1 Vectors and Matrices

1.1 Definition of Matrix

Without the use of matrices, the solution of a large set of linear simultaneous equations would be impractical and prone to various errors. The effectiveness of using matrices in practical calculations can be easily demonstrated with the following set:

$$\begin{aligned} 5x_1 - 4x_2 + x_3 &= 0 \\ -4x_1 + 6x_2 - 4x_3 + x_4 &= 1 \\ x_1 - 4x_2 + 6x_3 - 4x_4 &= 0 \\ x_2 - 4x_3 - 5x_4 &= 0 \end{aligned} \qquad \text{Eq. 1.1}$$

where the unknowns are x_1, x_2, x_3, and x_4. Using matrix notation, this set of equations is written as:

$$\begin{bmatrix} 5 & -4 & 1 & 0 \\ -4 & 6 & -4 & 1 \\ 1 & -4 & 6 & -4 \\ 0 & 1 & -4 & 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \qquad \text{Eq. 1.2}$$

The grouping of the coefficients in one array, the unknowns in another and the right-hand-side in another, allows us to write the set of simultaneous equations as:

$$\mathbf{Ax} = \mathbf{b} \qquad \text{Eq. 1.3}$$

where **A** is the matrix of the coefficients in the set of linear equations, **x** is the matrix of unknowns, and **b** is the matrix of known quantities (or constants):

$$\mathbf{A} = \begin{bmatrix} 5 & -4 & 1 & 0 \\ -4 & 6 & -4 & 1 \\ 1 & -4 & 6 & -4 \\ 0 & 1 & -4 & 5 \end{bmatrix}; \qquad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}; \qquad \mathbf{b} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \qquad \text{Eq. 1.4}$$

Definition[1]: *A matrix is an array of ordered numbers. A general matrix consists of mn numbers arranged in m rows and n columns, giving the following array:*

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & . & . & . & a_{1n} \\ a_{21} & a_{22} & . & . & . & a_{2n} \\ . & . & & & & . \\ . & . & & & & . \\ . & . & & & & . \\ a_{m1} & a_{m2} & . & . & . & a_{mn} \end{bmatrix} \qquad \text{Eq. 1.5}$$

The order of this matrix is then $m \times n$ (m by n). Matrices with one row (m = 1) or one column (n = 1) are called vectors. Conventionally matrices are represented by boldface letters, lower case for vectors and uppercase for non-vectors.

[1] p.18, “Finite Element Procedures”, Klaus-Jürgen Bathe,1996

1.2 Matrix Properties

It is possible to classify matrices that obey a certain form. The most common matrices that we will encounter in this course are square and symmetric matrices.

Definition[2]: *The transpose of the* $m \times n$ *matrix* **A**, *written as* $\mathbf{A}^T$, *is obtained by interchanging the rows and columns in* **A**. *If* $\mathbf{A} = \mathbf{A}^T$, *it follows that the number of rows and columns in* **A** *are equal and that* $a_{ij} = a_{ji}$. *Because* $m = n$, *we say that* **A** *is a* square matrix *of order n, and because* $a_{ij} = a_{ji}$, *we say that* **A** *is a* symmetric matrix. *Note that symmetry implies that* **A** *is square, but not vice versa; i.e., a square matrix need not be symmetric.*

The *identity* (or *unit*) matrix $\mathbf{I}_n$, which is a square matrix of order *n*, has all its diagonal entries equal to unity and all others equal to zero:

$$\mathbf{I}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \text{Eq. 1.6}$$

An identity matrix is also a *diagonal* matrix because only its diagonal terms are non-zero.

Due to the nature of Finite Element Models (discussed later), symmetric banded matrices are encountered more than often. All the elements of a banded matrix are zero beyond its bandwidth, which is very convenient in storing very large matrices.

