Deterministic computation of the characteristic polynomial in the time of matrix multiplication

Vincent Neiger .... U. Limoges, France

Clément Pernet .... Univ Grenoble Alpes, France

Inria LFANT seminar

Bordeaux, France (online), March 16, 2021

Outline

- Context, problem, state of the art
- Overview of the approach and complexity
- Obstacles and related spin-off results

Outline

- Context, problem, state of the art
- Overview of the approach and complexity
- Obstacles and related spin-off results

Context

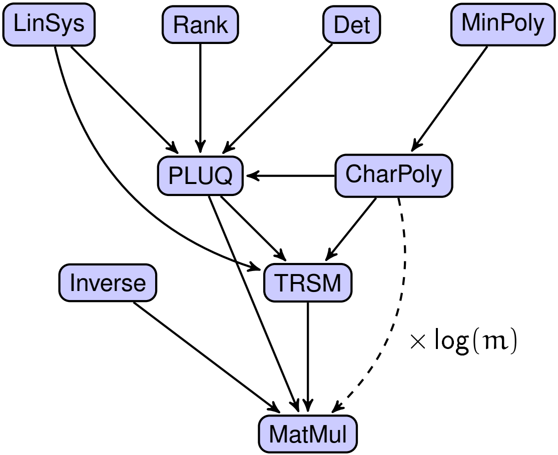
- field $\mathbb{K}$, algebraic complexity (counting operations in $\mathbb{K}$)
- ω: exponent of MatMul over $\mathbb{K}$: $m \times m$ by $m \times m$ in $O(m^{\omega})$

Reductions of most problems to matrix multiplication

Context

- field $\mathbb{K}$, algebraic complexity (counting operations in $\mathbb{K}$)
- ω: exponent of MatMul over $\mathbb{K}$: $m \times m$ by $m \times m$ in $O(m^\omega)$

Reductions of most problems to matrix multiplication

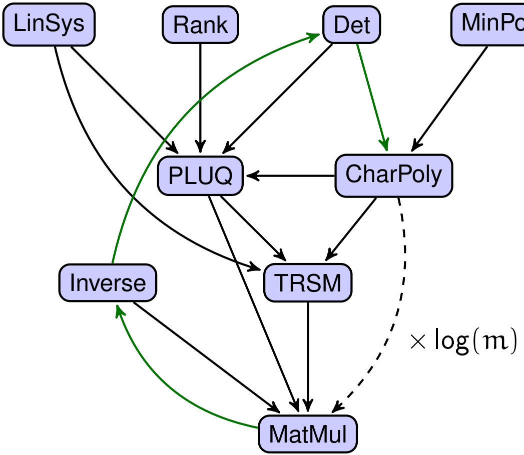


$$\left.\begin{array}{c}\mathsf{LinSys}\\ \mathsf{Det}\\ \mathsf{Rank}\\ \mathsf{PLUQ}\\ \mathsf{TRSM}\\ \mathsf{Inverse}\end{array}\right\} = O(\mathsf{MatMul})$$

Context

- field $\mathbb{K}$, algebraic complexity (counting operations in $\mathbb{K}$)
- ω: exponent of MatMul over $\mathbb{K}$: $m \times m$ by $m \times m$ in $O(m^\omega)$

Reductions of most problems to matrix multiplication



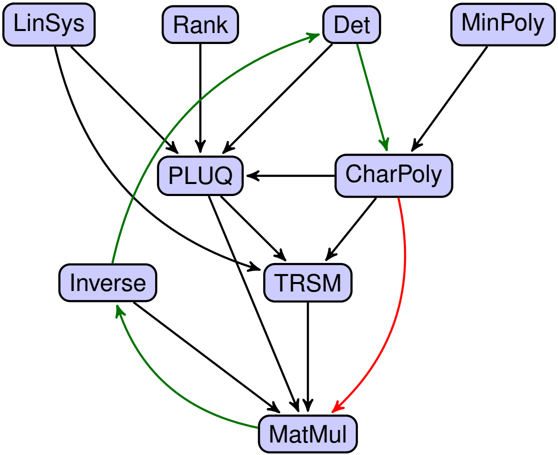
$$\left.\begin{matrix}\mathsf{LinSys}\\ \mathsf{Det}\\ \mathsf{Rank}\\ \mathsf{PLUQ}\\ \mathsf{TRSM}\\ \mathsf{Inverse}\end{matrix}\right\} = O(\mathsf{MatMul})$$

$$\mathsf{MatMul} = O\left(\begin{matrix}\mathsf{Det},\\ \mathsf{PLUQ},\\ \mathsf{CharPoly},\\ \mathsf{Inverse}\end{matrix}\right)$$

Context

- field $\mathbb{K}$, algebraic complexity (counting operations in $\mathbb{K}$)
- ω: exponent of MatMul over $\mathbb{K}$: $m \times m$ by $m \times m$ in $O(m^\omega)$

Reductions of most problems to matrix multiplication



$$\left.\begin{array}{c}\text{LinSys}\\ \text{Det}\\ \text{Rank}\\ \text{PLUQ}\\ \text{TRSM}\\ \text{Inverse}\end{array}\right\} = O(\text{MatMul})$$

$$\text{MatMul} = O\left(\begin{array}{c}\text{Det},\\ \text{PLUQ},\\ \text{CharPoly},\\ \text{Inverse}\end{array}\right)$$

$$\text{CharPoly} = O(\text{MatMul}) \ ?$$

Problem and result

Characteristic polynomial...

given $\mathbf{M} \in \mathbb{K}^{m \times m}$, compute $\det(x\mathbf{I}_m - \mathbf{M}) \in \mathbb{K}[x]$

- deterministic, general: $O(m^\omega \log(m))$ [Keller-Gehrig 1985]
- deterministic, generic input: $O(m^\omega)$ [Giorgi-Jeannerod-Villard 2003]
- randomized, general: $O(m^\omega)$ [P.-Storjohann 2007]

...in the time of matrix multiplication

Deterministic charpoly algorithm in $O(m^\omega)$

using any MatMul algorithm in $O(m^\omega)$ with $2 < \omega \leqslant 3$

(i.e. not relying on a $\tilde{O}(m^{\omega-\varepsilon})$ MatMul algorithm...)

arXiv: 2010.04662 / HAL: hal-02963147

Problem already solved?

[Bürgisser-Clausen-Shokrollahi, Algebraic Complexity Theory, 1997]

16.6* The Characteristic Polynomial

In Sect. 16.4 we saw that computing the determinant is about as hard as matrix multiplication. In this section we shall see that even the problem of computing *all* coefficients of the characteristic polynomial of a matrix has the same exponent as matrix multiplication.

Problem already solved?

