Polynomial system solving with the msolve library

https://msolve.lip6.fr

J. Berthomieu[1], C. Eder[2], V. Neiger[1], M. Safey El Din[1]

[1]PolSys Team, LIP6, CNRS, Sorbonne Université, Paris, France

[2]Rheinland-Pfälzische Technische Universität Kaiserslautern Landau, Germany

Polynomial systems

Let $\mathbb{K}$, $\mathbb{K}'$ be fields with $\mathbb{K} \subset \mathbb{K}'$ and $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $R = \mathbb{K}[x_1, \ldots, x_n]$

Polynomial system solving

"Solve" $f_1 = \cdots = f_s = 0$ over ${\mathbb{K}'}^n$ $\rightsquigarrow$ Solution set over ${\mathbb{K}'}^n$

Polynomial systems

Let $\mathbb{K}, \mathbb{K}'$ be fields with $\mathbb{K} \subset \mathbb{K}'$ and $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $R = \mathbb{K}[x_1, \ldots, x_n]$

Polynomial system solving

"Solve" $f_1 = \cdots = f_s = 0$ over ${\mathbb{K}'}^n$ $\rightsquigarrow$ Solution set over ${\mathbb{K}'}^n$

Typical settings.

☛ $\mathbb{K}$ is a finite field, $\mathbb{K}' = \mathbb{K}$ or $\mathbb{K}'$ is an algebraic closure of $\mathbb{K}$ (denoted by $\overline{K}$)

☛ $\mathbb{K} = \mathbb{Q}$ and $\mathbb{K}' = \mathbb{R}$ or $\mathbb{K}' = \mathbb{C}$

Polynomial systems

Let $\mathbb{K}$, $\mathbb{K}'$ be fields with $\mathbb{K} \subset \mathbb{K}'$ and $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $R = \mathbb{K}[x_1, \ldots, x_n]$

Polynomial system solving

"Solve" $f_1 = \cdots = f_s = 0$ over ${\mathbb{K}'}^n$ $\rightsquigarrow$ Solution set over ${\mathbb{K}'}^n$

Typical settings.

☛ $\mathbb{K}$ is a finite field, $\mathbb{K}' = \mathbb{K}$ or $\mathbb{K}'$ is an algebraic closure of $\mathbb{K}$ (denoted by $\overline{K}$)

☛ $\mathbb{K} = \mathbb{Q}$ and $\mathbb{K}' = \mathbb{R}$ or $\mathbb{K}' = \mathbb{C}$

- Finiteness of the solution set in $\overline{\mathbb{K}}^n$?

Polynomial systems

Let $\mathbb{K}, \mathbb{K}'$ be fields with $\mathbb{K} \subset \mathbb{K}'$ and $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $R = \mathbb{K}[x_1, \ldots, x_n]$

Polynomial system solving

"Solve" $f_1 = \cdots = f_s = 0$ over $\mathbb{K}'^n$ $\rightsquigarrow$ Solution set over $\mathbb{K}'^n$

Typical settings.

☛ $\mathbb{K}$ is a finite field, $\mathbb{K}' = \mathbb{K}$ or $\mathbb{K}'$ is an algebraic closure of $\mathbb{K}$ (denoted by $\overline{K}$)

☛ $\mathbb{K} = \mathbb{Q}$ and $\mathbb{K}' = \mathbb{R}$ or $\mathbb{K}' = \mathbb{C}$

- Finiteness of the solution set in $\overline{\mathbb{K}}^n$?
- $\mathcal{NP}$-hardness of multivariate solving
- Bézout bound $\rightsquigarrow$ Exponential number of solutions in n

Polynomial systems

Let $\mathbb{K}$, $\mathbb{K}'$ be fields with $\mathbb{K} \subset \mathbb{K}'$ and $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $R = \mathbb{K}[x_1, \ldots, x_n]$

Polynomial system solving

"Solve" $f_1 = \cdots = f_s = 0$ over $\mathbb{K}'^n$ $\rightsquigarrow$ Solution set over $\mathbb{K}'^n$

Typical settings.

☛ $\mathbb{K}$ is a finite field, $\mathbb{K}' = \mathbb{K}$ or $\mathbb{K}'$ is an algebraic closure of $\mathbb{K}$ (denoted by $\overline{K}$)

☛ $\mathbb{K} = \mathbb{Q}$ and $\mathbb{K}' = \mathbb{R}$ or $\mathbb{K}' = \mathbb{C}$

- Finiteness of the solution set in $\overline{\mathbb{K}}^n$?
- $\mathcal{NP}$-hardness of multivariate solving
- Bézout bound $\rightsquigarrow$ Exponential number of solutions in n
- Non-linearity $\rightsquigarrow$ numerical issues

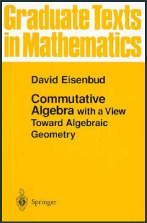
Algebra and geometry of polynomial system solving

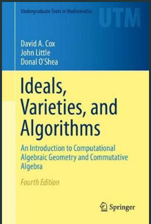
Algebraic representation → **Exact encoding** of the solution set

Equations → **Computing with solutions**

Zero test | **Sign determination** | **Dimension** | **Degree**

The algebra / geometry dictionary





☛ Non-linearity

⇝ vector spaces replaced by ideals

$$\langle f \rangle = \{ \sum_{i=1}^{s} q_i f_i, \, q_i \in R \}$$

☛ Need of normal forms (zero test)

⇝ computing "modulo" $I = \langle f \rangle$

The msolve library

plain C library implemented by Berthomieu, Eder, S.

$\simeq$ 55 000 lines, license GPLv2+

uses GMP and FLINT

https://msolve.lip6.fr

The msolve library

plain C library implemented by Berthomieu, Eder, S.

$\simeq$ 55 000 lines, license GPLv2+

uses GMP and FLINT

https://msolve.lip6.fr

https://github.com/algebraic-solving/msolve

~~https://gitlab.lip6.fr/safey/msolve~~

The msolve library

plain C library implemented by Berthomieu, Eder, S.

$\simeq$ 55 000 lines, license GPLv2+

uses GMP and FLINT

https://msolve.lip6.fr

https://github.com/algebraic-solving/msolve

~~https://gitlab.lip6.fr/safey/msolve~~

https://algebraic-solving.github.io/

The msolve library

plain C library implemented by Berthomieu, Eder, S.

$\simeq$ 55 000 lines, license GPLv2+

uses GMP and FLINT

`https://msolve.lip6.fr`

`https://github.com/algebraic-solving/msolve`

~~`https://gitlab.lip6.fr/safey/msolve`~~

`https://algebraic-solving.github.io/`

The msolve library

plain C library implemented by Berthomieu, Eder, S.

$\simeq$ 55 000 lines, license GPLv2+

uses GMP and FLINT

https://msolve.lip6.fr

https://github.com/algebraic-solving/msolve

~~https://gitlab.lip6.fr/safey/msolve~~

https://algebraic-solving.github.io/

msolve's background: Gröbner bases

Let $\succ$ be an admissible monomial ordering $\rightsquigarrow \mathrm{lm}_{\succ}(f)$ for any $f \in R$

Gröbner bases $G \subset I$ finite such that $\langle \mathrm{lm}_{\succ}(G) \rangle = \langle \mathrm{lm}_{\succ}(I) \rangle$

msolve's background: Gröbner bases

Let $\succ$ be an admissible monomial ordering $\rightsquigarrow \mathrm{lm}_\succ(f)$ for any $f \in R$

Gröbner bases $G \subset I$ finite such that $\langle \mathrm{lm}_\succ(G) \rangle = \langle \mathrm{lm}_\succ(I) \rangle$

Lexicographic ordering $\rightsquigarrow$ eliminates variables

The elimination theorem

Projection $\leftrightarrow$ Elimination

$$\begin{cases} G_1 = I \cap \mathbb{K}[x_1] \\ \vdots \\ G_n = I \cap \mathbb{K}[x_1, \ldots, x_n] \end{cases}$$

msolve's background: Gröbner bases

Let $\succ$ be an admissible monomial ordering $\rightsquigarrow \mathrm{lm}_{\succ}(f)$ for any $f \in R$

Gröbner bases $G \subset I$ finite such that $\langle \mathrm{lm}_{\succ}(G) \rangle = \langle \mathrm{lm}_{\succ}(I) \rangle$

Lexicographic ordering $\rightsquigarrow$ eliminates variables

The elimination theorem

Projection $\leftrightarrow$ Elimination

$$\begin{cases} G_1 = I \cap \mathbb{K}[x_1] \\ \vdots \\ G_n = I \cap \mathbb{K}[x_1, \ldots, x_n] \end{cases}$$

Description of finite solution set $V(I)$ in $\overline{\mathbb{K}}^n$

Ideals in shape position

Possible up to generic
linear change of coordinates

$$\begin{cases} w(x_1) = 0 \\ x_2 = w_2(x_1) \\ \vdots \\ x_n = w_n(x_1) \end{cases}$$

msolve's background: Gröbner bases

Let $\succ$ be an admissible monomial ordering $\rightsquigarrow$ $\mathrm{lm}_\succ(f)$ for any $f \in R$

Gröbner bases $G \subset I$ finite such that $\langle \mathrm{lm}_\succ(G) \rangle = \langle \mathrm{lm}_\succ(I) \rangle$

Lexicographic ordering $\rightsquigarrow$ eliminates variables

The elimination theorem

Projection $\leftrightarrow$ Elimination

$$\begin{cases} G_1 & = & I \cap \mathbb{K}[x_1] \\ & \vdots & \\ G_n & = & I \cap \mathbb{K}[x_1, \ldots, x_n] \end{cases}$$

Description of finite solution set $V(I)$ in $\overline{\mathbb{K}}^n$

Ideals in shape position

Possible up to generic
linear change of coordinates

$$\begin{cases} w(x_1) & = & 0 \\ x_2 & = & v_2(x_1)/w'(x_1) \\ & \vdots & \\ x_n & = & v_n(x_1)/w'(x_1) \end{cases}$$

Complexity issues

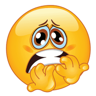
Grete Hermann. *Die Frage der endlich vielen Schritte in der Theorie der Polynomideale.* Math. Ann. 1926.

Constructive method, **doubly exponential bounds**.

Complexity issues

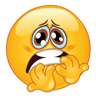
Grete Hermann. *Die Frage der endlich vielen Schritte in der Theorie der Polynomideale.* Math. Ann. 1926. Constructive method, **doubly exponential bounds**.

