I.1.(a) Krylov subspace projection methods
Orthogonal projection technique: framework

Let \( A \) be an \( n \times n \) complex matrix and \( \mathcal{K} \) be an \( m \)-dimensional subspace of \( \mathbb{C}^n \). An orthogonal projection technique seeks an approximate eigenpair

\[(\tilde{\lambda}, \tilde{u}) \text{ with } \tilde{\lambda} \in \mathbb{C} \text{ and } \tilde{u} \in \mathcal{K}.\]

This approximate eigenpair is obtained by imposing the following Galerkin condition:

\[A\tilde{u} - \tilde{\lambda}\tilde{u} \perp \mathcal{K},\]

or, equivalently,

\[v^H(A\tilde{u} - \tilde{\lambda}\tilde{u}) = 0, \quad \forall v \in \mathcal{K}.\] (2)

In matrix form, assume that an orthonormal basis \( \{v_1, v_2, \ldots, v_k\} \) of \( \mathcal{K} \) is available. Denote \( V = (v_1, v_2, \ldots, v_k) \), and let \( \tilde{u} = V y \). Then, the condition (2) becomes

\[v_j^H(AV y - \tilde{\lambda}V y) = 0, \quad j = 1, \ldots, k.\]
Therefore, \( y \) and \( \tilde{\lambda} \) must satisfy
\[
B_k y = \tilde{\lambda} y, \tag{3}
\]
where
\[
B_k = V^H A V.
\]
Each eigenvalue \( \tilde{\lambda}_i \) of \( B_k \) is called a **Ritz value**, and \( V y_i \) is called **Ritz vector**, where \( y_i \) is the eigenvector of \( B_k \) associated with \( \tilde{\lambda}_i \).
Rayleigh-Ritz procedure - orthogonal projection

1. Compute an orthonormal basis \( \{v_i\}_{i=1,...,k} \) of the subspace \( \mathcal{K} \).
   Let \( V = (v_1, v_2, \ldots, v_k) \).

2. Compute \( B_k = V^H A V \).

3. Compute the eigenvalues of \( B_k \) and select \( k_0 \) desired ones:
   \( \tilde{\lambda}_i, i = 1, 2, \ldots, k_0 \), where \( k_0 \leq k \).

4. Compute the eigenvectors \( y_i, i = 1, \ldots, k_0 \), of \( B_k \) associated with \( \tilde{\lambda}_i, i = 1, \ldots, k_0 \), and the corresponding approximate eigenvectors of \( A \),
   \( \tilde{u}_i = V y_i, i = 1, \ldots, k_0 \).
Oblique projection technique: framework

Select two subspaces $\mathcal{L}$ and $\mathcal{K}$ and then seek an approximate eigenpair

$$(\tilde{\lambda}, \tilde{u}) \text{ with } \lambda \in \mathbb{C} \text{ and } \tilde{u} \in \mathcal{K}$$

that satisfies the **Petrov-Galerkin condition**:

$$v^H(A\tilde{u} - \tilde{\lambda}\tilde{u}) = 0, \quad \forall \ v \in \mathcal{L}. \quad (4)$$

In matrix form, let $V$ denote the basis for the subspace $\mathcal{K}$ and $W$ for $\mathcal{L}$. Then, writing $\tilde{u} = V y$, the Petrov-Galerkin condition (4) yields the reduced eigenvalue problem

$$B_k y = \tilde{\lambda} C_k y,$$

where

$$B_k = W^H A V \quad \text{and} \quad C_k = W^H V.$$
If \( C_k = V^H V = I \), then the two bases are called \textit{biorthonormal}.

In order for a biorthonormal pair \( V \) and \( W \) to exist the following additional assumption for \( \mathcal{L} \) and \( \mathcal{K} \) must hold. \textit{For any two bases \( V \) and \( W \) of \( \mathcal{K} \) and \( \mathcal{L} \), respectively,}

\[
\det(W^H V) \neq 0 \quad .
\] (5)
Rayleigh-Ritz procedure - oblique projection

1. Compute an orthonormal bases \( \{v_i\}_{i=1}^{k} \) of the subspace \( \mathcal{K} \) and \( \{w_i\}_{i=1}^{k} \) of the subspace \( \mathcal{L} \).
   
   Let \( V = (v_1, v_2, \ldots, v_k) \) and \( W = (w_1, w_2, \ldots, w_k) \).

