New techniques for ideal to isogeny translations

Pierrick Dartois

2025, January 14

Aknowledgements

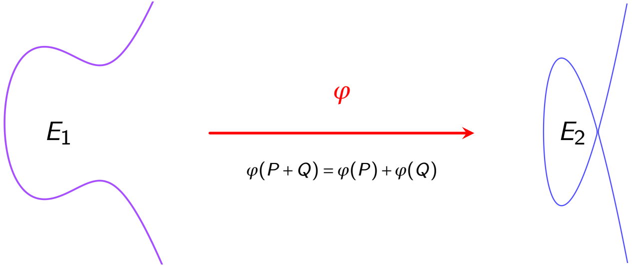
This presentation is based on two joint works:

- [BFD+24] A. Basso, P. Dartois, L. De Feo, A. Leroux, L. Maino, G. Pope, D. Robert and B. Wesolowski. SQIsign2D-West: the Fast, the Small, and the Safer. *Asiacrypt 2024*.
- P. Dartois, A. Herledan Le Merdy, R. Invernizzi, J. Komada Eriksen, T. B. Fouotsa, D. Robert, R. Rueger, F. Vercauteren and B. Wesolowski. *In preparation*.

Isogenies and the Deuring correspondence

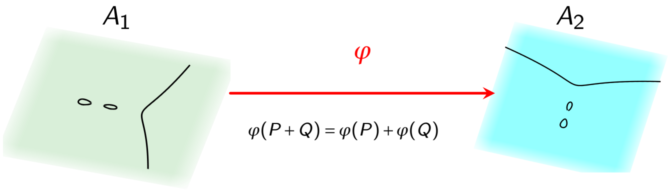
Isogenies between elliptic curves

Between elliptic curves, isogenies are non-zero morphisms of algebraic groups.



Isogenies between abelian varieties

- Abelian varieties are projective abelian group varieties, generalizing elliptic curves.
- Between abelian varieties, isogenies are morphisms which are surjective and of finite kernel.



The Deuring correspondence

Supersingular elliptic curves	Quaternions
$j(E)$ or $j(E)^p$ supersingular	$\mathcal{O} \cong \mathrm{End}(E)$ maximal order in $\mathcal{B}_{p,\infty}$
$\varphi : E \longrightarrow E'$	left $\mathcal{O}$-ideal and right $\mathcal{O}'$-ideal I_φ
$\varphi, \psi : E \longrightarrow E'$	$I_\varphi \sim I_\psi$ $(I_\psi = I_\varphi \alpha,\ \alpha \in \mathcal{B}_{p,\infty})$
$\widehat{\varphi} : E' \longrightarrow E$	$\overline{I_\varphi}$
$\varphi \circ \psi$	$I_\psi \cdot I_\varphi$
$\deg(\varphi)$	$\mathrm{nrd}(I_\varphi) = \sqrt{[\mathcal{O} : I_\varphi]}$

Computing isogenies via the Deuring correspondence

Problem: How to compute isogenies between elliptic curves of known endomorphism rings?

Computing isogenies via the Deuring correspondence

Problem: How to compute isogenies between elliptic curves of known endomorphism rings?

General method:

- Let E_1 and E_2 of known endomorphism rings $\mathcal{O}_1 \cong \mathrm{End}(E_1)$ and $\mathcal{O}_2 \cong \mathrm{End}(E_2)$.
- Compute a connecting ideal I between $\mathcal{O}_1$ and $\mathcal{O}_2$ (left $\mathcal{O}_1$-ideal and right $\mathcal{O}_2$-ideal).
- Translate I into an isogeny $\varphi_I : E_1 \longrightarrow E_2$.

Computing isogenies via the Deuring correspondence

Problem: How to compute isogenies between elliptic curves of known endomorphism rings?

General method:

- Let E_1 and E_2 of known endomorphism rings $\mathcal{O}_1 \cong \text{End}(E_1)$ and $\mathcal{O}_2 \cong \text{End}(E_2)$.
- Compute a connecting ideal I between $\mathcal{O}_1$ and $\mathcal{O}_2$ (left $\mathcal{O}_1$-ideal and right $\mathcal{O}_2$-ideal).
- Translate I into an isogeny $\varphi_I : E_1 \longrightarrow E_2$.

✓ Becomes hard when $\text{End}(E_1)$ or $\text{End}(E_2)$ is unknown.

Computing isogenies via the Deuring correspondence

Problem: How to compute isogenies between elliptic curves of known endomorphism rings?

General method:

- Let E_1 and E_2 of known endomorphism rings $\mathcal{O}_1 \cong \text{End}(E_1)$ and $\mathcal{O}_2 \cong \text{End}(E_2)$.
- Compute a connecting ideal I between $\mathcal{O}_1$ and $\mathcal{O}_2$ (left $\mathcal{O}_1$-ideal and right $\mathcal{O}_2$-ideal).
- Translate I into an isogeny $\varphi_I : E_1 \longrightarrow E_2$.

✓ Becomes hard when $\text{End}(E_1)$ or $\text{End}(E_2)$ is unknown.

✓ Takes polynomial time.

These are good features to build cryptographic schemes (like SQIsign).

Computing isogenies via the Deuring correspondence

Problem: How to compute isogenies between elliptic curves of known endomorphism rings?

General method:

- Let E_1 and E_2 of known endomorphism rings $\mathcal{O}_1 \cong \mathrm{End}(E_1)$ and $\mathcal{O}_2 \cong \mathrm{End}(E_2)$.
- Compute a connecting ideal I between $\mathcal{O}_1$ and $\mathcal{O}_2$ (left $\mathcal{O}_1$-ideal and right $\mathcal{O}_2$-ideal).
- Translate I into an isogeny $\varphi_I : E_1 \longrightarrow E_2$.

✓ Becomes hard when $\mathrm{End}(E_1)$ or $\mathrm{End}(E_2)$ is unknown.

✓ Takes polynomial time.

These are good features to build cryptographic schemes (like SQIsign).

? **Ideal Translation Problem:** How to translate I efficiently in practice?

What does it mean to "compute" an isogeny?

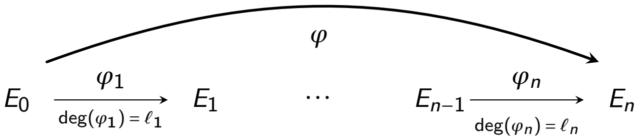
Definition (Efficient representation)

Let $\varphi : E \longrightarrow E'$ be a d-isogeny over $\mathbb{F}_q$. An efficient representation of φ with respect to an algorithm $\mathscr{A}$ is some data $D_\varphi \in \{0,1\}^*$ of size $\mathrm{poly}(\log(d), \log(q))$ s.t. on input $P \in E(\mathbb{F}_{q^k})$ and D_φ, $\mathscr{A}$ returns $\varphi(P)$ in time $\mathrm{poly}(\log(d), k\log(q))$.

What does it mean to "compute" an isogeny?

