"AN APPLICATION OF FINITE ELEMENT METHOD
TO THE SIMPLE BEAM PROBLEM"

by

Mark A. Bradshaw

Maritime Systems Engineering

Submitted in Partial Fulfillment of the Requirements of the
University Undergraduate Fellows Program

1983-1984

Approved by:

Tyn-Hsien Chang

Tyn-Hsien Chang

April 1984

ABSTRACT

Deflection is often the dominant design criterion of a flexural member. The theoretical methods commonly used to determine beam deflections become extremely complex for irregular and combined loading conditions. A finite element method is developed for determining the deflection of a simple beam for a variety of loading conditions. The results which are obtained by the method developed are compared with results determined by theoretical methods, and high accuracy is revealed. With relatively minor modification, the numerical method developed could be expanded to consider other structural members, such as cantilevers and beam-columns, or even simple frames and trusses. This potential for development, along with the present capability to consider successive beam cross sections quickly and accurately provide this method with flexibility and efficiency unmatched by standard theoretical methods.

To my partner in life,

my future wife, Shari.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

TABLE OF ILLUSTRATIONS

1. Introduction

1.1 Background

In the design of a beam, the deflection can be a major factor. Depending on the intended use of the beam, construction codes commonly limit the deflection to a fraction of the span length. For example, the American Institute of Steel Construction (AISC) permits a maximum deflection of 1/360 of the span length for beams and girders supporting plastered ceilings, and the American Association of State Highway and Traffic Officials (AASHTO) recommends a maximum deflection of 1/1000 of the span length, for bridges and in urban areas. In such cases, the determination of the deflection becomes paramount to the structural design in addition to the stress analysis.

1.2 Purpose of Research

Solving the differential equations, obtained from classical beam theory, for deflection can be very involved when the applied loadings are complex. Several other methods such as moment area, conjugate beam, and virtual work can also be used to compute the deflection. However, solution by any of these methods is manageable for simple loading conditions, but becomes increasingly difficult with the combination of two or more types of loading conditions. Hence, an alternative method for determining the deflection under any given loading is needed. With the advent of the computer, a numerical method appears to be very efficient for the deflection problem.

The purpose of this research is to develop a finite element method for the determination of the deflection in a beam under a variety of loading conditions. The computed deflections and slopes determined by this method are compared with those from other methods. Furthermore, because of the shape functions, the deflection curve can be displayed on the computer terminal. Thus, the location where maximum deflection occurs can be easily identified. Any excess deflection can be corrected immediately through the selection of a new cross section. The development of this method bears in mind the effects of boundary conditions, moments of inertia, span length, elastic properties, and loading conditions on the deflections of a simple beam.

1.3 Intended Audience

This report is directed to the reader who has a basic understanding of engineering mechanics. For those who are unfamiliar with the classical beam theory, the governing differential equations are derived in breif. The developed finite element method is presented in step form.

2. Classical Beam Theory

2.1 Derivation

In this section the classical relationship between bending moments and the deflection is derived. This is intended as a brief introduction to classical beam theory. More detailed derivations are available in most textbooks on the strengths of materials. The governing assumptions are as follows:

(1) Plane sections of the beam, originally plane, remain plane.
(2) The material in the beam is homogenous and elastic.
(3) The moduli of elasticity for tension and compression are equal.
(4) The beam is originally straight and of constant cross section.
(5) The plane of loading must contain a principal axis of the beam's cross section and the loads must be perpendicular to the longitudinal axis of the beam.

Consider the following beam:

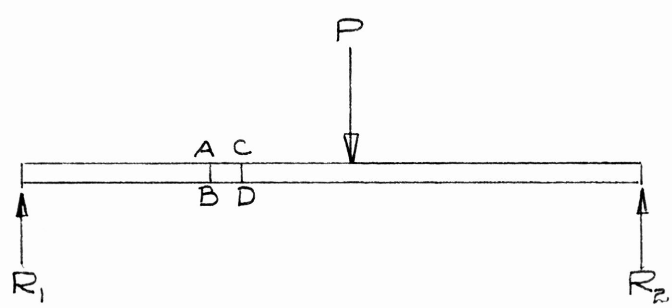


Figure 1: Classical Beam

When the load is applied, the beam bends downward causing sections A-B and C-D to rotate, relative to each other, by the amount $d\theta$ as shown in Fig. 2.

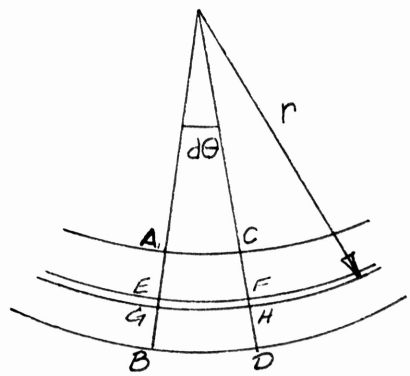


Figure 2: Deflected Beam for
Classical Theory

The top fiber (AC) shortens and the bottom fiber (BC) lengthens, while somewhere in between there exists a fiber whose length does not change. The line connecting this fiber, (EF), is defined as the neutral axis. The strain of the arbitrary section (GH) can therefore be expressed as:

(EQ 2.1) $$e = \frac{y\,d\theta}{ef} = \frac{y\,d\theta}{r\,d\theta} = \frac{y}{r}$$

And by Hooke's Law, the stress is:

(EQ 2.2) $$s = Ee = \frac{Ey}{r}$$

The bending moment must be balanced by a resistive moment.

(EQ 2.3) $$M = \int y(s_x da)$$

Therefore, through substitution EQ 2.2 becomes:

(EQ 2.4) $$M = \frac{E}{r}\int y^2 da = \frac{EI}{r}$$

Since the resulting deflection curve is very flat, its slope is very small and can be neglected. Thus, the curvature can be expressed in terms of the second derivative of the deflection, that is:

$$\frac{1}{r} = \frac{d^2y/dx^2}{[1+(dy/dx)^2]^{3/2}} \triangleq \frac{d^2y}{dx^2} \qquad \text{(EQ 2.5)}$$

On substituting EQ 2.5 into EQ 2.4 one has:

$$EI\,\frac{d^2y}{dx^2} = M \qquad \text{(EQ 2.6)}$$

This is the basic relation between applied bending moment and the flexure induced.

2.2 Governing Equations for Different Loadings

2.2.1 Concentrated Loading Conditions

For concentrated loadings the resulting moments must be found before EQ 2.6 can be solved. For example:

(a) Load Diagram

(b) Moment Diagram

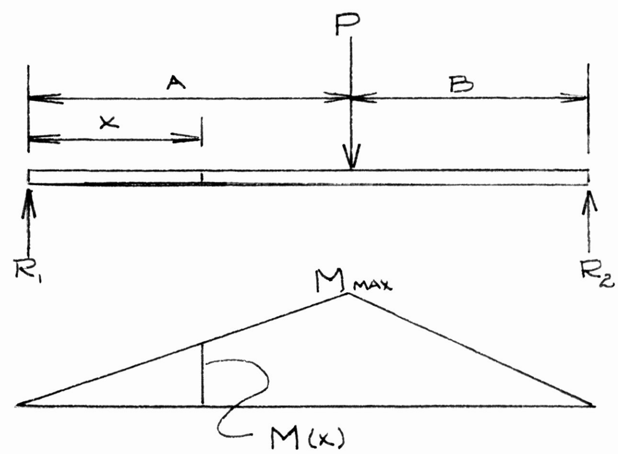


Figure 3: Concentrated Load for Classical Theory

Therefore, if the deflection is desired at some arbitrary point C, which is a distance x from the left end of the beam, the moment at that point must first be determined. Then EQ 2.6 can be solved for Y through double integration.

2.2.2 Distributed Loading Conditions

For distributed loading conditions, again the resulting moment must be determined before equation (2.6) is applied. For irregular loading conditions, however, the problem grows more complex. The governing equation becomes the fourth order ordinary differential equation:

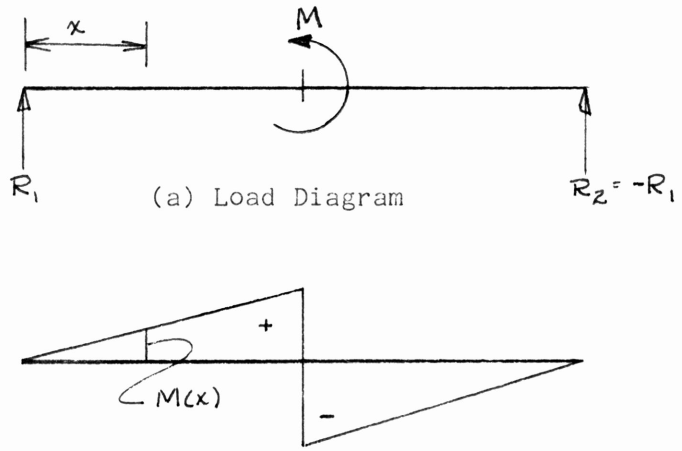
$$EI \frac{d^4 y}{dx^4} = p(x) \qquad \text{(EQ 2.7)}$$

where: P(X) = Load Function
EI = Flex. Rigidity
y = Deflection
x = Distance along the beam

Therefore, to obtain the deflection at any point, the above equation must be solved for y through four integrations.

2.2.3 Moment Loading Conditions

For moments applied directly to the beam, EQ 2.6 can be used with relative ease. Again, the resultant moment, which occurs at the arbitrary distance x, must first be determined.



(a) Load Diagram

(b) Moment Diagram

Figure 4: Moment Load for
Classical Theory

The deflection y, for the given location, can then be found through double integration of EQ (2.6).