2. Compute \( B_k = W^H A V \) and \( C_k = W^H V \).

3. Compute the eigenvalues of \( B_k - \lambda C_k \) and select \( k_0 \) desired ones: \( \tilde{\lambda}_i, i = 1, 2, \ldots, k_0 \), where \( k_0 \leq k \).

4. Compute the right and left eigenvectors \( y_i \) and \( z_i \), \( i = 1, \ldots, k_0 \), of \( B_k - \lambda C_k \) associated with \( \tilde{\lambda}_i, i = 1, \ldots, k_0 \), and the corresponding approximate right and left eigenvectors of \( A \), \( \tilde{u}_i = V y_i \), and \( \tilde{v}_i = W z_i \), \( i = 1, \ldots, k_0 \).
Optimality

Let \( Q = (Q_k, Q_u) \) be an \( n \)-by-\( n \) orthogonal matrix, where \( Q_k \) is \( n \)-by-\( k \), and \( Q_u \) is \( n \)-by-(\( n - k \)), and \( \text{span}(Q_k) = \mathcal{K} \). Then

\[
T = Q^T A Q = \begin{bmatrix}
Q_k^T A Q_k & Q_k^T A Q_u \\
Q_u^T A Q_k & Q_u^T A Q_u
\end{bmatrix} \equiv \begin{bmatrix}
T_k & T_{uk} \\
T_{ku} & T_u
\end{bmatrix}
\]

The Ritz values and Ritz vectors are considered optimal approximations to the eigenvalues and eigenvectors of \( A \) from the selected subspace \( \mathcal{K} = \text{span}(Q_k) \) as justified by the follows.

Theorem.

\[
\min_{S, k \times k} \| A Q_k - Q_k S \|_2 = \| A Q_k - Q_k T_k \|_2
\]
Krylov subspace

\[ \mathcal{K}_{k+1}(A, u_0) = \text{span}\{u_0, Au_0, A^2u_0, \ldots, A^k u_0\} \]
\[ = \{q(A)u_0 \mid q \in \mathcal{P}_k\}, \]

where \(\mathcal{P}_k\) is the set of all polynomial of degree less than \(k + 1\).

Properties of \(\mathcal{K}_{k+1}(A, u_0)\):

1. \(\mathcal{K}_k(A, u_0) \subset \mathcal{K}_{k+1}(A, u_0)\).
   \(A\mathcal{K}_k(A, u_0) \subset \mathcal{K}_{k+1}(A, u_0)\).

2. If \(\sigma \neq 0\), \(\mathcal{K}_k(A, u_0) = \mathcal{K}_k(\sigma A, u_0) = \mathcal{K}_k(A, \sigma u_0)\).

3. For any scalar \(\kappa\), \(\mathcal{K}_k(A, u_0) = \mathcal{K}_k(A - \kappa I, u_0)\).

4. If \(W\) is nonsingular, \(\mathcal{K}_k(W^{-1}AW, W^{-1}u_0) = W^{-1}\mathcal{K}_k(A, u_0)\).
Arnoldi decomposition

An explicit Krylov basis \( \{u_0, Au_0, A^2u_0, \ldots, A^ku_0\} \) is not suitable for numerical computing. It is extremely ill-conditioned. Therefore, our first task is to replace a Krylov basis with a better conditioned basis, say an orthonormal basis.

**Theorem.** Let the columns of \( K_{j+1} = (u_0 \ A u_0 \ \ldots \ A^j u_0) \) be linearly independent. Let

\[
K_{j+1} = U_{j+1} R_{j+1}
\]

be the QR factorization of \( K_{j+1} \). Then there is a \((j+1) \times j\) unreduced upper Hessenberg matrix \( \hat{H}_j \) such that

\[
A U_j = U_{j+1} \hat{H}_j.
\]

Conversely, if \( U_{j+1} \) is orthonormal and satisfies (7), then

\[
\text{span}(U_{j+1}) = \text{span}\{u_0, Au_0, \ldots, A^j u_0\}.
\]
Proof: Partitioning the QR decomposition (6), we have

\[
(K_j A^j u_0) = (U_j u_{j+1}) \begin{pmatrix} R_j & r_{j+1} \\ 0 & r_{j+1,j+1} \end{pmatrix},
\]

where \(K_j = U_j R_j\) is the QR decomposition of \(K_j\). Then

\[
AK_j = AU_j R_j
\]
or

\[
AU_j = AK_j R_j^{-1} = K_{j+1} \begin{pmatrix} 0 \\ R_j^{-1} \end{pmatrix} = U_{j+1} R_{j+1} \begin{pmatrix} 0 \\ R_{j}^{-1} \end{pmatrix}.
\]

