Isogeny Cryptography: Strengths, Weaknesses, and Challenges

Steven Galbraith

University of Auckland, New Zealand

Thanks

- Yuval Yarom
- David Kohel, Bryan Birch, Victor Miller, Gerhard Frey, George Rück, Florian Hess, Nigel Smart, Alfred Menezes, Scott Vanstone, David Jao, Fre Vercauteren, Drew Sutherland, Gaetan Bisson, Christophe Petit, John Voight, Luca de Feo, Tanja Lange, Dan Bernstein, etc.
- Anton Stolbunov, Ilya Chevyrev, Chang-An Zhao, Fangqian (Alice) Qiu, Christina Delfs, Barak Shani, Yan Bo Ti, Javier Silva, Joel Laity.

Plan

- Discrete logs and Diffie-Hellman
- Pollard rho and kangaroo
- Generalisations of Diffie-Hellman
- Isogenies
- SIDH/SIKE
- CSIDH

Please interrupt and ask questions.

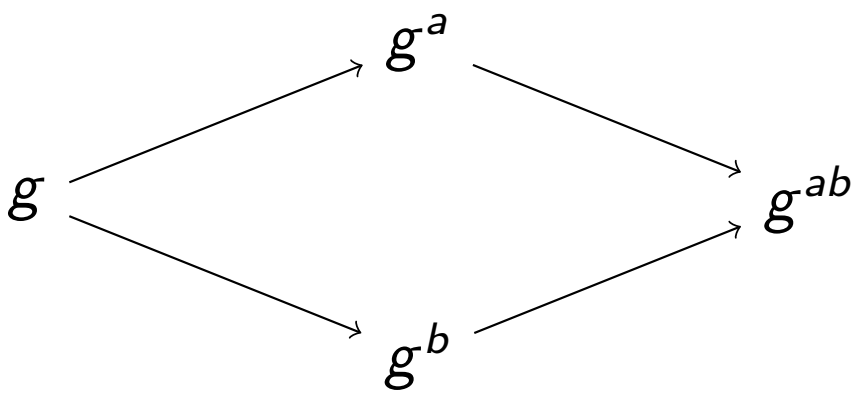
Discrete Logarithm Problem and Diffie-Hellman

Let G be a subgroup of $\mathbb{F}_q^*$ or $E(\mathbb{F}_q)$ of prime order.
Given $g \in G$ and $a \in \mathbb{N}$ it is **easy to compute** $h = g^a$.
Given $g \in G$ and $h = g^a$, it is hard to compute a.

Diffie-Hellman key exchange:

- Alice chooses a and sends $t_A = g^a$ to Bob.
- Bob chooses b and sends $t_B = g^b$ to Alice.
- Alice computes $t_B^a = g^{ab}$.
- Bob computes $t_A^b = g^{ab}$.

Diffie-Hellman key exchange



How hard is Discrete Logarithm Problem (DLP)?

- Let g have prime order r. Baby step giant step solves DLP in (deterministic) $O(\sqrt{r})$ time and space.
- Pollard rho solves DLP in (expected) $O(\sqrt{r})$ time and poly space.
- Idea: If can find integers a_1, a_2, b_1, b_2 such that

$$g^{a_1} h^{b_1} = g^{a_2} h^{b_2}.$$

  then solve the DLP as $a = (a_2 - a_1)(b_1 - b_2)^{-1} \pmod{r}$.
- Approach: Generate pseudorandom walks

$$x_i = g^{a_i} h^{b_i}$$

  such that (a_i, b_i) are known.
- **Pollard's big idea**: Use walks where the next step only depends on the current position.
- **Van Oorschot and Wiener's big idea**: Use distinguished points.

Pollard rho

Discrete logarithm "in an interval"