Mayr-Meyer'82 These bounds are "unavoidable".



Complexity issues

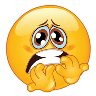
Grete Hermann. *Die Frage der endlich vielen Schritte in der Theorie der Polynomideale.* Math. Ann. 1926.
Constructive method, **doubly exponential bounds**.
Mayr-Meyer'82 These bounds are "unavoidable".



Giusti/Lecerf/Salvy, Lecerf ☛ System solving: polynomial in the Bézout bound

Complexity issues

Grete Hermann. *Die Frage der endlich vielen Schritte in der Theorie der Polynomideale.* Math. Ann. 1926.
Constructive method, **doubly exponential bounds**.
Mayr-Meyer'82 These bounds are "unavoidable".

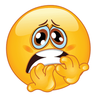


Giusti/Lecerf/Salvy, Lecerf ☛ System solving: polynomial in the Bézout bound

- ☛ Is the worst case the "generic" one? **NO!**
- ☛ Better complexity through extra requirements ? **YES!**

Complexity issues

Grete Hermann. *Die Frage der endlich vielen Schritte in der Theorie der Polynomideale.* Math. Ann. 1926.
Constructive method, **doubly exponential bounds**.
Mayr-Meyer'82 These bounds are "unavoidable".



Giusti/Lecerf/Salvy, Lecerf ☛ System solving: polynomial in the Bézout bound

☛ Is the worst case the "generic" one? **NO!**
☛ Better complexity through extra requirements ? **YES!**

Regular computations. **Bayer/Stillman/Lazard/Giusti, etc.**

$$E_d = \{\sum_{i=1}^{s} q_i f_i \mid q_i \in R, \deg(q_i f_i) \leq d\} \rightsquigarrow \text{finite dim. vector space}$$
$$B_{\succ,d} = \text{Basis of } E_d \text{ w.r.t. } \succ = \succ_{\text{graded}}$$
$$\langle \mathrm{lm}_{\succ}(B_{\succ,d}) \rangle = \langle \mathrm{lm}_{\succ}(I \cap R_{\leq d}) \rangle ?$$

Complexity

$$O\left(\left(\binom{n+\mathbb{D}_{\text{reg}}}{n}\right)^{\omega}\right) \text{ with } \mathbb{D}_{\text{reg}} = 1 + \sum_{i=1}^{s} (\deg(f_i) - 1)$$

Linearization technique and termination

$E_d = \{\sum_{i=1}^{s} q_i f_i \mid q_i \in R, \deg(q_i f_i) \leq d\} \rightsquigarrow$ finite dim. vector space

$B_d = \succ$-Basis of E_d with $\succ = \succ_{\text{graded}}$

$E_d \to B_{\succ,d}$ $\xrightarrow[\text{Gauss}]{\text{Generate}}$ $E_{d+1} \to B_{\succ,d+1}$ $\xrightarrow[\text{Gauss}]{\text{Generate}}$ $E_{d+2} \to B_{\succ,d+2}$

$\langle \mathrm{lm}_\succ(B_{\succ,\delta}) \rangle = \langle \mathrm{lm}_\succ(I) \rangle$?

Buchberger's criterion

☛ Multivariate division $\leftrightarrow$ Gaussian elimination

F4 algorithm

Faugère'98

$G \leftarrow (f_1, \ldots, f_s)$

$\{(a_{i,j} g_i, b_{i,j} g_j) \mid \mathsf{lm}_\succ(a_{i,j} g_i) = \mathsf{lm}_\succ(b_{i,j} g_j) = \mathsf{lcm}(\mathsf{lm}_\succ(g_i), \mathsf{lm}_\succ(g_j))\}$

$\mathscr{P} \leftarrow \text{Pairs}(G, \succ)$

F4 algorithm

Faugère'98

$G \leftarrow (f_1, \dots, f_s)$

$\{(a_{i,j} g_i, b_{i,j} g_j) \mid \mathrm{lm}_\succ(a_{i,j} g_i) = \mathrm{lm}_\succ(b_{i,j} g_j) = \mathrm{lcm}(\mathrm{lm}_\succ(g_i), \mathrm{lm}_\succ(g_j))\}$

$\mathscr{P} \leftarrow \text{Pairs}(G, \succ)$

Selection of the lcms of degree $d_{\min}$

$\mathscr{P}' \leftarrow \text{Select}(\mathscr{P})$ $\quad \mathscr{P} \leftarrow \mathscr{P} \setminus \mathscr{P}'$

$L \leftarrow \{af, bg \mid (af, bg) \in \mathscr{P}'\}$

$L' \leftarrow \text{SymbolicPreprocessing}(L, G)$

F4 algorithm

Faugère'98

$G \leftarrow (f_1, \dots, f_s)$

$\{(a_{i,j}g_i, b_{i,j}g_j) \mid \mathrm{lm}_\succ(a_{i,j}g_i) = \mathrm{lm}_\succ(b_{i,j}g_j) = \mathrm{lcm}(\mathrm{lm}_\succ(g_i), \mathrm{lm}_\succ(g_j))\}$

$\mathscr{P} \leftarrow \mathrm{Pairs}(G, \succ)$

Selection of the lcms of degree $d_{\min}$

$\mathscr{P}' \leftarrow \mathrm{Select}(\mathscr{P})$ $\mathscr{P} \leftarrow \mathscr{P} \setminus \mathscr{P}'$

$L \leftarrow \{af, bg \mid (af, bg) \in \mathscr{P}'\}$

Basis of $E_{d_{\min}}$

$L' \leftarrow \mathrm{SymbolicPreprocessing}(L, G)$

$H \leftarrow \mathrm{GaussianReduction}(\mathrm{Macaulay}(L', \succ))$

for $h \in H$, if $\mathrm{lm}_\succ(h) \notin \langle \mathrm{lm}_\succ(G) \rangle$
  $\mathscr{P} \leftarrow \mathscr{P} \cup \mathrm{Update}(G, h, \succ)$, $G \leftarrow G \cup \{h\}$

F4 algorithm

Faugère'98

$G \leftarrow (f_1, \ldots, f_s)$

$\{(a_{i,j}g_i, b_{i,j}g_j) \mid \mathrm{lm}_\succ(a_{i,j}g_i) = \mathrm{lm}_\succ(b_{i,j}g_j) = \mathrm{lcm}(\mathrm{lm}_\succ(g_i), \mathrm{lm}_\succ(g_j))\}$

$\mathscr{P} \leftarrow \mathrm{Pairs}(G, \succ)$

Selection of the lcms of degree $d_{\min}$

$\mathscr{P}' \leftarrow \mathrm{Select}(\mathscr{P})$ $\quad \mathscr{P} \leftarrow \mathscr{P} \setminus \mathscr{P}'$

$L \leftarrow \{af, bg \mid (af, bg) \in \mathscr{P}'\}$

Basis of $E_{d_{\min}}$

$L' \leftarrow \mathrm{SymbolicPreprocessing}(L, G)$

$H \leftarrow \mathrm{GaussianReduction}(\mathrm{Macaulay}(L', \succ))$

for $h \in H$, if $\mathrm{lm}_\succ(h) \notin \langle \mathrm{lm}_\succ(G) \rangle$

$\quad \mathscr{P} \leftarrow \mathscr{P} \cup \mathrm{Update}(G, h, \succ)$, $G \leftarrow G \cup \{h\}$

$\mathscr{P} \neq \emptyset$?

F4 algorithm

Faugère'98

$g_j) = \mathrm{lcm}(\mathrm{lm}_\succ(g_i), \mathrm{lm}_\succ(g_j))\}$

More accurate "complexity estimate":

- ☛ Number of matrices + their sizes + their ranks
- ☛ Number of reducers and new elements in the GB

of the lcms of degree $d_{\min}$

$\mathscr{P}' \leftarrow \mathrm{Select}(\mathscr{P})$ $\quad \mathscr{P} \leftarrow \mathscr{P} \setminus \mathscr{P}'$

Basis of $E_{d_{\min}}$

$L \leftarrow \{af, bg \mid (af, bg) \in \mathscr{P}'\}$

$L' \leftarrow \mathrm{SymbolicPreprocessing}(L, G)$

$H \leftarrow \mathrm{GaussianReduction}(\mathrm{Macaulay}(L', \succ))$

$\mathscr{P} \neq \emptyset$?

for $h \in H$, if $\mathrm{lm}_\succ(h) \notin \langle \mathrm{lm}_\succ(G) \rangle$

$\mathscr{P} \leftarrow \mathscr{P} \cup \mathrm{Update}(G, h, \succ)$, $G \leftarrow G \cup \{h\}$

F4 algorithm

Faugère'98

$g_j) = \mathrm{lcm}(\mathrm{lm}_\succ(g_i), \mathrm{lm}_\succ(g_j))\}$

More accurate "complexity estimate":

☛ Number of matrices + their sizes + their ranks

☛ Number of reducers and new elements in the GB

of the lcms of degree $d_{\min}$

☛ Many terms of reducers only useful "locally" (for their specific Macaulay-like matrix)

☛ To **save memory**: one global hash table + one secondary local hash table

Basis of $E_{d_{\min}}$

$H \leftarrow$ GaussianReduction(Macaulay$(L', \succ)$)

for $h \in H$, if $\mathrm{lm}_\succ(h) \notin \langle \mathrm{lm}_\succ(G) \rangle$

$\mathscr{P} \leftarrow \mathscr{P} \cup$ Update$(G, h, \succ)$, $G \leftarrow G \cup \{h\}$

$\mathscr{P} \neq \emptyset$?

$g_j) = \mathsf{lcm}(\mathsf{lm}_\succ(g_i), \mathsf{lm}_\succ(g_j))\}$

More accurate "complexity estimate":

☛ Number of matrices + their sizes + their ranks

☛ Number of reducers and new elements in the GB

of the lcms of degree $d_{\min}$

☛ Many terms of reducers only useful "locally" (for their specific Macaulay-like matrix)

☛ To **save memory**: one global hash table + one secondary local hash table

Basis of $E_{d_{\min}}$

☛ Need of **fast** divisibility checks

☛ Use of divisor masks

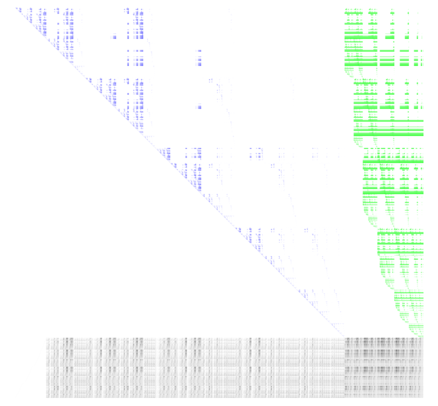
$y(L', \succ))$

for $h \in H$, if $\mathsf{lm}_\succ(h) \notin \langle \mathsf{lm}_\succ(G) \rangle$

$\mathscr{P} \leftarrow \mathscr{P} \cup \mathsf{Update}(G, h, \succ)$, $G \leftarrow G \cup \{h\}$

$\mathscr{P} \neq \emptyset$?

