ECC2011 summer school

September 15–16, 2011

Point counting algorithms on hyperelliptic curves

F. Morain

I. Introduction and motivations

Goal: build an effective group of cryptographic strength, resisting all known attacks.

Dream: find Nechaev groups G, in which the best attack will be $O(\sqrt{\#G})$ (existence?)

Best groups so far: hyperelliptic curves of genus g, with size $\approx q^g$ over some finite field $\mathbb{F}_q$. Typical size $q^g \approx 2^{160--200} \approx 10^{50--60}$.

- Miller, Koblitz (1986): elliptic curves are suggested for use, following the breakthrough of Lenstra in integer factorization (1985).
- Koblitz (1988): hyperelliptic cryptosystems.

In this series of talks

- Put the emphasis on elliptic curves, but take a more general view from time to time; $g > 1$ is the next case; sometimes, hec's yield info on ec's.

- Consider any base field, with some preference for large prime fields, or $\mathbb{F}_{2^n}$; few places where it really matters.

General overview of the lectures

I. Point counting algorithms: basic approaches.
II. Point counting algorithms: elaborate methods.

Bibliography and links

- *A course in algorithmic algebraic number theory* (Cohen);
- *The arithmetic of elliptic curves* (Silverman);
- *Elliptic curve public key cryptosystems* (Menezes);
- *Elliptic curves in cryptography* (Blake, Seroussi, Smart);
- *Advances in Elliptic curves in cryptography* (Blake, Seroussi, Smart);
- *Handbook of Elliptic and Hyperelliptic Curve Cryptography* (Cohen, Frey);
- *Algebraic aspects of cryptography* (Koblitz, appendix on hec by Menezes, Wu, Zuccherato).

ECC2011 summer school

September 15, 2011

Point counting algorithms:
I. basic approaches

F. Morain

Plan

I. Elements of theory.

II. Particular curves.

III. Generic methods.

IV. Schoof's algorithm.

I. Elements of theory

Let C be a plane smooth projective curve of genus g with equation $F(X, Y) = 0$ with coefficients in $\mathbb{K}$, $\mathrm{char}(\mathbb{K}) = p$.

Conic: (genus 0) $x^2 + y^2 = 1$.

Elliptic curve: (genus 1) $y^2 = x^3 + x + 1$.

Hyperelliptic curve: (genus g) $y^2 = x^{2g+1} + \cdots$ (or in some cases $y^2 = x^{2g+2} + \cdots$).

Rem. To simplify things, we assume that C is “at most” hyperelliptic (no C_{ab} or $X_0(N)$).

Def. $C(\mathbb{K}) = \{P = (x, y) \in \mathbb{K}^2, F(x, y) = 0\}$.

Thm. When $g \leq 1$, there is a group law on $C(\mathbb{K})$. When $g > 1$, there is a group law on the **jacobian** of the curve.

Elliptic curves

$$E : Y^2 + a_1XY + a_3Y = X^3 + a_2X^2 + a_4X + a_6$$

$$b_2 = a_1^2 + 4a_2, b_4 = 2a_4 + a_1a_3, b_6 = a_3^2 + 4a_6,$$

$$b_8 = a_1^2a_6 + 4a_2a_6 - a_1a_3a_4 + a_2a_3^2 - a_4^2,$$

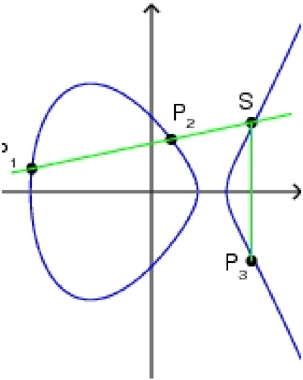
$$c_4 = b_2^2 - 24b_4, c_6 = b_2^3 + 36b_2b_4 - 216b_6,$$

$$\Delta = -b_2^2b_8 - 8b_4^3 - 27b_6^2 + 9b_2b_4b_6 \neq 0$$

$$j(E) = \frac{c_4^3}{\Delta}$$

When $p = 2$: $Y^2 + XY = X^3 + a_2X^2 + a_6$, $j = 1/a_6$.
When $p > 3$: $Y^2 = X^3 + AX + B$, $\Delta = -16(4A^3 + 27B^2)$.
$E(\mathbb{K})$, tangent-and-chord ($\oplus$, O_E), multiplication by n noted $[n]P$.

Group law



$$P_3 = P_1 \oplus P_2$$

$$[k]P = \underbrace{P \oplus \cdots \oplus P}_{k \text{ times}}$$

Hyperelliptic curves

$$y^2 + h(x)y = f(x) = x^{2g+1} + \cdots$$

IMPORTANT WARNING:

For almost all topics (properties, algorithms, etc.),
$g > 1$ is exponentially more difficult than $g = 1$.

