Recent progress on computing Riemann-Roch spaces

Simon Abelard
Laboratoire d'informatique de l'École Polytechnique
Institut Polytechnique de Paris, CNRS, Inria

Part of a project funded by the Agence de l'Innovation de Défense.

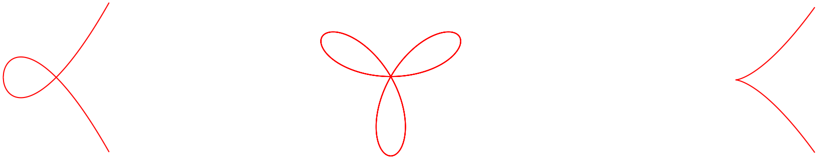
October 13, 2020

First assumption: ordinary curves

Input curves

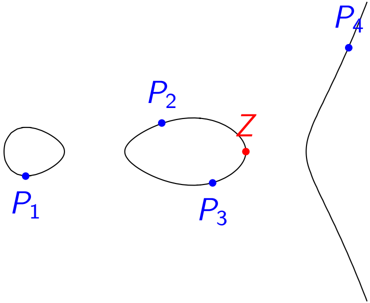
An absolutely irreducible ordinary plane projective curve.
Given by an equation $\mathcal{C} : Q(X, Y, Z) = 0$.
Ordinary: multiple points with distinct tangents at each branch.

Nodal curve | Ordinary curve | Non-ordinary curve



Results given for characteristic 0, see papers for other perfect fields.

Riemann-Roch problem

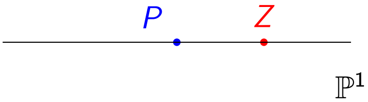


Goal: find all functions $\frac{G(X,Y)}{H(X,Y)}$ such that:

- Z **has to be** a zero of G.
- The P_i's **may** be zeros H.
- G/H has no other pole (including at infinity).

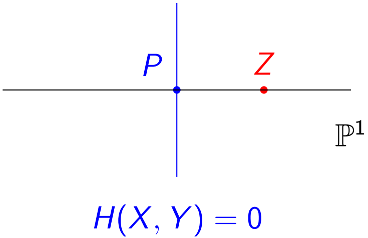
A toy example

Set $\mathcal{C} = \mathbb{P}^1$, $P = [0 : 1]$, $Z = [1 : 1]$ and $D = P - Z$.
Previous slide : $\frac{X-1}{X}$ is a solution (one pole in P and one zero in Z).
Riemann-Roch theorem: $\frac{X-1}{X}$ generates the solution space.



A toy example

Set $\mathcal{C} = \mathbb{P}^1$, $P = [0:1]$, $Z = [1:1]$ and $D = P - Z$.
Previous slide : $\frac{X-1}{X}$ is a solution (one pole in P and one zero in Z).
Riemann-Roch theorem: $\frac{X-1}{X}$ generates the solution space.

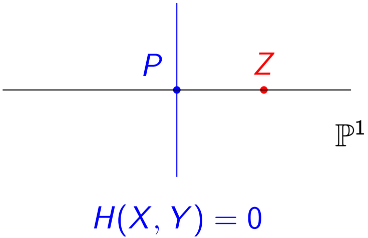


Our strategy

Denominator H passes through P.
This means $H(X,Y) \bmod X = 0$.

A toy example

Set $\mathcal{C} = \mathbb{P}^1$, $P = [0:1]$, $Z = [1:1]$ and $D = P - Z$.
Previous slide : $\frac{X-1}{X}$ is a solution (one pole in P and one zero in Z).
Riemann-Roch theorem: $\frac{X-1}{X}$ generates the solution space.



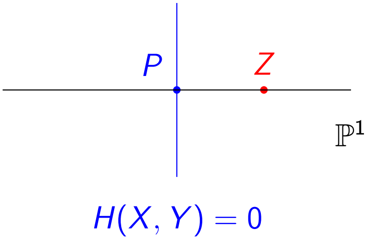
Our strategy

Denominator H passes through P.
This means $H(X, Y) \bmod X = 0$.

Numerators G pass through Z.
It means $G(X, Y) = 0 \bmod (X - 1)$.

