Algorithms for structured approximation and interpolation

Sara KHICHANE
Université Paris Cité

Supervised by: Vincent NEIGER
At: LIP6, Sorbonne Université

September 1st, 2025

Context

And notations

Goal

Design algorithms with good complexity bounds for fundamental operations on polynomials and matrices

Exact Computations

Base Field $\mathbb{K}$

$\mathbb{Z}/p\mathbb{Z}$, $\mathbb{F}_{p^e}$, $\mathbb{Q}$, ...

Algebraic Complexity

Upper bound on the number of operations in $\mathbb{K}$

Context

And notations

Goal

Design algorithms with good complexity bounds for fundamental operations on polynomials and matrices

Exact Computations

Base Field $\mathbb{K}$

$\mathbb{Z}/p\mathbb{Z}$, $\mathbb{F}_{p^e}$, $\mathbb{Q}$, ...

Algebraic Complexity

Upper bound on the number of operations in $\mathbb{K}$

Notations:

- $\tilde{O}$: asymptotic bound hiding logarithmic factors.
  $\rightsquigarrow$ Fast polynomial multiplication: $\tilde{O}(d)$ using FFT-based methods.
- ω: exponent of matrix multiplication, $2 < \omega \leq 3$, best known $\omega \approx 2.371$.

Modular approximation

Problem and previous works

Problem 1:

$$\text{ApproxMod}(M, \mathsf{F} \in \mathbb{K}[x]^{n \times m}, \nu = \{\nu_1, \ldots, \nu_m\})$$
$$\text{find } \mathsf{g} \text{ such that } \mathsf{F} \cdot \mathsf{g} = 0 \bmod M \text{ and } \deg(\mathsf{g}_j) < \nu_j$$

Modular approximation

Problem and previous works

Problem 1:

$$\text{ApproxMod}(M, \mathsf{F} \in \mathbb{K}[x]^{n\times m}, \nu = \{\nu_1, \ldots, \nu_m\})$$
$$\text{find g such that } \mathsf{F} \cdot \mathsf{g} = 0 \bmod M \text{ and } \deg(\mathsf{g}_j) < \nu_j$$

Fundamental problems:

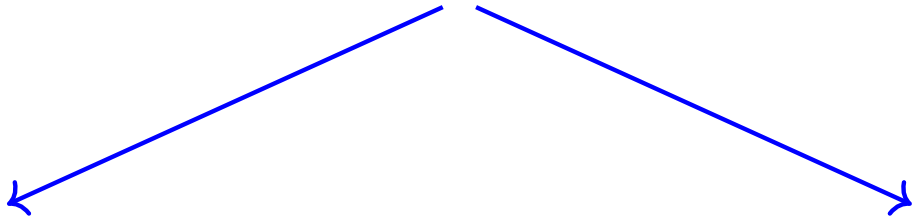
- Hermite-Padé approximation when $M = x^d$.
- Berlekamp–Massey algorithm when $M = x^d$ and $m = 2$.
- Rational interpolation when $M = (x - a_1) \cdots (x - a_d)$ with a_i's distinct.

Modular approximation

Problem and previous works

Problem 1:

$$\text{ApproxMod}(M, \mathsf{F} \in \mathbb{K}[x]^{n\times m}, \nu = \{\nu_1, \ldots, \nu_m\})$$
$$\text{find g such that } \mathsf{F} \cdot \mathsf{g} = 0 \bmod M \text{ and } \deg(\mathsf{g}_j) < \nu_j$$



Family of polynomials
ApproxMultMod($\mathfrak{M}$, F, ν)

$\mathfrak{M} = (\mathsf{M}_1, \ldots, \mathsf{M}_n)$ with
$D = \sum_{i=1}^{n} \deg(\mathsf{M}_i)$

Polynomial matrix
ApproxMatrixMod(M, F, ν)

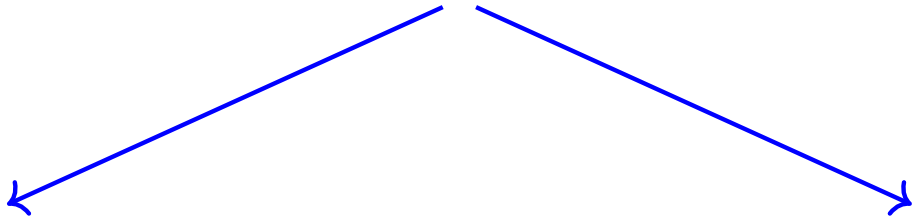
$\mathsf{M} \in \mathbb{K}[x]^{n\times n}$
with $D = \sum_{i=1}^{n} \deg(\mathsf{M}_{ii})$

Modular approximation

Problem and previous works

Problem 1:

$$\text{ApproxMod}(M, \mathsf{F} \in \mathbb{K}[x]^{n\times m}, \nu = \{\nu_1, \ldots, \nu_m\})$$
$$\text{find g such that } \mathsf{F}\cdot\mathsf{g} = 0 \bmod M \text{ and } \deg(\mathsf{g}_j) < \nu_j$$



Family of polynomials
ApproxMultMod($\mathfrak{M}$, F, ν)

$\mathfrak{M} = (\mathsf{M}_1, \ldots, \mathsf{M}_n)$ with
$D = \sum_{i=1}^{n} \deg(\mathsf{M}_i)$

Polynomial matrix
ApproxMatrixMod(M, F, ν)

$\mathsf{M} \in \mathbb{K}[x]^{n\times n}$
with $D = \sum_{i=1}^{n} \deg(\mathsf{M}_{ii})$

Polynomial approach:

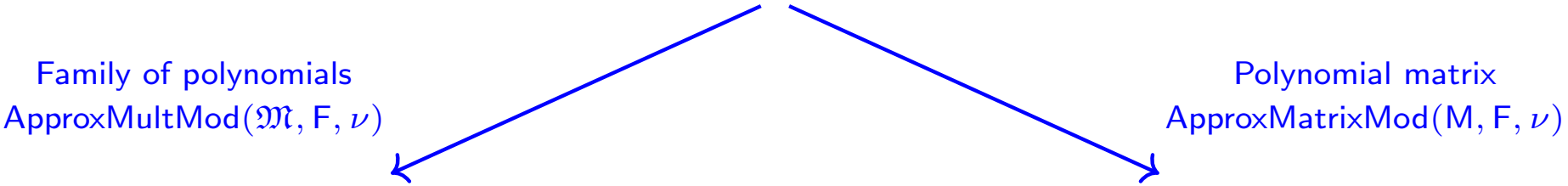
- Beckermann and Labahn 1994: ApproxMultMod solved in $\tilde{O}(m^\omega \cdot D)$.
- Neiger 2016: ApproxMultMod solved in $\tilde{O}(\max(n,m)^{\omega-1} \cdot D)$.
- Neiger and Vu 2017: ApproxMatrixMod solved in $\tilde{O}(\max(n,m)^{\omega-1} \cdot D)$.

Modular approximation

Problem and previous works

Problem 1:

$$\text{ApproxMod}(M, \mathsf{F} \in \mathbb{K}[x]^{n\times m}, \nu = \{\nu_1, \ldots, \nu_m\})$$
$$\text{find g such that } \mathsf{F} \cdot \mathsf{g} = 0 \bmod M \text{ and } \deg(\mathsf{g}_j) < \nu_j$$



Structured approach:

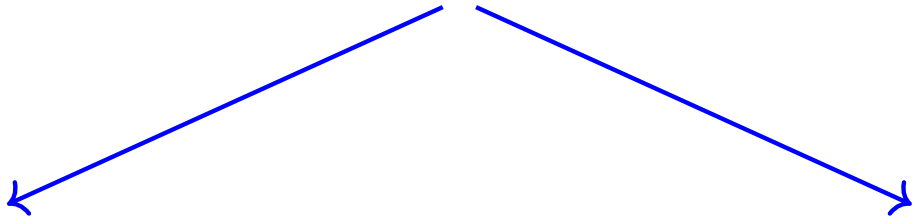
- Chowdhury et al. 2015: ApproxMultMod solved in $\tilde{O}(\max(n,m)^{\omega-1} \cdot n)$.
- **Not done yet** for ApproxMatrixMod.

Modular approximation

Problem and previous works

Problem 1:

$$\text{ApproxMod}(M, \mathsf{F} \in \mathbb{K}[x]^{n\times m}, \nu = \{\nu_1, \ldots, \nu_m\})$$
$$\text{find g such that } \mathsf{F}\cdot \mathsf{g} = 0 \bmod M \text{ and } \deg(\mathsf{g}_j) < \nu_j$$



Family of polynomials
ApproxMultMod($\mathfrak{M}$, F, ν)

$\mathfrak{M} = (\mathsf{M}_1, \ldots, \mathsf{M}_n)$ with
$D = \sum_{i=1}^{n} \deg(\mathsf{M}_i)$

Polynomial matrix
ApproxMatrixMod(M, F, ν)

$\mathsf{M} \in \mathbb{K}[x]^{n\times n}$
with $D = \sum_{i=1}^{n} \deg(\mathsf{M}_{ii})$

Structured approach:

- Chowdhury et al. 2015: ApproxMultMod solved in $\tilde{O}(\max(n,m)^{\omega-1}\cdot n)$.
- **Not done yet** for ApproxMatrixMod.

Reduction to a structured linear system

Structured matrices

Displacement rank approach

Example: Toeplitz matrix

Structured matrices

Displacement rank approach

Example: Toeplitz matrix

$$\mathbb{K}^{4\times 4}$$

$$\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ a_{-1} & a_0 & a_1 & a_2 \\ a_{-2} & a_{-1} & a_0 & a_1 \\ a_{-3} & a_{-2} & a_{-1} & a_0 \end{bmatrix}$$

Structured matrices

Displacement rank approach

Example: Toeplitz matrix

$$\mathbb{K}^{4\times 4}$$

$$\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ a_{-1} & a_0 & a_1 & a_2 \\ a_{-2} & a_{-1} & a_0 & a_1 \\ a_{-3} & a_{-2} & a_{-1} & a_0 \end{bmatrix}$$

$A \in \mathbb{K}^{n\times m}$

Displacement rank:

For $P \in \mathbb{K}^{n\times n}$ and $Q \in \mathbb{K}^{m\times m}$, we define it as the rank of the displacement operator $\Delta[P, Q]$ such that:

$$\Delta[P, Q](A) = A - P \cdot A \cdot Q$$

Structured matrices

Displacement rank approach

Example: Toeplitz matrix

$$\underset{\mathbb{K}^{4\times 4}}{\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ a_{-1} & a_0 & a_1 & a_2 \\ a_{-2} & a_{-1} & a_0 & a_1 \\ a_{-3} & a_{-2} & a_{-1} & a_0 \end{bmatrix}} - \underset{\mathbb{Z}_{n,0}\cdot \mathsf{A}\cdot \mathbb{Z}_{m,0}^{\mathsf{T}}}{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & a_0 & a_1 & a_2 \\ 0 & a_{-1} & a_0 & a_1 \\ 0 & a_{-2} & a_{-1} & a_0 \end{bmatrix}} = \underset{\text{Displacement result}}{\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ a_{-1} & 0 & 0 & 0 \\ a_{-2} & 0 & 0 & 0 \\ a_{-3} & 0 & 0 & 0 \end{bmatrix}}$$

$\mathsf{A} \in \mathbb{K}^{n\times m}$

Displacement rank:

For $\mathsf{P} \in \mathbb{K}^{n\times n}$ and $\mathsf{Q} \in \mathbb{K}^{m\times m}$, we define it as the rank of the displacement operator $\Delta[\mathsf{P},\mathsf{Q}]$ such that:

$$\Delta[\mathsf{P},\mathsf{Q}](\mathsf{A}) = \mathsf{A} - \mathsf{P}\cdot\mathsf{A}\cdot\mathsf{Q}$$

Quasi-Toeplitz Structure: $\mathsf{P} = \mathbb{Z}_{n,0}$, $\mathsf{Q} = \mathbb{Z}_{m,0}^{\mathsf{T}}$

$$\mathbb{Z}_{n,0} = \begin{bmatrix} 0 & & & \\ 1 & 0 & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{bmatrix} \in \mathbb{K}^{n\times n}$$

Structured matrices

Displacement rank approach

Example: Toeplitz matrix

$$\underset{\mathbb{K}^{4\times 4}}{\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ a_{-1} & a_0 & a_1 & a_2 \\ a_{-2} & a_{-1} & a_0 & a_1 \\ a_{-3} & a_{-2} & a_{-1} & a_0 \end{bmatrix}} - \underset{\mathbb{Z}_{n,0}\cdot \mathsf{A}\cdot \mathbb{Z}_{m,0}^{\mathsf{T}}}{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & a_0 & a_1 & a_2 \\ 0 & a_{-1} & a_0 & a_1 \\ 0 & a_{-2} & a_{-1} & a_0 \end{bmatrix}} = \underset{\text{Displacement result}}{\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ a_{-1} & 0 & 0 & 0 \\ a_{-2} & 0 & 0 & 0 \\ a_{-3} & 0 & 0 & 0 \end{bmatrix}}$$

$\mathsf{A} \in \mathbb{K}^{n\times m}$

Displacement rank:
For $\mathsf{P} \in \mathbb{K}^{n\times n}$ and $\mathsf{Q} \in \mathbb{K}^{m\times m}$, we define it as the rank of the displacement operator $\Delta[\mathsf{P},\mathsf{Q}]$ such that:

$$\Delta[\mathsf{P},\mathsf{Q}](\mathsf{A}) = \mathsf{A} - \mathsf{P}\cdot\mathsf{A}\cdot\mathsf{Q}$$

