SPQR: A new Package for Polynomial Division and Elimination Theory via Finite Fields

Giulio Crisanti

MathemAmplitudes, Mainz, 23/09/25

Based on upcoming work with

Vsevolod Chestnov

Introduction and Motivation (1/2)

Talk Outline and Goal

Explore technology behind SPQR as well as various applications

Introduction and Motivation (1/2)

Talk Outline and Goal

Explore technology behind SPQR as well as various applications

General overview of polynomial division

New methods/implementation: finite fields

Introduction and Motivation (1/2)

Talk Outline and Goal

Explore technology behind SPQR as well as various applications

General overview of polynomial division

New methods/implementation: finite fields

Elimination Theory

Landau Analysis

Super Quick Review of Polynomial Division

Univariate Polynomial Division

Polynomial division allows us to decompose functions as

$$f(x) = q(x)p(x) + r(x)$$

$$\updownarrow$$

$$f(x) = r(x) \mod p(x)$$

Super Quick Review of Polynomial Division

Univariate Polynomial Division

Polynomial division allows us to decompose functions as

$$f(x) = q(x)p(x) + r(x)$$

$$\deg(r) < \deg(p)$$

$$f(x) = r(x) \mod p(x)$$

Super Quick Review of Polynomial Division

Univariate Polynomial Division

Polynomial division allows us to decompose functions as

$$f(x) = q(x)p(x) + r(x)$$

$$\deg(r) < \deg(p)$$

$$f(x) = r(x) \mod p(x)$$

Can always be done — best seen by example!

$$f(x) = x^3 + ax^2 - (4 + 2a)x + 1$$

$$p(x) = x^2 - 2x - 1 \longrightarrow x^2 = p(x) + 2x + 1$$

Super Quick Review of Polynomial Division

Univariate Polynomial Division

Polynomial division allows us to decompose functions as

$$f(x) = q(x)p(x) + r(x)$$

$$\deg(r) < \deg(p)$$

$$f(x) = r(x) \mod p(x)$$

Can always be done — best seen by example!

$$f(x) = x^3 + ax^2 - (4+2a)x + 1 \qquad p(x) = x^2 - 2x - 1 \longrightarrow x^2 = p(x) + 2x + 1$$

$$\begin{aligned} f(x) &= x\,(p(x) + 2x + 1) + a\,(p(x) + 2x + 1) - (4+2a)x + 1 \\ &= p(x)(x+a) + 2x^2 + a - 3x + 1 \\ &= p(x)(x+a) + 2(p(x) + 2x + 1) + a - 3x + 1 \\ &= p(x)(x+a+2) + x + a + 3 \end{aligned}$$

Towards Finite Fields

Polynomial division as row reduction

If all we care about is the remainder, we can reformulate polynomial division as a row reduction problem

$$p(x) = 0 \mod p(x) \longrightarrow x^n p(x) = 0 \mod p(x)$$

Towards Finite Fields

Polynomial division as row reduction

If all we care about is the remainder, we can reformulate polynomial division as a row reduction problem

$$p(x) = 0 \mod p(x) \longrightarrow x^n p(x) = 0 \mod p(x)$$

Can generate a linear system of equations this way

$$x^2 - 2x - 1 = 0 \mod p(x)$$
$$x^3 - 2x^2 - x = 0 \mod p(x)$$
$$\vdots$$

Towards Finite Fields

Polynomial division as row reduction

If all we care about is the remainder, we can reformulate polynomial division as a row reduction problem

$$p(x) = 0 \mod p(x) \longrightarrow x^n p(x) = 0 \mod p(x)$$

Can generate a linear system of equations this way

$$x^2 - 2x - 1 = 0 \mod p(x)$$
$$x^3 - 2x^2 - x = 0 \mod p(x)$$
$$\vdots$$
$$f(x) - x^3 - ax^2 + (5 + 2a)x + 1 = 0 \mod p(x)$$

Towards Finite Fields

Polynomial division as row reduction

If all we care about is the remainder, we can reformulate polynomial division as a row reduction problem

$$p(x) = 0 \mod p(x) \longrightarrow x^n p(x) = 0 \mod p(x)$$

Can generate a linear system of equations this way

$$x^2 - 2x - 1 = 0 \mod p(x)$$
$$x^3 - 2x^2 - x = 0 \mod p(x)$$
$$\vdots$$
$$f(x) - x^3 - ax^2 + (5+2a)x + 1 = 0 \mod p(x)$$

Cast in (Macaulay) matrix form:

$$\begin{bmatrix} -1 & 1 & a & -(4+2a) & 1 \\ 0 & -1 & 2 & 1 & 0 \\ 0 & 0 & -1 & 2 & 1 \end{bmatrix} \begin{bmatrix} f(x) \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \mathbf{0}$$

Towards Finite Fields

Polynomial division as row reduction

If all we care about is the remainder, we can reformulate polynomial division as a row reduction problem

$$p(x) = 0 \mod p(x) \longrightarrow x^n p(x) = 0 \mod p(x)$$

Can generate a linear system of equations this way

$$x^2 - 2x - 1 = 0 \mod p(x)$$
$$x^3 - 2x^2 - x = 0 \mod p(x)$$
$$\vdots$$
$$f(x) - x^3 - ax^2 + (5 + 2a)x + 1 = 0 \mod p(x)$$

Cast in (Macaulay) matrix form:

$$\begin{bmatrix} -1 & 1 & a & -(4+2a) & 1 \\ 0 & -1 & 2 & 1 & 0 \\ 0 & 0 & -1 & 2 & 1 \end{bmatrix} \begin{bmatrix} f(x) \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \mathbf{0} \xrightarrow{\text{RowRed}} \begin{bmatrix} 1 & 0 & 0 & -1 & -3-a \\ 0 & 1 & 0 & -5 & -2 \\ 0 & 0 & 1 & -2 & -1 \end{bmatrix} \begin{bmatrix} f(x) \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \mathbf{0}$$

Towards Finite Fields

Polynomial division as row reduction

If all we care about is the remainder, we can reformulate polynomial division as a row reduction problem

$$p(x) = 0 \mod p(x) \longrightarrow x^n p(x) = 0 \mod p(x)$$

Can generate a linear system of equations this way

$$x^2 - 2x - 1 = 0 \mod p(x)$$
$$x^3 - 2x^2 - x = 0 \mod p(x)$$
$$\vdots$$
$$f(x) - x^3 - ax^2 + (5+2a)x + 1 = 0 \mod p(x)$$

$$f(x) - 3 - a - x = 0 \mod p(x)$$

Cast in (Macaulay) matrix form:

$$\begin{bmatrix} -1 & 1 & a & -(4+2a) & 1 \\ 0 & -1 & 2 & 1 & 0 \\ 0 & 0 & -1 & 2 & 1 \end{bmatrix} \begin{bmatrix} f(x) \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \mathbf{0} \xrightarrow{\text{RowRed}} \begin{bmatrix} 1 & 0 & 0 & -1 & -3-a \\ 0 & 1 & 0 & -5 & -2 \\ 0 & 0 & 1 & -2 & -1 \end{bmatrix} \begin{bmatrix} f(x) \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \mathbf{0}$$

Towards Finite Fields

Polynomial division as row reduction

If all we care about is the remainder, we can reformulate polynomial division as a row reduction problem

$$p(x) = 0 \mod p(x) \longrightarrow x^n p(x) = 0 \mod p(x)$$

Can generate a linear system of equations this way

$$x^2 - 2x - 1 = 0 \mod p(x)$$
$$x^3 - 2x^2 - x = 0 \mod p(x)$$
$$\vdots$$
$$f(x) - x^3 - ax^2 + (5+2a)x + 1 = 0 \mod p(x)$$

$$f(x) - 3 - a - x = 0 \mod p(x)$$

Cast in (Macaulay) matrix form:

$$\begin{bmatrix} -1 & 1 & a & -(4+2a) & 1 \\ 0 & -1 & 2 & 1 & 0 \\ 0 & 0 & -1 & 2 & 1 \end{bmatrix} \begin{bmatrix} f(x) \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \mathbf{0} \xrightarrow{\text{RowRed}} \begin{bmatrix} 1 & 0 & 0 & -1 & -3-a \\ 0 & 1 & 0 & -5 & -2 \\ 0 & 0 & 1 & -2 & -1 \end{bmatrix} \begin{bmatrix} f(x) \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \mathbf{0}$$

Useful because there exist very quick ways to do row reduction: Sample over finite fields and reconstruct output

Finite Field Reconstruction

Operations on Matrices

$\mathbb{M}(a_1, \cdots, a_n)$

Finite Field Reconstruction

Operations on Matrices

$$
\begin{tikzcd}
& & \mathbb{M}(a_1, \cdots, a_n) \arrow[dll] \arrow[dl] \arrow[d] \arrow[dr] & \\
\mathbb{M}_1 + \mathbb{M}_2 & \mathbb{M}_1 \cdot \mathbb{M}_2 & \mathbb{M}^{-1} & \mathrm{RowReduce}(\mathbb{M})
\end{tikzcd}
$$

Finite Field Reconstruction

Operations on Matrices

$$\mathbb{M}(a_1, \cdots, a_n)$$

$$\mathbb{M}_1 + \mathbb{M}_2 \qquad \mathbb{M}_1 \cdot \mathbb{M}_2 \qquad \mathbb{M}^{-1} \qquad \mathrm{RowReduce}(\mathbb{M})$$

Algebraic post processing simplification — can become very intensive!

Finite Field Reconstruction

Operations on Matrices

$$
\begin{tikzcd}
 & & \mathbb{M}(a_1, \cdots, a_n) \arrow[dll] \arrow[dl] \arrow[d] \arrow[dr] & \\
\mathbb{M}_1 + \mathbb{M}_2 & \mathbb{M}_1 \cdot \mathbb{M}_2 & \mathbb{M}^{-1} & \mathrm{RowReduce}(\mathbb{M})
\end{tikzcd}
$$

Algebraic post processing simplification — can become very intensive!

Finite Fields Approach

Substitute numerical values for parameters $\longrightarrow$ Perform all operations numerically mod primes $\longrightarrow$ Reconstruct functional output from numerical sampling

Finite Field Reconstruction

Operations on Matrices

$$\mathbb{M}(a_1, \cdots, a_n)$$

$$\mathbb{M}_1 + \mathbb{M}_2 \qquad \mathbb{M}_1 \cdot \mathbb{M}_2 \qquad \mathbb{M}^{-1} \qquad \mathrm{RowReduce}(\mathbb{M})$$

Algebraic post processing simplification — can become very intensive!

Finite Fields Approach

Substitute numerical values for parameters → Perform all operations numerically mod primes → Reconstruct functional output from numerical sampling

Complicated cancellations will happen numerically — final reconstructed output already "simplified"

All operations over finite fields handled by the C++/Mathematica package *FiniteFlow* [FiniteFlow, Peraro, 2019]

Finite Field Reconstruction

Operations on Matrices

$$\mathbb{M}(a_1, \cdots, a_n)$$

$$\mathbb{M}_1 + \mathbb{M}_2 \qquad \mathbb{M}_1 \cdot \mathbb{M}_2 \qquad \mathbb{M}^{-1} \qquad \mathrm{RowReduce}(\mathbb{M})$$

Algebraic post processing simplification — can become very intensive!

Finite Fields Approach

Substitute numerical values for parameters $\longrightarrow$ Perform all operations numerically mod primes $\longrightarrow$ Reconstruct functional output from numerical sampling

Complicated cancellations will happen numerically — final reconstructed output already "simplified"

All operations over finite fields handled by the C++/Mathematica package *FiniteFlow* [FiniteFlow, Peraro, 2019]

What is reconstructed?

$$f(x) = x^3 + ax^2 - (5+2a)x + 1$$
$$p(x) = x^2 - 2x - 1$$

$$\begin{bmatrix} -1 & 1 & a & -(4+2a) & 1 \\ 0 & -1 & 2 & 1 & 0 \\ 0 & 0 & -1 & 2 & 1 \end{bmatrix} \begin{bmatrix} f(x) \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \mathbf{0}$$

Finite Field Reconstruction

Operations on Matrices

$$\mathbb{M}(a_1, \cdots, a_n)$$

$$\mathbb{M}_1 + \mathbb{M}_2 \qquad \mathbb{M}_1 \cdot \mathbb{M}_2 \qquad \mathbb{M}^{-1} \qquad \mathrm{RowReduce}(\mathbb{M})$$

Algebraic post processing simplification — can become very intensive!

