Universiti Teknologi Malaysia

SME 3023 Applied Numerical Methods

Solution of Matrix Eigenvalue Problems

Abu Hasan Abdullah

Faculty of Mechanical Engineering

Sept 2012
Outline

1 Introduction
1. Introduction

2. Engineering Applications
Outline

1. Introduction

2. Engineering Applications

3. Definitions and Basic Facts
   - Eigenvalue & Eigenvector
   - Characteristic Equation
   - Standard & General Eigenvalue Problems
Outline

1. Introduction

2. Engineering Applications

3. Definitions and Basic Facts
   - Eigenvalue & Eigenvector
   - Characteristic Equation
   - Standard & General Eigenvalue Problems

4. Power Method
Outline

1. Introduction
2. Engineering Applications
3. Definitions and Basic Facts
   - Eigenvalue & Eigenvector
   - Characteristic Equation
   - Standard & General Eigenvalue Problems
4. Power Method
5. Faddeev-Leverrier Method
Outline

1 Introduction

2 Engineering Applications

3 Definitions and Basic Facts
   - Eigenvalue & Eigenvector
   - Characteristic Equation
   - Standard & General Eigenvalue Problems

4 Power Method

5 Faddeev-Leverrier Method

6 Matlab eig Function
Outline

1 Introduction

2 Engineering Applications

3 Definitions and Basic Facts
   - Eigenvalue & Eigenvector
   - Characteristic Equation
   - Standard & General Eigenvalue Problems

4 Power Method

5 Faddeev-Leverrier Method

6 Matlab eig Function

7 Bibliography
Eigenvalues play an important role in situations where the matrix is a transformation from one vector space onto itself.

Systems of linear ordinary differential equations are the primary examples.

The eigenvalues can correspond to critical values of stability parameters, or energy levels of atoms, or frequencies of vibration—see Figure 1.

Figure 1: Vibration of mass-spring system with three degrees of freedom.

The equations of motion for a system of masses and spring shown in Figure 1 are

\[
\begin{align*}
    m_1 \ddot{q}_1 + (k_1 + k_2 + k_4)q_1 - k_2q_2 - k_4q_3 &= 0 \\
    m_2 \ddot{q}_2 - k_2q_1 + (k_2 + k_3)q_2 - k_3q_3 &= 0 \\
    m_3 \ddot{q}_3 - k_4q_1 - k_3q_2 + (k_3 + k_4)q_3 &= 0
\end{align*}
\]

(1)
Figure 2: Forging hammer.
Problem Statement:
A forging hammer of mass $m_1$ is mounted on a concrete foundation block of mass $m_2$. The stiffness of the springs underneath the forging hammer and the foundation block are given by $k_2$ and $k_1$, respectively (see Figure 2). The system undergoes simple harmonic motion at one of its natural frequencies $\omega$, which are given by

$$\omega^2 = \frac{k_1 + k_2 - k_2}{m_1 + m_2}$$

where $\omega^2$ is the eigenvalue and $\bar{X}^T = \{x_1 \ x_2\}$ is the eigenvector or mode shape (displacement pattern) of the system. Determine the natural frequencies and mode shapes of the system for the following data:

Solution:
Work through the example.
Problem Statement:
The pin-ended column shown in Figure 3(a) is subjected to a compressive (axial) force, $P$. If the column is perturbed slightly as shown in Figure 3(b), as might occur from a slight vibration of the supports, it may not return to horizontal position even after removal of the disturbances; instead, the deflection might grow if the load is sufficiently large. Such load is called the buckling load.

Solution:
Work through the example.

Figure 3: Pin-ended column.
Definitions and Basic Facts

- A matrix is called **symmetric** if it is equal to its transpose,

\[ A = A^T \quad \text{or} \quad a_{ij} = a_{ji} \tag{2} \]

- A matrix is called **Hermitian** or **self-adjoint** if it equals the complex conjugate of its transpose (its **Hermitian conjugate**, denoted by “†”)

\[ A = A^\dagger \quad \text{or} \quad a_{ij} = a_{ji}^* \tag{3} \]

- A matrix is termed **orthogonal** if its transpose equals its inverse,

\[ A^T \cdot A = A \cdot A^T = 1 \tag{4} \]

and unitary if its Hermitian conjugate equals its inverse.

