Recent improvements to the SEA algorithm in genus 1

F. Morain

Laboratoire d'Informatique de l'École polytechnique

Toronto, October 31st, 2006

Plan

I. Introduction.

II. An overview of the SEA algorithm.

III. Fast isogeny computations.

IV. Computing modular equations (AE; RD).

V. Finding the eigenvalue (PG+FM; PM+FM).

VI. Records.

RD = R. Dupont, AE = A. Enge, PG = P. Gaudry, PM = P. Mihăilescu

I. Introduction

Problem: given

$$E : y^2 + a_1 xy + a_3 y = x^3 + a_2 x^2 + a_4 x + a_6$$

defined over some finite field $\mathbf{K} = \mathbb{F}_q$, $q = p^r$, compute its cardinality.

Which methods:

- Enumeration: $O(q)$, $O(q^{1/2})$;
- Baby steps/giant steps, kangaroos, etc.: $O(q^{1/4})$;
- Any q: Schoof's algorithm (1985) and extensions $\tilde{O}((\log q)^5)$;
- p small: p-adic methods à la Satoh $\tilde{O}(r^3)$ since 1999.

In this talk: $q = p$ large, $E : y^2 = x^3 + Ax + B$; we ignore CM curves of small discriminant, as well as supersingular curves, that should be tested beforehand.

II. An overview of the Schoof-Elkies-Atkin (SEA) algorithm

Def. (torsion points) For $n \in \mathbb{N}$, $E[n] = \{P \in E(\overline{\mathbf{K}}), [n]P = O_E\}$.

Division polynomials: (for $E : y^2 = x^3 + Ax + B$)

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

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

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

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

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

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

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

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

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

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

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

Thm. $P = (x, y) \in E[\ell] \iff [2]P = O_E$ or $f_\ell(x) = 0$.

The Frobenius endomorphism

Ordinary:

$$\begin{array}{rccc} \varphi: & \overline{\mathbf{K}} & \rightarrow & \overline{\mathbf{K}} \\ & x & \mapsto & x^p \end{array}$$

Extension to E:

$$\begin{array}{rccc} \varphi: & E(\overline{\mathbf{K}}) & \rightarrow & E(\overline{\mathbf{K}}) \\ & (X, Y) & \mapsto & (X^p, Y^p) \end{array}$$

Thm. The minimal polynomial of φ is $\chi(T) = T^2 - cT + p$, $|c| \leq 2\sqrt{p}$ and $\#E = \chi(1)$.

Schoof's algorithm (1985)

The fundamental idea: let ℓ be prime to p. Then φ restricted to $E[\ell]$ satisfies

$$\varphi_\ell^2 - c\varphi_\ell + p \equiv 0 \bmod \ell$$

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

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

in $\mathbf{K}[X, Y]/(E, f_\ell(X))$ and use CRT once $\prod \ell > 4\sqrt{p}$ ($\Rightarrow \ell = O(\log p)$).

Thm. Schoof's algorithm is deterministic polynomial with bit-complexity $O(\log p \cdot \log p \mathrm{M}(\ell^2 \log p)) = \tilde{O}((\log p)^5)$.

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

Atkin and Elkies (1986–1990)

Start again from:

$$\varphi_\ell^2 - c\varphi_\ell + p = 0, \quad \Delta = c^2 - 4p.$$

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

As a consequence, f_ℓ has a factor of degree $(\ell - 1)/2$.

Fact: there exists a polynomial $\Phi_\ell(X, Y) \in \mathbb{Z}[X, Y]$ s.t. E and E^* are ℓ-isogenous over **K** iff $\#E = \#E^*$ and $\Phi_\ell(j(E), j(E^*)) = 0$.

Elkies's algorithm

for prime ℓ **until** $\prod_{\ell \text{ good}} \ell > 4\sqrt{p}$ **do**

0. Compute $\Phi_\ell(X, Y)$. [precomputation?]

1. find the roots of $\Phi_\ell(X, j(E))$ over $\mathbf{K}$; if none, use next ℓ;

2. let j_0 be one of the roots:

   2.1 build $E^* = E/F$ corresponding to j_0; deduce $f_\lambda \mid f_\ell$;

   2.2 find λ mod ℓ s.t. $\varphi_\ell(X, Y) = [\lambda](X, Y)$ mod (E, f_λ);

   2.3 $c_\ell = \lambda + p/\lambda$ mod ℓ.

