Multivariate polynomials modulo prime powers: their roots, zeta-function and applications

M.Tech Thesis Defense

Sayak Chakrabarti

17807648
Supervisor: Prof. Nitin Saxena

List of Papers

[CDS22] Factoring modular polynomials via Hilbert's Nullstellensatz,
Sayak Chakrabarti, Ashish Dwivedi and Nitin Saxena,
Manuscript, 2022.

List of Papers

[CDS22] Factoring modular polynomials via Hilbert's Nullstellensatz,
Sayak Chakrabarti, Ashish Dwivedi and Nitin Saxena,
Manuscript, 2022.

[CS22] Describing the roots of multivariates mod p^k and efficient computation of Igusa's local zeta function,
Sayak Chakrabarti and Nitin Saxena,
Manuscript, 2022.

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

Finite Field $\mathbb{F}_p$:

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

Finite Field $\mathbb{F}_p$:

- a, $a \in \{0, \ldots, p-1\}$,

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

Finite Field $\mathbb{F}_p$:

- a, $a \in \{0, \ldots, p-1\}$,
- Integral domain,

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

Finite Field $\mathbb{F}_p$:

- a, $a \in \{0, \ldots, p-1\}$,
- Integral domain,
- 'Nice' properties.

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

Finite Field $\mathbb{F}_p$:

- a, $a \in \{0, \ldots, p-1\}$,
- Integral domain,
- 'Nice' properties.

p-adic integers $\mathbb{Z}_p$:

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

Finite Field $\mathbb{F}_p$:

- a, $a \in \{0, \ldots, p-1\}$,
- Integral domain,
- 'Nice' properties.

p-adic integers $\mathbb{Z}_p$:

- $a_0 + a_1 p + a_2 p^2 + \ldots$, $a_i \in \{0, \ldots, p-1\}$,

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

Finite Field $\mathbb{F}_p$:

- a, $a \in \{0, \ldots, p-1\}$,
- Integral domain,
- 'Nice' properties.

p-adic integers $\mathbb{Z}_p$:

- $a_0 + a_1 p + a_2 p^2 + \ldots$, $a_i \in \{0, \ldots, p-1\}$,
- Integral domain,

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

Finite Field $\mathbb{F}_p$:

- a, $a \in \{0, \ldots, p-1\}$,
- Integral domain,
- 'Nice' properties.

p-adic integers $\mathbb{Z}_p$:

- $a_0 + a_1 p + a_2 p^2 + \ldots,\ a_i \in \{0, \ldots, p-1\}$,
- Integral domain,
- (Less) 'nice' properties.

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

$\boxed{\mathbb{Z}/p^k\mathbb{Z}}$

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

$$\boxed{\mathbb{Z}/p^k\mathbb{Z}}$$

- $k = 1$: $\mathbb{F}_p$.

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

$$\boxed{\mathbb{Z}/p^k\mathbb{Z}}$$

- $k = 1$: $\mathbb{F}_p$.
- $k \to \infty$: $\mathbb{Z}_p$.

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

$$\boxed{\mathbb{Z}/p^k\mathbb{Z}}$$

- $k = 1$: $\mathbb{F}_p$.
- k is 'large': $\mathbb{Z}_p$.

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

$$\boxed{\mathbb{Z}/p^k\mathbb{Z}}$$

- $k = 1$: $\mathbb{F}_p$.
- k is 'large': $\mathbb{Z}_p$.
- $2 < k < C$?

From $\mathbb{F}_p$ to $\mathbb{Z}/p^k\mathbb{Z}$ to $\mathbb{Z}_p$

$$\boxed{\mathbb{Z}/p^k\mathbb{Z}}$$

- $k = 1$: $\mathbb{F}_p$.
- k is 'large': $\mathbb{Z}_p$.
- $2 < k < C$?
  - *not* integral domain!

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.
- $a_0 + a_1 p + \cdots + a_{k-1} p^{k-1} \longrightarrow$ extract a_i at i-th step:

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.
- $a_0 + a_1 p + \cdots + a_{k-1}p^{k-1} \longrightarrow$ extract a_i at i-th step:
  - $\tilde{f}(x) := p^{-v} f(a_0 + px)$; $v = v_p(f(a_0 + px))$.

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.
- $a_0 + a_1 p + \cdots + a_{k-1}p^{k-1} \longrightarrow$ extract a_i at i-th step:
  - $\tilde{f}(x) := p^{-v} f(a_0 + px)$; $v = v_p(f(a_0 + px))$.

Lifting Step

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.
- $a_0 + a_1 p + \cdots + a_{k-1}p^{k-1} \longrightarrow$ extract a_i at i-th step:
  - $\tilde{f}(x) := p^{-v} f(a_0 + px)$; $v = v_p(f(a_0 + px))$.

    Lifting Step

  - Find $\mathbb{F}_p$ roots of $\tilde{f}(x)$ using [CZ81].

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.
- $a_0 + a_1 p + \cdots + a_{k-1}p^{k-1} \longrightarrow$ extract a_i at i-th step:
  - $\tilde{f}(x) := p^{-v} f(a_0 + px)$; $v = v_p(f(a_0 + px))$.

    Lifting Step

  - Find $\mathbb{F}_p$ roots of $\tilde{f}(x)$ using [CZ81].
  - What if $v > 1$?

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.
- $a_0 + a_1 p + \cdots + a_{k-1} p^{k-1} \longrightarrow$ extract a_i at i-th step:
  - $\tilde{f}(x) := p^{-v} f(a_0 + px)$; $v = v_p(f(a_0 + px))$.

    Lifting Step

  - Find $\mathbb{F}_p$ roots of $\tilde{f}(x)$ using [CZ81].
  - What if $v > 1$?
  - $k := k - v$.

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.
- $a_0 + a_1 p + \cdots + a_{k-1}p^{k-1} \longrightarrow$ extract a_i at i-th step:
  - $\tilde{f}(x) := p^{-v} f(a_0 + px)$; $v = v_p(f(a_0 + px))$.

$$\boxed{\text{Lifting Step}}$$

  - Find $\mathbb{F}_p$ roots of $\tilde{f}(x)$ using [CZ81].
  - What if $v > 1$?
  - $k := k - v$.
- Continue until required exponent achieved.

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.
- $a_0 + a_1 p + \cdots + a_{k-1}p^{k-1} \longrightarrow$ extract a_i at i-th step:
    - $\tilde{f}(x) := p^{-v} f(a_0 + px)$; $v = v_p(f(a_0 + px))$.

      Lifting Step

    - Find $\mathbb{F}_p$ roots of $\tilde{f}(x)$ using [CZ81].
    - What if $v > 1$?
    - $k := k - v$.
- Continue until required exponent achieved.
- Modification: system of polynomials:

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.
- $a_0 + a_1 p + \cdots + a_{k-1} p^{k-1} \longrightarrow$ extract a_i at i-th step:
  - $\tilde{f}(x) := p^{-v} f(a_0 + px)$; $v = v_p(f(a_0 + px))$.

    Lifting Step

  - Find $\mathbb{F}_p$ roots of $\tilde{f}(x)$ using [CZ81].
  - What if $v > 1$?
  - $k := k - v$.
- Continue until required exponent achieved.
- Modification: system of polynomials:
  - Find roots of $\tilde{f}_1(x) := p^{-v_1} f_1(a_0 + px), \ldots, \tilde{f}_m(x) := p^{-v_m} f_m(a_0 + px)$.

Univariate root-finding [BLQ13, DMS21]

Main Idea:

- Reduce $\mathbb{Z}/p^k\mathbb{Z}$ root-finding to $\mathbb{F}_p$ root-finding.
- $a_0 + a_1 p + \cdots + a_{k-1}p^{k-1} \longrightarrow$ extract a_i at i-th step:
  - $\tilde{f}(x) := p^{-v} f(a_0 + px)$; $v = v_p(f(a_0 + px))$.

    $\boxed{\text{Lifting Step}}$

  - Find $\mathbb{F}_p$ roots of $\tilde{f}(x)$ using [CZ81].
  - What if $v > 1$?
  - $k := k - v$.
- Continue until required exponent achieved.
- Modification: system of polynomials:
  - Find roots of $\tilde{f}_1(x) := p^{-v_1} f_1(a_0 + px), \ldots, \tilde{f}_m(x) := p^{-v_m} f_m(a_0 + px)$.
  - $k_1 := k_1 - v_1, \ldots, k_m := k_m - v_m$

Univariate root-finding: Representative roots [Pan95, BLQ13, DMS21]

- $v > 1 \implies$ less than k lifting steps.

Univariate root-finding: Representative roots [Pan95, BLQ13, DMS21]

- $v > 1 \implies$ less than k lifting steps.
- $a_0 + a_1 p + \cdots + a_r p^r + p^{r+1} *$, $*$ represents $\mathbb{Z}/p^{k-r}\mathbb{Z}$– compact data-structure.