A toy example

Set $\mathcal{C} = \mathbb{P}^1$, $P = [0:1]$, $Z = [1:1]$ and $D = P - Z$.
Previous slide : $\frac{X-1}{X}$ is a solution (one pole in P and one zero in Z).
Riemann-Roch theorem: $\frac{X-1}{X}$ generates the solution space.



Our strategy

Denominator H passes through P.
This means $H(X,Y) \bmod X = 0$.

Numerators G pass through Z.
It means $G(X,Y) = 0 \bmod (X-1)$.

We recover the solution $\frac{X-1}{X}$.

Divisors and Riemann-Roch spaces

Smooth divisor D: finite formal sum $\sum_P m_P P$ of smooth points on $\mathcal{C}$.
Degree of a divisor: $\deg(D) = \sum_P m_P$.

Riemann-Roch space $L(D)$: set of rational fractions h such that

- If $m_P < 0$, P **has to be a zero** of h with multiplicity $\geq -m_P$.
- If $m_P > 0$, P **can be a pole** of h with multiplicity $\leq m_P$.

Divisors and Riemann-Roch spaces

Smooth divisor D: finite formal sum $\sum_P m_P P$ of smooth points on $\mathcal{C}$.
Degree of a divisor: $\deg(D) = \sum_P m_P$.

Riemann-Roch space $L(D)$: set of rational fractions h such that

- If $m_P < 0$, P **has to be a zero** of h with multiplicity $\geq -m_P$.
- If $m_P > 0$, P **can be a pole** of h with multiplicity $\leq m_P$.

Remember: zeros constrained by D_- and poles allowed by D_+.

Divisors and Riemann-Roch spaces

Smooth divisor D: finite formal sum $\sum_P m_P P$ of smooth points on $\mathcal{C}$.
Degree of a divisor: $\deg(D) = \sum_P m_P$.

Riemann-Roch space $L(D)$: set of rational fractions h such that

- If $m_P < 0$, P **has to be a zero** of h with multiplicity $\geq -m_P$.
- If $m_P > 0$, P **can be a pole** of h with multiplicity $\leq m_P$.

Remember: zeros constrained by D_- and poles allowed by D_+.

Our problem:

Given input ordinary curve $\mathcal{C}$ and smooth divisor D,
Compute a basis of the vector space $L(D)$.

Applications

- Diophantine equations (Coates, 1970)
- Symbolic integration (Davenport, 1981)
- Group operations in Jacobians of curves (cryptography in 1990's)
- Geometric codes (need to evaluate functions in $L(D)$)

Geometric vs arithmetic methods

Geometric methods:
Based on Brill-Noether theory.

Arithmetic methods:
Ideals in function fields.

Geometric vs arithmetic methods

Geometric methods:
Based on Brill-Noether theory.

- Goppa, Le Brigand-Risler (80's)
- Huang-Ierardi, Volcheck (90's)
- Khuri-Makdisi (2007)
- Le Gluher-Spaenlehauer (2018)

Arithmetic methods:
Ideals in function fields.

- Coates (1970)
- Davenport (1981)
- Hess' algorithm (2001)

Geometric vs arithmetic methods

Geometric methods:
Based on Brill-Noether theory.

- Goppa, Le Brigand-Risler (80's)
- Huang-Ierardi, Volcheck (90's)
- Khuri-Makdisi (2007)
- Le Gluher-Spaenlehauer (2018)

Arithmetic methods:
Ideals in function fields.

- Coates (1970)
- Davenport (1981)
- Hess' algorithm (2001)

Brief comparison

	Geometric	Arithmetic
Advantage:	faster (so far).	Very general.
Weakness:	for particular curves.	Unclear complexity bounds.
Complexity:	exponent ω (lin. alg.).	

Plan for today

- Geometric methods (joint with A. Couvreur & G. Lecerf)
  - Brill-Noether theory
  - Representing and handling divisors
  - Riemann-Roch spaces through interpolation
- Arithmetic Methods
  - Overview
  - Computing integral bases

What's new?

Brill-Noether theory: conditions to belong to a Riemann-Roch space.
State of the art: conditions $\rightsquigarrow$ linear algebra.
Novelty: use $K[X]$-module structure instead (faster algorithms).

[1] V. Neiger, Fast computation of shifted Popov forms of polynomial matrices via systems of modular polynomial equations, ISSAC 2016.

