ITERATIVE PROJECTION METHODS FOR SPARSE LINEAR SYSTEMS AND EIGENPROBLEMS

CHAPTER 10: KRYLOV SUBSPACE METHODS

Heinrich Voss
voss@tu-harburg.de

Hamburg University of Technology
Institute of Numerical Simulation
We already discussed in Sections 4, 5, and 6 how to construct an orthonormal basis $V_m$ of the Krylov space $K_m(v^1, A)$ by the Arnoldi method and for symmetric matrices by the Lanczos method.
We already discussed in Sections 4, 5, and 6 how to construct an orthonormal basis \( V_m \) of the Krylov space \( \mathcal{K}_m(v^1, A) \) by the Arnoldi method and for symmetric matrices by the Lanczos method.

Along with the bases one obtains the orthogonal projection

\[
    H_m := V_m^T AV_m \quad (H_m := V_m^H AV_m \text{ in the complex case})
\]

of \( A \) onto \( \mathcal{K}_m(v^1, A) \) which is a Hessenberg matrix for general matrix \( A \), and triangular for symmetric (Hermitean) \( A \).
We already discussed in Sections 4, 5, and 6 how to construct an orthonormal basis \( V_m \) of the Krylov space \( \mathcal{K}_m(v^1, A) \) by the Arnoldi method and for symmetric matrices by the Lanczos method.

Along with the bases one obtains the orthogonal projection

\[
H_m := V_m^T A V_m \quad (H_m := V_m^H A V_m \text{ in the complex case})
\]

of \( A \) onto \( \mathcal{K}_m(v^1, A) \) which is a Hessenberg matrix for general matrix \( A \), and triangular for symmetric (Hermitean) \( A \).

In Section 8 we discussed the two-sided Lanczos process which constructs bases \( V_m \) and \( W_m \) of the Krylov spaces \( \mathcal{K}_m(v^1, A) \) and \( \mathcal{K}_m(w^1, A^T) \), respectively, along with the oblique projection of \( A \) onto \( \mathcal{K}_m(v^1, A) \) along \( \mathcal{K}_m(w^1, A^T) \)

\[
T_m = W_m^T A V_m.
\]
We already discussed in Sections 4, 5, and 6 how to construct an orthonormal basis $V_m$ of the Krylov space $K_m(v^1, A)$ by the Arnoldi method and for symmetric matrices by the Lanczos method.

Along with the bases one obtains the orthogonal projection

$$H_m := V_m^T A V_m \quad (H_m := V_m^H A V_m \text{ in the complex case})$$

of $A$ onto $K_m(v^1, A)$ which is a Hessenberg matrix for general matrix $A$, and triangular for symmetric (Hermitean) $A$.

In Section 8 we discussed the two-sided Lanczos process which constructs bases $V_m$ and $W_m$ of the Krylov spaces $K_m(v^1, A)$ and $K_m(w^1, A^T)$, respectively, along with the oblique projection of $A$ onto $K_m(v^1, A)$ along $K_m(w^1, A^T)$

$$T_m = W_m^T A V_m.$$ 

$T_m$ is a triangular matrix. If look-ahead is included, then $T_m$ is block-triangular.
Arnoldi method; compact form

\[ AV_m = V_m H_m + h_{m+1,m} v^{m+1}_m e_T^m \]

\[ V_m^T V_m = I_m, \quad V_m^T v^{m+1}_m = 0, \]

\[ V_m^T A V_m = H_m = \begin{pmatrix}
    h_{11} & h_{12} & h_{13} & \cdots & \cdots & h_{1m} \\
    h_{21} & h_{22} & h_{23} & \cdots & \cdots & h_{2m} \\
    0 & h_{32} & h_{33} & \cdots & \cdots & h_{3m} \\
    \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\
    \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\
    0 & \cdots & \cdots & \cdots & \cdots & h_{m,m-1} & h_{mm}
\end{pmatrix} \]
Arnoldi method; compact form

\[ AV_m = V_m H_m + h_{m+1,m} v^{m+1} e_m^T \]

\[ V_m^T V_m = I_m, \quad V_m^T v^{m+1} = 0, \]

\[
\begin{pmatrix}
    h_{11} & h_{12} & h_{13} & \ldots & \ldots & h_{1m} \\
    h_{21} & h_{22} & h_{23} & \ldots & \ldots & h_{2m} \\
    0 & h_{32} & h_{33} & \ldots & \ldots & h_{3m} \\
    \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
    \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\
    0 & 0 & 0 & \ldots & h_{m,m-1} & h_{mm} \\
\end{pmatrix}
\]

The eigenvalue problem

\[ H_m s = \theta s \]

can be solved inexpensively by the QR algorithm.
The Arnoldi method may terminate with $h_{j,j+1} = 0$ for some $j$. 

Hence, $K_{j}(A, v_1)$ is an invariant subspace of $A$, and therefore every eigenvalue $\theta_j$ of $H_j$ is an eigenvalue of $A$, and the corresponding Ritz vectors are eigenvectors of $A$. 

Lucky termination
Lucky termination

The Arnoldi method may terminate with $h_{j,j+1} = 0$ for some $j$.

Then $v^{j+1} = 0$, and therefore

$$Av^j = \sum_{k=1}^{j} h_{kj} v^k \in \mathcal{K}_j(A, v^1).$$
The Arnoldi method may terminate with \( h_{j,j+1} = 0 \) for some \( j \).

Then \( \nu^{j+1} = 0 \), and therefore

\[
A\nu^j = \sum_{k=1}^{j} h_{kj} \nu^k \in \mathcal{K}_j(A, \nu^1).
\]

For \( i < j \) it holds by construction

\[
A\nu^i = \sum_{k=1}^{i+1} h_{ki} \nu^k \in \mathcal{K}_j(A, \nu^1).
\]

Hence, \( \mathcal{K}_j(A, \nu^1) \) is an invariant subspace of \( A \), and therefore every eigenvalue \( \theta_i^{(j)} \) of \( H_j \) is an eigenvalue of \( A \), and the corresponding Ritz vectors are eigenvectors of \( A \).
If $\theta_i^{(m)}$ are the Ritz values (eigenvalues of $H_m$), $s_i^{(m)}$ the corresponding eigenvectors, and $x_i^{(m)} = V_m s_i^{(m)}$ the Ritz vectors, then (as for the Lanczos method) it holds

$$(A - \theta_i^{(m)} I)x_i^{(m)} = h_{m+1,m} v^{m+1} e^T s_i^{(m)},$$

which implies

$$\| (A - \theta_i^{(m)} I)x_i^{(m)} \|_2 = h_{m+1,m} |s_i^{(m)}|.$$
If $\theta_i^{(m)}$ are the Ritz values (eigenvalues of $H_m$), $s_i^{(m)}$ the corresponding eigenvectors, and $x_i^{(m)} = V_m s_i^{(m)}$ the Ritz vectors, then (as for the Lanczos method) it holds

$$(A - \theta_i^{(m)} I)x_i^{(m)} = h_{m+1,m} v^{m+1} e^T_s s_i^{(m)},$$

which implies

$$\| (A - \theta_i^{(m)} I)x_i^{(m)} \|_2 = h_{m+1,m} |s_{m,i}^{(m)}|.$$ 

If $A$ is symmetric then by the Krylov & Bogoliubov Theorem there exists an eigenvalue $\tilde{\lambda}$ of $A$ such that

$$|\tilde{\lambda} - \theta_i^{(m)}| \leq h_{m+1,m} |s_{m,i}^{(m)}|.$$
If \( \theta_i^{(m)} \) are the Ritz values (eigenvalues of \( H_m \)), \( s_i^{(m)} \) the corresponding eigenvectors, and \( x_i^{(m)} = V_m s_i^{(m)} \) the Ritz vectors, then (as for the Lanczos method) it holds

\[
(A - \theta_i^{(m)} I)x_i^{(m)} = h_{m+1,m} v^{m+1} e_T s_i^{(m)},
\]

which implies

\[
\| (A - \theta_i^{(m)} I)x_i^{(m)} \|_2 = h_{m+1,m} |s_{m,i}|.
\]

If \( A \) is symmetric then by the Krylov & Bogoliubov Theorem there exists an eigenvalue \( \tilde{\lambda} \) of \( A \) such that

\[
|\tilde{\lambda} - \theta_i^{(m)}| \leq h_{m+1,m} |s_{m,i}|.
\]

Notice, that this error bound can be computet without determining the Ritz vector \( x_i^{(m)} = V_m s_i^{(m)} \).
If $\theta_i^{(m)}$ are the Ritz values (eigenvalues of $H_m$), $s_i^{(m)}$ the corresponding eigenvectors, and $x_i^{(m)} = V_m s_i^{(m)}$ the Ritz vectors, then (as for the Lanczos method) it holds

$$(A - \theta_i^{(m)} I)x_i^{(m)} = h_{m+1,m} v^{m+1} e^T_m s_i^{(m)},$$

which implies

$$\| (A - \theta_i^{(m)} I)x_i^{(m)} \|_2 = h_{m+1,m} |s_{m,i}^{(m)}|.$$ 

If $A$ is symmetric then by the Krylov & Bogoliubov Theorem there exists an eigenvalue $\tilde{\lambda}$ of $A$ such that

$$|\tilde{\lambda} - \theta_i^{(m)}| \leq h_{m+1,m} |s_{m,i}^{(m)}|.$$ 

Notice, that this error bound can be computed without determining the Ritz vector $x_i^{(m)} = V_m s_i^{(m)}$.

If $A$ is non-normal, then the Krylov & Bogoliubov Theorem does not hold. Nonetheless, in this case the backward error $h_{m+1,m} |s_{m,i}^{(m)}|$ is used as an error indicator.
Arnoldi method

1: choose initial vector $\mathbf{v}^1$ with $\|\mathbf{v}^1\| = 1$, $\mathbf{V}_1 = [\mathbf{v}^1]$
2: compute $\mathbf{w} = \mathbf{A}\mathbf{v}^1$, $\mathbf{h} = (\mathbf{v}^1)^T \mathbf{w}$, $\mathbf{r} = \mathbf{w} - \mathbf{v}^1 \mathbf{h}$, $\mathbf{H}_1 = [\mathbf{h}]$, $\beta = \|\mathbf{r}\|_2$
3: for $j = 1, 2, \ldots$ do
4: $\mathbf{v}^{j+1} = \mathbf{r} / \beta$
5: $\mathbf{V}_{j+1} = [\mathbf{V}_j, \mathbf{v}^{j+1}]$, $\mathbf{\hat{H}}_j = \begin{bmatrix} \mathbf{H}_j \\ \beta \mathbf{e}_j^T \end{bmatrix}$
6: $\mathbf{w} = \mathbf{A}\mathbf{v}^{j+1}$
7: $\mathbf{h} = \mathbf{V}_{j+1}^T \mathbf{w}$, $\mathbf{r} = \mathbf{w} - \mathbf{V}_{j+1} \mathbf{h}$
8: if $\|\mathbf{r}\|_2 < \eta \|\mathbf{h}\|_2$ then
9: $\mathbf{s} = \mathbf{V}_{j+1}^T \mathbf{r}$, $\mathbf{r} = \mathbf{r} - \mathbf{V}_{j+1} \mathbf{s}$
10: $\mathbf{h} = \mathbf{h} + \mathbf{s}$
11: end if
12: $\mathbf{H}_{j+1} = [\mathbf{\hat{H}}_j, \mathbf{h}]$, $\beta = \|\mathbf{r}\|_2$
13: compute approximate eigenvalues of $\mathbf{H}_{j+1}$
14: test for convergence
15: end for
For large matrices the Arnoldi method becomes costly, both in terms of computation and storage.
For large matrices the Arnoldi method becomes costly, both in terms of computation and storage.

We need to keep $m$ vectors of length $n$ plus an $m \times m$ Hessenberg matrix.
For large matrices the Arnoldi method becomes costly, both in terms of computation and storage.

We need to keep $m$ vectors of length $n$ plus an $m \times m$ Hessenberg matrix.

For the arithmetic costs, we need to multiply $v^{j+1}$ by $A$, at the cost of $2N_z$, where $N_z$ is the number of nonzero elements of $A$, and then orthogonalize the result against $j$ basis vectors, at the cost of $4(j + 1)n$. 
For large matrices the Arnoldi method becomes costly, both in terms of computation and storage.

We need to keep $m$ vectors of length $n$ plus an $m \times m$ Hessenberg matrix.

For the arithmetic costs, we need to multiply $v^{j+1}$ by $A$, at the cost of $2N_z$, where $N_z$ is the number of nonzero elements of $A$, and then orthogonalize the result against $j$ basis vectors, at the cost of $4(j + 1)n$.

Thus, an $m$-dimensional Arnoldi costs $\approx nm + 0.5m^2$ in storage and $\approx 2nN_z + 2nm^2$ in arithmetic operations.
For large matrices the Arnoldi method becomes costly, both in terms of computation and storage.

We need to keep $m$ vectors of length $n$ plus an $m \times m$ Hessenberg matrix.

For the arithmetic costs, we need to multiply $v^{j+1}$ by $A$, at the cost of $2N_z$, where $N_z$ is the number of nonzero elements of $A$, and then orthogonalize the result against $j$ basis vectors, at the cost of $4(j + 1)n$.

Thus, an $m$-dimensional Arnoldi costs $\approx nm + 0.5m^2$ in storage and $\approx 2nN_z + 2nm^2$ in arithmetic operations.

In the symmetric case only the last two basis vectors $v^j$ are needed when determining the projection $H_m$. The previous vectors are not even needed to determine an error bound, and can be stored on secondary storage until the Ritz vectors are computed.
The Arnoldi method is a generalization of the power method where approximations to eigenvectors are obtained from the last iterate only (or the two last iterates in case of a complex eigenvalue).
The Arnoldi method is a generalization of the power method where approximations to eigenvectors are obtained from the last iterate only (or the two last iterates in case of a complex eigenvalue).

As for the power method we therefore can expect fast convergence to the eigenvalue which is maximal in modulus and to the corresponding eigenvector.
The Arnoldi method is a generalization of the power method where approximations to eigenvectors are obtained from the last iterate only (or the two last iterates in case of a complex eigenvalue).

As for the power method we therefore can expect fast convergence to the eigenvalue which is maximal in modulus and to the corresponding eigenvector.

One can influence the convergence of the power method by shifts, either to separate the wanted eigenvalue more from the remaining spectrum or to enforce convergence to a different eigenvalue.
The Arnoldi method is a generalization of the power method where approximations to eigenvectors are obtained from the last iterate only (or the two last iterates in case of a complex eigenvalue).

As for the power method we therefore can expect fast convergence to the eigenvalue which is maximal in modulus and to the corresponding eigenvector.

One can influence the convergence of the power method by shifts, either to separate the wanted eigenvalue more from the remaining spectrum or to enforce convergence to a different eigenvalue.

The Arnoldi method is independent of shifts since

\[ \mathcal{K}_m(v^1, A) = \mathcal{K}_m(v^1, A + \alpha I) \quad \text{for all } \alpha \in \mathbb{C}. \]

Hence, we can expect convergence of the Arnoldi method to extreme eigenvalues first.
Eigenvalues (blue plus) of a random tridiagonal (100, 100) matrix and approximations (red circle) after 10 steps of Arnoldi:
Lanczos algorithm

We analyze the convergence of the Lanczos method.

1: Choose initial vector $v^1$ with $\|v^1\| = 1$
2: Set $v^0 = 0; \beta_0 = 0$
3: for $j = 1, 2, \ldots$ do
4: \[ v^{j+1} = Av^j - \beta_{j-1}v^{j-1} \]
5: \[ \alpha_j = (v^j)^Hv^{j+1} \]
6: \[ v^{j+1} = v^{j+1} - \alpha_j v^j \]
7: \[ \beta_j = \|v^{j+1}\| \]
8: \[ v^{j+1} = v^{j+1}/\beta_j \]
9: end for
Lanczos algorithm

We analyze the convergence of the Lanczos method.

