A Jacobi-Davidson method for two real parameter nonlinear eigenvalue problems arising from delay differential equations

Heinrich Voss
voss@tuhh.de

Joint work with Karl Meerbergen (KU Leuven) and Christian Schröder (TU Berlin)

Hamburg University of Technology
Institute of Mathematics
1. Problem definition
2. Small critical delay problems
3. Iterative projection methods for nonlinear eigenproblems
4. A Jacobi-Davidson-type method for two-real-param. EVP
5. Numerical Experience
Outline

1. Problem definition
2. Small critical delay problems
3. Iterative projection methods for nonlinear eigenproblems
4. A Jacobi-Davidson-type method for two-real-param. EVP
5. Numerical Experience
Consider the linear time-invariant delay differential equation

\[ M \dot{x}(t) + Ax(t) + Bx(t - \tau) = 0 \]

with given \( M, A, B \in \mathbb{C}^{n \times n}; \tau \geq 0 \) is the delay.
Consider the linear time-invariant delay differential equation

\[
M \dot{x}(t) + Ax(t) + Bx(t - \tau) = 0
\]

with given \( M, A, B \in \mathbb{C}^{n \times n} \); \( \tau \geq 0 \) is the delay.

This system is (asymptotically) stable if, for every bounded initial condition, it holds that \( x(t) \to 0 \) as \( t \to \infty \).
Consider the linear time-invariant delay differential equation

\[ M \dot{x}(t) + Ax(t) + Bx(t - \tau) = 0 \]

with given \( M, A, B \in \mathbb{C}^{n \times n}; \tau \geq 0 \) is the delay.

This system is (asymptotically) stable if, for every bounded initial condition, it holds that \( x(t) \to 0 \) as \( t \to \infty \).

Necessary and sufficient is that the spectrum of the nonlinear eigenvalue problem

\[ \lambda Mu + Au + e^{-\tau \lambda} Bu = 0 \]

is contained in the open left half–plane.
Approach at hand: homotopy, i.e.
follow eigenvalues close to imaginary axis for changing $\tau$
Approach at hand: homotopy, i.e.
follow eigenvalues close to imaginary axis for changing $\tau$

Problem expensive and unreliable; a different eigenvalue curve may have crossed the imaginary axis before the followed one arrives it.
Approach at hand: homotopy, i.e. follow eigenvalues close to imaginary axis for changing $\tau$

Problem expensive and unreliable; a different eigenvalue curve may have crossed the imaginary axis before the followed one arrives it.

Wanted: critical delays $\tau$ where system changes stability necessary: system has a purely imaginary eigenvalue
1. Problem definition

2. small critical delay problems

3. Iterative projection methods for nonlinear eigenproblems

4. A Jacobi-Davidson-type method for two-real-param. EVP

5. Numerical Experience
Recall the DEVP

\[(i\omega M + A + e^{-i\omega \tau} B)u = 0\]

where \(M, A, B \in \mathbb{C}^{n \times n}\).

We are interested in solutions \((\omega, \tau, u) \in \mathbb{R} \times \mathbb{R} \times \mathbb{C}^n, u \neq 0\).
Solving small critical delay problems

Recall the DEVP

\[(i\omega M + A + e^{-i\omega \tau} B)u = 0\]

where \(M, A, B \in \mathbb{C}^{n \times n}\).

We are interested in solutions \((\omega, \tau, u) \in \mathbb{R} \times \mathbb{R} \times \mathbb{C}^n, u \neq 0\).

Introducing the parameter \(\mu = e^{-i\omega \tau}\) translates the problem to the two-parameter problem

\[i\omega Mu + Au + \mu Bu = 0.\]
Recall the DEVP

\[(i\omega M + A + e^{-i\omega \tau} B)u = 0\]

where \(M, A, B \in \mathbb{C}^{n \times n}\).

We are interested in solutions \((\omega, \tau, u) \in \mathbb{R} \times \mathbb{R} \times \mathbb{C}^n, u \neq 0\).

Introducing the parameter \(\mu = e^{-i\omega \tau}\) translates the problem to the two-parameter problem

\[i\omega Mu + Au + \mu Bu = 0.\]

Note that \(\mu\) lies on the unit circle, thus \(\bar{\mu} = \mu^{-1}\).
Solving small critical delay problems

Recall the DEVP

\[(i\omega M + A + e^{-i\omega \tau} B)u = 0\]

where \(M, A, B \in \mathbb{C}^{n \times n}\).

We are interested in solutions \((\omega, \tau, u) \in \mathbb{R} \times \mathbb{R} \times \mathbb{C}^n, u \neq 0\).

Introducing the parameter \(\mu = e^{-i\omega \tau}\) translates the problem to the two-parameter problem

\[i\omega Mu + Au + \mu Bu = 0.\]

Note that \(\mu\) lies on the unit circle, thus \(\bar{\mu} = \mu^{-1}\).

Hence the complex conjugate equation reads

\[-i\omega \bar{M}\bar{u} + \bar{A}\bar{u} + \mu^{-1} \bar{B}\bar{u} = 0.\]
Using Kronecker products we eliminate $\omega$.
Using Kronecker products we eliminate $\omega$

\[
0 = -(i\omega Mu) \otimes \bar{M}\bar{u} - Mu \otimes (-i\omega \bar{M}\bar{u}) \\
= (Au + \mu Bu) \otimes \bar{M}\bar{u} + Mu \otimes (\bar{A}\bar{u} + \mu^{-1}\bar{B}\bar{u}) \\
= ((A + \mu B) \otimes \bar{M} + M \otimes (\bar{A} + \mu^{-1}\bar{B}))(u \otimes \bar{u}).
\]

and arrive at a rational eigenvalue problem.
Solving small critical delay problems

Using Kronecker products we eliminate $\omega$

\[
0 = -(i\omega Mu) \otimes \tilde{M} \tilde{u} - Mu \otimes (-i\omega \tilde{M} \tilde{u}) \\
= (Au + \mu Bu) \otimes \tilde{M} \tilde{u} + Mu \otimes (\tilde{A} \tilde{u} + \mu^{-1} \tilde{B} \tilde{u}) \\
= ((A + \mu B) \otimes \tilde{M} + M \otimes (\tilde{A} + \mu^{-1} \tilde{B})) (u \otimes \tilde{u}).
\]

and arrive at a rational eigenvalue problem.

Expanding and multiplication by $\mu$ yields the quadratic eigenvalue problem

\[
\mu^2 (B \otimes \tilde{M}) z + \mu (A \otimes \tilde{M} + M \otimes \tilde{A}) z + (M \otimes \tilde{B}) z = 0.
\]
Solving small critical delay problems

Using Kronecker products we eliminate \( \omega \)

\[
0 = -(i\omega Mu) \otimes \bar{M}\bar{u} \rightarrow Mu \otimes (i\omega\bar{M}\bar{u}) \\
= (Au + \mu Bu) \otimes \bar{M}\bar{u} + Mu \otimes (\bar{A}\bar{u} + \mu^{-1}\bar{B}\bar{u}) \\
= ((A + \mu B) \otimes \bar{M} + M \otimes (\bar{A} + \mu^{-1}\bar{B}))(u \otimes \bar{u}).
\]

and arrive at a rational eigenvalue problem.

Expanding and multiplication by \( \mu \) yields the quadratic eigenvalue problem

\[
\mu^2 (B \otimes \bar{M})z + \mu (A \otimes \bar{M} + M \otimes \bar{A})z + (M \otimes \bar{B})z = 0.
\]

We have thus shown that if \((\omega, \tau, u)\) is a solution of the DEVP and \(\mu = e^{-i\omega \tau}\) is on the unit circle, then \((\mu, u \otimes \bar{u})\) is an eigenpair of the quadratic eigenvalue problem (QEVP) above.
The converse is also true:
The converse is also true:

**Theorem**
Let $M, A, B \in \mathbb{C}^{n \times n}$ with $M$ nonsingular. Then any eigenvector $z \in \mathbb{C}^{n^2}$ of QEVP corresponding to a simple eigenvalue $\mu \in \mathbb{C}$ can be written as $z = \alpha u_1 \otimes u_2$ for some vectors $u_1, u_2 \in \mathbb{C}^n$ and some $\alpha \in \mathbb{C}$.
Solving small critical delay problems

The converse is also true:

**Theorem**
Let \( M, A, B \in \mathbb{C}^{n \times n} \) with \( M \) nonsingular. Then any eigenvector \( z \in \mathbb{C}^{n^2} \) of QEVP corresponding to a simple eigenvalue \( \mu \in \mathbb{C} \) can be written as \( z = \alpha u_1 \otimes u_2 \) for some vectors \( u_1, u_2 \in \mathbb{C}^n \) and some \( \alpha \in \mathbb{C} \).

Moreover, if \( |\mu| = 1 \) then \( u_1 = \overline{u}_2 = u \) and there is an \( \omega \in \mathbb{R} \) such that DEVP holds.
The converse is also true:

**Theorem**

Let $M, A, B \in \mathbb{C}^{n \times n}$ with $M$ nonsingular. Then any eigenvector $z \in \mathbb{C}^{n^2}$ of QEVP corresponding to a simple eigenvalue $\mu \in \mathbb{C}$ can be written as $z = \alpha u_1 \otimes u_2$ for some vectors $u_1, u_2 \in \mathbb{C}^n$ and some $\alpha \in \mathbb{C}$.

