Abelian varieties in the theta model and applications to cryptography

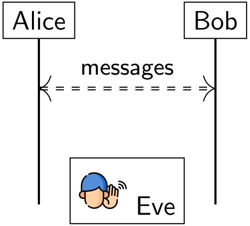
Candidate: Alessandro Sferlazza
Advisors: Benjamin Smith (INRIA Saclay, LIX Ecole Polytechnique, France)
Davide Lombardo (UniPi)

University of Pisa
Laurea Magistrale in Matematica

27 September 2024

Context: public key cryptography

Achieving secure communication over an insecure channel

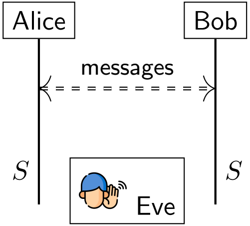


Context: public key cryptography

Achieving secure communication over an insecure channel

Example Diffie–Hellman key exchange

Goal: A, B establish a shared secret S.



Context: public key cryptography

Achieving secure communication over an insecure channel

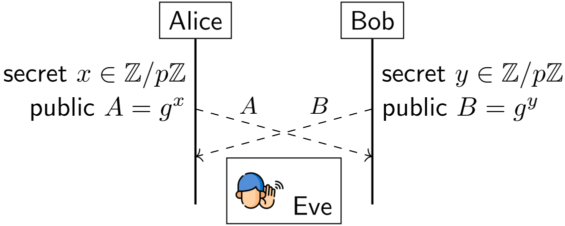
Example Diffie–Hellman key exchange

Goal: A, B establish a shared secret S.

Parameters:

- p large prime,
- $G = \langle g \rangle$ of order p.

example: $G \leq \mathbb{F}_q^\times$.



Context: public key cryptography

Achieving secure communication over an insecure channel

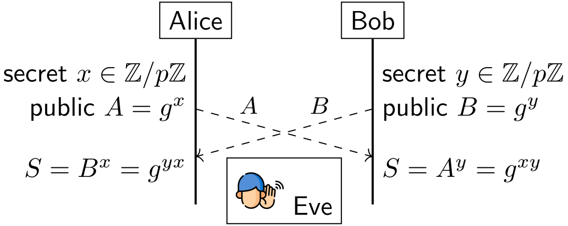
Example Diffie–Hellman key exchange

Goal: A, B establish a shared secret S.

Parameters:

- p large prime,
- $G = \langle g \rangle$ of order p.

example: $G \leq \mathbb{F}_q^\times$.



Context: public key cryptography

Achieving secure communication over an insecure channel

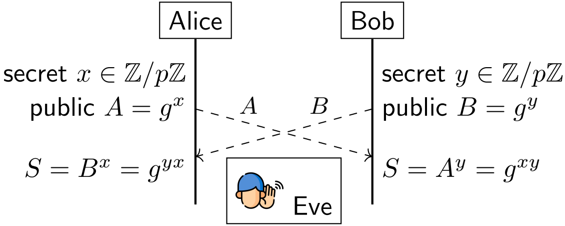
Example Diffie–Hellman key exchange

Goal: A, B establish a shared secret S.

Parameters:

- p large prime,
- $G = \langle g \rangle$ of order p.

example: $G \leq \mathbb{F}_q^\times$.



Hard problem (Discrete Log Problem) given g, g^a, find a.

Security Eve recovers a secret key $\Leftrightarrow$ she solves DLP.

Context: public key cryptography

Achieving secure communication over an insecure channel

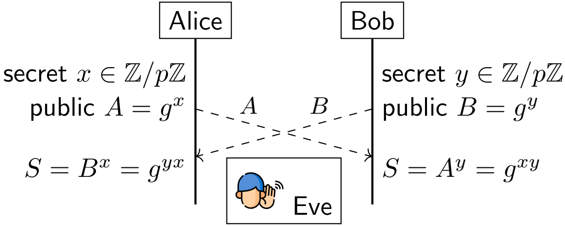
Example Diffie–Hellman key exchange

Goal: A, B establish a shared secret S.

Parameters:

- p large prime,
- $G = \langle g \rangle$ of order p.

example: $G \leq \mathbb{F}_q^\times$.

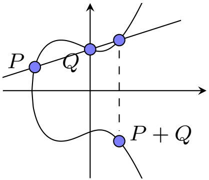


Hard problem (Discrete Log Problem) given g, g^a, find a.

Security Eve recovers a secret key $\Leftrightarrow$ she solves DLP.

Often G comes from elliptic curves:

- Defined by $E : Y^2 Z = X^3 + aXZ^2 + bZ^3$ with $a, b \in \mathbb{F}_q$
- $E(\overline{\mathbb{F}}_q) = \{(X : Y : Z) \in \mathbb{P}^2(\overline{\mathbb{F}}_q)$ satisfying eq$\}$ abelian group



Context: public key cryptography

Achieving secure communication over an insecure channel

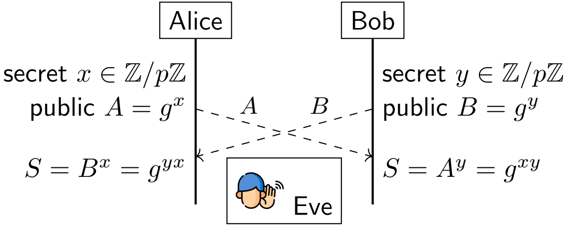
Example Diffie–Hellman key exchange

Goal: A, B establish a shared secret S.

Parameters:

- p large prime,
- $G = \langle g \rangle$ of order p.

example: $G \leq \mathbb{F}_q^\times$.

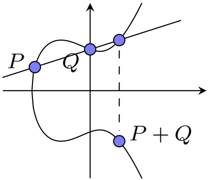


Hard problem (Discrete Log Problem) given g, g^a, find a.

Security Eve recovers a secret key $\Leftrightarrow$ she solves DLP.

Often G comes from elliptic curves:

- Defined by $E : Y^2Z = X^3 + aXZ^2 + bZ^3$ with $a, b \in \mathbb{F}_q$
- $E(\overline{\mathbb{F}}_q) = \{(X : Y : Z) \in \mathbb{P}^2(\overline{\mathbb{F}}_q)$ satisfying eq$\}$ abelian group
- If $G \leq E(\mathbb{F}_q)$ has large prime order, DLP is exponentially hard $O(\sqrt{\#G})$.



Isogeny-based cryptography

Premise Elliptic curve cryptography ubiquitous in today's internet.
Security $\longleftrightarrow$ hardness of order-p DLP: fastest algorithms are exponential-time in $\log p$.

Isogeny-based cryptography

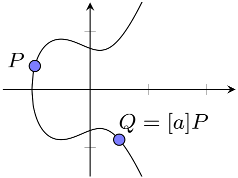
Premise Elliptic curve cryptography ubiquitous in today's internet.
Security $\longleftrightarrow$ hardness of order-p DLP: fastest algorithms are exponential-time in $\log p$.

Problem Shor's quantum algorithm solves DLP in poly$(\log p)$ $\rightsquigarrow$ ECC not quantum-secure.
$\rightsquigarrow$ need for post-quantum cryptography.
$\triangleright$ New paradigms: lattices, error correcting codes, ..., isogenies of elliptic curves.

Isogeny-based cryptography

Premise Elliptic curve cryptography ubiquitous in today's internet.
Security $\longleftrightarrow$ hardness of order-p DLP: fastest algorithms are exponential-time in $\log p$.

