Vincent Neiger

Laboratoire LIP6, Sorbonne Université

vincent.neiger@lip6.fr

polynomial matrices: introduction, motivations, and basic algorithms

Algorithmes Efficaces en Calcul Formel
Master Parisien de Recherche en Informatique
24 November 2024

outline

- introduction
- matrices? polynomials?
- polynomial matrices
- reduced forms

outline

- **introduction**
  - definitions and algebraic properties
  - examples you already know
  - three flagship applications
- **matrices? polynomials?**
- **polynomial matrices**
- **reduced forms**

polynomial matrices – introduction

definitions and algebraic properties

- working over a base field $\mathbb{K}$

$\mathbb{K} =$ finite field $\mathbf{F}_q$, extension $\mathbf{F}_q[X]/\langle f(X)\rangle$, rational numbers $\mathbb{Q}$, …

- considering polynomials in one indeterminate X

$\mathbb{K}[X]$ is a principal ideal domain (what does that mean?)

- in $\mathbb{K}[X]$, many operations cost $O(\mathsf{M}(d))$ or $O(\mathsf{M}(d)\log(d))$ field ops.

where $d \mapsto \mathsf{M}(d)$ is a cost function for polynomial multiplication in degree d

polynomial matrices – introduction

definitions and algebraic properties

- working over a base field $\mathbb{K}$

$\mathbb{K} =$ finite field $\mathbf{F}_q$, extension $\mathbf{F}_q[X]/\langle f(X)\rangle$, rational numbers $\mathbb{Q}$, ...

- considering polynomials in one indeterminate X

$\mathbb{K}[X]$ is a principal ideal domain (what does that mean?)

- in $\mathbb{K}[X]$, many operations cost $O(M(d))$ or $O(M(d)\log(d))$ field ops.

where $d \mapsto M(d)$ is a cost function for polynomial multiplication in degree d

- addition $f+g$, multiplication $f*g$
- division with remainder $f = qg + r$
- truncated inverse $f^{-1} \bmod X^d$
- extended GCD $uf + vg = \gcd(f, g)$
- multipoint eval. $f \mapsto f(x_1), \ldots, f(x_d)$
- interpolation $f(x_1), \ldots, f(x_d) \mapsto f$
- Padé approximation $f = \frac{p}{q} \bmod X^d$
- minpoly of linearly recurrent sequence

polynomial matrices – introduction

definitions and algebraic properties

- working over a base field $\mathbb{K}$

$\mathbb{K}$ = finite field $\mathbf{F}_q$, extension $\mathbf{F}_q[X]/\langle f(X)\rangle$, rational numbers $\mathbb{Q}$, ...

- considering polynomials in one indeterminate X

$\mathbb{K}[X]$ is a principal ideal domain (what does that mean?)

- in $\mathbb{K}[X]$, many operations cost $O(M(d))$ or $O(M(d)\log(d))$ field ops.

where $d \mapsto M(d)$ is a cost function for polynomial multiplication in degree d

$O(M(d))$	$O(M(d)\log(d))$
▸ addition $f+g$, multiplication $f*g$	▸ multipoint eval. $f \mapsto f(x_1), \ldots, f(x_d)$
▸ division with remainder $f = qg + r$	▸ interpolation $f(x_1), \ldots, f(x_d) \mapsto f$
▸ truncated inverse $f^{-1} \bmod X^d$	▸ Padé approximation $f = \frac{p}{q} \bmod X^d$
▸ extended GCD $uf + vg = \gcd(f, g)$	▸ minpoly of linearly recurrent sequence

$\mathbb{K}[X]^{m\times n}$ = set of $m \times n$ matrices over $\mathbb{K}[X]$

called polynomial matrices in what follows

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

polynomial matrices – introduction

definitions and algebraic properties

$$\mathbb{K}[X]^{m\times n} = \text{set of } m \times n \text{ matrices over } \mathbb{K}[X]$$

called polynomial matrices in what follows

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

- structure: matrices over $\mathbb{K}[X] \longleftrightarrow$ "free" modules over $\mathbb{K}[X]$

similarly to: matrices over $\mathbb{K} \longleftrightarrow$ vector spaces over $\mathbb{K}$

- basic operations: addition and multiplication

defined as usual (multiplication requires compatible dimensions)

- $\mathbb{K}[X]$ is not a field

what does this change? what operations are allowed / not allowed?

polynomial matrices – introduction

definitions and algebraic properties

$$\mathbb{K}[X]^{m \times n} = \text{set of } m \times n \text{ matrices over } \mathbb{K}[X]$$

called polynomial matrices in what follows

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

- structure: matrices over $\mathbb{K}[X] \longleftrightarrow$ "free" modules over $\mathbb{K}[X]$

similarly to: matrices over $\mathbb{K} \longleftrightarrow$ vector spaces over $\mathbb{K}$

- basic operations: addition and multiplication

defined as usual (multiplication requires compatible dimensions)

- $\mathbb{K}[X]$ is not a field

what does this change? what operations are allowed / not allowed?

$\rightsquigarrow$ algorithms may work in $\mathbb{K}(X)^{m \times n}$, but be careful with "degree explosion"!

(exercise: Gaussian elimination is exponential-time)

polynomial matrices – introduction

examples you already know

polynomial matrices – introduction

examples you already know

large matrices with small degrees:
characteristic polynomial $\det(X\mathbf{I}_m - \mathbf{M}) \in \mathbb{K}[X]$ of a matrix $\mathbf{M} \in \mathbb{K}^{m\times m}$
$\rightsquigarrow$ determinant of polynomial matrix $X\mathbf{I}_m - \mathbf{M} \in \mathbb{K}[X]^{m\times m}$

- fastest known algorithm uses this viewpoint [N.-Pernet, 2021]
- gradually transforms $X\mathbf{I}_m - \mathbf{M}$ to smaller matrices with larger degrees

polynomial matrices – introduction

examples you already know

large matrices with small degrees:
characteristic polynomial $\det(X\mathbf{I}_m - \mathbf{M}) \in \mathbb{K}[X]$ of a matrix $\mathbf{M} \in \mathbb{K}^{m \times m}$
$\rightsquigarrow$ determinant of polynomial matrix $X\mathbf{I}_m - \mathbf{M} \in \mathbb{K}[X]^{m \times m}$

- fastest known algorithm uses this viewpoint [N.-Pernet, 2021]
- gradually transforms $X\mathbf{I}_m - \mathbf{M}$ to smaller matrices with larger degrees

small matrices with large degree:
extended GCD $uf + vg = \gcd(f, g)$ for polynomials $f, g \in \mathbb{K}[X]_{\leqslant d}$
$\rightsquigarrow$ corresponds to a polynomial matrix transformation

$$\begin{bmatrix} u & v \\ \tilde{g} & \tilde{f} \end{bmatrix} \begin{bmatrix} f \\ g \end{bmatrix} = \begin{bmatrix} \gcd(f, g) \\ 0 \end{bmatrix}$$

with the leftmost (polynomial) matrix of determinant in $\mathbb{K} \setminus \{0\}$

- fastest known "half-gcd" algorithms use this viewpoint
[Knuth, 1970] [Schönhage, 1971] [Brent-Gustavson-Yun, 1980]

polynomial matrices – introduction

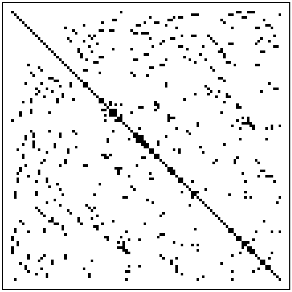
three flagship applications

1. operations on sparse matrices

- solving sparse linear systems over $\mathbb{K}$
- computing the minimal polynomial / Frobenius form
- introducing parallelism in these computations

