Hensel-lifting black-box algorithms and fast trace computation for elliptic-curve endomorphisms

Alessandro Sferlazza
with Lorenz Panny, Damien Robert

Technical University of Munich

Thursday 09 July 2026
ANTS XVII

Motivation: computing traces

Main characters:

- Elliptic curves over a finite field $\mathbb{F}_q$, $\quad q = p^m$.
- An endomorphism $\varphi \in \mathrm{End}(E)$, over $\mathbb{F}_q$ as well.

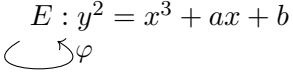
$$E : y^2 = x^3 + ax + b$$

φ

Motivation: computing traces

Main characters:

- Elliptic curves over a finite field $\mathbb{F}_q$, $\quad q = p^m$.
- An endomorphism $\varphi \in \mathrm{End}(E)$, over $\mathbb{F}_q$ as well.



Abstractly, $\mathrm{End}(E) \cong \mathcal{O}$ isomorphic to an order in a quadratic/quaternion algebra over $\mathbb{Q}$.

Problem: how to make this isomorphism explicit?

Motivation: computing traces

Main characters:

- Elliptic curves over a finite field $\mathbb{F}_q$, $q = p^m$.
- An endomorphism $\varphi \in \mathrm{End}(E)$, over $\mathbb{F}_q$ as well.

$$E : y^2 = x^3 + ax + b$$

Abstractly, $\mathrm{End}(E) \cong \mathcal{O}$ isomorphic to an order in a quadratic/quaternion algebra over $\mathbb{Q}$.

Problem: how to make this isomorphism explicit?

Endomorphisms $\varphi \in \mathrm{End}(E)$ are *quadratic integers*: they satisfy

$$\varphi^2 - [t]\varphi + [d] = 0, \qquad t = \mathrm{tr}(\varphi), \quad d = \deg(\varphi).$$

- The degree $d = \deg \varphi$ is usually intrinsic to the representation of φ on a computer.
- Together with the trace $t = \mathrm{tr}\, \varphi$ $\rightsquigarrow$ complete description of $\varphi \in \mathcal{O}$. ✓

Motivation: computing traces

Main characters:

- Elliptic curves over a finite field $\mathbb{F}_q$, $\quad q = p^m$.
- An endomorphism $\varphi \in \mathrm{End}(E)$, over $\mathbb{F}_q$ as well.

$$E : y^2 = x^3 + ax + b$$

Abstractly, $\mathrm{End}(E) \cong \mathcal{O}$ isomorphic to an order in a quadratic/quaternion algebra over $\mathbb{Q}$.
Problem: how to make this isomorphism explicit?

Endomorphisms $\varphi \in \mathrm{End}(E)$ are *quadratic integers*: they satisfy

$$\varphi^2 - [t]\varphi + [d] = 0, \qquad t = \mathrm{tr}(\varphi), \quad d = \deg(\varphi).$$

- The degree $d = \deg \varphi$ is usually intrinsic to the representation of φ on a computer.
- Together with the trace $t = \mathrm{tr}\,\varphi$ $\rightsquigarrow$ complete description of $\varphi \in \mathcal{O}$. ✓

Cost of computing $\mathrm{tr}\,\varphi$ on a curve $E/\mathbb{F}_{p^m}$, where φ is stored via $n = O(\log pd)$ field elements

[BCEMP18] `arXiv 1804.04063`	$\widetilde{O}(n^7)$	bit ops
[MPSW25] `arXiv 2501.16321`	$O(n^4 \log(n)^2)$	bit ops
this work	$\widetilde{O}(n^2)$	bit ops

Inspiration: point counting algorithms

Classical problem in number theory: Given $E/\mathbb{F}_q$ with $q = p^m$, compute $\#E(\mathbb{F}_q)$.

Inspiration: point counting algorithms

Classical problem in number theory: Given $E/\mathbb{F}_q$ with $q = p^m$, compute $\#E(\mathbb{F}_q)$.

How to solve? Take $\pi\colon (x, y) \mapsto (x^q, y^q)$ the q-Frobenius endomorphism on E.

- Satisfies quadratic equation
$$\Phi(\pi) = \pi^2 - t\pi + q = 0, \qquad t = \mathrm{tr}(\pi)$$
- The number of q-rational points is
$$\#E(\mathbb{F}_q) = q + 1 - t.$$

Inspiration: point counting algorithms

Classical problem in number theory: Given $E/\mathbb{F}_q$ with $q = p^m$, compute $\#E(\mathbb{F}_q)$.

How to solve? Take $\pi\colon (x, y) \mapsto (x^q, y^q)$ the q-Frobenius endomorphism on E.

- Satisfies quadratic equation

$$\Phi(\pi) = \pi^2 - t\pi + q = 0, \qquad t = \mathrm{tr}(\pi)$$

- The number of q-rational points is

$$\#E(\mathbb{F}_q) = q + 1 - t.$$

Point counting $\longleftrightarrow$ computing the trace of Frobenius.

Inspiration: point counting algorithms

Classical problem in number theory: Given $E/\mathbb{F}_q$ with $q = p^m$, compute $\#E(\mathbb{F}_q)$.

How to solve? Take $\pi\colon (x, y) \mapsto (x^q, y^q)$ the q-Frobenius endomorphism on E.

- Satisfies quadratic equation

$$\Phi(\pi) = \pi^2 - t\pi + q = 0, \qquad t = \mathrm{tr}(\pi)$$

- The number of q-rational points is

$$\#E(\mathbb{F}_q) = q + 1 - t.$$

Point counting $\longleftrightarrow$ computing the trace of Frobenius.

- SEA algorithm (Schoof 1985, Elkies, Atkin 1990s, ...):
  - computes $\mathrm{tr}(\pi) \bmod \ell$ for many small primes ℓ, combine via Chinese Remainder Theorem

$\rightsquigarrow$ state-of-the-art trace computation algorithms are variants of SEA.