Consider the symmetric matrix in Figure 1-1; it is a square matrix of order 6. In addition to that, it is a banded matrix where all elements beyond its bandwidth ($2m_A$+1=5) are zero. Because **A** is symmetric, this condition can be stated as:

$$a_{ij} = 0 \quad \text{for} \quad j > i + m_A \qquad \text{Eq. 1.7}$$

As the matrix in Figure 1-1a is both ***symmetric*** and ***banded***, it is possible to use a more efficient storage scheme rather than using a two-dimensional array of 36 elements. This scheme is called the ***Skyline*** method. The skyline is a virtual line above which all elements of the matrix are zero. This does not preclude the presence of zero elements below the skyline as it is the case for element **A**(2,3). Note the difference between the skyline and the bandwidth; the bandwidth of a given matrix is a fixed property whereas its skyline can vary (discussed later).

The idea behind the Skyline method is very simple: ***all columns of non-zero elements above the diagonal are stacked in a one-dimensional array while keeping track of the index of the diagonal elements.***

The advantage of using the Skyline method is to spare large amounts of in core memory while handling larger matrices. In core memories (RAM) are about two orders of magnitude faster to access than virtual memory stored on disks. Therefore the advantage of using more efficient storage techniques is not invalidated by the availability of cheaper in core memory, as however plentiful they might be, a larger model can always be solved with a more efficient storage.

The Skyline method is easily described in Figure 1-1a-b, where **AA** is the one-dimensional array containing the non-zero elements above the diagonal and **AMAX** is the on-dimensional array that keeps track of the index of each diagonal term. By knowing the index in **AA** of the

[2] p.19, ibid

diagonal terms, it is possible to access the elements of **A** using a simple algorithm. Consider the element **A**(i,j) where i = 4 and j = 5. The value of j gives us the column of the element, so we look in **AMAX** the index of the 5th diagonal in **AA**, which is 8. The value of i tells us how far away **A**(i,j) is from the diagonal (j-i = 1). Therefore **A**(i,j) = **AA**(**AMAX**(j)+(j-i)) or **A**(4,5)=**AA**(**AMAX**(5)+(5-4))= 9. Note that this formulation is only valid for $j \geq i$. Since the matrix is symmetric, i and j can always be transposed to conform with the formulation.

$m_A+1=3$ Skyline

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 3 & 0 & 3 & 5 & 2 & 0 \\ 0 & 0 & 5 & 4 & 9 & 1 \\ 0 & 0 & 2 & 9 & 8 & 4 \\ 0 & 0 & 0 & 1 & 4 & 7 \end{bmatrix}$$

(a)

AA(1)=1, AA(2)=3
AA(3)=3, AA(4)=0
AA(5)=3, AA(6)=4
AA(7)=5, AA(8)=8
AA(9)=9, AA(10)=2
AA(11)=7, AA(12)=4
AA(13)=1

(b)

AMAX(1)=1
AMAX(2)=2
AMAX(3)=3
AMAX(4)=6
AMAX(5)=8
AMAX(6)=11

(c)

Figure 1-1 Storage of matrix A in a one-dimensional array

1.3 Equality, addition, subtraction and multiplication by a scalar

As matrices are defined as ordered arrays of numbers, it is also necessary to define rules governing operations such as addition, substraction, multiplication, division and equality.

Definition*[3]*: *The matrices* **A** *and* **B** *are equal if and only if*

1. **A** *and* **B** *have the same number of rows and columns.*
2. *All corresponding elements are equal; i.e.* a_{ij} = b_{ij} *for all i and j.*

Definition*[4]*: *Two matrices* **A** *and* **B** *can be* added *if and only if they have the same number of rows and columns. The addition of the matrices is performed by adding all corresponding elements; i.e., if* a_{ij} *and* b_{ij} *denote general elements of* **A** *and* **B**, *respectively, then* $c_{ij} = a_{ij}+b_{ij}$ *denotes a general element of* **C**, *where* **C** = **A** + **B**. *it follows that* **C** *has the same number of rows and columns as* **A** *and* **B**.