1: Choose initial vector $v^1$ with $\|v^1\| = 1$
2: Set $v^0 = 0; \beta_0 = 0$
3: for $j = 1, 2, \ldots$ do
4: $v^{j+1} = Av^j - \beta_{j-1}v^{j-1}$
5: $\alpha_j = (v^j)^Hv^{j+1}$
6: $v^{j+1} = v^{j+1} - \alpha_jv^j$
7: $\beta_j = \|v^{j+1}\|$
8: $v^{j+1} = v^{j+1}/\beta_j$
9: end for

The matrix of the projected problem then is the tridiagonal matrix

$$T_m = \begin{pmatrix}
\alpha_1 & \beta_1 & 0 & \ldots & 0 \\
\beta_1 & \alpha_2 & \beta_2 & \ldots & 0 \\
& \ddots & \ddots & \ddots & \ddots \\
& \ddots & \ddots & \ddots & \ddots \\
& & & \beta_{m-1} & \alpha_m
\end{pmatrix}.$$
Monotonicity

Assume that the eigenvalues of $A$ and the Ritz values with respect to $K_m$ are ordered by magnitude:

$$
\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n, \quad \theta_1^{(m)} \leq \theta_2^{(m)} \leq \cdots \leq \theta_m^{(m)},
$$
Monotonicity

Assume that the eigenvalues of $A$ and the Ritz values with respect to $\mathcal{K}_m$ are ordered by magnitude:

$$\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n, \quad \theta_1^{(m)} \leq \theta_2^{(m)} \leq \cdots \leq \theta_m^{(m)},$$

Then the minmax principle yields ($S$ denotes a subspace of $\mathbb{C}^m$ and $\tilde{S}$ a subspace of $\mathbb{C}^n$)

$$\theta_j^{(m)} = \min_{\dim S=j} \max_{y \in S} \frac{y^H T_m y}{y^H y} = \min_{\dim S=j} \max_{y \in S} \frac{y^H V_m^H AV_m y}{y V_m^H V_m y}$$

$$= \min_{\dim \tilde{S}=j, \tilde{S} \subset \mathcal{K}_m} \max_{x \in \tilde{S}} \frac{x^H A x}{x^H x} \geq \min_{\dim \tilde{S}=j, \tilde{S} \subset \mathcal{K}_{m+1}} \max_{x \in \tilde{S}} \frac{x^H A x}{x^H x}$$

$$= \theta_j^{(m+1)} \geq \min_{\dim \tilde{S}=j} \max_{x \in \tilde{S}} \frac{x^H A x}{x^H x} = \lambda_j.$$
Monotonicity

Assume that the eigenvalues of $A$ and the Ritz values with respect to $\mathcal{K}_m$ are ordered by magnitude:

$$\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n, \quad \theta_1^{(m)} \leq \theta_2^{(m)} \leq \cdots \leq \theta_m^{(m)},$$

Then the minmax principle yields ($S$ denotes a subspace of $\mathbb{C}^m$ and $\tilde{S}$ a subspace of $\mathbb{C}^n$)

$$\theta_j^{(m)} = \min_{\dim S = j} \max_{y \in S} \frac{y^H T_m y}{y^H y} = \min_{\dim S = j} \max_{y \in S} \frac{y^H V_m^H A V_m y}{y V_m^H V_m y}$$

$$= \min_{\dim \tilde{S} = j} \max_{x \in \tilde{S}} \frac{x^H A x}{x^H x} \geq \min_{\dim \tilde{S} = j} \max_{x \in \tilde{S}} \frac{x^H A x}{x^H x}$$

$$= \theta_j^{(m+1)} \geq \min_{\dim \tilde{S} = j} \max_{x \in \tilde{S}} \frac{x^H A x}{x^H x} = \lambda_j.$$

Each (finite) sequence $\{\theta_j^{(m)}\}_{m=j,j+1,j+2}$ therefore is monotonically decreasing and bounded below by $\lambda_j$.
Likewise the sequence of the $j$-largest eigenvalues of $T_m$ are monotonically increasing and bounded above by the $j$-largest eigenvalue of $A$. 
Before getting results about the speed of convergence of the Lanczos method we first prove a bound for the angle between an eigenvector of $A$ and a Krylov space $\mathcal{K}_m(v^1, A)$. Denote by $u^i$ a system of orthonormal eigenvectors corresponding to the eigenvalues $\lambda_i$. 

**Lemma 10.1** Let $P_i$ be the orthogonal projector onto the eigenspace corresponding to $\lambda_i$. If $P_i v^1 \neq 0$ then 

$$\tan \delta(u^i, K_m) = \min_{p \in \Pi_{m-1}} \frac{\|p(A)y^i\|_2}{\|P_i v^1\|_2} \tan \delta(u^i, v^1),$$

where $y^i = \begin{cases} (I - P_i)v^1 & \|P_i v^1\|_2 \\
0 & \text{otherwise} \end{cases}$.
Before getting results about the speed of convergence of the Lanczos method we first prove a bound for the angle between an eigenvector of $A$ and a Krylov space $\mathcal{K}_m(v^1, A)$. Denote by $u^i$ a system of orthonormal eigenvectors corresponding to the eigenvalues $\lambda_i$.

**LEMMA 10.1**

Let $P_i$ be the orthogonal projector onto the eigenspace corresponding to $\lambda_i$. If $P_i v^1 \neq 0$ then

$$\tan \delta(u^i, \mathcal{K}_m) = \min_{p \in \Pi_{m-1}, p(\lambda_i)=1} \frac{\|p(A)y^i\|_2 \tan \delta(u^i, v^1)}{\|y^i\|_2},$$

where

$$y^i = \begin{cases} 
\frac{(I-P_i)v^1}{\|(I-P_i)v^1\|_2} & \text{if } (I - P_i)v^1 \neq 0 \\
0 & \text{otherwise}
\end{cases}$$
The Krylov space $\mathcal{K}_m(v^1, A)$ consists of all vectors which can be written as $x = q(A)v^1$ where $q \in \Pi_{m-1}$ is any polynomial of degree $m - 1$. 
Proof

The Krylov space $\mathcal{K}_m(v^1, A)$ consists of all vectors which can be written as $x = q(A)v^1$ where $q \in \Pi_{m-1}$ is any polynomial of degree $m - 1$. With the orthogonal decomposition

$$x = q(A)v^1 = q(A)P_i v^1 + q(A)(I - P_i)v^1$$

it holds for the angle $\delta(x, u^i)$ between $x$ and $u^i$

$$\tan \delta(x, u^i) = \frac{\|q(A)(I - P_i)v^1\|_2}{\|q(A)P_i v^1\|_2} = \frac{\|q(A)y^i\|_2}{\|q(\lambda_i)\|} \frac{\|(I - P_i)v^1\|_2}{\|P_i v^1\|_2},$$
The Krylov space $\mathcal{K}_m(v^1, A)$ consists of all vectors which can be written as $x = q(A)v^1$ where $q \in \Pi_{m-1}$ is any polynomial of degree $m - 1$.

With the orthogonal decomposition

$$x = q(A)v^1 = q(A)P_i v^1 + q(A)(I - P_i)v^1$$

it holds for the angle $\delta(x, u^i)$ between $x$ and $u^i$

$$\tan \delta(x, u^i) = \frac{\|q(A)(I - P_i)v^1\|_2}{\|q(A)P_i v^1\|_2} = \frac{\|q(A)y^i\|_2}{|q(\lambda_i)|} \frac{\|(I - P_i)v^1\|_2}{\|P_i v^1\|_2},$$

and the scaling $p(\lambda) := q(\lambda)/q(\lambda_i)$ yields

$$\tan \delta(x, u^i) = \|p(A)y^i\|_2 \tan \delta(v^1, u^i)$$

from which we get the statement by minimizing over all $x \in \mathcal{K}(v^1, A)$. □
Inserting any polynomial of degree $m - 1$ which satisfies $p(\lambda_i) = 1$ one obtains an upper bound for $\tan \delta(u_i, K_m(v^1, A))$ from the last lemma.
Inserting any polynomial of degree $m - 1$ which satisfies $p(\lambda_i) = 1$ one obtains an upper bound for $\tan \delta(u_i, K_m(v^1, A))$ from the last lemma.

We already have

**THEOREM 10.2**

Let $\alpha, \beta, \gamma \in \mathbb{R}$ with $\alpha < \beta$ and $\gamma \not\in (\alpha, \beta)$. Then the minimization problem

$$
\min_{p \in \Pi_m, \; p(\gamma) = 1} \max_{t \in [\alpha, \beta]} |p(t)|
$$

has a unique solution, and is solved by the scaled Chebyshev polynomial

$$
\tilde{c}_m(t) := \begin{cases} 
\frac{c_m(1 + 2 \frac{t-\beta}{\beta-\alpha})}{c_m(1 + 2 \frac{\gamma-\beta}{\beta-\alpha})} & \text{für } \gamma > \beta \\
\frac{c_m(1 + 2 \frac{\alpha-t}{\beta-\alpha})}{c_m(1 + 2 \frac{\alpha-\gamma}{\beta-\alpha})} & \text{für } \gamma < \alpha 
\end{cases}
$$
THEOREM 10.3

The angle $\delta(u^i, \mathcal{K}_m(v^1, A))$ between the exact eigenvector $u^i$ and the $m$-th Krylov space satisfies the inequality

$$\tan \delta(u^i, \mathcal{K}_m) \leq \frac{\kappa_i}{c_{m-i}(1 + 2\rho_i)} \tan \delta(u^i, v^1)$$

(1)

where

$$\kappa_1 = 1, \quad \kappa_i = \prod_{j=1}^{i-1} \frac{\lambda_n - \lambda_j}{\lambda_i - \lambda_j} \text{ für } i > 1$$

(2)

and

$$\rho_i = \frac{\lambda_{i+1} - \lambda_i}{\lambda_n - \lambda_{i+1}}.$$ 

(3)
THEOREM 10.3

The angle $\delta(u^i, K_m(v^1, A))$ between the exact eigenvector $u^i$ and the $m$-th Krylov space satisfies the inequality

$$\tan \delta(u^i, K_m) \leq \frac{\kappa_i}{c_{m-i}(1 + 2\rho_i)} \tan \delta(u^i, v^1)$$  \hspace{1cm} (1)

where

$$\kappa_1 = 1, \quad \kappa_i = \prod_{j=1}^{i-1} \frac{\lambda_n - \lambda_j}{\lambda_i - \lambda_j} \text{ für } i > 1 \hspace{1cm} (2)$$

and

$$\rho_i = \frac{\lambda_{i+1} - \lambda_i}{\lambda_n - \lambda_{i+1}}. \hspace{1cm} (3)$$

In particular for $i = 1$ one gets the estimate

$$\tan \delta(u^1, K_m(v^1, A)) \leq \frac{1}{c_{m-1}(1 + 2\rho_1)} \tan \delta(v^1, u^1) \text{ where } \rho_1 = \frac{\lambda_2 - \lambda_1}{\lambda_n - \lambda_2}.$$
**THEOREM 10.3**

The angle $\delta(u^i, K_m(v^1, A))$ between the exact eigenvector $u^i$ and the $m$-th Krylov space satisfies the inequality

\[
\tan \delta(u^i, K_m) \leq \frac{\kappa_i}{c_{m-i}(1 + 2\rho_i)} \tan \delta(u^i, v^1) \quad (1)
\]

where

\[
\kappa_1 = 1, \quad \kappa_i = \prod_{j=1}^{i-1} \frac{\lambda_n - \lambda_j}{\lambda_i - \lambda_j} \text{ für } i > 1 \quad (2)
\]

and

\[
\rho_i = \frac{\lambda_{i+1} - \lambda_i}{\lambda_n - \lambda_{i+1}}. \quad (3)
\]

In particular for $i = 1$ one gets the estimate

\[
\tan \delta(u^1, K_m(v^1, A)) \leq \frac{1}{c_{m-1}(1 + 2\rho_1)} \tan \delta(v^1, u^1) \quad \text{where } \rho_1 = \frac{\lambda_2 - \lambda_1}{\lambda_n - \lambda_2}.
\]

Crucial for the convergence is the distance of the two smallest eigenvalues relative to the width of the entire spectrum.
Proof

We consider the case $i = 1$ first.
Proof

We consider the case $i = 1$ first.

Expanding the vector $y^i$ in the basis $\{u^i\}$ of eigenvectors yields

$$y^i = \sum_{j=1}^{n} \alpha_j u^i, \quad \text{where} \quad \sum_{j=1}^{n} |\alpha_j|^2 = 1$$
Proof

We consider the case \( i = 1 \) first.

Expanding the vector \( y^i \) in the basis \( \{ u^i \} \) of eigenvectors yields

\[
y^i = \sum_{j=1}^{n} \alpha_j u^j,
\]

where \( \sum_{j=1}^{n} |\alpha_j|^2 = 1 \)

from which we get

\[
\|p(A)y^1\|_2 = \sum_{j=2}^{n} |p(\lambda_j)\alpha_j|^2 \leq \max_{j=2,\ldots,n} |p(\lambda_j)|^2 \leq \max_{\lambda \in [\lambda_2,\lambda_n]} |p(\lambda)|^2,
\]

and the statement follows from Theorem 10.2.
Proof ct.

For $i > 1$ we consider in Lemma 10.1 polynomials of the form

$$p(\lambda) := \frac{(\lambda - \lambda_1) \cdots (\lambda - \lambda_{i-1})}{(\lambda_i - \lambda_1) \cdots (\lambda_i - \lambda_{i-1})} q(\lambda)$$

with $q \in \Pi_{m-i}$ and $q(\lambda_i) = 1$. 
For $i > 1$ we consider in Lemma 10.1 polynomials of the form

$$p(\lambda) := \frac{(\lambda - \lambda_1) \cdots (\lambda - \lambda_{i-1})}{(\lambda_i - \lambda_1) \cdots (\lambda_i - \lambda_{i-1})} q(\lambda)$$

with $q \in \Pi_{m-i}$ and $q(\lambda_i) = 1$.

Then one gets as before

$$\|p(A)y^i\|_2 \leq \max_{\lambda \in [\lambda_{i+1}, \lambda_n]} \left| \prod_{j=1}^{i-1} \frac{\lambda - \lambda_j}{\lambda_i - \lambda_j} q(\lambda) \right| \leq \prod_{j=1}^{i-1} \frac{\lambda_n - \lambda_j}{\lambda_i - \lambda_j} \max_{\lambda \in [\lambda_{i+1}, \lambda_n]} |q(\lambda)|.$$
Proof ct.

For $i > 1$ we consider in Lemma 10.1 polynomials of the form

$$p(\lambda) := \frac{(\lambda - \lambda_1) \cdots (\lambda - \lambda_{i-1})}{(\lambda_i - \lambda_1) \cdots (\lambda_i - \lambda_{i-1})} q(\lambda)$$

with $q \in \Pi_{m-i}$ and $q(\lambda_i) = 1$.

Then one gets as before

$$\|p(A)y^i\|_2 \leq \max_{\lambda \in [\lambda_{i+1}, \lambda_n]} \left| \prod_{j=1}^{i-1} \frac{\lambda - \lambda_j}{\lambda_i - \lambda_j} q(\lambda) \right| \leq \prod_{j=1}^{i-1} \frac{\lambda_n - \lambda_j}{\lambda_i - \lambda_j} \max_{\lambda \in [\lambda_{i+1}, \lambda_n]} |q(\lambda)|.$$ 

The result follows by minimizing this expression over all polynomials $q$ satisfying the constraint $q(\lambda_i) = 1$. □
Let $A \in \mathbb{C}^{n \times n}$ be Hermitian with eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ and corresponding orthonormal eigenvectors $u^1, \ldots, u^n$. 
THEOREM 10.4 (Kaniel & Paige; 1. eigenvalue)

Let $A \in \mathbb{C}^{n \times n}$ be Hermitean with eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ and corresponding orthonormal eigenvectors $u^1, \ldots, u^n$.

If $\theta_1^{(m)} \leq \cdots \leq \theta_m^{(m)}$ denote the eigenvalues of the matrix $T_m$ obtained after $m$ steps of Lanczos’ method, then

$$0 \leq \theta_1^{(m)} - \lambda_1 \leq (\lambda_n - \lambda_1) \left( \frac{\tan \delta(u^1, v^1)}{c_{m-1}(1 + 2\rho_1)} \right)^2,$$

where $\rho_1 = (\lambda_2 - \lambda_1)/(\lambda_n - \lambda_2)$. 
THEOREM 10.4 (Kaniel & Paige; 1. eigenvalue)

Let \( A \in \mathbb{C}^{n \times n} \) be Hermitian with eigenvalues \( \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n \) and corresponding orthonormal eigenvectors \( u^1, \ldots, u^n \).

