Practical Effective Class Group Action using 4-Dimensional Isogenies

Pierrick Dartois, Jonathan Komada Eriksen, Tako Boris Fouotsa, Arthur Herlédan Le Merdy, Riccardo Invernizzi, Damien Robert, Ryan Rueger, Frederik Vercauteren and Benjamin Wesolowski

2025, April 30

Introduction: class group action on oriented curves

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

- $\mathrm{Cl}(\mathfrak{O})$ acts faithfully and (almost) transitively on the set of $\mathfrak{O}$-oriented curves.

Orientations

- $\mathrm{Cl}(\mathfrak{O})$ 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:

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

- There is also an $\mathfrak{O}$-orientation

$$\iota_{\mathfrak{a}} := (\varphi_{\mathfrak{a}})_*(\iota) : \alpha \longmapsto \frac{1}{N(\mathfrak{a})}\varphi_{\mathfrak{a}} \circ \iota(\alpha) \circ \widehat{\varphi}_{\mathfrak{a}}$$

on $E_{\mathfrak{a}}$.

Orientations

- $\mathrm{Cl}(\mathfrak{O})$ 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:

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

- There is also an $\mathfrak{O}$-orientation

$$\iota_{\mathfrak{a}} := (\varphi_{\mathfrak{a}})_*(\iota) : \alpha \longmapsto \frac{1}{N(\mathfrak{a})} \varphi_{\mathfrak{a}} \circ \iota(\alpha) \circ \widehat{\varphi}_{\mathfrak{a}}$$

  on $E_{\mathfrak{a}}$.

- The action is trivial $(E, \iota) \simeq (E_{\mathfrak{a}}, \iota_{\mathfrak{a}})$ if and only if $\mathfrak{a}$ is principal.

Example: CSIDH and CSURF

- Let $p \equiv 7 \mod 8$. Consider a supersingular Montgomery curve

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

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

- Then $\mathrm{End}_{\mathbb{F}_p}(E)$ contains the Frobenius endomorphism

$$\pi_p : (x, y) \in E \longmapsto (x^p, y^p) \in E,$$

which satisfies $\pi_p^2 = -[p]$.

- Hence E is $\mathbb{Z}[\sqrt{-p}]$-oriented:

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

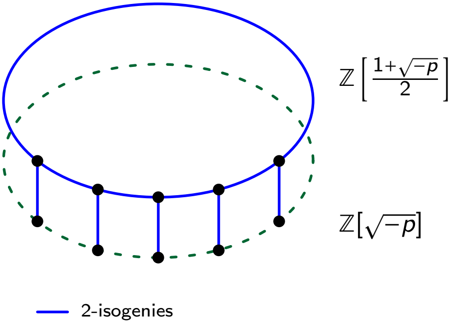
Example: CSIDH and CSURF

- Let $p \equiv 7 \mod 8$. Consider a supersingular Montgomery curve
$$E : y^2 = x^3 + Ax^2 + x$$
with $A \in \mathbb{F}_p$.
- Then $\mathrm{End}_{\mathbb{F}_p}(E)$ contains the Frobenius endomorphism
$$\pi_p : (x, y) \in E \longmapsto (x^p, y^p) \in E,$$
which satisfies $\pi_p^2 = -[p]$.
- Hence E is $\mathbb{Z}[\sqrt{-p}]$-oriented:
$$\begin{array}{ccc} \mathbb{Z}[\sqrt{-p}] & \hookrightarrow & \mathrm{End}_{\mathbb{F}_p}(E) \\ \sqrt{-p} & \longmapsto & \pi_p \end{array}.$$
- This orientation is not always primitive: there are two cases.

Example: CSIDH and CSURF

E is either:

- On the surface: primitively $\mathbb{Z}[(1+\sqrt{-p})/2]$-oriented (CSURF).
- On the floor: primitively $\mathbb{Z}[\sqrt{-p}]$-oriented (CSIDH).



