Abstract Perturbed Krylov Methods

Jens-Peter M. Zemke
zemke@tu-harburg.de

Institut für Numerische Simulation
Technische Universität Hamburg-Harburg

30.03.2006

Outline

Outline

Philosophical considerations

- A matrix equation
- The iterative point of view
- The polynomial point of view

The results on ...

- Ritz vectors
- QOR iterates
- QOR residuals

Outline

Philosophical considerations
- A matrix equation
- The iterative point of view
- The polynomial point of view

The results on ...
- Ritz vectors
- QOR iterates
- QOR residuals

A "numerical" experiment
- Eigenvectors using Lanczos' method

A matrix-theoretical beginning

We start with the equation

$$\begin{aligned} AQ_k &= Q_{k+1}\underline{C}_k \\ &= Q_k C_k + q_{k+1} c_{k+1,k} e_k^T. \end{aligned} \tag{1}$$

A matrix-theoretical beginning

We start with the equation

$$\begin{aligned} AQ_k &= Q_{k+1}\underline{C}_k \\ &= Q_k C_k + q_{k+1} c_{k+1,k} e_k^T. \end{aligned} \tag{1}$$

This equation is known as Krylov decomposition.

A matrix-theoretical beginning

We start with the equation

$$\begin{aligned} AQ_k + F_k &= Q_{k+1}\underline{C}_k \\ &= Q_k C_k + q_{k+1} c_{k+1,k} e_k^T. \end{aligned} \tag{1}$$

This equation is known as perturbed Krylov decomposition.

A matrix-theoretical beginning

We start with the equation

$$\begin{aligned} AQ_k + F_k &= Q_{k+1}\underline{C}_k \\ &= Q_k C_k + q_{k+1} c_{k+1,k} e_k^T. \end{aligned} \tag{1}$$

This equation is known as perturbed Krylov decomposition.

Here, we suppose that

A matrix-theoretical beginning

We start with the equation

$$\begin{aligned} AQ_k + F_k &= Q_{k+1}\underline{C}_k \\ &= Q_k C_k + q_{k+1} c_{k+1,k} e_k^T. \end{aligned} \tag{1}$$

This equation is known as perturbed Krylov decomposition.

Here, we suppose that

$A \in \mathbb{C}^{(n,n)}$ is a general square matrix,

A matrix-theoretical beginning

We start with the equation

$$\begin{aligned} AQ_k + F_k &= Q_{k+1}\underline{C}_k \\ &= Q_k C_k + q_{k+1} c_{k+1,k} e_k^T. \end{aligned} \tag{1}$$

This equation is known as perturbed Krylov decomposition.

Here, we suppose that

$$A \in \mathbb{C}^{(n,n)}$$ is a general square matrix,

$$Q_{k+1} = \begin{pmatrix} Q_k & q_{k+1} \end{pmatrix} \in \mathbb{C}^{(n,k+1)}$$ is a matrix of “basis” vectors,

A matrix-theoretical beginning

We start with the equation

$$\begin{aligned} AQ_k + F_k &= Q_{k+1}\underline{C}_k \\ &= Q_k C_k + q_{k+1} c_{k+1,k} e_k^T. \end{aligned} \qquad (1)$$

This equation is known as perturbed Krylov decomposition.

Here, we suppose that

$A \in \mathbb{C}^{(n,n)}$	is a general square matrix,
$Q_{k+1} = \begin{pmatrix} Q_k & q_{k+1} \end{pmatrix} \in \mathbb{C}^{(n,k+1)}$	is a matrix of "basis" vectors,
$\underline{C}_k = \begin{pmatrix} C_k \\ c_{k+1,k} e_k^T \end{pmatrix} \in \mathbb{C}^{(k+1,k)}$	is an extended Hessenberg matrix,

A matrix-theoretical beginning

We start with the equation

$$\begin{aligned} AQ_k + F_k &= Q_{k+1}\underline{C}_k \\ &= Q_k C_k + q_{k+1} c_{k+1,k} e_k^T. \end{aligned} \tag{1}$$

This equation is known as perturbed Krylov decomposition.

Here, we suppose that

$A \in \mathbb{C}^{(n,n)}$	is a general square matrix,
$Q_{k+1} = \begin{pmatrix} Q_k & q_{k+1} \end{pmatrix} \in \mathbb{C}^{(n,k+1)}$	is a matrix of “basis” vectors,
$\underline{C}_k = \begin{pmatrix} C_k \\ c_{k+1,k} e_k^T \end{pmatrix} \in \mathbb{C}^{(k+1,k)}$	is an extended Hessenberg matrix,
$F_k \in \mathbb{C}^{(n,k)}$	is a general perturbation.

The dependence on the iteration

We investigate this matrix equation iteratively:

$$\begin{aligned} AQ_k e_l + F_k e_l &= Q_{k+1} \underline{C}_k e_l \\ &= Q_k C_k e_l + q_{k+1} c_{k+1,k} \delta_{kl}, \qquad \forall l \leqslant k. \end{aligned} \tag{2}$$

The dependence on the iteration

We investigate this matrix equation iteratively:

$$\begin{aligned} AQ_k e_l + F_k e_l &= Q_{k+1} \underline{C}_k e_l \\ &= Q_k C_k e_l + q_{k+1} c_{k+1,k} \delta_{kl}, \qquad \forall l \leqslant k. \end{aligned} \tag{2}$$

Observations:

The dependence on the iteration

We investigate this matrix equation iteratively:

$$\begin{aligned} AQ_k e_l + F_k e_l &= Q_{k+1} \underline{C}_k e_l \\ &= Q_k C_k e_l + q_{k+1} c_{k+1,k} \delta_{kl}, \qquad \forall l \leqslant k. \end{aligned} \tag{2}$$

Observations:

- The perturbations $\{f_l = F_k e_l\}_{l=1}^k$ enter stage after step l.

The dependence on the iteration

We investigate this matrix equation iteratively:

$$\begin{aligned} AQ_k e_l + F_k e_l &= Q_{k+1}\underline{C}_k e_l \\ &= Q_k C_k e_l + q_{k+1} c_{k+1,k}\delta_{kl}, \qquad \forall l \leqslant k. \end{aligned} \tag{2}$$

Observations:

- The perturbations $\{f_l = F_k e_l\}_{l=1}^{k}$ enter stage after step l.
- Without perturbations, the Hessenberg structure results in polynomial dependence of the basis vectors $\{q_j = Q_k e_j\}_{j=1}^{k}$ from A.

The dependence on the iteration

We investigate this matrix equation iteratively:

$$\begin{aligned} AQ_k e_l + F_k e_l &= Q_{k+1} \underline{C}_k e_l \\ &= Q_k C_k e_l + q_{k+1} c_{k+1,k} \delta_{kl}, \qquad \forall l \leqslant k. \end{aligned} \tag{2}$$

Observations:

- The perturbations $\{f_l = F_k e_l\}_{l=1}^k$ enter stage after step l.
- Without perturbations, the Hessenberg structure results in polynomial dependence of the basis vectors $\{q_j = Q_k e_j\}_{j=1}^k$ from A.
- We are concerned with quantities from Krylov subspaces. These are represented as linear combinations $Q_k z$ for some $z \in \mathbb{C}^k$.

The dependence on the iteration

We investigate this matrix equation iteratively:

$$\begin{aligned} AQ_k e_l + F_k e_l &= Q_{k+1} \underline{C}_k e_l \\ &= Q_k C_k e_l + q_{k+1} c_{k+1,k} \delta_{kl}, \qquad \forall l \leqslant k. \end{aligned} \tag{2}$$

Observations:

- The perturbations $\{f_l = F_k e_l\}_{l=1}^{k}$ enter stage after step l.
- Without perturbations, the Hessenberg structure results in polynomial dependence of the basis vectors $\{q_j = Q_k e_j\}_{j=1}^{k}$ from A.
- We are concerned with quantities from Krylov subspaces. These are represented as linear combinations $Q_k z$ for some $z \in \mathbb{C}^k$.

Idea:

The dependence on the iteration

We investigate this matrix equation iteratively:

$$\begin{aligned} AQ_k e_l + F_k e_l &= Q_{k+1} \underline{C}_k e_l \\ &= Q_k C_k e_l + q_{k+1} c_{k+1,k} \delta_{kl}, \qquad \forall l \leqslant k. \end{aligned} \tag{2}$$

Observations:

- The perturbations $\{f_l = F_k e_l\}_{l=1}^k$ enter stage after step l.
- Without perturbations, the Hessenberg structure results in polynomial dependence of the basis vectors $\{q_j = Q_k e_j\}_{j=1}^k$ from A.
- We are concerned with quantities from Krylov subspaces. These are represented as linear combinations $Q_k z$ for some $z \in \mathbb{C}^k$.