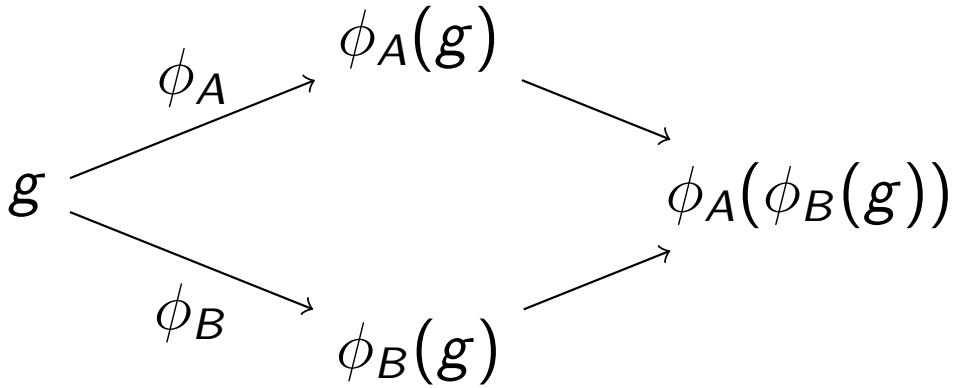
- Let g have order r, and let $0 < N < r$.
  The interval DLP is: Given $h = g^a$ where $0 \leq a < N$, to compute a.
- BSGS in $O(\sqrt{N})$ time and $O(\sqrt{N})$ space.
- Pollard kangaroo method using distinguished points (van Oorschot and Wiener) solves problem in average case expected $(2 + o(1))\sqrt{N}$ group operations.

Pollard kangaroo algorithm

- **Key properties:** one-dimensional space with a sense of direction to the right.
- Let $h = g^a$ with $a \in [0, N]$.
- The tame kangaroo starts in the middle of the interval $[0, N]$.
- The wild kangaroo starts at a.
- Steps in the walk are, on average, distance $m = \sqrt{N}/2$.
- Average case expected running time $(2 + o(1))\sqrt{N}$ group operations.

Generalised Diffie-Hellman 1: Endomorphisms

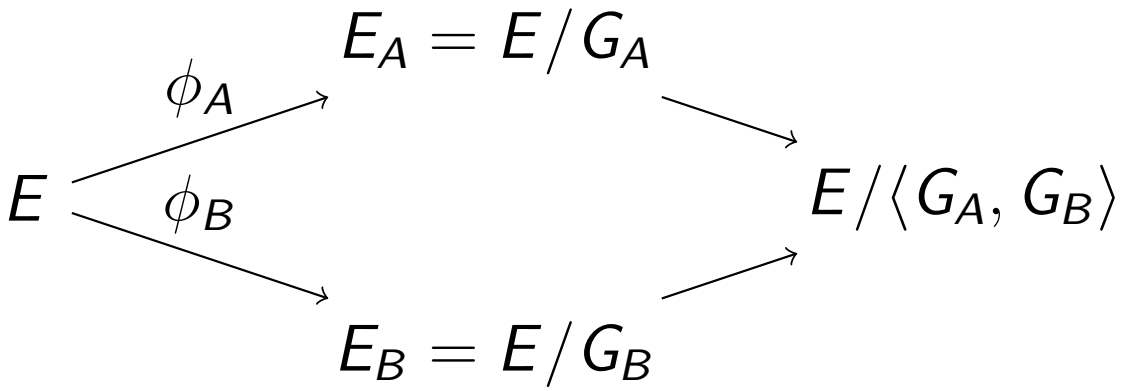
Fix an element g in a group G.
The map $\phi_A(x) = x^a$ is group homomorphism.
DLP: Given g and $\phi_A(g) = g^a$ it is hard to compute ϕ_A.



The maps $\phi_A(x) = x^a$ and $\phi_B(x) = x^b$ are group homomorphisms such that $\phi_A \circ \phi_B = \phi_B \circ \phi_A$.

Generalised Diffie-Hellman 1: Homomorphisms

Let E be an elliptic curve and G_A, G_B finite subgroups such that $G_A \cap G_B = \{0\}$.



Alice and Bob need to give each other enough information that they can compute $\phi_A(G_B)$ and $\phi_B(G_A)$.
Alice computes $E_B/\phi_B(G_A)$ and Bob computes $E_A/\phi_A(G_B)$.

\begin{scary isogeny section}

Example of an isogeny

- Let $E_1 : y^2 = x^3 + x$ and $E_2 : Y^2 = X^3 - 4X$.
- The point $(0, 0)$ on E_1 has order 2.
- There is an isogeny $\phi : E_1 \to E_2$ with kernel generated by $(0, 0)$, given by the rational function

