Modular Techniques in Computational Algebraic Geometry

Janko Boehm
joint with W. Decker, C. Fieker, S. Laplagne, G. Pfister

Technische Universität Kaiserslautern

15 January 2016

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.
- Fundamental approach:
    1. Compute modulo primes.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.
- Fundamental approach:
  1. Compute modulo primes.
  2. Reconstruct result over $\mathbb{Q}$.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.
- Fundamental approach:
  1. Compute modulo primes.
  2. Reconstruct result over $\mathbb{Q}$.
- Benefits:
  - Avoid intermediate coefficient growth.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.
- Fundamental approach:
  1. Compute modulo primes.
  2. Reconstruct result over $\mathbb{Q}$.
- Benefits:
  - Avoid intermediate coefficient growth.
  - Obtain parallel version of the algorithm.

Modular computations

- Many exact computations in computer algebra are carried out over $\mathbb{Q}$ and extensions thereof.
- Modular techniques are an important tool to improve performance of algorithms over $\mathbb{Q}$.
- Fundamental approach:
  1. Compute modulo primes.
  2. Reconstruct result over $\mathbb{Q}$.
- Benefits:
  - Avoid intermediate coefficient growth.
  - Obtain parallel version of the algorithm.
- Goal:
  General reconstruction scheme for algorithms in commutative algebra, algebraic geometry, number theory.

Outline

- Modular computations and rational reconstruction
- Bad primes

Outline

- Modular computations and rational reconstruction
- Bad primes

- Error tolerant lifting
- General reconstruction scheme

Outline

- Modular computations and rational reconstruction
- Bad primes

- Error tolerant lifting
- General reconstruction scheme

- Normalization
- Local-to-global algorithm for adjoint ideals
- Modular version and verification

Modular computations

Example

Compute

$$\frac{3}{4}+\frac{1}{3}=\frac{13}{12}$$

using modular techniques:

Modular computations

Example

Compute

$$\frac{3}{4} + \frac{1}{3} = \frac{13}{12}$$

using modular techniques:

$$\begin{array}{ccccccccccc} & & \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ \frac{3}{4} & \mapsto & (\,\overline{2} & , & \overline{6} & , & \overline{9} & , & \overline{26}\,) & & \end{array}$$

Modular computations

Example

Compute
$$\frac{3}{4}+\frac{1}{3}=\frac{13}{12}$$
using modular techniques:

$$\begin{array}{ccccccccccc} & & \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ \frac{3}{4} & \mapsto & (\,\overline{2} & , & \overline{6} & , & \overline{9} & , & \overline{26}\,) & & \\ & & & & + & & & & & & \\ \frac{1}{3} & \mapsto & (\,\overline{2} & , & \overline{5} & , & \overline{4} & , & \overline{34}\,) & & \end{array}$$

Modular computations

Example

Compute

$$\frac{3}{4} + \frac{1}{3} = \frac{13}{12}$$

using modular techniques:

$$\begin{array}{ccccccccccc}
 & & \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\
\frac{3}{4} & \mapsto & (\,\overline{2} & , & \overline{6} & , & \overline{9} & , & \overline{26}\,) & & \\
 & & & & & + & & & & & \\
\frac{1}{3} & \mapsto & (\,\overline{2} & , & \overline{5} & , & \overline{4} & , & \overline{34}\,) & & \\
 & & & & & \shortparallel & & & & & \\
 & & (\,\overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\,) & &
\end{array}$$

Modular computations

Example

Compute

$$\frac{3}{4} + \frac{1}{3} = \frac{13}{12}$$

using modular techniques:

$$\begin{array}{ccccccccccc}
 & & \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\
\frac{3}{4} & \mapsto & (\ \overline{2} & , & \overline{6} & , & \overline{9} & , & \overline{26}\ ) & & \\
 & & & & & + & & & & & \\
\frac{1}{3} & \mapsto & (\ \overline{2} & , & \overline{5} & , & \overline{4} & , & \overline{34}\ ) & & \\
 & & & & & \shortparallel & & & & & \\
 & & (\ \overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\ ) & \mapsto & \overline{22684}
\end{array}$$

Modular computations

Example

Compute

$$\frac{3}{4}+\frac{1}{3}=\frac{13}{12}$$

using modular techniques:

$$\begin{array}{ccccccccccc}
 & & \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\
\frac{3}{4} & \mapsto & (\,\overline{2} & , & \overline{6} & , & \overline{9} & , & \overline{26}\,) & & \\
 & & & & & + & & & & & \\
\frac{1}{3} & \mapsto & (\,\overline{2} & , & \overline{5} & , & \overline{4} & , & \overline{34}\,) & & \\
 & & & & & \shortparallel & & & & & \\
 & & (\,\overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\,) & \mapsto & \overline{22684}
\end{array}$$

How to obtain a rational number from $\overline{22684}$?

Rational reconstruction

Theorem (Kornerup, Gregory, 1983)

The **Farey map**

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{c} \gcd(a,b)=1 \\ \gcd(b,N)=1 \end{array} \quad |a|\,,|b| \leq \sqrt{(N-1)/2} \right\} \longrightarrow \mathbb{Z}/N$$

$$\tfrac{a}{b} \longmapsto \overline{a} \cdot \overline{b}^{-1}$$

Rational reconstruction

Theorem (Kornerup, Gregory, 1983)

The **Farey map**

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{c} \gcd(a,b)=1 \\ \gcd(b,N)=1 \end{array} \quad |a|\,,|b| \leq \sqrt{(N-1)/2} \right\} \longrightarrow \mathbb{Z}/N$$

$$\tfrac{a}{b} \longmapsto \overline{a} \cdot \overline{b}^{-1}$$

is injective.

Rational reconstruction

Theorem (Kornerup, Gregory, 1983)

The **Farey map**

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \,\middle|\, \begin{array}{c} \gcd(a,b) = 1 \\ \gcd(b,N) = 1 \end{array} \quad |a|, |b| \leq \sqrt{(N-1)/2} \right\} \longrightarrow \mathbb{Z}/N$$

$$\tfrac{a}{b} \longmapsto \overline{a} \cdot \overline{b}^{-1}$$

is injective. Efficient algorithm for preimage.

Rational reconstruction

Theorem (Kornerup, Gregory, 1983)

The **Farey map**

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{c} \gcd(a,b) = 1 \\ \gcd(b,N) = 1 \end{array} \quad |a|\,, |b| \leq \sqrt{(N-1)/2} \right\} \longrightarrow \mathbb{Z}/N$$

$$\tfrac{a}{b} \longmapsto \overline{a} \cdot \overline{b}^{-1}$$

is injective. Efficient algorithm for preimage.

Example

Indeed, in the above example

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{array}{r} \gcd(a,b) = 1 \\ \gcd(b,38885) = 1 \end{array} \quad |a|\,, |b| \leq 139 \right\} \longrightarrow \mathbb{Z}/38885$$

Rational reconstruction

Theorem (Kornerup, Gregory, 1983)

The **Farey map**

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{matrix} \gcd(a,b) = 1 \\ \gcd(b,N) = 1 \end{matrix} \quad |a|\,,|b| \leq \sqrt{(N-1)/2} \right\} \longrightarrow \mathbb{Z}/N$$

$$\tfrac{a}{b} \longmapsto \overline{a} \cdot \overline{b}^{-1}$$

is injective. Efficient algorithm for preimage.

Example

Indeed, in the above example

$$\left\{ \tfrac{a}{b} \in \mathbb{Q} \;\middle|\; \begin{matrix} \gcd(a,b) = 1 \\ \gcd(b,38885) = 1 \end{matrix} \quad |a|\,,|b| \leq 139 \right\} \longrightarrow \mathbb{Z}/38885$$

$$\tfrac{13}{12} \longmapsto \overline{22684}$$

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \;\cong\; \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \cong \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

3. If exists, compute preimage w.r.t injective Farey map.

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \;\cong\; \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

3. If exists, compute preimage w.r.t injective Farey map.
4. Verify correctness of lift.

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \cong \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

3. If exists, compute preimage w.r.t injective Farey map.
4. Verify correctness of lift.

This will yield correct result, provided

- N is large enough s.t. the $\mathbb{Q}$-result is in source of Farey map, and

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \;\cong\; \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

3. If exists, compute preimage w.r.t injective Farey map.
4. Verify correctness of lift.

This will yield correct result, provided

- N is large enough s.t. the $\mathbb{Q}$-result is in source of Farey map, and
- none of the p_i is bad.