Inspiration: point counting algorithms

Classical problem in number theory: Given $E/\mathbb{F}_q$ with $q = p^m$, compute $\#E(\mathbb{F}_q)$.

How to solve? Take $\pi\colon (x, y) \mapsto (x^q, y^q)$ the q-Frobenius endomorphism on E.

- Satisfies quadratic equation
$$\Phi(\pi) = \pi^2 - t\pi + q = 0, \qquad t = \mathrm{tr}(\pi)$$
- The number of q-rational points is
$$\#E(\mathbb{F}_q) = q + 1 - t.$$

Point counting $\longleftrightarrow$ computing the trace of Frobenius.

- SEA algorithm (Schoof 1985, Elkies, Atkin 1990s, ...):
  - computes $\mathrm{tr}(\pi) \bmod \ell$ for many small primes ℓ, combine via Chinese Remainder Theorem

  $\rightsquigarrow$ state-of-the-art trace computation algorithms are variants of SEA.
- Satoh's algorithm (Satoh 2000, ...): different approach $\nleftrightsquigarrow$ we adapt this one!
  - p-adic lifting of curves and endomorphisms
  - action of morphisms on invariant differentials

Ingredient #1: p-adic lifting, algorithmically

Let $q = p^m$. We can define a ring $\mathbb{Z}_q$ of characteristic 0, with a natural projection on $\mathbb{F}_q$:

$$\mathbb{Z}_q = W(\mathbb{F}_q) \;\approx\; \{\sum_{i=0}^{\infty} a_i\, p^i \mid a_i \in \mathbb{F}_q\} \qquad \sum a_i p^i \mapsto a_0 \in \mathbb{F}_q$$

Ingredient #1: p-adic lifting, algorithmically

Let $q = p^m$. We can define a ring $\mathbb{Z}_q$ of characteristic 0, with a natural projection on $\mathbb{F}_q$:

$$\mathbb{Z}_q = W(\mathbb{F}_q) \;\approx\; \Big\{\sum_{i=0}^{\infty} a_i\, p^i \mid a_i \in \mathbb{F}_q\Big\} \qquad \sum a_i p^i \mapsto a_0 \in \mathbb{F}_q$$

Hensel's lemma:
(stated for $p > 2$)

Let $F\colon \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ a polynomial map, a a non-degenerate root $\bmod\, p$.

Ingredient #1: p-adic lifting, algorithmically

Let $q = p^m$. We can define a ring $\mathbb{Z}_q$ of characteristic 0, with a natural projection on $\mathbb{F}_q$:

$$\mathbb{Z}_q = W(\mathbb{F}_q) \;\approx\; \Big\{\sum_{i=0}^{\infty} a_i\, p^i \mid a_i \in \mathbb{F}_q\Big\} \qquad \sum a_i p^i \mapsto a_0 \in \mathbb{F}_q$$

Hensel's lemma:
(stated for $p > 2$)

Let $F\colon \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ a polynomial map, a a non-degenerate root $\bmod p$.
We can lift a uniquely to $\widetilde{a} \in \mathbb{Z}_q^n$ with $\widetilde{a} \equiv a \pmod p$ and $F(\widetilde{a}) = 0$.

Ingredient #1: p-adic lifting, algorithmically

Let $q = p^m$. We can define a ring $\mathbb{Z}_q$ of characteristic 0, with a natural projection on $\mathbb{F}_q$:

$$\mathbb{Z}_q = W(\mathbb{F}_q) \;\approx\; \Big\{\sum_{i=0}^{\infty} a_i\, p^i \mid a_i \in \mathbb{F}_q\Big\} \qquad \sum a_i p^i \mapsto a_0 \in \mathbb{F}_q$$

Hensel's lemma:
(stated for $p > 2$)

Let $F\colon \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ a polynomial map, a a non-degenerate root $\bmod p$.
We can lift a uniquely to $\widetilde{a} \in \mathbb{Z}_q^n$ with $\widetilde{a} \equiv a \pmod p$ and $F(\widetilde{a}) = 0$.

generalizes to F rational functions on $\mathrm{Frac}(\mathbb{Z}_q) = \mathbb{Q}_q \to \mathbb{Q}_q$!

Ingredient #1: p-adic lifting, algorithmically

Let $q = p^m$. We can define a ring $\mathbb{Z}_q$ of characteristic 0, with a natural projection on $\mathbb{F}_q$:

$$\mathbb{Z}_q = W(\mathbb{F}_q) \;\approx\; \{\sum_{i=0}^{\infty} a_i\, p^i \mid a_i \in \mathbb{F}_q\} \qquad \sum a_i p^i \mapsto a_0 \in \mathbb{F}_q$$

Hensel's lemma:
(stated for $p > 2$)

Let $F\colon \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ a polynomial map, a a non-degenerate root $\bmod p$.
We can lift a uniquely to $\widetilde{a} \in \mathbb{Z}_q^n$ with $\widetilde{a} \equiv a \pmod p$ and $F(\widetilde{a}) = 0$.

generalizes to F rational functions on $\mathrm{Frac}(\mathbb{Z}_q) = \mathbb{Q}_q \to \mathbb{Q}_q$!

Concretely, use Newton iterations:

$\cdots$
$a + pv + p^2 v_2 \bmod p^4$
$a + pv \bmod p^2$
$a \bmod p$

Ingredient #1: p-adic lifting, algorithmically

Let $q = p^m$. We can define a ring $\mathbb{Z}_q$ of characteristic 0, with a natural projection on $\mathbb{F}_q$:

$$\mathbb{Z}_q = W(\mathbb{F}_q) \;\approx\; \Big\{\sum_{i=0}^{\infty} a_i\, p^i \mid a_i \in \mathbb{F}_q\Big\} \qquad \sum a_i p^i \mapsto a_0 \in \mathbb{F}_q$$

Hensel's lemma:
(stated for $p > 2$)

Let $F\colon \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ a polynomial map, a a non-degenerate root $\bmod p$.
We can lift a uniquely to $\widetilde{a} \in \mathbb{Z}_q^n$ with $\widetilde{a} \equiv a \pmod p$ and $F(\widetilde{a}) = 0$.

generalizes to F rational functions on $\mathrm{Frac}(\mathbb{Z}_q) = \mathbb{Q}_q \to \mathbb{Q}_q$!