It is easy to verify that

\[
\hat{H}_j = R_{j+1} \begin{pmatrix} 0 \\ R_{j}^{-1} \end{pmatrix}
\]
is a \((j + 1) \times j\) unreduced upper Hessenberg matrix. Therefore we complete the proof of (7).
Conversely, suppose that \(U_{j+1}\) satisfies (7), then by induction, we can prove the identity (8).
Arnoldi decomposition: by partitioning,

\[ \hat{H}_j = \left( \begin{array}{c} H_j \\ h_{j+1,j} e_j^T \end{array} \right), \]

the decomposition (7) can be written as follows:

\[ AU_j = U_j H_j + h_{j+1,j} u_{j+1} e_j^T. \] (9)

We call (9) an Arnoldi decomposition of order \( j \). The decomposition (7) is a compact form.
Arnoldi procedure

By the Arnoldi decomposition (9), we deduce the following process to generate an orthogonormal basis \( \{v_1, v_2, \ldots, v_m\} \) of the Krylov subspace \( \mathcal{K}_m(A, v) \):

**Arnoldi Process:**

1. \( v_1 = v / \|v\|_2 \)
2. for \( j = 1, 2, \ldots, k \)
3. compute \( w = Av_j \)
4. for \( i = 1, 2, \ldots, j \)
5. \( h_{ij} = v_i^Tw \)
6. \( w = w - h_{ij}v_i \)
7. end for
8. \( h_{j+1,j} = \|w\|_2 \)
9. If \( h_{j+1,j} = 0 \), stop
10. \( v_{j+1} = w_j / h_{j+1,j} \)
11. endfor
Remarks:

1. The matrix $A$ is only referenced via the matrix-vector multiplication $Av_j$. Therefore, it is ideal for large scale matrices. Any sparsity or structure of a matrix can be exploited.

2. The main storage requirement is $(m + 1)n$ for storing Arnoldi vectors $\{v_i\}$

3. The cost of arithmetic is $m$ matrix-vector products plus $2m^2n$ for the rest. It is common that the matrix-vector multiplication is the dominant cost.

4. The Arnoldi procedure breaks down when $h_{j+1,j} = 0$ for some $j$. It is easy to see that if the Arnoldi procedure breaks down at step $j$ (i.e. $h_{j+1,j} = 0$), then $\mathcal{K}_j = \text{span}(V_j)$ is invariant subspace of $A$.

5. Some care must be taken to insure that the vectors $v_j$ remain orthogonal to working accuracy in the presence of rounding error. The usual technique is reorthogonalization.
Arnoldi decomposition

- Denote

\[ V_k = (v_1 \ v_2 \ \ldots \ v_k) \]

and

\[ H_k = \begin{pmatrix}
  h_{11} & h_{12} & \ldots & h_{1,k-1} & h_{1k} \\
  h_{21} & h_{22} & \ldots & h_{2,k-1} & h_{2k} \\
  h_{32} & \ddots & \ddots & \ddots & \ddots \\
  \vdots & \vdots & \ddots & h_{3,k-1} & h_{3k} \\
  h_{k,k-1} & \cdots & \cdots & h_{k,k-1} & h_{kk}
\end{pmatrix}. \]

- The Arnoldi process can be expressed in the following governing relations:

\[ AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T \]

and

\[ V_k^H V_k = I \quad \text{and} \quad V_k^H v_{k+1} = 0. \]
• The decomposition is uniquely determined by the starting vector \( v \) (the implicit \( Q \)-Theorem).

• Since \( V_k^H v_{k+1} = 0 \), we have
  \[ H_k = V_k^T A V_k. \]

• Let \( \mu \) be an eigenvalue of \( H_k \) and \( y \) be a corresponding eigenvector \( y \), i.e.,
  \[ H_k y = \mu y, \quad \|y\|_2 = 1. \]

Then the corresponding Ritz pair is \((\mu, V_k y)\). Applying \( y \) to the right hand side of (10), the residual vector for \((\mu, V_k y)\) is given by
  \[ A(V_k y) - \mu (V_k y) = h_{k+1,k} v_{k+1}(e_k^T y). \]
Using the backward error interpretation, we know that \((\mu, V_k y)\) is an exact eigenpair of \(A + E\):

\[
(A + E)(V_k y) = \mu (V_k y),
\]

where

\[
\|E\|_2 = |h_{k+1,k}| \cdot |e_k^T y|.
\]