Basic concept for modular computations

1. Compute result over $\mathbb{Z}/p_i$ for distinct primes $p_1, \ldots, p_r$.
2. For $N = p_1 \cdot \ldots \cdot p_r$ compute lift w.r.t Chinese remainder isomorphism

$$\mathbb{Z}/N \cong \mathbb{Z}/p_1 \times \ldots \times \mathbb{Z}/p_r$$

3. If exists, compute preimage w.r.t injective Farey map.
4. Verify correctness of lift.

This will yield correct result, provided

- N is large enough s.t. the $\mathbb{Q}$-result is in source of Farey map, and
- none of the p_i is bad.

Definition

A prime p is called **bad** if the result over $\mathbb{Q}$ does not reduce modulo p to the result over $\mathbb{Z}/p$.

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G.

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ *with* f_i *primitve,*

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ with f_i primitve, and

- *G is the reduced Gröbner basis of $\langle F \rangle \subset \mathbb{Q}[X]$,*

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ with f_i primitve, and

- *G is the reduced Gröbner basis of $\langle F \rangle \subset \mathbb{Q}[X]$,*
- *$G(p)$ is the reduced Gröbner basis of $\langle F_p \rangle$, and*

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ with f_i primitve, and

- *G is the reduced Gröbner basis of $\langle F \rangle \subset \mathbb{Q}[X]$,*
- *$G(p)$ is the reduced Gröbner basis of $\langle F_p \rangle$, and*
- *$G_{\mathbb{Z}}$ a minimal strong Gröbnerbasis of $\langle F \rangle \subset \mathbb{Z}[X]$.*

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ with f_i primitve, and

- *G is the reduced Gröbner basis of $\langle F \rangle \subset \mathbb{Q}[X]$,*
- *$G(p)$ is the reduced Gröbner basis of $\langle F_p \rangle$, and*
- *$G_{\mathbb{Z}}$ a minimal strong Gröbnerbasis of $\langle F \rangle \subset \mathbb{Z}[X]$. Then*

$$p \text{ does not divide any lead coefficient in } G_{\mathbb{Z}} \iff \mathrm{LM}\, G = \mathrm{LM}\, G(p) \iff G_p = G(p)$$

Bad primes in Gröbner basis computations

For $G \subset K[X] = K[x_1, \ldots, x_n]$ and a monomial ordering $>$, let $\mathrm{LM}(G)$ be the set of lead monomials of G. For $G \subset \mathbb{Z}[X]$ define

$$G_p := \overline{G} \subset \mathbb{Z}/p\,[X].$$

Theorem (Arnold, 2003)

Suppose $F = \{f_1, ..., f_r\} \subset \mathbb{Z}[X]$ with f_i primitve, and

- *G is the reduced Gröbner basis of $\langle F \rangle \subset \mathbb{Q}[X]$,*
- *$G(p)$ is the reduced Gröbner basis of $\langle F_p \rangle$, and*
- *$G_{\mathbb{Z}}$ a minimal strong Gröbnerbasis of $\langle F \rangle \subset \mathbb{Z}[X]$. Then*

$$p \text{ does not divide any lead coefficient in } G_{\mathbb{Z}} \iff \mathrm{LM}\, G = \mathrm{LM}\, G(p) \iff G_p = G(p)$$

that is, p is not bad.

Bad primes in Gröbner basis computations

Example

Let

$$f = x^5 + y^{11} + xy^9 + x^3y^9 \in \mathbb{Z}[x, y].$$

Bad primes in Gröbner basis computations

Example

Let
$$f = x^5 + y^{11} + xy^9 + x^3y^9 \in \mathbb{Z}[x, y].$$

Then $G_{\mathbb{Z}}$ for
$$\left\langle \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right\rangle$$

w.r.t *lp* is

$$264627y^{39} + \ldots,$$

$$12103947791971846719838321886393392913750065060875xy^8 - \ldots,$$

$$40754032969602177507873137664624218564815033875x^4 + \ldots.$$

Bad primes in Gröbner basis computations

Example

Let

$$f = x^5 + y^{11} + xy^9 + x^3y^9 \in \mathbb{Z}[x, y].$$

Then $G_{\mathbb{Z}}$ for

$$\left\langle \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right\rangle$$

w.r.t *lp* is

$$264627y^{39} + \ldots,$$

$$1210394779197184671983832188639339291375006506087 5xy^8 - \ldots,$$

$$40754032969602177507873137664624218564815033875x^4 + \ldots.$$

and $\mathrm{LM}\, G = \mathrm{LM}\, G(p)$ for all primes p except

$$p = 3, 5, 11, 809, 65179, 531264751, 431051934846786628615463393.$$

Bad primes

Classification of bad primes:

- Type 1: Input modulo p not valid (no problem)

Bad primes

Classification of bad primes:

- Type 1: Input modulo p not valid (no problem)
- Type 2: Failure in the course of the algorithm (e.g. matrix not invertible modulo p, wastes computation time if happens)

Bad primes

Classification of bad primes:

- Type 1: Input modulo p not valid (no problem)
- Type 2: Failure in the course of the algorithm (e.g. matrix not invertible modulo p, wastes computation time if happens)
- Type 3: Computable invariant with known expected value (e.g. Hilbert polynomial) is wrong (have to do expensive test for each prime, although set of bad primes usually is finite)

Bad primes

Classification of bad primes:

- Type 1: Input modulo p not valid (no problem)
- Type 2: Failure in the course of the algorithm (e.g. matrix not invertible modulo p, wastes computation time if happens)
- Type 3: Computable invariant with known expected value (e.g. Hilbert polynomial) is wrong (have to do expensive test for each prime, although set of bad primes usually is finite)
- Type 4: Computable invariant with unknown expected value (e.g. lead ideal in Gröbner basis computations) is wrong (to detect by a majority vote, have to compute and store value of invariant for all modular results)

Bad primes

Classification of bad primes:

- Type 1: Input modulo p not valid (no problem)
- Type 2: Failure in the course of the algorithm (e.g. matrix not invertible modulo p, wastes computation time if happens)
- Type 3: Computable invariant with known expected value (e.g. Hilbert polynomial) is wrong (have to do expensive test for each prime, although set of bad primes usually is finite)
- Type 4: Computable invariant with unknown expected value (e.g. lead ideal in Gröbner basis computations) is wrong (to detect by a majority vote, have to compute and store value of invariant for all modular results)
- Type 5: otherwise.

Example of type 5 bad prime

For ideal $I \subset \mathbb{Q}[X]$ and prime p define $I_p = (I \cap \mathbb{Z}[X])_p$.

Example

Consider the algorithm $I \mapsto \sqrt{I + \mathrm{Jac}(I)}$ for

$$I = \langle x^6 + y^6 + 7x^5 z + x^3 y^2 z - 31x^4 z^2 - 224x^3 z^3 + 244x^2 z^4 + 1632xz^5 + 576z^6 \rangle$$

Example of type 5 bad prime

For ideal $I \subset \mathbb{Q}[X]$ and prime p define $I_p = (I \cap \mathbb{Z}[X])_p$.

Example

Consider the algorithm $I \mapsto \sqrt{I + \mathrm{Jac}(I)}$ for

$$I = \langle x^6 + y^6 + 7x^5z + x^3y^2z - 31x^4z^2 - 224x^3z^3 + 244x^2z^4 + 1632xz^5 + 576z^6 \rangle$$

Then w.r.t dp $\qquad \mathrm{LM}(I) = \langle x^6 \rangle = \mathrm{LM}(I_5)$

Example of type 5 bad prime

For ideal $I \subset \mathbb{Q}[X]$ and prime p define $I_p = (I \cap \mathbb{Z}[X])_p$.