[Wiedemann 1986]
[Coppersmith 1993]
[Villard 1997]

example of sparse matrix in $\mathbb{K}^{m \times m}$
typical case: $O(m)$ nonzero entries



uses **polynomial matrix** generator
of linearly recurrent **matrix** sequence

polynomial matrices – introduction

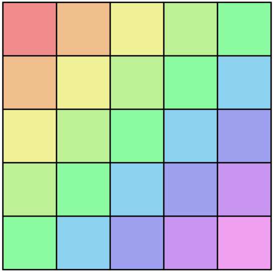
three flagship applications

2. operations on structured matrices

- matrix-vector multiplication
- linear system solving
- nullspace computation

[Kailath-Kung-Morf 1979]
[Bostan et al. 2017]

example of Hankel matrix
$\rightsquigarrow$ block-Hankel matrices
$\rightsquigarrow$ Hankel-like matrices



uses **polynomial matrix** multiplication and
matrix-Padé approximation / **matrix**-GCD

polynomial matrices – introduction

three flagship applications

3. bivariate interpolation and multipoint evaluation

problem: given points $(\alpha_1, \beta_1), \dots, (\alpha_n, \beta_n)$ in $\mathbb{K}^2$,

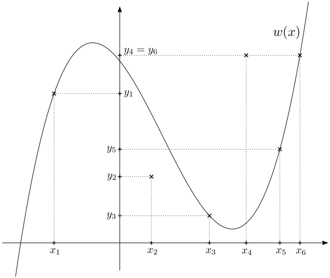
- given $p(x, y)$, compute $p(\alpha_i, \beta_i)$ for $1 \leqslant i \leqslant n$
- find $p(x, y)$ of small degree such that $p(\alpha_i, \beta_i) = 0$

[Nüsken-Ziegler 2004]

[Beckermann 1992] [van Barel-Bultheel 1992]
[Marinari-Möller-Mora 1993]

bivariate interpolation = main step
in Reed-Solomon list-decoding
(univariate interpolation with errors)

[Guruswami-Sudan 1999] [Kötter-Vardy 2003]



uses **polynomial matrix** multiplication and
matrix rational reconstruction / **algebraic approximants**

outline

- **introduction**
  - definitions and algebraic properties
  - examples you already know
  - three flagship applications
- **matrices? polynomials?**
- **polynomial matrices**
- **reduced forms**

outline

- introduction
  - definitions and algebraic properties
  - examples you already know
  - three flagship applications
- matrices? polynomials?
  - using matrix arithmetic
  - using polynomial arithmetic
  - limitations of these viewpoints
- polynomial matrices
- reduced forms

the matrix and the polynomial viewpoints

using matrix arithmetic

matrices in $\mathbb{K}[X]^{m \times n}$ are also in $\mathbb{K}(X)^{m \times n}$

(and $\mathbb{K}(X)$ is a field)

$\Rightarrow$ usual definition of addition, multiplication, determinant
these do not involve fractions anyway (... in algorithms?)

$\Rightarrow$ usual definition of inverse
but with inverse over $\mathbb{K}(X)$

$\Rightarrow$ usual definition of rank
... which one, by the way?

the matrix and the polynomial viewpoints

using matrix arithmetic

matrices in $\mathbb{K}[X]^{m \times n}$ are also in $\mathbb{K}(X)^{m \times n}$

(and $\mathbb{K}(X)$ is a field)

this point of view is hardly usable for algorithms:
it easily yields "garbage" cost bounds
e.g. addition in $\mathbb{K}[X]^{m \times n}$ costs mn additions... in $\mathbb{K}(X)$

- what is the cost of naive addition in $\mathbb{K}[X]^{m \times m}$?
- what is the cost of naive multiplication in $\mathbb{K}[X]^{m \times m}$?
- let $2 < \omega < 3$ be such that we can multiply two $m \times m$ matrices over a commutative ring in $O(m^{\omega})$ ring operations: what do you deduce about the cost of multiplying two matrices in $\mathbb{K}[X]^{m \times m}$?

the matrix and the polynomial viewpoints

using matrix arithmetic

matrices in $\mathbb{K}[X]^{m\times n}$ are also in $\mathbb{K}(X)^{m\times n}$

(and $\mathbb{K}(X)$ is a field)

this point of view is hardly usable for algorithms:
it easily yields "garbage" cost bounds
e.g. addition in $\mathbb{K}[X]^{m\times n}$ costs mn additions... in $\mathbb{K}(X)$

- what is the cost of naive addition in $\mathbb{K}[X]^{m\times m}$?
- what is the cost of naive multiplication in $\mathbb{K}[X]^{m\times m}$?
- let $2 < \omega < 3$ be such that we can multiply two $m \times m$ matrices over a commutative ring in $O(m^\omega)$ ring operations: what do you deduce about the cost of multiplying two matrices in $\mathbb{K}[X]^{m\times m}$?

for algorithms&complexity, considering the degrees of entries is essential

the matrix and the polynomial viewpoints

using matrix arithmetic

matrices in $\mathbb{K}[X]^{m \times n}$ are also in $\mathbb{K}(X)^{m \times n}$

(and $\mathbb{K}(X)$ is a field)

exercise: matrix equation $\mathbf{AU} = \mathbf{V}$

let $\mathbf{A} \in \mathbb{K}[X]^{m \times m}$ be nonsingular with all entries of degree $\leqslant d_1$

let $\mathbf{V} \in \mathbb{K}[X]^{m \times k}$ with all entries of degree $\leqslant d_2$

- show that $\mathbf{A}^{-1}\mathbf{V}$ can be represented as a fraction with numerator a matrix $\mathbf{U}$ in $\mathbb{K}[X]^{m \times k}$ and denominator a polynomial Δ in $\mathbb{K}[X]$
- give an upper bound on $\deg \det(\mathbf{A})$
- give an upper bound on $\deg(\Delta)$ and on the degrees of entries of $\mathbf{U}$
- prove that $\mathbf{A}^{-1} \in \mathbb{K}[X]^{m \times m} \Leftrightarrow \det(\mathbf{A}) \in \mathbb{K} \setminus \{0\}$

matrices with determinant in $\mathbb{K} \setminus \{0\}$ are called **unimodular**

the matrix and the polynomial viewpoints

using polynomial arithmetic

$\mathbb{K}[X]^{m\times n}$ is isomorphic to $\mathbb{K}^{m\times n}[X]$

(as $\mathbb{K}[X]$-modules)

$$\begin{bmatrix} 3X+4 & X^3+4X+1 & 4X^2+3 \\ 5 & 5X^2+3X+1 & 5X+3 \\ 3X^3+X^2+5X+3 & 6X+5 & 2X+1 \end{bmatrix}$$

$$= \begin{bmatrix} 4 & 1 & 3 \\ 5 & 1 & 3 \\ 3 & 5 & 1 \end{bmatrix} + \begin{bmatrix} 3 & 4 & 0 \\ 0 & 3 & 5 \\ 5 & 6 & 2 \end{bmatrix} X + \begin{bmatrix} 0 & 0 & 4 \\ 0 & 5 & 0 \\ 1 & 0 & 0 \end{bmatrix} X^2 + \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 3 & 0 & 0 \end{bmatrix} X^3$$

the matrix and the polynomial viewpoints

using polynomial arithmetic

$\mathbb{K}[X]^{m \times n}$ **is isomorphic to** $\mathbb{K}^{m \times n}[X]$

(as $\mathbb{K}[X]$-modules)