Problem Shor's quantum algorithm solves DLP in poly$(\log p)$ $\rightsquigarrow$ ECC not quantum-secure.
$\rightsquigarrow$ need for post-quantum cryptography.
▷ New paradigms: lattices, error correcting codes, ..., isogenies of elliptic curves.



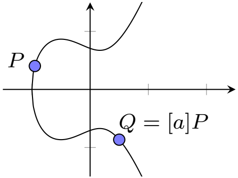
Elliptic curve:

- projective algebraic variety
- abelian group

Isogeny-based cryptography

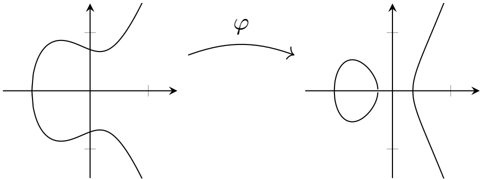
Premise Elliptic curve cryptography ubiquitous in today's internet.
Security $\longleftrightarrow$ hardness of order-p DLP: fastest algorithms are exponential-time in $\log p$.

Problem Shor's quantum algorithm solves DLP in poly$(\log p)$ $\rightsquigarrow$ ECC not quantum-secure.
$\rightsquigarrow$ need for post-quantum cryptography.
▷ New paradigms: lattices, error correcting codes, ..., isogenies of elliptic curves.



Elliptic curve:

- projective algebraic variety
- abelian group



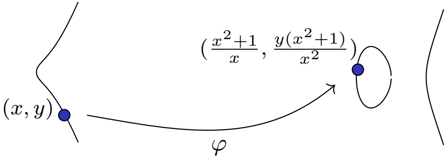
Isogeny:

- morphism of algebraic varieties (defined by rational maps)
- group homomorphism with finite kernel

Isogenies: definitions and examples

Isogeny: "nice" map $E_0 \xrightarrow{\varphi} E_1$:

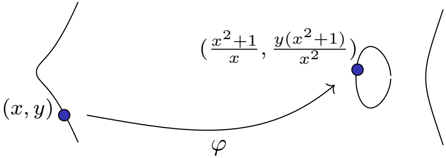
- defined by rational maps
- group homomorphism with finite kernel



Isogenies: definitions and examples

Isogeny: "nice" map $E_0 \xrightarrow{\varphi} E_1$:

- defined by rational maps
- group homomorphism with finite kernel

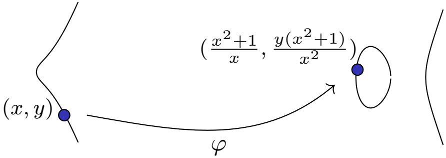


Def $\deg \varphi = x$-degree of its rational maps $\overset{\text{when } p \nmid \deg \varphi}{=} \# \ker \varphi$

Isogenies: definitions and examples

Isogeny: “nice” map $E_0 \xrightarrow{\varphi} E_1$:

- defined by rational maps
- group homomorphism with finite kernel



Def $\deg\varphi = x$-degree of its rational maps $\overset{\text{when } p\nmid\deg\varphi}{=} \#\ker\varphi$

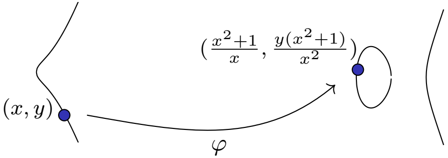
Examples $E : Y^2Z = X^3 + aXZ^2 + bZ^3$ defined over $\mathbb{F}_q$.

- Frobenius $\pi_q\colon E \to E,\quad (X : Y : Z) \mapsto (X^q : Y^q : Z^q)\quad \deg\pi_q = q$
- Scalar multiplication $[n]\colon E \to E,\quad P \mapsto \underbrace{P + P + \cdots + P}_{n \text{ times}} = nP\quad \deg[n] = n^2$

Isogenies: definitions and examples

Isogeny: "nice" map $E_0 \xrightarrow{\varphi} E_1$:

- defined by rational maps
- group homomorphism with finite kernel



<u>Def</u> $\deg\varphi = x$-degree of its rational maps $\overset{\text{when } p\nmid\deg\varphi}{=} \#\ker\varphi$

<u>Examples</u> $E : Y^2Z = X^3 + aXZ^2 + bZ^3$ defined over $\mathbb{F}_q$.

- Frobenius $\quad \pi_q\colon E \to E, \quad (X:Y:Z) \mapsto (X^q:Y^q:Z^q) \quad \deg\pi_q = q$
- Scalar multiplication $\quad [n]\colon E \to E, \quad P \mapsto \underbrace{P + P + \cdots + P}_{n \text{ times}} = nP \quad \deg[n] = n^2$

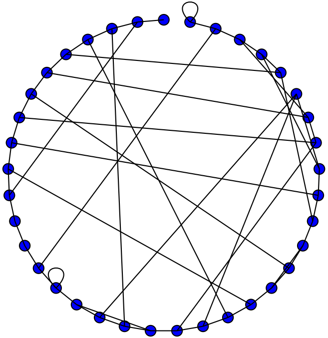
<u>Decomposing isogenies</u> Factor $\deg\varphi = \prod_{i=1}^r \ell_i$ into primes.
Isogenies can be factored too: $\varphi = \varphi_1 \circ \ldots \circ \varphi_r$, $\deg\varphi_i = \ell_i$.

- We can study isogenies of prime degree.

$$\begin{tikzcd} E_0 \arrow[r, "\varphi_1"] \arrow[rrrr, bend left, "\varphi"] & E^{(1)} \arrow[r, "\varphi_2"] & E^{(2)} \arrow[r, "\varphi_3"] & \ldots \arrow[r, "\varphi_r"] & E_1 \end{tikzcd}$$

A hard problem with isogenies

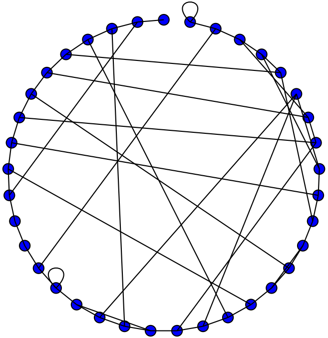
Fact If $\varphi\colon E_0 \to E_1$ is an isogeny, then there is $\widehat{\varphi}\colon E_1 \to E_0$.
"Being isogenous" is an equivalence relation. $\rightsquigarrow$ isogeny graphs.



Vertices: elliptic curves (up to $\cong$)
Edges: isogenies of fixed prime degree

A hard problem with isogenies

Fact If $\varphi\colon E_0 \to E_1$ is an isogeny, then there is $\widehat{\varphi}\colon E_1 \to E_0$.
"Being isogenous" is an equivalence relation. $\rightsquigarrow$ isogeny graphs.

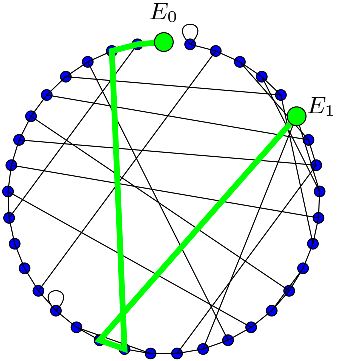


Vertices: elliptic curves (up to $\cong$)
Edges: isogenies of fixed prime degree

Hard problem (Isogeny problem) Given isogenous curves E_0, E_1, find a φ connecting them. $\longleftrightarrow$ find a path.

A hard problem with isogenies

Fact If $\varphi\colon E_0 \to E_1$ is an isogeny, then there is $\widehat{\varphi}\colon E_1 \to E_0$.
"Being isogenous" is an equivalence relation. $\rightsquigarrow$ isogeny graphs.

