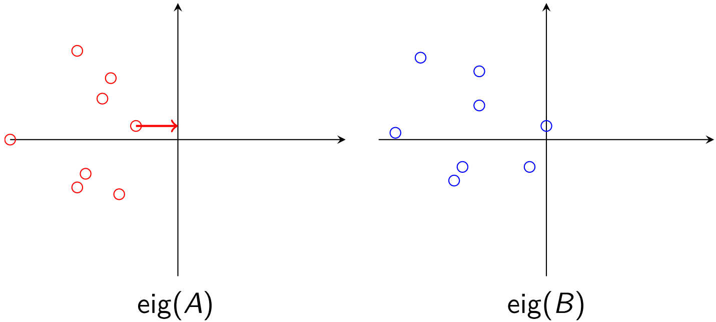
Computing the nearest stable matrix via optimization on matrix manifolds

Federico Poloni (University of Pisa)
Joint work with V. Noferini (Aalto university)

Oselot online seminars, October 2020

Two problems: "the ∃ problem"



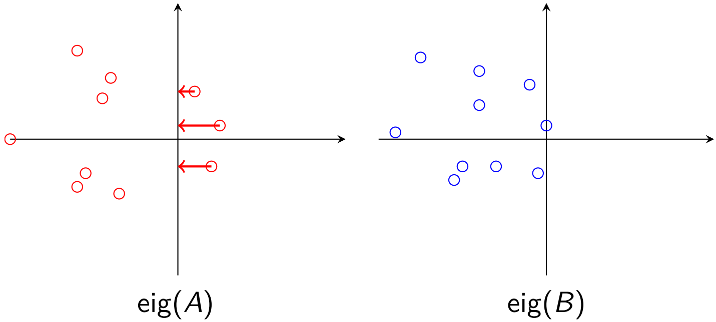
Given Hurwitz stable $A \in \mathbb{C}^{n\times n}$, find nearest non-stable B.

More generally: given A and closed region $\Omega \subseteq \mathbb{C}$, find

$$\min_{\substack{B\in\mathbb{C}^{n\times n}\\ \exists\,\lambda\in\Lambda(B)\cap\Omega}} \|A-B\|_F$$

Application: how much noise can we add so that $\dot{x} = Ax$ stays stable?

Two problems: "the ∀ problem"



Given non-stable $A \in \mathbb{C}^{n\times n}$, find nearest stable B.

More generally: given A and closed region $\Omega \subseteq \mathbb{C}$, find

$$\min_{\substack{B\in\mathbb{C}^{n\times n}\\ \Lambda(B)\subseteq\Omega}} \|A-B\|_F$$

Application: noise made $\dot{x} = Ax$ unstable; how to 'fix' A?

Comparing the two problems

When A is non-normal, there is no simple solution.

Previous work on these problems or variants: Benner, Burke, Byers, Gillis, Guglielmi, He, Hinrichsen, Karow, Kostić, Lewis, Meerbergen, Mehl, Mehrmann, Mengi, Michiels, Międlar, Mitchell, Nesterov, Overton, Pritchard, Protasov, Sharma, Stolwijk, Van Dooren, Watson, . . . (and surely I have missed many).

Keywords: nearest Ω-stable matrix, pseudospectral abscissa, robust stability, distance to (in)stability.

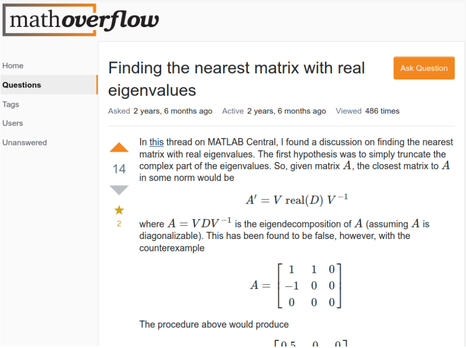
Most focus on the Frobenius norm $\|M\|_F = \left(\sum_{i,j=1}^{n} |M_{ij}|^2\right)^{1/2}$.

The $\forall$ problem is considered more difficult; we need to juggle multiple eigenvalues at the same time.

