CHAPTER 11

EIGENVALUES AND EIGENVALUE PROBLEMS

SOLVED EXAMPLE PROBLEMS

for

NUMERICAL METHODS FOR SCIENTISTS AND ENGINEERS With Pseudocodes

By Zekeriya ALTAÇ

May 2025

EXAMPLE 11.1: Finding Eigenpairs of a real Matrix

Determine its eigen pairs of the given matrix $\mathbf{A}$.

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 2 & -3 \\ -3 & -1 & 4 & 1 \\ -3 & -1 & 7 & -2 \\ -3 & -7 & 4 & 7 \end{bmatrix}$$

SOLUTION:

We obtain the characteristic polynomial by setting $p_4(\lambda) = \det(\mathbf{A} - \lambda\mathbf{I}) = 0$. Using the co-factor expansion along any row or column yields:

$$\det(\mathbf{A} - \lambda\mathbf{I}) = \begin{vmatrix} 1-\lambda & 1 & 2 & -3 \\ -3 & -1-\lambda & 4 & 1 \\ -3 & -1 & 7-\lambda & -2 \\ -3 & -7 & 4 & 7-\lambda \end{vmatrix} = p_4(\lambda) = \lambda^4 - 14\lambda^3 + 67\lambda^2 - 126\lambda + 72 = 0$$

Next, we can apply a root finding algorithm to find the zeros of $p_4(\lambda)$. Using any numerical method, you should be able to find $\lambda = 1$, 3, 4, and 6. To find the eigenvector of λ_k, we solve the homogeneous system: $(\mathbf{A} - \lambda_k\mathbf{I})\mathbf{x}_k = \mathbf{0}$ for $k = 1$, 2, 3, and 4.

$$\text{For } \lambda_1 = 1 \implies \begin{bmatrix} 0 & 1 & 2 & -3 \\ -3 & -2 & 4 & 1 \\ -3 & -1 & 6 & -2 \\ -3 & -7 & 4 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \mathbf{x}_1 = (1, 1, 1, 1)x_1$$

$$\text{For } \lambda_2 = 3 \implies \begin{bmatrix} -2 & 1 & 2 & -3 \\ -3 & -4 & 4 & 1 \\ -3 & -1 & 4 & -2 \\ -3 & -7 & 4 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \mathbf{x}_1 = (1, -1, 0, -1)x_1$$

$$\text{For } \lambda_3 = 4 \implies \begin{bmatrix} -3 & 1 & 2 & -3 \\ -3 & -5 & 4 & 1 \\ -3 & -1 & 3 & -2 \\ -3 & -7 & 4 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \mathbf{x}_1 = (0, 1, 1, 1)x_2$$

$$\text{For } \lambda_4 = 6 \implies \begin{bmatrix} -5 & 1 & 2 & -3 \\ -3 & -7 & 4 & 1 \\ -3 & -1 & 1 & -2 \\ -3 & -7 & 4 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \mathbf{x}_1 = (1, 0, 1, -1)x_1$$

Note that the eigenvectors are expressed in terms of x_1 or x_2; that is, an eigenvalue can have an infinite number of eigenvectors (except $x_1 = 0$ or $x_2 = 0$), but they all lie along the same eigenspace, which is a linear subspace of the vector space. One can simply set $x_1 = 1$ or $x_2 = 1$ to find eigenvectors.

Discussion: Eigenvectors can be scaled by any non-zero constant and still be valid. Normalizing gives a unique representative (usually of unit length) to avoid ambiguity, i.e., $\mathbf{x}_k / \|\mathbf{x}_k\|_2 = 1$. In many practical applications, unit vectors are easier to interpret geometrically or probabilistically. A normalized eigenvector has a magnitude (length) of 1, which helps with projections, rotations, and understanding direction only. In computations or algorithms, using normalized vectors can prevent numerical overflow or underflow. Normalized eigenvectors define directions (principal components) along which variance is measured; in this regard, normalization makes comparisons meaningful.

EXAMPLE 11.2: Estimating Eigenvalue Locations using Gerschgorin Circles

Estimate the eigenvalue locations of the following symmetric tridiagonal 5×5 matrix:

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 2 \end{bmatrix}$$

SOLUTION:

Gerschgorin Circle Theorem states that each eigenvalue of a matrix $\mathbf{A} = [a_{ij}]$ lies within at least one of the Gerschgorin circles:

$$C(a_{ii}, R_i), \quad \text{where } r_i = \sum_{j \neq i} |a_{ij}|$$

For the matrix $\mathbf{A}$, the Gerschgorin intervals are computed row-wise as follows:

- Row 1: center $a_{11} = 2$, $r_1 = |-1| = 1 \Rightarrow C_1 : \{|z-2| \leqslant 1\}$; thus, interval: $[1, 3]$
- Row 2: center $a_{22} = 2$, $r_2 = |-1| + |-1| = 2 \Rightarrow C_2 : \{|z-2| \leqslant 2\}$; thus, $[0, 4]$
- Row 3: center $a_{33} = 2$, $r_3 = |-1| + |-1| = 2 \Rightarrow C_3 : \{|z-2| \leqslant 2\}$; thus, $[0, 4]$
- Row 4: center $a_{44} = 2$, $r_4 = |-1| + |-1| = 2 \Rightarrow C_4 : \{|z-2| \leqslant 2\}$; thus, $[0, 4]$
- Row 5: center $a_{55} = 2$, $r_5 = |-1| = 1 \Rightarrow C_5 : \{|z-2| \leqslant 1\}$; thus, interval: $[1, 3]$

Notice that the column-wise intervals will be the same since the matrix symmetric.

Discussion: Estimating the intervals (or bounds) in which eigenvalues lie is extremely valuable in theory and applications. Computing exact eigenvalues can be expensive, especially for large matrices. Estimating intervals where eigenvalues lie gives quick insight without full eigenvalue decomposition. In differential equations, control systems, or vibrations, we often do not need exact eigenvalues—just knowing if they lie in certain intervals (e.g., all negative, all inside the unit circle) is enough to prove stability, predict oscillations or divergence, or understand response times or resonance risks.

Many iterative algorithms (like Power Method, Conjugate Gradient, GMRES) perform better or only converge under certain eigenvalue distributions. Knowing the interval of eigenvalues helps choose better step sizes or tune parameters. Condition number (which affects numerical error) is based on eigenvalue spread. In structural analysis, eigenvalues relate to natural frequencies. Knowing intervals helps ensure that designs avoid resonant frequencies, and so on.

One of the common tools to estimate eigenvalue intervals is the Gershgorin Circle Theorem, which gives a region (disk) where each eigenvalue must lie. In this example, the Gerschgorin's theorem predicts that all eigenvalues of $\mathbf{A}$ lie within the union of the intervals: [0, 4]. The given matrix is a *tridiagonal Toeplitz matrix*i whose eigenpairs can be calculated analytically. Recall that the exact eigenvalues of this matrix are given by the formula (*see* Section 2.1.6):

$$\lambda_k = 2\left(1 - \cos\left(\frac{k\pi}{6}\right)\right), \quad k = 1, 2, 3, 4, 5$$

These evaluate approximately to $\{0.2679,\ 1,\ 2,\ 3,\ 3.732\}$, which are indeed lie within the interval $[0, 4]$, confirming the validity of the Gerschgorin estimate.

