FGLM algorithm over Tate algebras

Xavier Caruso[1] Tristan Vaccon[2] Thibaut Verron[3]

1. Université de Bordeaux, CNRS, Inria, Bordeaux, France
2. Université de Limoges, CNRS, XLIM, Limoges, France
3. Johannes Kepler University, Institute for Algebra, Linz, Austria

Seminar *Algebra and Discrete Mathematics*, 2021/04/15

Setting and definitions

Valued field, valuation ring

- Field with a valuation $\mathrm{val} : K \to \mathbb{Z} \cup \infty$ $\mathbb{Q}_p$ $k((X))$
- Integer ring $K^\circ = \{x : \mathrm{val}(x) \geq 0\}$ $\mathbb{Z}_p$ $k[[X]]$
- Uniformizer π s.t. $\pi K^\circ = \{x : \mathrm{val}(x) \geq 1\}$ p X

$\} \ \mathrm{val}(a) = 3$

$a = a_3\pi^3 + a_4\pi^4 + \cdots$

$b = b_{-3}\pi^{-3} + \cdots$

$\} \ \mathrm{val}(b) = -3$

Metric and topology

- "a is small" $\iff$ "$\mathrm{val}(a)$ is large"
- Non-archimedean metric: "small + small = small"
- $\mathbb{Q}_p$, $\mathbb{Z}_p$, $k((X))$, $k[[X]]$ are complete for that topology

Rigid geometry and Tate series

- "Algebraic geometry, analytic geometry" bridge for non-archimedean geometry
- Main object: Tate series

Tate series

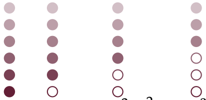
Definitions

$\mathbf{r} \in \mathbb{Q}^n$: convergence (log)-radii

- Tate algebra $K\{X_1, \ldots, X_n; r_1, \ldots, r_n\} = K\{\mathbf{X}; \mathbf{r}\}$
- Set of series $\sum_{\alpha \in \mathbb{N}^n} a_\alpha X_1^{\alpha_1} \cdots X_n^{\alpha_n}$ with $\operatorname{val}(a_\alpha) - \sum r_j \alpha_j \to \infty$
- *"Convergent for substitutions with* $\operatorname{val}(x_i) \geq -r_i$*"*
- smaller $r_i \iff$ smaller convergence radius $\iff$ larger algebra
- Convention: $r_i = \infty \to$ finitely many terms in X_i (polynomial)

Examples:

- Polynomials are Tate series for all radii (finite sums)



- $f = \sum_{i,j=0}^{\infty} \pi^i X^i = 1 + \pi X + \pi^2 X^2 + \pi^3 X^3 + \cdots$
  - $f \in K\{X\} = K\{X; 0\}$
  - $f \notin K\{X; 1\}$: for all terms, $\operatorname{val}(\pi^\alpha) - \alpha = 0 \not\to \infty$

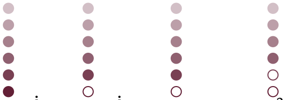
Gröbner bases over Tate algebras

Gröbner bases:

- Multi-purpose tool for ideal arithmetic in polynomial algebras
- Membership testing, elimination, intersection...
- Uses successive (terminating) reductions
- Requires the definition of an ordering on terms

Construction for Tate series

- Term order considering terms according to the valuation of their coefficient
- First compare $\mathrm{val}(a_\alpha) - \sum r_j \alpha_j$, break ties with a monomial order



$$\cdots > 1\,\mathbf{X}^{\mathbf{i}_1} > \pi\,\mathbf{X}^{\mathbf{i}_2} > \pi \cdot 1 > \pi^2\,\mathbf{X}^{\mathbf{i}_3} > \cdots$$

- Convergent reductions (interrupted at the precision bound) instead of terminating ones
- Allows to use usual algorithms (Buchberger, F4) to compute Gröbner bases

Complexity bottleneck: reductions

Cost of reductions

- Not unusual with Gröbner bases
- Tate case: reductions are interrupted at the precision bound
- The cost grows badly with the precision
- **Question:** can we compute reductions in time quasi-linear in the precision?