Idea:

- Interpret perturbed Krylov methods as overlay of several polynomial methods.

Introducing: polynomials

We study polynomials based on the computed C_k or $\underline{C}_k$ with certain useful properties.

Introducing: polynomials

We study polynomials based on the computed C_k or $\underline{C}_k$ with certain useful properties.

These polynomials are named by their property.

Introducing: polynomials

We study polynomials based on the computed C_k or $\underline{C}_k$ with certain useful properties.

These polynomials are named by their property.

In [Z, 2005] we considered the following five types of polynomials:

Introducing: polynomials

We study polynomials based on the computed C_k or $\underline{C}_k$ with certain useful properties.

These polynomials are named by their property.

In [Z, 2005] we considered the following five types of polynomials:

- basis polynomials $\mathcal{B}_k$,

Introducing: polynomials

We study polynomials based on the computed C_k or $\underline{C}_k$ with certain useful properties.

These polynomials are named by their property.

In [Z, 2005] we considered the following five types of polynomials:

- basis polynomials $\mathcal{B}_k$,
- adjugate polynomials $\mathcal{A}_k$,

Introducing: polynomials

We study polynomials based on the computed C_k or $\underline{C}_k$ with certain useful properties.

These polynomials are named by their property.

In [Z, 2005] we considered the following five types of polynomials:

- basis polynomials $\mathcal{B}_k$,
- adjugate polynomials $\mathcal{A}_k$,
- Lagrange polynomials $\mathcal{L}_k[z^{-1}]$ and $\underline{\mathcal{L}}_k[z^{-1}]$,

Introducing: polynomials

We study polynomials based on the computed C_k or $\underline{C}_k$ with certain useful properties.

These polynomials are named by their property.

In [Z, 2005] we considered the following five types of polynomials:

- basis polynomials $\mathcal{B}_k$,
- adjugate polynomials $\mathcal{A}_k$,
- Lagrange polynomials $\mathcal{L}_k[z^{-1}]$ and $\underline{\mathcal{L}}_k[z^{-1}]$,
- Lagrange polynomials $\mathcal{L}_k[1-\delta_{z0}]$ and $\underline{\mathcal{L}}_k[1-\delta_{z0}]$,

Introducing: polynomials

We study polynomials based on the computed C_k or $\underline{C}_k$ with certain useful properties.

These polynomials are named by their property.

In [Z, 2005] we considered the following five types of polynomials:

- basis polynomials $\mathcal{B}_k$,
- adjugate polynomials $\mathcal{A}_k$,
- Lagrange polynomials $\mathcal{L}_k[z^{-1}]$ and $\underline{\mathcal{L}}_k[z^{-1}]$,
- Lagrange polynomials $\mathcal{L}_k[1 - \delta_{z0}]$ and $\underline{\mathcal{L}}_k[1 - \delta_{z0}]$,
- residual polynomials $\mathcal{R}_k$ and $\underline{\mathcal{R}}_k$.

Introducing: polynomials

We study polynomials based on the computed C_k or $\underline{C}_k$ with certain useful properties.

These polynomials are named by their property.

In [Z, 2005] we considered the following five types of polynomials:

- basis polynomials $\mathcal{B}_k$,
- adjugate polynomials $\mathcal{A}_k$,
- Lagrange polynomials $\mathcal{L}_k[z^{-1}]$ and $\underline{\mathcal{L}}_k[z^{-1}]$,
- Lagrange polynomials $\mathcal{L}_k[1-\delta_{z0}]$ and $\underline{\mathcal{L}}_k[1-\delta_{z0}]$,
- residual polynomials $\mathcal{R}_k$ and $\underline{\mathcal{R}}_k$.

We restrict ourselves to $\mathcal{A}_k$, $\mathcal{L}_k[z^{-1}]$, $\mathcal{L}_k[1-\delta_{z0}]$ and $\mathcal{R}_k$.

Adjugate polynomials

First we consider certain bivariate polynomials – the adjugate polynomials.

Adjugate polynomials

First we consider certain bivariate polynomials – the adjugate polynomials.

- Property:

$$\mathcal{A}_k(z, C_k) = \operatorname{adj}(zI_k - C_k).$$

Adjugate polynomials

First we consider certain bivariate polynomials – the adjugate polynomials.

- Property:

$$\mathcal{A}_k(z, C_k) = \operatorname{adj}(zI_k - C_k).$$

- This implies [Z, 2006]

$$\mathcal{A}_k(\theta_j, C_k)e_1 = s_j, \qquad C_k s_j = \theta_j s_j$$

for all eigenvalues (Ritz values) θ_j of C_k.

Adjugate polynomials

First we consider certain bivariate polynomials – the adjugate polynomials.

- Property:

$$\mathcal{A}_k(z, C_k) = \operatorname{adj}(zI_k - C_k).$$

- This implies [Z, 2006]

$$\mathcal{A}_k(\theta_j, C_k)e_1 = s_j, \qquad C_k s_j = \theta_j s_j$$

  for all eigenvalues (Ritz values) θ_j of C_k.

- Definition:

$$\mathcal{A}_k(\theta, z) \equiv \frac{\chi_k(\theta) - \chi_k(z)}{\theta - z}.$$

Adjugate polynomials

First we consider certain bivariate polynomials – the adjugate polynomials.

- Property:

$$\mathcal{A}_k(z, C_k) = \operatorname{adj}(zI_k - C_k).$$

- This implies [Z, 2006]

$$\mathcal{A}_k(\theta_j, C_k)e_1 = s_j, \qquad C_k s_j = \theta_j s_j$$

  for all eigenvalues (Ritz values) θ_j of C_k.

- Definition:

$$\mathcal{A}_k(\theta, z) \equiv \frac{\chi_k(\theta) - \chi_k(z)}{\theta - z}.$$

- Generalization:

$$\mathcal{A}_{l+1:k}(\theta, z) \equiv \frac{\chi_{l+1:k}(\theta) - \chi_{l+1:k}(z)}{\theta - z}, \qquad l = 0, 1, \ldots, k.$$

Adjugate polynomials and Ritz vectors

Theorem (Ritz vectors)

Let $C_k S_\theta = S_\theta J_\theta$ (for a certain S_θ). Let the Ritz matrix be given by $Y_\theta \equiv Q_k S_\theta$. Then

$$\operatorname{vec}(Y_\theta) = \begin{pmatrix} \mathcal{A}_k(\theta, A) \\ \mathcal{A}'_k(\theta, A) \\ \vdots \\ \dfrac{\mathcal{A}_k^{(\alpha-1)}(\theta, A)}{(\alpha-1)!} \end{pmatrix} q_1 + \sum_{l=1}^{k} c_{1:l-1} \begin{pmatrix} \mathcal{A}_{l+1:k}(\theta, A) \\ \mathcal{A}'_{l+1:k}(\theta, A) \\ \vdots \\ \dfrac{\mathcal{A}_{l+1:k}^{(\alpha-1)}(\theta, A)}{(\alpha-1)!} \end{pmatrix} f_l, \qquad (3)$$

with derivation with respect to the shift θ.

Adjugate polynomials and Ritz vectors

Theorem (Ritz vectors)

Let $C_k S_\theta = S_\theta J_\theta$ (for a certain S_θ). Let the Ritz matrix be given by $Y_\theta \equiv Q_k S_\theta$. Then

$$\operatorname{vec}(Y_\theta) = \begin{pmatrix} \mathcal{A}_k(\theta, A) \\ \mathcal{A}'_k(\theta, A) \\ \vdots \\ \frac{\mathcal{A}_k^{(\alpha-1)}(\theta, A)}{(\alpha-1)!} \end{pmatrix} q_1 + \sum_{l=1}^{k} c_{1:l-1} \begin{pmatrix} \mathcal{A}_{l+1:k}(\theta, A) \\ \mathcal{A}'_{l+1:k}(\theta, A) \\ \vdots \\ \frac{\mathcal{A}_{l+1:k}^{(\alpha-1)}(\theta, A)}{(\alpha-1)!} \end{pmatrix} f_l, \qquad (3)$$

with derivation with respect to the shift θ.

We might scale differently such that (here only for approximate eigenvectors)

$$y = \frac{\mathcal{A}_k(\theta, A)}{\prod_{\ell=1}^{k-1} c_{\ell+1,\ell}} q_1 + \sum_{l=1}^{k} \frac{\mathcal{A}_{l+1:k}(\theta, A)}{\prod_{\ell=l+1}^{k-1} c_{\ell+1,\ell}} \frac{f_l}{c_{l+1,l}}.$$

Lagrange polynomials

We consider Lagrange interpolation polynomials interpolating the inverse and a singularly perturbed identity.