EXAMPLE 11.3: Finding the Dominant Eigenvalue

Consider matrix $\mathbf{A}$ in **Example 11.4**. (i) Using the initial guess vector $\mathbf{x}^{(0)} = [\,1\ 1\ 0\ 0\,]^T$ estimate the dominant (largest magnitude) eigenvalue of matrix $\mathbf{A}$ using the Power method with Rayleigh quotient. (ii) Repeat the procedure for $\mathbf{x}^{(0)} = [\,1\ 0\ 0\ 0\,]^T$. Use $|\lambda^{(p+1)} - \lambda^{(p)}| < 10^{-5}$ for convergence.

SOLUTION:

(i) We begin with the starting guess of $\mathbf{x}^{(0)} = [\,1\ 1\ 0\ 0\,]^T$. We first evaluate

$$\mathbf{w} = \mathbf{A}\mathbf{x}^{(0)} = \begin{bmatrix} 1 & 1 & 2 & -3 \\ -3 & -1 & 4 & 1 \\ -3 & -1 & 7 & -2 \\ -3 & -7 & 4 & 7 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ -4 \\ -4 \\ -10 \end{bmatrix}$$

Then we calculate $(\mathbf{w}, \mathbf{x}^{(0)}) = -2$ and $(\mathbf{x}^{(0)}, \mathbf{x}^{(0)}) = 2$. The first eigenvalue estimate is obtained as follows:

$$\lambda^{(1)} = \frac{(\mathbf{x}^{(0)}, \mathbf{w})}{(\mathbf{x}^{(0)}, \mathbf{x}^{(0)})} = \frac{-2}{2} = -1$$

Next, we normalized this vector estimate as $\mathbf{x}^{(1)} = \mathbf{w}/\|\mathbf{w}\|_\infty = [\,-1/5\ \ 2/5\ \ 2/5\ \ 1\,]^T$. The product vector $\mathbf{w}$ at the end of the second iteration becomes

$$\mathbf{w} = \mathbf{A}\mathbf{x}^{(1)} = \begin{bmatrix} 1 & 1 & 2 & -3 \\ -3 & -1 & 4 & 1 \\ -3 & -1 & 7 & -2 \\ -3 & -7 & 4 & 7 \end{bmatrix} \begin{bmatrix} -1/5 \\ 2/5 \\ 2/5 \\ 1 \end{bmatrix} = \begin{bmatrix} -2 \\ 14/5 \\ 1 \\ 32/5 \end{bmatrix}$$

This leads to $(\mathbf{x}^{(1)}, \mathbf{w}) = 208/25$ and $(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) = 34/25$. Then, the second estimate for the dominant eigenvalue is obtained as follows:

$$\lambda^{(2)} = \frac{(\mathbf{x}^{(1)}, \mathbf{w})}{(\mathbf{x}^{(1)}, \mathbf{x}^{(1)})} = \frac{104}{17} = 6.1176471$$

Likewise, the estimates for the dominant eigenvalue and associated eigenvector in the subsequent iterations are obtained in the same manner and summarized below:

$$\begin{aligned}
\lambda^{(3)} &= 5.4565056 & \mathbf{x}^{(3)} &= [-0.465909 \quad 0.3863640 \quad -0.073864 \quad 1\,]^T \\
\lambda^{(4)} &= 5.5952039 & \mathbf{x}^{(4)} &= [-0.597895 \quad 0.3178950 \quad -0.278947 \quad 1\,]^T \\
\lambda^{(5)} &= 5.7737836 & \mathbf{x}^{(5)} &= [-0.703861 \quad 0.2494210 \quad -0.454247 \quad 1\,]^T \\
\lambda^{(6)} &= 5.8885484 & \mathbf{x}^{(6)} &= [-0.786306 \quad 0.1883650 \quad -0.597906 \quad 1\,]^T \\
\lambda^{(7)} &= 5.9491769 & \mathbf{x}^{(7)} &= [-0.848641 \quad 0.1378950 \quad -0.710741 \quad 1\,]^T \\
\lambda^{(8)} &= 5.9781442 & \mathbf{x}^{(8)} &= [-0.894475 \quad 0.0984832 \quad -0.795991 \quad 1\,]^T \\
\lambda^{(9)} &= 5.9911042 & \mathbf{x}^{(9)} &= [-0.927349 \quad 0.0690144 \quad -0.858335 \quad 1\,]^T \\
\lambda^{(10)} &= 5.9966048 & \mathbf{x}^{(10)} &= [-0.950456 \quad 0.0476838 \quad -0.902772 \quad 1\,]^T \\
\lambda^{(11)} &= 5.9988203 & \mathbf{x}^{(11)} &= [-0.966448 \quad 0.0326074 \quad -0.933840 \quad 1\,]^T \\
\lambda^{(12)} &= 5.9996573 & \mathbf{x}^{(12)} &= [-0.977390 \quad 0.0221328 \quad -0.955257 \quad 1\,]^T \\
\lambda^{(13)} &= 5.9999442 & \mathbf{x}^{(13)} &= [-0.984816 \quad 0.0149438 \quad -0.969872 \quad 1\,]^T \\
\lambda^{(14)} &= 6.0000254 & \mathbf{x}^{(14)} &= [-0.989827 \quad 0.0100523 \quad -0.979775 \quad 1\,]^T \\
\lambda^{(15)} &= 6.0000371 & \mathbf{x}^{(15)} &= [-0.993195 \quad 0.0067442 \quad -0.986451 \quad 1\,]^T \\
\lambda^{(16)} &= 6.0000296 & \mathbf{x}^{(16)} &= [-0.995453 \quad 0.0045163 \quad -0.990937 \quad 1\,]^T
\end{aligned}$$

We already know the largest eigenvalue of the matrix **A** from the solution of **Example 11.1**. The numerical solution yields the largest eigenvalue of 6 and satisfies the convergence criterion in 16 iterations. However, the final associated eigenvector is not as accurate as the eigenvalue yet.

(ii) We recalculate the dominant eigenvalue using the starting guess of $\mathbf{x}^{(0)} = [\,1\ 0\ 0\ 0\,]^T$. The dominant eigenvalue estimate along with its associated eigenvector computed at every iteration step until convergence is achieved are presented below.

$$
\begin{aligned}
\lambda^{(1)} &= 1.0000000 \quad & \mathbf{x}^{(1)} &= [-0.333333 \quad 1 \quad 1 \quad 1\,]^T \\
\lambda^{(2)} &= 4.8571429 \quad & \mathbf{x}^{(2)} &= [-0.066667 \quad 1 \quad 1 \quad 1\,]^T \\
\lambda^{(3)} &= 4.1952663 \quad & \mathbf{x}^{(3)} &= [-0.015873 \quad 1 \quad 1 \quad 1\,]^T \\
\lambda^{(4)} &= 4.0473631 \quad & \mathbf{x}^{(4)} &= [-0.003922 \quad 1 \quad 1 \quad 1\,]^T \\
\lambda^{(5)} &= 4.0117493 \quad & \mathbf{x}^{(5)} &= [-0.000978 \quad 1 \quad 1 \quad 1\,]^T \\
\lambda^{(6)} &= 4.0029316 \quad & \mathbf{x}^{(6)} &= [-0.000244 \quad 1 \quad 1 \quad 1\,]^T \\
\lambda^{(7)} &= 4.0007325 \quad & \mathbf{x}^{(7)} &= [-0.000061 \quad 1 \quad 1 \quad 1\,]^T \\
\lambda^{(8)} &= 4.0001831 \quad & \mathbf{x}^{(8)} &= [-0.000015 \quad 1 \quad 1 \quad 1\,]^T \\
\lambda^{(9)} &= 4.0000458 \quad & \mathbf{x}^{(9)} &= [-0.000004 \quad 1 \quad 1 \quad 1\,]^T \\
\lambda^{(10)} &= 4.0000114 \quad & \mathbf{x}^{(10)} &= [-3.43\times10^{-5} \quad 1 \quad 1 \quad 1\,]^T \\
\lambda^{(11)} &= 4.0000029 \quad & \mathbf{x}^{(11)} &= [-8.58\times10^{-6} \quad 1 \quad 1 \quad 1\,]^T
\end{aligned}
$$

Notice that the converged eigenpair, $\lambda = 4$ and **x**=[0 1 1 1], is one of the eigenpairs of matrix **A** but not that of the dominant one. This is because after the first iteration the estimated eigenvector aligns with the eigenvector of $\lambda = 4$. Recall that the Power method can converge to the wrong eigenpair if the initial guess or subsequent eigenvector coincides with an eigenvector other than the dominant eigenvalue of **A**.

