Three numerical methods for the palindromic eigenvalue problem

Christian Mehl
Institut für Mathematik
Technische Universität Berlin

joint work with Niloufer Mackey, Steve Mackey, Volker Mehrmann

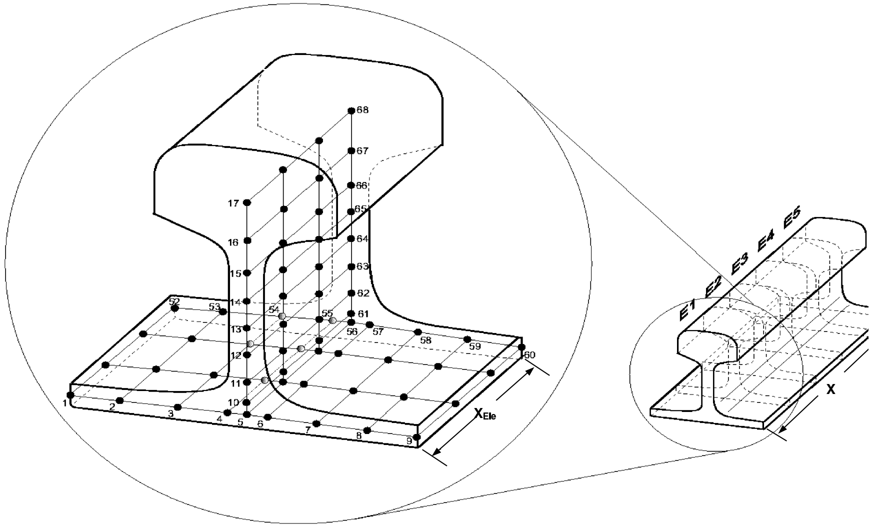
ICIAM 07

Zürich, July 16–20, 2007

Work supported by DFG Research Center
MATHEON – Mathematics for key technologies

Structured polynomial eigenvalue problems

Application: vibration analysis of rail tracks excited by high speed trains



Finite element discretization leads to the **palindromic eigenvalue problem**

$$\left(\lambda^2 A_0^T + \lambda A_1 + A_0\right) x = 0,$$

where $A_0, A_1 \in \mathbb{C}^{n\times n}$, and $A_1^T = A_1$.

Palindromic matrix polynomials

Definition: A matrix polynomial $P(\lambda) = A_0 + \lambda A_1 + \cdots + \lambda^k A_k$ is called T-palindromic (in short: **palindromic**) if

$$P(\lambda) = \sum_{j=0}^{k} \lambda^{k-j} A_j^T.$$

Examples:

- $P(\lambda) = A + \lambda B + \lambda^2 B^T + \lambda^3 A^T$;
- $P(\lambda) = A_2^T + \lambda A_1^T + \lambda^2 A_0 + \lambda^3 A_1 + \lambda^4 A_2$, where A_0 is symmetric;
- palindromic pencils $\lambda Z + Z^T$.

Formal resemblance with linguistic palindroms like "**I prefer pi**".

Properties of palindromic matrix polynomials

General assumption: all matrix polynomials under consideration are regular, i.e., $\det P(\lambda) \not\equiv 0$.

Spectral symmetry: Palindromic matrix polynomials have a symplectic spectrum.

- if λ_0 is an eigenvalue of $P(\lambda)$, then so is λ_0^{-1};
- pairing occurs also in algebraic, geometric, and partial multiplicities;
- symmetry degenerates for $\lambda_0 = 1$ and $\lambda_0 = -1$;

"Palindromic matrix polynomials generalize symplectic matrices".

How to solve palindromic eigenvalue problems

Linearization: Mackey, Mackey, M., Mehrmann: linearization theory for general and structured matrix polynomials

- Under modest assumptions, any polynomial palindromic eigenvalue problem can be transformed to a linear palindromic eigenvalue problem.

Example: $P(\lambda) = \lambda^2 A_0^T + \lambda A_1 + A_0$. Then

$$\lambda Z + Z^T := \lambda \begin{bmatrix} A_0^T & A_1 - A_0 \\ A_0^T & A_0^T \end{bmatrix} + \begin{bmatrix} A_0 & A_0 \\ A_1 - A_0^T & A_0 \end{bmatrix}$$

is a linearization for $P(\lambda)$ if -1 is not an eigenvalue of $P(\lambda)$.

Benefit: Symplectic spectrum preserved in finite precision arithmetic.

How to solve linear palindromic eigenvalue problems

Task: Solve the generalized eigenvalue problem for $\lambda Z + Z^T$.

- T-congruence transformations preserve the structure:

$$(\lambda Z + Z^T) \mapsto P^T(\lambda Z + Z^T)P, \quad P \text{ invertible}$$

- Numerical stability: Choose $P = U$ unitary if possible.

- Look for condensed forms under simultaneous unitary consimilarity:

$$(\lambda Z + Z^T) \mapsto \overline{U}^{-1}(\lambda Z + Z^T)U, \quad U \text{ unitary}$$

Advantage: We have to store and work on Z only.

The anti-triangular Schur form

Theorem: Let $Z \in \mathbb{C}^{n\times n}$. Then there exists a unitary matrix $U \in \mathbb{C}^{n\times n}$ such that

$$U^T Z U = \begin{bmatrix} 0 & \dots & 0 & z_{1n} \\ \vdots & \iddots & z_{2,n-1} & \vdots \\ 0 & \iddots & \iddots & \vdots \\ z_{n1} & \dots & \dots & z_{nn} \end{bmatrix}$$

is in **anti-triangular Schur form**.