Lagrange polynomials

We consider Lagrange interpolation polynomials interpolating the inverse and a singularly perturbed identity.

The Lagrange interpolation of the inverse is denoted by $\mathcal{L}_k[z^{-1}](z)$.

Lagrange polynomials

We consider Lagrange interpolation polynomials interpolating the inverse and a singularly perturbed identity.

The Lagrange interpolation of the inverse is denoted by $\mathcal{L}_k[z^{-1}](z)$.

- Property:

$$\mathcal{L}_k[z^{-1}](C_k) = C_k^{-1}.$$

Lagrange polynomials

We consider Lagrange interpolation polynomials interpolating the inverse and a singularly perturbed identity.

The Lagrange interpolation of the inverse is denoted by $\mathcal{L}_k[z^{-1}](z)$.

- Property:

$$\mathcal{L}_k[z^{-1}](C_k) = C_k^{-1}.$$

- Definition:

$$\mathcal{L}_k[z^{-1}](z) \equiv \frac{\chi_k(0) - \chi_k(z)}{z\chi_k(0)} = -\frac{\mathcal{A}_k(0,z)}{\chi_k(0)}.$$

Lagrange polynomials

We consider Lagrange interpolation polynomials interpolating the inverse and a singularly perturbed identity.

The Lagrange interpolation of the inverse is denoted by $\mathcal{L}_k[z^{-1}](z)$.

- Property:

$$\mathcal{L}_k[z^{-1}](C_k) = C_k^{-1}.$$

- Definition:

$$\mathcal{L}_k[z^{-1}](z) \equiv \frac{\chi_k(0) - \chi_k(z)}{z\chi_k(0)} = -\frac{\mathcal{A}_k(0,z)}{\chi_k(0)}.$$

- Generalization:

$$\mathcal{L}_{l+1:k}[z^{-1}](z) \equiv \frac{\chi_{l+1:k}(0) - \chi_{l+1:k}(z)}{z\chi_{l+1:k}(0)} = -\frac{\mathcal{A}_{l+1:k}(0,z)}{\chi_{l+1:k}(0)}, \qquad l = 0, 1, \ldots, k.$$

Lagrange polynomials and QOR iterates

Theorem (QOR iterates)

Suppose all $C_{l+1:k}$ *are regular. Define* $z_k \equiv C_k^{-1} e_1 \|r_0\|$ *and* $x_k \equiv Q_k z_k$. *Then*

$$x_k = \mathcal{L}_k[z^{-1}](A) r_0 - \sum_{l=1}^{k} z_{lk} \, \mathcal{L}_{l+1:k}[z^{-1}](A) f_l. \qquad (4)$$

Lagrange polynomials and QOR iterates

Theorem (QOR iterates)

Suppose all $C_{l+1:k}$ *are regular. Define* $z_k \equiv C_k^{-1} e_1 \|r_0\|$ *and* $x_k \equiv Q_k z_k$. *Then*

$$x_k = \mathcal{L}_k[z^{-1}](A) r_0 - \sum_{l=1}^{k} z_{lk} \, \mathcal{L}_{l+1:k}[z^{-1}](A) f_l. \tag{4}$$

Really sloppily speaking, in case of convergence,

$$x_\infty = A^{-1} r_0 + A^{-1} F_\infty z_\infty = A^{-1}(r_0 + F_\infty z_\infty).$$

Lagrange polynomials and QOR iterates

Theorem (QOR iterates)

Suppose all $C_{l+1:k}$ *are regular. Define* $z_k \equiv C_k^{-1} e_1 \|r_0\|$ *and* $x_k \equiv Q_k z_k$*. Then*

$$x_k = \mathcal{L}_k[z^{-1}](A) r_0 - \sum_{l=1}^{k} z_{lk} \, \mathcal{L}_{l+1:k}[z^{-1}](A) f_l. \tag{4}$$

Really sloppily speaking, in case of convergence,

$$x_\infty = A^{-1} r_0 + A^{-1} F_\infty z_\infty = A^{-1}(r_0 + F_\infty z_\infty).$$

Proving convergence is the hard task.

Lagrange polynomials (continued)

We consider Lagrange interpolation polynomials interpolating the inverse and a singularly perturbed identity.

Lagrange polynomials (continued)

We consider Lagrange interpolation polynomials interpolating the inverse and a singularly perturbed identity.

The Lagrange interpolation of the singularly perturbed identity is denoted by $\mathcal{L}_k^0[1 - \delta_{z0}](z)$.

Lagrange polynomials (continued)

We consider Lagrange interpolation polynomials interpolating the inverse and a singularly perturbed identity.

The Lagrange interpolation of the singularly perturbed identity is denoted by $\mathcal{L}_k^0[1-\delta_{z0}](z)$.

- Properties:

$$\mathcal{L}_k^0[1-\delta_{z0}](C_k) = I_k, \qquad \mathcal{L}_k^0[1-\delta_{z0}](0) = 0.$$

Lagrange polynomials (continued)

We consider Lagrange interpolation polynomials interpolating the inverse and a singularly perturbed identity.

The Lagrange interpolation of the singularly perturbed identity is denoted by $\mathcal{L}_k^0[1-\delta_{z0}](z)$.

- Properties:

$$\mathcal{L}_k^0[1-\delta_{z0}](C_k) = I_k, \qquad \mathcal{L}_k^0[1-\delta_{z0}](0) = 0.$$

- Definition:

$$\mathcal{L}_k^0[1-\delta_{z0}](z) \equiv \frac{\chi_k(0)-\chi_k(z)}{\chi_k(0)} = \mathcal{L}_k[z^{-1}](z)z.$$

Lagrange polynomials (continued)

We consider Lagrange interpolation polynomials interpolating the inverse and a singularly perturbed identity.

The Lagrange interpolation of the singularly perturbed identity is denoted by $\mathcal{L}_k^0[1-\delta_{z0}](z)$.

- Properties:

$$\mathcal{L}_k^0[1-\delta_{z0}](C_k) = I_k, \qquad \mathcal{L}_k^0[1-\delta_{z0}](0) = 0.$$

- Definition:

$$\mathcal{L}_k^0[1-\delta_{z0}](z) \equiv \frac{\chi_k(0)-\chi_k(z)}{\chi_k(0)} = \mathcal{L}_k[z^{-1}](z)z.$$

- Generalization:

$$\mathcal{L}_{l+1:k}^0[1-\delta_{z0}](z) \equiv \frac{\chi_{l+1:k}(0)-\chi_{l+1:k}(z)}{\chi_{l+1:k}(0)} = \mathcal{L}_{l+1:k}[z^{-1}](z)z. \qquad l = 0,1,\ldots,k.$$

Residual polynomials

We consider the well-known residual polynomials [Stiefel, 1995] denoted by $\mathcal{R}_k(z)$.

Residual polynomials

We consider the well-known residual polynomials [Stiefel, 1995] denoted by $\mathcal{R}_k(z)$.

- Properties:

$$\mathcal{R}_k(C_k) = O_k, \qquad \mathcal{R}_k(0) = 1.$$

Residual polynomials

We consider the well-known residual polynomials [Stiefel, 1995] denoted by $\mathcal{R}_k(z)$.

- Properties:

$$\mathcal{R}_k(C_k) = O_k, \qquad \mathcal{R}_k(0) = 1.$$

- Definition:

$$\mathcal{R}_k(z) \equiv \frac{\chi_k(z)}{\chi_k(0)} = 1 - \mathcal{L}_k^0[1 - \delta_{z0}](z) = \det(I_k - zC_k^{-1}).$$

Residual polynomials

We consider the well-known residual polynomials [Stiefel, 1995] denoted by $\mathcal{R}_k(z)$.

- Properties:

$$\mathcal{R}_k(C_k) = O_k, \qquad \mathcal{R}_k(0) = 1.$$

- Definition:

$$\mathcal{R}_k(z) \equiv \frac{\chi_k(z)}{\chi_k(0)} = 1 - \mathcal{L}_k^0[1 - \delta_{z0}](z) = \det(I_k - zC_k^{-1}).$$

- Generalization:

$$\mathcal{R}_{l+1:k}(z) \equiv \frac{\chi_{l+1:k}(z)}{\chi_{l+1:k}(0)} = 1 - \mathcal{L}_{l+1:k}^0[1 - \delta_{z0}](z). \qquad l = 0, 1, \ldots, k.$$

Residual polynomials

We consider the well-known residual polynomials [Stiefel, 1995] denoted by $\mathcal{R}_k(z)$.