Univariate root-finding: Representative roots [Pan95, BLQ13, DMS21]

- $v > 1 \implies$ less than k lifting steps.
- $a_0 + a_1 p + \cdots + a_r p^r + p^{r+1}*$, $*$ represents $\mathbb{Z}/p^{k-r}\mathbb{Z}$– compact data-structure.
- Example: $x^2 \bmod p^{2n}$

Univariate root-finding: Representative roots [Pan95, BLQ13, DMS21]

- $v > 1 \implies$ less than k lifting steps.
- $a_0 + a_1 p + \cdots + a_r p^r + p^{r+1}*$, $*$ represents $\mathbb{Z}/p^{k-r}\mathbb{Z}$– compact data-structure.
- Example: $x^2 \bmod p^{2n}$– root given by $0 + p^n*$

Part I

Roots of $f(x_1, x_2)$ mod p^k in deterministic poly$((d + p + k^) d)$ time.

Describing the roots of multivariates mod p^k and efficient computation of Igusa's local zeta function

Importance of the problem

- Data-structure to give all the roots:
  - Root finding of curves: elliptic curves, Diophantine equations.
  - Root counting: cryptography, #P-complete.
  - System of equations: NP-complete.

Describing the roots of multivariates mod p^k and efficient computation of Igusa's local zeta function

Importance of the problem

- Data-structure to give all the roots:
  - Root finding of curves: elliptic curves, Diophantine equations.
  - Root counting: cryptography, #P-complete.
  - System of equations: NP-complete.
- Roots over $\mathbb{Z}_p$.

Describing the roots of multivariates mod p^k and efficient computation of Igusa's local zeta function

Importance of the problem

- Data-structure to give all the roots:
  - Root finding of curves: elliptic curves, Diophantine equations.
  - Root counting: cryptography, #P-complete.
  - System of equations: NP-complete.
- Roots over $\mathbb{Z}_p$.
- Rationality of Poincaré series and computation of Igusa's local zeta function.

Describing the roots of multivariates mod p^k and efficient computation of Igusa's local zeta function

Importance of the problem

- Data-structure to give all the roots:
  - Root finding of curves: elliptic curves, Diophantine equations.
  - Root counting: cryptography, #P-complete.
  - System of equations: NP-complete.
- Roots over $\mathbb{Z}_p$.
- Rationality of Poincaré series and computation of Igusa's local zeta function.

Previous work

- Restricted to univariates:
  - Lifting of roots: [BLQ13, NRS17, Pan95, DMS21].
  - Counting of roots: [DMS19, CGRW19, KRRZ20, RRZ21].
  - Igusa's LZF: [DS20, ZG03].
- $\mathbb{Z}_p$: [Chi21, DS20].
- Igusa's LZF: [Den84]

Roots of bivariates

Main ideas

- **Lifting step:** $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$ $(\deg(f) = d, \deg(f \bmod p) = d_1)$.

Roots of bivariates

Main ideas

- **Lifting step:** $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$ $(\deg(f) = d, \deg(f \bmod p) = d_1)$.
- Enumerate over roots of $\tilde{f}(x_1, x_2)$ in $\mathbb{F}_p^2$: branches of a tree.

Roots of bivariates

Main ideas

- **Lifting step:** $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$ $(\deg(f) = d, \deg(f \bmod p) = d_1)$.
- Enumerate over roots of $\tilde{f}(x_1, x_2)$ in $\mathbb{F}_p^2$: branches of a tree.
- Lift again...

Roots of bivariates

Main ideas

- **Lifting step:** $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$ $(\deg(f) = d, \deg(f \bmod p) = d_1)$.
- Enumerate over roots of $\tilde{f}(x_1, x_2)$ in $\mathbb{F}_p^2$: branches of a tree.
- Lift again...
- **Goal:** exhaust k

Roots of bivariates

Main ideas

- **Lifting step:** $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$ $(\deg(f) = d, \deg(f \bmod p) = d_1)$.
- Enumerate over roots of $\tilde{f}(x_1, x_2)$ in $\mathbb{F}_p^2$: branches of a tree.
- Lift again...
- **Goal:** exhaust k (or nice roots that lift to any power of p)

Roots of bivariates

Main ideas

- **Lifting step:** $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$ $(\deg(f) = d, \deg(f \bmod p) = d_1)$.
- Enumerate over roots of $\tilde{f}(x_1, x_2)$ in $\mathbb{F}_p^2$: branches of a tree.
- Lift again...
- **Goal:** exhaust k (or nice roots that lift to any power of p)
- Bound on number of lifting steps (depth of tree):

Roots of bivariates

Main ideas

- **Lifting step:** $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$ $(\deg(f) = d, \deg(f \bmod p) = d_1)$.
- Enumerate over roots of $\tilde{f}(x_1, x_2)$ in $\mathbb{F}_p^2$: branches of a tree.
- Lift again...
- **Goal:** exhaust k (or nice roots that lift to any power of p)
- Bound on number of lifting steps (depth of tree):
  - Effective degree 'usually' reduces.

Roots of bivariates

Main ideas

- **Lifting step:** $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$ $(\deg(f) = d, \deg(f \bmod p) = d_1)$.
- Enumerate over roots of $\tilde{f}(x_1, x_2)$ in $\mathbb{F}_p^2$: branches of a tree.
- Lift again...
- **Goal:** exhaust k (or nice roots that lift to any power of p)
- Bound on number of lifting steps (depth of tree):
  - Effective degree 'usually' reduces.
  - Bad case: reduce to univariate root finding.

Roots of bivariates: behavior of effective degree

Theorem

Given $f(x_1, x_2)$ *of effective degree* d_1 *which has a root* $(a_1, a_2) \bmod p$ *of val-multiplitcity* v*, define* $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$*. Let* $d_2 := \deg(\tilde{f}(x_1, x_2) \bmod p)$*. We have the following:*

Roots of bivariates: behavior of effective degree

Theorem

Given $f(x_1, x_2)$ *of effective degree* d_1 *which has a root* $(a_1, a_2) \bmod p$ *of val-multiplitcity* v, *define* $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$. *Let* $d_2 := \deg(\tilde{f}(x_1, x_2) \bmod p)$. *We have the following:*

- *If* $d_1 \geq 1$, $d_2 \leq v \leq d_1$. *Equality holds iff* $v = d_1$.

Roots of bivariates: behavior of effective degree

Theorem

Given $f(x_1, x_2)$ *of effective degree* d_1 *which has a root* $(a_1, a_2) \bmod p$ *of val-multiplitcity* v*, define* $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$*. Let* $d_2 := \deg(\tilde{f}(x_1, x_2) \bmod p)$*. We have the following:*

- *If* $d_1 \geq 1$*,* $d_2 \leq v \leq d_1$*. Equality holds iff* $v = d_1$*.*
- *If* $d_1 = 1$*, then* $d_2 = 1$*.*

Roots of bivariates: behavior of effective degree

Theorem

Given $f(x_1, x_2)$ *of effective degree* d_1 *which has a root* $(a_1, a_2) \bmod p$ *of val-multiplitcity* v*, define* $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$*. Let* $d_2 := \deg(\tilde{f}(x_1, x_2) \bmod p)$*. We have the following:*

- *If* $d_1 \geq 1$, $d_2 \leq v \leq d_1$*. Equality holds iff* $v = d_1$*.*
- *If* $d_1 = 1$*, then* $d_2 = 1$*. (Hensel's lifting of roots)*

Roots of bivariates: behavior of effective degree

Theorem

Given $f(x_1, x_2)$ of effective degree d_1 which has a root $(a_1, a_2) \bmod p$ of val-multiplitcity v, define $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$. Let $d_2 := \deg(\tilde{f}(x_1, x_2) \bmod p)$. We have the following:

- *If $d_1 \geq 1$, $d_2 \leq v \leq d_1$. Equality holds iff $v = d_1$.*
- *If $d_1 = 1$, then $d_2 = 1$. (Hensel's lifting of roots)*

Proof idea:

- Taylor's expansion terms,

$$f(a_1 + px_1, a_2 + px_2) = \sum_{\ell=0}^{d} \left(\sum_{|\mathbf{i}|=\ell} \frac{\partial_{\mathbf{x}^{\mathbf{i}}} f(\mathbf{a})}{\mathbf{i}!} \cdot (px_1)^{i_1} (px_2)^{i_2} \right).$$

Roots of bivariates: behavior of effective degree

Theorem

Given $f(x_1, x_2)$ of effective degree d_1 which has a root (a_1, a_2) mod p of val-multiplitcity v, define $\tilde{f}(x_1, x_2) := p^{-v} f(a_1 + px_1, a_2 + px_2)$. Let $d_2 := \deg(\tilde{f}(x_1, x_2) \bmod p)$. We have the following:

- *If $d_1 \geq 1$, $d_2 \leq v \leq d_1$. Equality holds iff $v = d_1$.*
- *If $d_1 = 1$, then $d_2 = 1$. (Hensel's lifting of roots)*

Proof idea:

- Taylor's expansion terms,

$$f(a_1 + px_1, a_2 + px_2) = \sum_{\ell=0}^{d} \left(\sum_{|\mathbf{i}|=\ell} \frac{\partial_{\mathbf{x}^{\mathbf{i}}} f(\mathbf{a})}{\mathbf{i}!} \cdot (px_1)^{i_1} (px_2)^{i_2} \right).$$

- When $d_1 = 1$, $\ell x_1 + m x_2 + n + p.g(x_1, x_2) \mapsto p^{-1}(\ell(a_1 + px_1) + m(a_2 + px_2) + n + p.g(a_1 + px_1, a_2 + px_2)$.
  - Fix x_1 to any value and find x_2 at every step.

