the chain and U a signed permutation matrix, so that $U^{-\mathsf{T}} = U$. Suppose that D_k acts on the basis $(s;t)$ inducing the structure of A_k by $\mathrm{diag}(U, U)$ modulo 4, and that the 8-torsion points T'' with $2T'' = t$ used by the step satisfy $D_k T'' = T''U$ exactly, as points. Then D_{k+1} acts on the basis $(f(s); f(T''))$, which induces the structure of A_{k+1}, by the same block matrix modulo 4.*

The 8-torsion points T'' of the hypothesis are the chain's kernel generators, doubled and pushed to step k. Since every step commutes with D, they inherit an exact pairing $Dg_0 = \pm g_2$, $Dg_1 = \pm g_3$ from the generators on A, if the generators have it.

Theorem 1 (Conditional on the base case)**.** *Let U be a signed permutation matrix whose reduction modulo 2 is the permutation matrix of $(0\ 2)(1\ 3)$. Suppose that, for one half-chain, the four kernel generators on A satisfy $Dg_0 = \pm g_2$ and $Dg_1 = \pm g_3$ as points, and that D acts on the basis inducing the theta structure of the gluing codomain by $\mathrm{diag}(U, U)$ modulo 4. Then on every fourfold of that half-chain from the gluing codomain on, D_k acts on the level-2 theta coordinates as π, that is, $\theta_a(D_k z) = \mu_k\, \theta_{\pi(a)}(z)$ for all z and a, and the two-point chain of Section 5.7 computes the same kernels, isogenies and codomains as the four-point chain.*

Lemma 7 (The structured kernel block)**.** *In the verifier's coefficient layout, where row r of an 8×8 matrix is the coefficient of the base point $\lfloor r/4 \rfloor$ on copy $r \bmod 4$, with copies $0, 1$ on E_{com} and $2, 3$ on E_{chl}, D acts on coefficient vectors by $(v_0, v_1, v_2, v_3) \mapsto (v_1, -v_0, v_3, -v_2)$ on each half. The kernel block of the starting matrix has the columns, in this order,*

$$\begin{aligned} g_0 &= (a_1, a_2, 1, 0 \mid 0, 0, 0, 0), & g_1 &= (0, 0, 0, 0 \mid a_1, a_2, q, 0), \\ g_2 &= (-a_2, a_1, 0, 1 \mid 0, 0, 0, 0), & g_3 &= (0, 0, 0, 0 \mid -a_2, a_1, 0, q), \end{aligned}$$

where the bar separates the two halves, and D maps g_0 to $-g_2$ and g_1 to $-g_3$.

The verifier's generators are not this block, however: they are columns of matrices obtained from it through generic symplectic completions, as the next subsection explains.

5.6 The base case

This subsection proves the hypotheses of Theorem 1 for the verifier's algorithm, on the theta structures that algorithm uses (Definition 2): for the verifier's own starting matrices, whatever symplectic completion it uses, and for every input for which its half-chains are defined. Whether our implementation computes those structures is a separate question, taken up in Appendix B.

Base points Let (P, Q) be the basis of $E_{\mathrm{chl}}[2^r]$ and $(R, S) = ([2^{f-r}]R_0, [2^{f-r}]S_0)$ the basis of $E_{\mathrm{com}}[2^r]$ that the verifier uses (Section 4.2), and k the integer with $e(R, S) = e(P, Q)^k$, where e is the Weil pairing on 2^r-torsion. Bases built from hints need not be bases: a crafted hint can give points of smaller order, and then $e(R, S)$ has smaller order and k is even, or k does not exist. The verifier therefore rejects unless $e(P, Q)$ and $e(R, S)$ both have exact order 2^r, which makes k odd (Section 7.2). The verifier reads the transmitted matrix $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$, completed by the rule of Section 7.2, as the homomorphism $\tau\colon E_{\mathrm{com}}[2^r] \to E_{\mathrm{chl}}[2^r]$ with $\tau(R) = aP + bQ$ and $\tau(S) = cP + dQ$; for an honest signature, τ is the restriction of the dual $\widehat{\sigma}$ of the response. Let $\lambda \equiv q^{-1} \pmod{2^r}$. The base points are R and S on copies 0 and 1, and $\tau(R)$ and $\lambda\tau(S)$ on copies 2 and 3: write b_n, for $0 \le n \le 7$, for the base point $\lfloor n/4 \rfloor$ on copy $n \bmod 4$, so that a coefficient vector $v \in (\mathbb{Z}/2^r)^8$ stands for the point $\sum_n v_n b_n$ of $A[2^r]$, and a matrix acting on coefficient vectors has as column n the image of b_n. Let $\Omega = \left(\begin{smallmatrix} 0 & I \\ -I & 0 \end{smallmatrix}\right)$, with 4×4 blocks, and $\omega(u, v) = u^{\mathsf{T}} \Omega v$.

Lemma 8 (Base points)**.** *Suppose that q is odd, that $e(P, Q)$ and $e(R, S)$ have exact order 2^r, and that the rule of Section 7.2 is defined, that is, the entry it inverts is odd. Then $e(\tau R, \tau S) = e(R, S)^q$, the points $b_0, \ldots, b_7$ form a basis of $A[2^r]$, and, for the pairing e_A of the product polarization and $\zeta = e(R, S)$,*

$$e_A\Big(\sum_n u_n b_n, \sum_n v_n b_n\Big) = \zeta^{\omega(u,v)}.$$

In particular, a matrix over $\mathbb{Z}/2^r$ is symplectic for ω exactly when its columns, applied to the base points, form a ζ-symplectic basis of $A[2^r]$.

When q is even, when either pairing has smaller order, or when the entry the rule inverts is even, the verifier rejects; such inputs are not among those for which the half-chains are defined.

The Kani matrices Since $q\lambda \equiv 1$, we have $\tau(xR + yS) = x \cdot \tau R + qy \cdot \lambda\tau S$. Let $\widehat{\tau}\colon E_{\mathrm{chl}}[2^r] \to E_{\mathrm{com}}[2^r]$ map $u \cdot \tau R + v \cdot \lambda\tau S$ to $qu \cdot R + v \cdot S$; then $\widehat{\tau}\tau = q$ and $\tau\widehat{\tau} = q$, and for an honest signature $\widehat{\tau}$ is the restriction of σ. With $T = \mathrm{diag}(\tau, \tau)$ and $\widehat{T} = \mathrm{diag}(\widehat{\tau}, \widehat{\tau})$, the verifier uses, on $A[2^r] = E_{\mathrm{com}}^2[2^r] \times E_{\mathrm{chl}}^2[2^r]$,

$$F = \begin{pmatrix} \alpha & \widehat{T} \\ -T & \widehat{\alpha} \end{pmatrix}, \qquad \widehat{F} = \begin{pmatrix} \widehat{\alpha} & -\widehat{T} \\ T & \alpha \end{pmatrix},$$

which for an honest signature are the restrictions to $A[2^r]$ of the endomorphisms F and $\widehat{F}$ of Section 5.3 with the two curves exchanged, as stated there. The kernel lifts of Lemma 7 are the images under $\widehat{F}$ of b_0, b_4, b_1 and b_5, in that order.

Lemma 9 (The Kani matrices)**.** *On coefficient vectors, $F = \mathrm{diag}(F_1, F_2)$ and $\widehat{F} = \mathrm{diag}(\widehat{F}_1, \widehat{F}_2)$, where $F_1, \widehat{F}_1$ act on the coefficients of the first base points (rows 0 to 3) and $F_2, \widehat{F}_2$ on those of the second (rows 4 to 7), and, in 2×2 blocks with α, $\widehat{\alpha}$ the integer matrices of Lemma 1 and an integer standing for that multiple of I_2,*

$$F_1 = \begin{pmatrix} \alpha & q \\ -1 & \widehat{\alpha} \end{pmatrix}, \quad F_2 = \begin{pmatrix} \alpha & 1 \\ -q & \widehat{\alpha} \end{pmatrix}, \quad \widehat{F}_1 = \begin{pmatrix} \widehat{\alpha} & -q \\ 1 & \alpha \end{pmatrix}, \quad \widehat{F}_2 = \begin{pmatrix} \widehat{\alpha} & -1 \\ q & \alpha \end{pmatrix}.$$

Let $\tilde{D}$ be the action of D on coefficient vectors (Lemma 7). The following identities hold over $\mathbb{Z}[a_1, a_2, q]$, the first with $a_1^2 + a_2^2 = 2^e - q$ by the choice of a_1, a_2:

$$\begin{gathered} F\widehat{F} = \widehat{F}F = (a_1^2 + a_2^2 + q)\, I = 2^e I, \qquad \omega(Fu, v) = \omega(u, \widehat{F}v), \\ F\tilde{D} = \tilde{D}F, \qquad \widehat{F}\tilde{D} = \tilde{D}\widehat{F}, \qquad \omega(\tilde{D}u, \tilde{D}v) = \omega(u, v). \end{gathered} \tag{4}$$

In particular the first kernel block of Lemma 7 is $X^{(1)} = \widehat{F}E$, where E is the matrix with columns the standard basis vectors e_0, e_4, e_1 and e_5, in that order. We have $e \geq r$: $r = \lceil e/2 \rceil + 2 \leq e$ as soon as $e \geq 4$, and $(e, r) = (174, 89)$, $(264, 134)$ and $(346, 175)$ at levels I, III and V (Table 1). So the matrix $2^e I$ vanishes modulo 2^r.

A starting matrix M has as columns 0 to 3 the completion Y and as columns 4 to 7 the kernel lifts X; by Lemma 8 its columns, applied to the base points, give a symplectic basis of $A[2^r]$ exactly when M is symplectic for ω.

Lemma 10 (Any completion)**.** *Let $M = (Y \mid X)$ be symplectic over $\mathbb{Z}/2^r$, that is, $\omega(Y, X) = I$ and $\omega(Y, Y) = \omega(X, X) = 0$ column by column, and suppose that $\tilde{D}X = XU$ for a signed permutation matrix U. Then the columns of X span a Lagrangian direct summand of $(\mathbb{Z}/2^r)^8$, and $\tilde{D}Y = YU + XS$ with $S = \omega(Y, \tilde{D}Y - YU)$, a 4×4 matrix over $\mathbb{Z}/2^r$.*

Lemma 11 (The generators are paired)**.** *For the two starting matrices built by the verifier, $M_1 = (Y^{(1)} \mid -\widehat{F}Y^{(2)})$ and $M_2 = (Y^{(2)} \mid X^{(2)})$, with any symplectic completions $Y^{(1)}$, $Y^{(2)}$, both matrices are symplectic, and $\tilde{D}$ acts on the kernel columns of each by the same signed permutation U, exactly modulo 2^r: it maps the first kernel column to minus the third, the second to minus the fourth, the third to the first and the fourth to the second. Consequently $Dg_0 = -g_2$, $Dg_1 = -g_3$ as points of order 2^r, for the generators of both half-chains, and $\tilde{D}Y^{(h)} = Y^{(h)}U + X_h S_h$ for $h = 1, 2$, where $X_1 = -\widehat{F}Y^{(2)}$, $X_2 = X^{(2)}$ and S_h is as in Lemma 10.*

Definition 2 (The algorithm's theta structures)**.** Let $e_1 = \lceil e/2 \rceil$ and $e_2 = e - e_1$. The verifier computes F as two half-chains from A: Φ_1, the first e_1 steps of F, and Φ_2, the first e_2 steps of $\widehat{F}$, so that $F = \widehat{\Phi}_2 \circ \Phi_1$ up to isomorphism. For $h \in \{1, 2\}$, applying the columns of $2^{r-2-e_h} M_h$ to the base points gives, by Lemmas 8 and 11, a symplectic basis $(y'_{h,0}, \ldots, y'_{h,3}; x'_{h,0}, \ldots, x'_{h,3})$ of $A[2^{e_h+2}]$, with $\ker \Phi_h = \langle 4x'_{h,j} \rangle$. For $m \le k \le e_h$, with m as in Lemma 12 below, write $\Phi_{h,k}$ for the first k steps of Φ_h and $A_{h,k}$ for its codomain, and put $x'_{h,k,j} = 2^{e_h-k} x'_{h,j}$, of order 2^{k+2}, and $y_{h,j} = 2^{e_h} y'_{h,j} \in A[4]$. The *algorithm's theta structure* on $A_{h,k}$ is the level-2 structure induced by

$$B_{h,k} = \big(\Phi_{h,k}(y_{h,0}), \ldots, \Phi_{h,k}(y_{h,3});\ \Phi_{h,k}(x'_{h,k,0}), \ldots, \Phi_{h,k}(x'_{h,k,3})\big).$$

These are the structures in which SQIsignHD's half-chain algorithm computes [DLRW24, Appendix F], [Dar24, Section 5.3 and Appendix C], as an induction on k shows. Each half-chain starts with m steps in dimension 2, and the gluing step $m+1$ is computed in the structure induced by the symplectic basis $B_{h,m}$ of $A_{h,m}[4]$. This is [Dar24, Appendix C.2.2 and Lemma 31] in our notation. There the symplectic basis of $A[2^r]$ given by M_h lists the completion first and the kernel lifts second, the kernel of Φ_h is 2^{r-e_h} times the span of the kernel lifts, and the gluing uses the basis formed by $2^{r-2}\Phi_{h,m}$ of the completion and $2^{r-2-m}\Phi_{h,m}$ of the kernel lifts, which is $B_{h,m}$; Proposition 6 proves that the change of basis the algorithm applies at the gluing expresses exactly this basis. A step f from $A_{h,k}$ uses 8-torsion points T'' with $4T''$ spanning its kernel; completed to a symplectic basis $(S''; T'')$ of $A_{h,k}[8]$ above the basis inducing the structure of $A_{h,k}$, it gives the codomain the structure induced by the symplectic basis $(2f(S''); f(T''))$ [Dar24, Sections 4.1 and 4.2]. This is [DLRW24, Theorem 56] in the form in which [Dar24] states and applies it, with the half above the kernel second, as here. With $2S'' = \Phi_{h,k}(y_h)$ and $T'' = \Phi_{h,k}(x'_{h,k+1})$, that basis is $B_{h,k+1}$. Theorem 2 below is a statement about this algorithm.