Finite Fields Approach

Substitute numerical values for parameters → Perform all operations numerically mod primes → Reconstruct functional output from numerical sampling

Complicated cancellations will happen numerically — final reconstructed output already “simplified”

All operations over finite fields handled by the C++/Mathematica package *FiniteFlow* [FiniteFlow, Peraro, 2019]

What is reconstructed?

$$f(x) = x^3 + ax^2 - (5+2a)x + 1$$
$$p(x) = x^2 - 2x - 1$$

parameters

$$\begin{bmatrix} -1 & 1 & a & -(4+2a) & 1 \\ 0 & -1 & 2 & 1 & 0 \\ 0 & 0 & -1 & 2 & 1 \end{bmatrix} \begin{bmatrix} f(x) \\ x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \mathbf{0}$$

$$R = \mathbb{Q}(a)[x]$$

variables

Only need to reconstruct parameters!

Multivariate Polynomial Division (1/3)

Monomial Orderings

For one variable, sorting the monomials from “worst” to “best” is unambiguous

$$x^n > x^{n-1} > \cdots > x^3 > x^2 > x > 1$$

Multivariate Polynomial Division (1/3)

Monomial Orderings

For one variable, sorting the monomials from “worst” to “best” is unambiguous

$$x^n > x^{n-1} > \cdots > x^3 > x^2 > x > 1$$

For more than one variable there are multiple choices one can take

$$x > y$$

Multivariate Polynomial Division (1/3)

Monomial Orderings

For one variable, sorting the monomials from “worst” to “best” is unambiguous

$$x^n > x^{n-1} > \cdots > x^3 > x^2 > x > 1$$

For more than one variable there are multiple choices one can take

$x > y$ → Lexicographic: $\cdots > x^2 > xy^\infty > xy > x > y^\infty > \cdots > y > 1$

Multivariate Polynomial Division (1/3)

Monomial Orderings

For one variable, sorting the monomials from “worst” to “best” is unambiguous

$$x^n > x^{n-1} > \cdots > x^3 > x^2 > x > 1$$

For more than one variable there are multiple choices one can take

$x > y$

Lexicographic: $\cdots > x^2 > xy^\infty > xy > x > y^\infty > \cdots > y > 1$

Degree Lexicographic: $\cdots > y^3 > x^2 > xy > y^2 > x > y > 1$

Multivariate Polynomial Division (1/3)

Monomial Orderings

For one variable, sorting the monomials from “worst” to “best” is unambiguous

$$x^n > x^{n-1} > \cdots > x^3 > x^2 > x > 1$$

For more than one variable there are multiple choices one can take

$x > y$

Lexicographic: $\cdots > x^2 > xy^\infty > xy > x > y^\infty > \cdots > y > 1$

Degree Lexicographic: $\cdots > y^3 > x^2 > xy > y^2 > x > y > 1$

Is the division unique?

Unfortunately, this is not enough to uniquely determine a multivariate polynomial division

Consider $I = \langle xy - x, xy - y - 1\rangle$. What is $x\,y = \ \ ? \mod I$

Multivariate Polynomial Division (1/3)

Monomial Orderings

For one variable, sorting the monomials from "worst" to "best" is unambiguous

$$x^n > x^{n-1} > \cdots > x^3 > x^2 > x > 1$$

For more than one variable there are multiple choices one can take

$x > y$

- Lexicographic: $\cdots > x^2 > xy^\infty > xy > x > y^\infty > \cdots > y > 1$
- Degree Lexicographic: $\cdots > y^3 > x^2 > xy > y^2 > x > y > 1$

Is the division unique?

Unfortunately, this is not enough to uniquely determine a multivariate polynomial division

Consider $I = \langle xy - x, xy - y - 1 \rangle$. What is $x\,y = \ ? \mod I$

$$xy = x \mod I$$

Multivariate Polynomial Division (1/3)

Monomial Orderings

For one variable, sorting the monomials from "worst" to "best" is unambiguous

$$x^n > x^{n-1} > \cdots > x^3 > x^2 > x > 1$$

For more than one variable there are multiple choices one can take

$x > y$

Lexicographic: $\cdots > x^2 > xy^\infty > xy > x > y^\infty > \cdots > y > 1$

Degree Lexicographic: $\cdots > y^3 > x^2 > xy > y^2 > x > y > 1$

Is the division unique?

Unfortunately, this is not enough to uniquely determine a multivariate polynomial division

Consider $I = \langle xy - x, xy - y - 1 \rangle$. What is $x\,y = \ ? \mod I$

$xy = x \mod I$ $\qquad$ $xy = y + 1 \mod I$

Problem normally fixed by introducing Groebner Bases

[Buchberger, 1965]

Multivariate Polynomial Division (2/3)

Groebner Bases

A Groebner basis G is a set of generators for the ideal I that has many nice properties

Multivariate Polynomial Division (2/3)

Groebner Bases

A Groebner basis G is a set of generators for the ideal I that has many nice properties

For this talk: Roots of $G = 0$ are the same as $I = 0$, and polynomial division ambiguities fixed

Multivariate Polynomial Division (2/3)

Groebner Bases

A Groebner basis G is a set of generators for the ideal I that has many nice properties

For this talk: Roots of $G = 0$ are the same as $I = 0$, and polynomial division ambiguities fixed

$$I = \langle xy - x, xy - y - 1\rangle \longrightarrow G = \langle y^2 - 1, x - y - 1\rangle \quad \text{(Lexicograhpic)}$$

Any possible combination of the elements of G will result in the same polynomial remainder

Multivariate Polynomial Division (2/3)

Groebner Bases

A Groebner basis G is a set of generators for the ideal I that has many nice properties

For this talk: Roots of $G = 0$ are the same as $I = 0$, and polynomial division ambiguities fixed

$$I = \langle xy - x, xy - y - 1\rangle \longrightarrow G = \langle y^2 - 1, x - y - 1\rangle$$ (Lexicograhpic)

Any possible combination of the elements of G will result in the same polynomial remainder

$$xy \overset{2}{=} (y+1)y = y + y^2 \overset{1}{=} y + 1 \mod G$$

Multivariate Polynomial Division (2/3)

Groebner Bases

A Groebner basis G is a set of generators for the ideal I that has many nice properties

For this talk: Roots of $G = 0$ are the same as $I = 0$, and polynomial division ambiguities fixed

$$I = \langle xy - x, xy - y - 1\rangle \longrightarrow G = \langle y^2 - 1, x - y - 1\rangle$$ (Lexicograhpic)

Any possible combination of the elements of G will result in the same polynomial remainder

$$xy \overset{2}{=} (y+1)y = y + y^2 \overset{1}{=} y + 1 \mod G$$

$$\begin{aligned} xy^2 &\overset{1}{=} x \overset{2}{=} y + 1 \\ xy^2 &\overset{2}{=} (y+1)y^2 \overset{1}{=} y + 1 \end{aligned} \mod G$$

Multivariate Polynomial Division (2/3)

Groebner Bases

A Groebner basis G is a set of generators for the ideal I that has many nice properties

For this talk: Roots of $G = 0$ are the same as $I = 0$, and polynomial division ambiguities fixed

$$I = \langle xy - x, xy - y - 1\rangle \longrightarrow G = \langle y^2 - 1, x - y - 1\rangle$$ (Lexicograhpic)

Any possible combination of the elements of G will result in the same polynomial remainder

$$xy \overset{2}{=} (y+1)y = y + y^2 \overset{1}{=} y + 1 \mod G$$

$$\begin{array}{r} xy^2 \overset{1}{=} x \overset{2}{=} y + 1 \\ xy^2 \overset{2}{=} (y+1)y^2 \overset{1}{=} y + 1 \end{array} \mod G$$

Groebner bases can be very difficult to calculate and are often computational bottlenecks!

Multivariate Polynomial Division (2/3)

Groebner Bases

A Groebner basis G is a set of generators for the ideal I that has many nice properties

For this talk: Roots of $G = 0$ are the same as $I = 0$, and polynomial division ambiguities fixed

$$I = \langle xy - x, xy - y - 1 \rangle \longrightarrow G = \langle y^2 - 1, x - y - 1 \rangle$$ (Lexicograhpic)

Any possible combination of the elements of G will result in the same polynomial remainder

$$xy \overset{2}{=} (y+1)y = y + y^2 \overset{1}{=} y + 1 \mod G$$

$$\begin{aligned} xy^2 &\overset{1}{=} x \overset{2}{=} y + 1 \\ xy^2 &\overset{2}{=} (y+1)y^2 \overset{1}{=} y + 1 \end{aligned} \mod G$$

Groebner bases can be very difficult to calculate and are often computational bottlenecks!

Avoiding Groebner Bases

Claim: We can explicitly avoid computing a Groebner basis, and still obtain the correct result from polynomial division, using row reduction [Faugére, 1999] [Buchberger, 1985]

Allows us to compute polynomial divisions without needing to reconstruct the “intermediate” Groebner Basis

Multivariate Polynomial Division (3/3)

Row Reduction Again

We consider again $I = \langle xy - x, xy - y - 1 \rangle$ and let $f(x, y) = xy^2$

Multivariate Polynomial Division (3/3)

Row Reduction Again

We consider again $I = \langle xy - x, xy - y - 1 \rangle$ and let $f(x, y) = xy^2$

Seed a linear system by multiplying I by $x^n y^m$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} f(x) \\ x^2y^2 \\ x^2y \\ x^2 \\ xy^2 \\ xy \\ x \\ y^2 \\ y \\ 1 \end{bmatrix}$$

Multivariate Polynomial Division (3/3)

Row Reduction Again

We consider again $I = \langle xy - x, xy - y - 1 \rangle$ and let $f(x,y) = xy^2$

Seed a linear system by multiplying I by $x^n y^m$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} f(x) \\ x^2y^2 \\ x^2y \\ x^2 \\ xy^2 \\ xy \\ x \\ y^2 \\ y \\ 1 \end{bmatrix} \xrightarrow{\text{RowRed}} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -2 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -2 & -2 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -2 & -2 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} f(x) \\ x^2y^2 \\ x^2y \\ x^2 \\ xy^2 \\ xy \\ x \\ y^2 \\ y \\ 1 \end{bmatrix}$$

Multivariate Polynomial Division (3/3)

Row Reduction Again

We consider again $I = \langle xy - x, xy - y - 1 \rangle$ and let $f(x,y) = xy^2$

Seed a linear system by multiplying I by $x^n y^m$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} f(x) \\ x^2y^2 \\ x^2y \\ x^2 \\ xy^2 \\ xy \\ x \\ y^2 \\ y \\ 1 \end{bmatrix} \xrightarrow{\text{RowRed}} \left[\begin{array}{cccccccc|cc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -2 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -2 & -2 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -2 & -2 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} f(x) \\ x^2y^2 \\ x^2y \\ x^2 \\ xy^2 \\ xy \\ x \\ y^2 \\ \hline y \\ 1 \end{array}\right]$$

Irreducible monomials

Read off from top row: $f(x,y) = y + 1 \mod I$ No explicit Groebner Basis required!

Multivariate Polynomial Division (3/3)

Row Reduction Again

We consider again $I = \langle xy - x, xy - y - 1 \rangle$ and let $f(x,y) = xy^2$

Seed a linear system by multiplying I by $x^n y^m$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} f(x) \\ x^2y^2 \\ x^2y \\ x^2 \\ xy^2 \\ xy \\ x \\ y^2 \\ y \\ 1 \end{bmatrix} \xrightarrow{\text{RowRed}} \left[\begin{array}{cccccccc|cc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -2 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -2 & -2 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -2 & -2 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} f(x) \\ x^2y^2 \\ x^2y \\ x^2 \\ xy^2 \\ xy \\ x \\ y^2 \\ \hline y \\ 1 \end{array}\right]$$

Irreducible monomials

Read off from top row: $f(x,y) = y + 1 \mod I$ No explicit Groebner Basis required!

Important to note: this approach does not escape the complexity of computing Groebner Bases

Multivariate Polynomial Division (3/3)

Row Reduction Again

We consider again $I = \langle xy - x, xy - y - 1 \rangle$ and let $f(x,y) = xy^2$

Seed a linear system by multiplying I by $x^n y^m$

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 \\
0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix} f(x) \\ x^2y^2 \\ x^2y \\ x^2 \\ xy^2 \\ xy \\ x \\ y^2 \\ y \\ 1 \end{bmatrix}
\xrightarrow{\text{RowRed}}
\left[\begin{array}{cccccccc|cc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -2 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -2 & -2 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -2 & -2 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & -1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right]
\left[\begin{array}{c} f(x) \\ x^2y^2 \\ x^2y \\ x^2 \\ xy^2 \\ xy \\ x \\ y^2 \\ \hline y \\ 1 \end{array}\right]
$$

Irreducible monomials

Read off from top row: $f(x,y) = y + 1 \mod I$ No explicit Groebner Basis required!

