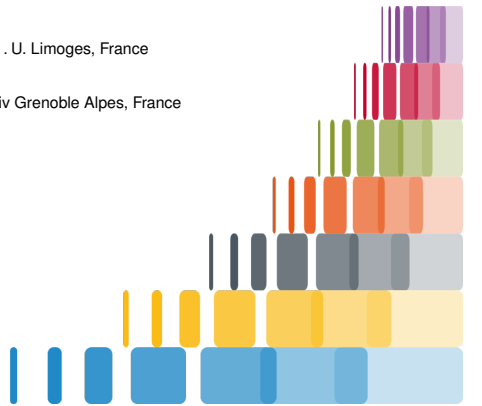
Deterministic computation of the characteristic polynomial in the time of matrix multiplication

Vincent Neiger ..... U. Limoges, France

Clément Pernet ..... Univ Grenoble Alpes, France

Journées Nationales de Calcul Formel 2021

Luminy, France (online), March 2, 2021



xlim INSTITUT DE RECHERCHE

Université de Limoges

UGA Université Grenoble Alpes

Outline

- Context, problem, state of the art
- Obstacles, overview of the approach
- Complexity, Spin-off results

Outline

- Context, problem, state of the art
- Obstacles, overview of the approach
- Complexity, Spin-off results

Context

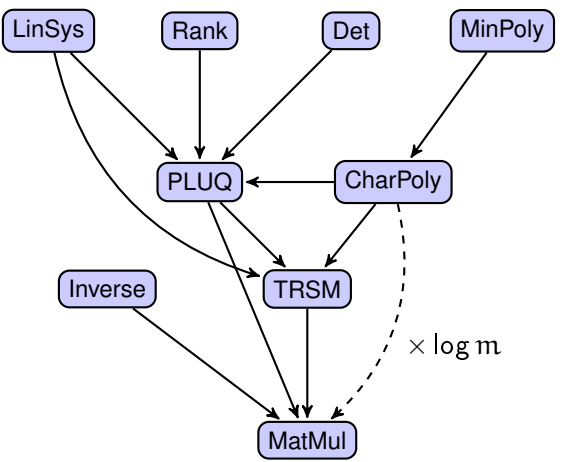
- field $\mathbb{K}$, algebraic complexity (counting operations in $\mathbb{K}$)
- ω: exponent of MatMul over $\mathbb{K}$: $m \times m$ by $m \times m$ in $O(m^\omega)$

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

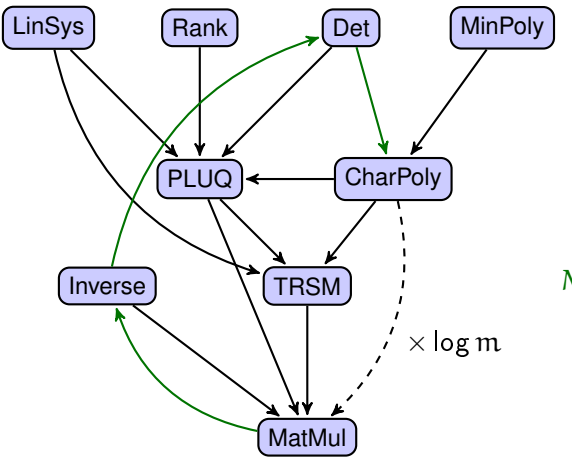


$$\left.\begin{array}{c} LinSys \\ Det \\ Rank \\ PLUQ \\ TRSM \\ Inverse \end{array}\right\} = O(MatMul)$$