- natural notion of **degree** of a polynomial matrix
- **addition** of $\mathbf{A}, \mathbf{B} \in \mathbb{K}[X]^{m \times n}$ is in $O(mnd)$ operations in $\mathbb{K}$
where $d = \min(\deg(\mathbf{A}), \deg(\mathbf{B}))$
- **some** other polynomial operations available:
truncation $\mathbf{A}$ rem X^N, shift $X^d \mathbf{A}$, evaluation $\mathbf{A}(\alpha)$
what is the complexity of evaluation?
what about Lagrange interpolation?

the matrix and the polynomial viewpoints

using polynomial arithmetic

$\mathbb{K}[X]^{m \times n}$ **is isomorphic to** $\mathbb{K}^{m \times n}[X]$

(as $\mathbb{K}[X]$-modules)

when $m = n$, $\mathbb{K}^{m \times m}$ is a (non-commutative) ring

- **multiplication** in $\mathbb{K}[X]^{m \times m}$ seen as a product of polynomials
complexity?

- **truncated inversion** via power series & Newton iteration
condition for $\mathbf{A}$ to be invertible as a power series? complexity?

- fast **Euclidean division with remainder**
does this make any sense?

multiplication

On fast multiplication of polynomials over arbitrary algebras

David G. Cantor[1] **and Erich Kaltofen**[2]★

[1] Department of Mathematics, University of California, Los Angeles, CA 90024-1555, USA
[2] Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180-3590, USA

Received January 22, 1988 / May 10, 1991

1 Introduction

In this paper we generalize the well-known Schönhage-Strassen algorithm for multiplying large integers to an algorithm for multiplying polynomials with coefficients from an arbitrary, not necessarily commutative, not necessarily associative, algebra $\mathscr{A}$. Our main result is an algorithm to multiply polynomials of degree $<n$ in $O(n \log n)$ algebra multiplications and $O(n \log n \log\log n)$ algebra additions/subtractions (we count a subtraction as an addition). The constant implied by the "O" does not depend upon the algebra $\mathscr{A}$. The parallel complexity of our algorithm, i.e., the depth of the corresponding arithmetic circuit, is

the matrix and the polynomial viewpoints

using polynomial arithmetic

multiplication

On fast multiplication of polynomials over arbitrary algebras

David G. Cantor[1] and Erich Kaltofen[2] ⋆

[1] Department of Mathematics, University of California, Los Angeles, CA 90024-1555, USA
[2] Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180-3590, USA

Received January 22, 1988 / May 10, 1991

multiplication in $\mathbb{K}^{m\times m}[X]$ with degree $\leqslant d$:
- $O(d\log(d))$ multiplications in $\mathbb{K}^{m\times m}$
- $O(d\log(d)\log\log(d))$ additions in $\mathbb{K}^{m\times m}$

$\mathsf{MM}(m,d) \in O(m^{\omega} d\log(d) + m^2 d\log(d)\log\log(d))$

1 Introduction

In this paper we generalize the well-known Schönhage-Strassen algorithm for multiplying large integers to an algorithm for multiplying polynomials with coefficients from an arbitrary, not necessarily commutative, not necessarily associative, algebra $\mathscr{A}$. Our main result is an algorithm to multiply polynomials of degree $<n$ in $O(n\log n)$ algebra multiplications and $O(n\log n\log\log n)$ algebra additions/subtractions (we count a subtraction as an addition). The constant implied by the "O" does not depend upon the algebra $\mathscr{A}$. The parallel complexity of our algorithm, i.e., the depth of the corresponding arithmetic circuit, is

truncated inversion – reminder from October 28 & from AECF

Details on power series inversion

Algorithm (series inversion by Newton iteration)

Input: Truncation T to order $N \in \mathbb{N}_{>0}$ of a series $F \in \mathbb{K}[[x]]$ with $F(0) \neq 0$.

Output: The truncation S to order N of the inverse series F^{-1}.

If $N = 1$, return $T(0)^{-1}$. Otherwise:

1. Recursively compute the truncation G to order $\lceil N/2 \rceil$ of T^{-1}.
2. Return $S := G + \operatorname{rem}((1 - GT)G, x^N)$.

Correctness proof Assume $T^{-1} = G + O(x^{\lceil N/2 \rceil})$ by induction. By Lemma,

$$\mathcal{N}(G) - T^{-1} = O(x^{2\lceil N/2 \rceil}) = O(x^N).$$

Write $F = T + O(x^N) = T(1 + O(x^N))$ to observe $F^{-1} = T^{-1} + O(x^N)$. Then,

$$F^{-1} - S = (F^{-1} - T^{-1}) + (T^{-1} - \mathcal{N}(G)) + (\mathcal{N}(G) - S) = O(x^N).$$

truncated inversion – reminder from October 28 & from AECF

Entrée Un entier $N > 0$, $F \bmod X^N$ une série tronquée.
Sortie $F^{-1} \bmod X^N$.
Si $N = 1$, alors renvoyer f_0^{-1}, où $f_0 = F(0)$.
Sinon :
1. Calculer récursivement l'inverse G de F mod $X^{\lceil N/2 \rceil}$.
2. Renvoyer $G + (1 - GF)G \bmod X^N$.

Algorithme 3.2 – Inverse de série par itération de Newton.

Convergence quadratique pour l'inverse d'une série formelle

Lemme 3.2 Soient $\mathbb{A}$ un anneau non nécessairement commutatif, $F \in \mathbb{A}[[X]]$ une série formelle de terme constant inversible et G une série telle que $G - F^{-1} = O(X^n)$ ($n \geq 1$). Alors la série

$$\mathcal{N}(G) = G + (1 - GF)G \tag{3.2}$$

vérifie $\mathcal{N}(G) - F^{-1} = O(X^{2n})$.

Démonstration. Par hypothèse, on peut définir $H \in \mathbb{A}[[X]]$ par $1 - GF = X^n H$. Il suffit alors de récrire $F = G^{-1}(1 - X^n H)$ et d'inverser :

$$F^{-1} = (1 + X^n H + O(X^{2n}))G = G + X^n HG + O(X^{2n})G = \mathcal{N}(G) + O(X^{2n}). \qquad \blacksquare$$

Algorithme

Lemme 3.3 L'Algorithme 3.2 d'inversion est correct.

