Computing the endomorphism ring of an elliptic curve over a number field

John Cremona

University of Warwick

—

with Andrew Sutherland, MIT

LuCaNT – LMFDB, Computation, and Number Theory
ICERM, 14 July 2023

Overview: identifying $\mathrm{End}(E)$ by recognising HCPs

1. From $\mathrm{End}(E)$ to CM to HCPs
2. CM facts
3. Properties of HCPs
4. The algorithm
5. Results

See `http://arxiv.org/abs/2301.11169` for our preprint, and `https://github.com/AndrewVSutherland/EndECNF` for implementations in Pari/GP, SageMath and Magma.

Our algorithm is now in SageMath (version 10.0) and will be in Pari/GP (version 2.16).

Introduction

For many questions concerning elliptic curves E over number fields K, it is important to know whether or not the curve has Complex Multiplication (CM).

By definition, this means that $\mathrm{End}(E)$ is an order in an imaginary quadratic field; otherwise $\mathrm{End}(E) \cong \mathbb{Z}$.

The question we are asking is in two parts:

Given an elliptic curve E defined over a number field K,
(1) does E have CM?; if not, then $\mathrm{End}(E) \cong \mathbb{Z}$*;*
(2) if so, what is the CM discriminant D such that $\mathrm{End}(E) \cong \mathcal{O}_D$*?*

Endomorphism rings and orders

- Recall: for each negative discriminant D (i.e. $D \equiv 0, 1 \pmod 4$) there is a unique order $\mathcal{O}_D$ of discriminant D. Hence elliptic curves with CM have a unique CM discriminant.
- By $\mathrm{End}(E)$ we always mean the ring of geometric endomorphisms, defined over the algebraic closure; the additional endomorphisms will only be defined over K when $\sqrt{D} \in K$.

j-invariants

- $\mathrm{End}(E)$ only depends on the j-invariant $j(E)$
- CM curves have *integral* j-invariants.

So we can rephrase our questions:

Given an algebraic integer j,
(1) is j a CM j-invariant ("singular modulus")?
(2) if so, what is the associated discriminant D?

These questions are independent of the field K containing j.

Hilbert Class Polynomials

- For each negative discriminant D, the number of CM j-invariants with discriminant D is $h(D)$, the class number of the order $\mathcal{O}_D$;
- they are all Galois conjugate, being the roots of the *Hilbert Class Polynomial* (HCP) H_D, which is monic and irreducible with integer coefficients.

So we can rephrase our questions again, in terms of the minimal polynomial H of the algebraic integer j:

Given a monic irreducible polynomial H in $\mathbb{Z}[X]$,
(1) is H an HCP?
(2) if so, for which D is $H = H_D$*?*

The exhaustive method

- For each class number h there are only finitely many discriminants D with $h(D) = h$, so finitely many HCPs of degree h.
- If we know them all we can simply do a table lookup.
- E.g. for $h = 1$ we have 13: $D = -3, -4, -7, -8, -11, -12, \ldots, -163$ and $H_D = X, X - 1728, X + 3375, X - 8000, X + 32768, X + 54000, X + 262537412640768000$.
- For $h \leq 100$ there are 66758 discriminants and over 2GB of HCPs!
- There are $29, 25, 84, 29, 101, 38, 208, 55, 123$ discriminants for $h = 2, \ldots, 10$. So this is only useful for very small h.

CM facts

- Let D be a negative discriminant and $K = \mathbb{Q}(\sqrt{D})$. After embedding $\mathcal{O}_D \hookrightarrow \mathbb{C}$, each invertible ideal $\mathfrak{a} \subset \mathcal{O}_D$ becomes a lattice in $\mathbb{C}$ and hence has a j-invariant $j(\mathfrak{a})$ which only depends on the ideal class $[\mathfrak{a}]$.
- For each $\mathfrak{a}$, $L = K(j([\mathfrak{a}]))$ is the *ring class field* for $\mathcal{O}_D$; it is an Abelian Galois extension of K of degree $h(D)$, with $\mathrm{Gal}(L/K) \cong \mathcal{C}_D$.
- The action of $\mathcal{C}_D$ is given by $[\mathfrak{b}] : j([\mathfrak{a}]) \mapsto j([\mathfrak{a}\mathfrak{b}^{-1}])$.
- L is also Galois over $\mathbb{Q}$ with $\mathrm{Gal}(L/\mathbb{Q}) \cong \mathcal{C}_D \rtimes C_2$, where C_2 acts on $\mathcal{C}_D$ by inversion.
- $F = \mathbb{Q}(j([\mathfrak{a}]))$ is only Galois when $\mathcal{C}_D$ has exponent 2.