Lemma 12 (The gluing codomain)**.** *For every input for which the verifier's half-chains are defined, a_1 is odd, a_2 is even and nonzero, and the dimension-4 gluing is step $m+1$ of each half-chain, with $m = v_2(a_2) \ge 1$. So the gluing codomain is $A_{h,m+1}$, with $m + 1 \ge 2$.*

Theorem 2 (The base case)**.** *For every input for which the verifier's half-chains are defined, honest or not, with any symplectic completions, for $h \in \{1, 2\}$ and every k with $\max(2, m) \le k \le e_h$, the automorphism $D_{h,k}$ of $A_{h,k}$ induced by D acts on the coordinates of the algorithm's theta structure on $A_{h,k}$ as the permutation π: there is $\mu \in \Bbbk^*$ with $\theta_a(D_{h,k} z) = \mu\, \theta_{\pi(a)}(z)$ for all $z \in A_{h,k}$ and all a. In particular the hypotheses of Theorem 1 hold for both half-chains, and from the gluing codomain on, the two-point algorithm computes the same kernels, isogenies and codomains as the four-point algorithm.*

Proposition 5 (Symmetric null points)**.** *For $h \in \{1, 2\}$ and $\max(2, m) \le k \le e_h$, the theta null point of $A_{h,k}$ in the algorithm's theta structure satisfies $\theta_{\pi(a)}(0) = \theta_a(0)$ for every index a, and so does its Hadamard transform. Both lie in the 10-dimensional space of vectors fixed by π.*

The per-step constants derived from the null point, the dual null point and its inverse, are therefore determined by their values on the ten orbits of π. Points pushed along the chain enjoy no such symmetry: no relation $\theta_a = \pm\theta_{\pi(a)}$ holds identically on $A_{h,k}$, since the θ_a are linearly independent.

The bound $k \ge 2$ matters only when $m = 1$: on $A_{h,1}$ the action is the permutation twisted by signs, which does not affect the verifier, whose two-point chain starts at the gluing codomain $A_{h,m+1}$.

5.7 The two-point verifier

Algorithm 2 gives the plain steps of a half-chain with two kernel points. For a vector x of sixteen coordinates, $\pi(x)$ denotes the vector with coordinates $\pi(x)_i = x_{\pi(i)}$.

Algorithm 2 The plain steps of a half-chain with two kernel points.

Require: the gluing codomain $A_{h,m+1}$ and its theta null point $\theta(0)$; the gluing images $\theta(g_0), \ldots, \theta(g_3)$ of the kernel generators; the points to evaluate; a strategy
Ensure: the codomains $A_{h,k}$, $m+1 < k \leq e_h$, and the images of the points to evaluate
1: **if** $\theta(g_2) \neq \pi(\theta(g_0))$ or $\theta(g_3) \neq \pi(\theta(g_1))$ projectively, or $\pi(\theta(0)) \neq \theta(0)$ **then**
2: **return** the four-point chain from the same inputs
3: **end if**
4: **for** each plain step **do**
5: $T_0, T_1 \leftarrow$ the 8-torsion multiples of g_0, g_1 above the step's kernel, by doubling (3) as the strategy prescribes
6: $T_2 \leftarrow \pi(T_0)$, $T_3 \leftarrow \pi(T_1)$ (Theorem 1)
7: compute the codomain from $T_0, \ldots, T_3$ by (2), on the ten orbits of π (Proposition 5)
8: push the kept multiples of g_0, g_1 and the points to evaluate through the step by (1)
9: **end for**

Since theta coordinates are even functions, it follows that $\theta(g_2) = \pi(\theta(g_0))$ and $\theta(g_3) = \pi(\theta(g_1))$ up to scalars at every step, and the step formulas use each kernel point only up to a scalar. A verifier can therefore double and push only g_0 and g_1 along each half-chain and obtain the other two generators by permuting coordinates; it then computes exactly the isogenies and codomains of the four-point verifier (Theorem 1). This removes half of the doublings and evaluations of kernel points, and in our measurements the two-point verifier took 0.63, 0.62 and 0.60 times the cycles of the four-point one at levels I, III and V, on the same signature (Section 8.3).

The symmetry also shortens the computation of each step's codomain. Since squaring and the Hadamard transform commute with the permutation of coordinates by π, two of the four squared kernel vectors from which the codomain is computed are permutations of the other two. Moreover, by Proposition 5, the codomain's null point and its dual are fixed by π, so the vectors that the step computes from them, the dual null point, its inverse and the next step's doubling constants, are determined by their values on the ten orbits of π. MIKE's reference implementation exploits the same structure [BDD+26], and we adopt two of its techniques: the dual null point and its inverse are each read off the two pushed kernel points as products along a fixed tree on the ten orbits, a path through nine of them and one leaf, in 25 multiplications, and the Hadamard transform of a vector fixed by π takes 32 additions and 4 doublings instead of 64 additions. Like MIKE's, this computation uses the symmetry without checking it: it rests on Theorem 2, which applies to our verifier by Appendix B. It never divides by zero: a vanishing coordinate among those the codomain formula reads sends the step back to the full computation, and a vanishing coordinate among the doubling constants makes the verifier reject, as it does with four points. Finally, as in MIKE's implementation, a codomain's doubling constants are computed only when a doubling follows, which happens after about 43 % of the steps of an optimal strategy at level I. All of them are included in the measurements of Table 4.

6 The change of basis at the gluing

Each half-chain of the verifier starts in dimension 2. Its first m steps, where $m = v_2(a_2)$, are a chain φ of 2-isogenies of abelian surfaces from $E_{\mathrm{com}} \times E_{\mathrm{chl}}$, applied to the pairs of copies $(0, 2)$ and $(1, 3)$ [Dar24, Lemmas 23 and 28]; step $m+1$, the gluing, then joins the two resulting surfaces into a fourfold. The dimension-2 steps compute in the product of the theta structures that φ induces on its codomain, whereas the dimension-4 steps that follow need the structure induced by the basis $B_{h,m}$ of Definition 2, made of the images of the columns of the verifier's starting matrix. Before the gluing, the verifier therefore changes

coordinates by Dartois's change-of-coordinates theorem [Dar24, Theorem 12], applied to the matrix N that expresses $B_{h,m}$ in the product basis. Dartois gives N in closed form for the first half-chain, computed via Weil pairings [Dar24, Lemma 30], and its analog for the second, from a dimension-2 basis given only modulo 4 [Dar24, Lemma 31 and Appendix C.1].

Proposition 6 (The change of basis at the gluing)**.** *Fix $h \in \{1,2\}$, put $\epsilon = 1$ if $h = 1$ and $\epsilon = -1$ if $h = 2$, let m be as in Lemma 12, and write $\kappa = 2^{m+1}$. By [Dar24, Lemmas 23 and 28], $\Phi_{h,m}(x_0,x_1,x_2,x_3) = (\varphi(x_0,x_2),\varphi(x_1,x_3))$ for a chain φ of m 2-isogenies of abelian surfaces from $E_{\mathrm{com}} \times E_{\mathrm{chl}}$, with kernel $\{(a_1x, \epsilon\tau(x)) : x \in E_{\mathrm{com}}[2^m]\}$. Let $R' = 2^{r-m-2}R$, $S' = 2^{r-m-2}S$, $\tilde{B}_0 = ((R',0),(0,\tau R'),(S',0),(0,\lambda\tau S'))$, and $\tilde{B} = (s_1,s_2;t_1,t_2) = \tilde{B}_0\tilde{M}$, where the columns of $\tilde{M}$ are, modulo 2^{m+2},*

$$(0,\eta,\kappa,\xi), \quad (\mu,\kappa,0,-\mu a_2), \quad (a_1,\epsilon,-\epsilon a_2,0), \quad (\epsilon a_2,0,a_1,\epsilon q),$$

with $\eta = \kappa a_2 q^{-1}$, $\xi = -\epsilon(1+\kappa a_1)$ and $\mu = (1-\epsilon\kappa q)a_1^{-1}$.

(i) $\tilde{B}$ is a symplectic basis of $(E_{\mathrm{com}} \times E_{\mathrm{chl}})[2^{m+2}]$, $\ker\varphi = \langle 4t_1, 4t_2\rangle$, and $2^m\tilde{B}$ is the basis of $(E_{\mathrm{com}} \times E_{\mathrm{chl}})[4]$ in which the algorithm starts the dimension-2 chain [Dar24, Lemmas 24 and 25, Appendix C.1]. After the dimension-2 steps, the structure on $A_{h,m}$ is therefore the product structure induced by $C \times C$, where

$$C = \big(2^m\varphi(s_1), 2^m\varphi(s_2); \varphi(t_1), \varphi(t_2)\big),$$

in the order of [Dar24, Appendix C.2.2].

(ii) The change of basis from $C \times C$ to $B_{h,m}$ is the matrix whose column j is obtained from column v of M_h as follows. Let $z = (v_0, v_2, v_4, v_6)$, $z' = (v_1, v_3, v_5, v_7)$, and $u = \tilde{M}^{-1}z$, $u' = \tilde{M}^{-1}z'$ modulo 2^{m+2}, that is,

$$\begin{aligned} u_1 &= -\epsilon a_2 z_0 - a_1 z_2 - \epsilon z_3, & u_2 &= a_1 z_0 + \epsilon q z_1 - \epsilon a_2 z_2, \\ u_3 &= -\kappa z_0 - \xi z_1 + \eta z_3, & u_4 &= \mu a_2 z_1 + \mu z_2 + \kappa z_3, \end{aligned}$$

and likewise u' from z'. A completion column ($j < 4$) and a kernel-lift column ($j \geq 4$) are, respectively,

$$\begin{aligned} &(u_1, u_2, u_1', u_2', 2^m u_3, 2^m u_4, 2^m u_3', 2^m u_4') \bmod 4, \\ &(u_1/2^m, u_2/2^m, u_1'/2^m, u_2'/2^m, u_3, u_4, u_3', u_4') \bmod 4, \end{aligned}$$

the divisions being exact. For $h = 1$ this is [Dar24, Lemma 30].

Proposition 6 makes this precise. Part (i) checks that $\tilde{B}$ is symplectic, adapted to the kernel and congruent modulo 4 to Dartois's bases, so that it is the basis in which the dimension-2 steps compute; of the six pairings of its vectors, four are units or zero by inspection, and the other two reduce to the norm equation $a_1^2 + a_2^2 + q = 2^e$. Part (ii) expresses each column of the starting matrix in $\tilde{B}$: since $\tilde{M}$ is symplectic, its inverse is obtained by transposing and swapping its 2×2 blocks, which gives the linear forms $u_1, \ldots, u_4$, and the division by 2^m in the kernel-lift columns is exact because four times a kernel lift lies in $\ker\varphi$.

For the first half-chain, reducing these formulas modulo 4 gives Dartois's Lemma 30, which the proposition thus derives; for the second, it supplies the derivation of the analog. Two consequences matter here. The structure of Definition 2 is justified on every fourfold from the gluing on, which Theorem 2 needs in order to show that D acts there as the permutation π. And the formulas that a verifier applies can be compared, term by term, with the closed form of the proposition, which Appendix B does for our implementation. The proof is in Appendix A.

7 Parameters and wire format

This section gives the engineering choices on which our implementation and the measurements of Section 8 rest: the parameters at the round-3 primes (Section 7.1) and the wire format (Section 7.2).

7.1 Parameters for the round-3 primes

SQIsignHD's exponent e bounds the response degree, $q < 2^e$. Every ideal class contains an ideal of norm at most about $0.90\sqrt{p}$ [DLRW24, Lemma 12], so 2^e must exceed $\sqrt{p}$, and the paper takes "a few bits over $\sqrt{p}$" as sufficient on heuristic grounds (their Section 4.2). The library's rule $e = \lambda + \lceil \log_2 2\lambda \rceil$ is phrased in terms of the security parameter because it was set for primes with $p \approx 2^{2\lambda}$, where it amounts to $\log_2 \sqrt{p}$ plus 10 to 13 bits. The round-3 level-I prime, however, has 70 bits more than $2^{2\lambda}$, and there the same formula gives $2^e = 2^{136}$, below $2^{162.6}$, the bound under which [DLRW24, Lemma 12] guarantees an ideal in every class: no response would exist. What transfers is the content of the rule, $\sqrt{p}$ plus about ten bits.