Matrices in F4



☛ rows stored in general in sparse format

☛ rows stored in sparse-dense hybrid format for denser matrices

☛ CPU intrinsics: AVX2 $\rightsquigarrow$ store eight 32-bit (unsigned) coefficients in one 256-bit `__m256i` type

☛ Probabilistic and deterministic reductions

☛ Implementation of a tracer for multi-modular computations

Traverso'88

F4 timings

Gröbner bases for grevlex order computations modulo primes $< 2^{31}$

```
./msolve -g 2 -f in.ms -o out.ms
./msolve -g 1 -f in.ms -o out.ms
```

Examples	msolve F4 learn	msolve F4 tracer	(learn/tracer)	msolve prob	(prob / tracer)	maple	magma
Katsura-9	0.17	0.03	5.67	0.06	2	0.10	
Katsura-10	0.81	0.09	9	0.24	2.67	0.36	
Katsura-11	6.26	0.45	13.9	1.34	2.98	1.82	
Katsura-12	56.1	3.10	18.1	8.61	2.78	8.50	
Katsura-13	425	19	22.4	53	2.79	60.9	
Katsura-14	3336	128	26.1	318	2.5	393	
Katsura-15	27960	738	27.96	2209	2.71	n.m.	
Katsura-16	259240	5548	46.7	12474	2.24	n.m.	

F4 timings

Gröbner bases for grevlex order computations modulo primes $< 2^{31}$

```
./msolve -g 2 -f in.ms -o out.ms
./msolve -g 1 -f in.ms -o out.ms
```

Examples	msolve F4 learn	msolve F4 tracer	(learn/tracer)	msolve prob	(prob / tracer)	maple	magma
Katsura-9	0.17	0.03	5.67	0.06	2	0.10	
Katsura-10	0.81	0.09	9	0.24	2.67	0.36	
Katsura-11	6.26	0.45	13.9	1.34	2.98	1.82	
Katsura-12	56.1	3.10	18.1	8.61	2.78	8.50	
Katsura-13	425	19	22.4	53	2.79	60.9	
Katsura-14	3336	128	26.1	318	2.5	393	
Katsura-15	27960	738	27.96	2209	2.71	n.m.	
Katsura-16	259240	5548	46.7	12474	2.24	n.m.	
Katsura-11	3.60			1.15			1.63
Katsura-12	28.53			6.30			9.10
Katsura-13	246.37			39.43			57.77

F4 timings

Gröbner bases for grevlex order computations modulo primes $< 2^{31}$

```
./msolve -g 2 -f in.ms -o out.ms
./msolve -g 1 -f in.ms -o out.ms
```

Examples	msolve F4 learn	msolve F4 tracer	(learn/tracer)	msolve prob	(prob / tracer)	maple	magma
Katsura-9	0.17	0.03	5.67	0.06	2	0.10	
Katsura-10	0.81	0.09	9	0.24	2.67	0.36	
Katsura-11	6.26	0.45	13.9	1.34	2.98	1.82	
Katsura-12	56.1	3.10	18.1	8.61	2.78	8.50	
Katsura-13	425	19	22.4	53	2.79	60.9	
Katsura-14	3336	128	26.1	318	2.5	393	
Katsura-15	27960	738	27.96	2209	2.71	n.m.	
Katsura-16	259240	5548	46.7	12474	2.24	n.m.	
Katsura-11	3.60			1.15			1.63
Katsura-12	28.53			6.30			9.10
Katsura-13	246.37			39.43			57.77
Eco-10	0.28	0.05	5.6	0.1	2	0.14	
Eco-11	1.21	0.17	7.11	0.39	2.29	0.56	
Eco-12	11.6	1.1	10.54	2.25	2.05	2.97	
Eco-13	67.3	6.6	10.2	11.7	1.77	15.1	
Eco-14	516	34.8	14.8	67	1.92	104.8	
Eco-15	3476	153	22.7	466.15	3	n.m.	

F4 timings

Gröbner bases for grevlex order computations modulo primes $< 2^{31}$

```
./msolve -g 2 -f in.ms -o out.ms
./msolve -g 1 -f in.ms -o out.ms
```

Examples	msolve F4 learn	msolve F4 tracer	(learn/tracer)	msolve prob	(prob / tracer)	maple	magma
Katsura-9	0.17	0.03	5.67	0.06	2	0.10	
Katsura-10	0.81	0.09	9	0.24	2.67	0.36	
Katsura-11	6.26	0.45	13.9	1.34	2.98	1.82	
Katsura-12	56.1	3.10	18.1	8.61	2.78	8.50	
Katsura-13	425	19	22.4	53	2.79	60.9	
Katsura-14	3336	128	26.1	318	2.5	393	
Katsura-15	27960	738	27.96	2209	2.71	n.m.	
Katsura-16	259240	5548	46.7	12474	2.24	n.m.	
Katsura-11	3.60			1.15			1.63
Katsura-12	28.53			6.30			9.10
Katsura-13	246.37			39.43			57.77
Eco-10	0.28	0.05	5.6	0.1	2	0.14	
Eco-11	1.21	0.17	7.11	0.39	2.29	0.56	
Eco-12	11.6	1.1	10.54	2.25	2.05	2.97	
Eco-13	67.3	6.6	10.2	11.7	1.77	15.1	
Eco-14	516	34.8	14.8	67	1.92	104.8	
Eco-15	3476	153	22.7	466.15	3	n.m.	
Eco-11	0.71			0.37			0.46
Eco-12	4.94			1.95			2.61
Eco-13	33.75			9.27			11.77

F4 timings

Gröbner bases for grevlex order computations modulo primes $< 2^{31}$

```
./msolve -g 2 -f in.ms -o out.ms
./msolve -g 1 -f in.ms -o out.ms
```

Examples	msolve F4 learn	msolve F4 tracer	(learn/tracer)	msolve prob	(prob / tracer)	maple	magma
Katsura-9	0.17	0.03	5.67	0.06	2	0.10	
Katsura-10	0.81	0.09	9	0.24	2.67	0.36	
Katsura-11	6.26	0.45	13.9	1.34	2.98	1.82	
Katsura-12	56.1	3.10	18.1	8.61	2.78	8.50	
Katsura-13	425	19	22.4	53	2.79	60.9	
Katsura-14	3336	128	26.1	318	2.5	393	
Katsura-15	27960	738	27.96	2209	2.71	n.m.	
Katsura-16	259240	5548	46.7	12474	2.24	n.m.	
Katsura-11	3.60			1.15			1.63
Katsura-12	28.53			6.30			9.10
Katsura-13	246.37			39.43			57.77
Eco-10	0.28	0.05	5.6	0.1	2	0.14	
Eco-11	1.21	0.17	7.11	0.39	2.29	0.56	
Eco-12	11.6	1.1	10.54	2.25	2.05	2.97	
Eco-13	67.3	6.6	10.2	11.7	1.77	15.1	
Eco-14	516	34.8	14.8	67	1.92	104.8	
Eco-15	3476	153	22.7	466.15	3	n.m.	
Eco-11	0.71			0.37			0.46
Eco-12	4.94			1.95			2.61
Eco-13	33.75			9.27			11.77
Henrion-6	0.22	0.07	3.14	0.11	1.57	0.17	
Henrion-7	27.5	6.5	4.23	9.55	1.47	12.8	

F4 timings

Gröbner bases for grevlex order computations modulo primes $< 2^{31}$

```
./msolve -g 2 -f in.ms -o out.ms
./msolve -g 1 -f in.ms -o out.ms
```

Examples	msolve F4 learn	msolve F4 tracer	(learn/tracer)	msolve prob	(prob / tracer)	maple	magma
Katsura-9	0.17	0.03	5.67	0.06	2	0.10	
Katsura-10	0.81	0.09	9	0.24	2.67	0.36	
Katsura-11	6.26	0.45	13.9	1.34	2.98	1.82	
Katsura-12	56.1	3.10	18.1	8.61	2.78	8.50	
Katsura-13	425	19	22.4	53	2.79	60.9	
Katsura-14	3336	128	26.1	318	2.5	393	
Katsura-15	27960	738	27.96	2209	2.71	n.m.	
Katsura-16	259240	5548	46.7	12474	2.24	n.m.	
Katsura-11	3.60			1.15			1.63
Katsura-12	28.53			6.30			9.10
Katsura-13	246.37			39.43			57.77
Eco-10	0.28	0.05	5.6	0.1	2	0.14	
Eco-11	1.21	0.17	7.11	0.39	2.29	0.56	
Eco-12	11.6	1.1	10.54	2.25	2.05	2.97	
Eco-13	67.3	6.6	10.2	11.7	1.77	15.1	
Eco-14	516	34.8	14.8	67	1.92	104.8	
Eco-15	3476	153	22.7	466.15	3	n.m.	
Eco-11	0.71			0.37			0.46
Eco-12	4.94			1.95			2.61
Eco-13	33.75			9.27			11.77
Henrion-6	0.22	0.07	3.14	0.11	1.57	0.17	
Henrion-7	27.5	6.5	4.23	9.55	1.47	12.8	
CP(3,6,2)	0.6	0.12	5	0.22	1.83	0.31	
CP(3,7,2)	8.18	1.23	6.65	1.97	1.6	2.78	
CP(3,8,2)	111.5	12.6	8.85	18.5	1.47	24.6	

F4 timings

Gröbner bases for grevlex order computations modulo primes $< 2^{31}$

```
./msolve -g 2 -f in.ms -o out.ms
./msolve -g 1 -f in.ms -o out.ms
```