Quasi-Toeplitz Structure: $\mathsf{P} = \mathbb{Z}_{n,0}$, $\mathsf{Q} = \mathbb{Z}_{m,0}^{\mathsf{T}}$

Generators:
$(\mathsf{G},\mathsf{H}) \in \mathbb{K}^{n\times\alpha} \times \mathbb{K}^{m\times\alpha}$ are called a ϕ_+-generator of length α for A if

$$\Delta[\mathbb{Z}_{n,0}, \mathbb{Z}_{m,0}^{\mathsf{T}}](\mathsf{A}) = \mathsf{G}\cdot\mathsf{H}^{\mathsf{T}}.$$

Structured matrices

Displacement rank approach

Example: Toeplitz matrix

$$\underset{\mathbb{K}^{4\times 4}}{\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ a_{-1} & a_0 & a_1 & a_2 \\ a_{-2} & a_{-1} & a_0 & a_1 \\ a_{-3} & a_{-2} & a_{-1} & a_0 \end{bmatrix}} - \underset{\mathbb{Z}_{n,0}\cdot \mathsf{A}\cdot \mathbb{Z}_{m,0}^{\mathsf{T}}}{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & a_0 & a_1 & a_2 \\ 0 & a_{-1} & a_0 & a_1 \\ 0 & a_{-2} & a_{-1} & a_0 \end{bmatrix}} = \underset{\text{Displacement result}}{\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ a_{-1} & 0 & 0 & 0 \\ a_{-2} & 0 & 0 & 0 \\ a_{-3} & 0 & 0 & 0 \end{bmatrix}}$$

$$\underset{\mathsf{G}}{\begin{bmatrix} a_0 & 1 \\ a_{-1} & 0 \\ a_{-2} & 0 \\ a_{-3} & 0 \end{bmatrix}} \times \underset{\mathsf{H}^{\mathsf{T}}}{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & a_1 & a_2 & a_3 \end{bmatrix}} = \underset{\text{Displacement result}}{\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ a_{-1} & 0 & 0 & 0 \\ a_{-2} & 0 & 0 & 0 \\ a_{-3} & 0 & 0 & 0 \end{bmatrix}}$$

$\mathsf{G} \in \mathbb{K}^{4\times 2}$ $\mathsf{H} \in \mathbb{K}^{4\times 2}$

Displacement rank $\alpha = 2$

Structured linear system

Definition and previous works

$\mathsf{A} \in \mathbb{K}^{n \times m}$

Problem 2:

$$\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha](\mathsf{G}, \mathsf{H}, \mathsf{v})$$

Using generators of A: $\mathsf{G} \in \mathbb{K}^{n \times \alpha}, \mathsf{H} \in \mathbb{K}^{m \times \alpha}$

find u such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

Structured linear system

Definition and previous works

$\mathsf{A} \in \mathbb{K}^{n \times m}$

Problem 2:

$$\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha](\mathsf{G}, \mathsf{H}, \mathsf{v})$$

Using generators of A: $\mathsf{G} \in \mathbb{K}^{n \times \alpha}, \mathsf{H} \in \mathbb{K}^{m \times \alpha}$

find u such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

Previous works:

- Kailath, Kung, Morf 1979: solved in $\tilde{O}(\alpha \cdot \max(n, m)^2)$.
- Kaltofen 1994, 1995: solved in $\tilde{O}(\alpha^2 \cdot \max(n, m))$. Better since $\alpha \leq \min(n, m)$
- Bostan, Jeannerod, Schost, Mouilleron 2008, 2017: solved in $\tilde{O}(\alpha^{\omega-1} \cdot \max(n, m))$.

Structured linear system

Definition and previous works

$\mathsf{A} \in \mathbb{K}^{n \times m}$

Problem 2:

$$\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha](\mathsf{G}, \mathsf{H}, \mathsf{v})$$

Using generators of A: $\mathsf{G} \in \mathbb{K}^{n \times \alpha}, \mathsf{H} \in \mathbb{K}^{m \times \alpha}$

find u such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

Previous works:

- Kailath, Kung, Morf 1979: solved in $\tilde{O}(\alpha \cdot \max(n, m)^2)$.
- Kaltofen 1994, 1995: solved in $\tilde{O}(\alpha^2 \cdot \max(n, m))$. Better since $\alpha \leq \min(n, m)$
- Bostan, Jeannerod, Schost, Mouilleron 2008, 2017: solved in $\tilde{O}(\alpha^{\omega-1} \cdot \max(n, m))$.

The algorithms are probabilistic.

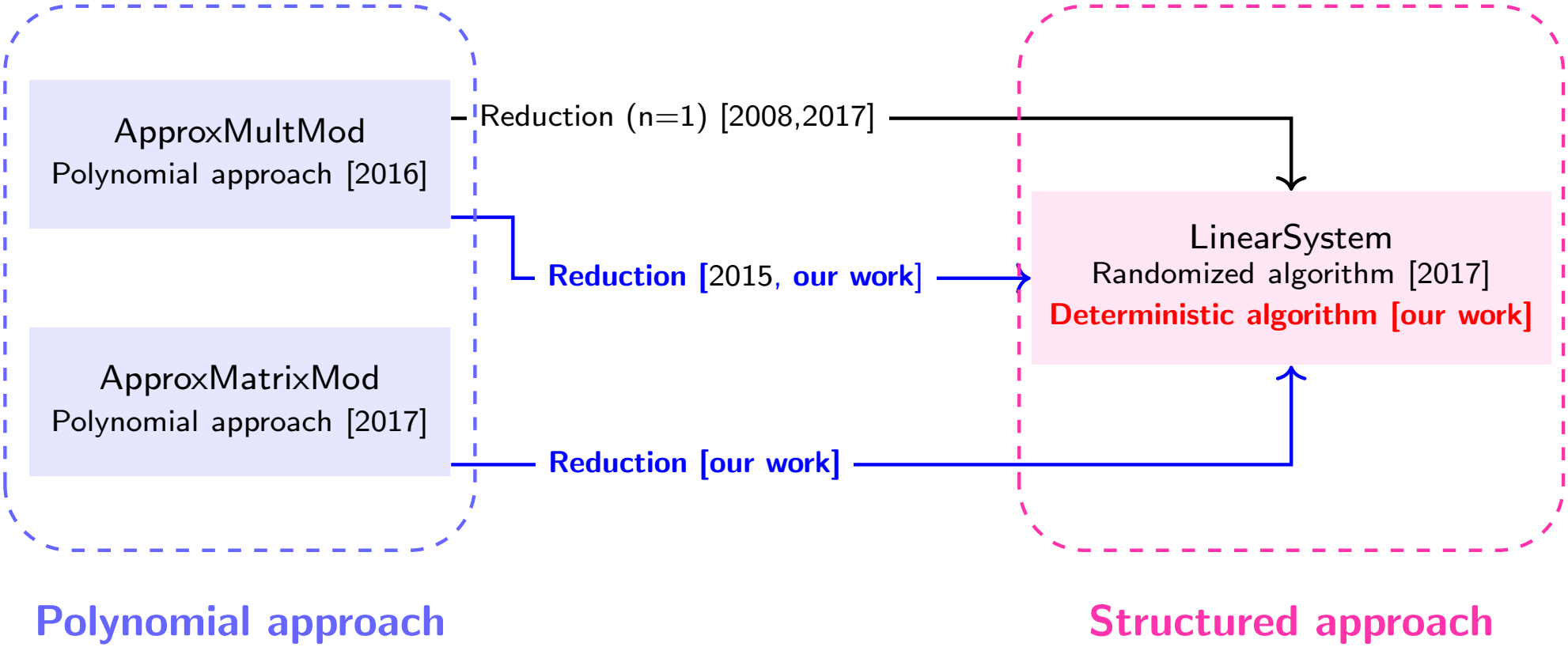
The field $\mathbb{K}$ has to be large enough for the algorithms to work with positive probability.

e.g., 15000×15000 system with $\alpha = 30$ solved in < 1 sec, but requires $|\mathbb{K}| > 2.25 \cdot 10^8 = 15000^2$.

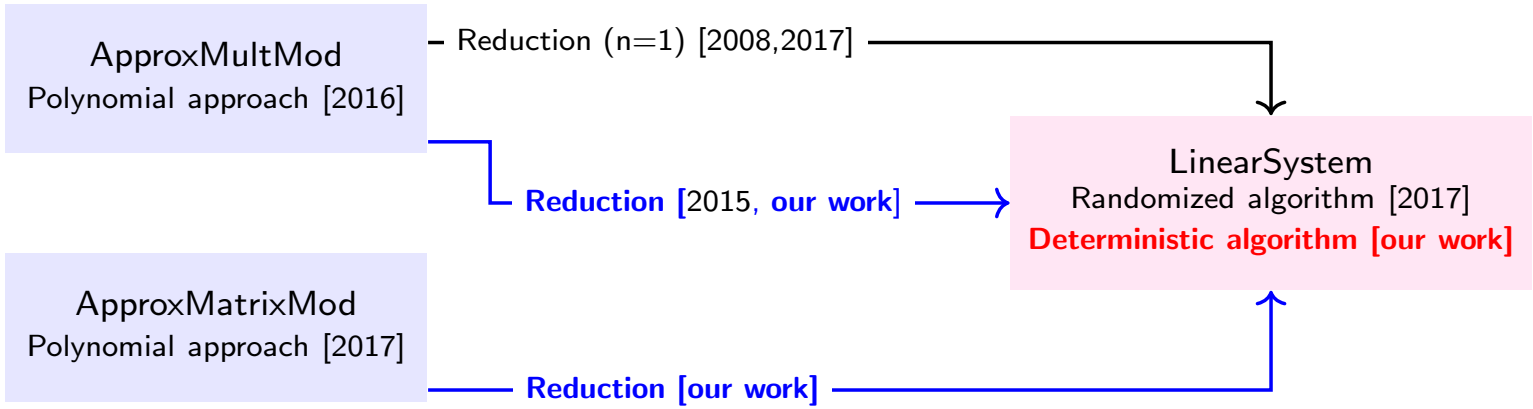
Main results

Bridging the gap between the two approaches

Our contributions:



Matrix modular approximation: reduction



Matrix modular approximation

Modulus polynomial matrices

ApproxMatrixMod	
Input:	a modulus polynomial matrix $\mathsf{M} \in \mathbb{K}[x]^{n \times n}$, a polynomial matrix $\mathsf{F} \in \mathbb{K}[x]^{n \times m}$ such that $\mathrm{rdeg}(\mathsf{F}) < \mathrm{rdeg}(\mathsf{M})$, a set of positive integers $\nu = \{\nu_1, \ldots, \nu_m\}$.
Output:	nonzero $\mathsf{g} \in \mathbb{K}[x]^{m \times 1}$ such that $\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$ and $\deg(\mathsf{g}_j) < \nu_j$.

Matrix modular approximation

Modulus polynomial matrices

ApproxMatrixMod	
Input:	a modulus polynomial matrix $\mathsf{M} \in \mathbb{K}[x]^{n \times n}$, a polynomial matrix $\mathsf{F} \in \mathbb{K}[x]^{n \times m}$ such that $\mathsf{rdeg}(\mathsf{F}) < \mathsf{rdeg}(\mathsf{M})$, a set of positive integers $\nu = \{\nu_1, \ldots, \nu_m\}$.
Output:	nonzero $\mathsf{g} \in \mathbb{K}[x]^{m \times 1}$ such that $\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$ and $\deg(\mathsf{g}_j) < \nu_j$.

