Polynomial system solving and/by multidimensional realization theory

'Back to the Roots'

Philippe Dreesen[1] Kim Batselier[2] Bart De Moor[1]

[1] KU Leuven, ESAT/STADIUS
[2] TU Delft, 3mE-DCSC

Jan 28, 2022

Control e-seminar
Mechatronics in Mechanical and Process Engineering (MEC)
University of Kaiserslautern

Outline

The history of polynomial system solving

Univariate polynomials and eigenvalue decompositions

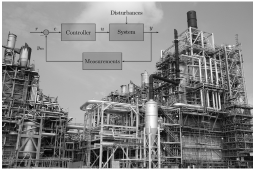
Multivariate polynomials and nD systems

Recent developments in the Macaulay spirit

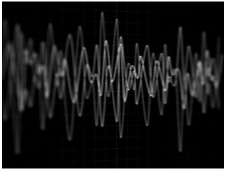
Conclusions and Perspectives

Why Study Polynomial Equations?

- fundamental mathematical objects
- powerful modelling tools
- ubiquitous in Science and Engineering (often *hidden*)



Systems and Control

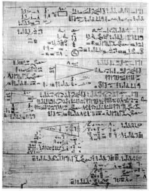


Signal Processing

Computational Biology

Kinematics/Robotics

Polynomial root-finding has a long and rich history. . .



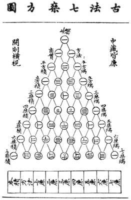
Egypt
(3000BCE-300BCE)

Babylon
(3000BCE-539BCE)

Euclid
(fl. 300BCE)

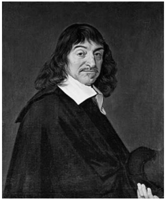
Diophantus
(c200-c284)

Al-Khwarizmi
(c780-c850)

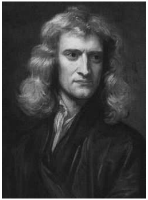


Zhu Shijie
(c1260-c1320)

Pierre de Fermat
(c1601-1665)



René Descartes
(1596-1650)



Isaac Newton
(1643-1727)

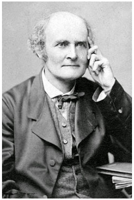
Gottfried Leibniz
(1646-1716)

Etienne Bézout
(1730-1783)

Carl Friedrich Gauss
(1777-1755)

Jean-Victor Poncelet
(1788-1867)

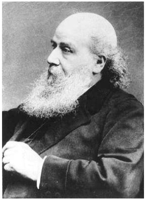
Evariste Galois
(1811-1832)



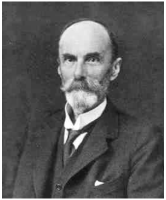
Arthur Cayley
(1821-1895)

Leopold Kronecker
(1823-1891)

Edmond Laguerre
(1834-1886)



James J. Sylvester
(1814-1897)

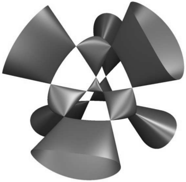


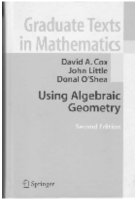
Francis S. Macaulay
(1862-1937)

David Hilbert
(1862-1943)

. . . leading to Algebraic Geometry and Computer Algebra

– large body of literature
– emphasis not (anymore) on *solving* equations
– computer algebra: symbolic manipulations (e.g., Gröbner Bases)
– numerical issues!



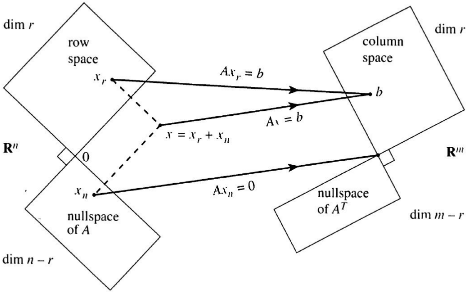



Wolfgang Gröbner
(1899-1980)


Bruno Buchberger

Back to the roots! Let's use linear algebra!?

