FINITE ELEMENT PROGRAM UTILIZING FRONTAL TECHNIQUE

by

Amir Salehpour

Submitted in Partial Fulfillment of the Requirements
for the Degree of
Master of Science
in the
Mechanical Engineering
Program

11/8/84
Adviser Date

Sally M. Hotchkiss November 8, 1984
Dean of the Graduate School Date

Youngstown State University
March, 1985

ABSTRACT

FINITE ELEMENT PROGRAM UTILIZING FRONTAL TECHNIQUE

AMIR SALEKPOUR

MASTER OF SCIENCE

YOUNGSTOWN STATE UNIVERSITY

The objective in this thesis was to investigate a relatively new technique for handling Finite Element Calculations known as the FRONTAL TECHNIQUE. This method was proposed by Bruce M. Irons in about 1970, but has not as yet attained any prominence in existing well known software computer programs utilizing the Finite Element Method.

Study of the method herein includes the preparation of a computer program which may be used to solve plane stress or plane strain problems and problems involving the bending of plates. The program is tested by comparing the results obtained for three elasticity problems with the corresponding solution found in the literature. The problems studied include a thick walled cylinder pressure vessel subjected to internal pressure, a simply supported plate subjected to lateral loading, and a long bar having a symmetrical temperature distribution about its axes.

Also included are a comparison of results on the thick-walled pressure vessel and a simply supported plate problem with results obtained using the SAPIV program which is on file at Youngstown State University Computer Center.

In general, very good results were obtained for all three problems when compared to theoretical solutions. From this limited investigation there is evidence that the FRONTAL TECHNIQUE leads to shorter computer time and perhaps greater accuracy for the same mesh size.

ACKNOWLEDGEMENT

I would like to dedicate this thesis to my wife, Kelly and our sone, Sawson Jason, for their constant encouragement which has helped me complete this project.

I also extend my gratitude to professors Dr. Frank A. D'Isa and Dr. J. Alam for their time and effort in the development and review of my thesis. Sandie Arnold's time and effort in typing the thesis is also greatly appreciated.

TABLE OF CONTENTS

LIST OF SYMBOLS

Symbol	Definition
$\{U\}$	Matrix defining the displacement of any point within the element.
$[N]$	Matrix of shape functions
$\{\varepsilon\}$	Strain Matrix
$[B]$	Matrix relating assumed displacement field to nodal displacements
$\{d\}$	Matrix defining nodal displacements
$[D]$	Elasticity matrix relating the displacement to strain
$\{\sigma\}$	Matrix defining the stresses
b_1, b_2	Coefficients defining the state of displacement function
Fi	i = 1, 2, 3, . . . Element nodal forces
E	Modulus of Elasticity
$[K]$	Element Stiffness matrix
Π	Total potential energy
$\bar{U}$	Total Strain energy
W	Total External energy
X, Y	Global Co-ordinates
xi, yi	i = 1, 2, 3, . . . Co-ordinates of node i
ξi, ηi	i = 1, 2, 3, . . . The natural co-ordinates for an element
$[j]$	Jacobian Matrix of Transformation
Wi, Wj	Weight function
ni	i = 1, 2, 3, . . . order of Integration
t	Thickness

Symbol	**Definition**
α	Coefficient of thermal expansion
ν	Poisson's Ratio
Pi	Internal pressure
σrr, $\sigma\theta\theta$	Radial and Circumfrential stress, respectively
Ur	Radial displacement
a, b	Inner and outer radius
r	Radius at any point
q	Lateral loading
Mxx, Myy	Moment about x and y axes

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

The purpose of this thesis is to introduce a finite element program that utilizes the Frontal Technique and compare its efficiency with other existing finite element programs. Chapter I describes the concept of the finite element method and the procedures by which one can approach a problem. Chapter II discusses element formulation for the plane stress or plane strain situations used in the finite element program. Appendix A provides instructions for the use of the computer program in Appendix B. Chapter III discusses different methods of equation solution and their efficiency in handling finite element programs. Chapter IV describes the computer procedures for the finite element program given in appendix B. In Chapter V several examples are given in order to prepare input data and also familiarize the user with the input data coding given in Appendix A. Chapter VI discusses the accuracy and efficiency of the results for the sample problems given in Chapter V. Conclusions for the given examples are given in Chapter VII.

CHAPTER I

CONCEPT OF FINITE ELEMENT METHOD

The Finite Element Method is a numerical technique for obtaining solutions for a variety of problems. The basic concept of the Method, when applied to problems, is that the structure can be divided into small elements (finite elements) which are connected by a number of nodal points (see Figure I.1). The displacements of these nodal points are the unknowns of the problem. Each element is defined by a set of functions that discribe its deformation in terms of the nodal displacements. This displacement function states the strains within an element and therefore the stresses can be defined throughout the element.

One of the greatest advantages of the Finite Element Method is the ability to automate the equation formulation process and to solve problems with irregular and complex structures.

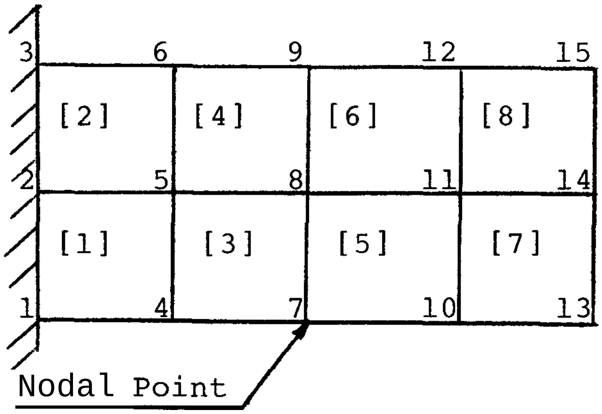


Figure 1.1 A typical Finite Element mesh for the beam problem

Another andvantage of the Finite Element Method is the flexibility with which one can formulate the properties of individual elements. There are four approaches.[1] Only the first two will be explained in detail in this thesis.

1. Direct approach
2. Minimization of Total Potential Energy Approach
3. Weighted Residuals Approach
4. Energy Balance Approach.

1. DIRECT APPROACH

An easy method to understand the relationships between the finite element analysis and the real structure problem is the Direct Approach Formulation. This method uses the combination of three sets of elasticity equations, namely, equations of equilibrium, strain-displacement, and the constitutive relationships.

---

[1] Kenneth L. Huebner, The Finite Element Method for Engineers, (New York: Wiley, 1975), pp. 3-15.

The direct approach method of stiffness formulation is presented as follows:

A. An element shown in Figure 1.2 is considered with two degrees of freedom per node. For node i the displacement relationship can be approximated as

$$\left\{U\right\} = \left\{\begin{matrix} u(x,y) \\ v(x,y) \end{matrix}\right\} = \sum_{i=1}^{n} [N_i]\left\{d\right\} = [N_i,\ N_j,\ \ldots]\left\{\begin{matrix} u_i \\ v_i \\ u_j \\ v_j \end{matrix}\right\},$$

where $\{U\}$ is the displacement at any point within the element; u_i, v_i, are the nodal displacements; and $[N]$ are prescribed functions of position called shape functions.

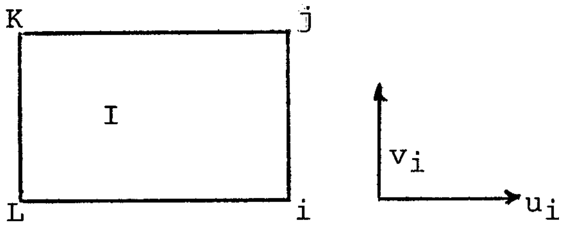


Figure I.2 A typical two-dimensional rectangular element

B. The strains $\{\varepsilon\}$ are defined in terms of displacements by the strain-displacement relationships, which for two dimensional plane stress and for plane strain are:

$$\{\varepsilon\} = \left\{\begin{matrix} \varepsilon,x \\ \varepsilon,y \\ \gamma,xy \end{matrix}\right\} = \begin{bmatrix} \frac{\partial u}{\partial x} & 0 \\ 0 & \frac{\partial V}{\partial y} \\ \frac{\partial u}{\partial y} & \frac{\partial V}{\partial x} \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \left\{\begin{matrix} U \\ V \end{matrix}\right\}$$

Substituting for $\left\{\begin{matrix} U \\ V \end{matrix}\right\}$ yields

$$\{\varepsilon\} = \sum_{i=1}^{n} \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 \\ 0 & \frac{\partial N_i}{\partial y} \\ \frac{\partial N_i}{\partial y} & \frac{\partial N_i}{\partial x} \end{bmatrix} \{di\}$$

or

$$\{\varepsilon\} = \sum_{i=1}^{n} [B]_i \{di\}$$

where [B] is called the strain matrix.

C. The constitutive law is introduced to relate stresses to the displacements.

The relation between stresses and strains, assuming initial strain to be zero, is of the form

$$\sigma = D\,\varepsilon$$

where D is the elasticity matrix. The equations that relate stresses to strains are as follows:

$$\varepsilon_x = \frac{1}{E}\sigma x - \frac{\nu}{E}\sigma y$$

$$\varepsilon_y = -\frac{\nu}{E}\sigma x + \frac{1}{E}\sigma y$$

$$\gamma_{xy} = [2(1+\nu)/E]\tau_{xy} \quad .$$

These relations can be written in matrix form as

$$\begin{Bmatrix} \varepsilon x \\ \varepsilon y \\ \gamma xy \end{Bmatrix} = \begin{bmatrix} \frac{1}{E} & -\frac{\nu}{E} & 0 \\ -\frac{\nu}{E} & \frac{1}{E} & 0 \\ 0 & 0 & \frac{2(1+\nu)}{E} \end{bmatrix} \begin{Bmatrix} \sigma x \\ \sigma y \\ \tau xy \end{Bmatrix} ,$$

or

$$\{\varepsilon\} = [H]\{\sigma\} ,$$

which can be solved for $\{\sigma\}$, by inverting [H].

$$\{\sigma\} = [H]^{-1}\{\varepsilon\} \quad ,$$

where

$$[D] = [H]^{-1} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad .$$

D. Equivalent nodal forces: The main concern in finding the element stiffness matrix is to find a set of nodal forces which are statically equivalent to the constant stress field acting at the edges of the element. Since the equations of $\{\sigma\}$ are in terms of displacements, it is possible to relate the nodal forces $\{F\}$ to the displacements. This will result in element stiffness equations.

1.a. EXAMPLE OF DIRECT APPROACH FORMULATION

Consider an axial member shown in Figure 1.3.

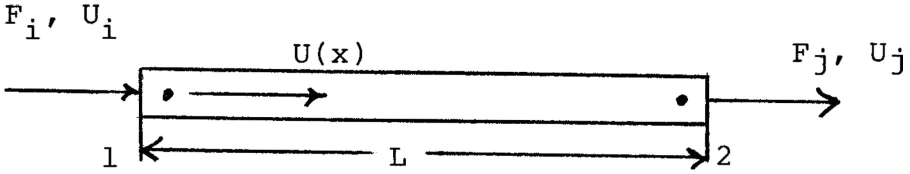


Figure 1.3 Axial member

a. Since there are two nodal displacements $\begin{Bmatrix} U_i \\ U_j \end{Bmatrix}$, the choice is to use a linear polynomial in the x-direction to describe the displacement functions? Thus:

$$U(x) = b_1 + b_2X = \left\{ 1 \quad X \right] \begin{Bmatrix} b_1 \\ b_2 \end{Bmatrix} \quad . \qquad \text{I.1}$$

---

[2] Richard H. Gallagher, <u>Finite Element Analysis Fundamentals</u> (Englewood Cliffs, New Jersy: Prentice-Hall, 1975), pp. 108-113.

Boundary conditions state that at $x=0$; $U=U_i$ and at $x=L$; $U=U_j$;

$$U(0) = U_i = \begin{Bmatrix} 1 & 0 \end{Bmatrix} \begin{Bmatrix} b_1 \\ b_2 \end{Bmatrix} \quad , \tag{I.2}$$

$$U(L) = U_j = \begin{Bmatrix} 1 & L \end{Bmatrix} \begin{Bmatrix} b_1 \\ b_2 \end{Bmatrix} \quad . \tag{I.3}$$

Combine these two to obtain:

$$\begin{Bmatrix} U_i \\ U_j \end{Bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & L \end{bmatrix} \begin{Bmatrix} b_1 \\ b_2 \end{Bmatrix} = [C] \begin{Bmatrix} b_1 \\ b_2 \end{Bmatrix} \quad . \tag{I.4}$$

In terms of b_1, b_2

$$\begin{Bmatrix} b_1 \\ b_2 \end{Bmatrix} = [C]^{-1} \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} = \frac{1}{L} \begin{bmatrix} L & 0 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} \quad . \tag{I.5}$$

Substituting this relationship into equation 1.1,

$$U(x) = \begin{Bmatrix} 1 & x \end{Bmatrix} \begin{bmatrix} 1 & 0 \\ -1/L & 1/L \end{bmatrix} \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} \quad , \tag{I.6}$$

or

$$U(x) = \begin{Bmatrix} (1-(1/L)x) & (1/L)x) \end{Bmatrix} \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} \quad , \tag{I.7}$$

$$U(x) = [N_1 \quad N_2] \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} = [N] \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} \quad , \tag{I.8}$$

where $(1-(x/L)) = N_1$, and $(x/L) = N_2$ are called the shape functions that represent the displacement field.

b. This step is the application of strain-displacement equations. Strain in x direction is found by differentiating EQ VIIa with respect to x.

$$\varepsilon x = \frac{dU(x)}{dx} = \begin{Bmatrix} -\frac{1}{L} & \frac{1}{L} \end{Bmatrix} \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} \quad . \qquad \text{I.9}$$

c. Introducing the stress-strain relationship

$$\{\sigma\} = [E] \; \{\varepsilon\} \qquad \text{I.10}$$

$$\{\sigma\} = E \begin{Bmatrix} -\frac{1}{L} & \frac{1}{L} \end{Bmatrix} \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} \quad . \qquad \text{I.11}$$

d. The final step is to transform the stresses into joint forces, which is done by multiplying stresses by the element cross-sectional area A. For this example

$$\begin{Bmatrix} F_1 \\ F_2 \end{Bmatrix} = A \begin{bmatrix} -1 \\ 1 \end{bmatrix} \sigma x \qquad \text{I.12}$$

Substituting II.c. into I.d results in

$$\begin{Bmatrix} F_1 \\ F_2 \end{Bmatrix} = AE \begin{bmatrix} -1 \\ 1 \end{bmatrix} \begin{Bmatrix} -\frac{1}{L} & \frac{1}{L} \end{Bmatrix} \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} \qquad \text{I.13}$$

or

$$\begin{Bmatrix} F_1 \\ F_2 \end{Bmatrix} = \frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} \quad . \qquad \text{I.14}$$

Therefore:

$$[K] = \frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

(Element Stiffness Equation)

and

$$F = [K] \; \{d\} \; .$$

(Hooke's Law)

2). **Minimization of Total Potential Energy Approach.**

The principle of total potential energy (Π) is given as the difference between the internal or strain energy U and the external energy W due to the applied loads.

$$\Pi = \bar{U} - W \qquad \text{I.15}$$

Strain energy density for a differential element is given by

$$d\bar{U}0 = \{\sigma\}^T \{d_\varepsilon\}, \qquad \text{I.16}$$

where

$$\{\sigma\} = [E]\{\varepsilon\} \quad . \qquad \text{I.17}$$

Substituting 1.17 into 1.16 yields

$$d\bar{U}0 = [[E]\{\varepsilon\}]^T \{d\varepsilon\}$$

$$= \{\varepsilon\}^T [E] \{d\varepsilon\} \quad .$$

Integrating dU0 from 0 to the final state

$$\bar{U}_0 = \int_0^\varepsilon \{\varepsilon\}^T [E] \{d\varepsilon\} \quad ,$$

$$\bar{U}_0 = \tfrac{1}{2}\{\varepsilon\}^T [E]\{\varepsilon\} \quad ,$$

and, the total strain energy U is given by

$$\bar{U} = \int_{Vol} \bar{U}odV \qquad \text{I}$$

or

$$\bar{U} = \tfrac{1}{2} \int_{Vol} \{\varepsilon\}^T [E] \{\varepsilon\} \, dv \qquad \text{I.18}$$

The total external energy W is given by

$$W = \int_v \{U_b\}^T \{fb\} \, dv + \int_S \{U_S\}^T \{fS\} \, dS - \{d\}^T \{P\}, \qquad \text{I.19}$$

where:

$\{U\}$ = Displacement at any point

$\{fb\}$ = Body forces

$\{fs\}$ = Applied surface forces

$\{P\}$ = Applied nodal forces

Therefore:

$$\Pi = \tfrac{1}{2} \int_V \{\varepsilon\}^T [E] \{\varepsilon\}\, dV - \int_V \{u_b\} \{f_b\}\, d_V - ,$$
$$\int_S \{u_s\}^T \{fs\}\, ds - \{d\}^T \{p\} \quad . \qquad I.20$$

In the finite element method, displacements $\{U\}$ can be described in terms of the nodal displacements $\{d\}$ and shape functions.

$$\{U\} = [N] \{d\}$$

The strain $\{\varepsilon\}$ within the element can be expressed in terms of nodal displacement as

$$\{\varepsilon\} = [B] \{d\} .$$

Now, total potential energy Π can be expressed in terms of shape functions and nodal displacements

$$\Pi = \tfrac{1}{2} \int_V \{d\}^T [B]^T [E][B]\{d\} dV - \int_V \{d\}^T [N]^T \{fb\} dv$$
$$- \int_S \{d\}^T [Ns]^T \{fs\} ds - \{d\}^T P \quad .$$