Example

Consider the algorithm $I \mapsto \sqrt{I + \mathrm{Jac}(I)}$ for

$$I = \langle x^6 + y^6 + 7x^5z + x^3y^2z - 31x^4z^2 - 224x^3z^3 + 244x^2z^4 + 1632xz^5 + 576z^6 \rangle$$

Then w.r.t dp $\qquad \mathrm{LM}(I) = \langle x^6 \rangle = \mathrm{LM}(I_5)$

$$U(0) = \sqrt{I + \mathrm{Jac}(I)} = \langle y, x - 4z \rangle \cap \langle y, x + 6z \rangle$$

$$U(5) = \sqrt{I_5 + \mathrm{Jac}(I_5)} = \langle y, x^2 - z^2 \rangle = \langle y, x - z \rangle \cap \langle y,\ x + z \rangle$$

Example of type 5 bad prime

For ideal $I \subset \mathbb{Q}[X]$ and prime p define $I_p = (I \cap \mathbb{Z}[X])_p$.

Example

Consider the algorithm $I \mapsto \sqrt{I + \mathrm{Jac}(I)}$ for

$$I = \langle x^6 + y^6 + 7x^5z + x^3y^2z - 31x^4z^2 - 224x^3z^3 + 244x^2z^4 + 1632xz^5 + 576z^6 \rangle$$

Then w.r.t dp $\qquad \mathrm{LM}(I) = \langle x^6 \rangle = \mathrm{LM}(I_5)$

$$U(0) = \sqrt{I + \mathrm{Jac}(I)} = \langle y, x - 4z \rangle \cap \langle y, x + 6z \rangle$$

$$U(5) = \sqrt{I_5 + \mathrm{Jac}(I_5)} = \langle y, x^2 - z^2 \rangle = \langle y, x - z \rangle \cap \langle y, \ x + z \rangle$$

$$U(0)_5 = \langle y, (x + z)^2 \rangle$$

Example of type 5 bad prime

For ideal $I \subset \mathbb{Q}[X]$ and prime p define $I_p = (I \cap \mathbb{Z}[X])_p$.

Example

Consider the algorithm $I \mapsto \sqrt{I + \mathrm{Jac}(I)}$ for

$$I = \langle x^6 + y^6 + 7x^5z + x^3y^2z - 31x^4z^2 - 224x^3z^3 + 244x^2z^4 + 1632xz^5 + 576z^6 \rangle$$

Then w.r.t dp $$\mathrm{LM}(I) = \langle x^6 \rangle = \mathrm{LM}(I_5)$$

$$U(0) = \sqrt{I + \mathrm{Jac}(I)} = \langle y, x - 4z \rangle \cap \langle y, x + 6z \rangle$$

$$U(5) = \sqrt{I_5 + \mathrm{Jac}(I_5)} = \langle y, x^2 - z^2 \rangle = \langle y, x - z \rangle \cap \langle y,\ x + z \rangle$$

$$U(0)_5 = \langle y, (x + z)^2 \rangle$$

Hence

$$U(0)_5 \neq U(5)$$

$$\mathrm{LM}(U(0)) = \langle y, x^2 \rangle = \mathrm{LM}(U(5))$$

Error tolerant reconstruction

Goal: Reconstruct $\frac{a}{b}$ from $\overline{r} \in \mathbb{Z}/N$ in the presence of bad primes.

Error tolerant reconstruction

Goal: Reconstruct $\frac{a}{b}$ from $\overline{r} \in \mathbb{Z}/N$ in the presence of bad primes.

Idea: Find (x, y) with $\frac{x}{y} = \frac{a}{b}$ in the lattice

$$\Lambda = \langle (N, 0), (r, 1) \rangle \subset \mathbb{Z}^2$$

Error tolerant reconstruction

Goal: Reconstruct $\frac{a}{b}$ from $\overline{r} \in \mathbb{Z}/N$ in the presence of bad primes.

Idea: Find (x, y) with $\frac{x}{y} = \frac{a}{b}$ in the lattice

$$\Lambda = \langle (N, 0), (r, 1) \rangle \subset \mathbb{Z}^2$$

Lemma (BDFP, 2015)

All $(x, y) \in \Lambda$ with $x^2 + y^2 < N$ are collinear.

Error tolerant reconstruction

Goal: Reconstruct $\frac{a}{b}$ from $\overline{r} \in \mathbb{Z}/N$ in the presence of bad primes.

Idea: Find (x, y) with $\frac{x}{y} = \frac{a}{b}$ in the lattice

$$\Lambda = \langle (N, 0), (r, 1) \rangle \subset \mathbb{Z}^2$$

Lemma (BDFP, 2015)

All $(x, y) \in \Lambda$ with $x^2 + y^2 < N$ are collinear.

Proof.

Let $\lambda = (x, y)$, $\mu = (c, d) \in \Lambda$ with $x^2 + y^2, c^2 + d^2 < N$. Then $y\mu - d\lambda = (yc - xd, 0) \in \Lambda$, so $N | (yc - xd)$. By Cauchy–Schwarz $|yc - xd| < N$, hence $yc = xd$. □

Error tolerant reconstruction

Goal: Reconstruct $\frac{a}{b}$ from $\overline{r} \in \mathbb{Z}/N$ in the presence of bad primes.

Idea: Find (x, y) with $\frac{x}{y} = \frac{a}{b}$ in the lattice

$$\Lambda = \langle (N, 0), (r, 1) \rangle \subset \mathbb{Z}^2$$

Lemma (BDFP, 2015)

All $(x, y) \in \Lambda$ *with* $x^2 + y^2 < N$ *are collinear.*

Proof.

Let $\lambda = (x, y)$, $\mu = (c, d) \in \Lambda$ with $x^2 + y^2, c^2 + d^2 < N$. Then $y\mu - d\lambda = (yc - xd, 0) \in \Lambda$, so $N | (yc - xd)$. By Cauchy–Schwarz $|yc - xd| < N$, hence $yc = xd$. □

Now suppose

$$N = N' \cdot M$$

with $\gcd(N', M) = 1$.

Error tolerant reconstruction

Think of N' as the product of the good primes with correct result $\overline{s}$, and of M as the product of the bad primes with wrong result $\overline{t}$.

Error tolerant reconstruction

Think of N' as the product of the good primes with correct result $\overline{s}$, and of M as the product of the bad primes with wrong result $\overline{t}$.

Theorem (BDFP, 2015)

If

$$\overline{r} \mapsto (\overline{s}, \overline{t}) \quad \textit{with respect to} \quad \mathbb{Z}/N \cong \mathbb{Z}/N' \times \mathbb{Z}/M$$

and

$$\frac{a}{b} \equiv s \bmod N'$$

Error tolerant reconstruction

Think of N' as the product of the good primes with correct result $\overline{s}$, and of M as the product of the bad primes with wrong result $\overline{t}$.

Theorem (BDFP, 2015)

If

$$\overline{r} \mapsto (\overline{s}, \overline{t}) \quad \textit{with respect to} \quad \mathbb{Z}/N \cong \mathbb{Z}/N' \times \mathbb{Z}/M$$

and

$$\frac{a}{b} \equiv s \bmod N'$$

then $(aM, bM) \in \Lambda$.

Error tolerant reconstruction

Think of N' as the product of the good primes with correct result $\overline{s}$, and of M as the product of the bad primes with wrong result $\overline{t}$.

Theorem (BDFP, 2015)

If

$$\overline{r} \mapsto (\overline{s}, \overline{t}) \quad \text{with respect to} \quad \mathbb{Z}/N \cong \mathbb{Z}/N' \times \mathbb{Z}/M$$

and

$$\frac{a}{b} \equiv s \bmod N'$$

then $(aM, bM) \in \Lambda$. *So if*

$$(a^2 + b^2)M < N',$$

then (by the lemma)

$$\frac{x}{y} = \frac{a}{b} \quad \text{for all } (x, y) \in \Lambda \text{ with } (x^2 + y^2) < N$$

and such vectors exist.

Error tolerant reconstruction

Think of N' as the product of the good primes with correct result $\overline{s}$, and of M as the product of the bad primes with wrong result $\overline{t}$.