Examples	msolve F4 learn	msolve F4 tracer	(learn/tracer)	msolve prob	(prob / tracer)	maple	magma
Katsura-9	0.17	0.03	5.67	0.06	2	0.10	
Katsura-10	0.81	0.09	9	0.24	2.67	0.36	
Katsura-11	6.26	0.45	13.9	1.34	2.98	1.82	
Katsura-12	56.1	3.10	18.1	8.61	2.78	8.50	
Katsura-13	425	19	22.4	53	2.79	60.9	
Katsura-14	3336	128	26.1	318	2.5	393	
Katsura-15	27960	738	27.96	2209	2.71	n.m.	
Katsura-16	259240	5548	46.7	12474	2.24	n.m.	
Katsura-11	3.60			1.15			1.63
Katsura-12	28.53			6.30			9.10
Katsura-13	246.37			39.43			57.77
Eco-10	0.28	0.05	5.6	0.1	2	0.14	
Eco-11	1.21	0.17	7.11	0.39	2.29	0.56	
Eco-12	11.6	1.1	10.54	2.25	2.05	2.97	
Eco-13	67.3	6.6	10.2	11.7	1.77	15.1	
Eco-14	516	34.8	14.8	67	1.92	104.8	
Eco-15	3476	153	22.7	466.15	3	n.m.	
Eco-11	0.71			0.37			0.46
Eco-12	4.94			1.95			2.61
Eco-13	33.75			9.27			11.77
Henrion-6	0.22	0.07	3.14	0.11	1.57	0.17	
Henrion-7	27.5	6.5	4.23	9.55	1.47	12.8	
CP(3,6,2)	0.6	0.12	5	0.22	1.83	0.31	
CP(3,7,2)	8.18	1.23	6.65	1.97	1.6	2.78	
CP(3,8,2)	111.5	12.6	8.85	18.5	1.47	24.6	
Pol-Bill	190	-	-			348	291
SDK-Bill	150	-	-			268	4208

F4 timings

Gröbner bases for grevlex order computations modulo primes $< 2^{31}$

```
./msolve -g 2 -f in.ms -o out.ms
./msolve -g 1 -f in.ms -o out.ms
```

Examples	msolve F4 learn	msolve F4 tracer	(learn/tracer)	msolve prob	(prob / tracer)	maple	magma
Katsura-9	0.17	0.03	5.67	0.06	2	0.10	
Katsura-10	0.81	0.09	9	0.24	2.67	0.36	
Katsura-11	6.26	0.45	13.9	1.34	2.98	1.82	
Katsura-12	56.1	3.10	18.1	8.61	2.78	8.50	
Katsura-13	425	19	22.4	53	2.79	60.9	
Katsura-14	3336	128	26.1	318	2.5	393	
Katsura-15	27960	738	27.96	2209	2.71	n.m.	
Katsura-16	259240	5548	46.7	12474	2.24	n.m.	
Katsura-11	3.60			1.15			1.63
Katsura-12	28.53			6.30			9.10
Katsura-13	246.37			39.43			57.77
Eco-10	0.28	0.05	5.6	0.1	2	0.14	
Eco-11	1.21	0.17	7.11	0.39	2.29	0.56	
Eco-12	11.6	1.1	10.54	2.25	2.05	2.97	
Eco-13	67.3	6.6	10.2	11.7	1.77	15.1	
Eco-14	516	34.8	14.8	67	1.92	104.8	
Eco-15	3476	153	22.7	466.15	3	n.m.	
Eco-11	0.71			0.37			0.46
Eco-12	4.94			1.95			2.61
Eco-13	33.75			9.27			11.77
Henrion-6	0.22	0.07	3.14	0.11	1.57	0.17	
Henrion-7	27.5	6.5	4.23	9.55	1.47	12.8	
CP(3,6,2)	0.6	0.12	5	0.22	1.83	0.31	
CP(3,7,2)	8.18	1.23	6.65	1.97	1.6	2.78	
CP(3,8,2)	111.5	12.6	8.85	18.5	1.47	24.6	
Pol-Bill	190	-	-			348	291
SDK-Bill	150	-	-			268	4208

F4 timings

Gröbner bases for grevlex order computations modulo primes $< 2^{31}$

```
./msolve -g 2 -f in.ms -o out.ms
./msolve -g 1 -f in.ms -o out.ms
```

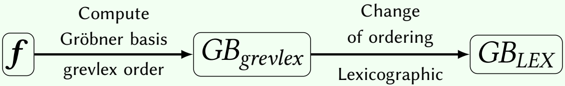
Examples	msolve F4 learn	msolve F4 tracer	(learn/tracer)	msolve prob	(prob / tracer)	maple	magma
Katsura-9	0.17	0.03	5.67	0.06	2	0.10	
Katsura-10	0.81	0.09	9	0.24	2.67	0.36	
Katsura-11	6.26	0.45	13.9	1.34	2.98	1.82	
Katsura-12	56.1	3.10	18.1	8.61	2.78	8.50	
Katsura-13	425	19	22.4	53	2.79	60.9	
Katsura-14	3336	128	26.1	318	2.5	393	
Katsura-15	27960	738	27.96	2209	2.71	n.m.	
Katsura-16	259240	5548	46.7	12474	2.24	n.m.	
Katsura-11	3.60			1.15			1.63
Katsura-12	28.53			6.30			9.10
Katsura-13	246.37			39.43			57.77
Eco-10	0.28	0.05	5.6	0.1	2	0.14	
Eco-11	1.21	0.17	7.11	0.39	2.29	0.56	
Eco-12	11.6	1.1	10.54	2.25	2.05	2.97	
Eco-13	67.3	6.6	10.2	11.7	1.77	15.1	
Eco-14	516	34.8	14.8	67	1.92	104.8	
Eco-15	3476	153	22.7	466.15	3	n.m.	
Eco-11	0.71			0.37			0.46
Eco-12	4.94			1.95			2.61
Eco-13	33.75			9.27			11.77
Henrion-6	0.22	0.07	3.14	0.11	1.57	0.17	
Henrion-7	27.5	6.5	4.23	9.55	1.47	12.8	
CP(3,6,2)	0.6	0.12	5	0.22	1.83	0.31	
CP(3,7,2)	8.18	1.23	6.65	1.97	1.6	2.78	
CP(3,8,2)	111.5	12.6	8.85	18.5	1.47	24.6	
Pol-Bill	190	-	-			348	291
SDK-Bill	150	-	-			268	4208

Grevlex one block elimination orderings are also available

```
./msolve -e k -g 2 -f in.ms -o out.ms
./msolve -e k -g 1 -f in.ms -o out.ms
```

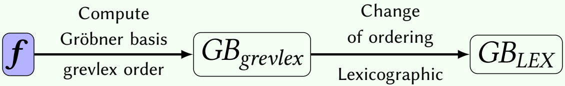
Describing solutions ⇝ Change of orders

The "usual" good way to do



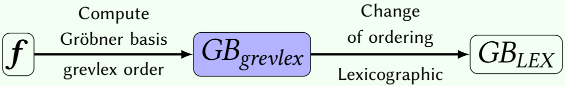
Describing solutions $\rightsquigarrow$ Change of orders

The "usual" good way to do



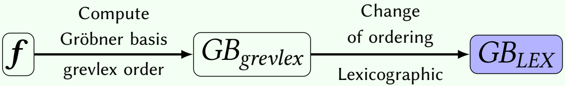
Describing solutions $\rightsquigarrow$ Change of orders

The "usual" good way to do



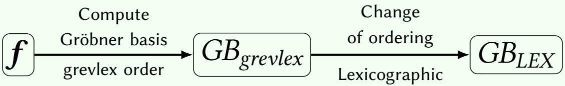
Describing solutions $\rightsquigarrow$ Change of orders

The "usual" good way to do



Describing solutions $\rightsquigarrow$ Change of orders

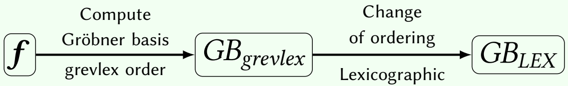
The "usual" good way to do



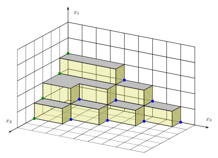
$\frac{\mathbb{K}[x_1,\ldots,x_n]}{I}$ is a finite dimensional vector space

Describing solutions $\rightsquigarrow$ Change of orders

The "usual" good way to do



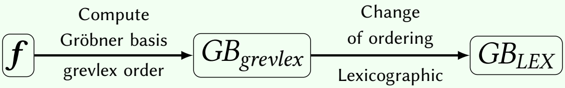
$\frac{\mathbb{K}[x_1,\ldots,x_n]}{I}$ is a finite dimensional vector space



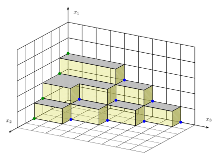
- Combinatorial structure of polynomial ideals
- Basis $\mathscr{B}$ of quotient ring $\frac{\mathbb{K}[x_1,\ldots,x_n]}{I}$
- Generic staircase **Moreno-Socías**
- $w(x_1) = 0, x_1 = w_2(x_1), \ldots, x_n = w_n(x_1)$

Describing solutions $\rightsquigarrow$ Change of orders

The "usual" good way to do



$\frac{\mathbb{K}[x_1,\dots,x_n]}{I}$ is a finite dimensional vector space



- Combinatorial structure of polynomial ideals
- Basis $\mathscr{B}$ of quotient ring $\frac{\mathbb{K}[x_1,\dots,x_n]}{I}$
- Generic staircase **Moreno-Socías**
- $w(x_1) = 0, x_1 = w_2(x_1), \dots, x_n = w_n(x_1)$

The generic staircase of grevlex Gröbner bases (Moreno-Socías)

For $m \in \mathscr{B}$, $mx_n \in \mathscr{B}$ or $mx_n \in \mathrm{LM}_{\succ_{grevlex}}(GB_{grevlex}) \rightarrow$ sparse matrix

Change of orders algorithms

Faugère/Lazard/Gianni/Mora $\rightsquigarrow$ FGLM algorithm

Complexity $O\left(D^3\right)$

relation reconstruction through linear algebra

Not implemented in `msolve`

Change of orders algorithms

Faugère/Lazard/Gianni/Mora $\rightsquigarrow$ FGLM algorithm

Complexity $O\left(D^3\right)$

relation reconstruction through linear algebra

Not implemented in `msolve`

Faugère/Mou $\rightsquigarrow$ connection to Wiedemann's algorithm (sparsity)

Computation of minimal polynomial

Berlekamp-Massey $\rightsquigarrow$ parametrizations

Implemented in `msolve`

$t = \sharp\{m \mid mx_n \in \mathscr{B}\}$

Complexity $O\left(tD^2\right)$

Change of orders algorithms

Faugère/Lazard/Gianni/Mora $\rightsquigarrow$ FGLM algorithm

Complexity $O(D^3)$

relation reconstruction through linear algebra

Not implemented in `msolve`

Faugère/Mou $\rightsquigarrow$ connection to Wiedemann's algorithm (sparsity)

Computation of minimal polynomial

Berlekamp-Massey $\rightsquigarrow$ parametrizations

Implemented in `msolve`

$t = \sharp\{m \mid mx_n \in \mathscr{B}\}$

Complexity $O(tD^2)$

Berthomieu/Neiger/S.