Important to note: this approach does not escape the complexity of computing Groebner Bases

For example: the size of the system one needs to generate is unclear a priori and can be very large

[Buchberger, 1985]

Companion Matrices (1/3)

Avoiding Large Systems

If we want to reduce polynomials with large monomial powers, the system to row reduce can become very large very quickly. Solution to this problem: Companion Matrices

[Buchberger, 1985]
[Sturmfels, 2002]
[Cox, Little O'Shea, 2015]
[Telen, 2020]
[Brunello, Chestnov, Mastrolia, 2024]

Companion Matrices (1/3)

Avoiding Large Systems

If we want to reduce polynomials with large monomial powers, the system to row reduce can become very large very quickly. Solution to this problem: Companion Matrices

[Buchberger, 1985]
[Sturmfels, 2002]
[Cox, Little O'Shea, 2015]
[Telen, 2020]
[Brunello, Chestnov, Mastrolia, 2024]

Companion matrices are built from taking polynomial divisions of a given function.

Companion Matrices (1/3)

Avoiding Large Systems

If we want to reduce polynomials with large monomial powers, the system to row reduce can become very large very quickly. Solution to this problem: Companion Matrices

[Buchberger, 1985]
[Sturmfels, 2002]
[Cox, Little O'Shea, 2015]
[Telen, 2020]
[Brunello, Chestnov, Mastrolia, 2024]

Companion matrices are built from taking polynomial divisions of a given function.

Let $m = \{\cdots y, 1\}$ be the irreducible monomials of a polynomial system.

Companion Matrices (1/3)

Avoiding Large Systems

If we want to reduce polynomials with large monomial powers, the system to row reduce can become very large very quickly. Solution to this problem: Companion Matrices

[Buchberger, 1985]
[Sturmfels, 2002]
[Cox, Little O'Shea, 2015]
[Telen, 2020]
[Brunello, Chestnov, Mastrolia, 2024]

Companion matrices are built from taking polynomial divisions of a given function.

Let $m = \{\cdots y, 1\}$ be the irreducible monomials of a polynomial system.

$$M_{f(x,y)} : \quad \begin{aligned} m_1 \cdot f(x,y) &= c_{11} m_1 + c_{12} m_2 + \cdots \\ m_2 \cdot f(x,y) &= c_{21} m_1 + c_{22} m_2 + \cdots \\ &\vdots \end{aligned} \quad \mod I(x,y)$$

Companion Matrices (1/3)

Avoiding Large Systems

If we want to reduce polynomials with large monomial powers, the system to row reduce can become very large very quickly. Solution to this problem: Companion Matrices

[Buchberger, 1985]
[Sturmfels, 2002]
[Cox, Little O'Shea, 2015]
[Telen, 2020]
[Brunello, Chestnov, Mastrolia, 2024]

Companion matrices are built from taking polynomial divisions of a given function.

Let $m = \{\cdots y, 1\}$ be the irreducible monomials of a polynomial system.

$$M_{f(x,y)} : \begin{array}{c} m_1 \cdot f(x,y) = c_{11} m_1 + c_{12} m_2 + \cdots \\ m_2 \cdot f(x,y) = c_{21} m_1 + c_{22} m_2 + \cdots \\ \vdots \end{array} \mod I(x,y) \longrightarrow M_{f(x,y)} = \begin{bmatrix} c_{11} & c_{12} & \cdots \\ c_{21} & c_{22} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

Companion Matrices (1/3)

Avoiding Large Systems

If we want to reduce polynomials with large monomial powers, the system to row reduce can become very large very quickly. Solution to this problem: Companion Matrices

[Buchberger, 1985]
[Sturmfels, 2002]
[Cox, Little O'Shea, 2015]
[Telen, 2020]
[Brunello, Chestnov, Mastrolia, 2024]

Companion matrices are built from taking polynomial divisions of a given function.

Let $m = \{\cdots y, 1\}$ be the irreducible monomials of a polynomial system.

$$M_{f(x,y)}: \begin{array}{c} m_1 \cdot f(x,y) = c_{11} m_1 + c_{12} m_2 + \cdots \\ m_2 \cdot f(x,y) = c_{21} m_1 + c_{22} m_2 + \cdots \\ \vdots \end{array} \quad \mod I(x,y) \longrightarrow M_{f(x,y)} = \begin{bmatrix} c_{11} & c_{12} & \cdots \\ c_{21} & c_{22} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

Key properties of companion matrices:

$$M_{f_1+f_2} = M_{f_1} + M_{f_2} \qquad M_{f_1\, f_2} = M_{f_1}\, M_{f_2} = M_{f_2}\, M_{f_1} \qquad M_{f_{\text{inv}}} = M_f^{-1}$$

Companion Matrices (1/3)

Avoiding Large Systems

If we want to reduce polynomials with large monomial powers, the system to row reduce can become very large very quickly. Solution to this problem: Companion Matrices

[Buchberger, 1985]
[Sturmfels, 2002]
[Cox, Little O'Shea, 2015]
[Telen, 2020]
[Brunello, Chestnov, Mastrolia, 2024]

Companion matrices are built from taking polynomial divisions of a given function.

Let $m = \{\cdots y, 1\}$ be the irreducible monomials of a polynomial system.

$$M_{f(x,y)} : \begin{array}{c} m_1 \cdot f(x,y) = c_{11} m_1 + c_{12} m_2 + \cdots \\ m_2 \cdot f(x,y) = c_{21} m_1 + c_{22} m_2 + \cdots \\ \vdots \end{array} \quad \mathrm{mod}\ I(x,y) \longrightarrow M_{f(x,y)} = \begin{bmatrix} c_{11} & c_{12} & \cdots \\ c_{21} & c_{22} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

Key properties of companion matrices:

$$M_{f_1+f_2} = M_{f_1} + M_{f_2} \qquad M_{f_1 f_2} = M_{f_1} M_{f_2} = M_{f_2} M_{f_1} \qquad M_{f_{\mathrm{inv}}} = M_f^{-1}$$

Remainder can be recovered as:

$$f(x,y) = (0, \cdots, 0, 1) \cdot M_{f(x,y)} \cdot m^T \mod I$$

Companion Matrices (1/3)

Avoiding Large Systems

If we want to reduce polynomials with large monomial powers, the system to row reduce can become very large very quickly. Solution to this problem: Companion Matrices

[Buchberger, 1985]
[Sturmfels, 2002]
[Cox, Little O'Shea, 2015]
[Telen, 2020]
[Brunello, Chestnov, Mastrolia, 2024]

Companion matrices are built from taking polynomial divisions of a given function.

Let $m = \{\cdots y, 1\}$ be the irreducible monomials of a polynomial system.

$$M_{f(x,y)}: \begin{array}{l} m_1 \cdot f(x,y) = c_{11} m_1 + c_{12} m_2 + \cdots \\ m_2 \cdot f(x,y) = c_{21} m_1 + c_{22} m_2 + \cdots \\ \qquad\qquad \vdots \end{array} \mod I(x,y) \longrightarrow M_{f(x,y)} = \begin{bmatrix} c_{11} & c_{12} & \cdots \\ c_{21} & c_{22} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

Key properties of companion matrices:

$$M_{f_1+f_2} = M_{f_1} + M_{f_2} \qquad M_{f_1\, f_2} = M_{f_1}\, M_{f_2} = M_{f_2}\, M_{f_1} \qquad M_{f_{\text{inv}}} = M_f^{-1}$$

Remainder can be recovered as:

$$f(x,y) = (0, \cdots, 0, 1) \cdot M_{f(x,y)} \cdot m^T \mod I$$

Strategy: Build M_x and M_y and you can reduce anything with just matrix multiplication!

Companion Matrices (2/3)

Examples

$$I = \langle xy - x, xy - y \rangle \longrightarrow m = \{y, 1\}$$

Companion Matrices (2/3)

Examples

$$I = \langle xy - x, xy - y \rangle \longrightarrow m = \{y, 1\}$$

$$M_x : \quad \begin{aligned} y \cdot x &= y + 1 \mod I \\ 1 \cdot x &= y + 1 \mod I \end{aligned} \longrightarrow M_x = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Companion Matrices (2/3)

Examples

$$I = \langle xy - x, xy - y \rangle \longrightarrow m = \{y, 1\}$$

$$M_x : \quad \begin{aligned} y \cdot x &= y + 1 \mod I \\ 1 \cdot x &= y + 1 \mod I \end{aligned} \longrightarrow M_x = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

$$M_y : \quad \begin{aligned} y \cdot y &= 1 \mod I \\ 1 \cdot y &= y \mod I \end{aligned} \longrightarrow M_y = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Companion Matrices (2/3)

Examples

$$I = \langle xy - x, xy - y \rangle \longrightarrow m = \{y, 1\}$$

$$M_x : \quad \begin{array}{ll} y \cdot x = y + 1 & \mod I \\ 1 \cdot x = y + 1 & \mod I \end{array} \longrightarrow M_x = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \qquad M_y : \quad \begin{array}{ll} y \cdot y = 1 & \mod I \\ 1 \cdot y = y & \mod I \end{array} \longrightarrow M_y = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Can now consider any function and build it with companion matrices

$$f(x, y) = xy^2 \longrightarrow M_{f(x,y)} = M_x \cdot M_y^2 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Companion Matrices (2/3)

Examples

$$I = \langle xy - x, xy - y \rangle \longrightarrow m = \{y, 1\}$$

$$M_x : \quad \begin{array}{ll} y \cdot x = y + 1 & \mod I \\ 1 \cdot x = y + 1 & \mod I \end{array} \longrightarrow M_x = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \qquad M_y : \quad \begin{array}{ll} y \cdot y = 1 & \mod I \\ 1 \cdot y = y & \mod I \end{array} \longrightarrow M_y = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Can now consider any function and build it with companion matrices

$$f(x,y) = xy^2 \longrightarrow M_{f(x,y)} = M_x \cdot M_y^2 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \longrightarrow r(x,y) = (0,1) \cdot \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} y \\ 1 \end{bmatrix} = y + 1$$

Companion Matrices (2/3)

Examples

$$I = \langle xy - x, xy - y\rangle \longrightarrow m = \{y, 1\}$$

$$M_x : \quad \begin{array}{ll} y \cdot x = y + 1 & \mod I \\ 1 \cdot x = y + 1 & \mod I \end{array} \longrightarrow M_x = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \qquad M_y : \quad \begin{array}{ll} y \cdot y = 1 & \mod I \\ 1 \cdot y = y & \mod I \end{array} \longrightarrow M_y = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Can now consider any function and build it with companion matrices

$$f(x, y) = xy^2 \longrightarrow M_{f(x,y)} = M_x \cdot M_y^2 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \longrightarrow r(x, y) = (0, 1) \cdot \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} y \\ 1 \end{bmatrix} = y + 1$$

Can immediately write down the answer for much more complicated expressions!

$$g(x, y) = a + \frac{x}{y^{100} - 3x + 2} \longrightarrow M_{g(x,y)} = \mathbb{1}\, a + M_x \left(M_y^{100} - 3M_x + 2\,\mathbb{1}\right)^{-1} = \begin{bmatrix} \frac{1}{3}(3a - 1) & -\frac{1}{3} \\ -\frac{1}{3} & \frac{1}{3}(3a - 1) \end{bmatrix}$$

$$g(x, y) = \frac{1}{3}(3a - 1) - \frac{y}{3} \mod I$$

Companion Matrices (3/3)

Variable Elimination

Often one is interested in projecting/eliminating an equation system down onto one variable "slices"

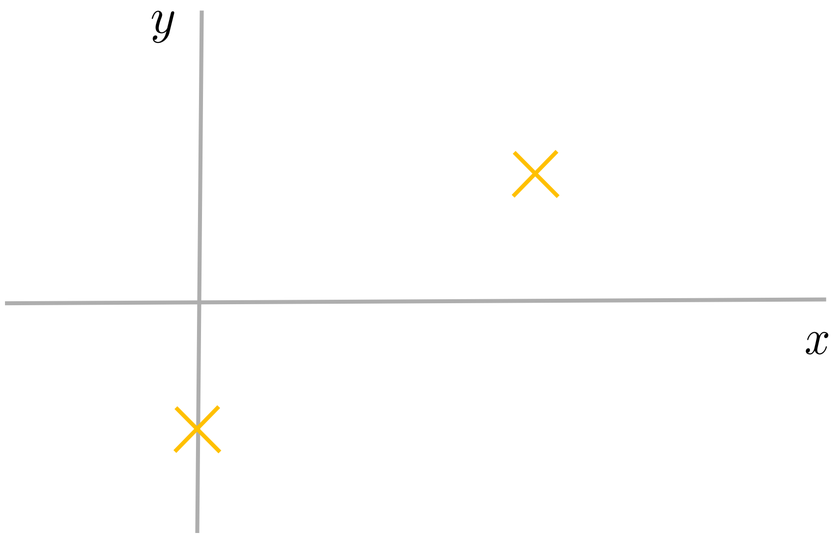
$$I = \langle xy - x, xy - y - 1 \rangle \longrightarrow \{x^* = 0\,,\ y^* = -1\} \cup \{x^* = 2\,,\ y^* = 1\}$$

