MATRIX STRUCTURAL ANALYSIS

MSA

Beam , Truss , Frame
2 & 3 Dimensions

HOSSEIN RAHAMI

Engineering Science Department

College of Engineering

University of Tehran

Tehran – Iran

March 2010

◇ CONTENTS

◇ DESCRIPTION

Program Data.m

```
function   D=Data
```

Note: x,y,z are indicators of local coordinate system and X,Y,Z are of global ones.

Variable	Size	Description	
m	1*1	total number of members	
n	1*1	total number of nodes	
Coord	n*3	coordinate of nodes	
		columns 1:3	X,Y,Z of coordinates
Con	m*4	connectivity & release	
		column 1	beginning node
		column 2	end node
		column 3	indicates if a member is released(=0) or not(=1) at its beginning
		column 4	indicates if a member is released(=0) or not(=1) at its end
		Note: for trusses Con(:,3:4)=0	
Re	n*6	degrees of freedom for each node (free=0 & fixed=1)	
		columns 1:3	flag indicating displacement in global X,Y,Z directions
		columns 4:6	flag indicating rotation about global X,Y,Z axes
		Note: for 2-D structures Re(:,3:5)=1	
Load	n*6	concentrated loads on nodes	
		columns 1:3	forces in global X,Y,Z direction
		columns 4:6	moments about global X,Y,Z axes
w	m*3	uniform loads in local coordinate system	
		columns 1:3	x,y,z component of w
E	1*m	material elastic modules	
G	1*m	shear elastic modules	
A	1*m	cross sectional area	
Iz	1*m	moment of inertia about its local z-z axis	
Iy	1*m	moment of inertia about its local y-y axis	
J	1*m	torsional constant	
St	n*6	settlement of supports & displacements of free nodes	
		columns 1:3	flag indicating displacement in global X,Y,Z directions
		columns 4:6	flag indicating rotation about global X,Y,Z axes
be	1*m	web rotation angle	
		Note: the angle assumes a counterclockwise convention about the local x-axis (in radians)	

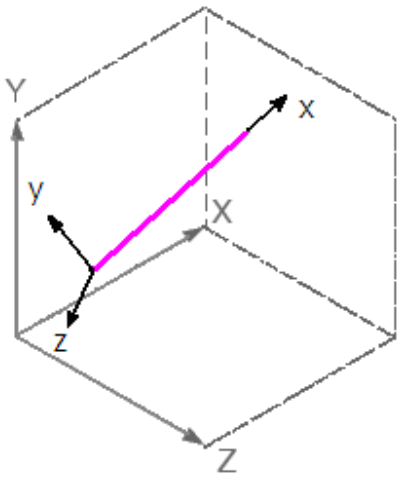


Fig.1

```
D=struct('m',m,'n',n,'Coord',Coord','Con',Con','Re',Re','Load',Load','w',w','E',E','G',G','A',A','Iz',Iz','Iy',Iy','J',J','St',St','be',be');
```

All of the variables are transposed and stored in a structure array in the name of D

Program MSA.m

```
1  function [Q,V,R]=MSA(D)
2  m=D.m;n=D.n;Ni=zeros(12,12,m);S=zeros(6*n);Pf=S(:,1);Q=zeros(12,m);Qfi=Q;Ei=Q;
3  for i=1:m
4      H=D.Con(:,i);C=D.Coord(:,H(2))-D.Coord(:,H(1));e=[6*H(1)-5:6*H(1),6*H(2)-5:6*H(2)];c=D.be(i);
5      [a,b,L]=cart2sph(C(1),C(3),C(2));ca=cos(a);sa=sin(a);cb=cos(b);sb=sin(b);cc=cos(c);sc=sin(c);
6      r=[1 0 0;0 cc sc;0 -sc cc]*[cb sb 0;-sb cb 0;0 0 1]*[ca 0 sa;0 1 0;-sa 0 ca];T=kron(eye(4),r);
7      co=2*L*[6/L 3 2*L L];x=D.A(i)*L^2;y=D.Iy(i)*co;z=D.Iz(i)*co;g=D.G(i)*D.J(i)*L^2/D.E(i);
8      K1=diag([x,z(1),y(1)]);K2=[0 0 0;0 0 z(2);0 -y(2) 0];K3=diag([g,y(3),z(3)]);K4=diag([-g,y(4),z(4)]);
9      K=D.E(i)/L^3*[K1 K2 -K1 K2;K2' K3 -K2' K4;-K1 -K2 K1 -K2;K2' K4 -K2' K3];
10     w=D.w(:,i)';Qf=-L^2/12*[6*w/L 0 -w(3) w(2) 6*w/L 0 w(3) -w(2)]';Qfs=K*T*D.St(e)';
11     A=diag([0 -0.5 -0.5]);B(2,3)=1.5/L;B(3,2)=-1.5/L;W=diag([1,0,0]);Z=zeros(3);M=eye(12);p=4:6;q=10:12;
12     switch 2*H(3)+H(4)
13         case 0;B=2*B/3;M(:,[p,q])=[-B -B;W Z;B B;Z W];case 1;M(:,p)=[-B;W;B;A];case 2;M(:,q)=[-B;A;B;W];
14     end
15     K=M*K;Ni(:,:,i)=K*T;S(e,e)=S(e,e)+T'*Ni(:,:,i);Qfi(:,i)=M*Qf;Pf(e)=Pf(e)+T'*M*(Qf+Qfs);Ei(:,i)=e;
16 end
17 V=1-(D.Re|D.St);f=find(V);V(f)=S(f,f)\(D.Load(f)-Pf(f));R=reshape(S*V(:)+Pf,6,n);R(f)=0;V=V+D.St;
18 for i=1:m
19     Q(:,i)=Ni(:,:,i)*V(Ei(:,i))+Qfi(:,i);
20 end
```

Line : 1 INPUTS & OUTPUTS

```
function [Q,V,R]=MSA(D)
```

Variable	Size	Description	
D		input data refers to program Data.m	
Q	12*m	internal forces and moments in local coordinate system	
		rows 1:3	components of forces in x,y,z directions at beginning node
		rows 4:6	components of moments about x,y,z directions at beginning node
		rows 7:9	components of forces in x,y,z directions at beginning node
		rows 10:12	components of moments about x,y,z directions at end node
V	6*n	deflections in global coordinate system	
		rows 1:3	displacement in X,Y,Z directions
		rows 4:6	rotation about X,Y,Z directions
R	6*n	reactions for supported nodes in global coordinate system	
		rows 1:3	components of forces in X,Y,Z directions
		rows 4:6	components of moments about X,Y,Z directions

Line : 2 INITIALIZATION

```
m=D.m;n=D.n;Ni=zeros(12,12,m);S=zeros(6*n);Pf=S(:,1);Q=zeros(12,m);Qfi=Q;Ei=Q;
```

Var.	Size	Description	
m	1*1	total number of members	
n	1*1	total number of nodes	
Ni	12*12*m	the matrix to store K*T for each member	
S	6n*6n	global stiffness matrix of the structure	
Pf	6n*1	element fixed end forces in global coordinate	
Q	12*m	internal forces and moments in local coordinate system for each member	
Qfi	12*m	element fixed end forces in local coordinate for each member	
Ei	12*m	member code numbers* (mcn) in global stiffness matrix for each member	
		rows 1:6	mcn corresponding to beginning node
		rows 7:12	mcn corresponding to end node

* number of degrees of freedom(ndof) and restrained coordinate numbers for a member

Lines : 4-6 **COORDINATE TRANSFORMATION**

```
H=D.Con(:,i);C=D.Coord(:,H(2))-D.Coord(:,H(1));e=[6*H(1)-5:6*H(1),6*H(2)-5:6*H(2)];c=D.be(i);
[a,b,L]=cart2sph(C(1),C(3),C(2));ca=cos(a);sa=sin(a);cb=cos(b);sb=sin(b);cc=cos(c);sc=sin(c);
r=[1 0 0;0 cc sc;0 -sc cc]*[cb sb 0;-sb cb 0;0 0 1]*[ca 0 sa;0 1 0;-sa 0 ca];T=kron(eye(4),r);
```

Variable	Size	Description
H	4*1	connectivity and release of the both member ends
C	3*1	difference of beginning and end nodes coordinates
e	1*12	member code numbers (mcn) in global stiffness matrix for a member
T	12*12	transformation matrix related to the coordinate transformation which in considering member orientation

See Fig.2

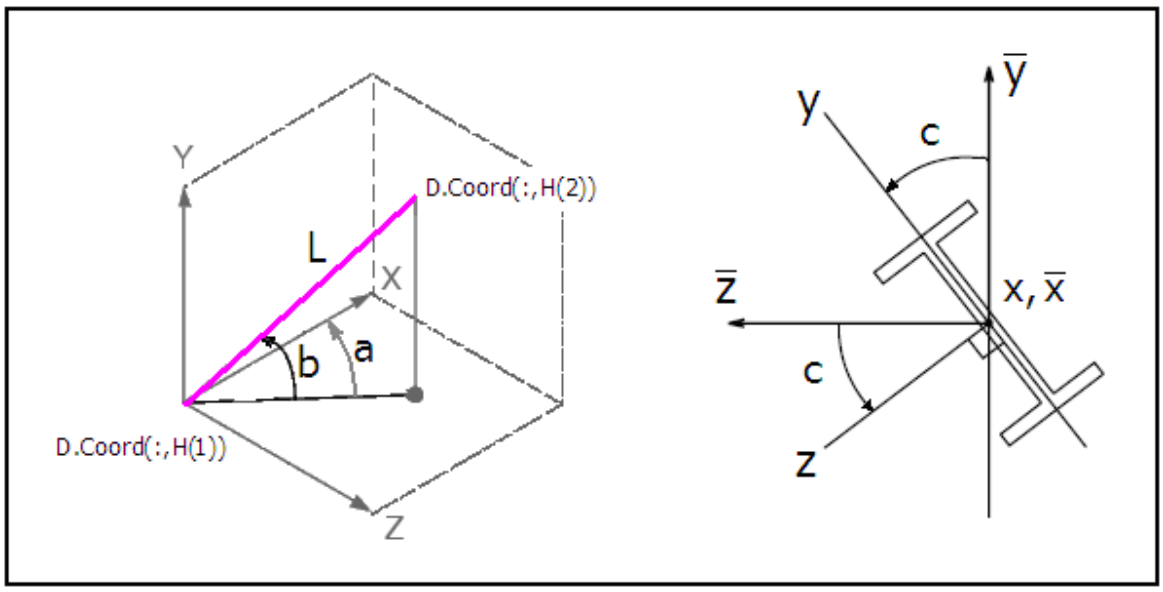


Fig.2 coordinate transformation

Lines : 7-9 **LOCAL STIFFNESS MATRIX**

```
co=2*L*[6/L 3 2*L  L];x=D.A(i)*L^2;y=D.Iy(i)*co;z=D.Iz(i)*co;g=D.G(i)*D.J(i)*L^2/D.E(i);
K1=diag([x,z(1),y(1)]);K2=[0 0 0;0 0 z(2);0 -y(2) 0];K3=diag([g,y(3),z(3)]);K4=diag([-g,y(4),z(4)]);
K=D.E(i)/L^3*[K1 K2 -K1 K2;K2' K3 -K2' K4;-K1 -K2 K1 -K2;K2' K4 -K2' K3];
```