Roots of bivariates: behavior of effective degree

Example

Consider $f(x_1, x_2) = x_1^2 + x_2^3$:

Roots of bivariates: behavior of effective degree

Example

Consider $f(x_1, x_2) = x_1^2 + x_2^3$:

- Root $(0, 0)$,
- Polynomial after lifting: $p^{-2}((0 + px_1)^2 + (0 + px_2)^3) = x_1^2 + px_2^3$.

Example

Consider $f(x_1, x_2) = x_1^3 + x_2^3$, $p = 5$:

Roots of bivariates: behavior of effective degree

Example

Consider $f(x_1, x_2) = x_1^2 + x_2^3$:

- Root $(0, 0)$,
- Polynomial after lifting: $p^{-2}((0 + px_1)^2 + (0 + px_2)^3) = x_1^2 + px_2^3$.

Example

Consider $f(x_1, x_2) = x_1^3 + x_2^3$, $p = 5$:

- Root $(1, 4)$,
- Polynomial after lifting: $x_1 + 4x_2 + 5(3x_1^2 + 12x_2^2 + 5x_1^3 + 5x_2^3)$.

Roots of bivariates: behavior of effective degree

Example

Consider $f(x_1, x_2) = x_1^2 + x_2^3$:

- Root $(0, 0)$,
- Polynomial after lifting: $p^{-2}((0 + px_1)^2 + (0 + px_2)^3) = x_1^2 + px_2^3$.

Example

Consider $f(x_1, x_2) = x_1^3 + x_2^3$, $p = 5$:

- Root $(1, 4)$,
- Polynomial after lifting: $x_1 + 4x_2 + 5(3x_1^2 + 12x_2^2 + 5x_1^3 + 5x_2^3)$.

Effective degree reduction– upto linear form

Roots of bivariates: Linear-representative roots

- Effective polynomial $\ell x_1 + mx_2 + n + p.g(x_1, x_2)$.

Roots of bivariates: Linear-representative roots

- Effective polynomial $\ell x_1 + mx_2 + n + p.g(x_1, x_2)$.
- For each precision coordinate, fix x_1 and find x_2.

Roots of bivariates: Linear-representative roots

- Effective polynomial $\ell x_1 + m x_2 + n + p.g(x_1, x_2)$.
- For each precision coordinate, fix x_1 and find x_2.
- n changes.

- Effective polynomial $\ell x_1 + m x_2 + n + p.g(x_1, x_2)$.
- For each precision coordinate, fix x_1 and find x_2.
- n changes.
- Computable function $c(\cdot)$, fixing x_1 coordinate gives x_2.

Roots of bivariates: Linear-representative roots

- Effective polynomial $\ell x_1 + m x_2 + n + p.g(x_1, x_2)$.
- For each precision coordinate, fix x_1 and find x_2.
- n changes.
- Computable function $c(\cdot)$, fixing x_1 coordinate gives x_2.
- $(*, c(*))$.

Roots of bivariates: Val-multiplicity$= d_1$ case

- **Goal:** Bring constant degree nodes into the same level.

Roots of bivariates: Val-multiplicity$=d_1$ case

- **Goal:** Bring constant degree nodes into the same level.
- Val-mult. $= d_1$ root exists $\implies f(x_1, x_2)$ mod p is a d_1-form:

$$\sum_{i=0}^{d_1} c_i (x_1 - a_1)^i (x_2 - a_2)^{d_1 - i}$$

Roots of bivariates: Val-multiplicity$= d_1$ case

- **Goal:** Bring constant degree nodes into the same level.
- Val-mult. $= d_1$ root exists $\implies f(x_1, x_2) \bmod p$ is a d_1-form:

$$\sum_{i=0}^{d_1} c_i (x_1 - a_1)^i (x_2 - a_2)^{d_1 - i}$$

- Multiple val-mult. $= d_1$ roots exist ($(0,0)$ and (a_1, a_2)):

$$f(x_1, x_2) \equiv c(a_2 x_1 - a_1 x_2)^{d_1} \bmod p$$

Roots of bivariates: Val-multiplicity$= d_1$ case

- Single val-mult. $= d_1$ root exists $\implies$ $f(x_1, x_2)$ mod p is a d_1-form: **[d_1-nonpower form]**

$$\sum_{i=0}^{d_1} c_i(x_1 - a_1)^i (x_2 - a_2)^{d_1 - i}.$$

- Multiple val-mult. $= d_1$ roots exist ($(0,0)$ and (a_1, a_2)): **[d_1-power]**

$$f(x_1, x_2) \equiv c(a_2 x_1 - a_1 x_2)^{d_1} \bmod p.$$

Roots of bivariates: Val-multiplicity$= d_1$ case

- Single val-mult. $= d_1$ root exists $\implies$ $f(x_1, x_2)$ mod p is a d_1-form: **[d_1-nonpower form]**

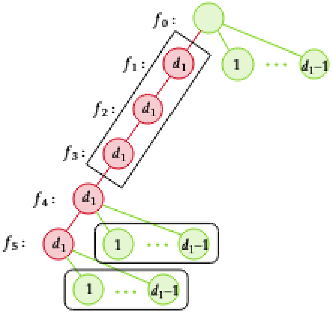
$$\sum_{i=0}^{d_1} c_i(x_1 - a_1)^i (x_2 - a_2)^{d_1 - i}.$$

- Multiple val-mult. $= d_1$ roots exist ($(0,0)$ and (a_1, a_2)): **[d_1-power]**

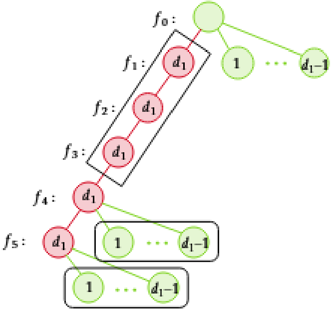
$$f(x_1, x_2) \equiv c(a_2x_1 - a_1x_2)^{d_1} \bmod p.$$

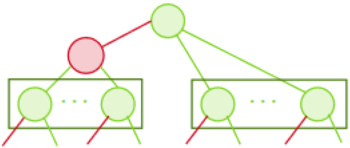
- d_1-nonpower form $\not\longrightarrow$ d_1-power
- d_1-nonpower form: contiguous chain clubbed to same level; $O(k)$-many degree reducing cases.
- d_1-power form: can lead to several constant degree cases.

Roots of bivariates: Val-multiplicity= d_1 case



Roots of bivariates: Val-multiplicity$= d_1$ case





Roots of bivariates: Structure of contiguous d_1-powers

- $f(x_1, x_2) \equiv L^{d_1} \bmod p$

Roots of bivariates: Structure of contiguous d_1-powers

- $f(x_1, x_2) \equiv L^{d_1} \bmod p$
- $\tilde{f}(L, x_2) \equiv L^{d_1} \bmod p$– change of basis

Roots of bivariates: Structure of contiguous d_1-powers

- $f(x_1, x_2) \equiv L^{d_1} \mod p$
- $\tilde{f}(L, x_2) \equiv L^{d_1} \mod p$– change of basis
- Lift $(L, x_2) \mapsto (pL, x_2)$, followed by division by p^{d_1}.

Roots of bivariates: Structure of contiguous d_1-powers

- $f(x_1, x_2) \equiv L^{d_1} \mod p$
- $\tilde{f}(L, x_2) \equiv L^{d_1} \mod p$– change of basis
- Lift $(L, x_2) \mapsto (pL, x_2)$, followed by division by p^{d_1}.
- Consider $f(x_1, x_2) = x_1^2 + 2px_1x_2 + p^2x_2$– contiguous d_1-power chains.

Roots of bivariates: Structure of contiguous d_1-powers

- $f(x_1, x_2) \equiv L^{d_1} \mod p$
- $\tilde{f}(L, x_2) \equiv L^{d_1} \mod p$– change of basis
- Lift $(L, x_2) \mapsto (pL, x_2)$, followed by division by p^{d_1}.
- Consider $f(x_1, x_2) = x_1^2 + 2px_1x_2 + p^2x_2$– contiguous d_1-power chains.
- How many possibilities?

Roots of bivariates: Structure of contiguous d_1-powers

- Basis change to (L, x_2).

Roots of bivariates: Structure of contiguous d_1-powers

- Basis change to (L, x_2).
- $f(L, x_2) =: L^{d_1} + p \cdot L^{d_1-1} \cdot u_1(x_2) + p \cdot L^{d_1-2} \cdot u_2(x_2) + \cdots + p \cdot u_{d_1}(x_2)$.