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

$$\mathsf{rdeg}(\mathsf{M}) = (2, 1, 2)$$

Matrix modular approximation

Modulus polynomial matrices

ApproxMatrixMod	
Input:	a modulus polynomial matrix $\mathsf{M} \in \mathbb{K}[x]^{n \times n}$, a polynomial matrix $\mathsf{F} \in \mathbb{K}[x]^{n \times m}$ such that $\mathsf{rdeg}(\mathsf{F}) < \mathsf{rdeg}(\mathsf{M})$, a set of positive integers $\nu = \{\nu_1, \ldots, \nu_m\}$.
Output:	nonzero $\mathsf{g} \in \mathbb{K}[x]^{m \times 1}$ such that $\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$ and $\deg(\mathsf{g}_j) < \nu_j$.

$$\mathsf{F} = \begin{bmatrix} x+1 & 2x & 3 & x \\ 4 & 1 & 6 & 2 \\ 7x & 8 & x+3 & 2x \end{bmatrix}$$

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

$\mathsf{rdeg}(\mathsf{F}) = (1, 0, 1)$

$\mathsf{rdeg}(\mathsf{M}) = (2, 1, 2)$

Matrix modular approximation

Modulus polynomial matrices

ApproxMatrixMod	
Input:	a modulus polynomial matrix $\mathsf{M} \in \mathbb{K}[x]^{n \times n}$, a polynomial matrix $\mathsf{F} \in \mathbb{K}[x]^{n \times m}$ such that $\mathsf{rdeg}(\mathsf{F}) < \mathsf{rdeg}(\mathsf{M})$, a set of positive integers $\nu = \{\nu_1, \ldots, \nu_m\}$.
Output:	nonzero $\mathsf{g} \in \mathbb{K}[x]^{m \times 1}$ such that $\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$ and $\deg(\mathsf{g}_j) < \nu_j$.

$$\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$$

$$\begin{bmatrix} x+1 & 2x & 3 & x \\ 4 & 1 & 6 & 2 \\ 7x & 8 & x+3 & 2x \end{bmatrix} \cdot \begin{bmatrix} g_1(x) \\ g_2(x) \\ g_3(x) \\ g_4(x) \end{bmatrix} = 0 \bmod \begin{bmatrix} x^2+3x+2 & 5x+3 & 6x \\ 6 & x+1 & 2 \\ 5x+3 & 3 & x^2+4 \end{bmatrix}$$

$\mathsf{rdeg}(\mathsf{F}) = (1, 0, 1)$ $\quad$ $\deg(\mathsf{g}_j) < \nu_j$ for $j = 1, 2, 3, 4$ $\quad$ $\mathsf{rdeg}(\mathsf{M}) = (2, 1, 2)$

Matrix modular approximation

Modulus polynomial matrices

ApproxMatrixMod	
Input:	a modulus polynomial matrix $\mathsf{M} \in \mathbb{K}[x]^{n \times n}$, a polynomial matrix $\mathsf{F} \in \mathbb{K}[x]^{n \times m}$ such that $\mathsf{rdeg}(\mathsf{F}) < \mathsf{rdeg}(\mathsf{M})$, a set of positive integers $\nu = \{\nu_1, \ldots, \nu_m\}$.
Output:	nonzero $\mathsf{g} \in \mathbb{K}[x]^{m \times 1}$ such that $\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$ and $\deg(\mathsf{g}_j) < \nu_j$.

$$\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$$

$$\begin{bmatrix} x+1 & 2x & 3 & x \\ 4 & 1 & 6 & 2 \\ 7x & 8 & x+3 & 2x \end{bmatrix} \cdot \begin{bmatrix} g_1(x) \\ g_2(x) \\ g_3(x) \\ g_4(x) \end{bmatrix} = 0 \bmod \begin{bmatrix} x^2+3x+2 & 5x+3 & 6x \\ 6 & x+1 & 2 \\ 5x+3 & 3 & x^2+4 \end{bmatrix}$$

$\mathsf{rdeg}(\mathsf{F}) = (1, 0, 1)$ $\quad$ $\deg(\mathsf{g}_j) < \nu_j$ for $j = 1, 2, 3, 4$ $\quad$ $\mathsf{rdeg}(\mathsf{M}) = (2, 1, 2)$

Matrix modular approximation

Modulus polynomial matrices

ApproxMatrixMod	
Input:	a modulus polynomial matrix $\mathsf{M} \in \mathbb{K}[x]^{n \times n}$, a polynomial matrix $\mathsf{F} \in \mathbb{K}[x]^{n \times m}$ such that $\mathrm{rdeg}(\mathsf{F}) < \mathrm{rdeg}(\mathsf{M})$, a set of positive integers $\nu = \{\nu_1, \ldots, \nu_m\}$.
Output:	nonzero $\mathsf{g} \in \mathbb{K}[x]^{m \times 1}$ such that $\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$ and $\deg(\mathsf{g}_j) < \nu_j$.

Theorem

The problem ApproxMatrixMod$(\mathsf{M}, \mathsf{F}, \nu)$ *can be solved in* $\tilde{O}(\max(m,n)^{\omega-1} \cdot D)$.

ApproxMatrixMod$(\mathsf{M}, \mathsf{F}, \nu)$

Find g such that

$\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$

Reduction →

LinearSystem$(\mathsf{G}, \mathsf{H}, \mathbf{0})$

Using generators of A

find u such that

$\mathsf{A} \cdot \mathsf{u} = \mathbf{0}$

Matrix modular approximation

Modulus polynomial matrices

ApproxMatrixMod	
Input:	a modulus polynomial matrix $\mathsf{M} \in \mathbb{K}[x]^{n \times n}$, a polynomial matrix $\mathsf{F} \in \mathbb{K}[x]^{n \times m}$ such that $\text{rdeg}(\mathsf{F}) < \text{rdeg}(\mathsf{M})$, a set of positive integers $\nu = \{\nu_1, \ldots, \nu_m\}$.
Output:	nonzero $\mathsf{g} \in \mathbb{K}[x]^{m \times 1}$ such that $\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$ and $\deg(\mathsf{g}_j) < \nu_j$.

Theorem

The problem ApproxMatrixMod$(\mathsf{M}, \mathsf{F}, \nu)$ *can be solved in* $\tilde{O}(\max(m,n)^{\omega-1} \cdot D)$.

ApproxMatrixMod$(\mathsf{M}, \mathsf{F}, \nu)$

Find g such that
$\mathsf{F} \cdot \mathsf{g} = 0 \bmod \mathsf{M}$

Reduction →

LinearSystem$(\mathsf{G}, \mathsf{H}, \mathbf{0})$

Using generators of A
find u such that
$\mathsf{A} \cdot \mathsf{u} = \mathbf{0}$

$\mathbb{K}[x]$ $\qquad$ $\mathbb{K}$

Multiplication matrix

From polynomials to coefficients

Multiplication matrix: For $M(x) = x^d + m_{d-1}x^{d-1} + \cdots + m_1 x + m_0 \in \mathbb{K}[x]$

$$\mathbb{X}(M) = \begin{bmatrix} 0 & \cdots & 0 & -m_0 \\ 1 & \cdots & 0 & -m_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & -m_{d-1} \end{bmatrix}$$

Multiplication matrix

From polynomials to coefficients

Multiplication matrix: For $M(x) = x^d + m_{d-1}x^{d-1} + \cdots + m_1 x + m_0 \in \mathbb{K}[x]$

$$\mathbb{X}(M) = \begin{bmatrix} 0 & \cdots & 0 & -m_0 \\ 1 & \cdots & 0 & -m_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & -m_{d-1} \end{bmatrix}$$

$$\mathbb{X}(M)^k \cdot \begin{bmatrix} \mathsf{p}_0 \\ \vdots \\ \mathsf{p}_{d-1} \end{bmatrix} \xrightarrow{\text{coefficient vector}} x^k \cdot (\mathsf{p}_0 + \cdots + \mathsf{p}_{d-1}x^{d-1}) \text{ rem } M$$

Multiplication matrix

From polynomials to coefficients

Multiplication matrix: For $M(x) = x^d + m_{d-1}x^{d-1} + \cdots + m_1 x + m_0 \in \mathbb{K}[x]$

$$\mathbb{X}(M) = \begin{bmatrix} 0 & \cdots & 0 & -m_0 \\ 1 & \cdots & 0 & -m_1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & -m_{d-1} \end{bmatrix}$$

$$\mathbb{X}(M)^k \cdot \begin{bmatrix} \mathsf{p}_0 \\ \vdots \\ \mathsf{p}_{d-1} \end{bmatrix} \xrightarrow{\text{coefficient vector}} x^k \cdot (\mathsf{p}_0 + \cdots + \mathsf{p}_{d-1}x^{d-1}) \text{ rem } M$$

Generalization to M: $\mathsf{rdeg}(\mathsf{M}) = [d_1, \ldots, d_n]$
Let $D = \sum_{i=1}^{n} d_i$.

$$\mathbb{X}(\mathsf{M}) = \begin{bmatrix} \mathbb{X}(\mathsf{M}_{11}) & \mathsf{C}_{12} & \cdots & \mathsf{C}_{1n} \\ \mathsf{C}_{21} & \mathbb{X}(\mathsf{M}_{22}) & \cdots & \mathsf{C}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathsf{C}_{n1} & \mathsf{C}_{n2} & \cdots & \mathbb{X}(\mathsf{M}_{nn}) \end{bmatrix} \qquad \mathsf{C}_{ij} = \begin{bmatrix} 0 & \cdots & 0 & -\mathsf{coeff}(\mathsf{M}_{ij}, 0) \\ 0 & \cdots & 0 & -\mathsf{coeff}(\mathsf{M}_{ij}, 1) \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & -\mathsf{coeff}(\mathsf{M}_{ij}, d_i - 1) \end{bmatrix}.$$

Definition of A

Reduction

Let $C = \nu_1 + \cdots + \nu_m$.
Let $D = |\mathsf{rdeg}(\mathsf{M})| = d_1 + \cdots + d_n$.

We define $\mathsf{A} = \begin{bmatrix} \mathsf{A}_1 & \mathsf{A}_2 & \cdots & \mathsf{A}_m \end{bmatrix} \in \mathbb{K}^{D \times C}$ with:

$$\mathsf{A}_j = \Big[\underset{\mathsf{F}_{*j}}{\overline{\mathsf{F}}_{*j}} \quad \underset{x \cdot \mathsf{F}_{*j} \text{ rem } \mathsf{M}}{\mathbb{X}(\mathsf{M}) \cdot \overline{\mathsf{F}}_{*j}} \quad \cdots \quad \underset{x^{\nu_j - 1} \cdot \mathsf{F}_{*j} \text{ rem } \mathsf{M}}{\mathbb{X}(\mathsf{M})^{\nu_j - 1} \cdot \overline{\mathsf{F}}_{*j}} \Big] \in \mathbb{K}^{D \times \nu_j}.$$

Here $\overline{\mathsf{F}}_{*j}$ is the vector of coefficients of F_{*j} the j-th column of F.

Definition of A

Reduction

Let $C = \nu_1 + \cdots + \nu_m$. Number of unknowns
Let $D = |\mathsf{rdeg}(\mathsf{M})| = d_1 + \cdots + d_n$. Number of equations

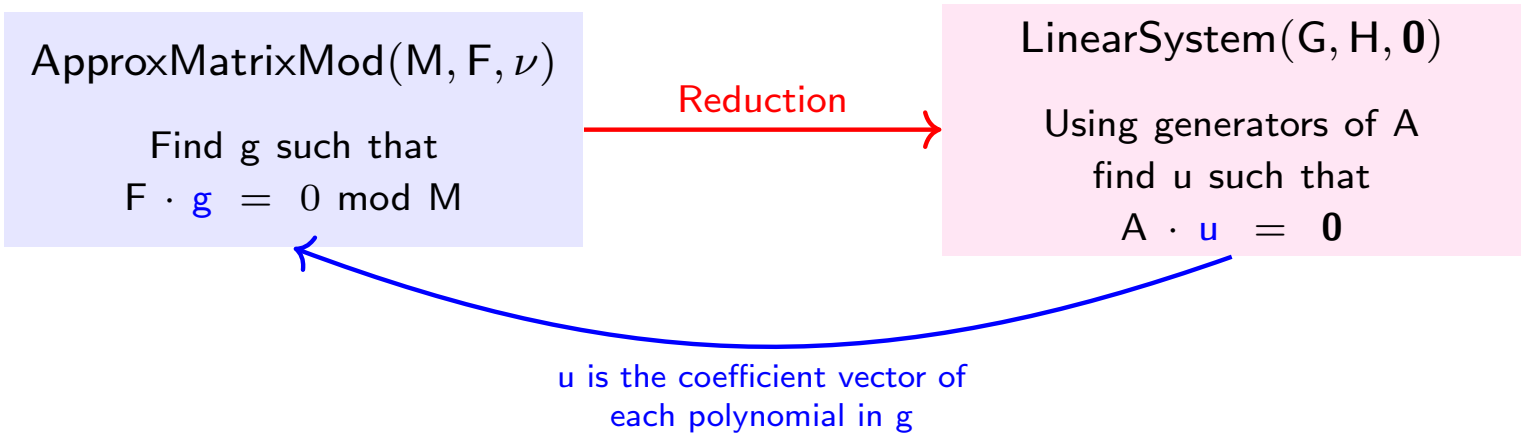
We define $\mathsf{A} = \begin{bmatrix} \mathsf{A}_1 & \mathsf{A}_2 & \cdots & \mathsf{A}_m \end{bmatrix} \in \mathbb{K}^{D \times C}$ with:

$$\mathsf{A}_j = \begin{bmatrix} \overline{\mathsf{F}}_{*j} & \mathbb{X}(\mathsf{M}) \cdot \overline{\mathsf{F}}_{*j} & \cdots & \mathbb{X}(\mathsf{M})^{\nu_j - 1} \cdot \overline{\mathsf{F}}_{*j} \end{bmatrix} \in \mathbb{K}^{D \times \nu_j}.$$

$\overline{\mathsf{F}}_{*j} \to \mathsf{F}_{*j}$ $\quad \mathbb{X}(\mathsf{M}) \cdot \overline{\mathsf{F}}_{*j} \to x \cdot \mathsf{F}_{*j}$ rem M $\quad \mathbb{X}(\mathsf{M})^{\nu_j - 1} \cdot \overline{\mathsf{F}}_{*j} \to x^{\nu_j - 1} \cdot \mathsf{F}_{*j}$ rem M

Here $\overline{\mathsf{F}}_{*j}$ is the vector of coefficients of F_{*j} the j-th column of F.

Reduction:



Generators of A

Reduction cost

Goal: Find (G, H) generators of length $O(\max(n, m))$ for A.

Generators of A

Reduction cost

Goal: Find (G, H) generators of length $O(\max(n, m))$ for A.

We proceed by splitting the displacement operator:

$$\mathsf{A} - \mathbb{Z}_{D,0} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} = \mathsf{A} - \mathbb{X}(\mathsf{M}) \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} + \mathbb{Y} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} - \delta \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} - \mathsf{N} \cdot \mathsf{E} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$$

$\mathsf{G} = \begin{bmatrix} \mathsf{Y} & \mathsf{R} & \mathsf{T} & \mathsf{N} \end{bmatrix} \in \mathbb{K}^{D \times (m+3n)}$ and $\mathsf{H} = \begin{bmatrix} \mathsf{Z} & \mathsf{U} & \mathsf{U} & \mathsf{Q} \end{bmatrix} \in \mathbb{K}^{C \times (m+3n)}$

Generators of A

Reduction cost

Goal: Find (G, H) generators of length $O(\max(n, m))$ for A.