- A matrix is called **normal** if it **commutes** with its Hermitian conjugate,

\[ A \cdot A^\dagger = A^\dagger \cdot A \tag{5} \]
Definitions and Basic Facts

Eigenvalue & Eigenvector

- An eigenvalue and eigenvector of a square matrix \([A]\) are a scalar \(\lambda\), and a nonzero vector \(\vec{X}\) so that
  \[
  [A]\vec{X} = \lambda\vec{X}
  \]

- Standard eigenvalue problem is defined by the homogeneous equations
  \[
  (a_{11} - \lambda)x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n = 0 \\
  a_{21}x_1 + (a_{22} - \lambda)x_2 + a_{23}x_3 + \ldots + a_{2n}x_n = 0 \\
  \vdots \\
  a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \ldots + (a_{nn} - \lambda)x_n = 0
  \]

  and in matrix form
  \[
  [A]\vec{X} = \lambda\vec{X}
  \]
Definitions and Basic Facts
Eigenvalue & Eigenvector

In Eq. (7),

1. \([A]\) is a known square, \((n \times n)\), matrix

\[
[A] = \begin{bmatrix}
a_{11} & a_{12} & \cdots & a_{1n} \\
a_{21} & a_{22} & \cdots & a_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{n1} & a_{n2} & \cdots & a_{nn}
\end{bmatrix}
\] (8)

2. \(\vec{X}\) is the unknown \(n\)-component vector,

\[
\vec{X} = \begin{bmatrix}
x_1 \\
x_2 \\
\vdots \\
x_n
\end{bmatrix}
\] (9)

3. \(\lambda\) is an unknown scalar, a.k.a. eigenvalue.

Eq. (7) can be re-written as

\[
[[A] - \lambda[I]]\vec{X} = \vec{0}
\] (10)

where \([I]\) is the identity matrix of order \(n\).

Eq. (10) has nontrivial solution because it represents a system of \(n\) homogeneous equation in \(n + 1\) unknowns. Thus the determinant of the coefficient matrix of \(\vec{X}\) in Eq. (10) must be zero:

\[
|[A] - \lambda[I]| = 0
\] (11)
Definitions and Basic Facts

Characteristic Equation

If we expand Eq. (11) we get

\[
\begin{vmatrix}
(a_{11} - \lambda) & a_{12} & \cdots & a_{1n} \\
a_{21} & (a_{22} - \lambda) & \cdots & a_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{n1} & a_{n2} & \cdots & (a_{nn} - \lambda)
\end{vmatrix} = 0
\] (12)

which, upon further expansion, gives an \( n^{\text{th}} \) order polynomial in \( \lambda \),

\[
P_n(\lambda) = a_n \lambda^n + a_{n-1} \lambda^{n-1} + \ldots + a_2 \lambda^2 + a_1 \lambda + a_0 = 0
\] (13)

called the characteristic equation, where \( a_0, a_1, a_2 \ldots a_n \) are coefficients of polynomial. Assuming \( \lambda_1, \lambda_2, \ldots, \lambda_n \) are roots of the characteristic equation, then there will be \( n \) solutions to Eq. (7).
Corresponding to each distinct eigenvalue $\lambda_i$, a nontrivial solution of linear equations in Eq. (7) can be determined through

$$[A]\vec{X}^{(i)} = \lambda_i\vec{X}^{(i)}$$ \hspace{1cm} (14)

where

$$\vec{X}^{(i)} = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \\ \vdots \\ x_n^{(i)} \end{bmatrix}$$ \hspace{1cm} (15)

is called the eigenvector corresponding to eigenvalue $\lambda_i$. 
### Standard Eigenvalue Problem

- Eigenvalue problem as expressed by Eq. (7), repeated below,

\[
(a_{11} - \lambda)x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = 0 \\
a_{21}x_1 + (a_{22} - \lambda)x_2 + \ldots + a_{2n}x_n = 0 \\
\ldots \\
a_{n1}x_1 + a_{n2}x_2 + \ldots + (a_{nn} - \lambda)x_n = 0
\]

and in matrix form

\[
[A]\vec{X} = \lambda[B]\vec{X}
\]

is known as the standard eigenvalue problem.