Examples of efficient representations:

- If $\deg(\varphi) = \prod_{i=1}^{r} \ell_i$, a chain of isogenies:



- If $\deg(\varphi)$ is smooth, a generator $P \in E(\mathbb{F}_q)$ s.t. $\ker(\varphi) = \langle P \rangle$ (Vélu).

- If $\deg(\varphi) < 2^e$ is odd and $E[2^e] = \langle P, Q \rangle$, the image points $(\varphi(P), \varphi(Q))$ (higher dimensional interpolation).

State of the art: translating smooth ideals in dimension 1

Translating smooth ideals (SQIsign)

Problem: How to compute isogenies between elliptic curves of known endomorphism rings?

The SQIsign IdealToIsogeny method:

- Let E_1 and E_2 of known endomorphism rings $\mathcal{O}_1 \cong \mathrm{End}(E_1)$ and $\mathcal{O}_2 \cong \mathrm{End}(E_2)$.
- Compute a connecting ideal I between $\mathcal{O}_1$ and $\mathcal{O}_2$ (left $\mathcal{O}_1$-ideal and right $\mathcal{O}_2$-ideal).
- Compute $J \sim I$ of smooth norm via [KLPT14].
- Translate J into an isogeny $\varphi_J : E_1 \longrightarrow E_2$.

Translating smooth ideals (SQIsign)

Problem: How to compute isogenies between elliptic curves of known endomorphism rings?

The SQIsign Ideal-to-isogeny method:

- Let E_1 and E_2 of known endomorphism rings $\mathcal{O}_1 \cong \mathrm{End}(E_1)$ and $\mathcal{O}_2 \cong \mathrm{End}(E_2)$.
- Compute a connecting ideal I between $\mathcal{O}_1$ and $\mathcal{O}_2$ (left $\mathcal{O}_1$-ideal and right $\mathcal{O}_2$-ideal).
- Compute $J \sim I$ of smooth norm via [KLPT14].
- Translate J into an isogeny $\varphi_J : E_1 \longrightarrow E_2$.

✗ Slow in practice because of the red steps.

The direct method [GPS20]

Input: $E/\mathbb{F}_{p^2}$ supersingular, $\mathcal{O} \cong \mathrm{End}(E)$ and J a left $\mathcal{O}$-ideal of smooth norm.

Output: $\varphi_J : E \longrightarrow E_J$.

The direct method [GPS20]

Input: $E/\mathbb{F}_{p^2}$ supersingular, $\mathcal{O} \cong \mathrm{End}(E)$ and J a left $\mathcal{O}$-ideal of smooth norm.

Output: $\varphi_J : E \longrightarrow E_J$.

- Compute

$$E[J] := \{P \in E \mid \forall \alpha \in J, \quad \alpha(P) = 0\}.$$

The direct method [GPS20]

Input: $E/\mathbb{F}_{p^2}$ supersingular, $\mathcal{O} \cong \mathrm{End}(E)$ and J a left $\mathcal{O}$-ideal of smooth norm.

Output: $\varphi_J : E \longrightarrow E_J$.

- Compute
$$E[J] := \{P \in E \mid \forall \alpha \in J, \quad \alpha(P) = 0\}.$$
- Compute φ_J of kernel $E[J]$ in $O(\mathrm{poly}(\max_{\ell \mid \mathrm{nrd}(J)} \ell))$ operations over $\mathbb{F}_{p^k}$, where $E[J] \subseteq E(\mathbb{F}_{p^k})$.

The direct method [GPS20]

Input: $E/\mathbb{F}_{p^2}$ supersingular, $\mathcal{O} \cong \mathrm{End}(E)$ and J a left $\mathcal{O}$-ideal of smooth norm.

Output: $\varphi_J : E \longrightarrow E_J$.

- Compute
$$E[J] := \{P \in E \mid \forall \alpha \in J, \quad \alpha(P) = 0\}.$$
- Compute φ_J of kernel $E[J]$ in $O(\mathrm{poly}(\max_{\ell \mid \mathrm{nrd}(J)} \ell))$ operations over $\mathbb{F}_{p^k}$, where $E[J] \subseteq E(\mathbb{F}_{p^k})$.

⚠ **Issue:** If J is a KLPT output, then $\mathrm{nrd}(J) \simeq p^{15/4} \gg p$ so k is exponentially big. Not practical for SQISign !

The SQIsign method [FLLW23]

Main idea: Cut the computation into smaller pieces. Write

$$J = J_0 \cdot J_1 \cdots J_{n-1} \quad \text{and} \quad \varphi_J = \varphi_{n-1} \circ \cdots \circ \varphi_1 \circ \varphi_0$$

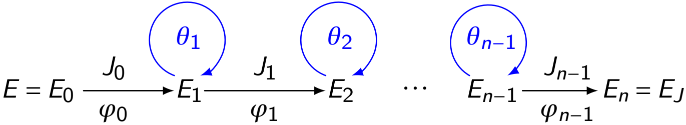
with $\mathrm{nrd}(J_0) = \cdots = \mathrm{nrd}(J_{n-1}) = \ell^f$.

The SQIsign method [FLLW23]

Main idea: Cut the computation into smaller pieces. Write

$$J = J_0 \cdot J_1 \cdots J_{n-1} \quad \text{and} \quad \varphi_J = \varphi_{n-1} \circ \cdots \circ \varphi_1 \circ \varphi_0$$

with $\mathrm{nrd}(J_0) = \cdots = \mathrm{nrd}(J_{n-1}) = \ell^f$.

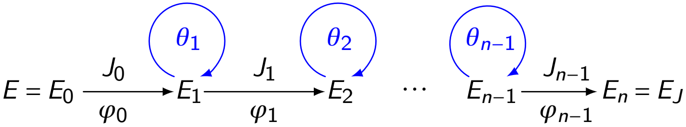


The SQIsign method [FLLW23]

Main idea: Cut the computation into smaller pieces. Write

$$J = J_0 \cdot J_1 \cdots J_{n-1} \quad \text{and} \quad \varphi_J = \varphi_{n-1} \circ \cdots \circ \varphi_1 \circ \varphi_0$$

with $\mathrm{nrd}(J_0) = \cdots = \mathrm{nrd}(J_{n-1}) = \ell^f$.



✗ This is slow in practice!

✗ Torsion requirements: $\ell^f T | p^2 - 1$ where $T \simeq p^{5/4}$ $(\deg(\theta_i) = T^2)$.

✓ Torsion requirements can be reduced with intermediate steps in dimension 2 [ON24].

State of the art: translating short ideals in dimension 4

d-isogenies and the dual isogeny in higher dimension

Definition (d-isogeny)

Let $\varphi : (A, \lambda_A) \longrightarrow (B, \lambda_B)$ be an isogeny between two principally polarized abelian varieties (PPAV). We define:

- $\widetilde{\varphi} := \lambda_A^{-1} \circ \widehat{\varphi} \circ \lambda_B : B \longrightarrow A$.

$$B \xrightarrow{\lambda_B} \widehat{B} \xrightarrow{\widehat{\varphi}} \widehat{A} \xrightarrow{\lambda_A^{-1}} A$$

- We say that φ is a <u>d-isogeny</u> or has <u>reduced degree d</u> if $\widetilde{\varphi} \circ \varphi = [d]_A$.

