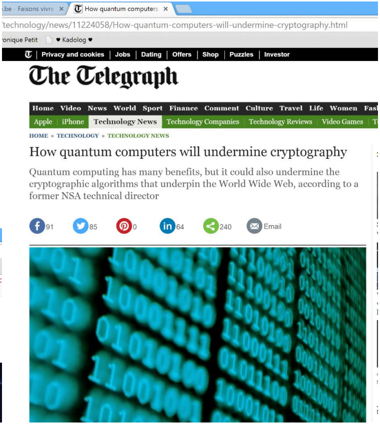
Supersingular Isogeny Graphs and Endomorphism Rings: Reductions and Solutions

Kirsten Eisenträger (Penn State), Sean Hallgren (Penn State),
Kristin Lauter (Microsoft Research), Travis Morrison (Penn State),
Christophe Petit (Birmingham)

Merge from the papers
Hard and Easy Problems for Supersingular Isogeny Graphs
Petit-Lauter [PL17]
On the Hardness of Computing Endomorphism Rings of Supersingular Elliptic Curves
Eisenträger-Hallgren-Morrison [EHM17]

The threat of quantum computers



Isogeny Problems

- Recently proposed for post-quantum cryptography
- Natural problems from a number theory point of view
- Classical and quantum algorithms still exponential time

Isogeny Problems

- Recently proposed for post-quantum cryptography
- Natural problems from a number theory point of view
- Classical and quantum algorithms still exponential time
- But still rather new, need further study

- Our results :
  - Efficient reductions between three hard problem variants
  - Efficient solutions for two (other) problems

Outline

Isogenies and related problems

Motivation : Charles-Goren-Lauter hash function

New results and techniques

Outline

Isogenies and related problems

Motivation : Charles-Goren-Lauter hash function

New results and techniques

Supersingular curves and isogenies

- Let p be a prime. Up to isomorphism, any supersingular elliptic curve is defined over $\mathbb{F}_{p^2}$
- An *isogeny* from a curve E_1 is a non trivial morphism $\phi : E_1 \rightarrow E_2$ sending 0 to 0
- In Weierstrass affine coordinates we can write

$$\phi : E_1 \rightarrow E_2 : \phi(x, y) = \left(\frac{\varphi(x)}{\psi^2(x, y)}, \frac{\omega(x, y)}{\psi^3(x, y)} \right)$$

- Isogeny *degree* is $\deg \phi = \max\{\deg \varphi, \deg \psi^2\}$
- An *endomorphism* of E is an isogeny $\phi : E \rightarrow E$ (examples : scalar multiplications, Frobenius)

Isogeny problems

- Isogeny problems with potential interest for cryptography are about "computing" isogenies between two curves, or some variant of this problem

Isogeny problems

- Isogeny problems with potential interest for cryptography are about "computing" isogenies between two curves, or some variant of this problem
  - A bit tricky to define : degree must be large for security, but then natural output representation is not efficient

Isogeny problems

- Isogeny problems with potential interest for cryptography are about “computing” isogenies between two curves, or some variant of this problem
  - A bit tricky to define : degree must be large for security, but then natural output representation is not efficient

- Endomorphism computation case : hard in general but
  - Easy for special curves
  - Scalar multiplications and Frobenius known trivially

Endomorphism rings

- The endomorphisms of a curve E have a ring structure, operations are addition law on E and composition
- The endomorphism ring of a supersingular curve over $\bar{\mathbb{F}}_p$ is a maximal order in the quaternion algebra $B_{p,\infty}$

Endomorphism rings

- The endomorphisms of a curve E have a ring structure, operations are addition law on E and composition
- The endomorphism ring of a supersingular curve over $\bar{\mathbb{F}}_p$ is a maximal order in the quaternion algebra $B_{p,\infty}$
- **Deuring correspondence** [D31] : bijection from supersingular curves over $\mathbb{F}_{p^2}$ (up to Galois conjugacy) to maximal orders in $B_{p,\infty}$ (up to conjugation)

$$E \to O \approx \mathrm{End}(E)$$

Isogeny graphs

- Over $\bar{\mathbb{F}}_p$ the ℓ-torsion $E[\ell]$ is isomorphic to $\mathbb{Z}_\ell \times \mathbb{Z}_\ell$
- There are $\ell + 1$ cyclic subgroups of order ℓ ; each one is the kernel of a degree ℓ isogeny