Ideas for possible improvement:

- Avoid useless reductions to zero
- Speed-up interreductions
- Exploit overconvergence

Series converging faster, *i.e.*, living in a smaller Tate algebra
Ex: polynomials (log-radii ∞) seen as Tate series

Complexity bottleneck: reductions

Cost of reductions

- Not unusual with Gröbner bases
- Tate case: reductions are interrupted at the precision bound
- The cost grows badly with the precision
- **Question:** can we compute reductions in time quasi-linear in the precision?

Ideas for possible improvement:

- Avoid useless reductions to zero $\longrightarrow$ Signature algorithms [CVV 2020]
- Speed-up interreductions
- Exploit overconvergence

Series converging faster, *i.e.*, living in a smaller Tate algebra
Ex: polynomials (log-radii ∞) seen as Tate series

Complexity bottleneck: reductions

Cost of reductions

- Not unusual with Gröbner bases
- Tate case: reductions are interrupted at the precision bound
- The cost grows badly with the precision
- **Question:** can we compute reductions in time quasi-linear in the precision?

Ideas for possible improvement:

- Avoid useless reductions to zero $\longrightarrow$ Signature algorithms [CVV 2020]
- Speed-up interreductions
- Exploit overconvergence

$\rightarrow$ In dim. 0: with FGLM **[this work]**

Series converging faster, *i.e.*, living in a smaller Tate algebra
Ex: polynomials (log-radii ∞) seen as Tate series

Change of ordering and the FGLM algorithm

Change of ordering:

- Useful in the classical case for two-steps strategies
- For zero-dimensional ideals, can be done efficiently with the FGLM algorithm [Faugère, Gianni, Lazard, Mora 1993]

For Tate algebras:

- Change of monomial ordering
- But also change of term ordering and radius of convergence

Idea for overconvergence:

1. Compute a Gröbner basis in the smaller Tate algebra
2. Use change of ordering to restrict to the larger one

Characteristics of the FGLM algorithm

0-dimensional ideals:

- Variety = finitely many points
- Quotient $K[\mathbf{X}]/I$ has finite dimension as a vector space over K
- Given a Gröbner basis G, the staircase under G is
  $B = \{m \text{ monomial not divisible by any LT of } G\}$
- B is a K-basis of $K[\mathbf{X}]/I$

Outline of the algorithm:

In: G_1 a reduced Gröbner basis wrt an order $<_1$
$<_2$ a monomial order

Out: G_2 a reduced Gröbner basis wrt $<_2$

1. Compute the matrices of multiplication by $X_1, \ldots, X_n$ in the basis B_1 (computing B_1)
2. Convert them into the Gröbner basis G_2 (computing B_2)

Characteristics of the FGLM algorithm

0-dimensional ideals:

- Variety = finitely many points
- Quotient $K[\mathbf{X}]/I$ has finite dimension as a vector space over K
- Given a Gröbner basis G, the staircase under G is
  $B = \{m \text{ monomial not divisible by any LT of } G\}$
- B is a K-basis of $K[\mathbf{X}]/I$

Outline of the algorithm:

In: G_1 a reduced Gröbner basis wrt an order $<_1$
$<_2$ a monomial order

Out: G_2 a reduced Gröbner basis wrt $<_2$

1. Compute the matrices of multiplication by $X_1, \ldots, X_n$ in the basis B_1 (computing B_1)
2. Convert them into the Gröbner basis G_2 (computing B_2)

Complexity

- Degree δ of the ideal = size of B = number of solutions (with multiplicity)
- Complexity cubic (or subcubic) in δ

FGLM algorithm for Tate ideals

0-dimensional Tate ideals

- Same definition as in the polynomial case: $K\{\mathbf{X}\}/I$ has finite dimension
- B is a K-basis of $K\{\mathbf{X}\}/I$
- Any element of $K\{\mathbf{X}\}/I$ can be represented as a **polynomial**

FGLM algorithm for Tate ideals

0-dimensional Tate ideals

- Same definition as in the polynomial case: $K\{\mathbf{X}\}/I$ has finite dimension
- B is a K-basis of $K\{\mathbf{X}\}/I$
- Any element of $K\{\mathbf{X}\}/I$ can be represented as a **polynomial**

Outline of the algorithm

In: G_1 a reduced Gröbner basis in $K\{\mathbf{X}; \mathbf{r}\}$ wrt an order $<_1$