Thm. $\tilde{O}((\log p)^2 \mathrm{M}(\ell \log p) = \tilde{O}((\log p)^4)$ probabilistic (half the primes are good).

III. Fast isogeny computations

INPUT: E and E^* related via an ℓ-isogeny with trace σ.
OUTPUT: $I(x) = N(x)/D(x)$.

$$E : y^2 = x^3 + Ax + B, E^* : y^2 = x^3 + \tilde{A}x + \tilde{B},$$

can be parametrized as $(x, y) = (\wp(z), \wp'(z)/2)$, where the function $\wp$ can be expanded as:

$$\wp(z) = \frac{1}{z^2} + \sum_{i \geq 1} c_i z^{2i},$$

with

$$c_1 = -\frac{A}{5}, c_2 = -\frac{B}{7}, \quad \text{for } k \geq 3, c_k = \frac{3}{(k-2)(2k+3)} \sum_{i=1}^{k-2} c_i c_{k-1-i}.$$

(see BMSS paper for fast expansion method)

Elkies's method

$$\frac{N(x)}{D(x)} = \tilde{\wp} \circ \wp^{-1}(x) = x + \sum_{i \geq 1} \frac{h_i}{x^i}$$

First: compute

$$h_k = \frac{3}{(k-2)(2k+3)} \sum_{i=1}^{k-2} h_i h_{k-1-i} - \frac{2k-3}{2k+3} A h_{k-2} - \frac{2(k-3)}{2k+3} B h_{k-3}$$

for all $k \geq 3$ with $h_1 = (A - \tilde{A})/5$ and $h_2 = (B - \tilde{B})/7$.

$\Rightarrow O(\ell^2)$ operations in **K.**

Second: get p_i's using:

$$h_i = (2i+1)p_{i+1} + (2i-1)Ap_{i-1} + (2i-2)Bp_{i-2}, \quad \text{for all } i \geq 1,$$

Third: recover $D(x)$ using Newton's formulas in $O(\ell^2)$ operations, or perhaps in $O(\mathrm{M}(\ell))$ with Schönhage's algorithm.

Total complexity: $O(\ell^2)$.

A fast variant (Bostan/M./Salvy/Schost)

Consider S s.t. $\tilde{R} = S \circ R$, with $R(z) = 1/\sqrt{\wp(z)}$ and $\tilde{R}(z) = 1/\sqrt{\tilde{\wp}(z)}$
One has:

$$S(z) = z + \frac{\tilde{A} - A}{10} z^5 + \frac{\tilde{B} - B}{14} z^7 + O(z^9) \in z + z^3 \mathbf{K}[[z^2]]$$

Claim:

$$\frac{N(x)}{D(x)} = \frac{1}{S\left(\frac{1}{\sqrt{x}}\right)^2}.$$

Applying the chain rule gives the following first order differential equation satisfied by $S(z)$:

$$(Bz^6 + Az^4 + 1)\, S'(z)^2 = 1 + \tilde{A}\, S(z)^4 + \tilde{B}\, S(z)^6.$$

Use fast computer algebra techniques to get $O(\mathsf{M}(\ell))$ method.

IV. Computing modular equations

Traditionnal modular polynomial: constructed via lattices and curves over $\mathbb{C}$. Remember that

$$j(q) = \frac{1}{q} + 744 + \sum_{n \geq 1} c_n q^n.$$

Then $\Phi_\ell^T(X, Y)$ is such that $\Phi_\ell^T(j(q), j(q^\ell))$ vanishes identically. This polynomial has a lot of properties: symmetrical $\mathbb{Z}[X, Y]$, degree in X and Y is $\ell + 1$ (hence $(\ell + 1)^2$ coefficients), etc. and moreover

Thm. [P. Cohen] the height of $\Phi_\ell^T(X, Y)$ is $O((\ell + 1) \log \ell)$.

Example:

$$\Phi_2(X, Y) = X^3 + X^2 \left(-Y^2 + 1488\, Y - 162000\right)$$

$$+X \left(1488\, Y^2 + 40773375\, Y + 8748000000\right)$$

$$+Y^3 - 162000\, Y^2 + 8748000000\, Y - 157464000000000.$$

Choosing another modular equation

Why? Always good to have the smallest polynomial so as not to fill the disks too rapidly... For small ℓ, Φ_ℓ^T is not a desperate choice.