We therefore take $e = \lceil (\log_2 p)/2 \rceil + 11 = 174/264/346$ and, by the library's rule for the torsion the two half-chains need, $r = \lceil e/2 \rceil + 2 = 89/134/175$ (Table 1). The margin of 11 bits lies within the 10 to 13 bits that the library's rule gives on the primes it was written for, while SQIsignHD's own prime uses 15.3. Two checks bound it from below. First, the number of ideal vectors of norm below 2^e grows as $2^{2\cdot\text{margin}}$: about $2\pi^2 2^{2e}/p \approx 2^{26.7}$ vectors at level I, against one good norm in a few hundred, so for a random class a good q exists with overwhelming probability. Second, Nakagawa and Onuki's conservative countermeasure for PRISM-id, whose response reveals an isogeny of large prime degree, is a degree of at least $4\lambda/3$ bits [NO25, Section 5.3]; transferred to e, that is 170.7/256.0/341.3 against 174/264/346, cleared at every level, whereas a margin of 7 bits would fail it at level I.

The exponent e also governs the signer's search for a good degree. The signer samples vectors of the response ideal below 2^e and keeps the first whose norm q is odd, congruent to 3 modulo 4, and has $2^e - q$ a probable prime (40 Miller–Rabin rounds), and for which the verifier's Cornacchia step succeeds with $m \le 40$ (Appendix B). The sampled norms are odd, so one candidate in two passes the congruence; if $2^e - q$ behaves as a random e-bit integer, the density of good q is then $1/(e \ln 2)$: one candidate in 121 at level I, of which 60 reach a primality test. Section 8 measures the loop.

That count assumes the norm form of the response class to behave generically, and some classes do not. If the commitment and challenge curves are both the base curve $E_0 \colon y^2 = x^3 + x$, the response is an endomorphism of E_0. In the maximal order $\mathcal{O}_0 = \mathbb{Z}\langle 1, i, (i+j)/2, (1+k)/2 \rangle$ of the specification, with $k = ij = -ji$, an element $a+bi+cj+dk$ has reduced norm $a^2 + b^2 + p(c^2 + d^2)$ with $c, d \in \frac{1}{2}\mathbb{Z}$, so a nonzero c or d gives a norm of at least $p/4$, which exceeds 2^e at every level. Every response of degree below 2^e therefore lies in $\mathbb{Z}[i]$, and its degree $a^2 + b^2$ is never congruent to 3 modulo 4: this class has no good degree at all.

Such degenerate classes are rare, and we can estimate how rare. The degrees of the responses in a class are the values of its norm form $q_I(x) = \text{nrd}(x)/\text{nrd}(I)$, a positive definite quaternary form of determinant $p^2/16$. Let $N_1 \le N_2 \le N_3 \le N_4$ be its successive minima, L_2 the sublattice spanned by the first two, and $X = 2^e$. By Minkowski's second theorem, $N_1 N_2 N_3 N_4 \le 64p^2/\pi^4$. Suppose that $N_3 \le X/2$. Each translate of L_2 by a vector whose projection orthogonal to L_2 has norm at most $X/2$ then contains at least $0.13\, X/\sqrt{N_1 N_2}$ vectors of norm at most X, since the covering radius of L_2 is at most $\sqrt{(N_1 + N_2)/4} \le \sqrt{X/4}$; counting such translates, separately for $N_4 \le X/2$ and for $N_4 > X/2$, and bounding $N_1 N_2$ by Minkowski's theorem, gives at least about $0.07\, X^2/p$ vectors of norm at most X outside L_2: $2^{18.6}$, $2^{19.4}$ and $2^{20.1}$ at levels I, III and V. The

norm form of a maximal order has no local obstruction at 2, so if about half of these norms are odd and the odd ones behave like random odd integers, about $2^{9.7}$ of them are good at level I, counting v and $-v$, which have the same norm, once, and the chance that none is is about $\exp(-2^{9.7})$. A degenerate class therefore essentially requires $N_3 > X/2$, in which case all short vectors lie in L_2, as for $\mathcal{O}_0$ above, where $N_1 = N_2 = 1$ and $N_3 \geq p/4$.

To estimate the probability of $N_3 > X/2$, we instrumented our signer to record the Gram matrix of the response lattice of each fresh commitment and computed its successive minima for 4 180 commitments at level I, those of the 100 signatures of Table 7 and of 4 080 further signatures of the same key. Normalized by $\sqrt{p}/2$, the fourth root of the determinant, the third minimum has median $2^{0.39}$ and maximum $2^{3.65}$, that is, $2^{-8.6}X$, and its tail follows $\Pr[N_3 > y\sqrt{p}/2] \approx 1.5\, y^{-4}$ over $1.4 \leq y \leq 4.8$, the range in which the sample has between 1 588 and 6 points: the product $y^4 \Pr$ has median 1.53 there and stays between 1.3 and 1.6 while at least 50 points remain, falling to 0.8 at the last 6. This is the behavior expected of a random lattice. Scale the form to determinant 1, so that $\nu_i = N_i/(\sqrt{p}/2)$ are its successive minima. For a lattice drawn from the invariant probability measure on lattices of determinant 1, Siegel's mean value theorem [Sie45] gives $\Pr[\nu_1 \leq x] \leq \pi^2 x^2/4 \approx 2.47\, x^2$, nearly with equality for small x; our sample gives between $1.8\, x^2$ and $3.0\, x^2$ over $2^{-5} \leq x \leq 2^{-1}$, within sampling error. For the third minimum, Minkowski's theorem gives $\nu_1\nu_2 < 1024/(\pi^4 y^2)$ when $\nu_3 > y$, so the primitive rank-2 sublattice containing the first two minimal vectors has determinant less than $32/(\pi^2 y)$. Kim's extension of Siegel's theorem to sublattices [Kim22, Theorem 3] states that the expected number of primitive rank-2 sublattices of determinant at most H of such a random lattice in dimension 4 is $a(4,2)\, H^4$, with $a(4,2) = \frac{3}{2} \cdot \frac{V(4)}{V(1)\zeta(4)} \cdot \frac{V(3)\zeta(2)}{V(2)\zeta(3)} \approx 6.24$, where $V(i)$ is the volume of the unit ball in $\mathbb{R}^i$. Markov's inequality then gives $\Pr[\nu_3 > y] \leq 6.24\,(32/\pi^2)^4\, y^{-4} \approx 690\, y^{-4}$: the exponent we measure, with a constant that reflects the slack in Minkowski's bound. Response lattices are ideal lattices of a quaternion order rather than random lattices, and that they follow the same statistics is a heuristic; for ideal lattices of cyclotomic fields, the analog of Siegel's theorem holds nearly exactly [GV24]. Extrapolating to $y = X/\sqrt{p}$, that is $2^{11.2}$, $2^{11.6}$ and $2^{12.0}$, a random commitment lands in a degenerate class with probability about 2^{-44}, 2^{-46} and 2^{-47} at levels I, III and V, and the bound for random lattices gives at most 2^{-35}, 2^{-37} and 2^{-38}. Both figures rest on the assumption that response lattices are distributed like random lattices of the same determinant, which our measurements support but do not prove.

Our signer gives up on a commitment after 4096 samples without a good degree and starts over with a fresh one, and it stops with an error after 64 commitments. A commitment is given up when its class is degenerate, with probability about 2^{-44} at level I by the estimate above, or when 4096 samples of a class that is not degenerate contain no good degree, with probability $(1 - 1/109)^{4096} \approx 2^{-54}$ at the measured density of one good degree in 109 samples (Table 7). A restart thus happens in fewer than one signature in 2^{40}, and the error, which takes 64 such commitments in a row, with probability about 2^{-2800}. The restart changes the distribution of the published commitments by at most the same probability in statistical distance.

7.2 The wire format

A dimension-4 signature carries the commitment curve, the degree q and three of the four entries of the transmitted action, with two hint bytes (Section 4.1). Which entry is dropped is less obvious than it seems. Write the transmitted action as $M = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$, so that $ad - bc \equiv kq \pmod{2^r}$. The verifier rejects unless both bases pair to roots of unity of exact order 2^r (Section 5.6), so k is odd; q is odd too, so $\det M$ is odd. Recovering the dropped entry means inverting another entry modulo 2^r, which must therefore be odd, and no single entry is odd on every signature. We therefore use the following rule. If a is

Table 1: Dimension-4 parameters at the round-3 primes. All three levels are built and measured.

	NIST I	NIST III	NIST V
p[a]	$3 \cdot 2^{324} - 1$	$27 \cdot 2^{500} - 1$	$17 \cdot 2^{664} - 1$
$\log_2 p$[a]	325.58	504.75	668.09
λ, challenge bits[a]	128	192	256
e, response bound $q < 2^{e}$[b]	174	264	346
$r = \lceil e/2 \rceil + 2$, transmitted torsion bits[c]	89	134	175
half-chain steps $\lceil e/2 \rceil$[c]	87	132	173
public key, bytes, predicted[d]	84	130	170
signature, bytes, predicted[d]	139	212	278
public key, bytes, measured	84	130	170
signature, bytes, measured	139	212	278
round-3 standard signature, bytes[a]	200	306	406

[a] The SQIsign round-3 specification [AAA+26].
[b] Ours: $e = \lceil (\log_2 p)/2 \rceil + 11$ (Section 7.1).
[c] SQIsignHD's rule for the torsion the half-chains need [DLRW24, Section 4.3].
[d] Signature: $2\lceil \log_2 p \rceil + (e-2) + 3r + 16$ bits, padded once (Section 7.2); 137 / 210 / 276 without the hints. Public key: the curve's two coordinates in whole bytes and two hint bytes.

odd, the signer drops d and the verifier computes $d \equiv (kq + bc)\, a^{-1} \pmod{2^r}$; if a is even, then $bc \equiv -kq$ is odd, so b and c are both odd, and the signer drops c, which the verifier computes as $c \equiv (ad - kq)\, b^{-1} \pmod{2^r}$. The verifier reads the case from the parity of a, so the rule costs no bit. The relation fixes r bits of the matrix and the format spends them on one entry; sending all four entries and only the top $e - r$ bits of q, whose low r bits the relation would then determine, would cost two bits more, since the four entries would then carry the two bits of $ad - bc \equiv 3k \pmod 4$ that $q \equiv 3 \pmod 4$ fixes in advance, so the pairings have nothing more to give.

The format packs its fields at bit granularity, which saves a few bytes over rounding each field up to whole bytes: at level I each scalar needs 89 bits, not 12 bytes, and q needs 172 bits, since $q \equiv 3 \pmod 4$ makes its two low bits known in advance. A signature is the commitment curve's Montgomery coefficient as two $\lceil \log_2 p \rceil$-bit integers, $(q-3)/4$ in $e-2$ bits, the three scalars in r bits each and the two hints in 8 bits each, least significant bit first, padded with zeros to a whole byte: $2\lceil \log_2 p \rceil + (e-2) + 3r + 16$ bits, that is, 1107, 1690 and 2223 bits, or 139, 212 and 278 bytes at levels I, III and V (Table 1), against 142, 214 and 280 with whole bytes. The decoder rejects a coordinate not below p and a nonzero padding bit, and the verifier rejects hints that are not the canonical ones of their curves, the first entries of the tables that give points on the curve, which are those the signer sends. Without this check any other hint that gives a basis would do: anyone could re-express the transmitted action in that basis, by discrete logarithms of order 2^r, and obtain a second valid encoding of the same signature. With the check switched off, 97 of the 399 other pairs of table entries gave such an encoding per signature on average at level I, over the 100 signatures of Table 7 (median 95, between 39 and 224), and the verifier rejected every one of them with the check on. With it, the hints leave a signature a single encoding (the top bit of a scalar, discussed below, is a separate matter), and unpacking costs a few hundred cycles, which is negligible compared to a verification. Since the canonical hints are a deterministic function of their curves, and the verifier already computes them to check them, the signature could omit them at no further cost to the verifier, for 137, 210 and 276 bytes; we keep them so that the library's verifier, which reads them, checks our signatures unmodified.

The compact format is not strongly unforgeable (SUF-CMA). The verifier accepts

any isogeny of degree q from the commitment curve to the challenge curve, given by its action on the torsion. For instance, if σ verifies, so does $-\sigma$, whose action is $-M$, and nothing in the format excludes it, so anyone can turn a signature into a second valid one on the same message. The round-3 specification discusses strong unforgeability only for its dimension-2 format, where the auxiliary isogeny can be replaced by another of the same degree [AAA+26, Section 8.4.2]; it says nothing about dimension 4. A second, weaker malleability concerns the top bit of each transmitted scalar. Changing it changes the transmitted action only by a map into 2-torsion, which moves only the last kernel point of each half-chain, and verification does not always detect it: at level I, flipping that bit of a or b gave a valid signature in 2 of 12 trials, while no other bit ever did. In such a case the verified isogeny is still σ.