If \( \theta_1^{(m)} \leq \cdots \leq \theta_m^{(m)} \) denote the eigenvalues of the matrix \( T_m \) obtained after \( m \) steps of Lanczos’ method, then

\[
0 \leq \theta_1^{(m)} - \lambda_1 \leq (\lambda_n - \lambda_1) \left( \frac{\tan \delta(u^1, v^1)}{c_{m-1}(1 + 2\rho_1)} \right)^2,
\]

where \( \rho_1 = (\lambda_2 - \lambda_1)/(\lambda_n - \lambda_2) \).

Crucial for the speed of convergence is the growth of \( c_{j-1}(1 + 2\rho_1) \), i.e. the separation of the first two eigenvalues relative to the width of the entire spectrum of \( A \).
The left inequality follows from Rayleigh’s principle.
Proof

The left inequality follows from Rayleigh’s principle.

We have

\[ \theta_1^{(m)} = \min_{x \in \mathcal{K}_m(v^1, A), x \neq 0} \frac{x^H A x}{x^H x}, \]

and, since each \( x \in \mathcal{K}_m(v^1, A) \) can be represented as \( x = q(A) v^1 \) for some \( q \in \Pi_{m-1} \), it follows

\[ \theta_1^{(m)} - \lambda_1 = \min_{x \in \mathcal{K}_m(v^1, A), x \neq 0} \frac{x^H (A - \lambda_1 I) x}{x^H x} \]
\[ = \min_{q \in \Pi_{m-1}, q \neq 0} \frac{(v^1)^H q(A)^H (A - \lambda_1 I) q(A) v^1}{(v^1)^H q(A)^2 v^1}. \]
Proof ct.

With $v^1 = \sum_{j=1}^{n} \alpha_j u^j$ it holds

$$\theta_1^{(m)} - \lambda_1 = \min_{q \in \Pi_{m-1}, q \neq 0} \frac{\sum_{j=2}^{n} (\lambda_j - \lambda_1) |\alpha_j q(\lambda_j)|^2}{\sum_{j=1}^{n} |\alpha_j q(\lambda_j)|^2}$$

$$\leq (\lambda_n - \lambda_1) \min_{q \in \Pi_{m-1}, q \neq 0} \frac{\sum_{j=2}^{n} |\alpha_j q(\lambda_j)|^2}{|\alpha_1 q(\lambda_1)|^2}$$

$$\leq (\lambda_n - \lambda_1) \min_{q \in \Pi_{m-1}, q \neq 0} \max_{j=2, \ldots, n} \frac{|q(\lambda_j)|^2}{|\alpha_1 q(\lambda_1)|^2} \cdot \frac{\sum_{j=2}^{n} |\alpha_j|^2}{|\alpha_1|^2}$$
With $v^1 = \sum_{j=1}^n \alpha_j u_j^j$ it holds

$$\theta_1^{(m)} - \lambda_1 = \min_{q \in \Pi_{m-1}, q \neq 0} \frac{\sum_{j=2}^n (\lambda_j - \lambda_1)|\alpha_j q(\lambda_j)|^2}{\sum_{j=1}^n |\alpha_j q(\lambda_j)|^2}$$

$$\leq (\lambda_n - \lambda_1) \min_{q \in \Pi_{m-1}, q \neq 0} \frac{\sum_{j=2}^n |\alpha_j q(\lambda_j)|^2}{|\alpha_1 q(\lambda_1)|^2}$$

$$\leq (\lambda_n - \lambda_1) \min_{q \in \Pi_{m-1}, q \neq 0} \max_{j=2,\ldots,n} \frac{|q(\lambda_j)|^2}{|q(\lambda_1)|^2} \cdot \frac{\sum_{j=2}^n |\alpha_j|^2}{|\alpha_1|^2}$$

Defining $p(\lambda) = q(\lambda)/q(\lambda_1)$, and observing that the set of all $p$'s when $q$ passes through the set $\Pi_{m-1}$ is the set of all polynomials of degree not exceeding $m - 1$ and satisfying the constraint $p(\lambda_1) = 1$ we get

$$\theta_1^{(m)} - \lambda_1 \leq (\lambda_n - \lambda_1) \tan^2 \delta(u^1, v^1) \min_{p \in \Pi_{m-1}, p(\lambda_1) = 1} \max_{\lambda \in [\lambda_2, \lambda_n]} |p(\lambda)|^2$$

and the statement follows from Theorem 10.2.
Example

Matrix $A$ has eigenvalue $\lambda_1 = 1$ and 99 eigenvalues uniformly distributed in $[\alpha, \beta]$.

<table>
<thead>
<tr>
<th>it.</th>
<th>$[\alpha, \beta] = [20, 100]$</th>
<th>$[\alpha, \beta] = [2, 100]$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>error</td>
<td>bound</td>
</tr>
<tr>
<td>2</td>
<td>$3.3890e+001$</td>
<td>$3.4832e+003$</td>
</tr>
<tr>
<td>3</td>
<td>$1.9708e+001$</td>
<td>$1.0090e+002$</td>
</tr>
<tr>
<td>4</td>
<td>$6.6363e+000$</td>
<td>$1.5083e+001$</td>
</tr>
<tr>
<td>5</td>
<td>$1.5352e+000$</td>
<td>$2.2440e+000$</td>
</tr>
<tr>
<td>10</td>
<td>$7.9258e-005$</td>
<td>$1.6293e-004$</td>
</tr>
<tr>
<td>15</td>
<td>$4.3730e-009$</td>
<td>$1.1827e-008$</td>
</tr>
<tr>
<td>20</td>
<td>$2.9843e-013$</td>
<td>$8.5861e-013$</td>
</tr>
<tr>
<td>25</td>
<td>$1.4843e-002$</td>
<td>$3.7876e+000$</td>
</tr>
<tr>
<td>30</td>
<td>$1.6509e-003$</td>
<td>$3.6109e-001$</td>
</tr>
<tr>
<td>35</td>
<td>$2.0658e-004$</td>
<td>$3.4423e-002$</td>
</tr>
<tr>
<td>40</td>
<td>$9.5770e-007$</td>
<td>$3.1285e-004$</td>
</tr>
<tr>
<td>45</td>
<td>$8.0487e-012$</td>
<td>$2.8432e-006$</td>
</tr>
<tr>
<td>55</td>
<td>$3.1530e-014$</td>
<td>$2.7105e-007$</td>
</tr>
<tr>
<td>60</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
THEOREM 10.5 (Kaniel & Paige; higher eigenvalues)

Under the conditions of Theorem 10.4 it holds

\[ 0 \leq \theta_j^{(m)} - \lambda_j \leq (\lambda_n - \lambda_1) \left( \frac{\kappa_j^{(m)} \tan \delta(v^1, u^i)}{c_{m-j}(1 + 2\rho_j)} \right)^2 \]

with

\[ \rho_j = \frac{(\lambda_{j+1} - \lambda_j)}{(\lambda_n - \lambda_{j+1})}, \]

and

\[ \kappa_1^{(m)} \equiv 1, \quad \kappa_j^{(m)} = \prod_{i=1}^{j-1} \frac{\lambda_n - \theta_i^{(m)}}{\lambda_j - \theta_i^{(m)}}. \]
THEOREM 10.5 (Kaniel & Paige; higher eigenvalues)

Under the conditions of Theorem 10.4 it holds

\[ 0 \leq \theta_j^{(m)} - \lambda_j \leq (\lambda_n - \lambda_1) \left( \frac{\kappa_j^{(m)} \tan \delta(v^1, u^i)}{c_{m-j}(1 + 2\rho_j)} \right)^2 \]

with

\[ \rho_j = (\lambda_{j+1} - \lambda_j)/(\lambda_n - \lambda_{j+1}), \]

and

\[ \kappa_1^{(m)} \equiv 1, \quad \kappa_j^{(m)} = \prod_{i=1}^{j-1} \frac{\lambda_n - \theta_i^{(m)}}{\lambda_j - \theta_i^{(m)}}. \]

The general case \( j > 1 \) can be proved by using the maxmin characterization of \( \theta_j^{(m)} \) of Courant and Fischer.
Under the conditions of Theorem 10.4 it holds

\[ 0 \leq \theta_j^{(m)} - \lambda_j \leq (\lambda_n - \lambda_1) \left( \frac{\kappa_j^{(m)} \tan \delta(v^1, u^i)}{c_{m-j}(1 + 2 \rho_j)} \right)^2 \]

with

\[ \rho_j = (\lambda_{j+1} - \lambda_j)/(\lambda_n - \lambda_{j+1}), \]

and

\[ \kappa_1^{(m)} \equiv 1, \quad \kappa_j^{(m)} = \prod_{i=1}^{j-1} \frac{\lambda_n - \theta_i^{(m)}}{\lambda_j - \theta_i^{(m)}}. \]

The general case \( j > 1 \) can be proved by using the maxmin characterization of \( \theta_j^{(m)} \) of Courant and Fischer.

Analogous results hold for the largest eigenvalues and Ritz values.
For nonsymmetric matrices and the Arnoldi process the speed of convergence was analyzed by Saad (1983).
For nonsymmetric matrices and the Arnoldi process the speed of convergence was analyzed by Saad (1983).

This was done by considering the distance of a particular eigenvector $u^1$ of $A$ from the subspace $\mathcal{K}_m(v^1, A)$.

\[ \epsilon(m) := \min_{p \in \Pi_{m-1}} \max_{\lambda \in \sigma(A)} \{ |p(\lambda)| \} \]

where $\sigma(A)$ is the spectrum of $A$, and $\Pi_{m-1}$ denotes the set of polynomials of maximum degree $m-1$ such that $p(\lambda_1) = 1$.

The following lemma relates the distance of this quantity to $\| (I - P_m) u^1 \|$ where $P_m$ denotes the projector onto $\mathcal{K}_m(v^1, A)$. 
Convergence of Arnoldi Method

For nonsymmetric matrices and the Arnoldi process the speed of convergence was analyzed by Saad (1983).

This was done by considering the distance of a particular eigenvector $u^1$ of $A$ from the subspace $\mathcal{K}_m(v^1, A)$.

Let

$$
\varepsilon^{(m)} := \min_{p \in \Pi^{*}_{m-1}} \max_{\lambda \in \sigma(A) \setminus \{\lambda_1\}} |p(\lambda)|
$$

where $\sigma(A)$ is the spectrum of $A$, and $\Pi^{*}_{m-1}$ denotes the set of polynomials of maximum degree $m - 1$ such that $p(\lambda_1) = 1$. 
For nonsymmetric matrices and the Arnoldi process the speed of convergence was analyzed by Saad (1983).

This was done by considering the distance of a particular eigenvector $u^1$ of $A$ from the subspace $\mathcal{K}_m(v^1, A)$.

Let

$$\varepsilon^{(m)} := \min_{p \in \Pi^*_{m-1}} \max_{\lambda \in \sigma(A) \setminus \{\lambda_1\}} |p(\lambda)|$$

where $\sigma(A)$ is the spectrum of $A$, and $\Pi^*_{m-1}$ denotes the set of polynomials of maximum degree $m - 1$ such that $p(\lambda_1) = 1$.

The following lemma relates the distance of this quantity to $\| (I - P_m)u^1 \|$ where $P_m$ denotes the projector onto $\mathcal{K}_m(v^1, A)$. 
LEMMA 10.6
Assume that $A$ is diagonalizable and that the initial vector $v^1$ of Arnoldi’s method has the expansion $v^1 = \sum_{j=1}^{n} \alpha_j u^j$ with respect to an eigenbasis $u^1, \ldots, u^n$ of $A$ where $\|u^j\| = 1$ and $\alpha_1 \neq 0$. Then the following inequality holds

$$\|(I - P_m)u^1\| \leq \xi \varepsilon^{(m)}$$

where $\xi = \sum_{j=2}^{n} |\alpha_j|/|\alpha_1|$. 

Hence, upper bounds of $\varepsilon^{(m)}$ estimate the speed of convergence of the Arnoldi method.
LEMMA 10.6
Assume that $A$ is diagonalizable and that the initial vector $\nu^1$ of Arnoldi’s method has the expansion $\nu^1 = \sum_{j=1}^{n} \alpha_j u^j$ with respect to an eigenbasis $u^1, \ldots, u^n$ of $A$ where $\|u^j\| = 1$ and $\alpha_1 \neq 0$. Then the following inequality holds

$$\|(I - P_m)u^1\| \leq \xi \varepsilon^{(m)}$$

where $\xi = \sum_{j=2}^{n} |\alpha_j|/|\alpha_1|$. Hence, upper bounds of $\varepsilon^{(m)}$ estimate the speed of convergence of the Arnoldi method.
LEMMA 10.6
Assume that $A$ is diagonalizable and that the initial vector $v^1$ of Arnoldi’s method has the expansion $v^1 = \sum_{j=1}^{n} \alpha_j u^j$ with respect to an eigenbasis $u^1, \ldots, u^n$ of $A$ where $\|u^i\| = 1$ and $\alpha_1 \neq 0$. Then the following inequality holds

$$\|(I - P_m)u^1\| \leq \xi \varepsilon^{(m)}$$

where $\xi = \sum_{j=2}^{n} |\alpha_j| / |\alpha_1|$.

Hence, upper bounds of $\varepsilon^{(m)}$ estimate the speed of convergence of the Arnoldi method.

THEOREM 10.7
Assume that all eigenvalues of $A$ but $\lambda_1$ are lying in an ellipse with center $c$, focal points $c - e$ and $c + e$ and large semiaxis $a$. Then it holds

$$\varepsilon^{(m)} \leq \frac{c_{m-1}(\frac{a}{c})}{|c_{m-1}(\frac{\lambda_1 - c}{e})|},$$

where $c_{m-1}$ denotes the Chebyshev polynomial of degree $m - 1$. 


Convergence of Arnoldi Method ct.

**LEMMA 10.6**
Assume that $A$ is diagonalizable and that the initial vector $v^1$ of Arnoldi’s method has the expansion $v^1 = \sum_{j=1}^{n} \alpha_j u^j$ with respect to an eigenbasis $u^1, \ldots, u^n$ of $A$ where $\|u^j\| = 1$ and $\alpha_1 \neq 0$. Then the following inequality holds

$$\| (I - P_m) u^1 \| \leq \xi \varepsilon^{(m)}$$

where $\xi = \sum_{j=2}^{n} |\alpha_j| / |\alpha_1|$.

Hence, upper bounds of $\varepsilon^{(m)}$ estimate the speed of convergence of the Arnoldi method.

**THEOREM 10.7**
Assume that all eigenvalues of $A$ but $\lambda_1$ are lying in an ellipse with center $c$, focal points $c - e$ and $c + e$ and large semiaxis $a$. Then it holds

$$\varepsilon^{(m)} \leq \frac{c_{m-1}(\frac{a}{c})}{|c_{m-1}(\frac{\lambda_1 - c}{e})|},$$

where $c_{m-1}$ denotes the Chebyshev polynomial of degree $m - 1$.

The relative difference between the right and left hand side converges to 0.
After a Ritz pair $(\theta, y)$ has been determined, the approximation $y$ to the eigenvector can be improved solving the optimization problem

\[ \|Az - \theta z\|_2 = \min, \ z \in K_m(v^1, A), \ |z|_2 = 1, \]

This improvement was introduced by Jia (1997) and was called refined Ritz vector although a solution in general is not a Ritz vector corresponding to $\theta$. Given a Ritz pair the refined Ritz vector can be obtained from the augmented Hessenberg matrix

\[
\begin{pmatrix}
 h_{11} & h_{12} & h_{13} & \cdots & h_{1m} \\
 h_{21} & h_{22} & h_{23} & \cdots & h_{2m} \\
 0 & h_{32} & h_{33} & \cdots & h_{3m} \\
 \vdots & \vdots & \vdots & \ddots & \vdots \\
 & & & & h_{m,m-1} \\
 & & & & h_{m,m} \\
\end{pmatrix}
\in \mathbb{R}^{(m+1) \times m}
\]
Refined Ritz vector

After a Ritz pair \((\theta, y)\) has been determined, the approximation \(y\) to the eigenvector can be improved solving the optimization problem

\[
\|Az - \theta z\|_2 = \min!, \; z \in \mathcal{K}_m(v^1, A), \; \|z\|_2 = 1,
\]

This improvement was introduced by Jia (1997) and was called refined Ritz vector although a solution in general is not a Ritz vector corresponding to \(\theta\).
After a Ritz pair \((\theta, y)\) has been determined, the approximation \(y\) to the eigenvector can be improved solving the optimization problem

\[
\|Az - \theta z\|_2 = \min!, \ z \in K_m(v^1, A), \|z\|_2 = 1
\]

This improvement was introduced by Jia (1997) and was called refined Ritz vector although a solution in general is not a Ritz vector corresponding to \(\theta\).

