Contour Integral Method for the Simulation of Accelerator Cavities

V. Pham-Xuan, W. Ackermann and H. De Gersem

Institut für Theorie Elektromagnetischer Felder

TECHNISCHE UNIVERSITÄT DARMSTADT

DESY meeting (14.11.2017)

Outline of the Talk

TECHNISCHE UNIVERSITÄT DARMSTADT

- Motivation
  - Iterative methods
  - Contour integral methods
- Formulation
  - Mathematical model
  - Eigenvalue algorithms
  - Contour integral methods
  - Multigrid method as a preconditioner
- Implementation
- Preliminary Results
- Possible Improvements

Presentation Outline

Motivation

Problem statement

Problem statement: we have to solve **a nonlinear eigenvalue problem (NEP)** where

Motivation

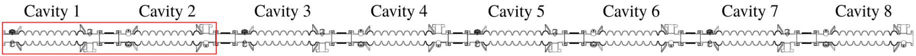
Problem statement

TECHNISCHE UNIVERSITÄT DARMSTADT

Problem statement: we have to solve **a nonlinear eigenvalue problem (NEP)** where

- the problem is large and sparse;
- the number of eigenvalues is large;

Figure: Chain of cavities (from [1])



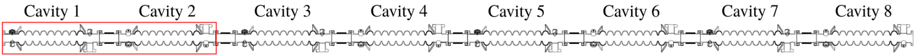
Motivation

Problem statement

Problem statement: we have to solve **a nonlinear eigenvalue problem (NEP)** where

- the problem is large and sparse;
- the number of eigenvalues is large;
- prior information about eigenvalues is available;

Figure: Chain of cavities (from [1])



Motivation

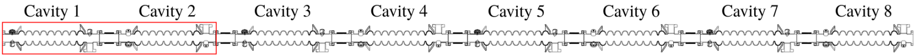
Problem statement

TECHNISCHE UNIVERSITÄT DARMSTADT

Problem statement: we have to solve **a nonlinear eigenvalue problem (NEP)** where

- the problem is large and sparse;
- the number of eigenvalues is large;
- prior information about eigenvalues is available;
- in several applications, one is only interested in a few eigenvalues within a certain range.

Figure: Chain of cavities (from [1])



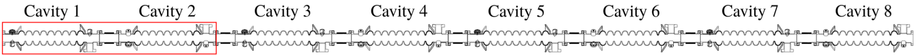
Motivation

Problem statement

Problem statement: we have to solve **a nonlinear eigenvalue problem (NEP)** where

- the problem is large and sparse;
- the number of eigenvalues is large;
- prior information about eigenvalues is available;
- in several applications, one is only interested in a few eigenvalues within a certain range.

Figure: Chain of cavities (from [1])



Available methods:

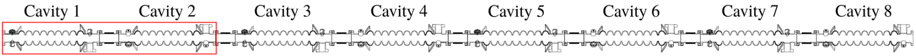
- Iterative methods: Jacobi-Davidson [2], Arnoldi, Lanczos, etc.

TECHNISCHE UNIVERSITÄT DARMSTADT

Problem statement: we have to solve **a nonlinear eigenvalue problem (NEP)** where

- the problem is large and sparse;
- the number of eigenvalues is large;
- prior information about eigenvalues is available;
- in several applications, one is only interested in a few eigenvalues within a certain range.

Figure: Chain of cavities (from [1])



Available methods:

- Iterative methods: Jacobi-Davidson [2], Arnoldi, Lanczos, etc.
- Contour integral methods: Beyn methods [3], resolvent sampling based Rayleigh-Ritz method (RSRR) [4], etc.

Presentation Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

Motivation
- Iterative methods
- Contour integral methods

Formulation
- Mathematical model
- Eigenvalue algorithms
- Contour integral methods
- Multigrid method as a preconditioner

Implementation

Preliminary Results

Possible Improvements

TECHNISCHE UNIVERSITÄT DARMSTADT

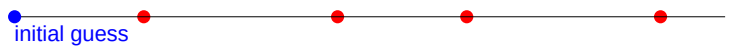
Lossless accelerator cavity: eigenvalues are on real axis

Motivation

Iterative methods (Jacobi-Davidson)

Lossless accelerator cavity: eigenvalues are on real axis

- choose an initial guess

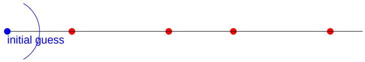


Motivation
Iterative methods (Jacobi-Davidson)

TECHNISCHE UNIVERSITÄT DARMSTADT

Lossless accelerator cavity: eigenvalues are on real axis

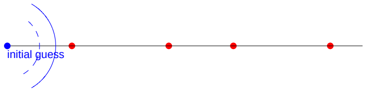
- choose an initial guess
- expand the search space ...



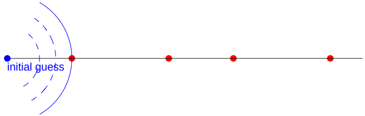
Motivation
Iterative methods (Jacobi-Davidson)

Lossless accelerator cavity: eigenvalues are on real axis

- choose an initial guess
- expand the search space ...



Lossless accelerator cavity: eigenvalues are on real axis



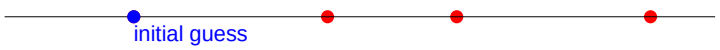
- choose an initial guess
- expand the search space ...
- until an approximate solution is found

Motivation

Iterative methods (Jacobi-Davidson)

TECHNISCHE UNIVERSITÄT DARMSTADT

Lossless accelerator cavity: eigenvalues are on real axis



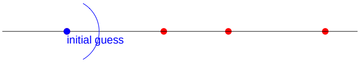
- choose an initial guess
- expand the search space ...
- until an approximate solution is found
- the solution becomes the new initial guess

Motivation

Iterative methods (Jacobi-Davidson)

TECHNISCHE UNIVERSITÄT DARMSTADT

Lossless accelerator cavity: eigenvalues are on real axis

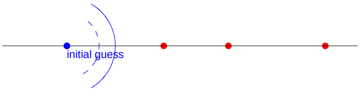


- choose an initial guess
- expand the search space ...
- until an approximate solution is found
- the solution becomes the new initial guess
- continue expanding the search space ...

Motivation

Iterative methods (Jacobi-Davidson)

Lossless accelerator cavity: eigenvalues are on real axis

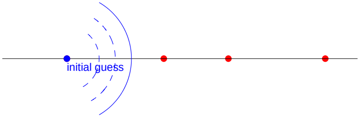


- choose an initial guess
- expand the search space ...
- until an approximate solution is found
- the solution becomes the new initial guess
- continue expanding the search space ...

Iterative methods (Jacobi-Davidson)

TECHNISCHE UNIVERSITÄT DARMSTADT

Lossless accelerator cavity: eigenvalues are on real axis

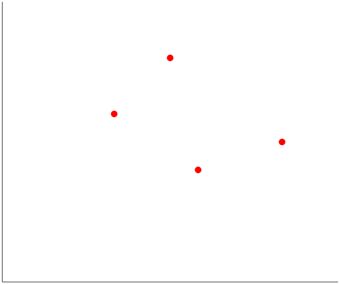


- choose an initial guess
- expand the search space ...
- until an approximate solution is found
- the solution becomes the new initial guess
- continue expanding the search space ...

Motivation

Iterative methods (Jacobi-Davidson)

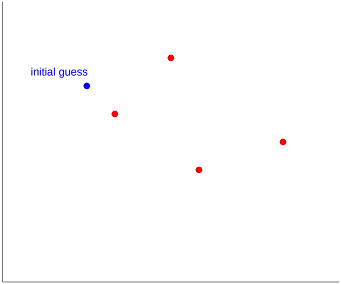
Lossy accelerator cavity: eigenvalues are in the complex plane



Motivation

Iterative methods (Jacobi-Davidson)

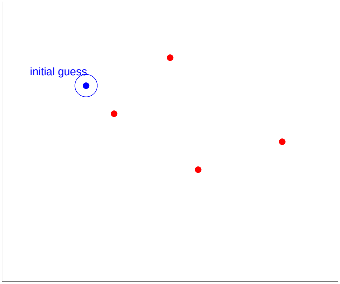
Lossy accelerator cavity: eigenvalues are in the complex plane



- choose an initial guess

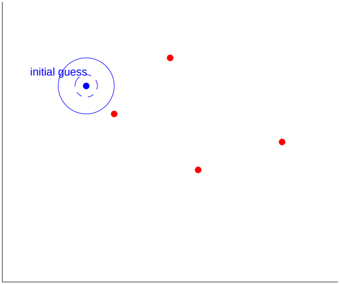
TECHNISCHE UNIVERSITÄT DARMSTADT

Lossy accelerator cavity: eigenvalues are in the complex plane



- choose an initial guess
- expand the search space ...

