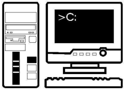
Efficient computation of the Hasse-Weil zeta function

Problem I:

Develop an efficient algorithm that determines the number of zeroes of any given polynomial $f(x_1, \ldots, x_n)$ with coefficients in a finite field $\mathbb{F}_q$.

$f \longrightarrow$ 

$\longrightarrow \#\left\{ (a_i) \in (\mathbb{F}_q)^n \,\middle|\, f(a_1, \ldots, a_n) = 0 \right\}$

Naively checking for every $(a_1, \ldots, a_n) \in (\mathbb{F}_q)^n$ whether $f(a_1, \ldots, a_n) = 0$ is not efficient!

$\rightsquigarrow$ takes at least q^n steps

Write $N_k := \# \left\{ (a_i) \in \left(\mathbb{F}_{q^k}\right)^n \,\middle|\, f(a_1, \ldots, a_n) = 0 \right\}$ and define the zeta function

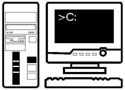
$$Z_f(T) = \exp\left(\sum_{k=1}^{\infty} N_k \frac{T^k}{k} \right) \qquad \in \mathbb{Q}[[T]]$$

which turns out to be a rational function (Dwork) that can be algorithmically determined (Bombieri).

Problem II:

Develop an efficient algorithm that determines the zeta function of any given polynomial $f(x_1, \ldots, x_n)$ with coefficients in a finite field $\mathbb{F}_q$.

$f \longrightarrow$



$\longrightarrow Z_f(T)$

In general, both problems are far from being solved (in every reasonable sense of the word efficient).

- If $n = 1 \rightsquigarrow$ solved
  - polynomial factorization over $\mathbb{F}_q$ (Berlekamp)
  - count the number of linear factors
- If $n = 2 \rightsquigarrow$ good progress
  - reduce to irreducible case via

$$\#Z(fg) = \#Z(f) + \#Z(g) - \#(Z(f) \cap Z(g))$$

    using polynomial factorization (Lenstra, Wan)
  - irreducible-but-not-absolutely-irreducible case is easy (point enumeration) $\rightsquigarrow$ reduce to absolutely irreducible case
  - use geometric and arithmetic properties of the curve $Z(f)$
- If $n > 2 \rightsquigarrow$ some generalizations, mostly only in theory

This talk: n=2, i.e.

- $f \in \mathbb{F}_q[x, y]$ is absolutely irreducible.
- Thus it defines a curve $\widetilde{C}$ in $\mathbb{A}^2_{\mathbb{F}_q}$.
- Generalized zeta function: for any quasi-projective curve $C/\mathbb{F}_q$ we define

$$Z_C(T) = \exp\left(\sum_{k=1}^{\infty} \#C(\mathbb{F}_{q^k})\frac{T^k}{k}\right).$$

  Thus $Z_{\widetilde{C}}(T) = Z_f(T)$.
- Note that $Z_C(T)$ only depends on the isomorphism class $[C]$.

Theorem (Weil):

Let C be a smooth projective curve of genus g. Then we can write

$$Z_C(T) = \frac{P(T)}{(1-T)(1-qT)}$$

for a degree $2g$ polynomial $P(T) \in \mathbb{Z}[T]$. Moreover

- $P(T)$ factors as
$$\prod_{i=1}^{2g}(1-\alpha_i T)$$
for algebraic integers $\alpha_i \in \mathbb{C}$.
- For $i = 1, \ldots, 2g$ we have $|\alpha_i| = \sqrt{q}$ (Riemann hypothesis).
- For a suitable choice of indices, we have $\alpha_i \alpha_{2g-i} = q$ for $i = 1, \ldots, g$.
- $\#C(\mathbb{F}_{q^k}) = q^k + 1 - \sum_{i=1}^{2g} \alpha_i^k$.

