Vincent Neiger
XLIM, Univ. Limoges, France
→ LIP6, Sorbonne Univ., France

Clément Pernet
LJK, Univ. Grenoble Alpes, France

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

Séminaire Aric, LIP, ENS de Lyon
March 9, 2022

outline

- context & result
- previous work
- overview of the approach
- obstacles & spin-offs

outline

- context & result
  - computing with matrices over $\mathbb{K}$ and $\mathbb{K}[x]$
  - reductions to matrix multiplication
  - framework for complexity bounds
- previous work
- overview of the approach
- obstacles & spin-offs

matrices: multiplication

$$\mathbf{M} = \begin{bmatrix} 28 & 68 & 75 & 70 \\ 38 & 25 & 75 & 55 \\ 24 & 1 & 56 & 28 \end{bmatrix} \in \mathbb{K}^{3\times 4} \longrightarrow 3 \times 4 \text{ matrix over } \mathbb{K} \text{ (here } \mathbb{F}_{97})$$

fundamental operations on $m \times m$ matrices:

- addition is "quadratic": $O(m^2)$ operations in $\mathbb{K}$
- naive multiplication is cubic: $O(m^3)$

[Strassen'69]

subcubic matrix multiplication

matrices: multiplication

$$\mathbf{M} = \begin{bmatrix} 28 & 68 & 75 & 70 \\ 38 & 25 & 75 & 55 \\ 24 & 1 & 56 & 28 \end{bmatrix} \in \mathbb{K}^{3\times 4}$$ $\longrightarrow$ 3×4 matrix over $\mathbb{K}$ (here $\mathbb{F}_{97}$)

fundamental operations on $m \times m$ matrices:

- addition is "quadratic": $O(m^2)$ operations in $\mathbb{K}$
- naive multiplication is cubic: $O(m^3)$

[Strassen'69]

subcubic matrix multiplication

- complexity exponent $\omega \approx 2.81$ — i.e. $O(m^{\omega})$ complexity
- used in practice for $m \geqslant$ a few 100s
  in NTL, FLINT, fflas-ffpack...

- best-known exponent $\omega \approx 2.373$
  [Le Gall'14] [Alman-Williams'20]
- "galactic" algorithms: strongly impractical as such

measuring efficiency: algebraic complexity

efficient algorithms for polynomials, matrices, ...
with coefficients in some base field $\mathbb{K}$

- low **complexity bound**
- low **execution time**

low memory usage, power consumption, ...

prime field $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$
field extension $\mathbb{F}_p[x]/\langle f(x) \rangle$
rational numbers $\mathbb{Q}$

measuring efficiency: algebraic complexity

efficient algorithms for polynomials, matrices, ...
with coefficients in some base field $\mathbb{K}$

- low **complexity bound**
- low **execution time**

low memory usage, power consumption, ...

prime field $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$
field extension $\mathbb{F}_p[x]/\langle f(x) \rangle$
rational numbers $\mathbb{Q}$

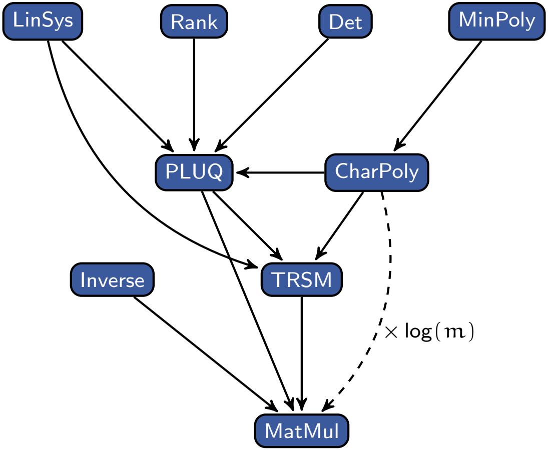
algebraic complexity bounds
$\rightsquigarrow$ count number of operations in $\mathbb{K}$

- 👍 standard complexity model for algebraic computations
- 👍 good predictor of practical performance for finite fields $\mathbb{K}$
- 👎 ignores coefficient growth, e.g. over $\mathbb{K} = \mathbb{Q}$

characteristic polynomial of a matrix

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

$\mathbb{K}$-linear algebra: reductions of most problems to matrix multiplication

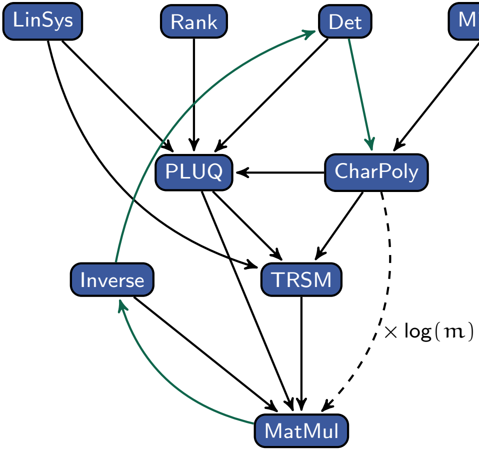


$$\left.\begin{array}{c}\text{LinSys}\\ \text{Det}\\ \text{Rank}\\ \text{PLUQ}\\ \text{TRSM}\\ \text{Inverse}\end{array}\right\} = O(\text{MatMul})$$

characteristic polynomial of a matrix

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

$\mathbb{K}$-linear algebra: reductions of most problems to matrix multiplication