Consequence: If $\det(\lambda Z + Z^T) \not\equiv 0$ then the eigenvalues of $\lambda Z + Z^T$ are

$$-\frac{z_{n1}}{z_{1n}}, \dots, -\frac{z_{1n}}{z_{n1}}, \qquad (\text{where } \frac{z}{0} := \infty).$$

Question: How do we compute the anti-triangular form numerically?

Method 1: The Laub-trick method

Theorem: (generalizes a trick by A. Laub for the computation of the Hamiltonian Schur form) Let $\lambda Z + Z^T \in \mathbb{C}^{2n\times 2n}$ be regular and let

$$\begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \left(\lambda \begin{bmatrix} X_{11} & X_{12} \\ 0 & X_{22} \end{bmatrix} + \begin{bmatrix} Y_{11} & Y_{12} \\ 0 & Y_{22} \end{bmatrix} \right) \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix}.$$

be its generalized Schur decomposition, where $X_{11}, Y_{11} \in \mathbb{C}^{n\times n}$. If

$$\mu \in \sigma(\lambda X_{11} + Y_{11}) \implies \frac{1}{\mu} \notin \sigma(\lambda X_{11} + Y_{11})$$

then

$$U = \begin{bmatrix} W_{11} & Q_{11}^T R_n \\ W_{21} & Q_{12}^T R_n \end{bmatrix}, \qquad \left(R_n := \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix} \right)$$

is unitary and

$$U^T Z U = \begin{bmatrix} 0 & Y_{11}^T R_n \\ R_n X_{11} & * \end{bmatrix}.$$

is in anti-triangular form.

Method 1: The Laub-trick method

Algorithm: (for regular $\lambda Z + Z^T$ not having eigenvalues with modulus 1)

1. Compute the generalized Schur decomposition

$$\lambda Z + Z^T = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \left(\lambda \begin{bmatrix} X_{11} & X_{12} \\ 0 & X_{22} \end{bmatrix} + \begin{bmatrix} Y_{11} & Y_{12} \\ 0 & Y_{22} \end{bmatrix} \right) \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix}.$$

2. Reorder the eigenvalues such that $\lambda X_{11} + Y_{11}$ contains all eigenvalues with $|\lambda| > 1$.

3. Set $U = \begin{bmatrix} W_{11} & Q_{11}^T R_n \\ W_{21} & Q_{12}^T R_n \end{bmatrix}$.

4. Compute $Z_{22} = \begin{bmatrix} R_n Q_{11} & R_n Q_{12} \end{bmatrix} Z \begin{bmatrix} Q_{11}^T R_n \\ Q_{12}^T R_n \end{bmatrix}$.

5. Set $\widetilde{Z} := \begin{bmatrix} 0 & Y_{11}^T R_n \\ R_n X_{11} & Z_{22} \end{bmatrix}$.

Method 1: The Laub-trick method

Properties:

+ cost is essentially the cost of `QZ` with reordering;

− only applicable if Z has even dimension and if $\lambda Z + Z^T$ does not have eigenvalues with modulus 1;

− problems if there are eigenvalues with modulus close to ± 1; $\rightsquigarrow$ `QZ` might detect more or less than n eigenvalues λ with $|\lambda| > 1$.

Questions: Are there other methods?

Method 2: A Jacobi-like method

Idea: Annihilate one diagonal or two off diagonal pivot elements in the strict upper anti-triangular part of Z in each Jacobi-step:

$$
\begin{bmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\
\cdot & \bullet & \cdot & \cdot & \cdot & \cdot & * & * \\
\cdot & \cdot & \cdot & \cdot & \cdot & * & * & * \\
\cdot & \cdot & \cdot & \cdot & * & * & * & * \\
\cdot & \cdot & \cdot & * & * & * & * & * \\
\cdot & \cdot & * & * & * & * & * & * \\
\cdot & * & * & * & * & * & * & * \\
* & * & * & * & * & * & * & *
\end{bmatrix}
\qquad
\begin{bmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\
\cdot & \cdot & \bullet & \cdot & \cdot & \cdot & * & * \\
\cdot & \bullet & \cdot & \cdot & \cdot & * & * & * \\
\cdot & \cdot & \cdot & \cdot & * & * & * & * \\
\cdot & \cdot & \cdot & * & * & * & * & * \\
\cdot & \cdot & * & * & * & * & * & * \\
\cdot & * & * & * & * & * & * & * \\
* & * & * & * & * & * & * & *
\end{bmatrix}
$$

This can always be achieved via a unitary consimilarity transformation.

Method 2: A Jacobi-like method

Diagonal pivots:

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & {\color{red}\bullet} & \cdot & \cdot & \cdot & \cdot & {\color{blue}\bullet} & * \\ \cdot & \cdot & \cdot & \cdot & \cdot & * & * & * \\ \cdot & \cdot & \cdot & \cdot & * & * & * & * \\ \cdot & \cdot & \cdot & * & * & * & * & * \\ \cdot & \cdot & * & * & * & * & * & * \\ \cdot & {\color{blue}\bullet} & * & * & * & * & {\color{blue}\bullet} & * \\ * & * & * & * & * & * & * & * \end{bmatrix}$$

Consider the colored 2×2 subproblem:

$$\begin{bmatrix} {\color{red}\bullet} & {\color{blue}\bullet} \\ {\color{blue}\bullet} & {\color{blue}\bullet} \end{bmatrix}$$