- Properties:

$$\mathcal{R}_k(C_k) = O_k, \qquad \mathcal{R}_k(0) = 1.$$

- Definition:

$$\mathcal{R}_k(z) \equiv \frac{\chi_k(z)}{\chi_k(0)} = 1 - \mathcal{L}_k^0[1 - \delta_{z0}](z) = \det(I_k - zC_k^{-1}).$$

- Generalization:

$$\mathcal{R}_{l+1:k}(z) \equiv \frac{\chi_{l+1:k}(z)}{\chi_{l+1:k}(0)} = 1 - \mathcal{L}_{l+1:k}^0[1 - \delta_{z0}](z). \qquad l = 0, 1, \ldots, k.$$

Two types of polynomials $\Rightarrow$ two expressions for the QOR residuals.

Residual polynomials and QOR residuals

Theorem (QOR residuals)

Suppose $q_1 = r_0/\|r_0\|$ *and let all* $C_{l+1:k}$ *be invertible. Let* x_k *denote the QOR iterate and* $r_k = r_0 - Ax_k$ *the corresponding residual.*
Then

$$\begin{aligned} r_k &= \mathcal{R}_k(A)r_0 + \sum_{l=1}^{k} z_{lk}\, \mathcal{L}^0_{l+1:k}[1-\delta_{z0}](A)f_l \\ &= \mathcal{R}_k(A)r_0 - \sum_{l=1}^{k} z_{lk}\, \mathcal{R}_{l+1:k}(A)f_l + F_k z_k. \end{aligned} \tag{5}$$

Residual polynomials and QOR residuals

Theorem (QOR residuals)

Suppose $q_1 = r_0/\|r_0\|$ *and let all* $C_{l+1:k}$ *be invertible. Let* x_k *denote the QOR iterate and* $r_k = r_0 - Ax_k$ *the corresponding residual.*
Then

$$\begin{aligned} r_k &= \mathcal{R}_k(A)r_0 + \sum_{l=1}^{k} z_{lk}\, \mathcal{L}^0_{l+1:k}[1 - \delta_{z0}](A) f_l \\ &= \mathcal{R}_k(A)r_0 - \sum_{l=1}^{k} z_{lk}\, \mathcal{R}_{l+1:k}(A) f_l + F_k z_k. \end{aligned} \tag{5}$$

First expression: related to perturbation amplification.

Residual polynomials and QOR residuals

Theorem (QOR residuals)

Suppose $q_1 = r_0/\|r_0\|$ and let all $C_{l+1:k}$ be invertible. Let x_k denote the QOR iterate and $r_k = r_0 - Ax_k$ the corresponding residual.
Then

$$\begin{aligned} r_k &= \mathcal{R}_k(A)r_0 + \sum_{l=1}^{k} z_{lk}\, \mathcal{L}^0_{l+1:k}[1 - \delta_{z0}](A) f_l \\ &= \mathcal{R}_k(A)r_0 - \sum_{l=1}^{k} z_{lk}\, \mathcal{R}_{l+1:k}(A) f_l + F_k z_k. \end{aligned} \tag{5}$$

First expression: related to perturbation amplification.

Second expression: related to the attainable accuracy.

An example: Lanczos’ method

We used the diagonal matrix

$$A = \texttt{diag([linspace(0,1,50),3])}$$

and the starting vector

$$e = \texttt{ones(51,1)}$$

in an implementation of Lanczos’ method in MATLAB on a PC conforming to ANSI/IEEE 754 with machine precision $\texttt{eps(1)} = 2^{-52} \approx 2.2204 \cdot 10^{-16}$.

An example: Lanczos’ method

We used the diagonal matrix

$$A = \texttt{diag([linspace(0,1,50),3])}$$

and the starting vector

$$e = \texttt{ones(51,1)}$$

in an implementation of Lanczos’ method in MATLAB on a PC conforming to ANSI/IEEE 754 with machine precision $\texttt{eps(1)} = 2^{-52} \approx 2.2204 \cdot 10^{-16}$.

At step 10 the first Ritz value has converged (up to machine precision) to the eigenvalue 3, at step 27 the second one has converged.

An example: Lanczos’ method

We used the diagonal matrix

$$A = \texttt{diag([linspace(0,1,50),3])}$$

and the starting vector

$$e = \texttt{ones(51,1)}$$

in an implementation of Lanczos’ method in MATLAB on a PC conforming to ANSI/IEEE 754 with machine precision $\texttt{eps(1)} = 2^{-52} \approx 2.2204 \cdot 10^{-16}$.

At step 10 the first Ritz value has converged (up to machine precision) to the eigenvalue 3, at step 27 the second one has converged.

Eigenvalues and eigenvectors are computed using MRRR, i.e., LAPACK’s routine `DSTEGR`, since MATLAB’s `eig` (using LAPACK’s `DSYEV`, i.e., the QR algorithm implemented as `DSTEQR`) fails in delivering accurate eigenvectors.

An example: Lanczos' method

We used the diagonal matrix

$$A = \texttt{diag([linspace(0,1,50),3])}$$

and the starting vector

$$e = \texttt{ones(51,1)}$$

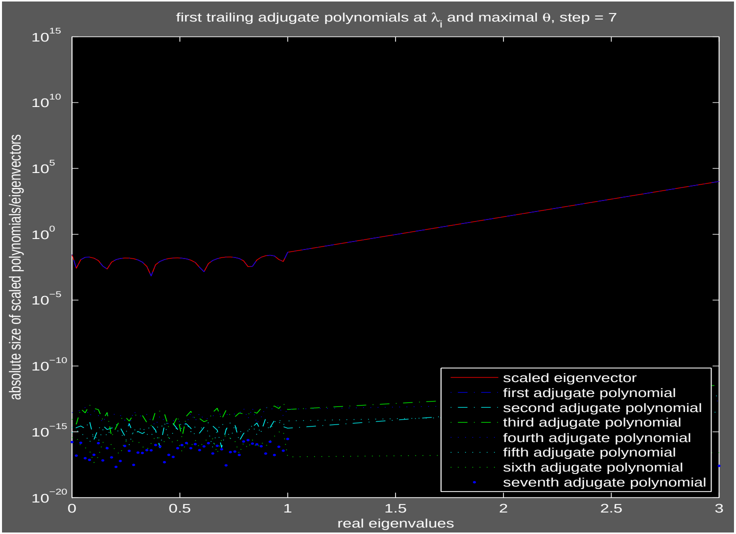
in an implementation of Lanczos' method in MATLAB on a PC conforming to ANSI/IEEE 754 with machine precision $\texttt{eps(1)} = 2^{-52} \approx 2.2204 \cdot 10^{-16}$.

At step 10 the first Ritz value has converged (up to machine precision) to the eigenvalue 3, at step 27 the second one has converged.

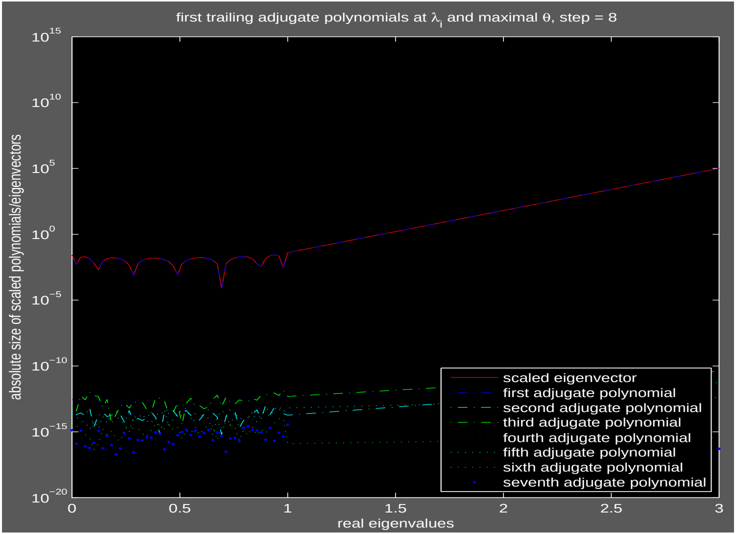
Eigenvalues and eigenvectors are computed using MRRR, i.e., LAPACK's routine `DSTEGR`, since MATLAB's `eig` (using LAPACK's `DSYEV`, i.e., the QR algorithm implemented as `DSTEQR`) fails in delivering accurate eigenvectors.

Additionally, we heavily used the symbolic toolbox, i.e., MAPLE.