This gives us a criterion of whether to accept the Ritz pair \((\mu, V_k y)\) as an accurate approximate eigenpair of \(A\).
Arnoldi method = RR + Arnoldi

1. Choose a starting vector \( v \);
2. Generate the Arnoldi decomposition of length \( k \) by the Arnoldi process;
3. Compute the Ritz pairs and decide which ones are acceptable;
4. If necessary, increase \( k \) and repeat.
An example

\[ A = \text{sprandn}(100, 100, 0.1) \text{ and } v = (1, 1, \ldots, 1)^T. \]

“+” are the eigenvalues of matrix \( A \)

“○” are the eigenvalues of the upper Hessenberg matrix \( H_{30} \)
Observation: *exterior eigenvalues converge first*, a typical convergence phenomenon.
The need of restarting

The algorithm has two nice aspects:

1. $H_k$ is already in the Hessenberg form, so we can immediately apply the QR algorithm to find its eigenvalues.

2. After we increase $k$ to, say $k + p$, we only have to orthogonalize $p$ vectors to compute the $(k + p)$th Arnoldi decomposition. The work already completed previously is not thrown away.

Unfortunately, the algorithm has its drawbacks, too:

1. If $A$ is large, we cannot increase $k$ indefinitely, since $V_k$ requires $nk$ memory locations to store.

2. We have little control over which eigenpairs the algorithm finds.
Implicit restarting

Goal: purge the unwanted eigenvalues $\mu$ from $H_k$.

1. Exact arithmetic case:

   By one step of the QR algorithm with shift $\mu$, we have
   \[
   R = U^H (H - \mu I) = \text{upper triangular}
   \]

   Note that $H - \mu I$ is singular, hence $R$ must have a zero on its diagonal. Because $H$ is unreduced, then $r_{nn} = 0$.

   Furthermore, note that $U = P_{12}P_{23}\cdots P_{n-1,n}$, where $P_{i,i+1}$ is a rotation in the $(i, i + 1)$-plane. Consequently, $U$ is Hessenberg:
   \[
   U = \begin{pmatrix}
   U_* & u \\
   u_{k,k-1}e_{k-1}^T & u_{k,k}
   \end{pmatrix}.
   \]

   Hence
   \[
   H' = RU + \mu I = \begin{pmatrix}
   \hat{H}_* & \hat{h} \\
   0 & \mu
   \end{pmatrix} = U^H H U.
   \]
In other words, one step of the shifted QR has found the eigenvalue $\mu$ exactly and has deflated the problem.

2. Finite precision arithmetic case:

- In the presence of rounding error, after one step of the shifted QR, we have
  \[ \widetilde{H}' = \hat{U}^H H \hat{U} = \begin{pmatrix} \hat{H}_* & \hat{h} \\ \hat{h}_{k,k-1} e_{k-1}^T & \hat{\mu} \end{pmatrix}. \]

- From the Arnoldi decomposition, we have
  \[ AV_k \hat{U} = V_k \hat{U} (\hat{U}^T H_k \hat{U}) + h_{k+1,k} v_{k+1} e_k^T \hat{U}. \]

- Partition
  \[ \hat{V}_k = V_k \hat{U} = \begin{pmatrix} \hat{V}_{k-1} \\ \hat{v}_k \end{pmatrix}. \]

Then
\[ A \left( \begin{pmatrix} \hat{V}_{k-1} \\ \hat{v}_k \end{pmatrix} \right) = \left( \begin{pmatrix} \hat{V}_{k-1} \\ \hat{v}_k \end{pmatrix} \right) \begin{pmatrix} \hat{H}_* & \hat{h} \\ \hat{h}_{k,k-1} e_{k-1}^T & \hat{\mu} \end{pmatrix} + h_{k+1,k} v_{k+1} \left( e_k^T \hat{U} \right). \]
• From the first $k-1$ columns of this partition, we get
\[
A\tilde{V}_{k-1} = \tilde{V}_{k-1}\tilde{H}_* + fe_{k-1}^T, \tag{11}
\]
where $f = \hat{h}_{k,k-1}\hat{v}_k + h_{k+1,k}\hat{u}_{k,k-1}v_{k+1}$.

• Note that $\tilde{H}_*$ is Hessenberg. $f$ is orthogonal to $\tilde{V}_{k-1}$. Hence (11) is an Arnoldi decomposition of length $k - 1$.