Effective group action

Definition

An *effective group action* (EGA) $G \curvearrowright X$ is:

1. Commutative.
2. Free: $\forall x \in X, g \in G, \quad g \cdot x = x \Longrightarrow g = e$.
3. Transitive: $\forall x, y \in X,\ \exists g \in G, \quad g \cdot x = y$.
4. Easy to compute: $g \cdot x$ can be evaluated in polynomial time for all $g \in G$ and $x \in X$.
5. One way: given x and $g \cdot x$, $g \in G$ is hard to find.

Effective group action

Definition

An *effective group action* (EGA) $G \curvearrowright X$ is:

1. Commutative.
2. Free: $\forall x \in X, g \in G, \quad g \cdot x = x \Longrightarrow g = e$.
3. Transitive: $\forall x, y \in X, \exists g \in G, \quad g \cdot x = y$.
4. Easy to compute: $g \cdot x$ can be evaluated in polynomial time for all $g \in G$ and $x \in X$.
5. One way: given x and $g \cdot x$, $g \in G$ is hard to find.

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

Restricted effective group actions

- Actually, group actions based on orientations are restricted effective 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

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

Restricted effective group actions

- Actually, group actions based on orientations are restricted effective 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 (*e.g.* CSI-FiSh).

The Clapoti method

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>polarized 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.:

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

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*

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

Computing 2^e-isogenies

Theorem (D. and Robert)

Let k be a field such that $\mathrm{char}(k) \neq 2$. Then there exists an algorithm that takes as input:

- *A principally polarised abelian variety A of dimension g defined over k;*
- *Points $T_1, \cdots, T_g \in A[2^{e+2}]$ defined over k forming a maximal isotropic subgroup of $A[2^{e+2}]$;*

And returns a 2^e-isogeny $F : A \longrightarrow B$ with kernel $\langle [4]T_1, \cdots, [4]T_g \rangle$ represented as a chain of 2-isogenies with a number of operations over k polynomial in e and 2^g.

The Clapoti method

Goal: Compute $E_{\mathfrak{a}}$ for any $\mathfrak{a} \subseteq \mathfrak{O}$.

Assumption: $p = c2^e - 1$.

- We solve:

$$uN(\mathfrak{b}) + vN(\mathfrak{c}) = 2^f,$$

  with $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$, $f \leq e-2$ and $\gcd(uN(\mathfrak{b}), vN(\mathfrak{c})) = 1$.
- If Φ_u and Φ_v are d-dimensional, the resulting Kani 2^f-isogeny

$$F : A_u \times A_v \longrightarrow E_{\mathfrak{a}}^d \times A'$$

  is $2d$-dimensional.

$$\begin{tikzcd}
A' \arrow[rr, "\Phi"] & & A_v \\
 & & E^d \arrow[u, "\Phi_v"'] \\
A_u \arrow[uu, "\Psi"] \arrow[r, "\widetilde{\Phi}_u"] & E^d \arrow[r, "\mathfrak{b}"] & E_{\mathfrak{a}}^d \arrow[u, "\bar{\mathfrak{c}}"']
\end{tikzcd}$$

The Clapoti method - Outline

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

The Clapoti method - Outline

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

Assumption: $p = c2^e - 1$.

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

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

The Clapoti method - Outline

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

Assumption: $p = c2^e - 1$.

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

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

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

The Clapoti method - Outline

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

Assumption: $p = c2^e - 1$.

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

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

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

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

The Clapoti method - Outline

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

Assumption: $p = c2^e - 1$.

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

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

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

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

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

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

The auxiliary isogenies Φ_u and Φ_v

- Φ_u and Φ_v are hard to compute in dimension $d=1$.

The auxiliary isogenies Φ_u and Φ_v

- Φ_u and Φ_v are hard to compute in dimension $d = 1$.
- In KlaPoTi [PPS24], they impose $u = v = 1$:

$$N(\mathfrak{b}) + N(\mathfrak{c}) = 2^f.$$

- KLPT [KLPT14] is used to find $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$.
- Only possible for small discriminants $|\mathrm{disc}(\mathfrak{O})| \leq 2^{f/3} \simeq p^{1/3}$.