Change of paradigm:

~~sparse~~ ☛ structured

Complexity $O(t^{\omega-1}D)$

Change of order timings

`msolve` implementation (prime fields, characteristic $< 2^{31}$)

☛ dedicated encoding of multiplication matrices

☛ AVX2 implementation

Examples	`msolve` FGLM	`maple` FGLM	ratio	`msolve` tracer	ratio (FGLM / tracer)
Katsura-10	0.11	0.15	1.36	0.09	1.2
Katsura-11	0.49	0.74	1.51	0.45	1.1
Katsura-12	3.96	5.4	1.36	3.10	1.28
Katsura-13	30.6	35.7	1.16	19	1.61
Katsura-14	210	271	1.29	128	1.64
Eco-11	0.07	0.12	1.71	0.17	0.41
Eco-12	0.34	0.85	2.5	1.07	0.31
Eco-13	2.12	6.7	3.16	6.6	0.32
Eco-14	25.9	69.1	2.67	34.8	0.74
Eco-15	146.3	n.m.		155.73	0.94
Henrion-6	0.11	0.11	1	0.07	1.57
Henrion-7	20.46	27.1	1.32	6.5	3.15
Noon-7	1.95	3.13	1.6	0.93	3.37
Noon-8	72.3	76.2	1.05	17.5	4.13

Change of order timings

msolve implementation (prime fields, characteristic $< 2^{31}$)

☛ dedicated encoding of multiplication matrices

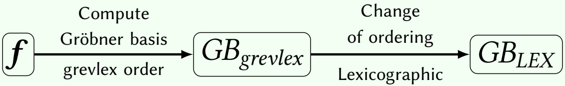
☛ AVX2 implementation

Examples	msolve FGLM	maple FGLM	ratio	msolve tracer	ratio (FGLM / tracer)
Katsura-10	0.11	0.15	1.36	0.09	1.2
Katsura-11	0.49	0.74	1.51	0.45	1.1
Katsura-12	3.96	5.4	1.36	3.10	1.28
Katsura-13	30.6	35.7	1.16	19	1.61
Katsura-14	210	271	1.29	128	1.64
Eco-11	0.07	0.12	1.71	0.17	0.41
Eco-12	0.34	0.85	2.5	1.07	0.31
Eco-13	2.12	6.7	3.16	6.6	0.32
Eco-14	25.9	69.1	2.67	34.8	0.74
Eco-15	146.3	n.m.		155.73	0.94
Henrion-6	0.11	0.11	1	0.07	1.57
Henrion-7	20.46	27.1	1.32	6.5	3.15
Noon-7	1.95	3.13	1.6	0.93	3.37
Noon-8	72.3	76.2	1.05	17.5	4.13

FGLM is **increasingly** dominant w.r.t. F4-tracer in msolve

Solving systems over the rational numbers

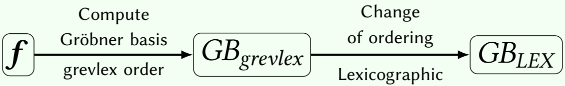
The “usual” good way to do



$$w(x_1) = 0, x_2 = w_2(x_1), \ldots, x_n = w_n(x_1) \quad \rightsquigarrow \quad w(x_1) = 0, x_2 = \frac{v_2(x_1)}{w'(x_1)}, \ldots, x_n = \frac{v_n(x_1)}{w'(x_1)}$$

Solving systems over the rational numbers

The “usual” good way to do



$$w(x_1) = 0, x_2 = w_2(x_1), \ldots, x_n = w_n(x_1) \rightsquigarrow w(x_1) = 0, x_2 = \frac{v_2(x_1)}{w'(x_1)}, \ldots, x_n = \frac{v_n(x_1)}{w'(x_1)}$$

- ☛ Multi-modular arithmetics
- ☛ Rational reconstruction
- ☛ Plenty of asymptotically optimal algorithms for univariate polynomials
- ☛ Dependency on the output bit size
- ☛ Probabilistic algorithm

Solving systems over the rational numbers

The "usual" good way to do

f —(Compute Gröbner basis, grevlex order)→ $GB_{grevlex}$ —(Change of ordering, Lexicographic)→ GB_{LEX}

$$w(x_1) = 0, x_2 = w_2(x_1), \ldots, x_n = w_n(x_1) \rightsquigarrow w(x_1) = 0, x_2 = \frac{v_2(x_1)}{w'(x_1)}, \ldots, x_n = \frac{v_n(x_1)}{w'(x_1)}$$

- ☛ Multi-modular arithmetics
- ☛ Rational reconstruction
- ☛ Plenty of asymptotically optimal algorithms for univariate polynomials
- ☛ Dependency on the output bit size
- ☛ Probabilistic algorithm

Lift $GB_{grevlex}$ or lift GB_{lex}?

To lift or not to lift ($GB_{grevlex}$)?

`./msolve -g 2 -f in.ms -o out.ms` versus `./msolve -f in.ms -o out.ms`

Examples	`msolve` Grevlex	nprimes	`msolve` Param	nprimes	ratio (time)	ratio (nprimes)
Katsura-10	3.26	21	21.75	141	0.15	0.15
Katsura-11	29.1	34	179.67	307	0.16	0.11
Katsura-12	260	56	2 025.82	643	0.13	0.09
Katsura-13	1 326	81	47 539.59	1336	0.03	0.06
Katsura-14	12 101	108	738 259.30	2941	0.02	0.04

To lift or not to lift ($GB_{grevlex}$)?

`./msolve -g 2 -f in.ms -o out.ms` versus `./msolve -f in.ms -o out.ms`

Examples	msolve Grevlex	nprimes	msolve Param	nprimes	ratio (time)	ratio (nprimes)
Katsura-10	3.26	21	21.75	141	0.15	0.15
Katsura-11	29.1	34	179.67	307	0.16	0.11
Katsura-12	260	56	2 025.82	643	0.13	0.09
Katsura-13	1 326	81	47 539.59	1336	0.03	0.06
Katsura-14	12 101	108	738 259.30	2941	0.02	0.04
Eco-11	31.03	131	47.22	174	0.66	0.75
Eco-12	128.91	188	494.52	317	0.26	0.59
Eco-13	3 544.29	531	4 147.28	650	0.85	0.82
Eco-14	44 732.69	1284	67 361.61	1347	0.66	0.95

To lift or not to lift ($GB_{grevlex}$)?

`./msolve -g 2 -f in.ms -o out.ms` versus `./msolve -f in.ms -o out.ms`

Examples	msolve Grevlex	nprimes	msolve Param	nprimes	ratio (time)	ratio (nprimes)
Katsura-10	3.26	21	21.75	141	0.15	0.15
Katsura-11	29.1	34	179.67	307	0.16	0.11
Katsura-12	260	56	2 025.82	643	0.13	0.09
Katsura-13	1 326	81	47 539.59	1336	0.03	0.06
Katsura-14	12 101	108	738 259.30	2941	0.02	0.04
Eco-11	31.03	131	47.22	174	0.66	0.75
Eco-12	128.91	188	494.52	317	0.26	0.59
Eco-13	3 544.29	531	4 147.28	650	0.85	0.82
Eco-14	44 732.69	1284	67 361.61	1347	0.66	0.95
Random-8	103.87	5 120	30.84	1 172	3.35	4.36
Random-9	1 616.59	12 800	318.23	2 661	5.08	4.81
Random-10	24 612.11	31 744	3 520.40	5 915	6.99	5.37
Random-11	568 577.42	73 728	46 085.00	13 000	12.34	5.67

To lift or not to lift ($GB_{grevlex}$)?

`./msolve -g 2 -f in.ms -o out.ms` versus `./msolve -f in.ms -o out.ms`

Examples	msolve Grevlex	nprimes	msolve Param	nprimes	ratio (time)	ratio (nprimes)
Katsura-10	3.26	21	21.75	141	0.15	0.15
Katsura-11	29.1	34	179.67	307	0.16	0.11
Katsura-12	260	56	2 025.82	643	0.13	0.09
Katsura-13	1 326	81	47 539.59	1336	0.03	0.06
Katsura-14	12 101	108	738 259.30	2941	0.02	0.04
Eco-11	31.03	131	47.22	174	0.66	0.75
Eco-12	128.91	188	494.52	317	0.26	0.59
Eco-13	3 544.29	531	4 147.28	650	0.85	0.82
Eco-14	44 732.69	1284	67 361.61	1347	0.66	0.95
Random-8	103.87	5 120	30.84	1 172	3.35	4.36
Random-9	1 616.59	12 800	318.23	2 661	5.08	4.81
Random-10	24 612.11	31 744	3 520.40	5 915	6.99	5.37
Random-11	568 577.42	73 728	46 085.00	13 000	12.34	5.67

Not so clear that there is in general interest to lift $GB_{grevlex}$

☛ msolve lifts GB_{lex}

Univariate real root isolation

Based on FLINT's univariate multiplication in `fmpz_poly`

Examples	♯ sols	msolve	maple		SLV		tdescartes	
		time	time	ratio	time	ratio	time	ratio
Katsura-10	120	3.1	4.8	1.5	3.8	1.2	20	6.5
Katsura-11	216	27	60	2.2	50.5	1.9	156	5.8
Katsura-12	326	207	656	3.2	555	2.7	2, 206	10.6
Katsura-13	582	2 220	16 852	7.6	13 651	6.1	22 945	10.3
Katsura-14	900	20 149	250 094	12.4	252 183	12.5	384 566	19.1
Katsura-15	1, 606	197 048	3 588 835	18.2	3 540 480	18.0	5 178 180	26.3
Katsura-16	2, 543	1 849 986	—	—	—	—	—	—
Katsura-17	4, 428	16 128 000	—	—	—	—	—	—