Representing $\mathrm{Jac}(C)$

1. Mumford: An element ($=$ a divisor) of $\mathrm{Jac}(C)$ is

$$D = \langle u(z), v(z)\rangle, \ \deg(u) \leq g, \ \deg(v) < \deg(u),$$

defined by (if $P_i = (x_i, y_i)$),

$$u(z) = \prod_{i=1}^{g}(z - x_i), \text{ and } v(x_i) = y_i, \ \forall i.$$

Rem. If $D = \langle u(z), v(z)\rangle$, then $-D = \langle u(z), -v(z)\rangle$.

Group law: Cantor's algorithm (or special formulae for fixed g *à la* Spallek, Harley, Nagao).

2. Theta representations: Chudnovsky& Chudnovsky, Gaudry, . . . , Robert, Cosset.

Cardinality

$\mathbb{K} = \mathbb{F}_q = \mathbb{F}_{p^n}$; $N_r = \#C(\mathbb{K}_r)$ where $[\mathbb{K}_r : \mathbb{K}] = r$:

$$Z(T) = \exp\left(\sum_{r\geq 1} N_r \frac{T^r}{r}\right).$$

Ex. $\mathbb{P}^1(\mathbb{F}_{q^r}) = \{(x_0, x_1) \neq (0,0) \in \mathbb{F}_{q^r}^2\}/\sim$.

$$\#\mathbb{P}^1(\mathbb{F}_{q^r}) = 1 + q^r$$

$$Z(T) = \frac{1}{(1-T)(1-qT)}.$$

Weil's theorem

Thm. (Weil) $Z(T) \in \mathbb{Q}[T]$

$$Z(T) = \frac{L(T)}{(1-T)(1-qT)}$$

(i) $L(T) = 1 + a_1 T + \cdots + q^g T^{2g}$, $a_i \in \mathbb{Z}$;
(ii) $a_{2g-i} = q^{g-i} a_i$ for $0 \leq i \leq g$;
(iii) if $L(T) = \prod(1 - \alpha_i T)$, then $\alpha_i \alpha_{g+i} = q$ and $|\alpha_i| = \sqrt{q}$.

Thm. $\#\mathrm{Jac}(C) = L(1)$.

Coro. $|\#C - (q+1)| \leq 2g\sqrt{q}$;
$(\sqrt{q}-1)^{2g} \leq \#\mathrm{Jac}(C) \leq (\sqrt{q}+1)^{2g}$.

ℓ-torsion

Def. $\mathrm{Jac}[n] = \{P \in \mathrm{Jac}(\overline{\mathbb{K}}), [n]P = O_J\}$.

Thm. If $(n, \mathrm{char}(\mathbb{K})) = 1$, $\mathrm{Jac}[n] \sim (\mathbb{Z}/n\mathbb{Z})^{2g}$; $\mathrm{Jac}[p^r] = (\mathbb{Z}/p\mathbb{Z}^r)^k$, $0 \leq k \leq g$.

Rem. In general $k = g$ (ordinary curves); when $g = 1$, the case $k = 0$ corresponds to **supersingular** curves.

Coro. $\mathrm{Jac}(C)/\mathbb{K}$ is at most $C_1 \times C_2 \times \cdots \times C_{2g}$.

For $g = 1$, this means E is cyclic (very often) or $C_1 \times C_2$ (rarely).

Division polynomials for elliptic curves

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

$$[n](X,Y) = \left(\frac{\phi_n(X,Y)}{\psi_n(X,Y)^2}, \frac{\omega_n(X,Y)}{\psi_n(X,Y)^3} \right)$$

$$\phi_n = X\psi_n^2 - \psi_{n+1}\psi_{n-1}$$

$$4Y\omega_n = \psi_{n+2}\psi_{n-1}^2 - \psi_{n-2}\psi_{n+1}^2$$

$$\phi_n, \psi_{2n+1}, \psi_{2n}/(2Y), \omega_{2n+1}/Y, \omega_{2n} \in \mathbb{Z}[A,B,X]$$

Rem. When $g > 1$, one can define analogous division polynomials – as a matter of fact, division ideals – (cf. Cantor).