Lossy accelerator cavity: eigenvalues are in the complex plane



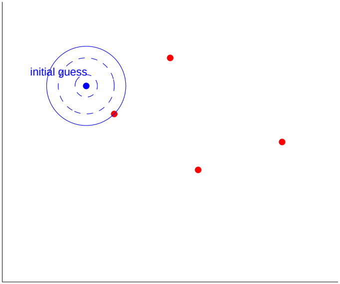
- choose an initial guess
- expand the search space ...

Motivation

Iterative methods (Jacobi-Davidson)

TECHNISCHE UNIVERSITÄT DARMSTADT

Lossy accelerator cavity: eigenvalues are in the complex plane



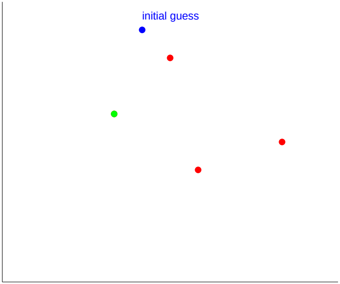
- choose an initial guess
- expand the search space ...
- until an approximate solution is found

Motivation

Iterative methods (Jacobi-Davidson)

TECHNISCHE UNIVERSITÄT DARMSTADT

Lossy accelerator cavity: eigenvalues are in the complex plane



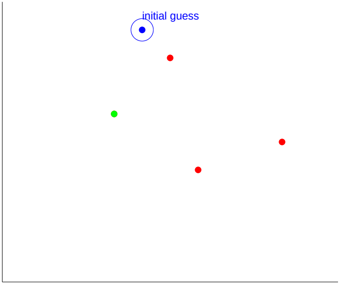
- choose an initial guess
- expand the search space ...
- until an approximate solution is found
- choose another initial guess

Motivation

Iterative methods (Jacobi-Davidson)

TECHNISCHE UNIVERSITÄT DARMSTADT

Lossy accelerator cavity: eigenvalues are in the complex plane



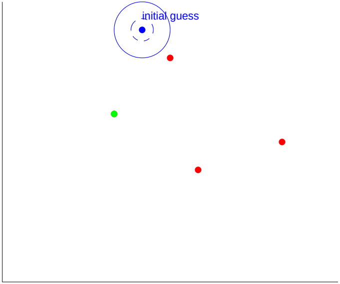
- choose an initial guess
- expand the search space ...
- until an approximate solution is found
- choose another initial guess
- continue expanding the search space ...

Motivation

Iterative methods (Jacobi-Davidson)

TECHNISCHE UNIVERSITÄT DARMSTADT

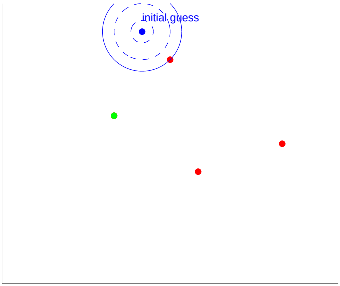
Lossy accelerator cavity: eigenvalues are in the complex plane



- choose an initial guess
- expand the search space ...
- until an approximate solution is found
- choose another initial guess
- continue expanding the search space ...

Iterative methods (Jacobi-Davidson)

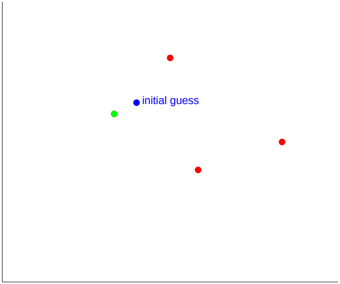
Lossy accelerator cavity: eigenvalues are in the complex plane



- choose an initial guess
- expand the search space ...
- until an approximate solution is found
- choose another initial guess
- continue expanding the search space ...
- find another approximate solution

TECHNISCHE UNIVERSITÄT DARMSTADT

Lossy accelerator cavity: eigenvalues are in the complex plane



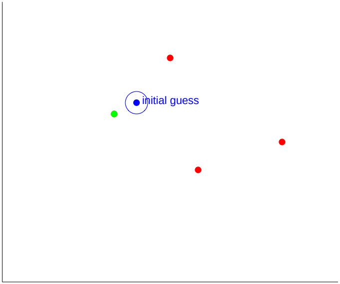
- choose an initial guess
- expand the search space ...
- until an approximate solution is found

- if we choose unsuitable initial guess

Motivation

Iterative methods (Jacobi-Davidson)

Lossy accelerator cavity: eigenvalues are in the complex plane



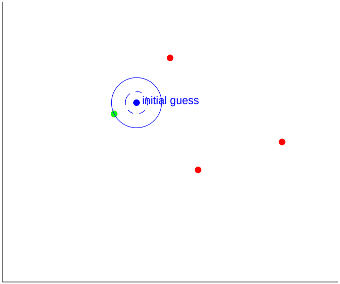
- choose an initial guess
- expand the search space ...
- until an approximate solution is found

- if we choose unsuitable initial guess
- the algorithm will converge to ...

Motivation
Iterative methods (Jacobi-Davidson)

TECHNISCHE UNIVERSITÄT DARMSTADT

Lossy accelerator cavity: eigenvalues are in the complex plane



- choose an initial guess
- expand the search space ...
- until an approximate solution is found

- if we choose unsuitable initial guess
- the algorithm will converge to ...
- a previously determined eigenvalue!!!!!

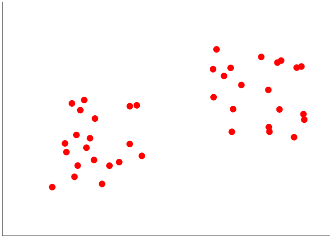
Presentation Outline

Motivation

Contour integral methods

TECHNISCHE UNIVERSITÄT DARMSTADT

An accurate computation of eigenpairs inside a region enclosed by a non-self-intersecting curve.

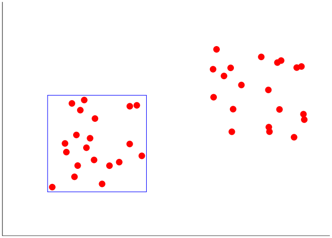


Motivation

Contour integral methods

TECHNISCHE UNIVERSITÄT DARMSTADT

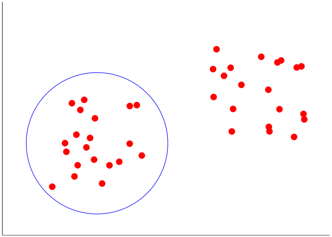
An accurate computation of eigenpairs inside a region enclosed by a non-self-intersecting curve.



- choose a region to look for eigenvalues
- the region can be of any shape, e.g rectangle ...

Motivation
Contour integral methods

An accurate computation of eigenpairs inside a region enclosed by a non-self-intersecting curve.

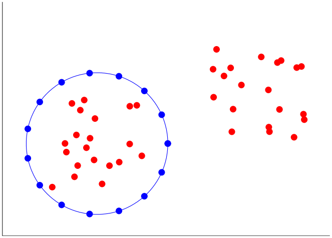


- choose a region to look for eigenvalues
- the region can be of any shape, e.g rectangle ...
- circle/ellipse

Motivation

Contour integral methods

An accurate computation of eigenpairs inside a region enclosed by a non-self-intersecting curve.



- choose a region to look for eigenvalues
- the region can be of any shape, e.g rectangle ...
- circle/ellipse
- most computation is spent to solve linear equation systems at different interpolation points which can be parallelized.