Concretely, use Newton iterations:

$$F(a + p^k v) = F(a) + p^k \mathrm{D}F(a) \cdot v + O(p^{2k}) \overset{?}{=} 0 \pmod{p^{2k}}$$

Linear equation: if $F(a), \mathrm{D}F(a)$ are known, solve for v.

$\cdots$
$a + pv + p^2 v_2 \bmod p^4$
$a + pv \bmod p^2$
$a \bmod p$

Ingredient #1: p-adic lifting, algorithmically

Let $q = p^m$. We can define a ring $\mathbb{Z}_q$ of characteristic 0, with a natural projection on $\mathbb{F}_q$:

$$\mathbb{Z}_q = W(\mathbb{F}_q) \;\approx\; \Big\{\sum_{i=0}^{\infty} a_i\, p^i \mid a_i \in \mathbb{F}_q\Big\} \qquad \sum a_i p^i \mapsto a_0 \in \mathbb{F}_q$$

Hensel's lemma: (stated for $p > 2$)

> Let $F\colon \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ a polynomial map, a a non-degenerate root $\bmod p$.
> We can lift a uniquely to $\widetilde{a} \in \mathbb{Z}_q^n$ with $\widetilde{a} \equiv a \pmod p$ and $F(\widetilde{a}) = 0$.

generalizes to F rational functions on $\mathrm{Frac}(\mathbb{Z}_q) = \mathbb{Q}_q \to \mathbb{Q}_q$!

Concretely, use Newton iterations:

$$F(a + p^k v) = F(a) + p^k \mathrm{D}F(a) \cdot v + O(p^{2k}) \overset{?}{=} 0 \pmod{p^{2k}}$$

Linear equation: if $F(a), \mathrm{D}F(a)$ are known, solve for v.

$\cdots$
$a + pv + p^2 v_2 \bmod p^4$
$a + pv \bmod p^2$
$a \bmod p$

No need to know $F, \mathrm{D}F$ explicitly! ✓ a black-box algo $a \mapsto F(a)$ is sufficient:

$$\begin{array}{rcll} a & \mapsto & F(a) & + O(p^{2k}) \\ a + e_i p^k & \mapsto & F(a) \;+ \partial_i F(a) p^k & + O(p^{2k}) \end{array}$$

$n+1$ black-box calls
$\rightsquigarrow$ get $F(a), \mathrm{D}F(a)$

Canonical lifts: Serre–Tate's theory

We can lift roots of polynomials. But would you like to lift some Elliptic Curves?

Canonical lifts: Serre–Tate's theory

We can lift roots of polynomials. But would you like to lift some Elliptic Curves?

Theorem (Serre–Tate–Grothendieck–Messing–Gross + tweaks)
Let $E/\mathbb{F}_q$ an elliptic curve, $\varphi \in \mathrm{End}_{\mathbb{F}_q}(E) \setminus \mathbb{Z}$ a separable endomorphism.
There is a $\approx$unique lift $(\widetilde{E}, \widetilde{\varphi})$ over $\mathbb{Z}_q$ with $\widetilde{\varphi} \in \mathrm{End}(\widetilde{E})$.

Canonical lifts: Serre–Tate's theory

We can lift roots of polynomials. But would you like to lift some ELLIPTIC CURVES?

Theorem (Serre–Tate–Grothendieck–Messing–Gross + tweaks)
Let $E/\mathbb{F}_q$ an elliptic curve, $\varphi \in \mathrm{End}_{\mathbb{F}_q}(E)\setminus\mathbb{Z}$ a separable endomorphism.
There is a $\approx$unique lift $(\widetilde{E},\widetilde{\varphi})$ over $\mathbb{Z}_q$ with $\widetilde{\varphi}\in\mathrm{End}(\widetilde{E})$.

Special case: when E is ordinary, lifting E with the dual $\widehat{\pi}$ of its Frobenius $\rightsquigarrow$ canonical lift $(\widetilde{E},\widetilde{\widehat{\pi}})$ satisfying $\mathrm{End}(\widetilde{E})\cong\mathrm{End}(E)$.

Canonical lifts: Serre–Tate's theory

We can lift roots of polynomials. But would you like to lift some Elliptic Curves?

Theorem (Serre–Tate–Grothendieck–Messing–Gross + tweaks)
Let $E/\mathbb{F}_q$ an elliptic curve, $\varphi \in \mathrm{End}_{\mathbb{F}_q}(E) \setminus \mathbb{Z}$ a separable endomorphism.
There is a $\approx$unique lift $(\widetilde{E}, \widetilde{\varphi})$ over $\mathbb{Z}_q$ with $\widetilde{\varphi} \in \mathrm{End}(\widetilde{E})$.

Special case: when E is ordinary, lifting E with the dual $\widehat{\pi}$ of its Frobenius $\rightsquigarrow$ canonical lift $(\widetilde{E}, \widetilde{\widehat{\pi}})$ satisfying $\mathrm{End}(\widetilde{E}) \cong \mathrm{End}(E)$.

The $\bmod\ p$ projection induces a ring hom $\mathrm{End}(\widetilde{E}) \hookrightarrow \mathrm{End}(E)$.
$\rightsquigarrow$ characteristic polynomials ($\rightsquigarrow$ traces too!) are preserved by lifting.

Canonical lifts: Serre–Tate's theory

We can lift roots of polynomials. But would you like to lift some Elliptic Curves?