Kani's embedding lemma [Kan97]

Definition (isogeny diamond)

An (a,b)-isogeny diamond is a commutative diagram s.t.:

$$\begin{array}{ccc} A' & \xrightarrow{\varphi'} & B' \\ \psi \uparrow & & \uparrow \psi' \\ A & \xrightarrow{\varphi} & B \end{array}$$

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

Lemma (Kani)

Consider the (a,b)-isogeny diamond on the left. Then:

- $F : A \times B' \longrightarrow B \times A'$,

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

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

- *If $a \wedge b = 1$, then*

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

Translating short ideals (SQIsignHD)

Problem: How to compute isogenies between elliptic curves of known endomorphism rings?

Ideal-to-isogeny in SQIsignHD:

- Let E_1 and E_2 of known endomorphism rings $\mathcal{O}_1 \cong \mathrm{End}(E_1)$ and $\mathcal{O}_2 \cong \mathrm{End}(E_2)$.
- Compute a connecting ideal I between $\mathcal{O}_1$ and $\mathcal{O}_2$ (left $\mathcal{O}_1$-ideal and right $\mathcal{O}_2$-ideal).
- Compute $J \sim I$ ~~of smooth norm via [KLPT14]~~ with $\mathrm{nrd}(J) \simeq \sqrt{p}$.
- Translate J into an isogeny $\varphi_J : E_1 \longrightarrow E_2$ using dimension 4.

✓ Faster in practice.

Translating short ideals (SQIsignHD)

Assume that:

- $E_1/\mathbb{F}_{p^2}$ is supersingular.
- $E_1[2^e] \subseteq E_1(\mathbb{F}_{p^2})$ with $2^e = \Omega(\sqrt{p})$.
- We have to translate $J \subseteq \mathrm{End}(E_1)$ with $\mathrm{nrd}(J) < 2^e$.
- $2^e - \mathrm{nrd}(J)$ is sum of two squares (e.g. prime $\equiv 1 \mod 4$).
- We know $(\varphi_J(P), \varphi_J(Q))$ where $E_1[2^e] = \langle P, Q \rangle$.

Translating short ideals (SQIsignHD)

Assume that:

- $E_1/\mathbb{F}_{p^2}$ is supersingular.
- $E_1[2^e] \subseteq E_1(\mathbb{F}_{p^2})$ with $2^e = \Omega(\sqrt{p})$.
- We have to translate $J \subseteq \mathrm{End}(E_1)$ with $\mathrm{nrd}(J) < 2^e$.
- $2^e - \mathrm{nrd}(J)$ is sum of two squares (e.g. prime $\equiv 1 \mod 4$).
- We know $(\varphi_J(P), \varphi_J(Q))$ where $E_1[2^e] = \langle P, Q \rangle$.

Goal: Obtain an(other) efficient representation of φ_J.

Translating short ideals (SQIsignHD)

Assume that:

- $E_1/\mathbb{F}_{p^2}$ is supersingular.
- $E_1[2^e] \subseteq E_1(\mathbb{F}_{p^2})$ with $2^e = \Omega(\sqrt{p})$.
- We have to translate $J \subseteq \mathrm{End}(E_1)$ with $\mathrm{nrd}(J) < 2^e$.
- $2^e - \mathrm{nrd}(J)$ is sum of two squares (e.g. prime $\equiv 1 \mod 4$).
- We know $(\varphi_J(P), \varphi_J(Q))$ where $E_1[2^e] = \langle P, Q \rangle$.

Goal: Obtain an(other) efficient representation of φ_J.

Step 1: compute $a_1, a_2 \in \mathbb{Z}$ s.t. $\mathrm{nrd}(J) + a_1^2 + a_2^2 = 2^e$ and consider

$$\alpha_i := \begin{pmatrix} a_1 & a_2 \\ -a_2 & a_1 \end{pmatrix} \in \mathrm{End}(E_i^2), \quad i \in \{1, 2\}.$$

Those are $(a_1^2 + a_2^2)$-isogenies.

Kani's embedding lemma in dimension 4

Applying Kani's lemma:

We have an $(\mathrm{nrd}(J), a_1^2 + a_2^2)$-isogeny diamond:

$$\begin{array}{ccc} E_2^2 & \xrightarrow{\alpha_2} & E_2^2 \\ \uparrow{\scriptstyle \Phi_J} & & \uparrow{\scriptstyle \Phi_J} \\ E_1^2 & \xrightarrow{\alpha_1} & E_1^2 \end{array}$$

with $\Phi_J := \mathrm{Diag}(\varphi_J, \varphi_J)$.

By Kani's lemma, we have the 2^e-isogeny $F \in \mathrm{End}(E_1^2 \times E_2^2)$,

$$F := \begin{pmatrix} \alpha_1 & \widetilde{\Phi}_J \\ -\Phi_J & \widetilde{\alpha}_2 \end{pmatrix}$$

with kernel given by (1).

Kani's embedding lemma in dimension 4

Applying Kani's lemma:

We have an $(\mathrm{nrd}(J), a_1^2+a_2^2)$-isogeny diamond:

$$\begin{array}{ccc} E_2^2 & \xrightarrow{\alpha_2} & E_2^2 \\ \uparrow{\scriptstyle \Phi_J} & & \uparrow{\scriptstyle \Phi_J} \\ E_1^2 & \xrightarrow{\alpha_1} & E_1^2 \end{array}$$

with $\Phi_J := \mathrm{Diag}(\varphi_J, \varphi_J)$.

By Kani's lemma, we have the 2^e-isogeny $F \in \mathrm{End}(E_1^2 \times E_2^2)$,

$$F := \begin{pmatrix} \alpha_1 & \widetilde{\Phi}_J \\ -\Phi_J & \widetilde{\alpha}_2 \end{pmatrix}$$

with kernel given by (1).

Step 2: Given $(\varphi_J(P), \varphi_J(Q))$, compute a basis of:

$$\ker(F) = \{([a_1]R - [a_2]S, [a_2]R + [a_1]S, \varphi_J(R), \varphi_J(S)) \mid R, S \in E_1[2^e]\}. \quad (1)$$

Algorithms for 4-dimensional isogeny computations

Step 3: computing F.

- The 2^e-isogeny F can be computed as a chain of 2-isogenies:

$$E_1^2 \times E_2^2 \xrightarrow{F_1} \mathscr{A}_1 \xrightarrow{F_2} \mathscr{A}_2 \quad \cdots \quad \mathscr{A}_{e-1} \xrightarrow{F_e} E_1^2 \times E_2^2$$

- Each 2-isogeny can be computed efficiently in the Θ-model [Dar24].
- Quasi-linear divide and conquer strategies running in $O(e \log(e))$ apply [JDF11; DLRW24].

Have we "computed" φ_J?

Lemma

F yields an efficient representation of φ_J.

Proof.