Variable	Size	Description
K	12*12	local stiffness matrix for each member

$$K = \frac{E}{L^3}\begin{bmatrix} x & 0 & 0 & 0 & 0 & 0 & -x & 0 & 0 & 0 & 0 & 0 \\ & z(1) & 0 & 0 & 0 & z(2) & 0 & -z(1) & 0 & 0 & 0 & z(2) \\ & & y(1) & 0 & -y(2) & 0 & 0 & 0 & -y(1) & 0 & -y(2) & 0 \\ & & & g & 0 & 0 & 0 & 0 & 0 & -g & 0 & 0 \\ & & & & y(3) & 0 & 0 & 0 & y(2) & 0 & y(4) & 0 \\ & & & & & z(3) & 0 & -z(2) & 0 & 0 & 0 & z(4) \\ & & & & & & x & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & z(1) & 0 & 0 & 0 & -z(2) \\ & & & & Sym. & & & & y(1) & 0 & y(2) & 0 \\ & & & & & & & & & g & 0 & 0 \\ & & & & & & & & & & y(3) & 0 \\ & & & & & & & & & & & z(3) \end{bmatrix}$$

$$K = \frac{E}{L^3}\begin{bmatrix} K_1 & K_2 & -K_1 & K_2 \\ K_2^t & K_3 & -K_2^t & K_4 \\ -K_1 & -K_2 & K_1 & -K_2 \\ K_2^t & K_4 & -K_2^t & K_3 \end{bmatrix}$$

$$x = A * L^2$$

$$y = 2 * L * I_y * \begin{bmatrix} 6/L & 3 & 2*L & L \end{bmatrix}$$

$$z = 2 * L * I_z * \begin{bmatrix} 6/L & 3 & 2*L & L \end{bmatrix}$$

$$g = G * J * L^2 / E$$

Note: when shear deformation is considerable, lines 7-9 should be changed to:

```
x=D.A(i)/L;g=D.G(i)*D.J(i)/(D.E(i)*L);ez=D.E(i)*D.Iz(i)/(D.Ay(i)*D.G(i));ey=D.E(i)*D.Iy(i)/(D.Az(i)*D.G(i));
z=D.Iz(i)/(L*(L^2/12+ez))*[1  L/2  (L^2/3+ez)  (L^2/6-ez)];
y=D.Iy(i)/(L*(L^2/12+ey))*[1  L/2  (L^2/3+ey)  (L^2/6-ey)];
K1=diag([x,z(1),y(1)]);K2=[0 0 0;0 0 z(2);0 -y(2) 0];K3=diag([g,y(3),z(3)]);K4=diag([-g,y(4),z(4)]);
K=D.E(i)*[K1 K2 -K1 K2;K2' K3 -K2' K4;-K1 -K2 K1 -K2;K2' K4 -K2' K3];
```

Line : 10 **LOCAL FIXED-END FORCE VECTOR**

```
w=D.w(:,i)';Qf=-L^2/12*[6*w/L 0 -w(3) w(2) 6*w/L 0 w(3) -w(2)]';Qfs=K*T*D.St(e)';
```

Variable	Size	Description
w	1*3	uniform loads in local coordinate system for each member
Qf	12*1	local fixed-end force vector for a member, corresponding to external loads
Qfs	12*1	local fixed-end force vector for a member, corresponding to support displacements

Local fixed-end force vector for the members of space frame is shown as follow: (Fig.3)

$$Q_f = \{ FA_b, FS_{by}, FS_{bz}, FT_b, FM_{by}, FM_{bz}, FA_e, FS_{ey}, FS_{ez}, FT_e, FM_{ey}, FM_{ez} \}^t$$

As instance, for a uniform loading:

$$Q_f = \left\{ -\frac{\omega_x L}{2}, -\frac{\omega_y L}{2}, -\frac{\omega_z L}{2}, 0, \frac{\omega_z L^2}{12}, -\frac{\omega_y L^2}{12}, -\frac{\omega_x L}{2}, -\frac{\omega_y L}{2}, -\frac{\omega_z L}{2}, 0, -\frac{\omega_z L^2}{12}, \frac{\omega_y L^2}{12} \right\}^t$$

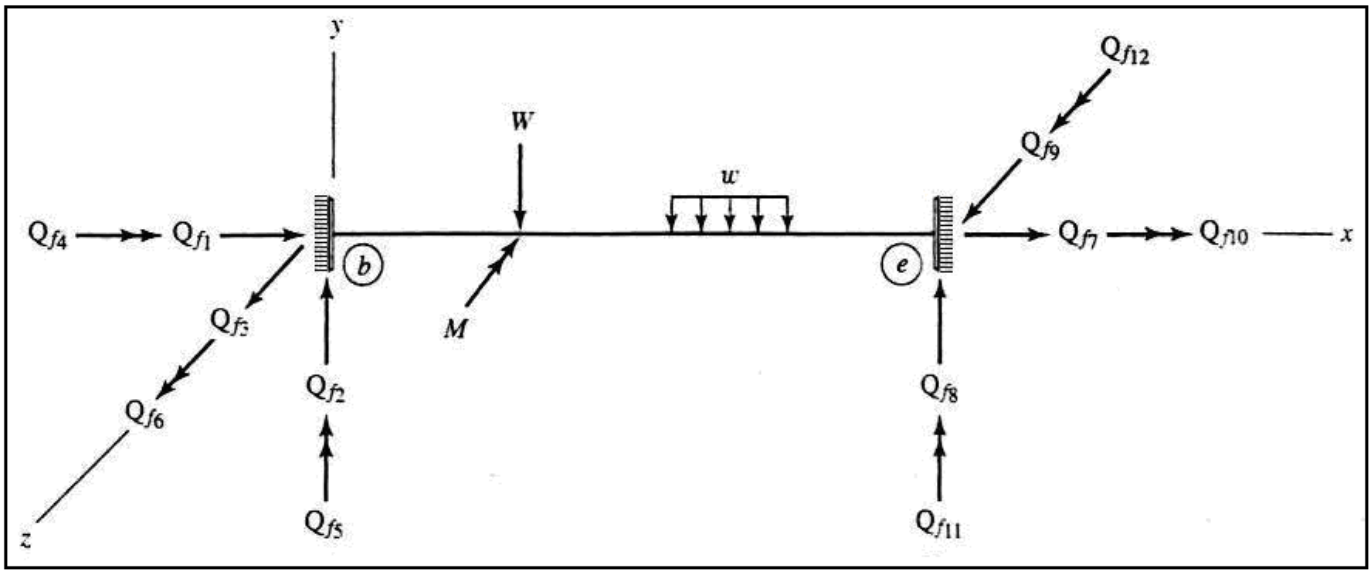


Fig.3 Member Fixed-End Forces in the Local Coordinate System

For other loadings, The member local Fixed-End force vector, must be computed as formulized in Table. 1

Table.1 Fixed-end moments ,shears and axial forces for various loading conditions

FIXED-END FORCE EXPRESSIONS

Fixed-End Moments, Shears and Axial Forces for Various Loading Conditions

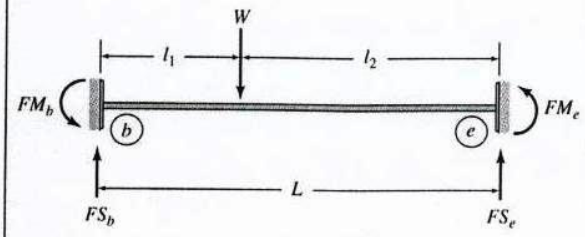
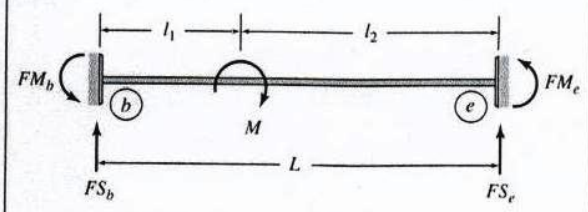
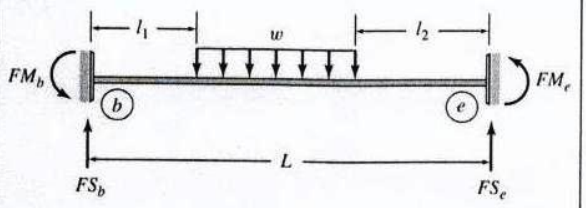
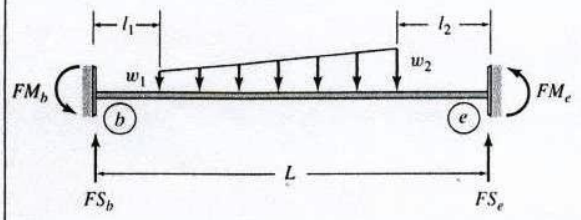
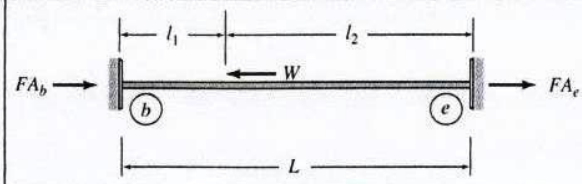
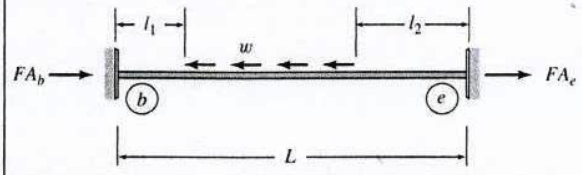
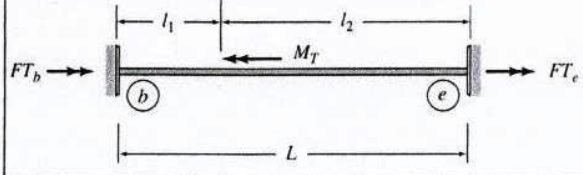
No.	Loading	Equations for Fixed-End Moments, Shears, and Axial Forces
1.		$FS_b = \frac{Wl_2^2}{L^3}(3l_1 + l_2)$ $FM_b = \frac{Wl_1l_2^2}{L^2}$ $FS_e = \frac{Wl_1^2}{L^3}(l_1 + 3l_2)$ $FM_e = -\frac{Wl_1^2l_2}{L^2}$
2.		$FS_b = -\frac{6Ml_1l_2}{L^3}$ $FM_b = \frac{Ml_2}{L^2}(l_2 - 2l_1)$ $FS_e = \frac{6Ml_1l_2}{L^3}$ $FM_e = \frac{Ml_1}{L^2}(l_1 - 2l_2)$
3.		$FS_b = \frac{wL}{2}\left[1 - \frac{l_1}{L^4}(2L^3 - 2l_1^2L + l_1^3) - \frac{l_2^3}{L^4}(2L - l_2)\right]$ $FM_b = \frac{wL^2}{12}\left[1 - \frac{l_1^2}{L^4}(6L^2 - 8l_1L + 3l_1^2) - \frac{l_2^3}{L^4}(4L - 3l_2)\right]$ $FS_e = \frac{wL}{2}\left[1 - \frac{l_1^3}{L^4}(2L - l_1) - \frac{l_2}{L^4}(2L^3 - 2l_2^2L + l_2^3)\right]$ $FM_e = -\frac{wL^2}{12}\left[1 - \frac{l_1^3}{L^4}(4L - 3l_1) - \frac{l_2^2}{L^4}(6L^2 - 8l_2L + 3l_2^2)\right]$