Theorem (Serre–Tate–Grothendieck–Messing–Gross + tweaks)
Let $E/\mathbb{F}_q$ an elliptic curve, $\varphi \in \mathrm{End}_{\mathbb{F}_q}(E) \setminus \mathbb{Z}$ a separable endomorphism.
There is a $\approx$unique lift $(\widetilde{E}, \widetilde{\varphi})$ over $\mathbb{Z}_q$ with $\widetilde{\varphi} \in \mathrm{End}(\widetilde{E})$.

Special case: when E is ordinary, lifting E with the dual $\widehat{\pi}$ of its Frobenius
$\rightsquigarrow$ canonical lift $(\widetilde{E}, \widetilde{\widehat{\pi}})$ satisfying $\mathrm{End}(\widetilde{E}) \cong \mathrm{End}(E)$.

The $\bmod p$ projection induces a ring hom $\mathrm{End}(\widetilde{E}) \hookrightarrow \mathrm{End}(E)$.
$\rightsquigarrow$ characteristic polynomials ($\rightsquigarrow$ traces too!) are preserved by lifting.

Can we lift (E, φ) algorithmically?

- Lifting curve = lifting coefficients. $\quad y^2 = x^3 + ax + b \quad \rightsquigarrow \quad y^2 = x^3 + \widetilde{a}x + \widetilde{b}$

Canonical lifts: Serre–Tate's theory

We can lift roots of polynomials. But would you like to lift some Elliptic Curves?

Theorem (Serre–Tate–Grothendieck–Messing–Gross + tweaks)
Let $E/\mathbb{F}_q$ an elliptic curve, $\varphi \in \mathrm{End}_{\mathbb{F}_q}(E) \setminus \mathbb{Z}$ a separable endomorphism.
There is a $\approx$unique lift $(\widetilde{E}, \widetilde{\varphi})$ over $\mathbb{Z}_q$ with $\widetilde{\varphi} \in \mathrm{End}(\widetilde{E})$.

Special case: when E is ordinary, lifting E with the dual $\widehat{\pi}$ of its Frobenius
$\rightsquigarrow$ canonical lift $(\widetilde{E}, \widetilde{\widehat{\pi}})$ satisfying $\mathrm{End}(\widetilde{E}) \cong \mathrm{End}(E)$.

The $\bmod p$ projection induces a ring hom $\mathrm{End}(\widetilde{E}) \hookrightarrow \mathrm{End}(E)$.
$\rightsquigarrow$ characteristic polynomials ($\rightsquigarrow$ traces too!) are preserved by lifting.

Can we lift (E, φ) algorithmically?

- Lifting curve = lifting coefficients. $\quad y^2 = x^3 + ax + b \quad \rightsquigarrow \quad y^2 = x^3 + \widetilde{a}x + \widetilde{b}$
- Lifting φ = ? $\quad$ ... depends on the algorithmic representation of φ.

Lifting isogenies and endomorphisms

Let $\varphi\colon E \to E'$ be a separable isogeny over k:

Lifting isogenies and endomorphisms

Let $\varphi\colon E \to E'$ be a separable isogeny over k:

Isogeny chains: When φ has smooth degree $N = \prod_i \ell_i$, then it factors into small steps:

$$E = E_0 \xrightarrow{\varphi_1} E_1 \xrightarrow{\varphi_2} E_2 \xrightarrow{\varphi_3} \dots \xrightarrow{\varphi_r} E_n = E', \qquad \varphi : E \to E' \text{ (over the arc)}$$

For small steps, we can use explicit formulas $\rightsquigarrow$ lifting φ is easy.

Lifting isogenies and endomorphisms

Let $\varphi\colon E \to E'$ be a separable isogeny over k:

Isogeny chains: When φ has smooth degree $N = \prod_i \ell_i$, then it factors into small steps:

$$E = E_0 \xrightarrow{\varphi_1} E_1 \xrightarrow{\varphi_2} E_2 \xrightarrow{\varphi_3} \dots \xrightarrow{\varphi_r} E_n = E' \qquad (\varphi : E \to E')$$

For small steps, we can use explicit formulas $\rightsquigarrow$ lifting φ is easy.

Endomorphisms: We can require domain and codomain to be isomorphic: there must be $u \in k$

$$E_0 : y^2 = x^3 + ax + b, \qquad E_n : y^2 = x^3 + (u^4 a)x + (u^6 b)$$

$\rightsquigarrow$ another polynomial constraint we can lift!

Lifting isogenies and endomorphisms

Let $\varphi\colon E \to E'$ be a separable isogeny over k:

Isogeny chains: When φ has smooth degree $N = \prod_i \ell_i$, then it factors into small steps:

$$E = E_0 \xrightarrow{\varphi_1} E_1 \xrightarrow{\varphi_2} E_2 \xrightarrow{\varphi_3} \dots \xrightarrow{\varphi_r} E_n = E' \qquad (E_0 \xrightarrow{\varphi} E_n)$$

For small steps, we can use explicit formulas $\rightsquigarrow$ lifting φ is easy.

Endomorphisms: We can require domain and codomain to be isomorphic: there must be $u \in k$

$$E_0 : y^2 = x^3 + ax + b, \qquad E_n : y^2 = x^3 + (u^4 a)x + (u^6 b)$$

$\rightsquigarrow$ another polynomial constraint we can lift! This constraint makes the lift $\approx$unique.

Lifting isogenies and endomorphisms

Let $\varphi\colon E \to E'$ be a separable isogeny over k:

Isogeny chains: When φ has smooth degree $N = \prod_i \ell_i$, then it factors into small steps:

$$E = E_0 \xrightarrow{\varphi_1} E_1 \xrightarrow{\varphi_2} E_2 \xrightarrow{\varphi_3} \dots \xrightarrow{\varphi_r} E_n = E' \qquad (\varphi : E \to E')$$

For small steps, we can use explicit formulas $\rightsquigarrow$ lifting φ is easy.