The abelian case

- When C_D is an elementary abelian 2-group, $F = \mathbb{Q}(j([\mathfrak{a}]))$ is itself Galois and $L = F(\sqrt{D})$ is abelian over $\mathbb{Q}$.
- For example, when $h(D) = 1$, $F = \mathbb{Q}$ and $L = K$ or when $h(D) = 2$.
- This only occurs for finitely many discriminants!
  There are 101 of these, listed in John Voight's PhD thesis (UC Berkeley, 2005), with $h(D) \leq 16$; the largest is $D = -7392$ with $h(D) = 16$.

I may tacitly exclude this case in what follows.

Action of Galois and complex conjugation

- The $h(D)$ elements of $\mathrm{Gal}(L/K)$ act via $j([\mathfrak{a}]) \mapsto j([\mathfrak{a}\mathfrak{b}^{-1}])$ for $[\mathfrak{b}] \in \mathcal{C}_D$.
- The other $h(D)$ elements of $\mathrm{Gal}(L/\mathbb{Q})$ have order 2, and act via $j([\mathfrak{a}]) \mapsto j([\mathfrak{a}^{-1}\mathfrak{b}])$ for $[\mathfrak{b}] \in \mathcal{C}_D$.
- As a special case, complex conjugation acts by $j([\mathfrak{a}]) \mapsto \overline{j([\mathfrak{a}])} = j([\mathfrak{a}]^{-1})$.
- Hence the number of *real* conjugates is $h_2 := \#\mathcal{C}_D[2]$.
- There is always at least one real conjugate $j([\mathcal{O}_D])$, and the conjugates are all real if and only if D is one of the abelian discriminants.

Properties of HCPs I: factorization over $\mathbb{R}$

- By definition,

$$H_D(X) = \prod_{[\mathfrak{a}] \in \mathcal{C}_D} (X - j([\mathfrak{a}]))$$

  so that H_D is monic, and it is irreducible, of degree $h(D)$, with integer coefficients.
- The root $j([\mathfrak{a}])$ is real if and only if $[\mathfrak{a}] \in \mathcal{C}_D[2]$, so the number h_2 of real roots is a power of 2, divides h, and is 1 if and only if h is odd.
- One way to show that some $f \in \mathbb{Z}[X]$ (monic irreducible of degree h) is *not* an HCP is to count its real roots and see if it satisfies these...

Identifying D using real roots

The algorithm used by the function `CMtest` in Magma V2.27-5 is to compute the real roots to high precision, check that their number is a power of 2 [dividing the degree] and inverting the j function.

For example if D is even and $h > 1$ then the largest positive real root $r = j(\sqrt{D}/2) \geq j(\sqrt{-5}) > 1264538$ and so $D \sim -\log((r - 744)/\pi)^2$.

Similarly in the case of odd D, using the largest negative root.

This method is fine for small degree (< 1s for $h \leq 45$) but very slow and memory bound for larger degrees.

Properties of HCPs II: factorization over $\mathbb{F}_p$

The factorization pattern of $H_D \pmod{p}$ is very constrained.
Assuming that $H_D \pmod{p}$ is squarefree:

- If p *splits* in K as $(p) = \mathfrak{p}\overline{\mathfrak{p}}$ then (considering the action of $\mathrm{Frob}_{\mathfrak{p}}$) we find that $H_D \pmod{p}$ factors as a product of h/f irreducible factors of degree f, where $f \mid h$ is the order of $[\mathfrak{p}]$ in $\mathcal{C}_D$.
- If p is *inert* in K then $H_D \pmod{p}$ factors *either* as a product of $h/2$ irreducible quadratics, *or* as h_2 linear and $(h - h_2)/2$ quadratics, where $h_2 = \#\mathcal{C}_D[2]$.
  The cases depend on whether $[\mathfrak{a}]$ is a square or not, where the action of $\mathrm{Frob}_{\mathfrak{p}}$ is given by $[\mathfrak{a}]$.