Key point: any function on $\Gamma_0(\ell)$ (or $\Gamma_0(\ell)/\langle w_\ell \rangle$) will do. In particular, if

$$f(q) = q^{-v} + \cdots$$

then there will exist a polynomial $\Phi_\ell[f](X, Y)$ s.t.

$$\Phi_\ell[f](j(q), f(q)) \equiv 0.$$

This polynomial will have $(v+1)(\ell+1)$ coefficients, and height $O(v \log \ell)$.

Choosing f

Atkin proposed several choices:

- canonical choice $f(q)$ using some power of $\eta(q)/\eta(q^\ell)$ where:
$$\eta(q) = q^{1/24} \prod_{n \geq 1} (1 - q^n).$$
- a conceptually difficult method (the laundry method) for finding (conjecturally) the f with smallest v (that he is now able to rewrite as θ-functions with characters).

Alternatively, one may use some linear algebra on functions obtained via Hecke operators.

Computing $\Phi_\ell[f]$ given f

- **Atkin** (analysis by Elkies): use q-expansion of j and f with $O(v\ell)$ terms, compute power sums of roots of $\Phi_\ell[f]$, write them as polynomials in J and go back to coefficients of $\Phi_\ell[f](X, J)$ via Newton's formulas; use CRT on small primes. $\tilde{O}(\ell^3 \mathrm{M}(p))$; used for $\ell \leq 1000$ fifteen years ago.
- **Charles+Lauter (2005):** compute Φ_ℓ^T modulo p using supersingular invariants mod p, Mestre *méthode des graphes*, ℓ torsion points defined over $\mathbb{F}_{p^{O(\ell)}}$ and interpolation. $\tilde{O}(\ell^4 \mathrm{M}(p))$
- **Enge (2004); Dupont (2004):** use complex floating point evaluation and interpolation. $\tilde{O}(\ell^3)$

Real life (Enge)

- Use
$$\frac{T_r(\eta\eta_\ell)}{\eta\eta_\ell}$$
where T_r is the Hecke operator
$$(T_r|f)(\tau) = f(r\tau) + \frac{1}{r}\sum_{k=0}^{r-1} f\left(\frac{\tau+k}{r}\right)$$
for some (small) r. Total overall cost $\tilde{O}(r\ell^3)$.
- Evaluation of η using the sparse expansion, $O(\sqrt{H})$ arithmetical operations per value: $O(\ell^2\sqrt{H}\mathsf{M}_{\mathrm{int}}(H))$.

Rem. sometimes, a combination of T_r's is better (i.e., smaller order v), but then evaluation is more costly.

Examples

ℓ	r	H	$\deg(J)$	eval(s)	interp(s)	tot (d)	Mb gz
3011	5	7560	200				368
3079	97	9018	254	7790	640	23	547
3527	13	9894	268	799	1440	3	746
3517	97	10746	290	12400	1110	42	850
4003	13	11408	308	1130	2320	4	1127
5009	5	13349	334	880	3110	3	1819
6029	5	16418	402	1550	6370	7	3251
7001	5	19473	466	2440	11700	13	5182
8009	5	22515	534	3500	20000	22	7905
9029	5	25507	602	5030	33100	35	11460
10079	5	28825	672	7690	56300	61	16152

V. Finding the eigenvalue

Pb: find λ, $1 \leq \lambda < \ell$ s.t.

$$(X^p, Y^p) = [\lambda](X, Y) \bmod (E, f_\lambda(X)).$$

A) previous methods

First approach: $O(\ell)$ iterations to find λ given X^p and Y^p.

When $\ell \equiv 3 \bmod 4$: enough to test $X^p = [\lambda](X)$ using Dewaghe's trick.

Maurer + Müller (1994/2001): [funny baby-steps/giant steps] find i and j s.t. $[i](X^p) = [j](X)$, with $i, j = O(\sqrt{\ell})$ yielding a $O(\sqrt{\ell}\mathrm{M}(\ell))$ method (given X^p).

Gaudry + FM (ISSAC 2006): practical improvements, for instance how to get X^p from Y^p; better constants in MM.