Minimizing total potential energy with respect to nodal displacements results in equilibruim equations. Thus:

$$\frac{d\,\Pi}{d\{d\}} = \int_V [B]^T[E][B]\{d\}\,dV - \int_V [N]^T\{fb\}dv$$

$$- \int_s [N]^T\{fs\}\,ds - \{P\} = 0 \quad ,$$

or
$$\int_V [B]^T[E][B]\{d\}\,dv = \int_V [N]^T\{fb\}dv + \int_s [N]^T\{fs\}ds$$

$$+ \{P\} \; .$$

$$[K]\{d\} = \{f\} \qquad \text{(Equilibruim EQ)}$$

$$[K] = \int_V [B]^T[E][B]dV \qquad \text{(Stiffness Matrix)}$$

$$\{f\} = \int_V [N]^T\{fb\}dv + \int_S [N]^T\{fs\}ds + \{P\}$$

(Equivalent Nodal Forces)

Consider the example of an axial member that was discussed in the previous section. It was found that the shape function for the member is

$$[N] = [N_1 \;\; N_2] = \left\{1 - \frac{X}{L} \quad \frac{X}{L}\right\}$$

and the displacement at any point is of the form

$$U(x) = [N]\{U\}$$

and strain $\varepsilon = \varepsilon_x$ is the derivative of $U(x)$ with respect to x.

$$\varepsilon_x = \frac{du}{dx} = \left\{-\frac{1}{L} \quad \frac{1}{L}\right\} \begin{Bmatrix} U_i \\ U_j \end{Bmatrix} ,$$

for which the general symbolism is

$$\{\varepsilon\} = [B]\{U\} ,$$

where

$$[B] = [B_1 \quad B_2] = \left\{ -\frac{1}{L} \quad \frac{1}{L} \right\} .$$

The total potential energy Π, assuming constant [E] and A, can be written as

$$\Pi = \frac{EA}{2}\{U\}^T \int_0^L [[B]^T[B]dx]\{U\} - \{U\}^T f \int_0^L [N]^T dx .$$

Minimizing Π with respect to U results in

$$\frac{d\Pi}{d\{u\}} = 0 = EA\int_0^L [B]^T[B]dx\{U\} - f\int_0^L [N]^T dx ,$$

where

$$[K] = EA\int_0^1 [B]^T[B]dx$$

$$= EA\int_0^1 \begin{Bmatrix} B_1 \\ B_2 \end{Bmatrix} [B_1 \quad B_2] \, dx .$$

$$= EA\int_0^1 \begin{bmatrix} B_1^2 & B_1B_2 \\ B_2B_1 & B_2^2 \end{bmatrix} dx .$$

Substituting for [B]

$$[K] = EA \begin{bmatrix} \int_0^1 \frac{1}{L^2}\,dx & \int_0^1 -\frac{1}{L^2}\,dx \\ \int_0^1 -\frac{1}{L^2}\,dx & \int_0^1 \frac{1}{L^2}\,dx \end{bmatrix}$$

$$[K] = \frac{EA}{L}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} .$$

This result is identical to that obtained by the Direct Approach.

CHAPTER II

ELEMENT FORMULATION FOR PLANE STRESS OR PLANE STRAIN

Element formulation is done by considering a two-dimensional parabolic isoparametric element shown in Figure II.1.

X = Gaussian Sampling Points

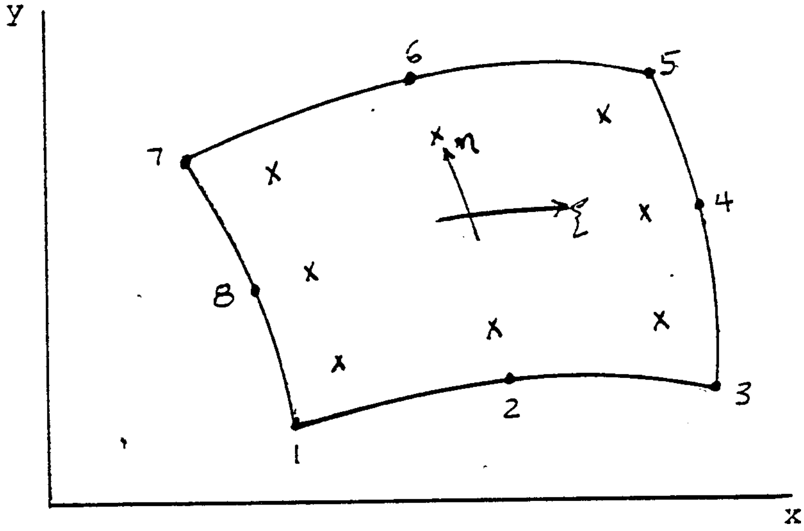


Figure II.1 Two-dimensional parabolic isoparametric element

The transformation of element co-ordinates ξ, η to 0 the global Cartesian co-ordinates is done by the use of shape functions [N].

$$X = \sum_{i=1}^{8} Ni(\xi, \eta) Xi$$

$$Y = \sum_{i=1}^{8} Ni(\xi, \eta) Yi \quad ,$$

or, in matrix form

$$\begin{Bmatrix} X \\ Y \end{Bmatrix} = \sum_{i=1}^{8} [Ni] \begin{Bmatrix} X_i \\ Y_i \end{Bmatrix} \quad .$$

The shape functions for this element are given as follows:[3]

For the corner nodes (1, 3, 5, 7)

$$N_i = \tfrac{1}{4}(1+\xi\xi_i)(1+\eta\eta_i)(\xi\xi_i+\eta\eta_i-1) \quad ,$$

where

$$\xi_i = \pm 1 \text{ and } \eta_i = \pm 1$$

For the mid-side nodes (2, 4, 6, 8)

$$\xi_i = 0, \quad N_i = \tfrac{1}{2}(1-\xi^2)(1+\eta\eta_i) \quad .$$

$$\eta_i = 0, \quad N_i = \tfrac{1}{2}(1+\xi\xi_i)(1-\eta^2) \quad .$$

The displacement field for plane stress or plane strain are $U(\xi, \eta)$ and $V(\xi, \eta)$, which can be expressed in matrix form as

$$\begin{matrix} U(\xi, \eta) \\ V(\xi, \nu) \end{matrix} = \sum_{i=1}^{8} [N_i] \begin{Bmatrix} U_i \\ V_i \end{Bmatrix} \quad ,$$

where U_i amd V_i are the nodal displacements at node i.

The next step is to find strain $\{\varepsilon\}$, where

$$\{\varepsilon\} = \begin{Bmatrix} \varepsilon\, x \\ \varepsilon\, y \\ \gamma\, xy \end{Bmatrix} = [B] \begin{Bmatrix} U_i \\ V_i \end{Bmatrix} \quad .$$

[B] is the strain matrix and can be evaluated by the following procedures.

---

[3] O. C. Zienkiewicz, The Finite Element Method (London: McGraw-Hill, 1977), pp. 155-157.

$$[B] = \sum_{i=1}^{8} \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 \\ 0 & \frac{\partial N_i}{\partial y} \\ \frac{\partial N_i}{\partial y} & \frac{\partial N_i}{\partial x} \end{bmatrix}$$

Using the chain rule and considering only U we obtain

$$U,\xi = \frac{\partial u}{\partial x} \circ \frac{\partial x}{\partial \xi} + \frac{\partial u}{\partial y} \circ \frac{\partial y}{\partial \xi}$$

$$U,\eta = \frac{\partial u}{\partial x} \circ \frac{\partial x}{\partial n} + \frac{\partial u}{\partial y} \circ \frac{\partial y}{\partial \eta}$$

In matrix form

$$\begin{Bmatrix} U,\xi \\ U,\eta \end{Bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \begin{Bmatrix} U,x \\ U,y \end{Bmatrix} ,$$

$$= [J] \begin{Bmatrix} U,x \\ U,y \end{Bmatrix}$$

where [J] is called Jacobian of Transformation. The derivative of U with respect to global co-ordinates x and y is performed by inverting [J].

$$\begin{Bmatrix} u,x \\ u,y \end{Bmatrix} = [J]^{-1} \begin{Bmatrix} u,\xi \\ u,\eta \end{Bmatrix} .$$

This procedure is the same for nodal displacement V.

Therefore,

$$\begin{Bmatrix} U,x \\ U,y \\ V,x \\ V,y \end{Bmatrix} = [J]^{-1} \begin{Bmatrix} U,\xi \\ U,\eta \\ V,\xi \\ V,\eta \end{Bmatrix}$$

The strain-displacement relationships are

$$\{\varepsilon\} = \begin{Bmatrix} \varepsilon x \\ \varepsilon y \\ \gamma xy \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \begin{Bmatrix} U,x \\ U,y \\ V,x \\ V,y \end{Bmatrix} ,$$

or

$$\{\varepsilon\} = [\alpha] \begin{Bmatrix} U,x \\ U,y \\ V,x \\ V,y \end{Bmatrix} = [\alpha]\ [j]^{-1} \begin{Bmatrix} U,\xi \\ U,\eta \\ V,\xi \\ V,\nu \end{Bmatrix}$$

and we know that

$$\frac{\partial u}{\partial \xi} = \sum_{i=1}^{8} \frac{\partial Ni}{\partial \xi} Ui ; \quad \frac{\partial V}{\partial \xi} \quad \sum_{i=1}^{8} \frac{\partial Ni}{\partial \xi}\ Vi$$

$$\frac{\partial U}{\partial \eta} = \sum_{i=1}^{8} \frac{\partial Ni}{\partial \eta}\ Ui ; \quad \frac{\partial V}{\partial \eta} = \sum_{i=1}^{8} \frac{\partial Ni}{\partial \eta}\ Vi \quad ,$$

or, in the matrix form

$$\begin{Bmatrix} U,\ \xi \\ U,\ \eta \\ V,\ \xi \\ V,\ \eta \end{Bmatrix} = [\gamma] \begin{Bmatrix} U_1 \\ V_1 \\ U_2 \\ V_2 \\ \vdots \\ U_8 \\ V_8 \end{Bmatrix} .$$

Substituting this expression in $\{\varepsilon\}$ results in

$$[B] = [\alpha][j]^{-1}[\gamma]$$

By knowing [B], the element stiffness matrix [K] can be found.

$$[K] = \int_V [B]^T[D][B]dv$$

$$= \int_S [B]^T[D][B]tdxdy \quad .$$

Area of an element dx dy can be determined by the expression

$$dx\,dy = \det[j]d\xi\,d\eta \quad .$$

Substituting this in [K] results in

$$[K] = \int_S [B]^T[D][B]t\,\det[j]d\xi\,d\eta \ .$$

A numerical integration is adopted to evaluate these stiffness integrals. Using the Gauss quadrature formula which states that [K] matrix can be solved by

$$\int_{-1}^{1} \int_{-1}^{1} f(x,y)dxdy = \sum_{j=1}^{2} \sum_{i=1}^{1} WiWjf(xi,xj) \quad ,$$

where

Wi, Wj = weight function

η_1, η_2 = order of integration

xi, xj = sampling points

The last step is to find stresses. From the strain-displacement relationship

$$\{\varepsilon\} = [B] \{U\} .$$

The stress at any point within the element may be written as

$$\{\sigma\} = [D]\{\varepsilon\}$$

$$\{\sigma\} = [D][B]\{U\} .$$

CHAPTER III

COMPARISON OF FRONTAL SOLUTION TECHNIQUE WITH THE BANDED SOLUTION TECHNIQUE

The method that is adopted in equation solution is a direct factor causing the efficiency of a Finite Element program. Two methods will be described in this section, which are as follows:

III.a). BANDED SOLUTION TECHNIQUE

A matrix is banded if all the nonzero coefficients are located about the diagonal and the zero coefficients are outside, above and below.

The number of nonzero coefficients in stiffness matrix [K] is independent of how nodes are numbered. But changing the node number changes their arrangement. Figure III.1 shows a banded matrix with semi-band width M. The size of M depends very much on the node numbering, therefore, node numbering is very important in banded solutions. The semi-band width M can be obtained as follows:

$M = (D+1) \times [\text{number of degrees of freedom}]$,

where

D = Maximum difference between any two node numbers in any element.

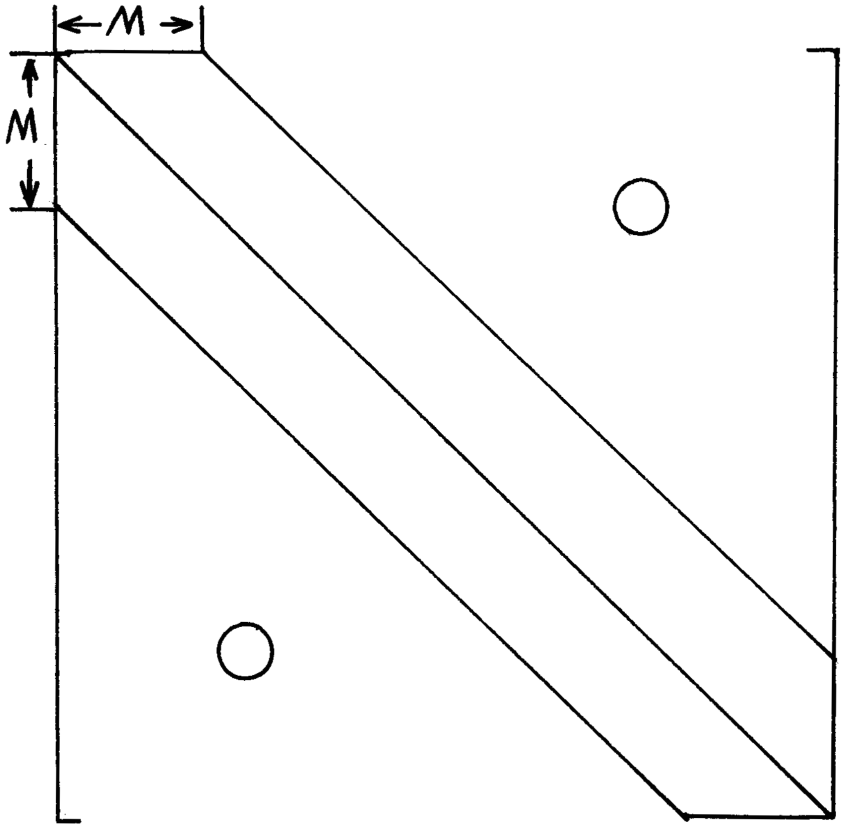


Figure III.1 Banded Stiffness Matrix

It is very important to minimize M by choosing the right node numbering. However, in problems where elements have midside nodes or large three-dimensional problems, band width minimization is very difficult. Therefore, analysts use other methods, like the FRONTAL TECHNIQUE. The FRONTAL TECHNIQUE will be discussed in detail in the next section.

To understand the concept of band solution an example will be considered which has one degree of freedom per node.[4]

---

[4] O. C. Zienkiewicz, The Finite Element Method (London: McGraw-Hill, 1977), pp. 178-204.

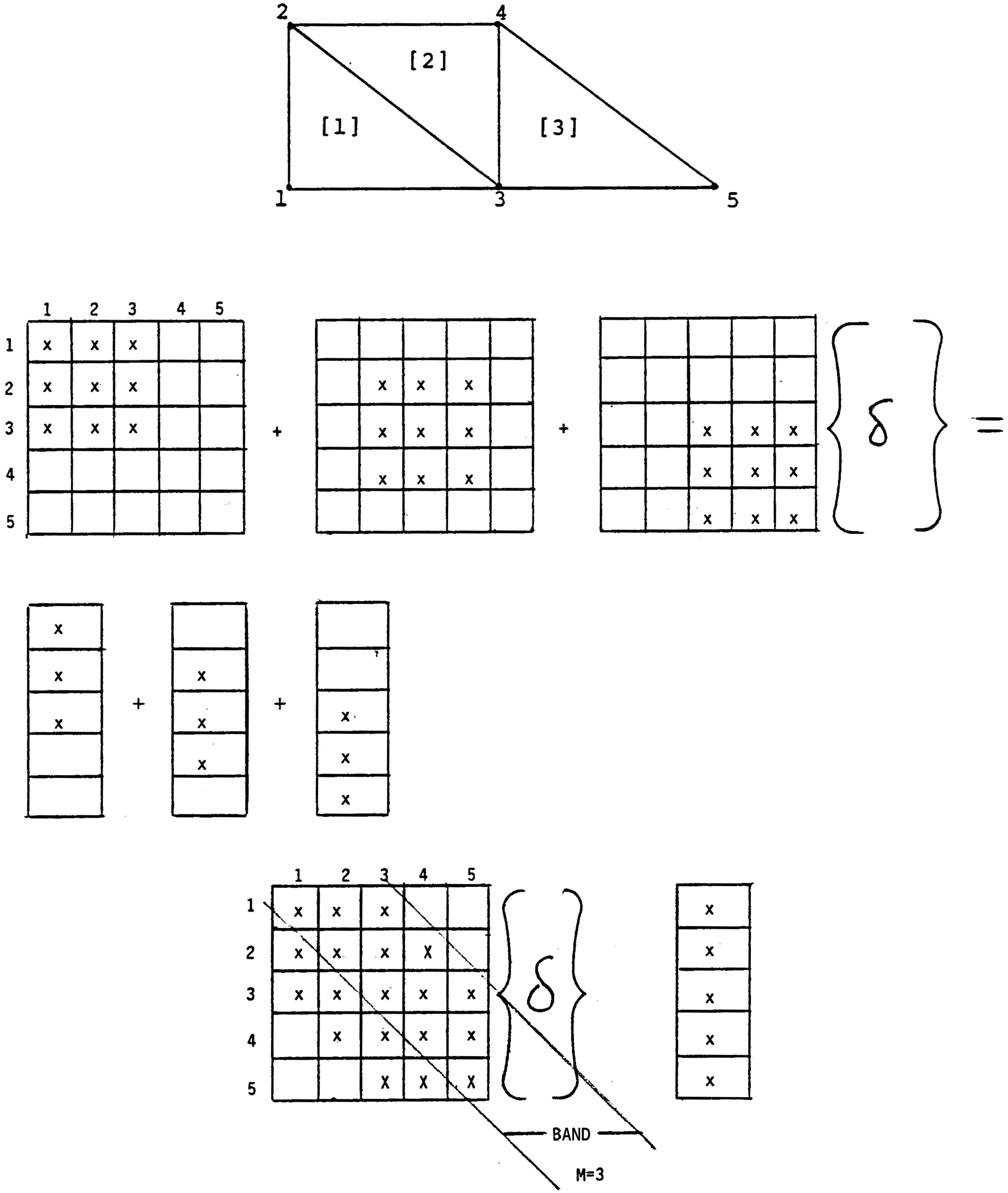


Figure II1.2 A Typical Finite Element Structure

Figure III.2 shows a finite element structure that consists of three triangular elements. Assuming that element properties are in global co-ordinates we can substitute each stiffness component $[K_{ij}]$ in its position. This is done separately for each element. The final step is to assemble all the equations. Assembly can be accomplished by adding all the terms in corresponding spaces of the global matrix.

