Computation of class polynomials for abelian surfaces

A. Enge[1], E. Thomé[2]

[1] INRIA/LFANT, Bordeaux ; [2] INRIA/CARAMEL, Nancy.

Oct. 9th, 2014

Plan

Introduction

CM in genus 1

Genus 2 prerequisites

Algorithm

Computer experiments

Motivations

Algebraic curves over finite fields are nice groups for cryptography.

Desired features:

- Compact representation of elements.
- Fast arithmetic.
- Hard discrete log problem.
  $\Rightarrow$Prefer almost prime group order.

Motivations

Algebraic curves over finite fields are nice groups for cryptography.

Desired features:
- Compact representation of elements.
- Fast arithmetic.
- Hard discrete log problem.
  $\Rightarrow$Prefer almost prime group order.

Typical candidates.

- Elliptic curves ($g = 1$)
  - Studied for crypto for 25+ years.
  - Efficient, secure.
- (Jacobians of) genus 2 curves.
  - Smaller base field for comparable group size.
  - Almost similar efficiency due to recent progress.
  - DL is hard as well.
- Higher genus: DL is comparatively easier. Avoided.

The cardinality issue

Strategy 1. Direct point counting.

Pick a curve at random (or select based on arithmetic properties). Compute $\#E(\mathbb{F}_p)$ (or $\#\operatorname{Jac}_{\mathcal{C}}(\mathbb{F}_p)$).

- Polynomial. Very fast for small characteristic (p-adic).
- $g = 1$: fast enough for crypto purposes (ℓ-adic, SEA).
- $g = 2$: now also possible, with some effort (SEA-like).

Strategy 2

Select some family of curves for easy point counting. Obtain an instance $(\mathbb{F}_p, E(\mathbb{F}_p), \#E(\mathbb{F}_p))$.

- The CM method is such a strategy.

Plan

Elliptic curves

Elliptic curves

- The moduli space of elliptic curves has dimension 1.
- It is parameterised by the j-invariant.
  Example: $y^2 = x^3 + ax + b \rightsquigarrow j = 1728\,\frac{4a^3}{4a^3+27b^2}$.

Endomorphism rings of elliptic curves classified by Deuring.

- In char. 0, either $\mathbb{Z}$ or an order in $\mathbb{Q}(\sqrt{D})$, for some $D < 0$.
- Over finite fields, ordinary: cannot be $\mathbb{Z}$.
  Any ordinary curve over $\mathbb{F}_p$ is the reduction of a curve over $\mathbb{C}$ with same $\mathrm{End}(E)$.

Strategy

- Pick an imaginary quadratic field $K = \mathbb{Q}(\sqrt{d})$; let $\mathcal{O}_K = \mathbb{Z} + \omega\mathbb{Z}$.
- Let $\mathcal{O} = \mathbb{Z} + f\omega\mathbb{Z}$ be an order in $\mathcal{O}_K$, $\mathrm{disc}(\mathcal{O}) = D = f^2 d$.

Aim at E mod some (yet unknown) p, with $\mathrm{End}(E) = \mathcal{O}$.

- First list E over $\mathbb{C}$ with CM by $\mathcal{O}$.
- j-invariants: roots of Hilbert class polynomial $H_D(x) \in \mathbb{Z}[x]$.
- Appropriate p have $p = \mathrm{Norm}(\pi \in \mathcal{O}_K,$ Weil number$)$.
- Roots of H_D mod p are j-invariants of their reductions.
- Those have $\#E(\mathbb{F}_p) = p + 1 \pm \mathrm{Tr}\,\pi$.

Effective complex multiplication

Given D, what are the curves over $\mathbb{C}$ with CM by $\mathcal{O}$?

Take $\mathfrak{a} = (\alpha_1, \alpha_2)$ ideal of $\mathcal{O}$ with $\Im(\tau = \frac{\alpha_2}{\alpha_1}) > 0$.