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

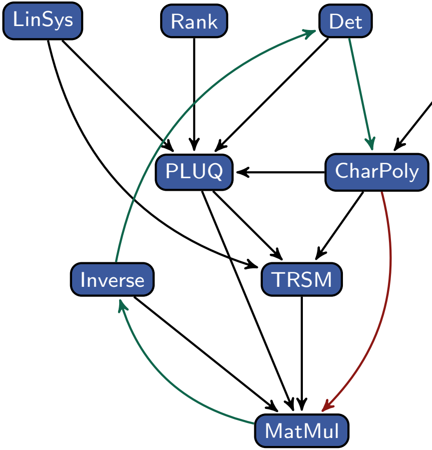
$$\text{MatMul} = O(\text{CharPoly})$$

[Baur-Strassen 1983]

characteristic polynomial of a matrix

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

$\mathbb{K}$-linear algebra: reductions of most problems to matrix multiplication



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

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

[Baur-Strassen 1983]

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

[Vincent Neiger & Clément Pernet, 2021]

deterministic algorithm with complexity $O(m^{\omega})$

- polynomial matrices
- partial triangularization
- ternary divide and conquer
- exploiting degree knowledge

characteristic polynomial in the time of matrix multiplication

[Vincent Neiger & Clément Pernet, 2021]

deterministic algorithm with complexity $O(m^\omega)$

- polynomial matrices
- partial triangularization
- ternary divide and conquer
- exploiting degree knowledge

characteristic polynomial in the time of matrix multiplication

summary of previous results

- 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]

[Vincent Neiger & Clément Pernet, 2021]
deterministic algorithm with complexity $O(m^\omega)$

- polynomial matrices
- partial triangularization
- ternary divide and conquer
- exploiting degree knowledge

characteristic polynomial in the time of matrix multiplication

framework for complexity — clarification is needed!

For any MatMul exponent ω feasible (as of today),
there is a MatMul algorithm in $O(m^{\omega-\varepsilon})$ for some $\varepsilon > 0$
$\Rightarrow$ the CharPoly algorithm of [Keller-Gehrig'85] is

- deterministic
- in $O(m^{\omega-\varepsilon}\log(m)) \subset O(m^\omega)$

not entirely satisfactory...

[Vincent Neiger & Clément Pernet, 2021]

deterministic algorithm with complexity $O(m^\omega)$

- polynomial matrices
- partial triangularization
- ternary divide and conquer
- exploiting degree knowledge

characteristic polynomial in the time of matrix multiplication

framework for complexity — classical requirements

matrix multiplication in $\mathbb{K}^{m\times m}$

- choose a MatMul algorithm in $O(m^\omega)$
- use this one for all MatMul instances

our requirement: $2 < \omega \leqslant 3$

we gladly accept $\omega = 2.1$, please provide the algorithm

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

polynomial multiplication in $\mathbb{K}[x]$

- choose a PolMul algorithm in $O(M(d))$
- use this one for all PolMul instances

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

$2M(d) \leqslant M(2d)$ $\quad M(d_1 d_2) \leqslant M(d_1)M(d_2)$

$M(d) \in O(d^{\omega-1-\varepsilon})$ for some $\varepsilon > 0$

polynomial matrices

$$\mathbf{A} = \begin{bmatrix} 3x+4 & x^3+4x+1 & 4x^2+3 \\ 5 & 5x^2+3x+1 & 5x+3 \\ 3x^3+x^2+5x+3 & 6x+5 & 2x+1 \end{bmatrix} \in \mathbb{K}[x]^{3\times 3}$$

3×3 matrix of degree 3
with entries in $\mathbb{K}[x] = \mathbb{F}_7[x]$

operations on $\mathbb{K}[x]^{m\times m}_{<d}$

- combination of matrix and polynomial computations
- addition in $O(m^2 d)$, naive multiplication in $O(m^3 d^2)$

[Cantor-Kaltofen'91]

multiplication in $O(m^\omega d \log(d) \;+\; m^2 d \log(d) \log\log(d))$

$\in O(m^\omega \mathsf{M}(d)) \subset O\tilde{~}(m^\omega d)$

polynomial matrices

$$\mathbf{A} = \begin{bmatrix} 3x+4 & x^3+4x+1 & 4x^2+3 \\ 5 & 5x^2+3x+1 & 5x+3 \\ 3x^3+x^2+5x+3 & 6x+5 & 2x+1 \end{bmatrix} \in \mathbb{K}[x]^{3\times 3}$$

3×3 matrix of degree 3
with entries in $\mathbb{K}[x] = \mathbb{F}_7[x]$

operations on $\mathbb{K}[x]^{m\times m}_{<d}$

- combination of matrix and polynomial computations
- addition in $O(m^2 d)$, naive multiplication in $O(m^3 d^2)$

[Cantor-Kaltofen'91]

multiplication in $O(m^\omega d \log(d) \; + \; m^2 d \log(d) \log\log(d))$

$\in O(m^\omega M(d)) \subset O\tilde{~}(m^\omega d)$

charpoly: matrix $x\mathbf{I}_m - \mathbf{M}$ is $m \times m$ of degree 1
$\rightarrow$ during algorithm: smaller size, larger degree