8 Experimental results

This section measures what the compact format saves and what it costs. We describe the setup (Section 8.1), report sizes, running times, the signer's search for a good degree and memory use (Section 8.2), and state the scope of these measurements (Section 8.3).

8.1 Setup

All measurements come from one session on one machine, a virtual machine with 32 vCPUs of an Intel Xeon at 2.80 GHz and 126 GiB of memory (Google Compute Engine, `n2-standard-32`), with the measuring process pinned to one core whose sibling was kept idle, on 2026-11-15. One binary measures the Rust code, compiled by `rustc` 1.100 in release mode with fat link-time optimization and no target-CPU flag; the code chooses its assembly field kernels at run time from CPUID. Each operation is timed with `rdtsc` and the wall clock around the call. The tables report medians: for dimension 2, over 52 key generations, 52 signatures and 200 verifications at each level; for the compact format at level I, over 52 key generations, 100 signatures and 200 verifications of five signatures in each mode; at levels III and V, over 20 key generations and 52 signatures, with verification as the mean over six signatures of the median of 15 runs. The C reference is the-sqisign at tag `nist-v3`, built with CMake in `Release` mode and `-DSQISIGN_BUILD_TYPE=broadwell`, the assembly-backed field arithmetic, and measured by its own benchmark (`--iterations=300` at level I, 100 at III and V), medians. Every row was measured in three passes of the session: key generation and signing of the C reference at levels III and V, whose benchmark draws a fresh seed on every run, differed by more than 1.5 % between the first two passes, so a third was run, and the tables report the median of the three passes' values; no row of the Rust code moved by more than 1.2 %. Verification is measured from bytes: decode the key, decode the signature, verify. The timings were taken with the bit-packed layout, and include the verifier's checks of the hints and of the orders of the pairings (Section 7.2) and its goodness test, which together cost 1.2, 3.2 and 5.5 megacycles per verification at levels I, III and V, under 2 % of a four-point verification and under 3 % of a two-point one. The verifier's tests of its divisors (Remark 1(j)) were added after this session; one further pass on the same core with them moved no verification row by more than 0.6 %, and the tables are those of the session.

8.2 Results

We begin with the sizes on the wire (Table 2). At level I the compact key and signature take 223 bytes, against 283 for the specification's format: a saving of 30.5 % on the signature, of which bit packing accounts for 1.5 points. The compressed dimension-2 format, with a 176-byte signature, sits between them.

Table 2: Public key + signature = total, in bytes, at the round-3 primes.

format	NIST I	NIST III	NIST V
standard (specification)	$83 + 200 = 283$	$129 + 306 = 435$	$169 + 406 = 575$
compressed (this crate)	$83 + 176 = 259$	$129 + 269 = 398$	$169 + 353 = 522$
compact, dim. 4 (this crate)	$84 + 139 = 223$	$130 + 212 = 342$	$170 + 278 = 448$

Table 3: Level I: signature bytes, and the bytes left for all other data under a budget of 250 and of 300 bytes for the signature and the data together. The budgets are the range reported in [SQI26a]; the signature sizes are those of Table 2 and the remainders are subtractions.

format	signature	left of 250	left of 300
standard (specification)	200	50	100
compressed, dimension 2	176	74	124
compact, dimension 4	139	111	161

Table 3 sets the three level-I signatures against the byte budgets reported for barcodes [SQI26a].

These savings come at a cost in time, which Table 4 gives in megacycles. At level I the dimension-4 verifier costs 6.8 times the dimension-2 verifier of the same crate, and dimension-4 signing costs 3.6 times dimension-2 signing, because the good degree is found by rejection. At levels III and V the dimension-4 verifier costs 6.5 and 6.0 times the dimension-2 one, the two-point verifier 4.0 and 3.6 times, and signing 2.4 and 2.2 times as much in median. Signing times vary widely at these levels, because of the search for a good degree: the middle half of the signatures took 398 to 1471 megacycles at level III and 869 to 3392 at level V. Key generation is the same in both formats up to the basis convention (35.3 against 32.9 megacycles). As a check on the implementation, the crate's dimension-2 code runs at 1.02 times the reference's assembly build for verification, 1.06 times for key generation and 1.09 times for signing, and the compressed format's recovery adds 19 % to a verification. Figure 2 sets these costs against signature size at level I: the compact format has the smallest signature and the costliest verification, and the compressed format lies between the other two on both counts.

Table 5 gives the corresponding wall-clock times at level I. On this machine the two-point verifier takes a level-I verification from 27.3 ms to 17.2 ms. The machine is not what a deployment runs; the ratios between the rows are the numbers to carry.

The same harness was also run on an Apple M4 Pro, at the commit the tables were measured at [Ray26b], with the crate's portable field arithmetic, since its assembly kernels are for x86-64, and with the wall clock alone (Table 6). These times are not comparable in absolute terms with those of the assembly build above; the ratios are the comparison. There the two-point verifier takes 0.62, 0.60 and 0.60 times the four-point one at levels I, III and V, and 4.1, 4.0 and 3.8 times the dimension-2 verifier of the same build, against 6.7, 6.6 and 6.3 with four points. The values are those of the first of two passes; the second differs by at most 1.2 % on these rows.

The higher signing cost is entirely due to the search for a good degree. Table 7 compares this search over the session's signatures with the expectation of Section 7.1. A candidate costs one norm computation; one in two reaches a Miller–Rabin test on an e-bit number, and composites fail at the first round. The loop also accounts for the spread of signing times: the median signature needs 95 samples, the mean 109, and the range is 1 to 446.

Finally, at its peak, one compact verification at level I allocates 1655 KiB (1.62 MiB) of heap with four points and 1631 KiB with two, measured with a counting allocator around the call, and 3443 and 6160 KiB with four points at levels III and V; the dimension-2 verifier allocates nothing.

Table 4: Megacycles at the round-3 primes, medians of the session. The last column is the dimension-4 verifier over the dimension-2 verifier of the same code. The C reference is the `broadwell` build. Run counts as in Section 8.1.

implementation	level	keygen	sign	verify	× dim. 2
C reference, dimension 2	I	31.2	92.0	11.0	
C reference, dimension 2	III	106.2	323.1	29.0	
C reference, dimension 2	V	216.9	679.6	75.3	
Rust, dimension 2	I	32.9	100.0	11.2	
Rust, dimension 2	III	109.0	375.4	29.8	
Rust, dimension 2	V	248.6	769.7	80.0	
Rust, dimension 2, compressed verify	I			13.3	
Rust, dimension 4 (compact)	I	35.3	360.5	76.5	6.8
two points	I			48.0	4.3
Rust, dimension 4 (compact)	III	126.3	913.9	194.6	6.5
two points	III			120.1	4.0
Rust, dimension 4 (compact)	V	249.8	1714.5	482.2	6.0
two points	V			291.6	3.6

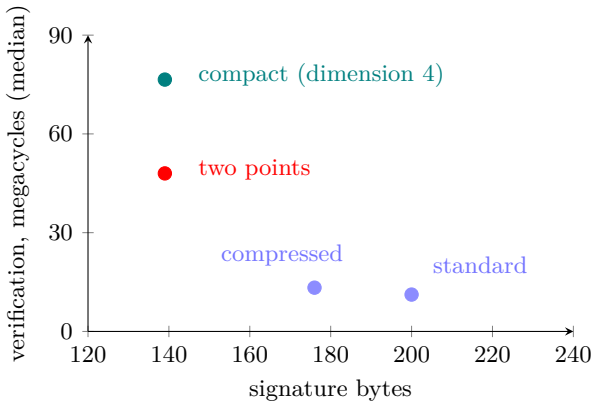


Figure 2: The trade at level I on the round-3 prime: signature bytes against verification cost for the three formats of one crate, from Tables 2 and 4; the red point is the two-point verifier.

8.3 Scope and reproducibility

These measurements come from one implementation; there is no independent round-3 dimension-4 signer to compare against. The SQIsignHD authors' verifier, re-parameterized to our level-I prime, accepts our signatures and rejects tampered ones [DLRW25]. At levels III and V, the non-residue tables of the basis-from-hint convention follow a documented deterministic rule of our own rather than the library's seeded one, so such a cross-check there would need the library's tables. The two-point verifier is not yet the default; it checks the pairing of the kernel generators at the gluing and falls back to four points otherwise, and on 24 signatures at level I and 5 each at levels III and V it reproduces every codomain and every verdict of the four-point verifier, on honest and tampered signatures alike (Appendix B).

Constant-time execution was not attempted. For the verifier this is no weakness in the usual setting, since it handles only public data, so its timing reveals nothing secret; the exception is an application in which the message or the signer's identity must itself stay confidential. SQIsign signing is variable-time by design in the reference as well. The

Table 5: Wall-clock milliseconds at level I on the round-3 prime, medians, on the machine of Section 8.1.

format	operation	ms
dimension 2 (standard)	key generation	11.8
	signing	35.7
	verification	4.0
	verification, compressed	4.7
dimension 4 (compact)	key generation	12.6
	signing	128.8
	verification	27.3
	verification, two points	17.2

Table 6: Verification in wall-clock milliseconds on an Apple M4 Pro (macOS, `rustc` 1.99, portable field arithmetic, commit 68a0017), medians of the first of two passes, with the run counts of Section 8.1. The last two columns are the multiple of the dimension-2 verifier of the same build.

level	four points	two points	two over four	four, × dim. 2	two, × dim. 2
I	14.7	9.1	0.62	6.7	4.1
III	42.2	25.4	0.60	6.6	4.0
V	77.8	46.4	0.60	6.3	3.8

implementation is experimental and not for production.

9 Conclusion

In this paper we proposed two-point verification for SQIsignHD's dimension-4 verifier, which halves the work spent on kernel points by exploiting an automorphism of order 4 of the verifier's chain, and proved that the automorphism acts as one fixed permutation of the theta coordinates for every input, that the null points along the chain are fixed by it, and that the change of basis at the gluing expresses the structure the chain needs (Section 5 and Appendix A). We also proved that the verifier's image check is sound with the signs left free (Proposition 1), and showed that the verifier must check the basis hints and the orders of the pairings, without which a forger could re-encode signatures and drive the chain outside the hypotheses of these proofs (Section 7.2). We implemented the scheme at the round-3 SQIsign primes, restating its response bound in terms of $\sqrt{p}$ (Section 7.1). At level I a signature takes 139 bytes, against 200 for the specification's format, and key and signature together take 223 bytes, the smallest level-I combination we are aware of at parameters that stand after Wesolowski's attack. The price is verification: a dimension-4 verification costs 4.3, 4.0 and 3.6 times a dimension-2 one in the same code at levels I, III and V with two points, and 6.8, 6.5 and 6.0 times with four, and signing costs 3.6, 2.4 and 2.2 times as much in median, because of the search for a good degree; the two-point verifier takes 0.63, 0.62 and 0.60 times the cycles of the four-point one (Section 8). The compact format therefore suits settings where bandwidth or storage dominates and a verification of a few tens of milliseconds at level I (Table 5), and of 43 to 172 milliseconds at levels III and V on the same machine, is acceptable, while the specification's dimension-2 format remains the better general-purpose choice. Under the 250-byte budget reported for barcodes [SQI26a], the compact signature at level I leaves 111 bytes for data where the standard format leaves 50 (Table 3), at 4.3 times the verification cost of dimension 2 with two points (Table 4).

Apart from the image check, the security argument is SQIsignHD's with our parameters

Table 7: The compact signer's good-degree loop at level I: measured over the session's signatures against the density $1/(e \ln 2)$ for odd q (and the figure for a candidate of random parity, which the sampler does not produce).

	measured	expected
signatures measured	100	
response samples per signature, mean	109	121
the same if q had random parity	—	241
response samples per signature, median (min, max)	95 (1, 446)	
primality tests per signature, mean	55	60
signing, Mcycles, median (min, max)	361 (46, 1529)	
signing, dimension 2, Mcycles, median	100	

substituted and has not been re-examined. That every class has a good degree is a heuristic in their paper and remains one here, but we estimate how often it fails: a random commitment lands in a class with no good degree with probability about 2^{-44} at level I, under the assumption that response lattices behave like random lattices (Section 7.1). Like the specification's format, the compact one is not strongly unforgeable (Section 7.2).

The main bottleneck remains the dimension-4 arithmetic itself, and narrowing it decides whether the 61 bytes saved at level I are worth having; there is reason to expect room. On the core of Section 8.1, MIKE's Rust implementation [BDD+26] computes a level-I shared secret, a chain of 228 steps, in 20.9 megacycles (median of 200 runs), about 0.09 megacycles per step, against about 0.28 for ours (48.0 megacycles over $2 \cdot 87$ steps, with two points). The comparison is loose, since MIKE works over $\mathbb{F}_p$ at a prime of 318 bits, in constant time, and our verifier also evaluates a point for its image check, but it suggests that much of our fourfold penalty lies in the implementation rather than in the dimension. Doubling on the quotient by the automorphism would save more, but it needs efficient image formulas there, which are not known (Appendix C); and MIKE's observation that additions, not multiplications, are a significant share of the cost of dimension-4 theta arithmetic [BDD+26, Section 6] points toward field representations and vectorization tuned to the Hadamard transform.