For this example the band width M = [(5-3)+1]x1 = 3. Consider now the node numbering illustrated on Figure 111.3. In this case the M = [(5-2)+1]x1 = 4 and this increases the storage requirement.

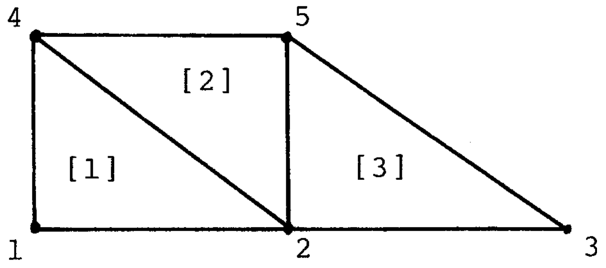


Figure III.3 A Typical Finite Element Structure

III.b). FRONTAL SOLUTION TECHNIQUE

The Frontal solution technique has been developed for the solution of finite element equations and is based on the Gaussian reduction process. The main feature of the frontal solution is to assemble the equations and eliminate them at the same time. Once the coefficients of an element are introduced and assembled the corresponding nodes can be eliminated. This elimination will leave gaps for new nodes as the front moves to the next element. Thus, when the $[K]_I$ for element I of Figure III.4.a is added to the structural stiffness matrix, variables 1, 2, 3, 8 and 11 are eliminated immediately, leaving zero rows and columns. Now $[K]_{II}$ for element II is introduced into the structural stiffness matrix; nodes 5, 6, 7, 10, 13, and 18 through 20 are added to the stiffness matrix. The total number of active variables in the front is 15. Now, the front moves to element III and variables 4 through 7, 9, 10, 12 and 13 are eliminated leaving zero rows and columns for the new variables 22, 23, 25, 26, and 30 through 33. The front moves to element IV and variables 18, 19, 20, 23, 26, 33, 32, and 31 are eliminated leaving zero rows and columns for the new variables 21, 24, and 27 through 29. After assembly of element IV the front consists of 12 variables.

Figure III.4.b shows the graphical representation of Frontal Techniques applied to the finite element assembly of Figure III.4.a. Figure III.4.b shows that when element III is assembled the new coefficients in the front are 21, 22, 24, 25 and 27 through 30. The variables that comprise the stiffness matrix [K] are 14 through 20, 21, 22, 24, 25, and 27 through 30, therefore, the size of the matrix [K] is 15 x 15 while the banded solution has a band width of [(30-14)+1] = 17.

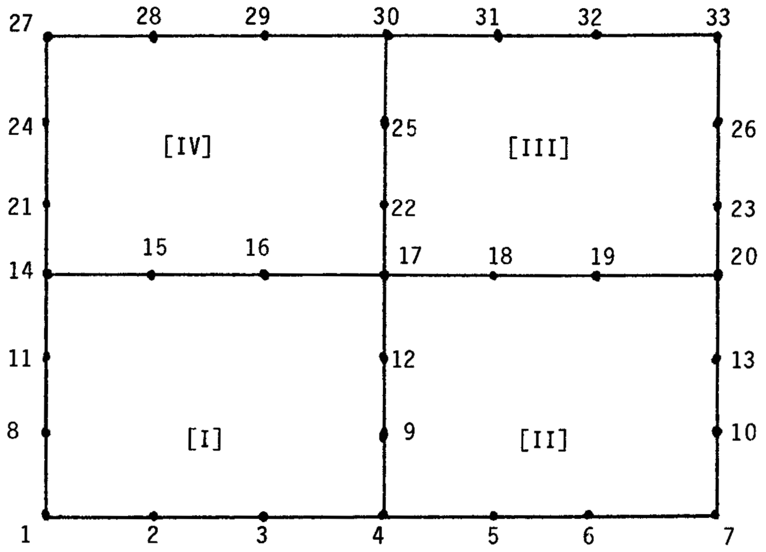


Figure III.4.a Finite element assembly

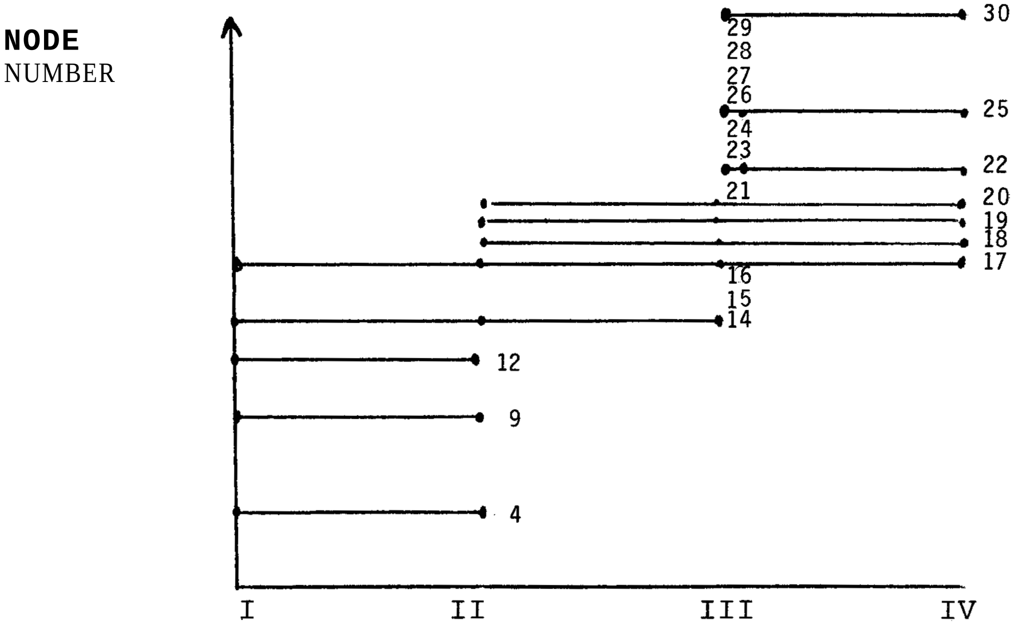
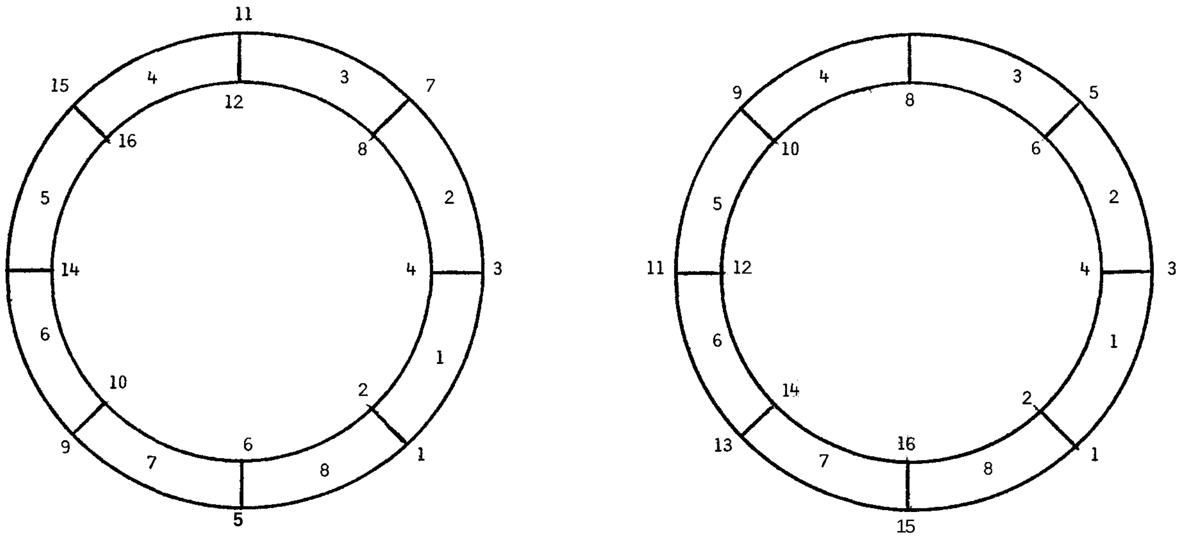


Figure III.4.b Graphical representation of Frontal treatment

Some other significant features of the Frontal Technique are explained as follows:

1). The nodes may be numbered at random, but the efficiency of the solution depends on the element numbering and is very crucial. In general, the objective in numbering elements for a Frontal solution is to keep a common boundary between elements numbered in sequential order. This is opposite to the requirement for a banded solution.

Consider a ring structure shown in Figure 111.5. The banded solution technique requires an artificial order of node numbers to obtain a small bandwidth, but the order of elements for the Frontal solution is critical.



Banded Technique **Frontal Technique**

Figure III.5 Numbering of nodes and element for ring structure

2). Consider a structure shown in Figure III.6.a. If the mesh is too coarse is some region, simply change the mesh with adding an element. This has little effect on the Frontal Technique Figure (III.6.b), but a banded solution may demand extensive re-numbering to prevent a large bandwidth (Figure III.6.c).[5]

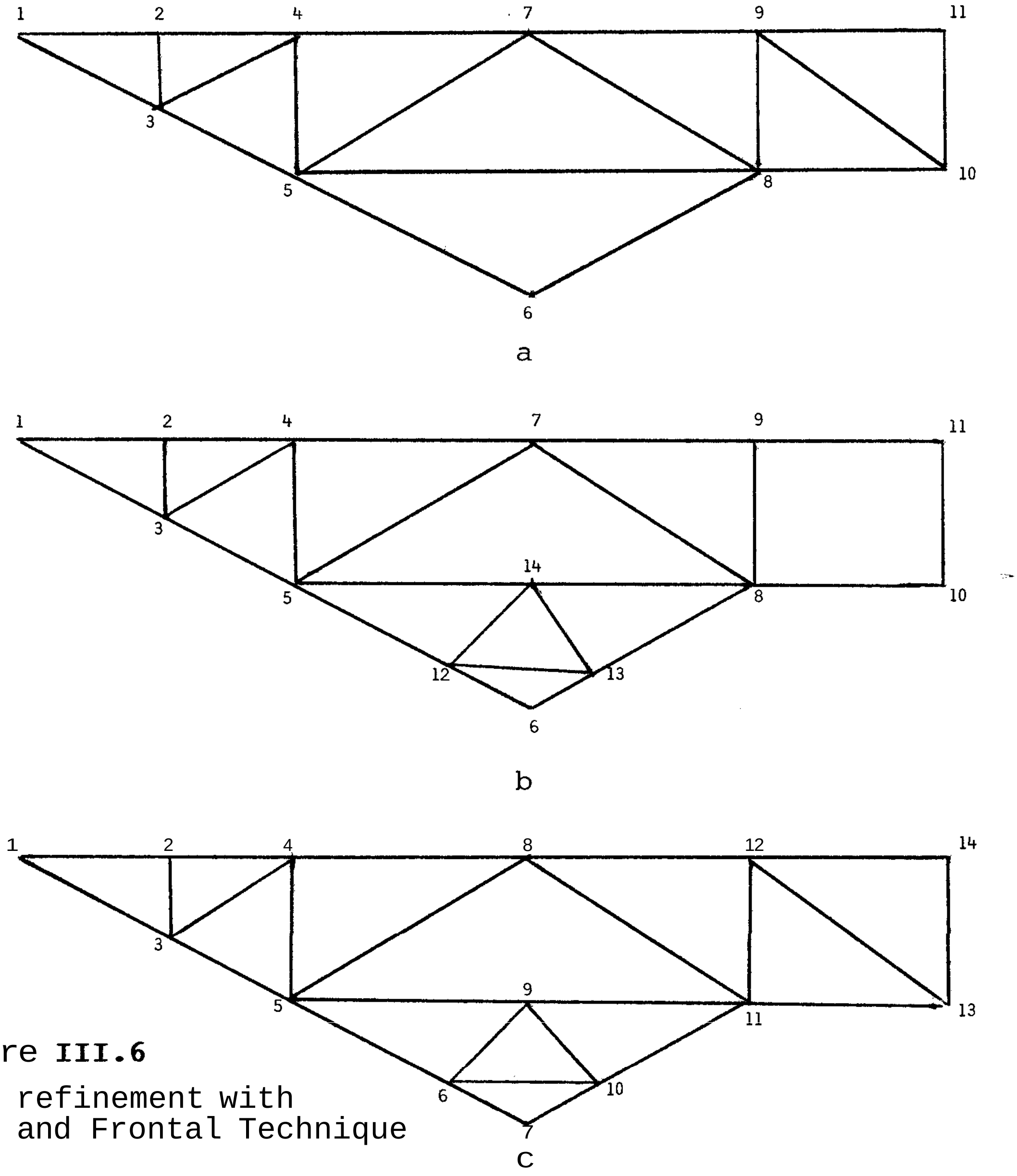


Figure III.6

Mesh refinement with
band and Frontal Technique

---

[5] Bruce M. Irons, "A Frontal Solution Program for Finite Element and Analysis," International Journal for Numerical Methods in Engineering (Vol. 2, 1970), pp. 5-12.

For a better understanding of the FRONTAL Technique, a numerical example is considered here. Figure III.7 shows a structure broken into 3 elements. Assuming one degree of freedom for node, the stiffness and loading for each element are:

$$\begin{bmatrix} 1 & -1 & 1 \\ -1 & 2 & 1 \\ 1 & 1 & 3 \end{bmatrix} \begin{Bmatrix} W_i \\ W_j \\ W_k \end{Bmatrix} = \begin{Bmatrix} 4 \\ 4 \\ 4 \end{Bmatrix}$$

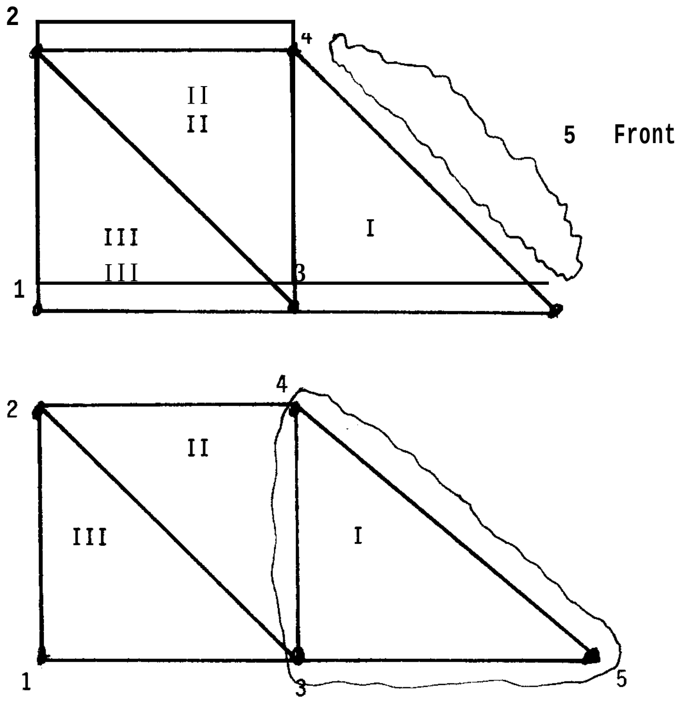
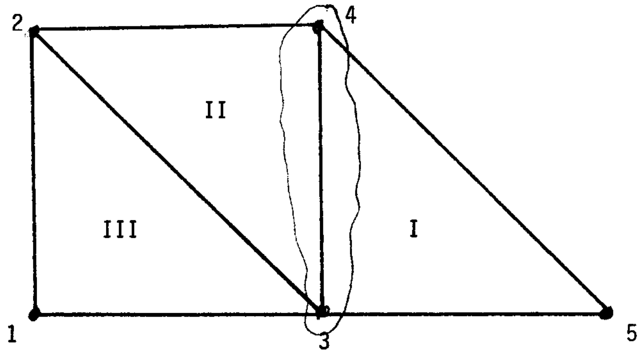


Figure III.7 Finite element assembly for Frontal application

First consider the assembly of element **I** with node numbers (3, 5, 4). The stiffness matrix would be

$$\begin{bmatrix} 1 & -1 & 1 \\ -1 & 2 & 1 \\ 1 & 1 & 3 \end{bmatrix} \begin{Bmatrix} W_3 \\ W_5 \\ W_4 \end{Bmatrix} = \begin{Bmatrix} 4 \\ 4 \\ 4 \end{Bmatrix}$$

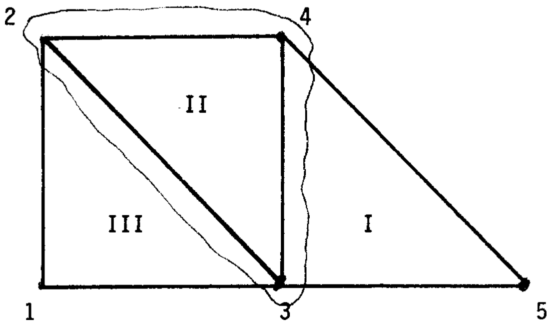


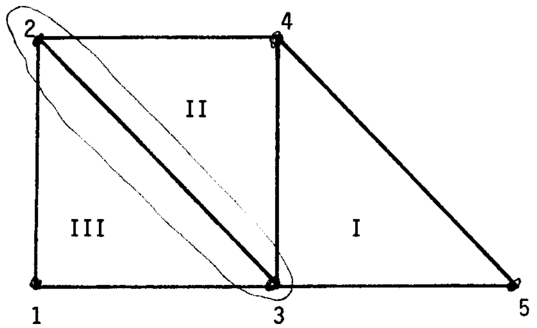
The front moves to element II and W_5 can be eliminated from the other two equations, therefore

$$-W_3 + 2W5 + W_4 = 4$$ (EQ. 111.1. Eliminated Equation)

$$\begin{bmatrix} \frac{1}{2} & 0 & \frac{3}{2} \\ 0 & 0 & 0 \\ \frac{3}{2} & 0 & \frac{5}{2} \end{bmatrix} \begin{Bmatrix} W_3 \\ 0 \\ W_4 \end{Bmatrix} = \begin{Bmatrix} 6 \\ 0 \\ 2 \end{Bmatrix}$$

The second step is assembly of element II. The new variable for element II is W_2 which takes the second position of stiffness matrix.