- some problems&techniques shared with matrices over $\mathbb{K}$
- some problems&techniques specific to entries in $\mathbb{K}[x]$

polynomial matrices: main computational problems

reductions of most problems to polynomial matrix multiplication

matrix $m \times m$ of degree d $\rightarrow$ $O\tilde{}(m^\omega d)$

of "average" degree $\frac{D}{m}$ $\rightarrow$ $O\tilde{}(m^\omega \frac{D}{m})$

classical matrix operations

- multiplication
- inversion $O\tilde{}(m^3 d)$
- kernel, system solving
- rank, determinant

univariate relations

- Hermite-Padé approximation
- vector rational interpolation
- syzygies, modular equations

transformation to normal forms

- triangularization: Hermite form
- row reduction: Popov form
- diagonalization: Smith form

outline

- context & result
  - computing with matrices over $\mathbb{K}$ and $\mathbb{K}[x]$
  - reductions to matrix multiplication
  - framework for complexity bounds
- previous work
- overview of the approach
- obstacles & spin-offs

outline

- context & result
  - computing with matrices over $\mathbb{K}$ and $\mathbb{K}[x]$
  - reductions to matrix multiplication
  - framework for complexity bounds
- previous work
  - based on matrices over $\mathbb{K}$
  - based on matrices over $\mathbb{K}[x]$
  - where do log factors come from?
- overview of the approach
- obstacles & spin-offs

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

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 CharPoly $\in \mathcal{NC}^2$

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 CharPoly $\in \mathcal{NC}^2$

determinant expansion $O(m^4)$

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

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 CharPoly $\in \mathcal{NC}^2$

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 randomized, requires large field $O(m^\omega)$

i.e. $\mathrm{card}(\mathbb{K}) \geqslant 2m^2$

charpoly via polynomial matrices

determinant of matrix $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$

charpoly via polynomial matrices

determinant of matrix $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$

evaluation-interpolation [folklore] $O(m^{\omega+1})$

at $\sim md$ points, requires large field

costs: for $\mathbf{A}$ of degree $d = 1$

charpoly via polynomial matrices

determinant of matrix $\mathbf{A} \in \mathbb{K}[x]^{m\times m}$

evaluation-interpolation [folklore] $O(m^{\omega+1})$

at $\sim md$ points, requires large field

costs: for $\mathbf{A}$ of degree $d = 1$

diagonalization [Storjohann 2003] $O(m^{\omega}\log(m)^2)$

Smith form: Las Vegas randomized, requires large field

charpoly via polynomial matrices

determinant of matrix $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$

evaluation-interpolation [folklore] $O(m^{\omega+1})$

at $\sim md$ points, requires large field

costs: for $\mathbf{A}$ of degree $d = 1$

diagonalization [Storjohann 2003] $O(m^{\omega} \log(m)^2)$

Smith form: Las Vegas randomized, requires large field

partial triangularization

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

- 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 $\begin{pmatrix} v & \mathbf{M}v & \cdots & \mathbf{M}^m v \end{pmatrix}$ $\rightarrow$ $\log(m)$

in $\mathbb{K}$-linear algebra

sources of log factors

for polynomial matrices

- 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 $\begin{pmatrix} v & \mathbf{M}v & \cdots & \mathbf{M}^m v \end{pmatrix}$ $\rightarrow \log(m)$

in $\mathbb{K}$-linear algebra

sources of log factors

for polynomial matrices

- 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 on constant matrices e.g. [Jeannerod-Neiger-Schost-Villard'17]
  since base cases of recursions on degree $=$ matrices over $\mathbb{K}$
  typically adds $O(m^\omega d \log(m))$ to the cost, non-negligible when $d = O(1)$
- looking for a matrix with unpredictable, unbalanced degrees
  $\log(m)$ steps in dimension $m \times m$, to uncover the degree profile [Zhou-Labahn'13]
  reminiscent of obstacles in the derandomization of [P.-Storjohann'07]

outline

- **context & result**
  - computing with matrices over $\mathbb{K}$ and $\mathbb{K}[x]$
  - reductions to matrix multiplication
  - framework for complexity bounds
- **previous work**
  - based on matrices over $\mathbb{K}$
  - based on matrices over $\mathbb{K}[x]$
  - where do log factors come from?
- **overview of the approach**
- **obstacles & spin-offs**

outline

- **context & result**
  - computing with matrices over $\mathbb{K}$ and $\mathbb{K}[x]$
  - reductions to matrix multiplication
  - framework for complexity bounds
- **previous work**
  - based on matrices over $\mathbb{K}$
  - based on matrices over $\mathbb{K}[x]$
  - where do log factors come from?
- **overview of the approach**
  - determinant via partial triangularization
  - overview of the new recursive approach
  - complexity of this ternary recursion
- **obstacles & spin-offs**

partial block triangularization

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

triangularization of $m \times m$ matrix $\mathbf{A}$ using $\frac{m}{2} \times \frac{m}{2}$ 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, Neiger-Labahn-Zhou 2017]

triangularization of $m \times m$ matrix $\mathbf{A}$ using $\frac{m}{2} \times \frac{m}{2}$ 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 $\left[\begin{smallmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{smallmatrix}\right]$ $\qquad$ $\mathbf{K}_1\mathbf{A}_2 + \mathbf{K}_2\mathbf{A}_4$ $\qquad$ row basis of $\left[\begin{smallmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{smallmatrix}\right]$

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

generic input $\Rightarrow \det(\mathbf{A})$ without $\log(m)$ [Giorgi-Jeannerod-Villard'03]

