Computing zeta functions of nondegenerate toric hypersurfaces via controlled reduction

Kiran S. Kedlaya

Department of Mathematics, University of California, San Diego
kedlaya@ucsd.edu
http://math.ucsd.edu/~kedlaya/slides/

Sage Days 53: Computational Number Theory, Geometry, and Physics
Mathematical Institute, University of Oxford, September 25, 2013

Joint work in preparation with David Harvey (U. New South Wales).

Supported by NSF (grant DMS-1101343), UCSD (Warschawski chair).

Contents

1. Generalities of zeta functions
2. Some examples of p-adic algorithms
3. Nondegenerate toric hypersurfaces
4. Controlled reduction in p-adic cohomology

Contents

1. Generalities of zeta functions
2. Some examples of p-adic algorithms
3. Nondegenerate toric hypersurfaces
4. Controlled reduction in p-adic cohomology

Zeta functions

For X an algebraic variety of dimension n over $\mathbb{F}_q$, its *zeta function* is

$$Z(X,T) = \exp\left(\sum_{n=1}^{\infty} \frac{T^n}{n} \#X(\mathbb{F}_{q^n})\right) \in \mathbb{Z}[\![T]\!]$$

This is a rational function of T.

Now assume X is smooth proper. Then

$$Z(X,T) = \prod_{i=0}^{2n} P_i(X,T)^{(-1)^{i+1}} = \frac{P_1(X,T)\cdots P_{2n-1}(X,T)}{P_0(X,T)\cdots P_{2n}(X,T)}$$

for some $P_i(X,T) \in 1 + T\mathbb{Z}[T]$ with $\mathbb{C}$-roots of norm $q^{-i/2}$. Moreover,

$$P_{2n-i}(X,T^{-1}) = \pm q^* T^* P_i(X,T).$$

If X lifts to characteristic 0, then $\deg P_i$ is the i-th Betti number of the lift.

The zeta function problem

Given X in an explicit form (i.e., defining equations), one would like to compute $Z(X,T)$. In principle this is a finite computation once one bounds the degree of the rational function, but in most cases the obvious computation is infeasible!

A better approach is to interpret $P_i(X,T)$ as the (reciprocal) characteristic polynomial of a linear transformation on some vector space. One such interpretation is provided by étale cohomology, but this is unsuitable for numerical computations.

By contrast, p-adic analogues of étale cohomology translate much more directly into algorithms. For instance, the first proof of rationality (by Dwork) can be made algorithmic (Lauder–Wan).

Sufficient p-adic precision

Write $q = p^a$ with p prime.

Suppose $\deg P_i$ is known for some i. Thanks to the bound on roots, for some *explicitly computable* N, we may determine P_i exactly from its coefficients modulo p^N.

That is, we may compute $P_i(X, T)$ by computing it as a p-adic polynomial to sufficient precision, or by identifying it as the reciprocal characteristic polynomial of a p-adic matrix computed to sufficient precision.

The Lefschetz hyperplane theorem

In the examples we will consider, X will be not just proper but also projective. In this case, for H a hyperplane section,

$$P_i(X,T) = P_i(H,T) \qquad (i = 0, \dots, n-1).$$

In practice, this will mean that we need only compute $P_n(X,T)$.

A precision refinement

If $P_n(X,T)$ has degree d, then it is determined by the coefficients of T^i for $i = 0, \dots, \lfloor d/2 \rfloor$. The coefficient of $T^{\lfloor d/2 \rfloor}$ has absolute value at most

$$\binom{d}{\lfloor d/2 \rfloor} q^{(n/2)\lfloor d/2 \rfloor};$$

if p^N exceeds twice this bound, then $P_n(X,T)$ is determined by its reduction modulo p^N.