Method 2: A Jacobi-like method

Diagonal pivots:

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & \bullet & \cdot & \cdot & \cdot & \cdot & \bullet & * \\ \cdot & \cdot & \cdot & \cdot & \cdot & * & * & * \\ \cdot & \cdot & \cdot & \cdot & * & * & * & * \\ \cdot & \cdot & \cdot & * & * & * & * & * \\ \cdot & \cdot & * & * & * & * & * & * \\ \cdot & \bullet & * & * & * & * & \bullet & * \\ * & * & * & * & * & * & * & * \end{bmatrix}$$

Compute the anti-triangular form of the 2×2 problem:

$$\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix} \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} = \begin{bmatrix} \circ & * \\ * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Diagonal pivots:

$$
\begin{bmatrix}
1 & & & & & & & \\
 & u_{11} & & & & & u_{21} & \\
 & & 1 & & & & & \\
 & & & 1 & & & & \\
 & & & & 1 & & & \\
 & & & & & 1 & & \\
 & u_{12} & & & & & u_{22} & \\
 & & & & & & & 1
\end{bmatrix}
\begin{bmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\
\cdot & {\color{red}\bullet} & \cdot & \cdot & \cdot & \cdot & {\color{blue}\bullet} & * \\
\cdot & \cdot & \cdot & \cdot & \cdot & * & * & * \\
\cdot & \cdot & \cdot & \cdot & * & * & * & * \\
\cdot & \cdot & \cdot & * & * & * & * & * \\
\cdot & \cdot & * & * & * & * & * & * \\
\cdot & {\color{blue}\bullet} & * & * & * & * & {\color{blue}\bullet} & * \\
* & * & * & * & * & * & * & *
\end{bmatrix}
\begin{bmatrix}
1 & & & & & & & \\
 & u_{11} & & & & & u_{12} & \\
 & & 1 & & & & & \\
 & & & 1 & & & & \\
 & & & & 1 & & & \\
 & & & & & 1 & & \\
 & u_{21} & & & & & u_{22} & \\
 & & & & & & & 1
\end{bmatrix}
$$

Then update the $n \times n$ matrix.

$$
\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix}
\begin{bmatrix} {\color{red}\bullet} & {\color{blue}\bullet} \\ {\color{blue}\bullet} & {\color{blue}\bullet} \end{bmatrix}
\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}
=
\begin{bmatrix} {\color{red}\circ} & * \\ * & * \end{bmatrix}
$$

Method 2: A Jacobi-like method

Diagonal pivots:

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & \circ & \cdot & \cdot & \cdot & \cdot & * & * \\ \cdot & \cdot & \cdot & \cdot & \cdot & * & * & * \\ \cdot & \cdot & \cdot & \cdot & * & * & * & * \\ \cdot & \cdot & \cdot & * & * & * & * & * \\ \cdot & \cdot & * & * & * & * & * & * \\ \cdot & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \end{bmatrix}$$

Then update the $n \times n$ matrix.

$$\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix} \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} = \begin{bmatrix} \circ & * \\ * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Off-diagonal pivots:

$$\begin{bmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\
\cdot & \cdot & \bullet & \cdot & \cdot & \cdot & * & * \\
\cdot & \bullet & \cdot & \cdot & \cdot & * & * & * \\
\cdot & \cdot & \cdot & \cdot & * & * & * & * \\
\cdot & \cdot & \cdot & * & * & * & * & * \\
\cdot & \cdot & * & * & * & * & * & * \\
\cdot & * & * & * & * & * & * & * \\
* & * & * & * & * & * & * & *
\end{bmatrix}$$

Question: Why consider two pivots?

Method 2: A Jacobi-like method

Off-diagonal pivots: assume, we only consider one off-diagonal pivot;

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & \cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & {\color{blue}\bullet} & * \\ \cdot & \cdot & \cdot & \cdot & \cdot & * & * & * \\ \cdot & \cdot & \cdot & \cdot & * & * & * & * \\ \cdot & \cdot & \cdot & * & * & * & * & * \\ \cdot & \cdot & {\color{blue}\bullet} & * & * & * & {\color{blue}\bullet} & * \\ \cdot & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \end{bmatrix}$$

Consider the colored 2×2 problem:

$$\begin{bmatrix} {\color{green}\bullet} & {\color{blue}\bullet} \\ {\color{blue}\bullet} & {\color{blue}\bullet} \end{bmatrix}$$

Method 2: A Jacobi-like method

Off-diagonal pivots: assume, we only consider one off-diagonal pivot;

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & \cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & {\color{blue}\bullet} & * \\ \cdot & \cdot & \cdot & \cdot & \cdot & * & * & * \\ \cdot & \cdot & \cdot & \cdot & * & * & * & * \\ \cdot & \cdot & \cdot & * & * & * & * & * \\ \cdot & \cdot & {\color{blue}\bullet} & * & * & * & {\color{blue}\bullet} & * \\ \cdot & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \end{bmatrix}$$

Compute the anti-triangular form of the 2×2 problem:

$$\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix} \begin{bmatrix} {\color{green}\bullet} & {\color{blue}\bullet} \\ {\color{blue}\bullet} & {\color{blue}\bullet} \end{bmatrix} \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix} = \begin{bmatrix} {\color{green}\circ} & * \\ * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Off-diagonal pivots: assume, we only consider one off-diagonal pivot;