$\mathbf{A}_1$ 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 $\qquad O(m^\omega \mathsf{M}(d)\log(d))$
- 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

complexity $\mathcal{C}(m, d) = \mathcal{C}(\frac{m}{2}, 2d) + O(m^\omega \mathsf{M}(d)\log(d))$

partial block triangularization

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

triangularization of $m \times m$ matrix $\mathbf{A}$ using $\frac{m}{2} \times \frac{m}{2}$ 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: $\det(\mathbf{A}) = \det(\mathbf{R})\det(\mathbf{B})$

general input $\Rightarrow$ $\det(\mathbf{A})$ with $\log(m)$ [Labahn-Neiger-Zhou'17]

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 $\frac{D}{m}$

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

partial block triangularization

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

triangularization of $m \times m$ matrix $\mathbf{A}$ using $\frac{m}{2} \times \frac{m}{2}$ 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 $\left[\begin{smallmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{smallmatrix}\right]$ $\qquad$ $\mathbf{K}_1\mathbf{A}_2 + \mathbf{K}_2\mathbf{A}_4$ $\qquad$ row basis of $\left[\begin{smallmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{smallmatrix}\right]$

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

be lazy: if hard to compute, don't compute [Neiger-P.'21]

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 (consequences of laziness)

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

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

further obstacles (consequences of 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 (consequences of 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

further obstacles (consequences of 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

solution: exploit degree knowledge to accelerate transformations

s-reduced $\Rightarrow$ s-weak Popov $\Rightarrow$ s-Popov

ternary recursion & complexity analysis

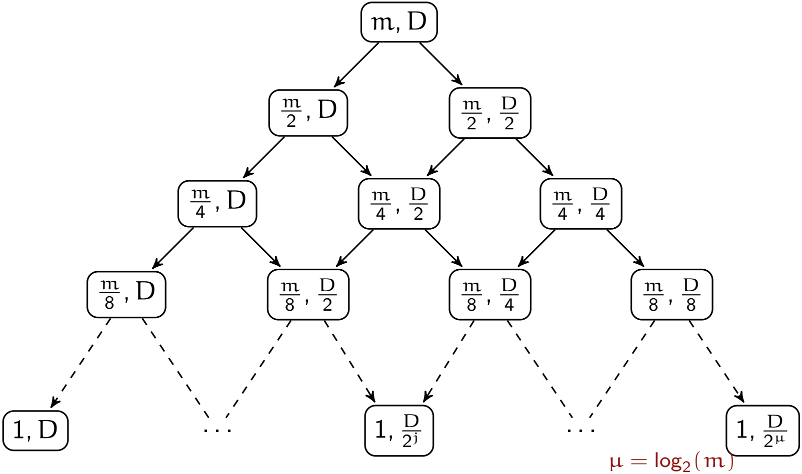
determinant of $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ of average row degree $\frac{D}{m} = \frac{\mathsf{degdet}}{m}$

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

ternary recursion & complexity analysis

determinant of $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ of average row degree $\frac{D}{m} = \frac{\mathrm{degdet}}{m}$

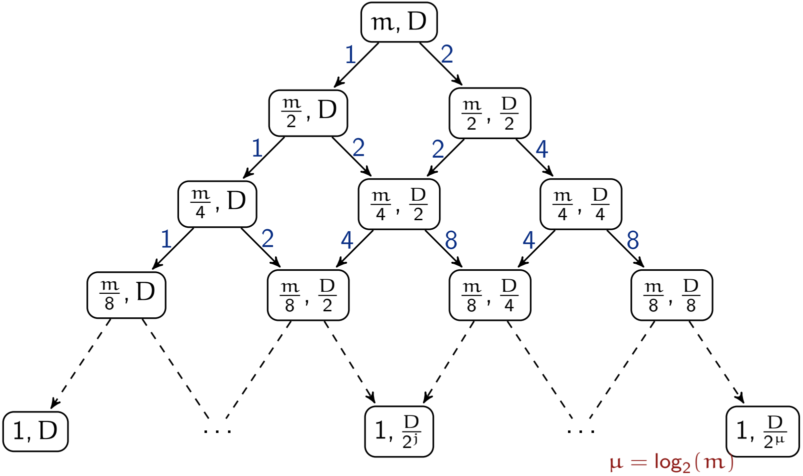
$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}(\tfrac{m}{2}, \tfrac{D}{2}) + \mathcal{C}(\tfrac{m}{2}, D) + O(m^{\omega} M'(\tfrac{D}{m}))$$



ternary recursion & complexity analysis

determinant of $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ of average row degree $\frac{D}{m} = \frac{\mathrm{degdet}}{m}$

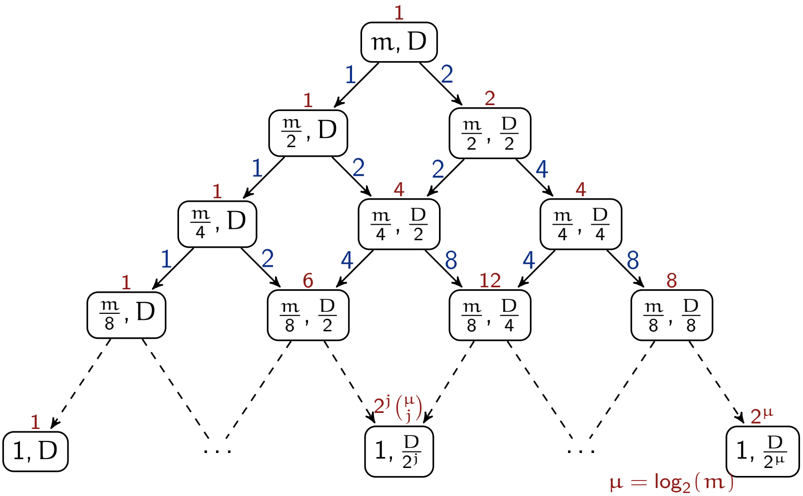
$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}(\tfrac{m}{2}, \tfrac{D}{2}) + \mathcal{C}(\tfrac{m}{2}, D) + O(m^{\omega} M'(\tfrac{D}{m}))$$



ternary recursion & complexity analysis

determinant of $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ of average row degree $\frac{D}{m} = \frac{\text{degdet}}{m}$