Table.1 (continued)

No.	Loading	Equations for Fixed-End Moments, Shears, and Axial Forces
4.		$FS_b = \frac{w_1(L-l_1)^3}{20L^3}\left\{(7L+8l_1) - \frac{l_2(3L+2l_1)}{(L-l_1)} \times \left[1+\frac{l_2}{L-l_1}+\frac{l_2^2}{(L-l_1)^2}\right] + \frac{2l_2^4}{(L-l_1)^3}\right\} + \frac{w_2(L-l_1)^3}{20L^3}\left\{(3L+2l_1)\left[1+\frac{l_2}{L-l_1} + \frac{l_2^2}{(L-l_1)^2}\right] - \frac{l_2^3}{(L-l_1)^2}\left[2+\frac{15L-8l_2}{L-l_1}\right]\right\}$ $FM_b = \frac{w_1(L-l_1)^3}{60L^2}\left\{3(L+4l_1) - \frac{l_2(2L+3l_1)}{L-l_1} \times \left[1+\frac{l_2}{L-l_1}+\frac{l_2^2}{(L-l_1)^2}\right] + \frac{3l_2^4}{(L-l_1)^3}\right\} + \frac{w_2(L-l_1)^3}{60L^2}\left\{(2L+3l_1)\left[1+\frac{l_2}{L-l_1} + \frac{l_2^2}{(L-l_1)^2}\right] - \frac{3l_2^3}{(L-l_1)^2}\left[1+\frac{5L-4l_2}{L-l_1}\right]\right\}$ $FS_e = \left(\frac{w_1+w_2}{2}\right)(L-l_1-l_2) - FS_b$ $FM_e = \frac{L-l_1-l_2}{6}[w_1(-2L+2l_1-l_2) - w_2(L-l_1+2l_2)] + FS_b(L) - FM_b$
5.		$FA_b = \frac{Wl_2}{L}$ $FA_e = \frac{Wl_1}{L}$
6.		$FA_b = \frac{w}{2L}(L-l_1-l_2)(L-l_1+l_2)$ $FA_e = \frac{w}{2L}(L-l_1-l_2)(L+l_1-l_2)$
7.		$FT_b = \frac{M_T l_2}{L}$ $FT_e = \frac{M_T l_1}{L}$

Lines : 11-14 **MEMBER RELEASE**

```
W=diag([1,0,0]);X=diag([0 -0.5 -0.5]);Y(2,3)=1.5/L;Y(3,2)=-1.5/L;Z=zeros(3);M=eye(12);p=4:6;q=10:12;
switch 2*H(3)+H(4)
    case 0;Y=2*Y/3;M(:,[p,q])=[-Y -Y;W Z;Y Y;Z W];case 1;M(:,p)=[-Y;W;Y;X];case 2;M(:,q)=[-Y;X;Y;W];
end
```

Variable	Size	Description	
2*H(3)+H(4)	1*1	type of member release*	
		=0	For a member released at both ends
		=1	For a member released at the beginning
		=2	For a member released at the end
		=3	For a member fixed at both ends
M	12*12	A matrix for modifying stiffness matrix and local fixed-end force vector of a released member ends such K=M*K , Qf=M*Qf and Qfs=M*Qfs	

* The bending moments about the y and z axes, are zero at the released member ends

As instance, For a member released at the beginning (2*H(3)+H(4)=1)

$$M=\left[\begin{array}{ccc|ccc|ccc|ccc} 1&0&0&0&0&0&0&0&0&0&0&0\\ 0&1&0&0&0&-\frac{1.5}{L}&0&0&0&0&0&0\\ 0&0&1&0&\frac{1.5}{L}&0&0&0&0&0&0&0\\ \hline 0&0&0&1&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0\\ \hline 0&0&0&0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&\frac{1.5}{L}&0&1&0&0&0&0\\ 0&0&0&0&-\frac{1.5}{L}&0&0&0&1&0&0&0\\ \hline 0&0&0&0&0&0&0&0&0&1&0&0\\ 0&0&0&0&-0.5&0&0&0&0&0&1&0\\ 0&0&0&0&0&-0.5&0&0&0&0&0&1 \end{array}\right]=\left[\begin{array}{cccc} I&-Y&Z&Z\\ Z&W&Z&Z\\ Z&Y&I&Z\\ Z&X&Z&I \end{array}\right]$$

Note: If the member releases are assumed to be in the form of spherical hinges, then the bending moments about the y and z axes, and the torsional moment are zero at the released member ends. In this case, X and W should be changed to:

$$X=\begin{bmatrix}1&0&0\\0&-0.5&0\\0&0&-0.5\end{bmatrix}\quad;\quad W=\text{zeros}(3)$$

$$K = M * K = \frac{E}{L^3}\begin{bmatrix} AL^2 & 0 & 0 & 0 & 0 & 0 & -AL^2 & 0 & 0 & 0 & 0 & 0 \\ & 3I_z & 0 & 0 & 0 & 0 & 0 & -3I_z & 0 & 0 & 0 & 3LI_z \\ & & 3I_y & 0 & 0 & 0 & 0 & 0 & -3I_y & 0 & -3LI_y & 0 \\ & & & GJL^2/E & 0 & 0 & 0 & 0 & 0 & -GJL^2/E & 0 & 0 \\ & & & & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & AL^2 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & 3I_z & 0 & 0 & 0 & -3LI_z \\ & & & Sym. & & & & & 3I_y & 0 & 3LI_y & 0 \\ & & & & & & & & & GJL^2/E & 0 & 0 \\ & & & & & & & & & & 3L^2I_y & 0 \\ & & & & & & & & & & & 3L^2I_z \end{bmatrix}$$

And for a uniform loading:

$$Q_f = M * Q_f = \left\{ -\frac{\omega_x L}{2}, -\frac{3\omega_y L}{8}, -\frac{3\omega_z L}{8}, 0, 0, 0, -\frac{\omega_x L}{2}, -\frac{5\omega_y L}{8}, -\frac{5\omega_z L}{8}, 0, -\frac{\omega_z L^2}{8}, \frac{\omega_y L^2}{8} \right\}^t$$

For a member fixed at both ends (2*H(3)+H(4)=3) then M=eye(12) that means no change in stiffness matrix and local fixed-end force vector.

Line : 15 ASSEMBLING AND STORING

```
K=M*K;Ni(:,:,i)=K*T;S(e,e)=S(e,e)+T'*Ni(:,:,i);Qfi(:,i)=M*Qf;Pf(e)=Pf(e)+T'*M*(Qf+Qfs);Ei(:,i)=e;
```

Variable	Size	Description
Ni	12*12*m	the matrix to store K*T for each member
S	6n*6n	global stiffness matrix of the structure
Qfi	12*m	element fixed end forces in local coordinate for each member
Pf	6n*1	element fixed end forces in global coordinate
Ei	12*m	member code numbers* (mcn) in global stiffness matrix for each member

* number of degrees of freedom(ndof) and restrained coordinate numbers for a member

Notes:
1- The local stiffness matrix is K=M*K
2- Matrix Ni is used for calculating of the internal forces and moments in local coordinate system.
3- It will be mentioned that a classical method saves the transformation matrix for all of the members, which will be stored at a 12*12*m matrix. Instead of it, we produce K*T and store that.

Line : 17 **DISPLACEMENTS AND REACTIONS**

```
V=1-(D.Re|D.St);f=find(V);V(f)=S(f,f)\(D.Load(f)-Pf(f));R=reshape(S*V(:)+Pf,6,n);R(f)=0;V=V+D.St;
```

Variable	Size	Description	
V	6*n	Deflections in global coordinate system	
		rows 1:3	displacement in X,Y,Z direction
		rows 4:6	rotation about X,Y,Z direction
R	6*n	Supports reactions in global coordinate system	
		rows 1:3	forces in X,Y,Z direction
		rows 4:6	moments about X,Y,Z direction
f	ndof*1	A vector that indicates the number of degree of freedom	

Note: by the V=1-(D.Re|D.St) we can have a specific displacement or rotation in a free node.

Lines : 18-20 **INTERNAL FORCES AND MOMENTS**

```
for i=1:m
    Q(:,i)=Ni(:,:,i)*V(Ei(:,i))+Qfi(:,i);
end
```

Variable	Size	Description	
Q	12*m	internal forces and moments in local coordinate system	
		rows 1:3	components of forces in x,y,z directions at beginning node
		rows 4:6	components of moments about x,y,z directions at beginning node
		rows 7:9	components of forces in x,y,z directions at beginning node
		rows 10:12	components of moments about x,y,z directions at end node

Note: the internal forces and moments can be achieved by using Q=K*U+Qf that U refers to Deflections in local coordinate system. It can be rewritten as:

$$Q = K * U + Q_f = K * (T * V) + Q_f = (K * T) * V + Q_f = N * V + Q_f$$

Fig.4 shows the end forces and end displacements in the local and global coordinate system