Presentation Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

Presentation Outline

Motivation
- Iterative methods
- Contour integral methods

Formulation
- Mathematical model
- Eigenvalue algorithms
- Contour integral methods
- Multigrid method as a preconditioner

Implementation

Preliminary Results

Possible Improvements

Formulation
Mathematical model

The combination of Maxwell-Ampère equation and the Maxwell-Faraday equation results in the double-curl equation

$$\nabla \times \frac{1}{\mu} \nabla \times \vec{E} - j\omega\sigma\vec{E} = \varepsilon\omega^2\vec{E} \tag{1}$$

Formulation
Mathematical model

TECHNISCHE UNIVERSITÄT DARMSTADT

The combination of Maxwell-Ampère equation and the Maxwell-Faraday equation results in the double-curl equation

$$\nabla \times \frac{1}{\mu} \nabla \times \vec{E} - j\omega\sigma\vec{E} = \varepsilon\omega^2\vec{E} \qquad (1)$$

Applying the Galerkin's approach to discretize (1) results in an eigenvalue problem

$$A^{3D}\vec{x} + j\omega\mu_0 C^{3D}\vec{x} - \omega^2\mu_0\varepsilon_0 B^{3D}\vec{x} = 0 \qquad (2)$$

TECHNISCHE UNIVERSITÄT DARMSTADT

The combination of Maxwell-Ampère equation and the Maxwell-Faraday equation results in the double-curl equation

$$\nabla \times \frac{1}{\mu} \nabla \times \vec{E} - j\omega\sigma\vec{E} = \varepsilon\omega^2\vec{E} \quad (1)$$

Applying the Galerkin's approach to discretize (1) results in an eigenvalue problem

$$A^{3D}\vec{x} + j\omega\mu_0 C^{3D}\vec{x} - \omega^2\mu_0\varepsilon_0 B^{3D}\vec{x} = 0 \quad (2)$$

which includes only losses from volumetric lossy material. Special treatment is carried out to incorporate 2D losses at port interfaces into (2), resulting in a nonlinear eigenvalue problem (NEP)

Formulation
Mathematical model

The combination of Maxwell-Ampère equation and the Maxwell-Faraday equation results in the double-curl equation

$$\nabla \times \frac{1}{\mu} \nabla \times \vec{E} - j\omega\sigma\vec{E} = \varepsilon\omega^2\vec{E} \quad (1)$$

Applying the Galerkin's approach to discretize (1) results in an eigenvalue problem

$$A^{3D}\vec{x} + j\omega\mu_0 C^{3D}\vec{x} - \omega^2\mu_0\varepsilon_0 B^{3D}\vec{x} = 0 \quad (2)$$

which includes only losses from volumetric lossy material. Special treatment is carried out to incorporate 2D losses at port interfaces into (2), resulting in a nonlinear eigenvalue problem (NEP)

$$\boxed{P(\omega)\vec{x} = 0} \quad (3)$$

Presentation Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

Motivation

- Iterative methods
- Contour integral methods

Formulation

- Mathematical model
- Eigenvalue algorithms
- Contour integral methods
- Multigrid method as a preconditioner

Implementation

Preliminary Results

Possible Improvements

Formulation
Eigenvalue algorithms

TECHNISCHE UNIVERSITÄT DARMSTADT

To solve $P(z)x = 0$ most of the standard eigenvalue algorithms exploit a projection procedure in order to extract approximate eigenvectors from a given subspace.

Formulation

Eigenvalue algorithms

TECHNISCHE UNIVERSITÄT DARMSTADT

To solve $P(z)x = 0$ most of the standard eigenvalue algorithms exploit a projection procedure in order to extract approximate eigenvectors from a given subspace.

Approximate an eigenspace Q

obtain matrix Q

Formulation

Eigenvalue algorithms

To solve $P(z)x = 0$ most of the standard eigenvalue algorithms exploit a projection procedure in order to extract approximate eigenvectors from a given subspace.

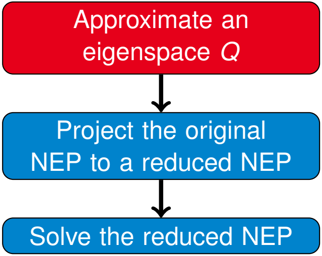
Approximate an eigenspace Q — obtain matrix Q

↓

Project the original NEP to a reduced NEP — using Rayleigh-Ritz procedure:
$P_Q(z) = Q^H P(z) Q$

TECHNISCHE UNIVERSITÄT DARMSTADT

To solve $P(z)x = 0$ most of the standard eigenvalue algorithms exploit a projection procedure in order to extract approximate eigenvectors from a given subspace.



obtain matrix Q

using Rayleigh-Ritz procedure:
$P_Q(z) = Q^H P(z) Q$

solve $P_Q(z)g = 0$
for eigenvalues z and eigenvectors g

TECHNISCHE UNIVERSITÄT DARMSTADT

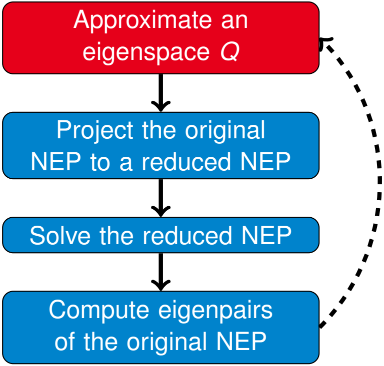
To solve $P(z)x = 0$ most of the standard eigenvalue algorithms exploit a projection procedure in order to extract approximate eigenvectors from a given subspace.

Approximate an eigenspace Q	obtain matrix Q
↓ Project the original NEP to a reduced NEP	using Rayleigh-Ritz procedure: $P_Q(z) = Q^H P(z) Q$
↓ Solve the reduced NEP	solve $P_Q(z)g = 0$ for eigenvalues z and eigenvectors g
↓ Compute eigenpairs of the original NEP	same eigenvalues as for the reduced problem; eigenvectors $x = Qg$

Formulation
Eigenvalue algorithms

TECHNISCHE UNIVERSITÄT DARMSTADT

To solve $P(z)x = 0$ most of the standard eigenvalue algorithms exploit a projection procedure in order to extract approximate eigenvectors from a given subspace.



obtain matrix Q

using Rayleigh-Ritz procedure:
$P_Q(z) = Q^H P(z) Q$

solve $P_Q(z)g = 0$
for eigenvalues z and eigenvectors g

same eigenvalues as for the reduced problem;
eigenvectors $x = Qg$

Presentation Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

Motivation
- Iterative methods
- Contour integral methods

Formulation
- Mathematical model
- Eigenvalue algorithms
- Contour integral methods
- Multigrid method as a preconditioner

Implementation

Preliminary Results

Possible Improvements

The resolvent $P(z)^{-1}$ reveals the existence of eigenvalues, indicates where eigenvalues are located, and show how sensitive these eigenvalues are to pertubation.

Formulation

Contour integral methods

Some basic spectral theory

TECHNISCHE UNIVERSITÄT DARMSTADT

The resolvent $P(z)^{-1}$ reveals the existence of eigenvalues, indicates where eigenvalues are located, and show how sensitive these eigenvalues are to pertubation.

As explained in [3], from Keldysh's theorem, we know that the resolvent function $P(z)^{-1}$ can be written (for simple eigenvalues λ_i) as

$$P(z)^{-1} = \sum_i v_i w_i^H \frac{1}{z - \lambda_i} + R(z) \tag{4}$$

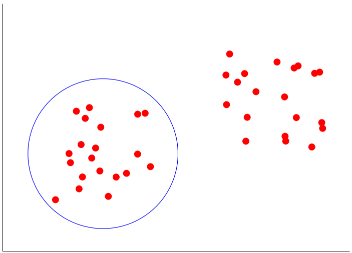
where

- v_i and w_i are suitably scaled right and left eigenvectors, respectively, corresponding to the (simple) eigenvalue λ_i
- $R(z)$ and $P(z)$ are analytic functions

Formulation

Contour integral methods

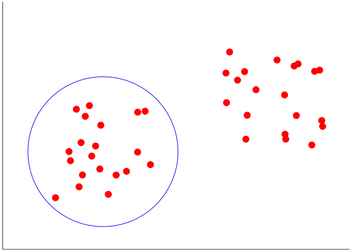
Some basic spectral theory



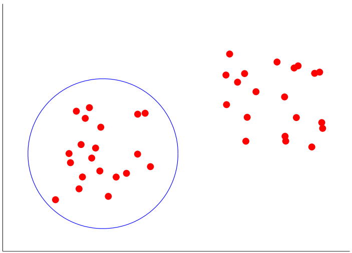
Formulation

Contour integral methods

Some basic spectral theory



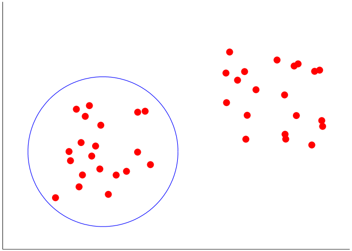
$Q = (q_1, q_2, ..., q_k)$



$Q = (q_1, q_2, ..., q_k)$

$\text{span}\{q_1, q_2, ..., q_k\} \supseteq \text{span}\{v_1, v_2, ..., v_{n(\Gamma)}\}$

$$P(z)^{-1} = \sum_i v_i w_i^H \frac{1}{z - \lambda_i} + R(z)$$



$Q = (q_1, q_2, ..., q_k)$

$\mathrm{span}\{q_1, q_2, ..., q_k\} \supseteq \mathrm{span}\{v_1, v_2, ..., v_{n(\Gamma)}\}$