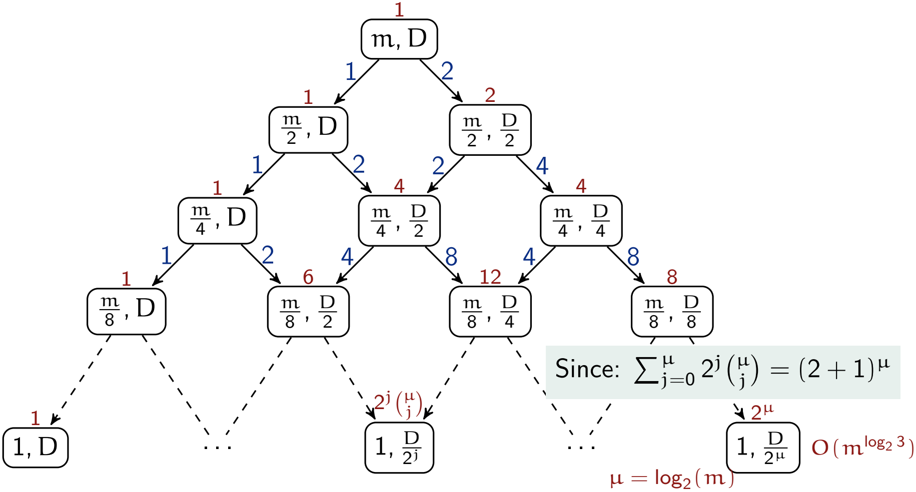
$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}(\tfrac{m}{2}, \tfrac{D}{2}) + \mathcal{C}(\tfrac{m}{2}, D) + O(m^{\omega} M'(\tfrac{D}{m}))$$



ternary recursion & complexity analysis

determinant of $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ of average row degree $\frac{D}{m} = \frac{\mathrm{degdet}}{m}$

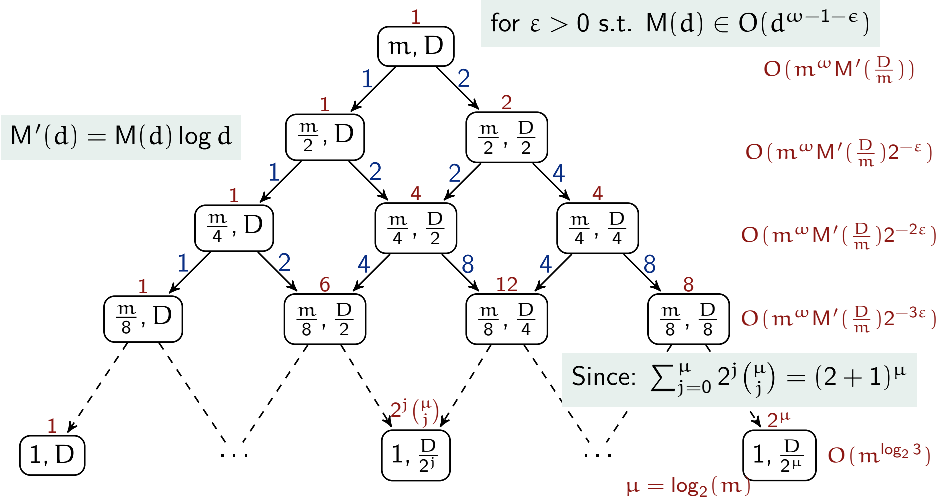
$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}(\tfrac{m}{2}, \tfrac{D}{2}) + \mathcal{C}(\tfrac{m}{2}, D) + O(m^{\omega} M'(\tfrac{D}{m}))$$



ternary recursion & complexity analysis

determinant of $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ of average row degree $\frac{D}{m} = \frac{\text{degdet}}{m}$

$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}(\tfrac{m}{2}, \tfrac{D}{2}) + \mathcal{C}(\tfrac{m}{2}, D) + O(m^{\omega} M'(\tfrac{D}{m}))$$



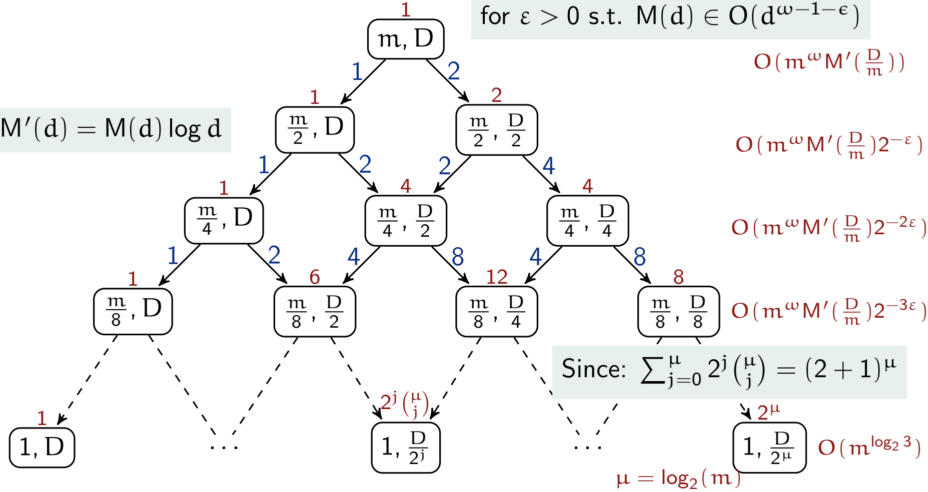
ternary recursion & complexity analysis

determinant of $\mathbf{A} \in \mathbb{K}[x]^{m \times m}$ of average row degree $\frac{D}{m} = \frac{\text{degdet}}{m}$