[Bürgisser-Clausen-Shokrollahi, Algebraic Complexity Theory, 1997]

16.6* The Characteristic Polynomial

In Sect. 16.4 we saw that computing the determinant is about as hard as matrix multiplication. In this section we shall see that even the problem of computing *all* coefficients of the characteristic polynomial of a matrix has the same exponent as matrix multiplication.

- Definition of ω: *infimum*? *feasible*?
- Which MatMul algorithm(s) can be used in the CharPoly algorithm?

For any ω feasible (as of today),
there is a MatMul algorithm in $O(m^{\omega-\varepsilon})$ for some $\varepsilon > 0$

$\Rightarrow$ Keller-Gehrig's CharPoly algorithm is in $O(m^{\omega-\varepsilon}\log(m)) \subset O(m^{\omega})$

Framework for complexity

Typical introduction of ω in computer algebra:

"Let ω be such that $m \times m$ MatMul costs $O(m^{\omega})$ field operations"

Matrix multiplication over $\mathbb{K}$

- choose a MatMul algorithm with complexity $O(m^{\omega})$
- use this specific algorithm for all arising MatMul instances

Our requirement: $2 < \omega \leqslant 3$ (we accept $\omega = 2.1$, if you provide the MatMul algorithm)

Framework for complexity

Typical introduction of ω in computer algebra:

"Let ω be such that $m \times m$ MatMul costs $O(m^\omega)$ field operations"

Matrix multiplication over $\mathbb{K}$

- choose a MatMul algorithm with complexity $O(m^\omega)$
- use this specific algorithm for all arising MatMul instances

Our requirement: $2 < \omega \leqslant 3$ (we accept $\omega = 2.1$, if you provide the MatMul algorithm)

Univariate polynomial multiplication over $\mathbb{K}[x]$

- choose a PolMul algorithm with complexity $O(M(d))$
- use this specific algorithm for all arising PolMul instances

Requirement: $M(d)$ is superlinear and submultiplicative and reasonably good

$2M(d) \leqslant M(2d)$ $\qquad$ $M(d_1 d_2) \leqslant M(d_1)M(d_2)$ $\qquad$ $M(d) \in O(d^{\omega-1-\varepsilon})$ for some $\varepsilon > 0$

Requirement: $m \times m$ matrices over $\mathbb{K}[x]_{\leqslant d}$ multiplied in $O(m^\omega M(d))$ field ops

All these requirements are satisfied by the classical MatMul/PolMul algorithms

Charpoly via $\mathbb{K}$-linear algebra

Traces of Powers: $O(m^4)$ or $O(m^{\omega+1})$

- [LeVerrier 1840] [Faddeev'49, Souriau'48, ...]
- used by [Csanky'75] to prove $\mathcal{NC}^2$ membership

Determinant expansion: $O(m^4)$

- [Samuelson'42, Berkowitz'84]
- suited to division free algorithms [Abdlejaoued-Malaschonok'01, Kaltofen-Villard'05]

Krylov methods: [Danilevskij'37, Keller-Gehrig'85, P.-Storjohann'07]

- Deterministic $O(m^3)$ or $O(m^\omega \log(m))$
- Generic $O(m^\omega)$
- Las-Vegas probabilistic for large fields ($|\mathbb{K}| \geqslant 2m^2$) $O(m^\omega)$

Charpoly via $\mathbb{K}[x]$-linear algebra

Determinant of a matrix $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ of degree d $\qquad d = 1$

Evaluation-Interpolation: [folklore] $\qquad O(m^{\omega+1})$
at $\sim md$ points: requires large enough field

Diagonalization (Smith form)**:** [Storjohann 2003] $\qquad O(m^{\omega} \log(m)^2)$
Las Vegas randomized + additional logs for small fields

Partial triangularization:

- Iterative [Mulders-Storjohann 2003] $\qquad O(m^3)$
  via weak Popov form computations
- Divide and conquer, generic [Giorgi-Jeannerod-Villard 2003] $\qquad O(m^{\omega})$
  diagonal of Hermite form must be $1, \ldots, 1, \det(\mathbf{A})$
- Divide and conquer [N.-Labahn-Zhou 2017] $\qquad \tilde{O}(m^{\omega})$
  logarithmic factors in m and d

Sources of log factors

In $\mathbb{K}$-linear algebra

- divide and conquer with half-dimension blocks $\rightarrow$ no $\log(m)$
- iterative approaches in m steps $\rightarrow$ sometimes no $\log(m)$ [P.-Storjohann'07]
- explicit Krylov iteration: compute $\begin{pmatrix} v & \mathbf{M}v & \cdots & \mathbf{M}^m v \end{pmatrix}$ $\rightarrow$ $\log(m) \times$ MatMul

In $\mathbb{K}[x]$-linear algebra

- divide and conquer with half-dimension blocks $\rightarrow$ no $\log(m)$

  provided degrees are controlled, e.g. kernel basis [Zhou-Labahn-Storjohann'12]
- divide and conquer on degree $\rightarrow$ $\log(d)$ but no $\log(m)$

  e.g. $\mathbb{K}[x]$-MatMul and approximant basis [Giorgi-Jeannerod-Villard'03]
- explicit Krylov iterations here as well $[\star]$

  because base cases of recursions on degree = matrices over $\mathbb{K}$ e.g. [Jeannerod-N.-Schost-Villard'17]
- looking for a matrix with unpredictable, unbalanced degrees

  up to $\sim \log(m)$ steps, each in dimension $m \times m$, to uncover the degree profile [Zhou-Labahn'13]

  reminiscent of long Krylov chains with small dimension drop & failure to derandomize [P.-Storjohann'07]

$[\star]$ typically contributes $O(m^\omega d \log(m))$ to the cost $\rightsquigarrow$ cannot be ignored for $d = O(1)$

Outline

- Context, problem, state of the art
- Overview of the approach and complexity
- Obstacles and related spin-off results

Partial block triangularization

[Mulders-Storjohann 2003, Giorgi-Jeannerod-Villard 2003, Zhou 2012, N.-Labahn-Zhou 2017]

Triangularization of $m \times m$ matrix $\mathbf{A}$ using $m/2 \times m/2$ blocks

not computed

$$\begin{bmatrix} * & * \\ \mathbf{K}_1 & \mathbf{K}_2 \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & * \\ \mathbf{0} & \mathbf{B} \end{bmatrix}$$

kernel basis of $\left[\begin{smallmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{smallmatrix}\right]$

$\mathbf{K}_1\mathbf{A}_2 + \mathbf{K}_2\mathbf{A}_4$

row basis of $\left[\begin{smallmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{smallmatrix}\right]$

Property: $\det(\mathbf{A}) = \det(\mathbf{R})\det(\mathbf{B})$

Partial block triangularization

[Mulders-Storjohann 2003, Giorgi-Jeannerod-Villard 2003, Zhou 2012, N.-Labahn-Zhou 2017]