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & \cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & {\color{blue}\bullet} & * \\ \cdot & \cdot & \cdot & \cdot & \cdot & * & * & * \\ \cdot & \cdot & \cdot & \cdot & * & * & * & * \\ \cdot & \cdot & \cdot & * & * & * & * & * \\ \cdot & \cdot & {\color{blue}\bullet} & * & * & * & {\color{blue}\bullet} & * \\ \cdot & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \end{bmatrix}$$

We may use different unitary transformation from the left and the right,

$$\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix} \begin{bmatrix} {\color{green}\bullet} & {\color{blue}\bullet} \\ {\color{blue}\bullet} & {\color{blue}\bullet} \end{bmatrix} \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix} = \begin{bmatrix} {\color{green}\circ} & * \\ * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Off-diagonal pivots: assume, we only consider one off-diagonal pivot;

$$
\begin{bmatrix}
1 &&&&&&& \\
& u_{11} &&&& u_{21} && \\
&& v_{11} &&&& v_{21} & \\
&&& 1 &&&& \\
&&&& 1 &&& \\
& u_{12} &&&& u_{22} && \\
&& v_{12} &&&& v_{22} & \\
&&&&&&& 1
\end{bmatrix}
\begin{bmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \ast \\
\cdot & \cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & {\color{blue}\bullet} & \ast \\
\cdot & \cdot & \cdot & \cdot & \cdot & \ast & \ast & \ast \\
\cdot & \cdot & \cdot & \cdot & \ast & \ast & \ast & \ast \\
\cdot & \cdot & \cdot & \ast & \ast & \ast & \ast & \ast \\
\cdot & \cdot & {\color{blue}\bullet} & \ast & \ast & \ast & {\color{blue}\bullet} & \ast \\
\cdot & \ast & \ast & \ast & \ast & \ast & \ast & \ast \\
\ast & \ast & \ast & \ast & \ast & \ast & \ast & \ast
\end{bmatrix}
\begin{bmatrix}
1 &&&&&&& \\
& u_{11} &&&& u_{12} && \\
&& v_{11} &&&& v_{12} & \\
&&& 1 &&&& \\
&&&& 1 &&& \\
& u_{21} &&&& u_{22} && \\
&& v_{21} &&&& v_{22} & \\
&&&&&&& 1
\end{bmatrix}
$$

because of the position of the subproblem. $\rightsquigarrow$ more freedom in the parameters

$$
\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix}
\begin{bmatrix} {\color{green}\bullet} & {\color{blue}\bullet} \\ {\color{blue}\bullet} & {\color{blue}\bullet} \end{bmatrix}
\begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}
=
\begin{bmatrix} {\color{green}\circ} & \ast \\ \ast & \ast \end{bmatrix}
$$

Method 2: A Jacobi-like method

Off-diagonal pivots: assume, we only consider one off-diagonal pivot;

$$\begin{bmatrix} 1 &&&&&&& \\ & u_{11} &&&& u_{21} && \\ && v_{11} &&&& v_{21} & \\ &&& 1 &&&& \\ &&&& 1 &&& \\ & u_{12} &&&& u_{22} && \\ && v_{12} &&&& v_{22} & \\ &&&&&&& 1 \end{bmatrix} \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & \cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & {\color{blue}\bullet} & * \\ \cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & \bullet & * & * \\ \cdot & \cdot & \cdot & \cdot & * & * & * & * \\ \cdot & \cdot & \cdot & * & * & * & * & * \\ \cdot & \cdot & {\color{blue}\bullet} & * & * & * & {\color{blue}\bullet} & * \\ \cdot & \bullet & * & * & * & \bullet & * & * \\ * & * & * & * & * & * & * & * \end{bmatrix} \begin{bmatrix} 1 &&&&&&& \\ & u_{11} &&&& u_{12} && \\ && v_{11} &&&& v_{12} & \\ &&& 1 &&&& \\ &&&& 1 &&& \\ & u_{21} &&&& u_{22} && \\ && v_{21} &&&& v_{22} & \\ &&&&&&& 1 \end{bmatrix}$$

Simultaneously, a second 2×2 system – marked by $\bullet$ – will be transformed.

$$\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix} \begin{bmatrix} {\color{green}\bullet} & {\color{blue}\bullet} \\ {\color{blue}\bullet} & {\color{blue}\bullet} \end{bmatrix} \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix} = \begin{bmatrix} {\color{green}\circ} & * \\ * & * \end{bmatrix}$$

$$\begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \begin{bmatrix} {\color{green}\bullet} & \bullet \\ \bullet & \bullet \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} = \begin{bmatrix} {\color{green}*} & * \\ * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Off-diagonal pivots: assume, we only consider one off-diagonal pivot;

$$
\begin{bmatrix}
1 &&&&&&& \\
& u_{11} &&&& u_{21} && \\
&& v_{11} &&&& v_{21} & \\
&&& 1 &&&& \\
&&&& 1 &&& \\
& u_{12} &&&& u_{22} && \\
&& v_{12} &&&& v_{22} & \\
&&&&&&& 1
\end{bmatrix}
\begin{bmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\
\cdot & \cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & {\color{blue}\bullet} & * \\
\cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & \bullet & * & * \\
\cdot & \cdot & \cdot & \cdot & * & * & * & * \\
\cdot & \cdot & \cdot & * & * & * & * & * \\
\cdot & \cdot & {\color{blue}\bullet} & * & * & * & {\color{blue}\bullet} & * \\
\cdot & \bullet & * & * & * & \bullet & * & * \\
* & * & * & * & * & * & * & *
\end{bmatrix}
\begin{bmatrix}
1 &&&&&&& \\
& u_{11} &&&& u_{12} && \\
&& v_{11} &&&& v_{12} & \\
&&& 1 &&&& \\
&&&& 1 &&& \\
& u_{21} &&&& u_{22} && \\
&& v_{21} &&&& v_{22} & \\
&&&&&&& 1
\end{bmatrix}
$$

