Computing the Nearest Singular Matrix Polynomial

Joseph Haraldson

with Mark Giesbrecht and George Labahn

David R. Cheriton School of Computer Science
University of Waterloo

January 24, 2018

Our Problem: Find the Nearest Singular Matrix Polynomial

Given a matrix $A \in \mathbb{R}[t]^{n\times n}$, find $\widehat{A} \in \mathbb{R}[t]^{n\times n}$ such that $\widehat{A}$ is singular and $\|A - \widehat{A}\|$ is minimized.

Equivalently, for some "reasonable" norm $\|\cdot\|$, solve the optimization problem

$$\min_{\widehat{A},b} \|A - \widehat{A}\| \text{ subject to } \begin{cases} \widehat{A}b = 0 \\ \|b\| = 1, \end{cases}$$

where $\|A_{ij}\|_2 = \sqrt{\sum_{1\leq k \deg A_{ij}} A_{ijk}^2}$ and $\|A\| = \|A\|_F = \sqrt{\sum_{1\leq i,j\leq n} \|A_{ij}\|_2^2}$.

- A matrix polynomial $A \in \mathbb{R}[t]^{n\times n}$ is singular if $\det(A) \equiv 0$
- Sometimes we say that A is irregular
- Singular $\implies \exists b \in \mathbb{R}[t]^{n\times 1}\backslash\{0\}$ such that $Ab \equiv 0$

Previous Algorithms

We would like an algorithm that is ...

- Numerically robust with rapid local convergence w.r.t. $\|\cdot\|$

Existing Algorithms:

- Restricted SVD [De Moor '93, '94]
  - At best linear convergence
- Structured Total Least Norm (STLN) [Rosen, Park & Glick '96]
  - Super linear convergence (not quadratic) [Lemmerling '99]
- Variable Projection [Golub & Pereyra '73, '03]
  - Use Gauss-Newton (Pure Newton)
  - Converges super-linearly (at least quadratic) if normalized
  - Problem size is much larger
- Lift and Project methods [Spaenlehauer & Schost '16]

Our Contributions

Recall we want to solve the optimization problem

$$\min_{\Delta A, b} \|\Delta A\|_F \text{ subject to } \begin{cases} \|b\|_F = 1 \\ (A + \Delta A)b = 0. \end{cases}$$

1. Prove that minimal solutions exist
2. Prove that minimal solutions are isolated
   - $\inf \|\Delta A_{opt} - \Delta A^{\star}_{opt}\|_F > 0$ when rank $A = n$ for two distinct minimal perturbations A_{opt} and $A^{\star}_{opt}$
3. Show that the problem is (locally) well-posed
   - Optimal value is isolated around minimizers
   - Algorithm is provably locally stable
4. Derive and implement a quadratically convergent algorithm

Applications

- Stability of solutions to linear time invariant systems
  - [Byers & Nicols '93] and [Byers, He & Mehrmann '98]
- Stability of polynomial eigenvalue problems
- Approximate GC(R)D of Ore polynomials
  - [Giesbrecht, H & Kaltofen '16]

Existence of Solutions

Theorem

Let $\Delta A \in \mathbb{R}[t]^{n \times n}$ *have the same support as* A*, that is* $\deg \Delta A_{ij} \leq \deg A_{ij}$*. The optimization problem*

$$\min_{\Delta A, b} \|\Delta A\| \text{ subject to } \begin{cases} \|b\| = 1 \\ (A + \Delta A)b = 0 \end{cases}$$

has an attainable global minimum $(\Delta A^\star, b^\star)$*.*

- Nearest singular matrix polynomial always exists
- i.e. $\exists(\Delta A^\star, b^\star)$ s.t. $(A + \Delta A^\star)b^\star = 0$ and $\|\Delta A^\star\|$ is minimal
- Minimization of $\|\Delta A\|$ under other structures has a solution if there is a finite solution to $(A + \Delta A)b = 0$
- In general bi-linear feasibility over $\mathbb{Q}$ is NP-Hard

Embedding $\mathbb{R}[t]^{n\times n}$ in $\mathbb{R}^{n(\mu+d)\times n\mu}$

Embed multiplication as a linear algebra problem over $\mathbb{R}$

- Recall that A is $n \times n$ with entries at most degree d
- Define $\mu = nd + 1$

Lemma

If $A \in \mathbb{R}[t]^{n\times n}$ is singular then there exists $b \in \ker A$ such that $\deg b \leq nd = \mu - 1$.