Triangularization of $m \times m$ matrix $\mathbf{A}$ using $m/2 \times m/2$ blocks

not computed

$$\begin{bmatrix} * & * \\ \mathbf{K}_1 & \mathbf{K}_2 \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & * \\ \mathbf{0} & \mathbf{B} \end{bmatrix}$$

kernel basis of $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$

$\mathbf{K}_1 \mathbf{A}_2 + \mathbf{K}_2 \mathbf{A}_4$

row basis of $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$

Property: $\det(\mathbf{A}) = \det(\mathbf{R}) \det(\mathbf{B})$

Generic case without log factor

[Mulders-Storjohann 2003, Giorgi-Jeannerod-Villard 2003, Zhou 2012, N.-Labahn-Zhou 2017]

Triangulation of $m \times m$ matrix $\mathbf{A}$ using $m/2 \times m/2$ blocks

not computed

$$\begin{bmatrix} * & * \\ \mathbf{K}_1 & \mathbf{K}_2 \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & * \\ \mathbf{0} & \mathbf{B} \end{bmatrix}$$

kernel basis of $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$ $\qquad$ $\mathbf{K}_1\mathbf{A}_2 + \mathbf{K}_2\mathbf{A}_4$ $\qquad$ row basis of $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$

Property: $\det(\mathbf{A}) = \det(\mathbf{R})\det(\mathbf{B})$

Generic input $\Rightarrow \det(\mathbf{A})$ **without** $\log(m)$ [Giorgi-Jeannerod-Villard 2003]

$\mathbf{A}_1$ and $\mathbf{A}_3$ are coprime $\Rightarrow \mathbf{R} = \mathbf{I}_{m/2} \Rightarrow \det(\mathbf{A}) = \det(\mathbf{B})$

- Compute kernel $[\mathbf{K}_1 \;\; \mathbf{K}_2]$; deduce $\mathbf{B}$ by MatMul $\qquad O(m^\omega M'(d))$
- Recursively, compute $\det(\mathbf{B})$, return it

$\mathbf{A}$ and $[\mathbf{K}_1 \;\; \mathbf{K}_2]$ have degree $d \Rightarrow \mathbf{B}$ has degree $2d$: controlled total degree

GCD in $\leqslant M'(d) \in O(M(d)\log(d))$ f.ops.

total cost: $O(m^\omega M'(d) + (m/2)^\omega M'(2d) + \cdots + M'(md)) \subset O(m^\omega M'(d))$

General case with log factor

[Mulders-Storjohann 2003, Giorgi-Jeannerod-Villard 2003, Zhou 2012, N.-Labahn-Zhou 2017]

Triangularization of $m \times m$ matrix $\mathbf{A}$ using $m/2 \times m/2$ blocks

not computed

$$\begin{bmatrix} * & * \\ \mathbf{K}_1 & \mathbf{K}_2 \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & * \\ \mathbf{0} & \mathbf{B} \end{bmatrix}$$

kernel basis of $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$ $\quad$ $\mathbf{K}_1\mathbf{A}_2 + \mathbf{K}_2\mathbf{A}_4$ $\quad$ row basis of $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$

Property: $\det(\mathbf{A}) = \det(\mathbf{R})\det(\mathbf{B})$

Matrix degree not controlled: degree of $\mathbf{B}$ up to $D = \sum \mathrm{rdeg}(\mathbf{A}) \leqslant md$
but controlled average row degree: at most $\frac{D}{m}$

General input $\Rightarrow$ $\det(\mathbf{A})$ in $\tilde{O}(m^{\omega}\frac{D}{m})$ [Labahn-N.-Zhou 2017]

- Compute kernel $[\mathbf{K}_1 \ \ \mathbf{K}_2]$; deduce $\mathbf{B}$ by MatMul $\quad O(m^{\omega}M'(\frac{D}{m}))$
- Compute row basis $\mathbf{R}$ $\quad \tilde{O}(m^{\omega}\frac{D}{m})$ with $\log(m)$
- Recursively, compute $\det(\mathbf{R})$ and $\det(\mathbf{B})$, return $\det(\mathbf{R})\det(\mathbf{B})$

Be lazy: if hard to compute, don't compute

[Mulders-Storjohann 2003, Giorgi-Jeannerod-Villard 2003, Zhou 2012, N.-Labahn-Zhou 2017]

Triangularization of $m \times m$ matrix $\mathbf{A}$ using $m/2 \times m/2$ blocks

not computed

$$\begin{bmatrix} * & * \\ \mathbf{K}_1 & \mathbf{K}_2 \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & * \\ \mathbf{0} & \mathbf{B} \end{bmatrix}$$

kernel basis of $\left[\begin{smallmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{smallmatrix}\right]$ $\quad$ $\mathbf{K}_1\mathbf{A}_2 + \mathbf{K}_2\mathbf{A}_4$ $\quad$ row basis of $\left[\begin{smallmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{smallmatrix}\right]$

Property: $\det(\mathbf{A}) = \det(\mathbf{R}) \det(\mathbf{B})$

Obstacle: removing log factors in row basis computation

$\Rightarrow$ solution: remove row basis computation

$$\begin{bmatrix} \mathbf{I}_{m/2} & \mathbf{0} \\ \mathbf{K}_1 & \mathbf{K}_2 \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{0} & \mathbf{B} \end{bmatrix}$$

Property: $\det(\mathbf{A}) = \det(\mathbf{A}_1) \det(\mathbf{B}) / \det(\mathbf{K}_2)$

Further obstacles (brought by laziness)

$$\begin{bmatrix} \mathbf{I}_{m/2} & \mathbf{0} \\ \mathbf{K}_1 & \mathbf{K}_2 \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{0} & \mathbf{B} \end{bmatrix}$$

Property: $\det(\mathbf{A}) = \det(\mathbf{A}_1)\det(\mathbf{B}) / \det(\mathbf{K}_2)$

- no $\log(m)$ in the computation of $\mathbf{A}_1$, $\mathbf{B}$, $\mathbf{K}_2$
- requires nonsingular $\mathbf{A}_1$, otherwise $\det(\mathbf{K}_2) = 0$
- 3 recursive calls in matrix size $m/2$ is 👍, but requires $\sum \mathrm{rdeg}(\mathbf{A}_1) \leqslant D/2$ otherwise degree control is too weak. (this implies $\sum \mathrm{rdeg}(\mathbf{K}_2) \leqslant D/2$)

Further obstacles (brought by laziness)

$$\begin{bmatrix} \mathbf{I}_{m/2} & \mathbf{0} \\ \mathbf{K}_1 & \mathbf{K}_2 \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{0} & \mathbf{B} \end{bmatrix}$$

Property: $\det(\mathbf{A}) = \det(\mathbf{A}_1)\det(\mathbf{B})/\det(\mathbf{K}_2)$