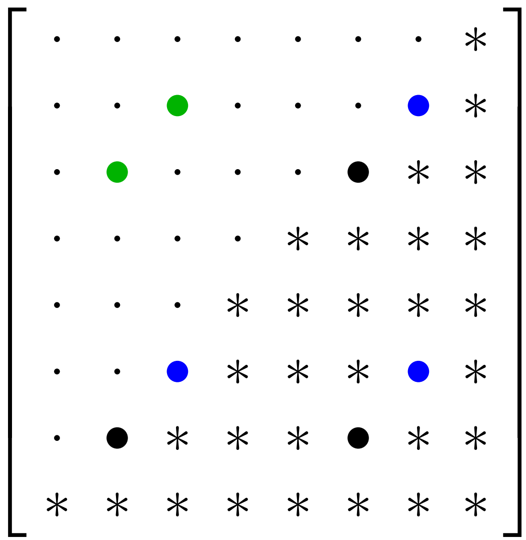
Use the freedom in the parameters to anti-triangularize the second system as well.

$$
\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix}
\begin{bmatrix} {\color{green}\bullet} & {\color{blue}\bullet} \\ {\color{blue}\bullet} & {\color{blue}\bullet} \end{bmatrix}
\begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}
=
\begin{bmatrix} {\color{green}\circ} & * \\ * & * \end{bmatrix}
$$

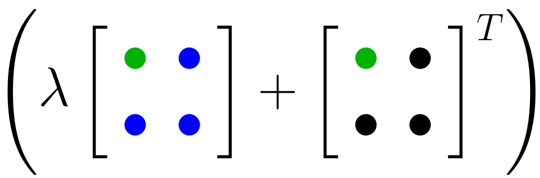
$$
\begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix}
\begin{bmatrix} {\color{green}\bullet} & \bullet \\ \bullet & \bullet \end{bmatrix}
\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}
=
\begin{bmatrix} {\color{green}*} & * \\ * & * \end{bmatrix}
$$

Method 2: A Jacobi-like method

Off-diagonal pivots:



Anti-triangularize the colored/black generalized 2×2 problem:



Method 2: A Jacobi-like method

Off-diagonal pivots:

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & \cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & {\color{blue}\bullet} & * \\ \cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & \bullet & * & * \\ \cdot & \cdot & \cdot & \cdot & * & * & * & * \\ \cdot & \cdot & \cdot & * & * & * & * & * \\ \cdot & \cdot & {\color{blue}\bullet} & * & * & * & {\color{blue}\bullet} & * \\ \cdot & \bullet & * & * & * & \bullet & * & * \\ * & * & * & * & * & * & * & * \end{bmatrix}$$

Anti-triangularize the colored/black generalized 2×2 problem:

$$\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix} \left(\lambda \begin{bmatrix} {\color{green}\bullet} & {\color{blue}\bullet} \\ {\color{blue}\bullet} & {\color{blue}\bullet} \end{bmatrix} + \begin{bmatrix} {\color{green}\bullet} & \bullet \\ \bullet & \bullet \end{bmatrix}^T \right) \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix} = \lambda \begin{bmatrix} {\color{green}\circ} & * \\ * & * \end{bmatrix} + \begin{bmatrix} {\color{green}\circ} & * \\ * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Off-diagonal pivots:

$$
\begin{bmatrix}
1 &&&&&&& \\
& u_{11} &&&& u_{21} && \\
&& v_{11} &&&& v_{21} & \\
&&& 1 &&&& \\
&&&& 1 &&& \\
& u_{12} &&&& u_{22} && \\
&& v_{12} &&&& v_{22} & \\
&&&&&&& 1
\end{bmatrix}
\begin{bmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\
\cdot & \cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & {\color{blue}\bullet} & * \\
\cdot & {\color{green}\bullet} & \cdot & \cdot & \cdot & \bullet & * & * \\
\cdot & \cdot & \cdot & \cdot & * & * & * & * \\
\cdot & \cdot & \cdot & * & * & * & * & * \\
\cdot & \cdot & {\color{blue}\bullet} & * & * & * & {\color{blue}\bullet} & * \\
\cdot & \bullet & * & * & * & \bullet & * & * \\
* & * & * & * & * & * & * & *
\end{bmatrix}
\begin{bmatrix}
1 &&&&&&& \\
& u_{11} &&&& u_{12} && \\
&& v_{11} &&&& v_{12} & \\
&&& 1 &&&& \\
&&&& 1 &&& \\
& u_{21} &&&& u_{22} && \\
&& v_{21} &&&& v_{22} & \\
&&&&&&& 1
\end{bmatrix}
$$

Update the $n \times n$ matrix.

$$
\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix}
\left(\lambda \begin{bmatrix} {\color{green}\bullet} & {\color{blue}\bullet} \\ {\color{blue}\bullet} & {\color{blue}\bullet} \end{bmatrix} + \begin{bmatrix} {\color{green}\bullet} & \bullet \\ \bullet & \bullet \end{bmatrix}^T \right)
\begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}
= \lambda \begin{bmatrix} {\color{green}\circ} & * \\ * & * \end{bmatrix} + \begin{bmatrix} {\color{green}\circ} & * \\ * & * \end{bmatrix}
$$