Démonstration. La preuve est une récurrence sur les entiers. Pour N = 1 la propriété est claire. Pour $N \geq 2$, si la propriété est vraie jusqu'à l'ordre $N - 1$, alors elle l'est pour

truncated inversion – conclusion

consider a (square) polynomial matrix $\mathbf{A} \in \mathbb{K}[X]^{m\times m}$

- $\mathbf{A}$ is invertible as a power series
  $\Leftrightarrow$ its constant term $\mathbf{A}(0) \in \mathbb{K}^{m\times m}$ is invertible

- if $\mathbf{A}$ is invertible as a power series,

computing its truncated inverse $\mathbf{A}^{-1} \bmod X^N$ costs

$$O(\mathrm{MM}(m, N)) \in O(m^\omega N \log(N) + m^2 N \log(N) \log\log(N))$$

operations in $\mathbb{K}$

division with remainder – reminder from October 28

Euclidean division for polynomials

[Strassen, 1973]

Pb: Given $F, G \in \mathbb{K}[x]_{\leq N}$, compute (Q, R) in Euclidean division $F = QG + R$

Naive algorithm: $O(N^2)$

Idea: look at $F = QG + R$ from infinity: $Q \sim_{+\infty} F/G$

Let $N = \deg(F)$ and $n = \deg(G)$. Then $\deg(Q) = N - n$, $\deg(R) < n$ and

$$\underbrace{F(1/x)x^N}_{\operatorname{rev}(F)} = \underbrace{G(1/x)x^n}_{\operatorname{rev}(G)} \cdot \underbrace{Q(1/x)x^{N-n}}_{\operatorname{rev}(Q)} + \underbrace{R(1/x)x^{\deg(R)}}_{\operatorname{rev}(R)} \cdot x^{N-\deg(R)}$$

Algorithm:

- Compute $\operatorname{rev}(Q) = \operatorname{rev}(F)/\operatorname{rev}(G) \mod x^{N-n+1}$ $O(\mathsf{M}(N))$
- Recover Q $O(1)$
- Deduce $R = F - QG$ $O(\mathsf{M}(N))$

division with remainder

problem:
given $\mathbf{A}, \mathbf{B} \in \mathbb{K}^{m \times m}[X]$,
compute $\mathbf{Q}, \mathbf{R} \in \mathbb{K}^{m \times m}[X]$ such that

$$\mathbf{A} = \mathbf{B}\mathbf{Q} + \mathbf{R} \quad \text{and} \quad \deg(\mathbf{R}) < \deg(\mathbf{B})$$

. . . are we not missing an assumption?

division with remainder

problem:
given $\mathbf{A}, \mathbf{B} \in \mathbb{K}^{m \times m}[X]$,
compute $\mathbf{Q}, \mathbf{R} \in \mathbb{K}^{m \times m}[X]$ such that

$$\mathbf{A} = \mathbf{B}\mathbf{Q} + \mathbf{R} \quad \text{and} \quad \deg(\mathbf{R}) < \deg(\mathbf{B})$$

... are we not missing an assumption?

rule 1: dividing by zero is generally a bad idea
rule 2: if you think you need to divide by zero, refer to rule 1
rule 3: neglecting to check that something is not zero does not make it nonzero
etc. etc.

the matrix and the polynomial viewpoints

using polynomial arithmetic

division with remainder

problem:
given $\mathbf{A}, \mathbf{B} \in \mathbb{K}^{m \times m}[X]$,
compute $\mathbf{Q}, \mathbf{R} \in \mathbb{K}^{m \times m}[X]$ such that

$$\mathbf{A} = \mathbf{B}\mathbf{Q} + \mathbf{R} \quad \text{and} \quad \deg(\mathbf{R}) < \deg(\mathbf{B})$$

... are we not missing an assumption?

for a polynomial $p \in \mathcal{A}[X]$, over some ring $\mathcal{A}$, division by p is feasible

- if p is monic (leading coefficient $1_{\mathcal{A}}$)
- and more generally if the leading coefficient of p is invertible in $\mathcal{A}$

assumption: the leading coefficient of $\mathbf{B}$ is invertible in $\mathbb{K}^{m \times m}$

recall $B = B_0 + B_1 X + \cdots + B_d X^d$ with $B_i \in \mathbb{K}^{m \times m}$

the matrix and the polynomial viewpoints

using polynomial arithmetic

division with remainder

problem:
given $\mathbf{A}, \mathbf{B} \in \mathbb{K}^{m \times m}[X]$ with $\mathrm{lc}(\mathbf{B})$ invertible,
compute $\mathbf{Q}, \mathbf{R} \in \mathbb{K}^{m \times m}[X]$ such that

$$\mathbf{A} = \mathbf{B}\mathbf{Q} + \mathbf{R} \quad \text{and} \quad \deg(\mathbf{R}) < \deg(\mathbf{B})$$

example:
let $\mathbf{B} = X\mathbf{I}_m - \mathbf{M}$ for some $\mathbf{M} \in \mathbb{K}^{m \times m}$
give a description of $\mathbf{R} = \mathbf{A} \text{ rem } \mathbf{B}$

division with remainder

problem:
given $\mathbf{A}, \mathbf{B} \in \mathbb{K}^{m \times m}[X]$ with $\mathrm{lc}(\mathbf{B})$ invertible,
compute $\mathbf{Q}, \mathbf{R} \in \mathbb{K}^{m \times m}[X]$ such that
$$\mathbf{A} = \mathbf{B}\mathbf{Q} + \mathbf{R} \quad \text{and} \quad \deg(\mathbf{R}) < \deg(\mathbf{B})$$

example:
let $\mathbf{B} = X\mathbf{I}_m - \mathbf{M}$ for some $\mathbf{M} \in \mathbb{K}^{m \times m}$
give a description of $\mathbf{R} = \mathbf{A} \text{ rem } \mathbf{B}$

from $X^k\mathbf{I}_m - \mathbf{M}^k = (X\mathbf{I}_m - \mathbf{M})(\sum_{1 \leqslant i \leqslant k-1} \mathbf{M}^i X^{k-i})$
we get $X^k\mathbf{I}_m = \mathbf{M}^k \bmod \mathbf{B}$, with $\deg(\mathbf{M}^k) < 1$

then by linearity
$$\begin{aligned}\mathbf{R} = \mathbf{A} \text{ rem } \mathbf{B} &= (\mathbf{A}_0 + \mathbf{A}_1 X + \mathbf{A}_2 X^2 + \cdots + \mathbf{A}_d X^d) \text{ rem } \mathbf{B} \\ &= \mathbf{A}_0 + \mathbf{M}\mathbf{A}_1 + \mathbf{M}^2\mathbf{A}_2 + \cdots + \mathbf{M}^d\mathbf{A}_d\end{aligned}$$

the matrix and the polynomial viewpoints

using polynomial arithmetic

division with remainder

problem:
given $\mathbf{A}, \mathbf{B} \in \mathbb{K}^{m \times m}[X]$ with $\mathrm{lc}(\mathbf{B})$ invertible,
compute $\mathbf{Q}, \mathbf{R} \in \mathbb{K}^{m \times m}[X]$ such that

$$\mathbf{A} = \mathbf{B}\mathbf{Q} + \mathbf{R} \quad \text{and} \quad \deg(\mathbf{R}) < \deg(\mathbf{B})$$

- under this assumption, the usual fast Euclidean algorithm works
- recall:
  1. reverse the equation,
  2. compute quotient by truncated inverse,
  3. deduce remainder
- complexity is $O(\mathrm{MM}(m, d))$ for $d = \max(\deg(\mathbf{A}), \deg(\mathbf{B}))$

the matrix and the polynomial viewpoints

limitations of these viewpoints

applying usual linear algebra algorithms to polynomial matrices:

- helps to understand some algebraic aspects
- leads too easily to computing in the fractions
- gives nonsensical complexity bounds

seeing polynomial matrices as polynomials with matrix coefficients

- allows direct use of some algorithms from polynomial arithmetic
- provides better control of the degree during computations
- remains restrictive and inefficient in many cases

- example for restrictive:

in division with remainder, the assumption "lc($\mathbf{B}$) invertible" can be relaxed into "$\mathbf{B}$ reduced" (and even to "$\mathbf{B}$ nonsingular")