Theorem (BDFP, 2015)

If

$$\overline{r} \mapsto (\overline{s}, \overline{t}) \quad \textit{with respect to} \quad \mathbb{Z}/N \cong \mathbb{Z}/N' \times \mathbb{Z}/M$$

and

$$\frac{a}{b} \equiv s \bmod N'$$

then $(aM, bM) \in \Lambda$. *So if*

$$(a^2 + b^2)M < N',$$

then (by the lemma)

$$\frac{x}{y} = \frac{a}{b} \quad \textit{for all } (x, y) \in \Lambda \textit{ with } (x^2 + y^2) < N$$

and such vectors exist. Moreover, if $\gcd(a, b) = 1$ *and* (x, y) *is a shortest vector* $\neq 0$ *in* Λ, *we also have* $\gcd(x, y) | M$.

Error tolerant reconstruction via Gauss-Lagrange

Hence, if $N' \gg M$, the Gauss-Lagrange-Algorithm for finding a shortest vector $(x, y) \in \Lambda$ gives $\frac{a}{b}$ independently of t, provided $x^2 + y^2 < N$.

Error tolerant reconstruction via Gauss-Lagrange

Hence, if $N' \gg M$, the Gauss-Lagrange-Algorithm for finding a shortest vector $(x, y) \in \Lambda$ gives $\frac{a}{b}$ independently of t, provided $x^2 + y^2 < N$.

Algorithm (Error tolerant reconstruction)

Input: *N and r.*

Output: *$\frac{a}{b}$ or* `false`.

$(a_0, b_0) := (N, 0)$, $(a_1, b_1) := (r, 1)$, $i := -1$

repeat

  $i = i + 1$

  $(a_{i+2}, b_{i+2}) = (a_i, b_i) - \left\lfloor \frac{\langle (a_i, b_i), (a_{i+1}, b_{i+1}) \rangle}{\| (a_{i+1}, b_{i+1}) \|^2} \right\rceil (a_{i+1}, b_{i+1})$

until $a_{i+2}^2 + b_{i+2}^2 \geq a_{i+1}^2 + b_{i+1}^2$

if $a_{i+1}^2 + b_{i+1}^2 < N$ **then**

  return $\frac{a_{i+1}}{b_{i+1}}$

else

  return `false`

Reconstruction via Gauss-Lagrange

Example

We reconstruct $\frac{13}{12}$ from

$$\overline{22684} \in \mathbb{Z}/38885$$

by determining a shortest vector in the lattice

$$\langle (38885, 0), (22684, 1) \rangle \subset \mathbb{Z}^2$$

via Gauss-Lagrange

Reconstruction via Gauss-Lagrange

Example

We reconstruct $\frac{13}{12}$ from

$$\overline{22684} \in \mathbb{Z}/38885$$

by determining a shortest vector in the lattice

$$\langle (38885, 0), (22684, 1) \rangle \subset \mathbb{Z}^2$$

via Gauss-Lagrange

$$\begin{aligned} (38885, 0) &= 2 \cdot (22684, 1) + (-6483, -2), \\ (22684, 1) &= -3 \cdot (-6483, -2) + (3235, -5), \\ (-6483, -2) &= 2 \cdot (3235, -5) + (-13, -12), \\ (3235, -5) &= -134 \cdot (-13, -12) + (1493, -1613). \end{aligned}$$

Reconstruction via Gauss-Lagrange

Example

Now introduce an error in the modular results:

$$\begin{array}{ccccccccccc} \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ (\,\overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\,) & \mapsto & \overline{22684} \end{array}$$

Reconstruction via Gauss-Lagrange

Example

Now introduce an error in the modular results:

$$\begin{array}{ccccccccc} \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ (\ \overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\ ) & \mapsto & \overline{22684} \\ (\ \overline{4} & , & \color{red}{\overline{2}} & , & \overline{2} & & \overline{60}\ ) & \mapsto & \overline{464} \end{array}$$

Reconstruction via Gauss-Lagrange

Example

Now introduce an error in the modular results:

$$\begin{array}{ccccccccccc} \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ (\,\overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\,) & \mapsto & \overline{22684} \\ (\,\overline{4} & , & \color{red}{\overline{2}} & , & \overline{2} & & \overline{60}\,) & \mapsto & \overline{464} \end{array}$$

Error tolerant reconstruction computes

$$\begin{aligned} (38885, 0) &= 84 \cdot (464, 1) + (-91, -84), \\ (464, 1) &= -3 \cdot (-91, -84) + (191, -251) \end{aligned}$$

Reconstruction via Gauss-Lagrange

Example

Now introduce an error in the modular results:

$$\begin{array}{ccccccccccc} \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ (\,\overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\,) & \mapsto & \overline{22684} \\ (\,\overline{4} & , & \color{red}{\overline{2}} & , & \overline{2} & & \overline{60}\,) & \mapsto & \overline{464} \end{array}$$

Error tolerant reconstruction computes

$$\begin{aligned} (38885, 0) &= 84 \cdot (464, 1) + (-91, -84), \\ (464, 1) &= -3 \cdot (-91, -84) + (191, -251) \end{aligned}$$

hence yields

$$\frac{91}{84} = \frac{7 \cdot 13}{7 \cdot 12} = \frac{13}{12}.$$

Reconstruction via Gauss-Lagrange

Example

Now introduce an error in the modular results:

$$\begin{array}{ccccccccccc} \mathbb{Z}/5 & \times & \mathbb{Z}/7 & \times & \mathbb{Z}/11 & \times & \mathbb{Z}/101 & \cong & \mathbb{Z}/38885 \\ (\,\overline{4} & , & \overline{4} & , & \overline{2} & , & \overline{60}\,) & \mapsto & \overline{22684} \\ (\,\overline{4} & , & \color{red}{\overline{2}} & , & \overline{2} & & \overline{60}\,) & \mapsto & \overline{464} \end{array}$$

Error tolerant reconstruction computes

$$\begin{aligned} (38885, 0) &= 84 \cdot (464, 1) + (-91, -84), \\ (464, 1) &= -3 \cdot (-91, -84) + (191, -251) \end{aligned}$$

hence yields

$$\frac{91}{84} = \frac{7 \cdot 13}{7 \cdot 12} = \frac{13}{12}.$$

Note that

$$(13^2 + 12^2) \cdot 7 = 2191 < 5555 = 5 \cdot 11 \cdot 101.$$

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*
- *For $N = \prod_{p \in \mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\operatorname{LM}(U(p))$.*
- *For $N = \prod_{p \in \mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*
- *Lift $U(N)$ by error tolerant rational reconstruction to U.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*
- *For $N = \prod_{p \in \mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*
- *Lift $U(N)$ by error tolerant rational reconstruction to U.*
- *Test $U_p = U(p)$ for random prime p.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*
- *For $N = \prod_{p \in \mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*
- *Lift $U(N)$ by error tolerant rational reconstruction to U.*
- *Test $U_p = U(p)$ for random prime p.*
- *Verify $U = U(0)$.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*
- *For $N = \prod_{p \in \mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*
- *Lift $U(N)$ by error tolerant rational reconstruction to U.*
- *Test $U_p = U(p)$ for random prime p.*
- *Verify $U = U(0)$.*
- *If lift, test or verification fails, then enlarge $\mathcal{P}$.*

General reconstruction scheme

Setup: For ideal $I \subset \mathbb{Q}[X]$ compute ideal (or module) $U(0)$ associated to I by deterministic algorithm.

Algorithm

- *For I_p compute result $U(p)$ over $\mathbb{Z}/p$ for p in finite set of primes $\mathcal{P}$.*
- *Reduce $\mathcal{P}$ according to majority vote on $\mathrm{LM}(U(p))$.*
- *For $N = \prod_{p \in \mathcal{P}} p$ compute termwise CRT–lift $U(N)$ to $\mathbb{Z}/N$.*
- *Lift $U(N)$ by error tolerant rational reconstruction to U.*
- *Test $U_p = U(p)$ for random prime p.*
- *Verify $U = U(0)$.*
- *If lift, test or verification fails, then enlarge $\mathcal{P}$.*

Theorem (BDFP, 2015)

If the bad primes form a Zariski closed true subset of $\operatorname{Spec} \mathbb{Z}$, then this algorithm terminates with the correct result.

Normalization

Setup: $A = K[X]/I$ domain.

Definition

The **normalization** $\overline{A}$ of A is the integral closure of A in its quotient field $Q(A)$.