Roots of bivariates: Structure of contiguous d_1-powers

- Basis change to (L, x_2).
- $f(L, x_2) \ =: \ L^{d_1} + p \cdot L^{d_1 - 1} \cdot u_1(x_2) + p \cdot L^{d_1 - 2} \cdot u_2(x_2) + \cdots + p \cdot u_{d_1}(x_2)$.
- Length $= 2$ chain $\implies$ effective polynomial $(L + u_1(x_2))^{d_1}$ after lifting
  $\implies u_j(x_2) \ \equiv \ p^{j-1} \binom{d_1}{j} \cdot (u_1(x_2)/d_1)^j \bmod p^j$.

Roots of bivariates: Structure of contiguous d_1-powers

- Basis change to (L, x_2).
- $f(L, x_2) =: L^{d_1} + p \cdot L^{d_1 - 1} \cdot u_1(x_2) + p \cdot L^{d_1 - 2} \cdot u_2(x_2) + \cdots + p \cdot u_{d_1}(x_2)$.
- Length $= 2$ chain $\implies$ effective polynomial $(L + u_1(x_2))^{d_1}$ after lifting
  $\implies u_j(x_2) \equiv p^{j-1}\binom{d_1}{j} \cdot (u_1(x_2)/d_1)^j \mod p^j$.
- Use [BLQ13] to solve the number of possibilities for length 2 contiguous chains.

Roots of bivariates: Structure of contiguous d_1-powers

- Basis change to (L, x_2).
- $f(L, x_2) =: L^{d_1} + p \cdot L^{d_1 - 1} \cdot u_1(x_2) + p \cdot L^{d_1 - 2} \cdot u_2(x_2) + \cdots + p \cdot u_{d_1}(x_2)$.
- Length $= 2$ chain $\implies$ effective polynomial $(L + u_1(x_2))^{d_1}$ after lifting
  $\implies u_j(x_2) \equiv p^{j-1}\binom{d_1}{j} \cdot (u_1(x_2)/d_1)^j \bmod p^j$.
- Use [BLQ13] to solve the number of possibilities for length 2 contiguous chains.
- d_1-many representatives.(What about $*$ part?)

Roots of bivariates: Structure of contiguous d_1-powers

- Basis change to (L, x_2).
- $f(L, x_2) =: L^{d_1} + p \cdot L^{d_1-1} \cdot u_1(x_2) + p \cdot L^{d_1-2} \cdot u_2(x_2) + \cdots + p \cdot u_{d_1}(x_2)$.
- Length $= 2$ chain $\implies$ effective polynomial $(L + u_1(x_2))^{d_1}$ after lifting
  $\implies u_j(x_2) \equiv p^{j-1}\binom{d_1}{j} \cdot (u_1(x_2)/d_1)^j \mod p^j$.
- Use [BLQ13] to solve the number of possibilities for length 2 contiguous chains.
- d_1-many representatives.(What about $*$ part?)
- Representative roots might lead to degree increase, e.g. $L^{d_1} + p^{d_1}x^{d_1+1}$:

Roots of bivariates: Structure of contiguous d_1-powers

- Basis change to (L, x_2).
- $f(L, x_2) \; =: \; L^{d_1} + p \cdot L^{d_1 - 1} \cdot u_1(x_2) + p \cdot L^{d_1 - 2} \cdot u_2(x_2) + \cdots + p \cdot u_{d_1}(x_2)$.
- Length $= 2$ chain $\implies$ effective polynomial $(L + u_1(x_2))^{d_1}$ after lifting
  $\implies \; u_j(x_2) \; \equiv \; p^{j-1} \binom{d_1}{j} \cdot (u_1(x_2)/d_1)^j \mod p^j$.
- Use [BLQ13] to solve the number of possibilities for length 2 contiguous chains.
- d_1-many representatives.(What about $*$ part?)
- Representative roots might lead to degree increase, e.g. $L^{d_1} + p^{d_1} x^{d_1+1}$:
  - Reduce precision of $*$ by 1,

Roots of bivariates: Structure of contiguous d_1-powers

- Basis change to (L, x_2).
- $f(L, x_2) =: L^{d_1} + p \cdot L^{d_1-1} \cdot u_1(x_2) + p \cdot L^{d_1-2} \cdot u_2(x_2) + \cdots + p \cdot u_{d_1}(x_2)$.
- Length $= 2$ chain $\implies$ effective polynomial $(L + u_1(x_2))^{d_1}$ after lifting
  $\implies u_j(x_2) \equiv p^{j-1}\binom{d_1}{j} \cdot (u_1(x_2)/d_1)^j \mod p^j$.
- Use [BLQ13] to solve the number of possibilities for length 2 contiguous chains.
- d_1-many representatives.(What about $*$ part?)
- Representative roots might lead to degree increase, e.g. $L^{d_1} + p^{d_1}x^{d_1+1}$:
  - Reduce precision of $*$ by 1,
  - $*$ replaced by $a + p*$, $a \in \{0, \ldots, p-1\}$.

Roots of bivariates: Structure of contiguous d_1-powers

- Longer chains: more equations!

Roots of bivariates: Structure of contiguous d_1-powers

- Longer chains: more equations!
- Form equations on $L + u_1(x_2)/d_1$.

Roots of bivariates: Structure of contiguous d_1-powers

- Longer chains: more equations!
- Form equations on $L + u_1(x_2)/d_1$.
- Loop over contiguous val-mult.$= d_1$ chains:

Roots of bivariates: Structure of contiguous d_1-powers

- Longer chains: more equations!
- Form equations on $L + u_1(x_2)/d_1$.
- Loop over contiguous val-mult.$= d_1$ chains:
  - d_1-power contiguous i_1 length,

Roots of bivariates: Structure of contiguous d_1-powers

- Longer chains: more equations!
- Form equations on $L + u_1(x_2)/d_1$.
- Loop over contiguous val-mult.$= d_1$ chains:
  - d_1-power contiguous i_1 length,
  - d_1-nonpower form contiguous i_2 length,

Roots of bivariates: Structure of contiguous d_1-powers

- Longer chains: more equations!
- Form equations on $L + u_1(x_2)/d_1$.
- Loop over contiguous val-mult.$= d_1$ chains:
  - d_1-power contiguous i_1 length,
  - d_1-nonpower form contiguous i_2 length,
  - $i_1 + i_2 \leq k/d_1$.

Roots of bivariates: Algorithm

1. If $d_1 = 1$, return linear representative roots $(*, c(*))$.

Roots of bivariates: Algorithm

1. If $d_1 = 1$, return linear representative roots $(*, c(*))$.
2. If $k = 0$, return $(*_1, *_2)$.

Roots of bivariates: Algorithm

1. If $d_1 = 1$, return linear representative roots $(*, c(*))$.
2. If $k = 0$, return $(*_1, *_2)$.
3. For val-mult.$< d_1$, recursively continue down the tree.

Roots of bivariates: Algorithm

1. If $d_1 = 1$, return linear representative roots $(*, c(*))$.
2. If $k = 0$, return $(*_1, *_2)$.
3. For val-mult.$< d_1$, recursively continue down the tree.
4. For each $i_1, i_2 \leq k/d_1$,

Roots of bivariates: Algorithm

1. If $d_1 = 1$, return linear representative roots $(*, c(*))$.
2. If $k = 0$, return $(*_1, *_2)$.
3. For val-mult.$< d_1$, recursively continue down the tree.
4. For each $i_1, i_2 \leq k/d_1$,
    1. Consider contiguous i_1 length d_1-power chains followed by i_2 length d_1-nonpower form chains.

Roots of bivariates: Algorithm

1. If $d_1 = 1$, return linear representative roots $(*, c(*))$.
2. If $k = 0$, return $(*_1, *_2)$.
3. For val-mult.$< d_1$, recursively continue down the tree.
4. For each $i_1, i_2 \leq k/d_1$,
    1. Consider contiguous i_1 length d_1-power chains followed by i_2 length d_1-nonpower form chains.
    2. Recursively continue on degree reducing branches from these nodes.

Roots of bivariates: Algorithm

1. If $d_1 = 1$, return linear representative roots $(*, c(*))$.
2. If $k = 0$, return $(*_1, *_2)$.
3. For val-mult.$< d_1$, recursively continue down the tree.
4. For each $i_1, i_2 \leq k/d_1$,
    1. Consider contiguous i_1 length d_1-power chains followed by i_2 length d_1-nonpower form chains.
    2. Recursively continue on degree reducing branches from these nodes.

Roots of bivariates: Algorithm

1. If $d_1 = 1$, return linear representative roots $(*, c(*))$.
2. If $k = 0$, return $(*_1, *_2)$.
3. For val-mult.$< d_1$, recursively continue down the tree.
4. For each $i_1, i_2 \leq k/d_1$,
    1. Consider contiguous i_1 length d_1-power chains followed by i_2 length d_1-nonpower form chains.
    2. Recursively continue on degree reducing branches from these nodes.

Time Complexity: poly$((k + p + d)^d)$.