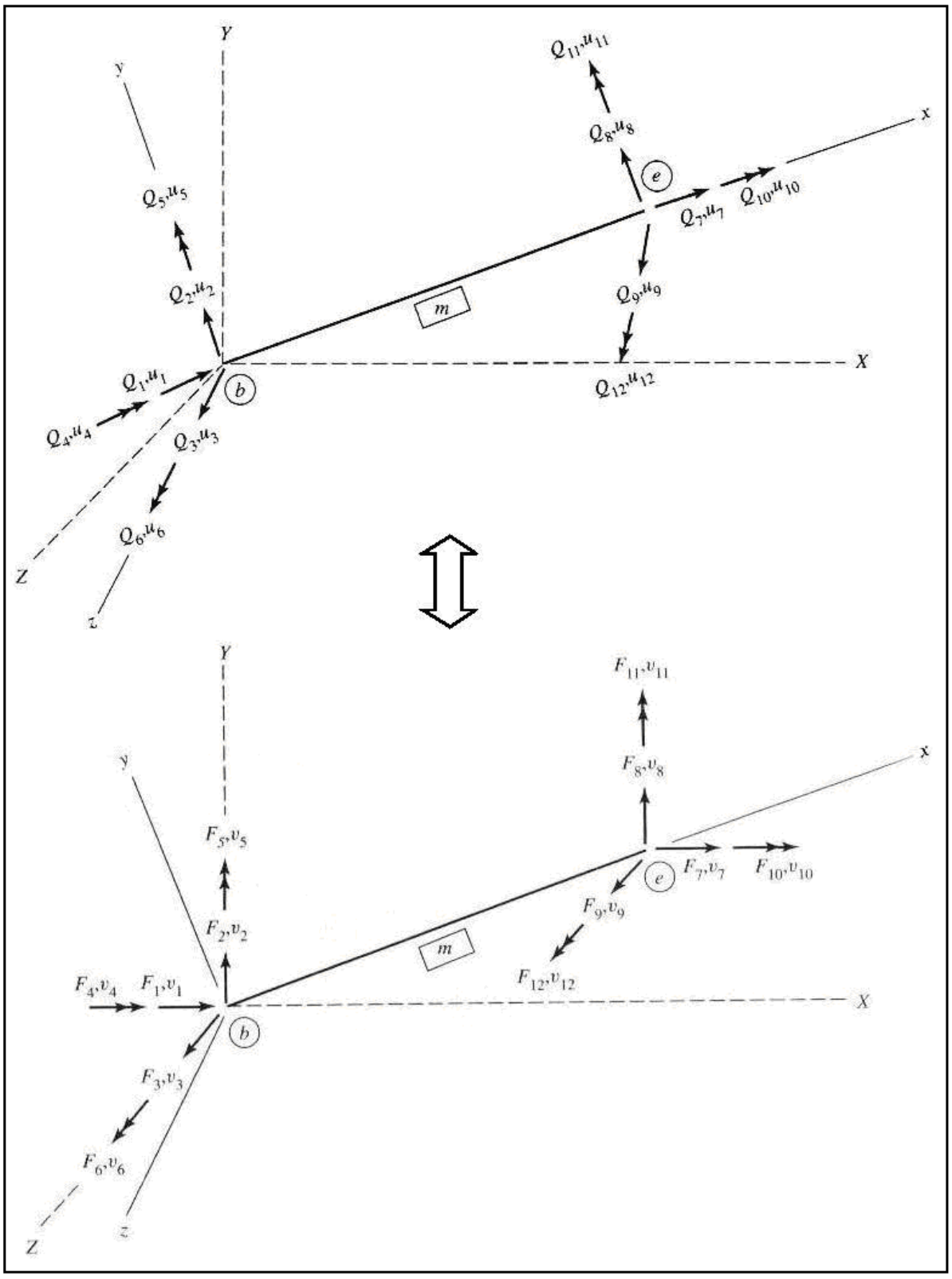


Fig.4 Member ends forces and end displacements in the local and global coordinate system

◇ EXAMPLES

Simple Illustrative Example: 3D Frame

Determine the joint displacements, member end forces, and support reactions for the three-member space frame shown in Fig.5, using the matrix stiffness method. [1]

The space frame has six degrees of freedom and 18 restrained coordinated, as shown in Fig.6

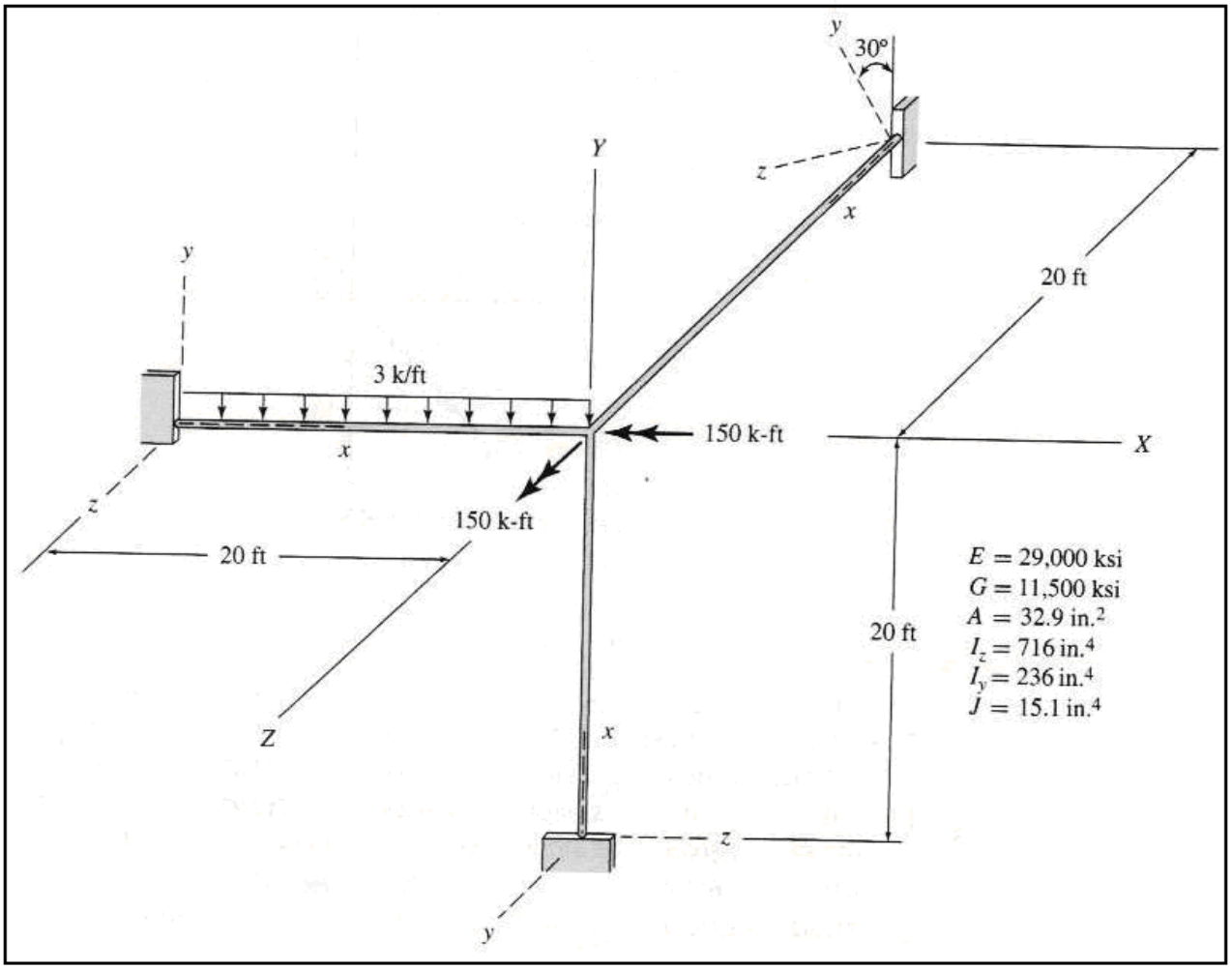


Fig.5 A space Frame [1]

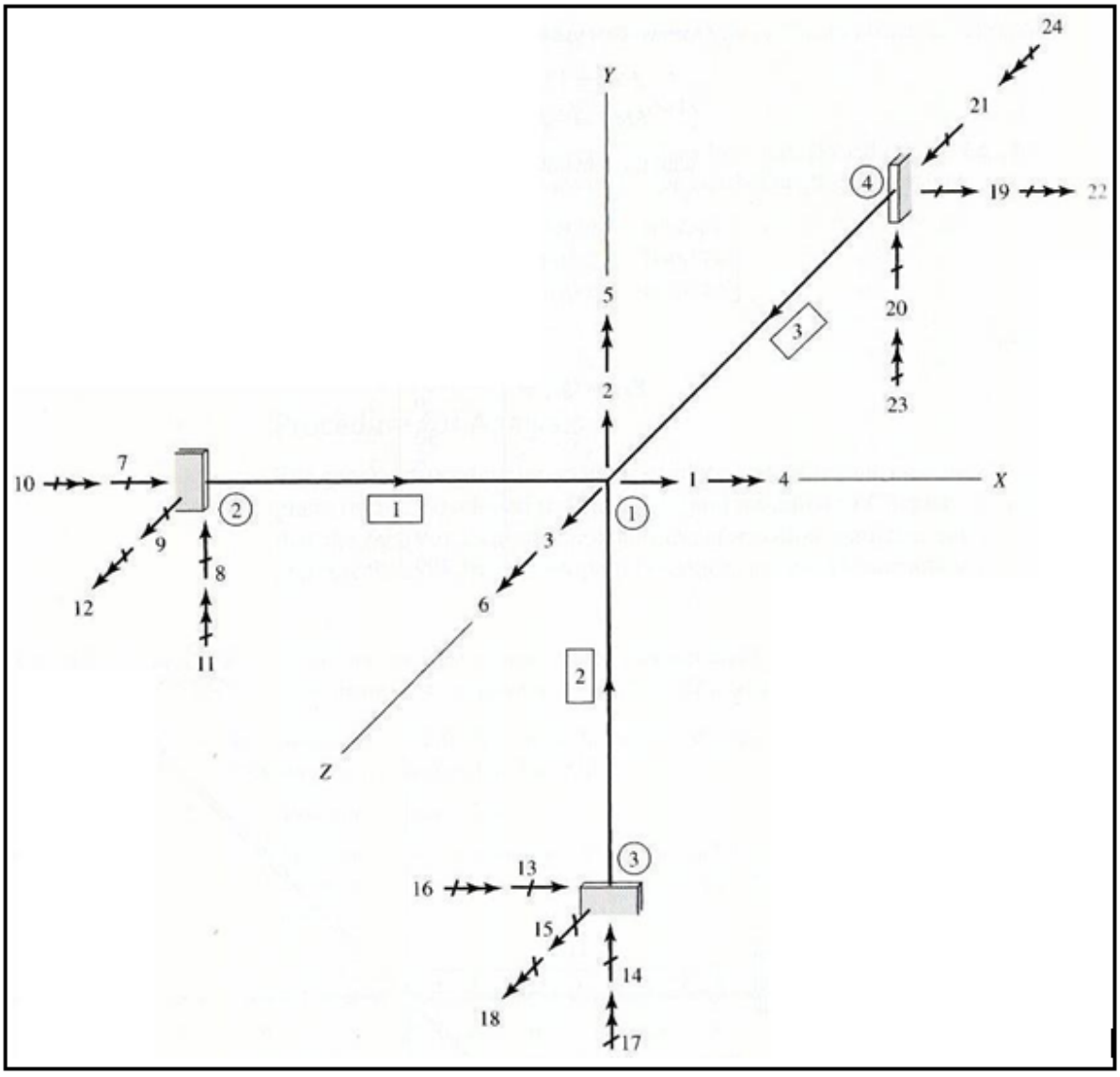


Fig.6 Analytical Model [1]