Some timings

For p with 1700dd, $\ell = 3881$:

$X^p \bmod \Phi$	17529
find j^* (deg=257)	1398
f_λ	2930
Y^p	8768
X^p from Y^p	2063
$j/i = 31/29$	
all N_j/D_j	149
$f_u(X^p)$	300
matchs	310

B) Abelian lifts (P. Mihăilescu)

(Joint work in progress...)

Finding λ: $O((\log p)\mathrm{M}(\ell) + \sqrt{\ell}\mathrm{M}(\ell))$.

Question: can we get rid of the $\log p$ term? Yes, in some cases.

Philosophy: f_λ behaves very much like a cyclotomic polynomial after all. Why not transfer all the theory?

First idea: factor f_λ, but requires $X^p \bmod f_\lambda$.

Second idea: use Gaussian periods, but then need $[a]X$ for $a \leq (\ell - 1)/2$. Cost is $O(\ell\mathrm{M}(\ell))$, ok if $\ell \ll \log p$, but in real life, $\ell = \log p$.

Third idea: look more closely at cyclotomic properties, or Abelian properties.

Principle: Let prime power $q = r^a \parallel d = (\ell - 1)/2$, $Q = (\ell - 1)/2/q$.

Write $(\mathbb{Z}/\ell\mathbb{Z})^* = \langle c \rangle$ and write $\lambda = c^x$. We will find $u = x \bmod q$.

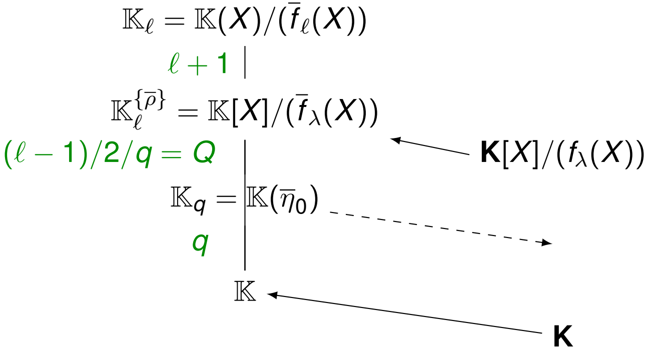
W.l.o.g: q odd.

Notation:

$$f_\lambda(Z) = \prod_{a=1}^{(\ell-1)/2} (Z - \rho_a(X))$$

where
$\rho_a(X) = ([a]P)_x$ in $\mathbf{K}[X]/(f_\lambda(X))$ and $1 \leq a \leq (\ell - 1)/2$.

Deuring lift $E/\mathbb{F}_p$ to $\overline{E}/\mathbb{K}$ and p to $\mathfrak{p}$.



There is an Abelian action:

$$\overline{\rho}_{ij} = \overline{\rho}_i\overline{\rho}_j = \overline{\rho}_j\overline{\rho}_i.$$

$\overline{f}_\lambda(Z) = \prod_{a=1}^{(\ell-1)/2}(Z - \overline{\rho}_a(X))$ is an Abelian lift of $f_\lambda(Z)$.

Elliptic Gaussian period

Let $(\mathbb{Z}/\ell\mathbb{Z})^*/\{\pm 1\} = \langle c \rangle$ and put:

$$(\mathbb{Z}/\ell\mathbb{Z})^*/\{\pm 1\} = H \times K = \langle h \rangle \times \langle k \rangle \quad \text{with } h = c^q, k = c^Q.$$

For $0 \leq i < q$:

$$\overline{\eta}_i = \sum_{a \in H} ([k^i \cdot a]\overline{P})_x$$

Since $\overline{\eta}_1 = \overline{\eta}_0 \circ \overline{\rho}_k$, there is a cyclic action:

$$\overline{\eta}_0 \stackrel{\overline{\rho}_k}{\longrightarrow} \overline{\eta}_1 \stackrel{\overline{\rho}_k}{\longrightarrow} \cdots \stackrel{\overline{\rho}_k}{\longrightarrow} \overline{\eta}_{q-1} \stackrel{\overline{\rho}_k}{\longrightarrow} \overline{\eta}_0,$$

The minimal polynomial of $\overline{\eta}_0$ is:

$$\overline{M}(T) = \prod_{i=0}^{q-1} (T - \overline{\eta}_i)$$

and belongs to $\mathbb{K}[T]$.

Fact: since the extension $\mathbb{K}_q/\mathbb{K}$ is Abelian, there exists $\overline{C}(T) \in \mathbb{K}[T]$ of degree $\leq q-1$ s.t. $\overline{\eta}_1 = \overline{C}(\overline{\eta}_0)$.