$$f_n(X) = \begin{cases} \psi_n(X,Y) & \text{for } n \text{ odd} \\ \psi_n(X,Y)/(2Y) & \text{for } n \text{ even} \end{cases}$$

$$f_{-1} = -1, \quad f_0 = 0, \quad f_1 = 1, \quad f_2 = 1$$

$$f_3(X,Y) = 3X^4 + 6AX^2 + 12BX - A^2$$

$$f_4(X,Y) = X^6 + 5AX^4 + 20BX^3 - 5A^2X^2 - 4ABX - 8B^2 - A^3$$

$$f_{2n} = f_n(f_{n+2}f_{n-1}^2 - f_{n-2}f_{n+1}^2)$$

$$f_{2n+1} = \begin{cases} f_{n+2}f_n^3 - f_{n+1}^3 f_{n-1}(16Y^4) & \text{if } n \text{ is odd} \\ (16Y^4)f_{n+2}f_n^3 - f_{n+1}^3 f_{n-1} & \text{otherwise.} \end{cases}$$

$$\deg(f_n(X)) = \begin{cases} (n^2-1)/2 & \text{if } n \text{ is odd} \\ (n^2-4)/2 & \text{otherwise.} \end{cases}$$

Thm. $P = (x,y)$ point of order ℓ in $E(\overline{\mathbb{K}})$
$\Longleftrightarrow [2]P = O_E$ or $f_\ell(x) = 0$.

II. Particular curves

A) Supersingular curves

Elliptic curves: E s.t. $\#E = q + 1 - c$, $p \mid c$ (not every c, all is known).
For instance: when $n = 2m + 1$, $q = 2^n$

E	c_n
$Y^2 + Y = X^3$	0
$Y^2 + Y = X^3 + X$	$-(2/n)\sqrt{2q}$
$Y^2 + Y = X^3 + X + 1$	$(2/n)\sqrt{2q}$

(See A. Menezes and S. Vanstone, *Utilitas Math.*, 38:135–153, 1990)
Pb: subject to the MOV reduction (see also Frey, Rück).

$g > 1$**:** can be generalized, but reductions still apply (see also Galbraith for security evaluation).

B) CM curves

$g = 1$:

Thm. (Katre) If $p = x^2 + 4y^2$ with $x \equiv 1 \bmod 4$ and $a \not\equiv 0 \bmod p$, then $E : Y^2 = X^3 + aX$ has cardinality

$$p + 1 - \begin{cases} 2x & \text{if } (a/p)_4 = 1, \\ -2x & \text{if } (a/p)_4 = -1, \\ -4y & \text{otherwise with } y \text{ s.t. } 2y(a/p)_4 = x. \end{cases}$$

There are 13 cases of curves defined over $\mathbb{Q}$ having such properties; in general, $4p = A^2 + DB^2$, $\#E = p + 1 - A$: basis for primality proving with elliptic curves (ECPP, Atkin, M.).

$g > 1$:

Spallek, Weng ($g = 2$); Buhler-Koblitz; Duursma-Sakurai; Chao, Matsuda, Nakamura, Tsujii; etc., etc.

$\Rightarrow$ M. Streng's talks.

Pb: too much structure?

C) Misc

- Weil-Koblitz: Build curves over $\mathbb{F}_q$ for q small and use $\mathrm{Jac}(C)/\mathbb{F}_{q^k}$. ECDL might be a little easier.
- Weil descent: Start from ec's to build hec's (Smart *et al.*).
- $Y^2 = X^{2g+1} + aX$, $Y^2 = X^{2g+1} + a$ (Jacobsthal sums: Furukawa/Kawazoe/Takahashi 2003, Haneda/Kawazoe/Takahashi 2005).
- Satoh: $Y^2 = X^5 + uX^3 + vX$ as covering of elliptic curves.

III. Generic methods

Input: a finite abelian group $(G, +)$ with $\#G \leq B$.
Output: $\#G$ together with a proof (factors of $\#G$ + structure with generators; for curves, use pairings).

1. Enumeration: $O(\#G)$ if one has a means of enumerating G...

2. Use Lagrange's theorem: for random $x \in G$, find $\omega =$ order of x. Deduce from this the order of G (take care to small orders, group structure with SNF, etc.; see Cohen). Relatively easy when G is cyclic and the number of generators important.

Easy method: try increasing value of ω: $O(\omega) \leq O(B)$, $O(1)$ space, deterministic.

Shanks's baby steps/giant steps method

Write $m = m_0 + m_1 b$ for some b, $0 \leq m_0 < b$, $0 \leq m_1 \leq B/b$ and write

$$[m]x = 0 \Longleftrightarrow [m_1]([-b]x) = [m_0]x.$$

1. **baby steps**: precompute $\mathcal{B} = \{[m_0]x, 0 \leq m_0 \leq b\}$;
2. **giant steps**: find all m_1 s.t. $[m_1]([-b]x) = [m_0]x$ for some m_0.