DataF3D.m

```
function  D=DataF3D
%  Units: kips & in
m=3;n=4;
Coord=[0 0 0;-240 0 0;0 -240 0;0 0 -240];
Con=[2 1 1 1;3 1 1 1;4 1 1 1];
Re=[0 0 0 0 0 0;1 1 1 1 1 1;1 1 1 1 1 1;1 1 1 1 1 1];
%  or:  Re=ones(n,6);Re(1,:)=zeros(1,6);
Load=[0 0 0 -1800 0 1800;0 0 0 0 0 0;0 0 0 0 0 0;0 0 0 0 0 0];
%  or:  Load=zeros(n,6);Load(1,[4,6])=[-1800 1800];
w=[0 -0.25 0;0 0 0;0 0 0];
%  or:  w=zeros(m,3);w(1,2)=-0.25;
E=ones(1,m)*29000;
G=ones(1,m)*11500;
A=ones(1,m)*32.9;
Iz=ones(1,m)*716;
Iy=ones(1,m)*236;
J=ones(1,m)*15.1;
St=zeros(n,6);
be=[0 90 30]*pi/180;
D=struct('m',m,'n',n,'Coord',Coord','Con',Con','Re',Re','Load',Load',...
     'w',w','E',E','G',G','A',A','Iz',Iz','Iy',Iy','J',J','St',St','be',be');
```

MSA.m

```
>> D=DataF3D;[Q,V,R]=MSA(D)
```

Member 1

i =

1

H =

2
1
1
1

e =

7	8	9	10	11	12	1	2	3	4	5	6

a =

0

b =

0

c =

0

T =

1	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	1

K =

1.0e+005 *

												mcn
0.0398	0	0	0	0	0	-0.0398	0	0	0	0	0	7
0	0.0002	0	0	0	0.0216	0	-0.0002	0	0	0	0.0216	8
0	0	0.0001	0	-0.0071	0	0	0	-0.0001	0	-0.0071	0	9
0	0	0	0.0072	0	0	0	0	0	-0.0072	0	0	10
0	0	-0.0071	0	1.1407	0	0	0	0.0071	0	0.5703	0	11
0	0.0216	0	0	0	3.4607	0	-0.0216	0	0	0	1.7303	12
-0.0398	0	0	0	0	0	0.0398	0	0	0	0	0	1
0	-0.0002	0	0	0	-0.0216	0	0.0002	0	0	0	-0.0216	2
0	0	-0.0001	0	0.0071	0	0	0	0.0001	0	0.0071	0	3
0	0	0	-0.0072	0	0	0	0	0	0.0072	0	0	4
0	0	-0.0071	0	0.5703	0	0	0	0.0071	0	1.1407	0	5
0	0.0216	0	0	0	1.7303	0	-0.0216	0	0	0	3.4607	6

Qf =

0
30
0
0
0
1200
0
30
0
0
0
-1200

```
Qfs =
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
M =
     1     0     0     0     0     0     0     0     0     0     0     0
     0     1     0     0     0     0     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0     0     0     0
     0     0     0     1     0     0     0     0     0     0     0     0
     0     0     0     0     1     0     0     0     0     0     0     0
     0     0     0     0     0     1     0     0     0     0     0     0
     0     0     0     0     0     0     1     0     0     0     0     0
     0     0     0     0     0     0     0     1     0     0     0     0
     0     0     0     0     0     0     0     0     1     0     0     0
     0     0     0     0     0     0     0     0     0     1     0     0
     0     0     0     0     0     0     0     0     0     0     1     0
     0     0     0     0     0     0     0     0     0     0     0     1

Ni(:,:,1) =
  1.0e+005 *
    0.0398         0         0         0         0         0   -0.0398         0         0         0         0         0
         0    0.0002         0         0         0    0.0216         0   -0.0002         0         0         0    0.0216
         0         0    0.0001         0   -0.0071         0         0         0   -0.0001         0   -0.0071         0
         0         0         0    0.0072         0         0         0         0         0   -0.0072         0         0
         0         0   -0.0071         0    1.1407         0         0         0    0.0071         0    0.5703         0
         0    0.0216         0         0         0    3.4607         0   -0.0216         0         0         0    1.7303
   -0.0398         0         0         0         0         0    0.0398         0         0         0         0         0
         0   -0.0002         0         0         0   -0.0216         0    0.0002         0         0         0   -0.0216
         0         0   -0.0001         0    0.0071         0         0         0    0.0001         0    0.0071         0
         0         0         0   -0.0072         0         0         0         0         0    0.0072         0         0
         0         0   -0.0071         0    0.5703         0         0         0    0.0071         0    1.1407         0
         0    0.0216         0         0         0    1.7303         0   -0.0216         0         0         0    3.4607

Qfi =
          0          0          0
         30          0          0
          0          0          0
          0          0          0
          0          0          0
       1200          0          0
          0          0          0
         30          0          0
          0          0          0
          0          0          0
          0          0          0
      -1200          0          0
```

```
Pf =
           0
          30
           0
           0
           0
       -1200
           0
          30
           0
           0
           0
        1200
           0
           0
           0
           0
           0
           0
           0
           0
           0
           0
           0
```

Member 2

```
i =
     2
H =
     3
     1
     1
     1
e =
    13    14    15    16    17    18     1     2     3     4     5     6
a =
     0
b =
    1.5708
c =
    1.5708

T =

     0     1     0     0     0     0     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0     0     0     0
     1     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     1     0     0     0     0     0     0     0
     0     0     0     0     0     1     0     0     0     0     0     0
     0     0     0     1     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     1     0     0     0     0
     0     0     0     0     0     0     0     0     1     0     0     0
     0     0     0     0     0     0     1     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     1     0
     0     0     0     0     0     0     0     0     0     0     0     1
     0     0     0     0     0     0     0     0     0     1     0     0
```

K =

1.0e+005 *

												mcn
0.0398	0	0	0	0	0	-0.0398	0	0	0	0	0	13
0	0.0002	0	0	0	0.0216	0	-0.0002	0	0	0	0.0216	14
0	0	0.0001	0	-0.0071	0	0	0	-0.0001	0	-0.0071	0	15
0	0	0	0.0072	0	0	0	0	0	-0.0072	0	0	16
0	0	-0.0071	0	1.1407	0	0	0	0.0071	0	0.5703	0	17
0	0.0216	0	0	0	3.4607	0	-0.0216	0	0	0	1.7303	18
-0.0398	0	0	0	0	0	0.0398	0	0	0	0	0	1
0	-0.0002	0	0	0	-0.0216	0	0.0002	0	0	0	-0.0216	2
0	0	-0.0001	0	0.0071	0	0	0	0.0001	0	0.0071	0	3
0	0	0	-0.0072	0	0	0	0	0	0.0072	0	0	4
0	0	-0.0071	0	0.5703	0	0	0	0.0071	0	1.1407	0	5
0	0.0216	0	0	0	1.7303	0	-0.0216	0	0	0	3.4607	6

Qf =

0
0
0
0
0
0
0
0
0
0
0
0

Qfs =

0
0
0
0
0
0
0
0
0
0
0
0

M =

1	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	1

Ni(:,:,2) =

1.0e+005 *

0.0000	0.0398	0	0	0	0	-0.0000	-0.0398	0	0	0	0
-0.0000	0.0000	0.0002	0.0216	-0.0000	0.0000	0.0000	-0.0000	-0.0002	0.0216	-0.0000	0.0000
0.0001	-0.0000	0.0000	0.0000	-0.0000	-0.0071	-0.0001	0.0000	-0.0000	0.0000	-0.0000	-0.0071
0	0	0	0.0000	0.0072	0	0	0	0	-0.0000	-0.0072	0
-0.0071	0.0000	-0.0000	-0.0000	0.0000	1.1407	0.0071	-0.0000	0.0000	-0.0000	0.0000	0.5703
-0.0000	0.0000	0.0216	3.4607	-0.0000	0.0000	0.0000	-0.0000	-0.0216	1.7303	-0.0000	0.0000
-0.0000	-0.0398	0	0	0	0	0.0000	0.0398	0	0	0	0
0.0000	-0.0000	-0.0002	-0.0216	0.0000	-0.0000	-0.0000	0.0000	0.0002	-0.0216	0.0000	-0.0000
-0.0001	0.0000	-0.0000	-0.0000	0.0000	0.0071	0.0001	-0.0000	0.0000	-0.0000	0.0000	0.0071
0	0	0	-0.0000	-0.0072	0	0	0	0	0.0000	0.0072	0
-0.0071	0.0000	-0.0000	-0.0000	0.0000	0.5703	0.0071	-0.0000	0.0000	-0.0000	0.0000	1.1407
-0.0000	0.0000	0.0216	1.7303	-0.0000	0.0000	0.0000	-0.0000	-0.0216	3.4607	-0.0000	0.0000

