Real root finding of determinants of linear matrices

D. Henrion [1,2,3] **S. Naldi** [1,2] **M. Safey El Din** [4]

[1]CNRS; LAAS; 7 avenue du colonel Roche, F-31400 Toulouse; France

[2]Université de Toulouse; LAAS, F-31400 Toulouse, France

[3]Faculty of Electrical Engineering, Czech Tech. Univ. in Prague, Technickà 2, CZ-16626 Prague, Cz. Rep.

[4]Équipe-projet POLSYS (INRIA/UPMC/LIP6). Laboratoire Lip6, 4 Place Jussieu F-75252, Paris.

Project ANR GeoLMI: www.laas.fr/geolmi

Geometry and Algebra of Linear Matrix Inequalities
C.I.R.M. Luminy, Nov 12-16, 2013

Problem statement

Given k, n integers and $A_0, A_1 \dots A_n \in \mathbb{Q}^{k \times k}$, find a sample set in

$$\left\{ \mathbf{x} \in \mathbb{R}^n \;\middle|\; \det A(\mathbf{x}) = 0 \right\}$$

where $A(\mathbf{x}) = A_0 + \mathbf{x}_1 A_1 + \dots + \mathbf{x}_n A_n$.

It is the locus of $\mathbf{x} \in \mathbb{R}^n$ such that $\operatorname{rank} A(\mathbf{x}) < k$.

* Compute a point in every connected component
* $n = 1$: Real Eigenvalue Problem
* $n \geq 2$: Positive dimensional problem
* Real solutions of $\det(A) = 0$, $\deg(\det(A)) = k$
* First step for solving $\det(A) > 0$ or $\det(A) \geq 0$.

Motivations

Dedicated algorithms for real solving structured systems

1. Symmetric matrices $\rightarrow$ Linear Matrix Inequalities:

$$\boxed{\text{Find } \mathbf{x} \text{ such that } A(\mathbf{x}) \succeq 0.}$$

Optimize a linear function over a spectrahedron / LMI set:

$$\begin{aligned} \min \quad & c_1\mathbf{x}_1 + \ldots + c_n\mathbf{x}_n \\ s.t. \quad & A_0 + \mathbf{x}_1 A_1 + \ldots + \mathbf{x}_n A_n \succeq 0. \end{aligned}$$

2. Determinantal semi-algebraic sets.

Stability Analysis of Dynamical Systems

Hurwitz Criterion: positivity of principal minors of Hurwitz matrices

$$P(z) = \sum_{i=0}^{n} \mathbf{x}_i z^{n-i} \qquad H_n = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_3 & \cdots & \\ \mathbf{x}_0 & \mathbf{x}_2 & \cdots & \\ \vdots & \vdots & \ddots & \\ & & & \mathbf{x}_n \end{bmatrix} \qquad \begin{array}{l} \det \Delta_1 > 0 \\ \vdots \\ \det \Delta_{n-1} > 0 \\ \det H_n > 0. \end{array}$$

State of the art

Existence of real roots

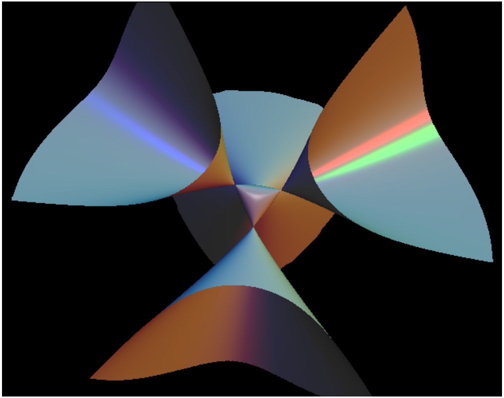
- $F(X_1 \ldots X_n) = 0$, $\deg F = k$: complexity $\mathbf{k}^{\mathcal{O}(\mathbf{n})}$, hard in practice (*Basu, Pollack, Roy, Grigoriev, Vorobjov, Heintz, Solerno*);
- Using polar varieties (*Bank, Giusti, Heintz, Mbakop, Pardo, Safey, Schost*):
  - $\mathcal{O}(\mathbf{k}^{\mathbf{3n}})$: regular case
  - $\mathcal{O}(\mathbf{k}^{\mathbf{4n}})$: singular case.
- Software with Gröbner Bases (`FGb`,`RAGlib`)
- Quadratic case. Complexity: polynomial in n, expon. in the codimension

Determinantal structure

- Extensively studied in Algebraic Geometry
- Zero-dimensional case: Gröbner Bases methods $\rightsquigarrow$ complexity bounds (*Faugère, Safey, Spaenlehauer*)

Generic singularity of determinants

We deal with singular varieties in positive dimension:



Contributions

A dedicated algorithm

- Computing at least one real point in every connected component of a hypersurface defined by the determinant of a linear matrix;
- Strong use of the geometrical structure
- Complexity:

k	:	size of the matrix
n	:	number of variables

$\rightarrow$ polynomial in $k, n, \binom{k+n}{n}$

- Asymptotically the complexity is $O(\mathbf{k}^{\alpha \mathbf{n}})$ with $\alpha < 3$.
- Implementation in `RAG` : experiments.