2.2.4 Combined Loading Conditions

When a combination of these loading conditions (i.e. concentrated load, distributed load, and moment) is applied to a simple beam, the solution is often obtained by the method of superposition. In this method the deflection curve due to each loading condition is computed separately according to the applicable governing equation, and then the curves are added to obtain the resultant deflection curve for the combined loading condition.

3. Finite Element Method of Solution

3.1 Introduction

The formulation of the Finite Element method describing a simple beam in flexure consists of the following seven steps:

(1) Discretization of the beam.

(2) Approximation of the deflection curve.

(3) Derivation of strain-displacement-stress relationships.

(4) Determination of element equations.

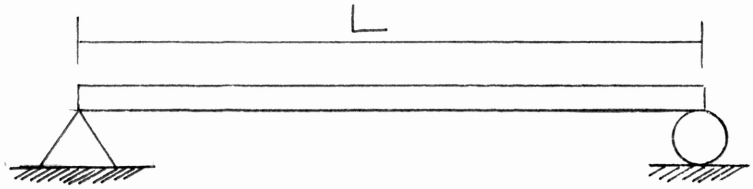
(5) Determination of the global equation.

(7) Determination of maximum strain and stress.

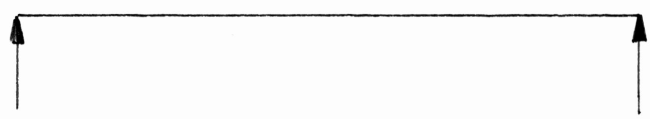
It should be noted that the assumptions for classical beam theory previously stated are observed. In the sections which follow, the steps above will be explained in greater detail.

3.2 Discretization of the Beam

The actual three dimensional beam is modeled by a one dimensional idealized beam. This idealized beam is then discretized into several line elements, as shown below in Fig. 5.



(a)



(b)

(c)

Figure 5: Discretized Beam

Notice that the element lengths need not be uniform. This allows greater flexibility in the placement of the nodal points. The local coordinate system for any element is shown below in Fig. 6.

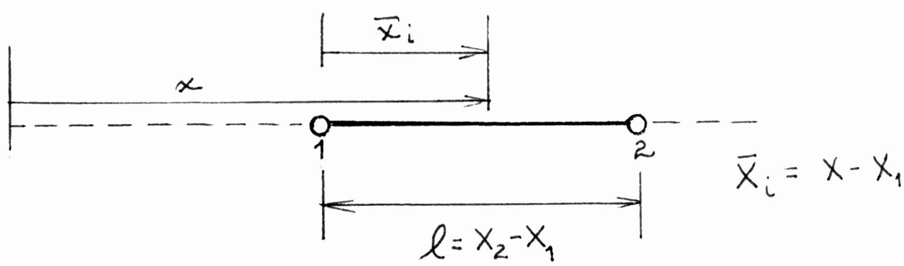


Figure 6: Random Element

3.3 Approximation of the Deflection Curve

Due to the properties of the material and the other assumptions of classical beam theory, the deflection curve must be smooth and continuous. Consider the loaded beam in Fig. 7(a). After loading, the beam will tend to bend downward as shown in Fig. 7(b).

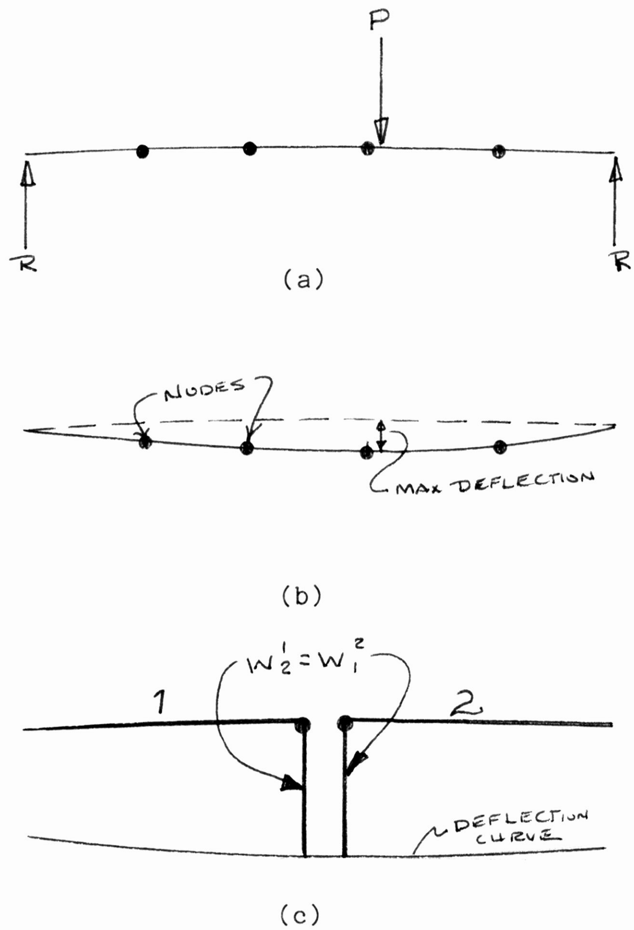


Figure 7: Deflected Beam for Finite Element Method

To ensure the continuity of the deflection curve, the adjacent nodes of neighboring elements must have the same slope and deflection as shown in Fig. 7(c). The deflection curve for any element can be approximated by shape functions in terms of nodal values.

Accordingly,

(EQ 3.3.1) $$W(x) = N_1 W_1 + N_2 \theta_1 + N_3 W_2 + N_4 \theta_2$$

$$W(x) = [N] \{q\}$$

Where the N_i functions are known as Hermitian functions, which are:

$$N_1 = 1 - 3(S)^2 + 2(S)^3$$

$$N_2 = L(S)(1 - 2(S) + S^2)$$

$$N_3 = S^2 (3-2(S))$$

$$N_4 = L(S)^2 (S-1)$$

where: $S = \frac{(x - x_i)}{L}$ (local coordinate)

in which:

x = global coordinates of any point
x_i = global coordinate of node (i)
L = length of element

This particular interpolation approximation model for deflection W(x) in local element coordinates, is quite adequate for the purposes of this research.

3.4 Derivation of Strain-Displacement and Strain-Stress Relationships

As shown previously, the strain-displacement relationship for any random point in the beam is:

(EQ 3.4.1) $$c(x,y) = \frac{dv}{dx} = -y\frac{d\ W}{dx} = -y\ W''$$

where:
v = axial displacement
y = vertical distance from neutral axis
W = deflection of beam

The EQ 3.3.1 is differentiated twice to obtain:

$$W''(x) = \frac{1}{L}\frac{d}{dS}[N]$$

$$W''(x) = L\,[B]\,\{q\} \qquad \text{(EQ 3.4.2)}$$

where [B] = transformation matrix whose coefficients are obtained through differentiation.

3.5 Determination of Element Equations

The principle of minimum potential energy is used to derive the element equations. The potential energy for any beam element experiencing distributed, concentrated, and moment loading conditions, is expressed as:

(EQ 3.5.1)

$$II = \int_{x_1}^{x_2} (0.5)F(W'')^2 - \int_{x_1}^{x_2} pW\,dx - \sum_{i=1}^{k} P_i W - \sum_{j=1}^{h} M_j W'$$

Since $\frac{dII}{dq} = 0$, differentiation of EQ 3.5.1 yeilds:

$$[k]\,\{q\} = \{Q\} \qquad \text{(EQ 3.5.2)}$$

where: $[k] = FL\int_0^1 [B]^T[B]\,dS$ (local stiffness)

$$\{Q\} = L\int [N]^T p(S)\,dS + \sum P_i\,[N(S_i)]^T + \sum Mj\,[n(S_j)]^T$$

and p(S) is found by linear interpolation of nodal load intensities (i.e.).

$$p(S) = (1-S)p_i + (S)p_{i+1}$$

It should be noted that EQ 3.5.3 allows for several applications of moment or concentrated loadings per element. For example, the following element configuration could easily be analyzed.

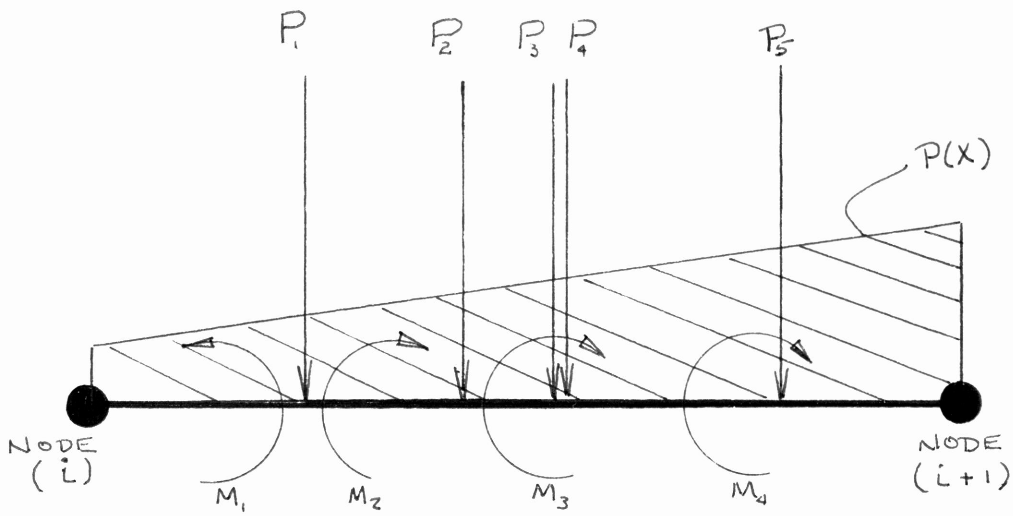


Figure 8: Arbitrary Element (i)

3.6 Determination of Global Equations

Similarly, for the global coordinate system, the deflection is related to the applied load and the internal stiffness of the beam by:

(EQ 3.6.1) $[K]\{q\} = \{Q\}$

where: $[K]$ = global stiffness matrix (square)

$\{q\}$ = global displacement matrix (column)

$\{Q\}$ = global load matrix (column)

The global stiffness and load matrices are determined by superimposing the local stiffness matrices. However, care must be taken to ensure interelement compatibility.