What's new?

Brill-Noether theory: conditions to belong to a Riemann-Roch space.
State of the art: conditions $\rightsquigarrow$ linear algebra.
Novelty: use $K[X]$-module structure instead (faster algorithms).

Main contributions:

- Replace linear algebra by structured linear algebra[1].
- Faster algorithms for divisor arithmetic.
- Existence of a nice suitable common denominator.

[1] V. Neiger, Fast computation of shifted Popov forms of polynomial matrices via systems of modular polynomial equations, ISSAC 2016.

What's new?

Brill-Noether theory: conditions to belong to a Riemann-Roch space.
State of the art: conditions $\rightsquigarrow$ linear algebra.
Novelty: use $K[X]$-module structure instead (faster algorithms).

Main contributions:

- Replace linear algebra by structured linear algebra[1].
- Faster algorithms for divisor arithmetic.
- Existence of a nice suitable common denominator.

Main complexity bound

Las Vegas algorithm computing $L(D)$ in $\widetilde{O}\left(((\deg \mathcal{C})^2 + \deg D_+)^{\frac{\omega+1}{2}}\right)$ field operations (previous best exponent is ω).

[1] V. Neiger, Fast computation of shifted Popov forms of polynomial matrices via systems of modular polynomial equations, ISSAC 2016.

A basis of $L(D)$ through Brill-Noether theory

Effective divisors

$D = \sum m_i P_i$ is positive or effective if for any i, $m_i \geq 0$.
Can split $D = D_+ - D_-$ as a difference of two effective divisors.
Denote $D \geq D'$ whenever $D - D'$ is effective.

A basis of $L(D)$ through Brill-Noether theory

Effective divisors

$D = \sum m_i P_i$ is positive or effective if for any i, $m_i \geq 0$.
Can split $D = D_+ - D_-$ as a difference of two effective divisors.
Denote $D \geq D'$ whenever $D - D'$ is effective.

Principal divisor: $(h) = \sum_{P \in \mathcal{C}} \text{ord}_P(h) P$ (zeros−poles with multiplicity)

A basis of $L(D)$ through Brill-Noether theory

Effective divisors

$D = \sum m_i P_i$ is positive or effective if for any i, $m_i \geq 0$.
Can split $D = D_+ - D_-$ as a difference of two effective divisors.
Denote $D \geq D'$ whenever $D - D'$ is effective.

Principal divisor: $(h) = \sum_{P \in \mathcal{C}} \mathrm{ord}_P(h) P$ (zeros−poles with multiplicity)

A description for $L(D)$ (Haché, Le Brigand-Risler)

Non-zero elements of $L(D)$ are of the form G/H where:

- The common denominator H satisfies $(H) \geq D$.
- H passes through singularities of $\mathcal{C}$ with given multiplicities.
- G is of degree $\deg H$, not divisible by Q and $(G) \geq (H) - D$.

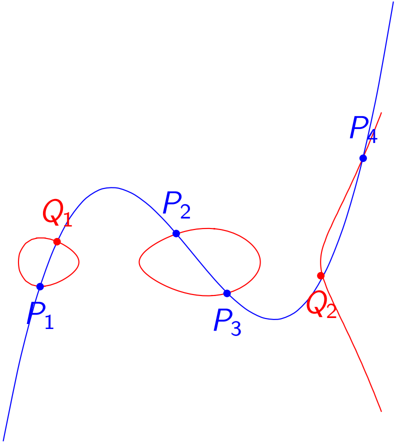
Sketch of the algorithm

Step 1 Find a denominator H.

Step 2 Compute (H).

Step 3 Compute $(H) - D$.

Step 4 Compute numerators.
(Very similar to step 1)



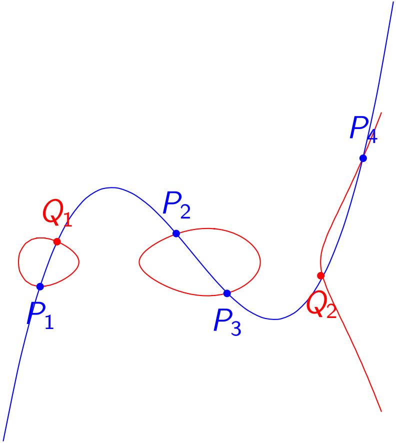
Sketch of the algorithm

Step 1 Find a denominator H.