[3] p.21, ibid

[4] p.21, ibid

Example 1.1:

Calculate **C = A + B**, where

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}; \qquad \mathbf{B} = \begin{bmatrix} 7 & 8 & 9 \\ 1 & 2 & 3 \end{bmatrix}$$

$$\Rightarrow \qquad \mathbf{C} = \mathbf{A} + \mathbf{B} = \begin{bmatrix} 8 & 10 & 12 \\ 5 & 7 & 9 \end{bmatrix}$$

Example 1.2:

Calculate **C =** k**A**, where

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}; \qquad k = 2$$

$$\Rightarrow \qquad \mathbf{C} = k\mathbf{A} = \begin{bmatrix} 2 & 4 & 6 \\ 8 & 10 & 12 \end{bmatrix}$$

1.4 Multiplication of Matrices

To calculate the product of two matrices the following definition is used:

Definition[5]: *Two matrices* **A** *and* **B** *can be* multiplied *to obtain* **C** = **AB** *if and only if the number of columns in* **A** *is equal to the number rows in* **B**. *Assume that* **A** *is of order* $p \times m$ *and* **B** *is of order* $m \times q$. *Then for each element in* **C** *we have*

$$c_{ij} = \sum_{r=1}^{m} a_{ir} b_{rj} \qquad \text{Eq. 1.8}$$

where **C** *is of order* $p \times q$*; i.e., the indices i and j in Eq. 1.8 vary from* 1 *to p and* 1 *to q, respectively.*

Example 1.3:

Calculate the matrix product **C = AB**, where

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}; \qquad \mathbf{B} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

The calculations would be:

$$c_{11} = (1)(1) + (2)(3) + (3)(5) = 22$$
$$c_{21} = (4)(1) + (5)(3) + (6)(5) = 49$$
$$c_{31} = (7)(1) + (8)(3) + (9)(5) = 76$$

$$c_{12} = (1)(2) + (2)(4) + (3)(6) = 28$$
$$c_{22} = (4)(2) + (5)(4) + (6)(6) = 64$$
$$c_{32} = (7)(2) + (8)(4) + (9)(6) = 100$$

$$\Rightarrow \quad \mathbf{C} = \begin{bmatrix} 22 & 28 \\ 49 & 64 \\ 76 & 100 \end{bmatrix}$$

[5] p.22, ibid

The number of multiplications required in this matrix multiplication is $p \times q \times m$, where p = 3, q = 3 and m = 2. However, when dealing with banded matrices, the number of operations can be reduced by skipping products where at least one of the multipliers is zero.

Being great in handling large numbers of linear equations, rules governing matrix operations may not be always intuitive, such as the non-commutative nature of the multiplication:

$$\mathbf{A} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}; \qquad \mathbf{B} = \begin{bmatrix} 3 & 4 \end{bmatrix} \qquad \text{Eq. 1.9}$$

$$\mathbf{AB} = \begin{bmatrix} 3 & 4 \\ 6 & 8 \end{bmatrix}; \qquad \mathbf{BA} = \begin{bmatrix} 11 \end{bmatrix} \qquad \text{Eq. 1.10}$$

Depending on the orders of **A** and **B**, their products **AB** and **BA** may also have different orders; such is the case in Eq. 1.10. A way of distinguishing the order of a matrix multiplication is to state: **A** *premultiplies* **B**, or **B** *postmultiplies* **A**, to define the product **AB**.

However, matrix multiplication is both distributive and associative.

The distributive law states:

$$\mathbf{E} = (\mathbf{A} + \mathbf{B})\mathbf{C} = \mathbf{AC} + \mathbf{BC} \qquad \text{Eq. 1.11}$$

The associative law states:

$$\mathbf{G} = (\mathbf{AB})\mathbf{C} = \mathbf{A}(\mathbf{BC}) = \mathbf{ABC} \qquad \text{Eq. 1.12}$$

<u>Some known properties:</u>

$$\mathbf{v}^T \mathbf{A} \mathbf{v} = \mathbf{v}^T \mathbf{A}^T \mathbf{v} \qquad \text{Eq. 1.13}$$

even if $\mathbf{A} \neq \mathbf{A}^T$, and where **v** is strictly a vector.