We have:

$$F(T,0,0,0) = ([a_1]T, -[a_2]T, -\varphi_J(T), 0).$$

□

Clapoti: translating with less restriction in dimension 2

What is Clapoti?

- **Clapoti:** class group action in polynomial time.
- Work by A. Page and D. Robert [PR23].
- **Goal:** compute efficiently the action of any ideal $\mathfrak{a} \subseteq \mathfrak{O}$ on $\mathfrak{O}$-oriented curves.
- Relies on Kani's lemma and the use of shorter equivalent ideals.
- Made practical with quaternion ideals in SQIsign2D-West [BFD+24] using 2-dimensional isogenies.

AnyIdealToIsogeny (SQIsign2D-West)

Assumptions:

- We work on $E_0 : y^2 = x^3 + x$ ($j = 1728$).
- $E_0[2^e] \subseteq E_0(\mathbb{F}_{p^2})$ with $2^e \approx p$.
- Let (P_0, Q_0) be a basis of $E_0[2^e]$
- If $u < 2^e$ is odd, RandIsogImages from QFESTA [NO23] outputs an efficient representation of a u-isogeny $\varphi : E_0 \longrightarrow E$ (using a 2-dimensional isogeny computation).

Input: Any ideal $I \subset \mathscr{O}_0 \cong \mathrm{End}(E_0)$.

Output: An efficient representation of $\varphi_I : E_0 \longrightarrow E_I$.

Main ingredient: shorter equivalent ideals $I_1, I_2 \sim I$.

AnyIdealToIsogeny (SQIsign2D-West)

Input: Any ideal $I \subset \mathcal{O}_0$.

Output: An efficient representation of $\varphi_I : E_0 \longrightarrow E_I$.

AnyIdealToIsogeny (SQIsign2D-West)

Input: Any ideal $I \subset \mathcal{O}_0$.

Output: An efficient representation of $\varphi_I : E_0 \longrightarrow E_I$.

The AnyIdealToIsogeny algorithm:

- Find ideals $I_1, I_2 \sim I$ of odd norms and $u, v \in \mathbb{N}$ odd s.t. $\gcd(u\,\mathrm{nrd}(I_1), v\,\mathrm{nrd}(I_2)) = 1$ and $u\,\mathrm{nrd}(I_1) + v\,\mathrm{nrd}(I_2) = 2^e$.

AnyIdealToIsogeny (SQIsign2D-West)

Input: Any ideal $I \subset \mathcal{O}_0$.

Output: An efficient representation of $\varphi_I : E_0 \longrightarrow E_I$.

The AnyIdealToIsogeny algorithm:

- Find ideals $I_1, I_2 \sim I$ of odd norms and $u, v \in \mathbb{N}$ odd s.t. $\gcd(u\,\mathrm{nrd}(I_1), v\,\mathrm{nrd}(I_2)) = 1$ and $u\,\mathrm{nrd}(I_1) + v\,\mathrm{nrd}(I_2) = 2^e$.
- Use RandIsogImages of QFESTA to obtain the images of (P_0, Q_0) via isogenies $\varphi_u : E_0 \longrightarrow E_u$ and $\varphi_v : E_0 \longrightarrow E_v$ of degrees u and v.

AnyIdealToIsogeny (SQIsign2D-West)

Input: Any ideal $I \subset \mathcal{O}_0$.

Output: An efficient representation of $\varphi_I : E_0 \longrightarrow E_I$.

The AnyIdealToIsogeny algorithm:

- Find ideals $I_1, I_2 \sim I$ of odd norms and $u, v \in \mathbb{N}$ odd s.t. $\gcd(u\,\mathrm{nrd}(I_1), v\,\mathrm{nrd}(I_2)) = 1$ and $u\,\mathrm{nrd}(I_1) + v\,\mathrm{nrd}(I_2) = 2^e$.
- Use RandIsogImages of QFESTA to obtain the images of (P_0, Q_0) via isogenies $\varphi_u : E_0 \longrightarrow E_u$ and $\varphi_v : E_0 \longrightarrow E_v$ of degrees u and v.
- Let $\beta_1, \beta_2 \in I$ s.t. $I_1 = I\overline{\beta_1}/\mathrm{nrd}(I)$ and $I_2 = I\overline{\beta_2}/\mathrm{nrd}(I)$.
- Then $\theta := \widehat{\varphi}_{I_2} \circ \varphi_{I_1} = \beta_2\overline{\beta_1}/\mathrm{nrd}(I)$.
- Compute $\theta(P_0, Q_0)$.

AnyIdealToIsogeny (SQIsign2D-West)

- Now, consider the Kani isogeny diamond:

$$\begin{array}{ccc} E' & \xrightarrow{\widehat{\varphi'}_v} & E_v \\ \uparrow{\scriptstyle \varphi'_u} & & \uparrow{\scriptstyle \varphi_v \circ \widehat{\varphi}_{I_2}} \\ E_u & \xrightarrow{\widehat{\varphi}_u \circ \varphi_{I_1}} & E_I \end{array}$$

- And the 2^e-isogeny:

$$\Phi := \begin{pmatrix} \varphi_{I_1} \circ \widehat{\varphi}_u & \varphi_{I_2} \circ \widehat{\varphi}_v \\ -\varphi'_u & \varphi'_v \end{pmatrix} : E_u \times E_v \longrightarrow E_I \times E'$$

AnyIdealToIsogeny (SQIsign2D-West)

- Now, consider the Kani isogeny diamond:

$$\begin{array}{ccc} E' & \xrightarrow{\widehat{\varphi'}_v} & E_v \\ \uparrow{\scriptstyle \varphi'_u} & & \uparrow{\scriptstyle \varphi_v \circ \widehat{\varphi}_{I_2}} \\ E_u & \xrightarrow{\widehat{\varphi}_u \circ \varphi_{I_1}} & E_I \end{array}$$

- And the 2^e-isogeny:

$$\Phi := \begin{pmatrix} \varphi_{I_1} \circ \widehat{\varphi}_u & \varphi_{I_2} \circ \widehat{\varphi}_v \\ -\varphi'_u & \varphi'_v \end{pmatrix} : E_u \times E_v \longrightarrow E_I \times E'$$

- It has kernel:

$$\ker(\Phi) = \{([\mathrm{nrd}(I_1)]\varphi_u(P), \varphi_v \circ \theta(P)) \mid P \in E_0[2^e]\}$$

- Using the images of $\theta, \varphi_u, \varphi_v$ of P_0, Q_0 and some DLPs, we obtain $\ker(\Phi)$.
- We then compute Φ in the Theta model.

AnyIdealToIsogeny (SQIsign2D-West)

- The 2^e-isogeny:

$$\Phi := \begin{pmatrix} \varphi_{I_1} \circ \widehat{\varphi}_u & \varphi_{I_2} \circ \widehat{\varphi}_v \\ -\varphi'_u & \varphi'_v \end{pmatrix} : E_u \times E_v \longrightarrow E_I \times E'$$

  represents $\varphi_{I_1} \circ \widehat{\varphi}_u$ and we can evaluate φ_u.
- Hence, we can evaluate φ_{I_1}.
- Besides, $[\mathrm{nrd}(I_1)]\varphi_I = \varphi_{I_1} \circ \beta_1$ so we can evaluate φ_I.