- $\mathbb{C}/_{\mathfrak{a}}$ has CM by $\mathcal{O}$.
  $j(\mathfrak{a}) := j(\tau)$ depends only on the ideal class of $\mathfrak{a}$.
  j is a modular function for the action of $\mathrm{SL}_2(\mathbb{Z})$ on$\mathcal{H}_1$.
- Curve with invariant $j(\mathfrak{a})$ has CM by $\mathcal{O}$,
- There are $h = \#\,\mathrm{Cl}(\mathcal{O})$ such curves (faithful action).

Main algorithm

- Fix $D < 0$ and Weil number π.
- Enumerate the h ideal classes of $\mathcal{O}_D$:

$$\left(A_i, \frac{-B_i + \sqrt{D}}{2}\right)$$

- Compute over $\mathbb{C}$ the class polynomial

$$H(X) = \prod_{i=1}^{h} \left(X - j\left(\frac{-B_i + \sqrt{D}}{2A_i}\right)\right) \in \mathbb{Z}[X]$$

- Find a root $\bar{j}$ modulo $p = \operatorname{Norm} \pi$.
- Curve with that invariant $\bmod p$ has $\#E = p + 1 \pm \operatorname{Tr} \pi$.

Complexity

- Size of H
  - Degree $h \in \tilde{O}\left(\sqrt{|D|}\right)$;
  - Coefficients with $\tilde{O}\left(\sqrt{|D|}\right)$ digits ;
  - Total size $\tilde{O}(|D|)$
- Evaluation of j: $\tilde{O}\left(\sqrt{|D|}\right)$
  - Precision: $\tilde{O}\left(\sqrt{|D|}\right)$ digits ;
  - Multievaluation of the “polynomial” j ;
  - Arithmetic-geometric mean.
- Total complexity

  $\tilde{O}(|D|)$ — quasi-linear in the output size.

Implementation

Record with complex analytic CM (Enge 2009):

- $D = -2\,093\,236\,031$;
- $h = 100\,000$;
- Precision 264 727 bits;
- 260 000 seconds = 3 days CPU time;
- 5 GB;
- benefited from using alternative class invariants.

Free, available software, based notably on MPFR/MPC/MPFRCX.

Further algorithms

See Belding–Bröker–Enge–Lauter 2008 and further works for comparison of other methods.

- *p*-adic lift.
- Chinese remaindering (CRT):
  - Enumerate CM curves over $\mathbb{F}_p$, compute H mod p;
  - Lift to $\mathbb{Z}$ or directly to $\mathbb{Z}/P\mathbb{Z}$.
- CRT has the edge for records (Enge–Sutherland 2010):
  - $D = -1\,000\,000\,013\,079\,299$;
  - $h = 10\,034\,174$;
  - $P \approx 2^{254}$;
  - Precision 21 533 832 bits;
  - 438 709 primes of $\leq$ 53 bits;
  - 200 days CPU time;
  - Size mod $P \approx 200$ MB;
  - Size over $\mathbb{Z} \approx 2$ PB (not computed explicitly).

AGM

Dupont: One can evaluate j at precision n in time

$$O(\mathrm{M}(n)\log n) = \tilde{O}(n).$$

Idea of the algorithm

- Newton iterations on a function built with the arithmetic-geometric mean (AGM).
- $j(\tau)$ is a zero of this function.

Genus 1 Theta constants — definition

$$a, b \in \frac{1}{2}\mathbb{Z}/\mathbb{Z}; \qquad q = e^{2\pi i\tau}$$

$$\theta_{a,b}(\tau) = \sum_{n\in\mathbb{Z}} e^{(2\pi i)(n+a)\tau(n+a)/2+(n+a)b} = e^{2\pi iab}\sum_{n\in\mathbb{Z}}(e^{2\pi ib})^n q^{(n+a)^2/2}$$