Discussion: The *Power method* is a simple and widely used iterative algorithm to find the dominant eigenvalue (i.e., the eigenvalue with the largest absolute value) and the associated eigenvector of a matrix. There is a unique dominant eigenvalue if the eigenvalue with the largest absolute value is strictly greater than all others, i.e., $|\lambda_1| > |\lambda_2| > \cdots |\lambda_n|$. In this case, the power method will converge to the dominant eigenvalue λ_1, and the associated eigenvector. The smaller the dominance ratio $|\lambda_2/\lambda_1|$, the faster the method will converge.

The method, however, may not always work perfectly in all scenarios and can fail under certain conditions. In this example, as $|\lambda_2/\lambda_1| = 2/3 < 1$, the method converges fairly quickly.

(i) If there is no unique dominant eigenvalue. For example, if $|\lambda_1| = |\lambda_2|$, the method might not converge or might oscillate. For a matrix with complex conjugate eigenvalues of equal magnitude (e.g., rotation matrices), the method can also diverge or cycle.

(ii) The initial vector is orthogonal to the dominant eigenvector. In this case, the projection onto the dominant eigenspace is zero, and the dominant component never shows up. Then it may converge to the next-most dominant eigenvalue/eigenvector (i.e., the eigenvalue with the second-largest absolute value) if the initial vector has a component in that direction.

(iii) Dominant eigenvalue is complex. If a matrix has a complex dominant eigenvalue and you are using real arithmetic, the method will not converge to a complex eigenpair. Instead, it may result in a cyclic or spiraling behavior.

EXAMPLE 11.4: Constructing a Matrix of known Eigenvalues

Construct a 4×4 matrix $\mathbf{A}$ that has known eigenvalues $\lambda_1 = 2, \lambda_2 = 3, \lambda_3 = 3, \lambda_4 = 3$.

SOLUTION:

In this example, we will make use of the similarity transformation. A matrix $\mathbf{A}$ is similar to a diagonal matrix $\mathbf{D}$ if there exists an invertible matrix $\mathbf{P}$ such that:

$$\mathbf{A} = \mathbf{PDP}^{-1}$$

where $\mathbf{D}$ is a diagonal matrix consisting of the eigenvalues.

We choose an invertible matrix $\mathbf{P}$ (and find its inverse) as follows:

$$\mathbf{P} = \begin{bmatrix} 1 & -2 & 2 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 0 & 1 & 2 \\ 1 & 0 & 1 & 1 \end{bmatrix}, \qquad \mathbf{P}^{-1} = \begin{bmatrix} -1 & -2 & -5 & 9 \\ 0 & 1 & 2 & -3 \\ 1 & 2 & 4 & -7 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

Note that the determinant of $\mathbf{P}$ is 1. We then compute:

$$\mathbf{A} = \mathbf{PDP}^{-1} = \begin{bmatrix} 1 & -2 & 2 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 0 & 1 & 2 \\ 1 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} -1 & -2 & -5 & 9 \\ 0 & 1 & 2 & -3 \\ 1 & 2 & 4 & -7 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & -1 & -2 & -5 \\ 1 & 2 & 1 & 3 \\ 2 & 1 & 4 & 4 \\ 1 & 0 & 1 & 4 \end{bmatrix}$$

Discussion: The matrix $\mathbf{P}$ in the similarity transformation $\mathbf{A}$=$\mathbf{PDP}^{-1}$ must meet certain conditions to ensure that everything works correctly. Key conditions for choosing $\mathbf{P}$ are

(a) $\mathbf{P}$ must be invertible. The similarity transformation requires $\mathbf{P}^{-1}$ to exist, i.e., det($\mathbf{P}$)$\neq 0$.

(b) Columns of $\mathbf{P}$ are eigenvectors (if constructing from known $\mathbf{A}$). If matrix $\mathbf{A}$ is given and we want to diagonalize it, $\mathbf{P}$ should be made from the eigenvectors of $\mathbf{A}$. In that case, $\mathbf{D}$ ends up being a diagonal matrix of eigenvalues.

(c) We can choose an arbitrary $\mathbf{P}$ when starting from $\mathbf{D}$. If you're constructing a matrix $\mathbf{A}$ from scratch (i.e., given eigenvalues and building $\mathbf{A}$=$\mathbf{PDP}^{-1}$, you can choose any invertible matrix $\mathbf{P}$. This means there are infinitely many matrices $\mathbf{A}$ that share the same eigenvalues—they are all similar to $\mathbf{D}$ but look different.

(d) Orthogonality (Optional but Nicer). If you choose $\mathbf{P}$ to be orthogonal (i.e., $\mathbf{P}^{-1}$=$\mathbf{P}^T$, the computation becomes cleaner:

(e) Optional: Symmetric Matrix Construction. To construct a real symmetric matrix with known eigenvalues, choose an orthogonal matrix $\mathbf{Q}$ (e.g., from orthonormalized random vectors), and define:

$$\mathbf{A} = \mathbf{Q}^T\mathbf{DQ}$$

This guarantees that $\mathbf{A}$ is symmetric and has real eigenvalues matching those in $\mathbf{D}$.

Warning! Not all matrices are diagonalizable, so if you're given $\mathbf{A}$, it may not be possible to find such a $\mathbf{P}$. But if you are constructing $\mathbf{A}$ from a diagonal matrix, you are always good. The matrix $\mathbf{P}$ is not unique—any invertible change of basis will result in a different matrix similar to $\mathbf{D}$.

EXAMPLE 11.5: Applying Jacobi Method

Apply the Jacobi method to find the eigenpairs of the following symmetric matrix:

$$\begin{bmatrix} 3.5 & 1 & 1.5 & 0 \\ 1 & 3.5 & 0 & 1.5 \\ 1.5 & 0 & 3.5 & 1 \\ 0 & 1.5 & 1 & 3.5 \end{bmatrix}$$

SOLUTION:

The Jacobi method can be employed to find eigenpairs of a symmetric matrix.

Step 1. Initialization. This step requires establishing the initial values of $\mathbf{A}$ and $\mathbf{X}$. This is done simply by setting $\mathbf{A}_0 = \mathbf{A}$ and $\mathbf{X}_0 = \mathbf{I}$, which will accumulate the eigenvectors. For accuracy tolerance, we assume $\varepsilon = 10^{-6}$

Step 2. Find the largest off-diagonal element. Search for the largest (in absolute value) of the off-diagonal elements and record its location (p, q). The off-diagonal elements are $|a_{12}| = |a_{34}| = 1$, $|a_{14}| = |a_{23}| = 0$, and $|a_{13}| = |a_{24}| = 1.5$. We will pick $|a_{13}| = 1.5$ as the element with the largest magnitude (we could have picked $|a_{24}|$ as well). In this case, we end up with $a_{33} = a_{11} = 3.5$ and $a_{13} = 1.5$.