Moreover, if $|\mu| = 1$ then $u_1 = \bar{u}_2 = u$ and there is an $\omega \in \mathbb{R}$ such that DEVP holds.

We have thus transformed the problem of finding solutions of the DEVP to the problem of finding eigenvalues of modulus one of a QEVP.
The converse is also true:

**Theorem**
Let $M, A, B \in \mathbb{C}^{n \times n}$ with $M$ nonsingular. Then any eigenvector $z \in \mathbb{C}^{n^2}$ of QEVP corresponding to a simple eigenvalue $\mu \in \mathbb{C}$ can be written as $z = \alpha u_1 \otimes u_2$ for some vectors $u_1, u_2 \in \mathbb{C}^n$ and some $\alpha \in \mathbb{C}$.

Moreover, if $|\mu| = 1$ then $u_1 = \bar{u}_2 = u$ and there is an $\omega \in \mathbb{R}$ such that DEVP holds.

We have thus transformed the problem of finding solutions of the DEVP to the problem of finding eigenvalues of modulus one of a QEVP.

This can be done by the structure preserving method similar to the method presented in Fassbender, Mackey, Mackey, Schröder (2008).
Once an eigenpair \((\mu, z)\) of QEVP is known, \(u\) can be recovered from \(z = \alpha u \otimes \bar{u}\).
Once an eigenpair \((\mu, z)\) of QEVP is known, \(u\) can be recovered from \(z = \alpha u \otimes \bar{u}\).

One possibility is to compute the dominant singular vector of the \(n \times n\) matrix \(Z\) with \(z = \text{vec}(Z)\). Alternatively, one could chose \(u\) as a column of \(Z\) scaled such that \(u\) has norm one.
Solving small critical delay problems

Once an eigenpair \((\mu, z)\) of QEVP is known, \(u\) can be recovered from \(z = \alpha u \otimes \bar{u}\).

One possibility is to compute the dominant singular vector of the \(n \times n\) matrix \(Z\) with \(z = \text{vec}(Z)\). Alternatively, one could choose \(u\) as a column of \(Z\) scaled such that \(u\) has norm one.

Subsequently \(\omega\) can be obtained by projection, i.e.

\[
\omega = \frac{-\text{Im}(Mu)^H(Au + \mu Bu)}{\|Mu\|^2_2}.
\]
Once an eigenpair \((\mu, z)\) of QEVP is known, \(u\) can be recovered from 
\[ z = \alpha u \otimes \bar{u}. \]

One possibility is to compute the dominant singular vector of the \(n \times n\) matrix \(Z\) with \(z = \text{vec}(Z)\). Alternatively, one could choose \(u\) as a column of \(Z\) scaled such that \(u\) has norm one.

Subsequently \(\omega\) can be obtained by projection, i.e.
\[
\omega = \frac{-\text{Im}(M u)^H (A u + \mu B u)}{\|M u\|_2^2}.
\]

Finally, \(\tau\) may be computed from \(\mu\) and \(\omega\) as 
\[
\tau = -\text{Im}(\ln(\mu))/\omega.
\]
Recall that the Kronecker product does, in general, not commute, i.e. $A \otimes B \neq B \otimes A$. However there is a symmetric permutation matrix $P$ such that $A \otimes B = P(B \otimes A)P$. Hence, QEVP can be rewritten as

$$\mu^2(B \otimes \tilde{M})z + \mu(A \otimes \tilde{M} + M \otimes \tilde{A})z + (M \otimes \tilde{B})z = 0.$$
Structure preserving method for QEVP

\[ \mu^2 (B \otimes \bar{M}) z + \mu (A \otimes \bar{M} + M \otimes \bar{A}) z + (M \otimes \bar{B}) z = 0. \]

Recall that the Kronecker product does, in general, not commute, i.e. \( A \otimes B \neq B \otimes A \). However there is a symmetric permutation matrix \( P \) such that \( A \otimes B = P(B \otimes A)P \).
Structure preserving method for QEVP

\[ \mu^2 (B \otimes \tilde{M}) z + \mu (A \otimes \tilde{M} + M \otimes \tilde{A}) z + (M \otimes \tilde{B}) z = 0. \]

Recall that the Kronecker product does, in general, not commute, i.e. \( A \otimes B \neq B \otimes A \). However there is a symmetric permutation matrix \( P \) such that \( A \otimes B = P (B \otimes A) P \).

Hence, QEVP can be rewritten as

\[ (\mu^2 A_2 + \mu A_1 + A_0) z = 0 \]

where

\[ A_2 = P \tilde{A}_0 P, \quad A_1 = P \tilde{A}_1 P, \quad \text{with} \quad P = P^{-1} = P^T \in \mathbb{R}^{n^2 \times n^2}. \]
Structure preserving method for QEVP

\[ \mu^2 (B \otimes \bar{M}) z + \mu (A \otimes \bar{M} + M \otimes \bar{A}) z + (M \otimes \bar{B}) z = 0. \]

Recall that the Kronecker product does, in general, not commute, i.e. \( A \otimes B \neq B \otimes A \). However there is a symmetric permutation matrix \( P \) such that \( A \otimes B = P (B \otimes A) P \).

Hence, QEVP can be rewritten as

\[ (\mu^2 A_2 + \mu A_1 + A_0) z = 0 \]

where

\[ A_2 = P \bar{A}_0 P, \quad A_1 = P \bar{A}_1 P, \quad \text{with} \quad P = P^{-1} = P^T \in \mathbb{R}^{n^2 \times n^2}. \]

Such problems are called PCP-palindromic eigenvalue problems.
Structure preserving method for QEVP

The method can be divided into three steps.
The method can be divided into three steps.

First, the quadratic eigenvalue problem is reformulated as

\[
\begin{pmatrix}
\mu & 0 & P \\
0 & P & 0 \\
\bar{A}_1 - \bar{A}_2 & \bar{A}_0 & 0 \\
\bar{A}_0 & \bar{A}_0 & P \\
\end{pmatrix}
\begin{pmatrix}
0 & P \\
A_1 - A_2 & A_0 \\
A_0 & A_0 \\
\end{pmatrix}
\begin{pmatrix}
\mu Z \\
Z \\
\end{pmatrix} = 0,
\]

which is a linear PCP-palindromic problem, because with \( P \) also \( [P \ P] \) is a real symmetric permutation.
Structure preserving method for QEVP

The method can be divided into three steps.

First, the quadratic eigenvalue problem is reformulated as

\[
\begin{pmatrix}
\mu & [0 & P] \\
& [\bar{A}_1 - \bar{A}_2 & \bar{A}_0] & [0 & P] \\
& [\bar{A}_0 & \bar{A}_0] & [0 & P] \\
\end{pmatrix} + 
\begin{pmatrix}
A_1 - A_2 & A_0 \\
& A_0 & A_0 \\
\end{pmatrix}
\begin{pmatrix}
\mu \\
z \\
\end{pmatrix} = 0,
\]

which is a linear PCP-palindromic problem, because with \(P\) also \([P \ P]\) is a real symmetric permutation.

Second, using the factorization

\[
\begin{pmatrix}
0 & P \\
P & 0 \\
\end{pmatrix} = UU^T,
\]

where

\[
U = \frac{1}{\sqrt{2}} \begin{pmatrix} I & iP \\ P & -iI \end{pmatrix}
\]

we define

\[
C := U^H \begin{pmatrix}
A_1 - A_2 & A_0 \\
& A_0 & A_0 \\
\end{pmatrix} \bar{U}.
\]
Since $U$ is unitary, i.e. $U^H U = I = U^T \bar{U}$, this pencil is equivalent to

$$U^H \left( \mu \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} \begin{bmatrix} \bar{A}_1 - \bar{A}_2 & \bar{A}_0 \\ \bar{A}_0 & \bar{A}_0 \end{bmatrix} \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} + \begin{bmatrix} A_1 - A_2 & A_0 \\ A_0 & A_0 \end{bmatrix} \right) \bar{U}$$

$$= \mu U^T \begin{bmatrix} \bar{A}_1 - \bar{A}_2 & \bar{A}_0 \\ \bar{A}_0 & \bar{A}_0 \end{bmatrix} U + U^H \begin{bmatrix} A_1 - A_2 & A_0 \\ A_0 & A_0 \end{bmatrix} \bar{U}$$

$$= \mu \tilde{C} + C.$$
Structure preserving method for QEVP

Since $U$ is unitary, i.e. $U^H U = I = U^T U^H$, this pencil is equivalent to

$$U^H \left( \mu \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} \begin{bmatrix} \bar{A}_1 - \bar{A}_2 & \bar{A}_0 \\ \bar{A}_0 & \bar{A}_0 \end{bmatrix} \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} + \begin{bmatrix} A_1 - A_2 & A_0 \\ A_0 & A_0 \end{bmatrix} \right) U$$

$$= \mu U^T \begin{bmatrix} \bar{A}_1 - \bar{A}_2 & \bar{A}_0 \\ \bar{A}_0 & \bar{A}_0 \end{bmatrix} U + U^H \begin{bmatrix} A_1 - A_2 & A_0 \\ A_0 & A_0 \end{bmatrix} \bar{U}$$

$$= \mu \bar{C} + C.$$  

Third, consider an eigenpair $(\theta, x)$ of the real generalized eigenvalue problem

$$\text{Re}(C)x = \theta \text{Im}(C)x$$

or, equivalently,

$$Cx + \frac{i + \theta}{i - \theta} \bar{C}x = 0.$$
Structure preserving method for QEVP