Endomorphisms: We can require domain and codomain to be isomorphic: there must be $u \in k$

$$E_0 : y^2 = x^3 + ax + b, \qquad E_n : y^2 = x^3 + (u^4 a)x + (u^6 b)$$

$\rightsquigarrow$ another polynomial constraint we can lift! This constraint makes the lift $\approx$unique.

Non-smooth degree: If $\deg \varphi$ is odd $\rightsquigarrow$ we can "embed" the 1-dimensional φ in dim 2:
$\exists\ 2^e$-isogeny of principally polarized abelian surfaces $\Phi\colon E \times E_a \to E' \times E_b$ s.t.

$$\varphi \quad \text{equals the composition} \quad E \hookrightarrow E \times E_a \xrightarrow{\Phi} E' \times E_b \twoheadrightarrow E'.$$

Lifting isogenies and endomorphisms

Let $\varphi\colon E \to E'$ be a separable isogeny over k:

Isogeny chains: When φ has smooth degree $N = \prod_i \ell_i$, then it factors into small steps:

$$E = E_0 \xrightarrow{\varphi_1} E_1 \xrightarrow{\varphi_2} E_2 \xrightarrow{\varphi_3} \dots \xrightarrow{\varphi_r} E_n = E' \qquad (\varphi_1 \circ \varphi_2 \circ \dots \circ \varphi_r = \varphi)$$

For small steps, we can use explicit formulas $\rightsquigarrow$ lifting φ is easy.

Endomorphisms: We can require domain and codomain to be isomorphic: there must be $u \in k$

$$E_0 : y^2 = x^3 + ax + b, \qquad E_n : y^2 = x^3 + (u^4 a)x + (u^6 b)$$

$\rightsquigarrow$ another polynomial constraint we can lift! This constraint makes the lift $\approx$unique.

Non-smooth degree: If $\deg \varphi$ is odd $\rightsquigarrow$ we can "embed" the 1-dimensional φ in dim 2:
$\exists\ 2^e$-isogeny of principally polarized abelian surfaces $\Phi\colon E \times E_a \to E' \times E_b$ s.t.

$$\varphi \quad \text{equals the composition} \quad E \hookrightarrow E \times E_a \xrightarrow{\Phi} E' \times E_b \twoheadrightarrow E'.$$

$\rightsquigarrow$ we get a *smooth-degree* isogeny chain (in dim 2), easy to lift
$\rightsquigarrow$ + quirky algebraic ($\implies$ liftable!) constraints coming from the 2-dim embedding.

Ingredient #2: differential scaling factors

An elliptic curve is a 1-dimensional algebraic group
$\rightsquigarrow$ admits a translation-invariant differential $\omega_E = dx/y$.

Ingredient #2: differential scaling factors

An elliptic curve is a 1-dimensional algebraic group
$\rightsquigarrow$ admits a translation-invariant differential $\omega_E = dx/y$.

Morphisms $\varphi\colon E \to E'$ interact with differentials:

$$\varphi^*\omega_{E'} = \omega_{E'} \circ \varphi = c_\varphi \cdot \omega_E, \qquad c_\varphi \in k.$$

where as a rational map, φ looks like:

$$\varphi(x, y) = \big(f(x),\ c_\varphi^{-1} \cdot y \cdot f'(x)\big).$$

Ingredient #2: differential scaling factors

An elliptic curve is a 1-dimensional algebraic group
$\rightsquigarrow$ admits a translation-invariant differential $\omega_E = dx/y$.

Morphisms $\varphi\colon E \to E'$ interact with differentials:

$$\varphi^*\omega_{E'} = \omega_{E'} \circ \varphi = c_\varphi \cdot \omega_E, \qquad c_\varphi \in k.$$

where as a rational map, φ looks like:

$$\varphi(x, y) = \big(f(x),\ c_\varphi^{-1} \cdot y \cdot f'(x)\big).$$

✓ c_φ is easy to compute: two (black-box) isogeny evaluations give $f(x), f'(x)$.

Ingredient #2: differential scaling factors

An elliptic curve is a 1-dimensional algebraic group
$\rightsquigarrow$ admits a translation-invariant differential $\omega_E = dx/y$.

Morphisms $\varphi\colon E \to E'$ interact with differentials:

$$\varphi^*\omega_{E'} = \omega_{E'} \circ \varphi = c_\varphi \cdot \omega_E, \qquad c_\varphi \in k.$$

where as a rational map, φ looks like:

$$\varphi(x, y) = \big(f(x),\ c_\varphi^{-1} \cdot y \cdot f'(x)\big).$$

$\varphi \in \mathrm{End}(E) \mapsto c_\varphi \in k$
is a ring homomorphism:

$$\varphi^2 - [t]\varphi + [d] = 0$$
$$\rightsquigarrow$$
$$c_\varphi^2 - t c_\varphi + d = 0.$$

✓ c_φ is easy to compute: two (black-box) isogeny evaluations give $f(x), f'(x)$.

Ingredient #2: differential scaling factors

An elliptic curve is a 1-dimensional algebraic group
$\rightsquigarrow$ admits a translation-invariant differential $\omega_E = dx/y$.

Morphisms $\varphi\colon E \to E'$ interact with differentials:

$$\varphi^*\omega_{E'} = \omega_{E'} \circ \varphi = c_\varphi \cdot \omega_E, \qquad c_\varphi \in k.$$

where as a rational map, φ looks like:

$$\varphi(x, y) = \big(f(x),\ c_\varphi^{-1} \cdot y \cdot f'(x)\big).$$

$\varphi \in \mathrm{End}(E) \mapsto c_\varphi \in k$
is a ring homomorphism:

$$\varphi^2 - [t]\varphi + [d] = 0$$
$$\rightsquigarrow$$
$$c_\varphi^2 - t c_\varphi + d = 0.$$

✓ c_φ is easy to compute: two (black-box) isogeny evaluations give $f(x), f'(x)$.

c_φ is quadratic too, with the same trace as φ $\quad\rightsquigarrow \mathrm{tr}(\varphi) = c_\varphi + d/c_\varphi$.