Given a Ritz pair the refined Ritz vector can be obtained from the augmented Hessenberg matrix

\[
\tilde{H}_m = \begin{pmatrix}
h_{11} & h_{12} & h_{13} & \ldots & \ldots & h_{1m} \\
h_{21} & h_{22} & h_{23} & \ldots & \ldots & h_{2m} \\
0 & h_{32} & h_{33} & \ldots & \ldots & h_{3m} \\
\vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\
h_{m,m-1} & h_{mm} & h_{m+1,m} \\
\end{pmatrix} \in \mathbb{R}^{(m+1) \times m}
\]
$z \in \mathcal{K}_m(v^1, A)$ can be written as $z = V_m t$, and $\|z\|_2 = 1$ holds if and only if $\|t\|_2 = 1$. 
Refined Ritz vector ct.

\( z \in \mathcal{K}_m(v^1, A) \) can be written as \( z = V_m t \), and \( \|z\|_2 = 1 \) holds if and only if \( \|t\|_2 = 1 \).

Hence,

\[
\| (AV_m - \theta V_m) t \|_2 = \| (V_{m+1} \tilde{H}_m - \theta V_m) t \|_2 \\
= \| V_{m+1} (\tilde{H}_m - \theta I_{m+1,m}) t \|_2 \\
= \| (\tilde{H}_m - \theta I_{m+1,m}) t \|_2
\]

and this expression attains its minimum under the constraint \( \|t\|_2 = 1 \) for the right singular vector of \( \tilde{H}_m - \theta I_{m+1,m} \) corresponding to the smallest singular value.
In exact arithmetic the Lanczos method generates an orthonormal basis of the Krylov space $\mathcal{K}_m(v^1, A)$. In the algorithm only the orthogonality with respect to two basis vectors $v^j$ and $v^{j-1}$ obtained in the last two steps is enforced, with respect to the previous $v^i$:s it follows from the symmetry of $A$. 
Orthogonality of basis vectors

In exact arithmetic the Lanczos method generates an orthonormal basis of the Krylov space \( \mathcal{K}_m(v^1, A) \). In the algorithm only the orthogonality with respect to two basis vectors \( v^j \) and \( v^{j-1} \) obtained in the last two steps is enforced, with respect to the previous \( v^i \)'s it follows from the symmetry of \( A \).

It can be shown that in floating point arithmetic the orthogonality is destroyed when a sequence \( \theta^{(j)}_i, j = 1, 2, \ldots \), of Ritz values has converged to an eigenvalue \( \tilde{\lambda} \) of \( A \), i.e. if the residual \( \beta_j s^{(j)}_{j,i} \) has become small.
Orthogonality of basis vectors

In exact arithmetic the Lanczos method generates an orthonormal basis of the Krylov space $K_m(v^1, A)$. In the algorithm only the orthogonality with respect to two basis vectors $v^j$ and $v^{j-1}$ obtained in the last two steps is enforced, with respect to the previous $v^i$:s it follows from the symmetry of $A$.

It can be shown that in floating point arithmetic the orthogonality is destroyed when a sequence $\theta_{i}^{(j)}$, $j = 1, 2, \ldots$, of Ritz values has converged to an eigenvalue $\tilde{\lambda}$ of $A$, i.e. if the residual $\beta_j s_{j,i}^{(j)}$ has become small.

Thereafter all $v^i$:s obtain a component in the direction of the eigenspace of the converged eigenvalue, and a duplicate copy of that eigenvalue will show up in the spectrum of the tridiagonal matrix $T_m$. 

Orthogonality of basis vectors

In exact arithmetic the Lanczos method generates an orthonormal basis of the Krylov space $\mathcal{K}_m(v^1, A)$. In the algorithm only the orthogonality with respect to two basis vectors $v^j$ and $v^{j-1}$ obtained in the last two steps is enforced, with respect to the previous $v^i$:s it follows from the symmetry of $A$.

It can be shown that in floating point arithmetic the orthogonality is destroyed when a sequence $\theta_i^{(j)}, j = 1, 2, \ldots$, of Ritz values has converged to an eigenvalue $\tilde{\lambda}$ of $A$, i.e. if the residual $\beta_j s_j^{(j), i}$ has become small.

Thereafter all $v^j$:s obtain a component in the direction of the eigenspace of the converged eigenvalue, and a duplicate copy of that eigenvalue will show up in the spectrum of the tridiagonal matrix $T_m$.

This effect was first observed and studied by Paige (1971). A detailed discussion is contained in the monograph of Parlett (1998).
Orthogonality of basis vectors ct.

Note that these multiple Ritz values have nothing to do with possible multiple eigenvalues of a given matrix, they occur simply as a result of a converged eigenvalue.
Orthogonality of basis vectors ct.

Note that these multiple Ritz values have nothing to do with possible multiple eigenvalues of a given matrix, they occur simply as a result of a converged eigenvalue.

For multiple eigenvalues the Lanczos (and Arnoldi) method in exact arithmetic can only detect one eigenvector, namely the projection of the initial vector $v^1$ to the corresponding eigenspace. Further eigenvectors can only be obtained restarting with a different initial vector or by a block Lanczos (Arnoldi) method.
Orthogonality of basis vectors ct.

Note that these multiple Ritz values have nothing to do with possible multiple eigenvalues of a given matrix, they occur simply as a result of a converged eigenvalue.

For multiple eigenvalues the Lanczos (and Arnoldi) method in exact arithmetic can only detect one eigenvector, namely the projection of the initial vector $v^1$ to the corresponding eigenspace. Further eigenvectors can only be obtained restarting with a different initial vector or by a block Lanczos (Arnoldi) method.

A simple trick to detect duplicate copies of an eigenvalue advocated by Cullum & Willoughby (1986) is the following.
Orthogonality of basis vectors ct.

Note that these multiple Ritz values have nothing to do with possible multiple eigenvalues of a given matrix, they occur simply as a result of a converged eigenvalue.

For multiple eigenvalues the Lanczos (and Arnoldi) method in exact arithmetic can only detect one eigenvector, namely the projection of the initial vector $v^1$ to the corresponding eigenspace. Further eigenvectors can only be obtained restarting with a different initial vector or by a block Lanczos (Arnoldi) method.

A simple trick to detect duplicate copies of an eigenvalue advocated by Cullum & Willoughby (1986) is the following.

Compute the eigenvalues of the reduced matrix $\hat{T}_m \in \mathbb{R}^{m-1 \times m-1}$ obtained from $T_m$ by detracting the first row and column. Those eigenvalues that differ less than a small multiple times machine precision from the eigenvalues of $T_m$ are the unwanted eigenvalues, i.e. the ones due to loss of orthogonality.
Convergence of the Lanczos process for $A = \text{diag} \ (\text{rand}(100,1))$
Complete reorthogonalization

In each step the new vector $\nu^{i+1}$ is reorthogonalized against all previous vectors $\nu^i$. 

$$\tilde{\nu}^{i+1} = \nu^{i+1} - \nu^i \nu^i H \nu^{i+1}.$$ 

If the norm is decreased by a nontrivial amount, say $\|\tilde{\nu}^{i+1}\| < \sqrt{2} \|\nu^{i+1}\|$, the reorthogonalization will have to be repeated.

Complete reorthogonalization is very reliable, but very expensive.
In each step the new vector $\nu^{j+1}$ is reorthogonalized against all previous vectors $\nu^i$.

With classical Gram–Schmidt this means: $\nu^{j+1}$ is replaced by

$$\tilde{\nu}^{j+1} = \nu^{j+1} - V_j V_j^H \nu^{j+1}.$$
Complete reorthogonalization

In each step the new vector \( \nu^{i+1} \) is reorthogonalized against all previous vectors \( \nu^i \).

With classical Gram–Schmidt this means: \( \nu^{i+1} \) is replaced by

\[
\tilde{\nu}^{i+1} = \nu^{i+1} - V_j V_j^H \nu^{i+1}.
\]

If the norm is decreased by a nontrivial amount, say

\[
\|\tilde{\nu}^{i+1}\| < \frac{1}{\sqrt{2}} \|\nu^{i+1}\|,
\]

the reorthogonalization will have to be repeated.
Complete reorthogonalization

In each step the new vector $\nu^{j+1}$ is reorthogonalized against all previous vectors $\nu^j$.

With classical Gram–Schmidt this means: $\nu^{j+1}$ is replaced by

$$\tilde{\nu}^{j+1} = \nu^{j+1} - V_j V_j^H \nu^{j+1}.$$ 

If the norm is decreased by a nontrivial amount, say

$$\|\tilde{\nu}^{j+1}\| < \frac{1}{\sqrt{2}} \|\nu^{j+1}\|,$$

the reorthogonalization will have to be repeated.

Complete reorthogonalization is very reliable, but very expensive.
1: Choose initial vector $\mathbf{v}^1$ with $||\mathbf{v}^1|| = 1$
2: Set $\mathbf{v}^0 = 0; \beta_0 = 0; V = [v^1]$
3: for $j = 1, 2, \ldots$ do
4: $\mathbf{v}^{j+1} = A\mathbf{v}^j - \beta_{j-1}\mathbf{v}^{j-1}$
5: $\alpha_j = (\mathbf{v}^j)^H\mathbf{v}^{j+1}$
6: $\mathbf{v}^{j+1} = \mathbf{v}^{j+1} - \alpha_j\mathbf{v}^j$
7: $\mathbf{v}^{j+1} = \mathbf{v}^{j+1} - VV^H\mathbf{v}^{j+1}$
8: $\beta_j = ||\mathbf{v}^{j+1}||$
9: $\mathbf{v}^{j+1} = \mathbf{v}^{j+1}/\beta_j$
10: $V = [V, \mathbf{v}^{j+1}]$
11: Solve projected eigenproblem $T_j s = \theta s$
12: Test for convergence
13: end for
Example

Convergence of the Lanczos process with complete reorthogonalization for $A = \text{diag}(\text{rand}(100,1))$
THEOREM 10.8 (Paige)

Let $V_k = [v^1, \ldots, v^k]$ be the matrix of vectors actually obtained in the Lanczos algorithm, $\Theta_k = \text{diag}\{\theta_1, \ldots, \theta_k\}$ and $S_k = [s^1, \ldots, s^k]$ such that $T_k S_k = S_k \Theta_k$ and $S_k^H S_k = I_k$. Then it holds:

$$(y_k, i)_H v^k + 1 = O(\varepsilon \| A \|_2 \beta_k |s_k, i|).$$

Hence, the component $(y_k, i)_H v^k + 1$ of the computed Lanczos vector $v^k + 1$ in the direction of the Ritz vector $y_k, i$ is proportional to the reciprocal of the error bound for the Ritz value $\theta_i$. 

TUHH Heinrich Voss
Krylov subspace methods
Summer School 2006
THEOREM 10.8 (Paige)

Let $V_k = [v^1, \ldots, v^k]$ be the matrix of vectors actually obtained in the Lanczos algorithm, $\Theta_k = \text{diag}\{\theta_1, \ldots, \theta_k\}$ and $S_k = [s^1, \ldots, s^k]$ such that $T_k S_k = S_k \Theta_k$ and $S_k^H S_k = I_k$.

Let $y^{k,i} = V_k s^i$ be the corresponding Ritz vectors. Then it holds

$$(y^{k,i})^H v^{k+1} = \frac{O(\varepsilon \|A\|_2)}{\beta_k |s_{k,i}|}.$$
THEOREM 10.8 (Paige)

Let \( V_k = [v^1, \ldots, v^k] \) be the matrix of vectors actually obtained in the Lanczos algorithm, \( \Theta_k = \text{diag}\{\theta_1, \ldots, \theta_k\} \) and \( S_k = [s^1, \ldots, s^k] \) such that \( T_k S_k = S_k \Theta_k \) and \( S_k^H S_k = I_k \).

Let \( y^{k,i} = V_k s^i \) be the corresponding Ritz vectors. Then it holds

\[
(y^{k,i})^H v^{k+1} = \frac{O(\varepsilon \|A\|_2)}{\beta_k |s_{k,i}|}.
\]

Hence, the component \( (y^{k,i})^H v^{k+1} \) of the computed Lanczos vector \( v^{k+1} \) in the direction of the Ritz vector \( y^{k,i} \) is proportional to the reciprocal of the error bound for the Ritz value \( \theta_i \).
Selective reorthogonalization

By Paige’s theorem the $v^j : s$ lose orthogonality since the vector $v^{j+1}$ obtained in the final step has a large component with respect to the Ritz vector $y = [v^1, \ldots, v^j] * s$ corresponding to the converged Ritz value $\theta$ (measured by the error bound $\beta_j |s_j|$).

This suggests to monitor the error bounds $\beta_j |s_j|$ for all eigenvectors $s$ of $T_j$ in every iteration step, and to reorthogonalize $v^{j+1}$ against the Ritz vector $y$:

$$v^{j+1} = v^{j+1} - (y^H v^{j+1}) y.$$
Selective reorthogonalization

By Paige’s theorem the \( \nu^j : s \) lose orthogonality since the vector \( \nu^{j+1} \) obtained in the final step has a large component with respect to the Ritz vector 
\[ y = [\nu^1, \ldots, \nu^j] \star s \] 
corresponding to the converged Ritz value \( \theta \) (measured by the error bound \( \beta_j |s_j| \)).

This suggests to monitor the error bounds \( \beta_j |s_j| \) for all eigenvectors \( s \) of \( T_j \) in every iteration step, and to reorthogonalize \( \nu^{j+1} \) against the Ritz vector \( y \):
\[
\nu^{j+1} = \nu^{j+1} - (y^H \nu^{j+1})y.
\]
Selective reorthogonalization

By Paige’s theorem the \( v^j : s \) lose orthogonality since the vector \( v^{j+1} \) obtained in the final step has a large component with respect to the Ritz vector \( y = [v^1, \ldots, v^j] \times s \) corresponding to the converged Ritz value \( \theta \) (measured by the error bound \( \beta_j |s_j| \)).

This suggests to monitor the error bounds \( \beta_j |s_j| \) for all eigenvectors \( s \) of \( T_j \) in every iteration step, and to reorthogonalize \( v^{j+1} \) against the Ritz vector \( y \):

\[
v^{j+1} = v^{j+1} - (y^H v^{j+1}) y.
\]

This so called selective reorthogonalization is applied if

\[
\beta_j |s_j| < \sqrt{\varepsilon} \| T_j \|
\]

(actually \( \| A \| \) would have been needed on the right hand side, but \( \| A \| \) is not available).
Selective reorthogonalization

1: Choose initial vector $v^1$ with $\|v^1\| = 1$
2: Set $v^0 = 0; \beta_0 = 0; V = [v^1]$
3: for $j = 1, 2, \ldots$ do
   4: $v^{j+1} = Av^j - \beta_{j-1}v^{j-1}$
   5: $\alpha_j = (v^j)^Hv^{j+1}$
   6: $v^{j+1} = v^{j+1} - \alpha_j v^j$
   7: $\beta_j = \|v^{j+1}\|$ 
   8: Solve tridiag $\{\beta_{i-1}, \alpha_i, \beta_i\}S = S\Theta$
   9: for $i = 1, \ldots, j$ do
      10: if $|\beta_j s_j^{(i)}| < \sqrt{\varepsilon} \max(\text{diag}\Theta)$ then
         11: $y = [v^1, \ldots, v^j]\text{s}$
         12: $v^{j+1} = v^{j+1} - (y^Hv^{j+1})y$
      13: end if
   14: end for
   15: $\beta_j = \|v^{j+1}\|$
   16: $v^{j+1} = v^{j+1}/\beta_j$
   17: $V = [V, v^{j+1}]$
18: end for
It can be shown (Simon (1984)) that the properties of the Lanczos method are widely retained as long as the basis is semiorthogonal, i.e.

\[ V_j^H V_j = I_j + E \quad \text{with} \quad \|E\|_2 = \sqrt{\varepsilon}, \]

where \( \varepsilon \) denotes the rounding unit.
Partial reorthogonalization

It can be shown ([Simon (1984)]) that the properties of the Lanczos method are widely retained as long as the basis is semiorthogonal, i.e.

\[ V_j^H V_j = I_j + E \quad \text{with} \quad \| E \|_2 = \sqrt{\varepsilon}, \]

where \( \varepsilon \) denotes the rounding unit.