The stiffness matrix for element II can be written as

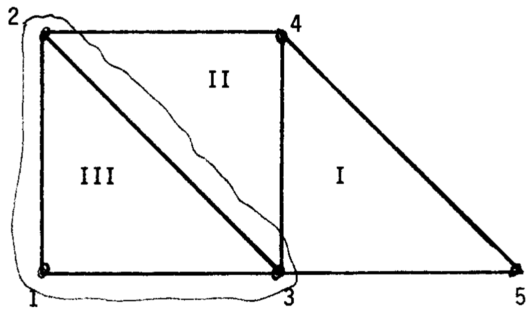
$$\begin{bmatrix} \frac{3}{2} & -1 & \frac{5}{2} \\ -1 & 2 & 1 \\ \frac{5}{2} & 1 & \frac{11}{2} \end{bmatrix} \begin{Bmatrix} W_3 \\ W_2 \\ W_4 \end{Bmatrix} = \begin{Bmatrix} 10 \\ 4 \\ 6 \end{Bmatrix}$$

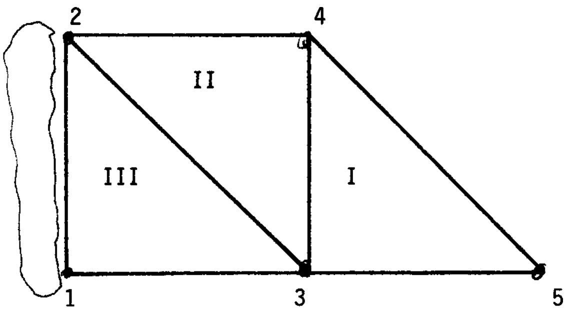
and the equation for the variable W_4 would be

$$\frac{5}{2}W_3 + W_2 + \frac{11}{2}W_4 = 6 \qquad \text{(EQ. III.2)}$$

The elimination process yields

$$\begin{bmatrix} \frac{4}{11} & -\frac{16}{11} & 0 \\ -\frac{16}{11} & \frac{20}{11} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} W_3 \\ W_2 \\ 0 \end{Bmatrix} = \begin{Bmatrix} \frac{80}{11} \\ -\frac{32}{11} \\ 0 \end{Bmatrix}$$





The front continues to move to element III as shown. The new variable in element III is w_1 which will take the third position of stiffness matrix.

$$\begin{bmatrix} \frac{15}{11} & -\frac{27}{11} & 1 \\ -\frac{27}{11} & \frac{42}{11} & 1 \\ 1 & 1 & 3 \end{bmatrix} \begin{Bmatrix} w_3 \\ w_2 \\ w_1 \end{Bmatrix} = \begin{Bmatrix} \frac{124}{11} \\ \frac{76}{11} \\ 4 \end{Bmatrix}$$

Finally, all the equations are fully summed and the next task is to eliminate w_1 and w_2.

$W3 + w_2 + 3w_1 = 4$ **(EQ.** 111.3)

The elimination of w_1 yields:

$$\begin{bmatrix} \frac{34}{33} & -\frac{92}{33} & 0 \\ -\frac{92}{33} & \frac{115}{33} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} w_3 \\ w_2 \\ 0 \end{Bmatrix} = \begin{Bmatrix} \frac{328}{33} \\ \frac{184}{33} \\ 0 \end{Bmatrix}$$

The elimination of w_2 yields:

$-\frac{92}{33}w_3 + \frac{115}{33}w_2 = \frac{64}{33}$ **(EQ.** III.4)

and

$$\begin{bmatrix} -39.6 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} w_3 \\ 0 \\ 0 \end{Bmatrix} = \begin{Bmatrix} 475.2 \\ 0 \\ 0 \end{Bmatrix}$$

$$-39.6W_3 = 475.2 \qquad \text{(EQ. III.5)}$$

From the last equation W_3 and by back substitution the remaining variables can be obtained.

$W_3 = -12$

$W_2 = -8$

$W_1 = 8$

$W_4 = 8$

$W_5 = -8$

CHAPTER IV

COMPUTER PROCEDURES FOR FINITE ELEMENT PROGRAM

The computer program that is given in Appendix B can be utilized to solve linear two-dimensional plane stress or plane strain problems and plate bending problems. The general form of program organization is shown in Figure IV.1.[6] In order to execute the program, the following coding must be used:

```
FIb̸5b̸DISKb̸Inputb̸Name
LOAD MAIN
START
```

The first step to solve a problem is to input the information required by the program. This information has to be in specified formats which is presented in detafl in Appendix A. The given input data are automatically checked by the subrountines, Check 1, ECHO, and Check 2. If any errors are detected, a set of messages will be printed and the program will stop running.

The second step in finite element programming is the solution and output to perform the actual analysis.

The accuracy of results and execution time are main concerns in finite element programming. A large

---

[6] Oktay Ural, Finite Element Method (New York: Intext Press, 1973).

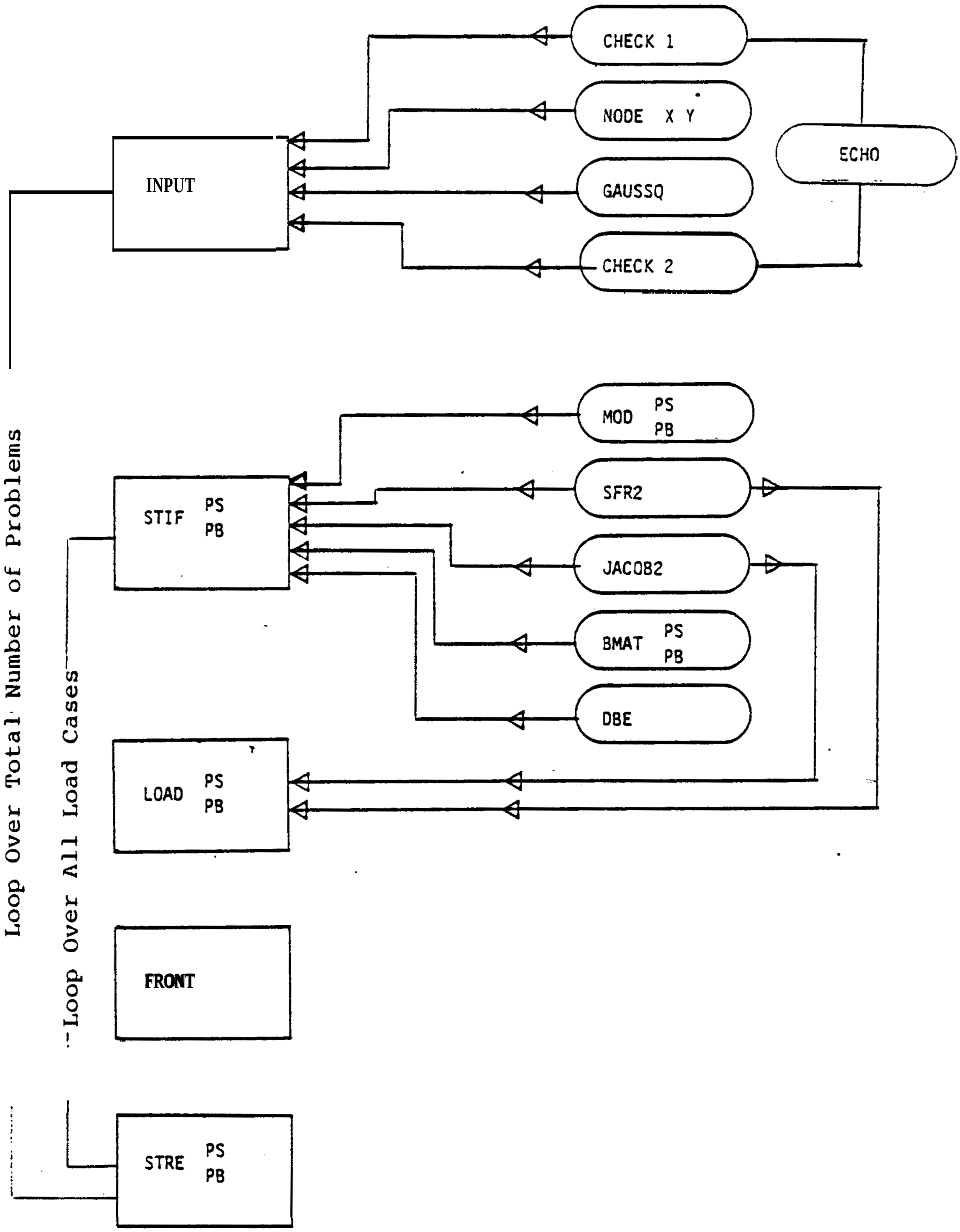


Figure IV.1 Program Organization

percentage of the execution time goes into the solution of the stiffness equations, and adopting a method to solve the stiffness equations is a very important factor to reduce the execution time.

The program shown in Appendix B uses the Frontal Technique to minimize the core storage requirements. The efficiency of the Frontal Technique will compare with the other methods (such as band techniques) throughout the examples that are presented in the latter sections.

Once the data are read into the program, the solution to the problem begins. There are nine subroutines that will perform the solutions, STIF, MOD, SFR2, JACOB2, BMAT, DBE, LOAD, FRONT, and STRE. A brief description of each subrountine is presented, as follows:

Subroutine STIFPS, STIFPB.

This subroutine calculates the stiffness matrix and stress matrix. Suffixes PS and PB denote plane stress or plain strain and plate bending.

Subroutine MODPS, MODPB.

This subroutine calculates the elasticity matrix [D] for plane stress or plain strain and plate bending.

Subroutine SFR2

This subroutine calculates the shape functions [N] and their derivatives.

Subroutine JACOB2

This subroutine calculates the co-ordinates of sampling points, determinant of JACABIAN MATRIX and its inverse, and Cartisian shape function derivatives.

Subroutine BMATPS, BMATPB

This subroutine calculates the strain matrix [B].

Subroutine DBE.

This subroutine will multiply matrix [D] by [B].

Subroutine LOADPS, LOADPB.

This subroutine will read the distributed, gravity, and thermal loadings into the program and reduce them to equivalent nodal forces.

Subroutine FRONT.

This subroutine is very important in finite element programming. This subroutine will assemble the element stiffness equations and solve for the unknown displacements as well as reactions and output of the results.

Subroutine STREPS, STREPB.

This subroutine calculates the stress resultants and principal stresses at the sampling points and will output the results.

CHAPTER V

SAMPLE PROBLEMS WITH INPUT AND OUTPUT

V.A). Thick-Walled Cylindrical Pressure Vessel Subject to Internal Pressure.

The thick-walled pressure vessel shown in Figure V.l is subjected to an internal pressure of 30,000 psi. The modulus of elasticity and Poisson's ratio are 30,000 ksi and 0.3, respectively.

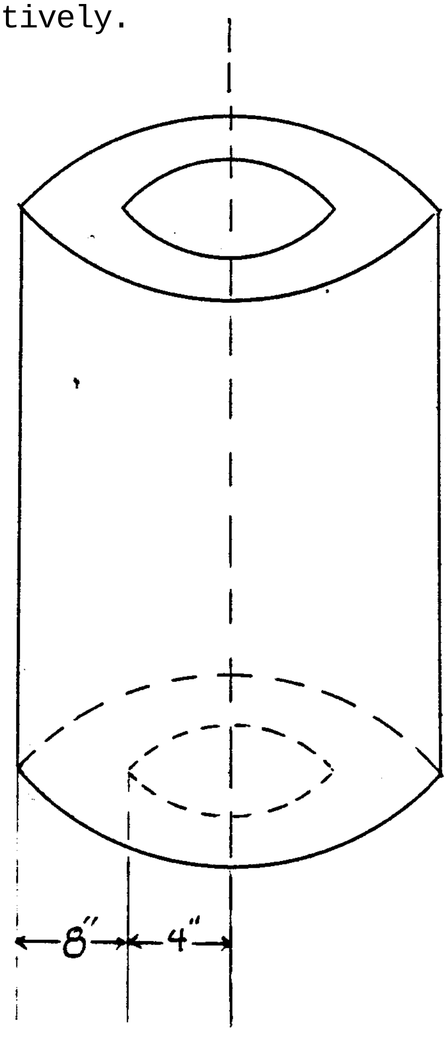


Figure V.1 Thick-walled cylindrical pressure vessel subjected to internal pressure

This problem is treated as a plane strain problem and because of symmetry one quadrant (¼) of cross-section will be considered.

The accuracy of the solutions using the finite element program given in Appendix B will be compared with the exact solutions (see section VI).

The finite element mapping for this problem is given in Figure V.2. The problem is divided into nine elements and 40 nodes. Nodes one through seven are fixed from moving in the y-direction while nodes 34 through 40 are fixed in the x-direction.