Ingredient #2: differential scaling factors

An elliptic curve is a 1-dimensional algebraic group
$\rightsquigarrow$ admits a translation-invariant differential $\omega_E = dx/y$.

Morphisms $\varphi\colon E \to E'$ interact with differentials:

$$\varphi^*\omega_{E'} = \omega_{E'} \circ \varphi = c_\varphi \cdot \omega_E, \qquad c_\varphi \in k.$$

where as a rational map, φ looks like:

$$\varphi(x, y) = \big(f(x),\ c_\varphi^{-1} \cdot y \cdot f'(x)\big).$$

$\varphi \in \mathrm{End}(E) \mapsto c_\varphi \in k$
is a ring homomorphism:

$$\varphi^2 - [t]\varphi + [d] = 0$$
$$\rightsquigarrow$$
$$c_\varphi^2 - t c_\varphi + d = 0.$$

✓ c_φ is easy to compute: two (black-box) isogeny evaluations give $f(x), f'(x)$.

c_φ is quadratic too, with the same trace as φ $\quad\rightsquigarrow \mathrm{tr}(\varphi) = c_\varphi + d/c_\varphi$.

× If $c_\varphi \in \mathbb{F}_q$, then $c_\varphi + d/c_\varphi$ is the image of an integer in $\mathbb{F}_q$: $\quad\rightsquigarrow$ only get $\mathrm{tr}(\varphi) \bmod p$.

Ingredient #2: differential scaling factors

An elliptic curve is a 1-dimensional algebraic group
$\rightsquigarrow$ admits a translation-invariant differential $\omega_E = dx/y$.

Morphisms $\varphi\colon E \to E'$ interact with differentials:

$$\varphi^*\omega_{E'} = \omega_{E'} \circ \varphi = c_\varphi \cdot \omega_E, \qquad c_\varphi \in k.$$

where as a rational map, φ looks like:

$$\varphi(x, y) = \big(f(x),\ c_\varphi^{-1} \cdot y \cdot f'(x)\big).$$

$\varphi \in \mathrm{End}(E) \mapsto c_\varphi \in k$
is a ring homomorphism:

$$\varphi^2 - [t]\varphi + [d] = 0$$
$$\rightsquigarrow$$
$$c_\varphi^2 - t c_\varphi + d = 0.$$

✓ c_φ is easy to compute: two (black-box) isogeny evaluations give $f(x), f'(x)$.

💡 c_φ is quadratic too, with the same trace as φ $\quad\rightsquigarrow \mathrm{tr}(\varphi) = c_\varphi + d/c_\varphi$.

✗ If $c_\varphi \in \mathbb{F}_q$, then $c_\varphi + d/c_\varphi$ is the image of an integer in $\mathbb{F}_q$: $\quad\rightsquigarrow$ only get $\mathrm{tr}(\varphi) \bmod p$.

✓ Replace $(E/\mathbb{F}_q, \varphi)$ with a lift $(\widetilde{E}, \widetilde{\varphi})$ over a characteristic-0 ring R.

Ingredient #2: differential scaling factors

An elliptic curve is a 1-dimensional algebraic group
$\rightsquigarrow$ admits a translation-invariant differential $\omega_E = dx/y$.

Morphisms $\varphi\colon E \to E'$ interact with differentials:

$$\varphi^*\omega_{E'} = \omega_{E'} \circ \varphi = c_\varphi \cdot \omega_E, \qquad c_\varphi \in k.$$

where as a rational map, φ looks like:

$$\varphi(x, y) = \big(f(x),\ c_\varphi^{-1} \cdot y \cdot f'(x)\big).$$

$\varphi \in \mathrm{End}(E) \mapsto c_\varphi \in k$
is a ring homomorphism:

$$\varphi^2 - [t]\varphi + [d] = 0$$
$$\rightsquigarrow$$
$$c_\varphi^2 - t c_\varphi + d = 0.$$

✓ c_φ is easy to compute: two (black-box) isogeny evaluations give $f(x), f'(x)$.

💡 c_φ is quadratic too, with the same trace as φ $\quad\rightsquigarrow \mathrm{tr}(\varphi) = c_\varphi + d/c_\varphi$.

✗ If $c_\varphi \in \mathbb{F}_q$, then $c_\varphi + d/c_\varphi$ is the image of an integer in $\mathbb{F}_q$: $\quad\rightsquigarrow$ only get $\mathrm{tr}(\varphi) \bmod p$.

✓ Replace $(E/\mathbb{F}_q, \varphi)$ with a lift $(\widetilde{E}, \widetilde{\varphi})$ over a characteristic-0 ring R.
Lifting preserves algebraic relations: $\mathrm{tr}(\widetilde{\varphi}) = \mathrm{tr}(\varphi)$.
$\rightsquigarrow$ Now $\mathrm{tr}(\widetilde{\varphi})$ lies in $\mathbb{Z} \hookrightarrow R$, i.e., an actual integer!

Recap: Computing traces: the steps

Main problem: given (E, φ) over $\mathbb{F}_q$ with $\varphi \in \mathrm{End}(E)$, compute $\mathrm{tr}\,\varphi$.

Recap: Computing traces: the steps

Main problem: given (E, φ) over $\mathbb{F}_q$ with $\varphi \in \mathrm{End}(E)$, compute $\mathrm{tr}\,\varphi$.

- Give an upper bound for the trace: if $\deg \varphi = d$, then $|\,\mathrm{tr}\,\varphi\,| \leq \sqrt{4d}$ ($\approx$ Hasse-Weil)

$\rightsquigarrow$ p-adic precision goal: need the smallest k s.t. $p^k > 2 \cdot \sqrt{4d}$

Recap: Computing traces: the steps

Main problem: given (E, φ) over $\mathbb{F}_q$ with $\varphi \in \mathrm{End}(E)$, compute $\mathrm{tr}\,\varphi$.