The auxiliary isogenies Φ_u and Φ_v

- Φ_u and Φ_v are hard to compute in dimension $d = 1$.
- In KlaPoTi [PPS24], they impose $u = v = 1$:

$$N(\mathfrak{b}) + N(\mathfrak{c}) = 2^f.$$

- KLPT [KLPT14] is used to find $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$.
- Only possible for small discriminants $|\mathrm{disc}(\mathfrak{O})| \leq 2^{f/3} \simeq p^{1/3}$.
- PEGASIS: a solution for $|\mathrm{disc}(\mathfrak{O})| \simeq p$ but with $d = 2$.
- We have to compute 4-dimensional isogenies!

From Clapoti to Pegasis: making it effective and efficient

Step 2: computing Φ_u and Φ_v - sums of squares

Goal: Given $u < 2^{e-2}$ odd and an $\mathfrak{O}$-oriented curve (E, ι), compute a u-isogeny $\Phi_u : E^2 \longrightarrow A_u$.

Step 2: computing Φ_u and Φ_v - sums of squares

Goal: Given $u < 2^{e-2}$ odd and an $\mathfrak{O}$-oriented curve (E, ι), compute a u-isogeny $\Phi_u : E^2 \longrightarrow A_u$.

Issue: With any u, it requires to compute a 4-dimensional isogeny [NO23].

Step 2: computing Φ_u and Φ_v - sums of squares

Goal: Given $u < 2^{e-2}$ odd and an $\mathfrak{O}$-oriented curve (E, ι), compute a u-isogeny $\Phi_u : E^2 \longrightarrow A_u$.

Issue: With any u, it requires to compute a 4-dimensional isogeny [NO23].
Idea: Require u of special form.

- Assume $u = g_u(x_u^2 + y_u^2)$.

Step 2: computing Φ_u and Φ_v - sums of squares

Goal: Given $u < 2^{e-2}$ odd and an $\mathfrak{O}$-oriented curve (E, ι), compute a u-isogeny $\Phi_u : E^2 \longrightarrow A_u$.

Issue: With any u, it requires to compute a 4-dimensional isogeny [NO23].
Idea: Require u of special form.

- Assume $u = g_u(x_u^2 + y_u^2)$.
- Then, we can define: We can define

$$\Phi_u := \begin{pmatrix} x_u & -y_u \\ y_u & x_u \end{pmatrix} \begin{pmatrix} \varphi_u & 0 \\ 0 & \varphi_u \end{pmatrix} : E^2 \longrightarrow E_u^2$$

  with $\deg(\varphi_u) = g_u$.
- g_u is a product of small primes that split in $\mathfrak{O}$ so that φ_u is given by an ideal action $\mathfrak{g}_u$.
- Only 1-dimensional computations are involved.

Step 1: tweaking the norm equation

- We want to solve:

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

with $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$, $\gcd(uN(\mathfrak{b}), vN(\mathfrak{c})) = 1$, $f \leq e-2$,

$$u = g_u(x_u^2 + y_u^2) \quad \text{and} \quad v = g_v(x_v^2 + y_v^2).$$

Step 1: tweaking the norm equation

- We want to solve:

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

  with $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$, $\gcd(uN(\mathfrak{b}), vN(\mathfrak{c})) = 1$, $f \leq e-2$,

$$u = g_u(x_u^2 + y_u^2) \quad \text{and} \quad v = g_v(x_v^2 + y_v^2).$$

- **Issue:** This might be to tight to be solved.