Roots of bivariates: $\mathbb{Z}_p$ roots

- [DS20] gave $k_0 = O(d^3 \log M)$, roots in $\mathbb{Z}/p^{k_0}\mathbb{Z} \iff$ roots in $\mathbb{Z}_p$.

Roots of bivariates: $\mathbb{Z}_p$ roots

- [DS20] gave $k_0 = O(d^3 \log M)$, roots in $\mathbb{Z}/p^{k_0}\mathbb{Z} \iff$ roots in $\mathbb{Z}_p$.
- Bivariates: linear-representative roots lift to $\mathbb{Z}_p$.

Roots of bivariates: $\mathbb{Z}_p$ roots

- [DS20] gave $k_0 = O(d^3 \log M)$, roots in $\mathbb{Z}/p^{k_0}\mathbb{Z} \iff$ roots in $\mathbb{Z}_p$.
- Bivariates: linear-representative roots lift to $\mathbb{Z}_p$.
- $\mathbb{Z}_p$ roots of resultant w.r.t. x_2.

Roots of bivariates: $\mathbb{Z}_p$ roots

- [DS20] gave $k_0 = O(d^3 \log M)$, roots in $\mathbb{Z}/p^{k_0}\mathbb{Z} \iff$ roots in $\mathbb{Z}_p$.
- Bivariates: linear-representative roots lift to $\mathbb{Z}_p$.
- $\mathbb{Z}_p$ roots of resultant w.r.t. x_2.
- $k_0 = O(d^{10} \log M)$.

Roots of bivariates: $\mathbb{Z}_p$ roots

- [DS20] gave $k_0 = O(d^3 \log M)$, roots in $\mathbb{Z}/p^{k_0}\mathbb{Z} \iff$ roots in $\mathbb{Z}_p$.
- Bivariates: linear-representative roots lift to $\mathbb{Z}_p$.
- $\mathbb{Z}_p$ roots of resultant w.r.t. x_2.
- $k_0 = O(d^{10} \log M)$.
- Linear representatives mod p^{k_0} as $\mathbf{a} \longrightarrow$ linear representatives $p^v\mathbf{a}$.

Roots of bivariates: Igusa's local zeta function

- $P(t) = \sum_{k=0}^{\infty} N_k(f) p^{-t} t^k$.

Roots of bivariates: Igusa's local zeta function

- $P(t) = \sum_{k=0}^{\infty} N_k(f) p^{-t} t^k$.
- Counting roots mod $p^{k_0} \implies$ counting roots mod p^k $\forall k$.

Roots of bivariates: Igusa's local zeta function

- $P(t) = \sum_{k=0}^{\infty} N_k(f) p^{-t} t^k$.
- Counting roots mod $p^{k_0} \implies$ counting roots mod $p^k\ \forall k$.
- Linear representative roots $\implies$ rational form!

Roots of bivariates: System of polynomial equations

- $f_1(x_1, x_2), \ldots, f_m(x_1, x_2)$.

Roots of bivariates: System of polynomial equations

- $f_1(x_1, x_2), \ldots, f_m(x_1, x_2)$.
- Isomorphic trees: individual trees in parallel.

Roots of bivariates: System of polynomial equations

- $f_1(x_1, x_2), \ldots, f_m(x_1, x_2)$.
- Isomorphic trees: individual trees in parallel.
- Similar algorithm till: all effective polynomials are linear forms.

Roots of bivariates: System of polynomial equations

- $f_1(x_1, x_2), \ldots, f_m(x_1, x_2)$.
- Isomorphic trees: individual trees in parallel.
- Similar algorithm till: all effective polynomials are linear forms.
- $L \equiv pg_1(L, x_2) \bmod p^k$; $L \equiv pg_2(L, x_2) \bmod p^k$; $\ldots$; $L \equiv pg_m(L, x_2) \bmod p^k$.

Roots of bivariates: System of polynomial equations

- $f_1(x_1, x_2), \ldots, f_m(x_1, x_2)$.
- Isomorphic trees: individual trees in parallel.
- Similar algorithm till: all effective polynomials are linear forms.
- $L \equiv pg_1(L, x_2) \bmod p^k$; $L \equiv pg_2(L, x_2) \bmod p^k$; $\ldots$; $L \equiv pg_m(L, x_2) \bmod p^k$.
- $L \mapsto pL$,

Roots of bivariates: System of polynomial equations

- $f_1(x_1, x_2), \ldots, f_m(x_1, x_2)$.
- Isomorphic trees: individual trees in parallel.
- Similar algorithm till: all effective polynomials are linear forms.
- $L \equiv pg_1(L, x_2) \bmod p^k$; $L \equiv pg_2(L, x_2) \bmod p^k$; $\ldots$; $L \equiv pg_m(L, x_2) \bmod p^k$.
- $L \mapsto pL$,
- $L \equiv g_1(pL, x_2) \bmod p^{k-1}$; $0 \equiv \tilde{g}_2(pL, x_2) \bmod p^{k-1}$; $\ldots$; $0 \equiv \tilde{g}_m(pL, x_2) \bmod p^{k-1}$.

Roots of bivariates: System of polynomial equations

- $f_1(x_1, x_2), \dots, f_m(x_1, x_2)$.
- Isomorphic trees: individual trees in parallel.
- Similar algorithm till: all effective polynomials are linear forms.
- $L \equiv pg_1(L, x_2) \bmod p^k$; $L \equiv pg_2(L, x_2) \bmod p^k$; $\dots$; $L \equiv pg_m(L, x_2) \bmod p^k$.
- $L \mapsto pL$,
- $L \equiv g_1(pL, x_2) \bmod p^{k-1}$; $0 \equiv \tilde{g}_2(pL, x_2) \bmod p^{k-1}$; $\dots$; $0 \equiv \tilde{g}_m(pL, x_2) \bmod p^{k-1}$.
- $L \equiv g_1(pL, x_2) \bmod p^{k-1}$; $0 \equiv \tilde{g}_2(pL, x_2) \bmod p^{k-1}$; $\dots$; $0 \equiv \tilde{g}_m(pL, x_2) \bmod p^{k-1}$.

Roots of bivariates: System of polynomial equations

- $f_1(x_1, x_2), \ldots, f_m(x_1, x_2)$.
- Isomorphic trees: individual trees in parallel.
- Similar algorithm till: all effective polynomials are linear forms.
- $L \equiv pg_1(L, x_2) \bmod p^k$; $L \equiv pg_2(L, x_2) \bmod p^k$; $\ldots$; $L \equiv pg_m(L, x_2) \bmod p^k$.
- $L \mapsto pL$,
- $L \equiv g_1(pL, x_2) \bmod p^{k-1}$; $0 \equiv \tilde{g}_2(pL, x_2) \bmod p^{k-1}$; $\ldots$; $0 \equiv \tilde{g}_m(pL, x_2) \bmod p^{k-1}$.
- $L \equiv g_1(pL, x_2) \bmod p^{k-1}$; $0 \equiv \tilde{g}_2(pL, x_2) \bmod p^{k-1}$; $\ldots$; $0 \equiv \tilde{g}_m(pL, x_2) \bmod p^{k-1}$.
- Find x_2 [BLQ13], find L.

Roots of n-variates

- Degree-reduction idea.

Roots of n-variates

- Degree-reduction idea.
- 3-variate:

Roots of n-variates

- Degree-reduction idea.
- 3-variate:
  - Rank 0 val-mult. $= d_1$ root: $\langle x_1 - a_1, x_2 - a_2, x_3 - a_3 \rangle^{d_1}$.

Roots of n-variates

- Degree-reduction idea.
- 3-variate:
  - Rank 0 val-mult. $= d_1$ root: $\langle x_1 - a_1, x_2 - a_2, x_3 - a_3 \rangle^{d_1}$.
  - Rank 1 val-mult.$= d_1$ root: $\langle a_1 x_2 - a_2 x_1, a_1 x_3 - a_3 x_1 \rangle^{d_1}$– root finding of univariates.

Roots of n-variates

- Degree-reduction idea.
- 3-variate:
  - Rank 0 val-mult. $= d_1$ root: $\langle x_1 - a_1, x_2 - a_2, x_3 - a_3 \rangle^{d_1}$.
  - Rank 1 val-mult.$= d_1$ root: $\langle a_1x_2 - a_2x_1, a_1x_3 - a_3x_1 \rangle^{d_1}$– root finding of univariates.
  - Rank 2 val-mult.$= d_1$ root: $\langle a_1x_2 - a_2x_1 \rangle^{d_1}$– root finding of bivariates.

Roots of n-variates

- Degree-reduction idea.
- 3-variate:
  - Rank 0 val-mult. $= d_1$ root: $\langle x_1 - a_1, x_2 - a_2, x_3 - a_3\rangle^{d_1}$.
  - Rank 1 val-mult.$= d_1$ root: $\langle a_1x_2 - a_2x_1, a_1x_3 - a_3x_1\rangle^{d_1}$– root finding of univariates.
  - Rank 2 val-mult.$= d_1$ root: $\langle a_1x_2 - a_2x_1\rangle^{d_1}$– root finding of bivariates.