Method 2: A Jacobi-like method

Off-diagonal pivots:

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & \cdot & \circ & \cdot & \cdot & \cdot & * & * \\ \cdot & \circ & \cdot & \cdot & \cdot & * & * & * \\ \cdot & \cdot & \cdot & \cdot & * & * & * & * \\ \cdot & \cdot & \cdot & * & * & * & * & * \\ \cdot & \cdot & * & * & * & * & * & * \\ \cdot & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * \end{bmatrix}$$

Update the $n \times n$ matrix.

$$\begin{bmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{bmatrix} \left(\lambda \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \end{bmatrix} + \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \end{bmatrix}^T \right) \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix} = \lambda \begin{bmatrix} \circ & * \\ * & * \end{bmatrix} + \begin{bmatrix} \circ & * \\ * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Sweep: Annihilate each pivot element at least once.

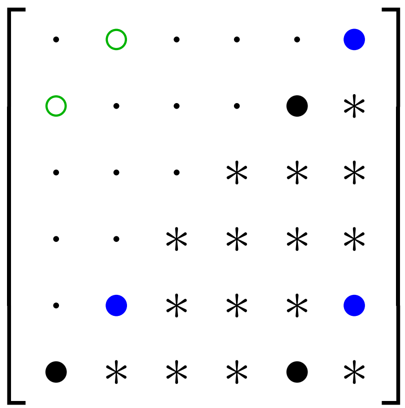
E.g., cyclic-by-row-sweep:

$$\begin{bmatrix} \circ & \cdot & \cdot & \cdot & \cdot & \bullet \\ \cdot & \cdot & \cdot & \cdot & * & * \\ \cdot & \cdot & \cdot & * & * & * \\ \cdot & \cdot & * & * & * & * \\ \cdot & * & * & * & * & * \\ \bullet & * & * & * & * & \bullet \end{bmatrix}$$

Method 2: A Jacobi-like method

Sweep: Annihilate each pivot element at least once.

E.g., cyclic-by-row-sweep:



Method 2: A Jacobi-like method

Sweep: Annihilate each pivot element at least once.

E.g., cyclic-by-row-sweep:

$$\begin{bmatrix} \cdot & \cdot & \circ & \cdot & \cdot & \bullet \\ \cdot & \cdot & \cdot & \cdot & * & * \\ \circ & \cdot & \cdot & \bullet & * & * \\ \cdot & \cdot & \bullet & * & * & \bullet \\ \cdot & * & * & * & * & * \\ \bullet & * & * & \bullet & * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Sweep: Annihilate each pivot element at least once.

E.g., cyclic-by-row-sweep:

$$\begin{bmatrix} \cdot & \cdot & \cdot & {\color{green}\circ} & \cdot & {\color{blue}\bullet} \\ \cdot & \cdot & \cdot & \cdot & * & * \\ \cdot & \cdot & \cdot & {\color{blue}\bullet} & * & {\color{blue}\bullet} \\ {\color{green}\circ} & \cdot & \bullet & * & * & * \\ \cdot & * & * & * & * & * \\ \bullet & * & \bullet & * & * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Sweep: Annihilate each pivot element at least once.

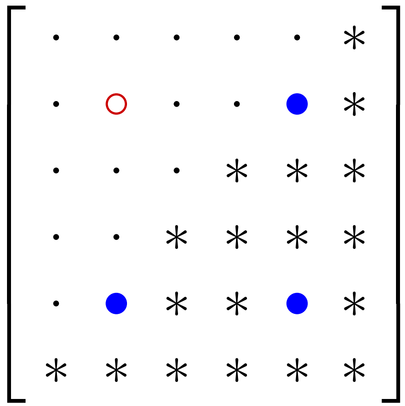
E.g., cyclic-by-row-sweep:

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \circ & \bullet \\ \cdot & \cdot & \cdot & \cdot & \bullet & \bullet \\ \cdot & \cdot & \cdot & * & * & * \\ \cdot & \cdot & * & * & * & * \\ \circ & \bullet & * & * & * & * \\ \bullet & \bullet & * & * & * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Sweep: Annihilate each pivot element at least once.

E.g., cyclic-by-row-sweep:



Method 2: A Jacobi-like method

Sweep: Annihilate each pivot element at least once.

E.g., cyclic-by-row-sweep:

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \ast \\ \cdot & \cdot & {\color{green}\circ} & \cdot & {\color{blue}\bullet} & \ast \\ \cdot & {\color{green}\circ} & \cdot & \bullet & \ast & \ast \\ \cdot & \cdot & {\color{blue}\bullet} & \ast & {\color{blue}\bullet} & \ast \\ \cdot & \bullet & \ast & \bullet & \ast & \ast \\ \ast & \ast & \ast & \ast & \ast & \ast \end{bmatrix}$$

Method 2: A Jacobi-like method

Sweep: Annihilate each pivot element at least once.

E.g., cyclic-by-row-sweep:

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & \cdot & \cdot & {\color{green}\circ} & {\color{blue}\bullet} & * \\ \cdot & \cdot & \cdot & {\color{blue}\bullet} & {\color{blue}\bullet} & * \\ \cdot & {\color{green}\circ} & \bullet & * & * & * \\ \cdot & \bullet & \bullet & * & * & * \\ * & * & * & * & * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Sweep: Annihilate each pivot element at least once.