- comprehensible and accessible language
- intuitive geometric interpretation
- computationally powerful framework
- well-established methods and stable numerics



Eigenvalue decompositions are at the core of root-finding

Eigenvalue equation

$$Av = \lambda v$$

and eigenvalue decomposition

$$A = V \Lambda V^{-1}$$

Enormous importance in (numerical) linear algebra and apps

- ‘understand’ the action of matrix A
- at the heart of a multitude of applications: oscillations, vibrations, quantum mechanics, data analytics, graph theory, and **many** more

Outline

From eigenvalues to roots . . . and back

Characteristic Polynomial

The eigenvalues of A are the roots of

$$p(\lambda) = |A - \lambda I|$$

Companion Matrix

Solving

$$q(x) = 7x^3 - 2x^2 - 5x + 1 = 0$$

leads to

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1/7 & 5/7 & 2/7 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ x^2 \end{bmatrix} = x \begin{bmatrix} 1 \\ x \\ x^2 \end{bmatrix}$$

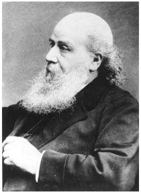
The Sylvester Matrix is used for finding common roots of multiple univariate polynomials

Consider two polynomial equations

$$\begin{array}{rcl} f(x) & = & x^3 - 6x^2 + 11x - 6 & = & (x-1)(x-2)(x-3) \\ g(x) & = & -x^2 + 5x - 6 & = & -(x-2)(x-3) \end{array}$$

Common roots if $|S(f,g)| = 0$

$$S(f,g) = \left[\begin{array}{ccccc} -6 & 11 & -6 & 1 & 0 \\ 0 & -6 & 11 & -6 & 1 \\ \hline -6 & 5 & -1 & 0 & 0 \\ 0 & -6 & 5 & -1 & 0 \\ 0 & 0 & -6 & 5 & -1 \end{array}\right]$$



James Joseph Sylvester

Sylvester's construction can be understood from

$$
\begin{array}{l}
f(x)=0 \\
x\cdot f(x)=0 \\
g(x)=0 \\
x\cdot g(x)=0 \\
x^2\cdot g(x)=0
\end{array}
\qquad
\overset{\begin{array}{ccccc} 1 & x & x^2 & x^3 & x^4 \end{array}}{\begin{bmatrix}
-6 & 11 & -6 & 1 & 0 \\
 & -6 & 11 & -6 & 1 \\
-6 & 5 & -1 & & \\
 & -6 & 5 & -1 & \\
 & & -6 & 5 & -1
\end{bmatrix}}
\begin{bmatrix}
1 & 1 \\
x_1 & x_2 \\
x_1^2 & x_2^2 \\
x_1^3 & x_2^3 \\
x_1^4 & x_2^4
\end{bmatrix} = 0
$$

where $x_1 = 2$ and $x_2 = 3$ are the common roots of f and g

The vectors in the Vandermonde-like null space K obey a 'shift structure':

$$\begin{bmatrix} 1 \\ x \\ x^2 \\ x^3 \end{bmatrix} x = \begin{bmatrix} x \\ x^2 \\ x^3 \\ x^4 \end{bmatrix}$$

The Vandermonde-like null space K is not available directly, instead we compute Z, for which $ZV = K$. We now have

$$\begin{aligned} KD &= K \\ ZVD &= ZV \end{aligned}$$

leading to the generalized eigenvalue problem

$$ZV = ZVD$$

Realization Theory Essentials

State-space formulation of linear dynamical system:

$$\left\{ \begin{array}{rcl} x_{k+1} & = & Ax_k + Bu_k \\ y_k & = & Cx_k + Bu_k \end{array} \right. \quad \text{with} \quad \left\{ \begin{array}{rcl} x_k & \in & \mathbb{R}^n \\ u_k & \in & \mathbb{R}^m \\ y_k & \in & \mathbb{R}^p \end{array} \right.$$

Controllability and Observability Matrices:

$$\begin{array}{rcl} \mathcal{C}_i & = & \left[\begin{array}{cccc} B & AB & \dots & A^{i-1}B \end{array} \right] \\ \mathcal{O}_i & = & \left[\begin{array}{cccc} C^T & (CA)^T & \dots & (CA^{i-1})^T \end{array} \right]^T \end{array}$$

and the corresponding **Gramians**:

$$\begin{array}{rcl} P & = & \mathcal{C}\mathcal{C}^* \\ Q & = & \mathcal{O}^*\mathcal{O} \end{array}$$

that solve the **Lyapunov equations**

$$\begin{array}{rcl} APA^* - P & = & -BB^* \\ A^*QA - Q & = & -C^*C \end{array}$$