3.7 Determination of Deflection

After the global stiffness and global load matrices have been found then the global displacements may be determined. One of the most common methods for the solution of a linear equation is matrix inversion.

(EQ 3.7.1) $\{q\} = [K]^{-1}\{Q\}$

The Gauss-Jordan elimination method can be used to invert the global stiffness matrix. Then the displacement matrix {q} can be found directly. The boundary conditions for a particular beam configuration may limit the translation or rotation at the ends, thereby reducing the number of unknowns and simplifying the governing equations. Hence, a considerable amount of computer time and memory can be saved if the given boundary conditions are imposed before the matrix inversion.

3.8 Determination of Strain and Stress

Once the slopes and deflections at the nodal points are known, then the Hermitian functions can be used to approximate the slopes and deflections at several points between the nodes, as discussed in section 3.3. Using this method, one can easily determine the location of the maximum deflection in each element. Once the location and magnitude of the maximum deflection is known, EQ 3.4.1 and EQ 2.2 may be used to determine the strain and stress at that point. Therefore, by utilizing this method, the maximum deflection, strain, and stress for the beam considered can be determined for any loading condition.

4. Results and Discussions

4.1 Loading Conditions

The following general types of loads were considered:

1. Distributed
2. Concentrated
3. Moment

As mentioned previously, for simple loadings the solution can be obtained with little difficulty using ordinary methods. But as the loading condition grows more complex, the advantage of numerical method becomes more evident. This section will discuss several particular loads and their combined as well as their individually induced deflections.

4.1.1 Distributed Loading

In the design of a structure, beams must commonly support distributed loads. For example, the dead weight of a concrete slab, roofing material, or machinery is not unusual. A uniformly distributed load of small magnitude, as shown in Fig. 9, can result in a relatively large deflection.

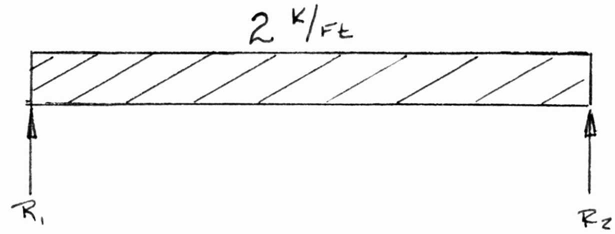


Figure 9: Uniformly Distributed Load

The distributed load for which the deflection curve was evaluated is shown in Fig. 10. The beam's dimensions and properties are also noted. The deflection curve which was obtained using the maximum possible number of points, is shown in Fig. 11. The maximum stress, along with its location on the beam, are also given with the computed curve.

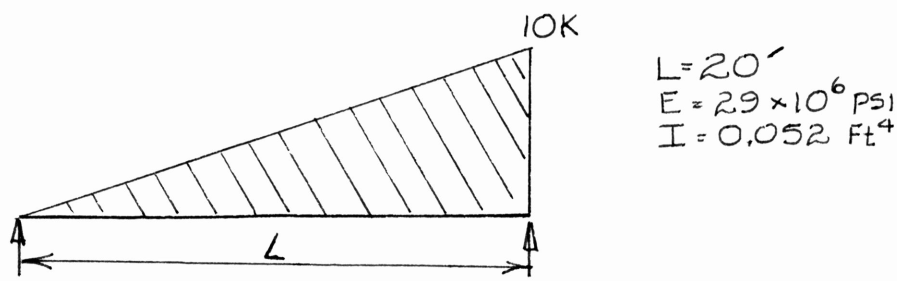


Figure 10: Distributed Load Considered

These results compare favorably with those obtained by other methods. The relative difference is only 0.26 percent, which can be explained by rounding errors commonly incurred during the manual calculations associated with theoretical solutions.

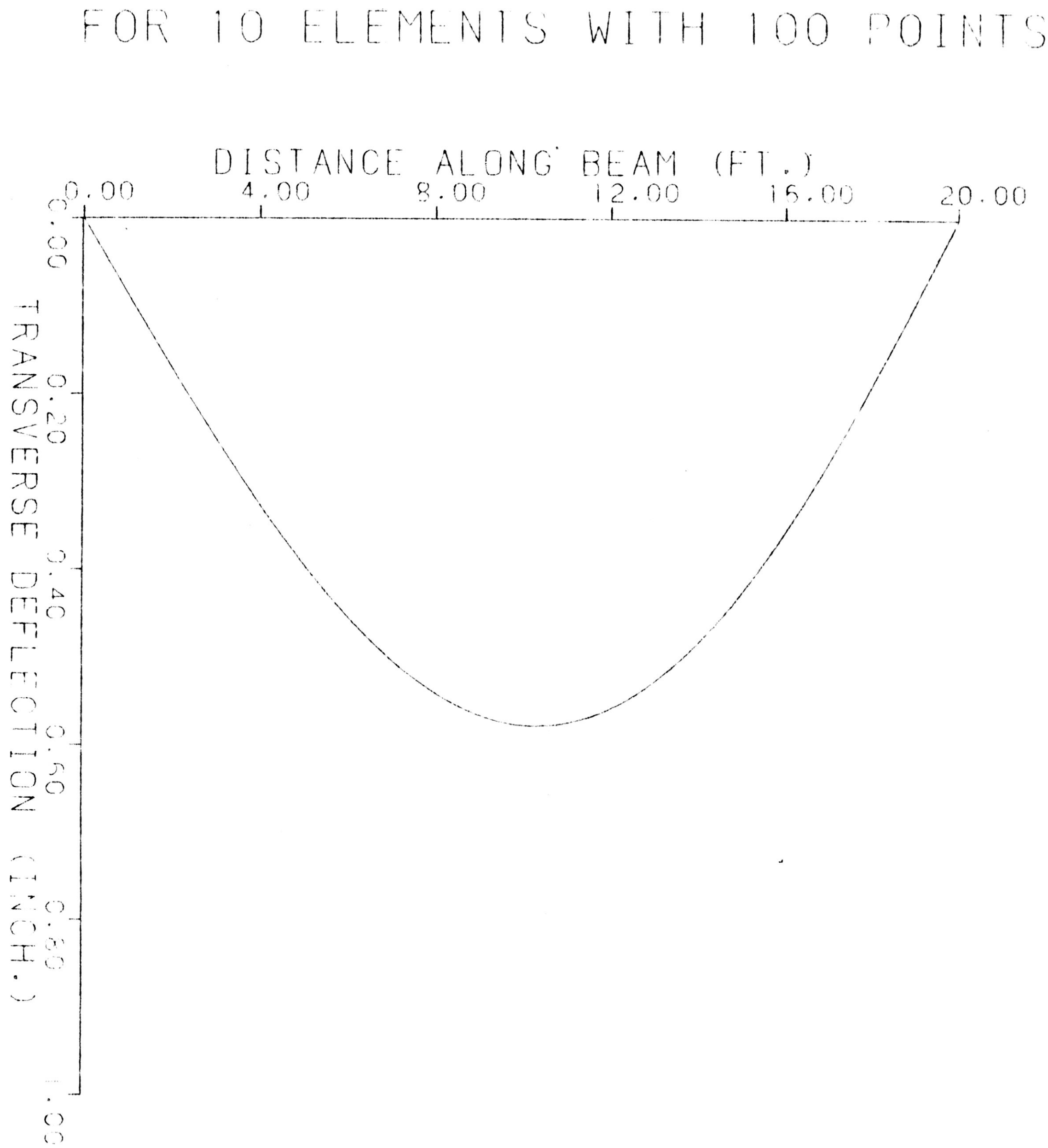
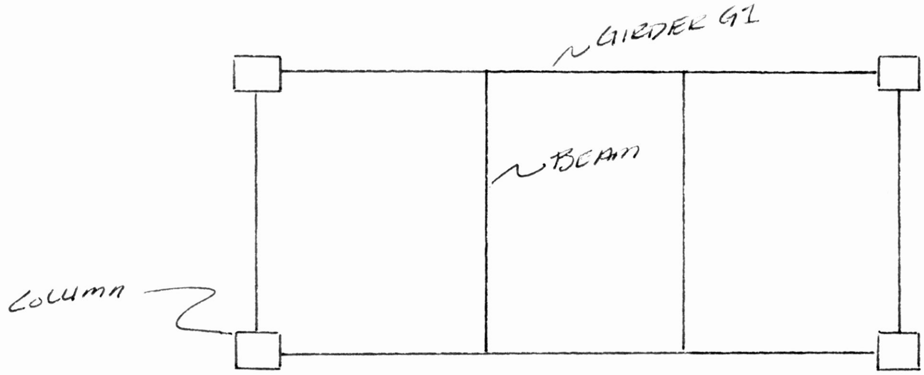

FOR 10 ELEMENTS WITH 100 POINTS
DISTANCE ALONG BEAM (FT.)
0.00
4.00
8.00
12.00
16.00
20.00
TRANSVERSE DEFLECTION (INCH.)
0.00
0.20
0.40
0.60
0.80

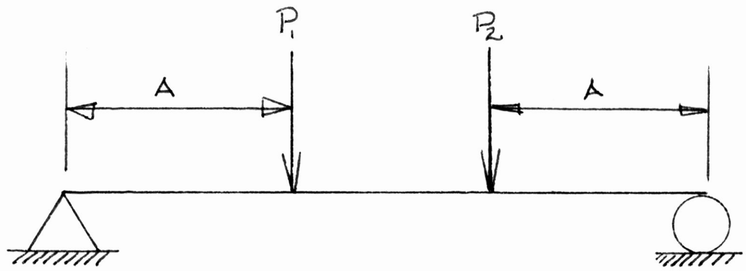
Figure 11

4.1.2 Concentrated Loading

Frequently a structure is designed such that the secondary members, which support the floor, are supported by primary members, such as the girders illustrated in Fig. 12.