Step 2 Compute (H).

Step 3 Compute $(H) - D$.

Step 4 Compute numerators.
(Very similar to step 1)



Problem: how do we handle divisors ?

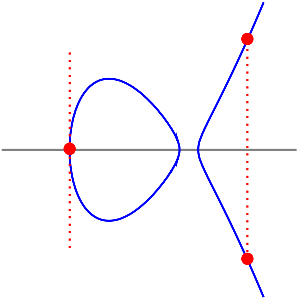
Representing effective divisors, expectation

Goal: transform divisor operations into polynomial operations.

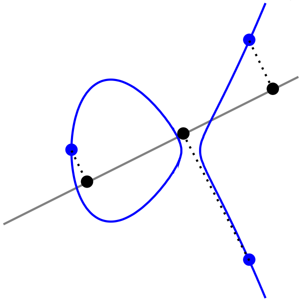
First try: $D = \sum_i m_i P_i$ with P_i of coordinates (x_i, y_i).
Encode $u(X) = \prod_i (X - x_i)^{m_i}$ and compute v such that $y_i = v(x_i)$.

Intuition: Project on line $y = 0$, u describes the projected points.

Problems:



Solution: $S = \lambda X + \mu Y$



Representing effective divisors, reality

Let D be a smooth effective divisor, *i.e.* a multi-set of smooth points. This set is put in primitive representation $(\lambda, \mu, \chi, u, v)$ with

- (λ, μ) yields a primitive element $\lambda X + \mu Y$
- χ monic of degree $\deg D$
- $\deg u$ and $\deg v$ are $< \deg D$
- $Q(u(S), v(S)) = 0 \bmod \chi(S)$
- $\lambda u(S) + \mu v(S) = S$
- $\mu \frac{\partial Q}{\partial X}(u(S), v(S)) - \lambda \frac{\partial Q}{\partial Y}(u(S), v(S))$ is coprime to $\chi(S)$.

Representing effective divisors, reality

Let D be a smooth effective divisor, *i.e.* a multi-set of smooth points. This set is put in primitive representation $(\lambda, \mu, \chi, u, v)$ with

- (λ, μ) yields a primitive element $\lambda X + \mu Y$
- χ monic of degree $\deg D$
- $\deg u$ and $\deg v$ are $< \deg D$
- $Q(u(S), v(S)) = 0 \bmod \chi(S)$
- $\lambda u(S) + \mu v(S) = S$
- $\mu \frac{\partial Q}{\partial X}(u(S), v(S)) - \lambda \frac{\partial Q}{\partial Y}(u(S), v(S))$ is coprime to $\chi(S)$.

Remarks: Such representation may not exist if base field too small. This is not unique, but it becomes unique once (λ, μ) is chosen.

Interface for divisors

- Change of primitive element
- Doubling a divisor
- Addition and subtraction:
  Find common primitive element (step above).
  For disjoint supports, product and CRT.
  For intersection, use doubling step.
- Computing representation of a principal divisor (H)

Conclusion: primitive representation has the routines we want.

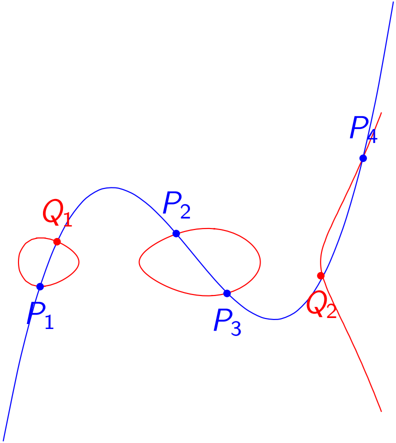
Sketch of the algorithm

Step 1 Find a denominator H.

Step 2 Compute (H).

Step 3 Compute $(H) - D$.

Step 4 Compute numerators.
(Very similar to step 1)



Problem: how about the interpolation step ?

Finding a denominator in practice

Conditions on H: passing through singularities and $(H) \geq D_+$.

In primitive form, $(H) \geq D_+ \Leftrightarrow H(X, v_+(X)) = 0 \bmod \chi_+(X)$.
Passing through singularities: similar equations.