However, this is not best possible! In fact, $P_n(X,T)$ is determined by its reduction modulo p^N provided that

$$p^N > \frac{2d}{i} q^{ni/2} \qquad (i = 0, \dots, \lfloor d/2 \rfloor).$$

This follows from the Newton identities and the fact that the i-th power sum of the reciprocal roots of $P_n(X,T)$ has norm at most $dq^{ni/2}$.

Zeta functions and the Hodge filtration

Suppose that X admits a smooth projective lift to characteristic 0 with Hodge numbers $h^{i,j}$. The values $h^{i,n-i}$ then imply some p-adic divisibility for coefficients of $P_n(X,T)$: the Newton polygon of $P_n(X,T)$ lies above the Hodge polygon. For example, if X is a quartic K3 surface in $\mathbb{P}^3$, then the coefficient of T^i is divisible by p^{i-1}.

If one is computing $P_n(X,T)$ as the characteristic polynomial of a matrix A over $\mathbb{Z}_q$ coming from p-adic cohomology, the Hodge numbers give lower bounds on the elementary divisors of A. This can be harnessed to reduce sufficient precision, e.g., for a quartic K3 surface over $\mathbb{F}_p$, from p^{11} to p^2 (say for $p > 17$).

Contents

1. Generalities of zeta functions
2. Some examples of p-adic algorithms
3. Nondegenerate toric hypersurfaces
4. Controlled reduction in p-adic cohomology

Extreme generality: the Lauder-Wan method

Dwork's proof of the rationality of $Z(X, T)$ reduces to the case of an affine hypersurface, for which one writes down a trace formula involving a compact operator on an infinite-dimensional p-adic vector space.

By careful bounding, Lauder and Wan extracted from this an algorithm for computing $Z(X, T)$. If X is of degree d and fixed dimension over $\mathbb{F}_q$ with $q = p^a$, this runs in time $\mathrm{poly}(p, d, a)$.

Unfortunately, the implied exponents and constants seem to make this algorithm infeasible. Some special cases can be made to work (e.g., Artin-Schreier curves).

Harvey is working on a variant of Lauder–Wan modeled on Hasse-Witt matrices.

Extreme specificity: Elliptic curves

For ordinary elliptic curves, Satoh described an algorithm for computing $Z(X, T)$ using the Deuring-Serre-Tate canonical lift. This runs in time $\mathrm{poly}(p)a^{3+o(1)}$ and is quite feasible for small p.

When $p = 2$, one can do better using Mestre's AGM iteration, replacing a^3 with a^2.

However, neither of these generalizes well even to genus 2 curves.

Less specificity: curves

For hyperelliptic curves of genus g (with $p > 2$ and having a rational Weierstrass point), Kedlaya described an algorithm for computing $Z(X, T)$ by realizing $P_1(X, T)$ as the characteristic polynomial of Frobenius on Monsky-Washnitzer cohomology of the affine curve obtained by removing the Weierstrass points. This runs in time $(pg^4a^3)^{1+\epsilon}$ and is feasible.

This can be generalized (with different exponents): hyperelliptic curves with $p = 2$ (Denef-Vercauteren) or having no rational Weierstrass point (Harrison), superelliptic curves (Gaudry–Gürel), $C_{a,b}$-curves (Denef–Vercauteren), nondegenerate curves (Castryck–Denef–Vercauteren), all curves (Tuitman).

An alternate approach, which may be more practical in the general case, uses the cup product duality (Besser–de Jeu–Escriva).

Some improvements for hyperelliptic curves

Harvey improved the dependence on p for hyperelliptic curves to $p^{1/2+o(1)}$. This uses a modified description of the Frobenius action which we will see again later, plus a method for accelerating matrix recurrences (Chudnovskys, Bostan–Gaudry–Schost).

For a hyperelliptic curve over $\mathbb{Q}$, Harvey described a method for amortizing the computation of zeta functions over $\mathbb{F}_p$ for all $p \leq x$, to get *average polynomial time* (i.e., time $\mathrm{poly}(\log(p), a, g)$ per prime). This incorporates an idea of Gerbicz from the context of computing Wilson quotients (i.e., $(p-1)! \mod p^2$) using *balanced remainder trees*.