Companion Matrices (3/3)

Variable Elimination

Often one is interested in projecting/eliminating an equation system down onto one variable "slices"

$$I = \langle xy - x, xy - y - 1 \rangle \longrightarrow \{x^* = 0\,,\, y^* = -1\} \cup \{x^* = 2\,,\, y^* = 1\}$$

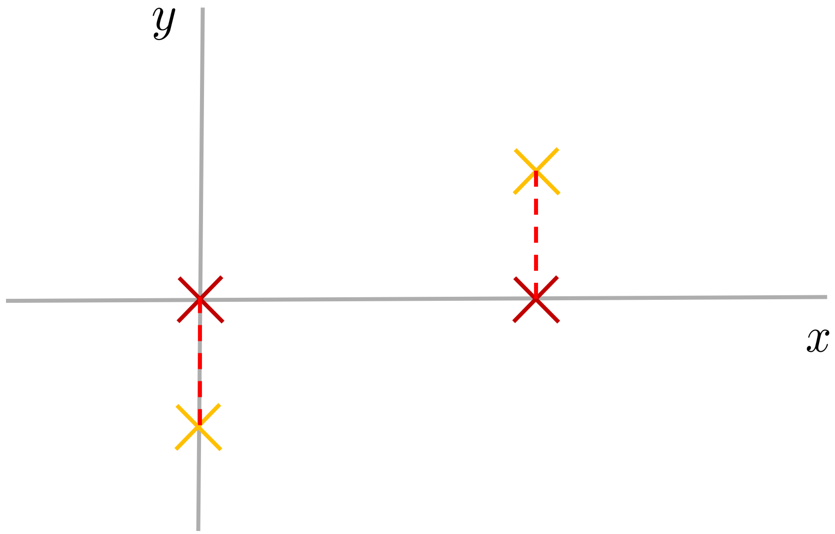


Companion Matrices (3/3)

Variable Elimination

Often one is interested in projecting/eliminating an equation system down onto one variable "slices"

$$I \;=\; \langle xy - x, xy - y - 1\rangle \longrightarrow \{x^* = 0\,,\; y^* = -1\} \cup \{x^* = 2\,,\; y^* = 1\}$$



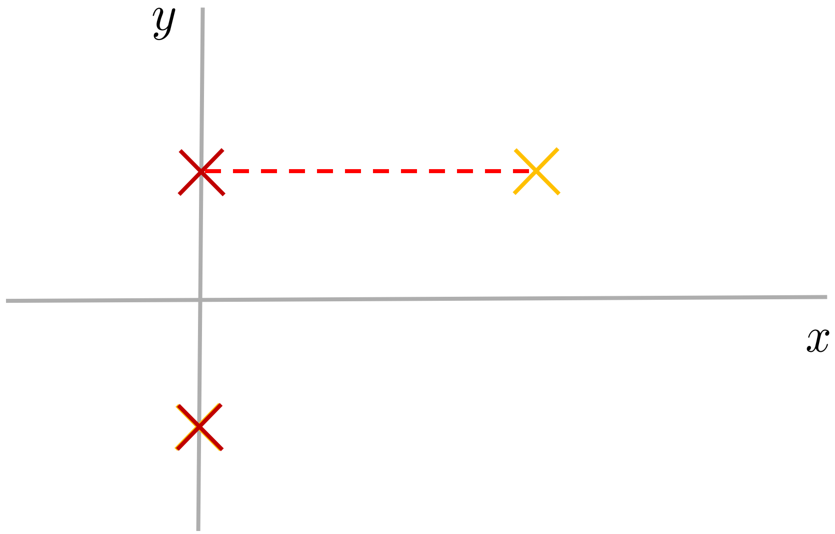
$$\mathrm{Elim}_y(\mathcal{I}) = \langle x(x-2)\rangle$$

Companion Matrices (3/3)

Variable Elimination

Often one is interested in projecting/eliminating an equation system down onto one variable “slices”

$$I = \langle xy - x, xy - y - 1\rangle \longrightarrow \{x^* = 0\,,\ y^* = -1\} \cup \{x^* = 2\,,\ y^* = 1\}$$



$$\mathrm{Elim}_y(\mathcal{I}) = \langle x(x-2)\rangle$$

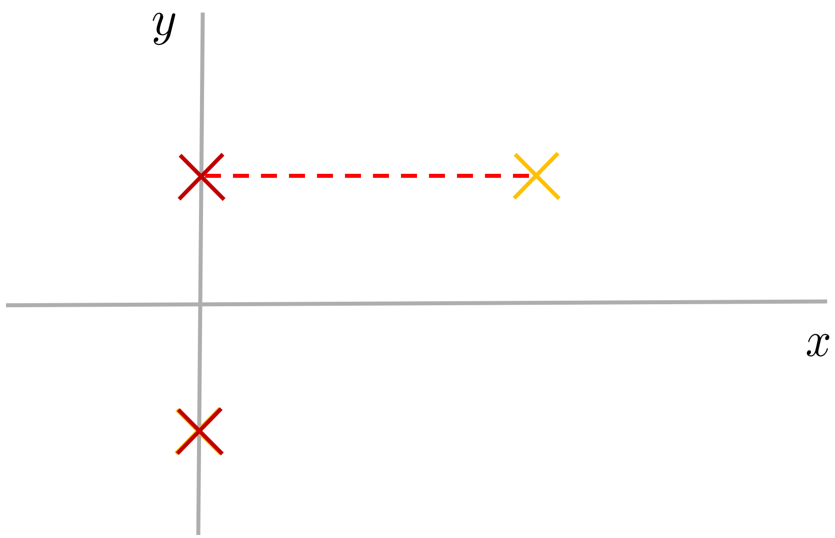
$$\mathrm{Elim}_x(\mathcal{I}) = \langle y^2 - 1\rangle$$

Companion Matrices (3/3)

Variable Elimination

Often one is interested in projecting/eliminating an equation system down onto one variable "slices"

$$I = \langle xy - x, xy - y - 1\rangle \longrightarrow \{x^* = 0\,,\ y^* = -1\} \cup \{x^* = 2\,,\ y^* = 1\}$$



$$\mathrm{Elim}_y(\mathcal{I}) = \langle x(x-2)\rangle$$

$$\mathrm{Elim}_x(\mathcal{I}) = \langle y^2 - 1\rangle$$

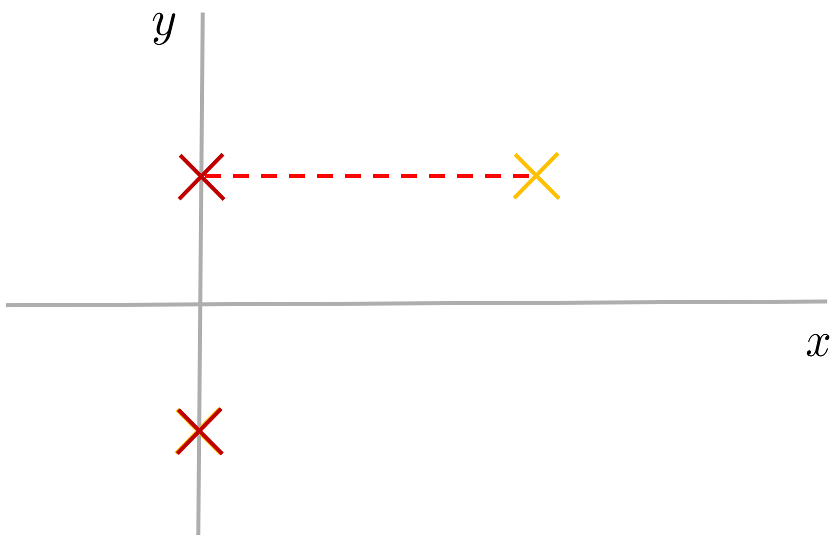
The eigenvalues of companion matrices encode the roots of the polynomial system. Can be conveniently accessed by computing characteristic polynomials

Companion Matrices (3/3)

Variable Elimination

Often one is interested in projecting/eliminating an equation system down onto one variable “slices”

$$I = \langle xy - x, xy - y - 1\rangle \longrightarrow \{x^* = 0\,,\ y^* = -1\} \cup \{x^* = 2\,,\ y^* = 1\}$$



$$\mathrm{Elim}_y(\mathcal{I}) = \langle x(x-2)\rangle$$

$$\mathrm{Elim}_x(\mathcal{I}) = \langle y^2 - 1\rangle$$

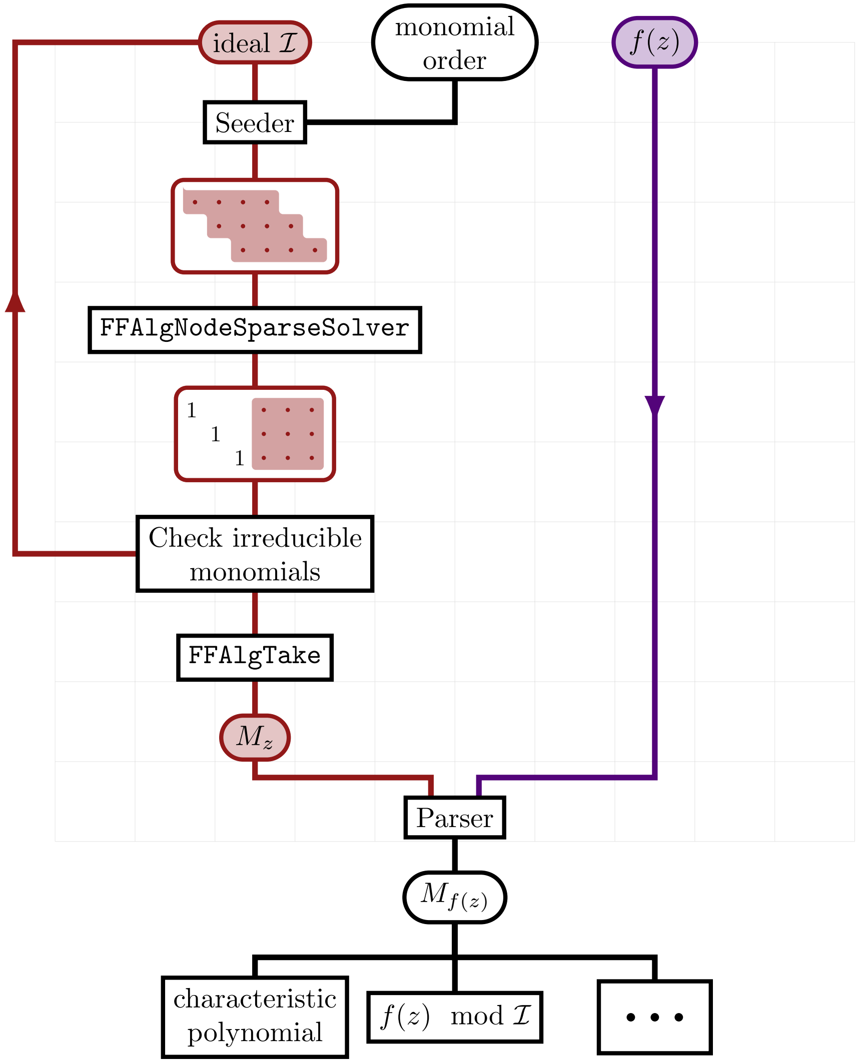
The eigenvalues of companion matrices encode the roots of the polynomial system. Can be conveniently accessed by computing characteristic polynomials

$$\mathrm{Elim}_x(\mathcal{I}) = \det(M_y - y\mathbb{1}) = \det\begin{bmatrix} -y & 1 \\ 1 & -y \end{bmatrix} = y^2 - 1$$

$$\mathrm{Elim}_y(\mathcal{I}) = \det(M_x - x\mathbb{1}) = \det\begin{bmatrix} 1-x & 1 \\ 1 & 1-x \end{bmatrix} = x(x-2)$$

