Efficiently computing Igusa's local-zeta function

Nitin Saxena (CSE@IIT Kanpur, India)

(appeared in ANTS-XIV'20; with Ashish Dwivedi)

<u>Oberseminar</u>

Nov 2021, Univ. Bayreuth (virtually)

- Zeta functions
- Igusa's local-zeta fn
- Algorithmic questions
- Root finding mod p^k
- Root counting mod p^k
- Compute Poincaré Series
- Conclusion

Zeta functions

- For function N_k there's generating-function $G(t) := \sum_{k>0} N_k t^k$.
 - This carries comprehensive information about N_{μ} .
 - Eg. the growth of N_{μ} decides how the **power-series** converges.
- Riemann zeta-fn: $\zeta(s) = \sum_{k \ge 1} 1/k^s$.
 - What's it encoding?
- Inspired many other zeta functions:
 - Selberg zeta fn of a manifold
 - Ruelle zeta fn of a dynamical system
 - Ihara zeta fn of a graph
- Local-zeta functions (based on a prime p):
 - Hasse-Weil zeta fn
 - Igusa local-zeta fn



Riemann 1826-66

Eg. Ramanujan tau-function

To count points

Galois field vs ring $Z/p^{k}Z$

- Zeta functions
- Igusa's local-zeta fn
- Algorithmic questions
- Root finding mod p^k
- Root counting mod p^k
- Compute Poincaré Series
- Conclusion

Igusa's local-zeta function

- Let Z_p denote p-adic integers.
 - → Elements are $\sum_{i\geq 0} a_i p^i$ ($a_i \in [0, p-1]$).
- Let $f = f(x_1, ..., x_n)$ be n-variate integral polynomial.
- Defn.1: Igusa's local-zeta fn $Z_{f,p}(s) = \int_{(Z_p)^n} |f(\mathbf{x})|_p^s |d\mathbf{x}|$.
 - Integrate using *p*-adic metric & Haar measure.
- This converges to a <u>rational</u> function in Q(p^s).
 - (Igusa'74) by resolving singularities.
 - (Denef'84) by p-adic cell decomposition.
- Counts roots f(x) mod p^k & `multiplies' by p^{-ks}.
- So, we can give another definition:



infinite sum



Igusa's local-zeta function

Defns: Analytic vs Discrete

- Define $N_k(f) := # \text{ roots of } f(\mathbf{x}) \mod p^k$.
- Defn.2: Poincaré Series P_{f,p}(t) = ∑_{k≥0} N_k(f)/p^{nk}.t^k.
 Eg. P_{0,p}(t) = ∑_{k≥0} t^k = 1/(1-t).
 - (Igusa'74) connected them at $t=p^{-s}$: P(t).(1-t) = 1 t.Z(s).
- (Igusa'74) $P_{f,p}(t)$ converges to a <u>rational</u> function in Q(t).
- This means that $N_k(f)$ is rather special!
 - Generally, power-series don't converge in Q(t).
 - Eg. $\sum_{k>0} (1/k!) \cdot t^k$ is irrational !
- Convergence proofs are quite *non*-explicit.
 - What do we learn about $N_k(f)$, for small k ?

- Zeta functions
- Igusa's local-zeta fn
- Algorithmic questions
- Root finding mod p^k
- Root counting mod p^k
- Compute Poincaré Series
- Conclusion

Algorithmic questions

- Qn: Could N_k(f) be computed efficiently?
- Trivially, in p^{kn} time.
 - Much faster unlikely.
 - It's NP-hard; even Permanent-hard !
- Could N_k(f) be computed efficiently, for univariate f(x)?
 Qn: In poly(deg(f), log p, k) time?
- Or, try to compute the integral in $Z_{f,p}(s)$.
- (Chistov'87) gave a randomized algorithm to factor f(x) over Z_{n} .

In p-adic extensions

- Using this one could factor f into roots,
- and attempt the integration ...?
- **Qn:** But, a *deterministic poly*-time algorithm for $N_k(f)$?

- Zeta functions
- Igusa's local-zeta fn
- Algorithmic questions
- Root finding mod p^k
- Root counting mod p^k
- Compute Poincaré Series
- Conclusion

Root-finding mod p^k

- Instead of integration, we take the route of roots mod p^k.
- Let $f \mod p^k$ be degree d univariate polynomial.
- (Berthomieu,Lecerf,Quintin'13) Roots of f mod p^k arrange as representative-roots:
 - → $\mathbf{a} =: \sum_{0 \le i < l} a_i p^i + *p^l$ ($a_i \in [0, p-1]$, *∈Z).
 - → **a** is minimal & $f(\mathbf{a}) = 0 \mod p^k$.
 - At most d rep.roots.
- Proof is *inductive*, based on the transformation:
 - $g(x) := f(\sum_{0 \le i \le m} a_i p^i + x.p^m) / p^v \mod p^{k-v}$. Reduces char p^k to p
 - Root of $g(x) \mod p$ gives a_m .
 - Continue with $\sum_{0 \le i \le m} a_i p^i$.