Finding a denominator in practice

Conditions on H: passing through singularities and $(H) \geq D_+$.

In primitive form, $(H) \geq D_+ \Leftrightarrow H(X, v_+(X)) = 0 \bmod \chi_+(X)$.
Passing through singularities: similar equations.

Set $d = \deg H$ and write $H = \sum_{i=1}^{d} h_i(X) Y^i$.
Above conditions on H: the h_i's are in a $K[X]$-module of rank $d + 1$.

Finding a denominator in practice

Conditions on H: passing through singularities and $(H) \geq D_+$.

In primitive form, $(H) \geq D_+ \Leftrightarrow H(X, v_+(X)) = 0 \bmod \chi_+(X)$.
Passing through singularities: similar equations.

Set $d = \deg H$ and write $H = \sum_{i=1}^{d} h_i(X) Y^i$.
Above conditions on H: the h_i's are in a $K[X]$-module of rank $d+1$.

Computing a solution basis (Neiger, 2016)

A basis of this $K[X]$-module costs $\widetilde{O}(d^{\omega-1} \deg \chi_+)$ field ops.
(Linear algebra on $d \times d$ polynomial matrices of degree $\leq \deg \chi_+$.)

How big is d? We prove that $d = \left\lceil \frac{(\deg \mathcal{C}-1)(\deg \mathcal{C}-2)+\deg \chi_+}{\deg \mathcal{C}} \right\rceil$ is enough.

Finding numerators of a basis

A similar condition on numerators

We have $G/H \in L(D)$ iff $G = 0$ or $(G) \geq (H) - D$.
By construction, smooth part of (H) is $D_+ + R$ with R effective.
Conditions on G: passing through singularities and $(G) \geq R + D_-$.

Finding numerators of a basis

A similar condition on numerators

We have $G/H \in L(D)$ iff $G = 0$ or $(G) \geq (H) - D$.
By construction, smooth part of (H) is $D_+ + R$ with R effective.
Conditions on G: passing through singularities and $(G) \geq R + D_-$.

Previous problem with $R + D_-$ instead of D_+, same d.
Only difference: now need basis and not single element in module.
But a solution basis is exactly what Neiger's algorithm computes.

Value of $d \rightsquigarrow$ both steps in $\widetilde{O}\left(((\deg \mathcal{C})^2 + \deg D_+)^{\frac{\omega+1}{2}}\right)$ field ops.

Overall complexity

Step 1 First guess for the common denominator:
Structured linear algebra in $\widetilde{O}\left(((\deg \mathcal{C})^2 + \deg D_+)^{\frac{\omega+1}{2}}\right)$.

Overall complexity

Step 1 First guess for the common denominator:
Structured linear algebra in $\widetilde{O}\left(((\deg \mathcal{C})^2 + \deg D_+)^{\frac{\omega+1}{2}}\right)$.

Step 2 Compute principal divisor (H):
Resultant and characteristic polynomial in $\widetilde{O}((\deg \mathcal{C})^3 + \deg \mathcal{C} \deg D_+)$.

Overall complexity

Step 1 First guess for the common denominator:
Structured linear algebra in $\widetilde{O}\left(((\deg \mathcal{C})^2 + \deg D_+)^{\frac{\omega+1}{2}}\right)$.

Step 2 Compute principal divisor (H):
Resultant and characteristic polynomial in $\widetilde{O}((\deg \mathcal{C})^3 + \deg \mathcal{C} \deg D_+)$.

Step 3 Compute $[(H) - D]_+$:
Arithmetic on divisors in $\widetilde{O}\left((\deg \mathcal{C})^{\omega/2+1} + (\deg D_+)^{(\omega+2)/3}\right)$.

Overall complexity

Step 1 First guess for the common denominator:
Structured linear algebra in $\widetilde{O}\left(((\deg \mathcal{C})^2 + \deg D_+)^{\frac{\omega+1}{2}}\right)$.

Step 2 Compute principal divisor (H):
Resultant and characteristic polynomial in $\widetilde{O}((\deg \mathcal{C})^3 + \deg \mathcal{C} \deg D_+)$.