Cost: $b + B/b$ minimized with $b = \sqrt{B}$. Time and space are $O(\sqrt{B})$ group operations, assuming membership testing is $O(1)$ (hashing), deterministic.

Rem. can be modified when $A \leq \#G \leq B$, yielding a method in $O(\sqrt{B-A})$.

Using kangaroos (Stein-Teske, Gaudry-Harley, Matsuo-Chao-Tsujii): probabilistic method in $O(\sqrt{B-A})$ time and $O(1)$ space.

Application to elliptic curves

- Enumeration: find all $x \in \mathbb{F}_q$ s.t. $f(x)$ is a square.
- Lagrange: $[q+1]P = [\pm c]P$ for $0 \leq c \leq 2\sqrt{q}$.
  Rem. If $\mathrm{ord}(P)$ is large enough, then

  $$\#\{c \in [-2\sqrt{q}, 2\sqrt{q}], [q+1-c]P = O_E\} = 1$$

  and we can bypass the structure problem (Mestre).
- Kangaroos: idem.
- Shanks: we can do slightly better finding c and not ω.
  Write $c = n_0 + n_1 W$, $0 \leq n_0 < W$, $|n_1| \leq 2\sqrt{q}/W$. Write

  $$[q+1-n_0]P = [\pm n_1][W]P, 0 \leq n_1 \leq 2\sqrt{q}/W$$

  Cost: $W = \sqrt{2\sqrt{q}}$, so $O(2\sqrt{2\sqrt{q}})$.

Application to hyperelliptic curves

$$L(1) = 1 - s_1 + \cdots + (-1)^g s_g + (-1)^{g+1} q s_{g-1} + \cdots - q^{g-1} s_1 + q^g,$$

$$|s_i| \leq \binom{2g}{i} q^{i/2}.$$

A) Enumeration

$g = 2$: compute $N_1(C)$ and $N_2(C)$ and deduce
$s_1 = q + 1 - N_1(C)$, $s_2 = (s_1^2 + N_2(C) - (q^2 + 1))/2$.

$g = 3$: $s_3 = (s_1^3 - 3 s_1 s_2 - N_3 + q^3 + 1)/3$.

Prop. Method in $O(q^g)$.

B) Lagrange

Hasse-Weil gives
$w = (\sqrt{q}+1)^{2g} - (\sqrt{q}-1)^{2g} = 4gq^{(2g-1)/2} + O(q^{(2g-3)/2})$ (for fixed g, $q \to +\infty$).

Prop. Method in $O(q^{(2g-1)/2})$ (for fixed g).

Shanks/Kangaroos: $O(q^{(2g-1)/4})$ (for fixed g).

Rem. Some improvements are possible (partial information – truncating $L(1)$, etc.).

The Frobenius endomorphism

Ordinary:

$$\begin{array}{rccc} \varphi : & \overline{\mathbb{K}} & \to & \overline{\mathbb{K}} \\ & x & \mapsto & x^q \end{array}$$

Extension to C and $\mathrm{Jac}(C)$:

$$\begin{array}{rccc} \varphi : & C(\overline{\mathbb{K}}) & \to & C(\overline{\mathbb{K}}) \\ & (X, Y) & \mapsto & (X^q, Y^q) \end{array}$$

Fundamental thm. The minimal polynomial $\chi(T)$ of φ is the reciprocal of $L(T)$. Moreover $\#\mathrm{Jac}(C)/\mathbb{F}_q = \chi(1)$.

Consequence: computing $\#\mathrm{Jac}(C)/\mathbb{F}_q$ boils down to computing $\chi(T)$.

$g = 1$: for E with $\chi(T) = T^2 - cT + q$, $|c| \leq 2\sqrt{q}$.
φ restricted to $E[\ell]$ satisfies:

$$\varphi^2 - c\varphi + q \equiv 0 \bmod \ell$$

so we can find $c_\ell \equiv c \bmod \ell$ such that

$$(X^{q^2}, Y^{q^2}) \oplus [q](X, Y) = [c_\ell](X^q, Y^q)$$

in $\mathbb{K}[X, Y]/(E, f_\ell(X))$ and use CRT once $\prod \ell > 4\sqrt{q}$. Yields a $O(\log^8 q)$ deterministic algorithm.

Pb. $\deg(f_\ell) = O(\ell^2)$.

$g > 1$: general algorithm by Pila (1990), but impossible to implement; Kampkötter (1991) for any hyperelliptic, with precise equations for $g = 2$ (uses Gröbner bases). More tomorrow!