- Embed A as $\mathcal{A} \in \mathbb{R}^{n(\mu+d)\times n\mu}$ and b as $\mathcal{B} \in \mathbb{R}^{n\mu\times 1}$
- $Ab = 0 \iff \mathcal{A}\mathcal{B} = 0$
- SVD provides a cheap lower bound on the distance to a singular matrix, hence a singular matrix polynomial

Embedding Example

The embedding preserves kernel vectors...

$$A = \begin{pmatrix} t^2 - 1 & t + 1 \\ t^2 - 2t + 1 & t - 1 \end{pmatrix} \quad \text{and} \quad \ker A = \begin{pmatrix} -1 \\ t - 1 \end{pmatrix}$$

$$\mathcal{A} = \left(\begin{array}{ccccc|ccccc}
-1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
1 & 0 & -1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 1 & 0 & -1 & 0 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
\hline
1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
-2 & 1 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\
1 & -2 & 1 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\
0 & 1 & -2 & 1 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 & 1 & -1 \\
0 & 0 & 0 & 1 & -2 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0
\end{array}\right) \qquad
\ker \mathcal{A} = \left(\begin{array}{cccc}
-1 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 \\
0 & 0 & -1 & 0 \\
0 & 0 & 0 & -1 \\
0 & 0 & 0 & 0 \\
\hline
-1 & 0 & 0 & 0 \\
1 & -1 & 0 & 0 \\
0 & 1 & -1 & 0 \\
0 & 0 & 1 & -1 \\
0 & 0 & 0 & 1
\end{array}\right)$$

- The embedding does not preserve rank information

Separation Bounds

Theorem

Suppose ΔA and $\Delta A^\star$ are distinct (local) minimal solutions then

$$\|\Delta A - \Delta A^\star\|_F \geq \frac{\|\Delta \mathcal{A} - \Delta \mathcal{A}^\star\|_2}{nd+1} \geq \frac{\sigma_{\min}(\mathcal{A})}{nd+1}.$$

Corollary

Minimal solutions $(\Delta A^\star, b^\star)$ are isolated modulo equivalence classes of $b^\star$.

- Normalize b over $\mathbb{R}[t] \implies$ locally unique solutions
- Proof uses first-order information
- Similar separation holds under $\|\cdot\|_1$ and other norms

Lagrange Multipliers

Definition

The Lagrangian is defined as

$$L = \|\Delta A\|_F^2 + \|\mathcal{B}\|_2^2 - 1 + \lambda^T \begin{pmatrix} \text{vec}((\mathcal{A} + \Delta\mathcal{A})\mathcal{B}) \\ \mathcal{B}^T\mathcal{B} - 1 \end{pmatrix}.$$

- ∇ is the gradient operator
- Let $x = (\text{vec}(\Delta A)^T, \text{vec}(b)^T)^T$
- ∇^2 (∇^2_{xx}) is the Hessian operator (w.r.t. to x)
- First order necessary (Karush-Kuhn-Tucker (KKT)) conditions

$$\nabla L(\Delta A^\star, \mathcal{B}^\star, \lambda^\star) = 0$$

- Idea is to solve $\nabla L = 0$ by Newton's method, i.e. compute

$$\begin{pmatrix} x^{k+1} \\ \lambda^{k+1} \end{pmatrix} = \begin{pmatrix} x^k + \Delta x^k \\ \lambda^k + \Delta\lambda^k \end{pmatrix} \text{ such that } \nabla^2 L \begin{pmatrix} \Delta x \\ \Delta\lambda \end{pmatrix} = -\nabla L$$

Lagrange Multipliers

Definition

The Jacobian of the constraints is

$$J = \nabla \begin{pmatrix} \text{vec}((\mathcal{A} + \Delta\mathcal{A})\mathcal{B}) \\ \mathcal{B}^T\mathcal{B} - 1 \end{pmatrix}^T = \begin{pmatrix} \psi(\mathcal{B}) & \mathcal{A} + \Delta\mathcal{A} \\ 0 & 2\mathcal{B}^T \end{pmatrix}.$$