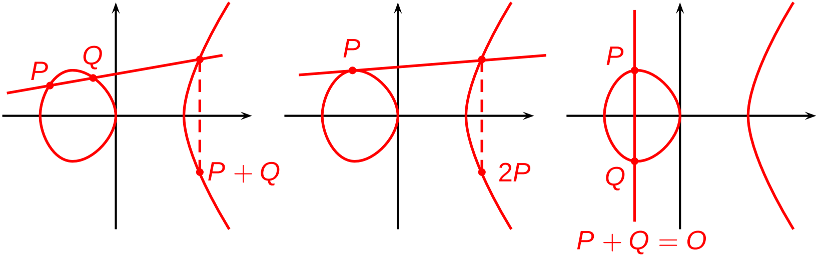
- If $\widetilde{C}$ is any quasi-projective curve, and C is its complete nonsingular model, then

$$Z_{\widetilde{C}}(T) = Z_C(T)(1 - T^{\kappa_1})(1 - T^{\kappa_2})(1 - T^{\kappa_t})$$

where $\kappa_1 + \cdots + \kappa_t$ is the number of 'points missing'.

- The κ_i depend on the degrees of the field extensions over which these points are defined.

In the case of a smooth projective $g = 1$ curve, we have the group law



For higher genus, a smooth projective curve $C/\mathbb{F}_q$ can be embedded in a 'smallest' abelian variety $\mathrm{Jac}_{\mathbb{F}_q}(C)$ (it has dimension g).

Theorem (Tate):

$\#\mathrm{Jac}_{\mathbb{F}_q}(C) = P(1)$

Most famous application and research motivation:

$$P \in E(\mathbb{F}_q)$$

$$a \in \mathbb{N} \quad \xrightarrow{aP}$$

$$\xleftarrow{bP} \quad b \in \mathbb{N}$$

$$\boxed{(ab)P = a(bP)} \qquad \boxed{(ab)P = b(aP)}$$

- Security is believed to depend on the hardness of the discrete log problem: given P and nP, find n ...
- ... which is easy if $\#E(\mathbb{F}_q)$ contains no big prime factors.

First method. Computing in $\mathrm{Jac}_{\mathbb{F}_q}(C)$.

Idea:

- Arithmetic in $\mathrm{Jac}_{\mathbb{F}_q}(C)$ can be performed efficiently (Hess, Khuri-Makdisi).
- Use this to compute the order of a generic point.
- Try to recover $Z_C(T)$ from $P(1) = \#\mathrm{Jac}_{\mathbb{F}_q}(C)$ ...
- ...and some additional info if $g > 1$ (becomes hard when g gets big).
- Example: in genus 2, q odd, every ordinary curve C has a quadratic twist C^t. If

$$Z_C(T) = \frac{P(T)}{(1-T)(1-qT)}$$

then

$$Z_{C^t}(T) = \frac{P(-T)}{(1-T)(1-qT)}$$

$\rightsquigarrow$ recover $Z_C(T)$ from $P(1)$ and $P(-1)$.

First method. Computing in $\mathrm{Jac}_{\mathbb{F}_q}(C)$.

Shanks' method to compute $N = \#\mathrm{Jac}_{\mathbb{F}_q}(C)$ (case $g = 1$).

- By Weil's theorem: $q + 1 - 2\sqrt{q} \leq N \leq q + 1 + 2\sqrt{q}$.
- Choose a random point $P \in C(\mathbb{F}_q) = \mathrm{Jac}_{\mathbb{F}_q}(C)$.
- Baby steps: make a list of the first $s \approx \sqrt[4]{q}$ multiples

$$0, \pm P, \pm 2P, \pm 3P, \ldots, \pm sP.$$

- Giant steps: compute $Q = (2s + 1)P$ and $R = (q + 1)P$ and for $t = \lceil 2\sqrt{q}/(2s + 1)\rceil \approx \sqrt[4]{q}$, produce the list

$$R, R \pm Q, R \pm 2Q, \ldots, R \pm tQ.$$

- Find match

$$R + iQ = jP.$$

- Then $mP = (q + 1 + (2s + 1)i - j)P = 0$. If the match is unique, then $\#C(\mathbb{F}_q) = m$. If not, try another P.
- Running time is $\widetilde{O}(\sqrt[4]{q})$. For $g \to \infty$, the advantage poured out of the Weil bound becomes smaller: $\widetilde{O}(q^{(2g-1)/4})$.