Qfi =

0	0	0
30	0	0
0	0	0
0	0	0
0	0	0
1200	0	0
0	0	0
30	0	0
0	0	0
0	0	0
0	0	0
-1200	0	0

Pf =

0
30
0
0
0
-1200
0
30
0
0
0
1200
0
0
0
0
0
0
0
0
0
0
0
0

Member 3

```
i =
     3
H =
     4
     1
     1
     1
e =
    19    20    21    22    23    24     1     2     3     4     5     6
a =
    1.5708
b =
     0
c =
    0.5236
T =
    0.0000         0    1.0000         0         0         0         0         0         0         0         0         0
   -0.5000    0.8660    0.0000         0         0         0         0         0         0         0         0         0
   -0.8660   -0.5000    0.0000         0         0         0         0         0         0         0         0         0
         0         0         0    0.0000         0    1.0000         0         0         0         0         0         0
         0         0         0   -0.5000    0.8660    0.0000         0         0         0         0         0         0
         0         0         0   -0.8660   -0.5000    0.0000         0         0         0         0         0         0
         0         0         0         0         0         0    0.0000         0    1.0000         0         0         0
         0         0         0         0         0         0   -0.5000    0.8660    0.0000         0         0         0
         0         0         0         0         0         0   -0.8660   -0.5000    0.0000         0         0         0
         0         0         0         0         0         0         0         0         0    0.0000         0    1.0000
         0         0         0         0         0         0         0         0         0   -0.5000    0.8660    0.0000
         0         0         0         0         0         0         0         0         0   -0.8660   -0.5000    0.0000

K =
  1.0e+005 *                                                                                                        mcn
    0.0398         0         0         0         0         0   -0.0398         0         0         0         0         0  19
         0    0.0002         0         0         0    0.0216         0   -0.0002         0         0         0    0.0216  20
         0         0    0.0001         0   -0.0071         0         0         0   -0.0001         0   -0.0071         0  21
         0         0         0    0.0072         0         0         0         0         0   -0.0072         0         0  22
         0         0   -0.0071         0    1.1407         0         0         0    0.0071         0    0.5703         0  23
         0    0.0216         0         0         0    3.4607         0   -0.0216         0         0         0    1.7303  24
   -0.0398         0         0         0         0         0    0.0398         0         0         0         0         0   1
         0   -0.0002         0         0         0   -0.0216         0    0.0002         0         0         0   -0.0216   2
         0         0   -0.0001         0    0.0071         0         0         0    0.0001         0    0.0071         0   3
         0         0         0   -0.0072         0         0         0         0         0    0.0072         0         0   4
         0         0   -0.0071         0    0.5703         0         0         0    0.0071         0    1.1407         0   5
         0    0.0216         0         0         0    1.7303         0   -0.0216         0         0         0    3.4607   6
Qf =
          0
          0
          0
          0
          0
          0
          0
          0
          0
          0
          0
          0
```

```
Qfs =
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
M =
     1     0     0     0     0     0     0     0     0     0     0     0
     0     1     0     0     0     0     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0     0     0     0
     0     0     0     1     0     0     0     0     0     0     0     0
     0     0     0     0     1     0     0     0     0     0     0     0
     0     0     0     0     0     1     0     0     0     0     0     0
     0     0     0     0     0     0     1     0     0     0     0     0
     0     0     0     0     0     0     0     1     0     0     0     0
     0     0     0     0     0     0     0     0     1     0     0     0
     0     0     0     0     0     0     0     0     0     1     0     0
     0     0     0     0     0     0     0     0     0     0     1     0
     0     0     0     0     0     0     0     0     0     0     0     1

Ni(:,:,3) =
  1.0e+005 *
    0.0000         0    0.0398         0         0         0   -0.0000         0   -0.0398         0         0         0
   -0.0001    0.0002    0.0000   -0.0187   -0.0108    0.0000    0.0001   -0.0002   -0.0000   -0.0187   -0.0108    0.0000
   -0.0001   -0.0000    0.0000    0.0036   -0.0062   -0.0000    0.0001    0.0000   -0.0000    0.0036   -0.0062   -0.0000
         0         0         0    0.0000         0    0.0072         0         0         0   -0.0000         0   -0.0072
    0.0062    0.0036   -0.0000   -0.5703    0.9878    0.0000   -0.0062   -0.0036    0.0000   -0.2852    0.4939    0.0000
   -0.0108    0.0187    0.0000   -2.9970   -1.7303    0.0000    0.0108   -0.0187   -0.0000   -1.4985   -0.8652    0.0000
   -0.0000         0   -0.0398         0         0         0    0.0000         0    0.0398         0         0         0
    0.0001   -0.0002   -0.0000    0.0187    0.0108   -0.0000   -0.0001    0.0002    0.0000    0.0187    0.0108   -0.0000
    0.0001    0.0000   -0.0000   -0.0036    0.0062    0.0000   -0.0001   -0.0000    0.0000   -0.0036    0.0062    0.0000
         0         0         0   -0.0000         0   -0.0072         0         0         0    0.0000         0    0.0072
    0.0062    0.0036   -0.0000   -0.2852    0.4939    0.0000   -0.0062   -0.0036    0.0000   -0.5703    0.9878    0.0000
   -0.0108    0.0187    0.0000   -1.4985   -0.8652    0.0000    0.0108   -0.0187   -0.0000   -2.9970   -1.7303    0.0000

Qfi =
         0         0         0
        30         0         0
         0         0         0
         0         0         0
         0         0         0
      1200         0         0
         0         0         0
        30         0         0
         0         0         0
         0         0         0
         0         0         0
     -1200         0         0
```

```
Pf =
           0
          30
           0
           0
           0
       -1200
           0
          30
           0
           0
           0
        1200
           0
           0
           0
           0
           0
           0
           0
           0
           0
           0
           0
```

Assembling

```
f =
     1
     2
     3
     4
     5
     6
S(f,f)
  1.0e+005 *
    0.0399   -0.0001    0.0000   -0.0063   -0.0108    0.0071
   -0.0001    0.0401    0.0000    0.0180    0.0063   -0.0216
    0.0000    0.0000    0.0400   -0.0216    0.0071   -0.0000
   -0.0063    0.0180   -0.0216    6.3486    1.0046   -0.0000
   -0.0108    0.0063    0.0071    1.0046    2.8686   -0.0000
    0.0071   -0.0216   -0.0000   -0.0000   -0.0000    4.6086
```

Final Result

V =

-0.0013522	0	0	0
-0.0027965	0	0	0
-0.001812	0	0	0
-0.0030021	0	0	0
0.0010569	0	0	0
0.0064986	0	0	0

Q =

5.3757	11.117	7.2034
44.106	-6.4607	4.5118
-0.74272	-4.6249	-1.7379
2.1722	-0.76472	-4.702
58.987	369.67	139.65
2330.5	-515.55	362.21
-5.3757	-11.117	-7.2034
15.894	6.4607	-4.5118
0.74272	4.6249	1.7379
-2.1722	0.76472	4.702
119.27	740.31	277.46
1055	-1035	720.63

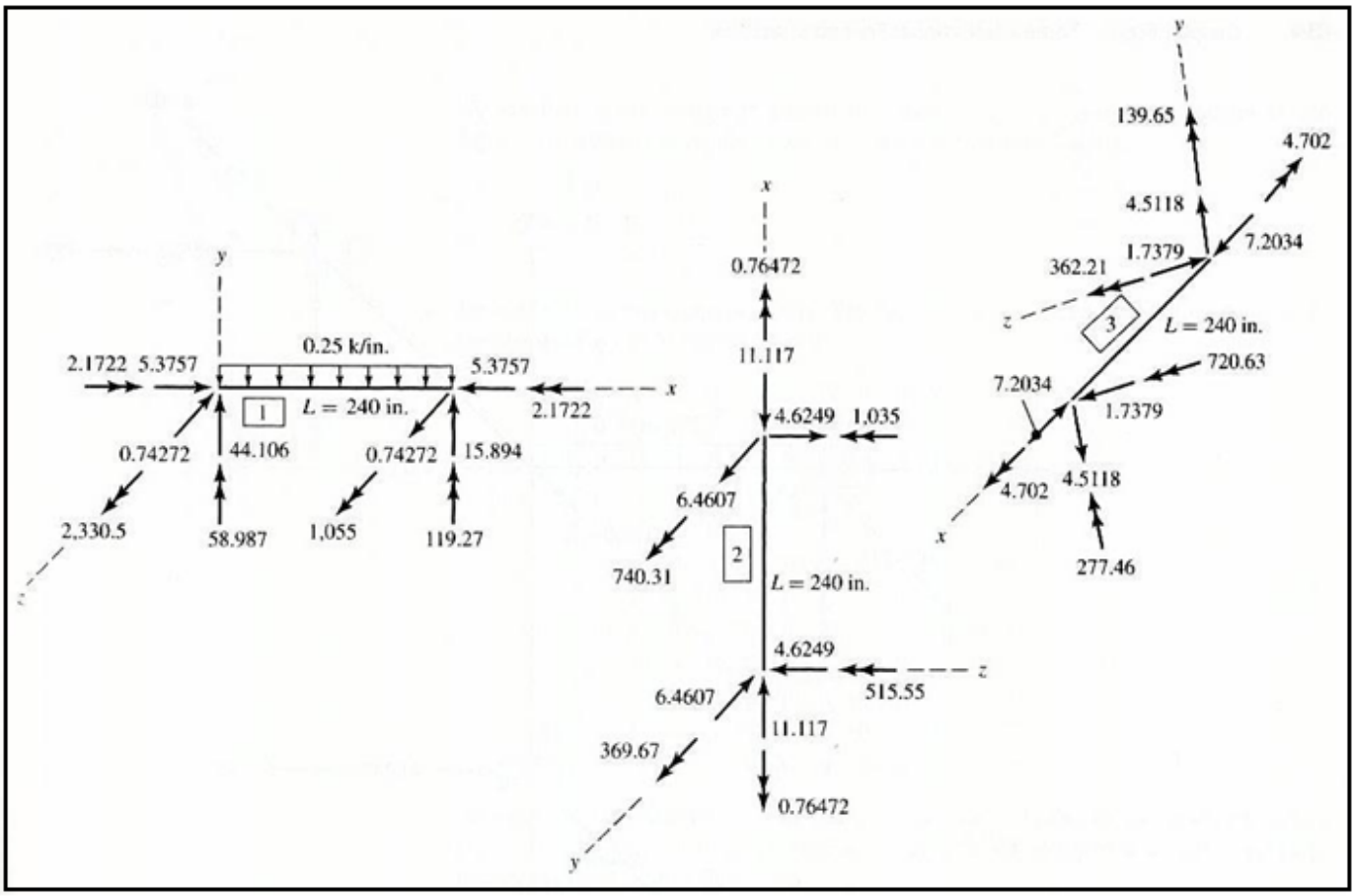


Fig.7 Member Local End Forces [1]

R =

0	5.3757	-4.6249	-0.75082
0	44.106	11.117	4.7763
0	-0.74272	-6.4607	7.2034
0	2.1722	-515.55	-383.5
0	58.987	-0.76472	-60.166
0	2330.5	369.67	-4.702