$<_2$ a monomial order

$\mathbf{u} \leq \mathbf{r}$ a system of log-radii

Out: G_2 a reduced Gröbner basis in $K\{\mathbf{X}; \mathbf{u}\}$ wrt $<_2$

1. Compute the matrices of multiplication by $X_1, \ldots, X_n$ in the basis $B_{1,\mathbf{r}}$
2. Convert them into matrices in the basis $B_{1,\mathbf{u}}$ (computing $B_{1,\mathbf{u}}$)
3. Convert them into the Gröbner basis G_2

FGLM algorithm for Tate ideals

0-dimensional Tate ideals

- Same definition as in the polynomial case: $K\{\mathbf{X}\}/I$ has finite dimension
- B is a K-basis of $K\{\mathbf{X}\}/I$
- Any element of $K\{\mathbf{X}\}/I$ can be represented as a **polynomial**

Outline of the algorithm

In: G_1 a reduced Gröbner basis in $K\{\mathbf{X};\mathbf{r}\}$ wrt an order $<_1$
$<_2$ a monomial order
$\mathbf{u} \leq \mathbf{r}$ a system of log-radii

Out: G_2 a reduced Gröbner basis in $K\{\mathbf{X};\mathbf{u}\}$ wrt $<_2$

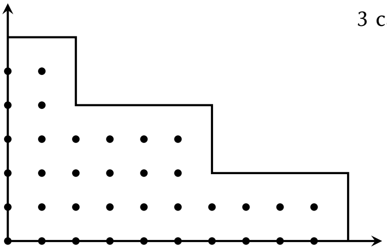
1. Compute the matrices of multiplication by $X_1, \ldots, X_n$ in the basis $B_{1,\mathbf{r}}$
2. Convert them into matrices in the basis $B_{1,\mathbf{u}}$ (computing $B_{1,\mathbf{u}}$)
3. Convert them into the Gröbner basis G_2

Complexity

- Complexity cubic in δ
- Base complexity quasi-linear in the precision

Iterative computation of the multiplication matrices

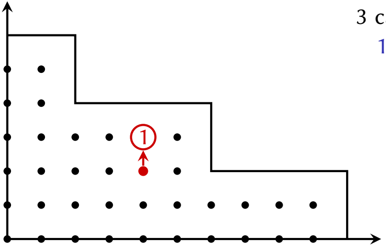
- Idea: need to compute $\mathrm{NF}(X_i m)$ for all $i \in \{1, \ldots, n\}$, $m \in B$
- Proceed in increasing order and reuse the computations



3 cases:

Iterative computation of the multiplication matrices

- Idea: need to compute $\mathrm{NF}(X_i m)$ for all $i \in \{1, \ldots, n\}$, $m \in B$
- Proceed in increasing order and reuse the computations

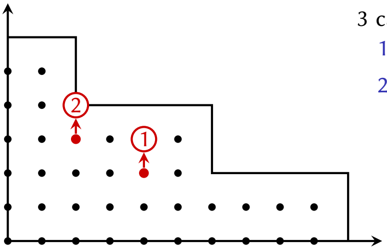


3 cases:

1. $X_i m \in B$: $\rightarrow \mathrm{NF}(X_i m) = X_i m$

Iterative computation of the multiplication matrices

- Idea: need to compute $\mathrm{NF}(X_i m)$ for all $i \in \{1, \ldots, n\}$, $m \in B$
- Proceed in increasing order and reuse the computations

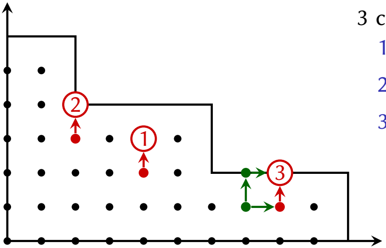


3 cases:

1. $X_i m \in B$: $\rightarrow \mathrm{NF}(X_i m) = X_i m$
2. $X_i m = \mathrm{LT}(g)$ for $g \in G \rightarrow \mathrm{NF}(X_i m) = X_i m - g$

Iterative computation of the multiplication matrices

- Idea: need to compute $\mathrm{NF}(X_i m)$ for all $i \in \{1, \ldots, n\}$, $m \in B$
- Proceed in increasing order and reuse the computations

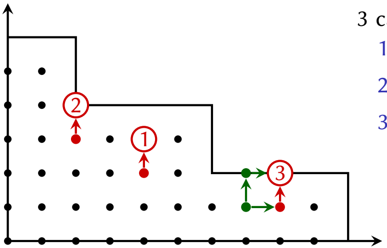


3 cases:

1. $X_i m \in B$: $\to \mathrm{NF}(X_i m) = X_i m$
2. $X_i m = \mathrm{LT}(g)$ for $g \in G \to \mathrm{NF}(X_i m) = X_i m - g$
3. Otherwise, write $m = X_j m'$ with
   $\mathrm{NF}(X_i m') = \sum a_\mu \mu$
   $\to \mathrm{NF}(X_i m) = \mathrm{NF}(X_j X_i m') = \sum a_\mu \mathrm{NF}(X_j \mu)$

Iterative computation of the multiplication matrices

- Idea: need to compute $\mathrm{NF}(X_i m)$ for all $i \in \{1, \dots, n\}$, $m \in B$
- Proceed in increasing order and reuse the computations



3 cases:

1. $X_i m \in B$: $\to \mathrm{NF}(X_i m) = X_i m$
2. $X_i m = \mathrm{LT}(g)$ for $g \in G \to \mathrm{NF}(X_i m) = X_i m - g$
3. Otherwise, write $m = X_j m'$ with
   $\mathrm{NF}(X_i m') = \sum a_\mu \mu$
   $\to \mathrm{NF}(X_i m) = \mathrm{NF}(X_j X_i m') = \sum a_\mu \mathrm{NF}(X_j \mu)$

Why does it work?

- Usual case: $\mathrm{NF}(m)$ only involves monomials smaller than m
- Tate case: not true, but if not their coefficient is smaller than 1 (i.e. divisible by π)
- So we can recover the value mod π, and repeating k times, the value mod π^k:

$a \cdot b = ab$

Two improvements on the computation of the multiplication matrices

Recursive computation:

- The previous algorithm relies on the order of the monomials
- Base complexity cubic in δ but quadratic in the precision
- Alternative: recursive algorithm, computing the coefficients mod π^k as needed
- Gives an order-agnostic algorithm which also works with non-0 log-radii
- Fast arithmetic + relaxed algorithms $\rightarrow$ base complexity quasi-linear in the precision
  [van der Hoeven 1997] [Berthomieu, van der Hoeven, Lecerf 2011] [Berthomieu, Lebreton 2012]

Two improvements on the computation of the multiplication matrices

Recursive computation:

- The previous algorithm relies on the order of the monomials
- Base complexity cubic in δ but quadratic in the precision
- Alternative: recursive algorithm, computing the coefficients mod π^k as needed
- Gives an order-agnostic algorithm which also works with non-0 log-radii
- Fast arithmetic + relaxed algorithms $\rightarrow$ base complexity quasi-linear in the precision
  [van der Hoeven 1997] [Berthomieu, van der Hoeven, Lecerf 2011] [Berthomieu, Lebreton 2012]

Non-reduced bases:

- Usual case: need bases to be reduced to ensure structure of the order
- Here, we have to consider monomials which we have not yet seen in any case
- As long as the basis is reduced mod π, the hypotheses hold
- So FGLM (with same order and log-radii as input and output)
  gives an algorithm for interreduction with complexity quasi-linear in precision
- The complexity is not only bounded in terms of δ anymore

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

- $K[x, y]$: $\mathbf{r} = (\infty, \infty)$
- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6

- $K\{x, y\}$: $\mathbf{u} = (0, 0)$

- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_2 = \{1, y\}$, degree 2!

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

- $K[x, y]$: $\mathbf{r} = (\infty, \infty)$
- $K\{x, y\}$: $\mathbf{u} = (0, 0)$

- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6
- $B_2 = \{1, y\}$, degree 2!

- Why does x disappear from the staircase?

  Consider $x^4 \cdot x$

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

- $K[x,y]$: $\mathbf{r} = (\infty, \infty)$
- $K\{x,y\}$: $\mathbf{u} = (0,0)$

- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6
- $B_2 = \{1, y\}$, degree 2!

- Why does x disappear from the staircase?

  Consider $x^4 \cdot x = \dfrac{1}{p} x^3 y^2$

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

- $K[x, y]$: $\mathbf{r} = (\infty, \infty)$
- $K\{x, y\}$: $\mathbf{u} = (0, 0)$

- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6
- $B_2 = \{1, y\}$, degree 2!

- Why does x disappear from the staircase?