Finally, every construction in this paper is young, ours included, and the numbers describe code as it stood at the time of writing, not a settled design.

AI disclosure

While the authors observed the opportunities for improving 4-dimensional SQIsign (notably the symmetry of the chain) on their own by reviewing the literature on the subject and designing exploratory tests, the final software prototype and test suites were developed on top of existing implementations of the signature scheme with the assistance of Anthropic's Claude Opus 5.5, which also drafted parts of the quantitative analysis and helped in carefully checking the proofs and references. The authors have reviewed the material and take full responsibility for the programs, the tests and their results, the measurements, and every claim in the paper.

Acknowledgments

The authors are grateful to Steven Galbraith and Jake Doliskani for their invaluable comments and feedback while this paper was being prepared.

References

[AAA+25] Marius A. Aardal, Gora Adj, Diego F. Aranha, Andrea Basso, Isaac Andrés Canales Martínez, Jorge Chávez-Saab, Maria Corte-Real Santos, Pierrick Dartois, Luca De Feo, Max Duparc, Jonathan Komada Eriksen, Tako Boris Fouotsa, Décio Luiz Gazzoni Filho, Basil Hess, David Kohel, Antonin Leroux, Patrick Longa, Luciano Maino, Michael Meyer, Kohei Nakagawa, Hiroshi Onuki, Lorenz Panny, Sikhar Patranabis, Christophe Petit, Giacomo Pope, Krijn Reijnders, Damien Robert, Francisco Rodríguez-Henríquez, Sina Schaeffler, and Benjamin Wesolowski. SQIsign – algorithm specifications and supporting documentation, version 2.0. NIST PQC Digital Signatures Round 2 Submission, https://sqisign.org/spec/sqisign-20250205.pdf, 2025.

[AAA+26] Marius A. Aardal, Gora Adj, Diego F. Aranha, Andrea Basso, Giacomo Borin, Isaac Andrés Canales Martínez, Jorge Chávez-Saab, Maria Corte-Real Santos, Pierrick Dartois, Luca De Feo, Max Duparc, Thomas Espitau, Jonathan Komada Eriksen, Tako Boris Fouotsa, Décio Luiz Gazzoni Filho, Basil Hess, Riccardo Invernizzi, David Kohel, Antonin Leroux, Patrick Longa, Luciano Maino, Michael Meyer, Marzio Mula, Kohei Nakagawa, Hiroshi Onuki, Lorenz Panny, Sikhar Patranabis, Christophe Petit, Giacomo Pope, Krijn Reijnders, Damien Robert, Francisco Rodríguez-Henríquez, Sina Schaeffler, Frederik Vercauteren, Alexandre Wallet, Benjamin Wesolowski, and Wessel van Woerden. SQIsign – algorithm specifications and supporting documentation, version 3.0. NIST PQC Digital Signatures Round 3 Submission, https://sqisign.org/spec/sqisign-20260901.pdf, 2026.

[BDD+26] Andrea Basso, Pierrick Dartois, Max Duparc, Jonathan Komada Eriksen, Sabrina Kunzweiler, Michael Meyer, Giacomo Pope, Krijn Reijnders, Damien Robert, Ryan Rueger, and Sina Schaeffler. MIKE: A fast and compact post-quantum NIKE. Cryptology ePrint Archive, Paper 2026/2377, https://eprint.iacr.org/2026/2377, 2026.

[BDF+24] Andrea Basso, Pierrick Dartois, Luca De Feo, Antonin Leroux, Luciano Maino, Giacomo Pope, Damien Robert, and Benjamin Wesolowski. SQIsign2D-West: The fast, the small, and the safer. In *Advances in Cryptology – ASIACRYPT 2024*, volume 15486 of *LNCS*, pages 339–370. Springer, 2024. ePrint 2024/760.

[CD23] Wouter Castryck and Thomas Decru. An efficient key recovery attack on SIDH. In *Advances in Cryptology – EUROCRYPT 2023*, volume 14008 of *LNCS*, pages 423–447. Springer, 2023.

[Dar24] Pierrick Dartois. Fast computation of 2-isogenies in dimension 4 and cryptographic applications. Cryptology ePrint Archive, Paper 2024/1180, https://eprint.iacr.org/2024/1180, 2024.

[Dar25] Pierrick Dartois. Theta dim4: SageMath library for 2-isogenies in dimension 4 in the theta model. https://github.com/Pierrick-Dartois/Theta_dim4, 2025.

[DLRW24] Pierrick Dartois, Antonin Leroux, Damien Robert, and Benjamin Wesolowski. SQIsignHD: New dimensions in cryptography. In *Advances in Cryptology – EUROCRYPT 2024*, volume 14651 of *LNCS*, pages 3–32. Springer, 2024. ePrint 2023/436, revised 2024-09-20, https://eprint.iacr.org/2023/436.

[DLRW25] Pierrick Dartois, Antonin Leroux, Damien Robert, and Benjamin Wesolowski. SQISignHD library, version 2.0: A C signer and a SageMath verifier. Master at

commit 0287566, https://github.com/Pierrick-Dartois/SQISignHD-lib, 2025.

[Don69] Peter Donovan. The Lefschetz–Riemann–Roch formula. *Bulletin de la Société Mathématique de France*, 97:257–273, 1969.

[FKL+20] Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski. SQISign: Compact post-quantum signatures from quaternions and isogenies. In *Advances in Cryptology – ASIACRYPT 2020*, volume 12491 of *LNCS*, pages 64–93. Springer, 2020. ePrint 2020/1240.

[GV24] Nihar Gargava and Maryna Viazovska. Mean value for random ideal lattices. arXiv:2411.14973, 2024.

[Kan97] Ernst Kani. The number of curves of genus two with elliptic differentials. *J. Reine Angew. Math.*, 485:93–122, 1997.

[Kim22] Seungki Kim. Mean value formulas on sublattices and flags of the random lattice. *Journal of Number Theory*, 2022. arXiv:2005.10874. doi:10.1016/j.jnt.2022.03.013.

[MMP+23] Luciano Maino, Chloe Martindale, Lorenz Panny, Giacomo Pope, and Benjamin Wesolowski. A direct key recovery attack on SIDH. In *Advances in Cryptology – EUROCRYPT 2023*, volume 14008 of *LNCS*, pages 448–471. Springer, 2023.

[Mum66] David Mumford. On the equations defining abelian varieties. I. *Inventiones Mathematicae*, 1(4):287–354, 1966. doi:10.1007/BF01389737.

[Mum74] David Mumford. *Abelian Varieties*, volume 5 of *Tata Institute of Fundamental Research Studies in Mathematics*. Oxford University Press, second edition, 1974.

[NO25] Kohei Nakagawa and Hiroshi Onuki. Attacks on PRISM-id via torsion over small extension fields. Cryptology ePrint Archive, Paper 2025/1602, https://eprint.iacr.org/2025/1602, 2025.

[Ray26a] Dustin Ray. Dimension-4 SQIsign at round-3 parameters: Sizes and costs of the compact format. Cryptology ePrint Archive, Paper 2026/2221, https://eprint.iacr.org/2026/2221, 2026.

[Ray26b] Dustin Ray. sqisign-rs: SQIsign round 3 in pure Rust, with the compressed and the dimension-4 compact formats. https://github.com/anchorageoss/sqisign-rs, 2026. The dimension-4 code and the session records of this paper are at tag paper-dim4-2026-11, commit 3c97d46, on branch agent/dim4-two-point. The tables were measured at commit 68a0017; the verifier's tests of its divisors were added afterwards, at commit 6302b7a. The dimension-2 round-3 code is that of the 0.6 release: tag v0.6.34 at commit 078cc2a, the version bump on the 0.6 merge, commit 8718c97.

[Rob23] Damien Robert. Breaking SIDH in polynomial time. In *Advances in Cryptology – EUROCRYPT 2023*, volume 14008 of *LNCS*, pages 472–503. Springer, 2023.

[Rob24] Damien Robert. The module action for isogeny based cryptography. Cryptology ePrint Archive, Report 2024/1556, https://eprint.iacr.org/2024/1556, 2024.

[Sie45] Carl Ludwig Siegel. A mean value theorem in geometry of numbers. *Annals of Mathematics*, 46(2):340–347, 1945.

[SQI26a] Technical notes on signature compression techniques. Issue 19 of the SQIsign reference repository, opened 2026-07-08; comment of 2026-10-07 on signatures in barcodes. https://github.com/SQISign/the-sqisign/issues/19, 2026.

[SQI26b] the-sqisign: Reference implementation of SQIsign, round 3. The SQIsign team. Tag nist-v3, commit 6d017708, https://github.com/SQISign/the-sqisign, 2026.

[Wes26] Benjamin Wesolowski. The supersingular isogeny problem in time and memory $p^{1/3+o(1)}$. Cryptology ePrint Archive, Paper 2026/1486, https://eprint.iacr.org/2026/1486, 2026.

A Proofs for Sections 4, 5 and 6

This appendix proves the results of Sections 4, 5 and 6, in the order in which they are stated there, with the notation of Section 5.1.

Proof for Section 4.2

Proof of Proposition 1. The verifier takes $R = [2^{f-r}]R_0$ and $S = [2^{f-r}]S_0$, and accepts only if $e(R, S)$ has exact order 2^r; so R has order 2^r and R_0 has order 2^f.

Next, F is an endomorphism of A with $\widehat{F}F = 2^e$. Every input that reaches the half-chains satisfies the hypotheses of Lemma 8, so by Lemma 11 the starting matrices are symplectic, and the kernel of Φ_h, generated by the points $4x'_{h,j}$ of Definition 2, is isotropic of order 2^{4e_h}, hence maximal isotropic in $A[2^{e_h}]$: Φ_h is a 2^{e_h}-isogeny of principally polarized abelian varieties, with $\widehat{\Phi}_h\Phi_h = 2^{e_h}$ and $\Phi_h\widehat{\Phi}_h = 2^{e_h}$. The step formulas compute it whenever they divide by no zero value [Dar24], and the verifier rejects otherwise. Acceptance means that the two middle codomains have the same theta null point, that is, they are the same principally polarized abelian variety with the same theta structure [DLRW24, Section 4.5]. So $F = \widehat{\Phi}_2\Phi_1$ is an endomorphism of A, and $\widehat{F}F = \widehat{\Phi}_1\Phi_2\widehat{\Phi}_2\Phi_1 = \widehat{\Phi}_1\,2^{e_2}\,\Phi_1 = 2^{e_1+e_2} = 2^e$.

Write $F = (F_{ij})$ as in Section 5.1, so that $F_{i1} \in \mathrm{Hom}(E_{\mathrm{com}}, E_i)$ with $E_1 = E_2 = E_{\mathrm{com}}$ and $E_3 = E_4 = E_{\mathrm{chl}}$. Since $\widehat{F}$ is the transpose of the matrix of duals, the $(1,1)$ entry of $\widehat{F}F$ is $\sum_i \widehat{F}_{i1}F_{i1} = \sum_i \deg F_{i1}$, so $\sum_i \deg F_{i1} = 2^e$. The chain gives $F(R_0, 0, 0, 0) = (F_{11}R_0, F_{21}R_0, F_{31}R_0, F_{41}R_0)$ up to the sign of the Kummer variety, which the x-coordinates of the components do not see; the test therefore says that $F_{11}R_0 = \epsilon_1 a_1 R_0$ and $F_{21}R_0 = \epsilon_2 a_2 R_0$ for some $\epsilon_1, \epsilon_2 \in \{\pm 1\}$, and that $F_{41}R_0 = 0$.

The degree is a positive definite quadratic form on $\mathrm{End}(E_{\mathrm{com}})$, so $\sqrt{\deg}$ satisfies the triangle inequality. Hence $\psi_1 = F_{11} - \epsilon_1 a_1$ has $\deg\psi_1 \le (\sqrt{\deg F_{11}} + |a_1|)^2 < (2^{e/2} + 2^{e/2})^2 = 2^{e+2}$, since $\deg F_{11} \le 2^e$ and $a_1^2 < 2^e$. But ψ_1 vanishes on the cyclic group $\langle R_0\rangle$ of order 2^f, so if it were nonzero its degree would be at least 2^f; and $2^f \ge 2^{e+2}$ at every level, with $f = 324$, 500 and 664 against $e + 2 = 176$, 266 and 348. So $F_{11} = \epsilon_1 a_1$, and in the same way $F_{21} = \epsilon_2 a_2$. Finally $\deg F_{41} \le 2^e < 2^f$ and F_{41} vanishes on $\langle R_0\rangle$, so $F_{41} = 0$. Then $\deg F_{31} = 2^e - a_1^2 - a_2^2 = q$, which is positive, and $\psi = F_{31}$ is an isogeny of degree q from E_{com} to E_{chl}. □

The argument is that of [DLRW24, Appendix E.5]; only the degrees of the first two entries enter it, which is why their signs may be left free. It also shows why the check on the order of the pairings matters beyond Lemma 8: a hint giving a point R_0 of small order would make the image check nearly vacuous.