If the tridiagonal matrix \( T_j \) is determined using a semiorthogonal basis \( V_j \) then there exists an orthonormal basis \( N_j \) of span \( V_j \) such that

\[ T_j = V_j^H A V_j = N_j^H A N_j + G \quad \text{with} \quad \| G \|_2 = O(\varepsilon \| A \|_2). \]
It can be shown (Simon (1984)) that the properties of the Lanczos method are widely retained as long as the basis is semiorthogonal, i.e.

\[ V_j^H V_j = I_j + E \quad \text{with} \quad \|E\|_2 = \sqrt{\varepsilon}, \]

where \(\varepsilon\) denotes the rounding unit.

If the tridiagonal matrix \(T_j\) is determined using a semiorthogonal basis \(V_j\) then there exists an orthonormal basis \(N_j\) of span \(V_j\) such that

\[ T_j = V_j^H AV_j = N_j^H AN_j + G \quad \text{with} \quad \|G\|_2 = O(\varepsilon \|A\|_2). \]

Hence, the eigenvalues of the problem projected to span \(V_j\) are obtained with full precision.
Partial reorthogonalization

1: Choose initial vector $v^1$ with $\|v^1\| = 1$
2: Set $v^0 = 0; \beta_0 = 0; V = [v^1]$
3: for $j = 1, 2, \ldots$ do
4: $v^{j+1} = Av^j - \beta_{j-1}v^{j-1}$
5: $\alpha_j = (v^j)^H v^{j+1}$
6: $v^{j+1} = v^{j+1} - \alpha_j v^j$
7: $\beta_j = \|v^{j+1}\|$
8: $v^{j+1} = v^{j+1} / \beta_j$
9: if $\| [V, v^{j+1}]^H [V, v^{j+1}] - I_{j+1} \| > \sqrt{\varepsilon}$ then
10: $v^{j+1} = v^{j+1} - VV^H v^{j+1}$
11: $\beta_j = \|v^{j+1}\|$
12: $v^{j+1} = v^{j+1} / \beta_j$
13: end if
14: $V = [V, v^{j+1}]$
15: end for
Explicit restarts

The growing storage and arithmetic cost may make restarts of the Arnoldi algorithm necessary.
Explicit restarts

The growing storage and arithmetic cost may make restarts of the Arnoldi algorithm necessary.

Since the Arnoldi method naturally starts with one vector, one of the most straightforward restarting schemes is to reduce the whole basis into one vector and start the new Arnoldi iteration with it.

If only one eigenvalue is required (for instance the one with the largest real part), we can choose to restart with the corresponding Ritz vector. If more than one eigenvalue is wanted, we may add all Ritz vectors together to form one starting vector, or use a block version of the Lanczos algorithm that has the same block size as the number of wanted eigenvalues. These options are simple to implement but not nearly as effective as the more sophisticated ones such as the implicit restarting scheme and the thick restart scheme.
Explicit restarts

The growing storage and arithmetic cost may make restarts of the Arnoldi algorithm necessary.

Since the Arnoldi method naturally starts with one vector, one of the most straightforward restarting schemes is to reduce the whole basis into one vector and start the new Arnoldi iteration with it.

If only one eigenvalue is required (for instance the one with the largest real part), we can choose to restart with the corresponding Ritz vector.
The growing storage and arithmetic cost may make restarts of the Arnoldi algorithm necessary.

Since the Arnoldi method naturally starts with one vector, one of the most straightforward restarting schemes is to reduce the whole basis into one vector and start the new Arnoldi iteration with it.

If only one eigenvalue is required (for instance the one with the largest real part), we can choose to restart with the corresponding Ritz vector.

If more than one eigenvalue is wanted, we may add all Ritz vectors together to form one starting vector, or use a block version of the Lanczos algorithm that has the same block size as the number of wanted eigenvalues.
Explicit restarts

The growing storage and arithmetic cost may make restarts of the Arnoldi algorithm necessary.

Since the Arnoldi method naturally starts with one vector, one of the most straightforward restarting schemes is to reduce the whole basis into one vector and start the new Arnoldi iteration with it.

If only one eigenvalue is required (for instance the one with the largest real part), we can choose to restart with the corresponding Ritz vector.

If more than one eigenvalue is wanted, we may add all Ritz vectors together to form one starting vector, or use a block version of the Lanczos algorithm that has the same block size as the number of wanted eigenvalues.

These options are simple to implement but not nearly as effective as the more sophisticated ones such as the implicit restarting scheme and the thick restart scheme.
Implicit restarts (Sorensen 1992)

With \( m = k + p \) steps of the Arnoldi method one determines a factorization

\[
AV_m = V_m H_m + r^m (e^m)^H.
\]
Implicit restarts (Sorensen 1992)

With \( m = k + p \) steps of the Arnoldi method one determines a factorization

\[
AV_m = V_m H_m + r^m (e^m)^H.
\]

With \( p \) steps of the QR algorithm with implicit shifts for \( H_m \) one gets

\[
AV^+_m = V^+_m H^+_m + r^m (e^m)^H Q \quad (*)
\]

where \( Q = Q_1 Q_2 \cdots Q_p \), and \( Q_j \) are the orthogonal matrices from the \( p \) QR–steps, and \( V^+_m = V_m Q \), \( H^+_m = Q^H H_m Q \).
Implicit restarts (Sorensen 1992)

With $m = k + p$ steps of the Arnoldi method one determines a factorization

$$AV_m = V_m H_m + r^m (e^m)^H.$$ 

With $p$ steps of the QR algorithm with implicit shifts for $H_m$ one gets

$$AV_m^+ = V_m^+ H_m^+ + r^m (e^m)^H Q \quad (\ast)$$

where $Q = Q_1 Q_2 \cdots Q_p$, and $Q_j$ are the orthogonal matrices from the $p$ QR–steps, and $V_m^+ = V_m Q$, $H_m^+ = Q^H H_m Q$.

The leading $k - 1$ components of $(e^m)^H Q$ are 0. Hence, the leading $k$ columns of ($\ast$) have the form

$$AV_k^+ = V_k^+ H_k^+ + (r^k)^+ (e^k)^H$$

with the updated residual $(r^k)^+ = V_m^+ e^{k+1} h_{k+1,k} + r^m Q(m, k)$. 
Implicitly restarted Arnoldi method (IRA)

1: Choose initial vector \( v^1 \) with \( \| v^1 \| = 1 \)
2: Determine \( AV_m = V_m H_m + r^m (e^m)^T \) for \( m = k + p \)
3: \textbf{while} \( \max_{j=1,\ldots,k} |t_{j+1,j}| > \text{tol} \) \textbf{do}
4: Determine eigenvalues of \( H_m \) and choose shifts \( \mu_1, \ldots, \mu_p \)
5: \( Q = I_m \)
6: \textbf{for} \( j = 1, \ldots, p \) \textbf{do}
7: \( \text{Compute QR factorization } Q_j R_j = T_m - \mu_j I; \)
8: \( H_m = Q_j^H H_m Q_j \)
9: \( Q = QQ_j \)
10: \textbf{end for}
11: \( V_k^+ = V_m Q(:, 1 : k); \)
12: \( H_k^+ = H_m(1 : k, 1 : k); \)
13: \( (r^k)^+ = V_m^+ e^{k+1} h_{k+1,k} + r^m Q(m, k) \)
14: Determine \( AV_m = V_m H_m + r^m (e_m)^H \) by \( p \) Arnoldi steps
   starting with \( AV_k^+ = V_k H_k^+ + (r^k)^+ (e^k)^H \)
15: \textbf{end while}
Advantages

From the standpoint of numerical stability the updating scheme has several advantages:

(i) Orthogonality can be maintained since the value $k$ is modest.
(ii) There is no question of spurious solutions.
(iii) There is a fixed storage requirement.
(iv) Deflation techniques similar to those associated with the QR iteration for dealing with numerically small diagonal elements of $H_k$ (or $T_k$ in the symmetric case) may be taken advantage of directly.
Advantages

From the standpoint of numerical stability the updating scheme has several advantages:

(i) Orthogonality can be maintained since the value $k$ is modest.
Advantages

From the standpoint of numerical stability the updating scheme has several advantages:

(i) Orthogonality can be maintained since the value $k$ is modest.
(ii) There is no question of spurious solutions
From the standpoint of numerical stability the updating scheme has several advantages:

(i) Orthogonality can be maintained since the value $k$ is modest.
(ii) There is no question of spurious solutions
(iii) There is a fixed storage requirement
From the standpoint of numerical stability the updating scheme has several advantages:

(i) Orthogonality can be maintained since the value $k$ is modest.
(ii) There is no question of spurious solutions
(iii) There is a fixed storage requirement
(iv) Deflation techniques similar to those associated with the QR iteration for dealing with numerically small diagonal elements of $H_k$ (or $T_k$ in the symmetric case) may be taken advantage of directly.
Choice of shifts

Applying one QR–step with shift $\mu$ is equivalent to multiplying $v^1$ by $A - \mu I$ (actually multiplying $e^1 \in \mathbb{R}^m$ by $H_m - \mu I_m$), $p$ QR steps with shifts $\mu_1, \ldots, \mu_p$ therefore corresponds to a multiplication

$$v^1 \leftarrow \psi(A)v^1 \quad \text{mit} \quad \psi(\lambda) = \prod_{j=1}^{p} (\lambda - \mu_j).$$
Choice of shifts

Applying one QR–step with shift $\mu$ is equivalent to multiplying $v^1$ by $A - \mu I$ (actually multiplying $e^1 \in \mathbb{R}^m$ by $H_m - \mu I_m$), $p$ QR steps with shifts $\mu_1, \ldots, \mu_p$ therefore corresponds to a multiplication

$$v^1 \leftarrow \psi(A)v^1 \text{ mit } \psi(\lambda) = \prod_{j=1}^{p} (\lambda - \mu_j).$$

If for instance $\lambda(A)$ is known to be contained in $D \subset \mathbb{C}$, and if the eigenvalues in $\tilde{D} \subset D$ are wanted, then it is reasonable to chose the shifts as roots of a polynomial $\psi$ the modulus of which is as large as possible on $\tilde{D}$ and as small as possible on $D$. 
Choice of shifts

Applying one QR-step with shift $\mu$ is equivalent to multiplying $v^1$ by $A - \mu I$ (actually multiplying $e^1 \in \mathbb{R}^m$ by $H_m - \mu I_m$), $p$ QR steps with shifts $\mu_1, \ldots, \mu_p$ therefore corresponds to a multiplication

$$v^1 \leftarrow \psi(A)v^1 \quad \text{mit} \quad \psi(\lambda) = \prod_{j=1}^{p}(\lambda - \mu_j).$$

If for instance $\lambda(A)$ is known to be contained in $D \subset \mathbb{C}$, and if the eigenvalues in $\tilde{D} \subset D$ are wanted, then it is reasonable to chose the shifts as roots of a polynomial $\psi$ the modulus of which is as large as possible on $\tilde{D}$ and as small as possible on $D$.

This suggests for instance the roots of a

- Chebyshev polynomial with respect to $D \setminus \tilde{D}$ (Saad (1984))
- least squares polynomial (Saad (1987))
- Leja polynomial for $D \setminus \tilde{D}$ (Baglama, Calvetti & Reichel (1998))
**Definition**: Let $K \subset \mathbb{C}$ be a compact set, and let $w : K \rightarrow \mathbb{R}^+$ be a continuous weight function.
**Definition:** Let $K \subset \mathbb{C}$ be a compact set, and let $w : K \rightarrow \mathbb{R}^+$ be a continuous weight function.

Let a sequence of points $z_k$ be defined recursively by

(i) $z_1 \in K : w(z_1)|z_1| = \max_{z \in K} w(z)|z|$

(ii) $z_k \in K : w(z_k) \prod_{j=1}^{k-1} |z_k - z_j| = \max_{z \in K} w(z) \prod_{j=1}^{k-1} |z - z_j|$, $k = 2, 3, \ldots$
**Definition:** Let $K \subset \mathbb{C}$ be a compact set, and let $w : K \rightarrow \mathbb{R}^+$ be a continuous weight function.

Let a sequence of points $z_k$ be defined recursively by

(i) $z_1 \in K : w(z_1) |z_1| = \max_{z \in K} w(z) |z|$

(ii) $z_k \in K : w(z_k) \prod_{j=1}^{k-1} |z_k - z_j| = \max_{z \in K} w(z) \prod_{j=1}^{k-1} |z - z_j|, \ k = 2, 3, \ldots$

Then $z_k$ are called **Leja points**, and the polynomial

$$
\psi(\lambda) = \prod_{k=1}^{p} (\lambda - z_k)
$$

is called **Leja polynomial** of degree $p$ with respect to $w$. 

There is no easy way to determine Leja points. However, Baglama, Calvetti & Reichel (1998) contains a method to determine approximations (called fast Leja points) in an efficient way.
**Definition**: Let $K \subset \mathbb{C}$ be a compact set, and let $w : K \rightarrow \mathbb{R}^+$ be a continuous weight function.

Let a sequence of points $z_k$ be defined recursively by

(i) $z_1 \in K : w(z_1)|z_1| = \max_{z \in K} w(z)|z|

(ii) $z_k \in K : w(z_k) \prod_{j=1}^{k-1} |z_k - z_j| = \max_{z \in K} w(z) \prod_{j=1}^{k-1} |z - z_j|, \ k = 2, 3, \ldots$

Then $z_k$ are called **Leja points**, and the polynomial

$$\psi(\lambda) = \prod_{k=1}^{p} (\lambda - z_k)$$

is called **Leja polynomial** of degree $p$ with respect to $w$.

There is no easy way to determine Leja points. However, Baglama, Calvetti & Reichel (1998) contains a method to determine approximations (called fast Leja points) in an efficient way.
Lehoucq & Sorensen (1996) suggested exact shifts, i.e. $\lambda(T_m)$ is decomposed into $k$ wanted and $p$ unwanted eigenvalues, and the unwanted eigenvalues are chosen as shifts.
Lehoucq & Sorensen (1996) suggested exact shifts, i.e. $\lambda(T_m)$ is decomposed into $k$ wanted and $p$ unwanted eigenvalues, and the unwanted eigenvalues are chosen as shifts.

Possible wanted eigenvalues are

- the $k$ largest / smallest eigenvalues
Lehoucq & Sorensen (1996) suggested exact shifts, i.e. $\lambda(T_m)$ is decomposed into $k$ wanted and $p$ unwanted eigenvalues, and the unwanted eigenvalues are chosen as shifts.

Possible wanted eigenvalues are

- the $k$ largest / smallest eigenvalues
- the $k$ largest / smallest eigenvalues in modulus
Lehoucq & Sorensen (1996) suggested exact shifts, i.e. \( \lambda(T_m) \) is decomposed into \( k \) wanted and \( p \) unwanted eigenvalues, and the unwanted eigenvalues are chosen as shifts.

Possible wanted eigenvalues are
- the \( k \) largest / smallest eigenvalues
- the \( k \) largest / smallest eigenvalues in modulus
- the \( k \) right most eigenvalues
Lehoucq & Sorensen (1996) suggested exact shifts, i.e. $\lambda(T_m)$ is decomposed into $k$ wanted and $p$ unwanted eigenvalues, and the unwanted eigenvalues are chosen as shifts.