Roots of n-variates

- Degree-reduction idea.
- 3-variate:
  - Rank 0 val-mult. $= d_1$ root: $\langle x_1 - a_1, x_2 - a_2, x_3 - a_3 \rangle^{d_1}$.
  - Rank 1 val-mult.$= d_1$ root: $\langle a_1 x_2 - a_2 x_1, a_1 x_3 - a_3 x_1 \rangle^{d_1}$– root finding of univariates.
  - Rank 2 val-mult.$= d_1$ root: $\langle a_1 x_2 - a_2 x_1 \rangle^{d_1}$– root finding of bivariates.

Time complexity- poly$((m + d + p)^{(2d(n-1))^{n-1}})$.

Part II

Root of $f_1(\mathbf{x}), \ldots, f_m(\mathbf{x})$ in randomized $\mathrm{poly}(m, d^{c_{nk}}, \log p)$, where $c_{nk} \leq (nk)^{O((nk)^2)}$.

Factoring modular polynomials via Hilbert's Nullstellensatz

Importance of the problem

- Hilbert's Nullstellensatz: NP-complete.

Factoring modular polynomials via Hilbert's Nullstellensatz

Importance of the problem

- Hilbert's Nullstellensatz: NP-complete.
- Connection between $\mathbb{F}_p$ and $\mathbb{Z}_p$.

Factoring modular polynomials via Hilbert's Nullstellensatz

Importance of the problem

- Hilbert's Nullstellensatz: NP-complete.
- Connection between $\mathbb{F}_p$ and $\mathbb{Z}_p$.
- Factorization modulo p^k.

Factoring modular polynomials via Hilbert's Nullstellensatz

Importance of the problem

- Hilbert's Nullstellensatz: NP-complete.
- Connection between $\mathbb{F}_p$ and $\mathbb{Z}_p$.
- Factorization modulo p^k.

Previous work

- Roots of polynomial system over different fields: [HW99, BKW19, LPT$^+$17, Kay05, CS22].
- Factoring over fields:
  - Finite fields: [Ber67, CZ81, Kal92, KU11, vzGP01].
  - p-adics: [CG00, Chi87, Chi94, GNP12].
- Famous Hensel's lifting: [Hen18].
- Factoring achieved only for small k's: [DMS21, Sir17, vzGH96, vzGH98].

Hilbert's Nullstellensatz $\bmod p^k$

Main ideas

- Store local roots in ideals (~~enumeration~~)
- Virtual roots at i-th step $\mathbf{y}$.
- **Lifting:** $f_j(\mathbf{x}) := p^{-v_j} f_j(\mathbf{y} + p\mathbf{x}) \ \forall j \in [m]$.
- Virtual roots such that $f_j(\mathbf{y}) \equiv 0 \bmod p$.
- $\mathtt{I} \leftarrow \mathtt{I} + \langle f_j(\mathbf{y}) \bmod p \rangle$.

Hilbert's Nullstellensatz $\bmod p^k$

Main ideas

- Store local roots in ideals (~~enumeration~~)

Hilbert's Nullstellensatz $\bmod p^k$

Main ideas

- Store local roots in ideals (~~enumeration~~)
- Virtual roots at i-th step $\mathbf{y}$.

Hilbert's Nullstellensatz $\bmod p^k$

Main ideas

- Store local roots in ideals (~~enumeration~~)
- Virtual roots at i-th step $\mathbf{y}$.
- **Lifting:** $f_j(\mathbf{x}) := p^{-v_j} f_j(\mathbf{y} + p\mathbf{x}) \; \forall j \in [m]$.

Hilbert's Nullstellensatz $\bmod p^k$

Main ideas

- Store local roots in ideals (~~enumeration~~)
- Virtual roots at i-th step $\mathbf{y}$.
- **Lifting:** $f_j(\mathbf{x}) := p^{-v_j} f_j(\mathbf{y} + p\mathbf{x})\ \forall j \in [m]$.
- Virtual roots such that $f_j(\mathbf{y}) \equiv 0 \bmod p$.

Hilbert's Nullstellensatz $\bmod p^k$

Main ideas

- Store local roots in ideals (~~enumeration~~)
- Virtual roots at i-th step $\mathbf{y}$.
- **Lifting:** $f_j(\mathbf{x}) := p^{-v_j} f_j(\mathbf{y} + p\mathbf{x}) \ \forall j \in [m]$.
- Virtual roots such that $f_j(\mathbf{y}) \equiv 0 \bmod p$.
- $\hat{\mathrm{I}} \leftarrow \hat{\mathrm{I}} + \langle f_j(\mathbf{y}) \bmod p \rangle$.

Hilbert's Nullstellensatz $\bmod p^k$

Main ideas

- Store local roots in ideals (~~enumeration~~)
- Virtual roots at i-th step $\mathbf{y}$.
- **Lifting:** $f_j(\mathbf{x}) := p^{-v_j} f_j(\mathbf{y} + p\mathbf{x})\ \forall j \in [m]$.
- Virtual roots such that $f_j(\mathbf{y}) \equiv 0 \bmod p$.
- $\hat{\mathrm{I}} \leftarrow \hat{\mathrm{I}} + \langle f_j(\mathbf{y}) \bmod p \rangle$.

Hilbert's Nullstellensatz $\bmod p^k$

Main ideas

- Store local roots in ideals (~~enumeration~~)
- Virtual roots at i-th step $\mathbf{y}$.
- **Lifting:** $f_j(\mathbf{x}) := p^{-v_j} f_j(\mathbf{y} + p\mathbf{x}) \; \forall j \in [m]$.
- Virtual roots such that $f_j(\mathbf{y}) \equiv 0 \bmod p$.
- $\hat{\mathrm{I}} \leftarrow \hat{\mathrm{I}} + \langle f_j(\mathbf{y}) \bmod p \rangle$.
- Division by $p \longrightarrow p$-adics.

Main ideas

- Store local roots in ideals (~~enumeration~~)
- Virtual roots at i-th step $\mathbf{y}$.
- **Lifting:** $f_j(\mathbf{x}) := p^{-v_j} f_j(\mathbf{y} + p\mathbf{x})\ \forall j \in [m]$.
- Virtual roots such that $f_j(\mathbf{y}) \equiv 0 \bmod p$.
- $\hat{\mathrm{I}} \leftarrow \hat{\mathrm{I}} + \langle f_j(\mathbf{y}) \bmod p \rangle$.
- Division by $p \longrightarrow p$-adics.
- **Goal:** Exhaust k, find solution of $\hat{\mathrm{I}}$.

Hilbert's Nullstellensatz $\bmod p^k$

Main ideas

- Store local roots in ideals (~~enumeration~~)
- Virtual roots at i-th step $\mathbf{y}$.
- **Lifting:** $f_j(\mathbf{x}) := p^{-v_j} f_j(\mathbf{y} + p\mathbf{x})\ \forall j \in [m]$.
- Virtual roots such that $f_j(\mathbf{y}) \equiv 0 \bmod p$.
- $\hat{\mathrm{I}} \leftarrow \hat{\mathrm{I}} + \langle f_j(\mathbf{y}) \bmod p \rangle$.
- Division by $p \longrightarrow p$-adics.
- **Goal:** Exhaust k, find solution of $\hat{\mathrm{I}}$.
- $\mathbb{Z}/p^k\mathbb{Z} \to \mathbb{F}_p \to \mathbb{Z}_p \to \mathbb{Z}/p^k\mathbb{Z}$.

Remark

- p-adic roots not unique.
- $a + pb$, replace $a := a - pt$, $b := b + t$.

Birationally equivalent hypersurface

- $\mathbf{V}(\mathtt{I}) \longleftrightarrow \mathtt{H} = \langle h \rangle$.

Birationally equivalent hypersurface

- $\mathbf{V}(\mathtt{I}) \longleftrightarrow \mathtt{H} = \langle h \rangle$.
- Primitive element theorem.

Hilbert's Nullstellensatz $\bmod p^k$

Birationally equivalent hypersurface

- $\mathbf{V}(\mathtt{I}) \longleftrightarrow \mathtt{H} = \langle h \rangle$.
- Primitive element theorem.
- $\phi_1 : \mathtt{H} \to \mathbf{V}(\mathtt{I})$; $\phi_2 : \mathbf{V}(\mathtt{I}) \to \mathtt{H}$.