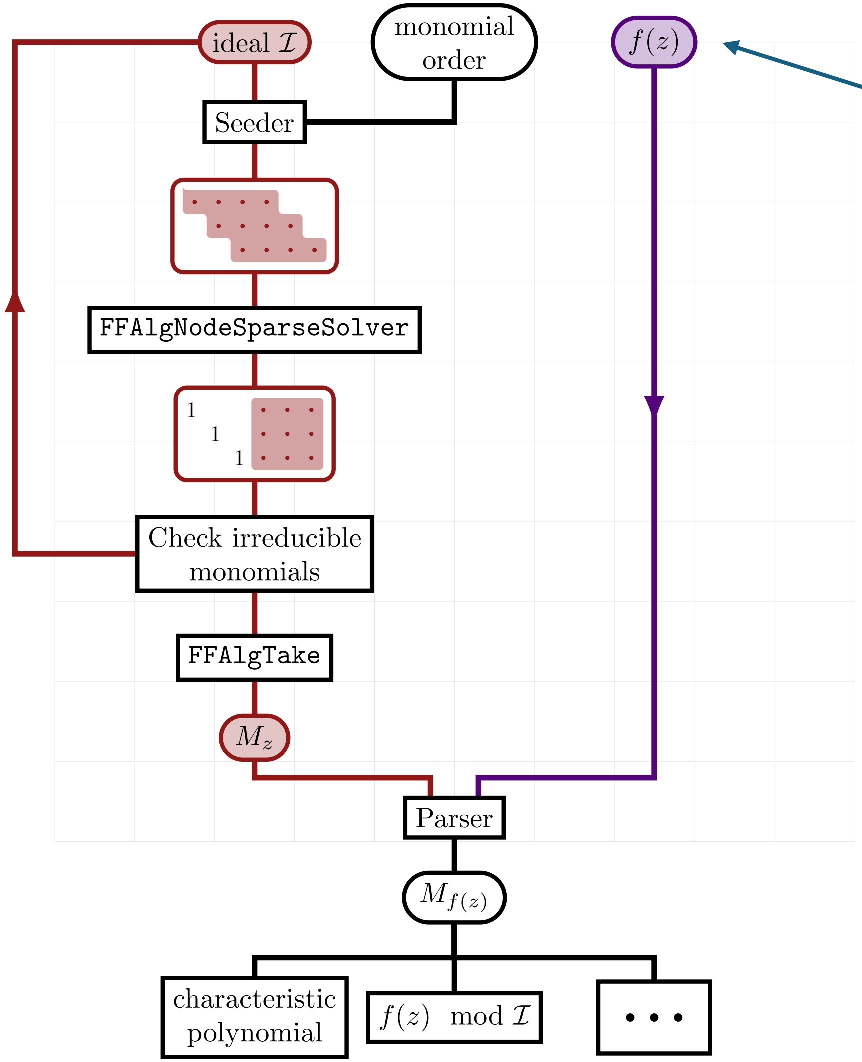
SPQR Summary

A (Mathematica) package that performs polynomial division and reconstruction over finite fields



SPQR Summary

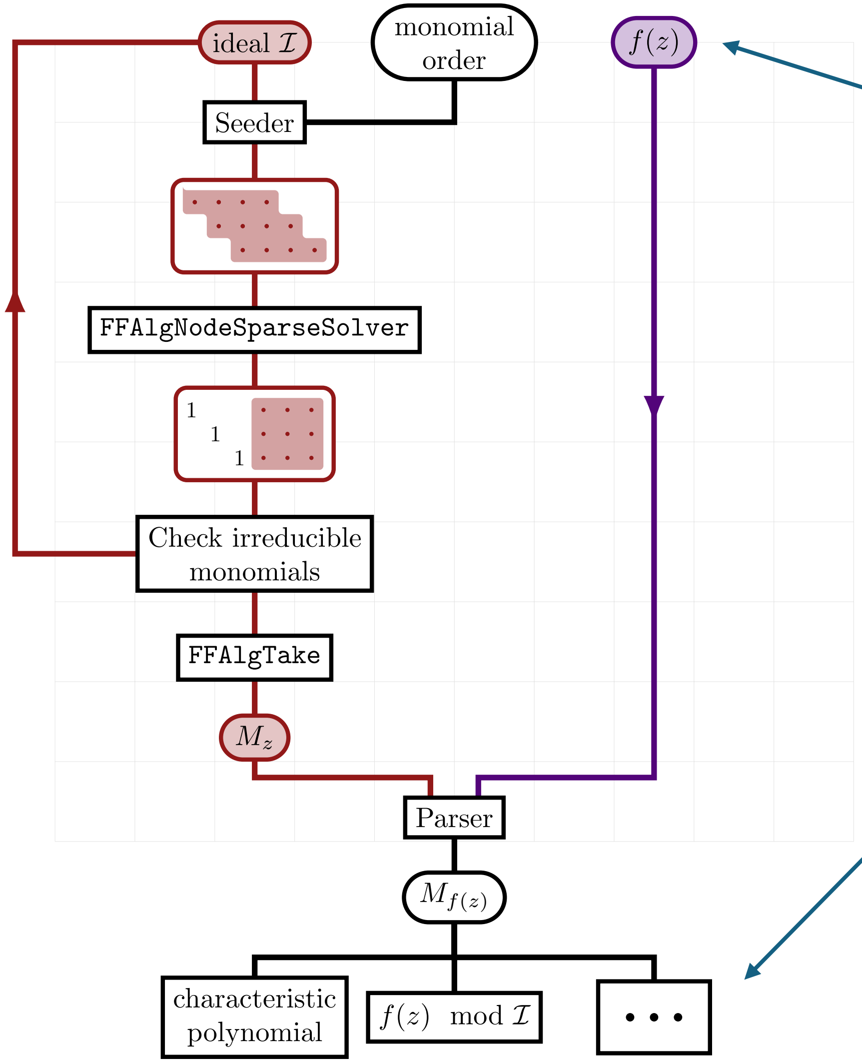
A (Mathematica) package that performs polynomial division and reconstruction over finite fields



Can handle polynomials or multivariate rational functions as input to arbitrary nested depth

SPQR Summary

A (Mathematica) package that performs polynomial division and reconstruction over finite fields



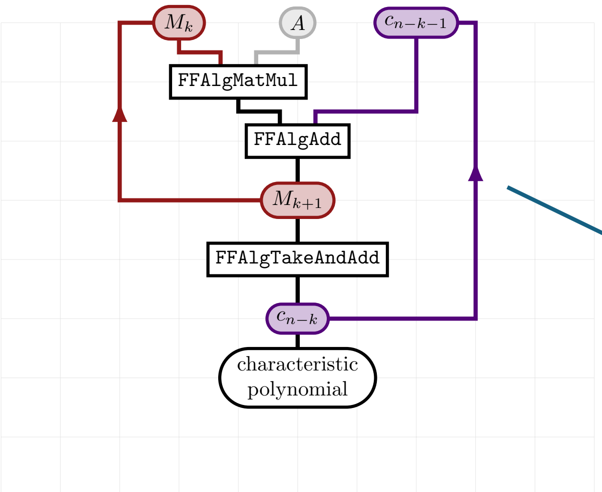
Can handle polynomials or multivariate rational functions as input to arbitrary nested depth

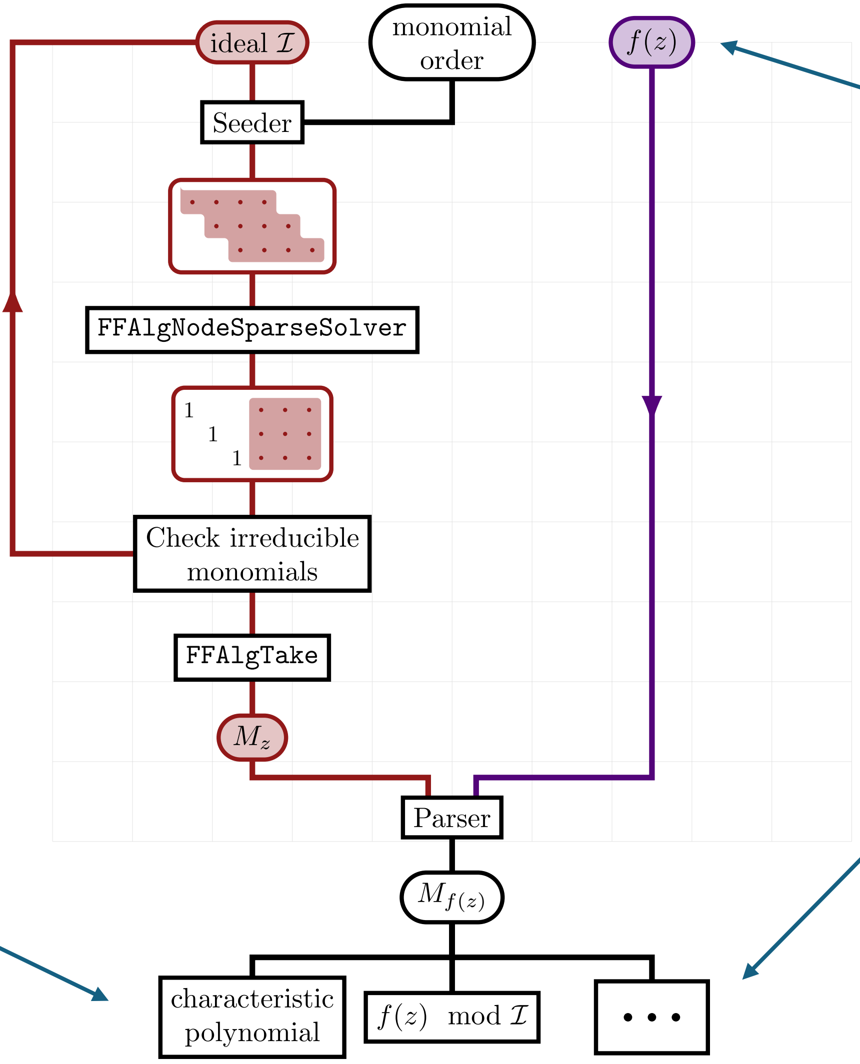
No intermediate reconstructions required ensuring the numerical cancellations of complex intermediate stages

SPQR Summary

A (Mathematica) package that performs polynomial division and reconstruction over finite fields

Characteristic polynomial also built inside FiniteFlow with the Faddeev–LeVerrier algorithm





Can handle polynomials or multivariate rational functions as input to arbitrary nested depth

No intermediate reconstructions required ensuring the numerical cancellations of complex intermediate stages

Advantages and Disadvantages (1/3)

Complexity of polynomial ideals

$$\underbrace{R = \mathbb{Q}[a_1 \cdots a_n]}_{\text{parameters}}\underbrace{[x_1 \cdots x_n]}_{\text{variables}}$$

Advantages and Disadvantages (1/3)

Complexity of polynomial ideals

$$R = \mathbb{Q}[a_1 \cdots a_n][x_1 \cdots x_n]$$

parameters

variables

“Variable complexity” : How many equations are needed for finding a Groebner Basis

Groebner Basis algorithms have fine tuned procedures for keeping this down to a minimum

Our approach (currently) relies on a much coarser and overcomplete system of equations

Advantages and Disadvantages (1/3)

Complexity of polynomial ideals

$$R = \mathbb{Q}[\underbrace{a_1 \cdots a_n}_{\text{parameters}}][\underbrace{x_1 \cdots x_n}_{\text{variables}}]$$

"Variable complexity" : How many equations are needed for finding a Groebner Basis

Groebner Basis algorithms have fine tuned procedures for keeping this down to a minimum

Our approach (currently) relies on a much coarser and overcomplete system of equations