In this talk: the $\forall$ problem, but the technique can be extended to the $\exists_k$ problem.

An "MO-hard" special case

Nearest matrix with all real eigenvalues: $\Omega = \mathbb{R}$.


mathoverflow

Home

Questions

Tags

Users

Unanswered

Finding the nearest matrix with real eigenvalues

Ask Question

Asked 2 years, 6 months ago Active 2 years, 6 months ago Viewed 486 times

14

2

In this thread on MATLAB Central, I found a discussion on finding the nearest matrix with real eigenvalues. The first hypothesis was to simply truncate the complex part of the eigenvalues. So, given matrix A, the closest matrix to A in some norm would be

$$A' = V\ \mathrm{real}(D)\ V^{-1}$$

where $A = VDV^{-1}$ is the eigendecomposition of A (assuming A is diagonalizable). This has been found to be false, however, with the counterexample

$$A = \begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The procedure above would produce

Attempts to find a closed-form solution (without luck) on Matlab Central and Mathoverflow, dating back to 2010.

The background

Distance from a closed set is a classical topic in mathematical analysis.

Given $\Omega \subseteq \mathbb{R}^N$ (or also $\mathbb{C} \simeq \mathbb{R}^2$), study functions

$$d_\Omega^2(x) = \min_{y\in\Omega} \|x-y\|^2, \qquad p_\Omega(x) = \arg\min_{y\in\Omega} \|x-y\|^2.$$

Known results: d_Ω^2 is continuous, semiconcave, and differentiable everywhere apart from a (measure-zero) set where $p_\Omega(x)$ is not unique.

squared distance from $\Omega = \{0\} \cup [2, \infty)$

The background

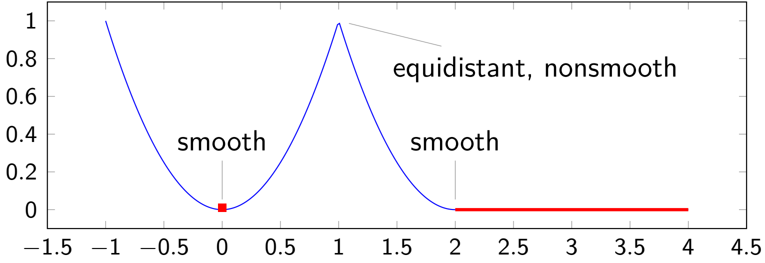
Distance from a closed set is a classical topic in mathematical analysis.

Given $\Omega \subseteq \mathbb{R}^N$ (or also $\mathbb{C} \simeq \mathbb{R}^2$), study functions

$$d^2_\Omega(x) = \min_{y\in\Omega} \|x-y\|^2, \qquad p_\Omega(x) = \arg\min_{y\in\Omega} \|x-y\|^2.$$

Known results: d^2_Ω is continuous, semiconcave, and differentiable everywhere apart from a (measure-zero) set where $p_\Omega(x)$ is not unique.

squared distance from $\Omega = \{0\} \cup [2,\infty)$

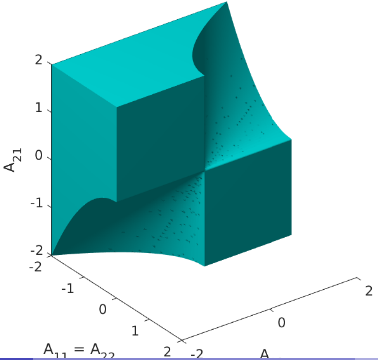


The set of Hurwitz stable matrices

The set $\{X \in \mathbb{C}^{n \times n} \colon \Lambda(X) \subseteq \Omega\}$ is closed, so the same results hold for nearest Ω-stable matrix problems.

Challenge 1: the set of Hurwitz stable matrices is non-smooth and non-convex, already for $n = 2$. Many local minima.

Challenge 2: minimizers often have multiple eigenvalues $\implies$ non-differentiability.



Yet another approach

Our approach: reformulation as optimization on matrix manifolds.

Basic idea simple enough that we can explain it in a few slides.