Proofs for Section 5.2

Proof of Lemma 1. (i) $J^2(P,Q) = J(Q,-P) = (-P,-Q)$. (ii) We have

$$(a_1 + a_2 J)(P,Q) = \big([a_1]P + [a_2]Q,\ -[a_2]P + [a_1]Q\big),$$

which is $\alpha(P,Q)$; the dual of an integer matrix is its transpose, and $J^\mathsf{T} = -J$. (iii) $(a_1 + a_2 J)(a_1 - a_2 J) = a_1^2 - a_2^2 J^2 = a_1^2 + a_2^2$, since J commutes with integers. (iv) $J^\mathsf{T} J = 1$ by direct computation, and an automorphism γ preserves the polarization exactly when $\widehat{\gamma}\gamma = 1$. □

Proofs for Section 5.3

Proof of Lemma 2. Every entry of F and $\widehat{F}$ is an integer matrix or a componentwise map $\mathrm{diag}(\psi,\psi)$, and these commute with each other: for an integer matrix M and any homomorphism ψ of curves, $\mathrm{diag}(\psi,\psi)M = M\,\mathrm{diag}(\psi,\psi)$, because ψ is a group homomorphism. The diagonal blocks of $\widehat{F}F$ are $\widehat{\alpha}\alpha + \widehat{\Sigma}\Sigma = (2^e - q) + q = 2^e$ and $\Sigma\widehat{\Sigma} + \alpha\widehat{\alpha} = 2^e$, by Lemma 1(iii) and $\widehat{\sigma}\sigma = \sigma\widehat{\sigma} = q$; the off-diagonal blocks are $\widehat{\alpha}\widehat{\Sigma} - \widehat{\Sigma}\widehat{\alpha} = 0$ and $\Sigma\alpha - \alpha\Sigma = 0$. The same holds for $F\widehat{F}$. Then $(\deg F)^2 = \deg[2^e] = 2^{8e}$ on a fourfold, so $\deg F = 2^{4e}$. Finally $\widehat{F}(A[2^e]) \subseteq \ker F$ because $F\widehat{F} = 2^e$, and both groups have 2^{4e} elements, since $\ker\widehat{F} \subseteq A[2^e]$ has 2^{4e} elements and $|A[2^e]| = 2^{8e}$. □

Proof of Proposition 2. Since $D = \mathrm{diag}(J,J)$, the identity $FD = DF$ holds block by block exactly when each block of F commutes with J. The blocks α and $\widehat{\alpha}$ are polynomials in J (Lemma 1(ii)), and Σ, $\widehat{\Sigma}$ are componentwise maps, which commute with every integer matrix (proof of Lemma 2). Explicitly, $J\Sigma(P,Q) = J(\sigma P, \sigma Q) = (\sigma Q, -\sigma P) = \Sigma(Q,-P) = \Sigma J(P,Q)$. If $F(x) = 0$ then $F(Dx) = DF(x) = 0$. The blocks of $\widehat{F}$ are of the same kinds, so $\widehat{F}D = D\widehat{F}$ as well. □

Proofs for Section 5.4

Proof of Proposition 3. D is a group automorphism with $D(K) = K$ (Proposition 2), hence $D([2^{e-k}]K) = [2^{e-k}]K$, so $F_k \circ D$ vanishes on $\ker F_k$ and factors as $D_k \circ F_k$; D_k is an automorphism because D is. The identities $D_k^2 = -1$ and $\widehat{D_k} = -D_k$ follow from $D^2 = -1$ and $\widehat{D} = -D$ (Lemma 1) by uniqueness of the factorization. The principal polarization of A_k is the one induced from 2^k times that of A through the isotropic kernel $\ker F_k$; D preserves both, so D_k preserves the induced one. □

Proof of Proposition 4. The verifier builds its kernel from the transmitted action rather than from σ: the transmitted matrix defines a map σ' on the 2^r-torsion (the map τ of Section 5.6, with the curves exchanged as in Section 5.3), and the verifier uses $\Sigma' = \mathrm{diag}(\sigma',\sigma')$ in place of Σ. Like Σ, the map Σ' acts componentwise, so it commutes with J, and the kernel generators $(\widehat{\alpha}(x), \Sigma'(x))$ satisfy $D(\widehat{\alpha}(x), \Sigma'(x)) = (\widehat{\alpha}(Jx), \Sigma'(Jx))$, which is again of that form. The proof of Proposition 3 uses only that the kernel is D-stable. □

Proofs for Section 5.5

Proof of Lemma 3. $\theta_{(i,j)}(Q,-P) = \theta_i(Q)\,\theta_j(-P) = \theta_j(P)\,\theta_i(Q)$, because level-2 theta functions are even. □

Proof of Lemma 4. The hypotheses of [Dar24, Theorem 12] are: $\mathcal{L}$ totally symmetric, here of type $\delta = (2,2,2,2)$; a symmetric pair of theta structures for $(\mathcal{L}, \mathcal{L}^2)$, which B provides by [Dar24, Theorem 5]; ζ a primitive $2d_g$-th root of unity, here $d_g = 2$; and a symmetric automorphism ψ of the Heisenberg group whose matrix in the ζ-canonical basis is M,

which exists because symmetric automorphisms surject onto $\mathrm{Sp}_8(\mathbb{Z}/4)$ [Dar24, Section 2.11]. Composing with ψ maps the ζ-canonical basis to BM, so the new structure is the one induced by BM, and it depends on M only [Dar24, Theorem 5(ii)]. With these identifications, and with Dartois's blocks $\left(\begin{smallmatrix} A & C \\ B & D \end{smallmatrix}\right)$ written $\left(\begin{smallmatrix} U & V \\ W & U' \end{smallmatrix}\right)$, Theorem 12 of [Dar24] is the displayed formula, with $\langle\cdot|\cdot\rangle$ the standard product since all $d_l = 2$. Its condition on $\mathbf{i}_0$ is a condition on sections: the proof obtains $\theta'_0 = \lambda T_{\mathbf{i}_0}$ from [Dar24, Proposition 8], for any $\mathbf{i}_0$ with $T_{\mathbf{i}_0} \neq 0$. Since θ' is determined by the new structure up to a scalar, the choice of admissible $\mathbf{i}_0$ does not matter.

(i) This is [Dar24, Lemma 11] with $\Delta = 1$: $U^{\mathsf{T}}U' - W^{\mathsf{T}}V \equiv 1 \pmod 4$. (ii) With $V = 0$ we have $\zeta^{-2\langle \mathbf{i}_0 | U'\mathbf{j}\rangle} = (-1)^{\langle U'^{\mathsf{T}}\mathbf{i}_0 | \mathbf{j}\rangle}$, so $T_{\mathbf{i}_0} = \theta^B_{\mathbf{i}_0} \sum_{\mathbf{j}} (-1)^{\langle U'^{\mathsf{T}}\mathbf{i}_0|\mathbf{j}\rangle}$. By (i), U' is invertible modulo 2, so the sum is 16 for $\mathbf{i}_0 = 0$ and 0 otherwise. The sixteen θ^B_a form a basis of the sections of $\mathcal{L}$ [Dar24, Section 2.4] and $16 \neq 0$ in odd characteristic, so $T_0 = 16\theta^B_0 \neq 0$. The condition concerns the section θ^B_0, not its value at the origin, so it holds even where the theta null point has $\theta^B_0(0) = 0$. (iii) With $V = 0$ and $\mathbf{i}_0 = 0$ the exponent is $\mathbf{i}^{\mathsf{T}}(1 - U^{\mathsf{T}}U')\mathbf{j} - \langle U\mathbf{i}|W\mathbf{i}\rangle$ for any integer lifts of $\mathbf{i}, \mathbf{j}$, and the first term vanishes modulo 4 by (i). The second does not depend on the lift of $\mathbf{i}$ either: replacing $\mathbf{i}$ by $\mathbf{i} + 2\mathbf{e}$ changes it by $4\,\mathbf{e}^{\mathsf{T}}U^{\mathsf{T}}W\mathbf{i} + 4\langle U\mathbf{e}|W\mathbf{e}\rangle \equiv 0 \pmod 4$, because $U^{\mathsf{T}}W$ is symmetric modulo 4 [Dar24, Lemma 11]. The summand no longer depends on $\mathbf{j}$, and the factor 16 goes into λ. □

Proof of Lemma 5. Since φ preserves the polarization, it preserves the Weil pairing e_4, so $\varphi(B)$ is ζ-symplectic. Next, $\varphi^*\mathcal{L} \cong \mathcal{L}$. Indeed $\varphi^*\mathcal{L}_0$ is symmetric, because φ commutes with $[-1]$, and algebraically equivalent to $\mathcal{L}_0$, because φ preserves the polarization; so $\varphi^*\mathcal{L}_0 \cong t_x^*\mathcal{L}_0$ for some x with $2x = 0$, and then $\varphi^*\mathcal{L} \cong t_x^*(\mathcal{L}_0^2) \cong \mathcal{L}$, since $A'[2]$ is the kernel of the polarization of $\mathcal{L}_0^2$.

Fix an isomorphism $\beta\colon \varphi^*\mathcal{L} \to \mathcal{L}$. The pair (φ, β) is an instance of the situation $(*)$ of [Mum66, p. 290], with $\pi = \varphi$ an isomorphism, so that its kernel and the corresponding level subgroup are trivial. By [Mum66, Proposition 2, p. 291] it induces an isomorphism τ of the theta group of $\mathcal{L}$ lying over φ (Mumford's map (2) on p. 302, the map α_f of [DLRW24, proof of Theorem 56]), and $(\varphi, \beta^{\otimes 2})$ likewise induces τ_2 on the theta group of $\mathcal{L}^2$. Mumford states [Mum66, p. 312, diagram (s), with $n = 2$], for every separable isogeny, that these maps commute with his maps ε_2 and η_2 [Mum66, pp. 309–311]. They therefore commute with $\delta_2 = \eta_2 \circ \varepsilon_2$ [Mum66, Proposition 5(iii), p. 311] on both theta groups, and hence with δ_{-1}, since $\delta_2(z) = z^3\delta_{-1}(z)$ [Mum66, p. 312]. Write the compatible pair of symmetric theta structures induced by B as Mumford's isomorphisms j_1 and j_2 from the theta groups of $\mathcal{L}$ and $\mathcal{L}^2$ onto his standard groups of types $(2,2,2,2)$ and $(4,4,4,4)$ [Mum66, p. 317]. By these commutations, $(j_1 \circ \tau^{-1}, j_2 \circ \tau_2^{-1})$ is again a compatible pair of symmetric theta structures in the sense of [Mum66, p. 317]. Since τ_2 lies over φ, its symplectic isomorphism onto $A'[4]$ maps the ζ-canonical basis to $\varphi(B)$, so by [Dar24, Theorem 5(ii)] $j_1 \circ \tau^{-1}$ is the structure induced by $\varphi(B)$. Finally, Mumford's isogeny theorem [Mum66, Theorem 4, pp. 302–303], applied to (φ, β) with trivial kernel and with the structure $j_1 \circ \tau^{-1}$ on the target, so that the map σ there is the identity, shows that φ^* maps the section with coordinates f for the structure induced by $\varphi(B)$ to c times the section with coordinates f for the structure induced by B, for a constant $c \in \Bbbk^*$. In the form of [Dar24, Theorem 3], this is $\theta_a^{\varphi(B)} \circ \varphi = c\,\theta_a^B$ for all a, that is, $\theta_a^{\varphi(B)}(\varphi(z)) = c\,\theta_a^B(z)$, which is the first assertion.