E.g., cyclic-by-row-sweep:

$$\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & * \\ \cdot & \cdot & \cdot & \cdot & * & * \\ \cdot & \cdot & \circ & \bullet & * & * \\ \cdot & \cdot & \bullet & \bullet & * & * \\ \cdot & * & * & * & * & * \\ * & * & * & * & * & * \end{bmatrix}$$

Method 2: A Jacobi-like method

Properties of the algorithm:

+ locally and asymptotically quadratically convergent;

+ globally convergent in experiments;

+ converges fast for matrices Z close to anti-triangular form

– expensive in general (cost of 3 sweeps $\hat{=}$ cost of `QZ`)

– convergence problems for badly scaled problems

– convergence problems for large n

Method 3: A hybrid method

Laub-trick:

- $+$ works for moderate sizes of n;
- $+$ essentially cost of `QZ`;
- $-$ problems for eigenvalues with modulus near one;

Jacobi:

- $+$ works nicely if problem is small and eigenvalues do not differ too much in modulus;

Idea: Combine the positive properties of these two algorithms. Use the Laub-trick for getting all eigenvalues sufficiently far away from the unit circle and use Jacobi for the eigenvalues near the unit circle.

Method 3: A hybrid method

$$\lambda Z + Z^T = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} \\ Q_{21} & Q_{22} & Q_{23} \\ Q_{31} & Q_{32} & Q_{33} \end{bmatrix} \left(\lambda \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ 0 & X_{22} & X_{23} \\ 0 & 0 & X_{33} \end{bmatrix} + \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ 0 & Y_{22} & Y_{23} \\ 0 & 0 & Y_{33} \end{bmatrix} \right) \begin{bmatrix} W_{11} & W_{12} & W_{13} \\ W_{21} & W_{22} & W_{23} \\ W_{31} & W_{32} & W_{33} \end{bmatrix}$$

Step 1: Given a tolerance $\alpha > 1$ and a regular $\lambda Z + Z^T \in \mathbb{C}^{2n \times 2n}$, compute its generalized Schur decomposition, where the eigenvalues are ordered in such a way that

$$\begin{aligned} \sigma(\lambda X_{11} + Y_{11}) &\subseteq \{\lambda \in \mathbb{C} : |\lambda| \geq \alpha\}, \\ \sigma(\lambda X_{22} + Y_{22}) &\subseteq \{\lambda \in \mathbb{C} : \alpha > |\lambda| > \frac{1}{\alpha}\}, \\ \sigma(\lambda X_{33} + Y_{33}) &\subseteq \{\lambda \in \mathbb{C} : |\lambda| \leq \frac{1}{\alpha}\}. \end{aligned}$$

Method 3: A hybrid method

$$\lambda Z + Z^T = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} \\ Q_{21} & Q_{22} & Q_{23} \\ Q_{31} & Q_{32} & Q_{33} \end{bmatrix} \left(\lambda \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ 0 & X_{22} & X_{23} \\ 0 & 0 & X_{33} \end{bmatrix} + \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ 0 & Y_{22} & Y_{23} \\ 0 & 0 & Y_{33} \end{bmatrix} \right) \begin{bmatrix} W_{11} & W_{12} & W_{13} \\ W_{21} & W_{22} & W_{23} \\ W_{31} & W_{32} & W_{33} \end{bmatrix}$$

Step 2: By the Laub trick, the matrix

$$\begin{bmatrix} W_{11} & Q_{11}^T R_m \\ W_{21} & Q_{12}^T R_m \\ W_{31} & Q_{13}^T R_m \end{bmatrix}$$

has orthonormal columns. Extend this matrix to a unitary matrix

$$U := \begin{bmatrix} W_{11} & U_{12} & Q_{11}^T R_m \\ W_{21} & U_{22} & Q_{12}^T R_m \\ W_{31} & U_{32} & Q_{13}^T R_m \end{bmatrix}.$$

Method 3: A hybrid method

$$\lambda Z + Z^T = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} \\ Q_{21} & Q_{22} & Q_{23} \\ Q_{31} & Q_{32} & Q_{33} \end{bmatrix} \left(\lambda \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ 0 & X_{22} & X_{23} \\ 0 & 0 & X_{33} \end{bmatrix} + \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ 0 & Y_{22} & Y_{23} \\ 0 & 0 & Y_{33} \end{bmatrix} \right) \begin{bmatrix} W_{11} & W_{12} & W_{13} \\ W_{21} & W_{22} & W_{23} \\ W_{31} & W_{32} & W_{33} \end{bmatrix}$$

Step 3: Compute

$$U^T Z U = \begin{bmatrix} 0 & 0 & Y_{11}^T R_m \\ 0 & Z_{22} & Z_{23} \\ R_m X_{11} & Z_{32} & Z_{33} \end{bmatrix},$$

where $Y_{11}^T R_m \in \mathbb{C}^{m \times m}$ and $R_m X_{11} \in \mathbb{C}^{m \times m}$ are in anti-triangular form and $Z_{22} \in \mathbb{C}^{(n-2m)\times(n-2m)}$ has only eigenvalues in $\{\lambda \in \mathbb{C} : \alpha > |\lambda| > \frac{1}{\alpha}\}$.