- Give an upper bound for the trace: if $\deg \varphi = d$, then $|\,\mathrm{tr}\,\varphi\,| \leq \sqrt{4d}$ ($\approx$ Hasse-Weil)

$\rightsquigarrow$ p-adic precision goal: need the smallest k s.t. $p^k > 2 \cdot \sqrt{4d}$

- Hensel-lift (E, φ) to precision $\geq k$ $\quad\rightsquigarrow\quad$ $(\widetilde{E}, \widetilde{\varphi})$ defined over $\mathbb{Z}_q/p^k\mathbb{Z}_q$

Recap: Computing traces: the steps

Main problem: given (E, φ) over $\mathbb{F}_q$ with $\varphi \in \mathrm{End}(E)$, compute $\mathrm{tr}\,\varphi$.

- Give an upper bound for the trace: if $\deg \varphi = d$, then $|\,\mathrm{tr}\,\varphi\,| \leq \sqrt{4d}$ ($\approx$ Hasse-Weil)

$\rightsquigarrow$ p-adic precision goal: need the smallest k s.t. $p^k > 2 \cdot \sqrt{4d}$

- Hensel-lift (E, φ) to precision $\geq k$ $\quad\rightsquigarrow\quad$ $(\widetilde{E}, \widetilde{\varphi})$ defined over $\mathbb{Z}_q/p^k\mathbb{Z}_q$
- Compute the scaling factor of $\widetilde{\varphi}$ as seen above $\rightsquigarrow$ $c_{\widetilde{\varphi}} \in \mathbb{Z}_q/p^k\mathbb{Z}_q$.

Recap: Computing traces: the steps

Main problem: given (E, φ) over $\mathbb{F}_q$ with $\varphi \in \mathrm{End}(E)$, compute $\mathrm{tr}\,\varphi$.

- Give an upper bound for the trace: if $\deg \varphi = d$, then $|\,\mathrm{tr}\,\varphi\,| \leq \sqrt{4d}$ ($\approx$ Hasse-Weil)

$\rightsquigarrow$ p-adic precision goal: need the smallest k s.t. $p^k > 2 \cdot \sqrt{4d}$

- Hensel-lift (E, φ) to precision $\geq k$ $\quad\rightsquigarrow\quad$ $(\widetilde{E}, \widetilde{\varphi})$ defined over $\mathbb{Z}_q/p^k\mathbb{Z}_q$
- Compute the scaling factor of $\widetilde{\varphi}$ as seen above $\rightsquigarrow$ $c_{\widetilde{\varphi}} \in \mathbb{Z}_q/p^k\mathbb{Z}_q$.
- Obtain the trace modulo p^k: $\quad t \bmod p^k = c_{\widetilde{\varphi}} + d/c_{\widetilde{\varphi}} \in \mathbb{Z}/p^k\mathbb{Z}$

Recap: Computing traces: the steps

Main problem: given (E, φ) over $\mathbb{F}_q$ with $\varphi \in \mathrm{End}(E)$, compute $\mathrm{tr}\,\varphi$.

- Give an upper bound for the trace: if $\deg \varphi = d$, then $|\,\mathrm{tr}\,\varphi\,| \leq \sqrt{4d}$ ($\approx$ Hasse-Weil)

$\rightsquigarrow$ p-adic precision goal: need the smallest k s.t. $p^k > 2 \cdot \sqrt{4d}$

- Hensel-lift (E, φ) to precision $\geq k$ $\quad\rightsquigarrow\quad$ $(\widetilde{E}, \widetilde{\varphi})$ defined over $\mathbb{Z}_q/p^k\mathbb{Z}_q$
- Compute the scaling factor of $\widetilde{\varphi}$ as seen above $\rightsquigarrow$ $c_{\widetilde{\varphi}} \in \mathbb{Z}_q/p^k\mathbb{Z}_q$.
- Obtain the trace modulo p^k: $\quad t \bmod p^k = c_{\widetilde{\varphi}} + d/c_{\widetilde{\varphi}} \in \mathbb{Z}/p^k\mathbb{Z}$

  Choose repr t in $\mathbb{Z} \cap [-p^k/2, p^k/2]$. From the bound, $\mathrm{tr}(\varphi) = t \in \mathbb{Z}$.

Recap: Computing traces: the steps

Main problem: given (E, φ) over $\mathbb{F}_q$ with $\varphi \in \mathrm{End}(E)$, compute $\mathrm{tr}\,\varphi$.

- Give an upper bound for the trace: if $\deg \varphi = d$, then $|\,\mathrm{tr}\,\varphi\,| \leq \sqrt{4d}$ ($\approx$ Hasse-Weil)

$\rightsquigarrow$ p-adic precision goal: need the smallest k s.t. $p^k > 2 \cdot \sqrt{4d}$

- Hensel-lift (E, φ) to precision $\geq k$ $\quad\rightsquigarrow\quad$ $(\widetilde{E}, \widetilde{\varphi})$ defined over $\mathbb{Z}_q/p^k\mathbb{Z}_q$
- Compute the scaling factor of $\widetilde{\varphi}$ as seen above $\rightsquigarrow$ $c_{\widetilde{\varphi}} \in \mathbb{Z}_q/p^k\mathbb{Z}_q$.
- Obtain the trace modulo p^k: $\quad t \bmod p^k = c_{\widetilde{\varphi}} + d/c_{\widetilde{\varphi}} \in \mathbb{Z}/p^k\mathbb{Z}$

  Choose repr t in $\mathbb{Z} \cap [-p^k/2, p^k/2]$. From the bound, $\mathrm{tr}(\varphi) = t \in \mathbb{Z}$.

Bulk of the computation: lifting to the highest precision.
i.e., $\approx$ re-evaluating the isogeny a handful of times at precision k.