Step 1: tweaking the norm equation

- We want to solve:

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

  with $\mathfrak{b}, \mathfrak{c} \sim \mathfrak{a}$, $\gcd(uN(\mathfrak{b}), vN(\mathfrak{c})) = 1$, $f \leq e - 2$,

$$u = g_u(x_u^2 + y_u^2) \quad \text{and} \quad v = g_v(x_v^2 + y_v^2).$$

- **Issue:** This might be to tight to be solved.
- **Solution:** 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 $\mathfrak{O}$.
- Solve the following instead:

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

Step 1: an algorithm

Goal: solve $uN(\mathfrak{b}_2) + vN(\mathfrak{c}_2) = 2^f$.

- Sample $\beta, \gamma \in \mathfrak{a}$ as follows:
  - Find a Lagrange-Gauss reduced basis (α_1, α_2) of $\mathfrak{a}$.
  - Sample small $x, y, z, t \in \mathbb{Z}$ and set $\beta := x\alpha_1 + y\alpha_2$ and $\gamma := z\alpha_1 + t\alpha_2$.
- Set $\mathfrak{b} := \mathfrak{a}\overline{\beta}/N(\mathfrak{a})$ and $\mathfrak{c} := \mathfrak{a}\overline{\gamma}/N(\mathfrak{a})$.
- Factor $\mathfrak{b} = \mathfrak{b}_1 \cdot \mathfrak{b}_2$ and $\mathfrak{c} = \mathfrak{c}_1 \cdot \mathfrak{c}_2$.
- Repeat until we can find suitable u, v.

Step 1: norm constraints

- By Minkowski's bounds, the Lagrange-Gauss reduced basis satisfies:

$$N(\alpha_1)N(\alpha_2) \simeq N(\mathfrak{a})^2|\Delta|$$

- So we expect $N(\alpha_1) \simeq N(\alpha_2) \simeq N(\mathfrak{a})\sqrt{|\Delta|}$, so that $N(\beta) \simeq N(\gamma) \simeq N(\mathfrak{a})\sqrt{|\Delta|}$ and:

$$N(\mathfrak{a}) \simeq N(\mathfrak{b}) \simeq \sqrt{|\Delta|}.$$

Step 1: norm constraints

- By Minkowski's bounds, the Lagrange-Gauss reduced basis satisfies:

$$N(\alpha_1)N(\alpha_2) \simeq N(\mathfrak{a})^2|\Delta|$$

- So we expect $N(\alpha_1) \simeq N(\alpha_2) \simeq N(\mathfrak{a})\sqrt{|\Delta|}$, so that $N(\beta) \simeq N(\gamma) \simeq N(\mathfrak{a})\sqrt{|\Delta|}$ and:

$$N(\mathfrak{a}) \simeq N(\mathfrak{b}) \simeq \sqrt{|\Delta|}.$$

- To solve $uN(\mathfrak{b}_2) + vN(\mathfrak{c}_2) = 2^f$, we need $N(\mathfrak{b}_2)N(\mathfrak{c}_2) \leq 2^f \simeq p$.
- We can solve it as long as $|\Delta| \leq p$.
- **Example:** In CSURF, $|\Delta| = p$.

Steps 3-5: applying Kani's lemma

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

$$\begin{tikzcd}
E'^2 \arrow[rr, "\Phi"] & & E_v^2 & \\
& & E_{\mathfrak{c}_1}^2 \arrow[u, "\Phi_v"'] & E^2 \arrow[l, "\mathfrak{c}_1"'] \\
E_u^2 \arrow[uu, "\Psi"] \arrow[r, "\widetilde{\Phi}_u"] & E_{\mathfrak{b}_1}^2 \arrow[r, "\mathfrak{b}_2"] & E_{\mathfrak{a}}^2 \arrow[u, "\bar{\mathfrak{c}}_2"'] & \\
& E^2 \arrow[u, "\mathfrak{b}_1"] & &
\end{tikzcd}$$