Application to HCP detection

The special factorization patterns of $H_D \pmod{p}$ provide ways of easily showing that $H \in \mathbb{Z}[X]$, monic irreducible of degree h, is *not* an HCP.

For example, if h is odd, then $h_2 = 1$, and the number of roots modulo p must be 0, 1 or h.

When h is even, the number of roots must be 0, h_2 or h, for some $h_2 > 1$, a power of 2 dividing h (which must be *the same* for all p which do not have 0 or h roots modulo p).

But to show that a polynomial *is* an HCP H_D, and to recover D, we need something more.

Using ordinary primes to recover D

As before, let p be a prime such that $H_D \pmod p$ is squarefree; these are unramified in K, so are split or inert.

Let E/L be an elliptic curve with j-invariant $j([\mathfrak{a}])$ for some $[\mathfrak{a}] \in \mathcal{C}_D$, so that E has CM by $\mathcal{O}_D$, and has good reduction modulo primes $\mathfrak{p} \mid p$.

The reduction $E_\mathfrak{p}$ is ordinary if and only if p splits in K; otherwise, for inert primes, it is supersingular.

Key fact: in the ordinary case,

$$\mathrm{End}(E_\mathfrak{p}) \cong \mathcal{O}_D \cong \mathrm{End}(E).$$

So we can recover D by computing $\mathrm{End}(E_\mathfrak{p})$!

Using ordinary primes to recover D (contd.)

In our algorithm we find ordinary primes p which split completely in L, so we only need work over $\mathbb{F}_p$.

But if we do not yet know $K = \mathbb{Q}(\sqrt{D})$, how do we find such primes?

Answer: they are primes such that $H_D \pmod{p}$ splits completely into linear factors. The density of these is $1/(2h)$ (and is likely to be much smaller for irreducible f of degree h which are not HCPs).

Using ordinary primes to recover D (contd.)

In our algorithm we find ordinary primes p which split completely in L, so we only need work over $\mathbb{F}_p$.

But if we do not yet know $K = \mathbb{Q}(\sqrt{D})$, how do we find such primes?

Answer: they are primes such that $H_D \pmod{p}$ splits completely into linear factors. The density of these is $1/(2h)$ (and is likely to be much smaller for irreducible f of degree h which are not HCPs).

Let $E_p/\mathbb{F}_p$ be an elliptic curve with $j(E_p)$ a root of H mod p. Computing $\mathrm{End}(E_p)$ for ordinary $E_p/\mathbb{F}_p$ is a previously solved problem which can be done in polynomial time (under GRH). [Kohel (1996); Bisson (2011); Bisson and Sutherland (2011)]

In our case we can make use of the fact that the class number of $\mathrm{End}(E_p)$ is known, to simplify the algorithm.

The algorithm

Given a monic irreducible $H \in \mathbb{Z}[X]$ of degree h, return `true`,*D if $H = H_D$ for some D, otherwise return* `false`.

Set $\mathcal{D} = \{h_2 = 2^k : h_2|h,\ h_2 \equiv h \pmod 2\}$.
For increasing primes $p \geq \lceil 37h^2(\log\log(h+1)+4)^4 \rceil$:

1. Compute $H_p := H \bmod p \in \mathbb{F}_p[x]$.
2. Compute $d := \deg\gcd(H_p(x), x^p - x)$.
3. If $d = 0$ or $\gcd(H_p, H_p') \neq 1$ then proceed to the next prime p.
4. If $d < h$ and $d \notin \mathcal{D}$ then return `false`.
5. Let $E_p/\mathbb{F}_p$ be an elliptic curve with $j(E_p)$ a root of H_p.
6. If E_p is supersingular then proceed to the next prime p.
7. Compute $D := \mathrm{disc}(\mathrm{End}(E_p)) \in \mathbb{Z}$.
8. If $h(D) \neq h$ then return `false`, else compute H_D.
9. If $H = H_D$ then return `true`, D; otherwise return `false`.

Proof of correctness

- The algorithm only returns `true` and D after checking that $H = H_D$.
- It terminates when it reaches a prime p that satisfies:
  1. $F = \mathbb{Q}[X]/(H)$ has a degree 1 prime $\mathfrak{p} \mid p$;
  2. every E/F with $j(E)$ a root of H has good ordinary reduction at every $\mathfrak{p} \mid p$.