(a)



(b)

Figure 12: Beams Supported By Girders

Therefore, this concentrated loading condition was selected for analysis. The actual beam and loads used are given in Fig. 13. The resulting deflection is plotted in Fig. 14.

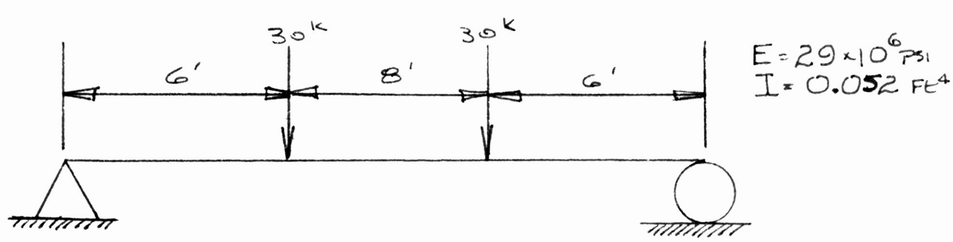


Figure 13: Concentrated Loading Considered

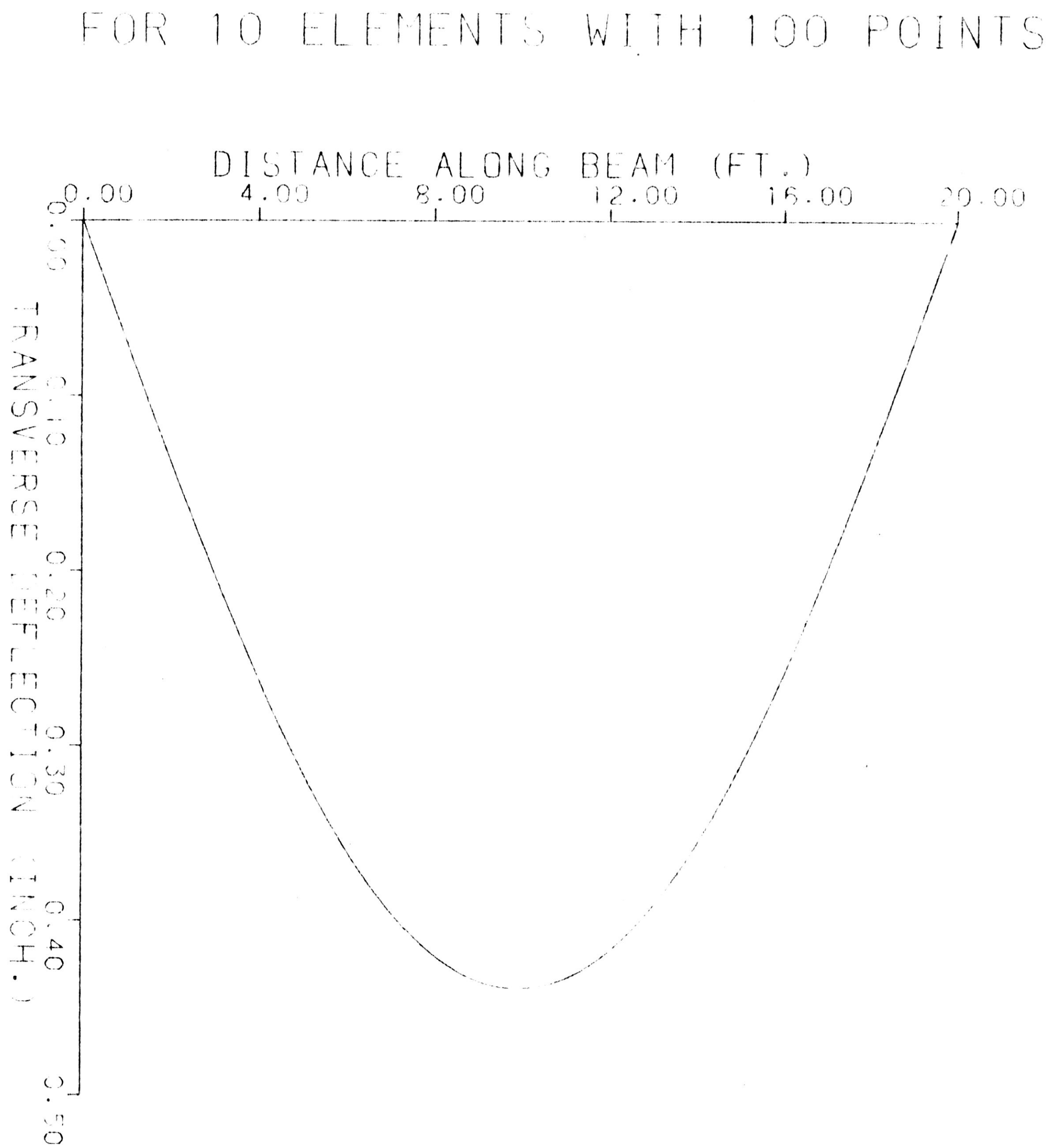

FOR 10 ELEMENTS WITH 100 POINTS
DISTANCE ALONG BEAM (FT.)
0.00
4.00
8.00
12.00
16.00
20.00
TRANSVERSE DEFLECTION (INCH.)
0.00
0.10
0.20
0.30
0.40
0.50

Figure 14

These results are very encouraging. The relative difference between the theoretical and numerical methods for computing maximum deflections was 0.10 percent.

4.1.3 Moment Loadings

The magnitude and direction of the applied moment was chosen arbitrarily. The selected moment and beam configuration is shown in Fig. 15, and the resulting deflection is shown in Fig. 16.

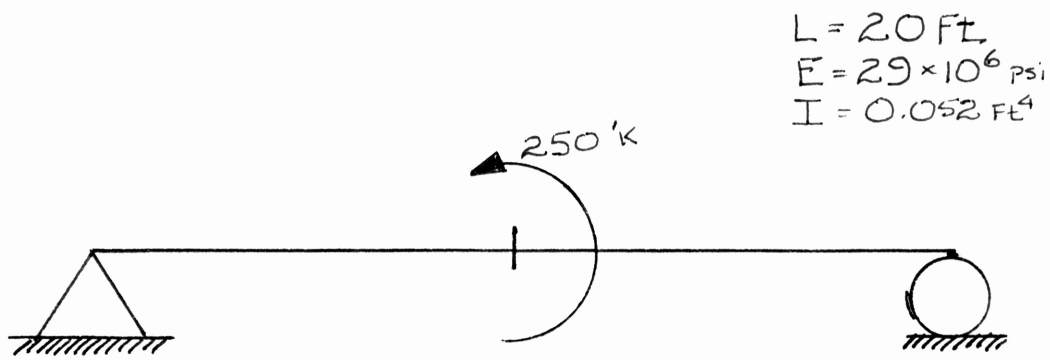


Figure 15: Moment Loading Considered

The relative difference between the results obtained using each of these methods was less than 0.84 percent. This is exceptional, considering the variations of the theoretical and finite element methods.