Normalization

Setup: $A = K[X]/I$ domain.

Definition

The **normalization** $\overline{A}$ of A is the integral closure of A in its quotient field $Q(A)$. We call A **normal** if $A = \overline{A}$.

Normalization

Setup: $A = K[X]/I$ domain.

Definition

The **normalization** $\overline{A}$ of A is the integral closure of A in its quotient field $Q(A)$. We call A **normal** if $A = \overline{A}$.

Theorem (Noether)

$\overline{A}$ is a finitely generated A-module.

Normalization

Setup: $A = K[X]/I$ domain.

Definition

The **normalization** $\overline{A}$ of A is the integral closure of A in its quotient field $Q(A)$. We call A **normal** if $A = \overline{A}$.

Theorem (Noether)

$\overline{A}$ is a finitely generated A-module.

Example

Curve $I = \langle x^3 + x^2 - y^2 \rangle \subset K[x, y]$

$$\begin{array}{rcccc} A = K[x,y]/I & \cong & K[t^2-1, t^3-t] & \subset & K[t] \cong \overline{A} \\ \overline{x} & \mapsto & t^2 - 1 & & \\ \overline{y} & \mapsto & t^3 - t & & \end{array}$$

Normalization

Setup: $A = K[X]/I$ domain.

Definition

The **normalization** $\overline{A}$ of A is the integral closure of A in its quotient field $Q(A)$. We call A **normal** if $A = \overline{A}$.

Theorem (Noether)

$\overline{A}$ is a finitely generated A-module.

Example

Curve $I = \langle x^3 + x^2 - y^2 \rangle \subset K[x, y]$

$$\begin{array}{rcl} A = K[x,y]/I & \cong & K[t^2-1, t^3-t] \quad \subset \quad K[t] \cong \overline{A} \\ \overline{x} & \mapsto & t^2 - 1 \\ \overline{y} & \mapsto & t^3 - t \end{array}$$

As an A-module $\overline{A} = \left\langle 1, \frac{\overline{y}}{\overline{x}} \right\rangle$.

Normalization

Lemma

If $J \subset A$ *is an ideal and* $0 \neq g \in J$*, then*

Normalization

Lemma

If $J \subset A$ is an ideal and $0 \neq g \in J$, then

$$\begin{array}{ccccccccc} A & \hookrightarrow & \mathrm{Hom}_A(J,J) & \cong & \frac{1}{g}(gJ :_A J) & \subset & \overline{A} \\ a & \mapsto & a\cdot & & & & \\ & & & \varphi & \mapsto & \frac{\varphi(g)}{g} & \end{array}$$

Normalization

Lemma

If $J \subset A$ is an ideal and $0 \neq g \in J$, then

$$\begin{array}{ccccccc} A & \hookrightarrow & \mathrm{Hom}_A(J,J) & \cong & \frac{1}{g}(gJ :_A J) & \subset & \overline{A} \\ a & \mapsto & a\cdot & & & & \\ & & & \varphi & \mapsto & \frac{\varphi(g)}{g} & \end{array}$$

Algorithm

Starting from $A_0 = A$ and $J_0 = J$,

Normalization

Lemma

If $J \subset A$ is an ideal and $0 \neq g \in J$, then

$$\begin{array}{ccccccc} A & \hookrightarrow & \mathrm{Hom}_A(J,J) & \cong & \frac{1}{g}(gJ :_A J) & \subset & \overline{A} \\ a & \mapsto & a\cdot & & & & \\ & & & \varphi & \mapsto & \frac{\varphi(g)}{g} & \end{array}$$

Algorithm

Starting from $A_0 = A$ and $J_0 = J$, setting

$$A_{i+1} = \frac{1}{g}(gJ_i :_{A_i} J_i) \qquad J_i = \sqrt{JA_i}$$

Normalization

Lemma

If $J \subset A$ is an ideal and $0 \neq g \in J$, then

$$\begin{array}{ccccccc} A & \hookrightarrow & \mathrm{Hom}_A(J,J) & \cong & \frac{1}{g}(gJ :_A J) & \subset & \overline{A} \\ a & \mapsto & a\cdot & & & & \\ & & \varphi & \mapsto & \frac{\varphi(g)}{g} & & \end{array}$$

Algorithm

Starting from $A_0 = A$ and $J_0 = J$, setting

$$A_{i+1} = \frac{1}{g}(gJ_i :_{A_i} J_i) \qquad J_i = \sqrt{JA_i}$$

we get a chain of extensions of reduced Noetherian rings

$$A = A_0 \subset \cdots \subset A_i \subset \cdots \subset A_m = A_{m+1}.$$

Terminates since A is Noetherian.

Grauert-Remmert criterion

Non-normal locus $N(A)$ is contained in **singular locus** $\mathrm{Sing}(A)$.

Grauert-Remmert criterion

Non-normal locus $N(A)$ is contained in **singular locus** $\mathrm{Sing}(A)$.

Theorem (Grauert-Remmert)

Let $0 \neq J \subset A$ *be an ideal with* $J = \sqrt{J}$

Grauert-Remmert criterion

Non-normal locus $N(A)$ is contained in **singular locus** $\mathrm{Sing}(A)$.

Theorem (Grauert-Remmert)

Let $0 \neq J \subset A$ *be an ideal with* $J = \sqrt{J}$ *and*

$$N(A) \subset V(J).$$

Grauert-Remmert criterion

Non-normal locus $N(A)$ is contained in **singular locus** $\operatorname{Sing}(A)$.

Theorem (Grauert-Remmert)

Let $0 \neq J \subset A$ *be an ideal with* $J = \sqrt{J}$ *and*

$$N(A) \subset V(J).$$

Then A *is normal iff the inclusion*

$$\begin{array}{ccc} A & \hookrightarrow & \operatorname{Hom}_A(J,J) \\ a & \mapsto & a\cdot \end{array}$$

is an isomorphism.

Grauert-Remmert criterion

Non-normal locus $N(A)$ is contained in **singular locus** $\mathrm{Sing}(A)$.

Theorem (Grauert-Remmert)

Let $0 \neq J \subset A$ *be an ideal with* $J = \sqrt{J}$ *and*

$$N(A) \subset V(J).$$

Then A *is normal iff the inclusion*

$$\begin{array}{ccc} A & \hookrightarrow & \mathrm{Hom}_A(J,J) \\ a & \mapsto & a\cdot \end{array}$$

is an isomorphism.

$\Longrightarrow$ For $J = \sqrt{\mathrm{Jac}(I)}$ algorithm terminates with $A_m = A_{m+1} = \overline{A}$,

Grauert-Remmert criterion

Non-normal locus $N(A)$ is contained in **singular locus** $\operatorname{Sing}(A)$.

Theorem (Grauert-Remmert)

Let $0 \neq J \subset A$ *be an ideal with* $J = \sqrt{J}$ *and*

$$N(A) \subset V(J).$$

Then A *is normal iff the inclusion*

$$\begin{array}{ccc} A & \hookrightarrow & \operatorname{Hom}_A(J,J) \\ a & \mapsto & a\cdot \end{array}$$

is an isomorphism.

$\Longrightarrow$ For $J = \sqrt{\operatorname{Jac}(I)}$ algorithm terminates with $A_m = A_{m+1} = \overline{A}$, since:

Lemma

$N(A_i) \subset V(\sqrt{JA_i})$

Local Techniques for Normalization

Theorem (BDLPSS, 2011)

Suppose

$$\mathrm{Sing}(A) = \{P_1, \ldots, P_r\}$$

Local Techniques for Normalization

Theorem (BDLPSS, 2011)

Suppose

$$\mathrm{Sing}(A) = \{P_1, \ldots, P_r\}$$

and

$$A \subset B_i \subset \overline{A}$$

is the ring given by the normalization algorithm applied to P_i *instead of* J

Local Techniques for Normalization

Theorem (BDLPSS, 2011)

Suppose

$$\mathrm{Sing}(A) = \{P_1, \ldots, P_r\}$$

and

$$A \subset B_i \subset \overline{A}$$

is the ring given by the normalization algorithm applied to P_i instead of J. Then

$$(B_i)_{P_i} = \overline{A_{P_i}}$$
$$(B_i)_Q = A_Q \text{ for all } P_i \neq Q \in \operatorname{Spec} A,$$

Local Techniques for Normalization

Theorem (BDLPSS, 2011)

Suppose

$$\operatorname{Sing}(A) = \{P_1, \ldots, P_r\}$$

and

$$A \subset B_i \subset \overline{A}$$

is the ring given by the normalization algorithm applied to P_i *instead of* J. *Then*

$$(B_i)_{P_i} = \overline{A_{P_i}}$$
$$(B_i)_Q = A_Q \text{ for all } P_i \neq Q \in \operatorname{Spec} A,$$

and

$$\overline{A} = B_1 + \ldots + B_r.$$

We call B_i *the* **minimal local contribution** *to* $\overline{A}$ *at* P_i.