$$\phi(x, y) = \left(\frac{x^2 + 1}{x} \ , \ y \frac{x^2 - 1}{x^2} \right).$$

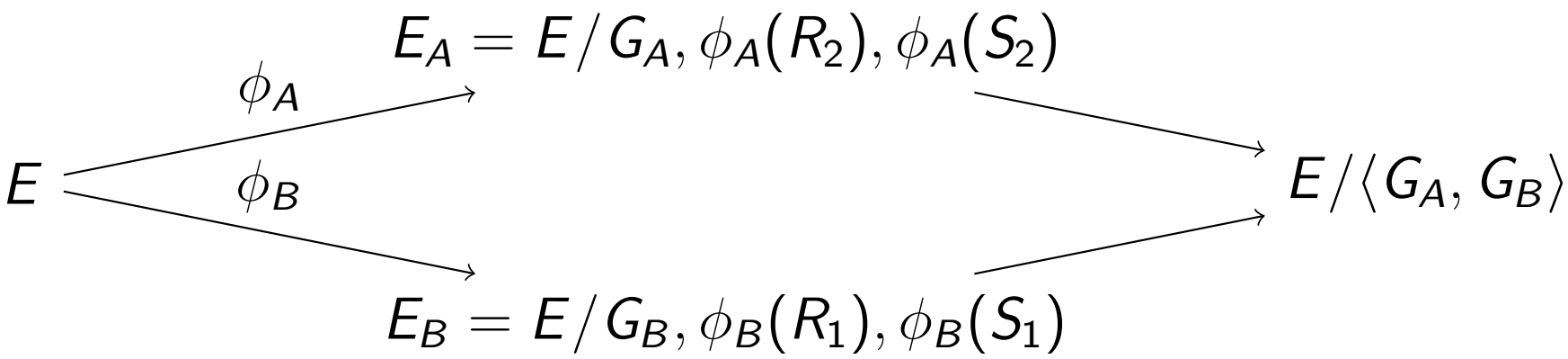
Isogenies

- An **isogeny** $\phi : E_1 \to E_2$ of elliptic curves is a (non-constant) morphism and a group homomorphism.
- An isogeny has finite kernel.
- Given a finite subgroup $G \subseteq E_1(\overline{\mathbb{F}}_q)$ there is a (unique separable) isogeny $\phi_G : E_1 \to E_2$ with kernel G.
- Can compute ϕ_G using Vélu in time **linear** in $\#G$ using operations in $\mathbb{F}_{q^t}$ where $G \subseteq E_1(\mathbb{F}_{q^t})$.
- We will write $E_2 = E_1/G$.
- We focus on separable isogenies, in which case $\deg(\phi) = \#\ker(\phi)$.
- $\mathrm{End}(E) = \{\text{isogenies } \phi : E \to E \text{ over } \overline{\mathbb{F}}_q\} \cup \{0\}$.

Jao and De Feo key exchange (SIDH)

- Let $p = 2^u 3^v f - 1$ be prime, where $2^u \approx 3^v$.
- Let E over $\mathbb{F}_{p^2}$ be a supersingular elliptic curve.
- Then $\#E(\mathbb{F}_{p^2}) = (p+1)^2$ group structure of $E(\mathbb{F}_{p^2})$ is a product of two cyclic groups of order $2^u 3^v f$.
- WTF? E is a supersingular curve with smooth group order?!?!
- They do this because we want to compute isogenies whose kernel is a point of order 2 or 3 defined over $\mathbb{F}_{p^2}$.
- Fix points $R_1, S_1 \in E[2^u]$ such that $\langle R_1, S_1 \rangle = E[2^u]$.
- Fix R_2, S_2 such that $\langle R_2, S_2 \rangle = E[3^v]$.
- The **system parameters** are (E, R_1, S_1, R_2, S_2).

Jao and De Feo key exchange (SIDH)



Jao and De Feo key exchange (SIDH)

- Alice chooses a secret subgroup of $E[2^u]$ by choosing an integer $0 \leq a < 2^u$ and setting $T_1 = R_1 + [a]S_1$.
- Alice computes an isogeny $\phi_A : E \to E_A$ with kernel generated by T_1 and publishes $(E_A, \phi_A(R_2), \phi_A(S_2))$.
- Bob chooses $0 \leq b < 3^v$, computes $\phi_B : E \to E_B$ with kernel generated by $T_2 = R_2 + [b]S_2$ and publishes $(E_B, \phi_B(R_1), \phi_B(S_1))$.
- Alice computes

$$T_1' = \phi_B(R_1) + [a]\phi_B(S_1) = \phi_B(R_1 + [a]S_1) = \phi_B(T_1)$$

  and then computes an isogeny $\phi_A' : E_B \to E_{AB}$ with kernel generated by T_1'.
- Bob computes an isogeny $\phi_B' : E_A \to E_{AB}'$ with kernel $\langle \phi_A(R_2) + [b]\phi_A(S_2) \rangle$.