The problem

$$B = \arg \min_{\Lambda(X) \subseteq \Omega} \|A - X\|_F.$$

Real and a complex version:

1. Nearest $X \in \mathbb{C}^{n \times n}$ to a given $A \in \mathbb{C}^{n \times n}$;
2. Nearest $X \in \mathbb{R}^{n \times n}$ to a given $A \in \mathbb{R}^{n \times n}$.

We start from the complex case.

On triangular matrices

Let us first solve a simpler problem: X upper triangular.

$$\mathcal{T}(A) = \arg \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ upper triangular}}} \|A - T\|_F$$

$$= \arg \min_{T_{ii}\in\Omega} \left\| \begin{bmatrix} A_{11} - T_{11} & A_{12} - T_{12} & A_{13} - T_{13} & \dots \\ A_{21} & A_{22} - T_{22} & A_{23} - T_{23} & \dots \\ A_{31} & A_{32} & A_{33} - T_{33} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \right\|_F$$

Clearly, the best choice is

- $T_{ij} = A_{ij}$ above the diagonal,
- $T_{ii} = p_\Omega(A_{ii})$ on the diagonal.

On triangular matrices

Let us first solve a simpler problem: X upper triangular.

$$\mathcal{T}(A) = \arg \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ upper triangular}}} \|A - T\|_F$$

$$= \arg \min_{T_{ii}\in\Omega} \left\| \begin{bmatrix} A_{11} - T_{11} & A_{12} - T_{12} & A_{13} - T_{13} & \dots \\ A_{21} & A_{22} - T_{22} & A_{23} - T_{23} & \dots \\ A_{31} & A_{32} & A_{33} - T_{33} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \right\|_F$$

Clearly, the best choice is

- $T_{ij} = A_{ij}$ above the diagonal,
- $T_{ii} = p_\Omega(A_{ii})$ on the diagonal.

Triangular case

Lemma

The solution to

$$\mathcal{T}(A) = \arg \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ upper triangular}}} \|A - T\|_F$$

is

$$\mathcal{T}(A)_{ij} = \begin{cases} A_{ij} & i < j, \\ p_\Omega(A_{ij}) & i = j, \\ 0 & i > j. \end{cases}$$

The optimum is $\|\mathcal{L}(A)\|_F$, where $\mathcal{L}(A) = A - \mathcal{T}(A)$ has entries

$$\mathcal{L}(A)_{ij} = \begin{cases} 0 & i < j, \\ A_{ij} - p_\Omega(A_{ij}) & i = j, \\ A_{ij} & i > j. \end{cases}$$

Example

With $\Omega = \{\lambda\colon \operatorname{Re}\lambda \le 0\}$ (nearest Hurwitz stable):

$$\underbrace{\begin{bmatrix} 3 & -1 & 1 & 2 \\ -1 & -2 & 4 & 0 \\ 2 & 1 & -1 & 1 \\ 1 & 2 & 2 & 1 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} 0 & -1 & 1 & 2 \\ 0 & -2 & 4 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathcal{T}(A)} + \underbrace{\begin{bmatrix} 3 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 1 & 2 & 2 & 1 \end{bmatrix}}_{\mathcal{L}(A)}$$

$\mathcal{T}(A)$ is the upper triangular Hurwitz stable matrix nearest to A, with distance $\|\mathcal{L}(A)\|_F$.

Schur trick

In an unknown basis, the solution X is upper triangular!

Schur form $X = UTU^*$: T upper triangular, $U \in \mathcal{U}_n$ (unitary matrices).

$$\min_{\Lambda(X)\subseteq\Omega} \|A - X\|_F = \min_{U\in\mathcal{U}_n} \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ triangular}}} \|A - UTU^*\|_F$$

$$= \min_{U\in\mathcal{U}_n} \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ triangular}}} \|U^*AU - T\|_F$$

$$= \min_{U\in\mathcal{U}_n} \|\mathcal{L}(U^*AU)\|_F.$$

We transformed a problem on $\{\Lambda(X) \subseteq \Omega\}$ into one on $\mathcal{U}_n$: simpler structure, half as many degrees of freedom.

Schur trick

In an unknown basis, the solution X is upper triangular!