- 👍 no $\log(m)$ in the computation of $\mathbf{A}_1$, $\mathbf{B}$, $\mathbf{K}_2$
- 👎 requires nonsingular $\mathbf{A}_1$, otherwise $\det(\mathbf{K}_2) = 0$
- 👎 3 recursive calls in matrix size $m/2$ is 👍, but requires $\sum \mathrm{rdeg}(\mathbf{A}_1) \leqslant D/2$ otherwise degree control is too weak. (this implies $\sum \mathrm{rdeg}(\mathbf{K}_2) \leqslant D/2$)

Solution: require $\mathbf{A}$ in weak Popov form

(the characteristic matrix $\mathbf{A} = x\mathbf{I}_m - \mathbf{M}$ is in Popov form)

- 👍 implies $\mathbf{A}_1$ nonsingular and $\sum \mathrm{rdeg}(\mathbf{A}_1) \leqslant D/2$ up to easy transformations
- 👍 both $\mathbf{A}_1$ and $\mathbf{B}$ are also in weak Popov form $\Rightarrow$ suitable for recursive calls
- 👎 $\mathbf{K}_2$ is in “shifted reduced” form… find weak Popov $\mathbf{P}$ with same determinant

Complexity

$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}\left(\frac{m}{2}, \left\lfloor\frac{D}{2}\right\rfloor\right) + \mathcal{C}\left(\frac{m}{2}, D\right) + O\left(m^{\omega} M'\left(\frac{D}{m}\right)\right)$$

where: $M'(d) = \mathsf{GCD}(d) \in O(M(d)\log(d))$

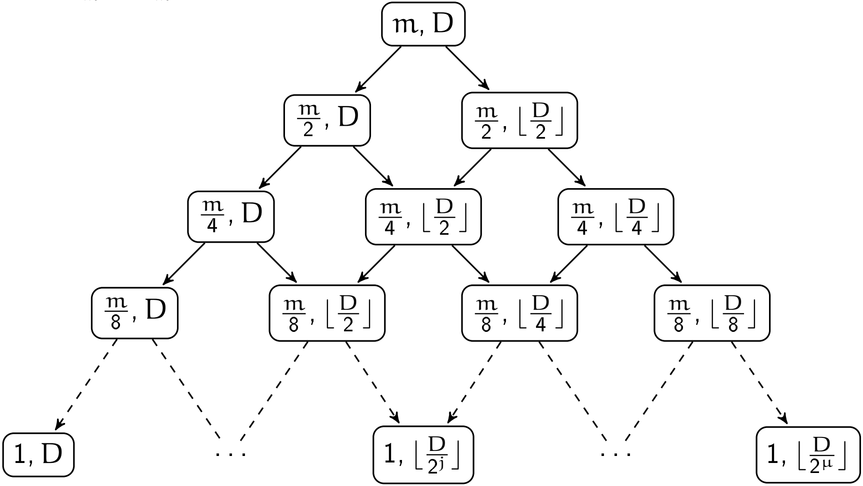
$\frac{D}{m} = \frac{\mathsf{degdet}}{m} =$ avg row degree

Complexity

$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}\left(\frac{m}{2}, \left\lfloor\frac{D}{2}\right\rfloor\right) + \mathcal{C}\left(\frac{m}{2}, D\right) + O\left(m^{\omega} M'\left(\frac{D}{m}\right)\right)$$

where: $M'(d) = \mathsf{GCD}(d) \in O(M(d)\log(d))$

$\frac{D}{m} = \frac{\mathsf{degdet}}{m} =$ avg row degree

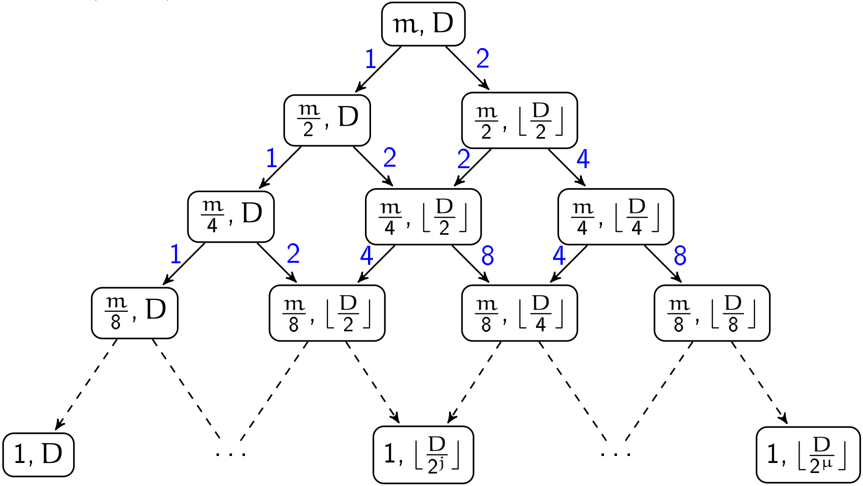


Complexity

$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}\left(\frac{m}{2}, \left\lfloor\frac{D}{2}\right\rfloor\right) + \mathcal{C}\left(\frac{m}{2}, D\right) + O\left(m^{\omega} M'\left(\frac{D}{m}\right)\right)$$

where: $M'(d) = \mathsf{GCD}(d) \in O(M(d)\log(d))$

$\frac{D}{m} = \frac{\mathsf{degdet}}{m} =$ avg row degree

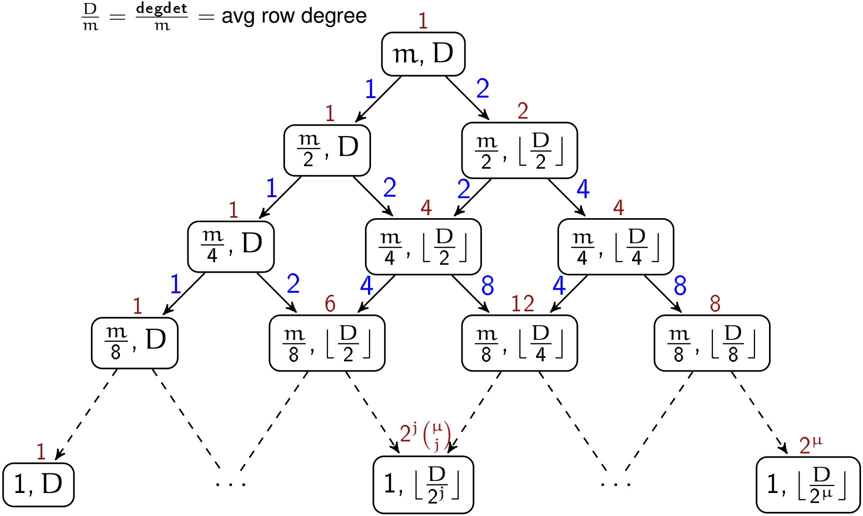


Complexity

$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}\left(\frac{m}{2}, \left\lfloor\frac{D}{2}\right\rfloor\right) + \mathcal{C}\left(\frac{m}{2}, D\right) + O\left(m^{\omega} M'\left(\frac{D}{m}\right)\right)$$