First method. Computing in $\mathrm{Jac}_{\mathbb{F}_q}(C)$.

State of the art: thanks to improvements by Mestre, Kedlaya, Sutherland, generic group methods make it feasible to compute $Z_C(T)$ for (roughly)

- $q < 10^{40}$ if $g = 1$, easily outperforms naive counting as soon as $q > 10^3$
- $q < 10^{13}$ if $g = 2$
- $q < 10^8$ if $g = 3$

If one is only interested in $\#\mathrm{Jac}_{\mathbb{F}_q}(C)$, then also higher genera can be dealt with, over moderately sized finite fields...

Second method. Computing in the Tate module.

Theorem (Tate):

For any prime ℓ different from the field characteristic p, and any $k \in \mathbb{N}$ we have that

$$\mathrm{Jac}_{\overline{\mathbb{F}}_q}(C)[\ell^k] \cong \left(\frac{\mathbb{Z}}{\ell^k \mathbb{Z}}\right)^{2g}.$$

Define

$$T_\ell(C) = \lim_{\overleftarrow{k}} \mathrm{Jac}_{\overline{\mathbb{F}}_q}(C)[\ell^k] \cong \mathbb{Z}_\ell^{2g}.$$

Let $\chi(T)$ be the characteristic polynomial of Frobenius acting on $T_\ell(C)$. Then

- $\chi(T) \in \mathbb{Z}[T]$ and does not depend on ℓ
- $$Z_C(T) = \frac{T^{2g}\chi(1/T)}{(1-T)(1-qT)}.$$

Second method. Computing in the Tate module.

Idea (Schoof):

- Compute
$$\chi(T) \bmod \ell$$
as the characteristic polynomial of Frobenius acting on $\#\mathrm{Jac}_{\overline{\mathbb{F}}_q}[\ell]$ for various primes ℓ.
- Use the Chinese Remainder Theorem to recover $\chi(T)$ mod $\prod \ell$.
- If $\prod \ell$ is big enough, Weil's theorem allows us to recover $\chi(T)$.

Second method. Computing in the Tate module.

In practice for elliptic curves $E : y^2 = x^3 + Ax + B$.

- The characteristic polynomial of Frobenius is of the form

$$T^2 - tT + q,$$

  and we need to recover t. By Weil's bound, $|t| \leq 2\sqrt{q}$.
- Caley-Hamilton: Frobenius map φ should satisfy its own characteristic polynomial

$$\varphi^2 - t\varphi + q = 0.$$

- There exist polynomials $\Psi_\ell \in \mathbb{F}_q[x]$ that vanish precisely at the ℓ-torsion points of E (example: $\Psi_2 = x$).
- For small ℓ, check for which $t' = t \bmod \ell$ the relation

$$(x^{q^2}, y^{q^2}) - t'(x^q, y^q) + (q \bmod \ell)(x, y)$$

  holds in $\mathbb{F}_q[x, y]/(\Psi_\ell, y^2 - x^3 - Ax - B)$.
- If $\prod \ell > 4\sqrt{q}$, use CRT to recover t.

Second method. Computing in the Tate module.

- Using smart speed-ups by Atkin and Elkies, Schoof's algorithm has become very efficient for elliptic curves ($q \approx 10^{60}$ in a couple of seconds).
- Seems hopeless to generalize this to high genera, because of the need of explicit formulas for $\#\mathrm{Jac}_{\mathbb{F}_q}(C)$.
- Small advances in genus 2 by Gaudry and Schost ($q \approx 10^{24}$ in about a week).

Third method. p-Adic cohomology.

First step: lift the curve to characteristic 0.