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



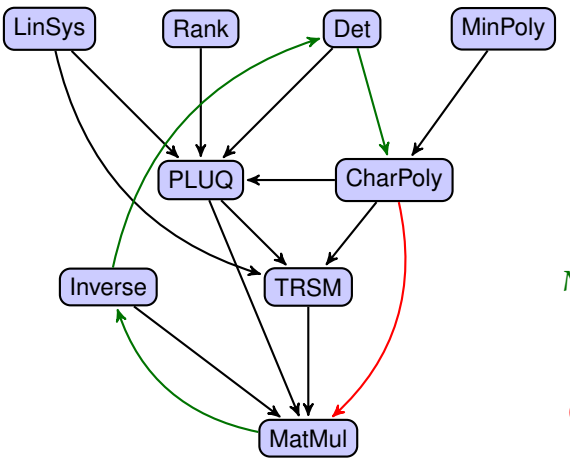
$$\left.\begin{matrix} \mathrm{LinSys} \\ \mathrm{Det} \\ \mathrm{Rank} \\ \mathrm{PLUQ} \\ \mathrm{TRSM} \\ \mathrm{Inverse} \end{matrix}\right\} = O(\mathrm{MatMul})$$

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

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



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

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

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

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

Motivation

Remark

In theory, for all ω, $\mathrm{CharPoly} = \mathrm{O}(\mathrm{m}^\omega)$ *is achieved by running*

- *any* $\mathrm{O}(\mathrm{m}^\omega \log \mathrm{m})$ *algorithm (like Keller-Gehrig's)*
- *using an* $\mathrm{O}(\mathrm{m}^{\omega-\varepsilon})$ *MatMul*

Motivation

Remark

In theory, for all ω, $\mathrm{CharPoly} = \mathrm{O}(\mathrm{m}^\omega)$ *is achieved by running*

- *any* $\mathrm{O}(\mathrm{m}^\omega \log \mathrm{m})$ *algorithm (like Keller-Gehrig's)*
- *using an* $\mathrm{O}(\mathrm{m}^{\omega-\varepsilon})$ *MatMul*

More relevant model

Rely on a given MatMul algorithm, and reach the same exponent

- No cheating w.r.t. the presence of $\log$
- Underlying hard question:

> *"How to compute an object related to a Krylov iteration without* $\log$ *?"*

Motivation

Remark

In theory, for all ω, $\mathrm{CharPoly} = \mathrm{O}(\mathrm{m}^\omega)$ *is achieved by running*

- *any* $\mathrm{O}(\mathrm{m}^\omega \log \mathrm{m})$ *algorithm (like Keller-Gehrig's)*
- *using an* $\mathrm{O}(\mathrm{m}^{\omega-\varepsilon})$ *MatMul*

More relevant model

Rely on a given MatMul algorithm, and reach the same exponent

- No cheating w.r.t. the presence of $\log$
- Underlying hard question:

> *"How to compute an object related to a Krylov iteration without* $\log$ *?"*

In practice:

- Only a few subcubic time MatMul are available (mainly Strassen's)
- [P. Storjohann'07] randomized algorithm inefficient with small fields

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

Traces of Powers: [LeVerrier1840] [Faddeev'49, Souriau'48, ...], $O(m^4)$ or $O(m^{\omega+1})$
used by [Csanky'75] to prove $\mathcal{NC}^2$ membership.

Determinant expansion: [Samuelson'42, Berkowitz'84] $O(m^4)$
suited for division free algorithms with later developments in
[Abdlejaoued and Malaschonok'01, Kaltofen and Villard'05]

Krylov methods: [Danilevskij'37,Keller-Gehrig'85, P. and 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 linear algebra over $\mathbb{K}[x]$

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

Evaluation-Interpolation: [folklore] $\quad O(m^{\omega+1})$

Diagonalization: Smith Form [Storjohann 2003] $\quad O(m^{\omega} \log(m)^2)$

Las Vegas randomized + additional logs for small fields

Triangularization:

- Generic: [Giorgi-Jeannerod-Villard 2003]: $\quad O(m^{\omega})$
  diagonal of Hermite form must be $1, \ldots, 1, \det(\mathbf{A})$
- [Neiger-Labahn-Zhou 2017] via triangularization: $\quad \tilde{O}(m^{\omega})$
  logarithmic factors in m and d

For polynomials of degree $\leqslant d$ in $\mathbb{K}[x]$:

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

Where do the log come from?