AnyIdealToIsogeny (SQIsign2D-West)

How to solve the norm equation:

$$u \,\mathrm{nrd}(I_1) + v \,\mathrm{nrd}(I_2) = 2^e$$

with $I_1, I_2 \sim I$ and $u, v \in \mathbb{N}$ s.t. $\gcd(u \,\mathrm{nrd}(I_1), v \,\mathrm{nrd}(I_2)) = 1$?

AnyIdealToIsogeny (SQIsign2D-West)

How to solve the norm equation:

$$u\,\mathrm{nrd}(I_1) + v\,\mathrm{nrd}(I_2) = 2^e$$

with $I_1, I_2 \sim I$ and $u, v \in \mathbb{N}$ s.t. $\gcd(u\,\mathrm{nrd}(I_1), v\,\mathrm{nrd}(I_2)) = 1$?

- Sample $\beta_1, \beta_2 \in I$ and set $I_1 = I\overline{\beta_1}/\mathrm{nrd}(I)$ and $I_2 = I\overline{\beta_2}/\mathrm{nrd}(I)$.
- Stop when we can solve $u\,\mathrm{nrd}(I_1) + v\,\mathrm{nrd}(I_2) = 2^e$.
- We need $\mathrm{nrd}(I_i) = \mathrm{nrd}(\beta_i)/\mathrm{nrd}(I) \simeq \sqrt{p}$ for $i \in \{1, 2\}$.
- ⚠ This may fail.

Clapoti original: class group action by any ideal

Orientations

- Let $\mathfrak{O}$ be a quadratic imaginary order.

Orientations

- Let $\mathfrak{O}$ be a quadratic imaginary order.
- Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve. A (primitive) $\mathfrak{O}$-orientation of E is an embedding:

$$\iota : \mathfrak{O} \hookrightarrow \mathrm{End}(E)$$

that is maximal (it does not extend to a superorder of $\mathfrak{O}$).

- We say that (E, ι) is $\mathfrak{O}$-oriented.

Orientations

- Let $\mathfrak{O}$ be a quadratic imaginary order.
- Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve. A (primitive) $\mathfrak{O}$-orientation of E is an embedding:

$$\iota : \mathfrak{O} \hookrightarrow \mathrm{End}(E)$$

  that is maximal (it does not extend to a superorder of $\mathfrak{O}$).
- We say that (E, ι) is $\mathfrak{O}$-oriented.

- $\mathrm{Cl}(\mathfrak{O})$ acts faithfully and (almost) transitively on the set of $\mathfrak{O}$-oriented curves.
- An ideal $\mathfrak{a} \subseteq \mathfrak{O}$ corresponds to an isogeny $\varphi_{\mathfrak{a}} : E \longrightarrow E_{\mathfrak{a}}$ of kernel:

$$E[\mathfrak{a}] := \{P \in E \mid \forall \alpha \in \mathfrak{a}, \quad \iota(\alpha)(P) = 0\}$$

Orientations

- Let $\mathfrak{O}$ be a quadratic imaginary order.
- Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve. A (primitive) $\mathfrak{O}$-orientation of E is an embedding:

$$\iota : \mathfrak{O} \hookrightarrow \mathrm{End}(E)$$

  that is maximal (it does not extend to a superorder of $\mathfrak{O}$).
- We say that (E, ι) is $\mathfrak{O}$-oriented.

- $\mathrm{Cl}(\mathfrak{O})$ acts faithfully and (almost) transitively on the set of $\mathfrak{O}$-oriented curves.
- An ideal $\mathfrak{a} \subseteq \mathfrak{O}$ corresponds to an isogeny $\varphi_{\mathfrak{a}} : E \longrightarrow E_{\mathfrak{a}}$ of kernel:

$$E[\mathfrak{a}] := \{P \in E \mid \forall \alpha \in \mathfrak{a}, \quad \iota(\alpha)(P) = 0\}$$

- The action is trivial $E \simeq E_{\mathfrak{a}}$ if and only if $\mathfrak{a}$ is principal.

Example: CSIDH

- Let $p \equiv 3 \mod 8$. Consider supersingular Montgomery curves

$$E : y^2 = x^3 + Ax^2 + x$$

with $A \in \mathbb{F}_p$.

- These curves E are all $\mathbb{Z}[\sqrt{-p}]$-oriented:

$$\begin{array}{ccc} \mathbb{Z}[\sqrt{-p}] & \hookrightarrow & \mathrm{End}_{\mathbb{F}_p}(E) \\ \sqrt{-p} & \longmapsto & \pi_p \end{array},$$

where $\pi_p : (x, y) \longmapsto (x^p, y^p)$ is the Frobenius endomorphism of E.

Example: CSIDH

- Let $p \equiv 3 \mod 8$. Consider supersingular Montgomery curves

$$E : y^2 = x^3 + Ax^2 + x$$

with $A \in \mathbb{F}_p$.

- These curves E are all $\mathbb{Z}[\sqrt{-p}]$-oriented:

$$\begin{array}{ccc} \mathbb{Z}[\sqrt{-p}] & \hookrightarrow & \mathrm{End}_{\mathbb{F}_p}(E) \\ \sqrt{-p} & \longmapsto & \pi_p \end{array},$$

where $\pi_p : (x, y) \longmapsto (x^p, y^p)$ is the Frobenius endomorphism of E.

- In CSIDH, the action of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ is used cryptographically (to build a key exchange).
- Other schemes are based on oriented curves (OSIDH, Scallop...).

Cryptographic group action

Definition

A *cryptographic group action* $G \curvearrowright X$ is:

1. Easy to compute: $g \cdot x$ can be evaluated in polynomial time for all $g \in G$ and $x \in X$.
2. One way: given x and $g \cdot x$, $g \in G$ is hard to find.

Cryptographic group action

Definition

A *cryptographic group action* $G \curvearrowright X$ is:

1. Easy to compute: $g \cdot x$ can be evaluated in polynomial time for all $g \in G$ and $x \in X$.
2. One way: given x and $g \cdot x$, $g \in G$ is hard to find.

- With cryptographic group actions, we can derive many schemes (including key exchange, signatures and more).

Cryptographic group action

- Actually, group actions based on orientations are restricted cryptographic group actions. We can act by ideals of small norms $\mathfrak{l}_1, \cdots, \mathfrak{l}_t$ that generate $\mathrm{Cl}(\mathfrak{O})$.

- To act with the whole of $\mathrm{Cl}(\mathfrak{O})$ we consider products

$$\mathfrak{a} = \prod_{i=1}^{t} \mathfrak{l}_i^{e_i}.$$

Cryptographic group action

- Actually, group actions based on orientations are restricted cryptographic group actions. We can act by ideals of small norms $\mathfrak{l}_1, \cdots, \mathfrak{l}_t$ that generate $\mathrm{Cl}(\mathfrak{O})$.