$$\mathcal{C}(m, D) \leqslant 2\mathcal{C}(\tfrac{m}{2}, \tfrac{D}{2}) + \mathcal{C}(\tfrac{m}{2}, D) + O(m^\omega M'(\tfrac{D}{m})) \in O(m^\omega M'(\tfrac{D}{m}))$$

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

$M'(d) = M(d) \log d$



Since: $\sum_{j=0}^{\mu} 2^j \binom{\mu}{j} = (2+1)^\mu$

outline

- **context & result**
  - computing with matrices over $\mathbb{K}$ and $\mathbb{K}[x]$
  - reductions to matrix multiplication
  - framework for complexity bounds
- **previous work**
  - based on matrices over $\mathbb{K}$
  - based on matrices over $\mathbb{K}[x]$
  - where do log factors come from?
- **overview of the approach**
  - determinant via partial triangularization
  - overview of the new recursive approach
  - complexity of this ternary recursion
- **obstacles & spin-offs**

outline

context & result

- computing with matrices over $\mathbb{K}$ and $\mathbb{K}[x]$
- reductions to matrix multiplication
- framework for complexity bounds

previous work

- based on matrices over $\mathbb{K}$
- based on matrices over $\mathbb{K}[x]$
- where do log factors come from?

overview of the approach

- determinant via partial triangularization
- overview of the new recursive approach
- complexity of this ternary recursion

obstacles & spin-offs

- main obstacles and solutions
- spin-off results on shifted forms
- summary and perspectives

Hermite and Popov forms

working over $\mathbb{K} = \mathbb{Z}/7\mathbb{Z}$

$$\mathbf{A} = \begin{bmatrix} 3x+4 & x^3+4x+1 & 4x^2+3 \\ 5 & 5x^2+3x+1 & 5x+3 \\ 3x^3+x^2+5x+3 & 6x+5 & 2x+1 \end{bmatrix}$$

using elementary row operations, transform $\mathbf{A}$ into...

Hermite form $\quad \mathbf{H} = \begin{bmatrix} x^6+6x^4+x^3+x+4 & 0 & 0 \\ 5x^5+5x^4+6x^3+2x^2+6x+3 & x & 0 \\ 3x^4+5x^3+4x^2+6x+1 & 5 & 1 \end{bmatrix}$

Popov form $\quad \mathbf{P} = \begin{bmatrix} x^3+5x^2+4x+1 & 2x+4 & 3x+5 \\ 1 & x^2+2x+3 & x+2 \\ 3x+2 & 4x & x^2 \end{bmatrix}$

Hermite and Popov forms

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

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

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

$$\mathbf{A} \xrightarrow{\text{elementary row transformations}} \text{Hermite form, Popov form}$$

Hermite form [Hermite, 1851]

- triangular
- column normalized

Popov form [Popov, 1972]

- row reduced/distinct pivots
- column normalized

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

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

Hermite and Popov forms

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

elementary row transformations

Hermite form [Hermite, 1851]
- triangular
- column normalized

Popov form [Popov, 1972]
- row reduced/distinct pivots
- column normalized

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

$\preccurlyeq_{\mathrm{pot}}$ reduced Gröbner basis $\preccurlyeq_{\mathrm{top}}$

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

Hermite and Popov forms

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

elementary row transformations

Hermite form [Hermite, 1851]
- triangular
- column normalized

Popov form [Popov, 1972]
- row reduced/distinct pivots
- column normalized

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

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

- average column degree is $\frac{D}{m}$
- size of object is $mD + m^2 = m^2(\frac{D}{m} + 1)$

Hermite and Popov forms

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

elementary row transformations

Hermite form [Hermite, 1851]
- triangular
- column normalized

Popov form [Popov, 1972]
- row reduced/distinct pivots
- column normalized

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

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

[Beckermann-Labahn-Villard, 1999; Mulders-Storjohann, 2003]