Kernel modelling: well known desingularization

From one polynomial equation ($\det A(\mathbf{x}) = 0$) we build the **bilinear system**

$$\begin{bmatrix} F_1(\mathbf{x}, \mathbf{y}) \\ \vdots \\ F_k(\mathbf{x}, \mathbf{y}) \end{bmatrix} = A(\mathbf{x}) \cdot \begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_k \end{bmatrix} = 0$$

It defines a set $V \subset \mathbb{C}^n \times \mathbb{P}_{\mathbb{C}}^{k-1}$.

- bilinear equations in $\mathbf{x}, \mathbf{y}$
- $\dim V = n - 1$, $\deg V = k + \binom{k}{2} + \cdots + \binom{k}{n}$
- $\Pi_{\mathbf{x}}(V) = \{\det A = 0\}$
- Real Points in V: $(\mathbf{x}, \mathbf{y}) \in V \cap \mathbb{R}^{n+k} \to \mathbf{x} \in \{\det A = 0\} \cap \mathbb{R}^n$

Result: generic smoothness and equidimensionality of affine sections of V.

Critical Points Method

Algebraic description Critical points are solutions of polynomial systems.

Efficient method In [Safey, Schost, 2003]: *polar varieties* $\rightarrow$ efficient algorithm for computing sample points (smooth case).

A roadmap of our algorithm:

1. Choose a line L with good properties
2. critical points of the restriction of $\pi_L : \mathbb{C}^{n+k} \rightarrow L$ to V
3. generic fibers of π_L + recursive call

Good properties: (ensured by generic projections)

- finite number of critical points
- $\pi_L(C)$ closed for all conn. component C of the $\{\det A = 0\}$

We prefer particular projections

If we choose a generic line in $\mathbb{R}^{n+k}$:

$$F_1(\mathbf{x}, \mathbf{y}) = \ldots = F_k(\mathbf{x}, \mathbf{y}) = a_1\mathbf{x}_1 + \ldots + a_n\mathbf{x}_n + a_{n+1}\mathbf{y}_1 + \ldots + a_{n+k}\mathbf{y}_k = 0.$$

If we choose a generic line in $\mathbb{R}^n$:

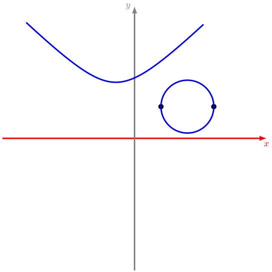
$$F_1(\mathbf{x}, \mathbf{y}) = \ldots = F_k(\mathbf{x}, \mathbf{y}) = a_1\mathbf{x}_1 + \ldots + a_n\mathbf{x}_n = 0.$$

Geometrical idea of the algorithm

* C connected component of $\{\mathbf{x} \in \mathbb{R}^n \mid \det A(\mathbf{x}) = 0\}$;
* $\Pi_a(\mathbf{x}, \mathbf{y}) = a \cdot \mathbf{x}$ is generically closed: $\Pi_a(C) = \mathbb{R}$ or $\partial \Pi_a(C) \neq \emptyset$;
* $\Pi_a(\mathbf{x}, \mathbf{y}) \in \partial \Pi_a(C) \implies \mathbf{x}$ critical point (because V is smooth).

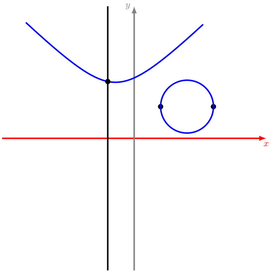
$\downarrow$

Either:



$C \cap (\mathrm{Crit}(\Pi_a, V)) \neq \emptyset$

or:



$\left(\Pi_a^{-1}(0)\right) \cap C \neq \emptyset$

Lagrange Systems

From k bilinear equations $F_1(\mathbf{x}, \mathbf{y}), \ldots, F_k(\mathbf{x}, \mathbf{y})$ to the **trilinear system**

$$
\begin{array}{rcl}
F_1(\mathbf{x}, \mathbf{y}) = \cdots = F_k(\mathbf{x}, \mathbf{y}) & = & 0 \\
\mathbf{z}_1 \nabla_{\mathbf{x}} F_1 + \ldots + \mathbf{z}_k \nabla_{\mathbf{x}} F_k & = & [a_1, \ldots, a_n]^T \\
\mathbf{z}_1 \nabla_{\mathbf{y}} F_1 + \ldots + \mathbf{z}_k \nabla_{\mathbf{y}} F_k & = & [0, \ldots, 0]^T
\end{array}
$$

where $a = (a_1, \ldots, a_n) \in \mathbb{R}^n$.