  Consider $x^4 \cdot x = \dfrac{1}{p} x^3 y^2 = \dfrac{1}{p^2} x y^4$

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

- $K[x, y]$: $\mathbf{r} = (\infty, \infty)$
- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6

- $K\{x, y\}$: $\mathbf{u} = (0, 0)$

- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_2 = \{1, y\}$, degree 2!

- Why does x disappear from the staircase?
  Consider $x^4 \cdot x = \dfrac{1}{p} x^3 y^2 = \dfrac{1}{p^2} xy^4 = \dfrac{1}{p^3} x^2 y$

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

- $K[x, y]$: $\mathbf{r} = (\infty, \infty)$
- $K\{x, y\}$: $\mathbf{u} = (0, 0)$

- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6
- $B_2 = \{1, y\}$, degree 2!

- Why does x disappear from the staircase?

  Consider $x^4 \cdot x = \frac{1}{p} x^3 y^2 = \frac{1}{p^2} xy^4 = \frac{1}{p^3} x^2 y = \frac{1}{p^4} y^3$

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

- $K[x, y]$: $\mathbf{r} = (\infty, \infty)$
- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6

- $K\{x, y\}$: $\mathbf{u} = (0, 0)$

- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_2 = \{1, y\}$, degree 2!

- Why does x disappear from the staircase?
  Consider $x^4 \cdot x = \frac{1}{p} x^3 y^2 = \frac{1}{p^2} x y^4 = \frac{1}{p^3} x^2 y = \frac{1}{p^4} y^3 = \frac{1}{p^5} x$

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

- $K[x, y]$: $\mathbf{r} = (\infty, \infty)$
- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6

- $K\{x, y\}$: $\mathbf{u} = (0, 0)$
- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_2 = \{1, y\}$, degree 2!

- Why does x disappear from the staircase?

  Consider $x^4 \cdot x = \frac{1}{p} x^3 y^2 = \frac{1}{p^2} xy^4 = \frac{1}{p^3} x^2 y = \frac{1}{p^4} y^3 = \frac{1}{p^5} x$

  so $x = p^5 x^5$

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

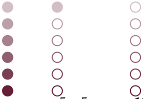
- $K[x, y]$: $\mathbf{r} = (\infty, \infty)$
- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6

- $K\{x, y\}$: $\mathbf{u} = (0, 0)$

- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_2 = \{1, y\}$, degree 2!

- Why does x disappear from the staircase?

  Consider $x^4 \cdot x = \frac{1}{p} x^3 y^2 = \frac{1}{p^2} xy^4 = \frac{1}{p^3} x^2 y = \frac{1}{p^4} y^3 = \frac{1}{p^5} x$



  so $x = p^5 x^5 = p^{10} x^9$

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

- $K[x, y]$: $\mathbf{r} = (\infty, \infty)$
- $K\{x, y\}$: $\mathbf{u} = (0, 0)$

- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6
- $B_2 = \{1, y\}$, degree 2!

- Why does x disappear from the staircase?

  Consider $x^4 \cdot x = \frac{1}{p} x^3 y^2 = \frac{1}{p^2} x y^4 = \frac{1}{p^3} x^2 y = \frac{1}{p^4} y^3 = \frac{1}{p^5} x$

  so $x = p^5 x^5 = p^{10} x^9 = \cdots = 0$

Changing log-radii: what happens to the staircase?

Example with $K = \mathbb{Q}_p$

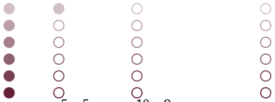
- $K[x, y]$: $\mathbf{r} = (\infty, \infty)$
- $I = \langle px^2 - y^2, py^3 - x \rangle$
- $B_1 = \{1, x, y, y^2, xy, xy^2\}$, degree 6

- $K\{x, y\}$: $\mathbf{u} = (0, 0)$

- $J = \langle y^2 - px^2, x - py^3 \rangle$
- $B_2 = \{1, y\}$, degree 2!

- Why does x disappear from the staircase?

  Consider $x^4 \cdot x = \frac{1}{p}x^3y^2 = \frac{1}{p^2}xy^4 = \frac{1}{p^3}x^2y = \frac{1}{p^4}y^3 = \frac{1}{p^5}x$

  so $x = p^5x^5 = p^{10}x^9 = \cdots = 0$ or equivalently $x(1 - p^5x^4) = 0 \implies x = 0$.