Hilbert's Nullstellensatz $\bmod p^k$

Birationally equivalent hypersurface

- $\mathbf{V}(\mathtt{I}) \longleftrightarrow \mathtt{H} = \langle h \rangle$.
- Primitive element theorem.
- $\phi_1 : \mathtt{H} \to \mathbf{V}(\mathtt{I})$; $\phi_2 : \mathbf{V}(\mathtt{I}) \to \mathtt{H}$.
- 'Most' points of $\mathbf{V}(\mathtt{I})$ are mapped.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

Lifting $\mathbb{F}_p$ roots to $\mathbb{Z}_p$

- I vanishes over $\mathbb{F}_p$, need to vanish over $\mathbb{Z}_p$.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

Lifting $\mathbb{F}_p$ roots to $\mathbb{Z}_p$

- I vanishes over $\mathbb{F}_p$, need to vanish over $\mathbb{Z}_p$.
- Hensel's lifting of roots.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

Lifting $\mathbb{F}_p$ roots to $\mathbb{Z}_p$

- $\mathtt{I}$ vanishes over $\mathbb{F}_p$, need to vanish over $\mathbb{Z}_p$.
- Hensel's lifting of roots.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

Lifting $\mathbb{F}_p$ roots to $\mathbb{Z}_p$

- $\mathtt{I}$ vanishes over $\mathbb{F}_p$, need to vanish over $\mathbb{Z}_p$.
- Hensel's lifting of roots. (One polynomial!)
- Find corresponding $\mathtt{H}$.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

Lifting $\mathbb{F}_p$ roots to $\mathbb{Z}_p$

- $\mathtt{I}$ vanishes over $\mathbb{F}_p$, need to vanish over $\mathbb{Z}_p$.
- Hensel's lifting of roots. (One polynomial!)
- Find corresponding $\mathtt{H}$.
- Lift to $\mathbb{Z}_p$.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

Lifting $\mathbb{F}_p$ roots to $\mathbb{Z}_p$

- $\mathtt{I}$ vanishes over $\mathbb{F}_p$, need to vanish over $\mathbb{Z}_p$.
- Hensel's lifting of roots. (One polynomial!)
- Find corresponding $\mathtt{H}$.
- Lift to $\mathbb{Z}_p$.
- Need for irreducible components $\mathtt{I} \rightarrow \mathtt{H} \rightarrow \mathtt{C}$.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

$$
\begin{tikzcd}
\widehat{\mathbb{G}}(\ell_1,\ldots,\ell_r)[\ell_{r+1},\ldots,\ell_N]/\hat{\mathtt{C}} \arrow[d, "\bmod p"'] & & \widehat{\mathbb{G}}(\ell_1,\ldots,\ell_r)[Y]/\langle\hat{h}\rangle \arrow[ll, "\hat{\psi}_2"'] \arrow[d, "\bmod p"] \\
\mathbb{F}_q(\ell_1,\ldots,\ell_r)[\ell_{r+1},\ldots,\ell_N]/\mathtt{C} \arrow[rr, shift left, "\psi_1"] & & \mathbb{F}_q(\ell_1,\ldots,\ell_r)[Y]/\langle h\rangle \arrow[ll, shift left, "\psi_2"]
\end{tikzcd}
$$

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

$$\boxed{\mathtt{C} \longrightarrow \hat{\mathtt{C}}}$$

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

$$\boxed{\mathtt{C} \longrightarrow \hat{\mathtt{C}}}$$

- Compute Gröbner basis of $\mathtt{C}$.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

$$\boxed{\mathtt{C} \longrightarrow \hat{\mathtt{C}}}$$

- Compute Gröbner basis of $\mathtt{C}$.
- Integral lift to $\mathbb{Z}_p$: $\hat{\mathtt{C}}$.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots

$$\boxed{\texttt{C} \longrightarrow \hat{\texttt{C}}}$$

- Compute Gröbner basis of $\texttt{C}$.
- Integral lift to $\mathbb{Z}_p$: $\hat{\texttt{C}}$.
- Roots nicely commute.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots: Absolutely irreducible components

- Hensel's lifting of roots (non-singular roots).

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots: Absolutely irreducible components

- Hensel's lifting of roots (non-singular roots).
- Roots that get lost in commutative diagram.

$\mathbb{F}_p$ coordinates to $\mathbb{Z}_p$ roots: Absolutely irreducible components

- Hensel's lifting of roots (non-singular roots).
- Roots that get lost in commutative diagram.
- Lesser dimension absolutely irreducible components.

Absolutely irreducible components: Branching

- Branches corresponding to each component.

Absolutely irreducible components: Branching

- Branches corresponding to each component.
- Variables $\mathbf{y}_0, \ldots, \mathbf{y}_\ell$.

Absolutely irreducible components: Branching

- Branches corresponding to each component.
- Variables $\mathbf{y}_0, \ldots, \mathbf{y}_\ell$.
- If $\mathtt{C} \cap \mathbb{F}_q[\mathbf{y}_0, \ldots, \mathbf{y}_{\ell-1}] \neq \mathtt{I} \cap \mathbb{F}_q[\mathbf{y}_0, \ldots, \mathbf{y}_{\ell-1}]$,

Absolutely irreducible components: Branching

- Branches corresponding to each component.
- Variables $\mathbf{y}_0, \ldots, \mathbf{y}_\ell$.
- If $\mathtt{C} \cap \mathbb{F}_q[\mathbf{y}_0, \ldots, \mathbf{y}_{\ell-1}] \neq \mathtt{I} \cap \mathbb{F}_q[\mathbf{y}_0, \ldots, \mathbf{y}_{\ell-1}]$,
    - Find min $s \leq \ell - 1$ s.t. $\mathtt{C} \leftarrow \mathtt{C} \cap \mathbb{F}_q[\mathbf{y}_0, \ldots, \mathbf{y}_s] \supsetneq \mathtt{I} \cap \mathbb{F}_q[\mathbf{y}_0, \ldots, \mathbf{y}_s]$.

Absolutely irreducible components: Branching

- Branches corresponding to each component.
- Variables $\mathbf{y}_0, \ldots, \mathbf{y}_\ell$.
- If $\mathtt{C} \cap \mathbb{F}_q[\mathbf{y}_0, \ldots, \mathbf{y}_{\ell-1}] \neq \mathtt{I} \cap \mathbb{F}_q[\mathbf{y}_0, \ldots, \mathbf{y}_{\ell-1}]$,
  - Find min $s \leq \ell - 1$ s.t. $\mathtt{C} \leftarrow \mathtt{C} \cap \mathbb{F}_q[\mathbf{y}_0, \ldots, \mathbf{y}_s] \supsetneq \mathtt{I} \cap \mathbb{F}_q[\mathbf{y}_0, \ldots, \mathbf{y}_s]$.
  - Backtracking.

Hilbert's Nullstellensatz mod p^k: Algorithm

1. If k exhausted, return ideals.

Hilbert's Nullstellensatz $\bmod p^k$: Algorithm

1. If k exhausted, return ideals.
2. $\mathtt{I} \leftarrow \langle f_1(\mathbf{y}_\ell), \ldots, f_m(\mathbf{y}_\ell)\rangle + \hat{\mathtt{I}} + \langle p \rangle$.

Hilbert's Nullstellensatz $\bmod p^k$: Algorithm

1. If k exhausted, return ideals.
2. $\mathtt{I} \leftarrow \langle f_1(\mathbf{y}_\ell), \ldots, f_m(\mathbf{y}_\ell)\rangle + \hat{\mathtt{I}} + \langle p\rangle$.
3. For each $\mathtt{C} \in \textsc{Abs\_Decomp}(\mathtt{I})$,

1. If k exhausted, return ideals.
2. $\mathtt{I} \leftarrow \langle f_1(\mathbf{y}_\ell), \ldots, f_m(\mathbf{y}_\ell)\rangle + \hat{\mathtt{I}} + \langle p \rangle$.
3. For each $\mathtt{C} \in \textsc{Abs\_Decomp}(\mathtt{I})$,
    1. $\tilde{f}_j(\mathbf{x}) := p^{-1} f_j(\mathbf{y}_\ell + p\mathbf{x}) \bmod \hat{\mathtt{C}}$

Hilbert's Nullstellensatz $\bmod\, p^k$: Algorithm

1. If k exhausted, return ideals.
2. $\mathtt{I} \leftarrow \langle f_1(\mathbf{y}_\ell), \ldots, f_m(\mathbf{y}_\ell)\rangle + \hat{\mathtt{I}} + \langle p\rangle$.
3. For each $\mathtt{C} \in \textsc{Abs\_Decomp}(\mathtt{I})$,
    1. $\tilde{f}_j(\mathbf{x}) := p^{-1} f_j(\mathbf{y}_\ell + p\mathbf{x}) \bmod \hat{\mathtt{C}}$

Hilbert's Nullstellensatz $\bmod p^k$: Algorithm

1. If k exhausted, return ideals.
2. $\mathtt{I} \leftarrow \langle f_1(\mathbf{y}_\ell), \ldots, f_m(\mathbf{y}_\ell)\rangle + \hat{\mathtt{I}} + \langle p\rangle$.
3. For each $\mathtt{C} \in \textsc{Abs\_Decomp}(\mathtt{I})$,
    1. $\tilde{f}_j(\mathbf{x}) := p^{-1} f_j(\mathbf{y}_\ell + p\mathbf{x}) \bmod \hat{\mathtt{C}}$ over $\mathbb{Z}_p$.
    2. Recursively return root on $\tilde{f}_j$'s.

- Ideals exactly capture *all* roots.

Hilbert's Nullstellensatz mod p^k: Algorithm

- Ideals exactly capture *all* roots.
- Find one [HW99].

Hilbert's Nullstellensatz $\bmod p^k$: Algorithm

- Ideals exactly capture *all* roots.
- Find one [HW99].
- Lift to $\mathbb{Z}_p$.