where: $M'(d) = GCD(d) \in O(M(d)\log(d))$

$\frac{D}{m} = \frac{\text{degdet}}{m} =$ avg row degree

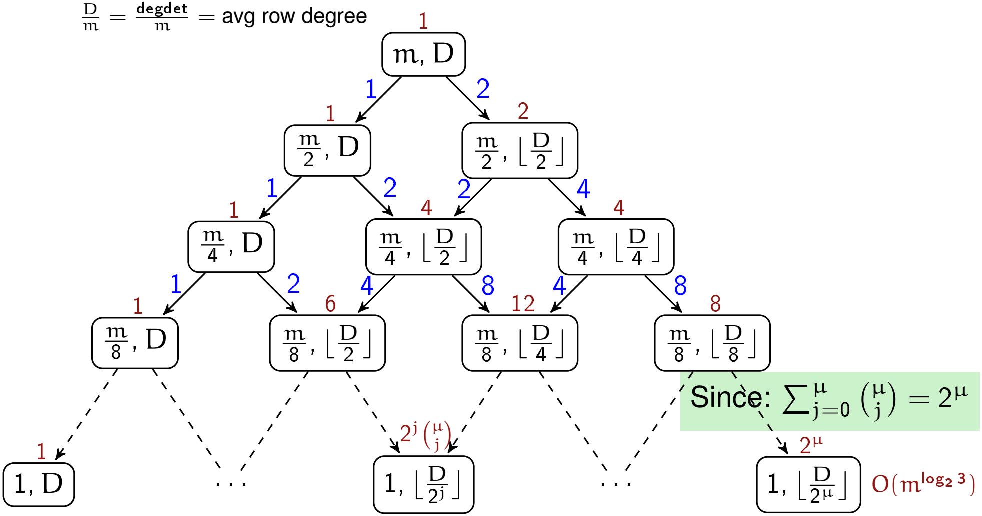


Complexity

$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}\left(\frac{m}{2}, \left\lfloor \frac{D}{2} \right\rfloor\right) + \mathcal{C}\left(\frac{m}{2}, D\right) + O\left(m^{\omega} M'\left(\frac{D}{m}\right)\right)$$

where: $M'(d) = \mathrm{GCD}(d) \in O(M(d)\log(d))$

$\frac{D}{m} = \frac{\mathrm{degdet}}{m} =$ avg row degree



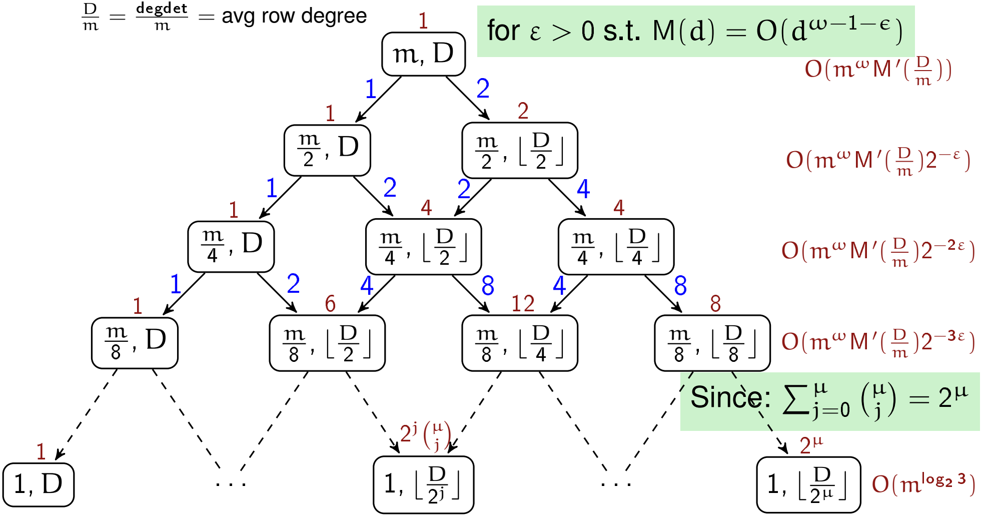
Complexity

$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}\left(\frac{m}{2}, \left\lfloor \frac{D}{2} \right\rfloor\right) + \mathcal{C}\left(\frac{m}{2}, D\right) + O\left(m^{\omega} M'\left(\frac{D}{m}\right)\right)$$

where: $M'(d) = \mathrm{GCD}(d) \in O(M(d)\log(d))$

$\frac{D}{m} = \frac{\mathrm{degdet}}{m} =$ avg row degree

for $\varepsilon > 0$ s.t. $M(d) = O(d^{\omega-1-\epsilon})$



Since: $\sum_{j=0}^{\mu} \binom{\mu}{j} = 2^\mu$

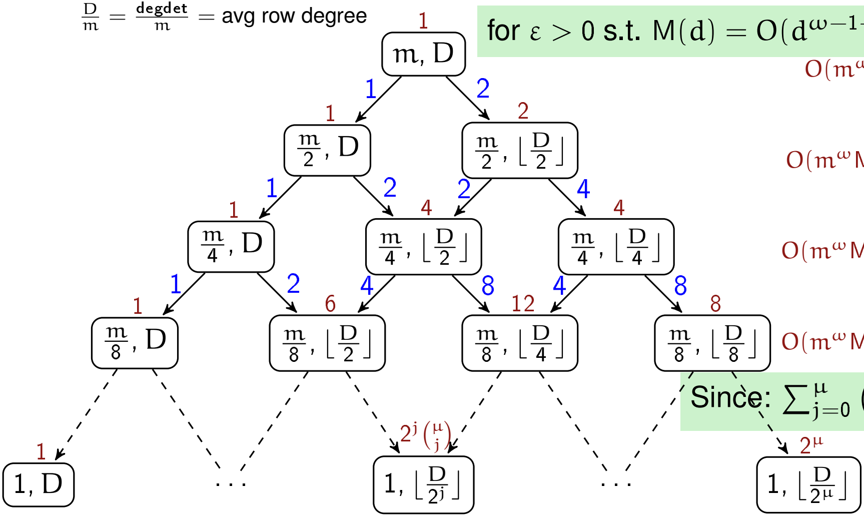
Complexity

$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}\left(\frac{m}{2}, \left\lfloor\frac{D}{2}\right\rfloor\right) + \mathcal{C}\left(\frac{m}{2}, D\right) + O\left(m^{\omega} M'\left(\frac{D}{m}\right)\right) \leqslant O\left(m^{\omega} M'\left(\frac{D}{m}\right)\right)$$

where: $M'(d) = \mathrm{GCD}(d) \in O(M(d)\log(d))$

$\frac{D}{m} = \frac{\mathrm{degdet}}{m}$ = avg row degree

for $\varepsilon > 0$ s.t. $M(d) = O(d^{\omega-1-\epsilon})$



$O(m^{\omega} M'(\frac{D}{m}))$

$O(m^{\omega} M'(\frac{D}{m}) 2^{-\varepsilon})$

$O(m^{\omega} M'(\frac{D}{m}) 2^{-2\varepsilon})$

$O(m^{\omega} M'(\frac{D}{m}) 2^{-3\varepsilon})$

Since: $\sum_{j=0}^{\mu} \binom{\mu}{j} = 2^{\mu}$

$O(m^{\log_2 3})$

Outline

- Context, problem, state of the art
- Overview of the approach and complexity
- Obstacles and related spin-off results