- example for inefficient:

for a matrix of degree d with many entries of degree $\ll d$, we want to take the individual degrees into account

outline

- **introduction**
  - definitions and algebraic properties
  - examples you already know
  - three flagship applications
- **matrices? polynomials?**
  - using matrix arithmetic
  - using polynomial arithmetic
  - limitations of these viewpoints
- **polynomial matrices**
- **reduced forms**

outline

- **introduction**
  - definitions and algebraic properties
  - examples you already know
  - three flagship applications
- **matrices? polynomials?**
  - using matrix arithmetic
  - using polynomial arithmetic
  - limitations of these viewpoints
- **polynomial matrices**
  - size and row/column degrees
  - evaluation-interpolation-based algorithms
  - partial linearization techniques
- **reduced forms**

size and row/column degrees

size of a polynomial matrix = number of coefficients from $\mathbb{K}$ needed for its dense representation

for $\mathbf{A} = (a_{i,j}) \in \mathbb{K}[X]^{m \times n}$,
$\mathrm{size}(\mathbf{A}) = \sum_{i,j} \mathrm{size}(a_{i,j}) = \sum_{i,j} 1 + \max(0, \deg(a_{i,j}))$

mixing matrix and polynomial tools

size and row/column degrees

size of a polynomial matrix = number of coefficients from $\mathbb{K}$ needed for its dense representation

for $\mathbf{A} = (a_{i,j}) \in \mathbb{K}[X]^{m \times n}$,
$\mathrm{size}(\mathbf{A}) = \sum_{i,j} \mathrm{size}(a_{i,j}) = \sum_{i,j} 1 + \max(0, \deg(a_{i,j}))$

recall $\deg(\mathbf{AB}) \leqslant \deg(\mathbf{A}) + \deg(\mathbf{B})$,
however:

in general the size is not compatible with matrix products

mixing matrix and polynomial tools

size and row/column degrees

size of a polynomial matrix = number of coefficients from $\mathbb{K}$ needed for its dense representation

for $\mathbf{A} = (a_{i,j}) \in \mathbb{K}[X]^{m \times n}$,
$\mathrm{size}(\mathbf{A}) = \sum_{i,j} \mathrm{size}(a_{i,j}) = \sum_{i,j} 1 + \max(0, \deg(a_{i,j}))$

recall $\deg(\mathbf{AB}) \leqslant \deg(\mathbf{A}) + \deg(\mathbf{B})$,
however:

in general the size is not compatible with matrix products

considering the degree matrices:

$$\begin{pmatrix} 100 & 50 & 40 & 10 \\ 100 & 50 & 40 & 10 \\ 100 & 50 & 40 & 10 \\ 100 & 50 & 40 & 10 \end{pmatrix} \begin{pmatrix} 50 & 50 & 50 & 50 \\ 50 & 50 & 50 & 50 \\ 50 & 50 & 50 & 50 \\ 50 & 50 & 50 & 50 \end{pmatrix} = \begin{pmatrix} 150 & 150 & 150 & 150 \\ 150 & 150 & 150 & 150 \\ 150 & 150 & 150 & 150 \\ 150 & 150 & 150 & 150 \end{pmatrix}$$

sizes of these three matrices?

mixing matrix and polynomial tools

size and row/column degrees

size of a polynomial matrix = number of coefficients from $\mathbb{K}$ needed for its dense representation

for $\mathbf{A} = (a_{i,j}) \in \mathbb{K}[X]^{m \times n}$,
$\mathrm{size}(\mathbf{A}) = \sum_{i,j} \mathrm{size}(a_{i,j}) = \sum_{i,j} 1 + \max(0, \deg(a_{i,j}))$

recall $\deg(\mathbf{AB}) \leqslant \deg(\mathbf{A}) + \deg(\mathbf{B})$,
however:

in general the size is not compatible with matrix products

but it may be, in some particular cases

mixing matrix and polynomial tools

size and row/column degrees

size of a polynomial matrix = number of coefficients from $\mathbb{K}$ needed for its dense representation

for $\mathbf{A} = (a_{i,j}) \in \mathbb{K}[X]^{m\times n}$,
size$(\mathbf{A}) = \sum_{i,j}$ size$(a_{i,j}) = \sum_{i,j} 1 + \max(0, \deg(a_{i,j}))$

recall $\deg(\mathbf{AB}) \leqslant \deg(\mathbf{A}) + \deg(\mathbf{B})$,
however:

in general the size is not compatible with matrix products

but it may be, in some particular cases

$$\begin{pmatrix} 100 & 100 & 100 & 100 \\ 50 & 50 & 50 & 50 \\ 40 & 40 & 40 & 40 \\ 10 & 10 & 10 & 10 \end{pmatrix} \begin{pmatrix} 50 & 50 & 50 & 50 \\ 50 & 50 & 50 & 50 \\ 50 & 50 & 50 & 50 \\ 50 & 50 & 50 & 50 \end{pmatrix} = \begin{pmatrix} 150 & 150 & 150 & 150 \\ 100 & 100 & 100 & 100 \\ 90 & 90 & 90 & 90 \\ 60 & 60 & 60 & 60 \end{pmatrix}$$

sizes of these three matrices?

mixing matrix and polynomial tools

size and row/column degrees

size of a polynomial matrix = number of coefficients from $\mathbb{K}$ needed for its dense representation

for $\mathbf{A} = (a_{i,j}) \in \mathbb{K}[X]^{m \times n}$,
$\mathrm{size}(\mathbf{A}) = \sum_{i,j} \mathrm{size}(a_{i,j}) = \sum_{i,j} 1 + \max(0, \deg(a_{i,j}))$

recall $\deg(\mathbf{AB}) \leqslant \deg(\mathbf{A}) + \deg(\mathbf{B})$,
however:

in general the size is not compatible with matrix products

but it may be, in some particular cases

- these particular cases include whole families of matrices
c.f. the degree profiles we just saw

- and they include reduced matrices often arising in algorithms
definition will come soon

row degree of a polynomial matrix
= the list of the maximum degree in each of its rows

for $\mathbf{A} = (a_{i,j}) \in \mathbb{K}[X]^{m \times n}$,

$$\begin{aligned} \mathrm{rdeg}(\mathbf{A}) &= (\mathrm{rdeg}(\mathbf{A}_{1,*}), \ldots, \mathrm{rdeg}(\mathbf{A}_{m,*})) \\ &= \left(\max_{1 \leqslant j \leqslant n} \deg(\mathbf{A}_{1,j}), \;\; \ldots, \;\; \max_{1 \leqslant j \leqslant n} \deg(\mathbf{A}_{m,j}) \right) \in \mathbb{Z}^m \end{aligned}$$

mixing matrix and polynomial tools

size and row/column degrees

row degree of a polynomial matrix
$=$ the list of the maximum degree in each of its rows

column degree of a polynomial matrix
$=$ the list of the maximum degree in each of its columns

mixing matrix and polynomial tools

size and row/column degrees

row degree of a polynomial matrix
$=$ the list of the maximum degree in each of its rows

column degree of a polynomial matrix
$=$ the list of the maximum degree in each of its columns

$$\text{average size} \leqslant \begin{matrix}\text{average row size}\\ \text{average column size}\end{matrix} \leqslant 1 + \deg(\mathbf{A})$$

mixing matrix and polynomial tools

size and row/column degrees

row degree of a polynomial matrix
= the list of the maximum degree in each of its rows

column degree of a polynomial matrix
= the list of the maximum degree in each of its columns

$$\text{average size} \leqslant \begin{matrix} \text{average row size} \\ \text{average column size} \end{matrix} \leqslant 1 + \deg(\mathbf{A})$$

consider $\mathbf{A}$ and $\mathbf{B}$ with respective degree matrices:

$$\begin{pmatrix} 100 & 50 & 40 & 10 \\ 100 & 50 & 40 & 10 \\ 100 & 50 & 40 & 10 \\ 100 & 50 & 40 & 10 \end{pmatrix} \text{ and } \begin{pmatrix} 100 & 100 & 100 & 100 \\ 50 & 50 & 50 & 50 \\ 40 & 40 & 40 & 40 \\ 10 & 10 & 10 & 10 \end{pmatrix}$$

row degree and column degree of these two matrices?