Higher dimensions: projective hypersurfaces

For smooth projective hypersurfaces, Abbott–Kedlaya–Roe described an algorithm for computing $Z(X, T)$ by working in the affine complement; we will see this trick again later. Unfortunately, the dependence on p goes like p^n for $n = \dim(X)$. The analogue of Castryck–Denef–Vercauteren behaves similarly.

Some alternatives that alleviate the dependence on p are Lauder's *deformation method* and *fibration method*. However, these seem to be feasible (so far) only for sparse polynomials.

Also available (and maybe feasible?) for sparse polynomials is Sperber–Voight, based on Dwork cohomology.

Hereafter, we describe a variant of AKR which has good (namely linear) dependence on p, can handle dense polynomials, and is feasible (shown by example!). One tradeoff is that we restrict the class of projective hypersurfaces slightly, but as a bonus we pick up many more examples.

Contents

1. Generalities of zeta functions
2. Some examples of p-adic algorithms
3. Nondegenerate toric hypersurfaces
4. Controlled reduction in p-adic cohomology

Lattices and differentials

Let R be a ring. Let L be a lattice of rank n. Let $L^\vee := \mathrm{Hom}_{\mathbb{Z}}(L, \mathbb{Z})$ denote the dual lattice.

Let $R[L]$ denote the monoid algebra. Concretely, if we fix a basis $\mathbf{e}_1, \dots, \mathbf{e}_n$ of L, we obtain an isomorphism

$$R[L] \cong R[x_1^\pm, \dots, x_n^\pm], \qquad [\mathbf{e}_i] \mapsto x_i.$$

Each $\lambda \in L^\vee$ defines a derivation ∂_λ on $R[L]$ via the formula

$$\partial_\lambda([\mathbf{v}]) = \lambda(\mathbf{v})[\mathbf{v}] \qquad (\mathbf{v} \in L);$$

these satisfy $\partial_{\lambda_1+\lambda_2} = \partial_{\lambda_1} + \partial_{\lambda_2}$. With a basis as above, for $\mathbf{e}_1^\vee, \dots, \mathbf{e}_n^\vee \in L^\vee$ the dual basis,

$$\partial_{\mathbf{e}_i^\vee} = x_i \frac{\partial}{\partial x_i}.$$

Polytopes and projective toric varieties

Let Δ be a convex lattice polytope of full dimension in $L_{\mathbb{R}} := L \otimes_{\mathbb{Z}} \mathbb{R}$, i.e., the convex hull of a finite subset of L not contained in any hyperplane. The cone over this polytope is then a fan defining a (polarized) projective toric variety over R. In simple cases, this can be computed as

$$X := \operatorname{Proj} P, \qquad P := \bigoplus_{d=0}^{\infty} P_d, \qquad P_d := R[d\Delta \cap L]$$

but in bad cases (e.g., for $\Delta = \operatorname{Conv}(0, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_1 + \mathbf{e}_2 + 3\mathbf{e}_3)$) one must take P to be Cox's *homogeneous coordinate ring*.

For example, for Δ the simplex with vertices $0, \mathbf{e}_1, \dots, \mathbf{e}_n$, we get projective space with its usual $\mathcal{O}(1)$. We similarly get weighted projective spaces, products, toric blowups, etc. Replacing Δ by $d\Delta$ preserves X but replaces the polarization by its d-th power.

Nondegeneracy

We say $f \in P_d$ is *nondegenerate* if the hypersurface

$$Z_f := \operatorname{Proj} P/(f)$$

cut out by f has transversal intersection with each torus in the natural stratification of X. In particular, this is required for the zero-dimensional strata, so f must have Newton polytope $d\Delta$.