Hermite and Popov forms

$\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ nonsingular

elementary row transformations

Hermite form [Hermite, 1851]

- triangular
- column normalized

$$\begin{bmatrix} \mathbf{16} & & & \\ 15 & \mathbf{0} & & \\ 15 & & \mathbf{0} & \\ 15 & & & \mathbf{0} \end{bmatrix} \qquad \begin{bmatrix} \mathbf{4} & & & \\ 3 & \mathbf{7} & & \\ 1 & 5 & \mathbf{3} & \\ 3 & 6 & 1 & \mathbf{2} \end{bmatrix}$$

Hermite and Popov forms

$\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ nonsingular

elementary row transformations

Hermite form [Hermite, 1851]

- triangular
- column normalized

$$\begin{bmatrix} \mathbf{16} & & & \\ 15 & \mathbf{0} & & \\ 15 & & \mathbf{0} & \\ 15 & & & \mathbf{0} \end{bmatrix} \qquad \begin{bmatrix} \mathbf{4} & & & \\ 3 & \mathbf{7} & & \\ 1 & 5 & \mathbf{3} & \\ 3 & 6 & 1 & \mathbf{2} \end{bmatrix}$$

Popov form [Popov, 1972]

- row reduced / distinct pivots
- column normalized

$$\begin{bmatrix} \mathbf{4} & 3 & 3 & 3 \\ 3 & \mathbf{4} & 3 & 3 \\ 3 & 3 & \mathbf{4} & 3 \\ 3 & 3 & 3 & \mathbf{4} \end{bmatrix} \qquad \begin{bmatrix} \mathbf{7} & 0 & 1 & 5 \\ 0 & \mathbf{1} & & 0 \\ & & \mathbf{2} & \\ 6 & 0 & 1 & \mathbf{6} \end{bmatrix}$$

Invariant: $D = \deg(\det(\mathbf{A})) = 4+7+3+2 = 7+1+2+6$

Hermite and Popov forms

$\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ nonsingular

elementary row transformations

Hermite form [Hermite, 1851]

- triangular
- column normalized

$$\begin{bmatrix} \mathbf{16} & & & \\ 15 & \mathbf{0} & & \\ 15 & & \mathbf{0} & \\ 15 & & & \mathbf{0} \end{bmatrix} \qquad \begin{bmatrix} \mathbf{4} & & & \\ 3 & \mathbf{7} & & \\ 1 & 5 & \mathbf{3} & \\ 3 & 6 & 1 & \mathbf{2} \end{bmatrix}$$

Popov form [Popov, 1972]

- row reduced / distinct pivots
- column normalized

$$\begin{bmatrix} \mathbf{4} & 3 & 3 & 3 \\ 3 & \mathbf{4} & 3 & 3 \\ 3 & 3 & \mathbf{4} & 3 \\ 3 & 3 & 3 & \mathbf{4} \end{bmatrix} \qquad \begin{bmatrix} \mathbf{7} & 0 & 1 & 5 \\ 0 & \mathbf{1} & & 0 \\ & & \mathbf{2} & \\ 6 & 0 & 1 & \mathbf{6} \end{bmatrix}$$

position over term

reduced Gröbner basis

term over position

$\mathbb{K}[x]$-module $\mathcal{M} \subset \mathbb{K}[x]^{1 \times m}$ of rank m

Invariant: $D = \deg(\det(\mathbf{A})) = 4 + 7 + 3 + 2 = 7 + 1 + 2 + 6 = \dim_{\mathbb{K}}(\mathbb{K}[x]^{1 \times m} / \mathcal{M})$

Hermite and Popov forms

$\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ nonsingular

elementary row transformations

Hermite form [Hermite, 1851]

- triangular
- column normalized

$$\begin{bmatrix} \mathbf{16} & & & \\ 15 & \mathbf{0} & & \\ 15 & & \mathbf{0} & \\ 15 & & & \mathbf{0} \end{bmatrix} \qquad \begin{bmatrix} \mathbf{4} & & & \\ 3 & \mathbf{7} & & \\ 1 & 5 & \mathbf{3} & \\ 3 & 6 & 1 & \mathbf{2} \end{bmatrix}$$

Popov form [Popov, 1972]

- row reduced / distinct pivots
- column normalized

$$\begin{bmatrix} \mathbf{4} & 3 & 3 & 3 \\ 3 & \mathbf{4} & 3 & 3 \\ 3 & 3 & \mathbf{4} & 3 \\ 3 & 3 & 3 & \mathbf{4} \end{bmatrix} \qquad \begin{bmatrix} \mathbf{7} & 0 & 1 & 5 \\ 0 & \mathbf{1} & & 0 \\ & & \mathbf{2} & \\ 6 & 0 & 1 & \mathbf{6} \end{bmatrix}$$

position over term

reduced Gröbner basis

term over position

$\mathbb{K}[x]$-module $\mathcal{M} \subset \mathbb{K}[x]^{1 \times m}$ of rank m

Invariant: $D = \deg(\det(\mathbf{A})) = 4 + 7 + 3 + 2 = 7 + 1 + 2 + 6 = \dim_{\mathbb{K}}(\mathbb{K}[x]^{1 \times m} / \mathcal{M})$

Weak Popov form [Beckermann-Labahn-Villard'99, Mulders-Storjohann'03]

not column normalized = minimal, non-reduced, t.o.p.-Gröbner basis

Shifted forms

Shift: integer tuple $s = (s_1, \ldots, s_m)$ acting as column weights

$\rightsquigarrow$ connects Popov and Hermite forms:

$\mathbf{s} = (0, 0, 0, 0)$
Popov

$$\begin{bmatrix} \mathbf{4} & 3 & 3 & 3 \\ 3 & \mathbf{4} & 3 & 3 \\ 3 & 3 & \mathbf{4} & 3 \\ 3 & 3 & 3 & \mathbf{4} \end{bmatrix} \qquad \begin{bmatrix} \mathbf{7} & 0 & 1 & 5 \\ 0 & \mathbf{1} & & 0 \\ & & \mathbf{2} & \\ 6 & 0 & 1 & \mathbf{6} \end{bmatrix}$$