Step 3 Compute $[(H) - D]_+$:
Arithmetic on divisors in $\widetilde{O}\left((\deg \mathcal{C})^{\omega/2+1} + (\deg D_+)^{(\omega+2)/3}\right)$.

Step 4 Computing numerators of the basis: same as Step 1.

Overall complexity: $\widetilde{O}\left(((\deg \mathcal{C})^2 + \deg D_+)^{\frac{\omega+1}{2}}\right)$ field operations.

Overall complexity

Step 1 First guess for the common denominator:
Structured linear algebra in $\widetilde{O}\left(((\deg \mathcal{C})^2 + \deg D_+)^{\frac{\omega+1}{2}}\right)$.

Step 2 Compute principal divisor (H):
Resultant and characteristic polynomial in $\widetilde{O}((\deg \mathcal{C})^3 + \deg \mathcal{C} \deg D_+)$.

Step 3 Compute $[(H) - D]_+$:
Arithmetic on divisors in $\widetilde{O}\left((\deg \mathcal{C})^{\omega/2+1} + (\deg D_+)^{(\omega+2)/3}\right)$.

Step 4 Computing numerators of the basis: same as Step 1.

Overall complexity: $\widetilde{O}\left(((\deg \mathcal{C})^2 + \deg D_+)^{\frac{\omega+1}{2}}\right)$ field operations.

Assumptions : ordinary curve, smooth divisor, base field large enough.

Prospective

- Implementation including fast structured linear algebra.
- Extend to non-ordinary curve.

Prospective

- Implementation including fast structured linear algebra.
- Extend to non-ordinary curve.

Main obstacles:

- Conditions on H are more complicated to rephrase.
- Too many equations to use Neiger's algorithm.

Prospective

- Implementation including fast structured linear algebra.
- Extend to non-ordinary curve.

Main obstacles:

- Conditions on H are more complicated to rephrase.
- Too many equations to use Neiger's algorithm.

Options:

- Approach based on linear algebra.
- Generalization of Neiger's work in overdetermined case.
- Find a suitable way to rephrase conditions on H.
- Use arithmetic methods (Hess).

Part 2, Arithmetic methods

Main ingredients

- Correspondance divisors on $\mathcal{C} \leftrightarrow$ ideals of $K(\mathcal{C})$.
- Computing Riemann-Roch spaces $\rightsquigarrow$ ideal arithmetic.
- Integral bases: ideal arithmetic $\rightsquigarrow$ polynomial matrices.

Questions: complexity bounds for this approach?
Today: cost of precomputing integral bases.

Algebraic function fields, integral bases

Algebraic function fields

Consider a plane curve $\mathcal{C}$ over perfect field K of equation $f(x, y) = 0$.
View $f \in K[x][y]$, monic of degree n, irreducible.
Function field $K(\mathcal{C}) = \mathrm{Frac}\left(K[x, y]/\langle f(x, y)\rangle\right)$.
Field of rational fractions modulo $\frac{f_1}{g_1} \sim \frac{f_2}{g_2}$ iff $f_1 g_2 - f_2 g_1 = 0 \bmod f$.

Algebraic function fields, integral bases

Algebraic function fields

Consider a plane curve $\mathcal{C}$ over perfect field K of equation $f(x, y) = 0$.
View $f \in K[x][y]$, monic of degree n, irreducible.
Function field $K(\mathcal{C}) = \mathrm{Frac}\left(K[x, y]/\langle f(x, y)\rangle\right)$.
Field of rational fractions modulo $\frac{f_1}{g_1} \sim \frac{f_2}{g_2}$ iff $f_1 g_2 - f_2 g_1 = 0 \bmod f$.

Integral elements

A function $g \in K(\mathcal{C})$ is integral (over $K[x]$) if there is a monic polynomial $\mu \in K[x][y]$ such that $\mu(g(x, y)) = 0$.

Algebraic function fields, integral bases

Algebraic function fields

Consider a plane curve $\mathcal{C}$ over perfect field K of equation $f(x, y) = 0$.
View $f \in K[x][y]$, monic of degree n, irreducible.
Function field $K(\mathcal{C}) = \mathrm{Frac}\left(K[x, y]/\langle f(x, y)\rangle\right)$.
Field of rational fractions modulo $\frac{f_1}{g_1} \sim \frac{f_2}{g_2}$ iff $f_1 g_2 - f_2 g_1 = 0 \bmod f$.