Why are rep.roots few?

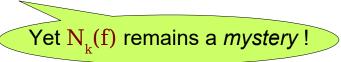
Root-finding mod p^k

- Rep.roots are few, but roots may be *exponentially* many!
 - Eg. $f := px \mod p^2$ has p roots,
 - but just one rep.root $\mathbf{a} =: \mathbf{0} + \mathbf{*p}$!
- (BLQ'13) yields fast randomized algorithm to find roots mod p^k.
 - Counting is easy, as rep.root a means p^{k-1} roots.
 - $\mathbf{a} = \sum_{0 \le i < l} a_i p^i + *p^l$.
 - Summing up over rep.roots, gives all roots.
- How to make it *deterministic* poly-time?
- Rep.roots yield $N_k(f) = \sum_i p^{k-l_i}$.
 - What does it say about Poincaré series P_{fn}(t) ?

 l_i depends on i, k

Root-finding mod p^k

- (Dwivedi,Mittal,S '19) gave fast deterministic algorithm to implicitly find roots mod p^k.
- Idea: Store rep.roots $\mathbf{a} = \sum_{0 \le i < l} a_i p^i + *p^l$ in maximal split ideals.
 - $\mathbf{I} = \langle \mathbf{h}_0(\mathbf{x}_0), \mathbf{h}_1(\mathbf{x}_0, \mathbf{x}_1), \dots, \mathbf{h}_{l-1}(\mathbf{x}_0, \dots, \mathbf{x}_{l-1}) \rangle$.
 - Each zero of **I** in $\mathbf{F}_{\mathbf{p}}^{-1}$ defines a rep.root.
 - Essentially, run (BLQ'13) mod I (without randomization!).
 - Keep `growing' I.
- (DMS'19) yields fast deterministic algorithm to count roots f mod p^k.



- Zeta functions
- Igusa's local-zeta fn
- Algorithmic questions
- Root finding mod p^k
- Root counting mod p^k
- Compute Poincaré Series
- Conclusion

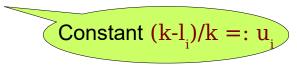
Root-counting mod p^k

- Intuitively, $N_k(f) = \sum_i p^{k-1_i}$ should behave better for large k.
 - Since, large k is like studying roots in Z_n .
- We show, for large $\mathbf{k} : \mathbf{l}_{i}$ is *linear* in \mathbf{k} .



- $k > k_0 := deg(f) * val_p(disc(rad(f)))$.
- $l_i = [(k val_p(f_i(\alpha_i))) / mult(\alpha_i)].$ ■ Where, α_i are all p-adic integer roots of f(x).
- Curiously, squarefree $f \& \text{ large } k \Rightarrow N_k(f)$ independent of k.

Roots *uniquely lift* as <u>k</u> grows.



- Zeta functions
- Igusa's local-zeta fn
- Algorithmic questions
- Root finding mod p^k
- Root counting mod p^k
- Compute Poincaré Series
- Conclusion

Compute Poincaré Series

• Got :
$$N_k(f) = \sum_i p^{k.u_i}$$
 for $k > k_0$.

• So,
$$P(t) = \sum_{k \ge 0} N_k(f)/p^k . t^k$$
,
= $P_0(t) + \sum_{k \ge k_0} N_k(f)/p^k . t^k$,
= $P_0(t) + \sum_{k \ge k_0} \sum_i p^{k.(u_i-1)} . t^k$.

- The infinite sum converges to a rational, in Q(t).
- Thus, P(t) is a rational function.
- Our algorithm computes N_k(f); hence, both P₀(t) and the infinite sum are *known*.
 - In poly(|f|, log p) time.

- Zeta functions
- Igusa's local-zeta fn
- Algorithmic questions
- Root finding mod p^k
- Root counting mod p^k
- Compute Poincaré Series
- Conclusion

At the end ...

- Det.poly-time algorithm for Igusa's local-zeta function.
 For univariate polynomial f.
- Could we do this for bivariate polynomial $f(x_1, x_2)$?

Relevant Qns:

- 1. Estimating the count $N_k(f(x_1, x_2)) = ?$
- 2. Counting factors of $f(x) \mod p^k$?
 - Irreducibility-testing of $f(x) \mod p^5$?