Since $U$ is unitary, i.e. $U^H U = I = U^T \bar{U}$, this pencil is equivalent to

$$U^H \left( \mu \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} \begin{bmatrix} \bar{A}_1 - \bar{A}_2 & \bar{A}_0 \\ \bar{A}_0 & \bar{A}_0 \end{bmatrix} \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} + \begin{bmatrix} A_1 - A_2 & A_0 \\ A_0 & A_0 \end{bmatrix} \right) \bar{U}$$

$$= \mu U^T \begin{bmatrix} \bar{A}_1 - \bar{A}_2 & \bar{A}_0 \\ \bar{A}_0 & \bar{A}_0 \end{bmatrix} U + U^H \begin{bmatrix} A_1 - A_2 & A_0 \\ A_0 & A_0 \end{bmatrix} \bar{U}$$

$$= \mu \bar{C} + C.$$

Third, consider an eigenpair $(\theta, x)$ of the real generalized eigenvalue problem

$$\text{Re}(C)x = \theta \text{Im}(C)x$$

or, equivalently,

$$C x + \frac{i + \theta}{i - \theta} \bar{C} x = 0.$$ 

Note that $\mu := (i + \theta)/(i - \theta)$ is on the unit circle if and only if $\theta$ is real or $\theta = \infty$ (the latter resulting in $\mu = 1$). Note that real simple eigenvalues of real pencils can be stably computed by e.g. the real QZ algorithm.
Structure preserving method for QEVP

**Require:** \( M, A, B \in \mathbb{C}^{n \times n} \)

**Ensure:** solutions \((\omega_j, \tau_j, u_j)_{j=1,...}^\infty\) of \((i\omega M + A + e^{-i\omega \tau} B)u = 0\)

1: \( A_0 = M \otimes \bar{B}, \ A_1 = A \otimes M + M \otimes \bar{A} \)

2: Construct the permutation \( P \)

3: \[ C = \frac{1}{2} \begin{bmatrix} I & iP \\ P & -il \end{bmatrix}^H \begin{bmatrix} A_1 - PA_0P & A_0 \\ A_0 & A_0 \end{bmatrix} \begin{bmatrix} I & iP \\ P & -il \end{bmatrix} \]

4: Compute all eigenpairs \((\theta_j, x_j)\) of \( \text{Re}(C)x = \theta \text{Im}(C)x \) where \( \theta_j \) is real

5: **for** \( j=1,... \) **do**

6: \( \mu_j = \frac{i+\theta_j}{i-\theta_j} \)

7: \( z_j = \begin{bmatrix} I & -iP \end{bmatrix} x_j \)

8: Compute \( u_j \) as dominant singular vector of \( \text{mat}(z_j) \)

9: \( \omega_j = \frac{-\text{Im}((M\mu_j)^H(Au_j+\mu_j Bu_j))}{\|M\mu_j\|_2^2} \)

10: \( \tau_j = \frac{-\text{Im}(\ln(\mu_j))}{\omega_j} \)

11: **end for**
Structure preserving method for QEVP

Require: \( M, A, B \in \mathbb{C}^{n \times n} \)

Ensure: solutions \((\omega_j, \tau_j, u_j)_{j=1,...} \) of \((i\omega M + A + e^{-i\omega \tau} B)u = 0\)

1: \( A_0 = M \otimes \bar{B}, A_1 = A \otimes M + M \otimes \bar{A} \)

2: Construct the permutation \( P \)

3: \( C = \frac{1}{2} \begin{bmatrix} 1 & iP \\ P & -il \end{bmatrix}^H \begin{bmatrix} A_1 - PA_0P & A_0 \\ A_0 & A_0 \end{bmatrix} \begin{bmatrix} 1 & iP \\ P & -il \end{bmatrix} \)

4: Compute all eigenpairs \((\theta_j, x_j)\) of \( \text{Re}(C)x = \theta \text{Im}(C)x \) where \( \theta_j \) is real

5: for \( j=1,... \) do

6: \( \mu_j = \frac{i+\theta_j}{i-\theta_j} \)

7: \( z_j = [I - iP]x_j \)

8: Compute \( u_j \) as dominant singular vector of \( \text{mat}(z_j) \)

9: \( \omega_j = \frac{-\text{Im}((Mu_j)^H(Au_j + \mu_j Bu_j))}{\|Mu_j\|_2^2} \)

10: \( \tau_j = \frac{-\text{Im}(\ln(\mu_j))}{\omega_j} \)

11: end for

The dominant computational part is step 4 with a complexity of \( O(n^6) \).
Outline

1. Problem definition
2. Small critical delay problems
3. Iterative projection methods for nonlinear eigenproblems
4. A Jacobi-Davidson-type method for two-real-param. EVP
5. Numerical Experience
Iterative projection methods

For linear sparse eigenproblems

\[ T(\lambda) = \lambda B - A \]

very efficient methods are iterative projection methods (Lanczos, Arnoldi, Jacobi–Davidson method, e.g.), where approximations to the wanted eigenvalues and eigenvectors are obtained from projections of the eigenproblem to subspaces of small dimension which are expanded in the course of the algorithm.
Iterative projection methods for nonlinear eigenproblems

For linear sparse eigenproblems

\[ T(\lambda) = \lambda B - A \]

very efficient methods are iterative projection methods (Lanczos, Arnoldi, Jacobi–Davidson method, e.g.), where approximations to the wanted eigenvalues and eigenvectors are obtained from projections of the eigenproblem to subspaces of small dimension which are expanded in the course of the algorithm.

Generalizations to nonlinear sparse eigenproblems

Iterative projection methods

For linear sparse eigenproblems

\[ T(\lambda) = \lambda B - A \]

very efficient methods are iterative projection methods (Lanczos, Arnoldi, Jacobi–Davidson method, e.g.), where approximations to the wanted eigenvalues and eigenvectors are obtained from projections of the eigenproblem to subspaces of small dimension which are expanded in the course of the algorithm.

Generalizations to nonlinear sparse eigenproblems

- Arnoldi method: quadratic probl.: Meerbergen (2001)
Iterative projection methods for nonlinear eigenproblems

For linear sparse eigenproblems

\[ T(\lambda) = \lambda B - A \]

very efficient methods are iterative projection methods (Lanczos, Arnoldi, Jacobi–Davidson method, e.g.), where approximations to the wanted eigenvalues and eigenvectors are obtained from projections of the eigenproblem to subspaces of small dimension which are expanded in the course of the algorithm.

Generalizations to nonlinear sparse eigenproblems

- Arnoldi method: quadratic probl.: Meerbergen (2001)
- Jacobi-Davidson:
Iterative projection method

**Require:** Initial basis $V$ with $V^H V = I$; set $m = 1$

1. **while** $m \leq$ number of wanted eigenvalues **do**
2. compute eigenpair $(\mu, y)$ of projected problem $V^T T(\lambda) V y = 0$.
3. determine Ritz vector $u = V y$, $\|u\| = 1$, and residual $r = T(\mu) u$
4. **if** $\|r\| < \varepsilon$ **then**
5. accept approximate eigenpair $\lambda_m = \mu$, $x_m = u$; increase $m \leftarrow m + 1$
6. reduce search space $V$ if necessary
7. choose approximation $(\lambda_m, u)$ to next eigenpair, and compute $r = T(\lambda_m) u$
8. **end if**
9. expand search space $V = [V, v_{\text{new}}]$
10. update projected problem
11. **end while**
Iterative projection method

Require: Initial basis $V$ with $V^HV = I$; set $m = 1$
1: while $m \leq$ number of wanted eigenvalues do
2: compute eigenpair $(\mu, y)$ of projected problem $V^T T(\lambda) V y = 0$.
3: determine Ritz vector $u = V y$, $\|u\| = 1$, and residual $r = T(\mu)u$
4: if $\|r\| < \varepsilon$ then
5: accept approximate eigenpair $\lambda_m = \mu$, $x_m = u$; increase $m \leftarrow m + 1$
6: reduce search space $V$ if necessary
7: choose approximation $(\lambda_m, u)$ to next eigenpair, and compute $r = T(\lambda_m)u$
8: end if
9: expand search space $V = [V, v_{\text{new}}]$
10: update projected problem
11: end while

Main tasks
- expand search space
- choose eigenpair of projected problem (locking, purging)
Given subspace $\mathcal{V} \subset \mathbb{C}^n$. Expand $\mathcal{V}$ by a direction with high approximation potential for the next wanted eigenvector.
Iterative projection methods for nonlinear eigenproblems

Expanding the subspace

Given subspace $\mathcal{V} \subset \mathbb{C}^n$. Expand $\mathcal{V}$ by a direction with high approximation potential for the next wanted eigenvector.

Let $\theta$ be an eigenvalue of the projected problem

$$V^H T(\lambda) V y = 0$$

and $x = V y$ corresponding Ritz vector, then inverse iteration yields suitable candidate

$$v := T(\theta)^{-1} T'(\theta) x$$

BUT: In each step have to solve large linear system with varying matrix and in a truly large problem the vector $v$ will not be accessible but only an inexact solution $\tilde{v} := v + e$ of $T(\theta) v = T'(\theta) x$, and the next iterate will be a solution of the projection of $T(\lambda) x = 0$ upon the expanded space $\tilde{V} := \text{span} \{ V, \tilde{v} \}$. 
Expanding the subspace

Given subspace $\mathcal{V} \subset \mathbb{C}^n$. Expand $\mathcal{V}$ by a direction with high approximation potential for the next wanted eigenvector.