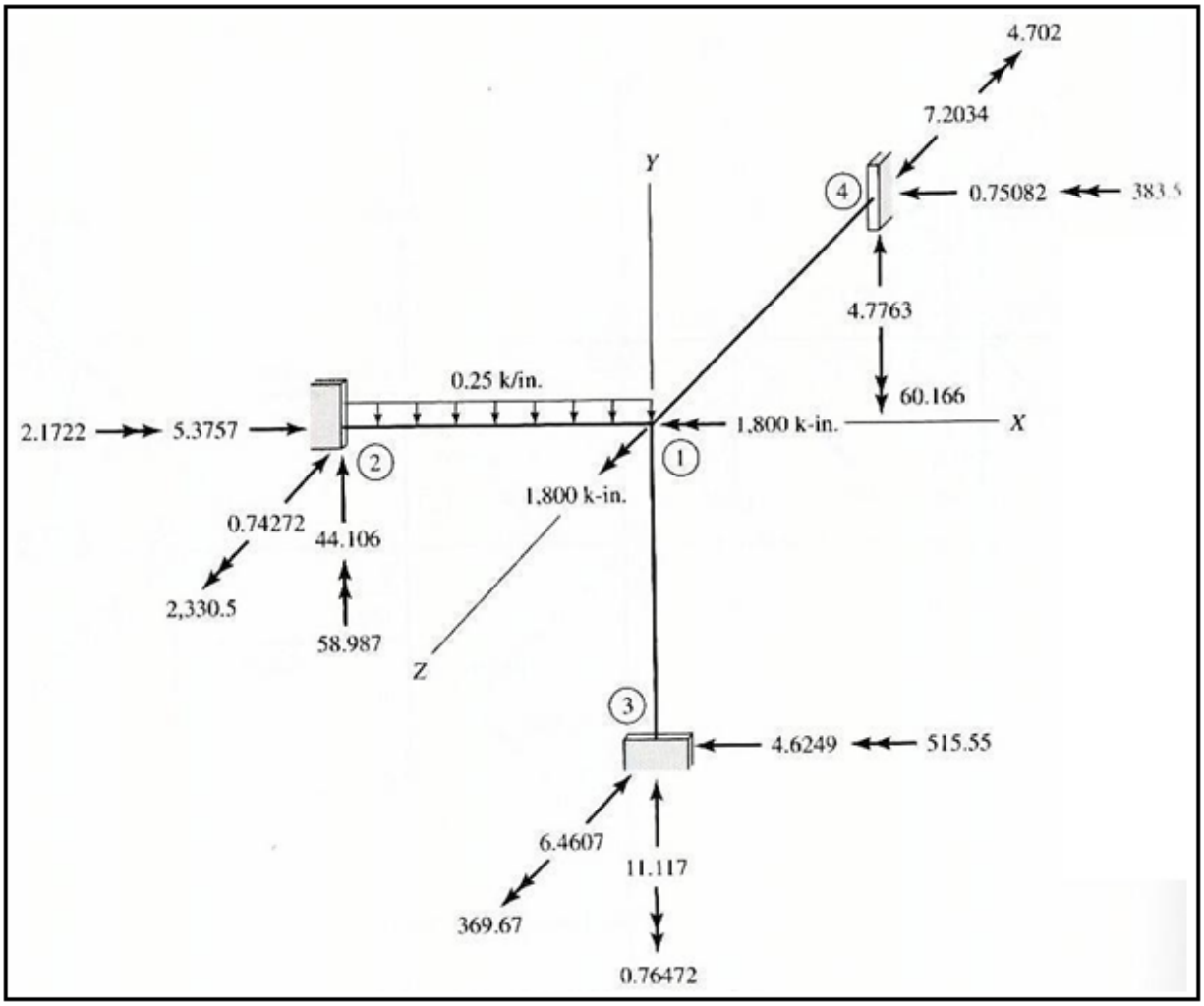
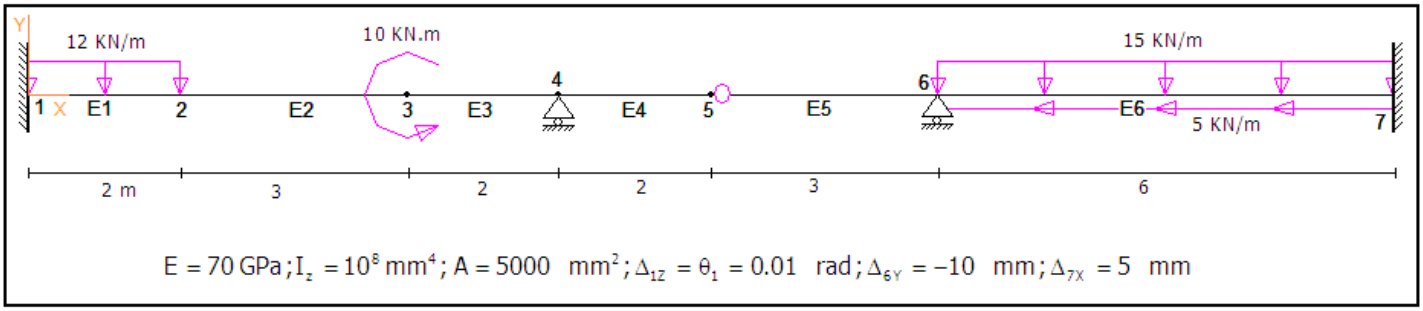


Fig.8 Support Reactions [1]

Other Examples

Example 1 : Beam



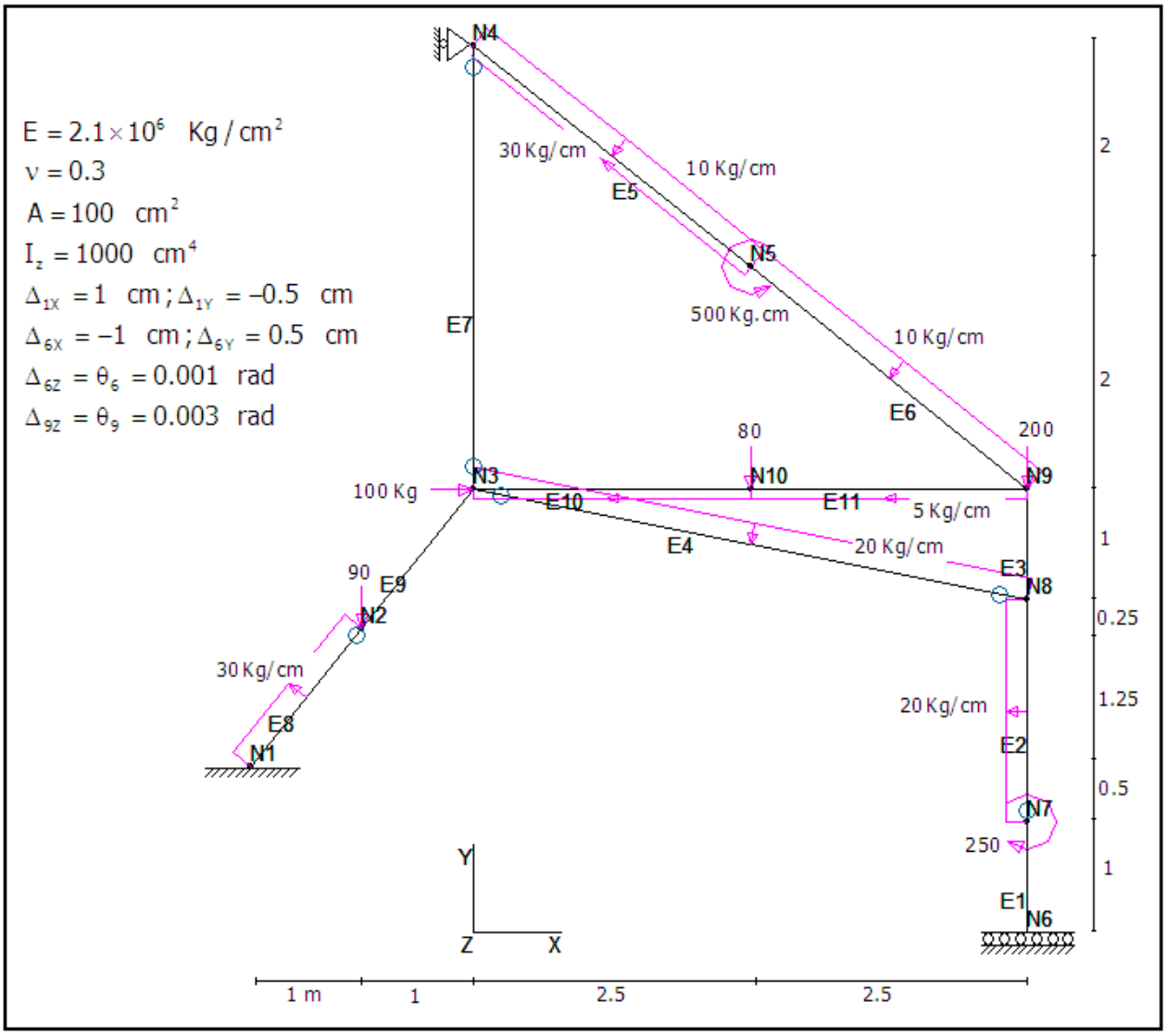
DataBeam.m

```
function D=DataBeam
%  Units: KN & m
m=6;n=7;
Coord=[0;2;5;7;9;12;18];Coord(:,2:3)=0;
Con=[1 2;2 3;3 4;4 5;5 6;6 7];Con(:,3:4)=1;Con(5,3)=0;
Re=ones(n,6);Re(:,[1,2,6])=[1 1 1;0 0 0;0 0 0;0 1 0;0 0 0;0 1 0;1 1 1];
Load=zeros(n,6);Load(3,6)=10;
w=zeros(m,3);w(1,2)=-12;w(6,1:2)=[-5 -15];
E=ones(1,m)*70e6;
nu=0.3;G=E/(2*(1+nu));
A=ones(1,m)*5e-3;
Iz=ones(1,m)*1e-4;
Iy=ones(1,m)*1e-4;
J=ones(1,m)*1e-4;
St=zeros(n,6);St(1,6)=0.01;St(6,2)=-0.01;St(7,1)=0.005;
be=zeros(1,m);
D=struct('m',m,'n',n,'Coord',Coord','Con',Con','Re',Re','Load',Load',...
    'w',w','E',E','G',G','A',A','Iz',Iz','Iy',Iy','J',J','St',St','be',be');
```