Isogeny graphs

- Over $\bar{\mathbb{F}}_p$ the ℓ-torsion $E[\ell]$ is isomorphic to $\mathbb{Z}_\ell \times \mathbb{Z}_\ell$
- There are $\ell + 1$ cyclic subgroups of order ℓ ; each one is the kernel of a degree ℓ isogeny
- ℓ-isogeny graph : each vertex is a j-invariant over $\bar{\mathbb{F}}_p$, each edge corresponds to one degree ℓ isogeny
- Isogeny graphs are undirected

Isogeny graphs

- Over $\bar{\mathbb{F}}_p$ the ℓ-torsion $E[\ell]$ is isomorphic to $\mathbb{Z}_\ell \times \mathbb{Z}_\ell$
- There are $\ell + 1$ cyclic subgroups of order ℓ ; each one is the kernel of a degree ℓ isogeny
- ℓ-isogeny graph : each vertex is a j-invariant over $\bar{\mathbb{F}}_p$, each edge corresponds to one degree ℓ isogeny
- Isogeny graphs are undirected
- In supersingular case all j and isogenies defined over $\mathbb{F}_{p^2}$ and graphs are Ramanujan (optimal expansion graphs)
- Isogeny problems $\sim$ finding paths in these graphs

Outline

Isogenies and related problems

Motivation : Charles-Goren-Lauter hash function

New results and techniques

Charles-Goren-Lauter hash function

Hash of the Future?

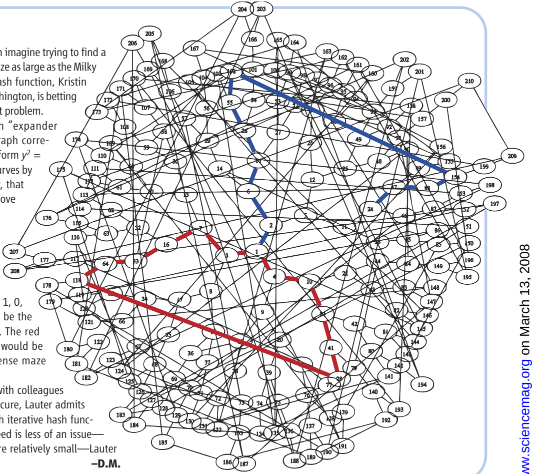
Have you ever struggled to solve a maze? Then imagine trying to find a path through a tangled, three-dimensional maze as large as the Milky Way. By incorporating such a maze into a hash function, Kristin Lauter of Microsoft Research in Redmond, Washington, is betting that neither you nor anyone else will solve that problem.

Technically, Lauter's maze is called an "expander graph" (see figure, right). Nodes in the graph correspond to elliptic curves, or equations of the form $y^2 = x^3 + ax + b$. Each curve leads to three other curves by a mathematical relation, now called isogeny, that Pierre de Fermat discovered while trying to prove his famous Last Theorem.

To hash a digital file using an expander graph, you would convert the bits of data into directions: 0 would mean "turn right," 1 would mean "turn left." In the maze illustrated here, after the initial step 1-2, the blue path encodes the directions 1, 0, 1, 1, 0, 0, 0, 0, 1, ending at point 24, which would be the digital signature of the string 101100001. The red loop shows a collision of two paths, which would be practically impossible to find in the immense maze envisioned by Lauter.

Although her hash function (developed with colleagues Denis Charles and Eyal Goren) is provably secure, Lauter admits that it is not yet fast enough to compete with iterative hash functions. However, for applications in which speed is less of an issue—for example, where the files to be hashed are relatively small—Lauter believes it might be a winner.

–D.M.