Let $\theta$ be an eigenvalue of the projected problem

$$V^H T(\lambda) V y = 0$$

and $x = V y$ corresponding Ritz vector, then inverse iteration yields suitable candidate

$$v := T(\theta)^{-1} T'(\theta) x$$

BUT: In each step have to solve large linear system with varying matrix
Expanding the subspace

Given subspace $\mathcal{V} \subset \mathbb{C}^n$. Expand $\mathcal{V}$ by a direction with high approximation potential for the next wanted eigenvector.

Let $\theta$ be an eigenvalue of the projected problem

$$V^H T(\lambda) V y = 0$$

and $x = V y$ corresponding Ritz vector, then inverse iteration yields suitable candidate

$$v := T(\theta)^{-1} T'(\theta) x$$

BUT: In each step have to solve large linear system with varying matrix and in a truly large problem the vector $v$ will not be accessible but only an inexact solution $\tilde{v} := v + e$ of $T(\theta) v = T'(\theta) x$, and the next iterate will be a solution of the projection of $T(\lambda) x = 0$ upon the expanded space $\tilde{\mathcal{V}} := \text{span}\{\mathcal{V}, \tilde{v}\}$. 
We assume that $x$ is already a good approximation to an eigenvector of $T(\cdot)$. Then $v$ will be an even better approximation, and therefore the eigenvector we are looking for will be very close to the plane $E := \text{span}\{x, v\}$. If the angle between these two planes is small, then the projection of $T(\lambda)$ upon $\tilde{V}$ should be similar to the one upon $\text{span}\{V, v\}$, and the approximation properties of inverse iteration should be maintained. If this angle can become large, then it is not surprising that the convergence properties of inverse iteration are not reflected by the projection method.
We assume that $x$ is already a good approximation to an eigenvector of $T(\cdot)$. Then $v$ will be an even better approximation, and therefore the eigenvector we are looking for will be very close to the plane $E := \text{span}\{x, v\}$.

We therefore neglect the influence of the orthogonal complement of $x$ in $\mathcal{V}$ on the next iterate and discuss the nearness of the planes $E$ and $\tilde{E} := \text{span}\{x, \tilde{v}\}$. 
We assume that \( x \) is already a good approximation to an eigenvector of \( T(\cdot) \). Then \( v \) will be an even better approximation, and therefore the eigenvector we are looking for will be very close to the plane \( E := \text{span}\{x, v\} \).

We therefore neglect the influence of the orthogonal complement of \( x \) in \( V \) on the next iterate and discuss the nearness of the planes \( E \) and \( \tilde{E} := \text{span}\{x, \tilde{v}\} \).

If the angle between these two planes is small, then the projection of \( T(\lambda) \) upon \( \tilde{V} \) should be similar to the one upon \( \text{span}\{V, v\} \), and the approximation properties of inverse iteration should be maintained.
We assume that $x$ is already a good approximation to an eigenvector of $T(\cdot)$. Then $v$ will be an even better approximation, and therefore the eigenvector we are looking for will be very close to the plane $E := \text{span}\{x, v\}$.

We therefore neglect the influence of the orthogonal complement of $x$ in $V$ on the next iterate and discuss the nearness of the planes $E$ and $\tilde{E} := \text{span}\{x, \tilde{v}\}$.

If the angle between these two planes is small, then the projection of $T(\lambda)$ upon $\tilde{V}$ should be similar to the one upon $\text{span}\{V, v\}$, and the approximation properties of inverse iteration should be maintained.

If this angle can become large, then it is not surprising that the convergence properties of inverse iteration are not reflected by the projection method.
Theorem

Let $\phi_0 = \arccos(x^Tv)$ denote the angle between $x$ and $v$, and the relative error of $\tilde{v}$ by $\varepsilon := \|e\|$. 
Let $\phi_0 = \arccos(x^T v)$ denote the angle between $x$ and $v$, and the relative error of $\tilde{v}$ by $\varepsilon := \|e\|$. 

Then the maximal possible acute angle between the planes $E$ and $\tilde{E}$ is

$$
\beta(\varepsilon) = \begin{cases} 
\arccos \sqrt{1 - \varepsilon^2 / \sin^2 \phi_0} & \text{if } \varepsilon \leq |\sin \phi_0| \\
\frac{\pi}{2} & \text{if } \varepsilon \geq |\sin \phi_0|
\end{cases}
$$
Iterative projection methods for nonlinear eigenproblems

“Proof”
Obviously for every $\alpha \in \mathbb{R}$, $\alpha \neq 0$ the plane $E$ is also spanned by $x$ and $x + \alpha v$. 
Obviously for every $\alpha \in \mathbb{R}$, $\alpha \neq 0$ the plane $E$ is also spanned by $x$ and $x + \alpha v$.

If $\tilde{E}(\alpha)$ is the plane which is spanned by $x$ and a perturbed realization $x + \alpha v + e$ of $x + \alpha v$ then by the same arguments as in the proof of the Theorem the maximum angle between $E$ and $\tilde{E}(\alpha)$ is

$$
\gamma(\alpha, \varepsilon) = \begin{cases} 
\arccos \sqrt{1 - \varepsilon^2 / \sin^2 \phi(\alpha)} & \text{if } \varepsilon \leq |\sin \phi(\alpha)| \\
\frac{\pi}{2} & \text{if } \varepsilon \geq |\sin \phi(\alpha)|
\end{cases}
$$

where $\phi(\alpha)$ denotes the angle between $x$ and $x + \alpha v$. 
Expansion by inexact inverse iteration

Obviously for every $\alpha \in \mathbb{R}$, $\alpha \neq 0$ the plane $E$ is also spanned by $x$ and $x + \alpha v$.

If $\tilde{E}(\alpha)$ is the plane which is spanned by $x$ and a perturbed realization $x + \alpha v + e$ of $x + \alpha v$ then by the same arguments as in the proof of the Theorem the maximum angle between $E$ and $\tilde{E}(\alpha)$ is

$$
\gamma(\alpha, \varepsilon) = \begin{cases} 
\arccos \sqrt{1 - \varepsilon^2 / \sin^2 \phi(\alpha)} & \text{if } \varepsilon \leq |\sin \phi(\alpha)| \\
\pi/2 & \text{if } \varepsilon \geq |\sin \phi(\alpha)|
\end{cases}
$$

where $\phi(\alpha)$ denotes the angle between $x$ and $x + \alpha v$.

Since the mapping

$$
\phi \mapsto \arccos \sqrt{1 - \varepsilon^2 / \sin^2 \phi}
$$

decreases monotonically the expansion of the search space by an inexact realization of $t := x + \alpha v$ is most robust with respect to small perturbations, if $\alpha$ is chosen such that $x$ and $x + \alpha v$ are orthogonal.
Expansion by inexact inverse iteration ct.

\[ t := x + \alpha v \] is orthogonal to \( x \) iff

\[ v = x - \frac{x^H x}{x^H T(\theta)^{-1} T'(\theta)x} T(\theta)^{-1} T(\theta)x.. \] \((*)\)
Expansion by inexact inverse iteration ct.

$t := x + \alpha v$ is orthogonal to $x$ iff

$$v = x - \frac{x^H x}{x^H T(\theta)^{-1} T'(\theta) x} T(\theta)^{-1} T(\theta)x.$$ (*)

which yields a maximum acute angle between $E$ and $\tilde{E}(\alpha)$

$$\gamma(\alpha, \varepsilon) = \begin{cases} 
\arccos \sqrt{1 - \varepsilon^2} & \text{if } \varepsilon \leq 1 \\
\frac{\pi}{2} & \text{if } \varepsilon \geq 1 
\end{cases}.$$
Expansion by inexact inverse iteration

\[ \phi_0 = 1 \]

angle between planes

inverse iteration

orthogonal correction

relative error
Expansion by inexact inverse iteration ct.

\[ \phi_0 = 0.1 \]

- Inverse Iteration
- Orthogonal Correction

Angle between planes vs. relative error.
Expansion by inexact inverse iteration ct.
The expansion

\[ \mathbf{v} = \mathbf{x} - \frac{\mathbf{x}^H \mathbf{x}}{\mathbf{x}^H \mathbf{T}^{-1} \mathbf{T}' \mathbf{x}} \mathbf{T}^{-1} \mathbf{T} \mathbf{t} \]

of the current search space \( \mathcal{V} \) is the solution of the equation

\[ \left( I - \frac{\mathbf{T}' \mathbf{x} \mathbf{x}^H}{\mathbf{x}^H \mathbf{T}' \mathbf{x}} \right) \mathbf{T} \left( I - \mathbf{x} \mathbf{x}^H \right) \mathbf{t} = -\mathbf{r}, \quad \mathbf{t} \perp \mathbf{x} \]
Iterative projection methods for nonlinear eigenproblems

Jacobi–Davidson method

The expansion

\[ \nu = x - \frac{x^H x}{x^H T(\theta)^{-1} T'(\theta)x} T(\theta)^{-1} T(\theta)x.. \quad (*) \]

of the current search space \( \mathcal{V} \) is the solution of the equation

\[ (I - \frac{T'(\theta)xx^H}{x^H T'(\theta)x}) T(\theta)(I - xx^H)t = -r, \quad t \perp x \]

This is the so called correction equation of the Jacobi–Davidson method which was derived in T. Betcke & Voss (2004) generalizing the approach of Sleijpen and van der Vorst (1996) for linear and polynomial eigenvalue problems.
Iterative projection methods for nonlinear eigenproblems

Jacobi–Davidson method

The expansion

\[ \nu = x - \frac{x^H x}{x^H T(\theta)^{-1} T'(\theta)x} T(\theta)^{-1} T(\theta)x, \] (*)

of the current search space \( \mathcal{V} \) is the solution of the equation

\[ (I - \frac{T'(\theta)xx^H}{x^H T'(\theta)x}) T(\theta)(I - xx^H)t = -r, \quad t \perp x. \]

This is the so called correction equation of the Jacobi–Davidson method which was derived in T. Betcke & Voss (2004) generalizing the approach of Sleijpen and van der Vorst (1996) for linear and polynomial eigenvalue problems.