• The process may be repeated to remove other unwanted values from $H$. 
The symmetric Lanczos procedure

- Observation: in the Arnoldi decomposition, if $A$ is symmetric, then the upper Hessenberg matrix $H_j$ is symmetric tridiagonal.

- The following is a simplified process to compute an orthonormal basis of a Krylov subspace:

**Lanczos process:**

1. $q_1 = v/\|v\|_2$, $\beta_0 = 0$; $q_0 = 0$;
2. for $j = 1$ to $k$, do
3. \[ w = Aq_j; \]
4. \[ \alpha_j = q_j^T w; \]
5. \[ w = w - \alpha_j q_j - \beta_{j-1} q_{j-1}; \]
6. \[ \beta_j = \|w\|_2; \]
7. if $\beta_j = 0$, quit;
8. \[ q_{j+1} = w/\beta_j; \]
9. endfor
The symmetric Lanczos algorithm: governing equation

Denote

\[ Q_k = (q_1 \ q_2 \ \ldots \ q_k) \]

and

\[ T_k = \begin{pmatrix}
\alpha_1 & \beta_1 \\
\beta_1 & \alpha_2 & \beta_2 \\
& \ddots & \ddots & \ddots \\
& & \alpha_{k-1} & \beta_{k-1} \\
& & \beta_{k-1} & \alpha_k
\end{pmatrix} = \text{tridiag}(\beta_j, \alpha_j, \beta_{j+1}), \]

the \( k \)-step Lanczos process yields

\[ AQ_k = Q_k T_k + f_k e_k^T, \quad f_k = \beta_k q_{k+1} \]  \hspace{1cm} (12)

and \( Q_k^T Q_k = I \) and \( Q_k^T q_{k+1} = 0. \)
Let

\[ T_ky = \mu y, \quad \|y\|_2 = 1. \]

Then

\[ A(Q_ky) = Q_kT_ky + f_k(e_k^T y) = \mu (Q_ky) + f_k(e_k^T y). \]

Here \( \mu \) is a Ritz value, and \( Q_ky \) is the corresponding Ritz vector.
**Error bound**

**Lemma.** Let $H$ be symmetric, and $Hz - \mu z = r$ and $z \neq 0$. Then

$$\min_{\lambda \in \lambda(H)} |\lambda - \mu| \leq \frac{\|r\|_2}{\|z\|_2}.$$ 

Proof: Let $H = U\Lambda U^T$ be the eigen-decomposition of $H$. Then

$$(H - \mu I)z = r \implies U(\Lambda - \mu I)U^T z = r \implies (\Lambda - \mu I)(U^T z) = U^T r.$$ 

Notice that $\Lambda - \mu I$ is diagonal. Thus

$$\|r\|_2 = \|U^T r\|_2 = \| (\Lambda - \mu I)(U^T z) \|_2$$

$$\geq \min_{\lambda \in \lambda(H)} |\lambda - \mu| \|U^T z\|_2$$

$$= \min_{\lambda \in \lambda(H)} |\lambda - \mu| \|z\|_2,$$

as expected.
Error bound

• If $f_k(e_k^T y) = 0$ for some $k$, then the associated Ritz value $\mu$ is an eigenvalue of $A$ with the corresponding eigenvector $Q_k y$.

• Let $\|r\|_2 = \|f_k(e_k^T y)\|_2$, then by the lemma, we know that for the Ritz pair $(\mu, Q_k y)$, there is an eigenvalue $\lambda$ of $A$, such that

$$|\lambda - \mu| \leq \frac{\|f_k(e_k^T y)\|_2}{\|Q_k y\|_2}$$
Lanczos method = RR + Lanczos

Simple Lanczos Algorithm:
1. \( q_1 = v / \|v\|_2 \), \( \beta_0 = 0 \), \( q_0 = 0 \);
2. for \( j = 1 \) to \( k \) do
3. \( w = Aq_j \);
4. \( \alpha_j = q_j^T w \);
5. \( w = w - \alpha_j q_j - \beta_{j-1} q_{j-1} \);
6. \( \beta_j = \|w\|_2 \);
7. if \( \beta_j = 0 \), quit;
8. \( q_{j+1} = w / \beta_j \);
9. Compute eigenvalues and eigenvectors of \( T_j \)
10. Test for convergence
11. endfor
Example

\[ A = \text{a random diagonal matrix} \quad A \text{ of order } n = 1000 \]
\[ v = (1, 1, \ldots, 1)^T \]

Convergence behavior:

![Graph showing convergence behavior](image-url)
We observe that

1. Extreme eigenvalues, i.e., the largest and smallest ones, converge first, and the interior eigenvalues converge last.

2. Convergence is monotonic, with the $i$th largest (smallest) eigenvalues of $T_k$ increasing (decreasing) to the $i$th largest (smallest) eigenvalue of $A$.