$\mathbf{s} = (0, 2, 4, 6)$
s-Popov

$$\begin{bmatrix} \mathbf{7} & 4 & 2 & 0 \\ 6 & \mathbf{5} & 2 & 0 \\ 6 & 4 & \mathbf{3} & 0 \\ 6 & 4 & 2 & \mathbf{1} \end{bmatrix} \qquad \begin{bmatrix} \mathbf{8} & 5 & 1 & \\ 7 & \mathbf{6} & 1 & \\ & & \mathbf{2} & \\ 0 & 1 & & \mathbf{0} \end{bmatrix}$$

$\mathbf{s} = (0, \mathrm{D}, 2\mathrm{D}, 3\mathrm{D})$
Hermite

$$\begin{bmatrix} \mathbf{16} & & & \\ 15 & \mathbf{0} & & \\ 15 & & \mathbf{0} & \\ 15 & & & \mathbf{0} \end{bmatrix} \qquad \begin{bmatrix} \mathbf{4} & & & \\ 3 & \mathbf{7} & & \\ 1 & 5 & \mathbf{3} & \\ 3 & 6 & 1 & \mathbf{2} \end{bmatrix}$$

- shifts arise naturally in algorithms (approximants, kernel, ...)
- they allow one to specify non-uniform degree constraints

Back to our obstacles: easy ones

Recall: $\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix}$ in weak Popov form, we want:

- $\mathbf{A}_1$ nonsingular: ok by definition
  (all principal submatrices of $\mathbf{A}$ are weak Popov $\Rightarrow$ are nonsingular)
- $\sum \mathrm{rdeg}(\mathbf{A}_1) \leqslant D/2$: either ok for $\mathbf{A}$, or ok for $\begin{bmatrix} \mathbf{A}_4 & \mathbf{A}_3 \\ \mathbf{A}_2 & \mathbf{A}_1 \end{bmatrix}$
  (*almost* weak Popov... easily transformed into it, with same determinant)

Back to our obstacles: easy ones

Recall: $\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix}$ in weak Popov form, we want:

- $\mathbf{A}_1$ nonsingular: ok by definition
  (all principal submatrices of $\mathbf{A}$ are weak Popov $\Rightarrow$ are nonsingular)
- $\sum \mathrm{rdeg}(\mathbf{A}_1) \leqslant D/2$: either ok for $\mathbf{A}$, or ok for $\begin{bmatrix} \mathbf{A}_4 & \mathbf{A}_3 \\ \mathbf{A}_2 & \mathbf{A}_1 \end{bmatrix}$
  (*almost* weak Popov... easily transformed into it, with same determinant)

Shifts in kernel basis computation [Zhou-Labahn-Storjohann'12]

$[\mathbf{K}_1 \;\; \mathbf{K}_2]$ kernel basis of $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$ computed in $\mathrm{rdeg}(\mathbf{A})$-weak Popov form:
cost $O(m^\omega M'(\frac{D}{m}))$, $\quad \sum \mathrm{rdeg}(\mathbf{K}_2) \leqslant D/2$, $\quad$ and $\mathbf{K}_2$ in $\mathbf{s}$-weak Popov form

$D = \sum \mathrm{rdeg}(\mathbf{A}) = \deg\det(\mathbf{A})$ $\qquad$ $\mathbf{s} = \mathrm{rdeg}(\mathbf{A}_4) =$ last $m/2$ entries of $\mathrm{rdeg}(\mathbf{A})$

Using the shift $\mathrm{rdeg}(\mathbf{A})$ (and $\mathbf{s}$) has many crucial advantages:

- *towards correctness:* product $\mathbf{B} = [\mathbf{K}_1 \;\; \mathbf{K}_2]\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$ is in $\mathbf{0}$-weak Popov form
- *towards efficiency:* implies small degrees in $\mathbf{K}_2$
  and best speed both for kernel and product $\mathbf{B}$

Back to our obstacles: easy ones

Recall: $\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix}$ in weak Popov form, we want:

- $\mathbf{A}_1$ nonsingular: ok by definition
  (all principal submatrices of $\mathbf{A}$ are weak Popov $\Rightarrow$ are nonsingular)
- $\sum \mathrm{rdeg}(\mathbf{A}_1) \leqslant D/2$: either ok for $\mathbf{A}$, or ok for $\begin{bmatrix} \mathbf{A}_4 & \mathbf{A}_3 \\ \mathbf{A}_2 & \mathbf{A}_1 \end{bmatrix}$
  (*almost* weak Popov... easily transformed into it, with same determinant)

Shifts in kernel basis computation [Zhou-Labahn-Storjohann'12]

$[\mathbf{K}_1 \;\; \mathbf{K}_2]$ kernel basis of $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$ computed in $\mathrm{rdeg}(\mathbf{A})$-weak Popov form:
cost $O(m^\omega M'(\frac{D}{m}))$, $\quad \sum \mathrm{rdeg}(\mathbf{K}_2) \leqslant D/2$, $\quad$ and $\mathbf{K}_2$ in $\mathbf{s}$-weak Popov form

$D = \sum \mathrm{rdeg}(\mathbf{A}) = \deg\det(\mathbf{A})$ $\qquad$ $\mathbf{s} = \mathrm{rdeg}(\mathbf{A}_4) =$ last $m/2$ entries of $\mathrm{rdeg}(\mathbf{A})$

Using the shift $\mathrm{rdeg}(\mathbf{A})$ (and $\mathbf{s}$) has many crucial advantages:

- *towards correctness:* product $\mathbf{B} = [\mathbf{K}_1 \;\; \mathbf{K}_2]\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$ is in $\mathbf{0}$-weak Popov form
- *towards efficiency:* implies small degrees in $\mathbf{K}_2$
  and best speed both for kernel and product $\mathbf{B}$

...but we cannot call the algorithm recursively on K_2

Approaching the main obstacle

Given $\mathbf{K}_2$ in $\mathbf{s}$-weak Popov form, with $\mathbf{s} \geqslant 0$
Find $\mathbf{P}$ in $\mathbf{0}$-weak Popov form with the same determinant

Idea 1.a: change of shift from $\mathbf{s}$ to $\mathbf{0}$, i.e. $\mathbf{P} = \mathsf{WeakPopov}(\mathbf{K}_2)$

known methods are only efficient for increasing $\mathbf{s}$ to a larger shift
[Jeannerod-N.-Schost-Villard'17]

Idea 1.b: normalization of $\mathbf{K}_2$ into its $\mathbf{s}$-Popov form $\mathbf{P}$

$\rightsquigarrow$ $\mathbf{P}^\mathsf{T}$ is weak Popov by construction, and $\det(\mathbf{P}^\mathsf{T}) = \det(\mathbf{P})$

amounts to a change of shift from $\mathbf{s}$ to $-\delta \leqslant 0$ [N.'16] $\Rightarrow$ same issue