Adjoint ideals

Setup: $\Gamma \subset \mathbb{P}^r$ integral, non-degenerate projective curve, $\pi : \overline{\Gamma} \to \Gamma$ normalization map, $I(\Gamma) \subsetneqq I \subset k[x_0, ..., x_r]$ saturated homogeneous ideal.

Adjoint ideals

Setup: $\Gamma \subset \mathbb{P}^r$ integral, non-degenerate projective curve, $\pi : \overline{\Gamma} \to \Gamma$ normalization map, $I(\Gamma) \subsetneqq I \subset k[x_0, ..., x_r]$ saturated homogeneous ideal. Let H be pullback of hyperplane, $\Delta(I)$ pullback of $\operatorname{Proj}(S/I)$.

Adjoint ideals

Setup: $\Gamma \subset \mathbb{P}^r$ integral, non-degenerate projective curve, $\pi : \overline{\Gamma} \to \Gamma$ normalization map, $I(\Gamma) \subsetneq I \subset k[x_0, ..., x_r]$ saturated homogeneous ideal. Let H be pullback of hyperplane, $\Delta(I)$ pullback of $\mathrm{Proj}(S/I)$. Then

$$0 \to \widetilde{I}\mathcal{O}_\Gamma \to \pi_*(\widetilde{I}\mathcal{O}_{\overline{\Gamma}}) \to \mathcal{F} \to 0$$

Adjoint ideals

Setup: $\Gamma \subset \mathbb{P}^r$ integral, non-degenerate projective curve, $\pi : \overline{\Gamma} \to \Gamma$ normalization map, $I(\Gamma) \subsetneqq I \subset k[x_0, ..., x_r]$ saturated homogeneous ideal. Let H be pullback of hyperplane, $\Delta(I)$ pullback of $\mathrm{Proj}(S/I)$. Then

$$0 \to \widetilde{I}\mathcal{O}_\Gamma \to \pi_*(\widetilde{I}\mathcal{O}_{\overline{\Gamma}}) \to \mathcal{F} \to 0$$

gives for $m \gg 0$ linear maps

$$0 \to I_m / I(\Gamma)_m \overset{\overline{\varrho_m}}{\to} H^0\left(\overline{\Gamma}, \mathcal{O}_{\overline{\Gamma}}(mH - \Delta(I))\right) \to H^0(\Gamma, \mathcal{F}) \to 0$$

Definition

I is an **adjoint ideal** of Γ if $\overline{\varrho_m}$ surjective for $m \gg 0$.

Adjoint ideals

Setup: $\Gamma \subset \mathbb{P}^r$ integral, non-degenerate projective curve, $\pi : \overline{\Gamma} \to \Gamma$ normalization map, $I(\Gamma) \subsetneq I \subset k[x_0, ..., x_r]$ saturated homogeneous ideal. Let H be pullback of hyperplane, $\Delta(I)$ pullback of $\mathrm{Proj}(S/I)$. Then

$$0 \to \widetilde{I}\mathcal{O}_\Gamma \to \pi_*(\widetilde{I}\mathcal{O}_{\overline{\Gamma}}) \to \mathcal{F} \to 0$$

gives for $m \gg 0$ linear maps

$$0 \to I_m / I(\Gamma)_m \overset{\overline{\varrho_m}}{\to} H^0\left(\overline{\Gamma}, \mathcal{O}_{\overline{\Gamma}}\left(mH - \Delta(I)\right)\right) \to H^0\left(\Gamma, \mathcal{F}\right) \to 0$$

Definition

I is an **adjoint ideal** of Γ if $\overline{\varrho_m}$ surjective for $m \gg 0$.

$$h^0\left(\Gamma, \mathcal{F}\right) = \sum_{P \in \mathrm{Sing}(\Gamma)} \ell(I_P \overline{\mathcal{O}_{\Gamma,P}} / I_P) \qquad \Longrightarrow$$

Theorem (Arbarello, Ciliberto, 1983)

$$I \textit{ adjoint} \iff I_P\overline{\mathcal{O}_{\Gamma,P}} = I_P \textit{ for all } P \in \mathrm{Sing}(\Gamma).$$

Conductor is largest ideal with this property.

Adjoint ideals

Definition

Gorenstein adjoint ideal is the unique largest homogeneous ideal $\mathfrak{G} \subset K[x_0, \ldots, x_r]$ with

$$\mathfrak{G}_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}} \text{ for all } P \in \operatorname{Sing}(\Gamma).$$

Adjoint ideals

Definition

Gorenstein adjoint ideal is the unique largest homogeneous ideal $\mathfrak{G} \subset K[x_0, \ldots, x_r]$ with

$$\mathfrak{G}_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}} \text{ for all } P \in \operatorname{Sing}(\Gamma).$$

Applications:

Example

If Γ is plane curve of degree n, then $\mathfrak{G}_{n-3}$ cuts out canonical linear series.

Adjoint ideals

Definition

Gorenstein adjoint ideal is the unique largest homogeneous ideal $\mathfrak{G} \subset K[x_0, \ldots, x_r]$ with

$$\mathfrak{G}_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}} \text{ for all } P \in \operatorname{Sing}(\Gamma).$$

Applications:

Example

If Γ is plane curve of degree n, then $\mathfrak{G}_{n-3}$ cuts out canonical linear series.

Example

If Γ is plane rational of degree n then $\mathfrak{G}_{n-2}$ maps Γ to rational normal curve of degree $n-2$ in $\mathbb{P}^{n-2}$.

Adjoint ideals

Definition

Gorenstein adjoint ideal is the unique largest homogeneous ideal $\mathfrak{G} \subset K[x_0, \ldots, x_r]$ with

$$\mathfrak{G}_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}} \text{ for all } P \in \mathrm{Sing}(\Gamma).$$

Applications:

Example

If Γ is plane curve of degree n, then $\mathfrak{G}_{n-3}$ cuts out canonical linear series.

Example

If Γ is plane rational of degree n then $\mathfrak{G}_{n-2}$ maps Γ to rational normal curve of degree $n-2$ in $\mathbb{P}^{n-2}$.

Example

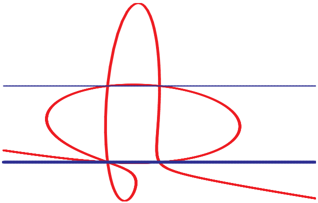
Brill-Noether-Algorithm for computing Riemann-Roch spaces.