Strategy to break CGL hash function

- Idea : use Deuring's correspondence $(E \leftrightarrow O \approx \mathrm{End}(E))$
  1. Translate collision and preimage resistance properties from the elliptic curve setting to the quaternion setting
  2. Break collision and preimage resistance for quaternions
  3. Translate the attacks back to elliptic curve setting

Strategy to break CGL hash function

- Idea : use Deuring's correspondence $(E \leftrightarrow O \approx \mathrm{End}(E))$
  1. Translate collision and preimage resistance properties from the elliptic curve setting to the quaternion setting
  2. Break collision and preimage resistance for quaternions
  3. Translate the attacks back to elliptic curve setting
- Steps 1 and 2 were solved in [KLPT14] : algorithms to compute elements in a given ideal with a given norm

Outline

Isogenies and related problems

Motivation : Charles-Goren-Lauter hash function

New results and techniques

Results in this paper

- Polynomial time collision attack on CGL hash function for "special" initial curves [PL17]
- Constructive Deuring correspondence in one direction : given a maximal order in $B_{p,\infty}$, can efficiently compute the corresponding j-invariant [PL17]
- Equivalence of hard problems [PL17]
  - Constructive Deuring correspondence in other direction
  - Endomorphism ring computation for random curves
  - Collision and preimage resistance of CGL hash function for random initial curves
- Other approach for some of these reductions, using an oracle for the *action on ℓ-torsion* problem [EHM17]

Key tools

- Converting quaternion ideals to isogenies [W69]
  - Let E_0 with known $\mathrm{End}(E_0) \approx O_0 \subset B_{p,\infty}$
  - Isogenies from E_0 correspond to left ideals of O_0
  - Correspondence computed by identifying kernels
  - Efficient for *powersmooth* norms/degrees
- "Quaternion ℓ-isogeny algorithm" [KLPT14,GPS17]
  - Replace ideal by equivalent one with powersmooth norm

Remember : CGL hash function

Hash of the Future?

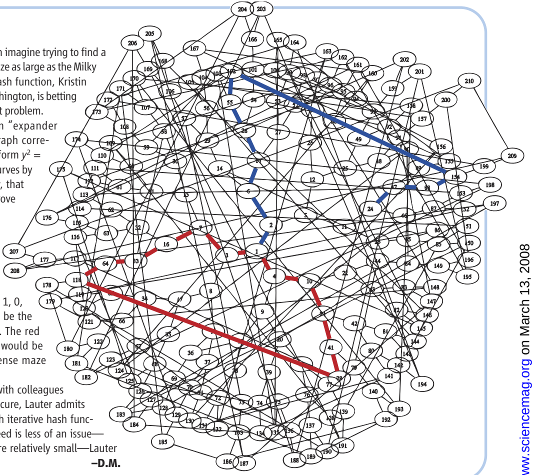
Have you ever struggled to solve a maze? Then imagine trying to find a path through a tangled, three-dimensional maze as large as the Milky Way. By incorporating such a maze into a hash function, Kristin Lauter of Microsoft Research in Redmond, Washington, is betting that neither you nor anyone else will solve that problem.

Technically, Lauter's maze is called an "expander graph" (see figure, right). Nodes in the graph correspond to elliptic curves, or equations of the form $y^2 = x^3 + ax + b$. Each curve leads to three other curves by a mathematical relation, now called isogeny, that Pierre de Fermat discovered while trying to prove his famous Last Theorem.

To hash a digital file using an expander graph, you would convert the bits of data into directions: 0 would mean "turn right," 1 would mean "turn left." In the maze illustrated here, after the initial step 1-2, the blue path encodes the directions 1, 0, 1, 1, 0, 0, 0, 0, 1, ending at point 24, which would be the digital signature of the string 101100001. The red loop shows a collision of two paths, which would be practically impossible to find in the immense maze envisioned by Lauter.

Although her hash function (developed with colleagues Denis Charles and Eyal Goren) is provably secure, Lauter admits that it is not yet fast enough to compete with iterative hash functions. However, for applications in which speed is less of an issue—for example, where the files to be hashed are relatively small—Lauter believes it might be a winner.