Step 3. Find the rotation angle. This algorithm basically requires the sine and cosine of the rotation angle rather (s and c) than the rotation angle itself. For this purpose, we compute α and β:

$$\alpha = \frac{a_{33} - a_{11}}{2} = \frac{3.5 - 3.5}{2} = 0, \quad \beta = \sqrt{a_{13}^2 + \alpha^2} = \sqrt{1.5^2 + 0^2} = 1.5$$

Then, the rotation sine and cosine are obtained from Eq. (11.44) and (11.45) as follows:

$$c = \sqrt{\frac{1}{2} + \frac{|\alpha|}{2\beta}} = \sqrt{\frac{1}{2} + \frac{0}{2(1.5)}} = \frac{1}{\sqrt{2}}, \quad s = -\frac{a_{13}}{2\,\beta\,c} = -\frac{1.5}{2(1.5)(1/\sqrt{2})} = -\frac{1}{\sqrt{2}}$$

Step 4. Construct the rotation matrix $\mathbf{U}$. The matrix $\mathbf{U}$ is an identity matrix, modified at rows and columns 1 and 3 as follows:

$$\mathbf{U} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 1 & 0 & 0 \\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Step 5. Update $\mathbf{A}$ and $\mathbf{X}$. New matrices are determined by $\mathbf{A}_k = \mathbf{U}^T \mathbf{A}_{k-1} \mathbf{U}$ and $\mathbf{X}_k = \mathbf{X}_{k-1} \mathbf{U}$ for $k > 0$. Then, we obtain $\mathbf{A}_1$ as

$$\mathbf{A}_1 = \mathbf{U}^T \mathbf{A}_0 \mathbf{U} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^T \begin{bmatrix} 3.5 & 1 & 1.5 & 0 \\ 1 & 3.5 & 0 & 1.5 \\ 1.5 & 0 & 3.5 & 1 \\ 0 & 1.5 & 1 & 3.5 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{A}_1 = \begin{bmatrix} 5 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 3.5 & -\frac{1}{\sqrt{2}} & 1.5 \\ 0 & -\frac{1}{\sqrt{2}} & 2 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 1.5 & \frac{1}{\sqrt{2}} & 3.5 \end{bmatrix}$$

and $\mathbf{X}_1$ becomes

$$\mathbf{X}_1 = \mathbf{X}_0\mathbf{U} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Step 6. Check for convergence. Steps 2-5 are repeated until the matrix becomes approximately diagonal, that is, the off-diagonal elements are near zero. Convergence is verified by checking whether the threshold value is met. The threshold value after the first iteration becomes

$$T_1 = (1/4)\sqrt{(\tfrac{1}{\sqrt{2}})^2 + (\tfrac{1}{\sqrt{2}})^2 + (-\tfrac{1}{\sqrt{2}})^2 + (3/2)^2 + (\tfrac{1}{\sqrt{2}})} = \sqrt{17}/8 = 0.51539 > \varepsilon$$

The *second iteration* starts with $\mathbf{A}_1$. The largest off-diagonal element and its location are searched. The element with the largest magnitude is $|a_{24}| = 1.5$. We also deduce $a_{44} = a_{22} = 3.5$ and $a_{42} = 1.5$. The values of α and β determined

$$\alpha = \frac{a_{44} - a_{22}}{2} = \frac{3.5 - 3.5}{2} = 0, \quad \beta = \sqrt{a_{24}^2 + \alpha^2} = \sqrt{1.5^2 + 0^2} = 1.5$$

and the rotation sine and cosine, as before, are calculated from Eq. (11.44) and (11.45):

$$c = \sqrt{\frac{1}{2} + \frac{|\alpha|}{2\beta}} = \sqrt{\frac{1}{2} + \frac{0}{2(1.5)}} = \frac{1}{\sqrt{2}}, \quad s = -\frac{a_{13}}{2\,\beta\,c} = -\frac{1.5}{2(1.5)(1/\sqrt{2})} = -\frac{1}{\sqrt{2}}$$

We can now construct the rotation matrix $\mathbf{U}$ by modifying the second and last rows and columns of the identity matrix:

$$\mathbf{U} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 & 1 & 0 \\ 0 & -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \end{bmatrix}$$

Finally, we find the updated matrix $\mathbf{A}_2$

$$\mathbf{A}_2 = \mathbf{U}^T\mathbf{A}_1\mathbf{U} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 & 1 & 0 \\ 0 & -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \end{bmatrix}^T \begin{bmatrix} 5 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 3.5 & -\frac{1}{\sqrt{2}} & 1.5 \\ 0 & -\frac{1}{\sqrt{2}} & 2 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 1.5 & \frac{1}{\sqrt{2}} & 3.5 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 & 1 & 0 \\ 0 & -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \end{bmatrix}$$

$$\mathbf{A}_2 = \begin{bmatrix} 5 & 1 & 0 & 0 \\ 1 & 5 & 0 & 0 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}$$

and the matrix $\mathbf{X}_2$:

$$\mathbf{X}_2 = \mathbf{X}_1\mathbf{U} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \end{bmatrix}$$

For the second iteration, the thresold value becomes $T_2 = (1/4)\sqrt{1^2 + 1^2} = \sqrt{2}/4 = 0.35355 > \varepsilon$.

Repeating this processes in the same manner, we obtain

$$\mathbf{A}_3 = \begin{bmatrix} 6 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix} \qquad \mathbf{X}_3 = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} & -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & -\frac{1}{\sqrt{2}} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & \frac{1}{\sqrt{2}} \end{bmatrix}, \quad T_3 = 1/4$$

$$\mathbf{A}_4 = \begin{bmatrix} 6 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{X}_4 = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}, \quad T_4 = 0$$

Discussion: The Jacobi method is a classical iterative technique used for finding eigenvalues and eigenvectors of a symmetric matrix. While it is useful in many situations, it comes with both advantages and disadvantages.

Simplicity and Conceptual Clarity. The method is relatively simple to understand and implement. The idea behind it is straightforward: rotate the matrix to zero out off-diagonal elements and iteratively improve the approximation of the eigenvalues and eigenvectors.

Works for Symmetric Matrices. The Jacobi method is only effective for symmetric matrices. If the matrix is symmetric, all the eigenvalues are real, and the eigenvectors can be orthogonal. The Jacobi method guarantees convergence to the exact eigenvalues and eigenvectors for symmetric matrices. It converges for all symmetric matrices within numerical precision limits after a sufficient number of iterations. (For nonsymmetric matrices, other techniques such as the QR algorithm or Power method are more appropriate.)

No Need for Initial Estimates. Unlike some other iterative methods (like power iteration or inverse iteration), the Jacobi method does not require an initial guess for the eigenvalues or eigenvectors. This makes the method simpler and more useful for situations where no prior knowledge of the matrix is known.

Computational cost. Convergence speed can be very slow, especially for large matrices or matrices with eigenvalues that are very close to each other. Each rotation typically only eliminates a small off-diagonal element, so it requires many iterations to make significant progress, which can make the method impractical for large problems. As a result, it can be computationally expensive for large matrices. The total number of operations grows rapidly as the size of the matrix increases, making it less efficient than more modern methods like the *QR decomposition* algorithm for large problems.

Memory and storage requirements. Storing and manipulating large matrices in this process can be memory-intensive, which makes it less suitable for very large-scale problems. The method also requires a suitable stopping criterion (e.g., a sufficiently small off-diagonal element). If the stopping criterion is too strict, it might lead to unnecessary computations, and if it's too loose, the result may not be accurate enough.