Schur form $X = UTU^*$: T upper triangular, $U \in \mathcal{U}_n$ (unitary matrices).

$$\min_{\Lambda(X)\subseteq\Omega} \|A - X\|_F = \min_{U\in\mathcal{U}_n} \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ triangular}}} \|A - UTU^*\|_F$$

$$= \min_{U\in\mathcal{U}_n} \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ triangular}}} \|U^*AU - T\|_F$$

$$= \min_{U\in\mathcal{U}_n} \|\mathcal{L}(U^*AU)\|_F.$$

We transformed a problem on $\{\Lambda(X) \subseteq \Omega\}$ into one on $\mathcal{U}_n$: simpler structure, half as many degrees of freedom.

Schur trick

In an unknown basis, the solution X is upper triangular!

Schur form $X = UTU^*$: T upper triangular, $U \in \mathcal{U}_n$ (unitary matrices).

$$
\begin{aligned}
\min_{\Lambda(X)\subseteq\Omega} \|A - X\|_F &= \min_{U\in\mathcal{U}_n} \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ triangular}}} \|A - UTU^*\|_F \\
&= \min_{U\in\mathcal{U}_n} \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ triangular}}} \|U^*AU - T\|_F \\
&= \min_{U\in\mathcal{U}_n} \|\mathcal{L}(U^*AU)\|_F.
\end{aligned}
$$

We transformed a problem on $\{\Lambda(X) \subseteq \Omega\}$ into one on $\mathcal{U}_n$: simpler structure, half as many degrees of freedom.

Schur trick

In an unknown basis, the solution X is upper triangular!

Schur form $X = UTU^*$: T upper triangular, $U \in \mathcal{U}_n$ (unitary matrices).

$$\begin{aligned}\min_{\Lambda(X)\subseteq\Omega} \|A - X\|_F &= \min_{U\in\mathcal{U}_n} \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ triangular}}} \|A - UTU^*\|_F \\ &= \min_{U\in\mathcal{U}_n} \min_{\substack{\Lambda(T)\subseteq\Omega \\ T \text{ triangular}}} \|U^*AU - T\|_F \\ &= \min_{U\in\mathcal{U}_n} \|\mathcal{L}(U^*AU)\|_F.\end{aligned}$$

We transformed a problem on $\{\Lambda(X) \subseteq \Omega\}$ into one on $\mathcal{U}_n$: simpler structure, half as many degrees of freedom.

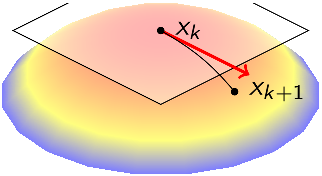
Optimization on (matrix) manifolds

Optimization on matrix manifolds has been studied widely recently: see e.g. [Absil, Mahony, Sepulchre book].

Many first- and second-order methods available.

Key ideas:

- switch to Riemannian gradient and Hessian;
- the Riemannian gradient lives in the tangent space; we need a way to "retract" $x_k + g_k$ onto the manifold.



Optimization on manifolds: the set-up

We just use these algorithms as black box (for now).

- Manifold: $\mathcal{U}_n$ (unitary matrices).
- Function: $f(U) = \|\mathcal{L}(U^* A U)\|_F^2$, with

$$\mathcal{L}(A)_{ij} = \begin{cases} 0 & i < j, \\ A_{ij} - p_\Omega(A_{ij}) & i = j, \\ A_{ij} & i > j. \end{cases}$$

- Gradient: $\nabla_U f = 2U \operatorname{skew}(TL^* - L^* T)$, where $L = \mathcal{L}(U^* A U)$, $T = \mathcal{T}(U^* A U)$, $\operatorname{skew}(M) = \frac{1}{2}(M - M^*)$.
- Algorithm: quasi-Newton (trust-region).

Remark There is nothing that computes eigenvalues here. (!!) The optimization procedure "does that" for us, and returns X in Schur form.

Differentiable formulation: both f and the constraint $U^* U = I$ are C^1 (outside of the medial axis).

Optimization on manifolds: the set-up

We just use these algorithms as black box (for now).