mixing matrix and polynomial tools

evaluation-interpolation-based algorithms

exercise: multiplication, determinant, inversion

1. adapting the evaluation-interpolation paradigm to matrices in $\mathbb{K}[X]^{m \times m}$,

- give an explicit **multiplication** algorithm
- give a **determinant** algorithm
- give an **inversion** algorithm ☕
  computing the inverse over the fractions $\mathbb{K}(X)$

2. for each of these algorithms,

- give a required lower bound on the **cardinality of** $\mathbb{K}$
- state and prove an upper bound on the **complexity**

two hints and one direction for further study:

- use known degree bounds on the output
- for inversion, assume you can do quasi-linear Cauchy interpolation
- could your complexity bounds take into account degree measures that refine the matrix degree such as the average row or column degree?

5.8. Cauchy interpolation

The polynomial interpolation problem is, given a collection of sample values $v_i = f(u_i) \in F$ for $0 \le i < n$ of an unknown function $f: F \longrightarrow F$ at distinct points $u_0, \ldots, u_{n-1}$ of a field F, to compute a polynomial $g \in F[x]$ of degree less than n that interpolates g at those points, so that $g(u_i) = v_i$ for all i. We saw in Section 5.2 that such a polynomial always exists uniquely and learned how to compute it using the Lagrange interpolation formula.

A more general problem is **Cauchy interpolation** or rational interpolation, where furthermore $k \in \{0, \ldots, n\}$ is given and we are looking for a rational function $r/t \in F(x)$, with $r, t \in F[x]$, such that

$$t(u_i) \neq 0 \text{ and } \frac{r(u_i)}{t(u_i)} = v_i \text{ for } 0 \le i < n, \quad \deg r < k, \quad \deg t \le n-k. \tag{20}$$

[von zur Gathen, Gerhard, Modern Computer Algebra]

see also [AECF, Definition 7.1] (in French)

we will describe a quasi-linear algorithm later in this course

which does not rely on polynomial matrix inversion...

mixing matrix and polynomial tools

partial linearization techniques

reduce **unbalanced** degrees to the **average** degree

where degree means row degree, column degree, or related refined measures

[Storjohann 2006] [Zhou-Labahn 2012] [Jeannerod-Neiger-Villard 2020]

typical properties:

from a matrix $\mathbf{A} \in \mathbb{K}[X]^{m \times m}$ with $D = |\mathrm{rdeg}(\mathbf{A})| \ll m \deg(\mathbf{A})$
construct a matrix $\bar{\mathbf{A}} \in \mathbb{K}[X]^{m' \times m'}$ with

- a slight increase of matrix dimension: $m \leqslant m' \leqslant 2m$
- a strong decrease of matrix degree: $\deg(\bar{\mathbf{A}}) \leqslant 2\frac{D}{m}$
- preservation of the features targeted by our computations

examples:

- product $\mathbf{AB}$ easily deduced from product $\bar{\mathbf{A}}\bar{\mathbf{B}}$
- preservation of the determinant $\det(\mathbf{A}) = \det(\bar{\mathbf{A}})$
- inverse of $\bar{\mathbf{A}}$ contains inverse of $\mathbf{A}$ as submatrix
- ...

mixing matrix and polynomial tools

partial linearization techniques

reduce **unbalanced** degrees to the **average** degree

basic illustration:

- let $\mathbf{A} \in \mathbb{K}[X]^{m \times m}$ of degree $< d$,
- let $\mathbf{u} \in \mathbb{K}[X]^{m \times 1}$ of degree $< md$,

then the matrix-vector product $\mathbf{Au}$ can be computed in

$$\mathsf{MM}(m, d) + O(m^2 d) \text{ operations in } \mathbb{K}$$

what would be the cost of the "naive" multiplication?

algorithm:

mixing matrix and polynomial tools

partial linearization techniques

reduce **unbalanced** degrees to the **average** degree

basic illustration:

- let $\mathbf{A} \in \mathbb{K}[X]^{m \times m}$ of degree $< d$,
- let $\mathbf{u} \in \mathbb{K}[X]^{m \times 1}$ of degree $< md$,

then the matrix-vector product $\mathbf{Au}$ can be computed in

$$\mathsf{MM}(m, d) + O(m^2 d) \text{ operations in } \mathbb{K}$$

what would be the cost of the "naive" multiplication?

algorithm:

$$\begin{bmatrix} & & \\ & \mathbf{A} & \\ & & \end{bmatrix} \begin{bmatrix} \\ \mathbf{u} \\ \\ \end{bmatrix} = \begin{bmatrix} & & \\ & \mathbf{A} & \\ & & \end{bmatrix} \begin{bmatrix} & & \\ & \bar{\mathbf{u}} & \\ & & \end{bmatrix} \begin{bmatrix} 1 \\ X^d \\ X^{2d} \\ \vdots \end{bmatrix}$$

where the columns of $\bar{\mathbf{u}} \in \mathbb{K}[X]^{m \times m}$ form the X^d-adic expansion of $\mathbf{u}$

$\Rightarrow$ here $\deg(\bar{\mathbf{u}}) < d$

outline

- **introduction**
  - definitions and algebraic properties
  - examples you already know
  - three flagship applications
- **matrices? polynomials?**
  - using matrix arithmetic
  - using polynomial arithmetic
  - limitations of these viewpoints
- **polynomial matrices**
  - size and row/column degrees
  - evaluation-interpolation-based algorithms
  - partial linearization techniques
- **reduced forms**

outline

introduction

- definitions and algebraic properties
- examples you already know
- three flagship applications

matrices? polynomials?

- using matrix arithmetic
- using polynomial arithmetic
- limitations of these viewpoints

polynomial matrices

- size and row/column degrees
- evaluation-interpolation-based algorithms
- partial linearization techniques

reduced forms

- motivations
- leading matrix and reducedness
- characterizations and main properties

polynomial matrices in reduced form

motivations

the above degree measures and techniques
- yield **faster algorithms** in some cases
- but leave **many remaining questions**

1. row and column degrees not compatible with multiplication
2. does not lift the restrictive assumption on $\mathrm{lc}(\mathbf{B})$ for `QuoRem`
3. can we get even faster determinant and inversion?

polynomial matrices in reduced form

motivations

the above degree measures and techniques
- yield **faster algorithms** in some cases
- but leave **many remaining questions**

1. row and column degrees not compatible with multiplication
2. does not lift the restrictive assumption on $\mathrm{lc}(\mathbf{B})$ for `QuoRem`
3. can we get even faster determinant and inversion?