Impulse Response: Markov parameters g_k

$$g_k = \left\{ \begin{array}{ll} D & k = 0 \\ CA^{k-1}B & k > 0 \end{array} \right.$$

Transfer function:

$$\begin{array}{rcl} G(q) & = & \sum_k g_k q^{-k} \\ & = & C(qI - A)^{-1}B + D \end{array}$$

Linear Realization Theory

Impulse response experiment: Markov parameters g_k

$$g_k = \begin{cases} D & k = 0 \\ CA^{k-1}B & k > 0 \end{cases}$$

Hankel matrix from data:

$$H = \begin{bmatrix} g_1 & g_2 & g_3 & g_4 & \cdot^{\cdot^{\cdot}} \\ g_2 & g_3 & g_4 & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} \\ g_3 & g_4 & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} \\ g_4 & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} \\ \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} & \cdot^{\cdot^{\cdot}} \end{bmatrix} = \mathcal{O}\mathcal{C}$$

rank(H) = McMillan Degree

Andrei Andreyevich Markov

Leopold Kronecker

Realization Theory – Algorithm

Algorithm (Realization Theory)

input: Markov parameters
$g_k, \quad k = 0, \ldots, K$
output: (Minimal order) realization
(A_r, B_r, C_r, D_r)

1 The matrix D_r is easily found as

$$D_r = g_0.$$

2 Construct the (block-)Hankel matrix $H_{i,j}$

$$H_{i,j} = \begin{bmatrix} g_1 & g_2 & g_3 & g_4 & \iddots \\ g_2 & g_3 & g_4 & \iddots & \iddots \\ g_3 & g_4 & \iddots & \iddots & \iddots \\ g_4 & \iddots & \iddots & \iddots & \iddots \\ \iddots & \iddots & \iddots & \iddots & \iddots \end{bmatrix}$$

Realization Theory – Algorithm

1 Perform an SVD on $\mathsf{H}_{i,j} = \mathsf{U}\Sigma\mathsf{V}^T$ and take

$$\begin{aligned}\mathcal{O}_i &= \mathsf{U}\Sigma^{1/2},\\ \mathcal{C}_j &= \Sigma^{1/2}\mathsf{V}^T.\end{aligned}$$

The rank of the block Hankel matrix, the minimal order of the underlying system, is equal to the number of nonzero singular values.

2 C_r is formed from the first p rows of $\mathcal{O}_i$, while B_r is formed from the first m columns of $\mathcal{C}_j$.

Realization Theory – Algorithm

1 From the observability matrix

$$\underline{\mathcal{O}_i}\mathsf{A} = \overline{\mathcal{O}_i},$$

A_r can be calculated as

$$\mathsf{A}_r = \left(\underline{\mathcal{O}_i}\right)^{\dagger} \overline{\mathcal{O}_i}.$$

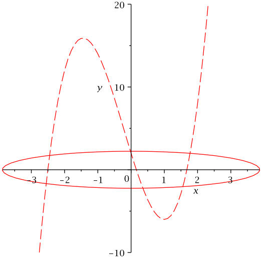
Analogously, A_r can also be calculated as

$$|\mathcal{C}_j \left(\mathcal{C}_j|\right)^{\dagger}.$$

Outline

Generalizing the Sylvester matrix to the multivariate case leads to the Macaulay matrix

Consider the system

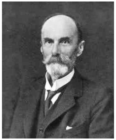


$$
\begin{array}{rcll}
p(x,y) & = & x^2 + 3y^2 - 15 & = & 0 \\
q(x,y) & = & y - 3x^3 - 2x^2 + 13x - 2 & = & 0
\end{array}
$$

Matrix representation of the system: Macaulay matrix M

	1	x	y	x^2	xy	y^2	x^3	x^2y	xy^2	y^3
$p(x,y)$	-15			1		3				
$x \cdot p(x,y)$		-15					1		3	
$y \cdot p(x,y)$			-15					1		3
$q(x,y)$	-2	13	1	-2			-3			

$$
\begin{aligned}
p(x,y) &= x^2 + 3y^2 - 15 &= 0\\
q(x,y) &= y - 3x^3 - 2x^2 + 13x - 2 &= 0
\end{aligned}
$$