- Manifold: $\mathcal{U}_n$ (unitary matrices).
- Function: $f(U) = \|\mathcal{L}(U^*AU)\|_F^2$, with
$$\mathcal{L}(A)_{ij} = \begin{cases} 0 & i < j, \\ A_{ij} - p_\Omega(A_{ij}) & i = j, \\ A_{ij} & i > j. \end{cases}$$
- Gradient: $\nabla_U f = 2U\,\mathrm{skew}(TL^* - L^*T)$, where $L = \mathcal{L}(U^*AU)$, $T = \mathcal{T}(U^*AU)$, $\mathrm{skew}(M) = \frac{1}{2}(M - M^*)$.
- Algorithm: quasi-Newton (trust-region).

Remark There is nothing that computes eigenvalues here. (!!) The optimization procedure "does that" for us, and returns X in Schur form.

Differentiable formulation: both f and the constraint $U^*U = I$ are C^1 (outside of the medial axis).

An aside: relation to Jacobi eigensolver

If we run the algorithm with $\Omega = \mathbb{C}$, the solution is $A = B = UTU^*$, i.e., the optimization algorithm just computes the Schur form of A.

This reminds of the Jacobi eigenvalue algorithm: apply a series of Givens rotations trying to zero out $\operatorname{tril}(A) \iff$ coordinate descent on $\mathcal{U}_n$.

$$A \mapsto Q_{k\ell}^T A Q_{k\ell} = A'.$$

$$\begin{bmatrix} * & & & \cdots & & & * \\ & \ddots & & & & & \\ & & a_{kk} & \cdots & a_{k\ell} & & \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ & & a_{\ell k} & \cdots & a_{\ell\ell} & & \\ & & & & & \ddots & \\ * & & & \cdots & & & * \end{bmatrix} \to \begin{bmatrix} * & & & \cdots & & & * \\ & \ddots & & & & & \\ & & a'_{kk} & \cdots & 0 & & \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ & & 0 & \cdots & a'_{\ell\ell} & & \\ & & & & & \ddots & \\ * & & & \cdots & & & * \end{bmatrix}.$$

In practice, coordinate descent did not perform well on this problem. However, many advanced computational tricks exist for eigensolvers; maybe we can borrow some.

The real case

The real case is more involved, because the real Schur form is more involved.

Easy case: $\Omega \subseteq \mathbb{R}$.
In this case, each admissible X can be written as $X = QTQ^\top$, where $Q \in \mathcal{O}_n$ (orthogonal matrices) and T is (truly) triangular.
Everything works like in the complex case.

This works for the 'nearest matrix with real eigenvalues' problem, for instance.

Hard case: general Ω.
We need to handle 2×2 blocks in the correct way.

General real case

Each real matrix is similar to

$$\begin{bmatrix} T_{11} & T_{12} & T_{13} & \dots \\ 0 & T_{22} & T_{23} & \dots \\ 0 & 0 & T_{33} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

where all T_{ii} are 2×2, except for a lone final entry if n odd.
(The T_{ii} may have real eigenvalues.)

We define $\mathcal{T}(A), \mathcal{L}(A)$ blockwise:

$$\mathcal{T}(A)_{ij} = \begin{cases} A_{ij} & i < j, \\ p_\Omega(A_{ij}) & i = j, \\ 0 & i > j, \end{cases} \quad \mathcal{L}(A)_{ij} = \begin{cases} 0 & i < j, \\ A_{ij} - p_\Omega(A_{ij}) & i = j, \\ A_{ij} & i > j. \end{cases}$$

(A_{ij} are 2×2 blocks here.)

Real case: 2×2 projection

We need a way to compute $p_\Omega(A_{ij})$, i.e., the 'projection' of $A_{ij} \in \mathbb{R}^{2\times 2}$ onto $\{\Lambda(X) \subseteq \Omega\}$.
I.e., we need a way to solve the 2×2 version of our problem.

This is more involved; we provide an implementation for the Hurwitz stable case.

Projection on Hurwitz stable 2×2 matrices

Let $A \in \mathbb{R}^{2\times 2}$, and $B = p_\Omega(A)$ the nearest Hurwitz stable matrix to A.

First result: we can reduce to matrices with equal diagonal entries.

