

APPENDIX A

VECTOR NOTATION

*Vector Notation is Based on
The Physical Laws of Statics*

A.1 INTRODUCTION

{ XE "Vector Notation" }To define member properties, skew boundary conditions and other information required to specify the input data for three-dimensional structures, the computer program user must have a working knowledge of vector notation. Because forces and moments are vectors in three-dimensional space, this appendix reviews, from a physical standpoint, vector notation and vector operations that are required to use a structural analysis program intelligently. Any force acting in three-dimensional space has a magnitude and direction or line of action, as shown in Figure A.1.

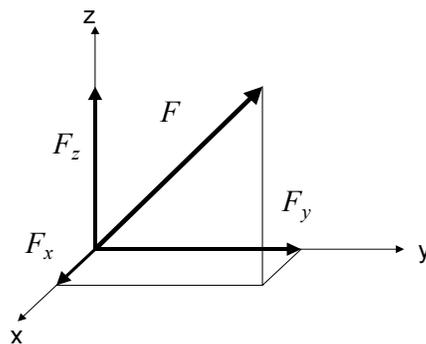


Figure A.1 Typical Force Vector

The point of application of the force on the structure is on this line of action. Also, a force can be expressed in terms of its components in the global x, y and z axes. In vector notation, the force is written in terms of its components as:

$$F = F_x \hat{x} + F_y \hat{y} + F_z \hat{z} \quad (\text{A.1})$$

where \hat{x} , \hat{y} and \hat{z} are by definition the unit vectors along the x, y, and z axes respectively. Note that a *vector equation always has three components*.

It is apparent that the absolute value of the magnitude of the force vector is given by:

$$|F| = \sqrt{F_x^2 + F_y^2 + F_z^2} \quad (\text{A.2})$$

We can now define the following dimensionless ratios:

$$V_{xf} = \frac{F_x}{|F|}, \quad V_{yf} = \frac{F_y}{|F|}, \quad \text{and} \quad V_{zf} = \frac{F_z}{|F|} \quad (\text{A.3})$$

In vector notation, these ratios are termed the direction cosines of the vector. Hence, the unit vector in the direction of the vector is:

$$\hat{f} = V_{xf} \hat{x} + V_{yf} \hat{y} + V_{zf} \hat{z} \quad (\text{A.4})$$

Therefore, the direction cosines are not independent because:

$$1 = V_{xf}^2 + V_{yf}^2 + V_{zf}^2 \quad (\text{A.5})$$

A.2 VECTOR CROSS PRODUCT

{ XE "Vector Cross Product" }The vector cross product can be defined using traditional, abstract mathematical notation. Or, the physical laws of statics can be applied to develop and explain the use of the cross product operation in defining the geometry of three-dimensional structural systems. The definition of a positive moment vector is defined by the right-hand rule, illustrated in Figure A.2.

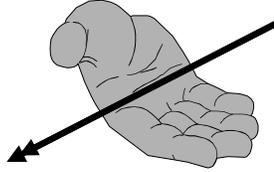


Figure A.2 Definition of Positive Moment (rotation) using the Right Hand Rule

Figure A.3 shows two vectors, a distance vector \mathbf{d} and a force vector \mathbf{F} . Point 2 is on the line of action of the force vector \mathbf{F} .

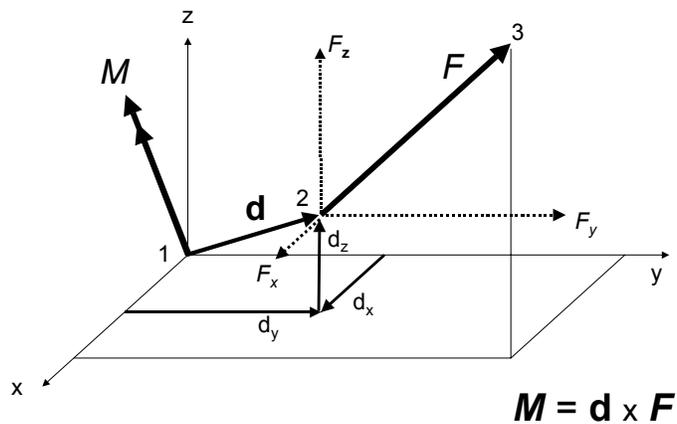


Figure A.3 Cross Product of Two Vectors

To calculate the moment acting at point 1, one can use the three components of the force vectors and the three components of the distance vector to calculate the three components of the resulting moment. Or:

$$M_x = d_y F_z - d_z F_y, \quad M_y = d_z F_x - d_x F_z \quad \text{and} \quad M_z = d_x F_y - d_y F_x \quad (\text{A.6})$$

The resulting moment at point 1 is written in vector notation as:

$$\mathbf{M} = M_x \hat{x} + M_y \hat{y} + M_z \hat{z} \quad (\text{A.7})$$

Therefore, this physical procedure defined as the cross, or vector, product of two vectors is defined as:

$$M = d \times F \quad (\text{A.8})$$

Because all of these calculations are performed within computer programs, it is not necessary to remember this cross product equation. The important physical fact to remember is that the resultant rotational vector is normal to the plane defined by points 1, 2 and 3.