Continue to enlarge the Macaulay matrix M:

		1	x	y	x^2	xy	y^2	x^3	x^2y	xy^2	y^3	x^4	x^3y	x^2y^2	xy^3	y^4	x^5	x^4y	x^3y^2	x^2y^3	xy^4	y^5	→
$d=3$	p	−15			1		3																
	xp		−15					1		3													
	yp			−15					1		3												
	q	−2	13	1	−2			−3															
$d=4$	x^2p				−15							1		3									
	xyp					−15							1		3								
	y^2p						−15							1		3							
	xq		−2		13	1		−2				−3											
	yq			−2		13	1		−2				−3										
$d=5$	x^3p							−15									1		3				
	x^2yp								−15									1		3			
	xy^2p									−15									1		3		
	y^3p										−15									1		3	
	x^2q				−2			13	1				−2				−3						
	xyq					−2			13	1				−2				−3					
	y^2q						−2			13	1				−2				−3				
↓								⋱	⋱	⋱	⋱	⋱	⋱	⋱	⋱	⋱	⋱	⋱	⋱	⋱	⋱	⋱	⋱

- Macaulay coefficient matrix M:

$$M = \begin{bmatrix} \times & \times & \times & \times & 0 & 0 & 0 \\ 0 & \times & \times & \times & \times & 0 & 0 \\ 0 & 0 & \times & \times & \times & \times & 0 \\ 0 & 0 & 0 & \times & \times & \times & \times \end{bmatrix}$$

- solutions generate vectors in null space

$$MK = 0$$

- number of solutions m = nullity

Multivariate Vandermonde basis for the null space:

$$\left[\begin{array}{c|c|c|c} 1 & 1 & \dots & 1 \\ \hline x_1 & x_2 & \dots & x_m \\ y_1 & y_2 & \dots & y_m \\ \hline x_1^2 & x_2^2 & \dots & x_m^2 \\ x_1 y_1 & x_2 y_2 & \dots & x_m y_m \\ y_1^2 & y_2^2 & \dots & y_m^2 \\ \hline x_1^3 & x_2^3 & \dots & x_m^3 \\ x_1^2 y_1 & x_2^2 y_2 & \dots & x_m^2 y_m \\ x_1 y_1^2 & x_2 y_2^2 & \dots & x_m y_m^2 \\ y_1^3 & y_2^3 & \dots & y_m^3 \\ \hline x_1^4 & x_2^4 & \dots & x_4^4 \\ x_1^3 y_1 & x_2^3 y_2 & \dots & x_m^3 y_m \\ x_1^2 y_1^2 & x_2^2 y_2^2 & \dots & x_m^2 y_m^2 \\ x_1 y_1^3 & x_2 y_2^3 & \dots & x_m y_m^3 \\ y_1^4 & y_2^4 & \dots & y_m^4 \\ \hline \vdots & \vdots & \vdots & \vdots \end{array}\right]$$

Select the 'top' m linear independent rows of K

$$S_1 \, K$$

$$\begin{bmatrix}
1 & 1 & \dots & 1 \\
x_1 & x_2 & \dots & x_m \\
y_1 & y_2 & \dots & y_m \\
x_1^2 & x_2^2 & \dots & x_m^2 \\
x_1 y_1 & x_2 y_2 & \dots & x_m y_m \\
y_1^2 & y_2^2 & \dots & y_m^2 \\
x_1^3 & x_2^3 & \dots & x_m^3 \\
x_1^2 y_1 & x_2^2 y_2 & \dots & x_m^2 y_m \\
x_1 y_1^2 & x_2 y_2^2 & \dots & x_m y_m^2 \\
y_1^3 & y_2^3 & \dots & y_m^3 \\
x_1^4 & x_2^4 & \dots & x_4^4 \\
x_1^3 y_1 & x_2^3 y_2 & \dots & x_m^3 y_m \\
x_1^2 y_1^2 & x_2^2 y_2^2 & \dots & x_m^2 y_m^2 \\
x_1 y_1^3 & x_2 y_2^3 & \dots & x_m y_m^3 \\
y_1^4 & y_2^4 & \dots & y_m^4 \\
\vdots & \vdots & \vdots & \vdots
\end{bmatrix}$$