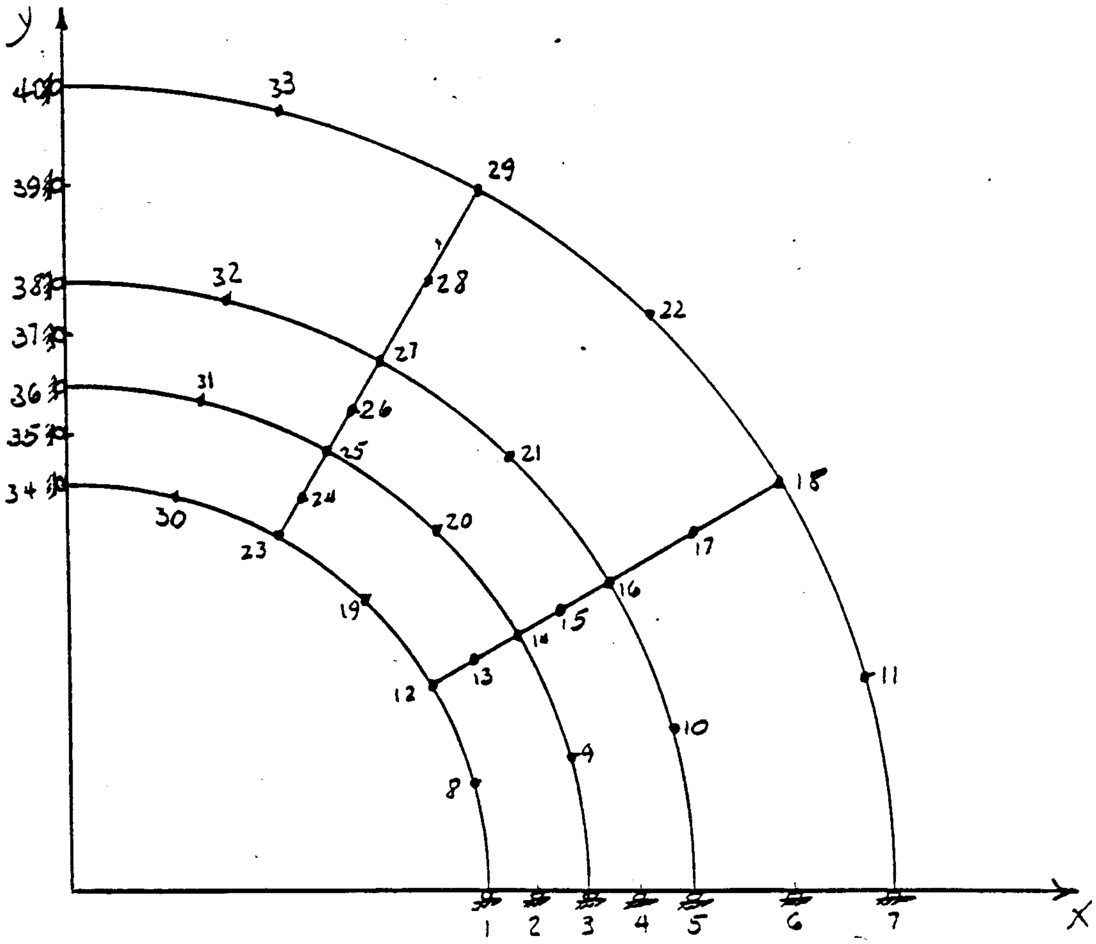


Figure V.2 Mesh generation for pressure vessel

Once numbering of nodes and elements are established, data can be input to the program according to the coding described in Appendix A.1. The output produced by the program is shown on the following pages.

```
      TOTAL NO. OF PROBLEM =.     1



      PROBLEM NO.    I            ****  THIC  CYLI  ER T  T **  **

  NPOIN =  40     NELEM =   9     NVFIX =  14     NCASE =   1     NTYPE =  2     NNODE =  8     NDOFN =  2

  NHATS =   1     NPROP =   5     NGAUS =   3     NDIME =   2     NSTRE =  3     NEVAB =  16

  ELEMENT   PROPERTY      NODE NUMBERS
      I         I           1    2    3    9   14   13   12    8
      2         1           3    4    5   10   16   15   14    9
      3         I           5    6    7   11   18   17   16   10
      4         1          12   13   14   20   25   24   23   19
      5         1          14   15   16   21   27   26   25   20
      6         1          16   17   18   22   29   28   27   21
      7         1          23   24   25   31   36   35   34   30
      8         1          25   26   27   32   38   37   36   31
      9         I          27   28   29   33   40   39   38   32


   NODAL POINT COORDINATES
   NODE         X          Y
      1       4.000      0.0
      2       4.500      0.0
      3       5.000      0.0
      4       5.500      0.0
      5       6.000      0.0
      6       7.000      0.0
      7       8.000      0.0
      8       3.864      1.035
      9       4.829      1.294
     10       5.795      1.553
     11       7.727      2.070
     12       3.464      2.000
     13       3.897      2.250
     14       4.330      2.500
     15       4.763      2.750
     16       5.196      3.000
     17       6.062      3.500
     18       6.928      4.500
     19       2.828      2.828
     20       3.535      3.535
     21       4.242      4.242
     22       5.657      5.657
     23       2.000      3.464
     24       2.250      3.897
     25       2.500      4.330
     26       2.750      4.763
```

```
   28      3.500      6.062
   29      4.500      6.928
   30      1.035      3.864
   31      1.294      4.829
   32      1.553      5.795
   33      2.070      7.727
   34      0.0        4.000
   35      0.0        4.500
   36      0.0        5.000
   37      0.0        5.500
   38      0.0        6.000
   39      0.0        7.000
   40      0.0        8.000

RESTRAINED NODES
NODE CODE        FIXED VALUES
                2
    1   01  0.0           0.0
    2   01  0.0           0.0
    3   01  0.0           0.0
    4   01  0.0           0.0
    5   01  0.0           0.0
    6   01  0.0           0.0
    7   01  0.0           0.0
   34   10  0.0           0.0
   35   10  0.0           0.0
   36   10  0.0           0.0
  :37   10  0.0           0.0
   38   10  0.0           0.0
   39   10  0.0           0.0
   40   10  0.0           0.0


 MATERIAL PROPERTIES
NUMBER,         PROPERTIES
    1          0.300000E+05  0.300000E+00  0.100000E+01  0.0          0.0


AX FRONTWIDTH ENCOUNTERED =    24

  ****  PRES  RE L  DING  **                                         LOAD CASE =  1
   0     0     1     0

     NO, OF LOADED EDGES=,     3

     LIST OF LOADED EDGES AND APPLIED LOADS,
               1          12      8     1
30000.000  30000.000  30000.000      0.0        0.0        0.0
               4          23     19    12
30000.000  30000.000  30000.000      0.0        0.0        0.0
               7          34     30    23
30000.000  30000.000  30000.000      0.0        0.0        0.0
```

```
    TOTAL NODAL FORCES FOR EACH ELEMENT
1        0.1070E+05  0.3998E+02  0.0         0.0         0.0         0.0         0.0         0.0
         0.0         0.0         0.0         0.0         0.9300E+04  0.5320E+04  0.4000E+05  0.1072E+05
2        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
3        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
4        0.9240E+04  0.5400E+04  0.0         0.0         0.0         0.0         0.0         0.0
         0.0         0.0         0.0         0.0         0.5400E+04  0.9240E+04  0.2928E+05  0.2928E+05
5        0.0         0.0         0.0         0.0         0.0         0.0         0.0         0.0
         0.6         0.0         0.0         0.0         0.0         0.0         0.0         0.0
6        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
'7       0.5320E+04  0.9300E+04  0.0         0.0         0.0         0.0         0.0         0.0
         0.0         0.0         0.0         0.0         0.3998E+02  0.1070E+05  0.1072E+05  0.4000E+05
8        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
9        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
       0.0                  0.0
       0.0
       0.0
       0.0


   DISPLACEMENTS

   NODE      X-DISP.          Y-DISP.
        0.0
     1  0.760839E-02  0.0
     2  0.693338E-02  0.0
     3  0.641384E-02  0.0
     4  0.600889E-02  0.0
     5  0.569132E-02  0.0
     6  0.515524E-02  0.0
     7  0.483296E-02  0.0
     8  0.733451E-02  0.193549E-02
     9  0.616682E-02  0.162936E-02
    10  0.544082E-02  0.143231E-02
    11  0.463644E-02  0.129754E-02
    12  0.655216E-02  0.375425E-02
    13  0.595490E-02  0.341680E-02
    14  0.549527E-02  0.315399E-02
    15  0.513071E-02  0.294391E-02
    16  0.483882E-02  0.277665E-02
    17  0.442293E-02  0.257653E-02
    18  0.392193E-02  0.262533E-02
    19  0.533022E-02  0.532923E-02
    20  0.447286E-02  0.447214E-02
    21  0.394151E-02  0.394095E-02
    22  0.339038E-02  0.339026E-02

    23  0.375481E-02  0.655077E-02
    24  0.341719E-02  0.595357E-02
    25  0.315428E-02  0.549402E-02
    26  0.294412E-02  0.512952E-02
```

```
  27  0.277678E-02   0.483768E-02
  28  0.257646E-02   0.442195E-02
  29  0.262498E-02   0.392130E-02
  30  0.193573E-02   0.733306E-02
  31  0.162948E-02   0.616535E-02
  32  0.143232E-02   0.543939E-02
  33  0.129729E-02   0.463516E-02
  34  0.0            0.760692E-02
  35  0.0            0.693186E-02
  36  0.0            0.641230E-02
  37  0.0            0.600733E-02
  38  0.0            0.568976E-02
  39  0.0            0.515374E-02
  40  0.0            0.483150E-02

REACTIONS

NODE       X-FORCE          Y-FORCE
   1  0.0             -0.798552E+04
   2  0.0             -0.272535E+05
   3  0.0             -0.114931E+05
   4  0.0             -0.202741E+05
   5  0.0             -0.155592E+05
   6  0.0             -0.296108E+05
   7  0.0             -0.780579E+04
  34 -0.798646E+04  0.0
  35 -0.272569E+05  0.0
  36 -0.114940E+05  0.0
  37 -0.202749E+05  0.0
  38 -0.155593E+05  0.0
  39 -0.296077E+05  0.0
  40 -0.780455E+04  0.0
```

```
     STRESSES

G.P.  X-COORD.  Y-COORD.   X-STRESS    Y-STRESS   XY-STRESS   I-STRESS    MAX P.S.    MIN P.S.     ANGLE

   ELEMENT NO.=     1
 1     4.1048     0.2462-0.26981E+05 0.46634E+05-0.45841E+04 0.58961E+04 0.46919E+05-0.27265E+05   3.5496
 2     3.9728     1.0642-0.22151E+05 0.42331E+05-0.16821E+05 0.60540E+04 0.47423E+05-0.27243E+05  15.1373
 3     3.6779     1.8390-0.12201E+05 0.32761E+05-0.30212E+05 0.61682E+04 0.47939E+05-0.27378E+05  26.6734
 4     4.4912     0.2694-0.21958E+05 0.40148E+05-0.39314E+04 0.54568E+04 0.40396E+05-0.22206E+05   3.6077
 5     4.3465     1.1645-0.18043E+05 0.36372E+05-0.15976E+05 0.54988E+04 0.40716E+05-0.22387E+05  15.2109
 6     4.0242     2.0122-0.94601E+04 0.28320E+05-0.25572E+05 0.56579E+04 0.41222E+05-0.22363E+05  26.7734
 7     4.8776     0.2926-0.15899E+05 0.36053E+05-0.33895E+04 0.60461E+04 0.36273E+05-0.16119E+05   3.7171
 8     4.7202     1.2648-0.12799E+05 0.32766E+05-0.13473E+05 0.59902E+04 0.36452E+05-0.16484E+05  15.2997
 9     4.9704     2.1854-0.54451E+04 0.26060E+05-0.21455E+05 0.61845E+04 0.36924E+05-0.16309E+05  26.8546
```

```
ELEMENT NO.=     2
1    5.1026    0.3061-0.13720E+05 0.33770E+05-0.31823E+04 0.60150E+04 0.33982E+05-0.13932E+05    3.8164
2    4.9379    1.3232-0.10993E+05 0.30670E+05-0.12432E+05 0.59032E+04 0.34098E+05-0.14420E+05   15.4145
3    4.5719    2.2862-0.44938E+04 0.24385E+05-0.19748E+05 0.59674E+04 0.34410E+05-0.14518E+05   26.9133
4    5.4891    0.3293-0.10843E+05 0.30300E+05-0.28779E+04 0.58372E+04 0.30500E+05-0.11043E+05    3.9820
5    5.3120    1.4235-0.86864E+04 0.27532E+05-0.10892E+05 0.56537E+04 0.30555E+05-0.11710E+05   15.5124
6    4.9183    2.4595-0.30483E+04 0.22094E+05-0.17207E+05 0.57136E+04 0.30833E+05-0.11787E+05   26.9247
7    5.8757    0.3525-0.74890E+04 0.27928E+05-0.26284E+04 0.61316E+04 0.28122E+05-0.76830E+04    4.2213
8    5.6861    1.5238-0.58567E+04 0.25467E+05-0.95140E+04 0.58831E+04 0.28130E+05-0.85200E+04   15.6385
9    5.2646    2.6327-0.10069E+04 0.20798E+05-0.14914E+05 0.59375E+04 0.28370E+05-0.85786E+04   26.9162

ELEMENT NO.=     3
1    6.2132    0.3566-0.70210E+04 0.26471E+05-0.16624E+04 0.58350E+04 0.26553E+05-0.71033E+04    2.8346
2    6.0127    1.5613-0.36404E+04 0.25541E+05-0.69855E+04 0.65703E+04 0.27127E+05-0.52264E+04   12.7916
3    5.5670    2.7339 0.32088E+03 0.21086E+05-0.12285E+05 0.64222E+04 0.26789E+05-0.53816E+04   24.8990
4    6.9863    0.3691-0.30807E+04 0.24728E+05-0.11132E+04 0.64941E+04 0.24772E+05-0.31252E+04    2.2887
5    6.7610    1.6865-0.19922E+04 0.21220E+05-0.63258E+04 0.57682E+04 0.22832E+05-0.36042E+04   14.2964
6    6.2597    3.0801 0.15250E+04 0.16913E+05-0.10238E+05 0.55315E+04 0.22026E+05-0.35879E+04   26.5373
7    7.7594    0.4153 0.70907E+03 0.22716E+05-0.10537E+04 0.70276E+04 0.22767E+05 0.65873E+03    2.7350
8    7.5093    1.9617-0.19975E+03 0.18247E+05-0.56509E+04 0.54143E+04 0.19841E+05-0.17932E+04   15.7470
9    6.9524    3.6926 0.25794E+04 0.14962E+05-0.85193E+04 0.52625E+04 0.19302E+05-0.17607E+04   26.9962

ELEMENT NO.=     4
1    3.4314    2.2655-0.43683E+04 0.25170E+05-0.34678E+05 0.62405E+04 0.48093E+05-0.27291E+05   33.4654
2    2.9077    2.9077 0.10688E+05 0.10690E+05-0.37627E+05 0.64134E+04 0.48316E+05-0.26938E+05   44.9995
3    2.2655    3.4314 0.25164E+05-0.43687E+04-0.34675E+05 0.62385E+04 0.48085E+05-0.27291E+05  -33.4667
4    3.7546    2.4788-0.30634E+04 0.21809E+05-0.29239E+05 0.56237E+04 6.41147E+05-0.22401E+05   33.4792
5    3.1815    3.1815 0.94648E+04 0.94670E+04-0.31769E+05 0.56795E+04 0.41235E+05-0.22303E+05   44.9990
6    2.4788    3.7546 0.21803E+05-0.30620E+04-0.29238E+05 0.56224E+04 0.41142E+05-0.22401E+05  -33.4819
7    4.0777    2.6922-0.15914E+03 0.20524E+05-0.24499E+05 0.61094E+04 0.36774E+05-0.16410E+05   33.5571
8    3.4553    3.4593 0.10107E+05 0.10110E+05-0.26658E+05 0.60650E+04 0.36766E+05-0.16549E+05   44.9983
9    2.6922    4.0777 0.20519E+05-0.15519E+03-0.24499E+05 0.61092E+04 0.36773E+05-0.16409E+05  -33.5615

ELEHENT NO.=     5
1    4.2658    2.8163 0.47936E+03 0.19447E+05-0.22469E+05 0.59778E+04 0.34351E+05-0.14425E+05   33.5581
2    3.6147    3.6147 0.10177E+05 0.10181E+05-0.24335E+05 0.61074E+04 0.34514E+05-0.14156E+05   44.9978
3    2.8163    4.2658 0.19442E+05 0.48325E+03-0.22470E+05 0.59776E+04 0.34350E+05-0.14425E+05  -33.5632
4    4.5889    3.0297 0.12361E+04 0.17824E+05-0.19550E+05 0.57181E+04 0.30767E+05-0.11706E+05   33.5053
5    3.8885    3.8885 0.96516E+04 0.96544E+04-0.21204E+05 0.57918E+04 0.30857E+05-0.11551E+05   44.9981
6    3.0297    4.5889 0.17821E+05 0.12387E+04-0.19551E+05 0.57178E+04 0.30766E+05-0.11707E+05  -33.5100
7    4.9121    3.2430 0.26699E+04 0.17078E+05-0.16926E+05 0.59244E+04 0.28270E+05-0.85216E+04   33.4723
8    4.1623    4.1623 0.99153E+04 0.99172E+04-0.18389E+05 0.59498E+04 0.28305E+05-0.84725E+04   44.9995
9    3.2430    4.9121 0.17076E+05 0.26716E+04-0.16928E+05 0.59243E+04 0.28270E+05-0.85227E+04  -33.4761

ELEMENT NO.=     6
1    5.1780    3.3793 0.22752E+04 0.16546E+05-0.15179E+05 0.56463E+04 0.26183E+05-0.73617E+04   32.4112
2    4.3515    4.3515 0.63405E+04 0.63429E+04-0.17761E+05 0.38050E+04 0.24102E+05-0.11419E+05   44.9980
3    3.3733    5.1780 0.16542E+05 0.22749E+04-0.15181E+05 0.56452E+04 0.26182E+05-0.73651E+04  -32.4154
4    5.7907    3.8061 0.58953E+04 0.15220E+05-0.12122E+05 0.63346E+04 0.23546E+05-0.24305E+04   34.4814
5    4.8245    4.8245 0.95548E+04 0.95562E+04-0.12809E+05 0.57333E+04 0.22364E+05-0.32530E+04   44.9985
6    3.8061    5.7907 0.15219E+05 0.58965E+04-0.12124E+05 0.63347E+04 0.23547E+05-0.24312E+04  -34.4947
7    6.4371    4.4992 0.64865E+04 0.14136E+05-0.94391E+04 0.61868E+04 0.20496E+05 0.12675E+03   33.9707
8    5.4475    5.4475 0.99299E+04 0.99308E+04-0.94016E+04 0.59582E+04 0.19332E+05 0.52871E+03   44.9986
9    4.4992    6.4371 0.14137E+05 0.64886E+04-0.94398E+04 0.61875E+04 0.20498E+05 0.12762E+03  -33.9738
```

```
 ELEMENT NO.=      7
1     1.8390     3.6779 0.32756E+05-0.12201E+05-0.30210E+05 0.61664E+04 0.47933E+05-0.27378E+05  -26.6734
2     1.0642     3.9720 0.42323E+05-0.22153E+05-0.18819E+05 0.60509E+04 0.47414E+05-0.27244E+05  -15.1371
3     0.2462     4.1048 0.46628E+05-0.26978E+05-0.45823E+04 0.58948E+04 0.46912E+05-0.27263E+05   -3.5487
4     2.01 22    4.0242 0.28317E+05-0.94590E+04-0.25571E+05 0.56573E+04 0.41219E+05-0.22361E+05  -26.7741
5     1.1645     4.3465 0.36366E+05-0.18044E+05-0.15975E+05 0.54967E+04 0.40710E+05-0.22388E+05  -15.2111
6     0.2694     4.4912 0.40143E+05-0.21957E+05-0.39303E+04 0.54559E+04 0.40391E+05-0.22204E+05   -3.6070
7     2.1854     4.3704 0.26058E+05-0.54437E+04-0.21454E+05 0.61844E+04 0.36923E+05-0.16308E+05  -26.8577
3     1.2648     4.7202 0.32762E+05-0.12799E+05-0.13473E+05 0.59887E+04 0.36448E+05-0.16485E+05  -15.3005
9     0.2926     4.8776 0.36049E+05-0.15898E+05-0.33889E+04 0.60453E+04 0.36269E+05-0.16118E+05   -3.7169

 ELEMENT NO.=      8
1     2.2862     4.5719 0.24383E+05-0.44929E+04-0.19748E+05 0.59671E+04 0.34409E+05-0.14518E+05  -26.9146
2     1.3232     4.9379 0.30666E+05-0.10994E+05-0.12432E+05 0.59017E+04 0.34094E+05-0.14422E+05  -15.4151
3     0.3061     5.1026 0.33767E+05-0.13719E+05-0.31818E+04 0.60142E+04 0.33979E+05-0.13931E+05   -3.8164
4     2.4595     4.9183 0.22092E+05-0.30485E+04-0.17207E+05 0.57132E+04 0.30832E+05-0.11788E+05  -26.9254
5     1.4235     5.3120 0.27529E+05-0.86872E+04-0.10892E+05 0.56526E+04 0.30552E+05-0.11710E+05  -15.5130
6     0.3293     5.4891 0.30298E+05-0.10841E+05-0.28778E+04 0.58373E+04 0.30499E+05-0.11041E+05   -3.9822
7     2.6327     5.2646 0.20798E+05-0.10079E+04-0.14915E+05 0.59369E+04 0.28370E+05-0.85800E+04  -26.9164
8     1.5238     5.6861 0.25466E+05-0.58571E+04-0.95143E+04 0.58826E+04 0.28129E+05-0.85205E+04  -15.6394
9     0.3525     5.8757 0.27927E+05-0.74856E+04-0.26287E+04 0.61325E+04 0.28121E+05-0.76797E+04   -4.2222

 ELEHENT NO.=      9
1     2.7339     5.5670 0.21086E+05 0.31982E+03-0.12286E+05 0.64218E+04 0.26789E+05-0.53830E+04  -24.8992
2     1.5613     6.0127 0.25541E+05-0.36401E+04-0.69856E+04 0.65702E+04 0.27127E+05-0.52262E+04  -12.7921
3     0.3566     6.2132 0.26471E+05-0.70194E+04-0.16627E+04 0.58355E+04 0.26553E+05-0.71017E+04   -2.8353
4     3.0801     6.2597 0.16915E+05 0.15246E+04-0.10239E+05 0.55318E+04 0.22028E+05-0.35889E+04  -26.5374
5     1.6865     6.7610 0.21221E+05-0.19922E+04-0.63264E+04 0.57686E+04 0.22833E+05-0.36044E+04  -14.2968
6     0.3691     6.9863 0.24730E+05-0.30793E+04-0.11137E+04 0.64953E+04 0.24775E+05-0.31239E+04   -2.2896
7     3.6926     6.9524 0.14965E+05 0.25801E+04-0.85206E+04 0.52634E+04 0.19305E+05-0.17607E+04  -26.9964
8     1.9617     7.5093 0.18250E+05-0.19987E+03-0.56519E+04 0.54150E+04 0.19844E+05-0.17936E+04  -15.7475
9     0.4153     7.7594 0.22721E+05 0.71022E+03-0.10545E+04 0.70293E+04 0.22771E+05 0.65982E+03   -2.7366
```