Formulation

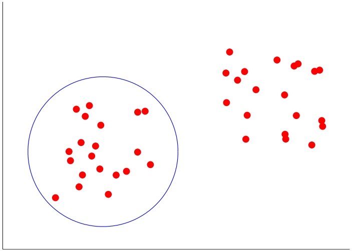
Contour integral methods

Some basic spectral theory

$$P(z)^{-1} = \sum_i v_i w_i^H \frac{1}{z - \lambda_i} + R(z)$$

Applying Cauchy's integral formula

$$\frac{1}{2\pi i} \oint_\Gamma f(z) P(z)^{-1} dz = \sum_{i=1}^{n(\Gamma)} f(\lambda_i) v_i w_i^H$$



$Q = (q_1, q_2, ..., q_k)$

$\text{span}\{q_1, q_2, ..., q_k\} \supseteq \text{span}\{v_1, v_2, ..., v_{n(\Gamma)}\}$

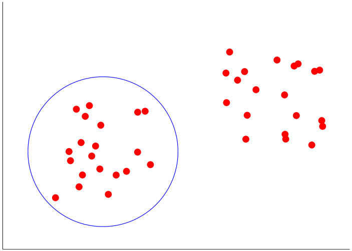
Formulation

Contour integral methods

Some basic spectral theory

TECHNISCHE UNIVERSITÄT DARMSTADT

$$P(z)^{-1} = \sum_i v_i w_i^H \frac{1}{z - \lambda_i} + R(z)$$



Applying Cauchy's integral formula

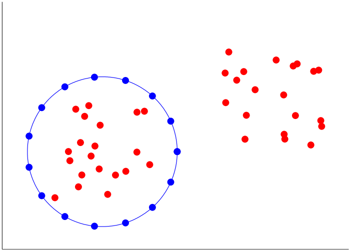
$$\frac{1}{2\pi i} \oint_\Gamma f(z) P(z)^{-1} dz = \sum_{i=1}^{n(\Gamma)} f(\lambda_i) v_i w_i^H$$

In practice, we evaluate the integral

$$\frac{1}{2\pi i} \oint_\Gamma f(z) P(z)^{-1} \hat{V} dz$$

$Q = (q_1, q_2, ..., q_k)$

$\text{span}\{q_1, q_2, ..., q_k\} \supseteq \text{span}\{v_1, v_2, ..., v_{n(\Gamma)}\}$



$$P(z)^{-1} = \sum_i v_i w_i^H \frac{1}{z - \lambda_i} + R(z)$$

Applying Cauchy's integral formula

$$\frac{1}{2\pi i} \oint_\Gamma f(z) P(z)^{-1} dz = \sum_{i=1}^{n(\Gamma)} f(\lambda_i) v_i w_i^H$$

In practice, we evaluate the integral

$$\frac{1}{2\pi i} \oint_\Gamma f(z) P(z)^{-1} \hat{V} dz$$

Using interpolation, we obtain

$$\frac{1}{2\pi i} \oint_\Gamma f(z) P(z)^{-1} \hat{V} dz = \sum_{i=1}^{n_{int}} \xi_i f(z_i) P(z_i)^{-1} \hat{V}$$

$Q = (q_1, q_2, ..., q_k)$

$\text{span}\{q_1, q_2, ..., q_k\} \supseteq \text{span}\{v_1, v_2, ..., v_{n(\Gamma)}\}$

Presentation Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

Formulation

Multigrid method as a preconditioner

The most expensive operation is to compute

$$X = P^{-1}(z_i)V \tag{5}$$

equivalent to solving the linear system

$$P(z_i)X = V \tag{6}$$

- Direct inverse becomes prohibitively expensive for large problems.

Formulation

Multigrid method as a preconditioner

The most expensive operation is to compute

$$X = P^{-1}(z_i)V \tag{5}$$

equivalent to solving the linear system

$$P(z_i)X = V \tag{6}$$

- Direct inverse becomes prohibitively expensive for large problems.
- For large-scale problems, iterative methods are preferable.

Formulation

Multigrid method as a preconditioner

The most expensive operation is to compute

$$X = P^{-1}(z_i)V \tag{5}$$

equivalent to solving the linear system

$$P(z_i)X = V \tag{6}$$

- Direct inverse becomes prohibitively expensive for large problems.
- For large-scale problems, iterative methods are preferable.
- Linear systems generated by Maxwell's equations are extremely ill-conditioned.

Formulation

Multigrid method as a preconditioner

The most expensive operation is to compute

$$X = P^{-1}(z_i)V \tag{5}$$

equivalent to solving the linear system

$$P(z_i)X = V \tag{6}$$

- Direct inverse becomes prohibitively expensive for large problems.
- For large-scale problems, iterative methods are preferable.
- Linear systems generated by Maxwell's equations are extremely ill-conditioned.
- Krylov iterative solvers with simple preconditioners often stagnate or diverge as when applied to these linear systems.

Formulation

Multigrid method as a preconditioner

The most expensive operation is to compute

$$X = P^{-1}(z_i)V \tag{5}$$

equivalent to solving the linear system

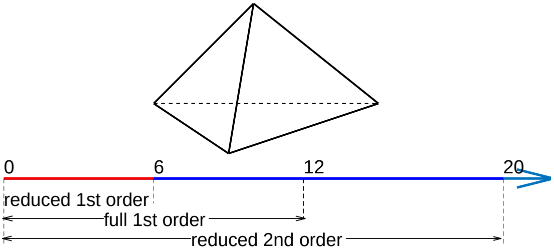
$$P(z_i)X = V \tag{6}$$

- Direct inverse becomes prohibitively expensive for large problems.
- For large-scale problems, iterative methods are preferable.
- Linear systems generated by Maxwell's equations are extremely ill-conditioned.
- Krylov iterative solvers with simple preconditioners often stagnate or diverge as when applied to these linear systems.
- Suitable preconditioners/iterative solvers should be applied to improve the convergence of the iterative solvers.

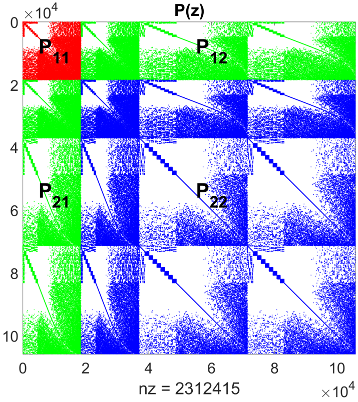
Formulation

Multigrid method as a preconditioner

TECHNISCHE UNIVERSITÄT DARMSTADT

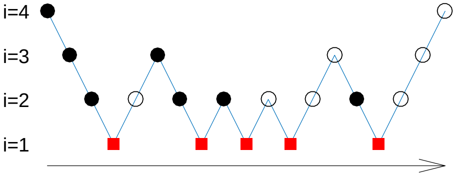


Pär Ingelström, "A New Set of H(curl) Conforming Hierarchical Basis Functions for Tetrahedral Meshes, IEEE Transactions on Microwave Theory and Techniques, vol. 54, no. 1, Jan. 2006.



Formulation

Multigrid method as a preconditioner



W-cycle

Pär Ingelström et al., "Comparison of Hierarchical Basis Functions for Efficient Multilevel Solvers", IET Science, Measurement and Technology, vol. 1, no. 1, Jan. 2007.

$$M^{-1}b = e \tag{7}$$

This equation is repeatedly computed at each iteration where M is the preconditioner, b is the input and e is the output. The output is computed by solving systems of the type

$$P_{ii}e_i = b_i - \sum_{i \neq j} P_{ij}e_j \tag{8}$$

where i and j refer to the order of the trial and test functions.