Lemma

Each A is similar to an $\hat{A} = Q^\top AQ$ with $\hat{A}_{11} = \hat{A}_{22}$.

Lemma

If $A_{11} = A_{22}$, then $B_{11} = B_{22}$.

Projection on Hurwitz stable 2×2 matrices

Second result: casework based on trace and determinant.

Lemma (Hurwitz)

$X \in \mathbb{R}^{2\times 2}$ Hurwitz stable iff $\mathrm{Tr}(X) \leq 0$, $\det(X) \geq 0$.

Lemma

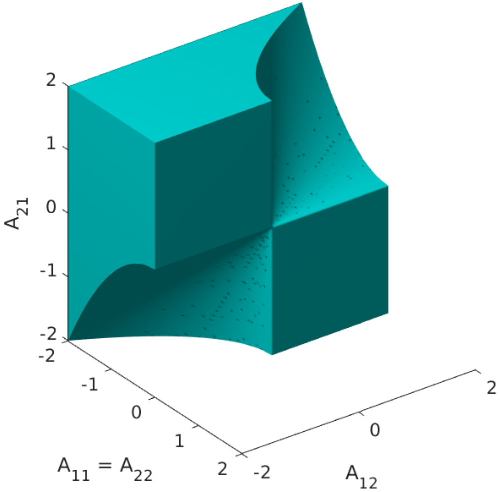
When A is not Hurwitz stable, B is either:

1. a (local) minimizer on $\{\mathrm{Tr}(X) = 0\}$,
2. a (local) minimizer on $\{\det(X) = 0\}$,
3. a (local) minimizer on $\{\mathrm{Tr}(X) = \det(X) = 0\}$.

Minimizers in all three cases can be computed explicitly with a little work (for instance, truncated SVD solves case 2).

The set of 2 × 2 Hurwitz stable matrices

We can now make more sense of this picture.



Optimization on manifolds: the set-up

We can formulate a real analogue of the algorithm.

- Manifold: $\mathcal{O}_n$ (orthogonal matrices).
- Function: $f(Q) = \|\mathcal{L}(Q^\top A Q)\|_F^2$, with
  $$\mathcal{L}(A)_{ij} = \begin{cases} 0 & i < j, \\ A_{ij} - p_\Omega(A_{ij}) & i = j, \\ A_{ij} & i > j. \end{cases}$$
  (the scalar version, if $\Omega \subseteq \mathbb{R}$, or the 2×2 block version).
- Gradient: $\nabla_Q f = 2Q\,\mathrm{skew}(TL^\top - L^\top T)$, where $L = \mathcal{L}(Q^\top A Q)$, $T = \mathcal{T}(Q^\top A Q)$, $\mathrm{skew}(M) = \frac{1}{2}(M - M^\top)$.
- Algorithm: quasi-Newton (trust-region).

A conjecture

Let us consider the complex version of the problem

$$B = \arg \min_{\substack{\Lambda(X)\subseteq\Omega \\ X\in\mathbb{C}^{n\times n}}} \|A - X\|_F.$$

Open problem

When A is a real matrix, is B also always a real matrix?

Experiments suggest so, at least for $\Omega =$ Hurwitz stable.

If the answer is yes, then one can also use the complex version of the algorithm for the real case.

Pros : simpler to write; no need to solve the 2×2 case by hand.

Cons : no reduction in dimensionality of the problem.

A conjecture

Let us consider the complex version of the problem

$$B = \arg \min_{\substack{\Lambda(X) \subseteq \Omega \\ X \in \mathbb{C}^{n \times n}}} \|A - X\|_F.$$

Open problem

When A is a real matrix, is B also always a real matrix?

Experiments suggest so, at least for $\Omega =$ Hurwitz stable.

If the answer is yes, then one can also use the complex version of the algorithm for the real case.

Pros : simpler to write; no need to solve the 2×2 case by hand.

Cons : no reduction in dimensionality of the problem.

Numerical experiments: setup

Tool Manopt [Boumal, Mishra, Absil, Sepulchre], a Matlab toolbox.