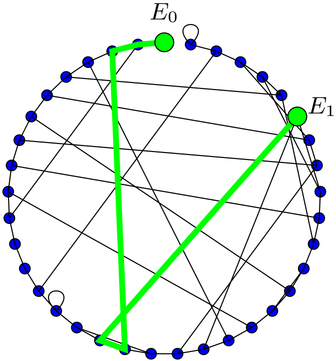


Vertices: elliptic curves (up to $\cong$)
Edges: isogenies of fixed prime degree

Hard problem (Isogeny problem) Given isogenous curves E_0, E_1, find a φ connecting them. $\longleftrightarrow$ find a path.
Cryptography Fix E_0. Consider $\varphi\colon E_0 \to E_1$
Secret key: isogeny path φ. Public key: destination curve E_1.

A hard problem with isogenies

Fact If $\varphi\colon E_0 \to E_1$ is an isogeny, then there is $\widehat{\varphi}\colon E_1 \to E_0$.
"Being isogenous" is an equivalence relation. $\rightsquigarrow$ isogeny graphs.



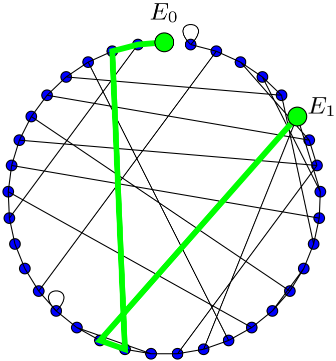
Vertices: elliptic curves (up to $\cong$)
Edges: isogenies of fixed prime degree

Hard problem (Isogeny problem) Given isogenous curves E_0, E_1, find a φ connecting them. $\longleftrightarrow$ find a path.
Cryptography Fix E_0. Consider $\varphi\colon E_0 \to E_1$
Secret key: isogeny path φ. Public key: destination curve E_1.

- The supersingular isogeny problem is exponentially hard even for quantum computers.

A hard problem with isogenies

Fact If $\varphi \colon E_0 \to E_1$ is an isogeny, then there is $\widehat{\varphi} \colon E_1 \to E_0$.
"Being isogenous" is an equivalence relation. $\rightsquigarrow$ isogeny graphs.



Vertices: elliptic curves (up to $\cong$)
Edges: isogenies of fixed prime degree

Hard problem (Isogeny problem) Given isogenous curves E_0, E_1, find a φ connecting them. $\longleftrightarrow$ find a path.
Cryptography Fix E_0. Consider $\varphi \colon E_0 \to E_1$
Secret key: isogeny path φ. Public key: destination curve E_1.

- The supersingular isogeny problem is exponentially hard even for quantum computers.
- Security of isogeny-based protocols $\longleftrightarrow$ hardness of isogeny problem.
- Efficiency $\longleftrightarrow$ fast evaluation of isogenies

SQIsign identification scheme

Basis of SQIsign signature: isogeny-based candidate for post-quantum standardization

Setup Public parameter E_0. Alice's keys: (secret isogeny $\varphi_{\mathrm{sk}} \colon E_0 \to E_{\mathrm{pk}}$, public E_{pk}).
Goal Alice proves her identity to Bob, showing she knows φ_{sk}.

$$E_0 \xrightarrow{\varphi_{\mathrm{sk}}} E_{\mathrm{pk}}$$

SQIsign identification scheme

Basis of SQIsign signature: isogeny-based candidate for post-quantum standardization

Setup Public parameter E_0. Alice's keys: (secret isogeny $\varphi_{\mathrm{sk}} \colon E_0 \to E_{\mathrm{pk}}$, public E_{pk}).
Goal Alice proves her identity to Bob, showing she knows φ_{sk}.

$$
\begin{tikzcd}
E_0 \arrow[r, dashed, "\varphi_{\mathrm{sk}}"] \arrow[d, dashed, "\varphi_{\mathrm{comm}}"'] & E_{\mathrm{pk}} \\
E_{\mathrm{comm}} &
\end{tikzcd}
$$

1. Alice sends E_{comm}

SQIsign identification scheme

Basis of SQIsign signature: isogeny-based candidate for post-quantum standardization

Setup Public parameter E_0. Alice's keys: (secret isogeny $\varphi_{\mathrm{sk}} \colon E_0 \to E_{\mathrm{pk}}$, public E_{pk}).
Goal Alice proves her identity to Bob, showing she knows φ_{sk}.

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "\varphi_{\mathrm{sk}}"] \arrow[d, dashed, "\varphi_{\mathrm{comm}}"'] & E_{\mathrm{pk}} \arrow[d, "\varphi_{\mathrm{chal}}"] \\
E_{\mathrm{comm}} & E_{\mathrm{chal}}
\end{tikzcd}$$

1. Alice sends E_{comm}
2. Bob sends $\varphi_{\mathrm{chal}}, E_{\mathrm{chal}}$

SQIsign identification scheme

Basis of SQIsign signature: isogeny-based candidate for post-quantum standardization

Setup Public parameter E_0. Alice's keys: (secret isogeny $\varphi_{\mathrm{sk}} \colon E_0 \to E_{\mathrm{pk}}$, public E_{pk}).
Goal Alice proves her identity to Bob, showing she knows φ_{sk}.

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "\varphi_{\mathrm{sk}}"] \arrow[d, dashed, "\varphi_{\mathrm{comm}}"'] & E_{\mathrm{pk}} \arrow[d, "\varphi_{\mathrm{chal}}"] \\
E_{\mathrm{comm}} \arrow[r, "\varphi_{\mathrm{resp}}"'] & E_{\mathrm{chal}}
\end{tikzcd}$$

1. Alice sends E_{comm}
2. Bob sends $\varphi_{\mathrm{chal}}, E_{\mathrm{chal}}$
3. Alice sends φ_{resp}

SQIsign identification scheme

Basis of SQIsign signature: isogeny-based candidate for post-quantum standardization

Setup Public parameter E_0. Alice's keys: (secret isogeny $\varphi_{\mathrm{sk}} : E_0 \to E_{\mathrm{pk}}$, public E_{pk}).
Goal Alice proves her identity to Bob, showing she knows φ_{sk}.

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "\varphi_{\mathrm{sk}}"] \arrow[d, dashed, "\varphi_{\mathrm{comm}}"'] & E_{\mathrm{pk}} \arrow[d, "\varphi_{\mathrm{chal}}"] \\
E_{\mathrm{comm}} \arrow[r, "\varphi_{\mathrm{resp}}"'] & E_{\mathrm{chal}}
\end{tikzcd}$$

1. Alice sends E_{comm}
2. Bob sends $\varphi_{\mathrm{chal}}, E_{\mathrm{chal}}$
3. Alice sends φ_{resp}

Performance
vs current PQ standards

Protocol	signature size (B)	signing time (kcycles)
ML-DSA (std)	2420	333
SQIsign	177	5,669,000

SQIsign identification scheme

Basis of SQIsign signature: isogeny-based candidate for post-quantum standardization

Setup Public parameter E_0. Alice's keys: (secret isogeny $\varphi_{\mathrm{sk}}\colon E_0 \to E_{\mathrm{pk}}$, public E_{pk}).
Goal Alice proves her identity to Bob, showing she knows φ_{sk}.

$$\begin{tikzcd}
 & E_0 \arrow[r, dashed, "\varphi_{\mathrm{sk}}"] \arrow[dl, dashed, "\varphi_{\mathrm{comm}}"'] & E_{\mathrm{pk}} \arrow[dr, "\varphi_{\mathrm{chal}}"] & \\
E_{\mathrm{comm}} \arrow[rrr, "\varphi_{\mathrm{resp}}"'] & & & E_{\mathrm{chal}}
\end{tikzcd}$$

1. Alice sends E_{comm}
2. Bob sends $\varphi_{\mathrm{chal}}, E_{\mathrm{chal}}$
3. Alice sends φ_{resp}

Performance
vs current PQ standards

Protocol	signature size (B)	signing time (kcycles)
ML-DSA (std)	2420	333
SQIsign	177	5,669,000

Why so slow? Bottleneck: computing isogenies of large prime degree

- We can choose (e.g.) $\deg \varphi_{\mathrm{chal}} = 2^e$: decomposable in small 2-isogenies.
- Then $\deg \varphi_{\mathrm{comm}}, \deg \varphi_{\mathrm{resp}}$ still have large prime factors.