It is equivalent to require that the *toric Jacobian ideal*

$$I_f = (f, \delta_\lambda(f) : \lambda \in L^\vee)$$

is irrelevant, that is, the *toric Jacobian ring* $J_f := P/I_f$ is module-finite over R. This condition is generic for "nice" P.

Note: if f is nondegenerate, then Z_f is "no more singular than X".

Some examples of nondegenerate hypersurfaces

n	Vertices of Δ	Resulting hypersurface
2	$0, d\mathbf{e}_1, d\mathbf{e}_2$	Smooth plane curve of genus $\binom{d-1}{2}$
2	$0, (2g+1)\mathbf{e}_1, \mathbf{e}_2$	Odd hyperelliptic curve of genus g
2	$0, a\mathbf{e}_1, b\mathbf{e}_2$	$C_{a,b}$-curve
2	$0, (g+1)\mathbf{e}_1,\ 2\mathbf{e}_2,$ $(g+1)\mathbf{e}_1 + 2\mathbf{e}_2$	Even hyperelliptic curve of genus g
3	$0, 4\mathbf{e}_1, 4\mathbf{e}_2, 4\mathbf{e}_3$	Quartic K3 surface
4	$0, 5\mathbf{e}_1, \dots, 5\mathbf{e}_5$	Quintic Calabi-Yau threefold

Contents

1. Generalities of zeta functions
2. Some examples of p-adic algorithms
3. Nondegenerate toric hypersurfaces
4. Controlled reduction in p-adic cohomology

Monsky-Washnitzer cohomology

From now on, work over $R = \mathbb{Z}_q$ and take $f \in P_1$ nondegenerate. (If $f \in P_d$ for $d > 0$, we may replace Δ with $d\Delta$ and then proceed.) Put $U_f := X \setminus Z_f$; this is an affine scheme with coordinate ring

$$S = \bigoplus_{m=0}^{\infty} f^{-m} P_m.$$

The *weak completion* $S^\dagger$ of S consists of infinite series $\sum_{m=0}^{\infty} g_m f^{-m}$ with $g_m \in P_m$ such that for some $a, b > 0$ (depending on the series),

$$v_p(g_m) \geq am - b \qquad (m \geq 0).$$

The *Monsky-Washnitzer cohomology* of $U_{\mathbb{F}_q}$ is the cohomology of the (continuous) de Rham complex $\Omega_{S^\dagger[p^{-1}]/\mathbb{Q}_q}$.

Action of Frobenius

Define a (semilinear) endomorphism σ of $S^\dagger$ as the absolute Frobenius lift on R, the substitution $[\mathbf{v}] \mapsto [\mathbf{v}]^p$ on monomials, and

$$g_m f^{-m} \mapsto \sum_{i=0}^{\infty} \sigma(g_m) \binom{-m}{i} (\sigma(f) - f^p)^i f^{-p(m+i)}.$$

The induced (linear) action of σ^a on MW cohomology computes $Z(Z_f, T)$. More precisely, for

$$H^n := \Omega^n / d(\Omega^{n-1}),$$

we have

$$Z(Z_f, T) = \frac{1}{(1-T)(1-qT)\cdots(1-q^{n-1}T)} P_f(T)^{(-1)^n}$$
$$P_f(T) = \det(1 - q^{-1}T\sigma^a, H^n \otimes_{\mathbb{Z}_q} \mathbb{Q}_q).$$

Griffiths-Dwork reduction

To compute the action of σ^a on the finite-dimensional $\mathbb{Q}_q$-vector space $H^n \otimes_{\mathbb{Z}_q} \mathbb{Q}_q$, we choose a basis, apply σ^a to each basis element, truncate the infinite sum somewhere, then reduce the result in cohomology. One way to do this is the Griffiths-Dwork reduction: for

$$\omega = \mathrm{dlog}[\mathbf{e}_1] \wedge \cdots \wedge \mathrm{dlog}[\mathbf{e}_n],$$

for $g_m \in P_m$, $\lambda \in L^\vee$ we have

$$\begin{aligned} \frac{g_m f}{f^{m+1}}\omega &\equiv \frac{g_m}{f^m}\omega \\ \frac{g_m \partial_\lambda(f)}{f^{m+1}}\omega &\equiv \frac{1}{m}\frac{\partial_\lambda(g_m)}{f^m}\omega \qquad (m > 0). \end{aligned}$$

Using a theorem of Macaulay, we lower the pole order to n and then finish with explicit linear algebra. This recovers the AKR algorithm.