### General Eigenvalue Problem

- Many physical problems, however, are expressed as the general eigenvalue problem given by

\[
[A]\vec{X} = \lambda[B]\vec{X} \quad (16)
\]

- General eigenvalue problem of Eq. (16) can be reduced to the standard eigenvalue problem of Eq. (7). We shall deal with this conversion later!
Since each eigenvector is associated with an eigenvalue, we often refer to an eigenvector $X$ and eigenvalue $\lambda$ that correspond to one another as an **eigenpair**.

Many of the “real world” applications are primarily interested in the **dominant eigenpair**.

The **dominant eigenvector** of a matrix is an eigenvector corresponding to the eigenvalue of largest magnitude (for real numbers, largest absolute value) of that matrix. The method that is used to find this eigenvector is called the **power method**.
Like the Jacobi and Gauss-Seidel methods, the **power method** for approximating eigenvalues is iterative.

1. First we assume that the matrix $A$ has a dominant eigenvalue with corresponding dominant eigenvectors.
2. Then we choose an initial approximation of one of the dominant eigenvectors of $A$. This initial approximation must be a nonzero vector in $\mathbb{R}^n$.
3. Finally we form the sequence given by

\[
x_1 = Ax_0 \\
x_2 = Ax_1 = A(Ax_0) = A^2x_0 \\
x_3 = Ax_2 = A(A^2x_0) = A^3x_0 \\
\vdots \\
x_k = Ax_{k-1} = A(A^{k-1}x_0) = A^kx_0
\]  

(17)
Power Method
Sequencing the Iteration–Example 1

We demonstrate those sequences on the matrix

\[ A = \begin{bmatrix} 3 & 6 \\ 1 & 4 \end{bmatrix} \]

and, for Step 1, arbitrarily choose

\[ x_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \]

\[ x_1 = Ax_0 = \begin{bmatrix} 3 & 6 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 9 \\ 5 \end{bmatrix} \rightarrow x'_1 = \begin{bmatrix} 1.8 \\ 1 \end{bmatrix} \]

\[ x_2 = Ax_1 = \begin{bmatrix} 3 & 6 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} 1.8 \\ 1 \end{bmatrix} = \begin{bmatrix} 11.4 \\ 5.8 \end{bmatrix} \rightarrow x'_2 = \begin{bmatrix} 1.965517241 \\ 1 \end{bmatrix} \]

\[ x_3 = Ax_2 = \begin{bmatrix} 3 & 6 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} 1.965517241 \\ 1 \end{bmatrix} = \begin{bmatrix} 11.89655172 \\ 5.96551724 \end{bmatrix} \rightarrow x'_3 = \begin{bmatrix} 1.994219653 \\ 1 \end{bmatrix} \]

\[ x_4 = Ax_3 = \begin{bmatrix} 3 & 6 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} 1.994219653 \\ 1 \end{bmatrix} = \begin{bmatrix} 11.98265896 \\ 5.99421965 \end{bmatrix} \rightarrow x'_4 = \begin{bmatrix} 1.99903568 \\ 1 \end{bmatrix} \]

It looks like an eigenvector is

\[ x = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \]
The corresponding eigenvalue is

\[
\frac{x^T Ax}{x^T x} = \frac{\begin{bmatrix} 2 & 1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 3 & 6 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix}}{\begin{bmatrix} 2 & 1 \\ 2 & 1 \end{bmatrix}} = \begin{bmatrix} 30 \\ 5 \end{bmatrix} \Rightarrow \lambda = 6
\]
The power method iteration in effect takes successively higher powers of matrix times initial starting vector