Hence, the Jacobi–Davidson method is the most robust realization of an expansion of a search space such that the direction of inverse iteration is contained in the expanded space in the sense that it is least sensitive to inexact solves of linear systems \( T(\theta)\nu = T'(\theta)x \).
Iterative projection methods for nonlinear eigenproblems

Nonlinear Jacobi-Davidson

1: Start with orthonormal basis \( V \); set \( m = 1 \)
2: determine preconditioner \( M \approx T(\sigma)^{-1} \); \( \sigma \) close to first wanted eigenvalue
3: while \( m \leq \) number of wanted eigenvalues do
4: compute eigenpair \( (\mu, y) \) of projected problem \( V^T T(\lambda) V y = 0 \).
5: determine Ritz vector \( u = V y, \|u\| = 1 \), and residual \( r = T(\mu)u \)
6: if \( \|r\| < \varepsilon \) then
7: accept approximate eigenpair \( \lambda_m = \mu, x_m = u \); increase \( m \leftarrow m + 1 \)
8: reduce search space \( V \) if necessary
9: choose new preconditioner \( M \approx T(\mu) \) if indicated
10: choose approximation \( (\lambda_m, u) \) to next eigenpair, and compute \( r = T(\lambda_m)u \)
11: end if
12: solve approximately correction equation

\[
\left( I - \frac{T'(\mu)u u^H}{u^H T'(\mu) u} \right) T(\mu) \left( I - \frac{u u^H}{u^H u} \right) t = -r, \quad t \perp u
\]

13: \( t = t - V V^T t, \tilde{v} = t/\|t\|, \) reorthogonalize if necessary
14: expand search space \( V = [V, \tilde{v}] \)
15: update projected problem
16: end while
The correction equation is solved approximately by a few steps of an iterative solver (GMRES or BiCGStab).
The correction equation is solved approximately by a few steps of an iterative solver (GMRES or BiCGStab).

The operator $T(\sigma)$ is restricted to map the subspace $x^\perp$ into itself. Hence, if $M \approx T(\sigma)$ is a preconditioner of $T(\sigma)$, $\sigma \approx \mu$, then a preconditioner for an iterative solver of the correction equation should be modified correspondingly to

$$\tilde{M} := (I - \frac{T'(\mu)xx^H}{x^HT(\mu)x})M(I - \frac{xx^H}{x^Hx}).$$
Correction equation

The correction equation is solved approximately by a few steps of an iterative solver (GMRES or BiCGStab).

The operator $T(\sigma)$ is restricted to map the subspace $x^\perp$ into itself. Hence, if $M \approx T(\sigma)$ is a preconditioner of $T(\sigma)$, $\sigma \approx \mu$, then a preconditioner for an iterative solver of the correction equation should be modified correspondingly to

$$\tilde{M} := (I - \frac{T'(\mu)xx^H}{x^HT(\mu)x})M(I - \frac{xx^H}{x^Hx}).$$

Taking into account the projectors in the preconditioner, i.e. using $\tilde{M}$ instead of $M$, raises the cost of the preconditioned Krylov solver only slightly (cf. Sleijpen, van der Vorst). Only one additional linear solve with system matrix $M$ is required.
Common approach to Jacobi–Davidson

Expand the search space $\mathcal{V}$ by the direction defined by one step of Newton’s method applied to

$$
\begin{pmatrix}
T(\theta) x \\
x^T x - 1
\end{pmatrix}
= \begin{pmatrix}
0 \\
0
\end{pmatrix},
$$

and the quadratic convergence of Newton’s method explains the fast convergence of the Jacobi–Davidson method.

BUT the resulting correction equation is solved only very inexactly which spoils the good approximation properties of Newton’s method.
Common approach to Jacobi–Davidson

Expand the search space $V$ by the direction defined by one step of Newton’s method applied to

\[
\begin{pmatrix}
T(\theta)x \\
x^T x - 1
\end{pmatrix} = \begin{pmatrix}
0 \\
0
\end{pmatrix},
\]

i.e.

\[
\begin{pmatrix}
T(\theta) & T'(\theta)x \\
2x^T & 0
\end{pmatrix}
\begin{pmatrix}
v \\
\alpha
\end{pmatrix} = \begin{pmatrix}
T(\theta)x \\
0
\end{pmatrix},
\]

and the quadratic convergence of Newton’s method explains the fast convergence of the Jacobi–Davidson method.
Common approach to Jacobi–Davidson

Expand the search space $\mathcal{V}$ by the direction defined by one step of Newton’s method applied to

$$
\begin{pmatrix}
T(\theta)x \\
x^T x - 1
\end{pmatrix} = \begin{pmatrix}
0 \\
0
\end{pmatrix},
$$

i.e.

$$
\begin{pmatrix}
T(\theta) & T'(\theta)x \\
2x^T & 0
\end{pmatrix}
\begin{pmatrix}
v \\
\alpha
\end{pmatrix} = \begin{pmatrix}
T(\theta)x \\
0
\end{pmatrix}
$$

$$
\iff T(\theta)v + \alpha T'(\theta)x = T(\theta)x, \quad 2x^T v = 0
$$
Common approach to Jacobi–Davidson

Expand the search space \( \mathcal{V} \) by the direction defined by one step of Newton’s method applied to

\[
\begin{pmatrix}
T(\theta)x \\
x^T x - 1
\end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},
\]

i.e.

\[
\begin{pmatrix}
T(\theta) & T'(\theta)x \\
2x^T & 0
\end{pmatrix}
\begin{pmatrix}
v \\ \alpha
\end{pmatrix} = \begin{pmatrix}
T(\theta)x \\ 0
\end{pmatrix}
\]

\(\iff\ T(\theta)v + \alpha T'(\theta)x = T(\theta)x, \ 2x^T v = 0\)

\(\iff\ v = x - \alpha T(\theta)^{-1} T'(\theta)x, \ x^T v = 0\)
Common approach to Jacobi–Davidson

Expand the search space $\mathcal{V}$ by the direction defined by one step of Newton’s method applied to

$$\begin{pmatrix} T(\theta)x \\ x^T x - 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

i.e.

$$\begin{pmatrix} T(\theta) & T'(\theta)x \\ 2x^T & 0 \end{pmatrix} \begin{pmatrix} v \\ \alpha \end{pmatrix} = \begin{pmatrix} T(\theta)x \\ 0 \end{pmatrix} \iff T(\theta)v + \alpha T'(\theta)x = T(\theta)x, \ 2x^T v = 0$$

$$\iff \ v = x - \alpha T(\theta)^{-1} T'(\theta)x, \ x^T v = 0$$

$$\iff \ v = x - \frac{x^T x}{x^T T(\theta)^{-1} T(\theta)x} T(\theta)^{-1} T'(\theta)x,$$

and the quadratic convergence of Newton’s method explains the fast convergence of the Jacobi–Davidson method.
Common approach to Jacobi–Davidson

Expand the search space $\mathcal{V}$ by the direction defined by one step of Newton’s method applied to

$$
\begin{pmatrix}
T(\theta)x \\
x^T x - 1
\end{pmatrix}
= 
\begin{pmatrix}
0 \\
0
\end{pmatrix},
$$

i.e.

$$
\begin{pmatrix}
T(\theta) & T'(\theta)x \\
2x^T & 0
\end{pmatrix}
\begin{pmatrix}
v \\
\alpha
\end{pmatrix}
= 
\begin{pmatrix}
T(\theta)x \\
0
\end{pmatrix}
\iff 
T(\theta)v + \alpha T'(\theta)x = T(\theta)x, \quad 2x^T v = 0
$$

$$
\iff 
v = x - \alpha T(\theta)^{-1} T'(\theta)x, \quad x^T v = 0
$$

$$
\iff 
v = x - \frac{x^T x}{x^T T(\theta)^{-1} T(\theta)x} T(\theta)^{-1} T'(\theta)x,
$$

and the quadratic convergence of Newton’s method explains the fast convergence of the Jacobi–Davidson method.