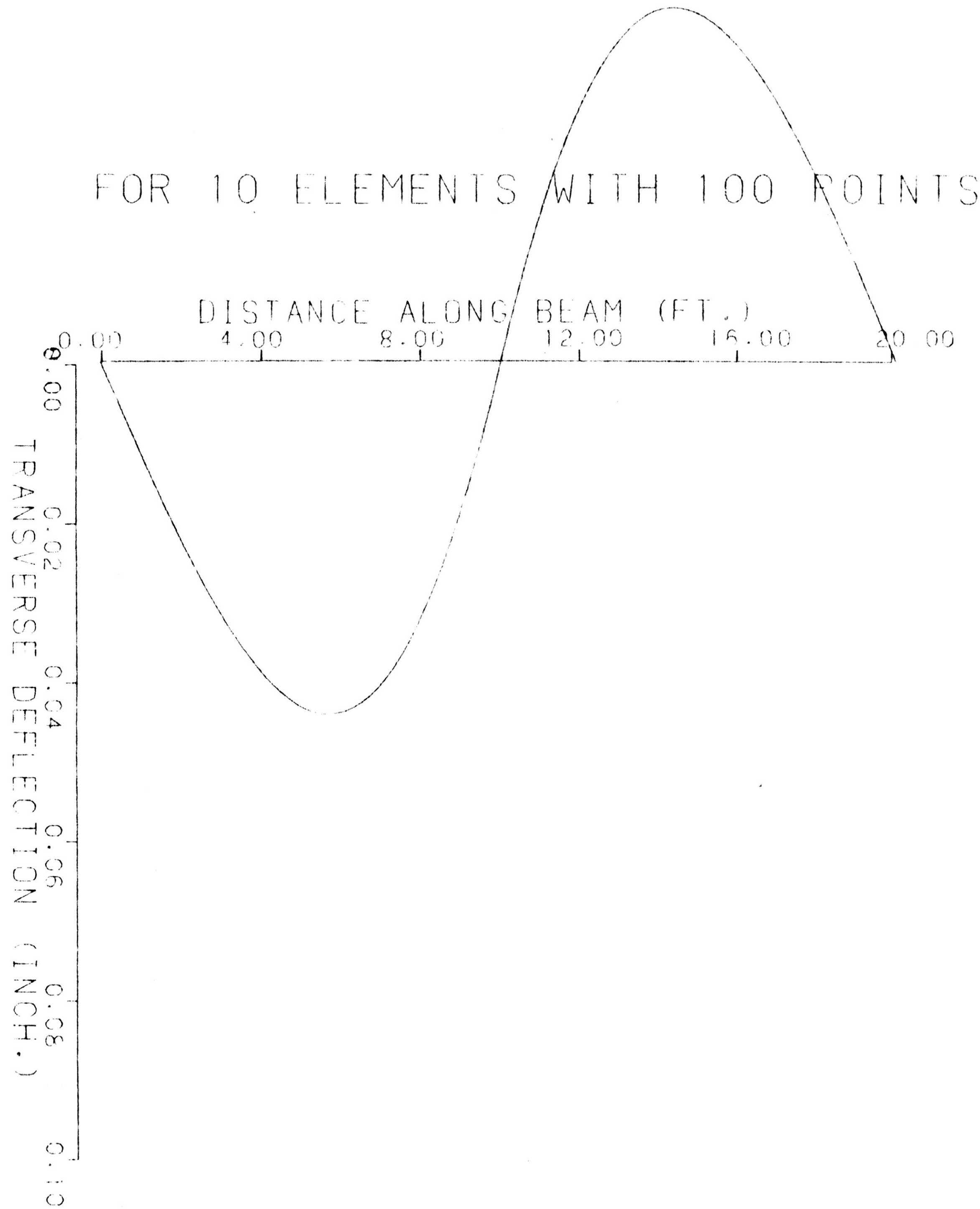

FOR 10 ELEMENTS WITH 100 POINTS
DISTANCE ALONG BEAM (FT.)
0.00
4.00
8.00
12.00
16.00
20.00
0.00
0.02
0.04
0.06
0.08
0.10
TRANSVERSE DEFLECTION (INCH.)

Figure 16

4.1.4 Combined Loadings

One of the most advantageous features of the numerical method which was developed during this research is its ability to consider multiple loading conditions. The program developed is designed to accomodate a maximum of 10 elements, each with a maximum of 100 concentrated loads and 100 applied moments. In addition, a coincident irregularly distributed load can also be applied. Fig. 17 indicates the flexibility of the method developed.

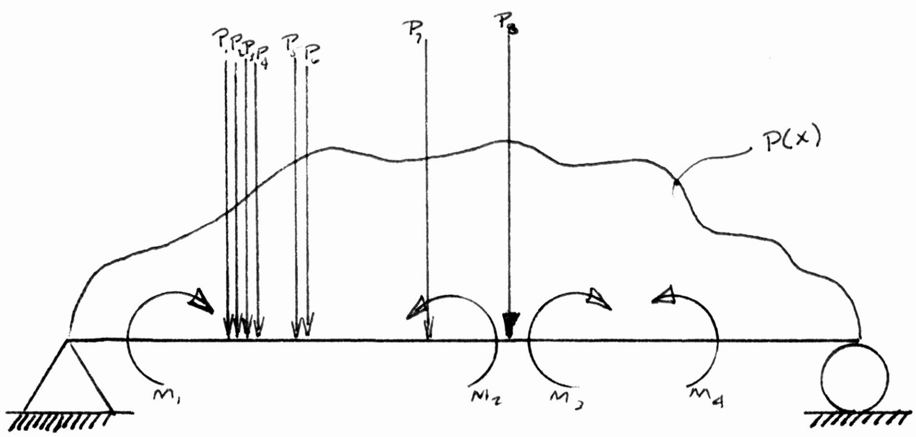


Figure 17: A Possible Combination of Loads and Moments

The principles used to evaluate simple loading conditions are the same for complex loading conditions. It is reasonable to assume that if numerical methods provide reliable results for simpler cases, then it should also comply for the maximum in complexity for which it was designed. See Appendix I.

The previous problems illustrated the reliability of the numerical method developed for each individual loading condition. Now, consider these three separate loading conditions applied simultaneously to the same beam as shown in Fig. 18. The resulting deflection is plotted in Fig. 19.

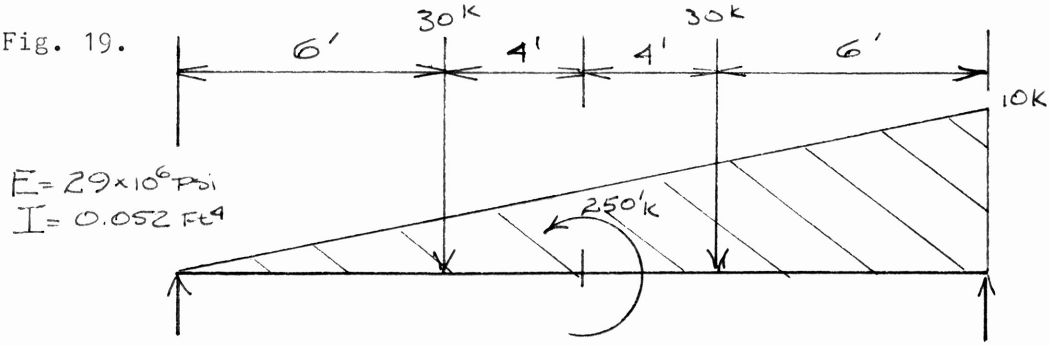


Figure 18: Combined Loadings Considered

From a brief inspection of Fig. 19, it can be seen that the deflection curve for the combined loadings is roughly equal to the summation of the deflection curves for each of the individual loading conditions. Comparison with theoretical methods reveals an extremly high accuracy in the deflection values, (less than 0.40 percent). Therefore, the numerical method developed provides reliable results for independent as well as combined loadings.

4.2 Effects of Various Functions on Accuracy

The major factors which influence the accuracy results are:

1. Load Approximation
2. Shape Functions
3. Beam Discretization

The effect on each of these and the particular application selected will be discussed in this section. For more detail about the actual functions the reader is referred to section 3.

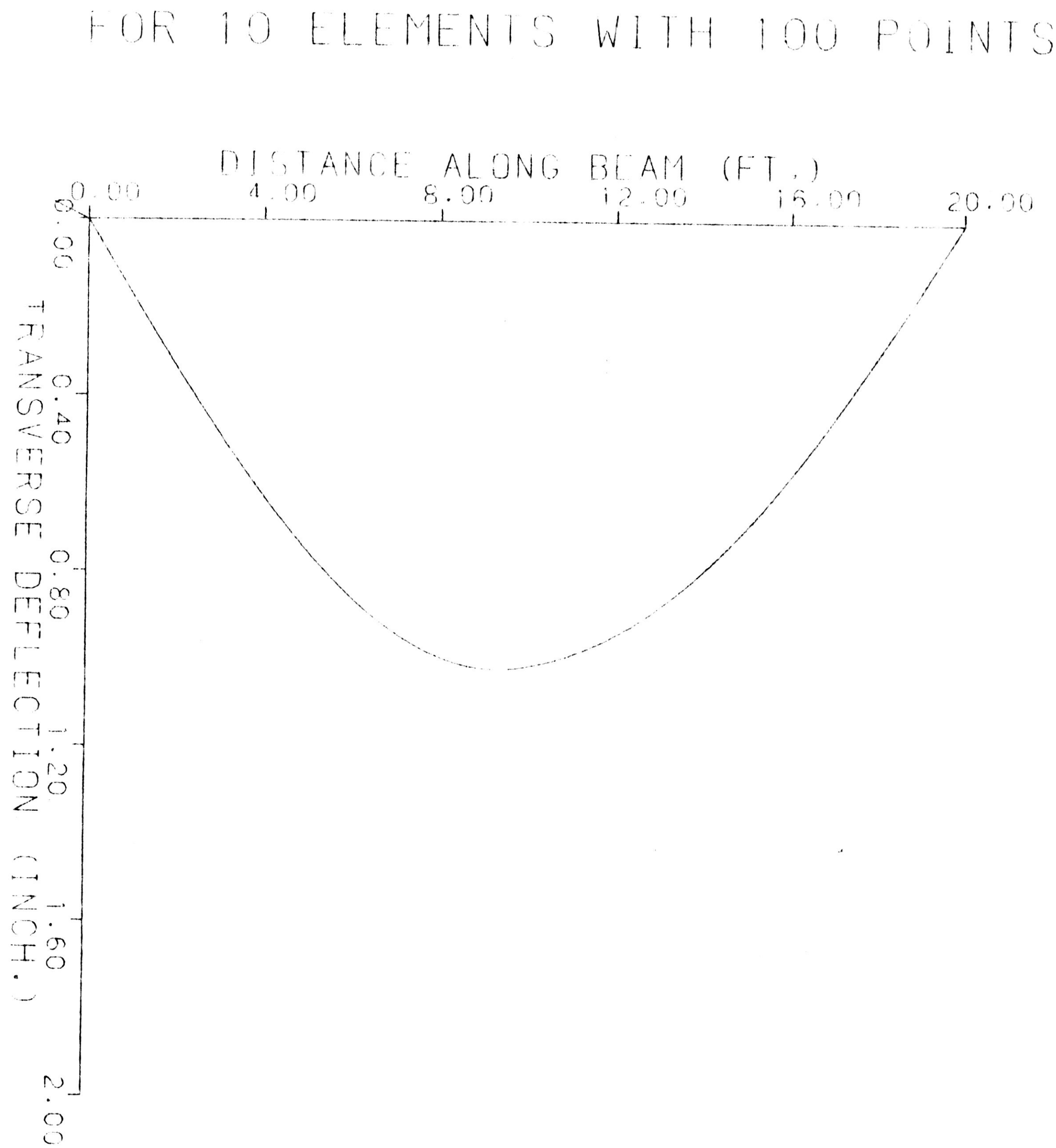
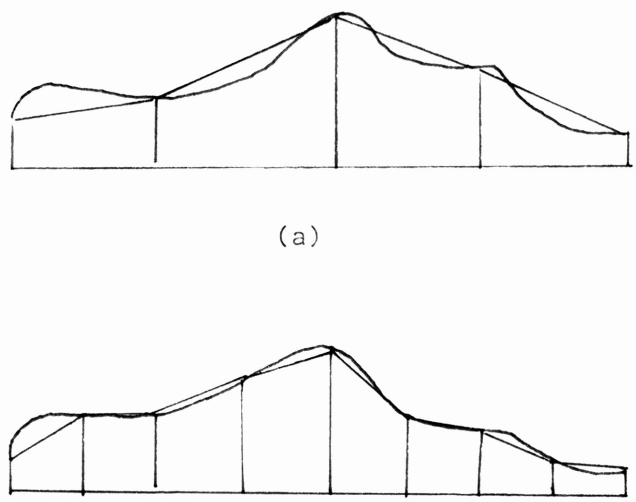