$$\begin{aligned}
\theta_{0,0}(\tau) &= \sum_{n\in\mathbb{Z}} q^{n^2/2} = 1 + 2q^{1/2} + 2q^2 + 2q^{9/2} + \dots \\
\theta_{0,\frac{1}{2}}(\tau) &= \sum_{n\in\mathbb{Z}} (-1)^n q^{n^2/2} = 1 - 2q^{1/2} + 2q^2 - 2q^{9/2} + \dots \\
\theta_{\frac{1}{2},0}(\tau) &= \sum_{n\in\mathbb{Z}} q^{(2n+1)^2/8} = q^{1/8}\left(1 + 2q + 2q^3 + \dots\right) \\
\theta_{\frac{1}{2},\frac{1}{2}}(\tau) &= 0
\end{aligned}$$

Theta constants — duplication formulæ

$$\theta_{0,0}^2(2\tau) = \frac{\theta_{0,0}^2(\tau) + \theta_{0,\frac{1}{2}}^2(\tau)}{2} \quad \theta_{0,\frac{1}{2}}^2(2\tau) = \sqrt{\theta_{0,0}^2(\tau)\theta_{0,\frac{1}{2}}^2(\tau)}$$

AGM

$$\theta_{0,0}^2(2\tau) = \frac{\theta_{0,0}^2(\tau) + \theta_{0,\frac{1}{2}}^2(\tau)}{2} \quad \theta_{0,\frac{1}{2}}^2(2\tau) = \sqrt{\theta_{0,0}^2(\tau)\theta_{0,\frac{1}{2}}^2(\tau)}$$

AGM for $a, b \in \mathbb{C}$

- $a_0 = a,\ b_0 = b$
- $a_{n+1} = \frac{a_n + b_n}{2}$
- $b_{n+1} = \sqrt{a_n b_n}$, closer to a_{n+1} than to its opposite.
- converges quadratically towards a common limit $\mathrm{AGM}(a, b)$

Evaluated in time $O(\mathrm{M}(n) \log n)$ at precision n.

For $\tau \in$ some region of $\mathcal{H}_1$,

$$\left\{ \left(\theta_{0,0}^2, \theta_{0,\frac{1}{2}}^2 \right) (2^n \tau) \right\}$$

is the AGM sequence starting from τ (whence the limit is 1).

Theta quotients

The AGM is an homogeneous bivariate function on $\mathbb{C}$. We define:

$$\mathrm{AGM}(a, b) = a \cdot \mathrm{AGM}(1, b/a) =: a \cdot M(b/a)$$

- $k'(\tau) = \left(\frac{\theta_{0,\frac{1}{2}}(\tau)}{\theta_{0,0}(\tau)}\right)^2$
- $k(\tau) = \left(\frac{\theta_{\frac{1}{2},0}(\tau)}{\theta_{0,0}(\tau)}\right)^2$
- $k^2(\tau) + {k'}^2(\tau) = 1$
- $j = 256\frac{(1-{k'}^2+{k'}^4)^3}{{k'}^4(1-{k'}^2)^2}$

j can be computed from k'

Newton iterations

- $M(k'(\tau)) = \frac{1}{\theta_{0,0}^2(\tau)}$,
- $M(k(\tau)) = M(k'(-1/\tau)) = \frac{1}{\theta_{0,0}^2(-1/\tau)} = \frac{i}{\tau\theta_{0,0}^2(\tau)}$,
- $k^2(\tau) + k'^2(\tau) = 1$
- $f_\tau(x) = iM(x) - \tau M(\sqrt{1-x^2})$
- $f_\tau(k'(\tau)) = 0$

$$x_{n+1} \leftarrow x_n - \frac{f_\tau(x_n)}{f'_\tau(x_n)}$$

converges quadratically towards $k'(\tau)$

Evaluated in time $O(\mathrm{M}(n)\log n)$ at precision n.

Caution

Care must be taken to consider τ for which the homogeneous AGM converges to 1 (which gives $M(k'(\tau)) = \frac{1}{\theta_{0,0}^2(\tau)}$).