ECC2011 summer school

September 15–16, 2011

Point counting algorithms: II. elaborate methods

F. Morain

Plan

I. What we saw yesterday.

II. Isogenies and point counting: Elkies, Atkin, Couveignes, Lercier.

III. Satoh's algorithm.

IV. Generalization to genus 2.

V. Generating cryptographically strong elliptic curves.

I. What we saw yesterday

$$\begin{array}{rccc} \varphi: & C(\overline{\mathbb{K}}) & \rightarrow & C(\overline{\mathbb{K}}) \\ & (X,Y) & \mapsto & (X^q, Y^q) \end{array}$$

Fundamental thm. The minimal polynomial $\chi(T)$ of φ is the reciprocal of $L(T)$. Moreover $\#\mathrm{Jac}(C)/\mathbb{F}_q = \chi(1)$.

Consequence: computing $\#\mathrm{Jac}(C)/\mathbb{F}_q$ boils down to computing $\chi(T)$.
$g = 1$: for E with $\chi(T) = T^2 - cT + q$, $|c| \leq 2\sqrt{q}$.
φ restricted to $E[\ell]$ satisfies:

$$\varphi^2 - c\varphi + q \equiv 0 \bmod \ell$$

so we can find $c_\ell \equiv c \bmod \ell$ such that

$$(X^{q^2}, Y^{q^2}) \oplus [q](X,Y) = [c_\ell](X^q, Y^q)$$

in $\mathbb{K}[X,Y]/(E, f_\ell(X))$ and use CRT once $\prod \ell > 4\sqrt{q}$. Yields a $O(\log^8 q)$ deterministic algorithm.
Pb. $\deg(f_\ell) = O(\ell^2)$.

II. Isogenies and point counting

A) Elements of theory

Def. $\phi : E \to E^*$, $\phi(O_E) = O_{E^*}$; induces a morphism of groups.

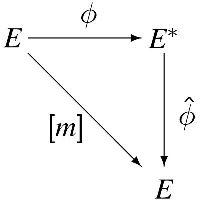
First examples

1.

$$[k](X,Y) = \left(\frac{A_k}{\psi_k^2}, \frac{B_k}{\psi_k^3}\right)$$

2. $[i](X,Y) = (-X, iY)$ on $E : Y^2 = X^3 - X$.
3. $\varphi(X,Y) = (X^q, Y^q)$, $\mathbb{K} = \mathbb{F}_q$.

Thm. (dual isogeny) There is a unique $\hat{\phi} : E^* \to E$, $\hat{\phi} \circ \phi = [m]$, $m = \deg \phi$.



Isogenies and subgroups

Thm. If F is a finite subgroup of E, then there exists ϕ and E^* s.t.

$$\phi : E \to E^* = E/F, \quad \ker(\phi) = F.$$

Ex. $E : y^2 = x^3 + ax^2 + bx$, $F = \langle (0,0) \rangle$;

$$E^* : Y^2 = X^3 - 2aX^2 + (a^2 - 4b)X,$$

$$\phi : (x, y) \mapsto \left(\frac{y^2}{x^2}, \frac{y(b - x^2)}{x^2} \right).$$

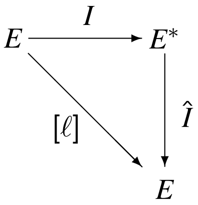
More generally: Vélu's formulas give

$$\phi(X, Y) = \left(\frac{G(X)}{H(X)^2}, \frac{J(X,Y)}{H(X)^3} \right).$$

(case $\deg \phi$ odd.)

Application to point counting

Suppose F is a subgroup of order ℓ of E:



$$I(X,Y) = \left(\frac{G}{H^2}, \ldots \right), \deg(H) = (\ell - 1)/2$$

$\ker(I) \subset E[\ell] \Rightarrow H(X) \mid f_\ell(X)$ in $\mathbb{K}[X]$.
Schoof's algorithm on a degree $O(\ell)$ polynomial.

Pb. When does such an F exist over $\mathbb{K}$?

B) Atkin and Elkies

Consider $\varphi : (X, Y) \mapsto (X^q, Y^q)$ and its restriction φ_ℓ to $E[\ell]$:

$$\varphi_\ell^2 - c\varphi_\ell + q = 0,$$

$$\Delta = c^2 - 4q.$$

If $(\Delta/\ell) = +1$, then over $\mathbb{F}_\ell$,
$\mathrm{Mat}(\varphi_\ell) \simeq \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \Leftrightarrow \exists F, \varphi(F) = F \Leftrightarrow F$ is a cyclic subgroup of order ℓ, defined over $\mathbb{K}$.

Clon. If $(\Delta/\ell) = +1, f_\ell$ has a factor of degree $(\ell-1)/2$.