How to represent an isogeny?

Computing isogenies Consider $\varphi\colon E_0 \to E_1$ with $\deg \varphi = \ell$ prime.

How to represent an isogeny?

Computing isogenies Consider $\varphi \colon E_0 \to E_1$ with $\deg \varphi = \ell$ prime. We want an algo :

- Input: Generators of $\ker \varphi \leq E_0$, possibly other info
- Output:
  - recover info about the codomain E_1
  - evaluate φ on any point $P \in E_0$

How to represent an isogeny?

Computing isogenies Consider $\varphi\colon E_0 \to E_1$ with $\deg \varphi = \ell$ prime. We want an algo :

- Input: Generators of $\ker \varphi \leq E_0$, possibly other info
- Output:
  - recover info about the codomain E_1
  - evaluate φ on any point $P \in E_0$

Goal Compute an isogeny φ of prime degree ℓ

- Small ℓ: Vélu's formulas give explicit rational maps from kernel points: $O(\ell)$
- Large ℓ: faster algo VéluSqrt (2020) runs in $O(\sqrt{\ell})$.

How to represent an isogeny?

Computing isogenies Consider $\varphi\colon E_0 \to E_1$ with $\deg \varphi = \ell$ prime. We want an algo :

- Input: Generators of $\ker \varphi \leq E_0$, possibly other info
- Output:
  - recover info about the codomain E_1
  - evaluate φ on any point $P \in E_0$

Goal Compute an isogeny φ of prime degree ℓ

- Small ℓ: Vélu's formulas give explicit rational maps from kernel points: $O(\ell)$
- Large ℓ: faster algo VéluSqrt (2020) runs in $O(\sqrt{\ell})$.

Problem This is already slow for $\ell \approx$ thousands.

How to represent an isogeny?

Computing isogenies Consider $\varphi\colon E_0 \to E_1$ with $\deg \varphi = \ell$ prime. We want an algo :

- Input: Generators of $\ker \varphi \leq E_0$, possibly other info
- Output:
  - recover info about the codomain E_1
  - evaluate φ on any point $P \in E_0$

Goal Compute an isogeny φ of prime degree ℓ

- Small ℓ: Vélu's formulas give explicit rational maps from kernel points: $O(\ell)$
- Large ℓ: faster algo VéluSqrt (2020) runs in $O(\sqrt{\ell})$.

Problem This is already slow for $\ell \approx$ thousands.

Solution (Castryck–Decru, 2022) Higher-dimensional representation, $O(\log^2 \ell) \leftarrow$ in my thesis

Kani's lemma

Fact $\varphi\colon E_0 \to E_1,\ \deg\varphi = m.$ There is a unique dual $\widehat{\varphi}\colon E_1 \to E_0,\ \varphi \circ \widehat{\varphi} = [m]$.
Fact Define the m-torsion $E[m] := \ker([m])$. If $p \nmid m$ then $E[m] \cong (\mathbb{Z}/m\mathbb{Z})^2$.

Kani's lemma

Fact $\varphi\colon E_0 \to E_1,\ \deg\varphi = m$. There is a unique dual $\widehat{\varphi}\colon E_1 \to E_0,\ \varphi\circ\widehat{\varphi} = [m]$.
Fact Define the m-torsion $E[m] := \ker([m])$. If $p \nmid m$ then $E[m] \cong (\mathbb{Z}/m\mathbb{Z})^2$.

Lemma ([Kani, 1997])
Fix $\varphi\colon E_0 \to E_1$ of degree m. Let $N > m$, suppose $N - m = a^2$ with $\gcd(m, a) = 1$.
The matrix $\Psi = \begin{pmatrix} [a] & -\widehat{\varphi} \\ \varphi & [a] \end{pmatrix} : E_0 \times E_1 \to E_0 \times E_1$ is an isogeny in dimension 2.

Kani's lemma

Fact $\varphi\colon E_0 \to E_1,\ \deg\varphi = m$. There is a unique dual $\widehat{\varphi}\colon E_1 \to E_0,\ \varphi\circ\widehat{\varphi} = [m]$.
Fact Define the m-torsion $E[m] := \ker([m])$. If $p \nmid m$ then $E[m] \cong (\mathbb{Z}/m\mathbb{Z})^2$.

Lemma ([Kani, 1997])
Fix $\varphi\colon E_0 \to E_1$ of degree m. Let $N > m$, suppose $N - m = a^2$ with $\gcd(m,a) = 1$.
The matrix $\Psi = \begin{pmatrix} [a] & -\widehat{\varphi} \\ \varphi & [a] \end{pmatrix} : E_0 \times E_1 \to E_0 \times E_1$ is an isogeny in dimension 2.

Proof

- Defined by rational maps ✓
- Homomorphism of abelian groups ✓

Kani's lemma

Fact $\varphi\colon E_0 \to E_1,\ \deg\varphi = m$. There is a unique dual $\widehat{\varphi}\colon E_1 \to E_0,\ \varphi \circ \widehat{\varphi} = [m]$.
Fact Define the m-torsion $E[m] := \ker([m])$. If $p \nmid m$ then $E[m] \cong (\mathbb{Z}/m\mathbb{Z})^2$.

Lemma ([Kani, 1997])
Fix $\varphi\colon E_0 \to E_1$ of degree m. Let $N > m$, suppose $N - m = a^2$ with $\gcd(m, a) = 1$.
The matrix $\Psi = \begin{pmatrix} [a] & -\widehat{\varphi} \\ \varphi & [a] \end{pmatrix} : E_0 \times E_1 \to E_0 \times E_1$ is an isogeny in dimension 2.

Proof

- Defined by rational maps ✓
- Homomorphism of abelian groups ✓
- Finite kernel:
  Define the *dual* $\widehat{\Psi} = \begin{pmatrix} [a] & \widehat{\varphi} \\ -\varphi & [a] \end{pmatrix}$. We have $\Psi \circ \widehat{\Psi} = \begin{pmatrix} [a^2+m] & 0 \\ 0 & [a^2+m] \end{pmatrix} = [N]$.

Kani's lemma

Fact $\varphi\colon E_0 \to E_1$, $\deg\varphi = m$. There is a unique dual $\widehat{\varphi}\colon E_1 \to E_0$, $\varphi \circ \widehat{\varphi} = [m]$.
Fact Define the m-torsion $E[m] := \ker([m])$. If $p \nmid m$ then $E[m] \cong (\mathbb{Z}/m\mathbb{Z})^2$.

Lemma ([Kani, 1997])
Fix $\varphi\colon E_0 \to E_1$ of degree m. Let $N > m$, suppose $N - m = a^2$ with $\gcd(m, a) = 1$.
The matrix $\Psi = \begin{pmatrix} [a] & -\widehat{\varphi} \\ \varphi & [a] \end{pmatrix} : E_0 \times E_1 \to E_0 \times E_1$ is an isogeny in dimension 2.