Plan

Introduction

CM in genus 1

Genus 2 prerequisites

Algorithm

Computer experiments

Generalization: Genus 2 CM

Let K be a CM field.

$$\begin{array}{c l} K & \\ \Big| & 2\text{, totally imaginary} \\ K_0 & \\ \Big| & g = 2\text{, totally real} \\ \mathbb{Q} & \end{array}$$

Workplan

- Enumerate principally polarized abelian varieties (PPAVs) with complex multiplication by $\mathcal{O}_K$ ($\mathrm{End} = \mathcal{O}_K$).
- Compute their invariants in $\mathbb{C}$ ($g = 2$, three invariants).
- Compute their defining polynomials: Igusa class polynomials.
- Recognize these (triples of) polynomials in $\mathbb{Q}[x]$.

The larger the discriminants, the bigger the polynomials.

Various approaches

- Complex analytic method: Spallek, Weng, Streng.
- p-adic: Gaudry, Houtmann, Kohel, Ritzenthaler, Weng, Carls, Lubicz.
- CRT: Eisentrager, Lauter, Bröker, Gruenewald, Robert.

Focus on the complex analytic method

- Streng: complete algorithm, and complexity upper bounds.
- Improve on keypoint: computation of invariants analytically.
- Recognize irreducible factors of class polynomials.

(1/5): CM fields

$$\begin{array}{c} K \\ | \\ K_0 \\ | \\ \mathbb{Q} \end{array}$$

- Preferred defining equation for K: $x^4 + Ax^2 + B$, with $A^2 - 4B = \square \times \mathrm{disc}(K_0)$.
- Let $D = \mathrm{disc}(K_0)$, and A minimal $\Rightarrow$ invariants $[D, A, B]$.

The CM field K may be either:

- Galois with $\mathrm{Gal}(K/\mathbb{Q}) = \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$; (degenerates to $g = 1$).
- Galois with $\mathrm{Gal}(K/\mathbb{Q}) = \mathbb{Z}/4\mathbb{Z}$; (cyclic case, rare).
- non-Galois, with $\mathrm{Gal}(L/\mathbb{Q}) = D_4 = \mathbb{Z}/4\mathbb{Z} \rtimes \mathbb{Z}/2\mathbb{Z}$; (typical). Study of the Galois structure reveals:
    - two non-conjugate pairs of embeddings $K \hookrightarrow \mathbb{C}$;
    - the reflex field K^r of K, which is another CM field.

(2/5): Period matrices

Siegel upper-half space $\mathcal{H}_2$: symm. + pos. def. imag. part.

- $\mathrm{Sp}_4(\mathbb{Z})$ acts on $\mathcal{H}_2$: $\begin{pmatrix} A & B \\ C & D \end{pmatrix}.\tau = (A\tau + B)(C\tau + D)^{-1}$.
- $\mathcal{F}_2$: fundamental domain for $\mathrm{Sp}_4 \backslash \mathcal{H}_2$.

PPAV= $\begin{array}{l} \mathbb{Z}\text{-lattice in } \mathbb{C}^2 \\ +\text{Riemann form} \end{array}$ $\rightarrow$ period matrix $\tau = \begin{pmatrix} \tau_1 & \tau_3 \\ \tau_3 & \tau_2 \end{pmatrix} \in \mathcal{H}_2$.

(3/5): θ-constants in genus 2

Theta constants for $\mathbf{a} = (a_1, a_2)$, $\mathbf{b} = (b_1, b_2)$, $a_i, b_i \in \{0, 1/2\}$:

$$\theta_{[\mathbf{a},\mathbf{b}]}(\tau) = \sum_{\mathbf{n}\in\mathbb{Z}^2} \exp\left(i\pi\left[(\mathbf{n}+\mathbf{a})\tau(\mathbf{n}+\mathbf{a})^t + 2(\mathbf{n}+\mathbf{a})\mathbf{b}^t\right]\right).$$

- Numbering (Dupont) $\theta_{[\mathbf{a},\mathbf{b}]} = \theta_{2b_1+4b_2+8a_1+16a_2}$.
- 10 even theta constants: $\theta_{0,1,2,3,4,6,8,9,12,15}$, other are 0.