- $\Pi_a(\mathbf{x}, \mathbf{y}) = a_1\mathbf{x}_1 + \ldots + a_n\mathbf{x}_n$
- $\mathbf{z} =$ Lagrange multipliers
- $(\mathbf{x}, \mathbf{y})$ critical for $\Pi_a \iff \exists\ \mathbf{z}\ :\ (\mathbf{x}, \mathbf{y}, \mathbf{z})$ is a solution
- $\#$ polynomials $= \#$ variables

Result 1: generic zero-dimensionality of the Lagrange system.
Result 2: Projections of the components on a generic line in $\mathbb{R}^n$ are closed .

Output and Complexity

Lagrange system is zero-dimensional $\rightarrow$ **rational parametrization**:

$$\mathbf{x}_1 = \frac{P_1(t)}{P_0(t)}, \ldots, \mathbf{x}_n = \frac{P_n(t)}{P_0(t)}, \quad P(t) = 0.$$

Suppose the Lagrange system is given by polynomials

$$F_1(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \ldots = F_s(\mathbf{x}, \mathbf{y}, \mathbf{z}) = 0.$$

The complexity cost to obtain a parametrization is essentially $O(\Delta^2)$ where

$$\Delta = \max_{1 \leq t \leq s} \deg \left\{ F_1 = \ldots = F_t \right\}.$$

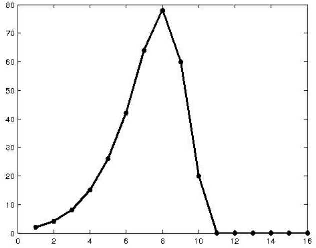
Final step

Use Multilinear Bézout bounds to estimate Δ.

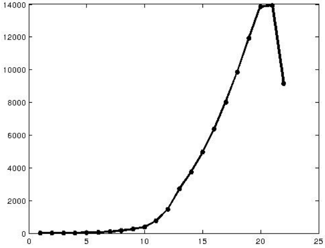
Multilinear Bézout Bounds

The system has a **trilinear structure** $\rightarrow$ better bounds on the degree. Two typical cases:

x axis : values of t
y axis : values of the Multilinear Bézout Bound



$(k, n) = (4, 10)$



$(k, n) = (10, 4)$

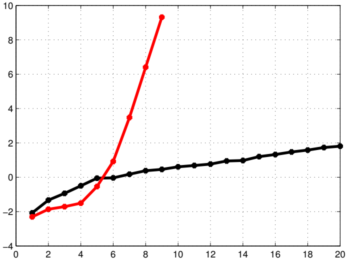
Interesting facts:

1. for $n \gg k$ there is no critical points at Step 1
2. non-sharpness of the last bound
3. $\Delta \ll \binom{k+n}{n}^{1.5} \rightarrow$ complexity is $O(\binom{k+n}{n}^3)$

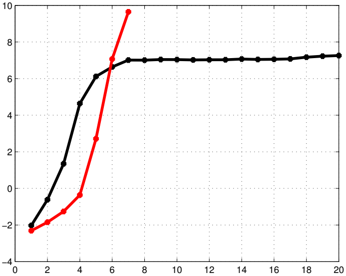
Timings in the dense case

Using FGb's Faugere Grobner basis engine.

x axis : values of n= number of variables
y axis : time in logarithmic scale, dense case



$k = 3$



$k = 4$

- $k = 4,\ n = 4 \rightarrow$ time ≈ 2 min
- $k = 4,\ n = 8 \rightarrow$ time ≈ 25 min
- $k = 4,\ n = 12 \rightarrow$ time ≈ 50 min
- $k = 4,\ n = 16 \rightarrow$ time ≈ 1.5 h

Remark: dense case is the worst → "sparse" cases much easier

Conclusions and outlooks

Conclusions

- an efficient algorithm solving the problem
- exploiting the geometry of determinants
- a new class of problems solved in polynomial time

Outlooks

- improve the implementation
- bigger rank defects of linear matrices: $A(\mathbf{x}) \cdot \mathbf{y}^{(1)} = \cdots = A(\mathbf{x}) \cdot \mathbf{y}^{(k-R)} = 0$
- interior of the LMI set: $A(\mathbf{x}) \succeq 0$
- bound the number of real connected components

Thank you