Real root isolation timings given in seconds

Warning: uses maple-v16

Timings for solving

Examples	DEG	msolve(trace)	msolve(prob)	speed-up	maple	speed-up	magma	speed-up
Katsura-9	256	4.89	7.49	1.53	104	21.27	2522	515
Katsura-10	512	43.7	70.5	1.61	1 278	29.24	82 540	1 888
Katsura-11	1024	424	814	1.92	7 812	18.4	-	
Katsura-12	2048	6 262	11 215	1.79	120 804	19.29	-	
Katsura-13	4096	89 390	148 372	1.66	-		-	
Katsura-14	8192	1 308 602	2 000 170	1.53	-		-	
Eco-10	256	12.5	21.2	1.69	26.3	2.1	6520	521.6
Eco-11	512	90.3	161	1.78	312	3.45	214 770	2378
Eco-12	1024	877	1 619	1.84	4 287	4.88	-	
Eco-13	2048	12 137	19 553	1.61	66 115	5.44	-	
Eco-14	4096	167 798	254 389	1.51	-		-	
Henrion-5	100	0.71	0.83	1.17	2.7	3.8	93	130.98
Henrion-6	720	138	157	1.13	1 470	10.65	-	
Henrion-7	5040	117 803	127 456	1.08	-		-	
CP(3,5,2)	288	18.1	19.2	1.06	249	13.75	-	
CP(3,6,2)	720	390	450	1.15	23 440	60	-	
CP(3,7,2)	1728	9 643	11 511	1.19	-		-	
CP(3,8,2)	4032	269 766	323 838	1.2	-		-	

Timings for solving

Examples	DEG	msolve(trace)	msolve(prob)	speed-up	maple	speed-up	magma	speed-up
Katsura-9	256	4.89	7.49	1.53	104	21.27	2522	515
Katsura-10	512	43.7	70.5	1.61	1 278	29.24	82 540	1 888
Katsura-11	1024	424	814	1.92	7 812	18.4	-	
Katsura-12	2048	6 262	11 215	1.79	120 804	19.29	-	
Katsura-13	4096	89 390	148 372	1.66	-		-	
Katsura-14	8192	1 308 602	2 000 170	1.53	-		-	
Eco-10	256	12.5	21.2	1.69	26.3	2.1	6520	521.6
Eco-11	512	90.3	161	1.78	312	3.45	214 770	2378
Eco-12	1024	877	1 619	1.84	4 287	4.88	-	
Eco-13	2048	12 137	19 553	1.61	66 115	5.44	-	
Eco-14	4096	167 798	254 389	1.51	-		-	
Henrion-5	100	0.71	0.83	1.17	2.7	3.8	93	130.98
Henrion-6	720	138	157	1.13	1 470	10.65	-	
Henrion-7	5040	117 803	127 456	1.08	-		-	
CP(3,5,2)	288	18.1	19.2	1.06	249	13.75	-	
CP(3,6,2)	720	390	450	1.15	23 440	60	-	
CP(3,7,2)	1728	9 643	11 511	1.19	-		-	
CP(3,8,2)	4032	269 766	323 838	1.2	-		-	
Noon-7	2173	4039	5 045	1.25	432	0.1	-	
Noon-8	6545	598 647	640 177	1.07	5997	0.01	-	

Timings for solving

Examples	DEG	msolve(trace)	msolve(prob)	speed-up	maple	speed-up	magma	speed-up
Katsura-9	256	4.89	7.49	1.53	104	21.27	2522	515
Katsura-10	512	43.7	70.5	1.61	1 278	29.24	82 540	1 888
Katsura-11	1024	424	814	1.92	7 812	18.4	-	
Katsura-12	2048	6 262	11 215	1.79	120 804	19.29	-	
Katsura-13	4096	89 390	148 372	1.66	-		-	
Katsura-14	8192	1 308 602	2 000 170	1.53	-		-	
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	6520	521.6
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]70	2378
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-	
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-	
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-	
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]93	130.98
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-	
Henrion-7	5040	117 803	127 456	1.08	-		-	
CP(3,5,2)	288	18.1	19.2	1.06	249	13.75	-	
CP(3,6,2)	720	390	450	1.15	23 440	60	-	
CP(3,7,2)	1728	9 643	11 511	1.19	-		-	
CP(3,8,2)	4032	269 766	323 838	1.2	-		-	
Noon-7	2173	4039	5 045	1.25	432	0.1	-	
Noon-8	6545	598 647	640 177	1.07	5997	0.01	-	

On Noon examples we suffer from the bit size of our output parametrizations (which could be split in many small components)

Using Gröbner bases in geometry

Take C_1, C_2, C_3, C_4, C_5 in $\mathbb{Q}[x_1, x_2]$ of degree 2.
Compute $U \in \mathbb{Q}[x_1, x_2]$ such that
$V(U)$ is tangent to $V(C_i)$ for $1 \leq i \leq 5$.

Using Gröbner bases in geometry

Take C_1, C_2, C_3, C_4, C_5 in $\mathbb{Q}[x_1, x_2]$ of degree 2.
Compute $U \in \mathbb{Q}[x_1, x_2]$ such that
$V(U)$ is tangent to $V(C_i)$ for $1 \leq i \leq 5$.

Breiding, Sturmfels, Timme'20 Solving means computing a Gröbner basis G. Indeed, crucial invariants, such as the dimension and degree of the solution variety, [...] The number of real solutions is found by applying techniques [...]. Yet Gröbner bases can take a very long time to compute. We found them impractical for Steiner's problem.

☛ Various modelings proposed, difficulty is to "force" U to be generic.
One suits better with numerical homotopy continuation

Using Gröbner bases in geometry

Take C_1, C_2, C_3, C_4, C_5 in $\mathbb{Q}[x_1, x_2]$ of degree 2.
Compute $U \in \mathbb{Q}[x_1, x_2]$ such that
$V(U)$ is tangent to $V(C_i)$ for $1 \leq i \leq 5$.

Breiding, Sturmfels, Timme'20 Solving means computing a Gröbner basis G. Indeed, crucial invariants, such as the dimension and degree of the solution variety, [...] The number of real solutions is found by applying techniques [...]. Yet Gröbner bases can take a very long time to compute. We found them impractical for Steiner's problem.

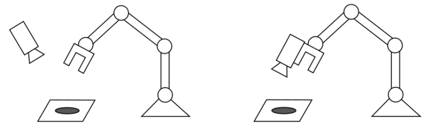
☛ Various modelings proposed, difficulty is to "force" U to be generic.
One suits better with numerical homotopy continuation

"New" alternative modeling which suits "well" to Gröbner bases

👍 `msolve` can solve one instance within $\simeq$ 2.5 hours (!)

👎 using 36 threads (memory consumption is ok but not tiny)...

Vision-based control schemes in robotics



☛ eye-in-hand with configuration camera

☛ dynamic control observation

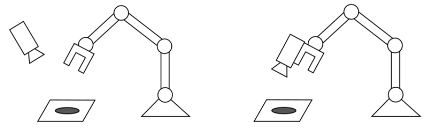
observation ⇝ desired position

Lyapunov theory ⇝ **critical points of a polynomial map**

local extrema ⇝ stability analysis

Briot/Chaumette/Colotti/Garcia-Fontan/Goldsztein/S.

Vision-based control schemes in robotics



- ☛ eye-in-hand with configuration camera
- ☛ dynamic control observation

observation $\rightsquigarrow$ desired position

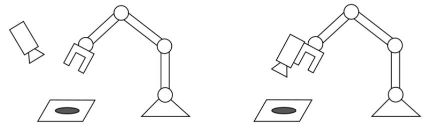
Lyapunov theory $\rightsquigarrow$ **critical points of a polynomial map**

local extrema $\rightsquigarrow$ stability analysis

System	msolve(×12)	HC.jl (×1)	Out. (algebraic)	Out. (numeric)
sys1	15 days	1630 secs	402/50	403/50
sys2	24 days	1495 secs	1016/44	1016/44
sys3	27 days	1950 secs	1064/48	871/32
sys4	41 days	2280 secs	3656/84	3537/95

Briot/Chaumette/Colotti/Garcia-Fontan/Goldsztein/S.

Vision-based control schemes in robotics



- eye-in-hand with configuration camera
- dynamic control observation

observation ⇝ desired position

Lyapunov theory ⇝ **critical points of a polynomial map**

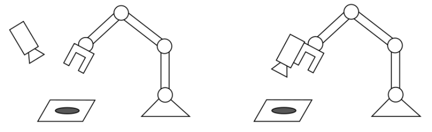
local extrema ⇝ stability analysis

System	msolve(×12)	HC.jl (×1)	Out. (algebraic)	Out. (numeric)
sys1	15 days	1630 secs	402/50	403/50
sys2	24 days	1495 secs	1016/44	1016/44
sys3	27 days	1950 secs	1064/48	871/32
sys4	41 days	2280 secs	3656/84	3537/95

System	msolve(×12)	HC.jl (×1)	Out. (algebraic)	Out. (numeric)
sys1	478 secs	14499 secs	402/50	402/50
sys2	21.2 h	15480 secs	1016/44	1016/44
sys3	18.4h	20099 secs	1064/48	871/32
sys4	41 days	2280 secs	3656/84	3537/95

Briot/Chaumette/Colotti/Garcia-Fontan/Goldsztein/S.

Vision-based control schemes in robotics



- ☛ eye-in-hand with configuration camera
- ☛ dynamic control observation

observation ⇝ desired position

Lyapunov theory ⇝ **critical points of a polynomial map**

local extrema ⇝ stability analysis

System	msolve(×12)	HC.jl (×1)	Out. (algebraic)	Out. (numeric)
sys1	15 days	1630 secs	402/50	403/50
sys2	24 days	1495 secs	1016/44	1016/44
sys3	27 days	1950 secs	1064/48	871/32
sys4	41 days	2280 secs	3656/84	3537/95

System	msolve(×12)	HC.jl (×1)	Out. (algebraic)	Out. (numeric)
sys1	478 secs	14499 secs	402/50	402/50
sys2	21.2 h	15480 secs	1016/44	1016/44
sys3	18.4h	20099 secs	1064/48	871/32
sys4	41 days	2280 secs	3656/84	3537/95

System	msolve(×12)	HC.jl	msolve(×12)
sys1	15 days	1630 secs	172 secs
sys2	24 days	1495 secs	10243 secs
sys3	27 days	1950 secs	8035 secs
sys4	41 days	-	26h

- Symetries arise naturally in the formulation.
- Using GBs one can rewrite the polynomial system w.r.t. invariants.
- Last column reports on timings.