Theta constants are used to compute invariants.

Duplication formulae

We have unambiguous formulae:

4-uple $(\theta_{0,1,2,3}(\tau/2)) \to$ 10-uple $(\theta^2_{0,1,2,3,4,6,8,9,12,15}(\tau))$.

(4/5): invariants of genus 2 curves

The moduli space of 2-dimensional PPAVs has dimension 3.

Igusa invariants can be computed from $\theta_{0,1,2,3,4,6,8,9,12,15}$.

- Several invariant sets floating around.
- Some “smaller” than others.
- Define (i_1, i_2, i_3) as those proposed by Streng.

$$i_1 = \frac{I_4(I_2 I_4 - 3I_6)}{2I_{10}} \quad i_2 = \frac{I_2 I_4^2}{I_{10}} \quad i_3 = \frac{I_4^5}{I_{10}^2}.$$

(5/5): Class polynomials

Consider $S(K)$ the set of PPAVs with CM by $\mathcal{O}_K$.

The set $\{i_1(\tau),\ \tau \in S(K)\}$ is defined over $\mathbb{Q}$.

- Minimal polynomials H_1, H_2, H_3 in $\mathbb{Q}[x]$.
- Better: $\{i_{1,2,3}(\tau)\}$ a 0-dimensional set in $\mathbb{C}^3$, defined over $\mathbb{Q}$.
- Triangular (Hecke) representation: H_1, $\hat{H}_2$, $\hat{H}_3$, with:

$$\hat{H}_2(i_1) = H_1'(i_1) i_2.$$

The triple $(H_1, \hat{H}_2, \hat{H}_3)$ is our target.

Obstacles:

- Large degree, (very) large coefficients.
- Need large precision for complex invariants, so that rational polynomials may be recognized.

Plan

Introduction

CM in genus 1

Genus 2 prerequisites

Algorithm

Computer experiments

Workplan (again)

$$
\begin{array}{c}
\text{List period matrices} \\
\downarrow \\
\text{Compute } \theta\text{-constants} \\
\downarrow \\
\text{Compute class polynomials} \\
\downarrow \\
\text{Compute a curve example}
\end{array}
$$

Plan

Algorithm

Principally polarized abelian varieties with CM by $\mathcal{O}_K$

Computing complex invariants

From θ-constants to class polynomials

PPAVs with CM by $\mathcal{O}_K$

$\mathcal{O}_K$-ideals to represent PPAVs.

Let $\mathfrak{a}$ be an $\mathcal{O}_K$-ideal with:

- $(\mathfrak{a}\bar{\mathfrak{a}}\mathcal{D}_{K/\mathbb{Q}})^{-1} = (\xi)$,
- $\Phi(\xi) \in i\mathbb{R}^{+*}$ for some CM-type Φ.

- Such $\mathfrak{a}$'s yield period matrices $\Omega \in \mathcal{M}_2(K^r) \hookrightarrow \mathcal{H}_2 \twoheadrightarrow \mathcal{F}_2$.
- Conversely, all PPAVs with CM by $\mathcal{O}_K$ are obtained this way.

Easy plan: enumerate representatives of $\mathrm{Cl}(\mathcal{O}_K)$ to find both.

Way more satisfactory: enumerate only irreducible components, working with Shimura group $\mathfrak{C}(K)$ and the reflex typenorm map.

Plan

Algorithm

Principally polarized abelian varieties with CM by $\mathcal{O}_K$

Computing complex invariants

From θ-constants to class polynomials

Computing theta constants

Input: $\tau \in \mathcal{F}_2$, whose entries are algebraic numbers (in K^r).
Goal: theta constants $\theta_{0,1,2,3,4,6,8,9,12,15}$ (and later $i_{1,2,3}$).