Jao and De Feo key exchange (SIDH)

- Note: Alice computes E_{AB} and Bob computes E'_{AB}.
- The elliptic curve equations E_{AB} and E'_{AB} computed by Alice and Bob are different, but isomorphic.
- The shared key for Alice and Bob is $j(E_{AB}) = j(E'_{AB})$.
- SIDH gives a convenient plug-and-play substitute for ephemeral Diffie-Hellman key exchange.
- Can use for encryption by making Bob's input static, but need to avoid an active attack due to Galbraith, Petit, Shani, Ti (Asiacrypt 2016).

Performance of SIDH/SIKE

- Alice computes an isogeny $\phi_A : E \to E_A$ of degree 2^u with kernel generated by T_1 and publishes $(E_A, \phi_A(R_2), \phi_A(S_2))$.
- She does this by computing a sequence of u isogenies of degree 2.
- Nice work by De Feo, Jao, Plût on making it efficient.
- **Philosophical question:** Is there a "square-and-multiply" algorithm for isogenies?

SIKE submission to NIST

- SIKE = Supersingular Isogeny Key Exchange.
- Submission to the NIST standardization process on post-quantum cryptography.
- Authors: Jao, Azarderakhsh, Campagna, Costello, De Feo, Hess, Jalali, Koziel, LaMacchia, Longa, Naehrig, Renes, Soukharev and Urbanik.
- Submission contains specification of an IND-CCA KEM.
- `http://sike.org/`
- Strengths: very short ciphertexts. CSIDH is even better.
- Weaknesses: Not as fast as one might hope.