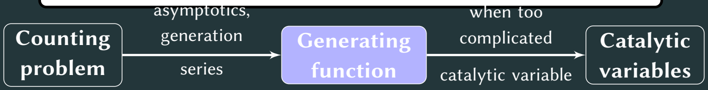
Briot/Chaumette/Colotti/Garcia-Fontan/Goldsztein/S.

Using Gröbner bases in combinatorics

The Flajolet-Sedgewick machinery for counting

Using Gröbner bases in combinatorics

The Flajolet-Sedgewick machinery for counting



Using Gröbner bases in combinatorics

The Flajolet-Sedgewick machinery for counting

Using Gröbner bases in combinatorics

The Flajolet-Sedgewick machinery for counting

Counting problem —(asymptotics, generation / series)→ **Generating function** —(when too complicated / catalytic variable)→ **Catalytic variables**

Algebraicity result **Bousquet-Mélou/Jéhanne'06, Popescu'86**

Let $f \in \mathbb{Q}[u]$ and $Q \in \mathbb{Q}[x, y, t, u]$.
Let $\mathscr{F} \in \mathbb{Q}[u][[t]]$ be the unique solution to
$$\boxed{\mathscr{F} = f(u) + tQ(\mathscr{F}, \Delta(\mathscr{F}), \ldots, \Delta^{(k)}(\mathscr{F}), t, u)}$$
where $\Delta = \frac{\mathscr{F}(t,u) - \mathscr{F}(t,1)}{u-1}$. Then, $\mathscr{F}$ is algebraic over $\mathbb{Q}(t, u)$
$\exists R \in \mathbb{Q}[t, z] - \{0\}, R(t, \mathscr{F}(t, 1)) \equiv 0$.

Using Gröbner bases in combinatorics

The Flajolet-Sedgewick machinery for counting

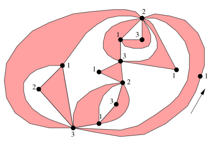
Counting problem —(asymptotics, generation / series)→ **Generating function** —(when too complicated / catalytic variable)→ **Catalytic variables**

Algebraicity result **Bousquet-Mélou/Jéhanne'06, Popescu'86**

Let $f \in \mathbb{Q}[u]$ and $Q \in \mathbb{Q}[x, y, t, u]$.
Let $\mathscr{F} \in \mathbb{Q}[u][[t]]$ be the unique solution to

$$\boxed{\mathscr{F} = f(u) + tQ(\mathscr{F}, \Delta(\mathscr{F}), \ldots, \Delta^{(k)}(\mathscr{F}), t, u)}$$

where $\Delta = \frac{\mathscr{F}(t,u) - \mathscr{F}(t,1)}{u-1}$. Then, $\mathscr{F}$ is algebraic over $\mathbb{Q}(t, u)$
$\exists R \in \mathbb{Q}[t, z] - \{0\}, R(t, \mathscr{F}(t, 1)) \equiv 0.$



DDE → Polynomial systems

Elimination

msolve used

Using Gröbner bases in combinatorics

The Flajolet-Sedgewick machinery for counting

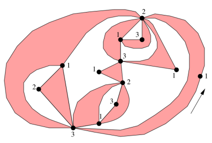
Counting problem —(asymptotics, generation / series)→ **Generating function** —(when too complicated / catalytic variable)→ **Catalytic variables**

Algebraicity result **Bousquet-Mélou/Jéhanne'06, Popescu'86**

Let $f \in \mathbb{Q}[u]$ and $Q \in \mathbb{Q}[x, y, t, u]$.
Let $\mathscr{F} \in \mathbb{Q}[u][[t]]$ be the unique solution to

$$\boxed{\mathscr{F} = f(u) + tQ(\mathscr{F}, \Delta(\mathscr{F}), \ldots, \Delta^{(k)}(\mathscr{F}), t, u)}$$

where $\Delta = \frac{\mathscr{F}(t,u) - \mathscr{F}(t,1)}{u-1}$. Then, $\mathscr{F}$ is algebraic over $\mathbb{Q}(t, u)$
$\exists R \in \mathbb{Q}[t, z] - \{0\}, R(t, \mathscr{F}(t, 1)) \equiv 0.$



DDE → Polynomial systems

Elimination

`msolve` used

Bostan/Chyzak/Notarantonio/S.

What's next: syzygies and ideal-theoretic operations

A module approach $fg = gf \rightsquigarrow \mathrm{lt}(f)g = gf - \mathrm{tail}(f)g$

Compact representations of module of syzygies (F5) **Eder/Faugère**

What's next: syzygies and ideal-theoretic operations

A module approach $fg = gf \rightsquigarrow \mathrm{lt}(f)g = gf - \mathrm{tail}(f)g$

Compact representations of module of syzygies (F5) **Eder/Faugère**

- Complexity issues in F5 algorithms
- Specializations of F5 in some structured setting
- Determinantal setting $\rightsquigarrow$ Crypto applications

Gopalakrishnan/Neiger/S.

What's next: syzygies and ideal-theoretic operations

A module approach $fg = gf \rightsquigarrow \mathrm{lt}(f)g = gf - \mathrm{tail}(f)g$

Compact representations of module of syzygies (F5) **Eder/Faugère**

- Complexity issues in F5 algorithms
- Specializations of F5 in some structured setting
- Determinantal setting $\rightsquigarrow$ Crypto applications

Gopalakrishnan/Neiger/S.

Ideal theoretic operations Nothing new since Bayer's PhD (!)

☛ F4 variant to compute saturation of ideals **Berthomieu/Eder/S.**

What's next: syzygies and ideal-theoretic operations

A module approach $fg = gf \rightsquigarrow \mathrm{lt}(f)g = gf - \mathrm{tail}(f)g$

Compact representations of module of syzygies (F5) **Eder/Faugère**

- Complexity issues in F5 algorithms
- Specializations of F5 in some structured setting
- Determinantal setting $\rightsquigarrow$ Crypto applications

Gopalakrishnan/Neiger/S.

Ideal theoretic operations Nothing new since Bayer's PhD (!)

☛ F4 variant to compute saturation of ideals **Berthomieu/Eder/S.**

☛ F5 variant for saturations + equidimensional decomposition

- Some reductions to 0 are unavoidable
- Exploit them $\rightsquigarrow$ decomposition of ideals

Eder/Lairez/Mohr/S.

The F4SAT algorithm

Berthomieu/Eder/S.

D_I: maximum degree reached to compute GB for I

$D_{\mathcal{J}}$: maximum degree reached to compute GB for $\mathcal{J}$

D_{rab}: maximum degree reached to compute GB for "rabinovitch" ideal

speedup1: Rabinowitsch / F4SAT (learn), speedup2: Rabinowitsch F4SAT (tracer)

speedup3: F4SAT / Maple (tracer)

system	D_I	$D_{\mathcal{J}}$	D_{rab}	learn	tracer	speedup1	speedup2	speedup3
d3-n6-p2	13	10	15	1.31	0.31	1.83	1.33	3.61
d3-n6-p3	16	13	18	43.7	1.84	3.25	9	19.24
d3-n6-p4	19	16	21	533	19.7	1.65	6.4	11.32
d4-n6-p3	24	20	27	31 101	596	1.4	10.6	14.8
d3-n7-p4	20	17	22	22 296	469	2.12	11.4	21.32
d3-n7-p5	23	20	25	126 006	2 881	1.62	7.96	11.67
d2-n8-p5	12	10	13	11.7	1.79	8.54	4.42	11.4
d3-n7-p3	17	14	19	1 263	32.4	2.89	12.53	30.37
d2-n9-p6	15	17	20	40 352	2 155	1.01	3.25	3.23
d2-n10-p5	13	14	15	66 845	2 141	0.9	10.8	32.1
steiner	19	19	24	115	67.2	5.34	2.28	3.57

Equidimensional decomposition

name	nb. comp.	equidim	Maple	Oscar	Magma	Magma (prime dec.)	Bertini
8-3-config-Li	23	1.6	16 ×10	> 1h	> 1h	65 ×40	4 ×2.5
cyclic8	6	381	> 5h	> 5h	> 5h	> 5h	⚠ 126 ×0.3
dgp6	3	✋ 0.2	53	2.2	> 1h	1.2	75
Gonnet	3	✋ 0.2	2.1	2.8	> 1h	1.4	74
P4L1	6	0.3	2.4	1.8	0.7	1.5	⚠ 21
P4L3	8	0.3	3.3	10	¡0.1	1.5	11
KdV		> 4h	353	> 4h	> 4h	7109 ×20	> 4h
Leykin-1	13	✋ 2.6 ×1.9	4 ×3.2	641 ×468	> 1h	1.4	✖
C1	4	129	> 1h	> 1h	> 1h	> 1h	✖
C2	4	0.3	100	152	> 1h	2.0	✖
C3	13	10	55	7	0.3	1.5	✖
MontesS16	6	1.9 ×1.4	2.7 ×1.9	2.0 ×1.4	1.4	1.5 ×1.1	7 ×5
Ps(10)	2	1.7	> 1h	30 ×17	> 1h	6 ×3.3	9 ×5
Ps(12)	2	51	> 1h	> 1h	> 1h	2060 ×40	⚠ 38 ×0.7
Ps(6)	2	¡0.1	0.2	1.7	0.5	0.3	2.0
Ps(8)	2	¡0.1	4	1.7	1.2	0.8	6
Sing(10)	2	0.4	> 1h	> 1h	> 1h	> 1h	⚠ 495
Sing(4)	2	¡0.1	76	2.2	8	5	5
Sing(5)	2	¡0.1	> 1h	4	7	1636	⚠ 1.0
Sing(6)	2	¡0.1	> 1h	51	> 1h	> 1h	⚠ 8
Sing(7)	2	¡0.1	1704	399	> 1h	> 1h	54
Sing(8)	2	0.1	> 1h	995	> 1h	> 1h	139
Sing(9)	2	0.2	> 1h	> 1h	> 1h	> 1h	⚠ 271
sos(4,2)	2	¡0.1	16	2.2	1.3	1.2	1.0
sos(4,3)	2	¡0.1	694	2.6	3.4	6	3.0
sos(5,2)	2	¡0.1	> 1h	1.8	3.7	1.2	3.0
sos(5,3)	2	¡0.1	> 1h	> 1h	> 1h	149	15
sos(5,4)	2	0.5	> 1h	> 1h	> 1h	> 1h	21
sos(6,2)	2	¡0.1	> 1h	2.0	5	1.6	5
sos(6,3)	2	0.1	> 1h	> 1h	> 1h	> 1h	34
sos(6,4)	2	5	> 1h	> 1h	> 1h	> 1h	69 ×14
sos(6,5)	2	14	> 1h	> 1h	> 1h	> 1h	⚠ 40 ×2.9
steiner	2	870	> 12h	> 12h	> 12h	> 12h	✖