1. more general partial linearizations

THEOREM 3.7. *Let $\mathbf{A} \in \mathbb{K}[x]^{m\times n}$, $\vec{s}$ a shift with entries bounding the column degrees of $\mathbf{A}$ and ξ, a bound on the sum of the entries of $\vec{s}$. Let $\mathbf{B} \in \mathbb{K}[x]^{n\times k}$ with $k \in O(m)$ and the sum θ of its $\vec{s}$-column degrees satisfying $\theta \in O(\xi)$. Then we can multiply $\mathbf{A}$ and $\mathbf{B}$ with a cost of $O^{\sim}(nm^{\omega-2}\xi)$.*

[Zhou-Labahn-Storjohann 2012]

shift s? **s-column degree?**

polynomial matrices in reduced form

motivations

the above degree measures and techniques
- yield **faster algorithms** in some cases
- but leave **many remaining questions**

1. row and column degrees not compatible with multiplication
2. does not lift the restrictive assumption on $\mathrm{lc}(\mathbf{B})$ for `QuoRem`
3. can we get even faster determinant and inversion?

2. more general division with remainder
is it reasonable that the `QuoRem` algorithm does not support the case of a division $\mathbf{A} = \mathbf{B}\mathbf{Q} + \mathbf{R}$ where $\mathbf{B}$ is the diagonal matrix $\mathbf{B} = \mathrm{diag}(X^{d_1}, \ldots, X^{d_m})$?

polynomial matrices in reduced form

motivations

the above degree measures and techniques
- yield **faster algorithms** in some cases
- but leave **many remaining questions**

1. row and column degrees not compatible with multiplication
2. does not lift the restrictive assumption on $\mathrm{lc}(\mathbf{B})$ for `QuoRem`
3. can we get even faster determinant and inversion?

2. more general division with remainder
is it reasonable that the `QuoRem` algorithm does not support the case of a division $\mathbf{A} = \mathbf{B}\mathbf{Q} + \mathbf{R}$ where $\mathbf{B}$ is the diagonal matrix $\mathbf{B} = \mathrm{diag}(X^{d_1}, \ldots, X^{d_m})$?

column reduced?

Algorithm 1: PM-QuoRem
Input:
- $\mathbf{M} \in \mathbb{K}[x]^{n\times n}$ column reduced,
- $\mathbf{F} \in \mathbb{K}[x]^{m\times n}$,
- $\delta \in \mathbb{Z}_{>0}$ such that $\mathrm{cdeg}(\mathbf{F}) < \mathrm{cdeg}(\mathbf{M}) + (\delta, \ldots, \delta)$.

Output: the quotient $\mathrm{Quo}(\mathbf{F}, \mathbf{M})$, the remainder $\mathrm{Rem}(\mathbf{F}, \mathbf{M})$.

1. `/* reverse order of coefficients */`
   $(d_1, \ldots, d_n) \leftarrow \mathrm{cdeg}(\mathbf{M})$
   $\mathbf{M}_{\mathrm{rev}} = \mathbf{M}(x^{-1})\,\mathrm{diag}(x^{d_1}, \ldots, x^{d_n})$
   $\mathbf{F}_{\mathrm{rev}} = \mathbf{F}(x^{-1})\,\mathrm{diag}(x^{\delta+d_1-1}, \ldots, x^{\delta+d_n-1})$
2. `/* compute quotient via expansion */`
   $\mathbf{Q}_{\mathrm{rev}} \leftarrow \mathbf{F}_{\mathrm{rev}}\mathbf{M}_{\mathrm{rev}}^{-1} \bmod x^{\delta}$
   $\mathbf{Q} \leftarrow x^{\delta-1}\mathbf{Q}_{\mathrm{rev}}(x^{-1})$
3. *Return* $(\mathbf{Q}, \mathbf{F} - \mathbf{Q}\mathbf{M})$

polynomial matrices in reduced form

motivations

the above degree measures and techniques
- yield **faster algorithms** in some cases
- but leave **many remaining questions**

1. row and column degrees not compatible with multiplication
2. does not lift the restrictive assumption on $\mathrm{lc}(\mathbf{B})$ for `QuoRem`
3. can we get even faster determinant and inversion?

3. even faster algorithms

for $\mathbf{A} \in \mathbb{K}[X]^{m \times m}$ of degree d, evaluation-interpolation yields determinant and inverse algorithms in $O\tilde{}(m^{\omega+1}d)$ ops.

how does this compare to the size of $\mathbf{A}$?
if you were to search for faster algorithms, what would you pick as your target complexity bound?

polynomial matrices in reduced form

motivations

the above degree measures and techniques
- yield **faster algorithms** in some cases
- but leave **many remaining questions**

1. row and column degrees not compatible with multiplication
2. does not lift the restrictive assumption on $\mathrm{lc}(\mathbf{B})$ for `QuoRem`
3. can we get even faster determinant and inversion?

3. even faster algorithms

for $\mathbf{A} \in \mathbb{K}[X]^{m \times m}$ of degree d, evaluation-interpolation yields determinant and inverse algorithms in $O\tilde{~}(m^{\omega+1} d)$ ops.

how does this compare to the size of $\mathbf{A}$?
if you were to search for faster algorithms, what would you pick as your target complexity bound?

$\rightsquigarrow$ cost $O\tilde{~}(m^{\omega} \frac{D}{m})$ achieved using operations on reduced matrices

[Zhou-Labahn-Storjohann 2015] [Labahn-Neiger-Zhou 2017]

polynomial matrices in reduced form

motivations

the above degree measures and techniques
- yield **faster algorithms** in some cases
- but leave **many remaining questions**

1. row and column degrees not compatible with multiplication
2. does not lift the restrictive assumption on $\mathsf{lc}(\mathbf{B})$ for `QuoRem`
3. can we get even faster determinant and inversion?

4. bonus: predictable degrees

in the two cases below,
- can you predict $\deg \det(\mathbf{A})$?
- can you predict the degrees in $\mathbf{BA}$ from the degrees in $\mathbf{B}$?

. case 1: $\mathbf{A} = X\mathbf{I}_m - \mathbf{M}$, with $\mathbf{M} \in \mathbb{K}^{m \times m}$
. case 2: $\mathbf{A} = X^d\mathbf{L} + \mathbf{R}$, with $\deg(\mathbf{R}) < d$ and $\mathbf{L} \in \mathbb{K}^{m \times m}$

polynomial matrices in reduced form

leading matrix and reducedness

notation:

let $\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ with no zero row,
define $\mathbf{d} = (d_1, \ldots, d_m) = \mathrm{rdeg}(\mathbf{A})$

and $$\mathbf{X}^{\mathbf{d}} = \begin{bmatrix} X^{d_1} & & \\ & \ddots & \\ & & X^{d_m} \end{bmatrix} \in \mathbb{K}[X]^{m \times m}$$

definition: (row-wise) leading matrix

the leading matrix of $\mathbf{A}$ is the unique matrix $\mathbf{L} \in \mathbb{K}^{m \times n}$
such that $\mathbf{A} = \mathbf{X}^{\mathbf{d}}\mathbf{L} + \mathbf{R}$ with $\mathrm{rdeg}(\mathbf{R}) < \mathbf{d}$ entry-wise

equivalently, $\mathbf{X}^{-\mathbf{d}}\mathbf{A} = \mathbf{L} +$ terms of strictly negative degree

polynomial matrices in reduced form

leading matrix and reducedness

notation:

let $\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ with no zero row,
define $\mathbf{d} = (d_1, \ldots, d_m) = \mathrm{rdeg}(\mathbf{A})$

and $\mathbf{X}^{\mathbf{d}} = \begin{bmatrix} X^{d_1} & & \\ & \ddots & \\ & & X^{d_m} \end{bmatrix} \in \mathbb{K}[X]^{m \times m}$

definition: (row-wise) leading matrix

the leading matrix of $\mathbf{A}$ is the unique matrix $\mathbf{L} \in \mathbb{K}^{m \times n}$
such that $\mathbf{A} = \mathbf{X}^{\mathbf{d}}\mathbf{L} + \mathbf{R}$ with $\mathrm{rdeg}(\mathbf{R}) < \mathbf{d}$ entry-wise

equivalently, $\mathbf{X}^{-\mathbf{d}}\mathbf{A} = \mathbf{L} +$ terms of strictly negative degree