- To act with the whole of $\mathrm{Cl}(\mathfrak{O})$ we consider products

$$\mathfrak{a} = \prod_{i=1}^{t} \mathfrak{l}_i^{e_i}.$$

- ⚠ **Issue:** it is non trivial (and not very efficient) to sample uniform classes in $\mathrm{Cl}(\mathfrak{O})$ with such products, as required in some protocols.

The Clapoti approach - Outline

Goal: Compute $E_{\mathfrak{a}}$ for any ideal $\mathfrak{a} \subseteq \mathfrak{O}$ and $\mathfrak{O}$-oriented curve (E, ι).

The Clapoti approach - Outline

Goal: Compute $E_{\mathfrak{a}}$ for any ideal $\mathfrak{a} \subseteq \mathfrak{O}$ and $\mathfrak{O}$-oriented curve (E, ι).

Step 1: Find ideals $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$ and $u, v \in \mathbb{N}$ such that $\gcd(uN(\mathfrak{b}), vN(\mathfrak{c})) = 1$ and

$$uN(\mathfrak{b}) + vN(\mathfrak{c}) = 2^e.$$

The Clapoti approach - Outline

Goal: Compute $E_{\mathfrak{a}}$ for any ideal $\mathfrak{a} \subseteq \mathfrak{O}$ and $\mathfrak{O}$-oriented curve (E, ι).

Step 1: Find ideals $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$ and $u, v \in \mathbb{N}$ such that $\gcd(uN(\mathfrak{b}), vN(\mathfrak{c})) = 1$ and

$$uN(\mathfrak{b}) + vN(\mathfrak{c}) = 2^e.$$

Step 2: Compute a u-isogeny $\Phi_u : E^2 \longrightarrow E_u^2$ and a v-isogeny $\Phi_v : E^2 \longrightarrow E_v^2$ in dimension 2.

The Clapoti approach - Outline

Goal: Compute $E_{\mathfrak{a}}$ for any ideal $\mathfrak{a} \subseteq \mathfrak{O}$ and $\mathfrak{O}$-oriented curve (E, ι).

Step 1: Find ideals $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$ and $u, v \in \mathbb{N}$ such that $\gcd(uN(\mathfrak{b}), vN(\mathfrak{c})) = 1$ and

$$uN(\mathfrak{b}) + vN(\mathfrak{c}) = 2^e.$$

Step 2: Compute a u-isogeny $\Phi_u : E^2 \longrightarrow E_u^2$ and a v-isogeny $\Phi_v : E^2 \longrightarrow E_v^2$ in dimension 2.

Step 3: Evaluate the endomorphism of E associated to $\mathfrak{b}\bar{\mathfrak{c}}$.

The Clapoti approach - Outline

Goal: Compute $E_{\mathfrak{a}}$ for any ideal $\mathfrak{a} \subseteq \mathfrak{O}$ and $\mathfrak{O}$-oriented curve (E, ι).

Step 1: Find ideals $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$ and $u, v \in \mathbb{N}$ such that $\gcd(uN(\mathfrak{b}), vN(\mathfrak{c})) = 1$ and

$$uN(\mathfrak{b}) + vN(\mathfrak{c}) = 2^e.$$

Step 2: Compute a u-isogeny $\Phi_u : E^2 \longrightarrow E_u^2$ and a v-isogeny $\Phi_v : E^2 \longrightarrow E_v^2$ in dimension 2.

Step 3: Evaluate the endomorphism of E associated to $\mathfrak{b}\bar{\mathfrak{c}}$.

Step 4: Compute a 4-dimensional isogeny $F : E_u^2 \times E_v^2 \longrightarrow E_{\mathfrak{a}}^2 \times E'^2$ embedding $\varphi_{\mathfrak{b}}, \varphi_{\mathfrak{c}}, \Phi_u, \Phi_v$.

Step 5: Extract $E_{\mathfrak{a}}$ from the codomain $E_{\mathfrak{a}}^2 \times E'^2$.

Several parameter tweaks (Steps 1 and 2)

- To simplify 2-dimensional isogeny computations in Step 2, we tweak the norm equation
$$uN(\mathfrak{b}) + vN(\mathfrak{c}) = 2^e.$$

Several parameter tweaks (Steps 1 and 2)

- To simplify 2-dimensional isogeny computations in Step 2, we tweak the norm equation

$$uN(\mathfrak{b}) + vN(\mathfrak{c}) = 2^e.$$

- **Tweak 1:** We require $u = g_u(x_u^2 + y_u^2)$ and $v = g_v(x_v^2 + y_v^2)$ where g_u and g_v are products of small primes that split in $\mathfrak{O}$, so that Φ_u and Φ_v are easier to compute.
- We can define

$$\Phi_u := \begin{pmatrix} x_u & -y_u \\ y_u & x_u \end{pmatrix} \begin{pmatrix} \varphi_u & 0 \\ 0 & \varphi_u \end{pmatrix}$$

with $\deg(\varphi_u) = g_u$, and similarly for Φ_v.
- Only dimension 1 computations are involved.

Several parameter tweaks (Steps 1 and 2)

- To simplify 2-dimensional isogeny computations in Step 2, we tweak the norm equation
$$uN(\mathfrak{b}) + vN(\mathfrak{c}) = 2^e.$$
- **Tweak 1:** We require $u = g_u(x_u^2 + y_u^2)$ and $v = g_v(x_v^2 + y_v^2)$ where g_u and g_v are products of small primes that split in $\mathfrak{O}$, so that Φ_u and Φ_v are easier to compute.
- We can define
$$\Phi_u := \begin{pmatrix} x_u & -y_u \\ y_u & x_u \end{pmatrix} \begin{pmatrix} \varphi_u & 0 \\ 0 & \varphi_u \end{pmatrix}$$
with $\deg(\varphi_u) = g_u$, and similarly for Φ_v.
- Only dimension 1 computations are involved.
- ⚠ **Issue:** This makes the equation harder to solve.

Several parameter tweaks (Steps 1 and 2)

- **Solution - Tweak 2:** Give more freedom to $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$.

Several parameter tweaks (Steps 1 and 2)

- **Solution - Tweak 2:** Give more freedom to $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$.

- Let $\mathfrak{b} = \mathfrak{b}_1 \cdot \mathfrak{b}_2$ and $\mathfrak{c} = \mathfrak{c}_1 \cdot \mathfrak{c}_2$, where $\mathfrak{b}_1$ and $\mathfrak{c}_1$ are a product of small prime ideals in $\mathfrak{O}$ (the "Elkies" part).
- We now solve

$$uN(\mathfrak{b}_2) + vN(\mathfrak{c}_2) = 2^e$$

instead of

$$uN(\mathfrak{b}) + vN(\mathfrak{c}) = 2^e.$$

Several parameter tweaks (Steps 1 and 2)

- **Solution** - **Tweak 2:** Give more freedom to $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$.