New change of order algorithm

Paradigm shift ~~sparse~~ → structured **Berthomieu/Neiger/S.**

$$\left[\begin{array}{cccc|cc|cc}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & -22 & -3 & -3 & -26 & -23 & 0 & -15 \\
\hline
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
-17 & 0 & -3 & 0 & -15 & -28 & -19 & -5 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
-3 & -9 & -19 & -18 & 0 & -27 & -2 & -24
\end{array}\right]$$

New change of order algorithm

Paradigm shift ~~sparse~~ $\to$ structured **Berthomieu/Neiger/S.**

$$\left[\begin{array}{cccc|cc|cc}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\
0 & -22 & -3 & -3 & -26 & -23 & 0 & -15\\
\hline
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0\\
-17 & 0 & -3 & 0 & -15 & -28 & -19 & -5\\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1\\
-3 & -9 & -19 & -18 & 0 & -27 & -2 & -24
\end{array}\right] \rightsquigarrow \begin{bmatrix} x_3^4+3x_3^3+3x_3^2+22x_3 & 23x_3+26 & 15x_3\\ 3x_3^2+17 & x_3^2+28x_3+15 & 5x_3+19\\ 18x_3^3+19x_3^2+9x_3+3 & 27x_3 & x_3^2+24x_3+2\end{bmatrix} \in \mathbb{K}[x_3]^{t\times t}$$

New change of order algorithm

Paradigm shift ~~sparse~~ → structured **Berthomieu/Neiger/S.**

$$\left[\begin{array}{cccc|cc|cc} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -22 & -3 & -3 & -26 & -23 & 0 & -15 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -17 & 0 & -3 & 0 & -15 & -28 & -19 & -5 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -3 & -9 & -19 & -18 & 0 & -27 & -2 & -24 \end{array}\right] \rightsquigarrow \begin{bmatrix} x_3^4 + 3x_3^3 + 3x_3^2 + 22x_3 & 23x_3 + 26 & 15x_3 \\ 3x_3^2 + 17 & x_3^2 + 28x_3 + 15 & 5x_3 + 19 \\ 18x_3^3 + 19x_3^2 + 9x_3 + 3 & 27x_3 & x_3^2 + 24x_3 + 2 \end{bmatrix} \in \mathbb{K}[x_3]^{t\times t}$$

New change of order algorithm

Paradigm shift ~~sparse~~ → structured

Berthomieu/Neiger/S.

$$\left[\begin{array}{cccc|cc|cc} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -22 & -3 & -3 & -26 & -23 & 0 & -15 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -17 & 0 & -3 & 0 & -15 & -28 & -19 & -5 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -3 & -9 & -19 & -18 & 0 & -27 & -2 & -24 \end{array}\right] \rightsquigarrow \begin{bmatrix} x_3^4 + 3x_3^3 + 3x_3^2 + 22x_3 & 23x_3 + 26 & 15x_3 \\ 3x_3^2 + 17 & x_3^2 + 28x_3 + 15 & 5x_3 + 19 \\ 18x_3^3 + 19x_3^2 + 9x_3 + 3 & 27x_3 & x_3^2 + 24x_3 + 2 \end{bmatrix} \in \mathbb{K}[x_3]^{t \times t}$$

basis of $\mathbb{K}[x_3]$-module of $I \cap (\mathbb{K}[x_3] + x_2\mathbb{K}[x_3] + x_1\mathbb{K}[x_3])$

Hermite normal form $\rightsquigarrow$ lex Gröbner basis

Complexity: $O\left(t^{\omega-1}D\right)$

			Step 1: $\mathcal{G}_{\mathrm{drl}} \approx P$		Step 2: $\mathcal{G}_{\mathrm{lex}} \approx H$		
			msolve		msolve	NTL	PML
n, d	D	t	F_4	F_4-tr	Wied.	bl-Wied.	HNF
11, 2	2048	462	11.6	1.1	1.2	1.7	0.8
12, 2	4096	924	115.9	8.3	6.5	14.5	5.3
13, 2	8192	1716	970	62	103.6	110	34.8
14, 2	16384	3432	7921	460	1011	880	240
15, 2	32768	6435	61381	3193	7844	6691	1665
16, 2	65536	12870	482515	24523	58744	52709	11359
8, 3	6561	1107	122.6	12.8	23.6	44.7	15.1
9, 3	19683	3139	3552.7	361	1302	1163	314
10, 3	59049	8953	95052	8664	34844	29974	6709
6, 4	4096	580	9.9	2.2	4	8.8	3.5
7, 4	16384	2128	876	128	575	545	157
8, 4	65536	8092	57237	6977	36454	33452	7231

msolve's perspectives

1. ~~Lift Gröbner bases over the rationals~~ (done)
2. ~~Improve parallelism in hashing~~ (done)
3. Test more and stabilize new algorithms for ideal saturation (started, on-going, almost done)
4. Implement ideal decompositions (zero dimensional and positive dimensional case)
5. Better continuous integration (started, on-going, almost done)
6. Mix F5 and F4 $\rightsquigarrow$ F6 algorithm (started, on-going)
7. Implement new change of orderings algorithms (started, on-going)
8. Implement Hilbert series computations
9. Implement weighted orderings
10. Develop the `AlgebraicSolving.jl` package (basic solving)
11. Develop the `AlgebraicSolving.jl` package for semi-algebraic geometry
12. Use AVX512 + Apple M2 chip instructions
13. Use MPI to have `msolve` running on clusters
14. Write an interface to the tracer (in `AlgebraicSolving.jl`)
15. Write a C interface with a documented API
16. Integrate Hensel lifting techniques $\rightsquigarrow$ quadratic convergence when lifting rationals
17. Modular arithmetics with floating point arithmetics
18. Linear algebra improvements: matrices are not only sparse
    but structured $\rightsquigarrow$ matrix multiplication $\leftrightarrow$ Gaussian elimination
19. Use code generation techniques
20. Have a dedicated implementation for the boolean field and extension fields
21. Investigate the use of GPUs
22. Solve challenging applications
23. Continue to disseminate `msolve` in computer algebra systems
    (Oscar ✓, SageMath ✓, Macaulay2 ✗, Symbolics.jl ✗)
24. **Hunt bugs, write documentations**, etc, etc, etc, etc...

msolve's perspectives

1. ~~Lift Gröbner bases over the rationals~~ (done)
2. ~~Improve parallelism in hashing~~ (done)
3. Test more and stabilize new algorithms for ideal saturation (started, on-going, almost done)
4. Impleme
5. Better c
6. Mix F5 a
7. Impleme
8. Impleme
9. Impleme
10. Develop
11. Develop
12. Use AVX
13. Use MP
14. Write ar

Acknowledgments. Marc Mezzarobba, Gleb Pogudin, Dima Pasechnik, Bill Alombert, Martin Helmer, Anton Leykin, the OSCAR team, Fredrik Johansson, Bill Hart, colleagues from robotics (Sébastien Briot, Jorge Garcia Fontan, Alexandre Goldzstein amongst others), Hadrien Notarantonio, Rémi Prébet, Clément Pernet, Pascal Giorgi and many others

Special thanks to Rafael Mohr and Jérémy Berthomieu

15. Write a C interface with a documented API
16. Integrate Hensel lifting techniques $\rightsquigarrow$ quadratic convergence when lifting rationals
17. Modular arithmetics with floating point arithmetics
18. Linear algebra improvements: matrices are not only sparse
    but structured $\rightsquigarrow$ matrix multiplication $\leftrightarrow$ Gaussian elimination
19. Use code generation techniques
20. Have a dedicated implementation for the boolean field and extension fields
21. Investigate the use of GPUs
22. Solve challenging applications
23. Continue to disseminate msolve in computer algebra systems
    (Oscar ✓, SageMath ✓, Macaulay2 ✗, Symbolics.jl ✗)
24. **Hunt bugs, write documentations**, etc, etc, etc, etc...

msolve's perspectives

1. ~~Lift Gröbner bases over the rationals~~ (done)
2. ~~Improve parallelism in hashing~~ (done)
3. Test more and stabilize new algorithms for ideal saturation (started, on-going, almost done)
4. Impleme
5. Better c
6. Mix F5 a
7. Impleme
8. Impleme
9. Impleme
10. Develop
11. Develop
12. Use AVX
13. Use MP
14. Write ar
15. Write a C interface with a documented API
16. Integrat
17. Modula
18. Linear a
19. Use cod
20. Have a
21. Investig
22. Solve ch
23. Continue to disseminate msolve in computer algebra systems (Oscar ✓, SageMath ✓, Macaulay2 ✗, Symbolics.jl ✗)
24. **Hunt bugs, write documentations**, etc, etc, etc, etc...

Acknowledgments. Marc Mezzarobba, Gleb Pogudin, Dima Pasechnik, Bill Alombert, Martin Helmer, Anton Leykin, the OSCAR team, Fredrik Johansson, Bill Hart, colleagues from robotics (Sébastien Briot, Jorge Garcia Fontan, Alexandre Goldzstein amongst others), Hadrien Notarantonio, Rémi Prébet, Clément Pernet, Pascal Giorgi and many others

Special thanks to Rafael Mohr and Jérémy Berthomieu

Requests.

☛ Star us on `github.com`

☛ Register to mailing list

☛ Continue coding

☛ Join forces (resource sharing)

Recent trends in computer algebra

https://rtca2023.github.io/

- Fundamental Algorithms and Algorithmic Complexity (Sep. 25-29)
- Geometry of Polynomial System Solving, Optimization and Topology (Oct. 16-20)
- Computer Algebra for Functional Equations in Combinatorics and Physics (Dec. 4-8)