Presentation Outline

Motivation

- Iterative methods
- Contour integral methods

Formulation

- Mathematical model
- Eigenvalue algorithms
- Contour integral methods
- Multigrid method as a preconditioner

Implementation

Preliminary Results

Possible Improvements

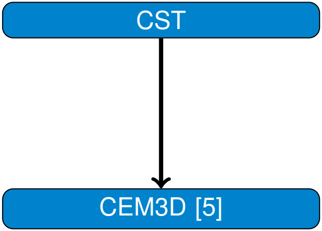
Implementation

Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)

CST — cavity design, FEM discretization

Implementation

Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)



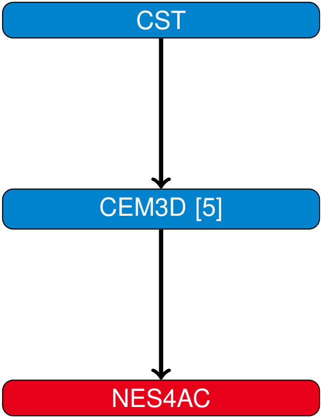
cavity design, FEM discretization

generate matrices $P(z_i)$

Implementation

Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)

TECHNISCHE UNIVERSITÄT DARMSTADT



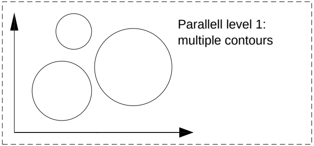
cavity design, FEM discretization

generate matrices $P(z_i)$

Solve nonlinear eigenvalue problem

Implementation

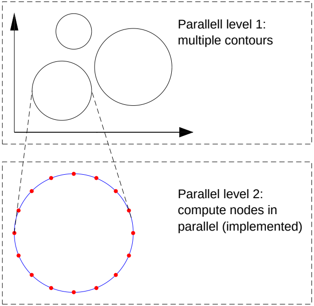
Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)



Implementation

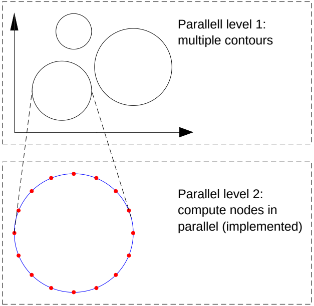
Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)

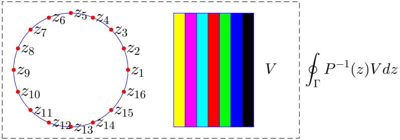
TECHNISCHE UNIVERSITÄT DARMSTADT



Implementation

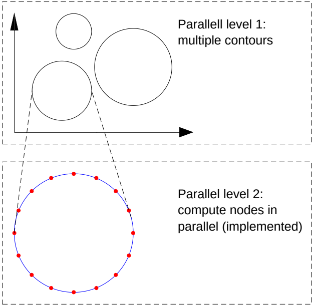
Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)

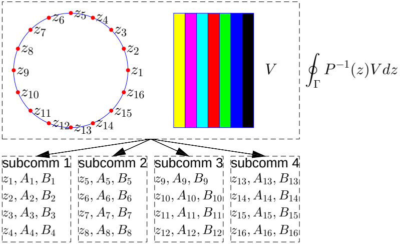




Implementation

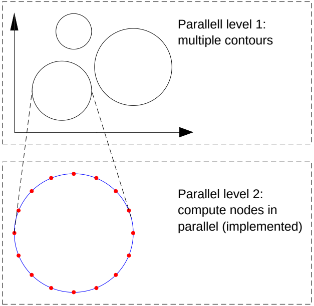
Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)

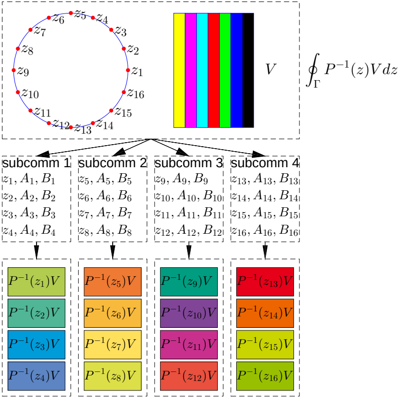




Implementation

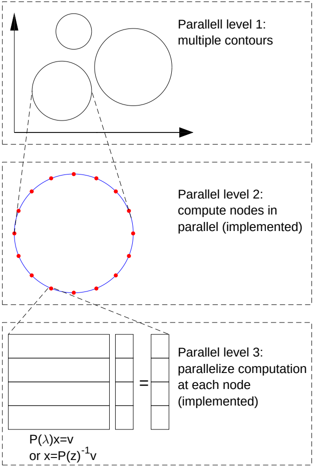
Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)

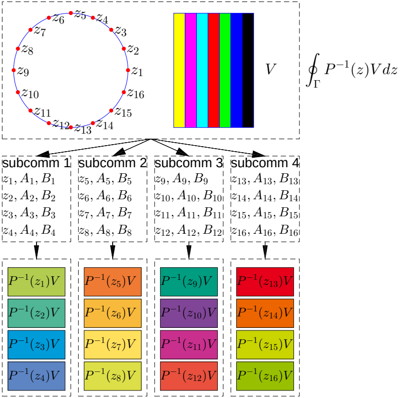




Implementation

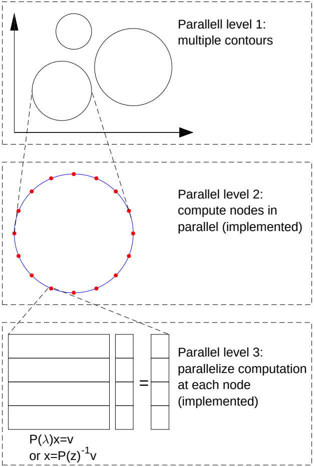
Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)

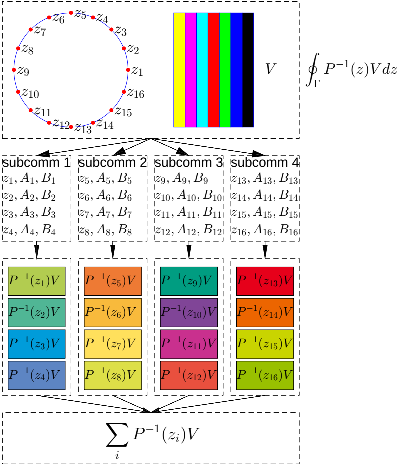




Implementation

Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)





Implementation

Nonlinear Eigenvalue Solver for Accelerator Cavities (NES4AC)

NES4AC highlights:

- extends the functionality of CEM3D [5].
- parallelized and developed in C++.
- based on PETSc (Portable, Extensible Toolkit for Scientific Computation) v3.3.0 and LAPACK.
- adopts the parallel scheme of the contour integral method from SLEPc (Scalable Library for Eigenvalue Problem Computations).
- uses the superLU_DIST for the computation of LU decompositions.
- including three contour integral algorithms for eigenvalue solution: Beyn1 (for a few eigenvalues), Beyn2 (for many eigenvalues) and RSRR.
- with two types of closed contour: ellipse and rectangle.

Presentation Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

Motivation
- Iterative methods
- Contour integral methods

Formulation
- Mathematical model
- Eigenvalue algorithms
- Contour integral methods
- Multigrid method as a preconditioner

Implementation

Preliminary Results

Possible Improvements

Preliminary Results

Nonlinear Eigenvalues Problems in [6]

TECHNISCHE UNIVERSITÄT DARMSTADT

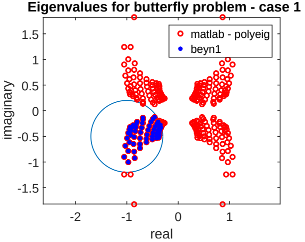
Butterfly Problem

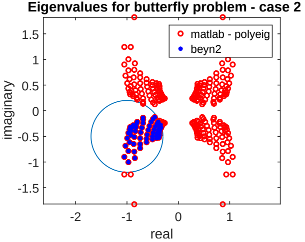
- Name: butterfly (Quartic matrix polynomial with T-even structure)
- $P(\lambda) = \lambda^4 A_4 + \lambda^3 A_3 + \lambda^2 A_2 + \lambda A_1 + A_0$
- Size: 64
- Region: circle(-1.0,-0.5,0.7)
- Number of eigenvalues: 55
- Algorithm parameters:
  - N = 60 (number of integration points)
  - L = 100:2:200 (number of columns of the random matrix)
  - K = 2 (for BEYN2)
  - Lorg = 20 (for RSRR)
  - Lred = 100 (for RSRR)
  - Nred = 30 (for RSRR)
  - Kred = 2 (for RSRR)
  - Rank tolerance = 1.0×10^{-12}

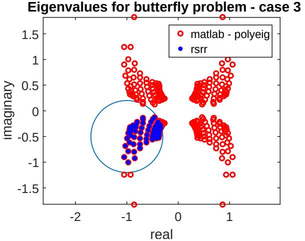
Preliminary Results

Nonlinear Eigenvalues Problems in [6]