We proceed by splitting the displacement operator:

$$\mathsf{A} - \mathbb{Z}_{D,0} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} = \mathsf{A} - \mathbb{X}(\mathsf{M}) \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} + \mathbb{Y} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} - \delta \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} - \mathsf{N} \cdot \mathsf{E} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$$

$\mathsf{G} = \begin{bmatrix} \mathsf{Y} & \mathsf{R} & \mathsf{T} & \mathsf{N} \end{bmatrix} \in \mathbb{K}^{D \times (m+3n)}$ and $\mathsf{H} = \begin{bmatrix} \mathsf{Z} & \mathsf{U} & \mathsf{U} & \mathsf{Q} \end{bmatrix} \in \mathbb{K}^{C \times (m+3n)}$

LinearSystem$(\mathsf{G}, \mathsf{H}, \mathbf{0})$

Using generators of length $O(\max(n, m))$ for A
find u such that $\mathsf{A} \cdot \mathsf{u} = \mathbf{0}$

$\tilde{O}(\max(n, m)^{\omega-1} \cdot D)$

Generators of A

Reduction cost

Goal: Find (G, H) generators of length $O(\max(n, m))$ for A.

We proceed by splitting the displacement operator:

$$\mathsf{A} - \mathbb{Z}_{D,0} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} = \mathsf{A} - \mathbb{X}(\mathsf{M}) \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} + \mathbb{Y} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} - \delta \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}} - \mathsf{N} \cdot \mathsf{E} \cdot \mathsf{A} \cdot \mathbb{Z}_{C,0}^{\mathsf{T}}$$

$\mathsf{G} = \begin{bmatrix} \mathsf{Y} & \mathsf{R} & \mathsf{T} & \mathsf{N} \end{bmatrix} \in \mathbb{K}^{D \times (m+3n)}$ and $\mathsf{H} = \begin{bmatrix} \mathsf{Z} & \mathsf{U} & \mathsf{U} & \mathsf{Q} \end{bmatrix} \in \mathbb{K}^{C \times (m+3n)}$

LinearSystem$(\mathsf{G}, \mathsf{H}, \mathbf{0})$

Using generators of length $O(\max(n, m))$ for A
find u such that $\mathsf{A} \cdot \mathsf{u} = \mathbf{0}$

$\tilde{O}(\max(n, m)^{\omega-1} \cdot D)$

The total cost of the reduction:
$\tilde{O}(n^{\omega-1} \cdot D + m \cdot n^{\omega-2} \cdot D + n^{\omega-1} \cdot C) \leq \tilde{O}(\max(n, m)^{\omega-1} \cdot D)$.

High degree coefficients

Problem overview

LastCoeffModMat	
Input:	$\mathsf{M} \in \mathbb{K}[x]^{n\times n}, \mathsf{F} \in \mathbb{K}[x]^{n\times m}$, $\nu = \{\nu_1, \ldots, \nu_m\}$ with $D = \mathsf{rdeg}(\mathsf{M}) = d_1 + \cdots + d_n$ and $C = \nu_1 + \cdots + \nu_m$
Output:	For $j = 1, \ldots, m$ and $k_j = 0, \ldots, \nu_j - 1$, $\mathsf{a}_{k_j,j} = [\mathsf{coeff}([x^{k_j} \cdot \mathsf{F} \text{ rem } \mathsf{M}]_{ij}, d_i - 1)]_{i=1,\ldots,n}$

High degree coefficients

Problem overview

LastCoeffModMat	
Input:	$\mathsf{M} \in \mathbb{K}[x]^{n\times n}, \mathsf{F} \in \mathbb{K}[x]^{n\times m}$, $\nu = \{\nu_1, \ldots, \nu_m\}$ with $D = \mathsf{rdeg}(\mathsf{M}) = d_1 + \cdots + d_n$ and $C = \nu_1 + \cdots + \nu_m$
Output:	For $j = 1, \ldots, m$ and $k_j = 0, \ldots, \nu_j - 1$, $\mathsf{a}_{k_j,j} = [\mathsf{coeff}([x^{k_j} \cdot \mathsf{F} \text{ rem } \mathsf{M}]_{ij}, d_i - 1)]_{i=1,\ldots,n}$

Compact computation: $x^{k_j+1} \cdot \mathsf{F}_{*j} = \mathsf{M} \cdot \underbrace{(x \cdot \mathsf{Q}_{k_j,j} + \mathsf{a}_{k_j,j})}_{\mathsf{Q}_{k_j+1,j}} + (\mathsf{a}_{k_j,j} \cdot \mathsf{M}' + x \cdot \mathbb{R}'_{k_j,j})$

$$
\begin{aligned}
\overline{\mathsf{Q}_{0,j}} &= 0,\\
\overline{\mathsf{Q}_{1,j}} &= x^0 \cdot \mathsf{a}_{0,j} + 0 = \mathsf{a}_{0,j},\\
&\vdots\\
\overline{\mathsf{Q}_{\nu_j,j}} &= x^{\nu_j-1} \cdot \mathsf{a}_{0,j} + x^{\nu_j-2} \cdot \mathsf{a}_{1,j} + \cdots + x^0 \cdot \mathsf{a}_{\nu_j-1,j}\ .
\end{aligned}
$$

High degree coefficients

Problem overview

LastCoeffModMat	
Input:	$\mathsf{M} \in \mathbb{K}[x]^{n\times n}, \mathsf{F} \in \mathbb{K}[x]^{n\times m}$, $\nu = \{\nu_1, \ldots, \nu_m\}$ with $D = \mathsf{rdeg}(\mathsf{M}) = d_1 + \cdots + d_n$ and $C = \nu_1 + \cdots + \nu_m$
Output:	For $j = 1, \ldots, m$ and $k_j = 0, \ldots, \nu_j - 1$, $\mathsf{a}_{k_j,j} = [\mathsf{coeff}([x^{k_j} \cdot \mathsf{F} \text{ rem } \mathsf{M}]_{ij}, d_i - 1)]_{i=1,\ldots,n}$

Compact computation: $x^{k_j+1} \cdot \mathsf{F}_{*j} = \mathsf{M} \cdot \underbrace{(x \cdot \mathsf{Q}_{k_j,j} + \mathsf{a}_{k_j,j})}_{\mathsf{Q}_{k_j+1,j}} + (\mathsf{a}_{k_j,j} \cdot \mathsf{M}' + x \cdot \mathbb{R}'_{k_j,j})$

$$\begin{aligned}
\overline{\mathsf{Q}_{0,j}} &= 0,\\
\overline{\mathsf{Q}_{1,j}} &= x^0 \cdot \mathsf{a}_{0,j} + 0 = \mathsf{a}_{0,j},\\
&\vdots\\
\overline{\mathsf{Q}_{\nu_j,j}} &= x^{\nu_j-1} \cdot \mathsf{a}_{0,j} + x^{\nu_j-2} \cdot \mathsf{a}_{1,j} + \cdots + x^0 \cdot \mathsf{a}_{\nu_j-1,j}\ .
\end{aligned}$$

$\rightsquigarrow \overline{\mathsf{Q}_\nu} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}} \text{ rrem } \mathbb{X}^\nu$

High degree coefficients

Problem overview

LastCoeffModMat	
Input:	$\mathsf{M} \in \mathbb{K}[x]^{n\times n}, \mathsf{F} \in \mathbb{K}[x]^{n\times m}$, $\nu = \{\nu_1, \ldots, \nu_m\}$ with $D = \mathsf{rdeg}(\mathsf{M}) = d_1 + \cdots + d_n$ and $C = \nu_1 + \cdots + \nu_m$
Output:	For $j = 1, \ldots, m$ and $k_j = 0, \ldots, \nu_j - 1$, $\mathsf{a}_{k_j,j} = [\mathsf{coeff}([x^{k_j} \cdot \mathsf{F} \text{ rem } \mathsf{M}]_{ij}, d_i - 1)]_{i=1,\ldots,n}$

Compact computation: $x^{k_j+1} \cdot \mathsf{F}_{*j} = \mathsf{M} \cdot \underbrace{(x \cdot \mathsf{Q}_{k_j,j} + \mathsf{a}_{k_j,j})}_{\mathsf{Q}_{k_j+1,j}} + (\mathsf{a}_{k_j,j} \cdot \mathsf{M}' + x \cdot \mathbb{R}'_{k_j,j})$

$$\begin{aligned}
\overline{\mathsf{Q}_{0,j}} &= 0,\\
\overline{\mathsf{Q}_{1,j}} &= x^0 \cdot \mathsf{a}_{0,j} + 0 = \mathsf{a}_{0,j},\\
&\vdots\\
\overline{\mathsf{Q}_{\nu_j,j}} &= x^{\nu_j-1} \cdot \mathsf{a}_{0,j} + x^{\nu_j-2} \cdot \mathsf{a}_{1,j} + \cdots + x^0 \cdot \mathsf{a}_{\nu_j-1,j}\ .
\end{aligned}$$

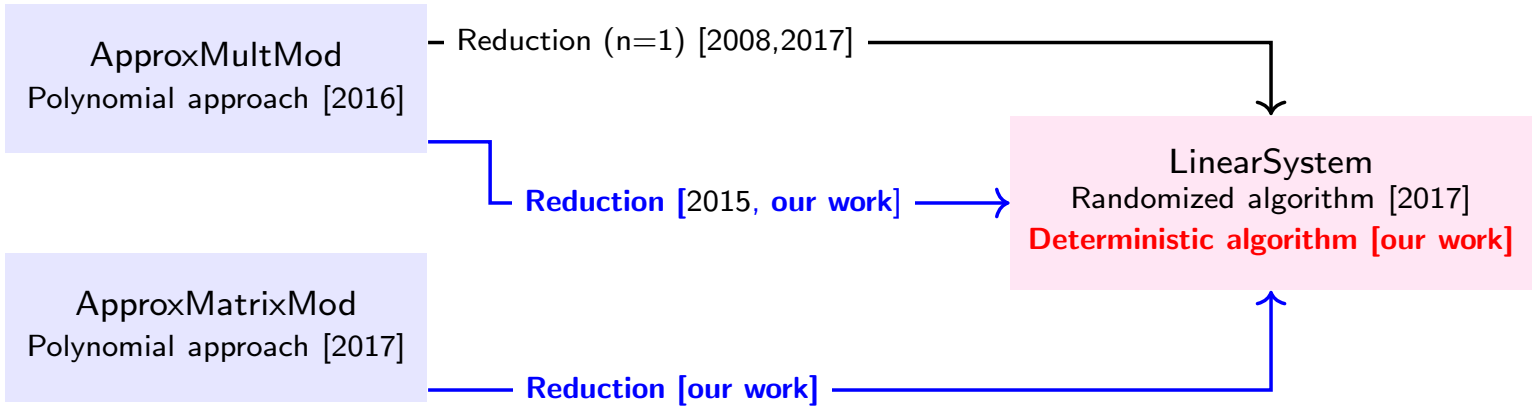
$\rightsquigarrow \overline{\mathsf{Q}_\nu} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}} \text{ rrem } \mathbb{X}^\nu$

Truncated inverse product lemma: $\tilde{O}(n^{\omega-1} \cdot C)$

- State of the art: the problem is solved under the assumption that $\max(\nu) \leq 2 \cdot \frac{C}{m}$.

Applying the algorithm on a selected set of columns of $\overline{\mathsf{F}}$.
$\rightsquigarrow \log_2(m)$ steps since $C = \sum_{j=1}^{m} \nu_j$.

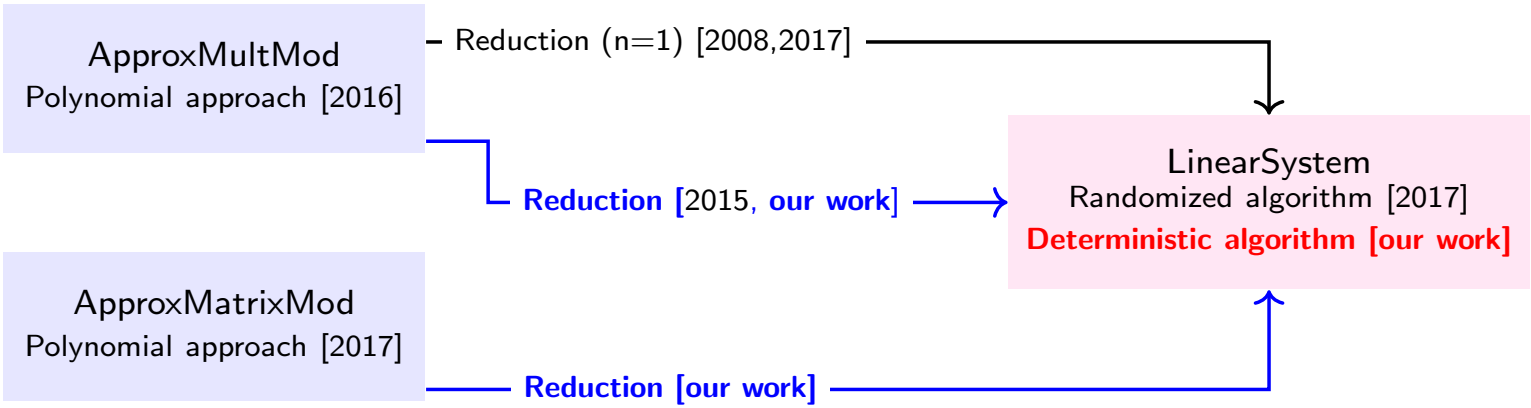
Structured linear system: deterministic algorithm



Quasi-Toeplitz linear system

Deterministic algorithm

We recall:



LinearSystem$[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha]$	
Input:	$\mathsf{G} \in \mathbb{K}^{n\times\alpha}, \mathsf{H} \in \mathbb{K}^{m\times\alpha}$ are ϕ_+-generators of length α for A, $\mathsf{v} \in \mathbb{K}^{n\times 1}$.
Output:	nonzero $\mathsf{u} \in \mathbb{K}^{m\times 1}$ such that $\mathsf{A}\cdot\mathsf{u} = \mathsf{v}$ if it exists, otherwise return $\emptyset$.