Multiplication matrices and slope factorization

- Problem: how to detect this phenomenon in general?

Consider the multiplication matrix by x:

$$T_x = \begin{array}{c} \\ \left(\begin{array}{cccccc} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & p^{-1} & 0 & p^{-2} & 0 & p^{-3} \\ 0 & 0 & 0 & 0 & 1 & 0 \end{array}\right) \\ \begin{array}{cccccc} 1 & x & y & xy & y^2 & xy^2 \end{array} \end{array} \begin{array}{c} 1 \\ x \\ y \\ xy \\ y^2 \\ xy^2 \\ \\ \end{array}$$

Characteristic polynomial:

$\chi_x = T^6 - p^{-5}T^2$

Multiplication matrices and slope factorization

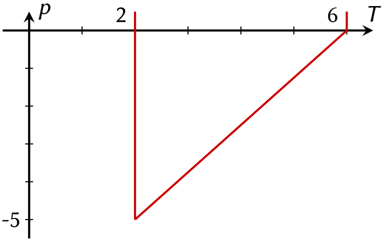
- Problem: how to detect this phenomenon in general?

Consider the multiplication matrix by x:

$$T_x = \begin{array}{c|c} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & p^{-1} & 0 & p^{-2} & 0 & p^{-3} \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} & \begin{array}{c} 1 \\ x \\ y \\ xy \\ y^2 \\ xy^2 \end{array} \\ \begin{array}{cccccc} 1 & x & y & xy & y^2 & xy^2 \end{array} & \end{array}$$

Characteristic polynomial:

$\chi_x = T^6 - p^{-5} T^2$



Multiplication matrices and slope factorization

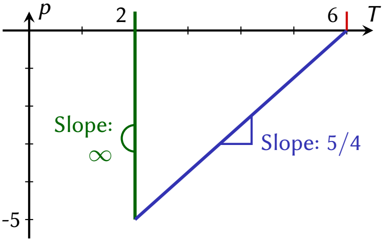
- Problem: how to detect this phenomenon in general?

Consider the multiplication matrix by x:

$$T_x = \begin{array}{cc} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & p^{-1} & 0 & p^{-2} & 0 & p^{-3} \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} & \begin{matrix} 1 \\ x \\ y \\ xy \\ y^2 \\ xy^2 \end{matrix} \\ \begin{matrix} 1 & \;x & \;y & \;xy & \;y^2 & \;xy^2 \end{matrix} & \end{array}$$

Characteristic polynomial:

$$\begin{aligned} \chi_x &= T^6 - p^{-5}T^2 \\ &= T^2 \cdot (T^4 - p^{-5}) \end{aligned}$$



Multiplication matrices and slope factorization

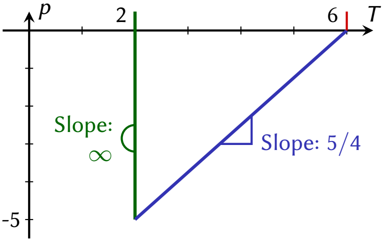
- Problem: how to detect this phenomenon in general?

Consider the multiplication matrix by x:

$$T_x = \begin{array}{cc} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & p^{-1} & 0 & p^{-2} & 0 & p^{-3} \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} & \begin{array}{c} 1 \\ x \\ y \\ xy \\ y^2 \\ xy^2 \end{array} \\ \begin{array}{cccccc} 1 & x & y & xy & y^2 & xy^2 \end{array} & \end{array}$$

Characteristic polynomial:

$$\chi_x = T^6 - p^{-5}T^2$$
$$= T^2 \cdot (T^4 - p^{-5})$$



Slope factorization:

- $\ker(T_x^4 - p^{-5})$: characteristic space with "eigenvalue" with valuation $-5/4 < 0$
  $\rightarrow$ vectors sent to 0
- $\ker(T_x^2)$: characteristic space with "eigenvalue" with valuation $\infty \geq 0$
  $\rightarrow$ vectors in the staircase