Pb. How do we know that $(\Delta/\ell) = +1$?

Modular polynomials

Thm. $\exists \Phi_\ell(X,Y) \in \mathbb{Z}[X,Y]$ s.t. E and E^* are ℓ-isogenous over $\mathbb{K}$ only if $\Phi_\ell(j(E), j(E^*)) = 0$.

This polynomial comes from the theory of elliptic curves over $\mathbb{C}$: for $\Im(\tau) > 0$, $\Phi_\ell(j(\tau), j(\tau/\ell)) = 0$.

There are $O(\ell^2)$ integer coefficients of size $O(\ell) \Rightarrow \Phi_\ell$ will occupy $O(\ell^3)$ bits. This yields a naive method for computing Φ_ℓ using linear algebra.

Ex.

$$\begin{aligned}\Phi_2(X,Y) = X^3 + X^2\left(-Y^2 + 1488\,Y - 162000\right) \\ +X\left(1488\,Y^2 + 40773375\,Y + 8748000000\right) \\ +Y^3 - 162000\,Y^2 + 8748000000\,Y - 157464000000000.\end{aligned}$$

Over finite fields

Thm. $E/\mathbb{F}_q$:

$$\Phi_\ell(X, j(E)) = \begin{cases} (1)(1)(s)\cdots(s) & \text{if } (\Delta/\ell) = +1, \\ (s)\cdots(s) & \text{if } (\Delta/\ell) = -1 \end{cases}$$

and s is the order of λ_1/λ_2.

Clon. $(\Delta/\ell) = +1$ iff $\Phi_\ell(X, j(E))$ has two distinct roots over $\mathbb{K}$.

Atkin's 1986 idea: use the splitting of Φ_ℓ to deduce information on t and combine it via a clever match and sort algorithm (see also Joux/Lercier).

Elkies's algorithm (circa 1989)

repeat

1. factor $\Phi_\ell(X, j(E))$ over $\mathbb{K}$.
2. if type = $(1)(1)(s)\cdots(s)$:
   2.1 build E^*;
   2.2 build I;
   2.3 find $c \bmod \ell$;

until $\prod_{\ell \text{ good}} \ell > 4\sqrt{q}$.

Thm. $O(\log^4 q)$ operations over $\mathbb{F}_q$, probabilistic.

Computing (E^*, I)

- use the theory of elliptic curves and lattices over $\mathbb{C}$ (Weierstrass $\wp$ function); rational formulas for E^*;
- computing I takes $O(\mathsf{M}(\ell))$ operations given E, E^* and the trace of the polynomial (Bostan/M./Salvy/Schost, Lercier/Sirvent);
- in small characteristic, this is more difficult: see CouveignesI+II, DeFeo; Lercier;
- Cf. D. Robert's talks for more.

Rem. Isogenies no longer used for computing cardinalities for p small, but used for computing modular polynomials (Bröker/Lauter/Sutherland), and enters some crypto primitives (cryptosystems, discrete log attacks, isogeny walks, etc.).

Modular polynomials

Historically: precompute huge tables of Φ_ℓ over $\mathbb{Z}$ and reduce them on the fly. Convenient for crypto targets.

- Find families of "smaller" modular polynomials (Weber functions, Atkin's laundry method – theta functions, Müller with Hecke operators, etc.); e.g., $\Phi_2[j^{1/3}] = U^3 - V^2U^2 + 495\,VU + V^3 - 54000$.
- Computing Φ_ℓ given f:
  - series expansions to recover coefficients;
  - floating point computations on huge complex numbers; best method is Enge, Dupont using evaluation/interpolation for $\tilde{O}(\ell^3)$ operations;
  - alternative p-adic approach by Bröker.
  - Vercauteren: special case of $p = 2$ enables many tricks that reduce the computations.

Modern times: directly compute Φ_ℓ over the ring we're interested in. Best algorithm uses CRT and isogeny volcanoes. (Bröker/Lauter/Sutherland) in time $\tilde{O}(\ell^3)$.

Point counting records

FM; then AEnge/PGaudry/FM (first home made; NTL)

what	500dd	1000dd	1500dd	2005dd	2500dd
when	1995			2005(!)	
X^p	6h	134h	35d	133d	224d
Total	10h	180h	77d	195d	404d

A. Sutherland (07/2010): $p = 16219299585 \times 2^{16612} - 1$ (5000dd),

```
Approximate timings on AMD Phenom II 3.0 GHz cores
  Phi_n(X,j(E)) mod p                      32 CPU days
  X^p mod Phi_n(X,j(E))                   995 CPU days
  Elkies kernel polynomial h(X)             3 CPU days
  Y^p mod h and derive X^p mod h          326 CPU days
  eigenvalue using BSGS                    22 CPU days
                                        ---------------
                                         1378 CPU days
```

Every day life (crypto)

- Optimal parameters for crypto size available since 1995 (Lercier+M.).
- well understood algo + implementation (see green books for convenience).
- Implementations available in MAGMA, `pari`, ...
- An exercise in NTL, or Sage. Ditto for modular polynomials, for which tables exist.