FOR 10 ELEMENTS WITH 100 POINTS
DISTANCE ALONG BEAM (FT.)
0.00
4.00
8.00
12.00
16.00
20.00
TRANSVERSE DEFLECTION (INCH.)
0.00
0.40
0.80
1.20
1.60
2.00

Figure 19

4.2.1 Load Approximation

In order to consider irregularly distributed loading functions, some general method of approximation is required. In the method developed, a linear approximation was used to describe the load function between nodal points as illustrated in Fig. 20.



(a)

(b)

Figure 20: Linear Approximation of Load Function

Although higher order polynomial approximations are available, they are not always necessary. For this research, the accuracy obtained using linear interpolation was sufficient, when the maximum number of nodal points were considered.

4.2.2 Shape Functions

Frequently, the maximum deflection may occur between nodal points. Shape functions, such as those given in section 3.3, can be used to generate the deflection at any point on an element, but their accuracy is very dependent on the accuracy of the nodal values of that element. In the method developed, the shape functions were used to determine the deflection at 10 points per element. Consequently, a very smooth deflected curve is obtained when plotted. This allows the maximum deflection and point of occurence, to be read directly from the curve whether it was plotted on paper or a computer terminal.

4.2.3 Beam Discretization

The size of the element considered affects the deflection directly, as described by element equations section 3.5, and indirectly, through load approximation and shape functions. By discretizing the beam into more elements of smaller lengths, the accuracy of the results can be improved. Hence, the highest accuracy is one obtained using the maximum possible nodal points.

5. Concluding Remarks

5.1 Summary of Results

As demonstrated by the results, the method developed is a viable alternative method for determination of the deflection of a simple beam for various loading conditions. The accuracy of the results obtained was quite acceptable for engineering analysis, and the ease with which several beam cross sections could be compared should prove invaluable to the overall design process.

5.2 Further Development

In this research, however, there is a potential for additional development. The method developed was designed so that it could be modified to consider more complex beams. For example, by increasing the types of boundary conditions which can be imposed, a fixed-end condition can be placed on the beam and axial loads can be considered. And, by defining the nodal coordinates in two or three dimensions, a simple frame or truss can be analyzed. In this manner, the capabilities of the method developed in this research can be dramatically increased without affecting its' basic structure.

APPENDIX I

The computer program developed is written in FORTRAN, however, to use it no knowledge of computer programming is needed. It was designed and tested using the PRIME 550 computer system at Texas A&M University at Galveston. The following data must be supplied:

Moment of Inertia
Modulus of Elasticity
Beam Length
Number of Elements
Coordinates of Nodes
Magnitude and Placement of Loads

Any system of units can be used, as long as consistency is maintained throughout the data. For example, if the nodal coordinates are in feet, then the moment of inertia must also be in feet. The maximum number of elements which can be evaluated by this program is 10. This program is applicable only for simple beams. A copy of the FORTRAN code can be found in APPENDIX II.