EXAMPLE 11.6: Applying the Householder Method

Apply the Householder method to transform the following 5×5 matrix into a tridiagonal matrix.

$$\mathbf{A}=\begin{bmatrix} 9 & -1 & 0 & -2 & -2 \\ -1 & 11 & -1 & -2 & -2 \\ 0 & -1 & 7 & -2 & -1 \\ -2 & -2 & -2 & 4 & -2 \\ -2 & -2 & -1 & -2 & 8 \end{bmatrix}$$

SOLUTION:

Given a symmetric 5×5 matrix $\mathbf{A}$, we will use Householder reflections to reduce it to a tridiagonal matrix $\mathbf{T}$, such that $\mathbf{T}=\mathbf{Q}^T\mathbf{AQ}$, where $\mathbf{Q}$ is an orthogonal matrix composed of Householder reflectors.

Step 1. We want to zero out entries below a_{21} (i.e., a_{31}, a_{41}, and a_{51}). The vector $\mathbf{x}$ is defined from $a_{2:5,1}$ (column elements) as

$$\mathbf{x}=\begin{bmatrix} -1 \\ 0 \\ -2 \\ -2 \end{bmatrix}$$

Step 2. To compute the Householder vector $\mathbf{u}$, we let

$$S=\text{sgn}(a_{21})\sqrt{a_{21}^2+a_{31}^2+a_{41}^2+a_{51}^2}=-\sqrt{(-1)^2+(0)^2+(-2)^2+(-2)^2}=-3$$

and $\mathbf{e}_1=[\,1\;\;0\;\;0\;\;0\,]^T$.

The vector $\mathbf{u}$ is then constructed as follows:

$$\mathbf{u}=\mathbf{x}+S\,\mathbf{e}_1=\begin{bmatrix} -1 \\ 0 \\ -2 \\ -2 \end{bmatrix}+(-3)\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}=\begin{bmatrix} -4 \\ 0 \\ -2 \\ -2 \end{bmatrix}$$

Step 3. To construct the Householder matrix $\mathbf{P}$, we first evaluate $d=S(S+a_{21})=(-3)(-3-1)=12$ and then $\mathbf{uu}^T$:

$$\mathbf{uu}^T=\begin{bmatrix} -4 \\ 0 \\ -2 \\ -2 \end{bmatrix}[-4\;\;0\;\;-2\;\;-2]=\begin{bmatrix} 16 & 0 & 8 & 8 \\ 0 & 0 & 0 & 0 \\ 8 & 0 & 4 & 4 \\ 8 & 0 & 4 & 4 \end{bmatrix}$$

Substituting $\mathbf{uu}^T$ into Eq. (11.56), we get

$$\mathbf{I}-\frac{1}{d}\mathbf{uu}^T=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}-\frac{1}{12}\begin{bmatrix} 16 & 0 & 8 & 8 \\ 0 & 0 & 0 & 0 \\ 8 & 0 & 4 & 4 \\ 8 & 0 & 4 & 4 \end{bmatrix}=\begin{bmatrix} -\frac{1}{3} & 0 & -\frac{2}{3} & -\frac{2}{3} \\ 0 & 1 & 0 & 0 \\ -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{3} \\ -\frac{2}{3} & 0 & -\frac{1}{3} & \frac{2}{3} \end{bmatrix}$$

Now, the first transformation matrix is found as

$$\mathbf{P}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{3} & 0 & -\frac{2}{3} & -\frac{2}{3} \\ 0 & 0 & 1 & 0 & 0 \\ 0 & -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{3} \\ 0 & -\frac{2}{3} & 0 & -\frac{1}{3} & \frac{2}{3} \end{bmatrix}$$

Step 4. Now we apply the orthogonal transformation (i.e., $\mathbf{P}_1\mathbf{A}\mathbf{P}_1$ product) that yields

$$\mathbf{A}_2 = \mathbf{P}_1\mathbf{A}_1\mathbf{P}_1 = \begin{bmatrix} 9 & 3 & 0 & 0 & 0 \\ 3 & 3 & \frac{7}{3} & \frac{4}{3} & -\frac{4}{3} \\ 0 & \frac{7}{3} & 7 & -\frac{1}{3} & \frac{2}{3} \\ 0 & \frac{4}{3} & -\frac{1}{3} & \frac{28}{3} & 2 \\ 0 & -\frac{4}{3} & \frac{2}{3} & 2 & \frac{32}{3} \end{bmatrix}$$

The transformation matrix $\mathbf{P}_2$ is constructed by repeating the four-step procedure above. We choose $\mathbf{x} = [\ 7/3 \quad 4/3 \quad -4/3\ ]$ and $\mathbf{e}_1 = [\ 1 \ \ 0 \ \ 0\ ]$. Then, S and d are computed as

$$S = \text{sgn}(a_{32})\sqrt{a_{32}^2 + a_{42}^2 + a_{52}^2} = \sqrt{\left(\frac{7}{3}\right)^2 + \left(\frac{4}{3}\right)^2 + \left(-\frac{4}{3}\right)^2} = 3$$

$$d = S(S + a_{32}) = 3\left(3 + \frac{7}{3}\right) = 16$$

The second step Householder vector $\mathbf{u}$ is found as

$$\mathbf{u} = \mathbf{x} + S\,\mathbf{e}_1 = \begin{bmatrix} \frac{7}{3} \\ \frac{4}{3} \\ -\frac{4}{3} \end{bmatrix} + (3)\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{16}{3} \\ \frac{4}{3} \\ -\frac{4}{3} \end{bmatrix}$$

Next, $\mathbf{u}\mathbf{u}^T$ is obtained:

$$\mathbf{u}\mathbf{u}^T = \begin{bmatrix} \frac{16}{3} \\ \frac{4}{3} \\ -\frac{4}{3} \end{bmatrix}\begin{bmatrix} \frac{16}{3} & \frac{4}{3} & -\frac{4}{3} \end{bmatrix}^T = \begin{bmatrix} \frac{256}{9} & \frac{64}{9} & -\frac{64}{9} \\ \frac{64}{9} & \frac{16}{9} & -\frac{16}{9} \\ -\frac{64}{9} & -\frac{16}{9} & \frac{16}{9} \end{bmatrix}$$

Finally, the Householder matrix $\mathbf{P}_2$ is constructed.

$$\mathbf{I}-\frac{1}{d}\mathbf{u}\mathbf{u}^T=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}-\frac{1}{16}\begin{bmatrix}\frac{256}{9}&\frac{64}{9}&-\frac{64}{9}\\\frac{64}{9}&\frac{16}{9}&-\frac{16}{9}\\-\frac{64}{9}&-\frac{16}{9}&\frac{16}{9}\end{bmatrix}=\begin{bmatrix}-\frac{7}{9}&-\frac{4}{9}&\frac{4}{9}\\-\frac{4}{9}&\frac{8}{9}&\frac{1}{9}\\\frac{4}{9}&\frac{1}{9}&\frac{8}{9}\end{bmatrix}$$

Next, the second Householder transformation matrix is found as

$$\mathbf{P}_2=\begin{bmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&-\frac{7}{9}&-\frac{4}{9}&\frac{4}{9}\\0&0&-\frac{4}{9}&\frac{8}{9}&\frac{1}{9}\\0&0&\frac{4}{9}&\frac{1}{9}&\frac{8}{9}\end{bmatrix}$$