Reduce everything modulo p: η_0 and η_1 live in $\mathbb{F}_p[X]/(f_\lambda(X))$ and are related through $\eta_1 = C(\eta_0)$, $M(\eta_0) = M(\eta_1) = 0$.

Suppose $T^p = C^{(v)}(T) \bmod M(T)$. Then

$$\eta_0^p = C^{(v)}(\eta_0) = \eta_v = [k^v]\eta_0.$$

But $\eta_0^p = [\lambda]\eta_0$ and therefore $c^u \equiv c^{Qv}$ or $u \equiv Qv \bmod q$.

Algorithm

Aim: given $q \mid\mid (\ell - 1)/2$, compute u mod q where $\lambda = c^u$.

1. Compute $\eta_0(X) \in \mathbb{F}_p[X]/(f_\lambda)$.
   Shoup's trace algorithm in $O((\log Q)(\mathcal{C}_2(\ell) + 0.5\mathcal{C}_3(\ell))$.
2. Compute $\eta_1(X) = \eta_0 \circ \rho_k(X) \bmod f_\lambda(X)$.
   $O(\mathcal{C}_1(\ell))$.
3. Compute the minimal polynomial $M(T)$ of η_0 mod f_λ.
   Shoup: $O(\mathrm{M}(q)q^{1/2} + q^2)$.
4. Compute $C(T)$ s.t. $\eta_1(X) = C(\eta_0(X))$.
   Shoup: $O(\ell^{(\omega+1)/2})$.
5. Compute $T_p = T^p \bmod M(T)$.
   $O((\log p)\mathrm{M}(q))$.
6. Find $0 \leq v < q$ s.t. $T_p = C^{(v)}(T) \bmod M(T)$.
   $O(q^{1/2}\mathcal{C}_{\sqrt{q}}(q))$.
7. Return vQ mod q.

$\mathcal{C}_r(\ell) = O(r^{1/2}\ell^{1/2}\mathrm{M}(\ell) + r^{(\omega-1)/2}\ell^{(\omega+1)/2})$ (Comp[23]Mod of NTL).

Trace computation: computing η_0 is analogous to Shoup's algorithm for computing

$$T_k(X) = \sum_{i=0}^{k} X^{p^i} \bmod f$$

using $T_{a+b} = T_a(X^{p^b}) + T_b$, hence $O(\log k)$ modular compositions by a divide-and-conquer algorithm.

Analysis:
When $q \ll \ell$: dominant step is step 1 in $O((\log Q)\mathcal{C}(\ell)) = O((\log \ell)\mathcal{C}(\ell))$.

When $q \approx \ell$: dominant term is step 5 in $O((\log p)\mathrm{M}(\ell)) \Rightarrow$ clearly not useful in that case.

A real life example

$p = 10^{2499} + 7131$, $\ell = 5861$, $\ell - 1 = 2^2 \cdot 5 \cdot 293$.

q	η_0	η_1	$M(T)$	$C(T)$	T^p	u
4	15418	732	13	100	2	0
5	8491	446	17	43	10	0
293	3615	446	160	2509	3203	250

for a total time of 36800 sec.

Traditional approach: Y^p costs 33001, X^p (from Y^p) 898; λ final is 3650.

Any improvement to $\mathcal{C}_r$ or trace computation would be crucial.

VI. Records

Modular equations computed using gmp, mpfr, mpc (C language).

SEA++ written in C++ (NTL).

Times for computing the cardinality of $E : Y^2 = X^3 + 4589X + 91128$ modulo the smallest p with given $\#$ dd, on an AMD 64 Processor 3400+ (2.4GHz).

what	500dd	1000dd	1500dd	2005dd	2100dd
X^p	6h	134h	35d	133d	121d
Total	10h	180h	77d	195d	190d

What's left to be done?

- Mihăilescu's approach: injecting more cyclotomic properties seems promising (Gauss and Jacobi sums, etc.).
- Computing E^* from E is a $O(\ell^2)$ process. Can we go down to $O(\mathrm{M}(\ell))$???
- Modular equations still the stumbling block of all this (as a result, AE has filled all our disks...). Can we dream of doing without Φ's????
- Much much harder: still a lot of work to be done in higher genus.