Algorithm: Power Method

\[
\begin{align*}
    x_0 &= \text{arbitrary nonzero vector} \\
    \text{for } k &= 1, 2, \ldots \\
    y_k &= A x_{k-1} \\
    x_k &= y_k / \|y_k\|_\infty \\
    \text{end}
\end{align*}
\]

Implementing the algorithm in Matlab:

Matlab Session

\[
\begin{align*}
    &>> A = \text{hilb}(5) \quad \% \text{matrix } A \\
    &>> x0 = \text{ones}(5,1) \quad \% \text{nonzero vector} \\
    &>> yk = A \times x0 \\
    &>> xk = yk / \text{norm}(yk)
\end{align*}
\]

- Compare the new value of \( x \) with the original.
- Repeat the last two lines (hint: use the scroll up button).
- Compare the newest value of \( x \) with the previous one and the original. Notice that there is less change between the second two.
- Repeat the last two commands over and over until the values stop changing.
Faddeev-Leverrier Method

- Let $A$ be an $n \times n$ matrix. The determination of eigenvalues and eigenvectors requires the solution of

$$AX = \lambda X$$

where $\lambda$ is the eigenvalue corresponding to the eigenvector $X$. The values $\lambda$ must satisfy the equation

$$\det(A - \lambda I) = 0$$

Hence $\lambda$ is a root of an $n$th degree polynomial $P(\lambda) = \det(A - \lambda I)$, which we write in the form

$$P(\lambda) = \lambda^n + c_1 \lambda^{n-1} + c_2 \lambda^{n-2} + \ldots + c_{n-2} \lambda^2 + c_{n-1} \lambda + c_n$$  \hspace{1cm} (18)$$

The Faddeev-Leverrier algorithm is an efficient method for finding the coefficients $c_k$ of the polynomial $P(\lambda)$. As an additional benefit, the inverse matrix $A^{-1}$ is obtained at no extra computational expense.
The trace of the matrix $A$, written $\text{Tr}[A]$, is

$$\text{Tr}[A] = a_{1,1} + a_{2,2} + \ldots + a_{n,n} \quad (19)$$

The algorithm generates a sequence of matrices $B_k = A$ and uses their traces to compute the coefficients of $P(\lambda)$,

- $B_1 = A$ and $p_1 = \text{Tr}[B_1]$
- $B_2 = A(B_1 - p_1I)$ and $p_2 = \frac{1}{2} \text{Tr}[B_2]$
- $\vdots$ \hspace{2cm} $\vdots$
- $B_k = A(B_{k-1} - p_{k-1}I)$ and $p_k = \frac{1}{k} \text{Tr}[B_k]$
- $\vdots$ \hspace{2cm} $\vdots$
- $B_n = A(B_{n-1} - p_{n-1}I)$ and $p_n = \frac{1}{n} \text{Tr}[B_n] \quad (20)$
Then the characteristic polynomial, Eq. 18, can be re-written as

\[ P(\lambda) = \lambda^n + p_1 \lambda^{n-1} + p_2 \lambda^{n-2} + \ldots + p_{n-2} \lambda^2 + p_{n-1} \lambda + p_n \]  

(21)

and the inverse matrix is given by

\[ A^{-1} = \frac{1}{p_n} (B_{n-1} - p_{n-1} I) \]  

(22)
Faddeev-Leverrier Method

Example 1

**Problem Statement:**
Use Faddeev’s method to find the characteristic polynomial and inverse of the matrix

\[
\begin{bmatrix}
2 & -1 & 1 \\
-1 & 2 & 1 \\
1 & -1 & 2
\end{bmatrix}
\]