The second orthogonal transformation (i.e., $\mathbf{P}_1\mathbf{A}\mathbf{P}_2$ product) yields

$$\mathbf{A}_3=\mathbf{P}_2\mathbf{A}_2\mathbf{P}_2=\begin{bmatrix}9&3&0&0&0\\3&3&-3&0&0\\0&-3&\frac{181}{27}&-\frac{2}{27}&\frac{11}{27}\\0&0&-\frac{2}{27}&\frac{256}{27}&\frac{50}{27}\\0&0&\frac{11}{27}&\frac{50}{27}&\frac{292}{27}\end{bmatrix}$$

We repeat the above process to construct the Householder vector. Then, by computing $\mathbf{P}_3$ and applying similarity transformation ($\mathbf{P}_3\mathbf{A}_3\mathbf{P}_3\rightarrow\mathbf{A}_4$) on the bottom-right 3×3 submatrix, we get

$$\mathbf{P}_3=\begin{bmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&1&0&0\\0&0&0&-\frac{2}{5\sqrt{5}}&\frac{11}{5\sqrt{5}}\\0&0&0&\frac{11}{5\sqrt{5}}&\frac{2}{5\sqrt{5}}\end{bmatrix},\quad \mathbf{A}_4=\begin{bmatrix}9&3&0&0&0\\3&3&-3&0&0\\0&-3&\frac{181}{27}&\frac{5\sqrt{5}}{27}&0\\0&0&\frac{5\sqrt{5}}{27}&\frac{34156}{3375}&\frac{246}{125}\\0&0&0&\frac{246}{125}&\frac{1272}{125}\end{bmatrix}$$

Discussion: The Householder tridiagonalization method is a cornerstone in numerical linear algebra, especially in the eigenvalue problem for symmetric matrices. Its importance lies in how it enables accurate, efficient, and stable computation of eigenvalues and eigenvectors.

Efficient Reduction for Eigenvalue Algorithms. Most eigenvalue algorithms (like the QR algorithm) are computationally expensive when applied directly to full matrices. By first reducing a symmetric matrix to tridiagonal form, the structure becomes simpler (only three diagonals); subsequent computations (like QR iterations) are much faster, reducing complexity from $\mathcal{O}(h^3)$ to $\mathcal{O}(h^2)$ per iteration, which dramatically speeds up eigenvalue computations.

Numerical Stability. Householder transformations are orthogonal (or unitary in complex space), meaning that they preserve the 2-norm and condition number, avoid numerical instabilities caused by round-off errors, and ensure backward stability, i.e., the computed results are the exact solution to a slightly perturbed problem.

Preservation of Symmetry. Unlike other methods (like Gaussian elimination), the Householder approach preserves the symmetry of the matrix throughout the reduction process. This is crucial, as many algorithms rely on symmetry to guarantee real eigenvalues. Also, exploiting symmetry halves memory and computation costs.

Foundation for Modern Eigenvalue Solvers. Nearly all high-performance software libraries (e.g., LAPACK, Eigen, MATLAB, SciPy) use Householder tridiagonalization as a first step in solving the symmetric eigenproblem. The algorithm provides a deterministic, predictable structure for the second stage (QR algorithm or divide-and-conquer method).

In summary, the Householder method has wide applications in science and engineering, since eigenvalue problems are central to quantum mechanics (Hamiltonians), structural analysis (vibration modes), principal component analysis (PCA), stability analysis in control systems, machine learning (kernel methods, covariance matrices), and so on.

EXAMPLE 11.7: Applying Sturm's Theorem

Use Sturm's theorem to find the characteristic polynomial of the following symmetric tridiagonal matrix and determine the roots (eigenvalues) of the resulting polynomial.

$$\mathbf{A} = \begin{bmatrix} 2 & 1 & 0 & 0 & 0 \\ 1 & 5 & 3 & 0 & 0 \\ 0 & 3 & 2 & 0 & 0 \\ 0 & 0 & 0 & 7 & 2 \\ 0 & 0 & 0 & 2 & 10 \end{bmatrix}$$

SOLUTION:

We note that $\mathbf{d} = \{2,\ 5,\ 2,\ 7,\ 19\}$ and $\mathbf{e} = \{1,\ 3,\ 0,\ 2\}$. The Sturm sequence of $\mathbf{A}$ can now be generated using Eq. (11.65)-(11.68). Starting with $p_0(\lambda) = 1$, we obtain

$$\begin{aligned} p_1(\lambda) =& d_1 - \lambda = 2 - \lambda \\ p_2(\lambda) =& (d_2 - \lambda)\, p_1(\lambda) - e_1^2\, p_0(\lambda) \\ =& (5 - \lambda)\, p_1(\lambda) - (1)^2\, p_0(\lambda) = \lambda^2 - 7\lambda + 9 \\ p_3(\lambda) =& (d_3 - \lambda)\, p_2(\lambda) - e_2^2\, p_1(\lambda) \\ =& (2 - \lambda)\, p_2(\lambda) - (3^2) p_1(\lambda) = -\lambda^3 + 9\lambda^2 - 14\lambda \\ p_4(\lambda) =& (d_4 - \lambda)\, p_3(\lambda) - e_3^2\, p_2(\lambda) \\ =& (7 - \lambda)\, p_3(\lambda) - (0^2)\, p_2(\lambda) = \lambda^4 - 16\lambda^3 + 77\lambda^2 - 98\lambda \\ p_5(\lambda) =& (d_5 - \lambda)\, p_4(\lambda) - e_4^2\, p_3(\lambda) \\ =& (10 - \lambda)\, p_4(\lambda) - (2^2)\, p_3(\lambda) = -\lambda^5 + 26\,\lambda^4 - 233\,\lambda^3 + 832\,\lambda^2 - 924\,\lambda \\ =& \lambda(2 - \lambda)(\lambda - 6)(\lambda - 7)(\lambda - 11) \end{aligned}$$

Clearly, it is seen that the characteristic polynomial is easily factored out and its eigenvalues are whole numbers, i.e., λ=0, 2, 6, 7, and 11. It is also noted that we find $\lambda_{\text{min}} > -1$ and $\lambda_{\text{max}} < 12$ using the Gerschgorin theorem *via* Eq. (11.69). In other words, all eigenvalues lie in the interval $(-1,12)$, which is consistent with our finding above.

Discussion: Sturm's method is a general-purpose method for finding or counting the number of real roots of any real polynomial within a given interval, i.e., it is not limited to finding eigenvalues *via* the characteristic polynomials. If a tridiagonal matrix is real symmetric, all eigenvalues are real, and Sturm's method (as shown in this example problem) is especially useful to count and isolate them.

The roots of polynomials of degree $n \leqslant 4$ can be obtained exactly. For $n = 3$ and for $n = 4$, the exact symbolic solutions can be obtained with Cardano's and Ferrari's methods, respectively. There is no general algebraic formula for polynomials of degree 5 or higher. We prefer numerical methods to find all real roots (or count them). One of these methods is the Sturm sequence, which provides a way to determine the number of distinct real roots of a polynomial in an interval (a, b). After each of the roots is isolated (a root in one interval), the roots are estimated within a tolerance level using a bracketing method such as the bisection method.