Complexity (in bit operations) when φ isogeny chain:
(# steps) $\times$ (cost of a step) $\times$ (cost of multiplications in highest precision: $\mathbb{Z}_q/p^k\mathbb{Z}_q$) $=$

$$n \cdot O(\ell) \cdot \widetilde{O}(k) = \widetilde{O}(n^2), \qquad k \approx \log d \approx n, \quad \ell = O(1)$$

Summary

What we've got:

- An algorithm to lift isogeny representations
  - Actually works on general arithmetic computations
    with sage implementation! ✓
    https://gitlab.inria.fr/isogenies/lifting_traces

Summary

What we've got:

- An algorithm to lift isogeny representations
  - Actually works on general arithmetic computations
    with sage implementation! ✓
    https://gitlab.inria.fr/isogenies/lifting_traces
- Byproduct: constructive existence proof of Serre–Tate lifts of elliptic curves

Summary

What we've got:

- An algorithm to lift isogeny representations
  - Actually works on general arithmetic computations
    with sage implementation! ✓
    https://gitlab.inria.fr/isogenies/lifting_traces
- Byproduct: constructive existence proof of Serre–Tate lifts of elliptic curves
- Algorithm to compute scaling factor of any isogeny

Summary

What we've got:

- An algorithm to lift isogeny representations
  - Actually works on general arithmetic computations
    with sage implementation! ✓
    https://gitlab.inria.fr/isogenies/lifting_traces
- Byproduct: constructive existence proof of Serre–Tate lifts of elliptic curves
- Algorithm to compute scaling factor of any isogeny
- Trace computation in the case of smooth chains, HD, radical isogenies
  - with sage implementation! ✓

Summary

What we've got:

- An algorithm to lift isogeny representations
  - Actually works on general arithmetic computations
    with sage implementation! ✓
    https://gitlab.inria.fr/isogenies/lifting_traces
- Byproduct: constructive existence proof of Serre–Tate lifts of elliptic curves
- Algorithm to compute scaling factor of any isogeny
- Trace computation in the case of smooth chains, HD, radical isogenies
  - with sage implementation! ✓
- Eprint: https://eprint.iacr.org/2026/137

Summary

What we've got:

- An algorithm to lift isogeny representations
  - Actually works on general arithmetic computations
    with sage implementation! ✓
    https://gitlab.inria.fr/isogenies/lifting_traces
- Byproduct: constructive existence proof of Serre–Tate lifts of elliptic curves
- Algorithm to compute scaling factor of any isogeny
- Trace computation in the case of smooth chains, HD, radical isogenies
  - with sage implementation! ✓
- Eprint: https://eprint.iacr.org/2026/137

Thank you for your attention :D

Questions?

Bonus details: Lifting a single isogeny step

Let $E_i : y^2 = x^3 + a_i x + b_i$ over $\mathbb{F}_q$ for $i = 0, 1$, $\varphi : E_0 \to E_1$ an ℓ-isogeny, ℓ small.

First lift the domain curve to $\widetilde{E_0} : y^2 = x^3 + \widetilde{a_0} x + \widetilde{b_0}$.

(sqrt)Vélu: Say $\ker \varphi = \langle P \rangle$. How do we lift to $\widetilde{P}$?

- $\widetilde{P}$ is of order ℓ: lift root of division polynomial $\psi_{\widetilde{E},\ell}(x)$, get $x_P \rightsquigarrow \widetilde{x_P}$

  
  better: $\psi_{\widetilde{E},\ell}$ depends polynomially on the curve: $\psi_\ell(a, b, x)$.
  $\rightsquigarrow$ compute partial derivatives wrt a, b, x $\rightsquigarrow$ linear dependency between $\widetilde{a_0}, \widetilde{b_0}, \widetilde{x_P}$

- $\widetilde{P}$ lies on $\widetilde{E}$: lift root of defining polynomial $y^2 - x^3 - \widetilde{a}x - \widetilde{b}$
  $\rightsquigarrow$ linear dependency between $\widetilde{a}, \widetilde{b}, \widetilde{x_P}, \widetilde{y_P}$

j-invariants: if we've got $(E_0, j(E_1))$, instead of a kernel point?

$\rightsquigarrow$ BMSS, 2008, computes E_1 with the correct j, and $\varphi : E_0 \to E_1$.

✓ The computation is algebraic: rational fn of the input

⚠ problems when $j(E_0)$ or $j(E_1)$ are in $\{0, 1728\}$... as usual

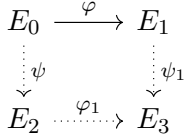
- φ non unique
- non-smooth map $j(E) \to E$

Note both this algo and Vélu compute a normalized φ (i.e. scaling factor = 1)

Lifting higher-dimensional representations

Embedding Lemma Let $\varphi\colon E_0 \to E_1$ be separable of odd degree d.
One can build a 2D polarized 2^e-isogeny $\Phi\colon E_0 \times E_3 \to E_1 \times E_2$ s.t.

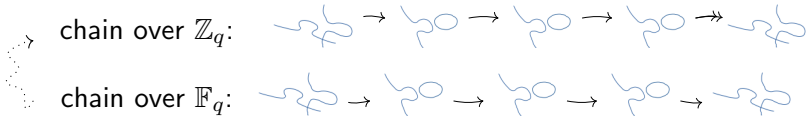
$$\varphi \quad \text{equals the composition} \quad E_0 \hookrightarrow E_0 \times E_3 \xrightarrow{\Phi} E_1 \times E_2 \twoheadrightarrow E_1.$$



The tuple (E_0, E_3, P, Q) gives an efficient representation for φ,
where $\langle P, Q\rangle = \ker \Phi \subseteq (E_0 \times E_3)[2^e]$. To lift the 2D-rep, make sure:

- both P and Q lie on the domain product surface and have order 2^e
- the 2D isogeny with kernel P, Q lands on a product
- (extra constraint if we're lifting an endo): $j(E_0) = j(E_1)$

} all algebraic



Same strategy generalizes to 4D, 8D representations.