$\implies$ *Convergence analysis*
Thick Restarting

• Selects two indices \( \ell \) and \( u \) to indicate those Ritz values to be kept at both ends of spectrum:

\[
\begin{array}{c}
\theta_1 \ \theta_2 \ \ldots \ \theta_i \ \ldots \ \theta_u \ \ldots \ \theta_m \\
\text{keep} \quad \ldots \quad \text{discard} \quad \ldots \quad \text{keep}
\end{array}
\]

• The corresponding kept Ritz vectors are denoted by

\[
\hat{Q}_k = [\hat{q}_1, \hat{q}_2, \ldots, \hat{q}_k] = Q_m Y_k, \tag{13}
\]

where

\[
k = \ell + (m - u + 1), \tag{14}
\]

\[
Y_k = [y_1, y_2, \ldots, y_\ell, y_u, y_{u+1}, \ldots, y_m], \tag{15}
\]

and \( y_i \) is the eigenvector of \( T_m \) corresponding to \( \theta_i \).
• Sets these Ritz vectors $\hat{Q}_k$ as the first $k$ basis vectors at the restart and $\hat{q}_{k+1} = q_{m+1}$.

• To compute the $(k + 2)$th basis vector $\hat{q}_{k+2}$, TRLan computes $A\hat{q}_{k+1}$ and orthonormalizes it against the previous $k + 1$ basis vectors. That is,

$$\hat{\beta}_{k+1}\hat{q}_{k+2} = A\hat{q}_{k+1} - \hat{Q}_k(\hat{Q}_k^H A\hat{q}_{k+1}) - \hat{q}_{k+1}(\hat{q}_{k+1}^H A\hat{q}_{k+1}).$$

Note that $A\hat{Q}_k$ satisfies the relation:

$$A\hat{Q}_k = \hat{Q}_k D_k + \beta_m \hat{q}_{k+1} s^H,$$

where $D_k$ is the $k \times k$ diagonal matrix whose diagonal elements are the kept Ritz values, and $s = Y_k^H e_m$.

Thus, the coefficients $\hat{Q}_k^H A\hat{q}_{k+1}$ can be computed efficiently:

$$\hat{Q}_k^H A\hat{q}_{k+1} = (A\hat{Q}_k)^H \hat{q}_{k+1} = (\hat{Q}_k D_k + \beta_m \hat{q}_{k+1} s^H)^H \hat{q}_{k+1} = D_k Y_k^H (Q_m^H q_{m+1}) + \beta_m s (\hat{q}_{k+1}^H \hat{q}_{k+1}) = \beta_m s.$$
• Then after the first iteration after the restart, we have

\[
A\hat{Q}_{k+1} = \hat{Q}_{k+1}\hat{T}_{k+1} + \hat{\beta}_{k+1}\hat{q}_{k+2}e_{k+i}^H,
\]

where

\[
\hat{T}_{k+1} = \begin{bmatrix}
D_k & \beta_m s \\
\beta_m s^H & \alpha_{k+1}
\end{bmatrix}
\]

• In general, at the \(i\)th iteration after the restart, the new basis vector \(\hat{q}_{k+i+1}\) satisfies the relation:

\[
A\hat{Q}_{k+i} = \hat{Q}_{k+i}\hat{T}_{k+i} + \hat{\beta}_{k+i}\hat{q}_{k+i+1}e_{k+i}^H,
\]

where \(\hat{T}_{k+i} = \hat{Q}_{k+i}^HA\hat{Q}_{k+i}\) is of the form

\[
\hat{T}_{k+i} = \begin{pmatrix}
D_k & \beta_m s \\
\beta_m s^H & \alpha_{k+1} & \beta_{k+1} \\
\beta_{k+1} & \alpha_{k+2} & \beta_{k+2} \\
\vdots & \vdots & \vdots \\
\beta_{k+i-2} & \alpha_{k+i-1} & \beta_{k+i-1} \\
\beta_{k+i-1} & \alpha_{k+i}
\end{pmatrix}.
\]
• Note that the three-term recurrence is not valid only for computing the \((k + 2)\)th basis vector and is resumed afterward.