For the second, Lemma 4 with $V = W = 0$ gives $\theta_a^{BM} = \lambda\,\theta^B_{Ua}$. Hence $\theta_a^B(z) = c^{-1}\theta_a^{\varphi(B)}(\varphi(z)) = c^{-1}\lambda\,\theta^B_{Ua}(\varphi(z))$, and putting $b = Ua$, which runs over all indices because U is invertible modulo 2, $\theta_b^B(\varphi(z)) = c\lambda^{-1}\theta^B_{U^{-1}b}(z)$. □

Proof of Lemma 6. By Proposition 3, $D_{k+1} \circ f = f \circ D_k$. The hypothesis on s is an identity between points of $A_k[4]$, so $D_{k+1}f(s) = f(D_k s) = f(s)\,U$; and $D_{k+1}f(T'') = f(D_kT'') = f(T'')\,U$ exactly. □

Proof of Theorem 1. Lemma 6, with the pairing of the pushed generators noted above, carries the block action from the gluing codomain to every later fourfold of the half-chain. On each of them, Lemma 5 turns it into $\theta_a(D_k z) = \mu_k\,\theta_{U^{-1}a}(z)$. Modulo 2, $U^{-1} = U^{\mathsf{T}}$ is the permutation matrix of $(0\ 2)(1\ 3)$, so by the index convention $U^{-1}a = \pi(a)$. The step from A_k uses the 8-torsion points $T_0'', \dots, T_3''$ above the kernel, and $T_2'' = \pm D_k T_0''$, $T_3'' = \pm D_k T_1''$. Since level-2 theta functions are even, $\theta(T_2'') = \theta(D_k T_0'') = \mu_k\,\pi(\theta(T_0''))$, and likewise for T_3''. The two-point chain uses $\pi(\theta(T_0''))$ and $\pi(\theta(T_1''))$ in place of the theta coordinates of T_2'' and T_3'': the same projective points, hence the same kernel, isogeny and codomain, and the doublings and images it computes for T_0'', T_1'' are those of the four-point chain. □

Proof of Lemma 7. On the first half, $(a_1, a_2, 1, 0) \mapsto (a_2, -a_1, 0, -1) = -(-a_2, a_1, 0, 1)$, so $Dg_0 = -g_2$; the same computation on the second half, with q in place of 1, gives $Dg_1 = -g_3$. □

Proofs for Section 5.6

Proof of Lemma 8. The rule enforces $ad - bc \equiv kq \pmod{2^r}$, so by bilinearity $e(\tau R, \tau S) = e(P,Q)^{ad-bc} = e(P,Q)^{kq} = e(R,S)^q$. Hence $e(\tau R, \lambda\tau S) = \zeta^{q\lambda} = \zeta$, which has order 2^r, so τR and $\lambda\tau S$ form a basis of $E_{\mathrm{chl}}[2^r]$. The pairing e_A is the product of the pairings on the four copies, so $e_A(b_n, b_{n+4}) = \zeta$ for $0 \le n \le 3$, while two base points on different copies, or equal ones, pair to 1. Bilinearity gives the formula. □

Proof of Lemma 9. On the first coefficients of each copy, τ acts as 1 and $\widehat{\tau}$ as q; on the second coefficients, τ acts as q and $\widehat{\tau}$ as 1. Substituting these into F and $\widehat{F}$ gives the blocks. Since $\alpha\widehat{\alpha} = \widehat{\alpha}\alpha = a_1^2 + a_2^2$ (Lemma 1(iii)) and integers commute with everything, multiplying the 2×2 blocks gives $F_i\widehat{F}_i = \widehat{F}_i F_i = (a_1^2 + a_2^2 + q)I$ for $i = 1, 2$. The blocks satisfy $\widehat{F}_2 = F_1^{\mathsf{T}}$ and $\widehat{F}_1 = F_2^{\mathsf{T}}$, because $\widehat{\alpha} = \alpha^{\mathsf{T}}$; this says $F^{\mathsf{T}}\Omega = \Omega\widehat{F}$, that is, $\omega(Fu, v) = \omega(u, \widehat{F}v)$. On coefficient vectors $\tilde{D} = \mathrm{diag}(K, K)$ with $K = \mathrm{diag}(J, J)$ and $J = \left(\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right)$ (Lemma 7). The blocks $\alpha = a_1 + a_2 J$ and $\widehat{\alpha} = a_1 - a_2 J$ commute with J, and so do integers, so K commutes with F_1, F_2, $\widehat{F}_1$ and $\widehat{F}_2$. Finally $K^{\mathsf{T}}K = I$, so $\tilde{D}^{\mathsf{T}}\Omega\tilde{D} = \Omega$. □

Proof of Lemma 10. A symplectic matrix is invertible, so its columns form a basis, and the span of X is a free direct summand with complement the span of Y. It is isotropic, since $\omega(X, X) = 0$. If $v = Y\mathbf{a} + X\mathbf{b}$ is orthogonal to every column of X, then $0 = \omega(v, X) = \mathbf{a}^{\mathsf{T}}$, so v lies in the span of X: the summand is its own orthogonal, that is, Lagrangian. Let $\Delta = \tilde{D}Y - YU$. Since $\tilde{D}$ preserves ω and $U^{\mathsf{T}}U = I$,

$$\omega(\Delta, XU) = \omega(\tilde{D}Y, \tilde{D}X) - U^{\mathsf{T}}\omega(Y, X)\,U = I - U^{\mathsf{T}}U = 0,$$

and U is invertible, so every column of Δ is orthogonal to the span of X and lies in it: $\Delta = XS$, and $\omega(Y, \Delta) = \omega(Y, X)S = S$. □

Proof of Lemma 11. The verifier builds its matrices in four steps; we follow them. (1) The first kernel block $X^{(1)}$ is that of Lemma 7, paired by U; its columns are columns of $\widehat{F}$, so $FX^{(1)} = F\widehat{F}E = 2^e E = 0$, with E as after Lemma 9. (2) Its completion $Y^{(1)}$ satisfies $\tilde{D}Y^{(1)} = Y^{(1)}U + X^{(1)}S_1$ by Lemma 10. (3) The second kernel block is $X^{(2)} = FY^{(1)}$, and $\tilde{D}X^{(2)} = F\tilde{D}Y^{(1)} = X^{(2)}U + FX^{(1)}S_1 = X^{(2)}U$. It is isotropic, since $\omega(FY^{(1)}, FY^{(1)}) = \omega(Y^{(1)}, \widehat{F}FY^{(1)}) = 0$, and its completion $Y^{(2)}$ satisfies $\tilde{D}Y^{(2)} = Y^{(2)}U + X^{(2)}S_2$, again by Lemma 10. (4) The first half-chain's kernel columns are $X_1 = -\widehat{F}Y^{(2)}$, and $\tilde{D}X_1 = -\widehat{F}Y^{(2)}U - \widehat{F}FY^{(1)}S_2 = X_1 U$, since $\widehat{F}F = 2^e = 0$. The matrix M_1 is symplectic: $\omega(Y^{(1)}, X_1) = -\omega(FY^{(1)}, Y^{(2)}) = -\omega(X^{(2)}, Y^{(2)}) = \omega(Y^{(2)}, X^{(2)})^{\mathsf{T}} = I$ and $\omega(X_1, X_1) = \omega(Y^{(2)}, F\widehat{F}Y^{(2)}) = 0$ by (4), while $\omega(Y^{(1)}, Y^{(1)}) = 0$ from step (2). Lemma 10 applied to M_1 and M_2 gives the last assertion. Applying the columns to the

base points, which are the same on both copies of each curve, turns these identities into identities between points. □

Proof of Lemma 12. The algorithm takes q odd and names the odd one of a_1, a_2 as a_1, which is possible because $a_1^2 + a_2^2 = 2^e - q$ is odd [Dar24, Section 5.1 and Algorithm 6]. It computes the first $m = v_2(a_2)$ steps of each half-chain in dimension 2 and step $m+1$ as the dimension-4 gluing [Dar24, Lemmas 23 and 28]. This needs m finite, that is, $a_2 \neq 0$, and then $m \geq 1$ because a_2 is even. For a good degree nothing is excluded: $2^e - q$ is then a prime, which is not a square, so $a_2 \neq 0$. □

Proof of Proposition 6. (i) The basis $\tilde{B}_0$ is symplectic by Lemma 8, its two pairs pairing to $e(R', S')$. In its coordinates, $\omega(s_1, t_1) = -\kappa a_1 - \epsilon\xi = 1$, $\omega(s_2, t_2) = \mu a_1 + \epsilon\kappa q = 1$, $\omega(s_1, t_2) = \epsilon(\eta q - \kappa a_2) = 0$, $\omega(s_2, t_1) = -\epsilon\mu a_2 + \epsilon\mu a_2 = 0$, $\omega(t_1, t_2) = a_1^2 + a_2^2 + q = 2^e \equiv 0$, and $\omega(s_1, s_2) = -\eta\mu a_2 - \kappa(\mu + \xi) \equiv 0$ modulo 2^{m+2}, since $\eta\mu a_2 = \kappa\mu a_2^2 q^{-1}$ with a_2 even and $\mu + \xi$ is even. Since 2^{m+2} divides $4a_2$, $4t_1 = (a_1 x, \epsilon\tau x)$ with $x = 4R'$ and $4t_2 = (a_1 y, \epsilon\tau y)$ with $y = 4S'$, using $q\lambda \equiv 1$; these generate $\ker\varphi$. Modulo 4, $\kappa \equiv \eta \equiv 0$, $\xi \equiv -\epsilon$ and $\mu \equiv a_1^{-1}$, and $\tilde{M}$ reduces to the matrix of [Dar24, proof of Lemma 25] for $h = 1$ and of [Dar24, Appendix C.1] for $h = 2$, which give the bases of $(E_{\mathrm{com}} \times E_{\mathrm{chl}})[4]$ with which the algorithm starts φ. The last assertion is [DLRW24, Theorem 56] for φ and the basis $\tilde{B}$, whose half above the kernel is (t_1, t_2), in the form of [Dar24, Section 4.2], on each pair of copies.

(ii) A point of $(E_{\mathrm{com}} \times E_{\mathrm{chl}})[2^{m+2}]$ with coordinates z in $\tilde{B}_0$ has coordinates $\tilde{M}^{-1}z$ in $\tilde{B}$. Since $\tilde{M}$ is symplectic, $\tilde{M}^{-1} = \begin{pmatrix} M_{22}^{\mathsf{T}} & -M_{12}^{\mathsf{T}} \\ -M_{21}^{\mathsf{T}} & M_{11}^{\mathsf{T}} \end{pmatrix}$ for $\tilde{M} = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix}$ in 2×2 blocks, which gives the displayed u. A completion column v gives the point $y_{h,j}$, whose components on copies $(0, 2)$ and $(1, 3)$ are $2^m w$ and $2^m w'$, with $w = \tilde{B}_0 z$ and $w' = \tilde{B}_0 z'$. So $\varphi(2^m w) = \sum_i u_i \cdot 2^m \varphi(s_i) + 2^m u_{2+i} \cdot \varphi(t_i)$, and the points of C are 4-torsion, which gives the coefficients modulo 4. A kernel-lift column gives $x'_{h,m,j}$, whose components are w and w'. Since $4x'_{h,m,j} \in \ker\Phi_{h,m}$, $4w \in \langle 4t_1, 4t_2 \rangle$, so 2^{m+2} divides $4u_1$ and $4u_2$; hence 2^m divides u_1 and u_2, and $\varphi(w) = \sum_i (u_i/2^m) \cdot 2^m \varphi(s_i) + u_{2+i} \cdot \varphi(t_i)$. The same holds for w'. For $h = 1$, reducing modulo 4 as in (i), with $2^m a_2 \equiv 0$, gives the rows of [Dar24, Lemma 30]. □

Proof of Theorem 2. The argument is the same for $h = 1$ and $h = 2$: the second half-chain is the chain of Φ_2, computed from A by the same algorithm from M_2 (Definition 2). Fix h and let U be the signed permutation of Lemma 11. The kernel of $\Phi_{h,k}$ is generated by the $4x'_{h,k,j}$ and is stable under D by Lemma 11, so the proof of Proposition 3 gives an automorphism $D_{h,k}$ with $D_{h,k} \circ \Phi_{h,k} = \Phi_{h,k} \circ D$, which preserves the principal polarization of $A_{h,k}$ (as Proposition 4 notes, that proof uses only the stability of the kernel).

Multiplying the identity $\tilde{D}X_h = X_h U$ of Lemma 11 by 2^{r-2-k} and applying it to the base points gives $Dx'_{h,k} = x'_{h,k} U$ exactly, hence $D_{h,k}\Phi_{h,k}(x'_{h,k}) = \Phi_{h,k}(x'_{h,k})\, U$. Multiplying $\tilde{D}Y^{(h)} = Y^{(h)}U + X_h S_h$ by 2^{r-2} in the same way gives $Dy_h = y_h U + x_h S_h$, where $x_{h,j} = 2^{e_h} x'_{h,j}$. The point $4x'_{h,k,j} = 2^{e_h - k + 2} x'_{h,j}$ lies in the kernel of $\Phi_{h,k}$, and $x_{h,j}$ is a multiple of it as soon as $k \geq 2$; then $\Phi_{h,k}(x_{h,j}) = 0$ and $D_{h,k}\Phi_{h,k}(y_h) = \Phi_{h,k}(y_h)\, U$. So $D_{h,k} B_{h,k} = B_{h,k} \operatorname{diag}(U, U)$, exactly.

The basis $B_{h,k}$ is symplectic for a primitive fourth root of unity (paragraph after Definition 2), so Lemma 5 now gives $\theta_a(D_{h,k} z) = \mu\, \theta_{U^{-1}a}(z)$. Modulo 2, $U^{-1} = U^{\mathsf{T}}$ is the permutation matrix of (0 2)(1 3), since U maps the first kernel generator to minus the third and the second to minus the fourth, and the third and fourth to the first and second. By the index convention of Section 5.5, the index map $\mathbf{i} \mapsto U^{-1}\mathbf{i}$ mod 2 exchanges bits 0 and 2 and bits 1 and 3: it is π.