BUT the resulting correction equation is solved only very inexactly which spoils the good approximation properties of Newton’s method.
More general approach

Assume that we are given a base method

\[ x^{k+1} = S(x^k, \theta_k), \quad \theta_{k+1} = p(x^{k+1}, \theta_k) \]

which converges locally

\[ \| \hat{x} - x^{k+1} \| = O(\| \hat{x} - x^k \|^{q_1}), \quad |\hat{\lambda} - \theta_{k+1}| = O(|\hat{\lambda} - \theta_k|^{q_2}). \]
More general approach

Assume that we are given a base method

\[ x^{k+1} = S(x^k, \theta_k), \quad \theta_{k+1} = p(x^{k+1}, \theta_k) \]

which converges locally

\[ \| \hat{x} - x^{k+1} \| = O(\| \hat{x} - x^k \|^q_1), \quad |\hat{\lambda} - \theta_{k+1}| = O(|\hat{\lambda} - \theta_k|^q_2). \]

Robustification:

(i) Find \( v^k = x^k + \alpha S(x^k, \theta_k)x^k, \quad (t^k)^H Bx^k = 0 \)

(ii) Choose \( x^{k+1} \in \text{span}\{x^k, v^k\} \), for instance solving the projected problem

\[ P_k^H T(\lambda) P_k z^k = 0, \quad x^{k+1} = P_k z^k \]

where \( P_k \) is the orthogonal projector onto \( \text{span}\{x^k, v^k\} \)

(iii) \( \theta_{k+1} = p(x^{k+1}) \)

where \( B \) is a Hermitian positive definite matrix (for instance if it is known in advance that the eigenvectors are \( B \)-orthogonal).
Outline

1. Problem definition
2. small critical delay problems
3. Iterative projection methods for nonlinear eigenproblems
4. A Jacobi-Davidson-type method for two-real-param. EVP
5. Numerical Experience
A JD-type method for two-real-param. EVP

Solve the NEP in two real parameters

\[ T(\omega, \tau)u := (i\omega M + A + e^{-i\omega \tau} B)u = 0 \]

by a straight forward adaption of (nonlinear) JD method:

1. Given an ansatz space \( \text{span}(V) \) of eigenvector approximations
2. Solve projected problem \( V^H T(\omega, \tau) Vz = 0 \)
3. Compute approximate eigenvectors \( u_i = Vz_i \) and residuals \( r_i = T(\omega_i, \tau_i) u_i \)
4. Stop, if enough eigentriples have converged
5. Compute a correction \( c \) of an approx. eigenvector \( \hat{u} \), i.e., \( \hat{u} + c \) is a better approximation
6. Expand \( V \leftarrow \text{orth}[V, c] \) and GOTO 2
A JD-type method for two-real-param. EVP

Solve the NEP in two real parameters

\[ T(\omega, \tau)u := (i\omega M + A + e^{-i\omega \tau} B)u = 0 \]

by a straightforward adaption of (nonlinear) JD method:

1. given an ansatz space \( \text{span}(V) \) of eigenvector approximations
2. known approximations of eigenvectors or random vector
3. solve projected problem
\[ V^H T(\omega, \tau) V z = 0 \]
4. compute approximate eigenvectors \( u_i \) and residuals \( r_i = T(\omega_i, \tau_i) u_i \)
5. stop, if enough eigentriples have converged
6. compute a correction \( c \) of an approx. eigenvector \( \hat{u} + c \) is a better approximation
7. expand \( V \leftarrow \text{orth}[V, c] \) and GOTO 2
A JD-type method for two-real-param. EVP

Solve the NEP in two real parameters

\[ T(\omega, \tau)u := (i\omega M + A + e^{-i\omega \tau} B)u = 0 \]

by a straightforward adaptation of (nonlinear) JD method:

1. given an ansatz space \( \text{span}(V) \) of eigenvector approximations
2. solve projected problem \( V^H T(\omega, \tau) Vz = 0 \)
3. use presented method for small problems
4. compute approximate eigenvectors \( u_i = Vz_i \) and residuals \( r_i = T(\omega_i, \tau_i) u_i \)
5. stop, if enough eigentriples have converged
6. compute a correction \( c \) of an approx. eigenvector \( \hat{u} \), i.e., \( \hat{u} + c \) is a better approximation
7. expand \( V \leftarrow \text{orth} \left[ V, c \right] \)

Beijing, May 2012
Solve the NEP in two real parameters

\[ T(\omega, \tau)u := (i\omega M + A + e^{-i\omega \tau} B)u = 0 \]

by a straightforward adaption of (nonlinear) JD method:

1. given an ansatz space \( \text{span}(V) \) of eigenvector approximations
2. solve projected problem \( V^H T(\omega, \tau) Vz = 0 \)
3. compute approximate eigenvectors \( u_i = Vz_i \) and residuals \( r_i = T(\omega_i, \tau_i)u_i \)
Solve the NEP in two real parameters

\[ T(\omega, \tau)u := (i\omega M + A + e^{-i\omega \tau} B)u = 0 \]

by a straightforward adaption of (nonlinear) JD method:

1. given an ansatz space \( \text{span}(V) \) of eigenvector approximations
2. solve projected problem \( V^H T(\omega, \tau) V z = 0 \)
3. compute approximate eigenvectors \( u_i = Vz_i \) and residuals \( r_i = T(\omega_i, \tau_i)u_i \)
4. stop, if enough eigentriples have converged

we use \( \|r_i\|_2 \leq \text{tol} \)
Solve the NEP in two real parameters

\[ T(\omega, \tau)u := (i\omega M + A + e^{-i\omega \tau} B)u = 0 \]

by a straightforward adaptation of (nonlinear) JD method:

1. given an ansatz space \( \text{span}(V) \) of eigenvector approximations
2. solve projected problem \( V^H T(\omega, \tau) Vz = 0 \)
3. compute approximate eigenvectors \( u_i = Vz_i \) and residuals \( r_i = T(\omega_i, \tau_i)u_i \)
4. stop, if enough eigentriples have converged
5. compute a correction \( c \) of an approx. eigenvector \( \hat{u} \)
   i.e., \( \hat{u} + c \) is a better approximation
   that’s the interesting part
Solve the NEP in two real parameters

\[
T(\omega, \tau)u := (i\omega M + A + e^{-i\omega \tau} B)u = 0
\]

by a straightforward adaptation of (nonlinear) JD method:

1. given an ansatz space \( \text{span}(V) \) of eigenvector approximations
2. solve projected problem \( V^H T(\omega, \tau) Vz = 0 \)
3. compute approximate eigenvectors \( u_i = Vz_i \) and residuals \( r_i = T(\omega_i, \tau_i) u_i \)
4. stop, if enough eigentriples have converged
5. compute a correction \( c \) of an approx. eigenvector \( \hat{u} \)
   i.e., \( \hat{u} + c \) is a better approximation
6. expand \( V \leftarrow \text{orth}[V, c] \) and GOTO 2
   repeat as long as necessary
Correction equation - medium sized problems

Given: approx. eigentriple \((\hat{\omega}, \hat{\tau}, \hat{u})\),
Wanted: correction \((\delta, \epsilon, c)\)
Correction equation - medium sized problems

Given: approx. eigentriple \((\hat{\omega}, \hat{\tau}, \hat{u})\),
Wanted: correction \((\delta, \epsilon, c)\)

- one step of Newton's method applied to
  \[
  \begin{bmatrix}
  T(\hat{\omega} + \delta, \hat{\tau} + \epsilon)(\hat{u} + c) \\
  \hat{u}^H c
  \end{bmatrix} = 0
  \]
Correction equation - medium sized problems

Given: approx. eigentriple \((\hat{\omega}, \hat{\tau}, \hat{u})\),
Wanted: correction \((\delta, \epsilon, c)\)

- one step of Newtons method applied to
  \[
  \begin{bmatrix}
  T(\hat{\omega} + \delta, \hat{\tau} + \epsilon)(\hat{u} + c) \\
  \hat{u}^H c
  \end{bmatrix} = 0
  \]

- \[
  \begin{bmatrix}
  T(\hat{\omega}, \hat{\tau}) & T_\omega(\hat{\omega}, \hat{\tau})\hat{u} & T_\tau(\hat{\omega}, \hat{\tau})\hat{u} \\
  \hat{u}^H & 0 & 0
  \end{bmatrix}
  \begin{bmatrix}
  c \\
  \delta \\
  \epsilon
  \end{bmatrix}
  = \begin{bmatrix}
  -\hat{r} \\
  0
  \end{bmatrix}.
  \]

linear system, \(n + 1\) complex equations,
\(n\) complex + 2 real unknowns \(\rightarrow\) no standard software available
Given: approx. eigentriple \((\hat{\omega}, \hat{\tau}, \hat{u})\),
Wanted: correction \((\delta, \epsilon, c)\)

- one step of Newtons method applied to

\[
\begin{bmatrix}
T(\hat{\omega} + \delta, \hat{\tau} + \epsilon)(\hat{u} + c)
\end{bmatrix} = 0
\]

\[
\begin{bmatrix}
T(\hat{\omega}, \hat{\tau}) & T_\omega(\hat{\omega}, \hat{\tau})\hat{u} & T_\tau(\hat{\omega}, \hat{\tau})\hat{u} \\
\hat{u}^H & 0 & 0
\end{bmatrix}
\begin{bmatrix}
c \\
\delta \\
\epsilon
\end{bmatrix}
= \begin{bmatrix}
-\hat{r} \\
0
\end{bmatrix}.
\]

Linear system, \(n + 1\) complex equations,
\(n\) complex + 2 real unknowns \(\rightarrow\) no standard software available
Correction equation - medium sized problems

Given: approx. eigentriple \((\hat{\omega}, \hat{\tau}, \hat{u})\),
Wanted: correction \((\delta, \epsilon, c)\)