Butterfly Problem







	beyn1	beyn2	rsrr
Min. residual	4.9×10^{-5}	2.05×10^{-14}	4.05×10^{-12}
Max. residual	0.0015	1.72×10^{-13}	4.38×10^{-9}

$$\epsilon = \frac{\|P(\lambda)x\|}{\|P(\lambda)\|\,\|x\|}$$

Preliminary Results

Nonlinear Eigenvalues Problems in [6]

Schrödinger

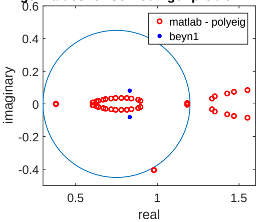
- Name: Schrödinger (QEP from Schrödinger operator)
- $P(\lambda) = K - 2\lambda C + \lambda^2 B$
- Size: 1998
- Region: circle(0.75,0.0,0.45)
- Number of eigenvalues: 30
- Algorithm parameters:
  - N = 20 (number of integration points)
  - L = 50:2:100 (number of columns of the random matrix)
  - K = 2 (for BEYN2)
  - Lorg = 20 (for RSRR)
  - Lred = 50 (for RSRR)
  - Nred = 30 (for RSRR)
  - Kred = 2 (for RSRR)
  - Rank tolerance = 1.0×10^{-12}

Preliminary Results

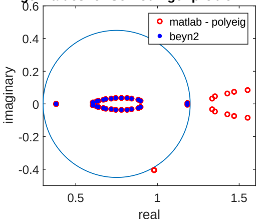
Nonlinear Eigenvalues Problems in [6]

Schrödinger

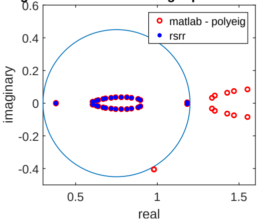
Eigenvalues for Schrödinger problem - case 1



Eigenvalues for Schrödinger problem - case 2



Eigenvalues for Schrödinger problem - case 3

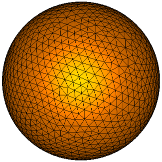


	beyn1	beyn2	rsrr
Min. residual	n/a	1.08×10^{-15}	1.97×10^{-17}
Max. residual	n/a	1.73×10^{-14}	2.52×10^{-17}

Preliminary Results

Spherical Cavity

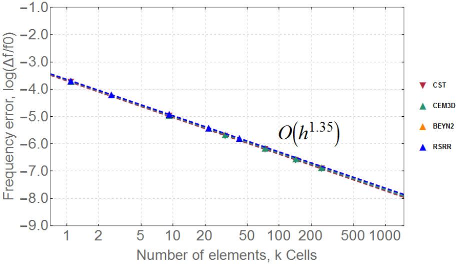
TECHNISCHE UNIVERSITÄT DARMSTADT

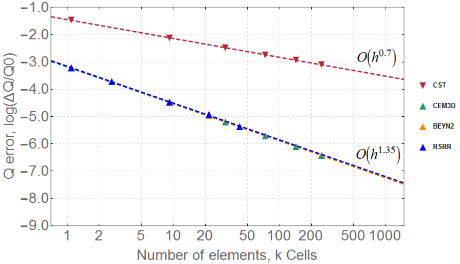


- Name: spherical cavity
- Electrical conductivity: 58×10^6 Ohm/sq
- Radius: 1m
- Size: 7614/17830/61106/143354/278138
- Target frequency: 125MHz
- Number of eigenvalues: 3
- Algorithm parameters:
  - Region: rectangle(1.0, 1.5, 1.0×10^{-15}, 0.05)
  - N = 20 (number of integration points)
  - L = 40:10:100 (number of columns of the random matrix)
  - K = 2 (for BEYN2)
  - Lorg = 20 / Lred = 20 / Nred = 20 / Kred = 2 (for RSRR)
  - Rank tolerance = 1.0×10^{-8}

Preliminary Results

Spherical Cavity



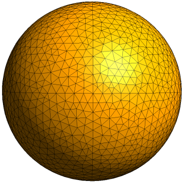


$$\epsilon_f = \frac{\| f - f_{analytical} \|}{\| f_{analytical} \|}$$

$$\epsilon_Q = \frac{\| Q - Q_{analytical} \|}{\| Q_{analytical} \|}$$

Preliminary Results

Multigrid preconditioner - Spherical Cavity



	# elements	# dofs	# dofs (1st order)
Dis. 1	342	1,652	250
Dis. 2	1,256	6,640	1,078
Dis. 3	2,918	16,300	2,755
Dis. 4	18,062	105,828	18,487

Table: Mesh refinement of a spherical cavity

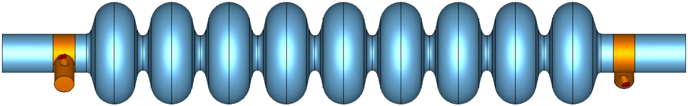
	Dis. 1	Dis. 2	Dis. 3	Dis. 4
mg-GMRES	17	21	23	23
GMRES	500 (1.0e-5)	500 (1.0e-2)	500 (1.0e-2)	500 (5.0e-2)

Table: Number of iterations required to achieve a residual of 1.0e-8

Preliminary Results

Multigrid preconditioner - Tesla Cavity

TECHNISCHE
UNIVERSITÄT
DARMSTADT



	# elements	# dofs	# dofs (1st order)
Dis. 1	48,819	286,026	50,067
Dis. 2	57,394	339,664	59,864
Dis. 3	76,113	373,812	80,552

Table: Mesh refinement of a Tesla cavity

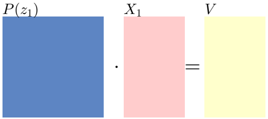
	Dis. 1	Dis. 2	Dis. 3
mg-GMRES	30	25	24
GMRES	500 (0.2)	500 (0.15)	500 (0.2)

Table: Number of iterations required to achieve a residual of 1.0e-8

Presentation Outline

Possible Improvements
Recycling Krylov subspace methods

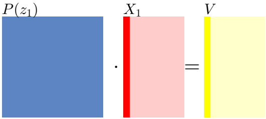
$$X_i = P^{-1}(z_i)V$$
$$P(z_i)X_i = V$$



Possible Improvements
Recycling Krylov subspace methods

$X_i = P^{-1}(z_i)V$

$P(z_i)X_i = V$



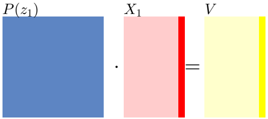
- Iterative method is repeatedly applied for different RHS.

Possible Improvements

Recycling Krylov subspace methods

$$X_i = P^{-1}(z_i)V$$

$$P(z_i)X_i = V$$

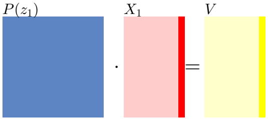


- Iterative method is repeatedly applied for different RHS.
- The matrix is unchanged.

Possible Improvements

Recycling Krylov subspace methods

$$X_i = P^{-1}(z_i)V$$
$$P(z_i)X_i = V$$



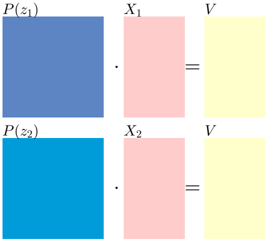
- Iterative method is repeatedly applied for different RHS.
- The matrix is unchanged.
- Opportunity for applying recycling Krylov subspace methods or block Krylov subspace methods.

Possible Improvements

Recycling Krylov subspace methods

TECHNISCHE UNIVERSITÄT DARMSTADT

$$X_i = P^{-1}(z_i)V$$
$$P(z_i)X_i = V$$



- Iterative method is repeatedly applied for different RHS.
- The matrix is unchanged.
- Opportunity for applying recycling Krylov subspace methods or block Krylov subspace methods.
- The system-matrices are slightly changed for different interpolation points

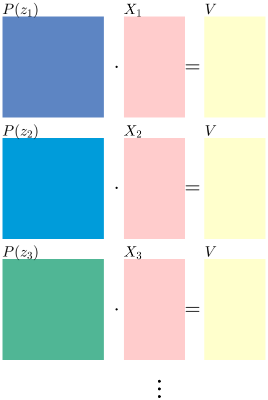
Possible Improvements

Recycling Krylov subspace methods

TECHNISCHE UNIVERSITÄT DARMSTADT

$$X_i = P^{-1}(z_i)V$$

$$P(z_i)X_i = V$$



- Iterative method is repeatedly applied for different RHS.
- The matrix is unchanged.
- Opportunity for applying recycling Krylov subspace methods or block Krylov subspace methods.
- The system-matrices are slightly changed for different interpolation points
- Opportunity for applying recycling Krylov subspace methods or block Krylov subspace methods.