Explicit Krylov iteration: to compute a Krylov basis

$$\begin{pmatrix} v & Av & \dots & A^{m-1}v \end{pmatrix}$$

$\rightarrow$ Fast matrix exponentiation: $\log m \times O(m^{\omega})$

Where do the log come from?

Explicit Krylov iteration: to compute a Krylov basis

$$\begin{pmatrix} v & Av & \dots & A^{m-1}v \end{pmatrix}$$

$\rightarrow$ Fast matrix exponentiation: $\log m \times O(m^\omega)$

Polynomial matrix determinants: If Divide and conquer is applied

- on matrix dimension $\rightarrow$ no $\log(m)$
- on degree (Polynomial arithmetic) $\rightarrow$ no $\log(m)$
- on both dimension and degrees:
  $\rightarrow$ manageable if the total degree remains controlled.
  Hard case: long Krylov chains with small dimension drop
  $\rightarrow$ reminiscent of the failure to de-randomize [P. Storjohann'07]

Outline

- Context, problem, state of the art
- Obstacles, overview of the approach
- Complexity, Spin-off results

Partial block triangularization

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

Triangularization of $\mathrm{m} \times \mathrm{m}$ matrix $\mathbf{A}$ using $\mathrm{m}/2 \times \mathrm{m}/2$ 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 $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$

Partial block triangularization

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

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

not computed

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

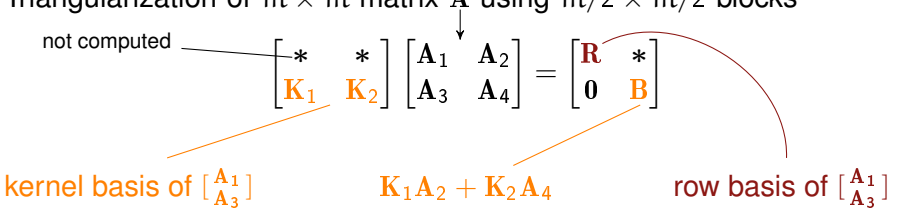
kernel basis of $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$ $\quad \mathbf{K}_1\mathbf{A}_2 + \mathbf{K}_2\mathbf{A}_4$ $\quad$ row basis of $\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_3 \end{bmatrix}$

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

Generic case without log factor

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

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



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 $\quad O(m^\omega M'(d))$
- Recursively, compute $\det(\mathbf{B})$, return it

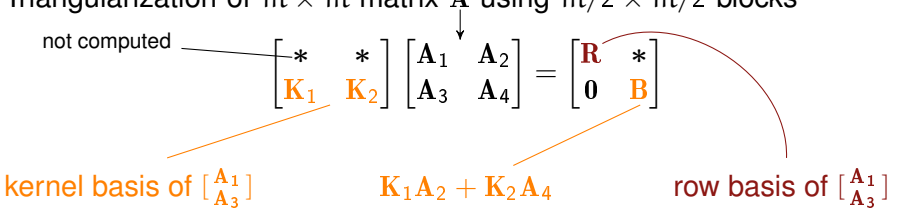
$\mathbf{A}$ and $[\mathbf{K}_1 \ \ \mathbf{K}_2]$ have degree $d \Rightarrow \mathbf{B}$ has degree $2d$: controlled total degree

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

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

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



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-Neiger-Zhou 2017]

- Compute kernel $[\mathbf{K}_1 \ \ \mathbf{K}_2]$; deduce $\mathbf{B}$ by MatMul $O(m^\omega M'(\frac{D}{m}))$
- Compute row basis $\mathbf{R}$ $\tilde{O}(m^\omega \frac{D}{m})$ 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