weak Popov form = **not** column normalized

= minimal, non-reduced, t.o.p. Gröbner basis

shifted forms

shift: integer tuple $\mathbf{s} = (s_1, \ldots, s_m)$ acting as column weights

$\rightarrow$ connects Popov and Hermite forms

$\mathbf{s} = (0,0,0,0)$ Popov	$\begin{bmatrix} \mathbf{4} & 3 & 3 & 3 \\ 3 & \mathbf{4} & 3 & 3 \\ 3 & 3 & \mathbf{4} & 3 \\ 3 & 3 & 3 & \mathbf{4} \end{bmatrix}$	$\begin{bmatrix} \mathbf{7} & 0 & 1 & 5 \\ 0 & \mathbf{1} & & 0 \\ & & \mathbf{2} & \\ 6 & 0 & 1 & \mathbf{6} \end{bmatrix}$
$\mathbf{s} = (0,2,4,6)$ $\mathbf{s}$-Popov	$\begin{bmatrix} \mathbf{7} & 4 & 2 & 0 \\ 6 & \mathbf{5} & 2 & 0 \\ 6 & 4 & \mathbf{3} & 0 \\ 6 & 4 & 2 & \mathbf{1} \end{bmatrix}$	$\begin{bmatrix} \mathbf{8} & 5 & 1 & \\ 7 & \mathbf{6} & 1 & \\ & & \mathbf{2} & \\ 0 & 1 & & \mathbf{0} \end{bmatrix}$
$\mathbf{s} = (0, D, 2D, 3D)$ Hermite	$\begin{bmatrix} \mathbf{16} & & & \\ 15 & \mathbf{0} & & \\ 15 & & \mathbf{0} & \\ 15 & & & \mathbf{0} \end{bmatrix}$	$\begin{bmatrix} \mathbf{4} & & & \\ 3 & \mathbf{7} & & \\ 1 & 5 & \mathbf{3} & \\ 3 & 6 & 1 & \mathbf{2} \end{bmatrix}$

- normal form, average column degree D/m
- shifts arise naturally in algorithms (approximants, kernel, ...)
- they allow one to specify non-uniform degree constraints

back to obstacles: easy ones

recall: $\mathbf{A} = \left[\begin{smallmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{smallmatrix}\right]$ in weak Popov form, we want:

- $\mathbf{A}_1$ nonsingular: ok by definition

(principal submatrices of $\mathbf{A}$ are weak Popov $\Rightarrow$ are nonsingular)

- $\sum \mathrm{rdeg}(\mathbf{A}_1) \leqslant \mathrm{D}/2$: either ok for $\mathbf{A}$, or ok for $\left[\begin{smallmatrix} \mathbf{A}_4 & \mathbf{A}_3 \\ \mathbf{A}_2 & \mathbf{A}_1 \end{smallmatrix}\right]$

(*almost* weak Popov... easily transformed into it, with same determinant)

back to 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

(principal submatrices of $\mathbf{A}$ are weak Popov $\Rightarrow$ are nonsingular)

- $\sum \mathrm{rdeg}(\mathbf{A}_1) \leqslant \mathrm{D}/2$: 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 $\mathrm{O}(\mathrm{m}^\omega \mathrm{M}'(\frac{\mathrm{D}}{\mathrm{m}}))$, $\quad \sum \mathrm{rdeg}(\mathbf{K}_2) \leqslant \mathrm{D}/2$, $\quad \mathbf{K}_2$ in $\mathbf{s}$-weak Popov form

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

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

- *towards correctness:* $\mathbf{B} = [\mathbf{K}_1 \;\; \mathbf{K}_2]\begin{bmatrix} \mathbf{A}_2 \\ \mathbf{A}_4 \end{bmatrix}$ 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 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

(principal submatrices of $\mathbf{A}$ are weak Popov $\Rightarrow$ are nonsingular)

- $\sum \mathrm{rdeg}(\mathbf{A}_1) \leqslant \mathrm{D}/2$: 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 $\mathrm{O}(\mathrm{m}^\omega \mathrm{M}'(\frac{\mathrm{D}}{\mathrm{m}}))$, $\quad \sum \mathrm{rdeg}(\mathbf{K}_2) \leqslant \mathrm{D}/2$, $\quad \mathbf{K}_2$ in $\boldsymbol{s}$-weak Popov form

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

using the shift $\mathrm{rdeg}(\mathbf{A})$ (and $\boldsymbol{s}$) has **crucial advantages**:

- *towards correctness:* $\mathbf{B} = [\mathbf{K}_1 \;\; \mathbf{K}_2]\begin{bmatrix} \mathbf{A}_2 \\ \mathbf{A}_4 \end{bmatrix}$ 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

input: $\mathbf{K}_2$ in $\mathbf{s}$-weak Popov form, with $\mathbf{s} \geqslant 0$
output: $\mathbf{P}$ in $\mathbf{0}$-weak Popov form, with $\det(\mathbf{P}) = \det(\mathbf{K}_2)$

approaching the main obstacle

input: $\mathbf{K}_2$ in $\mathbf{s}$-weak Popov form, with $\mathbf{s} \geqslant 0$
output: $\mathbf{P}$ in $\mathbf{0}$-weak Popov form, with $\det(\mathbf{P}) = \det(\mathbf{K}_2)$

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-Neiger-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 $\mathbf{0}$-weak Popov by construction, and $\det(\mathbf{P}^\mathsf{T}) = \det(\mathbf{P})$

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

approaching the main obstacle

input: $\mathbf{K}_2$ in $\mathbf{s}$-weak Popov form, with $\mathbf{s} \geqslant 0$
output: $\mathbf{P}$ in $\mathbf{0}$-weak Popov form, with $\det(\mathbf{P}) = \det(\mathbf{K}_2)$

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-Neiger-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 $\mathbf{0}$-weak Popov by construction, and $\det(\mathbf{P}^\mathsf{T}) = \det(\mathbf{P})$

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

Fact: $\mathbf{K}_2^\mathsf{T}$ is $-\mathbf{t}$-weak Popov, for some $-\mathbf{t} \leqslant 0$

- $\mathbf{t} = \mathrm{rdeg}_{\mathbf{s}}(\mathbf{K}_2) = \mathbf{s} + \boldsymbol{\delta} \geqslant 0$
- ignoring some row/column permutations for simplicity

approaching the main obstacle

input: $\mathbf{K}_2$ in $\mathbf{s}$-weak Popov form, with $\mathbf{s} \geqslant 0$
output: $\mathbf{P}$ in $\mathbf{0}$-weak Popov form, with $\det(\mathbf{P}) = \det(\mathbf{K}_2)$