Possible wanted eigenvalues are
- the $k$ largest / smallest eigenvalues
- the $k$ largest / smallest eigenvalues in modulus
- the $k$ right most eigenvalues
- the $k$ eigenvalues with largest/smallest imaginary part
Lehoucq & Sorensen (1996) suggested exact shifts, i.e. $\lambda(T_m)$ is decomposed into \textit{k wanted} and \textit{p unwanted} eigenvalues, and the unwanted eigenvalues are chosen as shifts.

Possible wanted eigenvalues are

- the \textit{k} largest / smallest eigenvalues
- the \textit{k} largest / smallest eigenvalues in modulus
- the \textit{k} right most eigenvalues
- the \textit{k} eigenvalues with largest/smallest imaginary part
- the \textit{k} eigenvalues which are closest to an excitation frequency
Lehoucq & Sorensen (1996) suggested exact shifts, i.e. $\lambda(T_m)$ is decomposed into $k$ wanted and $p$ unwanted eigenvalues, and the unwanted eigenvalues are chosen as shifts.

Possible wanted eigenvalues are
- the $k$ largest / smallest eigenvalues
- the $k$ largest / smallest eigenvalues in modulus
- the $k$ right most eigenvalues
- the $k$ eigenvalues with largest/smallest imaginary part
- the $k$ eigenvalues which are closest to an excitation frequency

Other strategies include
- refined shifts (Jia 1998)
- harmonic Ritz values (Morgan 1991)
It may happen that a Ritz pair converged \((\theta_i^{(m)}, V_{ms}^{(m)})\) without \(h_{j+1,j}\) or \(\beta_j\) having become small.
Locking and Purging

It may happen that a Ritz pair converged \((\theta_i^{(m)}, V_m s_i^{(m)})\) without \(h_{j+1,j}\) or \(\beta_j\) having become small.

If \(\theta_i^{(m)}\) is a wanted eigenvalue, then in the next step of the Arnoldi (or Lanczos) method the factorization can be curtailed to

\[
A v^1 = \theta_1 v^1 + \text{“small perturbation"}, \quad (v^1 := V_m s_i^{(m)})
\]

\[
A V_2 = V_2 T_2 + h_{k+1,k} r(e^{k-1})^H.
\]

with \(V_2^H v^1 = 0\). \((\theta_1, v_1)\) then is “locked” and will not be changed in the subsequent steps.
Locking and Purging

It may happen that a Ritz pair converged \((\theta_i^{(m)} , V_m s_i^{(m)})\) without \(h_{j+1,j}\) or \(\beta_j\) having become small.

If \(\theta_i^{(m)}\) is a wanted eigenvalue, then in the next step of the Arnoldi (or Lanczos) method the factorization can be curtailed to

\[
Av^1 = \theta_1 v^1 + \text{“small perturbation”}, \quad (v^1 := V_m s_i^{(m)})
\]

\[
AV_2 = V_2 T_2 + h_{k+1,k} r^r (e^{k-1})^H.
\]

with \(V_2^H v^1 = 0\). \((\theta_1, v_1)\) then is “locked” and will not be changed in the subsequent steps.

If \(\theta_i^{(m)}\) is unwanted it may happen that the influence of \(\theta_i^{(m)}\) can not be removed by the QR iteration. This situation can be handled by a special deflation technique called “purging” (cf. Lehoucq & Sorensen 1996).
Implementations of the Arnoldi method with implicit shifts by Lehoucq, Sorensen & Yang (1998) are freely available.
Implementations of the Arnoldi method with implicit shifts by Lehoucq, Sorensen & Yang (1998) are freely available

- ARPACK (Fortran 77)
- P_ARPACK (Fortran 77, parallel version)
- eigs (MATLAB) (which calls ARPACK)
Implementations of the Arnoldi method with implicit shifts by Lehoucq, Sorensen & Yang (1998) are freely available

- ARPACK (Fortran 77)
- P_ARPACK (Fortran 77, parallel version)
- eigs (MATLAB) (which calls ARPACK)

For symmetric eigenvalue problems Wu and Simon (2000) proposed an alternative restarted version of Lanczos (called thick restarts) which is mathematically equivalent to the implicitly restarted Lanczos method with exact shifts. An implementation

- TRLAN (Fortran 90)

is also freely available.
Implementations of the Arnoldi method with implicit shifts by Lehoucq, Sorensen & Yang (1998) are freely available

- ARPACK (Fortran 77)
- P_ARPACK (Fortran 77, parallel version)
- eigs (MATLAB) (which calls ARPACK)

For symmetric eigenvalue problems Wu and Simon (2000) proposed an alternative restarted version of Lanczos (called thick restarts) which is mathematically equivalent to the implicitly restarted Lanczos method with exact shifts. An implementation

- TRLAN (Fortran 90)

is also freely available. The code can run on a single address machine or in a distributed parallel environment, which requires MPI.
Thick restart

IRA projects $Ax = \lambda x$ to the Krylov space $\mathcal{K}_m(A, V_mQ(:, 1))$, and with exact shifts $\mu_j$ this subspace is

$$\text{span}\{y^1, \ldots, y^k, v^{k+1}, Av^{k+1}, \ldots, A^{p-1}v^{k+1}\}$$

(*)&

where $y^j$ denotes the Ritz vector corresponding to the kept Ritz values.
Thick restart

IRA projects $Ax = \lambda x$ to the Krylov space $\mathcal{K}_m(A, V_mQ(:,1))$, and with exact shifts $\mu_j$ this subspace is

$$\text{span}\{y^1, \ldots, y^k, \nu^{k+1}, A\nu^{k+1}, \ldots, A^{p-1}\nu^{k+1}\} \quad (*)$$

where $y^i$ denotes the Ritz vector corresponding to the kept Ritz values.

It was shown by Morgan (1996) that the subspace (*) is equal to

$$\text{span}\{y^1, \ldots, y^k, Ay^i, A^2y^i, \ldots, A^py^i\}$$

for every $i \in \{1, \ldots, k\}$. This helps to explain the efficiency of IRA, since for each Ritz vector $y^i$ the IRA subspace contains a Krylov subspace $\mathcal{K}_{p+1}(y^i, A)$ with starting vector $y^i$.
Thick restart

IRA projects $Ax = \lambda x$ to the Krylov space $K_m(A, V_m Q(:,1))$, and with exact shifts $\mu_j$ this subspace is

$$\text{span}\{y^1, \ldots, y^k, v^{k+1}, Av^{k+1}, \ldots, A^{p-1} v^{k+1}\} \quad (*)$$

where $y^j$ denotes the Ritz vector corresponding to the kept Ritz values.

It was shown by Morgan (1996) that the subspace $(*)$ is equal to

$$\text{span}\{y^1, \ldots, y^k, Ay^i, A^2 y^i, \ldots, A^p y^i\}$$

for every $i \in \{1, \ldots, k\}$. This helps to explain the efficiency of IRA, since for each Ritz vector $y^j$ the IRA subspace contains a Krylov subspace $K_{p+1}(y^j, A)$ with starting vector $y^j$.

Wu and Simon (2000) developed an alternative restarted version of the Lanczos method (called thick restarts) which is equivalent to IRA with exact shifts. Instead of using the QR algorithm they orthonormalize the vectors $y^1, \ldots, y^k, v^{k+1}, Av^{k+1}, \ldots, A^{p-1} v^{k+1}$ in order to generate an orthonormal basis of the subspace $(*)$. 
Generalized Hermitean eigenproblem

There are several variants of the Lanczos algorithm for the generalized Hermitean eigenvalue problem

\[ Ax = \lambda Bx, \quad A = A^H, \quad B = B^H, \quad B \text{ positive definite} \quad (\ast). \]

They all correspond to a reformulation as a standard eigenproblem \( Cy = \theta y \)
Generalized Hermitean eigenproblem

There are several variants of the Lanczos algorithm for the generalized Hermitean eigenvalue problem

\[ Ax = \lambda Bx, \quad A = A^H, \quad B = B^H, \quad B \text{ positive definite} \quad (\ast). \]

They all correspond to a reformulation as a standard eigenproblem \( Cy = \theta y \)

Problem (\ast) can be transformed to a symmetric eigenproblem

\[ Cy := R^{-H}AR^{-1}y = \lambda y, \quad x = R^{-1}y \]

where \( R \) denotes the Cholesky factor \( B = R^H R \).
Generalized Hermitean eigenproblem

There are several variants of the Lanczos algorithm for the generalized Hermitean eigenvalue problem

\[ Ax = \lambda Bx, \quad A = A^H, \quad B = B^H, \quad B \text{ positive definite} \quad (\ast). \]

They all correspond to a reformulation as a standard eigenproblem \( Cy = \theta y \)

Problem (\ast) can be transformed to a symmetric eigenproblem

\[ Cy := R^{-H}AR^{-1}y = \lambda y, \quad x = R^{-1}y \]

where \( R \) denotes the Cholesky factor \( B = R^H R \).

Alternatively the problem \( Cx = B^{-1}Ax = \lambda x \) is symmetric with respect to the scalar product \( \langle x, y \rangle_B := y^H Bx \). Hence, one can construct by the Lanczos process a \( B \)-orthogonal basis of the Krylov space \( \mathcal{K}_m(C, v) \) such that the projected problem is tridiagonal.
Generalized Hermitean eigenproblem

There are several variants of the Lanczos algorithm for the generalized Hermitean eigenvalue problem

\[ Ax = \lambda Bx, \quad A = A^H, \ B = B^H, \ B \text{ positive definite} \quad (\ast). \]

They all correspond to a reformulation as a standard eigenproblem \( Cy = \theta y \)

Problem (\ast) can be transformed to a symmetric eigenproblem

\[ Cy := R^{-H}AR^{-1}y = \lambda y, \quad x = R^{-1}y \]

where \( R \) denotes the Cholesky factor \( B = R^H R \).

Alternatively the problem \( Cx = B^{-1}Ax = \lambda x \) is symmetric with respect to the scalar product \( \langle x, y \rangle_B := y^H Bx \).

Hence, one can construct by the Lanczos process a \( B \)-orthogonal basis of the Krylov space \( K_m(C, v) \) such that the projected problem is tridiagonal.

Obviously in each step we have to multiply a vector by \( C \), i.e. we have to solve one linear system.
The Lanczos method for $C := B^{-1}A$ computes a basis $V_j$ of $K_j(v^1, C)$ and a real symmetric tridiagonal matrix $T_j$ such that

$$AV_j = BV_j T_j + r(e^i)^H$$

with $V_j^H B V_j = I_j$, $V_j^H A V_j = T_j$, $V_j^H B r = 0$. 

Ritz pairs $(\theta_j(i), x_i(i), (\theta_j(i)))$ are obtained from the tridiagonal eigenproblem $T_j s_i(i), (\theta_j(i)) = \theta_j(i)s_i(i), (\theta_j(i)) = V_j s_i(i), (\theta_j(i))$. For the residual it holds $r_i, (\theta_j(i)) = A x_i (\theta_j(i)) - B x_i (\theta_j(i)) = (AV_j - BV_j T_j)^s_i(i), (\theta_j(i)) = (r(e^i))^H s_i(i), (\theta_j(i)) = B v_j + \beta_j s_i(i), (\theta_j(i))$ from which we obtain $\|r_i, (\theta_j(i))\|_2 = (r_i, (\theta_j(i)))^H B^{-1} r_i, (\theta_j(i)) = |\beta_j|_2$. 

TUHH 
Heinrich Voss 
Krylov subspace methods 
Summer School 2006 
50 / 69
Lanczos method for $C := B^{-1}A$

The Lanczos method for $C := B^{-1}A$ computes a basis $V_j$ of $K_j(v^1, C)$ and a real symmetric tridiagonal matrix $T_j$ such that

$$AV_j = BV_j T_j + r(e^j)^H$$

with $V_j^H BV_j = I_j$, $V_j^H AV_j = T_j$, $V_j^H Br = 0$.

Ritz pairs $(\theta_i^{(j)}, x_i^{(j)})$ are obtained from the tridiagonal eigenproblem

$$T_j s_i^{(j)} = \theta_i^{(j)} s_i^{(j)}, \quad x_i^{(j)} = V_j s_i^{(j)}.$$
Lanczos method for $C := B^{-1}A$

The Lanczos method for $C := B^{-1}A$ computes a basis $V_j$ of $K_j(v^1, C)$ and a real symmetric tridiagonal matrix $T_j$ such that

$$AV_j = BV_j T_j + r(e^i)^H$$

with $V_j^H BV_j = I_j$, $V_j^H AV_j = T_j$, $V_j^H Br = 0$.

Ritz pairs $(\theta_j^{(i)}, x_i^{(j)})$ are obtained from the tridiagonal eigenproblem

$$T_j s_i^{(j)} = \theta_j^{(i)} s_i^{(j)}, \quad x_i^{(j)} = V_j s_i^{(j)}.$$

For the residual it holds

$$r_i^{(j)} = Ax_i^{(j)} - Bx_i^{(j)} \theta_i^{(j)} = AV_j s_i^{(j)} - BV_j s_i^{(j)} \theta_i^{(j)}$$

$$= (AV_j - BV_j T_j) s_i^{(j)} = r(e^i)^H s_i^{(j)} = Bv^j + 1 \beta_j s_j^{i,(j)}$$

from which we obtain

$$\|r_i^{(j)}\|_{B^{-1}}^2 = (r_i^{(j)})^H B^{-1} r_i^{(j)} = |\beta_j s_j^{i,(j)}|^2$$
Theorem 10.9

Let \( A, B \in \mathbb{C}^{n \times n} \) Hermitean and \( B \) positive definite, and denote by \( \lambda_j, j = 1, \ldots, n \) the eigenvalues of \( Ax = \lambda Bx \). Then it holds

\[
\min_{j=1, \ldots, n} |\lambda_j - \theta| \leq \frac{\| Ax - \theta Bx \|_{B^{-1}}}{\| X \|_B}.
\]
Theorem 10.9

Let $A, B \in \mathbb{C}^{n \times n}$ Hermitian and $B$ positive definite, and denote by $\lambda_j$, $j = 1, \ldots, n$ the eigenvalues of $Ax = \lambda Bx$. Then it holds

$$\min_{j=1,\ldots,n} |\lambda_j - \theta| \leq \frac{\|Ax - \theta Bx\|_{B^{-1}}}{\|x\|_B}.$$  

Proof: Let $u^j$ be a set of $B$-orthonormal eigenvalues corresponding to $\lambda_j$ and $x = \sum_{j=1}^n \alpha_j u^j$. Then it holds $\|x\|_B^2 = \sum_{j=1}^n |\alpha_j|^2$, and

$$\|Ax - \theta Bx\|_{B^{-1}} = \| \sum_{j=1}^n \alpha_j (Au^j - \theta Bu^j) \|_{B^{-1}} = \| \sum_{j=1}^n \alpha_j (\lambda_j - \theta) Bu^j \|_{B^{-1}}$$

$$= \sum_{j,k=1}^n \alpha_k (\lambda_k - \theta) \alpha_j (\lambda_j - \theta) (Bv^k)^H B^{-1} Bv^j = \sum_{j=1}^n |\alpha_j (\lambda_j - \theta)|^2$$

$$\geq \min_{j=1,\ldots,n} |\lambda_j - \theta|^2 \|x\|_B^2. \quad \square$$
Theorem 10.9

Let $A, B \in \mathbb{C}^{n \times n}$ Hermitian and $B$ positive definite, and denote by $\lambda_j$, $j = 1, \ldots, n$ the eigenvalues of $Ax = \lambda Bx$. Then it holds

$$\min_{j=1,\ldots,n} |\lambda_j - \theta| \leq \frac{\|Ax - \theta Bx\|_{B^{-1}}}{\|x\|_B}.$$  

Proof: Let $u^j$ be a set of $B$-orthonormal eigenvalues corresponding to $\lambda_j$ and $x = \sum_{j=1}^n \alpha_j u^j$. Then it holds $\|x\|_B^2 = \sum_{j=1}^n |\alpha_j|^2$, and

$$\|Ax - \theta Bx\|_{B^{-1}} = \| \sum_{j=1}^n \alpha_j (Au^j - \theta Bu^j) \|_{B^{-1}} = \| \sum_{j=1}^n \alpha_j (\lambda_j - \theta) Bu^j \|_{B^{-1}}$$

$$= \sum_{j,k=1}^n \alpha_k (\lambda_k - \theta) \alpha_j (\lambda_j - \theta) (Bv^k)^H B^{-1} Bv^j = \sum_{j=1}^n |\alpha_j (\lambda_j - \theta)|^2$$

$$\geq \min_{j=1,\ldots,n} |\lambda_j - \theta|^2 \|x\|_B^2. \quad \Box$$

As in the standard case we only have to monitor the subdiagonal elements $\beta_j$ of $T_j$ and the last component $s_j^{i,(j)}$ of its eigenvectors to control the errors of the Ritz values.
Lanczos method for $C := B^{-1}A$

1: Start with $q = x$, determine $r = Bq$, $\beta_0 = \sqrt{q^Hr}$
2: for $j=1,2,\ldots$ until convergence do
   3: $v_j = q/\beta_{j-1}$
   4: $w_j = r/\beta_{j-1}$
   5: $r = Av_j$
   6: $r = r - \beta_{j-1}w_{j-1}^{-1}$
   7: $\alpha_j = (v_j)^Hr$
   8: $r = r - \alpha_jw_j$
   9: reorthogonalize if necessary
10: solve $Bq = r$ for $q$
11: $\beta_j = \sqrt{q^Hr}$
12: solve eigenproblem $T_j = S\Theta_jS^H$
13: test for convergence
14: end for
15: compute approximate eigenvectors $X = V_jS$
To simplify the description of the $B$-orthogonalization we introduce an auxiliary basis $W_j := BV_j$, which is $B^{-1}$-orthogonal, i.e. $W_j^H B^{-1} W_j = I_j$, and for which $W_j^H V_j = I_j$. 