  A positive density of primes satisfy these.
- If $H = H_D$ then at step 7, H splits completely mod p and E is ordinary, so the D computed in step 7 is correct.

For details, see our paper.

Comments on the algorithm

- The computed starting value of p ensures that $4p > |D|$ when $H = H_D$ (under GRH), which is necessary for H_D to split completely mod p.
- When H is an HCP we expect (under GRH) to find a splitting prime in about $2h$ trials.
  The algorithm's correctness does not depend on these.
- For better practical performance and for the asymptotic complexity we should not reduce H modulo primes one by one, but use a product tree, first reducing H modulo a product of primes (in the range) which is large enough.
- In computing $\mathtt{End}(E)$ we may assume that its class number is h.

Complexity of the algorithm

Theorem (Heuristic)

Under reasonable heuristic assumptions (including GRH), the Algorithm can be implemented as a Las Vegas algorithm that runs in

$$h^2(\log h)^{3+o(1)} + h(h+|H|)\log(h+|H|)^{2+o(1)} = h(h+|H|)^{1+o(1)}$$

expected time (which is quasilinear in $|H|$), using at most

$$h(h+|H|)\log(h+|H|)^{1+o(1)}$$

space.

Here $|H|$ is the logarithm of the maximum absolute value of the coefficients of H.

An alternative algorithm

We have a second algorithm which admits a deterministic implementation that runs in

$$(h^2|H|)^{1+o(1)}$$

time using

$$(h|H|)^{1+o(1)}$$

space.

The input is again $H \in \mathbb{Z}[X]$, monic irreducible of degree h.

But

- It only returns `true`, not the value of D, when $H = H_D$; and
- its correctness is conditional on GRH!

See the paper for the other algorithm. Its implementation is simpler, but it is slower in practice, and gives less information.

Computational results

We have implemented the algorithm in PARI/GP, SAGEMATH, and MAGMA. Our code does not implement all the tweaks mentioned, but runs successfully on inputs of degree up to 1000, never taking more than 4.5m (and only up to 30s for $h \leq 500$).

In our timings we separate off the time to compute H_D, and test both H_D and $H_D + 1$ (which is not an HCP!) for many D up to about 28 million, with h up to 1000.

Computational results

			Magma			Pari/GP			SageMath		
h	$\|H\|$	$\|D\|$	t_{HCP}	t_{CM}	t_{noCM}	t_{HCP}	t_{CM}	t_{noCM}	t_{HCP}	t_{CM}	t_{noCM}
5	120	571	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00
10	294	2299	0.00	0.01	0.00	0.02	0.01	0.00	0.00	0.02	0.00
20	843	9124	0.00	0.01	0.00	0.02	0.02	0.00	0.00	0.02	0.00
30	1198	21592	0.00	0.02	0.00	0.05	0.06	0.00	0.00	0.07	0.00
40	1739	34180	0.01	0.02	0.00	0.05	0.06	0.00	0.00	0.02	0.00
50	2161	64203	0.02	0.02	0.00	0.09	0.09	0.00	0.00	0.04	0.00
100	4197	249451	0.15	0.23	0.00	0.29	0.37	0.00	0.03	0.30	0.00
200	9520	910539	1.32	1.86	0.00	0.77	1.24	0.00	0.19	1.21	0.00
300	14621	2127259	4.64	6.20	0.01	2.06	3.23	0.00	0.60	3.28	0.02
400	21707	3460787	12.90	16.99	0.00	5.91	8.45	0.00	1.50	5.66	0.00
500	28965	6423467	26.22	31.21	0.01	9.99	12.35	0.00	3.03	8.52	0.00
600	33802	7885067	45.68	49.61	0.01	14.97	17.57	0.01	4.73	10.93	0.02
700	39857	12955579	72.36	76.45	0.01	14.50	17.28	0.01	7.22	10.72	0.01
800	44169	13330819	106.77	122.06	0.02	20.26	28.64	0.01	9.73	27.43	0.02
900	47449	19028875	141.95	145.31	0.01	28.00	30.76	0.01	12.59	16.73	0.01
1000	56827	23519868	215.96	267.94	0.03	49.48	83.42	0.02	18.81	81.98	0.03