For the last assertion, Lemma 11 gives the pairing of the generators, and Lemma 12 shows that the gluing codomain is $A_{h,m+1}$ with $m + 1 \geq 2$, where the block action has just been proved; Theorem 1 applies. The verifier evaluates $\widehat{\Phi}_2$ as the dual of the second

half-chain, step by step, from the steps that chain computes; since the two-point algorithm computes the same steps, that evaluation is unchanged as well. □

Proof of Proposition 5. Write $A' = A_{h,k}$, $D' = D_{h,k}$, and $\mathcal{L} = \mathcal{L}_0^2$ for the line bundle of the structure, with $\mathcal{L}_0$ symmetric and representing the principal polarization, so that $\mathcal{L}$ is totally symmetric [Mum66, p. 307]. As in the proof of Lemma 5, $D'^*\mathcal{L} \cong \mathcal{L}$; since $D'(0) = 0$, we may choose an isomorphism $\beta\colon D'^*\mathcal{L} \to \mathcal{L}$ whose fiber at 0 is the identity. Let $T(s) = \beta(D'^*s)$ on the sections of $\mathcal{L}$.

First, $T^2 = 1$. Indeed, $\beta \circ D'^*\beta$ is an isomorphism $[-1]^*\mathcal{L} \to \mathcal{L}$ that is the identity at 0, hence the normalized isomorphism of [Mum66, p. 304], which fixes every theta function of a symmetric structure of level 2 by the inverse formula [Mum66, p. 331], because $-x = x$ on $(\mathbb{Z}/2)^4$. Second, by Theorem 2 and Lemma 5, $T(\theta_a) = \mu\,\theta_{\pi(a)}$ for all a and one scalar μ; then $T^2 = 1$ gives $\mu = \pm 1$, and evaluating at 0, where β is the identity, gives $\theta_a(0) = \mu\,\theta_{\pi(a)}(0)$. It remains to show that $\mu = 1$, which we do by computing the trace of T in two ways.

In the basis (θ_a), T is μ times the permutation matrix of π, which has trace 4, since π fixes the indices 0, 5, 10 and 15. On the other hand, D' has order 4, prime to p. Its fixed points form the kernel of $1-D'$, a group of order $\deg(1-D') = 16$, since $D'(1-D') = 1+D'$ and $(1+D')(1-D') = 2$; they are 2-torsion points. On the tangent space of E^2, J acts by $\left(\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right)$, so D acts on the tangent space of A with eigenvalues $i, i, -i, -i$, and so does D' at each of its fixed points, because $\Phi_{h,k}$ has degree a power of 2, hence is separable, and its differential intertwines D and D'. At a fixed point x, β acts on the fiber of $\mathcal{L}$ by a scalar φ_x, with $\varphi_0 = 1$ and $\varphi_x^2 = 1$, because φ_x^2 is the scalar by which the normalized isomorphism acts on the fiber at a point of order 2, which is 1 since $\mathcal{L}$ is totally symmetric [Mum66, p. 305, Proposition 1]. Since $\mathcal{L}$ is ample, $H^i(A', \mathcal{L}) = 0$ for $i > 0$ [Mum74, §16, p. 150]. Donovan's Lefschetz–Riemann–Roch formula [Don69, Corollary 5.5, with Lemmas 4.2 and 4.3], which holds for automorphisms of order prime to the characteristic, then gives, in the Witt ring of $\mathbb{k}$, which has characteristic 0,

$$\mathrm{Br}(T) = \sum_{x \in \ker(1-D')} \frac{w(\varphi_x)}{\prod_t (1 - w(\alpha_t))},$$

where Br is the Brauer trace, w the Teichmüller lift, and the α_t are the eigenvalues of D' on the cotangent space at x. The eigenvalues of T are ± 1, so $\mathrm{Br}(T) = 4\mu$. With $\iota = w(i)$, which satisfies $\iota^2 = w(-1) = -1$, every denominator is $(1-\iota)^2(1+\iota)^2 = (1-\iota^2)^2 = 4$. Hence $\sum_x w(\varphi_x) = 16\mu$, a sum of sixteen terms equal to ± 1, one of which, $w(\varphi_0)$, is 1; so $\mu = -1$ is impossible, and $\mu = 1$.

Finally, the Hadamard transform commutes with the permutation of coordinates by π, because π permutes the bits of the indices and so preserves the parity of $a \cdot b$: $a \cdot \pi(b) = \pi(a) \cdot b$. □

B Conformance of the implementation

Theorem 2 and Proposition 1 are statements about the verifier's algorithm, and apply to our verifier to the extent that the implementation computes that algorithm. We record what a reading of the crate's code establishes.

Remark 1 (Conformance). The following hold by inspection of our crate's code.

(a) *Index convention.* Our crate maps the multi-index (i_0, i_1, i_2, i_3) to $i_0 + 2i_1 + 4i_2 + 8i_3$, and its change of theta coordinates multiplies component l of the multi-index by column l of each block, with the blocks of [Dar24, Theorem 12] in the positions $\left(\begin{smallmatrix} U & V \\ W & U' \end{smallmatrix}\right)$ of Lemma 4. It takes the kernel generators from columns 4 to 7 of the starting matrices, in that order, and places copies 0 and 1 on E_{com} and copies 2 and

3 on E_{chl}. The coordinate permutation of the two-point verifier puts $\theta_{\pi(a)}$ in position a. So bit l is labeled by g_l, as Section 5.5 assumes, and the permutation used is π.

(b) *The index* $\mathbf{i}_0$. Our crate's change of theta coordinates takes for $\mathbf{i}_0$ the first multi-index, in the order $0, 1, \ldots, 15$, whose coefficient vector $T_{\mathbf{i}_0}$ is nonzero. It tests the condition on sections, as Lemma 4 requires, and takes $\mathbf{i}_0 = 0$ whenever $V = 0$.

(c) *The integers* a_1, a_2. Our crate's Cornacchia step returns the odd value as a_1, and rejects, among other cases, when $2^e - q$ is even or not congruent to 1 modulo 4. It computes m as the number of trailing zero bits of a_2, which exceeds 40 when $a_2 = 0$, and rejects every $m > 40$. So every input that reaches our half-chains has a_1 odd and $1 \le m \le 40$, as Lemma 12 requires. Before Cornacchia's algorithm, our verifier repeats the goodness test of Section 4.2 and rejects a degree that is not good; being deterministic, it runs its 40 Miller–Rabin rounds on the first 40 primes as bases. Our signer applies the same deterministic test and the same bound on m to every candidate degree, so it never produces a signature that the verifier rejects for either reason. The properties above are all that this appendix uses.

(d) *Projective kernel points.* Our crate computes the codomain of a step, and the images under it, from each kernel point only through ratios of that point's own coordinates, so a kernel point may be replaced by any nonzero multiple of its coordinate vector, as the proof of Theorem 1 assumes.

(e) *Base points and the dropped entry.* Our crate reads the transmitted matrix as the map τ of Section 5.6, from the commitment basis to points of E_{chl}, uses the base points of Lemma 8 with $\lambda \equiv q^{-1}$, and recovers the dropped entry by the rule of Section 7.2, rejecting a signature whose transmitted a and b are both even, where the rule is not defined. In order, it rejects hints that are not the canonical ones of their curves, then rejects unless $e(P, Q)$ has exact order 2^r, then rejects a and b both even, then rejects unless the discrete logarithm k of $e(R, S)$ to the base $e(P, Q)$ exists and is odd, that is, unless both pairings have exact order 2^r; the entry is recovered last. The basis (P, Q) is the image under the challenge isogeny of the hinted basis of the public-key curve, and $e(P, Q)$ is computed from the pairing of that basis. Every input that reaches our half-chains therefore satisfies the hypotheses of Lemma 8.

(f) *The change of basis at the gluing.* For both half-chains, our crate computes the coordinates $u_1, \ldots, u_4$ of Proposition 6(ii), with the constants of $\tilde{M}$, and assembles the matrix as stated there. By that proposition, the matrix expresses $B_{h,m}$ in the product basis.

The step formulas, checked against [Dar24], are also as the algorithm requires.

(g) *Plain steps.* Images are [Dar24, Algorithm 1], doublings [Dar24, Algorithm 8] with the two constants computed as stated there, and codomains [Dar24, Algorithms 2–4], applying [Dar24, Eq. (21)] along each edge, with kernel point l labeling index bit l as Definition 2 requires.

(h) *The dimension-2 chain.* One chain φ serves both pairs of copies, whose images are combined in the order of Proposition 6. Its kernel points are those of that proposition, and it starts from the change of coordinates of [Dar24, Theorem 12], computed as in (a) and (b) except that its four candidates for $\mathbf{i}_0$ are tried in the order $0, 2, 1, 3$, for the matrices of Proposition 6(i) modulo 4. Its plain steps are the algorithms of (g) in dimension 2, and its gluing evaluates by [Dar24, Eqs. (20), (24) and (25)] with one translate, computed on the elliptic curves [Dar24, Remarks 19 and 20].

(i) *The dimension-4 gluing step.* Its codomain is computed by [Dar24, Algorithm 4] from the four kernel points and the sum of the first two [Dar24, Remark 21], and its images by [Dar24, Algorithm 5] with two translates, computed on the elliptic curves [Dar24, Remarks 19 and 20].

(j) *Undefined cases.* Wherever these algorithms would divide by a vanishing value, our crate rejects the signature: it tests every divisor of the dimension-2 chain, of the two gluings, of the plain steps and of the image check before using it, and a test per site exercises the reject.

(k) *The image check.* Our crate evaluates the chain at $(R_0, 0, 0, 0)$, for the first point R_0 of the hinted basis of $E_{\mathrm{com}}[2^f]$, the point from which it derives R; it compares the x-coordinates of the first two components of the image with those of $[a_1]R_0$ and $[a_2]R_0$, and requires the fourth to be the point at infinity, as Proposition 1 assumes.

(l) *The middle comparison.* Our crate evaluates the second half-chain through its dual, and compares the null point of the first half-chain's last codomain with the Hadamard transform of that of the second's, projectively.

So our crate computes the theta structures of Definition 2 on every fourfold $A_{h,k}$ with $k \geq m$, and Theorem 2 and Proposition 1 apply to our verifier. These are findings of a reading of the code, not of a machine-checked proof.

Tests and safeguards The two-point verifier checks on each signature that the gluing images of g_2, g_3 are the π-permutations of those of g_0, g_1 and that the gluing codomain's null point is fixed by π, at the cost of four evaluations through the gluing, two projective comparisons of points for the images and six comparisons of coordinates for the null point, and falls back to four points otherwise. These checks would make the two-point chain correct even without the conformance above: every plain step consists of coordinatewise operations, Hadamard transforms and products with constants fixed by π, so it commutes with π, and by induction from the gluing the two-point chain computes what the four-point chain computes. Only the direct codomain formula of Section 5 needs more, the consistency of the equations [Dar24, Eq. (21)] it solves along a tree, which the conformance above provides. The fallback has not occurred on any honest signature tested. In the tests, the codomain computation of Section 5, which assumes that each codomain's dual null point is fixed by π, and the four-point verifier, which does not, produce the same codomain, projectively, at all 844 plain steps of both half-chains of five level-I test vectors, in both modes of the two-point verifier, with and without the check at the gluing, and the same verdict on those vectors and on 70 tampered variants. At levels III and V, the same tests give the same codomain at all 1 274 and 1 706 plain steps of five vectors each, and the same verdict on 75 and 70 tampered variants.

C Beyond two-point evaluation

MIKE exploits its automorphism $[i]$ in two ways, of which the verifier now shares the first. In the theta coordinates of MIKE's chain, induced like ours by the images of a symplectic basis, $[i]$ acts as a permutation of the sixteen coordinates [BDD+26, Corollary G.3], so that the codomain's null point has repeated coefficients and each codomain can be computed from two points above the kernel [BDD+26, Theorem E.1]. Theorem 2 and Propositions 5 and 6 give the corresponding facts for the verifier's chain, for every input, and Section 5 uses them in the same way. MIKE assumes that certain theta coordinates do not vanish [BDD+26, Heuristic E.1], which a forged signature could be built to violate; the verifier of Section 5 instead checks the symmetry it uses and falls back to the full

computation, so that it needs no such assumption. MIKE's coordinates in which $[i]$ acts diagonally, by signs [BDD+26, Definition G.1 and Proposition G.3], show the same phenomenon as Proposition 5: there the null point vanishes at the six coordinates that $[i]$ negates.

The second way is not available to the verifier. On the quotient by $[i]$, only the ten coordinates fixed by $[i]$ need to be kept, and doubling formulas in these coordinates are about 1.6 times faster than with all sixteen; but no efficient formulas are known for images under the 2-isogenies in this model [BDD+26, Remark E.2], and MIKE does not use it. The verifier doubles and pushes every kernel point it carries, so it would need both. Efficient image formulas on the quotient by D would reduce the dominant cost of a step, the doublings and images of kernel points, by up to about $1 - 1/1.6 \approx 38\,\%$; finding them is an open question.