Proof

- Defined by rational maps ✓
- Homomorphism of abelian groups ✓
- Finite kernel:
  Define the *dual* $\widehat{\Psi} = \begin{pmatrix} [a] & \widehat{\varphi} \\ -\varphi & [a] \end{pmatrix}$. We have $\Psi \circ \widehat{\Psi} = \begin{pmatrix} [a^2+m] & 0 \\ 0 & [a^2+m] \end{pmatrix} = [N]$.
  - We say Ψ has reduced degree N
  - $\ker\Psi \subseteq \ker([N]) = E_0[N] \times E_1[N]$ is finite. ✓

Kani's lemma

Fact $\varphi\colon E_0 \to E_1,\ \deg\varphi = m$. There is a unique dual $\widehat{\varphi}\colon E_1 \to E_0,\ \varphi\circ\widehat{\varphi} = [m]$.
Fact Define the m-torsion $E[m] := \ker([m])$. If $p \nmid m$ then $E[m] \cong (\mathbb{Z}/m\mathbb{Z})^2$.

Lemma ([Kani, 1997])
Fix $\varphi\colon E_0 \to E_1$ of degree m. Let $N > m$, suppose $N - m = a^2$ with $\gcd(m,a) = 1$.
The matrix $\Psi = \begin{pmatrix} [a] & -\widehat{\varphi} \\ \varphi & [a] \end{pmatrix} : E_0 \times E_1 \to E_0 \times E_1$ is an isogeny in dimension 2.

Proof

- Defined by rational maps ✓
- Homomorphism of abelian groups ✓
- Finite kernel:
  Define the *dual* $\widehat{\Psi} = \begin{pmatrix} [a] & \widehat{\varphi} \\ -\varphi & [a] \end{pmatrix}$. We have $\Psi\circ\widehat{\Psi} = \begin{pmatrix} [a^2+m] & 0 \\ 0 & [a^2+m] \end{pmatrix} = [N]$.
  - We say Ψ has reduced degree N
  - $\ker\Psi \subseteq \ker([N]) = E_0[N] \times E_1[N]$ is finite. ✓
    More precisely, $\ker\Psi = \{\widehat{\Psi}\begin{pmatrix} P \\ 0 \end{pmatrix} \mid P \in E_0[N]\}$.

Higher-dimensional (HD) representation

Goal Computing isogeny $\varphi\colon E_0 \to E_1$ of large prime degree ℓ.

Higher-dimensional (HD) representation

Goal Computing isogeny $\varphi\colon E_0 \to E_1$ of large prime degree ℓ.

If we find[1] $N = 2^n = \ell + a^2$ with $\ell \nmid a$,

$\Psi = \begin{pmatrix} [a] & -\widehat{\varphi} \\ \varphi & [a] \end{pmatrix}$ is a 2-dimensional isogeny of reduced degree 2^n (a 2^n-isogeny)

$\rightsquigarrow$ $(*, \varphi(Q)) = \Psi(Q, 0)$ for all Q. If we can compute Ψ, we can compute φ

[1] $N - \ell = a^2$ is restrictive. For general $N = 2^n$, we may have to use 4- or 8-dimensional isogenies.

Higher-dimensional (HD) representation

Goal Computing isogeny $\varphi\colon E_0 \to E_1$ of large prime degree ℓ.

If we find[1] $N = 2^n = \ell + a^2$ with $\ell \nmid a$,

$\Psi = \begin{pmatrix} [a] & -\widehat{\varphi} \\ \varphi & [a] \end{pmatrix}$ is a 2-dimensional isogeny of reduced degree 2^n (a 2^n-isogeny)

$\rightsquigarrow$ $(*, \varphi(Q)) = \Psi(Q, 0)$ for all Q. If we can compute Ψ, we can compute φ

Computing Ψ

- If we know torsion point images $\varphi(P)$ for $P \in E_0[N]$, we know $\ker \Psi = \{(aP, -\varphi(P))$ for $P \in E_0[N]\}$

[1] $N - \ell = a^2$ is restrictive. For general $N = 2^n$, we may have to use 4- or 8-dimensional isogenies.

Higher-dimensional (HD) representation

Goal Computing isogeny $\varphi\colon E_0 \to E_1$ of large prime degree ℓ.

If we find[1] $N = 2^n = \ell + a^2$ with $\ell \nmid a$,

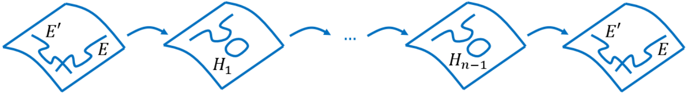
$\Psi = \begin{pmatrix} [a] & -\widehat{\varphi} \\ \varphi & [a] \end{pmatrix}$ is a 2-dimensional isogeny of reduced degree 2^n (a 2^n-isogeny)

$\rightsquigarrow$ $(*, \varphi(Q)) = \Psi(Q, 0)$ for all Q. If we can compute Ψ, we can compute φ

Computing Ψ

- If we know torsion point images $\varphi(P)$ for $P \in E_0[N]$, we know $\ker \Psi = \{(aP, -\varphi(P)) \text{ for } P \in E_0[N]\}$
- Ψ can be decomposed in smaller 2-isogeny pieces, but in dimension 2.

Credits: Wouter Castryck, CAIPI Symposium, Rennes 2024



Intermediate steps: principally polarized abelian surfaces ($\approx$ elliptic curves but 2-dim.)

[1] $N - \ell = a^2$ is restrictive. For general $N = 2^n$, we may have to use 4- or 8-dimensional isogenies.

Higher-dimensional (HD) representation

Goal Computing isogeny $\varphi\colon E_0 \to E_1$ of large prime degree ℓ.

If we find[1] $N = 2^n = \ell + a^2$ with $\ell \nmid a$,

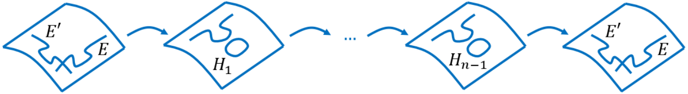
$\Psi = \begin{pmatrix} [a] & -\widehat{\varphi} \\ \varphi & [a] \end{pmatrix}$ is a 2-dimensional isogeny of reduced degree 2^n (a 2^n-isogeny)

$\rightsquigarrow$ $(*, \varphi(Q)) = \Psi(Q, 0)$ for all Q. If we can compute Ψ, we can compute φ

Computing Ψ

- If we know torsion point images $\varphi(P)$ for $P \in E_0[N]$, we know $\ker \Psi = \{(aP, -\varphi(P)) \text{ for } P \in E_0[N]\}$
- Ψ can be decomposed in smaller 2-isogeny pieces, but in dimension 2.

Credits: Wouter Castryck, CAIPI Symposium, Rennes 2024



Intermediate steps: principally polarized abelian surfaces ($\approx$ elliptic curves but 2-dim.)

New goal Computing 2-isogenies of PP abelian surfaces.

- In dim. 1, Vélu's formulas. In dim. 2: can we find explicit formulas from $\ker \Psi$?

[1] $N - \ell = a^2$ is restrictive. For general $N = 2^n$, we may have to use 4- or 8-dimensional isogenies.

How to represent principally polarized abelian varieties?

Definition Abelian variety: connected projective group variety.

How to represent principally polarized abelian varieties?

Definition Abelian variety: connected projective group variety.

- *projective*: there exists an embedding $A \hookrightarrow \mathbb{P}^N$ for some N. Explicitly?

How to represent principally polarized abelian varieties?