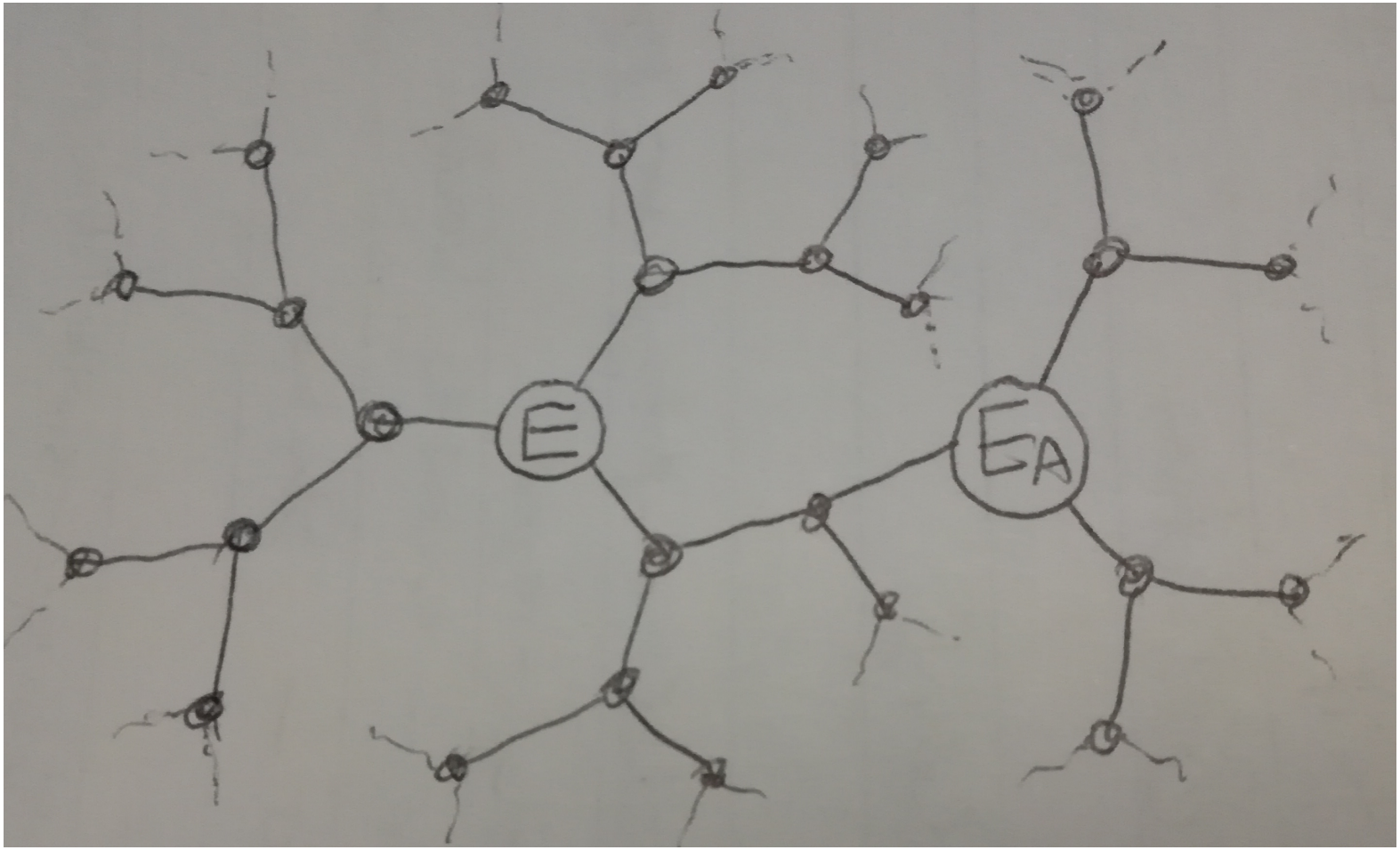
Security of SIDH/SIKE

- Given (E, E_A) an attacker wants to find the isogeny $\phi_A : E \to E_A$ of degree 2^u that maps the points R_2 and S_2 to the given points.
- An attacker needs to have exactly the right isogeny ϕ_A, otherwise they won't compute the correct shared key $E_{AB} = E/\langle G_A, G_B \rangle$.
  (Though, morally, if one can compute any isogeny one ought to be able to compute this special one.)
- There is a baby-step-giant-step algorithm that requires $O(2^{u/2})$ time and space. (Ditto $O(3^{v/2})$.)
  This is $O(p^{1/4})$ isogeny computations in the usual setting.
- Quantum speedup of BSGS using Tani "claw-finding" algorithm, giving $O(2^{u/3}) = O(p^{1/6})$ isogeny computations.

Security of SIDH/SIKE with random walks

- SIDH is analogous to the discrete log in an interval, because E_A is "much closer" to E than a random elliptic curve in the isogeny class would be.
- Can we attack using kangaroos?

Graph of 2-isogenies



Not one-dimensional. No sense of direction.

Security of SIDH/SIKE with random walks

- There is no known low-storage algorithm to break SIDH in $O(2^{u/2})$ isogeny computations.
- Adj, Cervantes-Vázquez, Chi-Domínguez, Menezes and Rodríguez-Henríquez, On the cost of computing isogenies between supersingular elliptic curves, eprint 2018/313 (to appear in proceedings SAC 2018).
- Uses a different idea of van Oorschot and Wiener.
- Gives an algorithm with $O(2^{3u/4})$ complexity.
- See Craig Costello's talk next week.

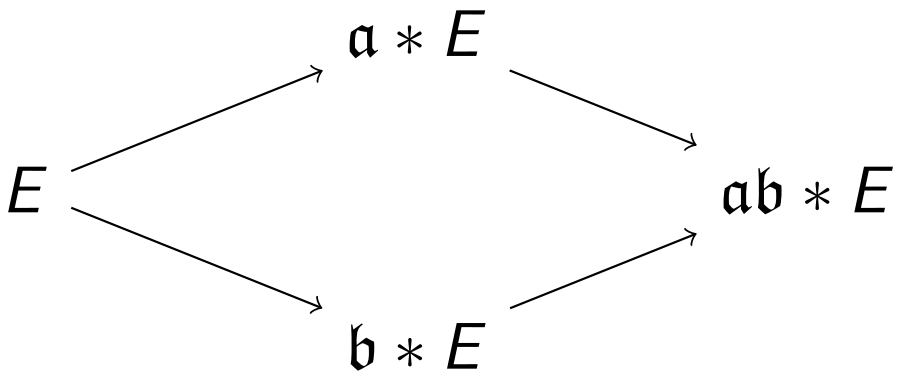
Generalised Discrete Logarithm Problem 2: Homogenous Spaces

(Couveignes 1997)
Let G be a subgroup of $\mathbb{F}_q^*$ or $E(\mathbb{F}_q)$ of prime order r.
For $a \in \mathbb{Z}_r$ and $g \in G$ define $a * g := g^a$.
Given $g \in G$ and $h = a * g$, hard to compute a.
Alternative formulation: an action of $\mathbb{Z}_r^*$ on $G - \{1\}$.

Generalised Diffie-Hellman key exchange:

- Alice chooses $a \in \mathbb{Z}_p$ and sends $t_A = a * g$ to Bob.
- Bob chooses $b \in \mathbb{Z}_p$ and sends $t_B = b * g$ to Alice.
- Alice computes $a * t_B$.
- Bob computes $b * t_A$.