Theorem

We can solve LinearSystem$[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha]$ *in* $\tilde{O}(\alpha^{\omega-1}\cdot\max(n,m))$ **deterministically**.

$\rightsquigarrow$ no constraint on the size of the field $\mathbb{K}$

Polynomial formulation

First step

$$\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}_{m,0}^{\mathsf{T}}, \alpha](\mathsf{G}, \mathsf{H}, \mathsf{v})$$

Using generators of A, find u such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

Polynomial formulation

First step

$$\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}_{m,0}^{\mathsf{T}}, \alpha](\mathsf{G}, \mathsf{H}, \mathsf{v})$$

Using generators of A, find u such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

Σ-LU decomposition of A:
$\phi_+(\mathsf{A}) = \mathsf{G} \cdot \mathsf{H}^{\mathsf{T}}$

$$\mathsf{A} \cdot \mathsf{u} = \mathsf{v} \Leftrightarrow \sum_{i=1}^{\alpha} \mathbb{L}(\mathsf{G}_{*i}) \cdot \mathbb{U}(\mathsf{H}_{*i}) \cdot \mathsf{u} = \mathsf{v}$$

$$\Leftrightarrow \sum_{i=1}^{\alpha} \begin{bmatrix} G_{1i} & 0 & \cdots & 0 \\ G_{2i} & G_{1i} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ G_{mi} & \cdots & G_{2i} & G_{1i} \end{bmatrix} \cdot \begin{bmatrix} H_{1i} & H_{2i} & \cdots & H_{mi} \\ 0 & H_{1i} & \ddots & \vdots \\ \vdots & \ddots & \ddots & H_{2i} \\ 0 & \cdots & 0 & H_{1i} \end{bmatrix} \cdot \mathsf{u} = \mathsf{v}$$

Polynomial formulation

First step

$$\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha](\mathsf{G}, \mathsf{H}, \mathsf{v})$$

Using generators of A, find u such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

Σ-LU decomposition of A:
$\phi_+(\mathsf{A}) = \mathsf{G} \cdot \mathsf{H}^{\mathsf{T}}$

$$\mathsf{A} \cdot \mathsf{u} = \mathsf{v} \Leftrightarrow \sum_{i=1}^{\alpha} \mathbb{L}(\mathsf{G}_{*i}) \cdot \mathbb{U}(\mathsf{H}_{*i}) \cdot \mathsf{u} = \mathsf{v}$$

$$\Leftrightarrow \sum_{i=1}^{\alpha} \begin{bmatrix} G_{1i} & 0 & \cdots & 0 \\ G_{2i} & G_{1i} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ G_{mi} & \cdots & G_{2i} & G_{1i} \end{bmatrix} \cdot \begin{bmatrix} H_{1i} & H_{2i} & \cdots & H_{mi} \\ 0 & H_{1i} & \ddots & \vdots \\ \vdots & \ddots & \ddots & H_{2i} \\ 0 & \cdots & 0 & H_{1i} \end{bmatrix} \cdot \mathsf{u} = \mathsf{v}$$

$$\Leftrightarrow \sum_{i=1}^{\alpha} f_i(x) \cdot (\mathsf{rev}_{n-1}(g_i) \cdot u(x) \text{ quo } x^{n-1}) \bmod x^n = v(x)$$

Polynomial formulation

First step

$$\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha](\mathsf{G}, \mathsf{H}, \mathsf{v})$$

Using generators of A, find u such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

Σ-LU decomposition of A:
$\phi_+(\mathsf{A}) = \mathsf{G} \cdot \mathsf{H}^{\mathsf{T}}$

$$\mathsf{A} \cdot \mathsf{u} = \mathsf{v} \Leftrightarrow \sum_{i=1}^{\alpha} \mathbb{L}(\mathsf{G}_{*i}) \cdot \mathbb{U}(\mathsf{H}_{*i}) \cdot \mathsf{u} = \mathsf{v}$$

$$\Leftrightarrow \sum_{i=1}^{\alpha} f_i(x) \cdot (\mathsf{rev}_{n-1}(g_i) \cdot u(x) \ \mathsf{quo} \ x^{n-1}) \bmod x^n = v(x)$$

Let $\mathsf{F} = \begin{bmatrix} f_1(x) & f_2(x) & \cdots & f_\alpha(x) \end{bmatrix}$.

Polynomial formulation

First step

$$\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha](\mathsf{G}, \mathsf{H}, \mathsf{v})$$

Using generators of A, find u such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

Σ-LU decomposition of A:
$\phi_+(\mathsf{A}) = \mathsf{G} \cdot \mathsf{H}^{\mathsf{T}}$

$$\mathsf{A} \cdot \mathsf{u} = \mathsf{v} \Leftrightarrow \sum_{i=1}^{\alpha} \mathbb{L}(\mathsf{G}_{*i}) \cdot \mathbb{U}(\mathsf{H}_{*i}) \cdot \mathsf{u} = \mathsf{v}$$

$$\Leftrightarrow \sum_{i=1}^{\alpha} f_i(x) \cdot (\mathsf{rev}_{n-1}(g_i) \cdot u(x) \ \mathsf{quo}\ x^{n-1}) \bmod x^n = v(x)$$

Let $\mathsf{F} = \begin{bmatrix} f_1(x) & f_2(x) & \cdots & f_\alpha(x) \end{bmatrix}$.

$\mathsf{LinearSystem}(\mathsf{G}, \mathsf{H}, \mathsf{v})$
Find u such that
$\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

Find the basis of solutions
$\{\mathsf{p} \in \mathbb{K}[x]^{\alpha \times 1} \mid \mathsf{F} \cdot \mathsf{p} = v(x) \bmod x^n\}$

Approximant basis

Second step

Find the basis of solutions

$$\{\mathsf{p} \in \mathbb{K}[x]^{\alpha \times 1} \mid \mathsf{F} \cdot \mathsf{p} = v(x) \bmod x^n\}$$

Approximant basis

Second step

Find the basis of solutions

$$\{\mathsf{p} \in \mathbb{K}[x]^{\alpha \times 1} \mid \mathsf{F} \cdot \mathsf{p} = v(x) \bmod x^n\}$$

- $\mathsf{P} \in \mathbb{K}[x]^{\alpha \times \alpha}$ the basis of solutions $\{\mathsf{q} \in \mathbb{K}[x]^{\alpha \times 1} \mid \mathsf{F} \cdot \mathsf{q} = 0 \bmod x^n\}$
- $\mathsf{s} \in \mathbb{K}[x]^{\alpha \times 1}$ such that $\mathsf{F} \cdot \mathsf{s} = v(x) \bmod x^n$

$$\mathsf{p} = \mathsf{P} \cdot \lambda + \mathsf{s}$$

Approximant basis

Second step

Find the basis of solutions

$$\{\mathsf{p} \in \mathbb{K}[x]^{\alpha\times 1} \mid \mathsf{F}\cdot\mathsf{p} = v(x) \bmod x^n\}$$

- $\mathsf{P} \in \mathbb{K}[x]^{\alpha\times\alpha}$ the basis of solutions $\{\mathsf{q} \in \mathbb{K}[x]^{\alpha\times 1} \mid \mathsf{F}\cdot\mathsf{q} = 0 \bmod x^n\}$
- $\mathsf{s} \in \mathbb{K}[x]^{\alpha\times 1}$ such that $\mathsf{F}\cdot\mathsf{s} = v(x) \bmod x^n$

$$\mathsf{p} = \mathsf{P}\cdot\lambda + \mathsf{s}$$

$$\bar{\mathsf{g}}\cdot u(x) \text{ quo } x^{n-1} = \mathsf{P}\cdot\lambda + \mathsf{s}$$

Approximant basis

Second step

Find the basis of solutions

$$\{\mathsf{p} \in \mathbb{K}[x]^{\alpha\times 1} \mid \mathsf{F}\cdot\mathsf{p} = v(x) \bmod x^n\}$$

- $\mathsf{P} \in \mathbb{K}[x]^{\alpha\times\alpha}$ the basis of solutions $\{\mathsf{q} \in \mathbb{K}[x]^{\alpha\times 1} \mid \mathsf{F}\cdot\mathsf{q} = 0 \bmod x^n\}$
- $\mathsf{s} \in \mathbb{K}[x]^{\alpha\times 1}$ such that $\mathsf{F}\cdot\mathsf{s} = v(x) \bmod x^n$

$$\mathsf{p} = \mathsf{P}\cdot\lambda + \mathsf{s}$$

$$\bar{\mathsf{g}}\cdot u(x) \text{ quo } x^{n-1} = \mathsf{P}\cdot\lambda + \mathsf{s}$$

Find the basis of solutions

$$\{\mathsf{p} \in \mathbb{K}[x]^{\alpha\times 1} \mid \mathsf{F}\cdot\mathsf{p} = v(x) \bmod x^n\}$$

Approximant basis

Find $u(x)$ and λ such that

$$\bar{\mathsf{g}}\cdot u(x) \text{ quo } x^{n-1} = \mathsf{P}\cdot\lambda + \mathsf{s}$$

Simultaneous extended Hermite-Padé approximation

Third step

Find $u(x)$ and λ such that

$$\bar{\mathsf{g}} \cdot u(x) \text{ quo } x^{n-1} = \mathsf{P} \cdot \lambda + \mathsf{s}$$

Simultaneous extended Hermite-Padé approximation

Third step

Find $u(x)$ and λ such that

$$\bar{\mathsf{g}} \cdot u(x) \text{ quo } x^{n-1} = \mathsf{P} \cdot \lambda + \mathsf{s}$$

$$\Leftrightarrow \mathsf{g} \cdot \mathrm{rev}_{n-1}(u) - \mathrm{rev}_{n-1}(\mathsf{s}) = \bar{\mathsf{P}}\bar{\lambda} \bmod x^n$$

Simultaneous extended Hermite-Padé approximation

Third step

Find $u(x)$ and λ such that

$$\bar{\mathsf{g}} \cdot u(x) \text{ quo } x^{n-1} = \mathsf{P} \cdot \lambda + \mathsf{s}$$

$\Leftrightarrow \mathsf{g} \cdot \mathrm{rev}_{n-1}(u) - \mathrm{rev}_{n-1}(\mathsf{s}) = \bar{\mathsf{P}}\bar{\lambda} \bmod x^n$

$\Leftrightarrow \bar{\mathsf{P}}^{-1} \cdot \mathsf{g} \cdot \mathrm{rev}_{n-1}(u) - \bar{\mathsf{P}}^{-1} \cdot \mathrm{rev}_{n-1}(\mathsf{s}) = \bar{\lambda} \bmod x^n$

P is in Popov form $\Rightarrow \bar{\mathsf{P}}$ is invertible mod x^n

Simultaneous extended Hermite-Padé approximation

Third step

Find $u(x)$ and λ such that

$$\bar{\mathsf{g}} \cdot u(x) \text{ quo } x^{n-1} = \mathsf{P} \cdot \lambda + \mathsf{s}$$

$$\Leftrightarrow \mathsf{g} \cdot \mathrm{rev}_{n-1}(u) - \mathrm{rev}_{n-1}(\mathsf{s}) = \bar{\mathsf{P}}\bar{\lambda} \bmod x^n$$

$$\Leftrightarrow \bar{\mathsf{P}}^{-1} \cdot \mathsf{g} \cdot \mathrm{rev}_{n-1}(u) - \bar{\mathsf{P}}^{-1} \cdot \mathrm{rev}_{n-1}(\mathsf{s}) = \bar{\lambda} \bmod x^n$$

P is in Popov form $\Rightarrow$ $\bar{\mathsf{P}}$ is invertible mod x^n

$$\Leftrightarrow \begin{bmatrix} \bar{\mathsf{P}}^{-1} \cdot \mathsf{g} & -\bar{\mathsf{P}}^{-1} \cdot \mathrm{rev}_{n-1}(\mathsf{s}) \end{bmatrix} \cdot \begin{bmatrix} \mathrm{rev}_{n-1}(u) \\ 1 \end{bmatrix} = \bar{\lambda} \bmod x^n$$

Simultaneous extended Hermite-Padé approximation

Third step

Find $u(x)$ and λ such that

$$\bar{\mathsf{g}} \cdot u(x) \text{ quo } x^{n-1} = \mathsf{P} \cdot \lambda + \mathsf{s}$$

$$\Leftrightarrow \mathsf{g} \cdot \mathrm{rev}_{n-1}(u) - \mathrm{rev}_{n-1}(\mathsf{s}) = \bar{\mathsf{P}}\bar{\lambda} \bmod x^n$$

$$\Leftrightarrow \bar{\mathsf{P}}^{-1} \cdot \mathsf{g} \cdot \mathrm{rev}_{n-1}(u) - \bar{\mathsf{P}}^{-1} \cdot \mathrm{rev}_{n-1}(\mathsf{s}) = \bar{\lambda} \bmod x^n$$