Definition Abelian variety: connected projective group variety.

- *projective*: there exists an embedding $A \hookrightarrow \mathbb{P}^N$ for some N. Explicitly?

Tool Theta coordinates of level n on a g-dimensional A:
n^g coordinates $(\theta_i)_{i \in (\mathbb{Z}/n\mathbb{Z})^g}$, with $A[n]$ in a special position.

How to represent principally polarized abelian varieties?

Definition Abelian variety: connected projective group variety.

- *projective*: there exists an embedding $A \hookrightarrow \mathbb{P}^N$ for some N. Explicitly?

Tool Theta coordinates of level n on a g-dimensional A:
n^g coordinates $(\theta_i)_{i\in(\mathbb{Z}/n\mathbb{Z})^g}$, with $A[n]$ in a special position.

$$J: \begin{array}{rcl} A & \hookrightarrow & \mathbb{P}^{n^g-1} \\ P & \mapsto & (\theta_i(P))_{i\in(\mathbb{Z}/n\mathbb{Z})^g} \end{array}$$

Fact If $n \geq 3$, J is injective. If $n = 2$, embedding of Kummer variety $\mathcal{K}_A = A/\pm 1 \hookrightarrow \mathbb{P}^{n^g-1}$.

- $n = 2 \rightsquigarrow$ fewer coordinates $\rightsquigarrow$ efficiency!

How to represent principally polarized abelian varieties?

Definition Abelian variety: connected projective group variety.

- *projective*: there exists an embedding $A \hookrightarrow \mathbb{P}^N$ for some N. Explicitly?

Tool Theta coordinates of level n on a g-dimensional A:
n^g coordinates $(\theta_i)_{i\in(\mathbb{Z}/n\mathbb{Z})^g}$, with $A[n]$ in a special position.

$$J\colon \quad A \hookrightarrow \mathbb{P}^{n^g-1}$$
$$P \mapsto (\theta_i(P))_{i\in(\mathbb{Z}/n\mathbb{Z})^g}$$

Fact If $n \geq 3$, J is injective. If $n = 2$, embedding of Kummer variety $\mathcal{K}_A = A/\pm 1 \hookrightarrow \mathbb{P}^{n^g-1}$.

- $n = 2 \rightsquigarrow$ fewer coordinates $\rightsquigarrow$ efficiency!

Examples

- $g = 1, n = 3$: elliptic curve as cubic curve in $\mathbb{P}^2$

How to represent principally polarized abelian varieties?

Definition Abelian variety: connected projective group variety.

- *projective*: there exists an embedding $A \hookrightarrow \mathbb{P}^N$ for some N. Explicitly?

Tool Theta coordinates of level n on a g-dimensional A:
n^g coordinates $(\theta_i)_{i \in (\mathbb{Z}/n\mathbb{Z})^g}$, with $A[n]$ in a special position.

$$\begin{aligned} J:\ A &\hookrightarrow \mathbb{P}^{n^g-1} \\ P &\mapsto (\theta_i(P))_{i \in (\mathbb{Z}/n\mathbb{Z})^g} \end{aligned}$$

Fact If $n \geq 3$, J is injective. If $n = 2$, embedding of Kummer variety $\mathcal{K}_A = A/\pm 1 \hookrightarrow \mathbb{P}^{n^g-1}$.

- $n = 2 \rightsquigarrow$ fewer coordinates $\rightsquigarrow$ efficiency!

Examples

- $g = 1, n = 3$: elliptic curve as cubic curve in $\mathbb{P}^2$
- $g = 1, n = 2$: $E[2] = \{(a : b) = 0_E, (b : a), (a : -b), (-b : a)\}$

How to represent principally polarized abelian varieties?

Definition Abelian variety: connected projective group variety.

- *projective*: there exists an embedding $A \hookrightarrow \mathbb{P}^N$ for some N. Explicitly?

Tool Theta coordinates of level n on a g-dimensional A:
n^g coordinates $(\theta_i)_{i\in(\mathbb{Z}/n\mathbb{Z})^g}$, with $A[n]$ in a special position.

$$J: \begin{array}{rcl} A & \hookrightarrow & \mathbb{P}^{n^g-1} \\ P & \mapsto & (\theta_i(P))_{i\in(\mathbb{Z}/n\mathbb{Z})^g} \end{array}$$

Fact If $n \geq 3$, J is injective. If $n = 2$, embedding of Kummer variety $\mathcal{K}_A = A/\pm 1 \hookrightarrow \mathbb{P}^{n^g-1}$.

- $n = 2 \rightsquigarrow$ fewer coordinates $\rightsquigarrow$ efficiency!

Examples

- $g = 1, n = 3$: elliptic curve as cubic curve in $\mathbb{P}^2$
- $g = 1, n = 2$: $E[2] = \{(a : b) = 0_E, (b : a), (a : -b), (-b : a)\}$
- $g = 2, n = 2$: quartic Kummer surface $\mathcal{K}_A$ in $\mathbb{P}^3$.

$$\begin{aligned} A[2] = \{&(a:b:c:d),\ (a:-b:c:-d),\ (a:b:-c:-d),\ (a:-b:-c:d),\\ &(b:a:d:c),\ (b:-a:d:-c),\ (b:a:-d:-c),\ (b:-a:-d:c),\\ &(c:d:a:b),\ (c:-d:a:-b),\ (c:d:-a:-b),\ (c:-d:-a:b),\\ &(d:c:b:a),\ (d:-c:b:-a),\ (d:c:-b:-a),\ (d:-c:-b:a)\} \end{aligned}$$

Algorithms on Kummer surfaces

Algorithms on Kummer surfaces

Representation Let A be a 2-dim. abelian variety, level-2 theta coordinates:

- Kummer variety $\mathcal{K}_A$: represent $(\pm P \in A) \mapsto (x : y : z : t) \in \mathbb{P}^3$

Algorithms on Kummer surfaces

Representation Let A be a 2-dim. abelian variety, level-2 theta coordinates:

- Kummer variety $\mathcal{K}_A$: represent $(\pm P \in A) \mapsto (x : y : z : t) \in \mathbb{P}^3$

Point arithmetic A is an algebraic group, but $\mathcal{K}_A = A/\pm 1$ is not. However:

Algorithms on Kummer surfaces

Representation Let A be a 2-dim. abelian variety, level-2 theta coordinates:

- Kummer variety $\mathcal{K}_A$: represent $(\pm P \in A) \mapsto (x : y : z : t) \in \mathbb{P}^3$

Point arithmetic A is an algebraic group, but $\mathcal{K}_A = A/\pm 1$ is not. However:

- Denote $\overline{P} = (x_P : y_P : z_P : t_P) = (\theta_i(P))_{i \in (\mathbb{Z}/2\mathbb{Z})^2}$.
  $\exists$ algebraic relations involving $\overline{P}, \overline{Q}, \overline{P+Q}, \overline{P-Q} \rightsquigarrow$ differential addition: algorithm

$$\text{diff\_add}(\overline{P}, \overline{Q}, \overline{P-Q}) = \overline{P+Q}$$

  Faster than normal point addition!