Shifting the selected rows gives (shown for 3 columns)

$$\begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ x_1^2 & x_2^2 & x_3^2 \\ x_1y_1 & x_2y_2 & x_3y_3 \\ y_1^2 & y_2^2 & y_3^2 \\ x_1^3 & x_2^3 & x_3^3 \\ x_1^2y_1 & x_2^2y_2 & x_3^2y_3 \\ x_1y_1^2 & x_2y_2^2 & x_3y_3^2 \\ y_1^3 & y_2^3 & y_3^3 \\ x_1^4 & x_2^4 & x_4^4 \\ x_1^3y_1 & x_2^3y_2 & x_3^3y_3 \\ x_1^2y_1^2 & x_2^2y_2^2 & x_3^2y_3^2 \\ x_1y_1^3 & x_2y_2^3 & x_3y_3^3 \\ y_1^4 & y_2^4 & y_3^4 \\ \vdots & \vdots & \vdots \end{bmatrix} \longrightarrow \text{“shift with } x\text{”} \longrightarrow \begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ x_1^2 & x_2^2 & x_3^2 \\ x_1y_1 & x_2y_2 & x_3y_3 \\ y_1^2 & y_2^2 & y_3^2 \\ x_1^3 & x_2^3 & x_3^3 \\ x_1^2y_1 & x_2^2y_2 & x_3^2y_3 \\ x_1y_1^2 & x_2y_2^2 & x_3y_3^2 \\ y_1^3 & y_2^3 & y_3^3 \\ x_1^4 & x_2^4 & x_4^4 \\ x_1^3y_1 & x_2^3y_2 & x_3^3y_3 \\ x_1^2y_1^2 & x_2^2y_2^2 & x_3^2y_3^2 \\ x_1y_1^3 & x_2y_2^3 & x_3y_3^3 \\ y_1^4 & y_2^4 & y_3^4 \\ \vdots & \vdots & \vdots \end{bmatrix}$$

simplified:

$$\begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ x_1y_1 & x_2y_2 & x_3y_3 \\ x_1^3 & x_2^3 & x_3^3 \\ x_1^2y_1 & x_2^2y_2 & x_3^2y_3 \end{bmatrix} \begin{bmatrix} x_1 & & \\ & x_2 & \\ & & x_3 \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & x_3 \\ x_1^2 & x_2^2 & x_3^2 \\ x_1y_1 & x_2y_2 & x_3y_3 \\ x_1^2y_1 & x_2^2y_2 & x_3^2y_3 \\ x_1^4 & x_2^4 & x_4^4 \\ x_1^3y_1 & x_2^3y_2 & x_3^3y_3 \end{bmatrix}$$

– finding the x-roots: let $D_x = \text{diag}(x_1, x_2, \ldots, x_s)$, then

$$S_1 K D_x = S_x K,$$

where S_1 and S_x select rows from K wrt. shift property

– reminiscent of **Realization Theory**

We have

$$S_1 KD_x = S_x K$$

However, K is not known, instead a basis Z is computed that satisfies

$$ZV = K$$

Which leads to

$$(S_x Z)V = (S_1 Z)VD_x$$

It is possible to shift with y as well...

We find

$$S_1 K D_y = S_y K$$

with D_y diagonal matrix of y-components of roots, leading to

$$(S_y Z)V = (S_1 Z)V D_y$$

Some interesting results:

- same eigenvectors V!
- $(S_y Z)^{-1}(S_1 Z)$ and $(S_x Z)^{-1}(S_1 Z)$ commute

Algorithm

1. Fix a monomial ordering scheme
2. Construct coefficient matrix M to sufficiently large dimensions
3. Compute basis for nullspace of M: nullity s and Z
4. Find s linear independent rows in Z
5. Choose shift function, e.g., x
6. Solve the GEVP
$$(S_2 Z)V = (S_1 Z)VD_x$$
S_1 selects linear independent rows in Z; S_2 the rows that are 'hit' by the shift