- what is the leading matrix of $\begin{bmatrix} 3X+4 & X^3+4X+1 & 4X^2+3 \\ 5 & 5X^2+3X+1 & 5X+3 \end{bmatrix}$?
- what is the leading matrix of $\mathbf{A} = X\mathbf{I}_m - \mathbf{M}$? of $\mathbf{A} = X^{\mathrm{d}}\mathbf{L} + \mathbf{R}$?

notation:

let $\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ with no zero row,
we write $\mathrm{lm}(\mathbf{A})$ for the leading matrix of $\mathbf{A}$

definition: (row-wise) reduced matrix

$\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ is said to be reduced
if $\mathrm{lm}(\mathbf{A})$ has full row rank

what does this imply on m and n?

polynomial matrices in reduced form

leading matrix and reducedness

notation:

let $\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ with no zero row,
we write $\mathrm{lm}(\mathbf{A})$ for the leading matrix of $\mathbf{A}$

definition: (row-wise) reduced matrix

$\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ is said to be reduced
if $\mathrm{lm}(\mathbf{A})$ has full row rank

what does this imply on m and n?

- is the matrix $\begin{bmatrix} 3X+4 & X^3+4X+1 & 4X^2+3 \\ 5 & 5X^2+3X+1 & 5X+3 \end{bmatrix}$ reduced?
- is $\mathbf{A} = X\mathbf{I}_m - \mathbf{M}$ row-wise reduced? column-wise reduced?
- is "$\mathbf{A} = X^d\mathbf{L} + \mathbf{R}$ is reduced" equivalent to "$\mathbf{L}$ is invertible"?

polynomial matrices in reduced form

characterizations and main properties

let $\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ with $m \leqslant n$,
the following are equivalent:

(i) $\mathbf{A}$ is reduced (i.e. $\mathsf{lm}(\mathbf{A})$ has full rank)

polynomial matrices in reduced form

characterizations and main properties

let $\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ with $m \leqslant n$,
the following are equivalent:

(i) $\mathbf{A}$ is reduced (i.e. $\mathrm{lm}(\mathbf{A})$ has full rank)

(ii) for any vector $\mathbf{u} = [\mathbf{u}_1 \;\; 1 \;\; \mathbf{u}_2] \in \mathbb{K}[X]^{1 \times m}$ with 1 at index i,
$\mathrm{rdeg}(\mathbf{u}\mathbf{A}) \geqslant \mathrm{rdeg}(\mathbf{A}_{i,*})$

polynomial matrices in reduced form

characterizations and main properties

let $\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ with $m \leqslant n$,
the following are equivalent:

(i) $\mathbf{A}$ is reduced (i.e. $\mathrm{lm}(\mathbf{A})$ has full rank)

(ii) for any vector $\mathbf{u} = [\mathbf{u}_1 \;\; 1 \;\; \mathbf{u}_2] \in \mathbb{K}[X]^{1 \times m}$ with 1 at index i,
$\mathrm{rdeg}(\mathbf{uA}) \geqslant \mathrm{rdeg}(\mathbf{A}_{i,*})$

(iii) predictable degree: for any vector $\mathbf{u} = [u_1 \cdots u_m] \in \mathbb{K}[X]^{1 \times m}$,
$\mathrm{rdeg}(\mathbf{uA}) = \max_{1 \leqslant i \leqslant m}(\deg(u_i) + \mathrm{rdeg}(\mathbf{A}_{i,*}))$

polynomial matrices in reduced form

characterizations and main properties

let $\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ with $m \leqslant n$,
the following are equivalent:

(i) $\mathbf{A}$ is reduced (i.e. $\mathrm{lm}(\mathbf{A})$ has full rank)

(ii) for any vector $\mathbf{u} = [\mathbf{u}_1 \;\; 1 \;\; \mathbf{u}_2] \in \mathbb{K}[X]^{1 \times m}$ with 1 at index i,
$\mathrm{rdeg}(\mathbf{uA}) \geqslant \mathrm{rdeg}(\mathbf{A}_{i,*})$

(iii) predictable degree: for any vector $\mathbf{u} = [u_1 \cdots u_m] \in \mathbb{K}[X]^{1 \times m}$,
$\mathrm{rdeg}(\mathbf{uA}) = \max_{1 \leqslant i \leqslant m}(\deg(u_i) + \mathrm{rdeg}(\mathbf{A}_{i,*}))$

(iv) degree minimality: $\mathrm{rdeg}(\mathbf{A}) \preccurlyeq \mathrm{rdeg}(\mathbf{UA})$ holds for any nonsingular matrix $\mathbf{U} \in \mathbb{K}[X]^{m \times m}$, where $\preccurlyeq$ sorts the tuples in nondecreasing order and then uses lexicographic comparison

polynomial matrices in reduced form

characterizations and main properties

let $\mathbf{A} \in \mathbb{K}[X]^{m \times n}$ with $m \leqslant n$,
the following are equivalent:

(i) $\mathbf{A}$ is reduced (i.e. $\mathrm{lm}(\mathbf{A})$ has full rank)

(ii) for any vector $\mathbf{u} = [\mathbf{u}_1 \;\; 1 \;\; \mathbf{u}_2] \in \mathbb{K}[X]^{1 \times m}$ with 1 at index i,
$\mathrm{rdeg}(\mathbf{uA}) \geqslant \mathrm{rdeg}(\mathbf{A}_{i,*})$

(iii) predictable degree: for any vector $\mathbf{u} = [u_1 \cdots u_m] \in \mathbb{K}[X]^{1 \times m}$,
$\mathrm{rdeg}(\mathbf{uA}) = \max_{1 \leqslant i \leqslant m}(\deg(u_i) + \mathrm{rdeg}(\mathbf{A}_{i,*}))$

(iv) degree minimality: $\mathrm{rdeg}(\mathbf{A}) \preccurlyeq \mathrm{rdeg}(\mathbf{UA})$ holds for any nonsingular matrix $\mathbf{U} \in \mathbb{K}[X]^{m \times m}$, where $\preccurlyeq$ sorts the tuples in nondecreasing order and then uses lexicographic comparison

(v) predictable determinantal degree: $\deg\det(\mathbf{A}) = |\mathrm{rdeg}(\mathbf{A})|$
(only when $m = n$)

summary

introduction

- definitions and algebraic properties
- examples you already know
- three flagship applications

matrices? polynomials?

- using matrix arithmetic
- using polynomial arithmetic
- limitations of these viewpoints

polynomial matrices

- size and row/column degrees
- evaluation-interpolation-based algorithms
- partial linearization techniques

reduced forms

- motivations
- leading matrix and reducedness
- characterizations and main properties