Algorithms on Kummer surfaces

Representation Let A be a 2-dim. abelian variety, level-2 theta coordinates:

- Kummer variety $\mathcal{K}_A$: represent $(\pm P \in A) \mapsto (x : y : z : t) \in \mathbb{P}^3$

Point arithmetic A is an algebraic group, but $\mathcal{K}_A = A/\pm 1$ is not. However:

- Denote $\overline{P} = (x_P : y_P : z_P : t_P) = (\theta_i(P))_{i \in (\mathbb{Z}/2\mathbb{Z})^2}$.
  $\exists$ algebraic relations involving $\overline{P}, \overline{Q}, \overline{P+Q}, \overline{P-Q} \rightsquigarrow$ differential addition: algorithm
  $$\mathrm{diff\_add}(\overline{P}, \overline{Q}, \overline{P-Q}) = \overline{P+Q}$$
  Faster than normal point addition!
- Doubling algo: $\overline{2P} = \mathrm{diff\_add}(\overline{P}, \overline{P}, \overline{0_A})$.
- More generally, efficient scalar multiplication $\overline{mP}$ for any $m \in \mathbb{Z}$.

Algorithms on Kummer surfaces

Representation Let A be a 2-dim. abelian variety, level-2 theta coordinates:

- Kummer variety $\mathcal{K}_A$: represent $(\pm P \in A) \mapsto (x : y : z : t) \in \mathbb{P}^3$

Point arithmetic A is an algebraic group, but $\mathcal{K}_A = A/\pm 1$ is not. However:

- Denote $\overline{P} = (x_P : y_P : z_P : t_P) = (\theta_i(P))_{i \in (\mathbb{Z}/2\mathbb{Z})^2}$.
  $\exists$ algebraic relations involving $\overline{P}, \overline{Q}, \overline{P+Q}, \overline{P-Q} \rightsquigarrow$ differential addition: algorithm
  $$\mathsf{diff\_add}(\overline{P}, \overline{Q}, \overline{P-Q}) = \overline{P+Q}$$
  Faster than normal point addition!
- Doubling algo: $\overline{2P} = \mathsf{diff\_add}(\overline{P}, \overline{P}, \overline{0_A})$.
- More generally, efficient scalar multiplication $\overline{mP}$ for any $m \in \mathbb{Z}$.

2-isogenies Consider $\Psi \colon A \to B$ a 2-isogeny (i.e. $\ker \Psi \cong (\mathbb{Z}/2\mathbb{Z})^2$), fix coords $(\theta_i^A)_i$ on $\mathcal{K}_A$.

- We can choose compatible theta coordinates $(\theta_i^B)_i$ on $\mathcal{K}_B$.

Algorithms on Kummer surfaces

Representation Let A be a 2-dim. abelian variety, level-2 theta coordinates:

- Kummer variety $\mathcal{K}_A$: represent $(\pm P \in A) \mapsto (x : y : z : t) \in \mathbb{P}^3$

Point arithmetic A is an algebraic group, but $\mathcal{K}_A = A/\pm 1$ is not. However:

- Denote $\overline{P} = (x_P : y_P : z_P : t_P) = (\theta_i(P))_{i \in (\mathbb{Z}/2\mathbb{Z})^2}$.
  $\exists$ algebraic relations involving $\overline{P}, \overline{Q}, \overline{P+Q}, \overline{P-Q} \rightsquigarrow$ differential addition: algorithm
  $$\text{diff\_add}(\overline{P}, \overline{Q}, \overline{P-Q}) = \overline{P+Q}$$
  Faster than normal point addition!
- Doubling algo: $\overline{2P} = \text{diff\_add}(\overline{P}, \overline{P}, \overline{0_A})$.
- More generally, efficient scalar multiplication $\overline{mP}$ for any $m \in \mathbb{Z}$.

2-isogenies Consider $\Psi \colon A \to B$ a 2-isogeny (i.e. $\ker \Psi \cong (\mathbb{Z}/2\mathbb{Z})^2$), fix coords $(\theta_i^A)_i$ on $\mathcal{K}_A$.

- We can choose compatible theta coordinates $(\theta_i^B)_i$ on $\mathcal{K}_B$.
- $\exists$ alg. relations between $(\theta_i^A)_i, (\theta_j^B)_j$. Same techniques as above $\rightsquigarrow$ explicit formulas for Ψ.

Bonus: general pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

Bonus: general pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

- G_1, G_2 subgroups/quotients of elliptic curves, $G_T \leq k^\times$

Bonus: general pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

- G_1, G_2 subgroups/quotients of elliptic curves, $G_T \leq k^\times$
- Pairings: ubiquitous tool in curve- and isogeny-based crypto

Bonus: general pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

- G_1, G_2 subgroups/quotients of elliptic curves, $G_T \leq k^\times$
- Pairings: ubiquitous tool in curve- and isogeny-based crypto

State of the art Algorithm for general pairing computations: Miller, 2004

- Using theta functions: faster algo
- Also applicable to higher-dimensional abelian varieties

Conclusions

Useful for efficiency of isogeny-based cryptography:

- Computing isogenies of elliptic curves of large prime degree

Conclusions

Useful for efficiency of isogeny-based cryptography:

- Computing isogenies of elliptic curves of large prime degree
  reduced to Computing isogenies of reduced degree 2, but in higher dimensions

Conclusions

Useful for efficiency of isogeny-based cryptography:

- Computing isogenies of elliptic curves of large prime degree
  reduced to Computing isogenies of reduced degree 2, but in higher dimensions

We got:

- How to represent higher-dimensional abelian varieties (lots of symmetries)

Conclusions

Useful for efficiency of isogeny-based cryptography:

- Computing isogenies of elliptic curves of large prime degree
  reduced to Computing isogenies of reduced degree 2, but in higher dimensions

We got:

- How to represent higher-dimensional abelian varieties (lots of symmetries)
- Point arithmetic (addition, doubling, scalar multiplication)

Conclusions

Useful for efficiency of isogeny-based cryptography:

- Computing isogenies of elliptic curves of large prime degree
  reduced to Computing isogenies of reduced degree 2, but in higher dimensions

We got:

- How to represent higher-dimensional abelian varieties (lots of symmetries)
- Point arithmetic (addition, doubling, scalar multiplication)
- 2-isogenies
- General pairing algorithms

Conclusions

Useful for efficiency of isogeny-based cryptography:

- Computing isogenies of elliptic curves of large prime degree
  reduced to Computing isogenies of reduced degree 2, but in higher dimensions

We got:

- How to represent higher-dimensional abelian varieties (lots of symmetries)
- Point arithmetic (addition, doubling, scalar multiplication)
- 2-isogenies
- General pairing algorithms
- Improved performance:

Protocol	signing time (kcycles)	verification time
ML-DSA (std)	333	118
SQIsign	5,669,000	108,000
SQIsign2D	124,000	11,000

Conclusions

Useful for efficiency of isogeny-based cryptography:

- Computing isogenies of elliptic curves of large prime degree
  reduced to Computing isogenies of reduced degree 2, but in higher dimensions

We got:

- How to represent higher-dimensional abelian varieties (lots of symmetries)
- Point arithmetic (addition, doubling, scalar multiplication)
- 2-isogenies
- General pairing algorithms
- Improved performance:

Protocol	signing time (kcycles)	verification time
ML-DSA (std)	333	118
SQIsign	5,669,000	108,000
SQIsign2D	124,000	11,000

Thank you for your attention! Questions?