- one step of Newtons method applied to
  
  \[
  \begin{bmatrix}
  T(\hat{\omega} + \delta, \hat{\tau} + \epsilon)(\hat{u} + c) & \\
  \hat{u}^H c
  \end{bmatrix} = 0
  \]

- \[
  \begin{bmatrix}
  T(\hat{\omega}, \hat{\tau}) & T_\omega(\hat{\omega}, \hat{\tau}) \hat{u} & T_\tau(\hat{\omega}, \hat{\tau}) \hat{u} \\
  \hat{u}^H & 0 & 0
  \end{bmatrix}
  \begin{bmatrix}
  c \\
  \delta
  \end{bmatrix} =
  \begin{bmatrix}
  -\hat{r} \\
  0
  \end{bmatrix}.
  \]

Linear system, \(n + 1\) complex equations,
\(n\) complex + 2 real unknowns → no standard software available

- \[
  \begin{bmatrix}
  \hat{T} & 0 & \hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u} \\
  0 & \hat{T} & \hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u} \\
  \hat{u}^H & 0 & 0 & 0 \\
  0 & \hat{u}^H & 0 & 0
  \end{bmatrix}
  \begin{bmatrix}
  c \\
  d
  \end{bmatrix} =
  \begin{bmatrix}
  -\hat{r} \\
  -\hat{r}
  \end{bmatrix}.
  \]

**Lemma:** \(d = \overline{c}, \delta, \epsilon \in \mathbb{R}\)
Given: approx. eigentriple \((\hat{\omega}, \hat{\tau}, \hat{u})\),
Wanted: correction \((\delta, \epsilon, c)\)

- one step of Newton's method applied to
  \[
  \begin{bmatrix}
    T(\hat{\omega} + \delta, \hat{\tau} + \epsilon)(\hat{u} + c) \\
    \hat{u}^H c
  \end{bmatrix} = 0
  \]

\[
\begin{bmatrix}
  T(\hat{\omega}, \hat{\tau}) & T_\omega(\hat{\omega}, \hat{\tau})\hat{u} & T_\tau(\hat{\omega}, \hat{\tau})\hat{u} \\
  \hat{u}^H & 0 & 0 \\
  0 & \hat{u}^H & 0
\end{bmatrix}
\begin{bmatrix}
  c \\
  \delta \\
  \epsilon
\end{bmatrix} = \begin{bmatrix}
  -\hat{r} \\
  0
\end{bmatrix}.
\]

linear system, \(n + 1\) complex equations,
\(n\) complex + 2 real unknowns → no standard software available

\[
\begin{bmatrix}
  \hat{T} & 0 & \hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u} \\
  0 & \hat{T} & \hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u} \\
  \hat{u}^H & 0 & 0 & 0 \\
  0 & \hat{u}^H & 0 & 0
\end{bmatrix}
\begin{bmatrix}
  c \\
  d \\
  \delta \\
  \epsilon
\end{bmatrix} = \begin{bmatrix}
  -\hat{r} \\
  -\hat{r}
\end{bmatrix}.
\]

\textbf{Lemma:} \(d = \overline{c}, \delta, \epsilon \in \mathbb{R}\)

What if \(n\) is large and a preconditioner is available for \(\hat{T}\) only?
Correction equation - large size

With

\[ T = \begin{bmatrix} \hat{T} & 0 \\ 0 & \hat{T} \end{bmatrix}, \quad K = \begin{bmatrix} \hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u} \\ \hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u} \end{bmatrix}, \quad U = \begin{bmatrix} \hat{u} & 0 \\ 0 & \hat{u} \end{bmatrix} \]

system reads

\[
\begin{bmatrix} T & K \\ U^H & 0 \end{bmatrix} \begin{bmatrix} c \\ \bar{c} \\ \delta \\ \varepsilon \end{bmatrix} = - \begin{bmatrix} r \\ \bar{r} \\ 0 \\ 0 \end{bmatrix}.
\]
A Jacobi-Davidson-type method for two-real-param. EVP

Correction equation - large size

With

\[
T = \begin{bmatrix}
\hat{T} & 0 \\
0 & \hat{T}
\end{bmatrix}, \quad K = \begin{bmatrix}
\hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u} \\
\hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u}
\end{bmatrix}, \quad U = \begin{bmatrix}
\hat{u} & 0 \\
0 & \hat{u}
\end{bmatrix}
\]

system reads

\[
\begin{bmatrix}
T & K \\
U^H & 0
\end{bmatrix}
\begin{bmatrix}
c \\ \bar{c} \\ \delta \\ \varepsilon
\end{bmatrix} = -\begin{bmatrix}
r \\ \bar{r} \\ 0 \\ 0
\end{bmatrix}.
\]

Eliminating \((\delta, \varepsilon)\) yields

\[
(I - K(U^HK)^{-1}U^H)T(I - UU^H)\begin{bmatrix}
c \\ \bar{c}
\end{bmatrix} = -(I - K(U^HK)^{-1}U^H)\begin{bmatrix}
r \\ \bar{r}
\end{bmatrix}
\]
Correction equation - large size

With

\[ T = \begin{bmatrix} \hat{T} & 0 \\ 0 & \overline{\hat{T}} \end{bmatrix}, \quad K = \begin{bmatrix} \hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u} \\ \overline{\hat{T}_\omega \hat{u}} & \overline{\hat{T}_\tau \hat{u}} \end{bmatrix}, \quad U = \begin{bmatrix} \hat{u} & 0 \\ 0 & \overline{\hat{u}} \end{bmatrix} \]

system reads

\[
\begin{bmatrix} T & K \\ U^H & 0 \end{bmatrix} \begin{bmatrix} c \\ \overline{c} \end{bmatrix} = - \begin{bmatrix} r \\ \overline{r} \end{bmatrix}.
\]

Eliminating \((\delta, \varepsilon)\) yields

\[
(I - K(U^H K)^{-1} U^H) T (I - U U^H) \begin{bmatrix} c \\ \overline{c} \end{bmatrix} = -(I - K(U^H K)^{-1} U^H) \begin{bmatrix} r \\ \overline{r} \end{bmatrix}
\]

looks like other JD correction equations:

\[
T \begin{bmatrix} c \\ \overline{c} \end{bmatrix} = - \begin{bmatrix} r \\ \overline{r} \end{bmatrix}
\]

is complemented by projectors
Correction equation - large size

With

\[ T = \begin{bmatrix} \hat{T} & 0 \\ 0 & \hat{T} \end{bmatrix}, \quad K = \begin{bmatrix} \hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u} \\ \hat{T}_\omega \hat{u} & \hat{T}_\tau \hat{u} \end{bmatrix}, \quad U = \begin{bmatrix} \hat{u} & 0 \\ 0 & \hat{u} \end{bmatrix} \]

system reads

\[
\begin{bmatrix}
T & K \\
U^H & 0
\end{bmatrix}
\begin{bmatrix}
c \\
\bar{c} \\
\delta \\
\varepsilon
\end{bmatrix}
=
\begin{bmatrix}
r \\
\bar{r} \\
0 \\
0
\end{bmatrix}.
\]

Eliminating \((\delta, \varepsilon)\) yields

\[
(I - K(U^H K)^{-1} U^H) T (I - U U^H) \begin{bmatrix}
\bar{c} \\
\bar{c}
\end{bmatrix}
= -(I - K(U^H K)^{-1} U^H) \begin{bmatrix}
r \\
\bar{r}
\end{bmatrix}
\]

looks like other JD correction equations: \(T \begin{bmatrix}
\bar{c} \\
\bar{c}
\end{bmatrix} = - \begin{bmatrix}
r \\
\bar{r}
\end{bmatrix}\)

is complemented by projectors

How does this help with the preconditioner?
Preconditioning the correction equation

Correction equation

\[
(I - K(U^H K)^{-1} U^H) \mathbf{T} (I - UU^H) \begin{bmatrix} \mathbf{c} \\ \mathbf{c} \end{bmatrix} = -(I - K(U^H K)^{-1} U^H) \begin{bmatrix} \mathbf{r} \\ \mathbf{r} \end{bmatrix}
\]
Preconditioning the correction equation

Correction equation

\[(I - K(U^H K)^{-1} U^H)T(I - UU^H) \begin{bmatrix} c \\ \bar{c} \end{bmatrix} = -(I - K(U^H K)^{-1} U^H) \begin{bmatrix} r \\ \bar{r} \end{bmatrix}\]

preconditioner:

\[(I - K(U^H K)^{-1} U^H) P(I - UU^H) \text{ with } P = \begin{bmatrix} P & 0 \\ 0 & \bar{P} \end{bmatrix}\]
Preconditioning the correction equation

Correction equation

\[
(I - \mathbf{K}(\mathbf{U}^H\mathbf{K})^{-1}\mathbf{U}^H)\mathbf{T}(I - \mathbf{U}\mathbf{U}^H) \begin{bmatrix} \mathbf{c} \\ \mathbf{c} \end{bmatrix} = -(I - \mathbf{K}(\mathbf{U}^H\mathbf{K})^{-1}\mathbf{U}^H) \begin{bmatrix} \mathbf{r} \\ \mathbf{r} \end{bmatrix}
\]