Steps 3-5: applying Kani's lemma

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

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

Steps 3-5: applying Kani's lemma

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

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

- The 2^{f+2} torsion above $\ker(F)$ can be computed by evaluating Φ_u, Φ_v and:

$$\widehat{\varphi}_{\mathfrak{c}_2} \circ \varphi_{\mathfrak{b}_2} = \frac{1}{N(\mathfrak{b}_1)N(\mathfrak{c}_1)} \varphi_{\mathfrak{c}_1} \circ \iota\left(\frac{\overline{\beta}\gamma}{N(\mathfrak{a})}\right) \circ \widehat{\varphi}_{\mathfrak{b}_1}$$

Steps 3-5: applying Kani's lemma

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

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

- The 2^{f+2} torsion above $\ker(F)$ can be computed by evaluating Φ_u, Φ_v and:

$$\widehat{\varphi}_{\mathfrak{c}_2} \circ \varphi_{\mathfrak{b}_2} = \frac{1}{N(\mathfrak{b}_1)N(\mathfrak{c}_1)} \varphi_{\mathfrak{c}_1} \circ \iota\left(\frac{\overline{\beta}\gamma}{N(\mathfrak{a})}\right) \circ \widehat{\varphi}_{\mathfrak{b}_1}$$

- F can then be computed efficiently with theta coordinates [Dar24].

Steps 3-5: applying Kani's lemma

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

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

- The 2^{f+2} torsion above $\ker(F)$ can be computed by evaluating Φ_u, Φ_v and:

$$\widehat{\varphi}_{\mathfrak{c}_2} \circ \varphi_{\mathfrak{b}_2} = \frac{1}{N(\mathfrak{b}_1)N(\mathfrak{c}_1)} \varphi_{\mathfrak{c}_1} \circ \iota\left(\frac{\overline{\beta}\gamma}{N(\mathfrak{a})}\right) \circ \widehat{\varphi}_{\mathfrak{b}_1}$$

- F can then be computed efficiently with theta coordinates [Dar24].
- We can then extract $E_{\mathfrak{a}}$ from the codomain $E_{\mathfrak{a}}^2 \times E'^2$.

Steps 3-5: applying Kani's lemma

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

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

- The 2^{f+2} torsion above $\ker(F)$ can be computed by evaluating Φ_u, Φ_v and:

$$\widehat{\varphi}_{\mathfrak{c}_2} \circ \varphi_{\mathfrak{b}_2} = \frac{1}{N(\mathfrak{b}_1)N(\mathfrak{c}_1)} \varphi_{\mathfrak{c}_1} \circ \iota\left(\frac{\overline{\beta}\gamma}{N(\mathfrak{a})}\right) \circ \widehat{\varphi}_{\mathfrak{b}_1}$$

- F can then be computed efficiently with theta coordinates [Dar24].
- We can then extract $E_{\mathfrak{a}}$ from the codomain $E_{\mathfrak{a}}^2 \times E'^2$.
- The orientation $\iota_{\mathfrak{a}} := (\varphi_{\mathfrak{a}})_* \iota$ on $E_{\mathfrak{a}}$ can be evaluated with F (unnecessary for CSURF).

Implementation for CSURF

Parameter set	Step 1 (s)	Step 2-3 (s)	Steps 4-5 (s)	Total (s)
500	0.097	0.477	0.960	1.534
1000	0.212	1.159	2.838	4.210
1500	1.186	2.853	6.491	10.530
2000	1.675	8.337	11.327	21.339
4000	15.606	52.808	53.463	121.876

Table: SageMath 10.5 timings in sec on Intel Core i5-1235U. Step 1 is the time to solve the norm equation, Steps 2-3 the time to compute all required 1-dimensional isogenies, and Steps 4-5 the time to compute the 4-dimensional isogeny.