Comments

To simplify the description of the $B$-orthogonalization we introduce an auxiliary basis $W_j := BV_j$, which is $B^{-1}$-orthogonal, i.e. $W_j^H B^{-1} W_j = I_j$, and for which $W_j^H V_j = I_j$.

In step 9 we only have to reorthogonalize one of the bases $V_j$ and $W_j$ where we can use (as for the standard eigenproblem) complete or selective or partial reorthogonalization.
To simplify the description of the $B$-orthogonalization we introduce an auxiliary basis $W_j := BV_j$, which is $B^{-1}$-orthogonal, i.e. $W_j^H B^{-1} W_j = I_j$, and for which $W_j^H V_j = I_j$.

In step 9 we only have to reorthogonalize one of the bases $V_j$ and $W_j$ where we can use (as for the standard eigenproblem) complete or selective or partial reorthogonalization.

The complete reorthogonalization obtains the form

$$r = r - B(V_j(V_j^H r)),$$

and this step is repeated if the new residual $r$ and $V_j$ are not yet orthogonal.
To simplify the description of the $B$-orthogonalization we introduce an auxiliary basis $W_j := BV_j$, which is $B^{-1}$-orthogonal, i.e. $W_j^H B^{-1} W_j = I_j$, and for which $W_j^H V_j = I_j$.

In step 9 we only have to reorthogonalize one of the bases $V_j$ and $W_j$ where we can use (as for the standard eigenproblem) complete or selective or partial reorthogonalization.

The complete reorthogonalization obtains the form

$$r = r - B(V_j(V_j^H r)),$$

and this step is repeated if the new residual $r$ and $V_j$ are not yet orthogonal.

The additional multiplication by $B$ in the reorthogonalization step can be avoided if both bases, $V_j$ and $W_j$ are stored. Then the reorthogonalization can be performed by

$$r = r - W_j V_j^H r.$$
The algorithm is stopped when the Ritz values $\theta_i^{(j)}$ are sufficiently good approximations of the wanted eigenvalues of the pencil $Ax = \lambda Bx$. 
The algorithm is stopped when the Ritz values $\theta_i^{(j)}$ are sufficiently good approximations of the wanted eigenvalues of the pencil $Ax = \lambda Bx$.

The estimate $|\beta_j s_j^{i,(j)}|$ may be too optimistic if the basis $V_j$ is not fully $B$-orthogonal. Then the Ritz vector $x_j^{i,(j)}$ may have its norm smaller than 1, and we have to replace the estimate by

$$\| r_j^{i,(j)} \|_{B^{-1}} \approx |\beta_j s_j^{i,(j)}| / \| V_j s_j^{i,(j)} \|_B.$$
The algorithm is stopped when the Ritz values $\theta_i^{(j)}$ are sufficiently good approximations of the wanted eigenvalues of the pencil $Ax = \lambda Bx$.

The estimate $|\beta_j s_j^{i,(j)}|$ may be too optimistic if the basis $V_j$ is not fully $B$-orthogonal. Then the Ritz vector $x_j^{i,(j)}$ may have its norm smaller than 1, and we have to replace the estimate by

$$\|r_j^{i,(j)}\|^{-1} \approx |\beta_j s_j^{i,(j)}| \|V_j s_j^{i,(j)}\|_B.$$ 

The Ritz vectors of the original matrix pencil are computed only when the test in step 13 has indicated that the wanted eigenvalues have been converged.
Under general conditions an eigenvalue problem

\[ Lu(x) = \lambda Mu(x), \quad x \in \Omega, \quad Bu(x) = 0, \quad x \in \partial \Omega, \]

with elliptic operators \( L \) and \( M \) has a countable set of eigenvalues \( \lambda_n \), which are clustered only at \( \infty \).
Under general conditions an eigenvalue problem

\[ Lu(x) = \lambda Mu(x), \quad x \in \Omega, \quad Bu(x) = 0, \quad x \in \partial\Omega, \]

with elliptic operators \( L \) and \( M \) has a countable set of eigenvalues \( \lambda_n \), which are clustered only at \( \infty \).

For instance, for an ordinary differential operator \( L \) of second order and \( M = I \) it holds \( \lambda_n = O(n^2) \). Hence the small eigenvalues are relatively close to each other whereas for large eigenvalues the distances will grow.
Shift-and-invert

Under general conditions an eigenvalue problem

\[ Lu(x) = \lambda Mu(x), \quad x \in \Omega, \quad Bu(x) = 0, \quad x \in \partial \Omega, \]

with elliptic operators \( L \) and \( M \) has a countable set of eigenvalues \( \lambda_n \), which are clustered only at \( \infty \).

For instance, for an ordinary differential operator \( L \) of second order and \( M = I \) it holds \( \lambda_n = \mathcal{O}(n^2) \). Hence the small eigenvalues are relatively close to each other whereas for large eigenvalues the distances will grow.

Therefore, for a discretization by the Rayleigh–Ritz method the spectrum will be very widely extended, in the lower part the eigenvalues will be clustered (relatively to the width of the spectrum) and higher eigenvalues will be well separated.
Under general conditions an eigenvalue problem

\[ Lu(x) = \lambda Mu(x), \quad x \in \Omega, \quad Bu(x) = 0, \quad x \in \partial \Omega, \]

with elliptic operators \( L \) and \( M \) has a countable set of eigenvalues \( \lambda_n \), which are clustered only at \( \infty \).

For instance, for an ordinary differential operator \( L \) of second order and \( M = I \) it holds \( \lambda_n = \mathcal{O}(n^2) \). Hence the small eigenvalues are relatively close to each other whereas for large eigenvalues the distances will grow.

Therefore, for a discretization by the Rayleigh–Ritz method the spectrum will be very widely extended, in the lower part the eigenvalues will be clustered (relatively to the width of the spectrum) and higher eigenvalues will be well separated.

Usually one is interested in small eigenvalues. Therefore, the Lanczos method is applied to \( A^{-1}B \) or to \( (A - \sigma B)^{-1}B \) if eigenvalues in the vicinity of a fixed parameter \( \sigma \) are of interest.
Shift-and-invert
Shift–and–invert Lanczos method

The shift-and-invert variant corresponds to the application of the Lanczos method to \( C := B(A - \sigma B)^{-1} \) for some shift \( \sigma \).
The shift-and-invert variant corresponds to the application of the Lanczos method to $C := B(A - \sigma B)^{-1}$ for some shift $\sigma$.

It gives eigenvalues close to $\sigma$, and usually one gets convergence after a small number of steps. Even if systems with the shifted matrix $A - \sigma B$ are more laborious to solve than those with $B$ needed in the direct variant, the smaller number of required steps will very often compensate for this.
The shift-and-invert variant corresponds to the application of the Lanczos method to $C := B(A - \sigma B)^{-1}$ for some shift $\sigma$.

It gives eigenvalues close to $\sigma$, and usually one gets convergence after a small number of steps. Even if systems with the shifted matrix $A - \sigma B$ are more laborious to solve than those with $B$ needed in the direct variant, the smaller number of required steps will very often compensate for this.

The basic recursion of the shift-and-invert method is

$$B(A - \sigma B)^{-1} V_j = V_j T_j + r(e^j)^H. \quad (*)$$
The shift-and-invert variant corresponds to the application of the Lanczos method to $C := B(A − \sigma B)^{-1}$ for some shift $\sigma$.

It gives eigenvalues close to $\sigma$, and usually one gets convergence after a small number of steps. Even if systems with the shifted matrix $A − \sigma B$ are more laborious to solve than those with $B$ needed in the direct variant, the smaller number of required steps will very often compensate for this.

The basic recursion of the shift-and-invert method is

$$B(A − \sigma B)^{-1}V_j = V_j T_j + r(e_i)^H. \quad (\ast)$$

If $V_j$ is chosen to be $B^{-1}$-orthogonal, i.e. $V_j^H B^{-1} V_j = I_j$ then multiplying $(\ast)$ by $V_j^H B^{-1}$ one gets

$$V_j^H (A − \sigma B)^{-1} V_j = T_j,$$

and the Lanczos process yields $V_j^H B^{-1} r = 0$. 
An eigenvalues $\theta_i^{(j)}$ of the tridiagonal matrix $T_j$ is an approximate eigenvalue of

$$C := B(A - \sigma B)^{-1},$$

and therefore

$$\lambda_i^{(j)} = \sigma + \frac{1}{\theta_i^{(j)}}$$

is an approximate eigenvalue of the original pencil $Ax = \lambda Bx$. 
An eigenvalue $\theta_i^{(j)}$ of the tridiagonal matrix $T_j$ is an approximate eigenvalue of

$$C := B(A - \sigma B)^{-1},$$

and therefore

$$\lambda_i^{(j)} = \sigma + \frac{1}{\theta_i^{(j)}}$$

is an approximate eigenvalue of the original pencil $Ax = \lambda Bx$.

If $s_i^{(j)}$ is a corresponding eigenvector, then the Ritz vector

$$x_i^{(j)} := B^{-1} V_j s_i^{(j)}$$

is an approximation to the corresponding eigenvector of $Ax = \lambda Bx$. 
An eigenvalues $\theta_i^{(j)}$ of the tridiagonal matrix $T_j$ is an approximate eigenvalue of

$$C := B(A - \sigma B)^{-1},$$

and therefore

$$\lambda_i^{(j)} = \sigma + \frac{1}{\theta_i^{(j)}}$$

is an approximate eigenvalue of the original pencil $Ax = \lambda Bx$.

If $s_i^{(j)}$ is a corresponding eigenvector, then the Ritz vector

$$x_i^{(j)} := B^{-1} V_j s_i^{(j)}$$

is an approximation to the corresponding eigenvector of $Ax = \lambda Bx$.

Let $W_j$ be the auxiliary basis for which $V_j = BW_j$. Then $W_j$ is $B$-orthogonal, $V_j$ and $W_j$ are biorthogonal, i.e. $W_j^H V_j = I_j$, and the Ritz vectors are

$$x_i^{(j)} := W_j s_i^{(j)}.$$
Multiplying the basic recursion by $B^{-1}(A - \sigma B)$ from the left and $s_{i,(j)}$ from the right, one gets

$$V_j s_{i,(j)} = (A - \sigma B)B^{-1} V_j T_j s_{i,(j)} + \beta_j (A - \sigma B)B^{-1} v^{j+1} s_{j,(j)}.$$
Generalized Eigenvalue Problems

Shift–and–invert Lanczos method ct.

Multiplying the basic recursion by $B^{-1}(A - \sigma B)$ from the left and $s^{i,(j)}$ from the right, one gets

$$V_j s^{i,(j)} = (A - \sigma B) B^{-1} V_j T_j s^{i,(j)} + \beta_j (A - \sigma B) B^{-1} v^{j+1} s^{i,(j)},$$

and it follows for the residual of a Ritz pair

$$r^{i,(j)} = Ax^{i,(j)} - \lambda^{(j)}_i Bx^{i,(j)} = (A - \sigma B) x^{i,(j)} - \frac{1}{\theta_i^{(j)}} Bx^{i,(j)}$$

$$= \frac{1}{\theta_i^{(j)}} \left( (A - \sigma B) B^{-1} V_j s^{i,(j)} \theta_i^{(j)} - V_j s^{i,(j)} \right)$$

$$= - \frac{1}{\theta_i^{(j)}} (A - \sigma B) B^{-1} v^{j+1} \beta_j s^{i,(j)}.$$
Multiplying the basic recursion by $B^{-1}(A - \sigma B)$ from the left and $s^{i,(j)}$ from the right, one gets

$$V_j s^{i,(j)} = (A - \sigma B) B^{-1} V_j T_j s^{i,(j)} + \beta_j (A - \sigma B) B^{-1} v^{j+1} s^{i,(j)},$$

and it follows for the residual of a Ritz pair

$$r^{i,(j)} = Ax^{i,(j)} - \lambda^{(j)} B x^{i,(j)} = (A - \sigma B) x^{i,(j)} - \frac{1}{\theta^{(j)}} B x^{i,(j)}$$

$$= \frac{1}{\theta^{(j)}} \left( (A - \sigma B) B^{-1} V_j s^{i,(j)} \theta^{(j)} - V_j s^{i,(j)} \right)$$

$$= - \frac{1}{\theta^{(j)}} (A - \sigma B) B^{-1} v^{j+1} \beta_j s^{i,(j)}.$$

In this case we do not obtain an error bound for the Ritz values from the residual (but only for the harmonic Ritz values). Nevertheless, $|\beta_j s^{i,(j)}|$ is an error indicator and is used in termination conditions.
Shift–and–invert Lanczos algorithm

1: Start with $r = x$, compute $q = Br$, $\beta_0 = \sqrt{q^H r}$
2: for $j=1,2,\ldots$ until convergence do
3: \hspace{1em} $v^j = q/\beta_{j-1}$
4: \hspace{1em} $w^j = r/\beta_{j-1}$
5: \hspace{1em} Solve $(A - \sigma B)r = v^j$ for $r$
6: \hspace{1em} $r = r - \beta_{j-1} w^{j-1}$
7: \hspace{1em} $\alpha_j = (v^j)^H r$
8: \hspace{1em} $r = r - \alpha_j w^j$
9: \hspace{1em} reorthogonalize if necessary
10: $q = Br$
11: $\beta_j = \sqrt{q^H r}$
12: solve eigenproblem $T_j = S\Theta_j S^H$
13: test for convergence
14: end for
15: compute approximative eigenvectors $X = W_j S$
Since for the shift-and-invert method we can expect rapid convergence to eigenvalues close to the shift one usually applies complete reorthogonalization

\[ r = r - W_j(W_j^H(Br)) \]

or

\[ r = r - W_j(V_j^H r) \]

until \( r \) and the basis \( W_j \) are \( B \)-orthogonal.
Shift–and–invert Lanczos algorithm

Since for the shift-and-invert method we can expect rapid convergence to eigenvalues close to the shift one usually applies complete reorthogonalization

\[ r = r - W_j(W_j^H(Br)) \]

or

\[ r = r - W_j(V_j^Hr) \]

until \( r \) and the basis \( W_j \) are \( B \)-orthogonal.

The linear system in step 5 one uses a factorization

\[ LDL^H = P^T(A - \sigma B)P \]

for an appropriate sparsity preserving permutation \( P \), which is determined in the beginning using sparse Gaussian elimination. Then \( r \) in step 5 is obtained as

\[ r = P(L^{-H}(D^{-1}(L^{-1}(P^Tv^j))))). \]
Shift–and–invert Lanczos algorithm

Since for the shift-and-invert method we can expect rapid convergence to eigenvalues close to the shift one usually applies complete reorthogonalization

\[ r = r - W_j(W_j^H(Br)) \]

or

\[ r = r - W_j(V_j^H r) \]

until \( r \) and the basis \( W_j \) are \( B \)-orthogonal.