Method 3: A hybrid method

$$\lambda Z + Z^T = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} \\ Q_{21} & Q_{22} & Q_{23} \\ Q_{31} & Q_{32} & Q_{33} \end{bmatrix} \left(\lambda \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ 0 & X_{22} & X_{23} \\ 0 & 0 & X_{33} \end{bmatrix} + \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ 0 & Y_{22} & Y_{23} \\ 0 & 0 & Y_{33} \end{bmatrix} \right) \begin{bmatrix} W_{11} & W_{12} & W_{13} \\ W_{21} & W_{22} & W_{23} \\ W_{31} & W_{32} & W_{33} \end{bmatrix}$$

Step 3: Compute

$$U^T Z U = \begin{bmatrix} 0 & 0 & Y_{11}^T R_m \\ 0 & Z_{22} & Z_{23} \\ R_m X_{11} & Z_{32} & Z_{33} \end{bmatrix},$$

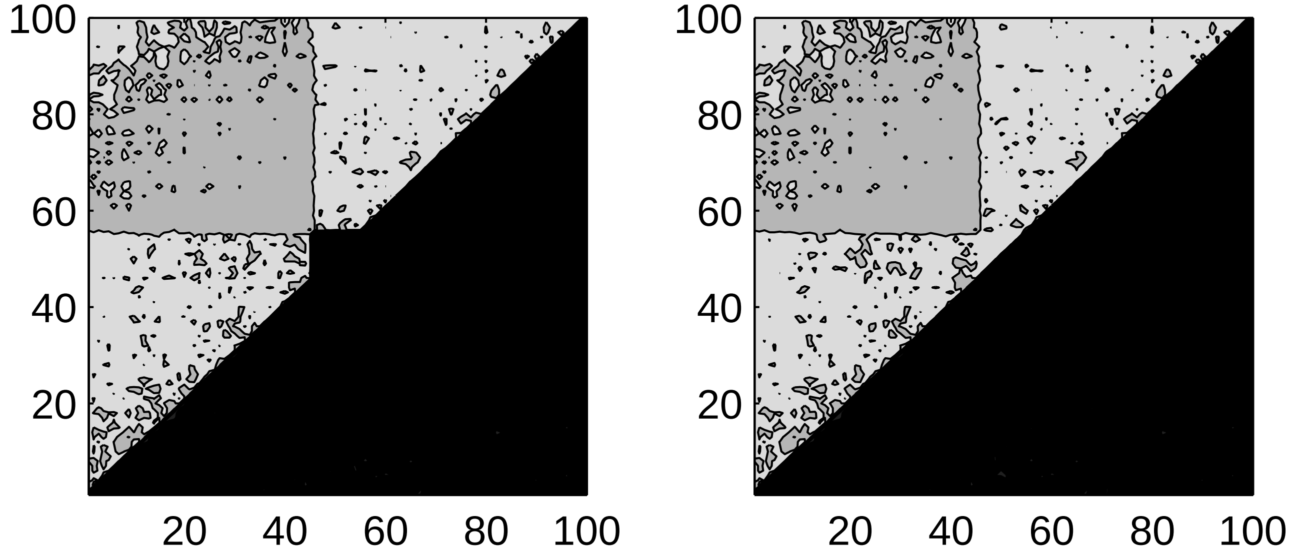
where $Y_{11}^T R_m \in \mathbb{C}^{m\times m}$ and $R_m X_{11} \in \mathbb{C}^{m\times m}$ are in anti-triangular form and $Z_{22} \in \mathbb{C}^{(n-2m)\times(n-2m)}$ has only eigenvalues in $\{\lambda \in \mathbb{C} : \alpha > |\lambda| > \frac{1}{\alpha}\}$.

Step 4: Anti-triangularize Z_{22} by some expensive, but accurate method (e.g., palindromic Jacobi algorithm, Schröder's palindromic QR algorithm).

Numerical experiments

Test: 100 random 100×100 matrices with 10 eigenvalues in an annulus in the complex plane with outer radius $1+10^{-12}$ and inner radius $1/(1+10^{-12})$.

Typical behavior: (black: elements of modulus larger than one; light grey: elements of modulus 10^{-15})

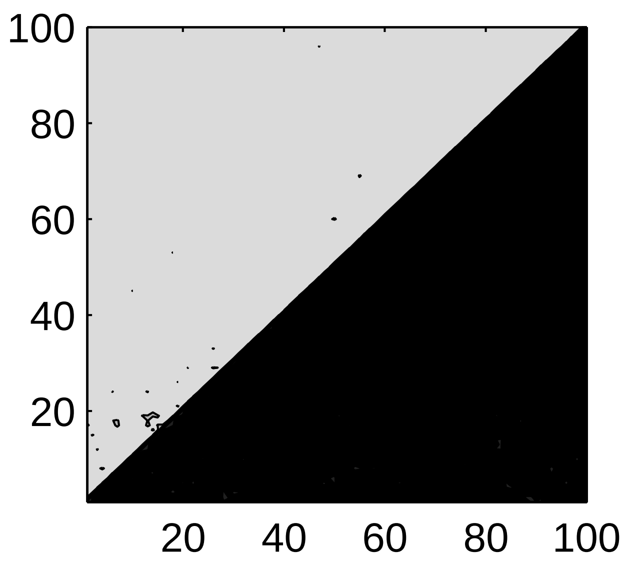


left: method 3 before applying Jacobi to the small problem
right: method 3 after applying Jacobi to the small problem

Numerical experiments

Test: 100 random 100×100 matrices with 10 eigenvalues in an annulus in the complex plane with outer radius $1+10^{-12}$ and inner radius $1/(1+10^{-12})$.

Typical behavior: (black: elements of modulus larger than one; light grey: elements of modulus 10^{-15})



method 3 with applying Jacobi to the small submatrix followed by one full sweep of Jacobi.