P is in Popov form $\Rightarrow$ $\bar{\mathsf{P}}$ is invertible mod x^n

$$\Leftrightarrow \underbrace{\begin{bmatrix} \bar{\mathsf{P}}^{-1} \cdot \mathsf{g} & -\bar{\mathsf{P}}^{-1} \cdot \mathrm{rev}_{n-1}(\mathsf{s}) \end{bmatrix}}_{\mathsf{G} \in \mathbb{K}[x]^{\alpha \times 2}} \cdot \begin{bmatrix} \mathrm{rev}_{n-1}(u) \\ 1 \end{bmatrix} = \bar{\lambda} \bmod x^n$$

Simultaneous extended Hermite-Padé approximation

Third step

Find $u(x)$ and λ such that

$$\bar{\mathsf{g}} \cdot u(x) \text{ quo } x^{n-1} = \mathsf{P} \cdot \lambda + \mathsf{s}$$

$$\Leftrightarrow \mathsf{g} \cdot \mathrm{rev}_{n-1}(u) - \mathrm{rev}_{n-1}(\mathsf{s}) = \bar{\mathsf{P}}\bar{\lambda} \bmod x^n$$

$$\Leftrightarrow \bar{\mathsf{P}}^{-1} \cdot \mathsf{g} \cdot \mathrm{rev}_{n-1}(u) - \bar{\mathsf{P}}^{-1} \cdot \mathrm{rev}_{n-1}(\mathsf{s}) = \bar{\lambda} \bmod x^n$$

P is in Popov form $\Rightarrow$ $\bar{\mathsf{P}}$ is invertible mod x^n

$$\Leftrightarrow \underbrace{\begin{bmatrix} \bar{\mathsf{P}}^{-1} \cdot \mathsf{g} & -\bar{\mathsf{P}}^{-1} \cdot \mathrm{rev}_{n-1}(\mathsf{s}) \end{bmatrix}}_{\mathsf{G} \in \mathbb{K}[x]^{\alpha \times 2}} \cdot \begin{bmatrix} \mathrm{rev}_{n-1}(u) \\ 1 \end{bmatrix} = \bar{\lambda} \bmod x^n$$

Find $u(x)$ and λ such that

$$\bar{\mathsf{g}} \cdot u(x) \text{ quo } x^{n-1} = \mathsf{P} \cdot \lambda + \mathsf{s}$$

Simultaneous extended Hermite-Padé approximation

Find $u(x)$ and λ such that

$$\mathsf{G} \cdot \begin{bmatrix} \mathrm{rev}_{n-1}(u) \\ 1 \end{bmatrix} = \bar{\lambda} \bmod x^n$$

Overview of the algorithm

And final complexity

$$\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha](\mathsf{G}, \mathsf{H}, \mathsf{v})$$

Using generators of A, find u such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

$\Updownarrow$

Find the basis of solutions

$$\{\mathsf{p} \in \mathbb{K}[x]^{\alpha \times 1} \mid \mathsf{F} \cdot \mathsf{p} = v(x) \bmod x^n\}$$

$\Updownarrow$

Find $u(x)$ and λ such that

$$\bar{\mathsf{g}} \cdot u(x) \text{ quo } x^{n-1} = \mathsf{P} \cdot \lambda + \mathsf{s}$$

$\Updownarrow$

Find $u(x)$ and λ such that

$$\underbrace{\mathsf{G}}_{\mathbb{K}[x]^{\alpha \times 2}} \cdot \begin{bmatrix} \mathsf{rev}_{n-1}(u) \\ 1 \end{bmatrix} = \bar{\lambda} \bmod x^n$$

$\Downarrow$

Computing $\mathsf{rev}_{n-1}(u)$

Overview of the algorithm

And final complexity

$$\mathsf{LinearSystem}[\mathbb{Z}_{n,0}, \mathbb{Z}^{\mathsf{T}}_{m,0}, \alpha](\mathsf{G}, \mathsf{H}, \mathsf{v})$$

Using generators of A, find u such that $\mathsf{A} \cdot \mathsf{u} = \mathsf{v}$

$\Updownarrow$

Find the basis of solutions

$$\{\mathsf{p} \in \mathbb{K}[x]^{\alpha \times 1} \mid \mathsf{F} \cdot \mathsf{p} = v(x) \bmod x^n\}$$

$\Updownarrow$ Approximant basis $\tilde{O}(\alpha^{\omega-1} \cdot n)$ Jeannerod, Neiger and Villard [2020]

Find $u(x)$ and λ such that

$$\bar{\mathsf{g}} \cdot u(x) \text{ quo } x^{n-1} = \mathsf{P} \cdot \lambda + \mathsf{s}$$

$\Updownarrow$

Find $u(x)$ and λ such that

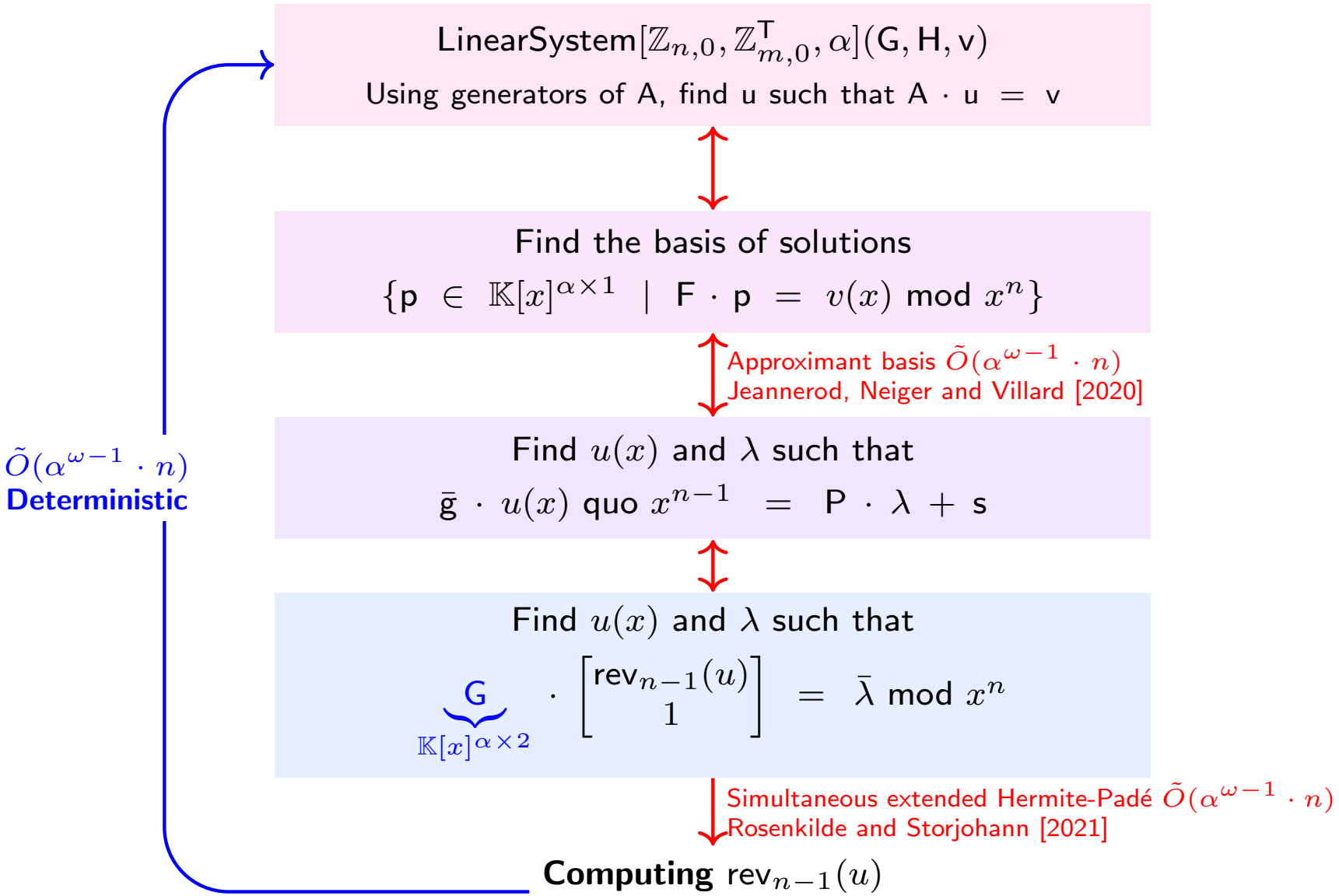
$$\underbrace{\mathsf{G}}_{\mathbb{K}[x]^{\alpha \times 2}} \cdot \begin{bmatrix} \mathsf{rev}_{n-1}(u) \\ 1 \end{bmatrix} = \bar{\lambda} \bmod x^n$$

$\Downarrow$ Simultaneous extended Hermite-Padé $\tilde{O}(\alpha^{\omega-1} \cdot n)$ Rosenkilde and Storjohann [2021]