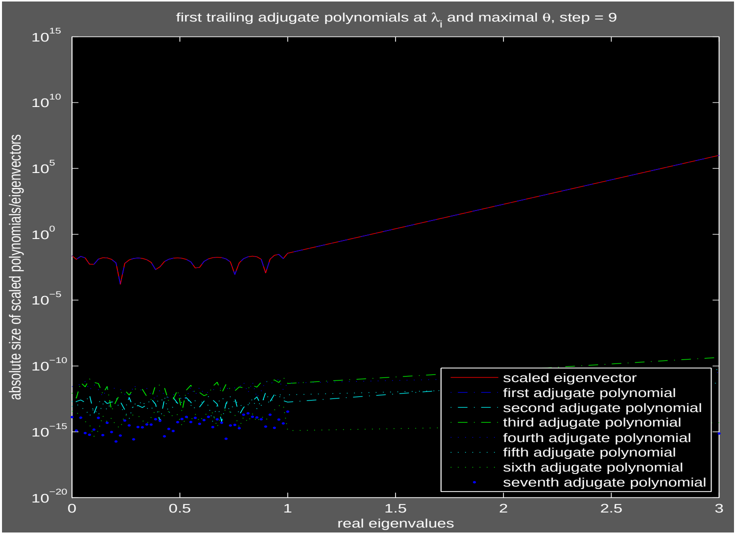
An example: Lanczos’ method



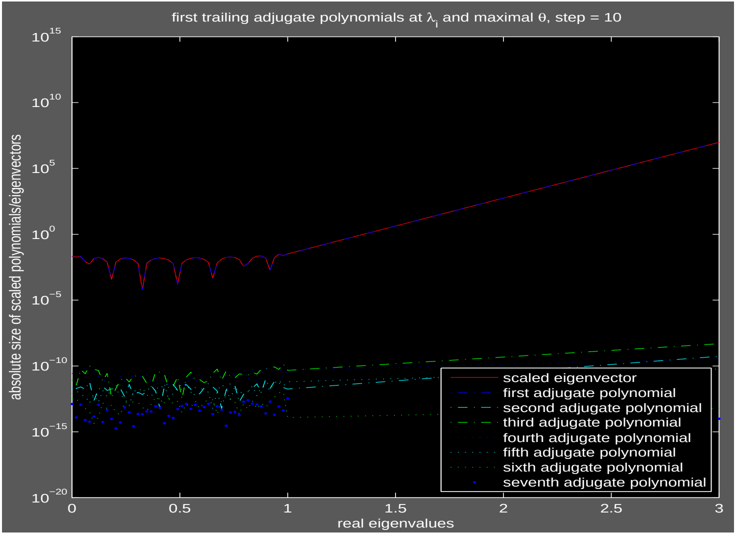
An example: Lanczos’ method



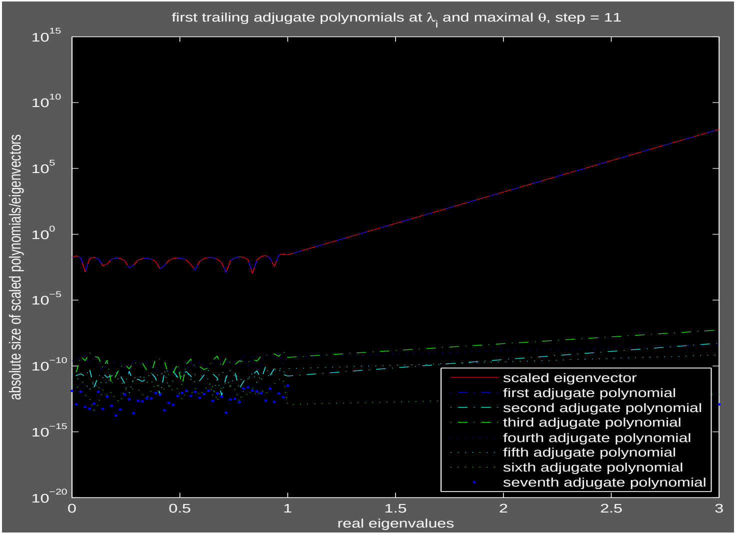
An example: Lanczos’ method



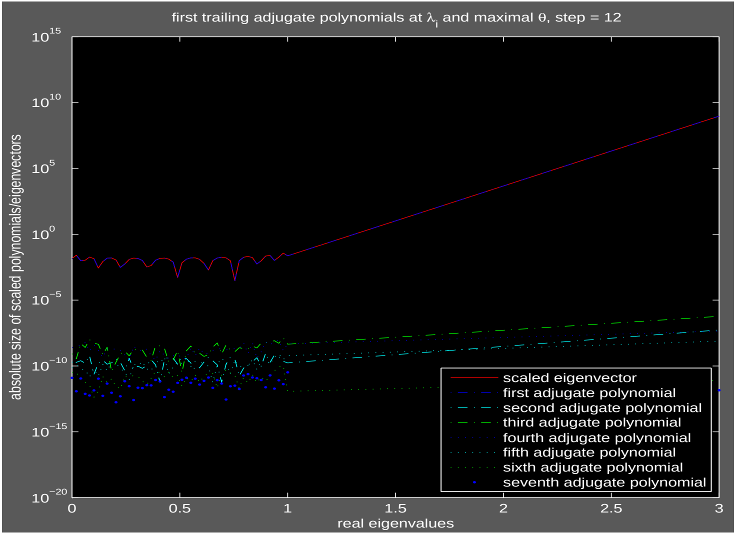
An example: Lanczos’ method



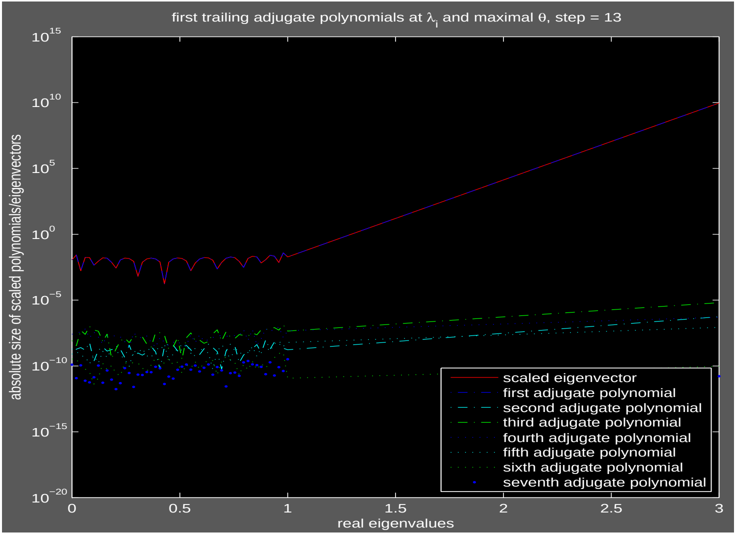
An example: Lanczos' method



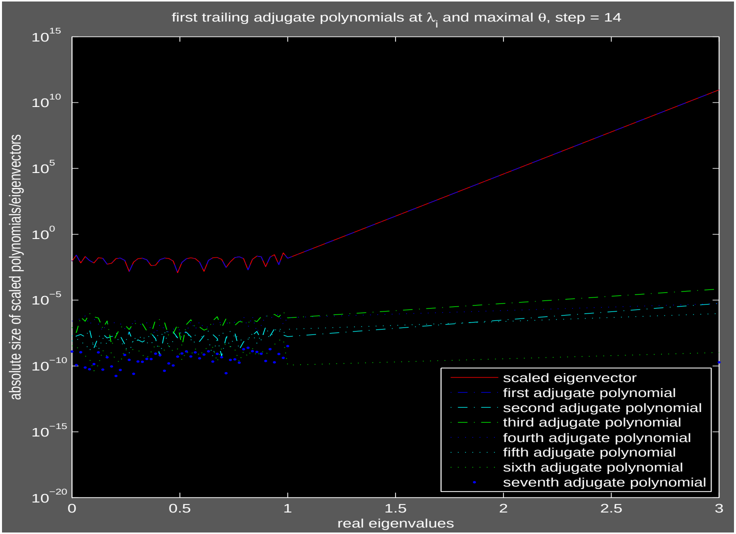
An example: Lanczos’ method



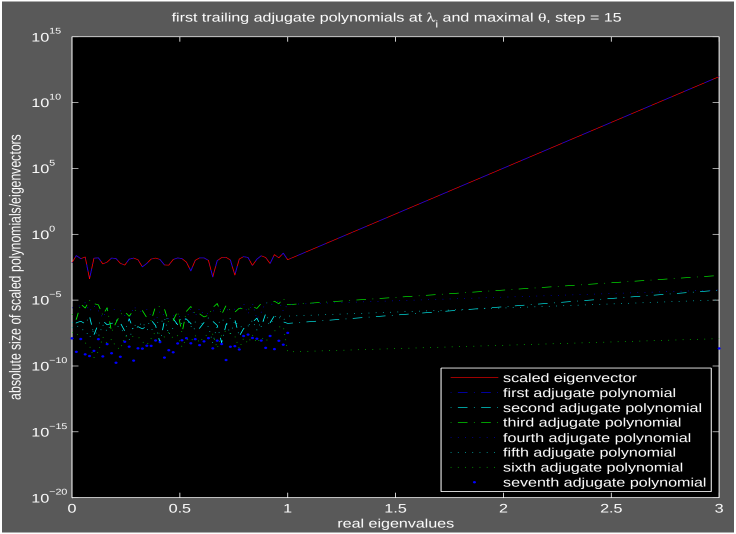
An example: Lanczos’ method



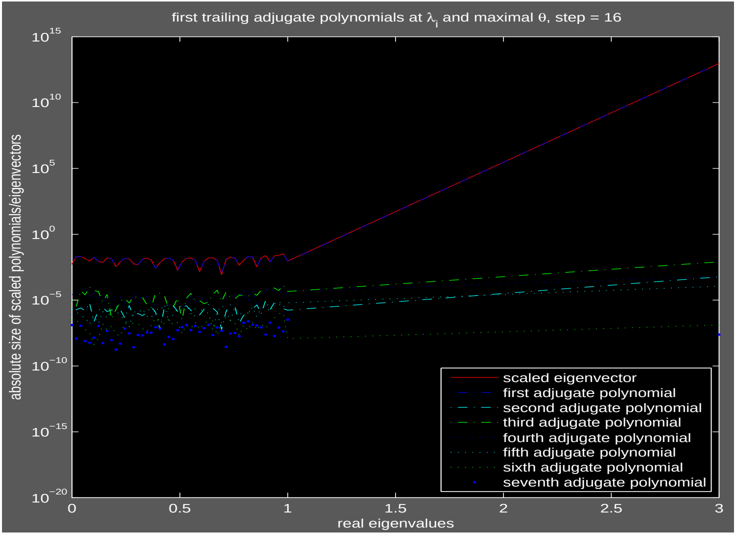
An example: Lanczos’ method



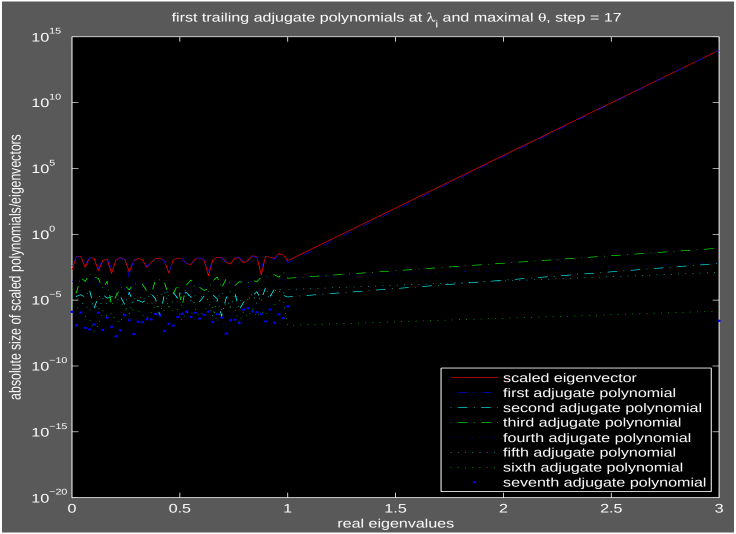