- Let $\mathfrak{b} = \mathfrak{b}_1 \cdot \mathfrak{b}_2$ and $\mathfrak{c} = \mathfrak{c}_1 \cdot \mathfrak{c}_2$, where $\mathfrak{b}_1$ and $\mathfrak{c}_1$ are a product of small prime ideals in $\mathfrak{O}$ (the "Elkies" part).
- We now solve

$$uN(\mathfrak{b}_2) + vN(\mathfrak{c}_2) = 2^e$$

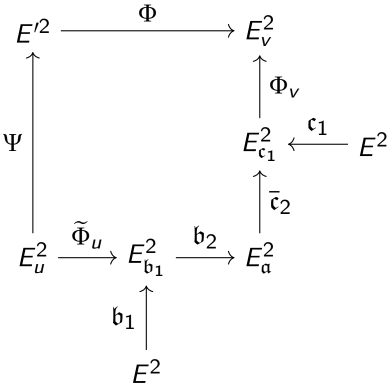
instead of

$$uN(\mathfrak{b}) + vN(\mathfrak{c}) = 2^e.$$

- And precompute the action of $\mathfrak{b}_1$ and $\mathfrak{c}_1$ on E.

Applying Kani's lemma (Steps 3 and 4)

- We have the following $(uN(\mathfrak{b}_2), vN(\mathfrak{c}_2))$-isogeny diamond:



Applying Kani's lemma (Steps 3-5)

- This isogeny diamond yields a 2^e-isogeny 4-dimensional

$$F = \begin{pmatrix} \Phi_{\mathfrak{b}_2} \circ \widetilde{\Phi}_u & \Phi_{\mathfrak{c}_2} \circ \widetilde{\Phi}_v \\ -\Psi & \widetilde{\Phi} \end{pmatrix} : E_u^2 \times E_v^2 \longrightarrow E_{\mathfrak{a}}^2 \times E'^2.$$

Applying Kani's lemma (Steps 3-5)

- This isogeny diamond yields a 2^e-isogeny 4-dimensional

$$F = \begin{pmatrix} \Phi_{\mathfrak{b}_2} \circ \widetilde{\Phi}_u & \Phi_{\mathfrak{c}_2} \circ \widetilde{\Phi}_v \\ -\Psi & \widetilde{\Phi} \end{pmatrix} : E_u^2 \times E_v^2 \longrightarrow E_{\mathfrak{a}}^2 \times E'^2.$$

- Its kernel can be computed by evaluating Φ_u, Φ_v, the action of $\mathfrak{b}_1$, $\mathfrak{c}_1$ and the endomorphism $\mathfrak{b}\overline{\mathfrak{c}}$ (Step 3).

Applying Kani's lemma (Steps 3-5)

- This isogeny diamond yields a 2^e-isogeny 4-dimensional

$$F = \begin{pmatrix} \Phi_{\mathfrak{b}_2} \circ \widetilde{\Phi}_u & \Phi_{\mathfrak{c}_2} \circ \widetilde{\Phi}_v \\ -\Psi & \widetilde{\Phi} \end{pmatrix} : E_u^2 \times E_v^2 \longrightarrow E_{\mathfrak{a}}^2 \times E'^2.$$

- Its kernel can be computed by evaluating Φ_u, Φ_v, the action of $\mathfrak{b}_1$, $\mathfrak{c}_1$ and the endomorphism $\mathfrak{b}\overline{\mathfrak{c}}$ (Step 3).

- F can then be computed efficiently with the Θ-model [Dar24].

Some preliminary results

A SageMath implementation adapted to CSIDH

Field size $\log_2(p)$	Norm eq. (step 1)	Dim 1 (steps 2-3)	Dim 4 (steps 4-5)	Total	Success rate
508	0.75	2.62	9.49	12.49	10/10
1008	1.20	8.75	26.58	36.53	10/10
1554	2.15	16.14	50.50	68.78	10/10
2032	28.22	40.01	77.88	146.11	10/10
4090	210.26	193.62	320.53	724.41	8/10

Table: Preliminary timings (in s) of our implementation on a 2,7 GHz Intel Core i5 dual core with 10 tests per prime size.

Some preliminary results

A SageMath implementation adapted to CSIDH

Field size $\log_2(p)$	Norm eq. (step 1)	Dim 1 (steps 2-3)	Dim 4 (steps 4-5)	Total	Success rate
508	0.75	2.62	9.49	12.49	10/10
1008	1.20	8.75	26.58	36.53	10/10
1554	2.15	16.14	50.50	68.78	10/10
2032	28.22	40.01	77.88	146.11	10/10
4090	210.26	193.62	320.53	724.41	8/10

Table: Preliminary timings (in s) of our implementation on a 2,7 GHz Intel Core i5 dual core with 10 tests per prime size.

- A concurrent work [PPS24] using dimension 2 isogenies adapted to Scallop took 2.5 s with 512 bits discriminant (and 1500 bits p size) in Rust... and 175 s in SageMath.
- We are faster with CSIDH and 4-dimensional isogenies.

Conclusion

To sum up:

- Previous ideal-to-isogeny algorithms involved restrictions on the ideal norm (either smooth or short).
- The most efficient method from SQIsignHD involved dimension 4 isogenies.

Conclusion

To sum up:

- Previous ideal-to-isogeny algorithms involved restrictions on the ideal norm (either smooth or short).
- The most efficient method from SQIsignHD involved dimension 4 isogenies.
- The Clapoti method is a powerful tool to:
  - Translate ideals of any norm with dimension 2 isogenies but only from the special curve E_0 (SQIsign2D-West).
  - Compute the action of any ideal in the oriented case.

Thanks for listening!

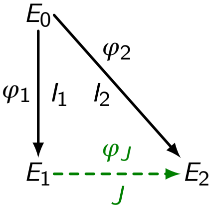
P. Dartois, A. Leroux, D. Robert and B. Wesolowski. SQIsignHD: New Dimensions in Cryptography. Eurocrypt 2024.
https://eprint.iacr.org/2023/436

A. Basso, P. Dartois, L. De Feo, A. Leroux, L. Maino, G. Pope, D. Robert and B. Wesolowski. SQIsign2D-West: The Fast, the Small, and the Safer. Asiacrypt 2024.
https://eprint.iacr.org/2024/760

Preliminary torsion evaluation in SQIsignHD

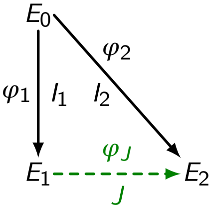
What about torsion images $(\varphi_J(P), \varphi_J(Q))$?

Main idea (SQIsignHD): Use an alternate isogeny path.



What about torsion images $(\varphi_J(P), \varphi_J(Q))$?

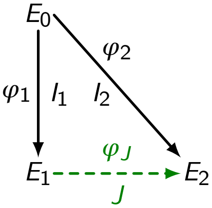
Main idea (SQIsignHD): Use an alternate isogeny path.



- Let $\gamma := \widehat{\varphi}_2 \circ \varphi_J \circ \varphi_1 \in \mathrm{End}(E_0)$.
- We have $\mathscr{O}_0 \gamma = I_1 \cdot J \cdot \overline{I}_2$ so we can compute γ.