Computing $\mathsf{rev}_{n-1}(u)$

Overview of the algorithm

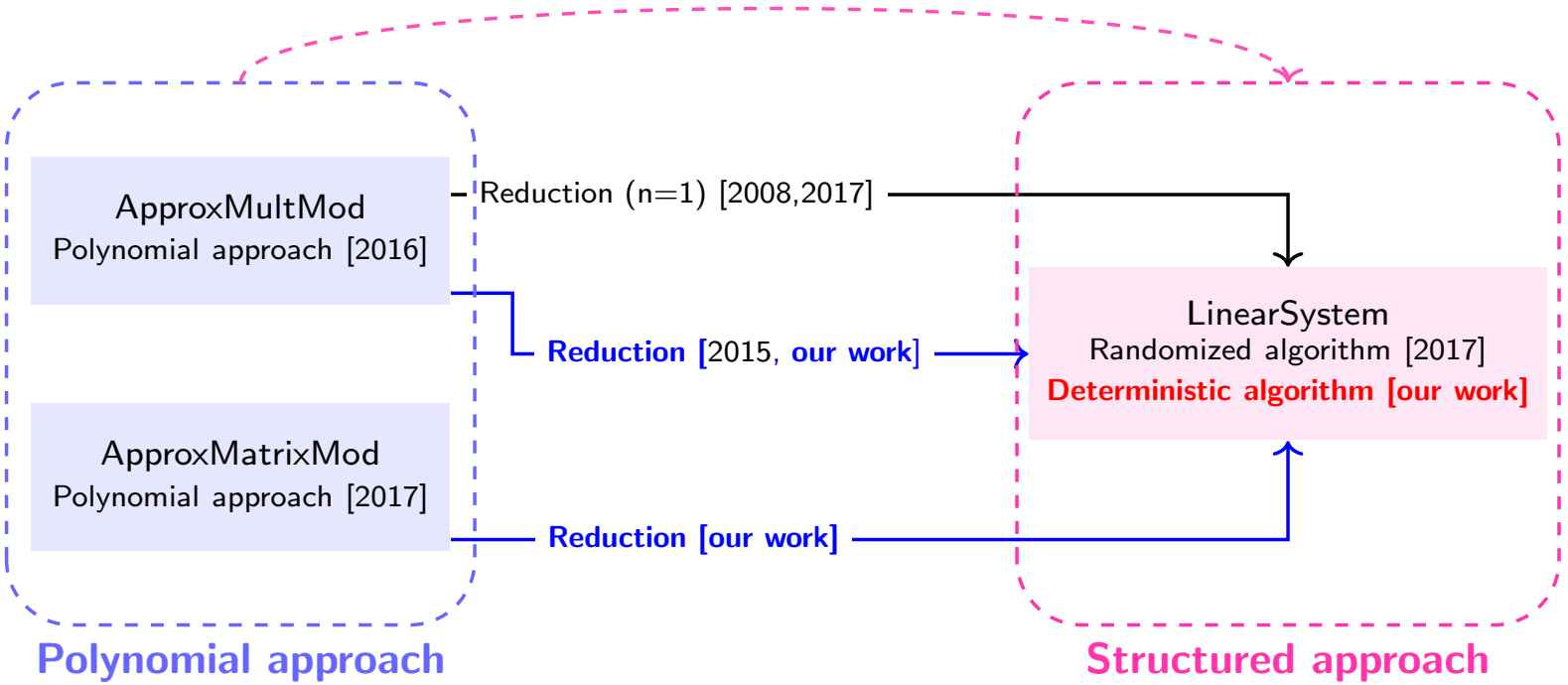
And final complexity



Conclusion

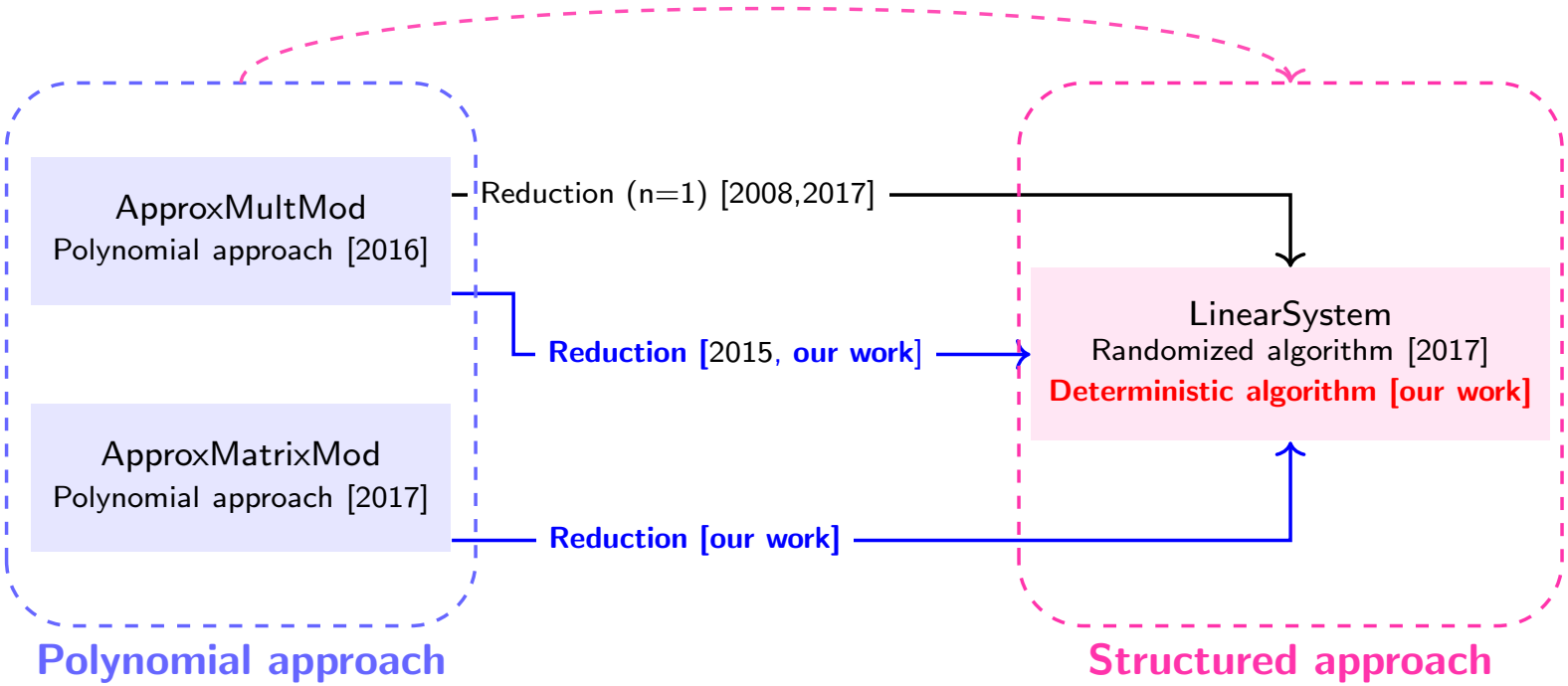
Conclusion

Summary and future work



Conclusion

Summary and future work

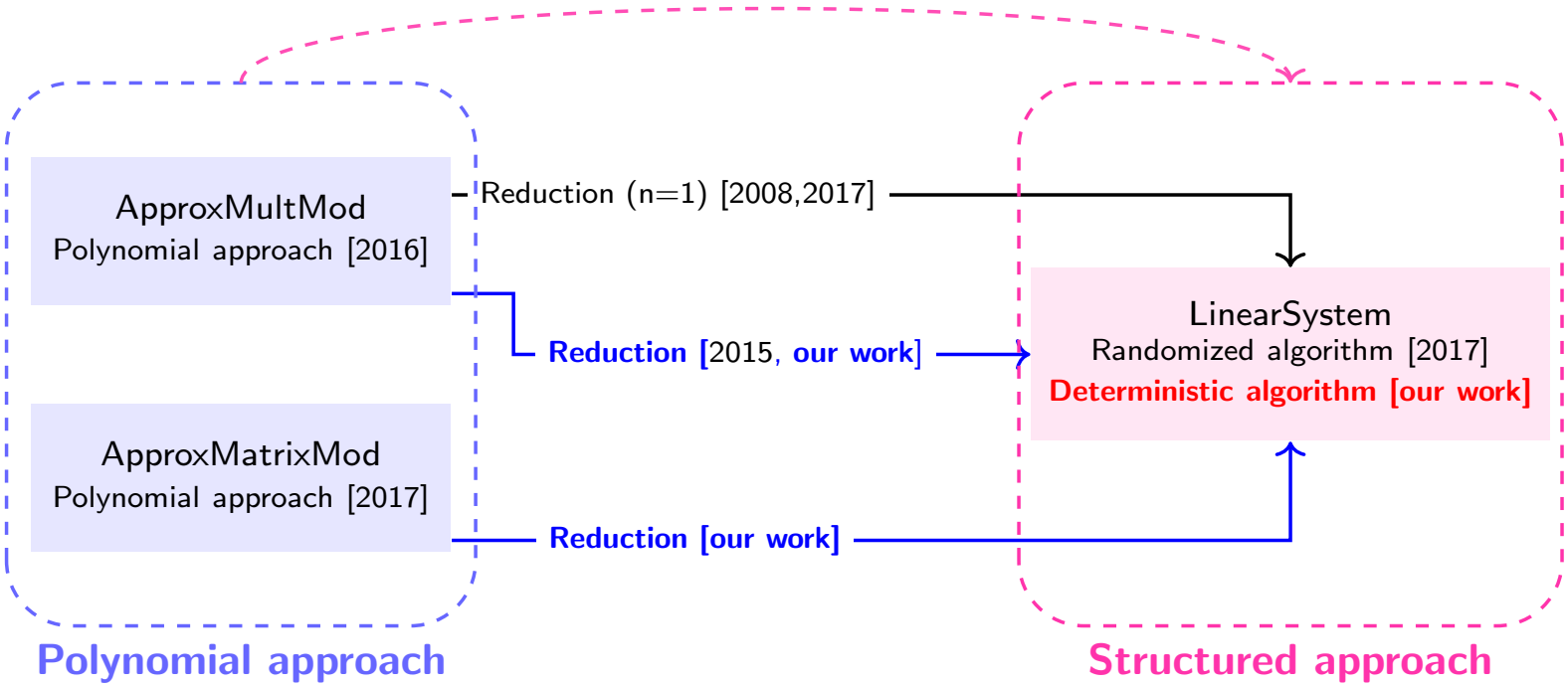


Perspectives:

- Extend our last result to other structured linear systems.
- Lift the $\tilde{O}$ notation.

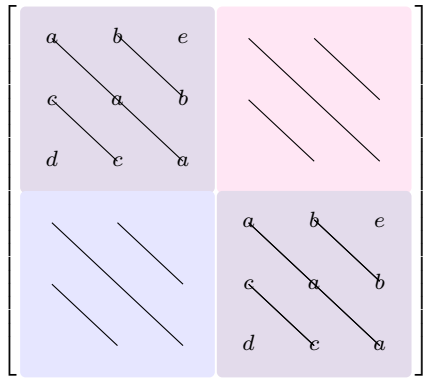
Conclusion

Summary and future work



Perspectives:

- Extend our last result to other structured linear systems.
- Lift the $\tilde{O}$ notation.
- Compute a basis of solutions for structured linear systems.
- Study cases with multiple nested structures.
  e.g., $\mathsf{F} = [1, p, p^2, \ldots, p^{m-1}]$ for some $p \in \mathbb{K}[x]$



BTTB

Acknowledgment

Thank you!

Appendix

Polynomial matrices

Modulus polynomial and row reduced

$\mathsf{M} \in \mathbb{K}[x]^{n \times n}$

Modulus polynomial:

M_{ii} monic, $\deg(\mathsf{M}_{ii}) = d_i$ $\quad \begin{bmatrix} \mathsf{M}_{11} & \mathsf{M}_{12} & \cdots & \mathsf{M}_{1n} \\ \mathsf{M}_{21} & \mathsf{M}_{22} & \cdots & \mathsf{M}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathsf{M}_{n1} & \mathsf{M}_{n2} & \cdots & \mathsf{M}_{nn} \end{bmatrix}$ $\quad \deg(\mathsf{M}_{ij}) < d_i$ for $j \neq i$

Row reduced:

A row reduced matrix is a matrix whose row leading coefficients matrix is invertible.

e.g. $\begin{bmatrix} x & x+1 \\ x & x^2 \end{bmatrix}$ $\qquad$ rlm $= \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$

Lemma

M *modulus polynomial* $\implies$ M *is row reduced.*

Proof. rlm of M is the identity matrix.

Polynomial matrices

Euclidean division

For any row reduced $\mathsf{M} \in \mathbb{K}[x]^{n\times n}$ with $\mathsf{rdeg}(\mathsf{M}) = [d_1, \ldots, d_n]$

and any $\mathsf{F} \in \mathbb{K}[x]^{n\times m}$ with $\mathsf{rdeg}(\mathsf{F}) < [d_1 + k, \ldots, d_n + k]$

Notation

$\mathsf{M} \in \mathbb{K}[x]^{n\times m}$:

- $\mathsf{rdeg}(\mathsf{M}) = [d_1, \ldots, d_n]$ such that $d_i = \max_{j=1,\ldots,m}(\deg(\mathsf{M}_{ij}))$
- $\deg(\mathsf{M}) = \max_{i=1,\ldots,n,j=1,\ldots,m}(\deg(\mathsf{M}_{ij}))$

There exists unique matrices $\mathsf{Q}, \mathsf{R} \in \mathbb{K}[x]^{n\times m}$ such that:

$$\mathsf{F} = \mathsf{M} \cdot \mathsf{Q} + \mathsf{R}$$

with $\mathsf{rdeg}(\mathsf{R}) < \mathsf{rdeg}(\mathsf{M})$ and $\deg(\mathsf{Q}) < k$.

Applications

Family of polynomials approximation:

- Block Wiedemann algorithm
- Factorization of integer numbers
- Polynomial multivariate systems solving

Polynomial matrix approximation:

- Acceleration of the computation of the characteristic polynomial
- Computations for univariate matrices

Structured linear systems:

- Error correcting codes: Reed-Solomon

Algorithms

Part 2

Algorithm 1: Coefficient $d-1$ of each of the polynomials $x^k P$ rem M for $k = 0, \ldots, \nu$

Input: $M = m_0 + m_1 x + \cdots + m_d x^d$,
$P = p_0 + p_1 x + \cdots + p_{d-1} x^{d-1}$, ν
Output: $[\mathsf{coeff}(P \text{ rem } M, d-1), \mathsf{coeff}(xP \text{ rem } M, d-1), \ldots, \mathsf{coeff}(x^{\nu-1}P \text{ rem } M, d-1)]$

$g = \mathsf{rev}_d(M) \text{ rem } x^\nu$;
$h = \mathsf{rev}_{d-1}(P) \text{ rem } x^\nu$;
$f = g^{-1} \cdot h \text{ rem } x^\nu$;
for $i = 0$ **to** $\nu - 1$ **do**
  $a_i = \mathsf{coeff}(f, i)$;
return $[a_0, \ldots, a_{\nu-1}]$;

Algorithm 2: Truncated Inverse Product

Input: $\mathsf{G} \in \mathbb{K}[x]^{n \times n}$ such that G is row reduced, $\mathsf{P} \in \mathbb{K}[x]^{n \times m}$ and $\nu = \{\nu_1, \ldots, \nu_m\}$.
Output: $\mathsf{G}^{-1} \cdot \mathsf{P} \text{ rem } \mathbb{X}^\nu$, where $\mathbb{X}^\nu = \mathsf{diag}(x^{\nu_1}, x^{\nu_2}, \ldots, x^{\nu_m})$.
Initialization: $\mathsf{P}' = \mathsf{P}$, $\nu' = \nu$, $S = \{1, \ldots, m\}$, $\ell = 1$, $C = \sum_{j=1}^m \nu_j$, $\mathsf{R} \in \mathbb{K}[x]^{n \times m}$ with $\mathsf{R} = [\mathsf{R}_{*j}]_{j=1,\ldots,m}$.
$S_\ell = \{j \in S \mid \nu_j \leq 2 \cdot \frac{C}{m}\}$; // S_1
for $\ell = 1$ **to** $\lceil \log_2(C/m) \rceil$ **do**
  $\mathsf{P}' \in \mathbb{K}[x]^{n \times |S_\ell|}$, where $\mathsf{P}' = [\mathsf{P}_{*j}]_{j \in S_\ell}$;
  $\nu' = (\nu_j)_{j \in S_\ell}$;
  $\mathbb{X}^{\nu'} \in \mathbb{K}[x]^{|S_\ell| \times |S_\ell|}$, where $\mathbb{X}^{\nu'} = \mathsf{diag}(x^{\nu_j})_{j \in S_\ell}$;
  Computing $\mathsf{B} = \mathsf{G}^{-1} \cdot \mathsf{P}' \text{ rem } \mathbb{X}^{\nu'}$;
  $i = 1$;
  for $j \in S_\ell$ **do**
    $\mathsf{R}_{*j} = \mathsf{B}_{*i}$;
    $i = i + 1$;
  $S_{\ell+1} = \{j \in S \mid 2^\ell \cdot \frac{C}{m} < \nu_j \leq 2^{\ell+1} \cdot \frac{C}{m}\}$;
return R;