–D.M.



www.sciencemag.org on March 13, 2008

Partial attack on CGL hash function

- Suppose CGL hash function uses a **special curve** E_0
- Goal : compute an endomorphism of E_0 of degree ℓ^e (this gives a collision with the void message)

Partial attack on CGL hash function

- Suppose CGL hash function uses a **special curve** E_0
- Goal : compute an endomorphism of E_0 of degree ℓ^e (this gives a collision with the void message)
- Compute $\alpha \in O_0 \approx \mathrm{End}(E_0)$ of norm ℓ^e (as in [KLPT14])
- Deduce a collision path in the quaternion setting $I_i = O_0\ell^i + O_0\alpha,\ i = 1, \ldots, e,$ where $n(I_i) = \ell^i$

Partial attack on CGL hash function

- Suppose CGL hash function uses a **special curve** E_0
- Goal : compute an endomorphism of E_0 of degree ℓ^e (this gives a collision with the void message)
- Compute $\alpha \in O_0 \approx \text{End}(E_0)$ of norm ℓ^e (as in [KLPT14])
- Deduce a collision path in the quaternion setting $I_i = O_0\ell^i + O_0\alpha,\ i = 1, \ldots, e,$ where $n(I_i) = \ell^i$
- For each i
  - Compute $J_i \approx I_i$ with powersmooth norm
  - Compute corresponding isogeny $\varphi_i : E_0 \to E_i$
- Deduce a collision path $(E_0, E_1, \ldots, E_e = E_0)$

Remember : CGL hash function

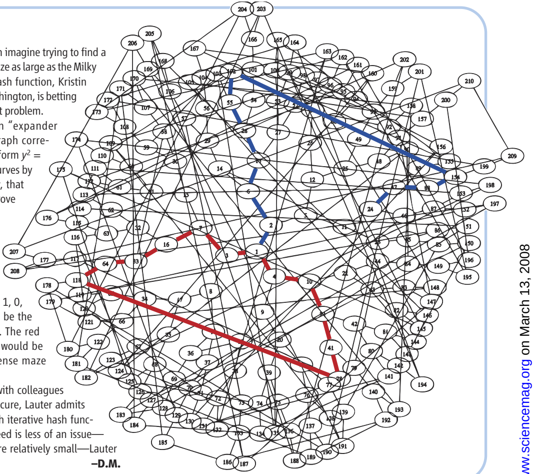
Hash of the Future?

Have you ever struggled to solve a maze? Then imagine trying to find a path through a tangled, three-dimensional maze as large as the Milky Way. By incorporating such a maze into a hash function, Kristin Lauter of Microsoft Research in Redmond, Washington, is betting that neither you nor anyone else will solve that problem.

Technically, Lauter's maze is called an "expander graph" (see figure, right). Nodes in the graph correspond to elliptic curves, or equations of the form $y^2 = x^3 + ax + b$. Each curve leads to three other curves by a mathematical relation, now called isogeny, that Pierre de Fermat discovered while trying to prove his famous Last Theorem.

To hash a digital file using an expander graph, you would convert the bits of data into directions: 0 would mean "turn right," 1 would mean "turn left." In the maze illustrated here, after the initial step 1-2, the blue path encodes the directions 1, 0, 1, 1, 0, 0, 0, 0, 1, ending at point 24, which would be the digital signature of the string 101100001. The red loop shows a collision of two paths, which would be practically impossible to find in the immense maze envisioned by Lauter.

Although her hash function (developed with colleagues Denis Charles and Eyal Goren) is provably secure, Lauter admits that it is not yet fast enough to compete with iterative hash functions. However, for applications in which speed is less of an issue—for example, where the files to be hashed are relatively small—Lauter believes it might be a winner. **–D.M.**



Equivalence of hard problems

1. Constructive Deuring correspondence in reverse direction : given a supersingular j-invariant, compute corresponding maximal order in $B_{p,\infty}$
2. Endomorphism ring computation for random curves
3. Collision and preimage resistance of CGL hash function for a random initial curve

Sketch (1) implies (2)