III. Satoh's algorithm

Def. $\mathbb{Z}_p$ ring of p-adic integers $(x_1, x_2, \ldots, x_n, \ldots)$ s.t. $x_n \in \mathbb{Z}/p^n\mathbb{Z}$ and $x_{n+1} \equiv x_n \mod p^n$. Denote by $\pi : \mathbb{Z}_p \to \mathbb{F}_p$ sending x to x_1.
Def. Let $q = p^r$ and $f(t) \in \mathbb{Z}_p[t]$ s.t. $\pi(f)$ is irreducible in $\mathbb{F}_p[t]$. Then $\mathbb{Z}_q = \mathbb{Z}_p[t]/(f(t))$.
An element of $\mathbb{Z}_q$ is $\mathcal{A} = a_{r-1}t^{r-1} + \cdots + a_0$ with $a_i \in \mathbb{Z}_p$; $\mathbb{Z}_q$ contains $\mathbb{Z}_p$ as a subring.

$$\pi(\mathcal{A}) = \sum_i \pi(a_i)t^i.$$

Prop. Let σ be the **little** Frobenius sending x in $\mathbb{F}_q$ to x^p. There is a canonical way to lift σ to $\Sigma : \mathbb{Z}_q \to \mathbb{Z}_q$.

Extend σ to points $\sigma(x, y) = (\sigma(x), \sigma(y))$ and to curves: $\sigma(E) = [\sigma(a_i)]$, so that if $P \in E(\mathbb{K})$, then $\sigma(P) \in \sigma(E)(\mathbb{K})$.

Thm (Lubin-Serre-Tate) Let $E/\mathbb{F}_q$ with $j = j(E) \in \mathbb{F}_q - \mathbb{F}_{p^2}$.
There is a unique $\mathcal{J}$ in $\mathbb{Z}_q$ s.t.

$$\Phi_p(\mathcal{J}, \Sigma(\mathcal{J})) = 0,$$

$\pi(\mathcal{J}) = j$; $\mathcal{J}$ is the invariant of the **canonical lift** $\mathcal{E}$ of E and $\mathrm{End}(\mathcal{E}) = \mathrm{End}(E)$.

Isogeny cycles:

$$\begin{array}{ccccccccc}
\mathcal{E}_0 & \xrightarrow{\Sigma_{r-1}} & \mathcal{E}_{r-1} & \xrightarrow{\Sigma_{r-2}} & \cdots & \xrightarrow{\Sigma_1} & \mathcal{E}_1 & \xrightarrow{\Sigma_0} & \mathcal{E}_0 \\
\downarrow \pi & & \downarrow \pi & & & & \downarrow \pi & & \\
E_0 & \xrightarrow{\sigma_{r-1}} & E_{r-1} & \xrightarrow{\sigma_{r-2}} & \cdots & \xrightarrow{\sigma_1} & E_1 & \xrightarrow{\sigma_0} & E_0
\end{array}$$

Prop. $\varphi = \sigma_0 \circ \sigma_1 \circ \cdots \circ \sigma_{r-1}$, $\mathcal{F} = \Sigma_0 \circ \Sigma_1 \circ \cdots \circ \Sigma_{r-1}$.
Thm. $\mathrm{Tr}(\varphi) = \mathrm{Tr}(\mathcal{F})$.

Computing $\mathrm{Tr}(\mathcal{F})$ (1/2)

Use the dual of Frobenius to get another isogeny cycle amenable to computations:

$$\begin{array}{ccccccccc}
\mathcal{E}_0 & \xrightarrow{\hat{\Sigma}_0} & \mathcal{E}_1 & \xrightarrow{\hat{\Sigma}_1} & \cdots & \xrightarrow{\hat{\Sigma}_{r-2}} & \mathcal{E}_{r-1} & \xrightarrow{\hat{\Sigma}_{r-1}} & \mathcal{E}_0 \\
\downarrow \pi & & \downarrow \pi & & & & \downarrow \pi & & \\
E_0 & \xrightarrow{\hat{\sigma}_0} & E_1 & \xrightarrow{\hat{\sigma}_1} & \cdots & \xrightarrow{\hat{\sigma}_{r-2}} & E_{r-1} & \xrightarrow{\hat{\sigma}_{r-1}} & E_0
\end{array}$$

Prop. $\hat{\varphi} = \hat{\sigma}_{r-1} \circ \hat{\sigma}_{r-2} \circ \cdots \circ \hat{\sigma}_0$ (idem for $\hat{\mathcal{F}}$) and also $\mathrm{Tr}(\hat{\mathcal{F}}) = \mathrm{Tr}(\mathcal{F}) = \mathrm{Tr}(\varphi)$.