Comparison with state of the art

Paper	Impl.	500	1000	1500	2000	4000
SCALLOP [FFK+23]*	C++	35s	12m30s	–	–	–
SCALLOP-HD [CLP24]*	Sage	88s	19m	–	–	–
PEARL-SCALLOP [ABE+24]	C++	30s	58s	12m	–	–
KLaPoTi [PPS24]	Sage	200s	–	–	–	–
	Rust	1.95s	–	–	–	–
PEGASIS (This work)	Sage	1.53s	4.21s	10.5s	21.3s	2m2s

Table: Comparison between PEGASIS and other effective group actions in the literature. The last 5 columns gives the timings corresponding to the different security levels, where s/m gives the number of seconds/minutes in wall-clock time. SCALLOP and SCALLOP-HD are starred because they were measured on a different hardware setup.

Conclusion

To sum up:

- We now have an unrestricted group action which is efficient in practice.
- Made possible with the use of 4-dimensional isogenies.

Conclusion

To sum up:

- We now have an unrestricted group action which is efficient in practice.
- Made possible with the use of 4-dimensional isogenies.

Future works/open questions:

- Need to implement 4-dimensional isogenies in C and/or Rust.
- CSURF was efficient because computations were done over $\mathbb{F}_p$. Need to better understand 4-dimensional isogeny computations over $\mathbb{F}_p$.
- Could we do better in dimension 2?
- What can be done when $|\mathrm{disc}(\mathfrak{O})| \gg p$?

Thanks for listening!

P. Dartois, J. Komada Eriksen, T. B. Fouotsa, A. Herlédan Le Merdy, R. Invernizzi, D. Robert, R. Rueger, F. Vercauteren and B. Wesolowski. PEGASIS: Practical Effective Class Group Action using 4-Dimensional Isogenies. e-Print https://eprint.iacr.org/2023/436

Further algorithmic details

Step 2: computing Φ_u and Φ_v - general case

Goal: Given $u < 2^{e-2}$ odd and an $\mathfrak{O}$-oriented curve (E, ι), compute a u-isogeny $\Phi_u : E^2 \longrightarrow A_u$.

- A method inspired from QFESTA [NO23].

Step 2: computing Φ_u and Φ_v - general case

Goal: Given $u < 2^{e-2}$ odd and an $\mathfrak{O}$-oriented curve (E, ι), compute a u-isogeny $\Phi_u : E^2 \longrightarrow A_u$.

- A method inspired from QFESTA [NO23].
- Let $\Delta := \mathrm{disc}(\mathfrak{O})$ and assume $u(2^f - u) := \Omega(|\Delta| \log(|\Delta|))$ (with $f \leq e-2$).
- Solve:
  $$x^2 + z^2 + |\Delta|(y^2 + t^2) = u(2^f - u),$$
  with $x, y, z, t \in \mathbb{Z}$.

Step 2: computing Φ_u and Φ_v - general case

Goal: Given $u < 2^{e-2}$ odd and an $\mathfrak{O}$-oriented curve (E,ι), compute a u-isogeny $\Phi_u : E^2 \longrightarrow A_u$.

- A method inspired from QFESTA [NO23].
- Let $\Delta := \mathrm{disc}(\mathfrak{O})$ and assume $u(2^f - u) := \Omega(|\Delta|\log(|\Delta|))$ (with $f \leq e-2$).
- Solve:
$$x^2 + z^2 + |\Delta|(y^2 + t^2) = u(2^f - u),$$
with $x, y, z, t \in \mathbb{Z}$.
- Let $\gamma_1 := x + \sqrt{\Delta}y, \gamma_2 := z + \sqrt{\Delta}t \in \mathfrak{O}$ and:
$$\Gamma := \begin{pmatrix} \iota(\gamma_1) & \iota(\overline{\gamma}_2) \\ -\iota(\gamma_2) & \iota(\overline{\gamma}_1) \end{pmatrix} \in \mathrm{End}(E^2).$$
- Then $\widetilde{\Gamma} \circ \Gamma := [N(\gamma_1) + N(\gamma_2)] = u(2^f - u)$.