V.B). SIMPLY SUPPORTED SQUARE PLATE

Consider a square plate shown in Figure V.3. The plate is simply supported around the edges with the modulus of elasticity and Poisson's ratio of 10920 $\frac{k}{in^2}$ and 0.3, respectively. The plate is subjected to a uniformly distributed loading of 1 unit and has a thickness of 0.1".

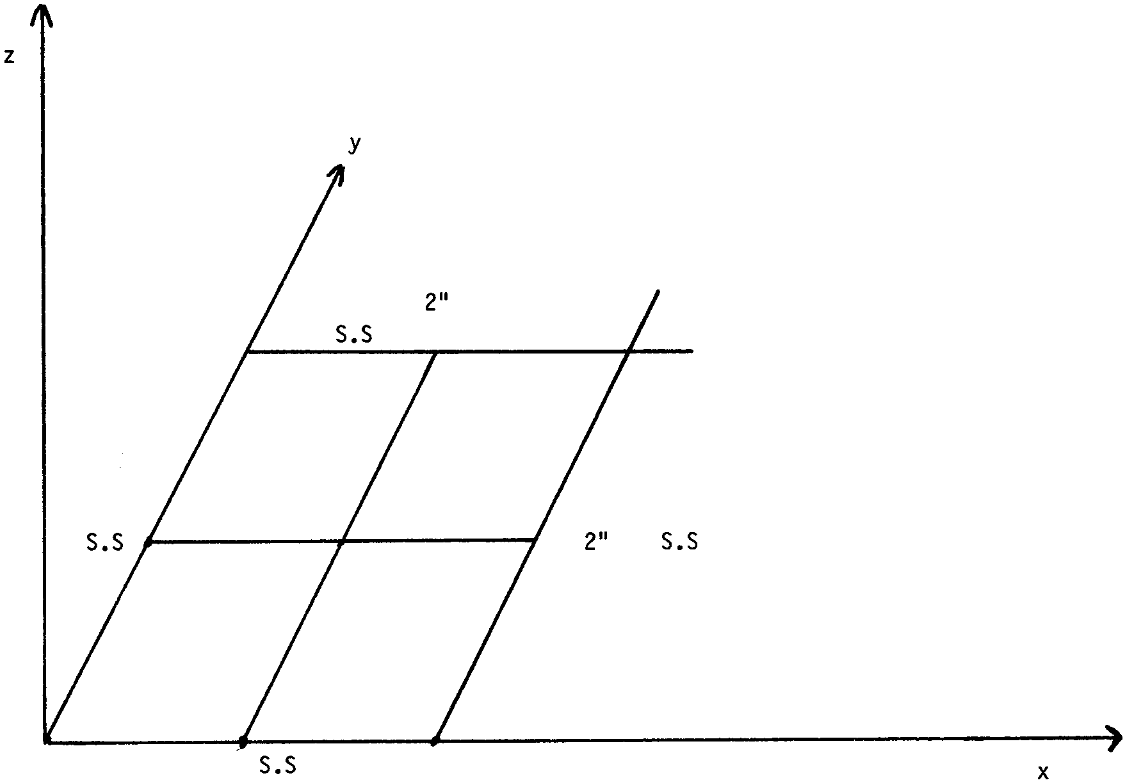


Figure V.3 Simply supported square plate

The plate is divided into four symmetrical sections; for finite element mapping only one quadrant is considered (see Figure V.4). The accuracy of the Finite Element Method solution will be compared with the theoretical solution (see section VI).

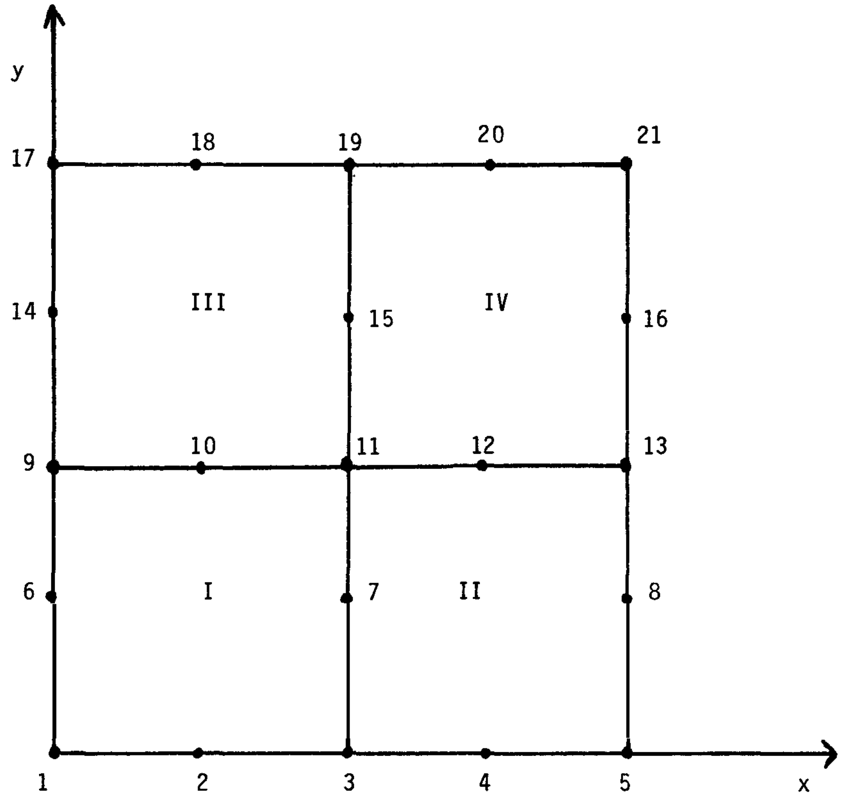


Figure V.4 Mesh generation for simply supported plate

The boundary condition requires that nodes 1 through 5 are restrained from moving in the z-direction, but have rotation about the x-axis. Nodes 1, 6, 9, 14, and 17 are restrained from moving in z-direction, but may have rotation about the y-axis. For nodes 17 through 21 the rotation θx is restrained while nodes 5, 8, 13, 16, and 21 are restrained in rotation about y-axis θy.

The instructions for preparing the input data, for this problem are given in Appendix A.2.

```
TOTAL NO. OF PROBLEH =,     1



PROBLEH NO.    1            ***S  PLY   FPOR  D PL  E***

NPOIN =  21    NELEM =   4    NVFIX =  16    NCASE =   1    NTYPE =   0    NNON =   8    NDOFN =   3

NHATS =   1    NPROP =   4    NGAUS =   2    NDIME =   2    NSTRE =   5    NEVAB =  24

ELEMENT   PROPERTY      NODE NUMBERS
    1         1          1   2   3   7   1   1   1   0   9   6
    2         1          3   4   5   8  13  12  11   7
    3         1          9  10  11  15  19  18  17  14
    4         1         11  12  13  16  21  20  19  15     .

 NODAL POINT COORDINATES
 NODE        X          Y
    1      0.0        0.0
    2      0.250      0.0
    3      0.500      0.0
    4      0.750      0.0
    5      1.000      0.0
    6      0.0        0.250
    7      0.300      0.250
    8      1.000      0.250
    9      0.0        0.500
   10      0.250      0.500
   11      0.500      0.500
   12      0.750      0.500
   13      1.000      0.500
   14      0.0        0.750
   15      0.500      0.750
   16      1.000      0.750
   17      0.0        1.000
   18      0.250      1.000
   19      0.500      1.000
   20      0.750      1.000
   21      1.000      1.000

RESTRAINED NODES
NODE CODE       FIXED  VALUES
  1  111   0.0          0.0          0.0
  2  110   0.0          0.0          0.0
  3  110   0.0          0.0          0.0
  4  110   0.0          0.0          0.0
  5  110   0.0          0.0          0.0
```

```
   6  101  0.0        0.0        0.0
   8  010  0.0        0.0        0.0
   9  101  0.0        0.0        0.0
  13  010  0.0        0.0        0.0
  14  101  0.0        0.0        0.0
  16  010  0.0        0.0        0.0
  17  101  0.0        0.0        0.0
  18  001  0.0        0.0        0.0
  19  001  0.0        0.0        0.0
  20  001  0.0        0.0        0.0
  21  011  0.0        0.0        0.0


  MATERIAL PROPERTIES
 NUMBER,        PROPERTIES
    1           0.109200E+05  0.300000E+00  0.100000E+00  0.100000E+01


 AX FRONTUIDTH ENCOUNTERED =   30

    UNI  RMLY  ISTR  UTED  OAD                                              LOAD CASE =  1
    0

      TOTAL NODAL FORCES FOR EACH ELEMENT
    1      -0.2083E-01  0.0          0.0          0.8333E-01  0.0          0.0          -0.2083E-01  0.0
            0.0         0.8333E-01   0.0          0.0         -0.2083E-01   0.0           0.0         0.8333E-01
            0.0         0.0         -0.2083E-01   0.0          0.0          0.8333E-01   0.0         0.0
    2      -0.2083E-01  0.0          0.0          0.8333E-01  0.0          0.0          -0.2083E-01  0.0
            0.0         0.8333E-01   0.0          0.0         -0.2083E-01   0.0           0.0         0.8333E-01
            0.0         0.0         -0.2083E-01   0.0          0.0          0.8333E-01   0.0         0.0
    3      -0.2083E-01  0.0          0.0          0.8333E-01  0.0          0.0          -0.2083E-01  0.0
            0.0         0.8333E-01   0.0          0.0         -0.2083E-01   0.0           0.0         0.8333E-01
            0.0         0.0         -0.2083E-01   0.0          0.0          0.8333E-01   0.0         0.0
    4      -0.2083E-01  0.0          0.0          0.8333E-01  0.0          0.0          -0.2083E-01  0.0
            0.0         0.8333E-01   0.0          0.0         -0.2083E-01   0.0           0.0         0.8333E-01
            0.0         0.0         -0.2083E-01   0.0          0.0          0.8333E-01   0.0         0.0

     DISPLACEMENTS

     NODE        DISP.           XZ-ROT.          YZ-ROT.
        1   0.0             0.0              0.0
        2   0.0             0.0              0.434907E-01
        3   0.0             0.0              0.784444E-01
        4   0.0             0.0              0.100684E+00
        5   0.0             0.0              0.107817E+00
        6   0.0             0.434907E-01     0.0
        7   0.192101E-01    0.278236E-01     0.703566E-01
        8   0.263735E-01    0.0              0.968956E-01
        9   0.0             0.784445E-01     0.0
       10   0.192103E-01    0.703562E-01     0.278245E-01
       11   0.345671E-01    0.503236E-01     0.503266E-01
       12   0.444602E-01    0.260920E-01     0.650122E-01
       13   0.476145E-01    0.0              0.697700E-01
```

```
   14  0.0           0.100688E+00  0.0
   15  0.444612E-01  0.650116E-01  0.260966E-01
   16  0.613140E-01  0.0           0.364722E-01
   17  0.0           0.107821E+00  0.0
   18  0.263745E-01  0.968974E-01  0.0
   19  0.476161E-01  0.697682E-01  0.0
   20  0.613147E-01  0.364674E-01  0.0
   21  0.657597E-01  0.0           0.0

REACTIONS

NODE      FORCE       XZ-MOMENT     YZ-MOMENT
   1  0.155821E-01 -0.951028E-02 -0.951016E-02
   2 -0.168217E+00 -0.395993E-01  0.0
   3 -0.426499E-01 -0.132917E-01  0.0
   4 -0.259576E+00 -0.125450E-01  0.0
   5 -0.373335E-01 -0.315692E-02  0.0
   4 -0.168232E+00  0.0          -0.396048E-01
   8  0.0          -0.266027E-01  0.0
   9 -0.426250E-01  0.0          -0.132953E-01
  13  0.0          -0.242275E-01  0.0
  14 -0.259594E+00  0.0          -0.125502E-01
  16  0.0          -0.591322E-01  0.0
  17 -0.373372E-01  0.0          -0.315808E-02
  18  0.0           0.0          -0.268093E-01
  19  0.0           0.0          -0.242339E-01
  20  0.0           0.0          -0.591422E-01
  21  0.0          -0.162477E-01 -0.162455E-01

     STRESSES


G,P.  X-COORD.  Y-COORD.   X-MOMENT    Y-MOMENT   XY-MOMENT  XZ-S.FORCE  YZ-S.FORCE

   ELEMENT NO.=     1
  1     0.1057    0.1057  0.10658E-01 0.10658E-01-0.11978E+00 0.96990E-01 0.97037E-01
  2     0.1057    0.3943  0.30701E-01 0.27279E-01-0.94212E-01 0.41021E+00 0.19795E+00
  3     0.3943    0.1057  0.27280E-01 0.30700E-01-0.94212E-01 0.19784E+00 0.41014E+00
  4     0.3943    0.3943  0.88727E-01 0.88722E-01-0.74645E-01 0.19529E+00 0.19525E+00

   ELEMENT NO.=     2
  1     0.6057    0.1037  0.34178E-01 0.41398E-01-0.63023E-01-0.91599E-01 0.50616E+00
  2     0.6057    0.3943  0.10760E+00 0.11741E+00-0.49806E-01 0.18082E+00 0.29490E+00
  3     0.8943    0.1057  0.37728E-01 0.47724E-01-0.17548E-01 0.11380E+00 0.57774E+00
  4     0.0943    0.3943  0.11952E+00 0.13659E+00-0.14133E-01-0.32232E-01 0.33085E+00

   ELEMENT NO.=     3
  1     0.1057    0.6057  0.41401E-01 0.34176E-01-0.63030E-01 0.50607E+00-0.91469E-01
  2     0.1057    0.8943  0.47734E-01 0.37734E-01-0.17549E-01 0.57774E+00 0.11387E+00
  3     0.3943    0.6057  0.11742E+00 0.10760E+00-0.49813E-01 0.29503E+00 0.18089E+00
  4     0.3943    0.8943  0.13661E+00 0.11954E+00-0.14134E-01 0.33114E+00-0.32229E-01

   ELEMENT NO.=     4
  1     0.6057    0.6057  0.14371E+00 0.14371E+00-0.34739E-01 0.16179E+00 0.16205E+00
  2     0.6057    0.0943  0.16739E+00 0.15982E+00-0.95856E-02 0.20816E+00 0.77207E-01
  3     0.8943    0.6057  0.15979E+00 0.16737E+00-0.95881E-02 0.77186E-01 0.20841E+00
  4     0.8943    0.0943  0.18800E+00 0.18802E+00-0.29699E-02 0.44791E-01 0.44949E-01
```

V.C) BAR SUBJECTED TO TEMPERATURE DISTRIBUTION.