Large precision N needed to successful reconstruct $H_1, \hat{H}_2, \hat{H}_3$.

Upper bounds on N exist. Difficult to make it tight.

Two strategies for computing θ's from τ.

- q-expansion of $\theta_{0,1,2,3}(\tau/2)$, letting $q_k = \exp(i\pi\tau_k/2)$:

$$\theta_{4b_1+2b_2}(\tau/2) = \sum_{m,n\in\mathbb{Z}} (-1)^{2(mb_1+nb_2)} q_0^{m^2} q_1^{2mn} q_2^{n^2}.$$

  Summation over $O(N)$ terms, can be done in $O(N\mathcal{M}(N))$.
  Finish with duplication formulae.

- Faster: Newton lifting.

Borchardt mean

Dupont defines a Borchardt sequence as $((x_n, y_n, z_n, t_n) \in \mathbb{C}^4)$:

$$x_{n+1} = \frac{1}{4}(x_n + y_n + z_n + t_n), \qquad y_{n+1} = \frac{1}{2}(\sqrt{x_n}\sqrt{y_n} + \sqrt{z_n}\sqrt{t_n}),$$
$$z_{n+1} = \frac{1}{2}(\sqrt{x_n}\sqrt{z_n} + \sqrt{y_n}\sqrt{t_n}), \quad t_{n+1} = \frac{1}{2}(\sqrt{x_n}\sqrt{t_n} + \sqrt{y_n}\sqrt{z_n}).$$

- Choice of $\sqrt{}$ at each iteration.
- Starting (x_0, y_0, z_0, t_0): set of possible limits $\mathcal{B}_2(x_0, y_0, z_0, t_0)$.
- Forcing consistent choice of roots: $B_2(x, y, z, t)$ well defined.

Let $\mathcal{U} = \{\tau \in \mathcal{H}_2,\ B_2(\theta^2_{0,1,2,3}(\tau)) = 1\}$. At least $\mathcal{F}_2 \subset \mathcal{U}$.

Homogeneity

$$B_2(\lambda x, \lambda y, \lambda z, \lambda t) = \lambda B_2(x, y, z, t).$$

Exploiting action of $\mathrm{Sp}_4(\mathbb{Z})$

Action of Γ_2 on the theta constants

Let $\tau \in \mathcal{H}_2$. Then

$$\left(\theta_j^2((\mathfrak{J}\mathfrak{M}_1)^2\tau)\right)_{j=0,1,2,3} = -i\tau_1 \left(\theta_j^2(\tau)\right)_{j=4,0,6,2},$$
$$\left(\theta_j^2((\mathfrak{J}\mathfrak{M}_2)^2\tau)\right)_{j=0,1,2,3} = -i\tau_2 \left(\theta_j^2(\tau)\right)_{j=8,9,0,1},$$
$$\left(\theta_j^2((\mathfrak{J}\mathfrak{M}_3)^2\tau)\right)_{j=0,1,2,3} = (\tau_3^2 - \tau_1\tau_2) \left(\theta_j^2(\tau)\right)_{j=0,8,4,12}.$$

Important: if $(\mathfrak{J}\mathfrak{M}_1)^2.\tau \in \mathcal{U}$, then $B_2(\theta_{4,0,6,2}^2(\tau)) = \frac{1}{-i\tau_1}$.

Conjecture

For $\tau \in \mathcal{F}_2$, $i \in \{0,1,2\}$: $(\mathfrak{J}\mathfrak{M}_i)^2.(\tau) \in \mathcal{U}$.

$\theta_{0,1,2,3}(\tau/2)$ as solutions of an equation

Input: $\tau \in \mathcal{F}_2$ known (to any precision we like).

Initially: low-precision $\theta_{0,1,2,3}(\tau/2)$.