- Let $\overline{C}(x,y) \in \mathbb{F}_q[x,y]$ define a smooth curve in $\mathbb{A}^2_{\mathbb{F}_q}$, and write
$$\overline{A} = \frac{\mathbb{F}_q[x,y]}{(\overline{C}(x,y))}$$
for its coordinate ring.
- Write $q = p^n$ where p is the field characteristic.
- Let $\mathbb{Q}_q$ be the unramified degree n extension of $\mathbb{Q}_p$.
- Let $\mathbb{Z}_q$ be its ring of integers. This is a complete DVR with local parameter p and residue field $\mathbb{F}_q$.
- Let $C(x,y) \in \mathbb{Z}_q[x,y]$ be such that it reduces to $\overline{C}(x,y)$ mod p and write
$$A = \frac{\mathbb{Z}_q[x,y]}{(C(x,y))}.$$

Third method. p-Adic cohomology.

Problem: Geometric properties of $C/\mathbb{Q}_q$ depend on the choice of the lift: different genus, different endomorphism ring, ...

- Define

$$\mathbb{Z}_q\langle x, y\rangle^\dagger = \left\{ \sum_{i,j\in\mathbb{N}} a_{ij}x^iy^j \,\middle|\, \exists \rho \in]0,1[: \frac{|a_{ij}|_p}{\rho^{i+j}} \to 0 \text{ if } i+j \to \infty \right\}.$$

- Note that $\mathbb{Z}_q\langle x, y\rangle^\dagger$ is closed under integration and that there is a natural map $\pi : \mathbb{Z}_q\langle x, y\rangle^\dagger \to \mathbb{F}_q[x, y]$.
- Define

$$A^\dagger = \frac{\mathbb{Z}_q\langle x, y\rangle^\dagger}{(C(x, y))}.$$

Theorem (Monsky, Washnitzer):

$A^\dagger$ does not depend on the choice of C, and for every morphism $\overline{\varphi} : \overline{A} \to \overline{A}$ there exists a morphism $\varphi : A^\dagger \to A^\dagger$ that lifts $\overline{\varphi}$ in the sense that $\overline{\varphi} \circ \pi = \pi \circ \varphi$.

Third method. p-Adic cohomology.

Consider the module of differentials

$$D^1(A^\dagger) = \frac{A^\dagger dx + A^\dagger dy}{\left(\frac{\partial C}{\partial x} dx + \frac{\partial C}{\partial y} dy\right)}$$

and let $d : A^\dagger \to D^1(A^\dagger)$ be the usual exterior derivation. Then define the cohomology space

$$H^1_{MW}(\overline{A}/\mathbb{Q}_q) = \frac{D^1(A^\dagger)}{d(A^\dagger)} \otimes_{\mathbb{Z}_q} \mathbb{Q}_q.$$

Every $\mathbb{Z}_q$-algebra morphism $\varphi : A^\dagger \to A^\dagger$ induces a map

$$\varphi^* : D^1(A^\dagger) \to D^1(A^\dagger) : fdx + gdy \mapsto \varphi(f)d\varphi(x) + \varphi(g)d\varphi(y)$$

which is well-defined on $H^1_{MW}(\overline{A}/\mathbb{Q}_q)$.

Third method. p-Adic cohomology.

Theorem (Monsky, Washnitzer):

Let $\overline{\mathcal{F}}_q : \overline{A} \to \overline{A} : a \mapsto a^q$ and let $\mathcal{F}_q : \overline{A}^\dagger \to \overline{A}^\dagger$ be a lift. Then

$$Z_{\overline{C}}(T) = \frac{\det\left(\mathbb{I} - q\mathcal{F}_q^{*-1}T \;\middle|\; H^1_{MW}(\overline{A}/\mathbb{Q}_q)\right)}{(1-qT)}.$$

If $\chi(T)$ is the characteristic polynomial of $\mathcal{F}_q^*$ acting on $H^1_{MW}(\overline{A}/\mathbb{Q}_q)$, then one can verify that

$$Z_{\overline{C}}(T) = \frac{\frac{1}{q^{g+R-1}}\chi(qT)}{(1-qT)}$$

where R is the number of points at infinity.

Third method. p-Adic cohomology.