- $\psi(\mathcal{B})\text{vec}(A + \Delta A) = 0 \iff (\mathcal{A} + \Delta\mathcal{A})\mathcal{B} = 0$
- J has full rank $\implies$ minimal solutions $(\Delta A^\star, b^\star)$ are isolated
- Multiple kernel vectors $\implies$ J is rank deficient

Definition

The Hessian matrix of L is $\nabla^2 L = \begin{pmatrix} \nabla^2_{xx} L & J \\ J^T & 0 \end{pmatrix}$.

$\nabla^2 L$ has full rank $\iff$ J has full rank

Minimal Kernel Embedding

Definition

A kernel vector $\mathcal{B}$ corresponding to $b \in \ker(A + \Delta A)$ is *minimally degree embedded* in $\mathcal{A} + \Delta\mathcal{A}$ if

1. $\ker(\mathcal{A} + \Delta\mathcal{A}) = \text{span}(\mathcal{B})$ and b is primitive
2. b comes from a column echelon reduced basis.

Theorem

J has full rank at $(\Delta A^\star, b^\star)$ if $b^\star$ is minimally degree embedded.

- Minimally embed by deleting rows/columns of $\mathcal{A}/\mathcal{B}$
- Compute via orthogonal eliminations

Corollary

Newton's method converges quadratically to $(\Delta A^\star, b^\star)$ with a suitable initial guess.

Minimal Kernel Example

$$A = \begin{pmatrix} t^2 - 1 & t + 1 \\ t^2 - 2t + 1 & t - 1 \end{pmatrix} \quad \text{and} \quad \ker A = \begin{pmatrix} -1 \\ t - 1 \end{pmatrix}$$

The minimal embedding gives us

$$\mathcal{A}_{min} = \left(\begin{array}{c|cc} -1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ \hline 1 & -1 & 0 \\ -2 & 1 & -1 \\ 1 & 0 & 1 \end{array}\right) \quad \text{and} \quad \mathcal{B}_{min} = \left(\begin{array}{c} -1 \\ \hline -1 \\ 1 \end{array}\right)$$

Implementation Information

Our Implementation:

- Initialize ΔA and b via SVD (or other method)
- Minimally degree embed system over $\Delta\mathcal{A}$ and $\mathcal{B}$
- Compute Lagrange multipliers via linear least squares

In general:

- Use any method to approximate $\nabla L = 0$
- Minimally degree embed and switch to Newton's method later

Cost Per Iteration:

- Maximum cost per iteration is $O(\text{size}^6) = O(n^{12}d^6)$
- Exploiting structure and sparsity reduces costs considerably

Algorithm

Input: Full rank $A \in \mathbb{R}[t]^{n \times n}$ with structure ΔA and
$C \in \mathbb{R}[t]^{n \times n}$ with (approx) kernel vector b

Output: $A + \Delta A^{\star}$ or an indication of failure

1. Embed input over $\mathbb{R}$
2. Compute λ^0 by solving $\nabla L|_{x^0} = 0$ via linear least squares
3. Compute $\begin{pmatrix} x + \Delta x \\ \lambda + \Delta \lambda \end{pmatrix}$ until $\left\| \begin{pmatrix} \Delta x \\ \Delta \lambda \end{pmatrix} \right\|_2$ is sufficiently small
4. Return the locally optimal solution or an indication of failure