An example: Lanczos’ method



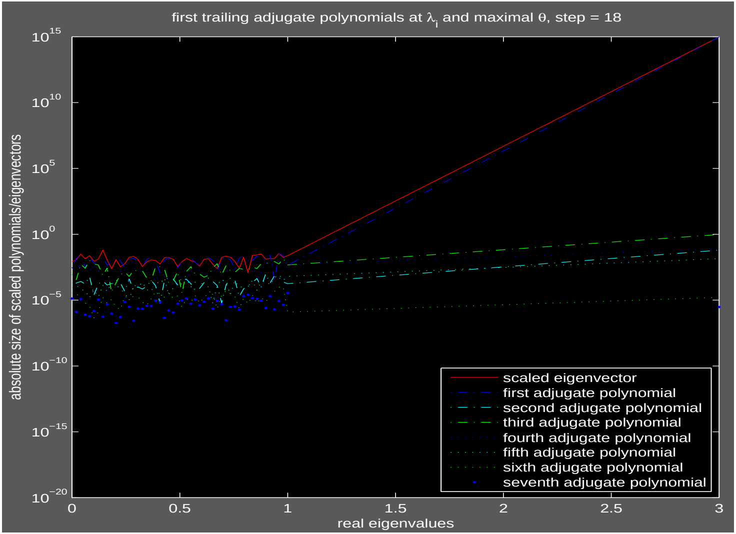
An example: Lanczos’ method



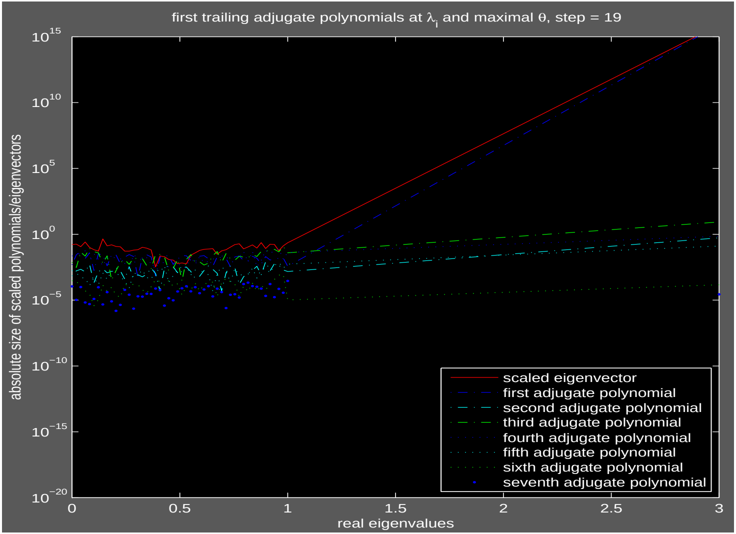
An example: Lanczos’ method



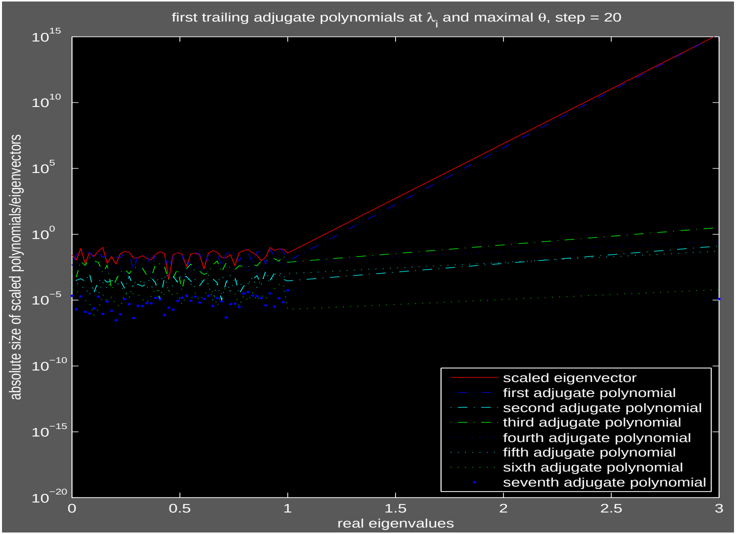
An example: Lanczos' method



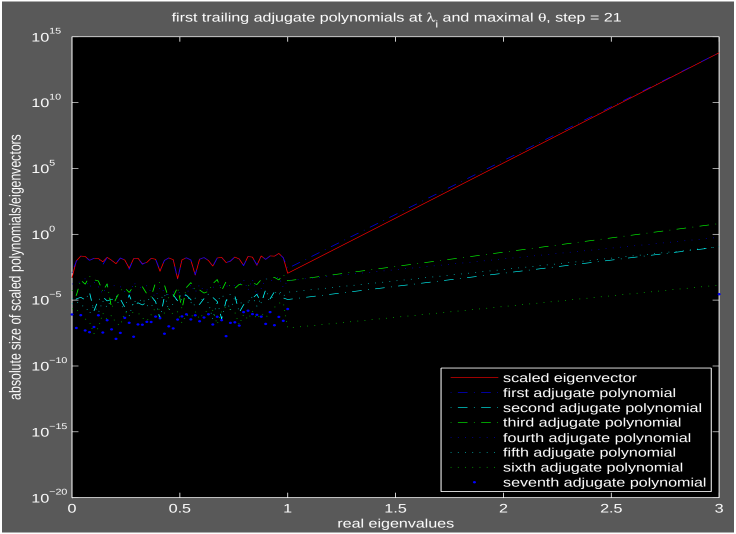
An example: Lanczos' method



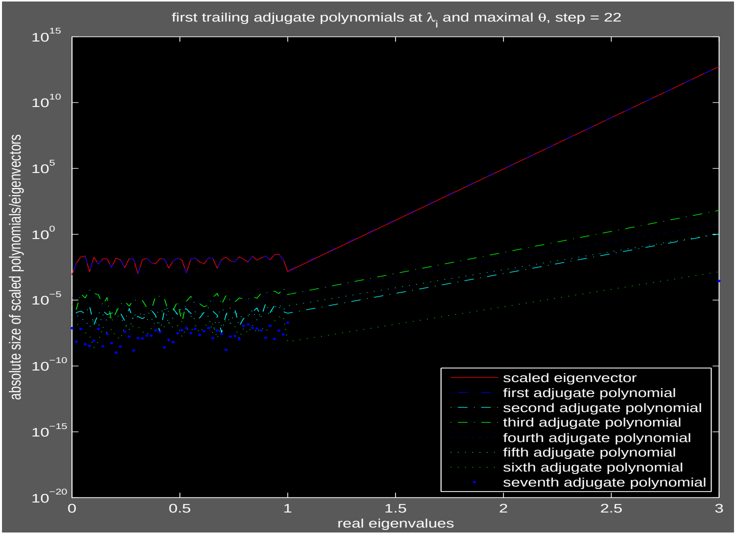
An example: Lanczos' method



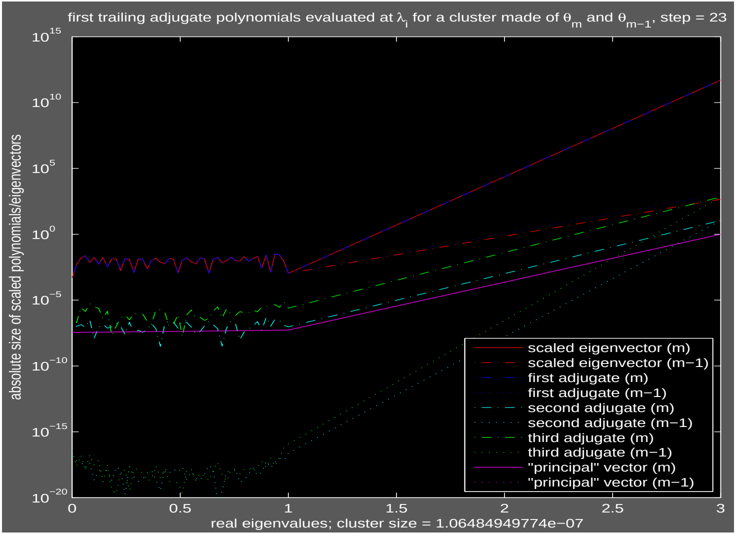
An example: Lanczos’ method



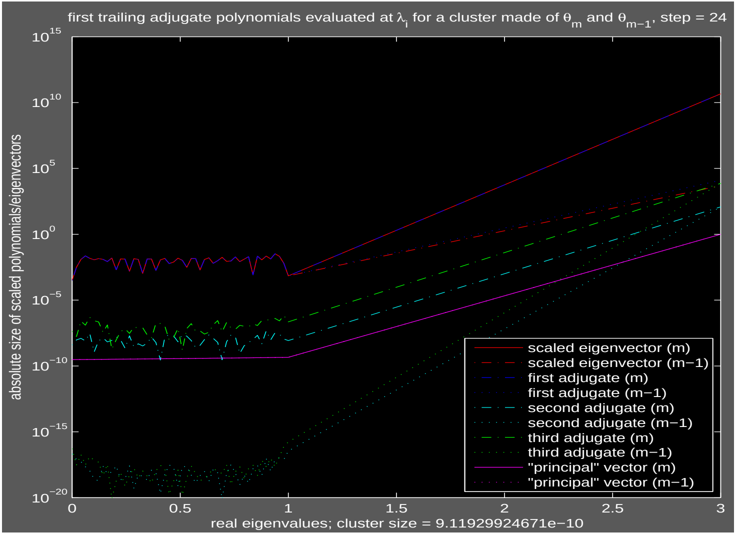
An example: Lanczos’ method



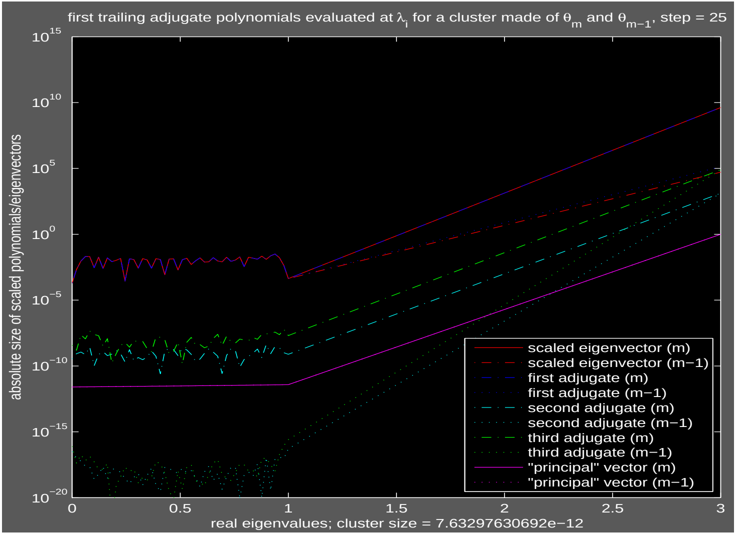