Computing $\mathrm{Tr}(\mathcal{F})$ (2/2)

Let τ (resp. τ_i) denote the local parameter of $\mathcal{E}$ (resp. $\mathcal{E}_i$).

$$\mathcal{F}(\tau) = \sum_{k \geq 1} c_k \tau^k$$

Prop. (Satoh) $\mathrm{Tr}(\mathcal{F}) = c_1 + q/c_1$.

$$c_1 = \prod_{i=0}^{d-1} g_i$$

where (Vélu's formulas again)

$$\hat{\Sigma}_i(\tau_i) = g_i \tau_i + O(\tau_i^2)$$

Satoh's algorithm in brief

1. Compute the curves E_0, E_1, E_{r-1} and their invariants j_i.
2. Lift all the j_i's simultaneously by a Newton iteration to get $\mathcal{J}_i$:

   $$\Theta((x_i)) = (\Phi_p(x_0, x_1), \Phi_p(x_1, x_2), \ldots, \Phi_p(x_{r-1}, x_0))$$

   as

   $$(x_i) \leftarrow (x_i) - ((D\Theta)^{-1}\Theta)((x_i)).$$

3. Lift each E_i coefficient by coefficient.
4. Lift the p-torsion subgroup of E_i.
5. Compute the $\hat{\Sigma}_i$'s.
6. Compute the trace.

Thm. (Satoh-FGH) For fixed p, Satoh-FGH requires $O(r^3)$ memory and $O(r^{3+\varepsilon})$ bit-operations.

IV. The situation in genus 2

- Division polynomials: Cantor.
- Schoof/Pila:
  - random curves: Gaudry/Harley ($p \approx 2^{61}$), Gaudry/Schost ($p \approx 2^{82}$), Pitcher, Gaudry/Schost (2010): $\tilde{O}((\log p)^7)$ operations in $\mathbb{F}_p$ (record $p = 2^{127} - 1$: 1000 CPU hours).
  - easy Real Multiplication: Gaudry/Kohel/Smith (2011) give a $\tilde{O}((\log p)^4)$ algorithm (record: $p \approx 2^{512}$; 128-bit takes 3 hours).
- Satoh's algorithm: LST valid. Need modular equation. Very fast for small p.
- Isogenies: Vélu's formulas for maximally isotropic kernels (Lubicz/Robert). See D. Robert, G. Bisson, R. Cosset (AVIsogenies).
- Modular polynomials: not usable yet.

Modular polynomials when $g = 2$

- **Gaudry + Schost:** the algebraic alternative is generic (Ξ_ℓ)
  - total degree is $d = (\ell^4 - 1)/(\ell - 1)$;
  - number of monomials is $O(\ell^{12})$;
  - can do $\ell = 3$: 50k but a lot of computing time (weblink still active);
  - use its factorization patterns à la Atkin to speedup cardinality computations.

- **The classical modular approach:**
  - Poincaré $\rightarrow$ Siegel (dim $2g$);
  - replace j by $(j_1, j_2, j_3) \Rightarrow$ triplet of modular polynomials, coefficients are rational fractions in j_i's;
  - Dupont (experimental conjectures proven more recently by Bröker+Lauter): stuck at $\ell = 2$ with 26.8 Mbgz (just the beginning of $\ell = 3$); uses evaluation/interpolation again; see Goren/Lauter.

V. Generating cryptographically strong curves

$\mathbb{F}_p$ with large p or $\mathbb{F}_{2^n}$ with n prime (Weil descent, see Menezes & Qu); subgroups of large prime order.

- Supersingular curves: too much structure (?).
- CM curves: quite efficient for $g = 1$ or $g = 2$, but who knows?
- Fixed curves: The NIST curves (?).
- Random curves:
  - $g = 1$: use SEA for large p, Satoh for $p = 2$. Very efficient when combined to the early-abort approach in Lercier's EUROCRYPT'97 article. Experiments conducted by FGH combining SEA and Satoh show that it takes 5 min on Alpha 750 MHz to build a good curve over $\mathbb{F}_{2^{233}}$.
  - $g = 2$ begins to be efficient (in particular RM).
  - $g > 2$: out of reach right now.