Generalised Diffie-Hellman 2: Group action



Class Group Actions from Isogenies

- J.-M. Couveignes "Hard Homogeneous Spaces", preprint (1997/2006)
- A. Stolbunov, Master thesis (2004)
- A. Rostovtsev, A. Stolbunov, preprint (2006)
- A. Stolbunov "Constructing public-key cryptographic schemes based on class group action on a set of isogenous elliptic curves", Adv. Math. Comm., (2010)
- Couveignes describes a Diffie-Hellman-type key exchange based on group actions.
  Does not mention post-quantum security.
- Rostovtsev and Stolbunov give key exchange and encryption.
  Suggest isogenies could be post-quantum secure.
- Stolbunov's thesis describes also mentions signatures.

Computational problems

- Vectorisation: Given E and $\mathfrak{a} * E$ to compute $\mathfrak{a}$.
- Parallelisation: Given E, $\mathfrak{a} * E$, $\mathfrak{b} * E$ to compute $(\mathfrak{ab}) * E$.
- See Ben Smith paper "Pre- and post-quantum DiffieHellman from groups, actions, and isogenies"
- New result (Galbraith, Panny, Vercauteren): parallelisation + Shor $\rightarrow$ vectorisation.

Pollard rho for vectorisation

- Let $\mathfrak{a} \in G$ act on a set of things called E.
- Problem instance: E_0 and $E_A = \mathfrak{a} * E$.
- Define pseudo-random walk $E \mapsto \mathfrak{b} * E$ for known $\mathfrak{b} \in G$.
- Define walk on pairs $(E, \mathfrak{c}) \mapsto (\mathfrak{b} * E, \mathfrak{b}\mathfrak{c})$.
- Start half the walks on $(\mathfrak{c} * E_0, \mathfrak{c})$ and the other half on $(\mathfrak{c} * E_A, \mathfrak{c})$.
- Run walks until hit a distinguished point.
- With probability $1/2$ a collision gives a solution to $\mathfrak{c}_i * E_0 = \mathfrak{c}_j * E_A$ and so the solution is $\mathfrak{c}_i \mathfrak{c}_j^{-1}$.