Kedlaya's method:

- Compute a lift of Frobenius $\mathcal{F}_q$.
- Compute a basis of $H^1_{MW}(\overline{A}/\mathbb{Q}_q)$.
- Let $\mathcal{F}_q^*$ act on this basis.
- Re-express the result in terms of the basis, hence obtain a matrix of Frobenius.
- Compute its characteristic polynomial.
- By Weil's theorem, it suffices to do this modulo a certain p-adic precision.
- Problem: the resulting algorithms have running time $O(q)$ and are therefore slower than generic methods.
- Solution if n is big and p is small: split up $\overline{\mathcal{F}_q} = \overline{\mathcal{F}_p} \circ \cdots \circ \overline{\mathcal{F}_p}$ $\rightsquigarrow$ running time becomes typically $O(p)$.
- Hopeless if p is big.

Third method. p-Adic cohomology.

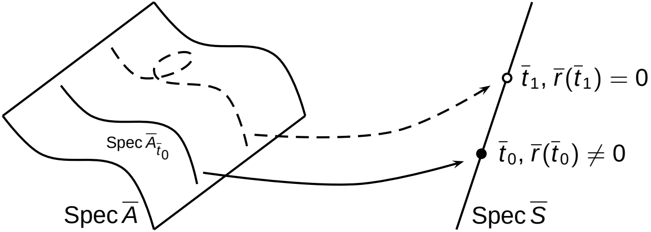
So far:

- Elliptic curves, in slightly different framework (Satoh, . . .): works extremely fast ($q \approx 10^{60}$ in a fraction of a second).
- Hyperelliptic curves (Kedlaya, Denef, Vercauteren): works fast (matter of seconds for cryptographic ranges and high genera).
- Superelliptic curves (Gaudry, Gürel): idem.
- C_{ab} curves (Denef, Vercauteren): slow performance due to different Frobenius lifting technique.
- Nondegenerate curves (curves in toric surfaces) (C., Denef, Vercauteren): idem.

Third method. *p*-Adic cohomology.

Deformation (Lauder):

- Idea: put the curve of interest into a 1-parameter family



  with $\overline{S} = \mathbb{F}_q[t, \bar{r}(t)^{-1}]$.

- Define the relative cohomology as above, now taking coefficients in a ring $S^\dagger$.
- 'Specifying' $t = t_0$ gives us the cohomology of the fibre above t_0.

Third method. *p*-Adic cohomology.

- The relative matrix of Frobenius $F(t)$ can be computed from an initial value by solving a differential equation

$$N \cdot F - \frac{d}{dt} F = q t^{q-1} \cdot F \cdot N(t^q),$$

where N is easy to compute (Gauss-Manin connection).

- Lauder's idea: take as initial value an 'easy' curve (e.g. one whose actual field of definition is a small subfield of $\mathbb{F}_q$), compute $F(t)$ and specify at the curve of interest.

Advantages:

- Avoid slow lifting of Frobenius.
- Algorithms become more memory efficient.
- Finding curves with prime order Jacobian is easier: specify at various values in the family.

So far:

- Works already well in elliptic and hyperelliptic case (Hubrechts).
- Gives satisfactory results in C_{ab} case (C., Hubrechts, Vercauteren).
- Probably as well in nondegenerate case (Tuitman, in progress)

Deformation might be the key towards dealing with arbitrary curves!

Remember: all this is over fields of small characteristic.

Some overall remarks on p-adic methods.

- The theoretical framework is very robust, results in algorithms that have polynomial running time in the genus, and applies to a wide range of varieties. In fact:

Theorem (Lauder, Wan):

If we fix the field characteristic p and the dimension n, there exists a polynomial running time algorithm (although nonpractical) to compute the zeta function of an arbitrary polynomial in n variables.

- Dependency on p is $O(p)$, but in case of hyperelliptic curves this has been reduced to $O(\sqrt{p})$ by Harvey $\rightsquigarrow$ outperforms generic methods from genus 3 on.
- Interesting question: can deformation be done in $O(\sqrt{p})$?

That's it (phew)!