Example

- We generate a singular matrix polynomial of rank 2
- Create A by adding noise; relative error is $\dfrac{\|\Delta A\|}{\|A\|} \approx .0512$
- Compare algorithm with different approximate kernel vectors

$$\frac{A}{\|A\|} = \frac{\Delta A}{\|A\|} + \begin{pmatrix} .11t^3 - .18t^2 - .075t + .11 & .062t^3 + .12t^2 + .15t - .12 & 0 & 0 \\ -.034t^3 - .057t^2 + .18t - .019 & .075t^3 - .0021t^2 - .16t + .053 & 0 & 0 \\ .092t^3 + .11t^2 - .42t + .062 & -.29t^3 - .21t^2 + .48t - .10 & 0 & 0 \\ .13t^3 - .079t^2 - .13t - .088 & .075t^3 - .0042t^2 - .042t + .066 & 0 & 0 \end{pmatrix}$$

$$+ \begin{pmatrix} 0 & 0 & .062t^3 - .0084t^2 + .16t - .066 & -.084t^3 - .057t^2 - .048t - .029 \\ 0 & 0 & -.0042t^3 - .017t^2 - .062t - .053 & -.057t^3 + .057t^2 - .0084t + .013 \\ 0 & 0 & -.0042t^3 - .038t^2 + .18t + .034 & .26t^3 + .048t - .026 \\ 0 & 0 & .070t^3 + .029t^2 + .14t + .10 & -.097t^3 + .026t^2 + .042t + .017 \end{pmatrix}$$

Example Cont.

$$b_1 = \begin{pmatrix} 0.06105t^4 + 0.06919t^3 - 0.30118t^2 + 0.01628t \\ 0.232t^4 - 0.0936t^3 - 0.334t^2 - 0.0855t - 0.0244 \\ 0.0 \\ 0.265t^4 + 0.0326t^3 - 0.789t^2 + 0.122t + 0.0977 \end{pmatrix}$$

$$b_2 = \begin{pmatrix} -0.118t^4 - 0.102t^3 - 0.172t^2 - 0.314t + 0.251 \\ -0.0392t^4 - 0.0784t^3 + 0.219t^2 - 0.165t + 0.188 \\ 0.255t^4 + 0.0314t^3 - 0.760t^2 + 0.118t + 0.0941 \\ 0.0 \end{pmatrix}$$

Example Cont.

- $\frac{||\Delta A_{b_1}||}{||A||} \approx .002756$
- $\frac{||\Delta A_{b_2}||}{||A||} \approx .002735$
- Actual separation: $\frac{||A_{b_1} - A_{b_2}||}{||A||} \approx .002761$
- Lower bound on separation: $\approx .000051$

Iteration	$\|\|x_{b_1}^{i-1} - x_{b_1}^{i}\|\|_2$	$\|\|x_{b_2}^{i-1} - x_{b_2}^{i}\|\|_2$
1	108.291	18.797
2	30.335	12.297
3	0.9396	1.008
4	2.6939e-3	6.261e-4
5	5.5337e-9	2.2271e-11
6	8.06113e-20	1.8074e-25
7	0	0

Some Experiments

- Generate a singular matrix polynomial and add noise
- Initialize with nearby singular matrix polynomial w/ kernel

n	d	iterations	$\frac{\|\mathcal{A}-\mathcal{A}_{init}\|_F}{\|\mathcal{A}\|_F}$	$\frac{\|\Delta\mathcal{A}\|_F}{\|\mathcal{A}\|_F}$	Status
6	2	7	1.64e-02	3.29e-03	
6	6	6	1.58e-04	3.40e-05	
6	6	1	1.51e-02	6.85e-03	S-FAIL
6	10	6	1.85e-04	4.16e-05	
6	10	2	1.70e-02	2.98e-02	FAIL
9	2	5	1.69e-04	3.75e-05	
9	2	1	1.66e-02	6.11e-03	S-FAIL
9	4	9	1.68e-02	2.29e-03	
12	2	5	1.75e-04	2.21e-05	
12	2	9	1.67e-02	2.28e-03	

- S-FAIL: In region of convergence; kernel vector not primitive
- FAIL: Outside region of convergence

Completed/In Progress

What we've done since...

- Quadratically convergent algorithm for "*rank-at-most*" approximation [Giesbrecht, H & Labahn '17]
- Technique can be adapted to other affine structures:
  - Approximate (multivariate) polynomial GCD
  - Approximate multivariate polynomial factorization i.e. use [Kaltofen, May, Yang & Zhi' 08]
- Implementation online in Maple

...and we are currently looking into

- Hardness results for nearest singular matrix polynomial
- Matrix polynomial Approximate GCD
- Nearest "interesting" Smith form
  - Wilkinson's Problem for Matrix Polynomials