Example

Minimal generators of $\mathfrak{G}$ for rational curve of degree 5:

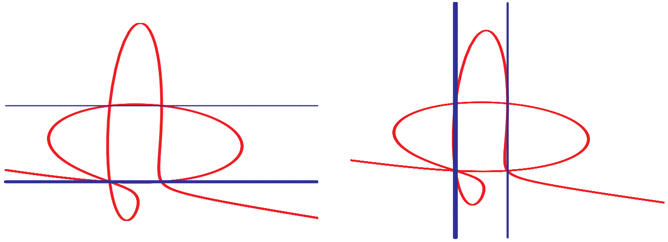
Example

Minimal generators of $\mathfrak{G}$ for rational curve of degree 5:



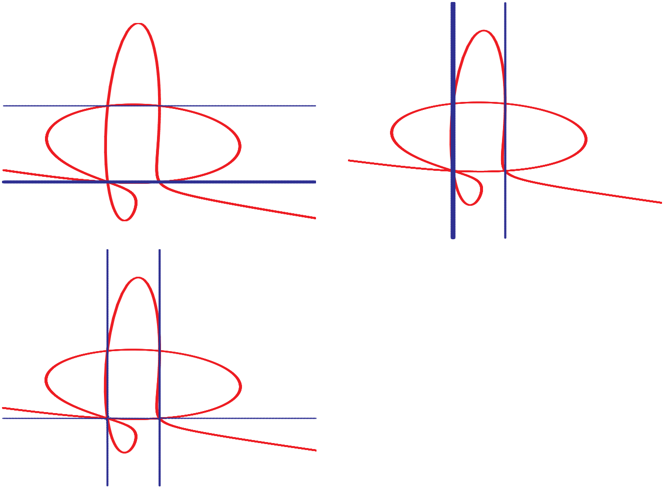
Example

Minimal generators of $\mathfrak{G}$ for rational curve of degree 5:



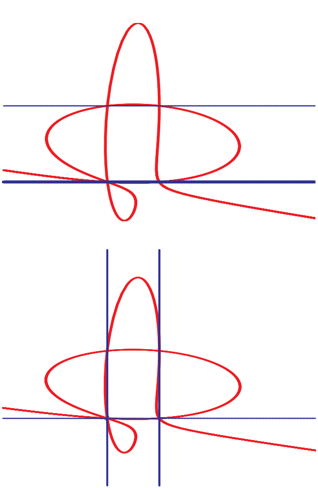
Example

Minimal generators of $\mathfrak{G}$ for rational curve of degree 5:

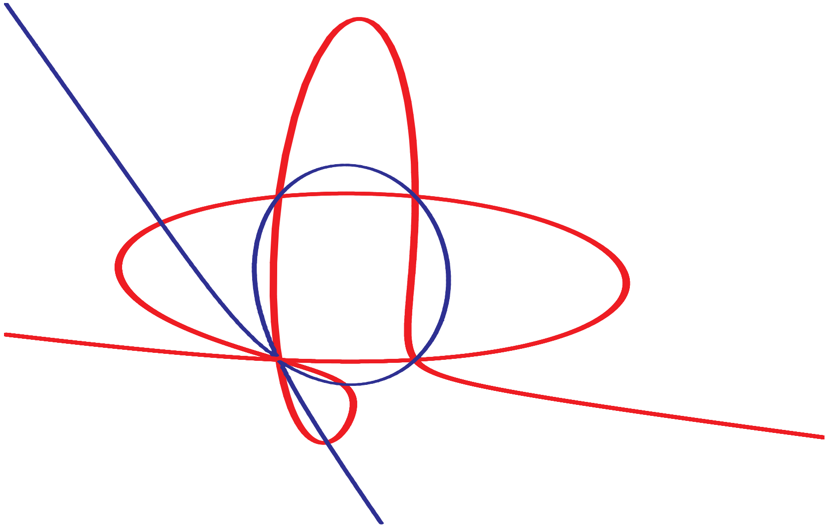


Example

Minimal generators of $\mathfrak{G}$ for rational curve of degree 5:



Example



Local-to-global algorithm

Definition

The **local adjoint ideal** of Γ at $P \in \operatorname{Sing}\Gamma$ is the largest homogeneous ideal $\mathfrak{G}(P) \subset k[x_0, \ldots, x_r]$ with

$$\mathfrak{G}(P)_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}}$$

Local-to-global algorithm

Definition

The **local adjoint ideal** of Γ at $P \in \operatorname{Sing}\Gamma$ is the largest homogeneous ideal $\mathfrak{G}(P) \subset k[x_0, \ldots, x_r]$ with

$$\mathfrak{G}(P)_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}}$$

Lemma (BDLP, 2015)

$$\mathfrak{G} = \bigcap_{P \in \operatorname{Sing}\Gamma} \mathfrak{G}(P)$$

The $\mathfrak{G}(P)$ can be computed in parallel via normalization.

Local-to-global algorithm

Definition

The **local adjoint ideal** of Γ at $P \in \operatorname{Sing}\Gamma$ is the largest homogeneous ideal $\mathfrak{G}(P) \subset k[x_0, \ldots, x_r]$ with

$$\mathfrak{G}(P)_P = \mathcal{C}_{\mathcal{O}_{\Gamma,P}}$$

Lemma (BDLP, 2015)

$$\mathfrak{G} = \bigcap_{P \in \operatorname{Sing}\Gamma} \mathfrak{G}(P)$$

The $\mathfrak{G}(P)$ can be computed in parallel via normalization.

Algorithm (BDLP, 2015)

If $\frac{1}{d}U$ is the minimal local contribution at P then

$$\mathfrak{G}(P) = (d : U)^h$$

Special types of singularities

If $\Gamma \subset \mathbb{P}^2$ has a singularity of type A_n at $P = (0:0:1)$, then given by

$$f = T^2 + W^{n+1} \quad \text{with} \quad T, W \in \mathbb{C}[[x, y]].$$

Special types of singularities

If $\Gamma \subset \mathbb{P}^2$ has a singularity of type A_n at $P = (0:0:1)$, then given by

$$f = T^2 + W^{n+1} \quad \text{with} \quad T, W \in \mathbb{C}[[x, y]].$$

Compute $T_j = T + O(j+1)$ inductively.

Special types of singularities

If $\Gamma \subset \mathbb{P}^2$ has a singularity of type A_n at $P = (0:0:1)$, then given by

$$f = T^2 + W^{n+1} \quad \text{with} \quad T, W \in \mathbb{C}[[x, y]].$$

Compute $T_j = T + O(j+1)$ inductively.

Lemma

If $P = (0,0)$ is of type A_n and $s = \left\lfloor \frac{n+1}{2} \right\rfloor$, then

$$\mathfrak{G}(P) = \langle x^s,\ T_{s-1},\ y^s \rangle^h \subset \mathbb{C}[x, y, z]$$

Special types of singularities

If $\Gamma \subset \mathbb{P}^2$ has a singularity of type A_n at $P = (0:0:1)$, then given by

$$f = T^2 + W^{n+1} \quad \text{with} \quad T, W \in \mathbb{C}[[x,y]].$$

Compute $T_j = T + O(j+1)$ inductively.

Lemma

If $P = (0,0)$ is of type A_n and $s = \left\lfloor \frac{n+1}{2} \right\rfloor$, then

$$\mathfrak{G}(P) = \left\langle x^s,\ T_{s-1},\ y^s \right\rangle^h \subset \mathbb{C}[x,y,z]$$

Similar results for D_n, E_n and other singularities in Arnold's list.

Example

$f = x^4 - y^2 + x^5$ with A_3 singularity. Then $\mathfrak{G}(P) = \left\langle x^2, y \right\rangle$.

Modular version of the algorithm

Applying the general modular strategy gives two-fold parallel algorithm.

Modular version of the algorithm

Applying the general modular strategy gives two-fold parallel algorithm.
Use primes p such that algorithm is applicable to Γ_p defined by $I(\Gamma)_p$.

Modular version of the algorithm

Applying the general modular strategy gives two-fold parallel algorithm.
Use primes p such that algorithm is applicable to Γ_p defined by $I(\Gamma)_p$.
Verification?

Modular version of the algorithm

Applying the general modular strategy gives two-fold parallel algorithm.
Use primes p such that algorithm is applicable to Γ_p defined by $I(\Gamma)_p$.
Verification?

Theorem (Arbarello, Ciliberto, 1983, Chiarli, 1984)

Let $I(\Gamma) \subsetneq I \subset k[x_0, \ldots, x_r]$ be saturated homogeneous. Then

$$\deg \Delta(I) \leq \deg I + \delta(\Gamma),$$

Modular version of the algorithm

Applying the general modular strategy gives two-fold parallel algorithm.
Use primes p such that algorithm is applicable to Γ_p defined by $I(\Gamma)_p$.
Verification?

Theorem (Arbarello, Ciliberto, 1983, Chiarli, 1984)

Let $I(\Gamma) \subsetneq I \subset k[x_0, \ldots, x_r]$ be saturated homogeneous. Then

$$\deg \Delta(I) \leq \deg I + \delta(\Gamma),$$

and I is an adjoint ideal of Γ iff

$$\deg \Delta(I) = \deg I + \delta(\Gamma).$$

Modular version of the algorithm

Applying the general modular strategy gives two-fold parallel algorithm.
Use primes p such that algorithm is applicable to Γ_p defined by $I(\Gamma)_p$.
Verification?