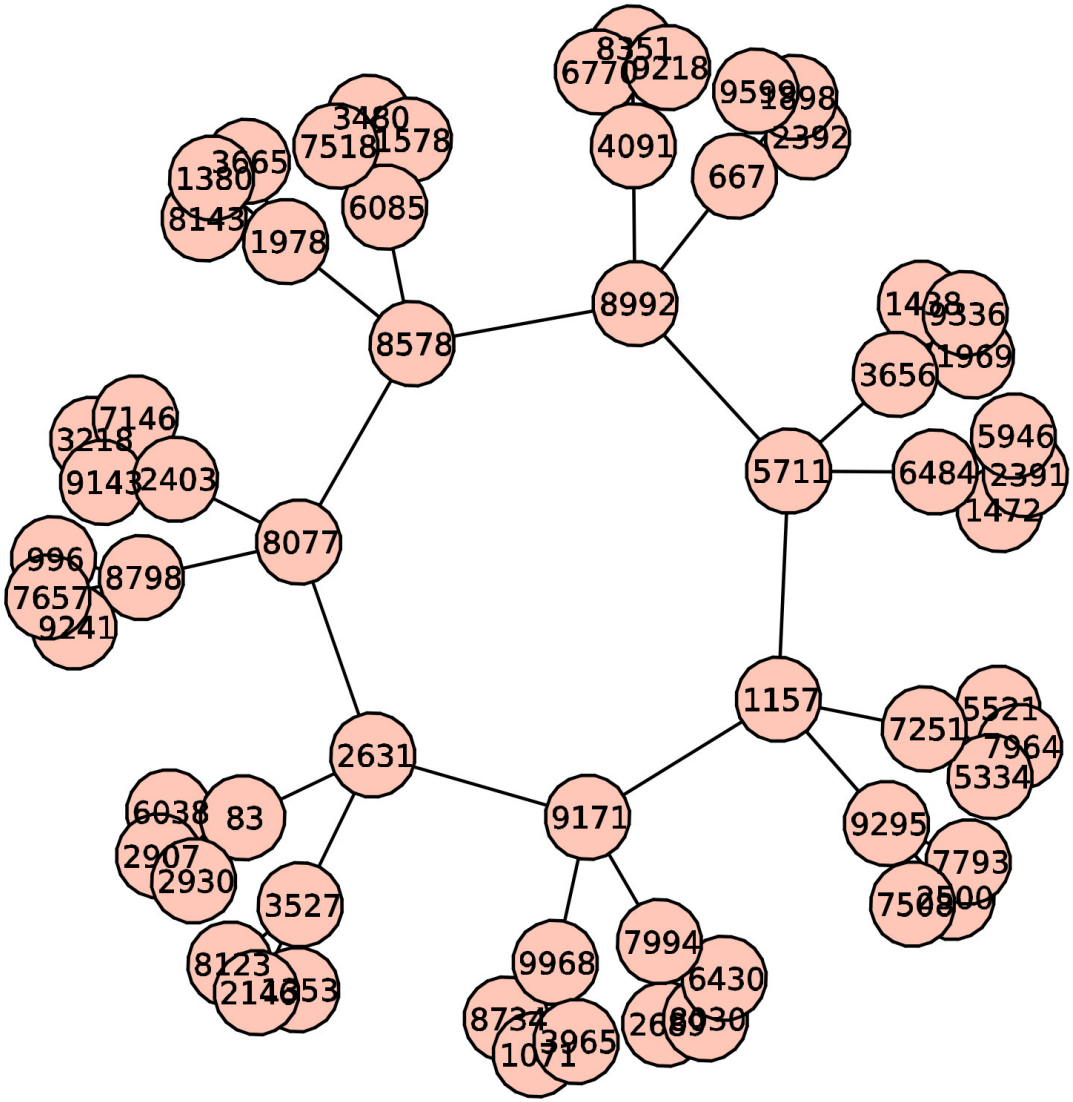
Class Group Action on Elliptic Curves

- Let E be an ordinary elliptic curve over $\mathbb{F}_q$ with $\mathrm{End}(E) \cong \mathcal{O}$ an order in an imaginary quadratic field.
- Let $\mathfrak{a}$ be an invertible $\mathcal{O}$-ideal.
- Can define the subgroup

$$E[\mathfrak{a}] = \{P \in E(\overline{\mathbb{F}}_q) : \phi(P) = 0 \ \forall \phi \in \mathfrak{a}\}.$$

  (Waterhouse 1969)
- There is an isogeny $E \to E'$ with kernel $E[\mathfrak{a}]$. Define $\mathfrak{a} * E$ to be $E' = E/E[\mathfrak{a}]$.
- $\mathfrak{a} * E$ depends only on the ideal class of $\mathfrak{a}$.
- This gives an action of the ideal class group $\mathrm{Cl}(\mathcal{O})$ on the set of E with $\mathrm{End}(E) \cong \mathcal{O}$.

Ordinary Isogeny Graph ($\ell = 3$)



Credit: Dustin Moody

How to compute $\mathfrak{a} * E$?

- Recall we can only compute an isogeny efficiently if its kernel G is small and preferably defined over the ground field.
- For a random ideal class $\mathfrak{a}$ the kernel will not satisfy this requirement.
- Let $\mathfrak{l}_1, \ldots, \mathfrak{l}_n$ be split prime ideals ("Elkies primes") of small norm. Write

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

  (Let's assume $\{\mathfrak{l}_i\}$ generates the class group.)
- Couveignes: time required "a few hours".
- Stolbunov: compute $\mathfrak{a} * E$ in 4 minutes or so.
- De Feo, Kieffer and Smith discuss choosing a special curve to make the isogeny computations faster.
- CSIDH improve this (see later)

Efficiency of computation

- In key exchange we turn things around: Choose uniform $e_i \in [-B, B]$ and define $\mathfrak{a} = \prod_i \mathfrak{l}_i^{e_i}$.
- Computing a walk $\prod_{i=1}^{k} \ell_i^{e_i} \pmod{p}$ in a graph requires $\sum_{i=1}^{k} |e_i| \approx kB$ operations.
- To minimise this cost subject to $(2B+1)^k \approx p$ one takes $k \approx \log(p)$ and $B = O(1)$.
- Actually: $\sum_i |e_i| l_i$ operations where l_i is the norm of ℓ_i. Stolbunov uses different intervals $e_i \in [-B_i, B_i]$ to optimise cost.