- Use duplication formulae to deduce $\theta^2_{0,1,2,3,4,6,8,9,12,15}(\tau)$.
- Use B_2 computations to deduce coefficients of τ.
- The accurate $x_{0,1,2,3} = \theta_{0,1,2,3}(\tau/2)$ are solutions to

$$\text{complicated-}B_2\text{-calculation}(x_{0,1,2,3}) = \tau.$$

Newton: use this feedback loop to find $\theta_{0,1,2,3}(\tau/2)$.

- Keeping track of derivatives is messy.
- A secant method also works, and is actually more convenient.

Computation of $\theta^2_{0,1,2,3}$ by Newton lifting

Convergence of the Newton iteration is quadratic:

- each iteration (almost) doubles the precision.
- it is possible to "lift higher" without restarting from scratch.

Complexity of the algorithm: quasi-linear $O(\mathcal{M}(N) \log N)$.

Performance measurements

	$\tau = \begin{pmatrix} \frac{-1+5i}{2} & \frac{i}{6} \\ \frac{i}{6} & \frac{-1+7i}{2} \end{pmatrix}$			$\tau = \begin{pmatrix} \frac{2+10i}{7} & \frac{1+2i}{6} \\ \frac{1+2i}{6} & \frac{4}{10}+8i \end{pmatrix}$		
bits	MAGMA	cmh-naive	cmh-Newton	MAGMA	cmh-naive	cmh-Newton
$\approx 2^{11}$	0.46	0	0.02	0.03	0	0.02
$\approx 2^{12}$	3.4	0.01	0.04	0.17	0.04	0.03
$\approx 2^{13}$	26	0.07	0.08	1.1	0.20	0.09
$\approx 2^{14}$	210	0.31	0.24	8.2	1.0	0.26
$\approx 2^{15}$	1700	1.3	0.69	60	5.2	0.75
$\approx 2^{16}$		6.4	2.0	430	27	2.2
$\approx 2^{17}$		32	5.7	3100	130	6.0
$\approx 2^{18}$		160	16		720	16
$\approx 2^{19}$		770	39		3100	40
$\approx 2^{20}$		3200	98			96
$\approx 2^{21}$			240			230
$\approx 2^{22}$			560			530
$\approx 2^{23}$			1400			1300
$\approx 2^{24}$			3200			3000
$\approx 2^{25}$			7600			7100
$\approx 2^{26}$			16000			16000

Table:

Computation of $\theta_0(\tau)$ (Intel i5-2500, 3.3GHz; MAGMA-2.19.4; cmh-1.0)

Plan

Algorithm

Principally polarized abelian varieties with CM by $\mathcal{O}_K$

Computing complex invariants

From θ-constants to class polynomials

Reconstruction

θ-constants $\rightsquigarrow$ three Igusa invariants : trivial.

From these, we compute:

- product trees yield $H_1, \hat{H}_2, \hat{H}_3 \in \mathbb{R}[x]$.
- Their coefficients belong to the quadratic real K_0^r.
  Recognize $x \in \mathbb{R}$ as short vector in:

$$\begin{pmatrix} 1 & \kappa_1 & 0 & 0 \\ \sqrt{D'} & 0 & \kappa_2 & 0 \\ x & 0 & 0 & \kappa_3 \end{pmatrix}$$

Success criterion: smooth denominators.

- Denominators can be predicted to some extent (not done).
- As long as reconstruction fails, keep on lifting $\theta^2_{0,1,2,3}(\tau)$.
  At most we're lifting twice higher than what we would need if we had sharp bounds on denominators.