Competitors Various algorithms available on N. Gillis' home page:

- [Burke, Henrion, Lewis, Overton]: non-smooth quasi-Newton methods
- [Orbandexivry, Nesterov, Van Dooren]: convex approximation
- [Gillis, Sharma]: reformulation as dissipative Hamiltonian system

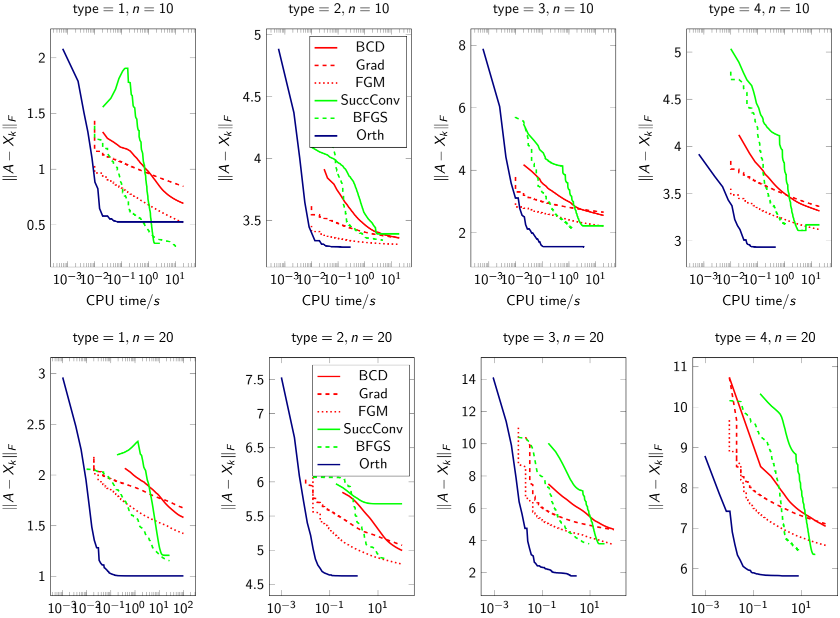
Not in these experiments, but some remarks later:

- [Guglielmi, Lubich, Manetta, Protasov]: reformulation as a system of ODEs

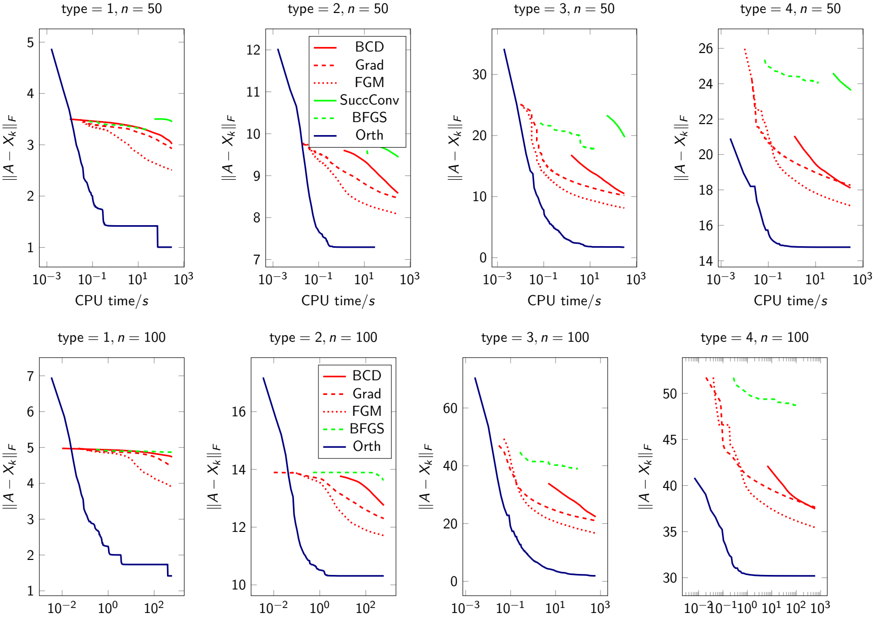
(arguably the best algorithm available so far).

All algorithms promise only local minima.