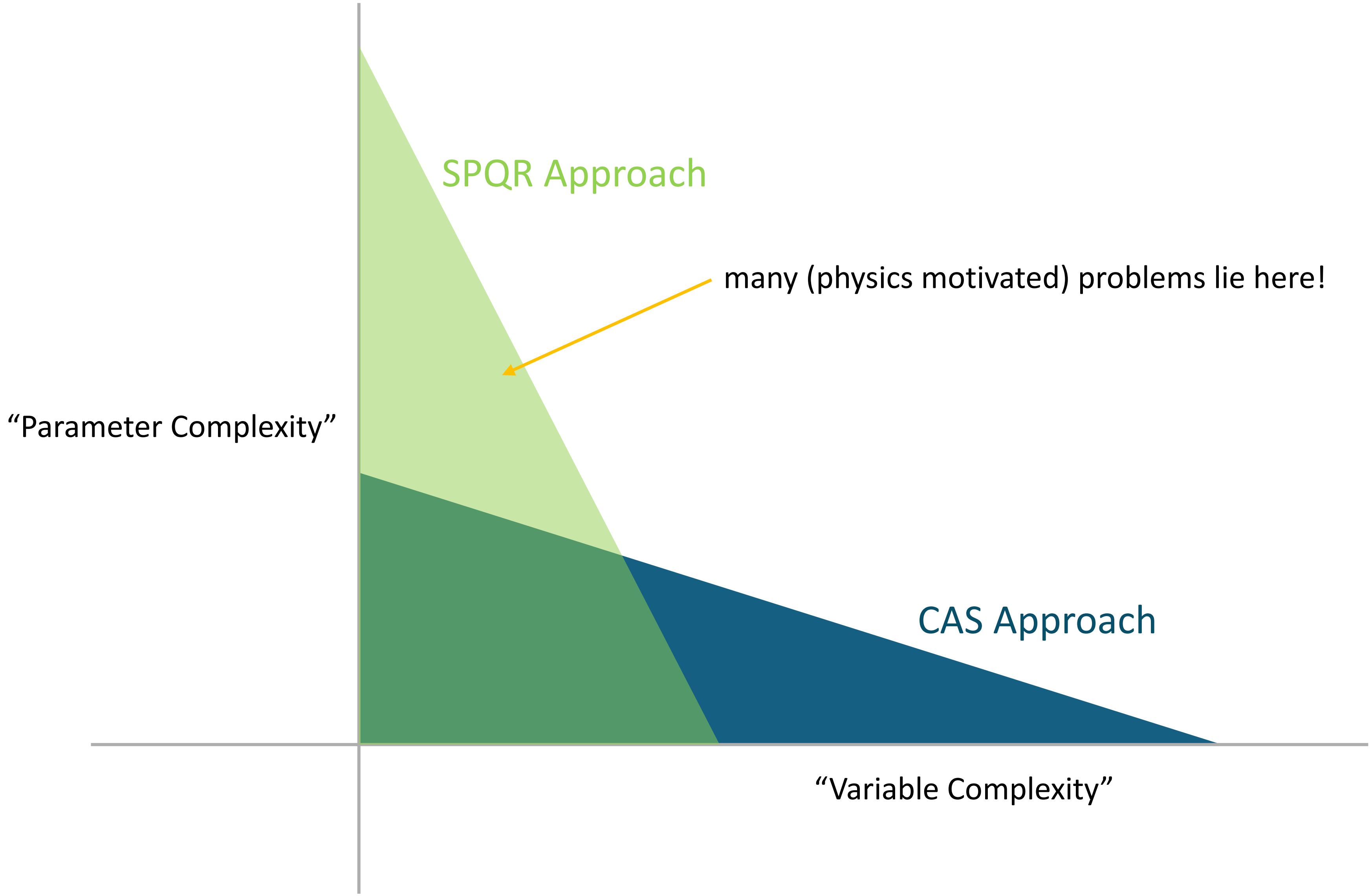
"Parameter complexity" : How complicated are the *coefficients* of the Groebner Basis polynomials

Groebner Basis algorithms suffer from complicated symbolic processing at intermediate stages

Reconstruction over finite fields completely avoids this problem as cancellations happen numerically!

Can speed up computations by orders of magnitude

Advantages and Disadvantages (2/3)



Advantages and Disadvantages (3/3)

Resultant Benchmarking (Preliminary)

$$R = \mathbb{Q}(a, b, c, d)[x, y, z]$$

$$I = \langle a + x^2y^2 + y^3 + z - 1, ax + cxy^2 + cy + z^2 - 2, a + bxy^2 + b + x^2y^2, -c + dxz + xyz + 1 \rangle$$

Task: Eliminate $\{x, y, z\} \longrightarrow \mathcal{R}(a, b, c, d)$

Advantages and Disadvantages (3/3)

Resultant Benchmarking (Preliminary)

$$R = \mathbb{Q}(a,b,c,d)[x,y,z]$$

$$I = \langle a + x^2y^2 + y^3 + z - 1, ax + cxy^2 + cy + z^2 - 2, a + bxy^2 + b + x^2y^2, -c + dxz + xyz + 1\rangle$$

Task: Eliminate $\{x,y,z\} \longrightarrow \mathcal{R}(a,b,c,d)$

Resultant	Singular	Macaulay 2	Mathematica (+ tricks)	SPQR
$\mathcal{R}(3,5,7,d)$				
$\mathcal{R}(3,5,c,d)$				
$\mathcal{R}(3,b,c,d)$				
$\mathcal{R}(a,b,c,d)$				

Advantages and Disadvantages (3/3)

Resultant Benchmarking (Preliminary)

$$R = \mathbb{Q}(a,b,c,d)[x,y,z]$$

$$I = \langle a + x^2y^2 + y^3 + z - 1, ax + cxy^2 + cy + z^2 - 2, a + bxy^2 + b + x^2y^2, -c + dxz + xyz + 1\rangle$$

Task: Eliminate $\{x, y, z\} \longrightarrow \mathcal{R}(a,b,c,d)$

Resultant	Singular	Macaulay 2	Mathematica (+ tricks)	SPQR
$\mathcal{R}(3,5,7,d)$	$\approx 0.01s$	$\approx 0.06s$	$\approx 0.03s$	$\approx 0.18s$
$\mathcal{R}(3,5,c,d)$				
$\mathcal{R}(3,b,c,d)$				
$\mathcal{R}(a,b,c,d)$				

Advantages and Disadvantages (3/3)

Resultant Benchmarking (Preliminary)

$$R = \mathbb{Q}(a,b,c,d)[x,y,z]$$

$$I = \langle a + x^2y^2 + y^3 + z - 1, ax + cxy^2 + cy + z^2 - 2, a + bxy^2 + b + x^2y^2, -c + dxz + xyz + 1 \rangle$$

Task: Eliminate $\{x, y, z\} \longrightarrow \mathcal{R}(a,b,c,d)$

Resultant	Singular	Macaulay 2	Mathematica (+ tricks)	SPQR
$\mathcal{R}(3,5,7,d)$	$\approx 0.01s$	$\approx 0.06s$	$\approx 0.03s$	$\approx 0.18s$
$\mathcal{R}(3,5,c,d)$	$\approx 53.29s$	$\approx 12.40s$	$\approx 5.03s$	$\approx 0.20s$
$\mathcal{R}(3,b,c,d)$				
$\mathcal{R}(a,b,c,d)$				

Advantages and Disadvantages (3/3)

Resultant Benchmarking (Preliminary)

$$R = \mathbb{Q}(a,b,c,d)[x,y,z]$$

$$I = \langle a + x^2y^2 + y^3 + z - 1, ax + cxy^2 + cy + z^2 - 2, a + bxy^2 + b + x^2y^2, -c + dxz + xyz + 1\rangle$$

Task: Eliminate $\{x,y,z\}$ $\longrightarrow$ $\mathcal{R}(a,b,c,d)$

Resultant	Singular	Macaulay 2	Mathematica (+ tricks)	SPQR
$\mathcal{R}(3,5,7,d)$	$\approx 0.01s$	$\approx 0.06s$	$\approx 0.03s$	$\approx 0.18s$
$\mathcal{R}(3,5,c,d)$	$\approx 53.29s$	$\approx 12.40s$	$\approx 5.03s$	$\approx 0.20s$
$\mathcal{R}(3,b,c,d)$	$> 7d$	$> 7d$	$\approx 4h\,39m$	$\approx 0.46s$
$\mathcal{R}(a,b,c,d)$				

Advantages and Disadvantages (3/3)

Resultant Benchmarking (Preliminary)

$$R = \mathbb{Q}(a,b,c,d)[x,y,z]$$

$$I = \langle a + x^2y^2 + y^3 + z - 1, ax + cxy^2 + cy + z^2 - 2, a + bxy^2 + b + x^2y^2, -c + dxz + xyz + 1 \rangle$$

Task: Eliminate $\{x,y,z\} \longrightarrow \mathcal{R}(a,b,c,d)$

Resultant	Singular	Macaulay 2	Mathematica (+ tricks)	SPQR
$\mathcal{R}(3,5,7,d)$	$\approx 0.01s$	$\approx 0.06s$	$\approx 0.03s$	$\approx 0.18s$
$\mathcal{R}(3,5,c,d)$	$\approx 53.29s$	$\approx 12.40s$	$\approx 5.03s$	$\approx 0.20s$
$\mathcal{R}(3,b,c,d)$	$> 7d$	$> 7d$	$\approx 4h\,39m$	$\approx 0.46s$
$\mathcal{R}(a,b,c,d)$	$?$	$?$	$> 7d$	$\approx 1.89s$

9 MB degree 34 polynomial

The Finite Fields approach is solving a less refined set of equations, but isn't slowed down by intermediate cancellations

Physics Motivated Examples (1/3)

Landau Singularities of Feynman (Euler) Integrals

Feynman integrals are Euler/Twisted Period integrals

Twist: polynomial raised to a non integer (generic) power

$$I(s_{ij}, m) = \int_0^\infty \mathcal{G}(x, s_{ij}, m)^{-d/2} \frac{dx_1}{x_1} \wedge \cdots \wedge \frac{dx_n}{x_n}$$

[Lee, 2013]

Parameters (Mandelstam variables)

Integration variables

Physics Motivated Examples (1/3)

Landau Singularities of Feynman (Euler) Integrals

Feynman integrals are Euler/Twisted Period integrals

Twist: polynomial raised to a non integer (generic) power

$$I(s_{ij}, m) = \int_0^\infty \mathcal{G}(x, s_{ij}, m)^{-d/2} \frac{dx_1}{x_1} \wedge \cdots \wedge \frac{dx_n}{x_n}$$

[Lee, 2013]

Parameters (Mandelstam variables)

Integration variables

Landau Analysis: where are the branch points of $I(s_{ij}, m)$ located?

[Landau, 1960]
[Cutkosky, 1960]

[Abreu, Berghoff, Bourjaily, Britto, Correia, Duhr, Fevola, Gardi, Giroux, Hannesdottir, Helmer, Lippestreu, Matsubara-Heo, McLeod, Mizera, Panzer, Papathanasiou, Polackova, Schwartz, Tellander, Telen, Vergu, Wiesmann 2017-2025]

Physics Motivated Examples (1/3)

Landau Singularities of Feynman (Euler) Integrals

Feynman integrals are Euler/Twisted Period integrals

Twist: polynomial raised to a non integer (generic) power

$$I(s_{ij}, m) = \int_0^\infty \mathcal{G}(x, s_{ij}, m)^{-d/2} \frac{dx_1}{x_1} \wedge \cdots \wedge \frac{dx_n}{x_n}$$

[Lee, 2013]

Parameters (Mandelstam variables)

Integration variables

Landau Analysis: where are the branch points of $I(s_{ij}, m)$ located?

[Landau, 1960]
[Cutkosky, 1960]

[Abreu, Berghoff, Bourjaily, Britto, Correia, Duhr, Fevola, Gardi, Giroux, Hannesdottir, Helmer, Lippestreu, Matsubara-Heo, McLeod, Mizera, Panzer, Papathanasiou, Polackova, Schwartz, Tellander, Telen, Vergu, Wiesmann 2017-2025]

Euler/Twisted Period representations allow us to associate an Euler characteristic χ to a given Feynman integral

Physics Motivated Examples (1/3)

Landau Singularities of Feynman (Euler) Integrals

Feynman integrals are Euler/Twisted Period integrals

Twist: polynomial raised to a non integer (generic) power

$$I(s_{ij}, m) = \int_0^\infty \mathcal{G}(x, s_{ij}, m)^{-d/2} \frac{dx_1}{x_1} \wedge \cdots \wedge \frac{dx_n}{x_n}$$

[Lee, 2013]

Parameters (Mandelstam variables) Integration variables

Landau Analysis: where are the branch points of $I(s_{ij}, m)$ located?

[Landau, 1960]
[Cutkosky, 1960]

[Abreu, Berghoff, Bourjaily, Britto, Correia, Duhr, Fevola, Gardi, Giroux, Hannesdottir, Helmer, Lippestreu, Matsubara-Heo, McLeod, Mizera, Panzer, Papathanasiou, Polackova, Schwartz, Tellander, Telen, Vergu, Wiesmann 2017-2025]

Euler/Twisted Period representations allow us to associate an Euler characteristic χ to a given Feynman integral

The Landau Variety can be defined as the values of $\{s_{ij}, m\}$ for which the Euler characteristic drops in value

[Chestnov, Matsubara-Heo, Munch, Takayama, 2023] [Mizera, Fevola, Telen, 2023/24]

Physics Motivated Examples (1/3)

Landau Singularities of Feynman (Euler) Integrals

Feynman integrals are Euler/Twisted Period integrals

Twist: polynomial raised to a non integer (generic) power

$$I(s_{ij}, m) = \int_0^\infty \mathcal{G}(x, s_{ij}, m)^{-d/2} \frac{dx_1}{x_1} \wedge \cdots \wedge \frac{dx_n}{x_n}$$

[Lee, 2013]

Parameters (Mandelstam variables)   Integration variables

Landau Analysis: where are the branch points of $I(s_{ij}, m)$ located?