($S_1 Z$ and $S_2 Z$ can be rectangular as long as $S_1 Z$ contains s linear independent rows)

The null space is an nD state sequence

The null space of the Macaulay matrix is the interface between polynomial system and nD state space description

- nD state-space model (for $n = 2$)

$$
\begin{aligned}
v(k+1, l) &= A_x v(k, l) \\
v(k, l+1) &= A_y v(k, l)
\end{aligned}
$$

- null space of Macaulay matrix: nD state sequence

$$
\left(\begin{array}{c|cc|ccc|cccc}
| & | & | & | & | & | & | & | & | & | \\
v_{00} & v_{10} & v_{01} & v_{20} & v_{11} & v_{02} & v_{30} & v_{21} & v_{12} & v_{03} \\
| & | & | & | & | & | & | & | & | & |
\end{array}\right)^T =
$$

$$
\left(\begin{array}{c|cc|c|cccc}
| & | & | & & | & | & | & | \\
v_{00} & A_x v_{00} & A_y v_{00} & \cdots & A_x^3 v_{00} & A_x^2 A_y v_{00} & A_x A_y^2 v_{00} & A_y^3 v_{00} \\
| & | & | & & | & | & | & |
\end{array}\right)^T
$$

- shift-invariance property, e.g., for y:

$$\begin{pmatrix} - v_{00} - \\ \hline - v_{10} - \\ - v_{01} - \\ \hline - v_{20} - \\ - v_{11} - \\ - v_{02} - \end{pmatrix} A_y^T = \begin{pmatrix} - v_{01} - \\ \hline - v_{11} - \\ - v_{02} - \\ \hline - v_{21} - \\ - v_{12} - \\ - v_{03} - \end{pmatrix},$$

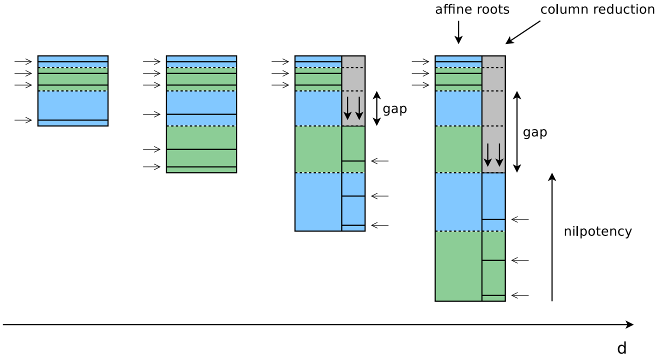
- corresponding nD system realization

$$\begin{aligned} v(k+1,l) &= A_x v(k,l) \\ v(k,l+1) &= A_y v(k,l) \\ v(0,0) &= v_{00} \end{aligned}$$

- choice of basis null space leads to different system realizations
- eigenvalues of A_x and A_y invariant: x and y components of roots

Roots at infinity? *Mind the Gap!*

- dynamics in the null space of $M(d)$ for increasing degree d
- nilpotency gives rise to a 'gap'
- mechanism to count and separate affine from infinity



Roots at infinity lead to nD descriptor systems

Weierstrass Canonical Form decouples affine/infty

$$\left[\begin{array}{c} v(k+1) \\ \hline w(k-1) \end{array}\right] = \left[\begin{array}{c|c} A & 0 \\ \hline 0 & E \end{array}\right] \left[\begin{array}{c} v(k) \\ \hline w(k) \end{array}\right]$$

nD descriptor state-space model

$$\begin{array}{rcl} v(k+1,l) & = & A_x v(k,l) \\ v(k,l+1) & = & A_y v(k,l) \\ \\ w(k-1,l) & = & E_x w(k,l) \\ w(k,l-1) & = & E_y w(k,l) \end{array}$$

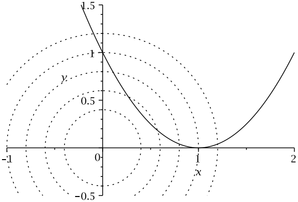
with nilpotent E_x and E_y

Polynomial root-finding is a question in linear algebra and multidimensional realization theory