Integral elements

A function $g \in K(\mathcal{C})$ is integral (over $K[x]$) if there is a monic polynomial $\mu \in K[x][y]$ such that $\mu(g(x, y)) = 0$.

Example: $1, y, \ldots, y^{n-1}$ are integral elements.
Integral elements form a $K[x]$-module of rank n.
A $K[x]$-basis of this module is an **integral basis**.

Incremental algorithms for integral bases

General principle

Start with $B = (1, y, \cdots, y^{n-1})$, it generates an integral module.

Compute a matrix $A \in K(x)^{n \times n}$ such that AB remains integral and generates a greater module.

Replace B by AB and repeat until a criterion is met.

Incremental algorithms for integral bases

General principle

Start with $B = (1, y, \cdots, y^{n-1})$, it generates an integral module.

Compute a matrix $A \in K(x)^{n \times n}$ such that AB remains integral and generates a greater module.

Replace B by AB and repeat until a criterion is met.

- **Trager's algorithm** (1984), criterion from commutative algebra.
  Finding A: Popov form in $K[x]^{n^2 \times n}$, Gaussian red. in $K(x)^{n^2 \times n}$.
- **Van Hoeij's algorithm** (1995) using Puiseux series as criterion.
  Finding A: solving $n^2 \times n$ linear systems.

Integral bases through factorization

Algorithm of Böhm, Decker, Laplagne, Pfister (2015):
Factor $f(x, y)$ in $K[[x]][y]$ (branch-wise approach).

Key idea: if f is irreducible, explicit formulas are known.

Integral bases through factorization

Algorithm of Böhm, Decker, Laplagne, Pfister (2015):
Factor $f(x, y)$ in $K[[x]][y]$ (branch-wise approach).

Key idea: if f is irreducible, explicit formulas are known.

- Factor f over $K[[x]][y]$ (Poteaux-Weimann).
- At each branch, deduce integral basis using Puiseux series.
- Glue each branch and perform CRT to deduce an integral basis.

Contributions

- Update well-known algorithms with state-of-the-art routines.
  - Puiseux series (characteristic $> n$), factorization in $K[[x]][y]$.
    (Poteaux, Rybowicz, Weimann)
  - Polynomial matrices.
    (Labahn, Neiger, Storjohann, Zhou and many more)
- Complexity bounds for these tailored versions.

Contributions

- Update well-known algorithms with state-of-the-art routines.
  - Puiseux series (characteristic $> n$), factorization in $K[[x]][y]$. (Poteaux, Rybowicz, Weimann)
  - Polynomial matrices. (Labahn, Neiger, Storjohann, Zhou and many more)
- Complexity bounds for these tailored versions.

Notation: $n = \deg_y(f)$, $\delta = \deg(\mathrm{Disc}_y(f))$, $\omega \leq 3$ exponent for lin. alg.

Algorithm	# Field Operations	Univariate factorization
Trager	$\widetilde{O}(n^5\delta)$	$\mathrm{Disc}_y(f)$
Van Hoeij	$\widetilde{O}(n^{\omega+2}\delta)$	$\mathrm{Disc}_y(f)$
Böhm et al.	$\widetilde{O}(n^2\delta)$	$\mathrm{Disc}_y(f)$

Contributions (simplified)

Set $D = \max(\deg_y(f), \deg_x(f))$, $\delta \leq D^2$, ignore factorization.

Algorithm	# Field Operations
Trager	$\widetilde{O}(D^7)$
Van Hoeij	$\widetilde{O}(D^{\omega+4})$
Böhm et al.	$\widetilde{O}(D^4)$

Input size: $f \in K[x, y]$ has $\leq D^2$ monomials.
Output size: integral basis has $O(D^4)$ field elements.

Future work

- Investigate Hess' algorithm.
  (Complexity bounds, exploit progress on polynomial matrices)
- Better representation for integral bases ?
  (Quasi-optimal is not good enough)

Future work

- Investigate Hess' algorithm.
  (Complexity bounds, exploit progress on polynomial matrices)
- Better representation for integral bases ?
  (Quasi-optimal is not good enough)

Thank you for your attention !