Step 2: computing Φ_u and Φ_v - general case

- Consider the isogeny diamond:

$$\begin{tikzcd}
A'_u \arrow[r, "\Phi'_u"] & E^2 \\
E^2 \arrow[u, "\Psi'_u"] \arrow[ur, "\Gamma"] \arrow[r, "\Phi_u"] & A_u \arrow[u, "\Psi_u"']
\end{tikzcd}$$

with $\deg(\Phi_u) = u$ and $\deg(\Psi_u) = 2^f - u$.

Step 2: computing Φ_u and Φ_v - general case

- Consider the isogeny diamond:

$$\begin{tikzcd}
A'_u \arrow[r, "\Phi'_u"] & E^2 \\
E^2 \arrow[u, "\Psi'_u"] \arrow[ur, "\Gamma"] \arrow[r, "\Phi_u"'] & A_u \arrow[u, "\Psi_u"']
\end{tikzcd}$$

with $\deg(\Phi_u) = u$ and $\deg(\Psi_u) = 2^f - u$.

- By Kani's lemma, it induces a 2^f-isogeny

$$F_u := \begin{pmatrix} \Phi_u & \widetilde{\Psi}_u \\ -\Psi'_u & \widetilde{\Phi'}_u \end{pmatrix} : E^2 \longrightarrow A_u \times A'_u,$$

Step 2: computing Φ_u and Φ_v - general case

- Consider the isogeny diamond:

$$\begin{tikzcd} A'_u \arrow[r, "\Phi'_u"] & E^2 \\ E^2 \arrow[u, "\Psi'_u"] \arrow[ur, "\Gamma"] \arrow[r, "\Phi_u"'] & A_u \arrow[u, "\Psi_u"'] \end{tikzcd}$$

with $\deg(\Phi_u) = u$ and $\deg(\Psi_u) = 2^f - u$.

- By Kani's lemma, it induces a 2^f-isogeny

$$F_u := \begin{pmatrix} \Phi_u & \widetilde{\Psi}_u \\ -\Psi'_u & \widetilde{\Phi'}_u \end{pmatrix} : E^2 \longrightarrow A_u \times A'_u,$$

- With kernel:

$$\ker(F_u) = \{([u]P, [u]Q, \Gamma(P,Q)) \\ \mid P, Q \in E[2^f]\}.$$

- Knowing ι, we can compute 2^{f+2}-torsion above $\ker(F_u)$ and F_u.

Step 2: computing Φ_u and Φ_v - general case

- Consider the isogeny diamond:

$$\begin{tikzcd} A'_u \arrow[r, "\Phi'_u"] & E^2 \\ E^2 \arrow[u, "\Psi'_u"] \arrow[ur, "\Gamma"] \arrow[r, "\Phi_u"'] & A_u \arrow[u, "\Psi_u"'] \end{tikzcd}$$

with $\deg(\Phi_u) = u$ and $\deg(\Psi_u) = 2^f - u$.

- By Kani's lemma, it induces a 2^f-isogeny

$$F_u := \begin{pmatrix} \Phi_u & \widetilde{\Psi}_u \\ -\Psi'_u & \widetilde{\Phi'}_u \end{pmatrix} : E^2 \longrightarrow A_u \times A'_u,$$

- With kernel:

$$\ker(F_u) = \{([u]P, [u]Q, \Gamma(P,Q)) \\ \mid P, Q \in E[2^f]\}.$$

- Knowing ι, we can compute 2^{f+2}-torsion above $\ker(F_u)$ and F_u.
- F_u represents Φ_u.

Step 1: rerandomization

Issue: What happens when $N(\alpha_1) \ll N(\alpha_2)$?

Step 1: rerandomization

Issue: What happens when $N(\alpha_1) \ll N(\alpha_2)$?

Solution:

- Replace $\mathfrak{a}$ by $\mathfrak{l}\mathfrak{a}$ for a small prime ideal $\mathfrak{l}$.
- Replace E by $E_{\bar{\mathfrak{l}}}$.
- Repeat until $N(\alpha_1) \simeq N(\alpha_2)$ (for the new ideal $\mathfrak{a}$).