APPENDIX II
FORTRAN FOR BEAM DEFLECTION BY F.E.M.

```
C     SIMPLE BEAM DEFLECTION BY FEM
C*********************************************************
C MAIN PROGRAM
C  VALID ONLY IF 10 ELEMENTS OR LESS CONSIDERED
C*********************************************************
      DIMENSION S(10,4,4),GS(22,22),XI(10),Q(10,4),P(11),F(10),
     *GQ(22),X(10),C(20,20),GP(20),D(22),PC(10,100),AM(10,22),
     *PX(10,100),SX(10,100),DX(11),NPEL(10),NMEL(10),QII(10,4),
     *QIII(10,4),AMX(10,100),SMX(10,100),XCOORD(11),XSX(10,10),
     *XINTER(10),W(10,10),XS(10,10),D2(101),ASX(101)

      DATA S,GS,XI,Q,P,F,GQ,X,GP,D,C,PC,AM/160*0,484*0,10*0,40*0,11*0,
     *10*0,22*0,10*0,20*0,22*0,400*0,1000*0,220*0/
      DATA SX,PX,DX,NPEL,NMEL/1000*0,1000*0,11*0,10*0,10*0/
      DATA QII,QIII,AMX,SMX,XCOORD/40*0,40*0,1000*0,1000*0,11*0/
      DATA XSX,XINTER,W,XS,ASX,D2/100*0,10*0,100*0,100*0,101*0,101*0/
         GRW=0.0
         SG=0.0
           NELJG=0
           NELGW=0
      DO 5 I=1,20
      DO 4 J=1,20
       C(I,J)=0.00
  4    CONTINUE
  5    CONTINUE
      WRITE(1,100)
 10   WRITE(1,105)
      READ(1,*)IA
      IF(IA.NE.0)GO TO 20
      CALL EXIT
 20   WRITE(1,107)
      WRITE(1,110)
      READ(1,*)NEL
      IF(NEL.GT.10) GO TO 250
      N3=NEL+1
      WRITE(1,120)
      NCORD=NEL+1
      DO 30 I=2,NCORD
      READ(1,*)XCOORD(I)
      J=I-1
      X(J)=XCOORD(I)-XCOORD(J)
      DX(I)=XCOORD(I)
      DA=DX(I)
```

```
30   CONTINUE
 47  WRITE(1,127)
     READ(1,*)IAL
     IF(IAL.EQ.1)GO TO 48
     IF(IAL.EQ.0)GO TO 60
       GO TO 47
 48  WRITE(1,129)
     READ(1,*)ITYPE
     IF(ITYPE.EQ.1)GO TO 50
     IF(ITYPE.EQ.2)GO TO 54
     IF(ITYPE.EQ.3)GO TO 50
50   WRITE(1,130)
     READ(1,*)IC
     IF(IC.EQ.1) GO TO 52
     IF(IC.EQ.0) GO TO 51
        GO TO 50
51   WRITE(1,136)
      READ(1,*)(P(I),I=1,N3)
     GO TO 65
52    WRITE(1,135)
     READ(1,*)P1
      DO 65 I=1,N3
        P(I)=P1
65    CONTINUE
     IF(ITYPE.EQ.1)GO TO 60
54   DO 58 I=1,NEL
     WRITE(1,137)I
     READ(1,*)NP
     NPEL(I)=NP
     IF(NPEL(I).EQ.0)GO TO 58
     WRITE(1,138)I
     DO 56 J=1,NP
     READ(1,*)PC(I,J),PX(I,J)
     SX(I,J)=(PX(I,J)-DX(I))/X(I)
56    CONTINUE
58    CONTINUE
60   WRITE(1,133)
     READ(1,*)IQM
     IF(IQM.EQ.1)GO TO 62
     IF(IQM.EQ.0)GO TO 90
      GO TO 60
62   DO 66 I=1,NEL
     WRITE(1,139)I
     READ(1,*)NM
     NMEL(I)=NM
     IF(NMEL(I).EQ.0)GO TO 66
     WRITE(1,143)
     DO 64 J=1,NM
     READ(1,*)AM(I,J),AMX(I,J)
     SMX(I,J)=(AMX(I,J)-DX(I))/X(I)
64     CONTINUE
66     CONTINUE
90   WRITE(1,145)
     READ(1,*)E
```

```
 198 WRITE(1,147)
     READ(1,*)AI
     IF(AI.EQ.1.0) GO TO 199
     IF(AI.EQ.2.0) GO TO 200
     GO TO 198
199  WRITE(1,149)
     READ(1,*) XT
     GO TO 212
200  WRITE(1,150)
     READ(1,*)IE
     IF(IE.EQ.1)GO TO 210
     IF(IE.EQ.0)GO TO 220
       GO TO 200
210  WRITE(1,155)
     READ(1,*)X2,Y2
     XT=(X2*Y2**3)/12
212  DO 215 J=1,NEL
        XI(J)=XI
215   CONTINUE
     GO TO 230
220  WRITE(1,156)
       DO 225 J=1,NEL
     READ(1,*)X2,Y2
       XI(J)=(X2*Y2**3)/12
 225 CONTINUE
230  DO 240 J=1,NEL
      F(J)=E*XI(J)
240  CONTINUE
     WRITE(1,185)
     WRITE(1,201)XT
     WRITE(1,202)E,DA
     WRITE(1,2003)NEL
     WRITE(1,205)
     DO 245 I=1,NCORD
      WRITE (1,207)XCOORD(I)
245  CONTINUE
     WRITE(1,208)
     DO 242 J=1,NEL
     IPNEL=NPEL(J)
      IF(IPNEL.EQ.0)GO TO 242
     DO 242 I=1,IPNEL
     WRITE(1,209)PC(J,I),PX(J,I)
242  CONTINUE
     DO 247 J=1,NEL
     MNEL=NMEL(J)
     IF(MNEL.EQ.0)GO TO 247
     DO 247 I=1,MNEL
     WRITE(1,211)AM(J,I),AMX(J,I)
247  CONTINUE
     WRITE(1,203)P1
     GO TO 300
250  WRITE(1,160)
     GO TO 10
300  DO 500 I=1,NEL
```

```
       CALL STIFF(I,X,S,F)
 500  CONTINUE
 600  DO 650 I=1,NEL
       CALL LOAD(I,X,P,NMEL,NPEL,SX,SMX,PC,AM,QII,QIII,Q)
 650  CONTINUE
 700  CALL DIAGS(NEL,S,GS,C)
 800  CALL DIAGQ(NEL,Q,GQ,GP)
      N=NEL*2
 900  CALL INVDET(C,N)
 1000 CALL DEFLEC(C,GP,N,D)
      WRITE(1,165)
       NA=N+2
       WRITE(1,170)(D(I),I=1,NA)
      WRITE(1,185)
      WRITE(1,1015)
      READ(1,*)ISHAPE
      IF(ISHAPE.EQ.0) CALL EXIT
      DO 3333 I=1,NEL
           XINTER(I) = X(I)/10.0
         DO 3777 J=2,10
            NJ=J-1
            XSX(I,J)= XSX(I,NJ) +XINTER(I)
            XS(I,J)=XSX(I,J)/X(I)
 3777    CONTINUE
 3333  CONTINUE
                CALL SHAPE(NEL,D,X,XS,W,GRW,NELGW,NELJG,XSI)
      WRITE(1,1025)NELGW,GRW
            K=1
      DO 4000 I=1,NEL
      DO 3999 J=1,10
C           WRITE(8,1050)XSX(I,J)
C           WRITE(9,1050)W(I,J)
            ASX(K)=XSX(I,J)+DX(I)
            D2(K)=W(I,J)
            K=K+1
 3999 CONTINUE
 4000 CONTINUE
      ASX(K)=DX(NEL+1)
      DO 5000 I=1,K
      WRITE(5,1050)ASX(I)
      WRITE(6,1050)D2(I)
 5000 CONTINUE
      WRITE(7,1060)XSI,NELJG
           J=NELGW
           PLACE = XSI*X(J) + DX(J)
      CALL STRAIN(J,X,XS,XSI,D,NELJG,SG)
           STRESS=E*SG
      WRITE(1,1030) SG,STRESS,PLACE
      CALL EXIT
C ****************************
C ** FORMAT STATEMENTS *******
C ****************************
 100  FORMAT(2X,'THIS PROGRAM DETERMINES THE DEFLECTION OF ',/,
     *'A SIMPLY SUPPORTED BEAM BY FINITE ELEMENT ANALYSIS ',/,
```

```
      *'IT SHOULD BE NOTED THAT THIS PROGRAM IS ONLY VALID FOR',/,
      *'10 ELEMENTS OR LESS')
105    FORMAT(2X,'DO YOU WISH TO CONTINUE? (1=YES,0=NO)')
107    FORMAT(2X,'PLEASE ENTER THE FOLLOWING DATA----')
110    FORMAT(2X,'HOW MANY ELEMENTS ARE TO BE CONSIDERED?')
120    FORMAT(2X,'PLEASE ENTER THE COORDINATE OF',/
      *'EACH NODE (FROM LEFT TO RIGHT)')
127    FORMAT(2X,'ARE ANY LOADS APPLIED?',
      *'(1=YES,0=NO)')
129    FORMAT(2X,'ARE THEY DISTRIBUTED OR CONCENTRATED?',
      *,/,15X,'1=DISTRIBUTED',/,15X,'2=CONCENTRATED',/,15X,'3=BOTH')
130    FORMAT(2X,'IS THE DISTRIBUTED LOAD CONSTANT OVER THE BEAM?',/,
      *'1=YES,0=NO')
133    FORMAT(2X,'ARE ANY MOMENTS APPLIED TO THE BEAM?',/,'1=YES,0=NO')
135    FORMAT(2X,'PLEASE ENTER THE UNIFORM LOAD')
136    FORMAT(2X,'PLEASE ENTER THE UNIFORM LOAD APPLIED A EACH NODE--'
      *,/,2X,'FROM LEFT TO RIGHT--')
137    FORMAT(2X,'HOW MANY CONCENTRATED LOADS ARE APPLIED IN ELEMENT ',
      *,I2,' ?')
138    FORMAT(2X,'PLEASE ENTER EACH CONCENTRATED LOAD APPLIED ',
      *'IN ELEMENT NUMBER ',I2,/, 'AND THE ',
      *'DISTANCE OF EACH FROM THE LEFT END OF THE BEAM')
139    FORMAT(2X,'HOW MANY MOMENTS ARE APPLIED IN ELEMENT ',I2,' ?')
143    FORMAT(2X,'PLEASE ENTER EACH APPLIED MOMENT ',
      *'AND THE DISTANCE OF EACH FROM THE LEFT END')
145    FORMAT(2X,'PLEASE ENTER THE MODULUS OF ELASTICITY'
      *,/,'EE6')
147    FORMAT(2X,'WHICH DO YOU WISH TO ENTER?',/,15X,'1=MOMENT OF',
      *' INERTIA',/,15X,'2=RECTANGULAR DIMENSIONS OF X-SECTION')
149    FORMAT(2X,'PLEASE ENTER THE MOMENT OF INERTIA')
150    FORMAT(2X,'IS THE CROSSECTION OF THE BEAM CONSTANT FOR EACH',/,
      *'ELEMENT?',/,2X,'(1=YES,0=NO)')
155    FORMAT(2X,'PLEASE ENTER THE X (HORIZ) AND Y (VERT) DIMENSIONS')
156    FORMAT(2X,'PLEASE ENTER THE X (HORIZ) AND Y (VERT) DIMENSIONS',/,
      *'FOR EACH ELEMENT',/,'X1= ,Y1=',/,'etc')
157    FORMAT(F3.0,X,F3.0)
160    FORMAT(2X,'I"M SORRY,BUT THIS PROGRAM CAN ONLY HANDLE A',/,
      *'MAXIMUM OF 10 ELEMENTS')
165    FORMAT(2X,'THE DEFLECTION AND SLOPE AT EACH NODAL POINT',/,
      *'LISTED BELOW IN DESCENDING ORDER (I=1,NUMBER OF ELEMENTS)')
170    FORMAT('W=',F10.4,5X,'SLOPE =',F10.4,)
185    FORMAT(//'**********************************************',/)
201    FORMAT(2X,'THE FOLLOWING IS A LIST OF THE INPUT DATA',/,
      *'**********************************************',//,
      *T5,'BEAM CHARACTERISTICS',/,T10,'MOMENT OF INERTIA',
      *T50,F5.3)
202    FORMAT(T10,'MODULUS OF ELASTICITY',T50,E10.4,/
      *T10,'LENGTH OF BEAM',T50,F4.1)
2003    FORMAT(T10,'NUMBER OF ELEMENTS',T50,I2)
205    FORMAT(T10,'COORDINATES OF NODAL POINTS ARE')