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

Triangularization of $\mathrm{m} \times \mathrm{m}$ matrix $\mathbf{A}$ using $\mathrm{m}/2 \times \mathrm{m}/2$ 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: $\det(\mathbf{A}) = \det(\mathbf{R}) \det(\mathbf{B})$

Obstacle: removing log factors in row basis computation

$\Rightarrow$ solution: remove row basis computation

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

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 ⇒ suitable for recursive calls
- 👎 $\mathbf{K}_2$ is in "shifted reduced" form… find weak Popov $\mathbf{P}$ with same determinant

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 $\rightsquigarrow$ $\mathbf{P}$ = transpose of the $-\mathrm{rdeg}_{\mathrm{rdeg}(\mathbf{A}_4)}(\mathbf{K}_2)$-Popov form of $\mathbf{K}_2^{\mathsf{T}}$

Outline

- Context, problem, state of the art
- Obstacles, overview of the approach
- Complexity, Spin-off results

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

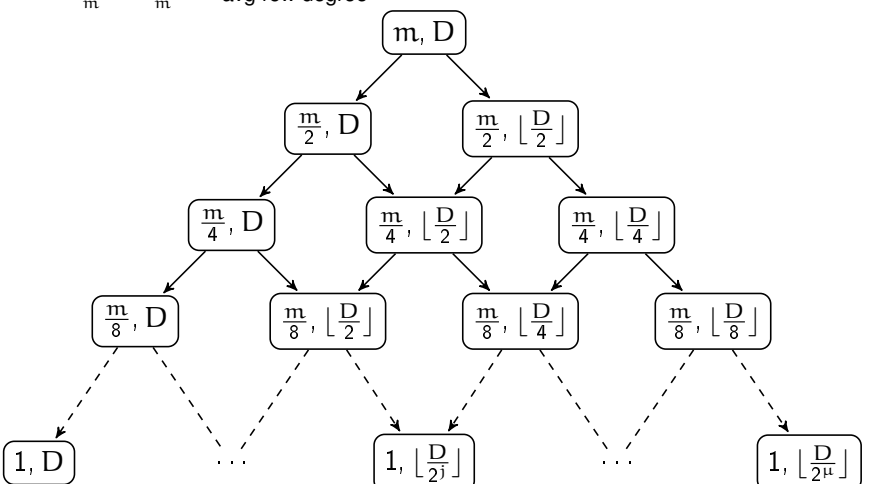
$\frac{D}{m} = \frac{\deg \det}{m} =$ avg row degree