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-Neiger-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 $\mathbf{0}$-weak Popov by construction, and $\det(\mathbf{P}^\mathsf{T}) = \det(\mathbf{P})$

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

Fact: $\mathbf{K}_2^\mathsf{T}$ is $-\mathbf{t}$-weak Popov, for some $-\mathbf{t} \leqslant 0$

- $\mathbf{t} = \mathrm{rdeg}_{\mathbf{s}}(\mathbf{K}_2) = \mathbf{s} + \boldsymbol{\delta} \geqslant 0$
- ignoring some row/column permutations for simplicity

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

input: $\mathbf{K}_2$ in $\mathbf{s}$-weak Popov form, with $\mathbf{s} \geqslant 0$
output: $\mathbf{P}$ in $\mathbf{0}$-weak Popov form, with $\det(\mathbf{P}) = \det(\mathbf{K}_2)$

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-Neiger-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 $\mathbf{0}$-weak Popov by construction, and $\det(\mathbf{P}^\mathsf{T}) = \det(\mathbf{P})$
amounts to a change of shift from $\mathbf{s}$ to $-\boldsymbol{\delta} \leqslant 0$ [Neiger'16] $\Rightarrow$ same issue

Fact: $\mathbf{K}_2^\mathsf{T}$ is $-\mathbf{t}$-weak Popov, for some $-\mathbf{t} \leqslant 0$

- $\mathbf{t} = \mathrm{rdeg}_{\mathbf{s}}(\mathbf{K}_2) = \mathbf{s} + \boldsymbol{\delta} \geqslant 0$
- ignoring some row/column permutations for simplicity

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-offs: faster transformations of shifted forms

weak Popov → Popov

Input:	$\mathbf{s} \in \mathbb{Z}^m$, a shift,
	$\mathbf{A} \in \mathbb{K}[x]^{m \times m}$, a matrix in $\mathbf{s}$-weak Popov form
Output:	the $\mathbf{s}$-Popov form of $\mathbf{A}$
Requirement:	$-\mathbf{s} \geqslant \mathrm{DiagonalDegrees}(\mathbf{A})$
Complexity:	$O(m^\omega \mathsf{M}(\frac{D}{m}) \log(\frac{D}{m}))$, where $D = \sum \mathbf{s}$

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 known output degrees
- introduction of partial linearization techniques for kernel bases

spin-offs: faster transformations of shifted forms

weak Popov → Popov

Input:	$\mathbf{s} \in \mathbb{Z}^m$, a shift,
	$\mathbf{A} \in \mathbb{K}[x]^{m \times m}$, a matrix in $\mathbf{s}$-weak Popov form
Output:	the $\mathbf{s}$-Popov form of $\mathbf{A}$
Requirement:	$-\mathbf{s} \geqslant \mathrm{DiagonalDegrees}(\mathbf{A})$
Complexity:	$O(m^\omega M(\frac{D}{m}) \log(\frac{D}{m}))$, where $D = \sum \mathbf{s}$

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 known output degrees
- introduction of partial linearization techniques for kernel bases

reduced → weak Popov

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

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

open questions: Frobenius and Smith forms

deterministic, log-free Frobenius form

$$\begin{bmatrix} & & \\ & \mathbf{M} & \\ & & \end{bmatrix} \longrightarrow \begin{bmatrix} \mathbf{C}_{s_1} & & & \\ & \mathbf{C}_{s_2} & & \\ & & \ddots & \\ & & & \mathbf{C}_{s_m} \end{bmatrix}$$

s_{i+1} divides s_i

- complexity $O(m^\omega)$ [P.-Storjohann'07]
- Las Vegas, requires large field
- exploit the new CharPoly techniques?

deterministic algo in $O(m^\omega)$?

open questions: Frobenius and Smith forms

deterministic, log-free Frobenius form

$$\begin{bmatrix} & & \\ & \mathbf{M} & \\ & & \end{bmatrix} \longrightarrow \begin{bmatrix} \mathbf{C}_{s_1} & & & \\ & \mathbf{C}_{s_2} & & \\ & & \ddots & \\ & & & \mathbf{C}_{s_m} \end{bmatrix}$$

s_{i+1} divides s_i

- complexity $O(m^\omega)$ [P.-Storjohann'07]
- Las Vegas, requires large field
- exploit the new CharPoly techniques?

deterministic algo in $O(m^\omega)$?

deterministic Smith form

$$\begin{bmatrix} & & \\ & \mathbf{A} & \\ & & \end{bmatrix} \longrightarrow \begin{bmatrix} s_1 & & & \\ & s_2 & & \\ & & \ddots & \\ & & & s_m \end{bmatrix}$$

s_{i+1} divides s_i

- complexity $O\tilde{~}(m^\omega \frac{D}{m})$ [Storjohann'03]
- Las Vegas, requires large field
- exploit progress on $\mathbb{K}[x]$-matrices?

deterministic algo in $O\tilde{~}(m^\omega \frac{D}{m})$?

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

summary

conclusion

perspectives

- efficient implementation and study of practical performance
  small fields, degenerate instances, ...
- alternative approach by exploiting a quasiseparable structure
  closer to the linear algebra approach in [P.-Storjohann 2007]
- Frobenius normal form & Smith normal form