The linear system in step 5 one uses a factorization

\[ LDL^H = P^T (A - \sigma B) P \]

for an appropriate sparsity preserving permutation \( P \), which is determined in the beginning using sparse Gaussian elimination. Then \( r \) in step 5 is obtained as

\[ r = P(L^{-H}(D^{-1}(L^{-1}(P^T v^j))))) \).

If all eigenvalues are on one side of the shift then \( A - \sigma B \) is definite, and one can use (sparse) Cholesky, otherwise the matrix \( A - \sigma B \) is indefinite, and one has to use a symmetric indefinite factorization.
The shift-and-invert idea can be used for non-Hermitean eigenvalue problems and regular pencils \( Ax = \lambda Bx \) (i.e. \( \det(A - \lambda B) \neq 0 \)) to determine eigenvalues in the vicinity of a given shift \( \sigma \).
The shift-and-invert idea can be used for non-Hermitean eigenvalue problems and regular pencils \( Ax = \lambda Bx \) (i.e. \( \det(A - \lambda B) \neq 0 \)) to determine eigenvalues in the vicinity of a given shift \( \sigma \).

Since \( B \) is no longer definite the Arnoldi method constructs an orthogonal basis (with respect to the Euclidean inner product) of \( \mathcal{K}_m(v_1, C) \) where \( C := (A - \sigma B)^{-1}B \) such that

\[
(A - \sigma B)^{-1}BV_m = V_mH_m + h_{m+1,m}v_{m+1}e_m^T
\]
The shift-and-invert idea can be used for non-Hermitean eigenvalue problems and regular pencils $Ax = \lambda Bx$ (i.e. $\det(A - \lambda B) \neq 0$) to determine eigenvalues in the vicinity of a given shift $\sigma$.

Since $B$ is no longer definite the Arnoldi method constructs an orthogonal basis (with respect to the Euclidean inner product) of $K_m(v_1, C)$ where $C := (A - \sigma B)^{-1}B$ such that

$$(A - \sigma B)^{-1}BV_m = V_mH_m + h_{m+1,m}v_{m+1}e_m^T$$

The method converges to eigenvalues close to the shift first, and the convergence is faster, the better these eigenvalues are separated from the rest of the spectrum.
The shift-and-invert idea can be used for non-Hermitean eigenvalue problems and regular pencils $Ax = \lambda Bx$ (i.e. $\det(A - \lambda B) \neq 0$) to determine eigenvalues in the vicinity of a given shift $\sigma$.

Since $B$ is no longer definite the Arnoldi method constructs an orthogonal basis (with respect to the Euclidean inner product) of $K_m(v_1, C)$ where $C := (A - \sigma B)^{-1} B$ such that

$$(A - \sigma B)^{-1} BV_m = V_m H_m + h_{m+1,m} v_{m+1} e^T_m$$

The method converges to eigenvalues close to the shift first, and the convergence is faster, the better these eigenvalues are separated from the rest of the spectrum.

ARPACK has driver routines for the generalized eigenvalue problem and uses shift-and-invert.
Rational Krylov subspace method

If one is interested in eigenvalues in a large interval \([\alpha, \beta] \subset \sigma(A)\) or an extended region of the complex plane then one can apply the shift-and-invert method in several runs with several parameters \(\sigma_j \in [\alpha, \beta]\).
Rational Krylov subspace method

If one is interested in eigenvalues in a large interval \([\alpha, \beta] \subset \sigma(A)\) or an extended region of the complex plane then one can apply the shift-and-invert method in several runs with several parameters \(\sigma_j \in [\alpha, \beta]\).

The cost can be reduced considerably if the eigenproblem is projected to a rational Krylov space

\[
\{ v, (A - \sigma_1 B)^{-1} Bv, \ldots, (A - \sigma_1 B)^{-i_1} Bv, (A - \sigma_2 B)^{-1} Bv, \ldots, (A - \sigma_2 B)^{-i_2} Bv, \ldots, (A - \sigma_k B)^{-1} Bv, \ldots, (A - \sigma_k B)^{-i_k} Bv \}
\]

and the projected eigenproblem is solved.
If one is interested in eigenvalues in a large interval $[\alpha, \beta] \subset \sigma(A)$ or an extended region of the complex plane then one can apply the shift-and-invert method in several runs with several parameters $\sigma_j \in [\alpha, \beta]$.

The cost can be reduced considerably if the eigenproblem is projected to a rational Krylov space

$$\{v, (A - \sigma_1 B)^{-1} Bv, \ldots, (A - \sigma_1 B)^{-i_1} Bv, (A - \sigma_2 B)^{-1} Bv, \ldots, (A - \sigma_2 B)^{-i_2} Bv, \ldots, (A - \sigma_k B)^{-1} Bv, \ldots, (A - \sigma_k B)^{-i_k} Bv\}$$

and the projected eigenproblem is solved.

In the Arnoldi method the eigenproblem is projected onto a subspace of the form

$$V_m = \{\psi((A - \sigma B)^{-1} B)v : \psi \in \Pi_{m-1}, \psi(0) = 1\}.$$
Rational Krylov subspace method

If one is interested in eigenvalues in a large interval \([\alpha, \beta] \subset \sigma(A)\) or an extended region of the complex plane then one can apply the shift-and-invert method in several runs with several parameters \(\sigma_j \in [\alpha, \beta]\).

The cost can be reduced considerably if the eigenproblem is projected to a rational Krylov space

\[
\{ v, (A - \sigma_1 B)^{-1} Bv, \ldots, (A - \sigma_1 B)^{-i_1} Bv, (A - \sigma_2 B)^{-1} Bv, \ldots, \\
(A - \sigma_2 B)^{-i_2} Bv, \ldots, (A - \sigma_k B)^{-1} Bv, \ldots, (A - \sigma_k B)^{-i_k} Bv \}
\]

and the projected eigenproblem is solved.

In the Arnoldi method the eigenproblem is projected onto a subspace of the form

\[
V_m = \{ \psi((A - \sigma B)^{-1} B)v : \psi \in \Pi_{m-1}, \psi(0) = 1 \}.
\]

For the rational Krylov method the subspace can be written as

\[
V = \{ \rho(A)v : \rho \text{ a suitable rational function, } \rho(0) = 1 \}.
\]
Rational Krylov subspace method

If one is interested in eigenvalues in a large interval \([\alpha, \beta] \subset \sigma(A)\) or an extended region of the complex plane then one can apply the shift-and-invert method in several runs with several parameters \(\sigma_j \in [\alpha, \beta]\).

The cost can be reduced considerably if the eigenproblem is projected to a rational Krylov space

\[
\{ v, (A - \sigma_1 B)^{-1} Bv, \ldots, (A - \sigma_1 B)^{-i_1} Bv, (A - \sigma_2 B)^{-1} Bv, \ldots, \\
(A - \sigma_2 B)^{-i_2} Bv, \ldots, (A - \sigma_k B)^{-1} Bv, \ldots, (A - \sigma_k B)^{-i_k} Bv \}
\]

and the projected eigenproblem is solved.

In the Arnoldi method the eigenproblem is projected onto a subspace of the form

\[
V_m = \{ \psi((A - \sigma B)^{-1} B)v : \psi \in \Pi_{m-1}, \psi(0) = 1 \}.
\]

For the rational Krylov method the subspace can be written as

\[
V = \{ \rho(A)v : \rho \text{ a suitable rational function, } \rho(0) = 1 \}.
\]

It can be shown (Ruhe 1998) that the rational Krylov method can be interpreted as a shift-and-invert Lanczos method with shift \(\sigma_k\) and a modified initial vector \(\tilde{v}^1\).
Rational Krylov subspace method ct.

Rational Krylov starts as shift-and-invert Arnoldi method with shift $\sigma_1$ and initial vector $v^1$, and determines an Arnoldi recursion

$$(A - \sigma_1 B)^{-1} B V_m = V_{m+1} H_{m+1,m}. \quad (1)$$
Rational Krylov subspace method ct.

Rational Krylov starts as shift-and-invert Arnoldi method with shift $\sigma_1$ and initial vector $v^1$, and determines an Arnoldi recursion

$$(A - \sigma_1 B)^{-1} B V_m = V_{m+1} H_{m+1,m}. \quad (1)$$

If $m$ is big enough then accurate approximations to eigenvalues in the vicinity of $\sigma_1$ are obtained from extreme eigenvalues of $H_m = H_{m+1,m}(1 : m, 1 : m)$. 
Rational Krylov subspace method ct.

Rational Krylov starts as shift-and-invert Arnoldi method with shift $\sigma_1$ and initial vector $\nu^1$, and determines an Arnoldi recursion

\[(A - \sigma_1 B)^{-1} B V_m = V_{m+1} H_{m+1,m}. \quad (1)\]

If $m$ is big enough then accurate approximations to eigenvalues in the vicinity of $\sigma_1$ are obtained from extreme eigenvalues of $H_m = H_{m+1,m}(1 : m, 1 : m)$.

To obtain further eigenvalue approximations we choose a new shift $\sigma_2$ and continue the Arnoldi process without throwing away the information gathered in the basis of the Krylov space $\mathcal{K} := \mathcal{K}_m((A - \sigma_1 B)^{-1} B, \nu^1)$. 
Rational Krylov subspace method ct.

Rational Krylov starts as shift-and-invert Arnoldi method with shift $\sigma_1$ and initial vector $v^1$, and determines an Arnoldi recursion

$$(A - \sigma_1 B)^{-1} B V_m = V_{m+1} H_{m+1,m}. \quad (1)$$

If $m$ is big enough then accurate approximations to eigenvalues in the vicinity of $\sigma_1$ are obtained from extreme eigenvalues of $H_m = H_{m+1,m}(1 : m, 1 : m)$.

To obtain further eigenvalue approximations we choose a new shift $\sigma_2$ and continue the Arnoldi process without throwing away the information gathered in the basis of the Krylov space $\mathcal{K} := \mathcal{K}_m((A - \sigma_1 B)^{-1} B, v^1)$.

This is indeed possible if we are able to determine an Arnoldi recursion

$$(A - \sigma_2 B)^{-1} B W_m = W_{m+1} \tilde{H}_{m+1,m}. \quad (2)$$

corresponding to the shift $\sigma_2$ and the initial vector $w^1$ such that $\tilde{H}_{m+1,m}$ has the same trapezoidal form as $H_{m+1,m}$ (i.e. $\tilde{h}_{ij} = 0$ for $i > j + 1$), and $\text{span}(V_{m+1}) = \text{span}(W_{m+1})$. 


Rewrite the recursion (1) as

\[ BV_m = (A - \sigma_1 B)V_{m+1}H_{m+1,m}, \]

which is equivalent to

\[ (\sigma_1 - \sigma_2)BV_{m+1}H_{m+1,m} + BV_m = (A - \sigma_2 B)V_{m+1}H_{m+1,m}, \]
Rewrite the recursion (1) as

\[ BV_m = (A - \sigma_1 B)V_{m+1}H_{m+1,m}, \]

which is equivalent to

\[ (\sigma_1 - \sigma_2)BV_{m+1}H_{m+1,m} + BV_m = (A - \sigma_2 B)V_{m+1}H_{m+1,m}, \]

and to

\[ BV_{m+1}(I_{m+1,m} + (\sigma_1 - \sigma_2)H_{m+1,m}) = (A - \sigma_2 B)V_{m+1}H_{m+1,m}, \]

where the matrix

\[ K_{m+1,m} := I_{m+1,m} + (\sigma_1 - \sigma_2)H_{m+1,m} \]

is trapezoidal of the same form as \( H_{m+1,m} \).
From

\[(A - \sigma_2 B)^{-1} BV_{m+1} K_{m+1,m} = V_{m+1} H_{m+1,m}\]

we obtain the desired Arnoldi recursion if we get rid of the factor \(K_{m+1,m}\) on the left.
Rational Krylov subspace method ct.

From
\[(A - \sigma_2 B)^{-1} B V_{m+1} K_{m+1,m} = V_{m+1} H_{m+1,m}\]
we obtain the desired Arnoldi recursion if we get rid of the factor \(K_{m+1,m}\) on the left.

If
\[K_{m+1,m} = Q_{m+1} \begin{bmatrix} R_m & \ \ \\ 0 & \end{bmatrix}\]
denotes the QR factorization of \(K_{m+1,m}\), then \(R_m\) is regular (otherwise a subdiagonal element of \(H_{m+1,m}\) would have been 0, and the Arnoldi process would have stopped with an invariant subspace), and it follows
\[(A - \sigma_2 B)^{-1} B V_{m+1} Q_{m+1} \begin{bmatrix} R_m & \ \ \\ 0 & \end{bmatrix} = V_{m+1} H_{m+1,m},\]
and multiplication by \(R_m^{-1}\) from the right yields
Rational Krylov subspace method ct.

\[(A - \sigma_2 B)^{-1} B V_{m+1} Q_{m+1,m} = V_{m+1} Q_{m+1} H_{m+1,m} R_m^{-1}. \]  

(3)
\[(A - \sigma_2 B)^{-1} B V_{m+1} Q_{m+1} = V_{m+1} Q_{m+1} Q_{m+1}^H H_{m+1, m} R_m^{-1}. \]  

(3)

Hence, with the orthogonal basis \( V_{m+1} Q_{m+1} \) of the Krylov space \( \mathcal{K} \) the projection of \((A - \sigma_2 B)^{-1} B\) is represented by the full matrix

\[ L_{m+1, m} = Q_{m+1}^H H_{m+1, m} R_m^{-1} \]

which can be transformed to trapezoidal form by applying Householder matrices from bottom upwards

\[ L_{m+1, m} = \begin{bmatrix} P_m & 0 \\ 0 & 1 \end{bmatrix} \tilde{H}_{m+1, m} P_m^H. \]

Multiplying equation (3) by \( P_m \) from the right one gets
Generalized Eigenvalue Problems

Rational Krylov subspace method ct.

\[(A - \sigma_2 B)^{-1} B V_{m+1} Q_{m+1, m} P_m = V_{m+1} Q_{m+1} \begin{bmatrix} P_m & 0 \\ 0 & 1 \end{bmatrix} \tilde{H}_{m+1, m},\]

i.e. an Arnoldi recursion

\[(A - \sigma_2 B)^{-1} B W_m = W_{m+1} \tilde{H}_{m+1, m}\]

with the new shift \(\sigma_2\), the new orthogonal basis

\[W_{m+1} = V_{m+1} Q_{m+1} \begin{bmatrix} P_m & 0 \\ 0 & 1 \end{bmatrix},\]

and the upper Hessenberg matrix \(\tilde{H}_{m+1, m}\).
Notice that all transformations are done without performing operations with the large matrices $A$ and $B$, and that it can even be avoided to form the matrix $W$ explicitly, thus avoiding all work on large vectors.
Rational Krylov subspace method ct.

Notice that all transformations are done without performing operations with the large matrices $A$ and $B$, and that it can even be avoided to form the matrix $W$ explicitly, thus avoiding all work on large vectors.

In practical implementations the rational Krylov method is combined with locking, purging, and implicit restarts.
Notice that all transformations are done without performing operations with the large matrices $A$ and $B$, and that it can even be avoided to form the matrix $W$ explicitly, thus avoiding all work on large vectors.

In practical implementations the rational Krylov method is combined with locking, purging, and implicit restarts.

Implicitly restarted shift-and-invert Arnoldi with shift $\sigma_1$ is run until an appropriate number of eigenvalues around $\sigma_1$ have converged. Then these eigenvalues are locked, and eigenvalues outside the interesting region are purged leaving an Arnoldi recursion of dimension $m$. Then a new shift $\sigma_2$ is introduced, $W_{m+1}$ and $\tilde{H}_{m+1,m}$ are determined, and the implicitly restarted shift-and-invert Arnoldi process with shift $\sigma_2$ is continued without touching the locked Ritz pairs. The same procedure is repeated until all interesting eigenvalues have converged.