CSIDH (Castryck, Lange, Martindale, Panny, Renes 2018)

- Let $p = 4\ell_1 \cdots \ell_k - 1$.
- Let X be the set of isomorphism classes of supersingular elliptic curves E with j-invariant in $\mathbb{F}_p$.
  Note that $\#X = O(\sqrt{p}\log(p))$.
- All $E \in X$ have $\mathrm{End}_{\mathbb{F}_p}(E)$ an order in $\mathbb{Q}(\sqrt{-p})$.
  Here $\mathrm{End}_{\mathbb{F}_p}(E) = \{\phi : E \to E \text{ defined over } \mathbb{F}_p\}$.
- C. Delfs and S. D. Galbraith (2016) showed that one can define class group actions on X.
- CSIDH is an instantiation of group action crypto using supersingular curves, which gives **massive** performance improvements.
- Features:
  - No public key validation needed, so can do non-interactive key exchange.
  - Better bandwidth.
  - Only sub-exponentially quantum secure.

Open problems

- How close to uniform is the distribution

$$\mathfrak{a} \equiv \prod_i \mathfrak{l}_i^{e_i}$$

  over uniform $e_i \in [-B, B]$, for fixed small prime ideals $\mathfrak{l}_i$? (Let's assume $\{\mathfrak{l}_i\}$ generates the class group.)
- Can small prime factors of $\#\mathrm{Cl}(\mathcal{O})$ be determined? Can subgroups of ideal class group be exploited?
- (Boneh): Find other homogeneous spaces/torsors for group actions that are efficient and secure for crypto.

Computational problems and algorithms

- Given E and $E' = \mathfrak{a} * E$ to determine the ideal (class) $\mathfrak{a}$.
- Pollard-rho-type algorithm due to Galbraith-Hess-Smart in time $\tilde{O}(\sqrt{\#G})$ (bug fixed by Stolbunov).
  Running time $\tilde{O}(p^{1/4})$ in the CSIDH case.
- Hidden shift problem: G an abelian group and $f, g : G \to S$ such that, for some $s \in G$, $g(x) = f(xs)$ for all $x \in G$. Problem: find s.
- Idea: Given $(E, E' = \mathfrak{a} * E)$ define $f(\mathfrak{b}) = \mathfrak{b} * E$ and $g(\mathfrak{b}) = \mathfrak{b} * E' = f(\mathfrak{b}\mathfrak{a})$.
- Kuperberg's algorithm is a subexponential time and space quantum algorithm for the hidden shift problem.
- A. Childs, D. Jao and V. Soukharev were the first to analyse Kuperberg's algorithm in the isogeny setting.
- See talk by J.-F. Biasse.

Candidate post-quantum pairing

Recent paper by Boneh, Glass, Krashen, Lauter, Sharif, Silverberg, Tibouchi and Zhandry (eprint 2018/665).

- Fix **ordinary** $E/\mathbb{F}_q$
- Fact: $(\mathfrak{a}_1 * E) \times (\mathfrak{a}_2 * E) \cong (\mathfrak{a}_1\mathfrak{a}_2 * E) \times E$ as unpolarized abelian varieties.
  (Result holds more generally for n terms; see Kani 2011.)
- This is essentially a bilinear pairing (resp. multilinear map).
- Not used for key exchange, but other more complex protocols.
- **Open problem:** To find a computable invariant of the isomorphism class.
- **Application:** Algorithm to solve the decisional Diffie-Hellman problem for class group actions in the ordinary case (but not the supersingular case).

Public Key Signatures

- L. De Feo and S. Galbraith "SeaSign: Compact isogeny signatures from class group actions", eprint 2018/824.
- Public key: E and $E_A = \mathfrak{a} * E$ where

$$\mathfrak{a} \equiv \prod_i \mathfrak{l}_i^{e_i}$$

  and $\mathfrak{l}_i$ ideals of small prime norm, $|e_i| \leq B$.
- Signer generates random ideals $\mathfrak{b}_k = \prod_{i=1}^{n} \mathfrak{l}_i^{f_{k,i}}$ for $1 \leq k \leq t$ and computes $\mathcal{E}_k = \mathfrak{b}_k * E$.
- Compute $H(j(\mathcal{E}_1), \ldots, j(\mathcal{E}_t), \text{message})$ where H is a cryptographic hash function with t-bit output $b_1, \ldots, b_t$.
- If $b_k = 0$ signature includes $\mathbf{f}_k = (f_{k,1}, \ldots, f_{k,n})$ and if $b_k = 1$ it includes

$$\mathbf{f}_k - \mathbf{e} = (f_{k,1} - e_1, \ldots, f_{k,n} - e_n).$$

- Use Lyubashevsky's "Fiat-Shamir with aborts".

Thank You