What about torsion images $(\varphi_J(P), \varphi_J(Q))$?

Main idea (SQIsignHD): Use an alternate isogeny path.

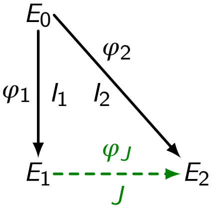


- Let $\gamma := \widehat{\varphi}_2 \circ \varphi_J \circ \varphi_1 \in \mathrm{End}(E_0)$.
- We have $\mathcal{O}_0 \gamma = I_1 \cdot J \cdot \overline{I}_2$ so we can compute γ.
- Then:

$$[\mathrm{nrd}(I_1)\,\mathrm{nrd}(I_2)]\varphi_J = \varphi_2 \circ \gamma \circ \widehat{\varphi}_1$$

What about torsion images $(\varphi_J(P), \varphi_J(Q))$?

Main idea (SQIsignHD): Use an alternate isogeny path.



- Let $\gamma := \widehat{\varphi}_2 \circ \varphi_J \circ \varphi_1 \in \mathrm{End}(E_0)$.
- We have $\mathcal{O}_0\gamma = I_1 \cdot J \cdot \overline{I}_2$ so we can compute γ.
- Then:

$$[\mathrm{nrd}(I_1)\mathrm{nrd}(I_2)]\varphi_J = \varphi_2 \circ \gamma \circ \widehat{\varphi}_1$$

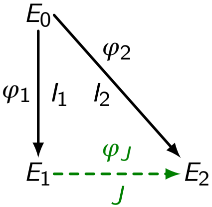
- We can evaluate φ_J on $P, Q \in E_1[2^e]$ provided $\mathrm{nrd}(I_1)\mathrm{nrd}(I_2)$ is odd:

$$\varphi_J(P, Q) = [\lambda]\varphi_2 \circ \gamma \circ \widehat{\varphi}_1(P, Q),$$

with $\lambda \,\mathrm{nrd}(I_1)\mathrm{nrd}(I_2) \equiv 1 \mod 2^e$.

What about torsion images $(\varphi_J(P), \varphi_J(Q))$?

Main idea (SQIsignHD): Use an alternate isogeny path.



- Let $\gamma := \widehat{\varphi}_2 \circ \varphi_J \circ \varphi_1 \in \mathrm{End}(E_0)$.
- We have $\mathcal{O}_0\gamma = I_1 \cdot J \cdot \overline{I}_2$ so we can compute γ.
- Then:

$$[\mathrm{nrd}(I_1)\mathrm{nrd}(I_2)]\varphi_J = \varphi_2 \circ \gamma \circ \widehat{\varphi}_1$$

- We can evaluate φ_J on $P, Q \in E_1[2^e]$ provided $\mathrm{nrd}(I_1)\mathrm{nrd}(I_2)$ is odd:

$$\varphi_J(P, Q) = [\lambda]\varphi_2 \circ \gamma \circ \widehat{\varphi}_1(P, Q),$$

with $\lambda\,\mathrm{nrd}(I_1)\mathrm{nrd}(I_2) \equiv 1 \mod 2^e$.

- $(\varphi_J(P), \varphi_J(Q))$ is also an efficient representation of φ_J.

Computing an isogeny of fixed degree (QFESTA)

Find an isogeny of fixed degree

Input: An odd number $u < 2^e$.

Output: An efficient representation of an isogeny $\varphi : E_0 \longrightarrow E$ of degree u.

Find an isogeny of fixed degree

Input: An odd number $u < 2^e$.

Output: An efficient representation of an isogeny $\varphi : E_0 \longrightarrow E$ of degree u.

If we are lucky:

- Assume $u = a^2 + b^2$.

Find an isogeny of fixed degree

Input: An odd number $u < 2^e$.

Output: An efficient representation of an isogeny $\varphi : E_0 \longrightarrow E$ of degree u.

If we are lucky:

- Assume $u = a^2 + b^2$.
- Then, we can set $\varphi := [a] + [b]\iota \in \mathrm{End}(E_0)$ with:

$$\iota : (x, y) \in E_0 \longmapsto (-x, \sqrt{-1}y) \in E_0.$$

Find an isogeny of fixed degree

Input: An odd number $u < 2^e$.

Output: An efficient representation of an isogeny $\varphi : E_0 \longrightarrow E$ of degree u.

If we are lucky:

- Assume $u = a^2 + b^2$.
- Then, we can set $\varphi := [a] + [b]\iota \in \mathrm{End}(E_0)$ with:

$$\iota : (x, y) \in E_0 \longmapsto (-x, \sqrt{-1}y) \in E_0.$$

⚠ Requiring u (and/or v) sums of two squares in $u\,\mathrm{nrd}(I_1) + v\,\mathrm{nrd}(I_2) = 2^e$ makes it harder to solve.

✓ In practice, we use RandIsogImages from QFESTA.

RandIsogImages [NO23]

Input: An odd number $u < 2^e$.

Output: An efficient representation of an isogeny $\varphi : E_0 \longrightarrow E$ of degree u.

RandIsogImages [NO23]

Input: An odd number $u < 2^e$.

Output: An efficient representation of an isogeny $\varphi : E_0 \longrightarrow E$ of degree u.

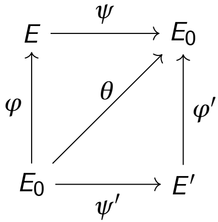
- Compute $\theta \in \mathcal{O}_0$ of norm $u(2^e - u) > p$.

RandIsogImages [NO23]

Input: An odd number $u < 2^e$.

Output: An efficient representation of an isogeny $\varphi : E_0 \longrightarrow E$ of degree u.

- Compute $\theta \in \mathcal{O}_0$ of norm $u(2^e - u) > p$.
- Consider the commutative diagram:



with $\theta = \psi \circ \varphi$, $\deg(\varphi) = u$ and $\deg(\psi) = 2^e - u$.

RandIsogImages [NO23]

- Compute $\theta(P_0, Q_0)$ to obtain the kernel:

$$\ker(\Phi) = \{([u]P, \theta(P)) \mid P \in E_0[2^e]\}$$

of

$$\Phi = \begin{pmatrix} \varphi & \widehat{\psi} \\ -\psi' & \widehat{\varphi'} \end{pmatrix} : E_0 \times E_0 \to E \times E'.$$

- Compute the 2^e-isogeny Φ with the Theta model.

RandIsogImages [NO23]

- Compute $\theta(P_0, Q_0)$ to obtain the kernel:

$$\ker(\Phi) = \{([u]P, \theta(P)) \mid P \in E_0[2^e]\}$$

of

$$\Phi = \begin{pmatrix} \varphi & \widehat{\psi} \\ -\psi' & \widehat{\varphi}' \end{pmatrix} : E_0 \times E_0 \to E \times E'.$$

- Compute the 2^e-isogeny Φ with the Theta model.
- We have $\Phi(P, 0) = (\varphi(P), *)$ so Φ represents φ.