References

- [Mum66] David Mumford, *On the equations defining abelian varieties I*, Inventiones mathematicae, vol. 1, pp. 287-354, Springer, 1966
- [Kan97] Ernst Kani, *The number of curves of genus two with elliptic differentials*, Journal für die reine und angewandte Mathematik, no. 485, pp. 99-122, 1997
- [Rob21] Damien Robert, *Efficient algorithms for abelian varieties and their moduli spaces*, Thesis (Habilitation à Diriger des Recherches), Université de Bordeaux, June 2021.
- [CD22] Wouter Castryck, Thomas Decru *An efficient key recovery attack on SIDH*, Advances in Cryptology – EUROCRYPT 2023, Springer-Verlag, no. 5, pp. 423-447, first appeared July 2022
- [DMPR22] Pierrick Dartois, Luciano Maino, Giacomo Pope, Damien Robert *An algorithmic approach to $(2,2)$-isogenies in the theta model and appli- cations to isogeny-based cryptography*, Cryptology ePrint Archive, Paper 2023/1747, 2023
- [BDD+24] Andrea Basso, Luca De Feo, Pierrick Dartois, Antonin Leroux, Luciano Maino, Giacomo Pope, Damien Robert, Benjamin Wesolowski, *SQIsign2D-West: The Fast, the Small, and the Safer*, Cryptology ePrint Archive, Paper 2024/760, 2024

Kani, HD-representation in dim. 4, 8

Let $\varphi\colon E_0 \to E_1$ be an isogeny of degree m.
Let $N = 2^n > m$.

- Suppose $N - m = a^2 + b^2$. Define $A_2 = \begin{pmatrix} a & -b \\ b & a \end{pmatrix}$ and $F_{\varphi,2} = \begin{pmatrix} \varphi & \\ & \varphi \end{pmatrix}$.
- Otherwise, write $N - m = a^2 + b^2 + c^2 + d^2$ (we can always do so!) and define

$$A_4 = \begin{pmatrix} a & -b & -c & -d \\ b & a & d & -c \\ c & -d & a & b \\ d & c & -b & a \end{pmatrix}, \qquad F_{\varphi,4} = \begin{pmatrix} \varphi & & & \\ & \varphi & & \\ & & \varphi & \\ & & & \varphi \end{pmatrix}$$

For $r = 2, 4$, the matrix $\Psi = \begin{pmatrix} A_r & F_{-\widehat{\varphi},r} \\ F_{\varphi,r} & A_r^T \end{pmatrix}$ is an endomorphism of $E_0^r \times E^r$.

If $\widehat{\Psi}$ is defined by $(\widehat{\Psi})_{i,j} = \widehat{(\Psi)_{j,i}}$, then $\Psi \circ \widehat{\Psi} = [N] = [2^n]$.

Finally, Ψ is a 2^n-isogeny: decompose it in smaller 2-isogenies in dimension r.

General pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

General pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

- Pairings: ubiquitous tool in curve- and isogeny-based crypto
- In this case: $G_1, G_2 \leq E$ ell curve, $\#G_i = \ell$, and $G_T = \mu_\ell = \{\ell\text{-th roots of } 1\} \leq k^\times$

General pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

- Pairings: ubiquitous tool in curve- and isogeny-based crypto
- In this case: $G_1, G_2 \leq E$ ell curve, $\#G_i = \ell$, and $G_T = \mu_\ell = \{\ell\text{-th roots of } 1\} \leq k^\times$
- *non-degenerate*: $G_2 \cong \mathrm{Hom}(G_1, G_T)$

General pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

- Pairings: ubiquitous tool in curve- and isogeny-based crypto
- In this case: $G_1, G_2 \leq E$ ell curve, $\#G_i = \ell$, and $G_T = \mu_\ell = \{\ell\text{-th roots of } 1\} \leq k^\times$
- *non-degenerate*: $G_2 \cong \mathrm{Hom}(G_1, G_T)$

State of the art Algorithm for general pairing computations: Miller, 2004

- Vast literature on optimized pairings, only for *specific* pairing-friendly curves $E/\mathbb{F}_p$

General pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

- Pairings: ubiquitous tool in curve- and isogeny-based crypto
- In this case: $G_1, G_2 \leq E$ ell curve, $\#G_i = \ell$, and $G_T = \mu_\ell = \{\ell\text{-th roots of } 1\} \leq k^\times$
- *non-degenerate*: $G_2 \cong \mathrm{Hom}(G_1, G_T)$

State of the art Algorithm for general pairing computations: Miller, 2004

- Vast literature on optimized pairings, only for *specific* pairing-friendly curves $E/\mathbb{F}_p$

Theta pairings Using theta functions on E, take as input $(\overline{0_E}, \overline{P}, \overline{Q}, \overline{P+Q})$:

- Compute $\overline{mP}, \overline{mP+Q}$ using theta point arithmetic
- A ratio of the coordinates of $\overline{mP}, \overline{mP+Q}$ gives the pairing $e(P, Q)$.

General pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

- Pairings: ubiquitous tool in curve- and isogeny-based crypto
- In this case: $G_1, G_2 \leq E$ ell curve, $\#G_i = \ell$, and $G_T = \mu_\ell = \{\ell\text{-th roots of } 1\} \leq k^\times$
- *non-degenerate*: $G_2 \cong \mathrm{Hom}(G_1, G_T)$

State of the art Algorithm for general pairing computations: Miller, 2004

- Vast literature on optimized pairings, only for *specific* pairing-friendly curves $E/\mathbb{F}_p$

Theta pairings Using theta functions on E, take as input $(\overline{0_E}, \overline{P}, \overline{Q}, \overline{P+Q})$:

- Compute $\overline{mP}, \overline{mP+Q}$ using theta point arithmetic
- A ratio of the coordinates of $\overline{mP}, \overline{mP+Q}$ gives the pairing $e(P,Q)$.

This gives:

- Efficient algorithm for general elliptic curves, improving Miller

General pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

- Pairings: ubiquitous tool in curve- and isogeny-based crypto
- In this case: $G_1, G_2 \leq E$ ell curve, $\#G_i = \ell$, and $G_T = \mu_\ell = \{\ell\text{-th roots of } 1\} \leq k^\times$
- *non-degenerate*: $G_2 \cong \mathrm{Hom}(G_1, G_T)$

State of the art Algorithm for general pairing computations: Miller, 2004

- Vast literature on optimized pairings, only for *specific* pairing-friendly curves $E/\mathbb{F}_p$

Theta pairings Using theta functions on E, take as input $(\overline{0_E}, \overline{P}, \overline{Q}, \overline{P+Q})$:

- Compute $\overline{mP}, \overline{mP+Q}$ using theta point arithmetic
- A ratio of the coordinates of $\overline{mP}, \overline{mP+Q}$ gives the pairing $e(P,Q)$.

This gives:

- Efficient algorithm for general elliptic curves, improving Miller

  $\rightsquigarrow$ Good for isogeny-based crypto (many different curves)

General pairing computations

Definition Pairing: non-degenerate bilinear map $e\colon G_1 \times G_2 \to G_T$.

- Pairings: ubiquitous tool in curve- and isogeny-based crypto
- In this case: $G_1, G_2 \leq E$ ell curve, $\#G_i = \ell$, and $G_T = \mu_\ell = \{\ell\text{-th roots of } 1\} \leq k^\times$
- *non-degenerate*: $G_2 \cong \mathrm{Hom}(G_1, G_T)$

State of the art Algorithm for general pairing computations: Miller, 2004

- Vast literature on optimized pairings, only for *specific* pairing-friendly curves $E/\mathbb{F}_p$

Theta pairings Using theta functions on E, take as input $(\overline{0_E}, \overline{P}, \overline{Q}, \overline{P+Q})$:

- Compute $\overline{mP}, \overline{mP+Q}$ using theta point arithmetic
- A ratio of the coordinates of $\overline{mP}, \overline{mP+Q}$ gives the pairing $e(P,Q)$.

This gives:

- Efficient algorithm for general elliptic curves, improving Miller

  $\rightsquigarrow$ Good for isogeny-based crypto (many different curves)
- Applicable to higher-dimensional abelian varieties