**Solution:**

\[
A = \begin{bmatrix}
2 & -1 & 1 \\
-1 & 2 & 1 \\
1 & -1 & 2
\end{bmatrix}
\]

\[
B_1 = \begin{bmatrix}
2 & -1 & 1 \\
-1 & 2 & 1 \\
1 & -1 & 2
\end{bmatrix}
\]

\[
p_1 = \text{Tr}[B_1] = 6
\]
Solution (continued):

\[ B_2 = A(B_1 - p_1 I) \]

\[
B_2 = \begin{bmatrix} 2 & -1 & 1 \\ -1 & 2 & 1 \\ 1 & -1 & 2 \end{bmatrix} \begin{bmatrix} 2 & -1 & 1 \\ -1 & 2 & 1 \\ 1 & -1 & 2 \end{bmatrix} - 6 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -6 & 1 & -3 \\ 3 & -8 & -3 \\ -1 & 1 & -8 \end{bmatrix}
\]

\[ p_2 = \frac{1}{2} \text{Tr}[B_2] = \frac{1}{2}(-22) = -11 \]
Solution (continued):

\[
B_3 = A(B_2 - p_2I)
\]

\[
B_3 = \begin{bmatrix}
2 & -1 & 1 \\
-1 & 2 & 1 \\
1 & -1 & 2
\end{bmatrix}
\begin{bmatrix}
-6 & 1 & -3 \\
3 & -8 & -3 \\
-1 & 1 & -8
\end{bmatrix}
- (-11)
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

\[
= \begin{bmatrix}
6 & 0 & 0 \\
0 & 6 & 0 \\
0 & 0 & 6
\end{bmatrix}
\]

\[
p_3 = \frac{1}{3} \text{Tr}[B_3] = \frac{1}{3}(18) = 6
\]

The characteristic polynomial is thus

\[
P(\lambda) = \lambda^3 - \sum_{i=1}^{3} p_i \lambda^{n-i} = -6 + 11\lambda - 6\lambda^2 + \lambda^3
\]
Faddeev-Leverrier Method

Example 1

Solution (continued):
The inverse matrix is

\[
A^{-1} = \frac{1}{p_3} (b_{3-1} - p_{3-1} I)
\]

\[
= \frac{1}{6} \left( \begin{bmatrix} -6 & 1 & -3 \\ 3 & -8 & -3 \\ -1 & 1 & -8 \end{bmatrix} - (-11) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right)
\]

\[
= \frac{1}{6} \left( \begin{bmatrix} -6 & 1 & -3 \\ 3 & -8 & -3 \\ -1 & 1 & -8 \end{bmatrix} - \begin{bmatrix} -11 & 0 & 0 \\ 0 & -11 & 0 \\ 0 & 0 & -11 \end{bmatrix} \right)
\]

\[
= \frac{1}{6} \left( \begin{bmatrix} 5 & 1 & -3 \\ 3 & 3 & -3 \\ -1 & 1 & 3 \end{bmatrix} \right)
\]

\[
= \frac{1}{6} \begin{bmatrix} 5 & 1 & -3 \\ 3 & 3 & -3 \\ -1 & 1 & 3 \end{bmatrix} = \begin{bmatrix} \frac{5}{6} & \frac{1}{6} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{6} & \frac{1}{6} & \frac{1}{2} \end{bmatrix}
\]
Matlab `eig` Function

Example 1

**Problem Statement:**

\[
\begin{bmatrix}
-149 & -50 & -154 \\
537 & 180 & 546 \\
-27 & -9 & -25
\end{bmatrix}
\begin{Bmatrix}
x_1 \\
x_2 \\
x_3
\end{Bmatrix}
= \begin{bmatrix}
\lambda_1 & 0 & 0 \\
0 & \lambda_2 & 0 \\
0 & 0 & \lambda_3
\end{bmatrix}
\begin{Bmatrix}
x_1 \\
x_2 \\
x_3
\end{Bmatrix}
\]

**Solution:**

Matlab Session

```matlab
>> A = [-149 -50 -154; 537 180 546; -27 -9 -25]
>> [X,lambda] = eig(A)
>> lambda1 = lambda(1,:)
>> lambda2 = lambda(2,:)
>> lambda3 = lambda(3,:)
>> X1 = X(:,1)
>> X2 = X(:,2)
>> X3 = X(:,3)
>> LHS1 = A*X1
>> RHS1 = lambda1*X1
>> % Check if LHS=RHS
```
Bibliography