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

$\frac{D}{m} = \frac{\deg \det}{m}$ = avg row degree

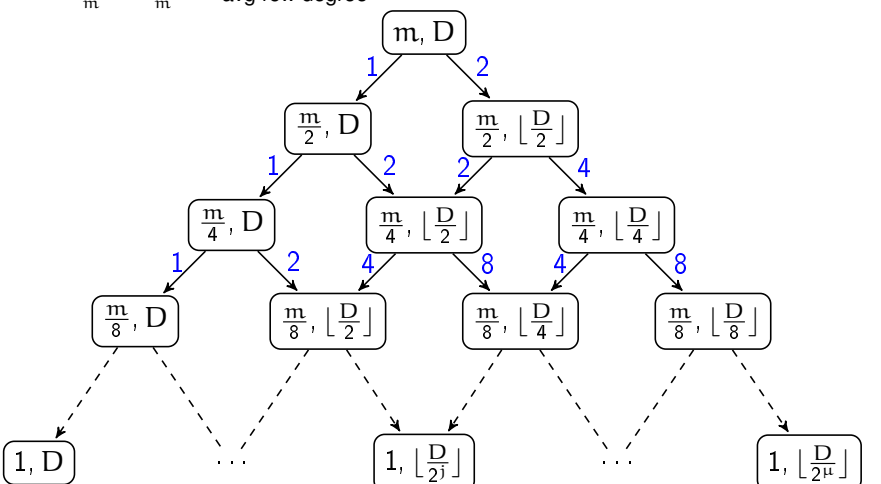


Complexity

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

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

$\frac{D}{m} = \frac{\deg \det}{m}$ = avg row degree

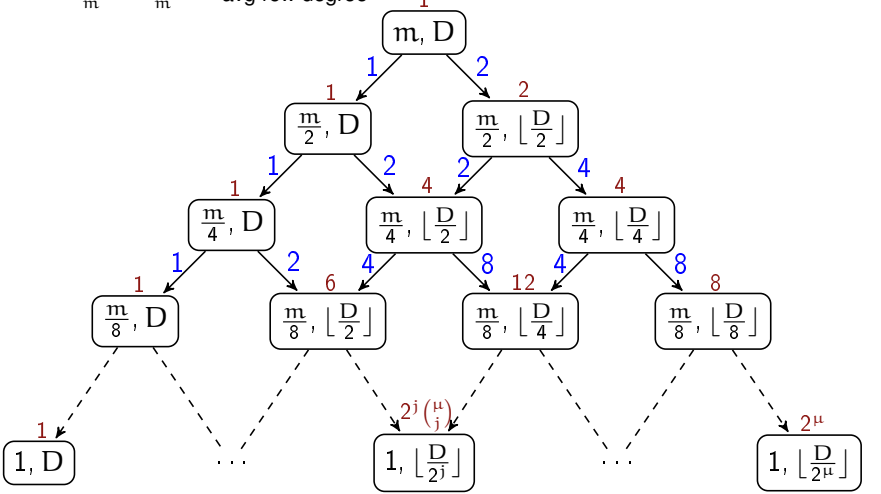


Complexity

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

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

$\frac{D}{m} = \frac{\deg \det}{m}$ = avg row degree

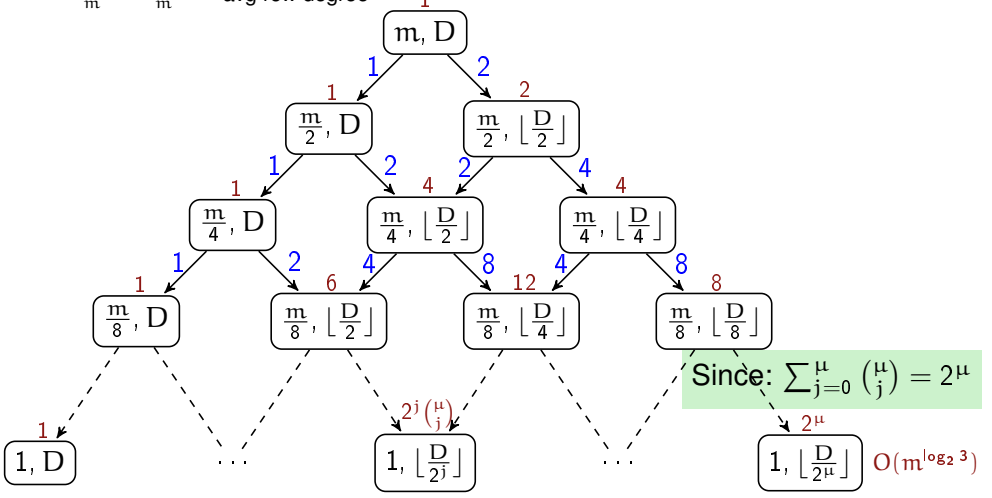


Complexity

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

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

$\frac{D}{m} = \frac{\deg \det}{m}$ = avg row degree



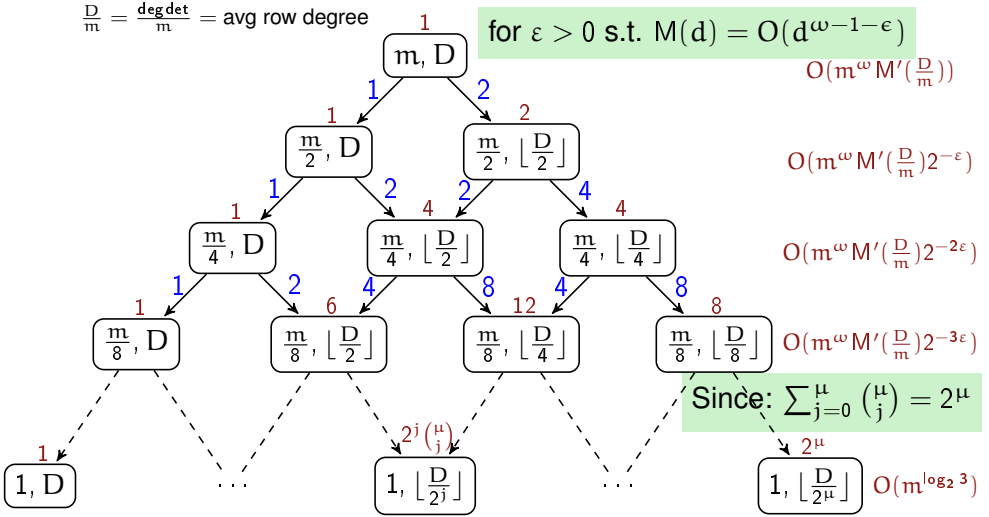
Since: $\sum_{j=0}^{\mu} \binom{\mu}{j} = 2^{\mu}$

Complexity

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

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

$\frac{D}{m} = \frac{\deg \det}{m}$ = avg row degree



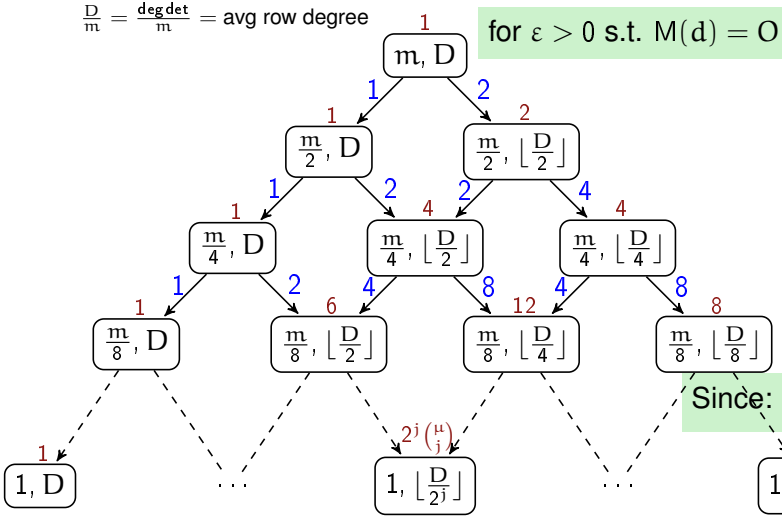
Complexity

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

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

$\frac{D}{m} = \frac{\deg \det}{m}$ = 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: $\sum_{j=0}^{\mu} \binom{\mu}{j} = 2^{\mu}$

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

New polynomial matrix transformations

Weak Popov to Popov

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

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

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

New polynomial matrix transformations

Weak Popov to 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:	$-\mathbf{s} \geqslant \text{pivotDegree}(\mathbf{A})$
Complexity:	$O(m^\omega M'(\frac{D}{m}))$, 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 a priori known output degrees
- introduction of partial linearization techniques for kernel bases

Reduced to 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 $D = \sum \text{rdeg}_{\mathbf{s}}(\mathbf{A}) - m \min(\mathbf{s})$

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

Summary and perspectives

Summary

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

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

Summary and perspectives

Summary

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

$\frac{\mathrm{D}}{\mathrm{m}} = \frac{\mathbf{degdet}}{\mathrm{m}}$ = 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