Algorithms

Part 2

Algorithm 3: Solving a structured linear system

Input: $\mathsf{G} \in \mathbb{K}^{n\times\alpha}$, $\mathsf{H} \in \mathbb{K}^{m\times\alpha}$, $\mathsf{v} \in \mathbb{K}^{n\times 1}$
Output: LinearSystem$(\mathsf{G}, \mathsf{H}, \mathsf{v})$
`// Transform to polynomial matrices`
for $j = 0$ **to** $\alpha - 1$ **do**
  $f_j(x) = \sum_{i=0}^{n-1} G_{ij} x^i$;
$\mathsf{F} = [f_0(x), f_1(x), \ldots, f_{\alpha-1}(x)]$;
for $j = 0$ **to** $\alpha - 1$ **do**
  $g_j(x) = \sum_{i=0}^{m-1} H_{ij} x^i$;
$v(x) = \sum_{i=0}^{n-1} \mathsf{v}_i x^i$;
`// Approximant basis computation`
$\begin{bmatrix} \mathsf{P} & \mathsf{s} \\ 0 & \mu \end{bmatrix} = \mathsf{ApproximantBasis}(\begin{bmatrix} \mathsf{F} & -v \end{bmatrix}, n)$;
if $\deg(\mu) > 0$ **then**
  return $\emptyset$;
else
  $\deg(\mu) = 0$
$\mu(x) = 1$;
`// Retrieve the solution`
$\mathsf{B} = \begin{bmatrix} \bar{\mathsf{P}}^{-1} \cdot \mathsf{g} & -\bar{\mathsf{P}}^{-1} \cdot \mathsf{rev}_{n-1}(\mathsf{s}) \end{bmatrix}$;
return $[\mathsf{coeff}(u, \alpha-1), \ldots, \mathsf{coeff}(u, 0)]^{\mathsf{T}}$;

Algorithm 4: Family of polynomials approximation

Input: $\mathfrak{M} \in \mathbb{K}[x]^n$, $\mathsf{F} \in \mathbb{K}[x]^{n\times m}$,
$\nu = \{\nu_1, \ldots, \nu_m\}$
Output: ApproxMultMod$(\mathfrak{M}, \mathsf{F}, \nu)$
$\mathsf{A} \in \mathbb{K}^{D\times C}$;
for $i = 1$ **to** n **do**
  for $j = 1$ **to** m **do**
    $\mathsf{f}_{ij} \in \mathbb{K}^{d_i}$ is the vector of coefficients;
    $\mathsf{A}_{ij} =$
    $\begin{bmatrix} \mathsf{f}_{ij} & \mathbb{X}(M_i)\mathsf{f}_{ij} & \cdots & \mathbb{X}(M_i)^{\nu_j - 1}\mathsf{f}_{ij} \end{bmatrix}$;
Compute $\mathsf{G}, \mathsf{H} \in \mathbb{K}^{*\times O(\max(m,n))}$;
return
 LinearSystem$[\mathbb{Z}_{D,0}, \mathbb{Z}_{C,0}^{\mathsf{T}}, O(\max(m,n))](\mathsf{G}, \mathsf{H})$;

High degree coefficients

Problem definition

Theorem

The problem LastCoeffModMat$(\mathsf{M}, \mathsf{F}, \nu)$ *is solved in* $\tilde{O}(n^{\omega-1} \cdot C)$.

Proof. For M, F and k, there exist unique Q_k and R_k such that $x^k \cdot \mathsf{F} = \mathsf{M} \cdot \mathsf{Q}_k + \mathsf{R}_k$

$\overline{\mathsf{M}} = \mathbb{X}^d \cdot \mathsf{M}(x^{-1})$,
$\overline{\mathsf{F}} = \mathbb{X}^{d-1} \cdot \mathsf{F}(x^{-1})$,
$\overline{\mathsf{Q}_k} = x^{k-1} \cdot \mathsf{Q}_k(x^{-1})$,
$\overline{\mathsf{R}_k} = \mathbb{X}^{d-1} \cdot \mathsf{R}_k(x^{-1})$.

$\overline{\mathsf{F}} = \overline{\mathsf{M}} \cdot \overline{\mathsf{Q}_k} + x^k \cdot \overline{\mathsf{R}_k}$
We rewrite the identity with $k = k_j$ for each column j as:

$$\overline{\mathsf{F}_{*j}} = \overline{\mathsf{M}} \cdot \overline{\mathsf{Q}_{k_j,j}} + x^{k_j} \cdot \overline{\mathsf{R}_{k_j,j}},$$

We get $\overline{\mathsf{Q}_{k_j,j}} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*j}}$ rem x^{k_j}.

High degree coefficients

Correctness

$\overline{\mathsf{Q}_{k_j,j}} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*j}} \text{ rem } x^{k_j}$

Lemma

For each j, $\overline{\mathsf{Q}_{\nu_j,j}}$ (the quotient with $k_j = \nu_j$) contains the coefficients of degree $d_i - 1$ of $[x^{k_j} \cdot \mathsf{F} \text{ rem } \mathsf{M}]_{ij}$ for each i.

Proof. For one column j

We recall the euclidean division $x^{k_j} \cdot \mathsf{F}_{*j} = \mathsf{M} \cdot \mathsf{Q}_{k_j,j} + \mathsf{R}_{k_j,j}$

We multipliy by x and get $x^{k_j+1} \cdot \mathsf{F}_{*j} = \mathsf{M} \cdot (x \cdot \mathsf{Q}_{k_j,j}) + (x \cdot \mathsf{R}_{k_j,j})$

Note that $x^{k_j+1} \cdot \mathsf{F}_{*j} =$

$$\mathsf{M} \cdot (x \cdot \mathsf{Q}_{k_j,j}) + \left(\begin{bmatrix} \text{coeff}([x^{k_j} \cdot \mathsf{F} \text{ rem } \mathsf{M}]_{1j}, d_1 - 1) \\ \vdots \\ \text{coeff}([x^{k_j} \cdot \mathsf{F} \text{ rem } \mathsf{M}]_{nj}, d_n - 1) \end{bmatrix} \cdot \begin{bmatrix} x^{d_1} & 0 & \cdots & 0 \\ 0 & x^{d_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x^{d_n} \end{bmatrix} + \mathsf{R}'_{k_j,j} \right)$$

For ease of notation, we denote $x^{k_j+1} \cdot \mathsf{F}_{*j} = \mathsf{M} \cdot (x \cdot \mathsf{Q}_{k_j,j}) + (\mathsf{a}_{k_j,j} \cdot \mathbb{X}^d + \mathsf{R}'_{k_j,j})$

Since M is modulus polynomial, we can write $\mathsf{M} = \mathbb{X}^d - \mathsf{M}'$ and get:

$$x^{k_j+1} \cdot \mathsf{F}_{*j} = \mathsf{M} \cdot (x \cdot \mathsf{Q}_{k_j,j} + \mathsf{a}_{k_j,j}) + (\mathsf{a}_{k_j,j} \cdot \mathsf{M}' + x \cdot \mathbb{R}'_{k_j,j})$$

High degree coefficients

Correctness

$\overline{\mathsf{Q}_{k_j,j}} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*j}} \text{ rem } x^{k_j}$

Lemma

For each j, $\overline{\mathsf{Q}_{\nu_j,j}}$ (the quotient with $k_j = \nu_j$) contains the coefficients of degree $d_i - 1$ of $[x^{k_j} \cdot \mathsf{F} \text{ rem } \mathsf{M}]_{ij}$ for each i.

Proof. For one column j

$$x^{k_j+1} \cdot \mathsf{F}_{*j} = \mathsf{M} \cdot (x \cdot \mathsf{Q}_{k_j,j} + \mathsf{a}_{k_j,j}) + (\mathsf{a}_{k_j,j} \cdot \mathsf{M}' + x \cdot \mathbb{R}'_{k_j,j})$$

By uniqueness of the quotient and remainder, we have:

- $\mathsf{Q}_{k_j+1,j} = x \cdot \mathsf{Q}_{k_j,j} + \mathsf{a}_{k_j,j}$,
- $\mathsf{R}_{k_j+1,j} = \mathsf{a}_{k_j,j} \cdot \mathsf{M}' + x \cdot \mathbb{R}'_{k_j,j}$.

High degree coefficients

Correctness

$\overline{\mathsf{Q}_{k_j,j}} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*j}} \text{ rem } x^{k_j}$

Lemma

For each j, $\overline{\mathsf{Q}_{\nu_j,j}}$ (the quotient with $k_j = \nu_j$) contains the coefficients of degree $d_i - 1$ of $[x^{k_j} \cdot \mathsf{F} \text{ rem } \mathsf{M}]_{ij}$ for each i.

Proof. For one column j
We focus on the quotient:

$$\mathsf{Q}_{k_j+1,j} = x \cdot \mathsf{Q}_{k_j,j} + \mathsf{a}_{k_j,j}$$

and enroll the recurrence:

$$\begin{aligned}
\overline{\mathsf{Q}_{0,j}} &= 0, \\
\overline{\mathsf{Q}_{1,j}} &= x^0 \cdot \mathsf{a}_{0,j} + 0 = \mathsf{a}_{0,j}, \\
\overline{\mathsf{Q}_{2,j}} &= x^1 \cdot a_{0,j} + \mathsf{a}_{1,j} = x^1 \cdot \mathsf{a}_{0,j} + \mathsf{a}_{1,j}, \\
&\vdots \\
\overline{\mathsf{Q}_{\nu_j,j}} &= x^{\nu_j-1} \cdot \mathsf{a}_{0,j} + x^{\nu_j-2} \cdot \mathsf{a}_{1,j} + \cdots + x^0 \cdot \mathsf{a}_{\nu_j-1,j} \ .
\end{aligned}$$

High degree coefficients

Complexity

$\overline{\mathsf{Q}_{\nu_j,j}} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*j}} \text{ rem } x^{\nu_j}$

Lemma

We can compute $\overline{\mathsf{Q}_\nu}$ in $\tilde{O}(n^{\omega-1} \cdot C)$.

Proof.

We define $\overline{\mathsf{Q}_\nu} \in \mathbb{K}[x]^{n \times m}$ as $\left[\overline{\mathsf{Q}_{\nu_1,1}} \quad \overline{\mathsf{Q}_{\nu_2,2}} \quad \cdots \quad \overline{\mathsf{Q}_{\nu_m,m}}\right]$

Since we can verify that

$$\begin{aligned}\overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}} \text{ rrem } \mathbb{X}^\nu &= \left[\overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*1}} \text{ rem } x^{\nu_1} \quad \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*2}} \text{ rem } x^{\nu_2} \quad \cdots \quad \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*m}} \text{ rem } x^{\nu_m}\right] \\ &= \left[\overline{\mathsf{Q}_{\nu_1,1}} \quad \overline{\mathsf{Q}_{\nu_2,2}} \quad \cdots \quad \overline{\mathsf{Q}_{\nu_m,m}}\right],\end{aligned}$$

it is equivalent to compute $\overline{\mathsf{Q}_\nu} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}} \text{ rem } \mathbb{X}^\nu$, where $\begin{bmatrix} x^{\nu_1} & 0 & \cdots & 0 \\ 0 & x^{\nu_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x^{\nu_m} \end{bmatrix}$

High degree coefficients

Complexity

$\overline{\mathsf{Q}_{\nu_j,j}} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*j}} \text{ rem } x^{\nu_j}$

Lemma

We can compute $\overline{\mathsf{Q}_\nu}$ in $\tilde{O}(n^{\omega-1} \cdot C)$.

Proof.

We compute $\overline{\mathsf{Q}_\nu} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}} \text{ rem } \mathbb{X}^\nu$.

- State of the art: the problem is solved under the assumption that $\max(\nu) \leq 2 \cdot \frac{C}{m}$.

Applying the algorithm on a selected set of columns of $\overline{\mathsf{F}}$:

i.e. For a given step ℓ of the algorithm, we have:

- $L_\ell = \{j \in \{1, \ldots, m\} \mid \nu_j \leq 2^\ell \cdot \frac{C}{m}\}$
- $\mathsf{F}' = [\mathsf{P}_{*j}]_{j \in L_\ell} \in \mathbb{K}[x]^{n \times |L_\ell|}$.
- $\nu' = (\nu_j)_{j \in L_\ell}$, we get $\mathbb{X}^{\nu'} \in \mathbb{K}[x]^{\frac{m}{2^\ell} \times \frac{m}{2^\ell}}$.

Number of steps: $\lceil \log_2(m) \rceil$.
Cost of each step: $\tilde{O}(n^{\omega-1} \cdot C)$,
(since $|L_\ell| \geq \frac{m}{2^\ell}$).
Total cost:
$\tilde{O}(n^{\omega-1} \cdot C \cdot \log_2(m)) = \tilde{O}(n^{\omega-1} \cdot C)$.

□

High degree coefficients

Summary

$\overline{\mathsf{Q}_{\nu_j,j}} = \overline{\mathsf{M}}^{-1} \cdot \overline{\mathsf{F}_{*j}} \text{ rem } x^{\nu_j}$

Correctness:

Lemma

For each j, $\overline{\mathsf{Q}_{\nu_j,j}}$ contains the coefficients of degree $d_i - 1$ of $[x^{k_j} \cdot \mathsf{F} \text{ rem } \mathsf{M}]_{ij}$ for each i.

Complexity:

Lemma

We can compute $\overline{\mathsf{Q}_\nu}$ in $\tilde{O}(n^{\omega-1} \cdot C)$.

Final result:

Theorem

The problem LastCoeffModMat$(\mathsf{M}, \mathsf{F}, \nu)$ *is solved in $\tilde{O}(n^{\omega-1} \cdot C)$.*

Notes

Berlekamp-Massey:
Finds the minimal recurrence (of order d) from the first $2d$ terms of a linearly recurrent sequence, and this is equivalent to compute a basis of relations for $F = [S, -1]$ modulo x^{2d}, where the input S is the polynomial defined from the $2d$ first terms of the sequence

- Dornstetter 1987 and Sugiyama et al. 1975: quadratic
- Brent-Gustavson-Yun 1980: quasi-linear

Randomization in current structured linear system algorithms:
The pivots in a Gaussian elimination-type of algorithm must be on the diagonal, and authors ensure this property by multiplying the input matrix by a random choice of matrices (with a special form for efficiency).