TECHNISCHE UNIVERSITÄT DARMSTADT

Thank you for your attention

References I

[1] T. Flisgen, J. Heller, T. Galek, L. Shi, N. Joshi, N. Baboi, R. M. Jones, and U. van Rienen, “Eigenmode Compendium of The Third Harmonic Module of the European X-Ray Free Electron Laser,” *Physical Review Accelerators and Beams*, vol. 20, p. 042002, Apr 2017.

[2] H. Voss, “A Jacobi-Davidson Method for Nonlinear and Nonsymmetric Eigenproblems,” *Computers and Structures*, vol. 85, no. 17-18, pp. 1284–1292, 2007.

[3] W.-J. Beyn, “An Integral Method for Solving Nonlinear Eigenvalue Problems,” *Linear Algebra and its Applications*, vol. 436, no. 10, pp. 3839 – 3863, 2012.

References II

[4] J. Xiao, C. Zhang, T. M. Huang, and T. Sakurai, "Solving Large-Scale Nonlinear Eigenvalue Problems by Rational Interpolation and Resolvent Sampling Based Rayleigh-Ritz Method," *International Journal for Numerical Methods in Engineering*, vol. 110, no. 8, pp. 776–800, 2017.

[5] T. Banova, W. Ackermann, and T. Weiland, "Accurate Determination of Thousands of Eigenvalues for Large-Scale Eigenvalue Problems," *IEEE Transactions on Magnetics*, vol. 50, pp. 481–484, Feb. 2014.

[6] T. Betcke, N. J. Higham, V. Mehrmann, C. Schröder, and F. Tisseur, "NLEVP: A Collection of Nonlinear Eigenvalue Problems," *ACM Transactions on Mathematical Software*, vol. 39, no. 2, pp. 7:1–7:28, 2013.

TECHNISCHE UNIVERSITÄT DARMSTADT

Define the matrices A_0 and $A_1 \in \mathbb{C}^{n\times k}$

$$A_0 = \frac{1}{2\pi i}\oint_\Gamma P(z)^{-1}\hat{V}dz \qquad (9)$$

$$A_1 = \frac{1}{2\pi i}\oint_\Gamma zP(z)^{-1}\hat{V}dz \qquad (10)$$

Then $A_0 = VW^H\hat{V}$ and $A_1 = V\Lambda W^H\hat{V}$ where

- $\Lambda = \text{diag}(\lambda_1, ..., \lambda_{n(\Gamma)})$
- $V = \begin{bmatrix} v_1 & \cdots & v_{n(\Gamma)} \end{bmatrix}$
- $W = \begin{bmatrix} w_1 & \cdots & w_{n(\Gamma)} \end{bmatrix}$

$\hat{V}$ is a random matrix $\hat{V} \in \mathbb{C}^{n\times L}$. L is smaller than n and equal or greater and k

TECHNISCHE UNIVERSITÄT DARMSTADT

Beyn's method is based on the singular value decomposition of A_0

$$A_0 = V_0 \Sigma_0 W_0^H \tag{11}$$

Beyn has shown that the matrix

$$B = V_0^H A_1 W_0^H \Sigma_0^{-1} \tag{12}$$

is diagonalizable. Its eigenvalues are the eigenvalues of P inside the contour and its eigenvectors lead to the corresponding eigenvectors of P.

Define the matrices $A_p \in \mathbb{C}^{n\times k}$

$$A_p = \frac{1}{2\pi i} \oint_{\Gamma} z^p P(z)^{-1} \hat{V} dz \qquad (13)$$

Then $A_p = V\Lambda^p W^H \hat{V}$. The matrices B_0 and B_1 are defined as follows

$$B_0 = \begin{pmatrix} A_0 & \cdots & A_{K-1} \\ \vdots & & \vdots \\ A_{K-1} & \cdots & A_{2K-2} \end{pmatrix} \quad ; \quad B_1 = \begin{pmatrix} A_1 & \cdots & A_K \\ \vdots & & \vdots \\ A_K & \cdots & A_{2K-1} \end{pmatrix} \qquad (14)$$

Performing the singular value decomposition of B_0

$$B_0 = V_0 \Sigma_0 W_0^H \tag{15}$$

Beyn has shown that the matrix

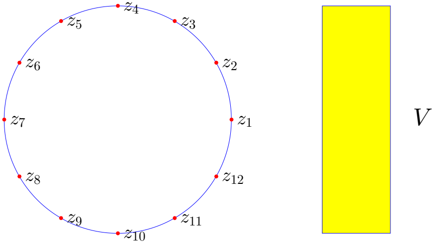
$$D = V_0^H B_1 W_0^H \Sigma_0^{-1} \tag{16}$$

is diagonalizable. Its eigenvalues are the eigenvalues of P inside the contour and its eigenvectors lead to the corresponding eigenvectors of P.

Appendix

Contour integral methods

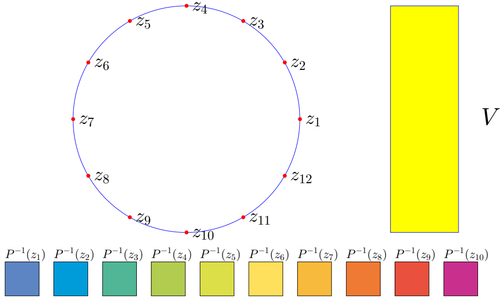
Resolvent Sampling based Rayleigh-Ritz method



Appendix

Contour integral methods

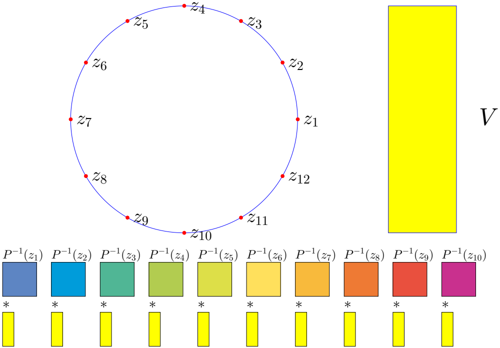
Resolvent Sampling based Rayleigh-Ritz method



Appendix

Contour integral methods

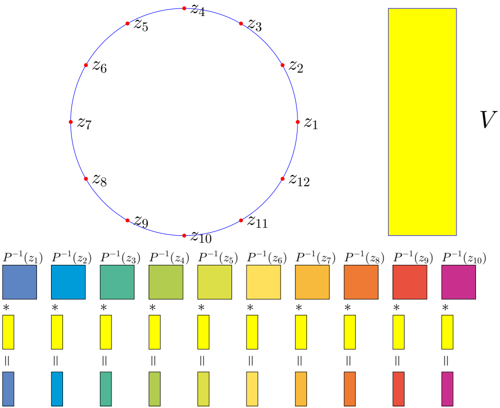
Resolvent Sampling based Rayleigh-Ritz method



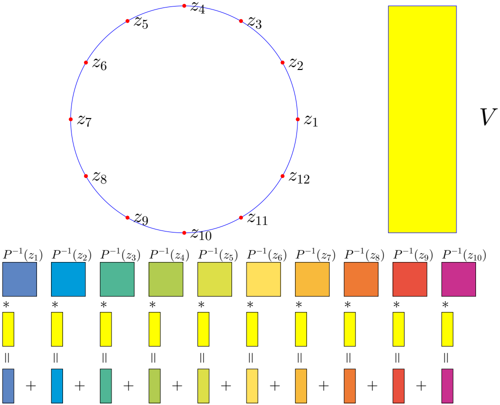
Appendix

Contour integral methods

Resolvent Sampling based Rayleigh-Ritz method



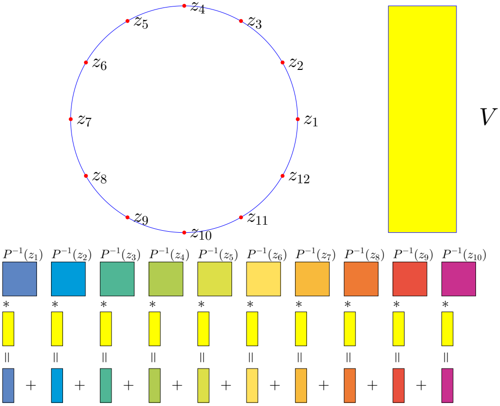
Appendix
Contour integral methods
Resolvent Sampling based Rayleigh-Ritz method



Appendix

Contour integral methods

Resolvent Sampling based Rayleigh-Ritz method

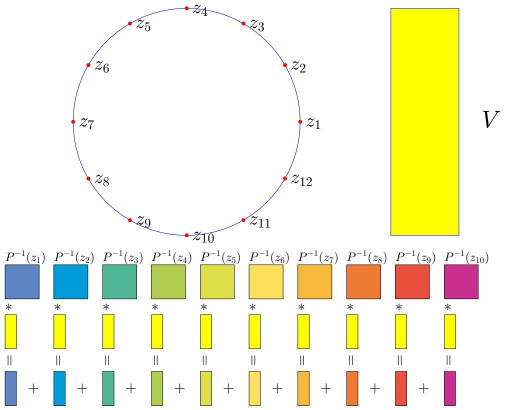


The Beyn2 algorithm is robust and accurate if a large L but a small K are used.