Hilbert's Nullstellensatz $\bmod p^k$: Algorithm

- Ideals exactly capture *all* roots.
- Find one [HW99].
- Lift to $\mathbb{Z}_p$.
- Size of tree $d^{(nk)^{O((nk)^2)}}$

Hilbert's Nullstellensatz $\bmod p^k$: Algorithm

- Ideals exactly capture *all* roots.
- Find one [HW99].
- Lift to $\mathbb{Z}_p$.
- Size of tree $d^{(nk)^{O((nk)^2)}}$
- Time complexity poly$(m, d^{c_{nk}}, \log p)$, where $c_{nk} \leq (nk)^{O((nk)^2)}$.

Application: Factoring $\bmod p^k$

- Hensel's lifting [Hen18] fails when $f(x) \equiv \varphi(x)^e \bmod p$.

Application: Factoring mod p^k

- Hensel's lifting [Hen18] fails when $f(x) \equiv \varphi(x)^e \bmod p$.
- [DMS21]: Factor $h(x) = \varphi(x)^a - py \longleftrightarrow$ Roots of $f(x)(\varphi^{a(k-1)} + \varphi^{a(k-2)}(py) + \cdots + (py)^{k-1}) \bmod \langle p^k, \varphi^{ak} \rangle$.

Application: Factoring $\bmod p^k$

- Hensel's lifting [Hen18] fails when $f(x) \equiv \varphi(x)^e \bmod p$.
- [DMS21]: Factor $h(x) = \varphi(x)^a - py \longleftrightarrow$ Roots of $f(x)(\varphi^{a(k-1)} + \varphi^{a(k-2)}(py) + \cdots + (py)^{k-1}) \bmod \langle p^k, \varphi^{ak} \rangle$.
- Reduced to root finding of system of polynomials over Galois rings.

Application: Factoring mod p^k

- Hensel's lifting [Hen18] fails when $f(x) \equiv \varphi(x)^e \bmod p$.
- [DMS21]: Factor $h(x) = \varphi(x)^a - py \longleftrightarrow$ Roots of $f(x)(\varphi^{a(k-1)} + \varphi^{a(k-2)}(py) + \cdots + (py)^{k-1}) \bmod \langle p^k, \varphi^{ak} \rangle$.
- Reduced to root finding of system of polynomials over Galois rings.
- Solved for constant a.

Questions?

Describing roots of multivariates:

- Small $\log p, d, n$.
- Iteratively finding each coordinate.
- Main ideas:
  - Degree reduction,
  - Reduction to $n - 1$ variates.
- Output:
  - Representative roots,
  - Linear-representative roots.

Hilbert's Nullstellensatz $\bmod p^k$

- Small n, k.
- Storing coordinates in ideals in $\mathbb{Z}_p$.
- Main ideas:
  - Reduction to $\mathbb{F}_p$ system solving,
  - Virtual roots in ideals.
- Output:
  - Ideals containing roots,
  - Coordinates given as $\mathbb{Z}_p$ points.

References I

Elwyn R Berlekamp.

Factoring polynomials over finite fields.

Bell System Technical Journal, 46(8):1853–1859, 1967.

Andreas Björklund, Petteri Kaski, and Ryan Williams.

Solving systems of polynomial equations over gf (2) by a parity-counting self-reduction.

In *46th International Colloquium on Automata, Languages, and Programming (ICALP), 2019, Patras, Greece*, volume 132 of *LIPIcs*, pages 26:1–26:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

References II

Jérémy Berthomieu, Grégoire Lecerf, and Guillaume Quintin.

Polynomial root finding over local rings and application to error correcting codes.

Applicable Algebra in Engineering, Communication and Computing, 24(6):413–443, 2013.

Sayak Chakrabarti, Ashish Dwivedi, and Nitin Saxena.

Factoring modular polynomials via Hilbert's Nullstellensatz.

Manuscript, 2022.

David G Cantor and Daniel M Gordon.

Factoring polynomials over p-adic fields.

In *International Algorithmic Number Theory Symposium*, pages 185–208. Springer, 2000.

References III

Qi Cheng, Shuhong Gao, J Maurice Rojas, and Daqing Wan.

Counting roots for polynomials modulo prime powers.

The Open Book Series (ANTS XIII), 2(1):191–205, 2019.

Alexander Leonidovich Chistov.

Efficient factorization of polynomials over local fields.

volume 293, pages 1073–1077. Russian Academy of Sciences, 1987.

Alexander L Chistov.

Algorithm of polynomial complexity for factoring polynomials over local fields.

Journal of mathematical sciences, 70(4):1912–1933, 1994.

References IV

Alexander L Chistov.

An effective algorithm for deciding solvability of a system of polynomial equations over p-adic integers.

Algebra i Analiz, 33(6):162–196, 2021.

Sayak Chakrabarti and Nitin Saxena.

Describing the roots of multivariates mod p^k and efficient computation of Igusa's local zeta function.

Manuscript, 2022.

David G Cantor and Hans Zassenhaus.

A new algorithm for factoring polynomials over finite fields.

Mathematics of Computation, 36(154):587–592, 1981.

Jan Denef.

The rationality of the poincaré series associated to the p-adic points on a variety.

Invent. math, 77(1):1–23, 1984.

Ashish Dwivedi, Rajat Mittal, and Nitin Saxena.

Counting Basic-Irreducible Factors Mod p^k in Deterministic Poly-Time and p-Adic Applications.

In Amir Shpilka, editor, *34th Computational Complexity Conference (CCC 2019)*, volume 137 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 15:1–15:29, 2019.

References VI

Ashish Dwivedi, Rajat Mittal, and Nitin Saxena.

Efficiently factoring polynomials modulo p^4.

Journal of Symbolic Computation, 104:805 – 823, 2021.

Preliminary version appeared in The 44th ACM International Symposium on Symbolic and Algebraic Computation (ISSAC) 2019.

Ashish Dwivedi and Nitin Saxena.

Computing Igusa's local zeta function of univariates in deterministic polynomial-time.

14*th Algorithmic Number Theory Symposium (ANTS XIV), Open Book Series*, 4(1):197–214, 2020.

References VII

Jordi Guàrdia, Enric Nart, and Sebastian Pauli.

Single-factor lifting and factorization of polynomials over local fields.

J. Symb. Comput., 47(11):1318–1346, November 2012.

Kurt Hensel.

Eine neue theorie der algebraischen zahlen.

Mathematische Zeitschrift, 2(3):433–452, Sep 1918.

M-D Huang and Y-C Wong.

Solvability of systems of polynomial congruences modulo a large prime.

computational complexity, 8(3):227–257, 1999.

Preliminary version appeared in The IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS) 1996.

References VIII

Erich Kaltofen.

Polynomial factorization 1987–1991.

In *Latin American Symposium on Theoretical Informatics*, pages 294–313. Springer, 1992.

Neeraj Kayal.

Solvability of a system of bivariate polynomial equations over a finite field.

In *International Colloquium on Automata, Languages, and Programming*, pages 551–562. Springer, 2005.

Leann Kopp, Natalie Randall, J Maurice Rojas, and Yuyu Zhu.

Randomized polynomial-time root counting in prime power rings.

Mathematics of Computation, 89(321):373–385, 2020.

References IX

Kiran S Kedlaya and Christopher Umans.

Fast polynomial factorization and modular composition.

SIAM Journal on Computing, 40(6):1767–1802, 2011.

Daniel Lokshtanov, Ramamohan Paturi, Suguru Tamaki, Ryan Williams, and Huacheng Yu.

Beating brute force for systems of polynomial equations over finite fields.

In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2190–2202. SIAM, 2017.

References X

Vincent Neiger, Johan Rosenkilde, and Éric Schost.

Fast computation of the roots of polynomials over the ring of power series.

In *Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation*, pages 349–356, 2017.

Peter N Panayi.

Computation of Leopoldt's P-adic regulator.

PhD thesis, University of East Anglia, Norwich, England, 1995.

Caleb Robelle, J Maurice Rojas, and Yuyu Zhu.

Sub-linear point counting for variable separated curves over prime power rings.

arXiv preprint arXiv:2102.01626, 2021.

References XI

Carlo Sircana.

Factorization of polynomials over $\mathbb{Z}/(p^n)$.

In *Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation*, pages 405–412. ACM, 2017.

Joachim von zur Gathen and Silke Hartlieb.

Factorization of polynomials modulo small prime powers.

Technical report, Paderborn Univ, 1996.

Joachim von zur Gathen and Silke Hartlieb.

Factoring modular polynomials.

Journal of Symbolic Computation, 26(5):583–606, 1998.

(Conference version in ISSAC'96).

References XII

Joachim von zur Gathen and Daniel Panario.

Factoring polynomials over finite fields: A survey.

Journal of Symbolic Computation, 31(1-2):3–17, 2001.

WA Zuniga-Galindo.

Computing igusa's local zeta functions of univariate polynomials, and linear feedback shift registers.

arXiv preprint cs/0309050, 2003.

Thank You!