EXAMPLE 11.8: Characteristic Value Problem

Consider a square thin plate of thickness h, under uniaxial in-plane compressive force P_x. The left-right edges of the plate are simply supported ($w = w_{xx} = 0$) while the front-back edges are fixed ($w = w_y = 0$), as depicted in **Figure 11.1**:

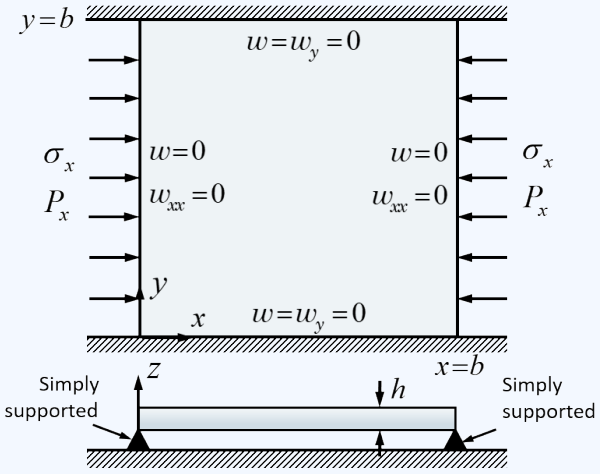

Figure 11.1

The plate buckling equation can be expressed as

$$\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} + \lambda \frac{\partial^2 w}{\partial x^2} = 0, \qquad 0 \leqslant x \leqslant b, \quad 0 \leqslant y \leqslant b$$

where $b = 4$ m. Notice that the buckling of a plate is in fact an eigenvalue problem, where $\lambda = P_x/D$ is the eigenvalue, $D = Eh^3/12(1-\nu^2)$ is the flexural rigidity, E is the Young modulus, ν is the Poisson's ratio, and w is out-of-plane deflection (or buckling mode shape). Apply the finite-difference method ($\Delta x = \Delta y = 1$) to determine the critical buckling load (i.e., the smallest eigenvalue).

SOLUTION:

The numerical solution of two-point BVP is like solving a problem along a line, resulting in fewer points and easier structure. On the other hand, the plate buckling equation is a partial differential equation, which needs to be solved on a plate surface with many more nodal points, more complexity at boundaries and in the solution method. The numerical solution of a PDE differs from that of a two-point BVP in several important ways, due to the nature of the equations themselves. However, the numerical solution steps are the same.

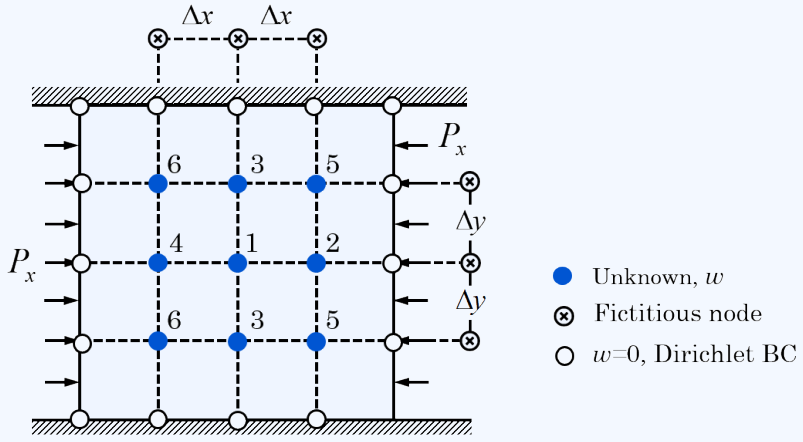

Figure 11.2

Step 1: Gridding. A two-dimensional uniform grid, shown in **Figure 11.2**, is generated using uniformly-spaced subintervals in both directions, $\Delta x = \Delta y = b/4 = 1$. The position of the nodal points can be found from $x_i = i\Delta x$ and $y_i = i\Delta y$ for $i = 0, 1, 2, 3, 4$. Also note that due to the Dirichlet BCs ($w = 0$) applied to the edges of the plate, the nodes corresponding to the edges are shown but not numbered.

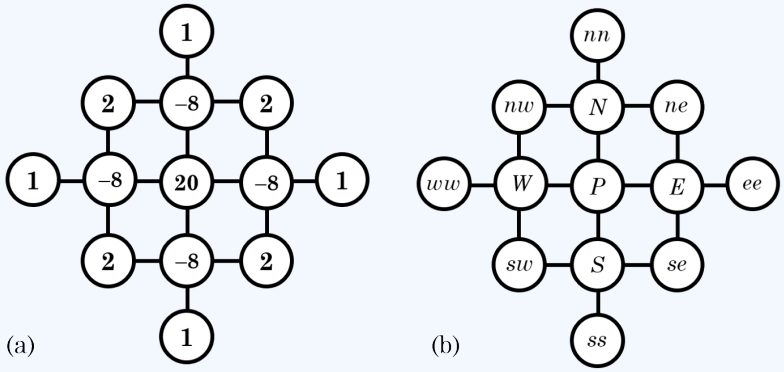

Figure 11.3: Discretization of biharmonic equation (a) computational molecule with the values of the coefficients in each node, (b) relative positions with respect to central node.

Step 2: Discretizing. The PDE is satisfied for all (x_i, y_j) in the given domain. So we may write

$$\Delta^2 w_{i,j} + \lambda\, \delta_x^2\, w_{i,j} = 0, \qquad \text{for } i = j = 1, 2, 3$$

Here, we will drop the double subscripts and adopt compass notation (where the position of neighboring grid points relative to the central node is described) to help make a complex stencil easier to understand and communicate. Also note that when assembling the entire finite difference system, we flatten the 2D grid into a 1D vector **w** with a single subscript; that is why the nodes in **Figure 11.2** have been coded from 1 to 6, with the symmetry in mind wrt $y = 2$. This further simplifies the system, leading to a standard matrix equation.

The central differencing of the biharmonic equation leads to a 13-point computational molecule:

$$\Delta^2 w_{i,j} \approx \frac{1}{h^4}\Big[\, 20\, w_P - 8\,(w_E + w_W + w_N + w_S) + 2\,(w_{ee} + w_{ww} + w_{ss} + w_{nn}) + w_{ne} + w_{se} + w_{nw} + w_{sw}\,\Big] + \mathcal{O}(h^2)$$

$$\delta_x^2 w_{i,j} \approx \frac{1}{h^2}(\, w_W - 2w_P + w_E) + \mathcal{O}(h^2)$$

for *P*=1, 2, ..., 6.

Step 3: Implementing BCs. Dirichlet boundary condition is applied to all 16 nodes (depicted as unnumbered circles) corresponding to the edges, i.e., $w = 0$ at 16 nodes.

At the top and bottom edges (for nodes 3, 5, and 6), the fictitious nodes will appear in the difference equations, which are eliminated by discretizing the Neumann BC with the central difference formula $w_y \approx (w_N - w_S)/2h = 0$, giving $w_N = w_S$ (either E or W is a fictitious node). At the left and right boundaries (for nodes 2, 4, 5 and 6), we also end up with fictitious nodes on the left or right sides, which are also eliminated by discretizing the BCs: $w_{xx} = 0$. Again, employing the central difference formula and noting that $w_P \approx 0$ on the boundary, we obtain $w_{xx} \approx (w_W - 2w_P + w_E)/h^2 = 0$, which results in $w_W = -w_E$ (W is a fictitious node) or $w_E = -w_W$ (E is a fictitious node).

Step 4: Constructing the Eigensystem. Making use of the discrete approximations in Step 2, the difference equation for node 1 becomes

$$20\,w_1 - 8\,w_2 - 16\,w_3 - 8\,w_4 + 4\,w_5 + 4\,w_6 = \lambda\,(-w_2 + 2\,w_1 - w_4)$$

For node 5 (with $P = 5$), we obtain

$$w_2 - 8\,w_2 - 8\,w_3 + 21\,w_5 + w_6 = \lambda\,(-w_3 + 2w_4)$$

where $w_E = w_N = 0$, $w_{ee} = -w_5$, and $w_{nn} = w_5$ due to the BCs.