Consider a free bar shown in Figure V.5 subjected to a temperature in such a way that $T_1 = 1000°F$ and $T_2 = 500°F$. The temperature distribution for the bar is $t(y) = T_1 - (T_1 - T_2)((2y)/d)^2$.

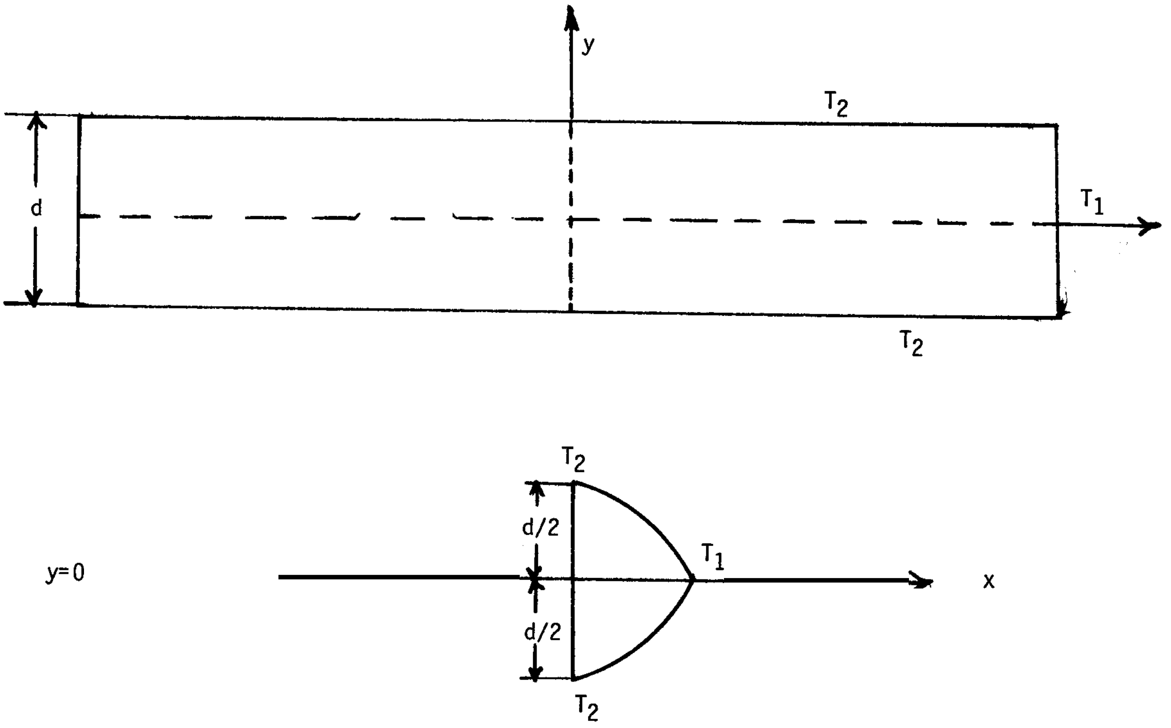


Figure V.5 Free Bar

The temperature distribution is a function of y only, therefore, for the finite element mapping a portion of bar will be considered. This is shown in Figure V.6.

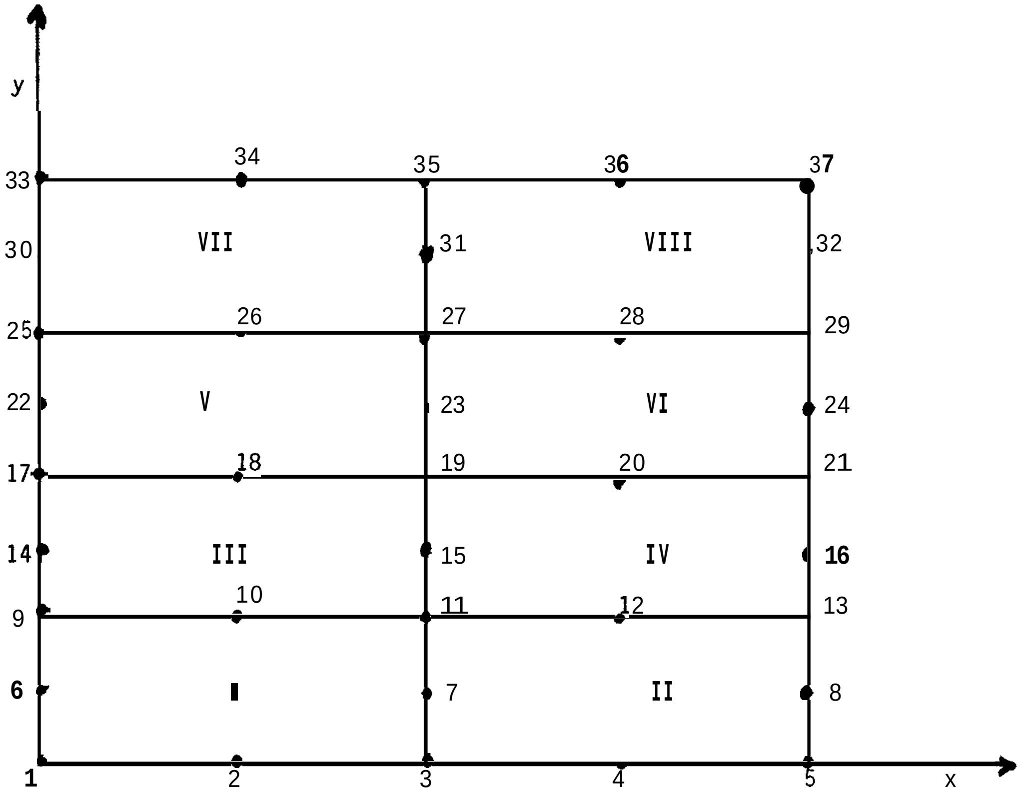


Figure V.6 Mesh generation for long bar

This problem is considered as a plane stress situation with thickness of one unit and the instruction for preparation of input data is given in Appendix A.1. The accuracy of the results obtained by the finite element analysis will be compared with the theoretical (see section VI). The output for this problem is shown on the following pages.

```
      TOTAL NO. OF PROBLEM =,      1



      PROBLEM NO.    1                    FREE   AR U  ER T  PRAT   E

  NPOIN =  37      NELEM =    8      NVFIX =    I      NCASE =    1      NTYPE =    I      NNODE =    e      NDOFN =    2

  NMATS =    I     NPROP =    5      NGAUS =    3      NDIME =    2      NSTRE =    3      NEVAB =  16

  ELEMENT    PRWERTY       NODE NUMBERS
      1         I           1     2     3     7    11    10     9     6
      2         I           3     4     5     8    13    12    11     7
      3         I           9    10    11    15    19    18    17    14
      4         1          It    12    13    16    21    20    19    15
      5         I          17    18    19    23    27    26    25    22
      6         1          19    20    21    24    29    28    27    23
      7         1          25    26    27    31    35    34    33    30
      8         1          27    20    29    32    37    36    35    31


   NODAL POINT COORDINATES
   NODE       X          Y
      1     0.0        0.0
      2     1.250      0.0
      3     2.500      0.0
      4     3.730      0.0
      5     5.000      0.0
      6     0.0        0.125
      7     2.590      0.125
      e     5.000      0.125
      9     0.0        0.250
     10     1.250      0.250
     II     2.500      0.250
     12     3.750      0.230
     13     5.090      0.250
     14     0.0        0.375
     15     2.500      0.375
     16     5.000      0.375
     17     0.0        0.500
     18     1.250      0.500
     19     2.500      0.500
     20     3.750      0. 500
     21     5.000      0. 500
     22     0.0        0.625
     23     2.500      0.625
     24     5.000      0.625
     25     0.0        0.750
     26     1.250      0.750
     27     2.500      0.750
```

```
25   875.000
26   875.000
27   875.000
28   875.000
29   875.000
30   718.750
31   718.750
32   718.750
33   500.000
34   500.000
35   500.000
36   500.000
37   500.000

    TOTAL NODAL FORCES FOR EACH ELEMENT
 1       -0.3051E+04 -0.1984E+05  0.7324E-03 -0.1687E+06  0.3051E+04 -0.1984E+05  0.1696E+05 -0.4464E+05
          0.5283E+04  0.6448E+05  0.7813E-02  0.1687E+06 -0.5283E+04  0.6448E+05 -0.1696E+05 -0.4464E+05
 2       -0.3051E+04 -0.1984E+05 -0.3174E-02 -0.1687E+06  0.3051E+04 -0.1984E+05  0.1696E+05 -0.4464E+05
          0.5283E+04  0.6448E+05  0.3906E-02  0.1686E+06 -0.5283E+04  0.6448E+05 -0.1696E+05 -0.4464E+05
 3       -0.5283E+04 -0.4960E+05  0.1221E-02 -0.2282E+06  0.5283E+04 -0.4960E+05  0.2292E+05 -0.1488E+05
          0.6027E+04  0.6448E+05  0.1172E-01  0.2282E+06 -0.6027E+04  0.6448E+05 -0.2292E+05 -0.1488E+05
 4       -0.5283E+04 -0.4960E+05  0.5127E-02 -0.2282E+06  0.5283E+04 -0.4960E+05  0.2292E+05 -0.1488E+05
          0.6027E+04  0.6448E+05  0.0          0.2282E+06 -0.6027E+04  0.6448E+05 -0.2292E+05 -0.1488E+05
 5       -0.6027E+04 -0.6448E+05  0.6592E-02 -0.2282E+06  0.6027E+04 -0.6448E+05  0.2292E+05  0.1488E+05
          0.5283E+04  0.4960E+05  0.3906E-02  0.2282E+06 -0.5283E+04  0.4960E+05 -0.2292E+05  0.1488E+05
 6       -0.6027E+04 -0.6448E+05  0.2686E-02 -0.2282E+06  0.6027E+04 -0.6448E+05  0.2292E+05  0.1488E+05
          0.5283E+04  0.4960E+05  0.1172E-01  0.2282E+06 -0.5283E+04  0.4960E+05 -0.2292E+05  0.1489E+05
 7       -0.5283E+04 -0.6448E+05  0.4199E-01 -0.1687E+06  0.5283E+04 -0.6448E+05  0.1696E+05  0.4464E+05
          0.3051E+04  0.1984E+05 -0.7813E-02  0.1687E+06 -0.3051E+04  0.1984E+05 -0.1696E+05  0.4464E+05
 8       -0.5283E+04 -0.6448E+05  0.3784E-01 -0.1687E+06  0.5283E+04 -0.6448E+05  0.1696E+05  0.4464E+05
          0.3051E+04  0.1984E+05 -0.7813E-02  0.1687E+06 -0.3051E+04  0.1984E+05 -0.1696E+05  0.4464E+05

   DISPLACEMENTS

   NODE       X-DISP.          Y-DISP.
          0.0
      1 -0.203421E+01 -0.452557E+00
      2 -0.104120E+01 -0.421693E+00
      3  0.439043E-02 -0.426014E+00
      4  0.104998E+01 -0.399825E+00
      5  0.204298E+01 -0.408934E+00
      6 -0.205687E+01 -0.380808E+00
      7  0.329371E-02 -0.356695E+00
      8  0.206348E+01 -0.337186E+00
      9 -0.208455E+01 -0.279492E+00
     10 -0.103987E+01 -0.254712E+00
     11  0.219625E-02 -0.256725E+00
     12  0.104429E+01 -0.232845E+00
     13  0.208899E+01 -0.235873E+00
     14 -0.210559E+01 -0.155387E+00
     15  0.109846E-02 -0.133489E+00
     16  0.210786E+01 -0.111772E+00
     17 -0.211998E+01 -0.217216E-01
```

```
  19   0.0             0.0
  20   0.103999E+01   0.109705E-01
  21   0.212009E+01   0.218887E-01
  22  -0.210775E+01   0.111946E+00
  23  -0.108973E-02   0.133494E+00
  24   0.210569E+01   0.155552E+00
  25  -0.208887E+01   0.236053E+00
  26  -0.104422E+01   0.232924E+00
  27  -0.218045E-02   0.256734E+00
  28   0.103992E+01   0.254784E+00
  29   0.208464E+01   0.279655E+00
  30  -0.206335E+01   0.337371E+00
  31  -0.327298E-02   0.356707E+00
  32   0.205695E+01   0.380969E+00
  33  -0.204285E+01   0.409123E+00
  34  -0.104989E+01   0.399907E+00
  35  -0.436714E-02   0.426029E+00
  36   0.104123E+01   0.421762E+00
  37   0.203428E+01   0.452718E+00

 REACTIONS

 NODE        X-FORCE         Y-FORCE
  19  -0.266699E+01  -0.106641E+01

      STRESSES

G.P.  X-COORD.  Y-COORD.   X-STRESS     Y-STRESS    XV-STRESS    2-STRESS'   MAX P.S.     MIN P.S.      ANGLE

   ELEMENT NO.=     1
  1     0.2010     0.0282 0.23873E+05 0.16721E+04-0.29376E+04 0.0          0.24255E+05 0.12900E+04   -7.4112
  2     0.2818     0.1250 0.87211E+04-0.87450E+03-0.41021E+04 0.0          0.10236E+05-0.23891E+04  -20.2653
  3     0.2010     0.2218-0.94187E+03 0.19763E+04-0.52560E+04 0.0          0.59720E+04-0.49376E+04   37.2426
  4     1.2500     0.0282 0.26420E+05 0.58406E+03 0.12002E+04 0.0          0.26475E+05 0.52842E+03    2.6542
  5     1.2500     0.1250 0.98975E+04-0.21135E+04 0.57532E+03 0.0          0.99250E+04-0.21410E+04    2.7361
  6     1.2500     0.2218-0.11354E+04 0.58575E+03-0.39819E+02 0.0          0.58667E+03-0.11364E+04    1.3246
  7     2.2102     0.0282 0.29630E+05 0.17091E+04 0.24206E+03 0.0          0.29632E+05 0.17070E+04    0.4967
  0     2.2102     0.1250 0.11738E+05-0.11394E+04 0.15809E+03 0.0          0.11740E+05-0.11414E+04    0.7033
  9     2.2102     0.2218-0.66456E+03 0.14096E+04 0.83811E+02 0.0          0.14129E+04-0.66794E+03   -2.3102

   ELEMENT NO.=     2
  1     2.7817     0.0282 0.29631E+05 0.17091E+04-0.24396E+03 0.0          0.29633E+05 0.17069E+04   -0.5066
  2     2.7818     0.1250 0.11739E+05-0.11391E+04-0.15882E+03 0.0          0.11741E+05-0.11410E+04   -0.7064
  3     2.7817     0.2218-0.66212E+03 0.14106E+04-0.83403E+02 0.0          0.14139E+04-0.66547E+03    2.3006
  4     3.7500     0.0282 0.26420E+05 0.58462E+03-0.11995E+04 0.0          0.26476E+05 0.52905E+03   -2.6526
  5     3.7500     0.1250 0.98983E+04-0.21134E+04-0.57515E+03 0.0          0.99258E+04-0.21408E+04   -2.7351
  6     3.7500     0.2218-0.11339E+04 0.58562E+03 0.39314E+02 0.0          0.58652E+03-0.11348E+04   -1.3091
  7     4.7182     0.0282 0.23873E+05 0.16718E+04 0.29376E+04 0.0          0.24255E+05 0.12897E+04    7.4114
  8     4.7182     0.1250 0.87209E+04-0.87575E+03 0.41026E+04 0.0          0.10236E+05-0.23906E+04   20.2654
  9     4.7182     0.2218-0.94131E+03 0.19742E+04 0.52576E+04 0.0          0.59724E+04-0.49394E+04  -37.2513
```

```
 ELEMENT NO.=     3
1     0.2818    0.2782-0.50862E+04 0.32543E+04-0.42143E+04 0.0          0.50129E+04-0.68448E+04    22.6503
2     0.2810    0.3750-0.11068E+05 0.71644E+03-0.30659E+04 0.0          0.14663E+04-0.11818E+05    13.7446
3     0.2818    0.4718-0.11868E+05 0.34842E+04-0.19105E+04 0.0          0.37184E+04-0.12102E+05     6.9882
4     1.2500    0.2782-0.67224E+04-0.78762E+03-0.45390E+03 0.0         -0.75311E+03-0.67569E+04     4.3483
5     1.2500    0.3750-0.13640E+05-0.34396E+04 0.32850E+03 0.0         -0.34290E+04-0.13650E+05    -1.8427
6     1.2500    0.4718-0.15375E+05-0.78687E+03 0.11171E+04 0.0         -0.70183E+03-0.15460E+05    -4.3538
7     2.2182    0.2782-0.62339E+04 0.22535E+04-0.15896E+03 0.0          0.22565E+04-0.62369E+04     1.0726
8     2.2182    0.3750-0.14086E+05-0.51244E+03 0.25867E+03 0.0         -0.50751E+03-0.14091E+05    -1.0914
Y     2.2182    0.4718-0.16755E+05 0.20268E+04 0.68261E+03 0.0          0.20516E+04-0.16780E+05    -2.0787

 ELEMENT NO.=     4
1     2.7017    0.2782-0.62309E+04 0.22541E+04 0.15679E+03 0.0          0.22570E+04-0.62338E+04    -1.0583
2     2.7818    0.3750-0.14081E+05-0.51150E+03-0.25979E+03 0.0         -0.50653E+03-0.14086E+05     1.0964
3     2.7017    0.4718-0.16750E+05 0.20284E+04-0.68264E+03 0.0          0.20532E+04-0.16775E+05     2.0792
4     3.7500    0.2782-0.67201E+04-0.78750E+03 0.45497E+03 0.0         -0.75281E+03-0.67547E+04    -4.3600
5     3.7500    0.3750-0.13636E+05-0.34397E+04-0.32832E+03 0.0         -0.34291E+04-0.13647E+05     1.8423
6     3.7500    0.4718-0.15371E+05-0.78712E+03-0.11180E+04 0.0         -0.70192E+03-0.15456E+05     4.3583
7     4.7182    0.2782-0.50846E+04 0.32527E+04 0.42151E+04 0.0          0.50124E+04-0.68442E+04   -22.6586
8     4.7102    0.3750-0.11066E+05 0.71431E+03 0.30663E+04 0.0          0.14646E+04-0.11817E+05   -13.7497
9     4.7182    0.4718-0.11866E+05 0.34812E+04 0.19110E+04 0.0          0.37156E+04-0.12100E+05    -6.9923

 ELEMENT NO.=     5
1     0.2818    0.5282-0.11869E+05 0.34839E+04 0.19085E+04 0.0          0.37176E+04-0.12102E+05    -6.9809
2     0.2818    0.6250-0.11069E+05 0.71762E+03 0.30641E+04 0.0          0.14666E+04-0.11818E+05   -13.7360
3     0.2818    0.7218-0.50868E+04 0.32561E+04 0.42137E+04 0.0          0.50139E+04-0.68447E+04   -22.6445
4     1.2500    0.5282-0.15375E+05-0.78731E+03-0.11183E+04 0.0         -0.70208E+03-0.15460E+05     4.3584
5     1.2500    0.6250-0.13641E+05-0.34401E+04-0.32925E+03 0.0         -0.34295E+04-0.13651E+05     1.8468
6     1.2500    0.7218-0.67243E+04-0.78819E+03 0.45335E+03 0.0         -0.75376E+03-0.67587E+04    -4.3422
7     2.2102    0.5282-0.16755E+05 0.20280E+04-0.68384E+03 0.0          0.20529E+04-0.16780E+05     2.0823
8     2.2102    0.6250-0.14086E+05-0.51037E+03-0.25946E+03 0.0         -0.50542E+03-0.14091E+05     1.0945
9     2.2102    0.7218-0.62351E+04 0.22561E+04 0.15829E+03 0.0          0.22590E+04-0.62380E+04    -1.0676

 ELEMENT NO.=     6
1     2.7817    0.5282-0.16750E+05 0.20297E+04 0.68494E+03 0.0          0.20546E+04-0.16775E+05    -2.0860
2     2.7810    0.6250-0.14081E+05-0.50900E+03 0.26153E+03 0.0         -0.50396E+03-0.14086E+05    -1.1036
3     2.7817    0.7218-0.62300E+04 0.22573E+04-0.15501E+03 0.0          0.22601E+04-0.62328E+04     1.0460
4     3.7500    0.5282-0.15371E+05-0.78775E+03 0.11191E+04 0.0         -0.70237E+03-0.15456E+05    -4.3629
5     3.7500    0.6250-0.13636E+05-0.34403E+04 0.32925E+03 0.0         -0.34297E+04-0.13646E+05    -1.8477
6     3.7500    0.7218-0.67194E+04-0.78819E+03-0.45421E+03 0.0         -0.75361E+03-0.67540E+04     4.3539
7     4.7102    0.5282-0.11865E+05 0.34814E+04-0.19097E+04 0.0          0.37154E+04-0.12099E+05     6.9877
8     4.7182    0.6250-0.11065E+05 0.71537E+03-0.30650E+04 0.0          0.14651E+04-0.11815E+05    13.7455
9     4.7102    0.7218-0.50821E+04 0.32547E+04-0.42139E+04 0.0          0.50136E+04-0.68409E+04    22.6554

 ELEMENT NO.=     7
1     0.2818    0.7782-0.94400E+03 0.19765E+04 0.52555E+04 0.0          0.59708E+04-0.49383E+04   -37.2359
2     0.2818    0.8750 0.87185E+04-0.87356E+03 0.41016E+04 0.0          0.10233E+05-0.23982E+04    20.2666
3     0.2818    0.9718 0.23870E+05 0.16732E+04 0.29384E+04 0.0          0.24252E+05 0.12909E+04     7.4147
4     1.2500    0.7782-0.11374E+04 0.58681E+03 0.39062E+02 0.0          0.58770E+03-0.11383E+04    -1.2972
5     1.2500    0.8750 0.98942E+04-0.21133E+04-0.57557E+03 0.0          0.99217E+04-0.21408E+04    -2.7361
6     1.2500    0.9718 0.26415E+05 0.58362E+03-0.12004E+04 0.0          0.26471E+05 0.52796E+03    -2.6549
7     2.2182    0.7782-0.66662E+03 0.14111E+04-0.85071E+02 0.0          0.14146E+04-0.67010E+03     2.3407
8     2.2182    0.8750 0.11734E+05-0.11383E+04-0.15901E+03 0.0          0.11736E+05-0.11403E+04    -0.7075
9     2.2182    0.9718 0.29625E+05 0.17089E+04-0.24334E+03 0.0          0.29627E+05 0.17068E+04    -0.4994
```

```
 ELEMENT NO.=     8
1     2.7017     0.7782-0.66150E+03 0.14121E+04 0.84873E+02 0.0           0.14156E+04-0.66497E+03    -2.3399
2     2.7818     0.8750 0.11740E+05-0.11371E+04 0.15968E+03 0.0           0.11742E+05-0.11391E+04     0.7104
3     2.7317     0.9718 0.29631E+05 0.17102E+04 0.24488E+03 0.0           0.29633E+05 0.17080E+04     0.5025
4     3.7500     0.7782-0.11325E+04 0.58612E+03-0.38710E+02 0.0           0.58700E+03-0.11334E+04     1.2896
5     3.7500     0.8750 0.98999E+04-0.21137E+04 0.57558E+03 0.0           0.99274E+04-0.21412E+04     2.7367
6     3.7500     0.9718 0.26422E+05 0.58356E+03 0.12000E+04 0.0           0.26477E+05 0.52795E+03     2.6533
7     4.7102     0.7782-0.93837E+03 0.19751E+04-0.52570E+04 0.0           0.59734E+04-0.49367E+04    37.2557
8     4.7102     0.8750 0.87243E+04-0.87425E+03-0.41021E+04 0.0           0.10239E+05-0.23885E+04   -20.2607
9     4.7182     0.9718 0.23877E+05 0.16731E+04-0.29371E+04 0.0           0.24259E+05 0.12912E+04    -7.4092
```