Characterization and construction of the new staircase

Construction

- Inclusion $K\{\mathbf{X};\mathbf{r}\} \to K\{\mathbf{X};\mathbf{u}\} \rightsquigarrow$ map $\Phi : V = K\{\mathbf{X};\mathbf{r}\}/I \to K\{\mathbf{X};\mathbf{u}\}/(I\,K\{\mathbf{X};\mathbf{u}\})$
- Φ is surjective but not injective
- Vectors sent to 0:

$$N = \bigcap \text{“Eigenspace” of } T_i \text{ with valuation} < u_i$$

Characterization and construction of the new staircase

Construction

- Inclusion $K\{\mathbf{X};\mathbf{r}\} \to K\{\mathbf{X};\mathbf{u}\} \rightsquigarrow$ map $\Phi : V = K\{\mathbf{X};\mathbf{r}\}/I \to K\{\mathbf{X};\mathbf{u}\}/(I\,K\{\mathbf{X};\mathbf{u}\})$
- Φ is surjective but not injective
- Vectors sent to 0:

$$N = \bigcap \text{“Eigenspace” of } T_i \text{ with valuation} < u_i$$

- New quotient:

$$K\{\mathbf{X};\mathbf{u}\}/(I+N) = \sum \text{“Eigenspace” of } T_i \text{ with valuation} \geq u_i$$

- Or simply compute a monomial basis of the quotient
- This linear algebra encodes a topological construction

Full FGLM algorithm for Tate algebras

In: G_1 a reduced Gröbner basis in $K\{\mathbf{X};\mathbf{r}\}$ wrt an order $<_1$
$<_2$ a monomial order
$\mathbf{u} \le \mathbf{r}$ a system of log-radii

Out: G_2 a reduced Gröbner basis wrt $<_2$ in $K\{\mathbf{X};\mathbf{u}\}$

1. Compute the matrices of multiplication by $X_1, \ldots, X_n$ in the basis $B_{1,\mathbf{r}}$
2. Convert them into matrices of multiplication by $X_1, \ldots, X_n$ in the basis $B_{1,\mathbf{u}}$ (slope factorization)
3. Convert into the basis G_2
   3.1 Use the usual algorithm modulo π (in $\mathbb{F}$) to compute $B_{2,\mathbf{u}}$ and $\overline{G_2}$
   3.2 Lift the linear algebra operations to obtain G_2

Full FGLM algorithm for Tate algebras

In: G_1 a reduced Gröbner basis in $K\{\mathbf{X};\mathbf{r}\}$ wrt an order $<_1$
$<_2$ a monomial order
$\mathbf{u} \leq \mathbf{r}$ a system of log-radii

Out: G_2 a reduced Gröbner basis wrt $<_2$ in $K\{\mathbf{X};\mathbf{u}\}$

1. Compute the matrices of multiplication by $X_1, \dots, X_n$ in the basis $B_{1,\mathbf{r}}$
2. Convert them into matrices of multiplication by $X_1, \dots, X_n$ in the basis $B_{1,\mathbf{u}}$ (slope factorization)
3. Convert into the basis G_2
   3.1 Use the usual algorithm modulo π (in $\mathbb{F}$) to compute $B_{2,\mathbf{u}}$ and $\overline{G_2}$
   3.2 Lift the linear algebra operations to obtain G_2

Complexity

- Step 1 has base complexity $\tilde{O}(n\delta^3 \mathrm{prec})$
- Each other step has arithmetic complexity $\tilde{O}(n\delta^3)$
- Final base complexity: $\tilde{O}(n\delta^3 \mathrm{prec})$

Conclusion

Summary

- FGLM algorithm for Tate series
- Allows to perform interreduction and change of log-radii in dimension 0
- Complexity cubic in degree, quasi-linear in precision

Conclusion

Summary

- FGLM algorithm for Tate series
- Allows to perform interreduction and change of log-radii in dimension 0
- Complexity cubic in degree, quasi-linear in precision

Future work

- Integrate FGLM in the `tate_algebra` package of SageMath
- Generalizations of the interreduction in the middle of GB calculations
- Improve the complexity of reduction in positive dimension

Conclusion

Summary

- FGLM algorithm for Tate series
- Allows to perform interreduction and change of log-radii in dimension 0
- Complexity cubic in degree, quasi-linear in precision

Future work

- Integrate FGLM in the `tate_algebra` package of SageMath
- Generalizations of the interreduction in the middle of GB calculations
- Improve the complexity of reduction in positive dimension

Thank you for your attention!

References

- *Gröbner bases over Tate algebras*, ISSAC 2019
- *Signature-based algorithms for Gröbner bases over Tate algebras*, ISSAC 2020
- *On FGLM algorithms with Tate algebras*, preprint 2021