Preconditioner:

\[
(I - \mathbf{K}(\mathbf{U}^H\mathbf{K})^{-1}\mathbf{U}^H)\mathbf{P}(I - \mathbf{U}\mathbf{U}^H) \text{ with } \mathbf{P} = \begin{bmatrix} \mathbf{P} & 0 \\ 0 & \overline{\mathbf{P}} \end{bmatrix}
\]

Need to apply in a left preconditioned Krylov solver:

\[
(I - \mathbf{U}\mathbf{U}^H)\mathbf{P}^{-1}(I - \mathbf{K}(\mathbf{U}^H\mathbf{P}^{-1}\mathbf{K})^{-1}\mathbf{U}^H\mathbf{P}^{-1})(I - \mathbf{K}(\mathbf{U}^H\mathbf{K})^{-1}\mathbf{U}^H)\mathbf{T}(I - \mathbf{U}\mathbf{U}^H)
\]
Preconditioning the correction equation

Correction equation

\[
(I - K(U^HK)^{-1}U^H)T(I - UU^H) \begin{bmatrix} c \\ c \end{bmatrix} = -(I - K(U^HK)^{-1}U^H) \begin{bmatrix} r \\ r \end{bmatrix}
\]

preconditioner:

\[
(I - K(U^HK)^{-1}U^H)P(I - UU^H)
\]

with \( P = \begin{bmatrix} P & 0 \\ 0 & \bar{P} \end{bmatrix} \)

need to apply in a left preconditioned Krylov solver:

\[
(I - UU^H)\hat{P}^{-1}(I - K(U^H\hat{P}^{-1}K)^{-1}U^H\hat{P}^{-1})(I - K(U^HK)^{-1}U^H)T(I - UU^H)
\]

Efficient implementation needs one application of \( P^{-1}\hat{T} \) per iteration and additionally 2 \( T \) products and 3 \( P \) solves as in other JD variants
Alternative expansion

- sometimes the projected problem has no solution \((\text{Re}(C)x = \theta \text{Im}(C)x)\) has no real eigenvalues)
Some details

Alternative expansion

- sometimes the projected problem has no solution ($\text{Re}(C)x = \theta \text{Im}(C)x$ has no real eigenvalues)
- then what to use as approximate Ritz triple for the correction equation?

we use $\hat{\omega} = 0$, $e^{i\hat{\omega}\hat{\tau}} = \sigma$ (given) and $\hat{u} \in \text{span}(V)$ such that $\|T(\hat{\omega}, \hat{\tau})\hat{u}\|_2 = \min$
Some details

Alternative expansion

- Sometimes the projected problem has no solution ($\text{Re}(C)x = \theta \text{Im}(C)x$ has no real eigenvalues)

- Then what to use as approximate Ritz triple for the correction equation?

- We use $\hat{\omega} = 0$, $e^{i\hat{\omega}\hat{\tau}} = \sigma$ (given) and $\hat{u} \in \text{span}(V)$ such that $\| T(\hat{\omega}, \hat{\tau}) \hat{u} \|_2 = \min$
Some details

Alternative expansion

- sometimes the projected problem has no solution ($\text{Re}(C)x = \theta \text{Im}(C)x$ has no real eigenvalues)
- then what to use as approximate Ritz triple for the correction equation?
  - we use $\hat{\omega} = 0$, $e^{i\hat{\omega}\hat{\tau}} = \sigma$ (given) and $
\hat{u} \in \text{span}(V)$ such that $\|T(\hat{\omega}, \hat{\tau})\hat{u}\|_2 = \text{min}$

Restart

- cost grows like $O(k^6)$, so becomes prohibitive for larger $k$
Some details

Alternative expansion

- sometimes the projected problem has no solution \((\text{Re}(C)x = \theta \text{Im}(C)x)\) has no real eigenvalues
- then what to use as approximate Ritz triple for the correction equation?
- we use \(\tilde{\omega} = 0, e^{i\tilde{\omega}\hat{\tau}} = \sigma\) (given) and \(\hat{u} \in \text{span}(V)\) such that \(\|T(\tilde{\omega}, \hat{\tau})\hat{u}\|_2 = \min\)

Restart

- cost grows like \(\mathcal{O}(k^6)\), so becomes prohibitive for larger \(k\)
- restart with space spanned by converged eigenvectors and a few unconverged ones with the best residuals
Some details

Alternative expansion

- sometimes the projected problem has no solution \((\text{Re}(C)x = \theta \text{Im}(C)x)\) has no real eigenvalues

- then what to use as approximate Ritz triple for the correction equation?

- we use \(\hat{\omega} = 0, e^{i\hat{\omega}\hat{\tau}} = \sigma\) (given) and \(\hat{u} \in \text{span}(V)\) such that \(\|T(\hat{\omega}, \hat{\tau})\hat{u}\|_2 = \min\)

Restart

- cost grows like \(O(k^6)\), so becomes prohibitve for larger \(k\)

- restart with space spanned by converged eigenvectors and a few unconverged ones with the best residuals

Real problems

- if \(M, A, B\) are real, then eigentriples come in pairs \((\omega, \tau, u), (\omega, \tau, \bar{u})\)
Some details

Alternative expansion

- sometimes the projected problem has no solution ($\text{Re}(C)x = \theta \text{Im}(C)x$ has no real eigenvalues)
- then what to use as approximate Ritz triple for the correction equation?
- we use $\hat{\omega} = 0$, $e^{i\hat{\omega}\hat{\tau}} = \sigma$ (given) and $\hat{u} \in \text{span}(V)$ such that $\|T(\hat{\omega}, \hat{\tau})\hat{u}\|_2 = \min$

Restart

- cost grows like $\mathcal{O}(k^6)$, so becomes prohibitive for larger $k$
- restart with space spanned by converged eigenvectors and a few unconverged ones with the best residuals

Real problems

- if $M, A, B$ are real, then eigentriples come in pairs $(\omega, \tau, u), (-\omega, \tau, \bar{u})$
- when an eigenvector $u$ converges, add $\bar{u}$ to searchspace
Outline

1. Problem definition
2. Small critical delay problems
3. Iterative projection methods for nonlinear eigenproblems
4. A Jacobi-Davidson-type method for two-real-param. EVP
5. Numerical Experience
Consider the parabolic problem

\[ u_t - \nabla((1 + x^2 + y^2 + z^2)\nabla u) + [1, 0, -1]\nabla u + u - \alpha(1 + x^2)u(t - \tau) = 0 \quad (1) \]

with spatial variables \( x, y \) and \( z \) on \( \Omega = (0, 1) \times (0, 1) \times (0, 1) \) with Dirichlet boundary condition \( u = 0 \) on \( \partial \Omega \).
Consider the parabolic problem

\[ u_t - \nabla((1 + x^2 + y^2 + z^2)\nabla u) + [1, 0, -1]\nabla u + u - \alpha(1 + x^2)u(t - \tau) = 0 \quad (1) \]

with spatial variables \(x, y, z\) on \(\Omega = (0, 1) \times (0, 1) \times (0, 1)\) with Dirichlet boundary condition \(u = 0\) on \(\partial\Omega\).

A discretization with piecewise quadratic ansatz functions on a tetrahedral grid using COMSOL yielded an eigenvalue problem

\[ M\dot{x}(t) + Ax(t) + Bx(t - \tau) = 0 \]

of dimension \(n = 80623\).
Consider the parabolic problem
\[ u_t - \nabla((1 + x^2 + y^2 + z^2) \nabla u) + [1, 0, -1] \nabla u + u - \alpha(1 + x^2)u(t - \tau) = 0 \quad (1) \]
with spatial variables \( x, y \) and \( z \) on \( \Omega = (0, 1) \times (0, 1) \times (0, 1) \) with Dirichlet boundary condition \( u = 0 \) on \( \partial\Omega \).

A discretization with piecewise quadratic ansatz functions on a tetrahedral grid using COMSOL yielded an eigenvalue problem
\[ M\dot{x}(t) + Ax(t) + Bx(t - \tau) = 0 \]
of dimension \( n = 80623 \).

For \( \alpha = 100 \) the problem has four pairs of eigentriples.
Numerical Example

Convergence History Example 3

iteration
residual norm

TUHH
Heinrich Voss
Delay JD
Beijing, May 2012
x axis: number of iteration; tic=5; together 31
Numerical Example

Convergence History Example 3

- **Y axis:** norm of residual: \( \text{tic} = 10^{-2}; \) tolerance = \( 10^{-10} \)
Convergence History Example 3

- first iteration: alternative selection, then Ritz values
linear system: $P = \text{luinc}(A, 1e-3)$; GMRES terminated after at most 10 steps or if residual was reduced by $10^{-1}$
- first converged triple: 9 iterations, then expand by $\bar{u}$
Numerical Example

Convergence History Example 3

iteration
residual norm

then $8 + 1, 5 + 1, 5 + 1$ iterations for further triple pairs
Convergence History Example 3

iteration
residual norm

robust (lots of runs with random start vector)
 restarts after 12, 16, 21, 25, 28 iterations, no slowdown
Conclusion

two real instead of one complex parameter
"direct" method for small critical delay problems
robust JD variant for larger problems
\[(i\omega M + A + e^{i\omega \tau} B)u = T(\omega, \tau)u = 0\]

- two real instead of one complex parameter
- "direct" method for small critical delay problems
- robust JD variant for larger problems

References

- K. Meerbergen, C. Schröder, and H. Voss

- K. Meerbergen, C. Schröder, and H. Voss

Thanks for your attention!