[Landau, 1960]
[Cutkosky, 1960]

[Abreu, Berghoff, Bourjaily, Britto, Correia, Duhr, Fevola, Gardi, Giroux, Hannesdottir, Helmer, Lippestreu, Matsubara-Heo, McLeod, Mizera, Panzer, Papathanasiou, Polackova, Schwartz, Tellander, Telen, Vergu, Wiesmann 2017-2025]

Euler/Twisted Period representations allow us to associate an Euler characteristic χ to a given Feynman integral

The Landau Variety can be defined as the values of $\{s_{ij}, m\}$ for which the Euler characteristic drops in value

[Chestnov, Matsubara-Heo, Munch, Takayama, 2023]   [Mizera, Fevola, Telen, 2023/24]

χ can be computed by solving systems of polynomial equations

Physics Motivated Examples (2/3)

Computing Euler Characteristics for Landau Analysis

Computing χ: $\omega = \mathrm{d}\log\left(G(z)^{-d/2}\right)$ $\omega = -\frac{d}{2}\left(\frac{\partial_1 G}{G}dx_1 + \cdots + \frac{\partial_n G}{G}dx_n\right)$ $\chi = \#\text{ solutions to }\omega = 0$

[Lee, 2013]
[Mastrolia, Mizera 2018]

Physics Motivated Examples (2/3)

Computing Euler Characteristics for Landau Analysis

Computing χ: $\omega = \mathrm{d}\log\left(G(z)^{-d/2}\right)$ $\omega = -\frac{d}{2}\left(\frac{\partial_1 G}{G}dx_1 + \cdots + \frac{\partial_n G}{G}dx_n\right)$ $\chi = \#\text{ solutions to }\omega = 0$

[Lee, 2013]
[Mastrolia, Mizera 2018]

$$\omega = 0 \longrightarrow \boxed{\begin{array}{c} \frac{\partial_1 G}{G} = 0 \\ \vdots \\ \frac{\partial_n G}{G} = 0 \end{array}}$$

Physics Motivated Examples (2/3)

Computing Euler Characteristics for Landau Analysis

Computing χ: $\omega = \mathrm{d}\log\left(G(z)^{-d/2}\right)$ $\omega = -\frac{d}{2}\left(\frac{\partial_1 G}{G}dx_1 + \cdots + \frac{\partial_n G}{G}dx_n\right)$ $\chi = \#\ \text{solutions to}\ \omega = 0$

[Lee, 2013]
[Mastrolia, Mizera 2018]

$$\omega = 0 \longrightarrow \boxed{\begin{aligned} \frac{\partial_1 G}{G} &= 0 \\ &\vdots \\ \frac{\partial_n G}{G} &= 0 \end{aligned}} \longrightarrow \underbrace{\boxed{\begin{aligned} \partial_1 G &= 0 \\ &\vdots \\ \partial_n G &= 0 \\ 1 - x_0 G &= 0 \end{aligned}}}_{\text{system of polynomial equations!}}$$

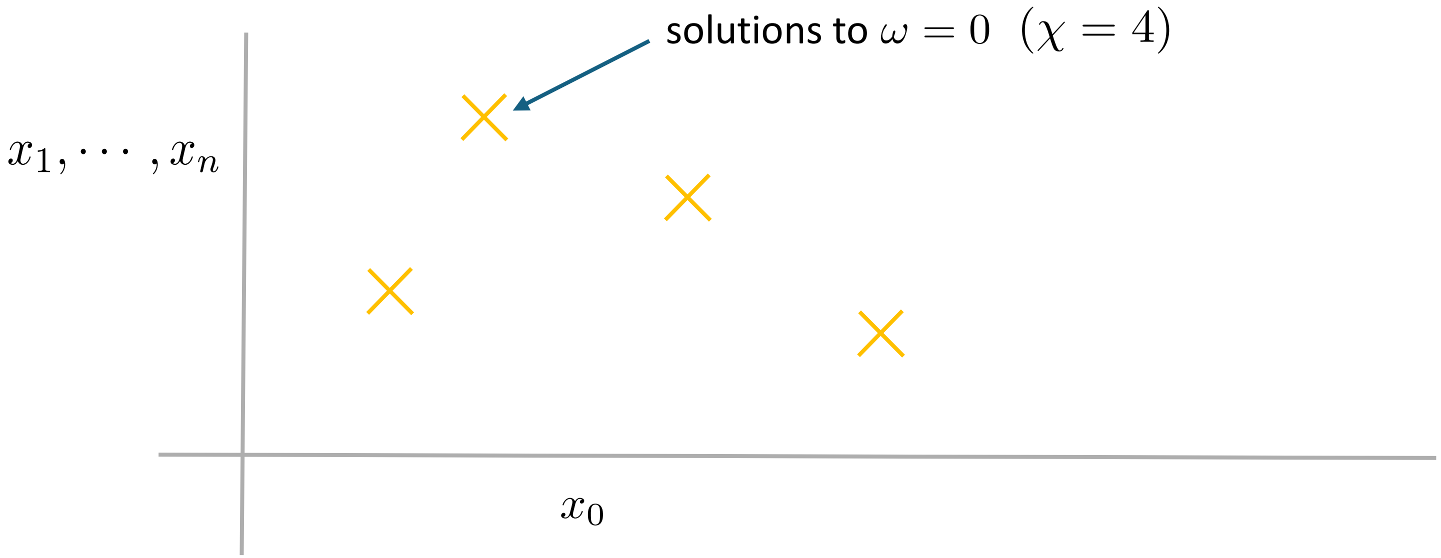
Physics Motivated Examples (2/3)

Computing Euler Characteristics for Landau Analysis

Computing χ: $\omega = \mathrm{d}\log\left(G(z)^{-d/2}\right)$ $\qquad \omega = -\frac{d}{2}\left(\frac{\partial_1 G}{G}dx_1 + \cdots + \frac{\partial_n G}{G}dx_n\right)$ $\qquad \chi = \#\ \text{solutions to } \omega = 0$

[Lee, 2013]
[Mastrolia, Mizera 2018]

$$\omega = 0 \longrightarrow \boxed{\begin{array}{c} \frac{\partial_1 G}{G} = 0 \\ \vdots \\ \frac{\partial_n G}{G} = 0 \end{array}} \longrightarrow \left.\boxed{\begin{array}{r} \partial_1 G = 0 \\ \vdots \\ \partial_n G = 0 \\ 1 - x_0 G = 0 \end{array}}\right\} \text{system of polynomial equations!}$$



Physics Motivated Examples (2/3)

Computing Euler Characteristics for Landau Analysis

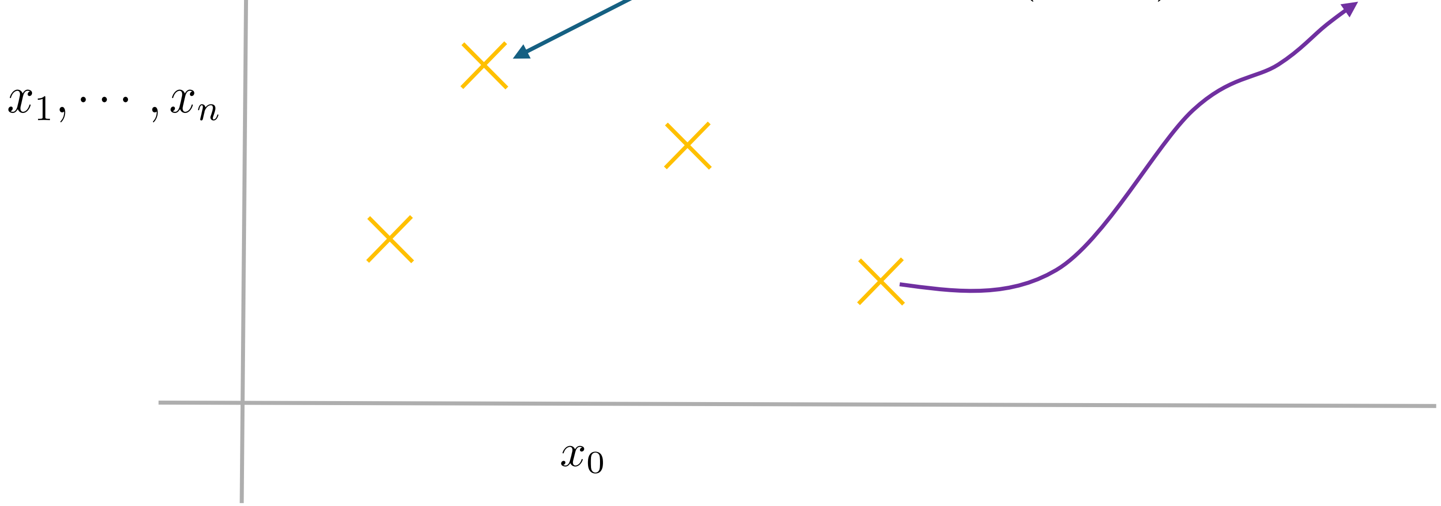
Computing χ: $\omega = \mathrm{d}\log\left(G(z)^{-d/2}\right)$ $\qquad \omega = -\frac{d}{2}\left(\frac{\partial_1 G}{G}dx_1 + \cdots + \frac{\partial_n G}{G}dx_n\right)$ $\qquad \chi = \#\ \text{solutions to } \omega = 0$

[Lee, 2013]
[Mastrolia, Mizera 2018]

$$\omega = 0 \longrightarrow \boxed{\begin{array}{c} \dfrac{\partial_1 G}{G} = 0 \\ \vdots \\ \dfrac{\partial_n G}{G} = 0 \end{array}} \longrightarrow \left.\boxed{\begin{array}{r} \partial_1 G = 0 \\ \vdots \\ \partial_n G = 0 \\ 1 - x_0 G = 0 \end{array}}\right\} \text{system of polynomial equations!}$$

solutions to $\omega = 0 \quad (\chi = 4)$

as the degenerate configuration is approached some solutions fly off to infinity



Physics Motivated Examples (2/3)

Computing Euler Characteristics for Landau Analysis

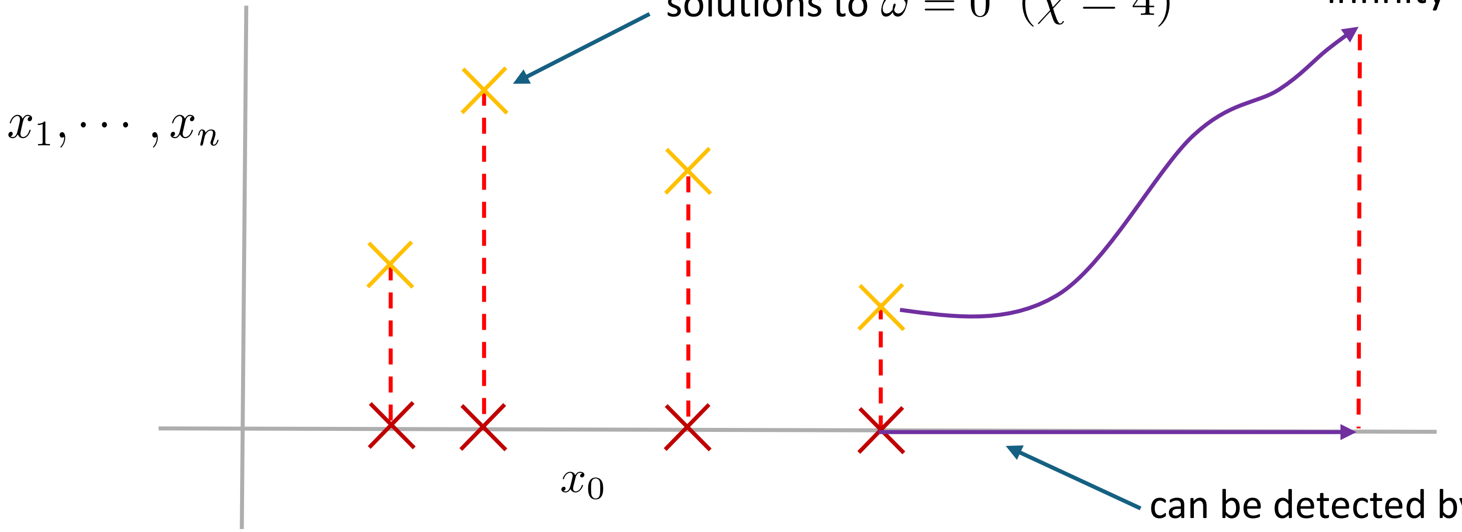
Computing χ: $\quad \omega = \mathrm{d}\log\left(G(z)^{-d/2}\right) \qquad \omega = -\frac{d}{2}\left(\frac{\partial_1 G}{G}dx_1 + \cdots + \frac{\partial_n G}{G}dx_n\right) \qquad \chi = \#\text{ solutions to } \omega = 0$