MSA.m

```
>> D=DataBeam;[Q,V,R]=MSA(D);
```

Example 2 : 2D Frame



Note: Joint 9 has a specific rotation

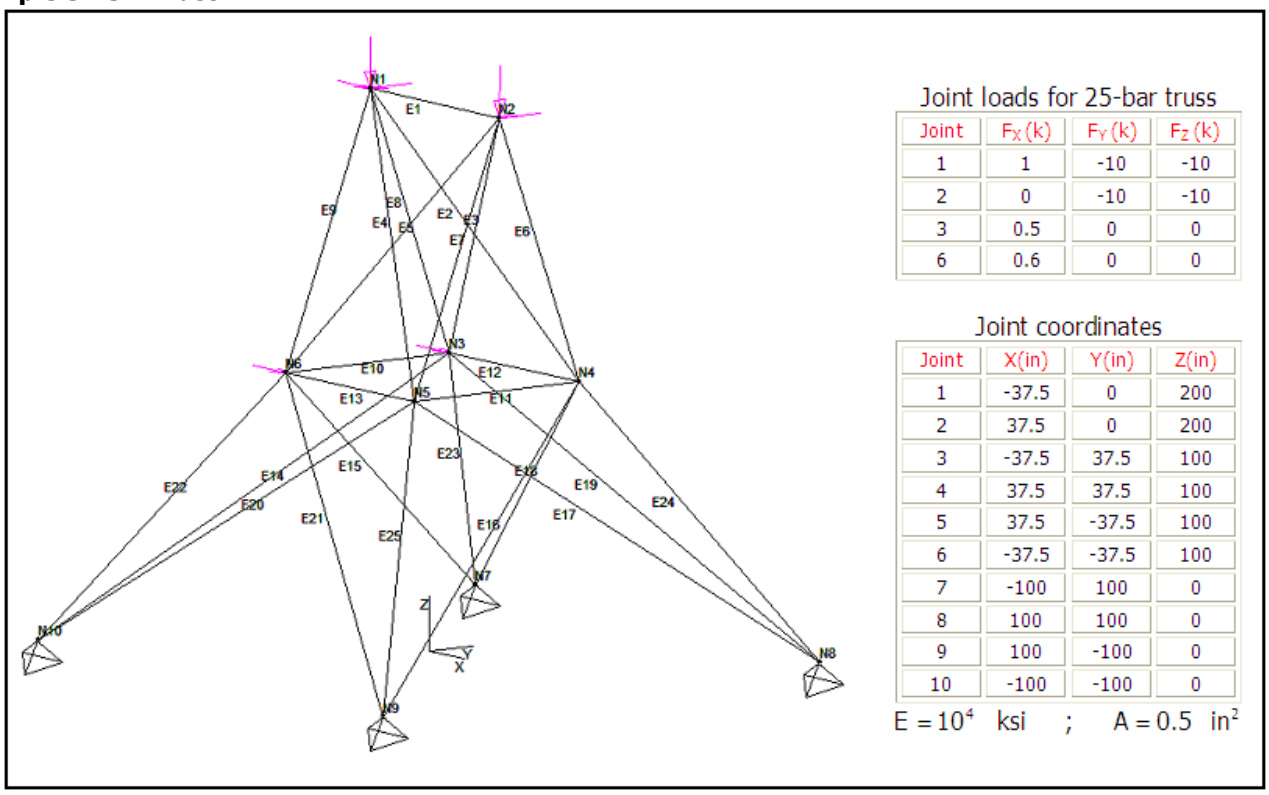
DataF2D.m

```
function D=DataF2D
%  Units: Kg & cm
m=11;n=10;
Coord=[-200 150;-100  275;0  400;0  800;250  600;500  0;500  100;500  300;500  400;250  400];
Coord(:,3)=0;
Con=[6 7 1 1;7 8 0 1;8 9 1 1;3 8 0 0;5 4 1 1;9 5 1 1;3 4 0 0;1 2 1 0;2 3 1 1;3 10 1 1;10 9 1 1];
Re=ones(n,6);Re(:,[1,2,6])=[1 1 1;0 0 0;0 0 0;1 0 0;0 0 0;0 1 1;0 0 0;0 0 0;0 0 0;0 0 0];
Load=zeros(n,6);Load(:,[1,2,6])=[0 0 0;0 -90 0;100 0 0;0 0 0;0 0 500;0 0 0;0 0 -250;0 0 0;...
    0 -200 0;0 -80 0];
w=zeros(m,3);w(:,1:2)=[0 0;0 20;0 0;0 -20;15 -10;0 -10;0 0;0 30;0 0;-5 0;-5 0];
E=ones(1,m)*2.1e6;
nu=0.3;G=E/(2*(1+nu));
A=ones(1,m)*100;
Iz=ones(1,m)*1000;
Iy=ones(1,m)*1000;
J=ones(1,m)*1000;
St=zeros(n,6);St([1,6,9],[1,2,6])=[1 -0.5 0;-1 0.5 0.001;0 0 0.003];
be=zeros(1,m);
D=struct('m',m,'n',n,'Coord',Coord','Con',Con','Re',Re','Load',Load',...
    'w',w','E',E','G',G','A',A','Iz',Iz','Iy',Iy','J',J','St',St','be',be');
```

MSA.m

```
>> D=DataF2D;[Q,V,R]=MSA(D);
```

Example 3 : 3D Truss



Joint loads for 25-bar truss

Joint	F_X (k)	F_Y (k)	F_Z (k)
1	1	-10	-10
2	0	-10	-10
3	0.5	0	0
6	0.6	0	0

Joint coordinates

Joint	X(in)	Y(in)	Z(in)
1	-37.5	0	200
2	37.5	0	200
3	-37.5	37.5	100
4	37.5	37.5	100
5	37.5	-37.5	100
6	-37.5	-37.5	100
7	-100	100	0
8	100	100	0
9	100	-100	0
10	-100	-100	0

$E = 10^4$ ksi ; $A = 0.5$ in^2

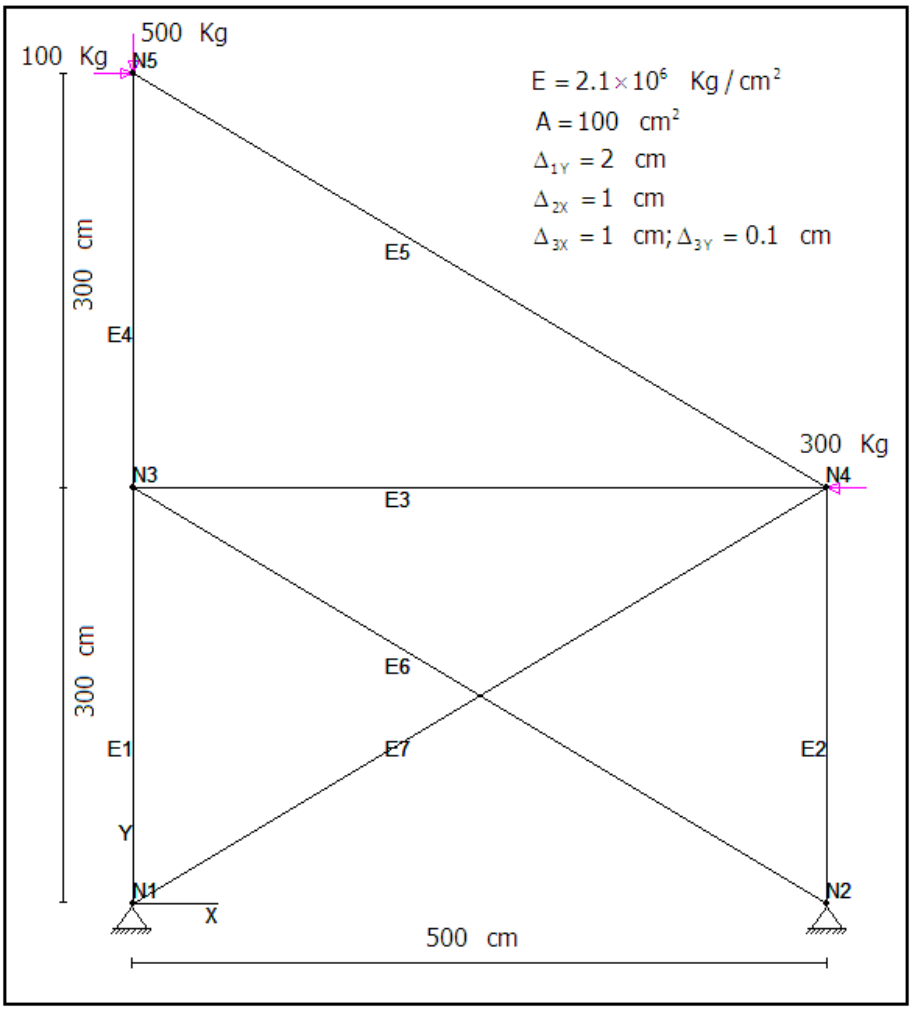
DataT3D.m

```
function D=DataT3D
%  Units: kips & in
m=25;n=10;
Coord=[-37.5 0 200;37.5 0 200;-37.5 37.5 100;37.5 37.5 100;37.5 -37.5 100;-37.5 -37.5 100;...
          -100 100 0;100 100 0;100 -100 0;-100 -100 0];
Con=[1 2;1 4;2 3;1 5;2 6;2 4;2 5;1 3;1 6;3 6;4 5;3 4;5 6;3 10;6 7;4 9;5 8;4 7;3 8;5 10;6 9;...
        6 10;3 7;4 8;5 9];Con(:,3:4)=0;
Re=ones(n,6);Re(1:6,1:3)=zeros(6,3);
Load=zeros(n,6);Load([1,2,3,6],1:3)=[1 -10 -10;0 -10 -10;0.5 0 0;0.6 0 0];
w=zeros(m,3);
E=ones(1,m)*1e4;nu=0.3;G=E/(2*(1+nu));
A=ones(1,m)*0.5;Iz=ones(1,m);Iy=ones(1,m);J=ones(1,m);
St=zeros(n,6);be=zeros(1,m);
D=struct('m',m,'n',n,'Coord',Coord','Con',Con','Re',Re','Load',Load',...
    'w',w','E',E','G',G','A',A','Iz',Iz','Iy',Iy','J',J','St',St','be',be');
```

MSA.m

```
>> D=DataT3D;[Q,V,R]=MSA(D);
```

Example 4 : 2D Truss



Note: Joint 3 has a specific displacement

DataT2D.m

```
function D=DataT2D
%  Units: Kg & cm
m=7;n=5;
Coord=[0 0;500 0;0  300;500  300;0 600];Coord(:,3)=0;
Con=[1 3;2 4;3 4;3 5;5 4;3 2;1 4];Con(:,3:4)=0;
Re=ones(n,6);Re(3:5,1:2)=zeros(3,2);
Load=zeros(n,6);Load(4,1)=-300;Load(5,1:2)=[100 -500];
w=zeros(m,3);
E=ones(1,m)*2.1e6;nu=0.3;G=E/(2*(1+nu));
A=ones(1,m)*100;Iz=ones(1,m)*1000;Iy=ones(1,m)*1000;J=ones(1,m)*50;
St=zeros(n,6);St(1:3,1:2)=[0 2;1 0;1 0.1];
be=zeros(1,m);
D=struct('m',m,'n',n,'Coord',Coord','Con',Con','Re',Re','Load',Load',...
    'w',w','E',E','G',G','A',A','Iz',Iz','Iy',Iy','J',J','St',St','be',be');
```

MSA.m

```
>> D=DataT2D;[Q,V,R]=MSA(D);
```

References

1- A. Kassimali, Matrix Analysis of Structures, Brooks/Cole Publishing Company, 1999.
2- W. McGuire, R.H. Gallagher and R.D. Ziemian, Matrix Structural Analysis, 2nd ed. John Wiley, 2000.
3- R.D. Ziemian, W. McGuire ,MASTAN2, educational analysis software for the 21st century, Proceedings of the 6th International Conference on Computation of Shell and Spatial Structures. IASS-IACM, 2008.

Hossein Rahami
E-Mail: hrahami@ut.ac.ir
Engineering Science Department, College of Engineering, University of Tehran

Personal Profile
Faculty of Engineering - Tehran University – Tehran - Iran

Professional Interests
Structural Mechanics , Optimization , Evolutionary Algorithms , Graph Theory

I would like to express my thanks to Mr. Payam Pezeshki for his careful reviews of the manuscript and for his constructive suggestions.
I welcome comments, suggestions, questions, and corrections that you might wish to offer. Please send your remarks to the following e-mail address hrahami@ut.ac.ir