$$\mathbf{B}^T \mathbf{A} \mathbf{B} = \mathbf{B}^T \mathbf{A}^T \mathbf{B} \qquad \text{Eq. 1.14}$$

if **A** is symmetric ➔ $\mathbf{A} = \mathbf{A}^T$, where **B** can be any matrix

$$\mathbf{AI} = \mathbf{A} \qquad \text{Eq. 1.15}$$

where **I** is the identity matrix.

$$\mathbf{AA}^{-1} = \mathbf{I} \qquad \text{Eq. 1.16}$$

where $\mathbf{A}^{-1}$ is the inverse matrix of **A**.

$$(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1} \qquad \text{Eq. 1.17}$$

It is sometimes useful to rewrite matrices in terms of identifiable ones such as rewriting a symmetric matrix as the sum of a lower triangular matrix and a diagonal matrix. This simple expansion can help reduce the amount of operations required to calculate a complex looking expression.

Example 1.4:

Evaluate the product $\mathbf{v}^T\mathbf{A}\mathbf{v}$, where

$$\mathbf{A} = \begin{bmatrix} 3 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 6 \end{bmatrix}; \qquad \mathbf{v} = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}$$

The straight on procedure would be to calculate $\mathbf{x} = \mathbf{A}\mathbf{v}$, and then $\mathbf{v}^T\mathbf{x}$. However the process can be simplified by using the following matrix manipulations:

$$\mathbf{A} = \mathbf{U} + \mathbf{D} + \mathbf{U}^T$$

where **U** is a lower triangular matrix and **D** is a diagonal matrix,

$$\mathbf{U} = \begin{bmatrix} 0 & 0 & 0 \\ 2 & 0 & 0 \\ 1 & 2 & 0 \end{bmatrix}; \qquad \mathbf{D} = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 6 \end{bmatrix}$$

➔ $$\mathbf{v}^T\mathbf{A}\mathbf{v} = \mathbf{v}^T(\mathbf{U} + \mathbf{D} + \mathbf{U}^T)\mathbf{v}$$

➔ $$\mathbf{v}^T\mathbf{A}\mathbf{v} = \mathbf{v}^T\mathbf{U}\mathbf{v} + \mathbf{v}^T\mathbf{D}\mathbf{v} + \mathbf{v}\mathbf{U}^T\mathbf{v}$$

recalling Eq. 1.13 $$\mathbf{v}^T\mathbf{U}^T\mathbf{v} = \mathbf{v}^T\mathbf{U}\mathbf{v}$$

➔ $$\mathbf{v}^T\mathbf{A}\mathbf{v} = 2\mathbf{v}^T\mathbf{U}\mathbf{v} + \mathbf{v}^T\mathbf{D}\mathbf{v}$$

knowing that the majority of the elements of **U** and **D** are zero, nil operations are avoided, and we can calculate $\mathbf{x} = \mathbf{U}\mathbf{v}$ as:

$$\begin{aligned} x_1 &= 0 \\ x_2 &= (2)(1) = 2 \\ x_3 &= (1)(1) + (2)(2) = 5 \end{aligned} \quad ➔ \quad \mathbf{x} = \begin{bmatrix} 0 \\ 2 \\ 5 \end{bmatrix}$$

next, we calculate

$$\mathbf{v}^T\mathbf{U}\mathbf{v} = \mathbf{v}^T\mathbf{x} = (2)(2) + (-1)(5) = -1$$

also $$\mathbf{v}^T\mathbf{D}\mathbf{v} = (1)(1)(3) + (2)(2)(4) + (-1)(-1)(6) = 25$$

➔ $$\mathbf{v}^T\mathbf{A}\mathbf{v} = 2\mathbf{v}^T\mathbf{U}\mathbf{v} + \mathbf{v}^T\mathbf{D}\mathbf{v} = (2)(-1) + 25 = 23$$