An example: Lanczos’ method



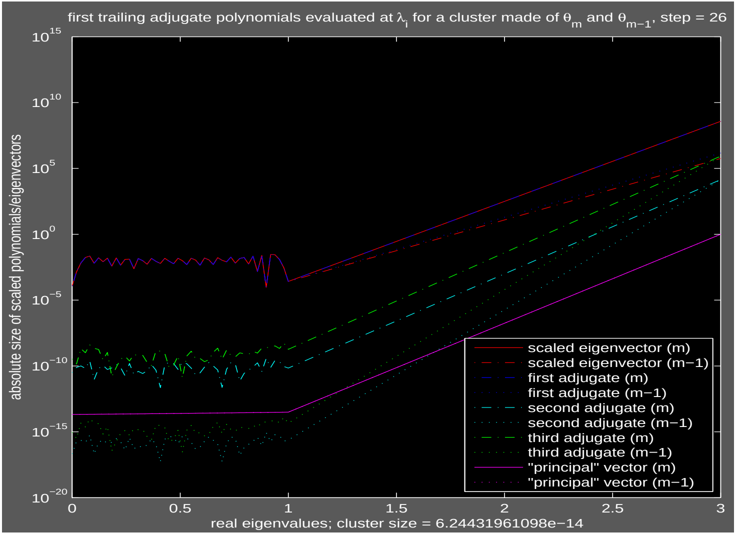
An example: Lanczos' method



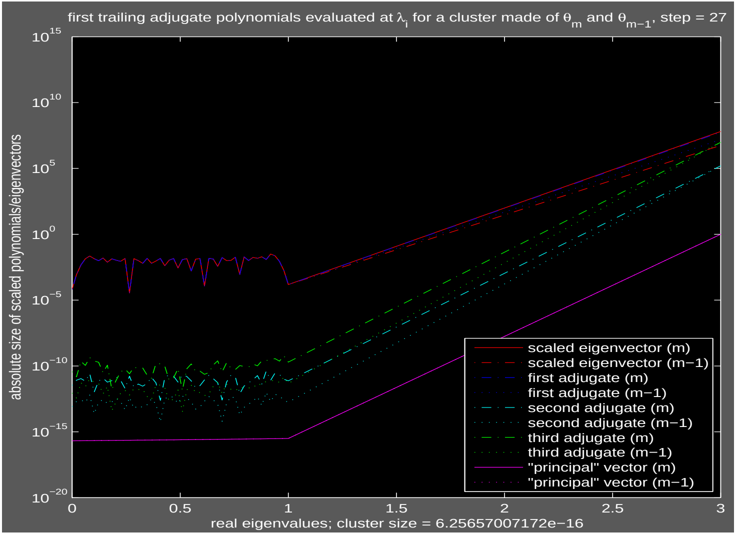
An example: Lanczos’ method



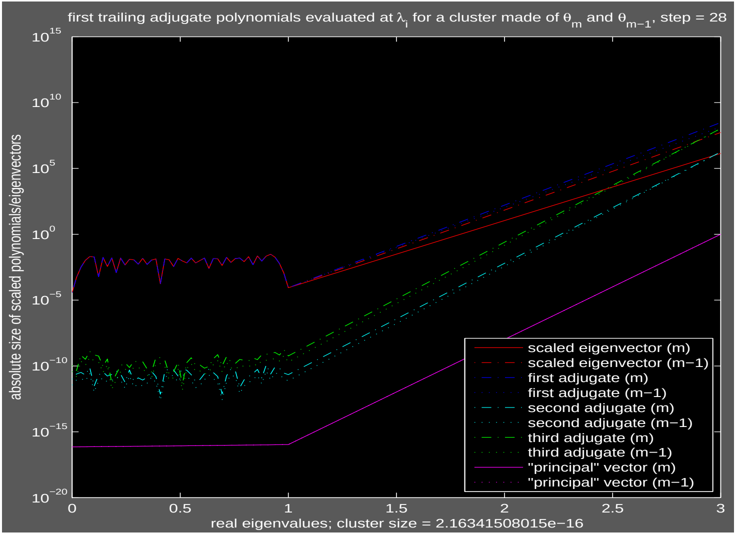
An example: Lanczos’ method



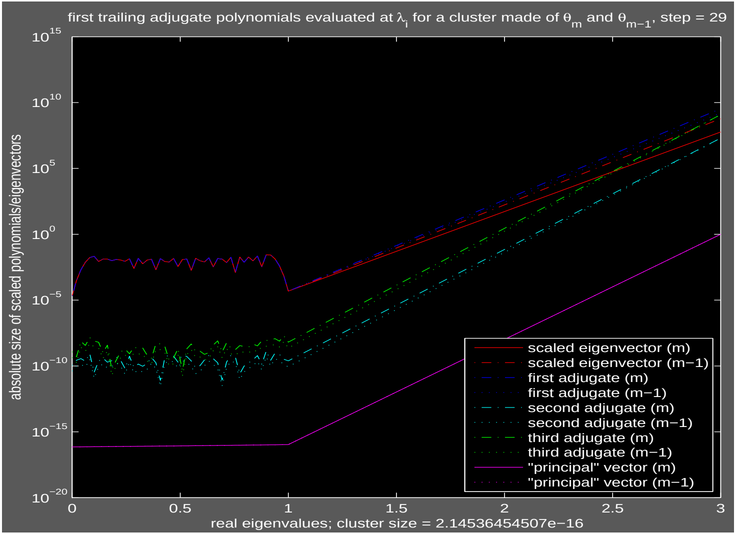
An example: Lanczos’ method



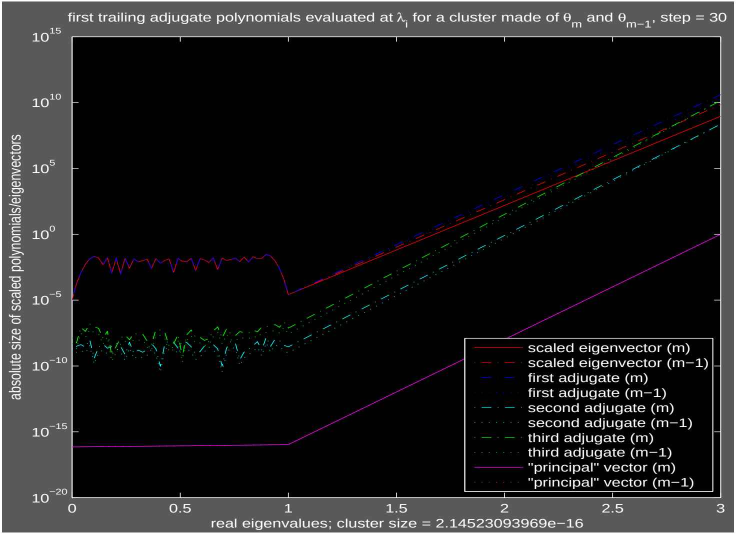
An example: Lanczos’ method



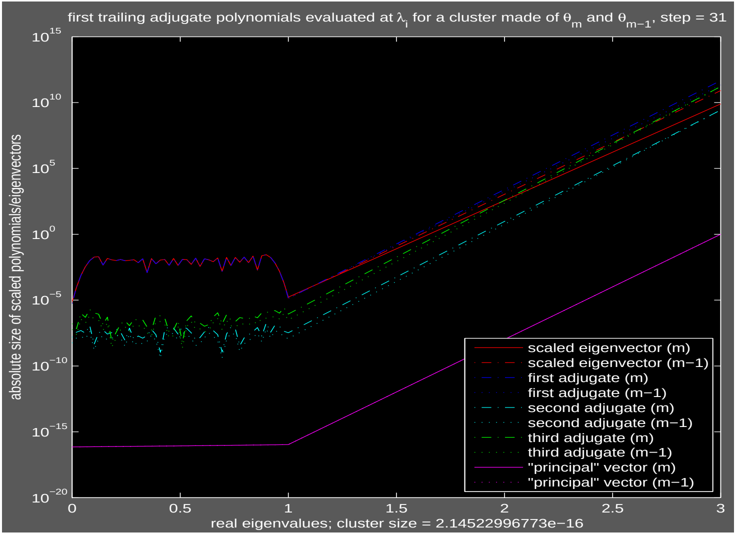
An example: Lanczos’ method



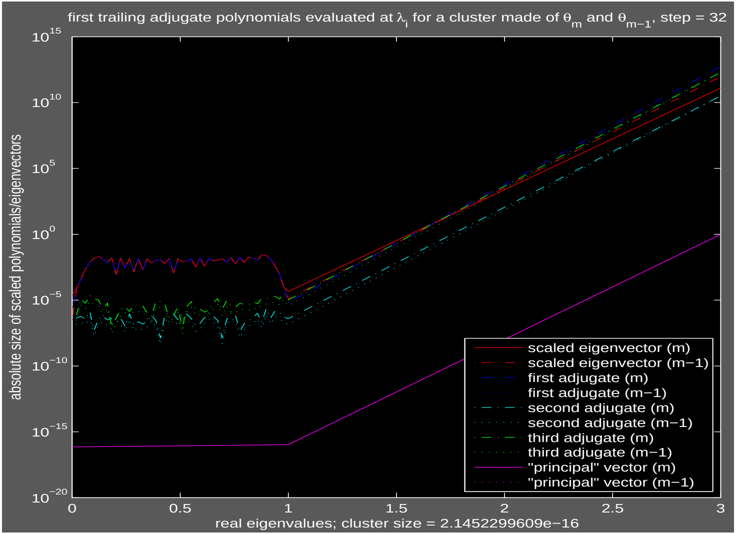
An example: Lanczos' method



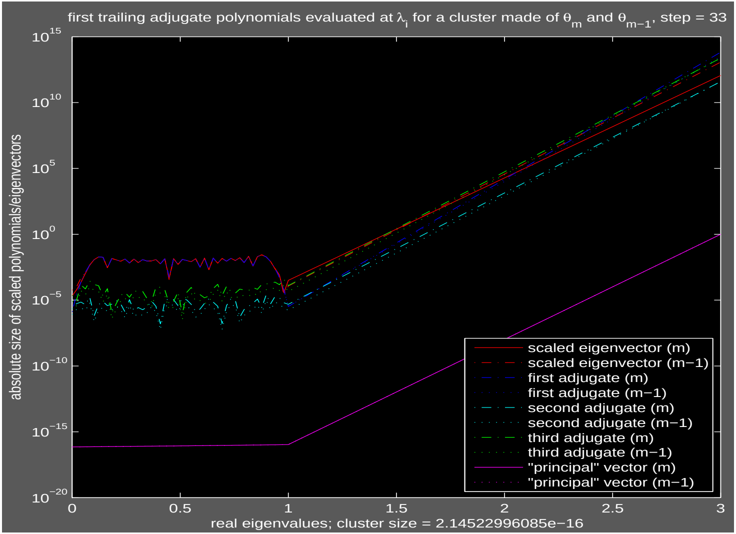
An example: Lanczos’ method



An example: Lanczos’ method



An example: Lanczos’ method



E. Stiefel
Relaxationsmethoden bester Strategie zur Lösung linearer Gleichungssysteme
Commentarii Mathematici Helvetici 29, 157–179, 1955,
http://www-gdz.sub.uni-goettingen.de/cgi-bin/digbib.cgi?PPN358147735_0029.

Jens-Peter M. Zemke
Hessenberg eigenvalue–eigenmatrix relations.
Linear Algebra and Its Applications, 414/2–3, pp. 589–606, 2006,
http://dx.doi.org/10.1016/j.laa.2005.11.002.

Jens-Peter M. Zemke
Abstract perturbed Krylov methods.
Bericht 89, Institut für Numerische Simulation, TUHH, 2005,
http://doku.b.tu-harburg.de/volltexte/2006/158/.