Approaching the main obstacle

Given $\mathbf{K}_2$ in $\mathbf{s}$-weak Popov form, with $\mathbf{s} \geqslant 0$
Find $\mathbf{P}$ in $\mathbf{0}$-weak Popov form with the same determinant

Idea 1.a: change of shift from $\mathbf{s}$ to $\mathbf{0}$, i.e. $\mathbf{P} = \mathrm{WeakPopov}(\mathbf{K}_2)$

known methods are only efficient for increasing $\mathbf{s}$ to a larger shift [Jeannerod-N.-Schost-Villard'17]

Idea 1.b: normalization of $\mathbf{K}_2$ into its $\mathbf{s}$-Popov form $\mathbf{P}$

$\rightsquigarrow$ $\mathbf{P}^\mathsf{T}$ is weak Popov by construction, and $\det(\mathbf{P}^\mathsf{T}) = \det(\mathbf{P})$

amounts to a change of shift from $\mathbf{s}$ to $-\delta \leqslant 0$ [N.'16] $\Rightarrow$ same issue

Fact: $\mathbf{K}_2^\mathsf{T}$ is $-\mathbf{t}$-weak Popov $\qquad \mathbf{t} = \mathrm{rdeg}_\mathbf{s}(\mathbf{K}_2) = \mathbf{s} + \delta \geqslant 0$

(for simplicity some row and column permutations are ignored)

Idea 2.a: change of shift from $-\mathbf{t}$ to $\mathbf{0}$, i.e. $\mathbf{P} = \mathrm{WeakPopov}(\mathbf{K}_2^\mathsf{T})$

increasing shift, but $\mathbf{K}_2^\mathsf{T}$ has large average rdeg (we control $\mathrm{cdeg}(\mathbf{K}_2^\mathsf{T}) = \mathrm{rdeg}(\mathbf{K}_2)$)

Approaching the main obstacle

Given $\mathbf{K}_2$ in s-weak Popov form, with $s \geqslant 0$
Find $\mathbf{P}$ in $\mathbf{0}$-weak Popov form with the same determinant

Idea 1.a: change of shift from s to $\mathbf{0}$, i.e. $\mathbf{P} = \mathrm{WeakPopov}(\mathbf{K}_2)$

known methods are only efficient for increasing s to a larger shift
[Jeannerod-N.-Schost-Villard'17]

Idea 1.b: normalization of $\mathbf{K}_2$ into its s-Popov form $\mathbf{P}$

$\rightsquigarrow$ $\mathbf{P}^\mathsf{T}$ is weak Popov by construction, and $\det(\mathbf{P}^\mathsf{T}) = \det(\mathbf{P})$

amounts to a change of shift from s to $-\delta \leqslant 0$ [N.'16] $\Rightarrow$ same issue

Fact: $\mathbf{K}_2^\mathsf{T}$ is $-\mathbf{t}$-weak Popov $\quad \mathbf{t} = \mathrm{rdeg}_s(\mathbf{K}_2) = s + \delta \geqslant 0$

(for simplicity some row and column permutations are ignored)

Idea 2.a: change of shift from $-\mathbf{t}$ to $\mathbf{0}$, i.e. $\mathbf{P} = \mathrm{WeakPopov}(\mathbf{K}_2^\mathsf{T})$

increasing shift, but $\mathbf{K}_2^\mathsf{T}$ has large average rdeg (we control $\mathrm{cdeg}(\mathbf{K}_2^\mathsf{T}) = \mathrm{rdeg}(\mathbf{K}_2)$)

Idea 2.b: normalization of $\mathbf{K}_2^\mathsf{T}$ into its $-\mathbf{t}$-Popov form $\mathbf{P}$

Spin-off: new transformations

Weak Popov to Popov

Input:	$\mathbf{t} \in \mathbb{Z}_{\geqslant 0}^{m}$ a nonnegative shift,
	$\mathbf{K} \in \mathbb{K}[x]^{m \times m}$ a matrix in $-\mathbf{t}$-weak Popov form
Output:	the $-\mathbf{t}$-Popov form of $\mathbf{K}$
Requirement:	$\mathbf{t} \geqslant \delta := \mathrm{pivotDegree}(\mathbf{K})$
Complexity:	$O(m^{\omega} M'(\frac{D}{m}))$, where $D = \sum \mathbf{t}$

Improvement and generalization of [Sarkar-Storjohann 2011, Section 4]
$\rightsquigarrow$ support nonzero shifts and involve average degree $\frac{D}{m}$

- problem viewed as a change of shift with a priori known output degrees
- introduction of partial linearization techniques for kernel bases

Spin-off: new transformations

Weak Popov to Popov

Input:	$t \in \mathbb{Z}^m_{\geqslant 0}$ a nonnegative shift,
	$\mathbf{K} \in \mathbb{K}[x]^{m\times m}$ a matrix in $-t$-weak Popov form
Output:	the $-t$-Popov form of $\mathbf{K}$
Requirement:	$t \geqslant \delta := \mathrm{pivotDegree}(\mathbf{K})$
Complexity:	$O(m^\omega M'(\frac{D}{m}))$, where $D = \sum t$

Improvement and generalization of [Sarkar-Storjohann 2011, Section 4]

$\rightsquigarrow$ support nonzero shifts and involve average degree $\frac{D}{m}$

- problem viewed as a change of shift with a priori known output degrees
- introduction of partial linearization techniques for kernel bases

Reduced to weak Popov

Input:	$s \in \mathbb{Z}^n$ a shift
	$\mathbf{A} \in \mathbb{K}[x]^{m\times n}$ a matrix in s-reduced form
Output:	an s-weak Popov form of $\mathbf{A}$
Complexity:	$O(m^{\omega-1} n(\frac{D}{m} + 1))$, where $D = \sum \mathrm{rdeg}_s(\mathbf{A}) - m \min(s)$

Easy extension of [Sarkar-Storjohann 2011, Section 3] to shifted forms

Summary and perspectives

Summary

- $\mathsf{CharPoly} = O(\mathsf{MatMul})$
- Determinant of reduced polynomial matrices in $O(m^{\omega}\mathsf{M}'(\frac{D}{m}))$
- Fast transformations between shifted forms of polynomial matrices

$\frac{D}{m} = \frac{\mathbf{degdet}}{m} =$ average row degree

Summary and perspectives

Summary

- $\mathsf{CharPoly} = O(\mathsf{MatMul})$
- Determinant of reduced polynomial matrices in $O(m^\omega M'(\frac{D}{m}))$
- Fast transformations between shifted forms of polynomial matrices

$\frac{D}{m} = \frac{\mathbf{degdet}}{m} =$ average row degree

Perspectives

- Implementation and practical efficiency (small fields, degenerate instances, ...)
- Approach without fast polynomial arithmetic
  $\rightarrow$ Exploit the quasiseparable struct. of linearized polynomial matrices
- Frobenius normal form & Smith normal form