A.3 VECTORS TO DEFINE A LOCAL REFERENCE SYSTEM

{ XE "Vectors Used to Define Local Reference System" } A local 1, 2, 3 reference system can be defined by the specification of three points in space, as shown in Figure A.4.

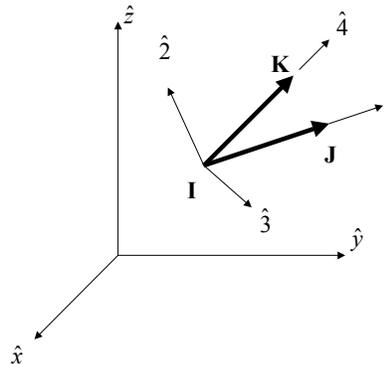


Figure A.4 Definition of Local Reference System from Points I, J and K

Unit vectors \hat{I} and \hat{J} can be defined from the vectors I to J and I to K respectively. Now, if we form the cross product vectors \hat{I} with \hat{J} , we can define a vector \hat{M} normal to the plane I-J-K. The unit vector \hat{L} is now defined by the cross product of the vectors \hat{M} with \hat{I} . The resulting local 1,2,3 right-hand reference system is related to the global x,y,z system by the following matrix equations of direction cosines:

$$\begin{bmatrix} \hat{1} \\ \hat{2} \\ \hat{3} \end{bmatrix} = \begin{bmatrix} V_{1x} & V_{1y} & V_{1z} \\ V_{2x} & V_{2y} & V_{2z} \\ V_{3x} & V_{3y} & V_{3z} \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} \quad (\text{A.9})$$

The 3 by 3 V matrix can now be used to transform displacements, rotations, forces and moments from one reference system to another reference system. For example, the displacement transformation equations are:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \mathbf{V} \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = \mathbf{V}^T \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (\text{A.10a and A.10b})$$

This allows element stiffness and load matrices to be formed in a local element reference system and then transformed to a global reference system to form the global equilibrium equations.

A.4 FORTRAN SUBROUTINES FOR VECTOR OPERATIONS

Within a structural analysis program, only two vector operations are required. To define a vector, the coordinates of the starting point “I” and ending point “J” must be given. The FORTRAN subroutine given in Table A.1 illustrates how the three direction cosines are calculated and how the length of the vector is calculated. The results are stored in the “V” array.

Table A.1 FORTRAN Subroutine to Define Vector

```

SUBROUTINE VECTOR (V,XI,YI,ZI,XJ,YJ,ZJ)
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION V(4)
C---- GIVEN TWO POINTS DEFINE VECTOR IN I-J DIRECTION -
X = XJ - XI           ! X PROJECTION
Y = YJ - YI           ! Y PROJECTION
Z = ZJ - ZI           ! Z PROJECTION
V(4) = DSQRT( X*X + Y*Y + Z*Z ) ! VECTOR LENGTH
C---- ERROR CHECK -----
IF (V(4).LE.0.0D0) THEN
WRITE (*,*) '*ERROR* ZERO LENGTH MEMBER OR VECTOR'
PAUSE 'CORRECT ERROR AND RERUN PROGRAM'
STOP ' '
ENDIF
C---- COMPUTER DIRECTION COSINES -----
V(3) = Z/V(4)
V(2) = Y/V(4)
V(1) = X/V(4)
C
RETURN
END

```

The subroutine given in Table A.2 produces the cross product vector “C,” given vectors “A” and “B.”

Table A.2 FORTRAN Subroutine to Perform Vector Cross Product

```

SUBROUTINE CROSS (A,B,C)
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION A(4),B(4),C(4)
C---- CROSS PRODUCT OF VECTORS "A" x "B" = VECTOR "C"-
X = A(2)*B(3) - A(3)*B(2) ! X COMPONENT
Y = A(3)*B(1) - A(1)*B(3) ! Y COMPONENT
Z = A(1)*B(2) - A(2)*B(1) ! Z COMPONENT
C(4) = DSQRT( X*X + Y*Y + Z*Z ) ! VECTOR LENGTH
C---- CHECK FOR ERROR -----
IF (C(4).LE.0.0D0) THEN
WRITE (*,*) '*ERROR* VECTORS ARE IN SAME DIRECTION'
PAUSE 'CORRECT ERROR AND RERUN PROGRAM'
STOP ' '
ENDIF
C---- COMPUTE DIRECTION COSINES -----
C(3) = Z/C(4)
C(2) = Y/C(4)
C(1) = X/C(4)
C
RETURN
END

```