Numerical experiments: results



Numerical experiments: results



Numerical experiments: quality of local minima found

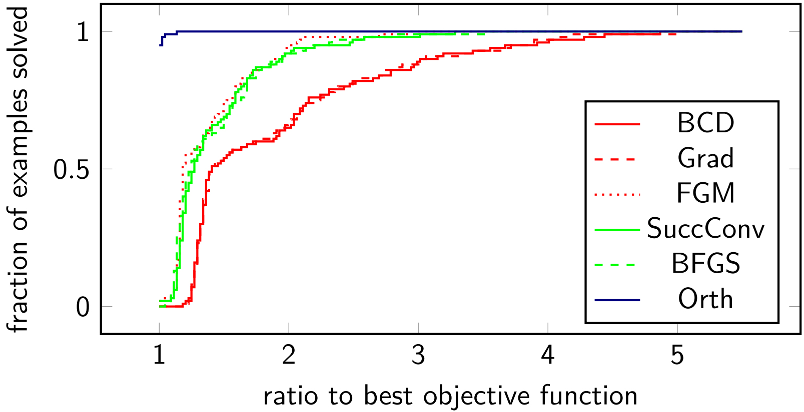
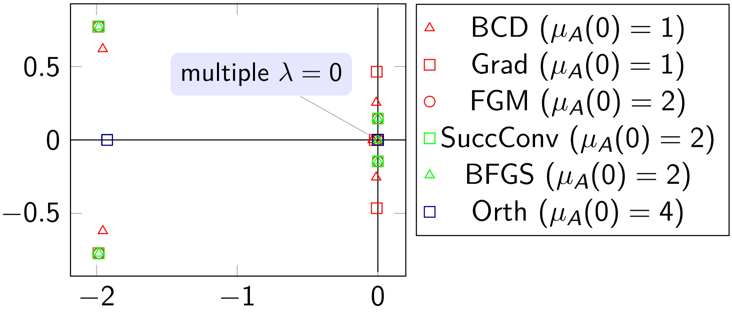


Figure: Performance profile of the values of $\|A - X\|_F$ obtained by the algorithms on 100 random 10×10 matrices (equal split of `rand` and `randn`).

Multiple eigenvalues

Empirical observation: often the other algorithms (especially BCD and Grad) cannot find local minima with multiple zero eigenvalues.

Eigenvalues of minimizer B for a random 6×6 matrix A



Related: in Orth, diag(T) gives multiple eigenvalues much more accurately than eig(B) (accuracy $\mathbf{u}^{1/k}$ from perturbation theory).

Comparison with ODE approach

No extensive comparison yet with ODE approach [Guglielmi, Lubich] (due to code availability).

- ☺ On a difficult small example (30×30 Grcar matrix), we seem to win both in terms of CPU time and quality of minimum $\|A - B\|_F$ (5.65 vs 6.50, by finding a minimizer with a pair of complex conjugate eigenvalues of multiplicity 14!).
- ☹ On large-scale problems (e.g. one with $n = 800$), the optimizer from `Manopt` does not converge.
- ☹ ODE method can handle various matrix structures and we cannot.

Conclusions

- New framework to attack nearest-stable-matrix problems via optimization on matrix manifolds.
- Avoids some of the main troubles with the problem: tricky feasible region, numerical difficulties with eigenvalue computation.
- Great numerical results for small matrices. Still work needed for larger matrices ($n \approx 100 - 1000$).
- Design space to explore: choose good initial value; fine-tune the optimization method; borrow tricks from eigensolvers.
- The approach works for a generic Ω, and can be generalized to variants (e.g., nearest matrix with at least k eigenvalues in Ω).

Thanks for your attention!

Conclusions

- New framework to attack nearest-stable-matrix problems via optimization on matrix manifolds.
- Avoids some of the main troubles with the problem: tricky feasible region, numerical difficulties with eigenvalue computation.
- Great numerical results for small matrices. Still work needed for larger matrices ($n \approx 100 - 1000$).
- Design space to explore: choose good initial value; fine-tune the optimization method; borrow tricks from eigensolvers.
- The approach works for a generic Ω, and can be generalized to variants (e.g., nearest matrix with at least k eigenvalues in Ω).

Thanks for your attention!