Computing Zeta functions of Picard curves

Francesc Fité (MIT)

Joint with S. Asif (MIT), D. Pentland (MIT), and A.V. Sutherland (MIT)

Based on the article: https://arxiv.org/abs/2010.07247

4/2/2021

Let C be a smooth projective curve defined over $\mathbb{Q}$ of genus g.

Throughout the talk p will be a prime of good reduction for C.

The zeta function of C at p is

$$Z_p(C,T) = \exp\left(\sum_{n=1}^{\infty} \#C(\mathbb{F}_{p^n})\frac{T^n}{n}\right) \in \mathbb{Q}[[T]]\,.$$

It is shown to be a rational function

$$Z_p(C,T) = \frac{L_p(C,T)}{(1-T)(1-pT)} \in \mathbb{Q}(T)\,,$$

where $L_p(C,T) \in \mathbb{Z}[T]$ has degree $2g$.

Computing $L_p(C,T)$ amounts to computing:

$$\#C(\mathbb{F}_p), \quad \#C(\mathbb{F}_{p^2}), \quad \dots, \quad \#C(\mathbb{F}_{p^g})\,.$$

A problem and its theoretical answers

Problem

- Compute $L_p(C, T)$ for large p.
- Compute $\{L_p(C, T)\}_{p \leq N}$ for some large bound N.

Theorem (Schoof–Pila; 1985-1990)

There is an algorithm[1] that computes $L_p(C, T)$ using $\log(p)^{e(g)+o(1)}$ bit operations.

(Adleman–Huang: $e(g)$ has a polynomial growth in g).

Theorem (Harvey; 2015)

There is an algorithm[2] that computes $\{L_p(C, T)\}_{p \leq N}$ using $N \log(N)^{3+o(1)}$ bit operations.

(Hence “polynomial on average": $\log(N)^{4+o(1)}$ operations per prime).

[1] In this talk all algorithms are deterministic.

[2] In fact, it is much more general: It applies to any scheme of finite type over $\mathbb{Z}$!

The computational challenge

We also seek for practical algorithms, i.e:

“able to produce answers when run by real hardware and p, N are, say, $\approx 2^{30}$ in a reasonable amount of time, say, less than a week."

- Practical algorithms are at disposal when $g \leq 2$ (Sutherland).
- Substantial progress has been made for superelliptic curves:

$$C : y^m = f(x)\,, \qquad \text{where } f \in \mathbb{Q}[x] \text{ is separable.}$$

Indeed:

Arul–Best–Costa–Magner–Triantafillou (2019)
compute [3] $L_p(C, T)$ in time $p^{1/2+o(1)}$.

Sutherland (2020)
computes $\{L_p(C, T)\}_{p\leq N}$ *modulo* p in time $N \log(N)^{3+o(1)}$.

---

[3] In the particular case of a Picard curve an algorithm of the same complexity had been implemented by Bauer, Teske, and Weng (2004).

Goal

A Picard curve defined over $\mathbb{Q}$ is a curve C given by an affine model

$$y^3 = f(x) \qquad \text{where } f(x) \in \mathbb{Q}[x] \text{ is separable of degree 4.}$$

WLOG we will assume $f(x) = x^4 + f_2x^2 + f_1x + f_0$ with $f_i \in \mathbb{Z}$.

There is a ring homomorphism

$$\mathbb{Z}[\zeta_3] \hookrightarrow \mathrm{End}(\mathrm{Jac}(C)_{\overline{\mathbb{Q}}})\,.$$

We will say that C is generic if $\mathrm{End}(\mathrm{Jac}(C)_{\overline{\mathbb{Q}}}) \simeq \mathbb{Z}[\zeta_3]$.

Goal

Describe a practical algorithm that, given a generic Picard curve, computes $\{L_p(C,T)\}_p$ for *almost* every $p \leq N$ in time $N\log(N)^{3+o(1)}$.

The Cartier–Manin matrix

The Cartier–Manin matrix at p is the matrix A_p of a certain operator $\mathcal{C}_p$ acting on $H^0(C_p, \Omega^1_{C_p/\mathbb{F}_p})$ (in the basis dx/y^2, xdx/y^2, dx/y).

It has the fundamental property:

$$L_p(C,T) \equiv \det(1 - TA_p) \pmod{p}.$$

Key point

For $p \equiv 1 \pmod 3$, A_p is of the form $\begin{pmatrix} * & * & 0 \\ * & * & 0 \\ 0 & 0 & * \end{pmatrix}$.

So A_p not only gives $L_p(C,T) \pmod{p}$, but a canonical factorization

$$L_p(C,T) \equiv g_p^1(T) \cdot g_p^2(T) \pmod{p}.$$

Sutherland (2020) computes $\{A_p\}_{p \le N}$ in time $N \log(N)^{3+o(1)}$.

First main result

Set the notations:

$\mathfrak{I}(C)$ for the set of good primes that are inert in $\mathbb{Q}(\zeta_3)$.

$\mathfrak{S}(C)$ for the set of good primes that split in $\mathbb{Q}(\zeta_3)$.

$\mathfrak{S}^{\mathrm{ord}}(C)$ for the set of good *ordinary* primes that split in $\mathbb{Q}(\zeta_3)$.