However, for large-scale problems, a small L is essential to reduce the computational burden.

TECHNISCHE UNIVERSITÄT DARMSTADT

Resolvent Sampling based Rayleigh-Ritz method



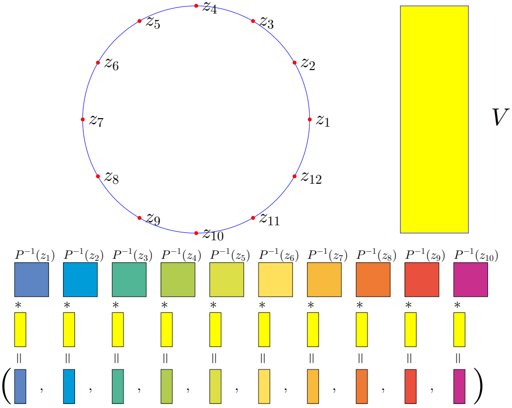
The Beyn2 algorithm is robust and accurate if a large L but a small K are used.

However, for large-scale problems, a small L is essential to reduce the computational burden.

Decrease L and increase K make the algorithm unstable and inaccurate.

TECHNISCHE UNIVERSITÄT DARMSTADT

Resolvent Sampling based Rayleigh-Ritz method



The Beyn2 algorithm is robust and accurate if a large L but a small K are used.

However, for large-scale problems, a small L is essential to reduce the computational burden.

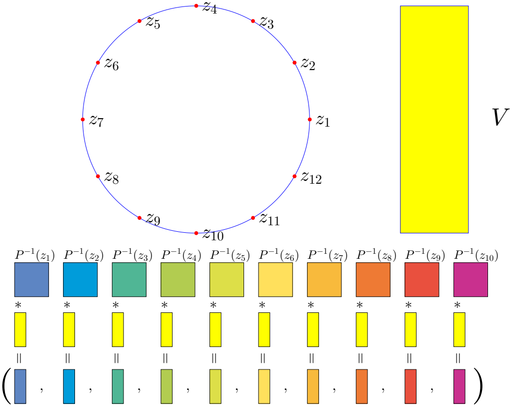
Decrease L and increase K make the algorithm unstable and inaccurate.

Appendix

Contour integral methods

Resolvent Sampling based Rayleigh-Ritz method

TECHNISCHE
UNIVERSITÄT
DARMSTADT



The Beyn2 algorithm is robust and accurate if a large L but a small K are used.

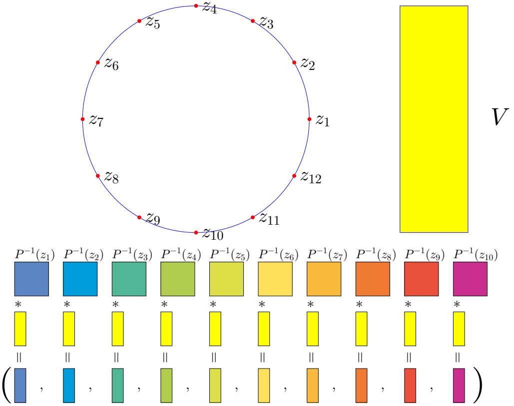
However, for large-scale problems, a small L is essential to reduce the computational burden.

Decrease L and increase K make the algorithm unstable and inaccurate.

RSRR reduce the number of columns of V.

TECHNISCHE UNIVERSITÄT DARMSTADT

Resolvent Sampling based Rayleigh-Ritz method



The Beyn2 algorithm is robust and accurate if a large L but a small K are used.

However, for large-scale problems, a small L is essential to reduce the computational burden.

Decrease L and increase K make the algorithm unstable and inaccurate.

RSRR reduce the number of columns of V.

Let $Q \in \mathbb{C}^{n\times k}$ be an orthogonal basis of search space, then the original NEP can be converted to the following reduced NEP

$$P_Q(z)g = 0$$

TECHNISCHE UNIVERSITÄT DARMSTADT

(1) Initialization: Fix the contour Γ, the number N and the sampling points z_i. Fix the number L and generate a $n \times L$ random matrix U

TECHNISCHE UNIVERSITÄT DARMSTADT

(1) Initialization: Fix the contour Γ, the number N and the sampling points z_i. Fix the number L and generate a $n \times L$ random matrix U

(2) Compute $P(z_i)^{-1}U$ for $i = 0, 1, \ldots, N-1$

Appendix

Contour integral methods

Resolvent Sampling based Rayleigh-Ritz method

(1) Initialization: Fix the contour Γ, the number N and the sampling points z_i. Fix the number L and generate a $n \times L$ random matrix U

(2) Compute $P(z_i)^{-1}U$ for $i = 0, 1, \ldots, N-1$

(3) Form S as follows

Appendix

Contour integral methods

Resolvent Sampling based Rayleigh-Ritz method

(1) Initialization: Fix the contour Γ, the number N and the sampling points z_i. Fix the number L and generate a $n \times L$ random matrix U

(2) Compute $P(z_i)^{-1}U$ for $i = 0, 1, \dots, N-1$

(3) Form S as follows

$$S = \left[P(z_0)^{-1}U, \quad P(z_1)^{-1}U, \quad \cdots, \quad P(z_{N-1})^{-1}U\right] \in \mathbb{C}^{n \times N \cdot L} \tag{17}$$

Appendix

Contour integral methods

Resolvent Sampling based Rayleigh-Ritz method

TECHNISCHE UNIVERSITÄT DARMSTADT

(1) Initialization: Fix the contour Γ, the number N and the sampling points z_i. Fix the number L and generate a $n \times L$ random matrix U

(2) Compute $P(z_i)^{-1}U$ for $i = 0, 1, \dots, N-1$

(3) Form S as follows

$$S = \left[P(z_0)^{-1}U, \quad P(z_1)^{-1}U, \quad \cdots, \quad P(z_{N-1})^{-1}U\right] \in \mathbb{C}^{n \times N \cdot L} \tag{17}$$

(4) Generate the matrix Q via the truncated singular value decomposition $S \approx Q \Sigma V^H$.

Appendix

Contour integral methods

Resolvent Sampling based Rayleigh-Ritz method

(1) Initialization: Fix the contour Γ, the number N and the sampling points z_i. Fix the number L and generate a $n \times L$ random matrix U

(2) Compute $P(z_i)^{-1}U$ for $i = 0, 1, ..., N-1$

(3) Form S as follows

$$S = \left[P(z_0)^{-1}U, \quad P(z_1)^{-1}U, \quad \cdots, \quad P(z_{N-1})^{-1}U\right] \in \mathbb{C}^{n\times N\cdot L} \tag{17}$$

(4) Generate the matrix Q via the truncated singular value decomposition $S \approx Q\Sigma V^H$.

(5) Compute $P_Q(z) = Q^H P(z) Q$, and solve the projected NEP $P_Q(\lambda)g = 0$ using the SS-FULL algorithm to obtain $n(\Gamma)$ eigenapairs (g_j, λ_j).

(1) Initialization: Fix the contour Γ, the number N and the sampling points z_i. Fix the number L and generate a $n \times L$ random matrix U
(2) Compute $P(z_i)^{-1}U$ for $i = 0, 1, \dots, N-1$
(3) Form S as follows

$$S = \left[P(z_0)^{-1}U, \quad P(z_1)^{-1}U, \quad \cdots, \quad P(z_{N-1})^{-1}U\right] \in \mathbb{C}^{n \times N \cdot L} \tag{17}$$

(4) Generate the matrix Q via the truncated singular value decomposition $S \approx Q\Sigma V^H$.
(5) Compute $P_Q(z) = Q^H P(z) Q$, and solve the projected NEP $P_Q(\lambda)g = 0$ using the SS-FULL algorithm to obtain $n(\Gamma)$ eigenapairs (g_j, λ_j).
(6) Compute the eigenpairs of the original NEP via the eigenpairs of the reduced NEP.