- Goal : given E and abstract representation of $\mathrm{End}(E)$ as a $\mathbb{Z}$-basis for a maximal order $O \subset B_{p,\infty}$, provide concrete representations of endomorphisms generating $\mathrm{End}(E)$

Sketch (1) implies (2)

- Goal : given E and abstract representation of $\mathrm{End}(E)$ as a $\mathbb{Z}$-basis for a maximal order $O \subset B_{p,\infty}$, provide concrete representations of endomorphisms generating $\mathrm{End}(E)$
- Let E_0 special curve with known $\mathrm{End}(E_0) \approx O_0 \subset B_{p,\infty}$
- Compute ideal I connecting O_0 and O. We then have

$$O \subset \frac{I \; O_0 \; \bar{I}}{n(I)}$$

Sketch (1) implies (2)

- Goal : given E and abstract representation of $\text{End}(E)$ as a $\mathbb{Z}$-basis for a maximal order $O \subset B_{p,\infty}$, provide concrete representations of endomorphisms generating $\text{End}(E)$
- Let E_0 special curve with known $\text{End}(E_0) \approx O_0 \subset B_{p,\infty}$
- Compute ideal I connecting O_0 and O. We then have

$$O \subset \frac{I \; O_0 \; \bar{I}}{n(I)}$$

- Translating I into an isogeny $\varphi : E_0 \to E$ we have

$$\text{End}(E) \subset \frac{\varphi \; \text{End}(E_0) \; \hat{\varphi}}{\deg \varphi}$$

Sketch (1) implies (2)

- Goal : given E and abstract representation of $\mathrm{End}(E)$ as a $\mathbb{Z}$-basis for a maximal order $O \subset B_{p,\infty}$, provide concrete representations of endomorphisms generating $\mathrm{End}(E)$

- Let E_0 special curve with known $\mathrm{End}(E_0) \approx O_0 \subset B_{p,\infty}$
- Compute ideal I connecting O_0 and O. We then have

$$O \subset \frac{I\ O_0\ \bar{I}}{n(I)}$$

- Translating I into an isogeny $\varphi : E_0 \to E$ we have

$$\mathrm{End}(E) \subset \frac{\varphi\ \mathrm{End}(E_0)\ \hat{\varphi}}{\deg \varphi}$$

(use [KLPT14] first to ensure $n(I)$ powersmooth)

Outline

Isogenies and related problems

Motivation : Charles-Goren-Lauter hash function

New results and techniques

Conclusion and perspectives

- With a random initial curve, CGL hash function is secure iff the endomorphism ring computation problem is hard
- For the later, "output representation does not matter"
- Initial curve in CGL hash function must be random (and beware of any backdoor)

Conclusion and perspectives

- With a random initial curve, CGL hash function is secure iff the endomorphism ring computation problem is hard
- For the later, "output representation does not matter"
- Initial curve in CGL hash function must be random (and beware of any backdoor)

- Our algorithms and reductions are heuristic
- Is SIDH secure ? only if endomorphism ring computation problem hard [GPST16], but this may not be enough [P17]

Thanks !

- Questions ?

References

- [CGL09] Charles-Goren-Lauter, Cryptographic Hash Functions from Expander Graphs
- [D31] Deuring, Die Typen der Multiplikatorenringe elliptischer Funktionenkörper
- [EHM17] Eisenträger-Hallgren-Morrison, On the Hardness of Computing Endomorphism Rings of Supersingular Elliptic Curves
- [GPS17] Galbraith-Petit-Silva, Identification Protocols and Signature Schemes Based on Supersingular Isogeny Problems
- [GPST16] Galbraith-Petit-Shani-Ti, On the Security of Supersingular Isogeny Cryptosystems

References

- [KLPT14] Kohel-Lauter-Petit-Tignol, On the quaternion ℓ-isogeny path problem
- [P17] Petit, Faster Algorithms for Isogeny Problems Using Torsion Point Images
- [PL17] Petit-Lauter, Hard and Easy Problems for Supersingular Isogeny Graphs
- [W69] Waterhouse, Abelian varieties over finite fields