- Solving multivariate polynomials
  - Macaulay formulation makes it a linear algebra question
  - Realization theory in null space of Macaulay matrix
  - nD autonomous (descriptor) state-space models
- Decisions made based upon (numerical) rank
  - # roots (nullity)
  - # affine roots (column reduction)
- Mind the Gap phenomenon: affine vs. infinity roots
- Not discussed
  - multiplicity of roots
  - column-space based method
  - over-constrained systems

Polynomial Optimization Problems are EVPs

$$\begin{array}{ll} \min_{x,y} & x^2 + y^2 \\ \text{s. t.} & y - x^2 + 2x - 1 = 0 \end{array}$$



Lagrange multipliers give conditions for optimality:

$$L(x, y, z) = x^2 + y^2 + z(y - x^2 + 2x - 1)$$

we find

$$\begin{array}{lll} \partial L/\partial x = 0 & \rightarrow & 2x - 2xz + 2z = 0 \\ \partial L/\partial y = 0 & \rightarrow & 2y + z = 0 \\ \partial L/\partial z = 0 & \rightarrow & y - x^2 + 2x - 1 = 0 \end{array}$$

Observations:

- everything remains polynomial
- system of polynomial equations
- shift with objective function to find minimum/maximum

Let

$$A_x V = xV$$

and

$$A_y V = yV$$

then find min/max eigenvalue of

$$(A_x^2 + A_y^2)V = (x^2 + y^2)V$$

Outline

Recent advances in the Macaulay spirit

Computing state-recursion polynomials
Batselier K., Wong N., *"Computing the state recursion polynomials for discrete linear mD systems"*, Automatica, vol. 64, pp.254-261, 2016.

"The CPD appears to be the joint EVD of the multiplication tables"
Vanderstukken J., Stegeman A., De Lathauwer L., *"Systems of polynomial equations, higher-order tensor decompositions and multidimensional harmonic retrieval: A unifying framework."* (two-part paper), KU Leuven ESAT-STADIUS TR 17-133 and TR 17-134, 2017.

Block-shifting with an objective function
Vermeersch C., De Moor B., *"Globally Optimal Least-Squares ARMA Model Identification is an Eigenvalue Problem"*, IEEE Control Systems Letters, 3:4, 1062–1067, 2019.

Exploring the non-autonomous case
Vergauwen B., Agudelo M., De Moor B., *"Order estimation of two dimensional systems based on rank decisions"*, IEEE CDC 2018.

Adapting the choice of basis for improved numerical stability
Telen S., Mourrain B., Van Barel M., *"Solving Polynomial Systems via Truncated Normal Forms"*, SIAM J Matrix Anal Appl, 39:3, 1421–1447, 2018.

Outline

The history of polynomial system solving

Univariate polynomials and eigenvalue decompositions

Multivariate polynomials and nD systems

Recent developments in the Macaulay spirit

Conclusions and Perspectives

Conclusions

- bridging the gap between algebraic geometry and engineering
- finding roots: linear algebra and realization theory!
- extension to over-constrained systems
- polynomial optimization: extremal eigenvalue problems

Open Problems

Many challenges remain

- exploiting sparsity and structure of M
- efficient (more direct) construction of the eigenvalue problem
- algorithms to find the minimizing solution efficiently (inverse power method?)

Dreesen P., Batselier K., De Moor B., *"Multidimensional realisation theory and polynomial system solving"*, Int J Control, 91:12, pp. 2692–2704, 2018. (arXiv 1805.02253)

Thank you for listening!

Acknowledgements

This work was supported by the KU Leuven Research Fund, FWO (EOS proj 30468160 (SeIMA), SBO S005319N, I013218N, TBM T001919N, PhD grants (SB/1SA1319N, SB1S93918, SB/151622). This research received funding from the Flemish Government (AI Research Program). BDM and PD are affiliated to Leuven.AI. This project received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant 885682). Part of this work was done while PD held a PhD scholarship of Flanders' Agency for Innovation by Science and Technology (IWT Vlaanderen).

Polynomial system solving and/by multidimensional realization theory

'Back to the Roots'

Philippe Dreesen[1] Kim Batselier[2] Bart De Moor[1]

[1] KU Leuven, ESAT/STADIUS
[2] TU Delft, 3mE-DCSC

Jan 28, 2022

Control e-seminar
Mechatronics in Mechanical and Process Engineering (MEC)
University of Kaiserslautern