Theorem (Arbarello, Ciliberto, 1983, Chiarli, 1984)

Let $I(\Gamma) \subsetneqq I \subset k[x_0, \ldots, x_r]$ be saturated homogeneous. Then

$$\deg \Delta(I) \leq \deg I + \delta(\Gamma),$$

and I is an adjoint ideal of Γ iff

$$\deg \Delta(I) = \deg I + \delta(\Gamma).$$

Theorem (BDLP, 2015, corollary to Lipman, 2006)

$$\delta(\Gamma) \leq \delta(\Gamma_p)$$

and δ-constant flat family admits a simultaneous normalization.

Verification

$\widetilde{d}(g) = \deg(\text{divisor cut out by } g \text{ away from } \operatorname{Sing}(\Gamma))$.

Verification

$\widetilde{d}(g) = \deg(\text{divisor cut out by } g \text{ away from } \mathrm{Sing}(\Gamma))$.

Theorem (BDLP, 2015)

Let $I \subset k[x_0, \ldots, x_r]$ be saturated homogeneous with $I(\Gamma) \subsetneq I$ and suppose G is reduced Gröbner basis of I. If p is a prime and $g \in I$ is homogeneous of degree m such that

1. $\mathrm{LM}(I(\Gamma_p)) = \mathrm{LM}(I(\Gamma))$

Verification

$\widetilde{d}(g) = \deg(\text{divisor cut out by } g \text{ away from } \mathrm{Sing}(\Gamma))$.

Theorem (BDLP, 2015)

Let $I \subset k[x_0, \ldots, x_r]$ be saturated homogeneous with $I(\Gamma) \subsetneq I$ and suppose G is reduced Gröbner basis of I. If p is a prime and $g \in I$ is homogeneous of degree m such that

1. $\mathrm{LM}(I(\Gamma_p)) = \mathrm{LM}(I(\Gamma))$
2. *$G_p = G(p)$ is reduced Gröbner basis of an adjoint ideal of Γ_p*

Verification

$\widetilde{d}(g) = \deg(\text{divisor cut out by } g \text{ away from } \mathrm{Sing}(\Gamma))$.

Theorem (BDLP, 2015)

Let $I \subset k[x_0, \ldots, x_r]$ be saturated homogeneous with $I(\Gamma) \subsetneq I$ and suppose G is reduced Gröbner basis of I. If p is a prime and $g \in I$ is homogeneous of degree m such that

1. $\mathrm{LM}(I(\Gamma_p)) = \mathrm{LM}(I(\Gamma))$
2. $G_p = G(p)$ *is reduced Gröbner basis of an adjoint ideal of* Γ_p
3. $\widetilde{d}(g_p) = (\deg \Gamma) \cdot m - \deg I_p - \delta(\Gamma)$

Verification

$\widetilde{d}(g) = \deg(\text{divisor cut out by } g \text{ away from } \mathrm{Sing}(\Gamma))$.

Theorem (BDLP, 2015)

Let $I \subset k[x_0, \ldots, x_r]$ be saturated homogeneous with $I(\Gamma) \subsetneq I$ and suppose G is reduced Gröbner basis of I. If p is a prime and $g \in I$ is homogeneous of degree m such that

1. $\mathrm{LM}(I(\Gamma_p)) = \mathrm{LM}(I(\Gamma))$
2. $G_p = G(p)$ *is reduced Gröbner basis of an adjoint ideal of* Γ_p
3. $\widetilde{d}(g_p) = (\deg \Gamma) \cdot m - \deg I_p - \delta(\Gamma)$
4. $|mH - \Delta(I_p)|$ *is non-special*

Verification

$\widetilde{d}(g) = \deg(\text{divisor cut out by } g \text{ away from } \mathrm{Sing}(\Gamma))$.

Theorem (BDLP, 2015)

Let $I \subset k[x_0, \ldots, x_r]$ be saturated homogeneous with $I(\Gamma) \subsetneq I$ and suppose G is reduced Gröbner basis of I. If p is a prime and $g \in I$ is homogeneous of degree m such that

1. $\mathrm{LM}(I(\Gamma_p)) = \mathrm{LM}(I(\Gamma))$
2. $G_p = G(p)$ *is reduced Gröbner basis of an adjoint ideal of* Γ_p
3. $\widetilde{d}(g_p) = (\deg \Gamma) \cdot m - \deg I_p - \delta(\Gamma)$
4. $|mH - \Delta(I_p)|$ *is non-special*

then

$$\deg \Delta(I) = \deg \Delta(I_p) = (\deg \Gamma) \cdot m - \widetilde{d}(g_p)$$
$$\delta(\Gamma) = \delta(\Gamma_p)$$

and I is an adjoint ideal.

- Computer algebra system for polynomial computations, over 30 development teams worldwide, over 130 libraries for advanced topics.

- Computer algebra system for polynomial computations, over 30 development teams worldwide, over 130 libraries for advanced topics.

- Special emphasis on algebraic geometry, commutative and non-commutative algebra, singularity theory, packages for convex and tropical geometry.

Timings in SINGULAR

Plane curve f_n of degree n with $\binom{n-1}{2}$ singularities of type A_1.

Timings in Singular

Plane curve f_n of degree n with $\binom{n-1}{2}$ singularities of type A_1.

	parallel	probablisitic	f_5		f_6		f_7	
locNormal			2.1		56		-	
Maple-IB			5.1		47		318	
LA			98		4400		-	
IQ			1.3		54		3800	
locIQ	■		1.3	(1)	54	(1)	3800	(1)
ADE	■		.18	(1)	1.2	(1)	49	(1)
modLocIQ			6.4	[33]	19	[53]	150	[75]
		■	6.2	[33]	18	[53]	104	[75]
	■		.36	(74)	1.6	(153)	51	(230)
	■	■	.21	(74)	0.48	(153)	5.2	(230)

[primes] (cores)

Timings in SINGULAR

Plane curve $f_{n,d}$ of degree d with one singularity of type D_n.
Curves h_1, h_2 of degree 20 and 28 in $\mathbb{P}^5$.

Timings in SINGULAR

Plane curve $f_{n,d}$ of degree d with one singularity of type D_n.
Curves h_1, h_2 of degree 20 and 28 in $\mathbb{P}^5$.

	parallel	probabilistic	$f_{50,500}$		$f_{400,500}$		h_1		h_2	
`locNormal`			.67		4.9		21		-	
`Maple-IB`			1830		-		N/A		N/A	
`LA`			-		-		N/A		N/A	
`IQ`			.67		5.0		30		-	
`locIQ`	■		.67	(1)	5.0	(1)	7.5	(6)	-	
`ADE`	■		.58	(1)	5.0	(1)	N/A		N/A	
`modLocIQ`		■	1.5	[2]	24	[2]	27	[3]	2600	[5]
	■	■	.77	(2)	17	(2)	4.0	[27]	59	(69)

[primes] (cores)

References

- J. Boehm, W. Decker, C. Fieker, G. Pfister. *The use of bad primes in rational reconstruction*, Math. Comp. 84 (2015).
- J. Boehm, W. Decker, S. Laplagne, G. Pfister, A. Steenpaß, S. Steidel. *Parallel algorithms for normalization*, J. Symb. Comp. 51 (2013).
- J. Boehm, W. Decker, G. Pfister, S. Laplagne. *Local to global algorithms for the Gorenstein adjoint ideal of a curve*, arXiv:1505.05040.
- J. Boehm, W. Decker, G. Pfister, S. Laplagne. *adjointideal.lib. A Singular 4 library for computing adjoint ideals*, SINGULAR distribution.

- P. Kornerup, R. T. Gregory, *Mapping integers and Hensel codes onto Farey fractions*, BIT 23 (1983).
- E. Arnold, *Modular algorithms for computing Gröbner bases*, J. Symb. Comp. 35 (2003).
- G.-M. Greuel, S. Laplagne, S. Seelisch, *Normalization of rings*, J. Symb. Comp. (2010).