[Lee, 2013]
[Mastrolia, Mizera 2018]

$$\omega = 0 \longrightarrow \boxed{\begin{array}{c} \frac{\partial_1 G}{G} = 0 \\ \vdots \\ \frac{\partial_n G}{G} = 0 \end{array}} \longrightarrow \left.\boxed{\begin{array}{r} \partial_1 G = 0 \\ \vdots \\ \partial_n G = 0 \\ 1 - x_0 G = 0 \end{array}}\right\}\text{ system of polynomial equations!}$$

as the degenerate configuration is approached some solutions fly off to infinity

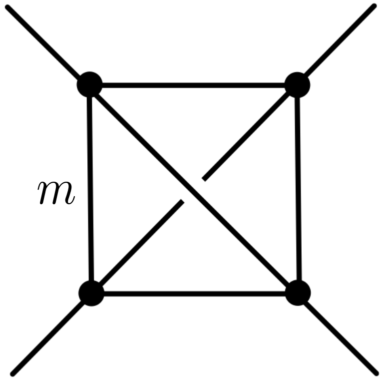
solutions to $\omega = 0 \quad (\chi = 4)$



can be detected by checking coefficients of the remaining one variable polynomial

Physics Motivated Examples (3/3)

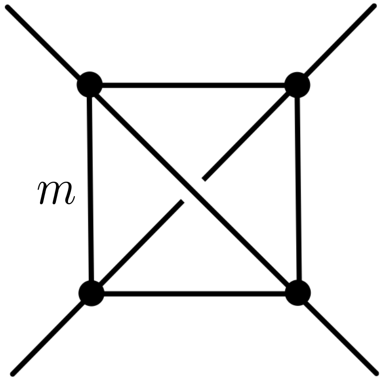
Computing Euler Characteristics for Landau Analysis: Three loop envelope (preliminary)



[Correia, Sever, Zhibodeev, 2021]

Physics Motivated Examples (3/3)

Computing Euler Characteristics for Landau Analysis: Three loop envelope (preliminary)

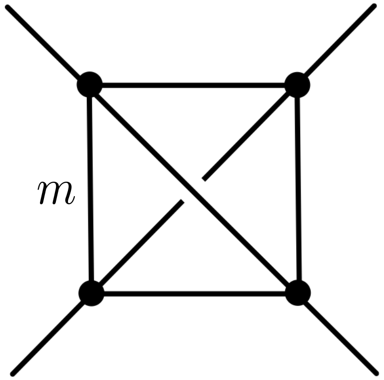


[Correia, Sever, Zhibodeev, 2021]

Horrendous integral: $\chi = 60(!)$ in the top (max cut) sector alone

Physics Motivated Examples (3/3)

Computing Euler Characteristics for Landau Analysis: Three loop envelope (preliminary)



[Correia, Sever, Zhibodeev, 2021]

Horrendous integral: $\chi = 60(!)$ in the top (max cut) sector alone

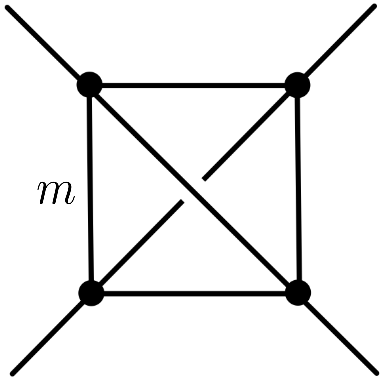
PLD/SOFIA most complicated letter found: $27(m^2)^3 + 4s^2t + 4st^2$

[Fevola, Mizera, Telen, 2023]

[Correia, Giroux, Mizera, 2025]

Physics Motivated Examples (3/3)

Computing Euler Characteristics for Landau Analysis: Three loop envelope (preliminary)



[Correia, Sever, Zhibodeev, 2021]

Horrendous integral: $\chi = 60(!)$ in the top (max cut) sector alone

PLD/SOFIA most complicated letter found: $27(m^2)^3 + 4s^2t + 4st^2$

[Fevola, Mizera, Telen, 2023]

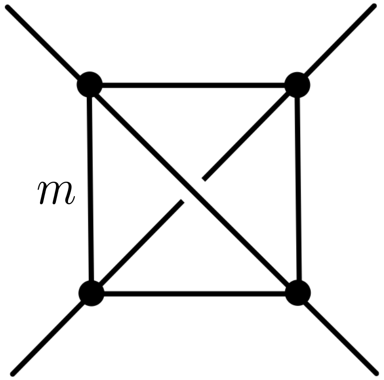
[Correia, Giroux, Mizera, 2025]

Euler characteristic strategy

Two new simple letters: $\{s^2 + st + t^2, m^2s^2 + m^2st + s^2t + m^2t^2 + st^2\}$

Physics Motivated Examples (3/3)

Computing Euler Characteristics for Landau Analysis: Three loop envelope (preliminary)



[Correia, Sever, Zhibodeev, 2021]

Horrendous integral: $\chi = 60(!)$ in the top (max cut) sector alone

PLD/SOFIA most complicated letter found: $27(m^2)^3 + 4s^2t + 4st^2$

[Fevola, Mizera, Telen, 2023]

[Correia, Giroux, Mizera, 2025]

Euler characteristic strategy

Two new simple letters: $\{s^2 + st + t^2, m^2s^2 + m^2st + s^2t + m^2t^2 + st^2\}$

Four new complicated letters:

$$
\begin{aligned}
&\{27\,\mathrm{m2}^4 s^2 + 108\,\mathrm{m2}^4 s\,t + 162\,\mathrm{m2}^3 s^2 t + 54\,\mathrm{m2}^2 s^3 t + 4\,\mathrm{m2}\,s^4 t + 108\,\mathrm{m2}^4 t^2 + 162\,\mathrm{m2}^3 s\,t^2 + 45\,\mathrm{m2}^2 s^2 t^2 - \\
&\quad 6\,\mathrm{m2}\,s^3 t^2 - s^4 t^2 - 18\,\mathrm{m2}^2 s\,t^3 - 20\,\mathrm{m2}\,s^2 t^3 - 2\,s^3 t^3 - 9\,\mathrm{m2}^2 t^4 - 10\,\mathrm{m2}\,s\,t^4 - s^2 t^4,\ 108\,\mathrm{m2}^4 s^2 - 9\,\mathrm{m2}^2 s^4 + 108\,\mathrm{m2}^4 s\,t + 162\,\mathrm{m2}^3 s^2 t - \\
&\quad 18\,\mathrm{m2}^2 s^3 t - 10\,\mathrm{m2}\,s^4 t + 27\,\mathrm{m2}^4 t^2 + 162\,\mathrm{m2}^3 s\,t^2 + 45\,\mathrm{m2}^2 s^2 t^2 - 20\,\mathrm{m2}\,s^3 t^2 - s^4 t^2 + 54\,\mathrm{m2}^2 s\,t^3 - 6\,\mathrm{m2}\,s^2 t^3 - 2\,s^3 t^3 + 4\,\mathrm{m2}\,s\,t^4 - s^2 t^4, \\
&27\,\mathrm{m2}^4 s^2 - 54\,\mathrm{m2}^4 s\,t + 162\,\mathrm{m2}^3 s^2 t - 54\,\mathrm{m2}^2 s^3 t + 4\,\mathrm{m2}\,s^4 t + 27\,\mathrm{m2}^4 t^2 + 162\,\mathrm{m2}^3 s\,t^2 - 117\,\mathrm{m2}^2 s^2 t^2 + 22\,\mathrm{m2}\,s^3 t^2 - s^4 t^2 - 54\,\mathrm{m2}^2 s\,t^3 + 22\,\mathrm{m2}\,s^2 t^3 - 2\,s^3 t^3 + 4\,\mathrm{m2}\,s\,t^4 - s^2 t^4, \\
&65\,536\,\mathrm{m2}^{12} + 270\,336\,\mathrm{m2}^{10} s^2 + 33\,024\,\mathrm{m2}^8 s^4 + 1024\,\mathrm{m2}^6 s^6 + 270\,336\,\mathrm{m2}^{10} s\,t - 458\,752\,\mathrm{m2}^9 s^2 t + 66\,048\,\mathrm{m2}^8 s^3 t - 1\,276\,416\,\mathrm{m2}^7 s^4 t + 3072\,\mathrm{m2}^6 s^5 t - 137\,472\,\mathrm{m2}^5 s^6 t - \\
&\quad 4096\,\mathrm{m2}^3 s^8 t + 270\,336\,\mathrm{m2}^{10} t^2 - 458\,752\,\mathrm{m2}^9 s\,t^2 + 99\,072\,\mathrm{m2}^8 s^2 t^2 - 2\,552\,832\,\mathrm{m2}^7 s^3 t^2 - 3\,427\,584\,\mathrm{m2}^6 s^4 t^2 - 412\,416\,\mathrm{m2}^5 s^5 t^2 + 149\,472\,\mathrm{m2}^4 s^6 t^2 - 16\,384\,\mathrm{m2}^3 s^7 t^2 + \\
&\quad 768\,\mathrm{m2}^2 s^8 t^2 + 66\,048\,\mathrm{m2}^8 s\,t^3 - 2\,552\,832\,\mathrm{m2}^7 s^2 t^3 - 6\,860\,288\,\mathrm{m2}^6 s^3 t^3 - 687\,360\,\mathrm{m2}^5 s^4 t^3 + 448\,416\,\mathrm{m2}^4 s^5 t^3 - 49\,888\,\mathrm{m2}^3 s^6 t^3 + 3072\,\mathrm{m2}^2 s^7 t^3 - 48\,\mathrm{m2}\,s^8 t^3 + \\
&\quad 33\,024\,\mathrm{m2}^8 t^4 - 1\,276\,416\,\mathrm{m2}^7 s\,t^4 - 3\,427\,584\,\mathrm{m2}^6 s^2 t^4 - 687\,360\,\mathrm{m2}^5 s^3 t^4 + 597\,888\,\mathrm{m2}^4 s^4 t^4 - 92\,320\,\mathrm{m2}^3 s^5 t^4 + 6144\,\mathrm{m2}^2 s^6 t^4 - 192\,\mathrm{m2}\,s^7 t^4 + s^8 t^4 + \\
&\quad 3072\,\mathrm{m2}^6 s\,t^5 - 412\,416\,\mathrm{m2}^5 s^2 t^5 + 448\,416\,\mathrm{m2}^4 s^3 t^5 - 92\,320\,\mathrm{m2}^3 s^4 t^5 + 7680\,\mathrm{m2}^2 s^5 t^5 - 336\,\mathrm{m2}\,s^6 t^5 + 4\,s^7 t^5 + 1024\,\mathrm{m2}^6 t^6 - 137\,472\,\mathrm{m2}^5 s\,t^6 + 149\,472\,\mathrm{m2}^4 s^2 t^6 - \\
&\quad 49\,888\,\mathrm{m2}^3 s^3 t^6 + 6144\,\mathrm{m2}^2 s^4 t^6 - 336\,\mathrm{m2}\,s^5 t^6 + 6\,s^6 t^6 - 16\,384\,\mathrm{m2}^3 s^2 t^7 + 3072\,\mathrm{m2}^2 s^3 t^7 - 192\,\mathrm{m2}\,s^4 t^7 + 4\,s^5 t^7 - 4096\,\mathrm{m2}^3 s\,t^8 + 768\,\mathrm{m2}^2 s^2 t^8 - 48\,\mathrm{m2}\,s^3 t^8 + s^4 t^8\}
\end{aligned}
$$

Conclusions and Outlook

SPQR is a new open-source Mathematica package for performing polynomial division. It leverages finite field sampling and reconstruction to process Macaulay matrices from complex polynomial systems with high efficiency.

This approach avoids heavy symbolic cancellations, often leading to significant speedups compared to traditional Groebner basis approaches.

The companion matrix formalism helps contain the Macaulay system size, making it possible to handle larger and more complicated problems

Future development directions: A way to generate a smaller set of equations for the row reduction would allow for the tackling of even more complicated problems — perhaps possible to build equation systems based on the steps carried out by Groebner Basis algorithms/interface to current Groebner basis tools?

A reconstruction algorithm that factors expressions could save many sample points for sparse rational function outputs

(Beta) version is available now at github.com/giu989/spqr

```
ResourceFunction["GitHubInstall"]["giu989","SPQR"];
```

Paper coming soon!

Thank you for listening!

(Beta) version is available now at github.com/giu989/spqr

```
ResourceFunction["GitHubInstall"]["giu989","SPQR"];
```

Paper coming soon!