Unfortunately, this involves dense polynomials of degree pn, and thus an unavoidable factor of p^n in the runtime. But there is another way...

A word on precision

Since the reduction process involves denominators, truncating σ modulo p^N does not guarantee correct computation of the matrix of action modulo p^N.

However, the loss of precision is bounded above by $n \log(pN)$, so the necessary working precision is not much larger than the sufficient final precision. We will hereafter ignore the distinction between the two. (It is particularly easy to analyze the situation when $p > n$.)

A sparse representation of Frobenius

Note that modulo p^N,

$$\begin{aligned}
\sigma\left(\frac{g_m}{f^m}\right) &\equiv \sigma(g_m) \sum_{i=0}^{N-1} \binom{-m}{i} (\sigma(f) - f^p)^i f^{-p(m+i)} \\
&= \sigma(g_m) \sum_{i=0}^{N-1} \binom{-m}{i} f^{-p(m+i)} \sum_{j=0}^{i} (-1)^{i-j} \binom{i}{j} \sigma(f)^j f^{p(i-j)} \\
&= \sigma(g_m) \sum_{j=0}^{N-1} \binom{-m}{j} \sigma(f)^j f^{-p(m+j)} \sum_{i=j}^{N-1} \binom{m+i-1}{m+j-1} \\
&= \sum_{j=0}^{N-1} \binom{-m}{j} \binom{m+N-1}{N-j-1} \sigma(g_m f^j) f^{-p(m+j)}.
\end{aligned}$$

The last expression is no longer the truncation of a p-adically convergent series, but no matter; it involves only p-th power monomials!

Controlled reduction

By the nondegeneracy hypothesis, we can construct linear maps $\pi_0, \dots, \pi_n : P_{n+1} \to P_n$ such that

$$P_{n+1}(g_{n+1}) = \pi_0(g_{n+1})f + \sum_{i=1}^n \pi_i(g_{n+1})\partial_{\mathbf{e}_i^*}(f).$$

Then for any $m, j \geq 0$ and any monomials $\mu \in P_1, \nu \in P_m$,

$$\frac{g_n \mu^{j+1}\nu}{f^{m+n+j+1}}\omega \equiv (m+n+j)^{-1}(R_{\mu,\nu}(g_n) + jS_\mu(g_n))\frac{\mu^j \nu}{f^{m+n+j}}\omega$$

for

$$R_{\mu,\nu}(x) := (m+n)\pi_0(\mu x) + \sum_{h=1}^n (\partial_{\mathbf{e}_h^*} + \mathbf{e}_h^*(\nu))(\pi_h(\mu x))$$

$$S_\mu(x) := \pi_0(\mu x) + \sum_{h=1}^n \mathbf{e}_h^*(\mu)\pi_h(\mu x).$$

More on controlled reduction

We thus can strip out μ^p by multiplying together p matrices of size

$$\#(n\Delta \cap L) \sim n^n \operatorname{Vol}(\Delta).$$

With a slightly more involved process, we can reduce the matrix size to $n! \operatorname{Vol}(\Delta)$, saving a factor of $(n^n/n!) \sim e^n$.

In case P is generated in degree 1, we can use controlled reduction to completely simplify the expressions occuring in the sparse Frobenius expansion.

Otherwise, the only issue is caused by monomials of the form $\sigma(g_m)$ for $m \in \{1, \dots, n\}$. This can be resolved in various ways, e.g., by writing a small power of g_m as a product of degree 1 monomials.