207    FORMAT(T50,F5.2)
 208   FORMAT(T5,'LOADING CONDITIONS')
209    FORMAT(T10,'CONCENTRATED LOAD',
      *T48,F7.1,/,T15,'POINT APPLIED',T50,F5.1,/,/)
```

```
 211  FORMAT(T10,'MOMENT MAGNITUDE',T48,F7.1,/,T15,'POINT APPLIED',
     *T50,F5.1,/,/)
 203  FORMAT(T10,'THE DISTRIBUTED LOAD IS',T50,F5.1)
 1015 FORMAT(2X,'DO YOU WISH TO GENERATE MORE DEFLECTIONS',/,
     *'USING SHAPE FUNCTIONS? (0=NO)')
 1025 FORMAT(2X,'THE MAXIMUM DEFLECTION OCCURS IN ELEMENT ',I2,/,
     *'AND IT"S MAGNITUDE IS',5X,E12.6)
 1030 FORMAT(T10,'MAXIMUM STRAIN IS',T50,E12.6,/,
     *T10,'MAXIMUM STRESS IS',T50,E12.6,/,/,T5,'THE DISTANCE FROM ',
     *'FROM THE LEFT END OF THE BEAM',/,'TO THE POINT OF MAXIMUM',
     *' STRESS IS',T50,F12.6)
 1050 FORMAT(X,F12.6)
 1060 FORMAT(F3.0,I4)

      END

C     ******************************
C      LOCAL STIFF MATRIX
C     ******************************
      SUBROUTINE STIFF(I,X,S,F)
      DIMENSION X(10),F(10),S(10,4,4)
      A=F(I)/X(I)**3
      B=X(I)**2

       S(I,1,1)=12.
       S(I,1,2)=6.*X(I)
       S(I,1,3)=-12.
       S(I,1,4)=6. *X(I)
       S(I,2,2)=4. *B
       S(I,2,3)=-6. *X(I)
       S(I,2,4)=2.*B
       S(I,3,3)=12.
       S(I,3,4)=-6.*X(I)
       S(I,4,4)=4.*B

       DO 10 K=1,4
        DO 9 J=1,K
        S(I,J,K)=S(I,J,K)*A
        S(I,K,J)=S(I,J,K)
   9  CONTINUE
  10  CONTINUE
      RETURN
      END
C ********************
C ** LOAD MATRIX *****
C ** SUBROUTINE  *****
C ********************
      SUBROUTINE LOAD(I,X,P,NMEL,NPEL,SX,SMX,PC,AM,QII,QIII,Q)
      DIMENSION NMEL(10),NPEL(10),SX(10,100),PC(10,100),AM(10,22),
     *X(10),Q(10,4),Q2(10,20,4),Q3(10,20,4),
     *Q1(10,4),P(11),QII(10,4),QIII(10,4),SMX(10,100)
      B=X(I)/3.
      A=X(I)/20.
      BI=1./X(I)
```

```
      NI=I+1
        NPL=NPEL(I)
        NML=NMEL(I)

        Q1(I,1)=(P(I)*7.+P(NI)*3.)*A
        Q1(I,2)=(B*(P(I)*3. + P(NI)*2.))*A
        Q1(I,3)=(P(I)*3. + P(NI)*7.)*A
        Q1(I,4)=(-B*(P(I)*2. + P(NI)*3.))*A

        DO 5 J=1,NPL

        Q2(I,J,1)=(1-3.*SX(I,J)**2 +2.*SX(I,J)**3)*PC(I,J)
       Q2(I,J,2)=((X(I)*SX(I,J)**2)*(1.-2.*SX(I,J)+SX(I,J)**2))*PC(I,J)
        Q2(I,J,3)=SX(I,J)**2*(3.-2.*SX(I,J))*PC(I,J)
        Q2(I,J,4)=X(I)*(SX(I,J)**2)*(SX(I,J)-1.)*PC(I,J)
    5 CONTINUE

        DO 7 J=1,NML

        Q3(I,J,1)=(-6.*SMX(I,J)+6.*SMX(I,J)**2)*AM(I,J)*BI
        Q3(I,J,2)=(X(I)*(1.-4.*SMX(I,J)+3.*SMX(I,J)**2))*AM(I,J)*BI
        Q3(I,J,3)=(6.*SMX(I,J)-6*SMX(I,J)**2)*AM(I,J)*BI
        Q3(I,J,4)=(X(I)*(3.*SMX(I,J)**2-2.*SMX(I,J)))*AM(I,J)*BI
    7 CONTINUE

      DO 30 K=1,4
      DO 29 J=1,NPL
       QII(I,K)=QII(I,K)+Q2(I,J,K)
   29 CONTINUE
   30 CONTINUE

      DO 40 K=1,4
      DO 39 J=1,NML
       QIII(I,K)=QIII(I,K)+ Q3(I,J,K)
   39 CONTINUE
   40 CONTINUE
       DO 20 K=1,4
          Q(I,K)=Q1(I,K)+QII(I,K)+QIII(I,K)
C
C               NOTE:   I=ELEM. NO.
C                       J=CONC. LOAD NO.
C                       K=MATRIX CHAR.
   20  CONTINUE
      RETURN
      END
C *********************
C ** DIAGONALIZATION **
C ** OF STIFF MATRIX **
C *********************
      SUBROUTINE DIAGS(NEL,S,GS,C)
      DIMENSION S(10,4,4),GS(22,22),C(20,20)
      DO 110 K=1,10
      KUL=(K-1)*2
        DO 100 J=1,4
```

```
        DO 90 I=1,4
        NEWI=KUL+I
        NEWJ=KUL+J
        GS(NEWI,NEWJ)=GS(NEWI,NEWJ)+S(K,I,J)
 90   CONTINUE
 100  CONTINUE
 110  CONTINUE
      N=NEL*2 +2
       NEWN=N-1
     A=0.0
     B=0.0
     DO 10 K=1,N
     A=GS(N,K)
     GS(N,K)=0.0
     GS(N,K)=GS(NEWN,K)
     GS(NEWN,K)=A
 10   CONTINUE
     DO 11 K=1,N
     B=GS(K,N)
      GS(K,N)=GS(K,NEWN)
      GS(K,NEWN)=B
 11  CONTINUE
      N2=NEL*2
     DO 15 K=1,N2
      K2=K+1
     DO 15 J=1,N2
      J2=J+1
     C(J,K)=GS(J2,K2)
 15  CONTINUE
      RETURN
     END
C **********************
C ** GLOBALIZATION   ***
C ** OF LOAD MATRIX  ***
C **********************
     SUBROUTINE DIAGQ(NEL,Q,GQ,GP)
     DIMENSION Q(10,4),GQ(22),GP(20)

     DO 11 K=1,NEL
       KXI=(K-1)*2
      DO 10 I=1,4
          M=I+KXI
       GQ(M)=GQ(M)+Q(K,I)
 10  CONTINUE
 11  CONTINUE
      NWN=NEL*2+2
     A2=GQ(NWN)
     GQ(NWN)=GQ(NWN-1)
     GQ(NWN-1)=A2
     NB=NWN-1
     DO 15 K=2,NB
       GP(K-1)=GQ(K)
 15  CONTINUE
     RETURN
```

```
      END
C ********************************
C ** INVERSION OF GLOBAL     ******
C ** STIFFNESS MATRIX        ******
C ** OBTAINED FROM NUMERICAL ******
C ** METHODS TEXT--HORNBECK  ******
C ** (GAUSS-JORDAN METHOD)   ******
C ********************************
      SUBROUTINE INVDET(C,N)
      DIMENSION C(20,20),J(50)
      REAL*8 PD,DETM
      PD=1.0
      DO 124 L=1,N
      DD=0.0
      DO 123 K=1,N
  123 DD=DD+C(L,K)*C(L,K)
      DD=SQRT(DD)
  124 PD=PD*DD
      DETM=1.0
      DO 125 L=1,N
  125 J(L+20)=L
      DO 144 L=1,N
      CC=0.0
      M=L
      DO 135 K=L,N
      IF((ABS(CC)-ABS(C(L,K))).GE.0.0)GO TO 135
  126 M=K
      CC=C(L,K)
  135 CONTINUE
  127 IF(L.EQ.M) GO TO 138
  128 K=J(M+20)
      J(M+20)=J(L+20)
      J(L+20)=K
      DO 137 K=1,N
      S=C(K,L)
      C(K,L)=C(K,M)
  137 C(K,M)=S
  138 C(L,L)=1.0
      DETM=DETM*CC
      DO 139 M=1,N
  139 C(L,M)=C(L,M)/CC
      DO 142 M=1,N
      IF (L.EQ.M) GO TO 142
  129 CC=C(M,L)
      IF(CC.EQ.0.0) GO TO 142
  130 C(M,L)=0.0
      DO 141 K=1,N
  141 C(M,K)=C(M,K)-CC*C(L,K)
  142 CONTINUE
  144 CONTINUE
      DO 143  L=1,N
      IF(J(L+20).EQ.L) GO TO 143
  131 M=L
  132 M=M+1
```

```
      IF (J(M+20).EQ.L) GO TO 133
 136  IF (N.GT.M) GO TO 132
 133  J(M+20)=J(L+20)
      DO 163 K=1,N
      CC=C(L,K)
      C(L,K)=C(M,K)
 163  C(M,K)=CC
      J(L+20)=L
 143  CONTINUE
      DETM=DABS(DETM)
      DTNRM=DETM/PD
      RETURN
      END
C *********************************
      SUBROUTINE DEFLEC(C,GP,N,D)
      DIMENSION C(20,20),GP(20),D(22)
      DO 10 I=1,N
      DO 10 J=1,N
       D(I+1)=D(I+1)+C(I,J)*GP(J)
   10 CONTINUE
      K=N+2
      D(K)= D(K-1)
      D(K-1)=0.0
      RETURN
      END
C ****************************************
C ******     SHAPE FUNCTIONS USED    ******
C ******     TO GENERATE CURVE PTS   ******
C ****************************************
      SUBROUTINE SHAPE(NEL,D,X,XS,W,GRW,NELGW,NELJG,XSI)
      DIMENSION XS(10,10),X(10),D(22),W(10,10)
           DO 10 I=1,NEL
       K1=I*2-1
        K2=K1+1
        K3=K1+2
        K4=K1+3
       DO 5 J=1,10


           XN1=1.0-3.0*XS(I,J)**2.0 + 2.0*XS(I,J)**3
           XN2= X(I)*XS(I,J)*( 1.0 - 2.0 *XS(I,J) +XS(I,J)**2.0)
           XN3= XS(I,J)**2.0 *(3.0 -2.0 *XS(I,J))
           XN4= X(I)*XS(I,J)**2.0*(XS(I,J) -1.0)
      W(I,J)= XN1*D(K1) +XN2*D(K2) +XN3*D(K3) +XN4*D(K4)

      A=W(I,J)
      IF(GRW.GT.A)GO TO 4
           GRW= W(I,J)
           NELGW=I
           NELJG=J
           XSI=XS(I,J)

  4    CONTINUE
```

```
  5    CONTINUE
 10    CONTINUE

           RETURN
           END
C *************************************
C ******  DETERMINATION OF STRAIN  ******
C *************************************

      SUBROUTINE STRAIN (J,X,XS,XSI,D,NEWJG,SG)
      DIMENSION X(10),XS(10,10),D(22)

           I=NEWJG
           K1= J*2-2
         K2 =K1 +1
         K3 =K1 +2
         K4 =K1 +3

           SG = 1.0/X(J)**2.0  *((-6.0 + 12.0*XS(J,I))*D(K1)
     *           + (-4.0* X(J) + 6.0 * X(J) *XS(J,I))* D(K2)
     *           +(6.0 -12.0*XS(J,I)) * D(K3)
     *           +(6.0 * X(J)*XS(J,I) -2.0*X(J)) *D(K4))

           RETURN
           END
```

REFERENCES

American Institute of Steel Construction. Manual of Steel Construction. Chicago: AISC, 1980.

Desai, C.S. Elementary Finite Element Method. New Jersey: Prentice-Hall, 1979

McCormac, Jack C. Structural Analysis. New York, Harper and Row Publishers, 1975.

McCormac, Jack C. Structural Steel Design. New York, Harper and Row Publishers, 1981

Singer and Pytel Strength of Materials. New York: Harper and Row Publishers, 1980.