When all the difference equations are assembled and the BCs are properly implemented, we obtain a generalized eigenvalue problem, **Aw**=λ**Bw**, where

$$\mathbf{A} = \begin{bmatrix} 20 & -8 & -16 & -8 & 4 & 4 \\ -8 & 19 & 4 & 1 & -16 & 0 \\ -8 & 2 & 22 & 2 & -8 & -8 \\ -8 & 1 & 4 & 19 & 0 & -16 \\ 2 & -8 & -8 & 0 & 21 & 1 \\ 2 & 0 & -8 & -8 & 1 & 21 \end{bmatrix},\ \mathbf{B} = \begin{bmatrix} 2 & -1 & 0 & -1 & 0 & 0 \\ -1 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & -1 & -1 \\ -1 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & -1 & 0 & 2 & 0 \\ 0 & 0 & -1 & 0 & 0 & 2 \end{bmatrix},\ \mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \end{bmatrix}$$

Step 5: Solving the eigensystem. We see that the finite difference equations with a 13-point stencil led to a generalized characteristic value problem: **Aw**=λ**Bw**. Note that **B** is a symmetric matrix while **A** is not. In this plate buckling problem, we do not need to find all eigenvalues. Only the smallest (first) eigenvalue, which corresponds to the critical buckling load, is essential. The smallest eigenvalue is the load at which the plate will buckle.

To convert the generalized EVP to a standard EVP, we first carry out $\mathbf{B} = \mathbf{L}\mathbf{L}^T$ decomposition with Cholesky's method since it is symmetrical. Employing the Cholesky decomposition algorithm (*see* Section 2.8.2), we find

$$\mathbf{L} = \begin{bmatrix} \sqrt{2} & 0 & 0 & 0 & 0 & 0 \\ -1/\sqrt{2} & \sqrt{3/2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{2} & 0 & 0 & 0 \\ -1/\sqrt{2} & -1/\sqrt{6} & 0 & 2/\sqrt{3} & 0 & 0 \\ 0 & 0 & -1/\sqrt{2} & 0 & \sqrt{3/2} & 0 \\ 0 & 0 & -1/\sqrt{2} & 0 & -1/\sqrt{6} & 2/\sqrt{3} \end{bmatrix}$$

Replacing $\mathbf{B}$ with $\mathbf{L}\mathbf{L}^T$ and defining $\mathbf{x} = \mathbf{L}^T\mathbf{w}$, we may also write $\mathbf{w} = \mathbf{L}^{-T}\mathbf{x}$. The generalized EVP then becomes $\mathbf{A}\mathbf{L}^{-T}\mathbf{x} = \mathbf{L}\mathbf{x}$. Next, left multiplying both sides by $\mathbf{L}^{-1}$, we convert this to a standard EVP:

$$\mathbf{C}\mathbf{x} = \lambda\,\mathbf{x}$$

where

$$\mathbf{C} = \mathbf{L}^{-1}\mathbf{A}\mathbf{L}^{-T} = \begin{bmatrix} 10 & 2/\sqrt{3} & -8 & 2\sqrt{2/3} & -4/\sqrt{3} & -4\sqrt{2/3} \\ 2/\sqrt{3} & 32/3 & -4/\sqrt{3} & 5\sqrt{2}/3 & -32/3 & -8\sqrt{2}/3 \\ -4 & -2/\sqrt{3} & 11 & -2\sqrt{2/3} & \sqrt{3} & \sqrt{6} \\ 2\sqrt{2/3} & 5\sqrt{2}/3 & -4\sqrt{2/3} & 37/3 & -8\sqrt{2}/3 & -40/3 \\ -2/\sqrt{3} & -16/3 & \sqrt{3} & -4\sqrt{2}/3 & 37/3 & 7\sqrt{2}/3 \\ -2\sqrt{2/3} & -4\sqrt{2}/3 & \sqrt{6} & -20/3 & 7\sqrt{2}/3 & 44/3 \end{bmatrix}$$

which is not symmetrical either.

Recall that we do not need to find all eigenvalues of $\mathbf{C}$, only the smallest in magnitude. To achieve this, we will employ the *inverse Power method* (*see* Section 11.2.3); that is,

$$\mathbf{C}\mathbf{x}^{(p+1)} = \mathbf{x}^{(p)} \quad \text{with} \quad \mu^{(p+1)} = \frac{(\mathbf{x}^{(p)}, \mathbf{x}^{(p+1)})}{(\mathbf{x}^{(p)}, \mathbf{x}^{(p)})}$$

Starting with **x**=[0 1 0 0 0 0], after 167 iterations with $\|\mathbf{x}\|_2 < 10^{-4}$, we find

$$\mu \to 0.2617051 = \frac{19+\sqrt{129}}{116}, \quad \text{or} \quad \lambda_{\text{min}} = \frac{1}{\mu} \to 3.821095 = \frac{19-\sqrt{129}}{2}$$

$$\mathbf{x} \to [0, \sqrt{\frac{193+\sqrt{129}}{435}}, 0, -\sqrt{\frac{193+\sqrt{129}}{870}}, \sqrt{\frac{97-\sqrt{129}}{435}}, -\sqrt{\frac{97-\sqrt{129}}{870}}]^T$$

and finally we obtain

$$\mathbf{w} = \mathbf{L}^{-T}\mathbf{x} = [0.005756,\ 0.597645,\ 0.003551,\ -0.589505,\ 0.386770,\ -0.381748]^T$$

Thus, the critical buckling load is obtained as $P_{\text{x,crit}} = 3.821092D$ since $\lambda_{\text{min}} = P_x/D = 3.821092$. The normal stress σ_x in the plate is spatially uniform, and we write it as $\sigma_x = -\sigma$. Then, the applied compressive stress is found as $\sigma = P_x/(hb)$. Note that higher eigenvalues correspond to higher-mode buckling patterns, which occur only if the structure is loaded beyond the first buckling load (rare and typically not of design concern).

Discussion: Coarse meshes tend to underestimate the smallest eigenvalue. As the mesh is refined (e.g., increasing the number of grid points), the smallest eigenvalue converges from below to the true value. Finer meshes provide much better accuracy but increase computational cost.

Buckling is a sudden, catastrophic failure mode. Thin plates, under compressive or shear loads, can fail at loads much lower than their material yield strength due to instability. In this regard, knowing the critical buckling load helps ensure that components remain stable under expected loads. Therefore, accurate prediction of the first mode shape is crucial in engineering and structural design, particularly in aerospace, civil, mechanical, and marine applications, as it determines the collapse mechanism. Critical buckling load prevents premature failure, guides material use, and supports compliant, optimized engineering design.

Table 11.1: Richardson extrapolation table for the computed minimum eigenvalue.

Grid	$\mathcal{O}(h^2)$	$\mathcal{O}(h^4)$	$\mathcal{O}(h^6)$	$\mathcal{O}(h^8)$
4×2	3.821092			
8×4	4.429254	4.631974		
16×8	4.654199	4.729181	4.735661	
32×16	4.720876	4.743102	4.744030	4.744163

The convergence rate depends on the order of the finite difference scheme. For second-order central difference, the eigenvalue error is typically $\mathcal{O}(h^2)$, where h is the grid spacing. For high-order accurate estimates (e.g., 4th, 6th, or 8th order), we may use Richardson extrapolation, provided that the grid interval is doubled (i.e., interval size h is reduced by half). The results of grid refinement and Richardson extrapolations up to the 8th order are presented **Table 11.1**. For the given plate dimensions, the critical buckling load is found to be $P_x \cong 4.744163D$.