In any case, one must do some residual linear algebra at the end to reduce the matrix to the correct size (roughly a factor of n). For instance, for a quartic K3 surface, one must reduce the matrix size from 64 to 21.

A bit of complexity analysis

Unless $\log_p q$ is large, the dominant factor is the rounds of controlled reduction. The number of such rounds is

$$\#((n+N)\Delta \cap L) \sim (n+N)^n \operatorname{Vol}(\Delta)$$

Each round involves multiplying p matrices of size $n! \operatorname{Vol}(\Delta)$, so with straightforward matrix arithmetic we have $O(p(n+N)^n (n!)^3 \operatorname{Vol}(\Delta)^4)$ arithmetic operations. Note that the dependence on p is linear! (Warning: one must also factor in the p-adic precision.)

One can easily adapt for square-root dependence in p or average polynomial time dependence in $\log p$, but we have not attempted this.

A numerical example

This example computed by Edgar Costa (NYU) using C++/NTL.

Take $n := 3$, $\Delta := \mathrm{Conv}(0, 4\mathbf{e}_1, 4\mathbf{e}_2, 4\mathbf{e}_3)$. Write x_0, x_1, x_2, x_3 for $[0], [\mathbf{e}_1], [\mathbf{e}_2], [\mathbf{e}_3]$ and put

$$\begin{aligned} f := {} & 25163x_0^4 + 9405x_0^3x_1 + 85x_0^2x_1^2 + 30034x_0x_1^3 + 21740x_1^4 \\ & + 14747x_0^3x_2 + 35394x_0^2x_1x_2 + 13683x_0x_1^2x_2 + 12720x_1^3x_2 \\ & + 36331x_0^2x_2^2 + 23023x_0x_1x_2^2 + 25667x_1^2x_2^2 + 7066x_0x_2^3 + 6479x_1x_2^3 \\ & + 8778x_2^4 + 40922x_0^3x_3 + 38119x_0^2x_1x_3 + 48775x_0x_1^2x_3 + 9720x_1^3x_3 \\ & + 20633x_0^2x_2x_3 + 41354x_0x_1x_2x_3 + 31769x_1^2x_2x_3 + 32904x_0x_2^2x_3 \\ & + 49443x_1x_2^2x_3 + 24957x_2^3x_3 + 37766x_0^2x_3^2 + 8622x_0x_1x_3^2 + 3377x_1^2x_3^2 \\ & + 15688x_0x_2x_3^2 + 10170x_1x_2x_3^2 + 19668x_2^2x_3^2 + 2486x_0x_3^3 + 13807x_1x_3^3 \\ & + 15264x_2x_3^3 + 27566x_3^4. \end{aligned}$$

Then Z_f is a nondegenerate quartic K3 surface in $\mathbb{P}^3_{\mathbb{Q}}$.

A numerical example (continued)

Take $p := 49999$. In 5h45m on a single-core 2.6GHz Intel Xeon (Sandy Bridge), one computes

$$\begin{aligned} P_2(Z_f, T) = &1 + a_1 T + a_2 p T^2 + \cdots + a_{10} p^9 T^{10} \\ &- a_{10} p^{10} T^{11} - \cdots - a_2 p^{18} T^{19} - a_1 p^{19} T^{20} + p^{21} \end{aligned}$$

with

$$\begin{aligned} (a_1, \dots, a_{10}) = &(33264, -81893, -32490, 86146, 23017, \\ &- 55214, -22632, -2392, 43164, 47726). \end{aligned}$$

This has roots in $\mathbb{C}$ as predicted by the Weil conjectures (see Sage notebook).

What next?

It would be worth trying to build a SAGE implementation which would allow for arbitrary polytopes (as long as they are generated in degree 1). This would allow experiments in many new examples!

To get reasonable results, it might be necessary to build the matrix multiplication part of controlled reduction as a compiled black box. However, one should be able to leave the rest in interpreted SAGE.