(p is called ordinary if p does not divide the central coefficient of $L_p(C,T)$).

Theorem 1

Let C be a Picard curve over $\mathbb{Q}$. Then:

For every p in $\mathfrak{S}^{\mathrm{ord}}(C)$ at least 53, A_p uniquely determines $L_p(C,T)$.

If C is generic, then $\mathfrak{S}^{\mathrm{nord}}(C) := \mathfrak{S}(C) - \mathfrak{S}^{\mathrm{ord}}(C)$ has zero density.

Second main result

Attached to $y^3 = f(x)$, with $f(x) = x^4 + f_2 x^2 + f_1 + f_0$, define

$$\begin{aligned}\psi_f(x) := x^9 &+ 24 f_2 x^7 - 168 f_1 x^6 + (1080 f_0 - 78 f_2^2) x^5 + 336 f_1 f_2 x^4 \\ &+ (1728 f_0 f_2 - 636 f_1^2 + 80 f_2^3) x^3 + (-864 f_0 f_1 - 168 f_1 f_2^2) x^2 \\ &+ (-432 f_0^2 + 216 f_0 f_2^2 - 120 f_1^2 f_2 - 27 f_2^4) x - 8 f_1^3 \,.\end{aligned}$$

Key property

The splitting field of $\psi_f(x^3/2)$ is the 2-torsion field of $\mathrm{Jac}(C)$.

Theorem 2

Let C be a Picard curve over $\mathbb{Q}$. For every p in $\mathfrak{I}(C)$, the knowledge of:

- f having or not a root modulo p;
- $\psi_f(x)$ being irreducible or not modulo p;
- the matrix A_p;

uniquely determine $L_p(C, T)$.

A practical algorithm

Corollary

The constructive proofs of the theorems yield a practical algorithm that computes $\{L_p(C,T)\}_{p\leq N, p\notin \mathfrak{S}^{\mathrm{nord}}(C)}$ in time $N\log(N)^{3+o(1)}$.

Remarks

One can speed up the ABCMT algorithm by a factor of 8 (for $p \notin \mathfrak{S}^{\mathrm{nord}}$).

In practice for $p \in \mathfrak{S}^{\mathrm{nord}}(C)$ one can use the ABCMT algorithm.

For C generic, it is conceivable that $\mathfrak{S}^{\mathrm{nord}}(C)$ is so small that this does not affect the complexity of the algorithm

(we were unable to prove that).

N	2^{20}		2^{24}		2^{28}	
Algorithm	[ABCMT]	[S]+[AFPS]	[ABCMT]	[S]+[AFPS]	[ABCMT]	[S]+[AFPS]
$y^3 = x^4 + x + 1$	215.1	0.57+**0.12**	1152.7	1.37+**0.13**	5051.4	4.63+**0.14**
$y^3 = x^4 + 3x^2 + 2x + 1$	213.5	0.59+**0.12**	1152.9	1.41+**0.13**	5053.9	4.74+**0.14**

Running time to compute $L_p(C,T)$ in ms for $p \approx N$ for the various algorithms.
The timings were taken on a 3.40GHz Intel(R) Xeon(R) E5-2687W CPU.

Sketch of proof of Theorem 1

Recall the factorization

$$L_p(C,T) \equiv g_p^1(T)\cdot g_p^2(T) \pmod{p}\,.$$

The injection $\mathbb{Z}[\zeta_3] \hookrightarrow \mathrm{End}(\mathrm{Jac}(C)_{\overline{\mathbb{Q}}})$ induces a *compatible* factorization

$$L_p(C,T) = L_p^1(T)\cdot L_p^2(T) \qquad \text{over } \mathbb{Z}[\zeta_3][T]\,.$$

It suffices to determine $L_p^1(T) = 1 - aT + bT^2 - cT^3$.

- Since $|a| \leq 3\sqrt{p}$, a is determined by $g_p^1(T)$ (for $p \geq 53$). Note that $g_p^1(T)$ only provides $b \pmod{\pi}$ and $c \pmod{\pi}$.
- One shows $pb = c\bar{a}$. So it suffices to determine c.
- One shows $c = \zeta p \pi$, with $\zeta^6 = 1$. So it suffices to determine ζ.
- Note that

$$\zeta = \frac{b}{\bar{a}\bar{\pi}} \equiv \frac{b}{\bar{a}\bar{\pi}} \pmod{\pi}$$

This makes sense only if p ordinary!

Sketch of proof of Theorem 2

Exercise: Show that if p is in $\mathfrak{I}(C)$, then

$$L_p(C,T) = (1+pT^2)(1-tT^2+p^2T^4) \qquad \text{where } |t| \le 2p.$$

Hint: Use that $\#C(\mathbb{F}_p) = p+1$ and $\#C(\mathbb{F}_{p^3}) = p^3+1$.

The theorem follows from the facts that:

- A_p determines $L_p(C,T)$ modulo p.
- the splitting behavior of f modulo p determines $L_p(C,T) \pmod 3$. (f is related to the 3-torison field of $\mathrm{Jac}(C)$).
- the splitting behavior of ψ_f modulo p determines $L_p(C,T) \pmod 2$. (ψ_f is related to the 2-torsion field of $\mathrm{Jac}(C)$).