Plan

Introduction

CM in genus 1

Genus 2 prerequisites

Algorithm

Computer experiments

Implementation

- Number theoretic computations: $\mathfrak{C}(K)$, (reduced) period matrices
  - Pari/GP
  - negligible effort
- Evaluation of theta and invariants
  - C
  - Libraries: GMP, MPFR, MPC
  - MPI for parallelisation
- Polynomial operations
  - MPFRCX
  - MPI for (partial) parallelisation

Software

`http://cmh.gforge.inria.fr/`

- GPLv3+
- `./configure --with-gmp=... ... --enable-mpi`
  `make install`
- Period matrices: `cmh-classpol.sh -p 35 65`
- Class polynomials: `cmh-classpol.sh -f 35 65`
- Curve for checking: `cmh-classpol.sh -c 35 65`
- Using MPI:
  `mpirun -n 4 cm2-mpi -i 965_35_65.in -o H123.pol`

Two baby examples

$X^4 + 144X^2 + 3500$
$\mathfrak{C} = \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/30\mathbb{Z}$

preparation	0.2
base, 2 000 bits	0.6
lift, 3 984 bits	0.8
lift, 7 944 bits	2.1
reconstruction	0.1
lift, 15 846 bits	6.2
$H_1, \hat{H}_2, \hat{H}_3 \in \mathbb{C}[X]$	0.1
$H_1, \hat{H}_2, \hat{H}_3 \in K_0^r[X]$	3×0.3
check	0.8
Total (incl. I/O)	12.4

$X^4 + 134X^2 + 712$
$\mathfrak{C} = \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/60\mathbb{Z}$

preparation	0.3
base, 2 000 bits	1.1
lift, 3 988 bits	1.6
lift, 7 958 bits	4.4
reconstruction	0.1
lift, 15 886 bits	13.1
reconstruction	0.2
lift, 31 744 bits	38.7
$H_1, \hat{H}_2, \hat{H}_3 \in \mathbb{C}[X]$	0.6
$H_1, \hat{H}_2, \hat{H}_3 \in K_0^r[X]$	$1.8 + 2\times1.4$
check	0.7
Total (incl. I/O)	69.2

Timings in seconds for two examples (Intel i5-2500, 3.3GHz).

One jumbo experiment

How far can we go ?

- $K = \mathbb{Q}[X]/(X^4 + 1357X^2 + 3299)$, $K_0 = \mathbb{Q}(\sqrt{1828253})$.
- $\mathfrak{C} \simeq \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/5004\mathbb{Z}$; $\mathfrak{C} = 20\,016$.

Computation breakdown:

- 10 008 symbolic period matrices: minutes.
- Lift up to 2 000 000 bits: hours (640 cores).
- Lift up to 8 000 000 bits: 3 days (160 cores, more RAM).
- Computing polynomials: 3 days (24 cores).
- Recognizing coefficients: 2 days (480 cores).
- Disk size for class polynomial triple: 90 GB.

$\mathrm{lc}(H_1)$ has 8 884 distinct prime factors, largest is 1 506 803 839.

A curve

$$\pi = 2587584949432298\alpha^3 + 598749326588980\alpha^2 + 34891101632059958 72\alpha - 17626367557116479015,$$

$$p^2 = \mathrm{Norm}(\pi) = \left(2^{128} + 5399685\right)^2,$$

$$y^2 = 329105434147215182703081697774190891717x^5 + 219357712933218699650940059644263138156x^4 + 94773520721686083389380651745963315116x^3 + 136122807144468181040303471221092158 19x^2 + 224591198286067822213326173663420732292x + 62350272396394045327709463978232206155,$$

$$\chi = t^4 - s_1 t^3 + s_2 t^2 - ps_1 t + p^2,\ (s_1 = -72130475900828407780, s_2 = 1980610692179048658315492237655054733182),$$

$$\#J = (p^2+1) - (p+1)s_1 + s_2 = 2^4 \cdot 3433 \cdot p_{73}.$$

Conclusion

- Complex analytic CM construction is effective in genus 2, not just for ridiculously small examples;
- We don't meet the sky-large class number requirements though;
- Computing θ-constants is fast.
  Never say it's a bottleneck. There's available software !

Further improvements:

- Higher genus ?
- Prove the conjectures ? (note: there are trivial workarounds anyway).
- Improve on our recognition step, which is too slow.
- Compute $\theta(\tau, z)$, not jsut $\theta(\tau, 0)$.
- Improve the CRT method to make it as effective.