CHAPTER VI

ACCURACY AND EFFICIENCY OF THE RESULTS

VI.a). **Accuracy of the results**

In this chapter the accuracy of the results found by Finite Element Analysis is compared with the theoretical solution.

First, consider the example of the thick-walled cylindrical pressure vessel subjected to internal pressure.

The theoretical values for radial displacements Ur, radial stress rr, and circumferential stress $\sigma\theta\theta$ are given as (See Figure VI.1):[7]

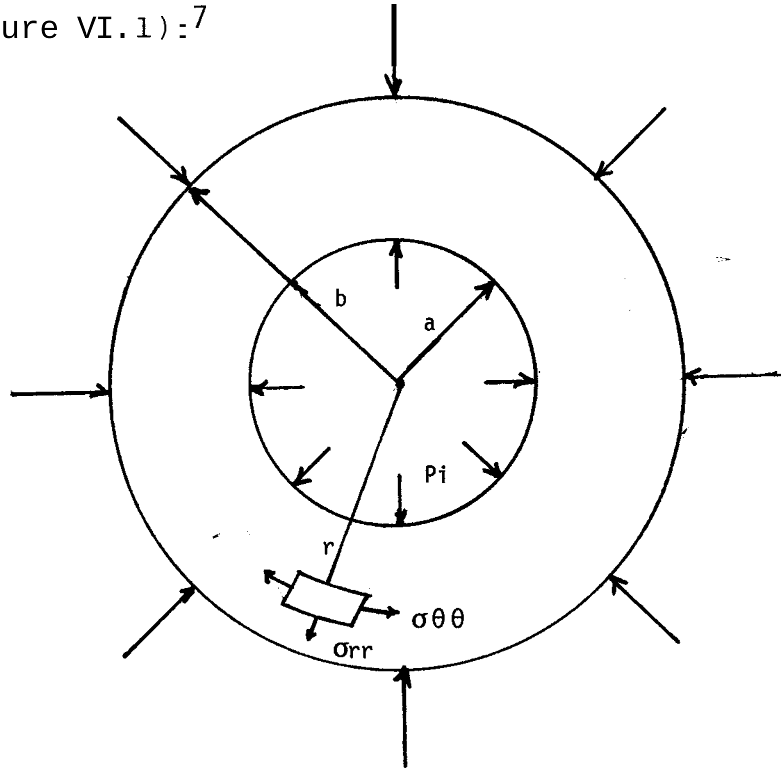


Figure (VI.a) Cross-sectional area for thick-walled pressure vessel

---

[7] Frank A. D'Isa, *Mechanics of Metals* (Reading, Massachusetts: Academic Press, 1977), pp. 109-111.

$$Ur = \frac{r}{2G} \quad \frac{a^2b^2(P0-Pi)}{a^2 - b^2} \circ \frac{1}{r^2} + \frac{Pi\ a^2 - b^2 P_O}{b^2 - a^2} \ (1-2\ )$$

where:

Po = External pressure

Pi = Internal pressure

a,b = Inner and outer radius

r = Radius at any point

ν = Poisson's ratio

For the problem given in section (V.a), Po=0, Pi=30,000psi, a=4", b=8", $\nu=0.3$, and $G=\frac{E}{2(1+\nu)}$ = 11538.46ksi. Substituting these values into Ur results in

$$Ur = 0.43335 \ r \ \frac{64}{r^2} + 0.4 \quad .$$

Radial stress

$$\sigma rr = -Pi \frac{(\frac{b}{r})^2 - 1}{(\frac{b}{a})^2 - 1} - Po \frac{1-(\frac{a}{r})^2}{1-(\frac{a}{b})^2}$$

Substituting Po, Pi, a, and b results in

$$arr = -10,000(\frac{64}{r^2} - 1)$$

The circumfrential stress $\sigma\theta\theta$ is found to be

$$\sigma\theta\theta = Pi \frac{(\frac{b}{r})^2 + 1}{(\frac{b}{a})^2 + 1} - P_0 \frac{1 + (\frac{a}{r})^2}{1 - (\frac{a}{b})^2} \quad .$$

After substitution it will reduce to

$$\sigma\theta\theta = 10,000 \ (\frac{64}{r^2} + 1)$$

Table VI.a shows the results of radial displacement Ur, obtained by Finite Element Analysis and Theory of Elasticity.

	Ur X 10^3					
r	$\theta = 0°$	$\theta = 30°$	$\theta = 60°$	$\theta = 90°$	Theory of Elasticity	Ave. % Error
4"	7.6079	7.5521	7.5530	7.6093	7.627"	0.6
5"	6.4131	6.3364	6.3375	6.4147	6.413"	0.58
6"	5.6904	5.5792	5.5802	5.6921	5.662	0.47
8"	4.8320	4.7199	4.7206	4.8337	4.853	1.57

Table VI.a Comparison of radial displacement Ur for the case of pressure vessel subjected to internal pressure

Table VI.b shows the radial and hoop stresses obtained by Finite Element Analysis and Theory of Elasticity.

r, in	σ_{rr} FE psi	$\sigma_{\theta\theta}$ FE psi	urr, Theory psi	$\sigma_{\theta\theta}$ Theory psi	σ_{rr} % Error	$\sigma_{\theta\theta}$ % Error
4.1048"	-26981.0	46634	-27983.6	47983.6	3.58	2.81
5.1026	-13720.0	33770	-14580.8	34550.8	5.89	2.34
6.2132	-7021.0	26471	-6578.6	26578.6	6.3	0.40
7.7594	-709.0	22716	-629.7	20629.7	11.28	9.18

Table VI.b Comparison of radial and circumferential stress for the case of pressure vessel subjected to interal pressure

Next, consider the simply supported plate subjected to a uniformly distributed loading of 1 lb/in^2.

The theoritical values of maximum displacement and maximum moment are given as follows.

$$W_{MAX} = 0.00406\frac{qa^4}{D} \quad ,$$

where

q = lateral loading

a = side length

$$D = \frac{Et^3}{12(1-\nu^2)} = \text{Flexural Rigidity}$$

$$D = \frac{(10920)(0.1)^3}{12(1-.3^2)} = 1.0 \quad k.in$$

Substituting these values into W_{MAX} results in

$$W_{MAX} = -\ 0.06496 \text{ in}$$

$$Mxx = Myy = 0.0479qa^2$$

or

$$Mxx = Myy = -\ .1916\frac{k\ in}{in}$$

Table VI.c shows the results that are obtained by Finite Element Analysis and by the Theory of Elasticity.

	FINITE ELEM. ANALYSIS	**THEORY OF ELASTICITY**	**% ERROR**
W_{MAX}, in	0.06575	0.06496	1.20
Mxx = Myy, $\frac{k\ in}{in}$	0.1880	0.1916	1.87

Table VI.c Comparison of maximum deflection and moment for the case of simply supported plate bending

Finally, consider a free bar subjected to a temperature as shown in Fig. V.5. An approximate solution could be obtained for this problem by assuming that temperature depends on the y direction only. Therefore the stress σx and deflections Ur and V can be written as[8]

$$\sigma x = 4\alpha E(T_1 - T_2)\left(\frac{y}{d}\right)^2 - \frac{1}{3}(T_1 - T_2)\alpha E$$

$$Ur = \left[-\frac{\alpha}{3}(T_1 - T_2) + \alpha T_1\right]x$$

$$V = -\frac{4}{3}\alpha(T_1 - T_2)\left(\frac{y^3}{d^2}\right)(1+\nu)+\alpha\left[\frac{1}{3}(T_1-T_2)+T_1\right]y$$

where

α = Thermal expansion coefficient

E = Modulus of Elasticity

T_2, T_1 = Temperatures at the outside and center line of the bar, respectively.

d = width of the bar.

The numerical values are given in section V.c. Therefore, after substitution, the equations reduce to

$$\sigma x = 200{,}000\left(\frac{y}{d}\right)^2 - 16666.67$$

$$Ur = (0.8334)x$$

$$V = -86667\left(\frac{y^3}{d^2}\right) + (1.05)y \quad .$$

Tables VI.d.1 and VI.d.2, are presented to compare the values of ox, Ur, and V obtained by Finite Element Analysis with the Theory of Elasticity.

---

[8] A. P. Boresi and P. O. Lynn, <u>Elasticity in Engineering Mechanics</u> (Englewood Cliffs, New Jersey: Prentice-Hall, 1974), pp. 400-402.

Co-ordinates for Analytical Equation		Co-ordinates for F-E Analysis		F-E	Theory	
X	Y	X	Y	x, PSI	x. PSI	% Error
0.0	0.0	2.5	2.5	-16755	-16666.6	0.52
0.0	0.125	2.5	0.625	-14086	-13541.6	3.86
0.0	0.375	2.5	1.875	11738	11458.3	2.38
0.0	0.5	2.5	1.0	29630	33333.34	11.1

Table VI.d.1

Co-ordinates for Analytical Equations		Co-ordinates for F-E Analysis		Finite Eelement Analysis		Theory of Elasticity		V
X	Y	X	Y	U, in	V, in	U, in	V, in	% Error
0.0	0.0	2.5	2.5	0.0	0.0	0.0	0.0	0.0
0.0	0.125	2.5	0.625	0.001098	-0.13348	0.0	0.12955	2.9
0.0	0.375	2.5	0.875	0.00329	-0.35669	0.0	0.3480	2.4
0.0	0.5	2.5	1.0	0.00439	-0.42601	0.0	0.4166	2.2

Table VI.d.2

Comparison of deflections and stress for the case
of long bar subjected to temperature

VI.b) Efficiency of the results

In this section, the efficiency of the results using the Frontal Solution Technique in Finite Element programming will be discussed. This is done by comparing the results obtained by the program given in Appendix B with the SAPIV. SAPIV is a Finite Element Program which uses the banded solution technique to perform the assembly and solution of the overall stiffness matrix.

Table VI.e shows the execution time for the case of the thick-walled cylinderical pressure vessel. This example is performed earlier in section V.a by using 9 parabolic isoparametric elements (See Figure V.2), while 24 quadrilateral elements are used in SAPIV in order to get an exact result.

	EXECUTION TIME, SEC	**% ERROR U**
24 ELEMENT SAPIV	1.91	0.87
9 ELEMENT FRONTAL TECHNIQUE	1.31	0.25

Table VI.e Efficiency of results for the case of pressure vessel

Table VI.f shows the execution time for the case of the simply supported plate. The results of this example are given in section V.b.

	EXECUTION TIME, SEC	% ERROR W_{MAX}
4-ELEMENT BAND TECHNIQUE, SAPIV	1.10	12.17
8-ELEMENT BANDED TECHNIQUE, SAPIV	1.15	9.51
16-ELEMENT BANDED TECHNIQUE, SAPIV	2.47	3.07
4-ELEMENT FRONTAL TECHNIQUE	0.86	1.20

Table VI.f Efficiency of results for the case of plate bending

CHAPTER VII

CONCLUSION

The Finite Element analysis is a tool to solve engineering problems. Appendix B contains a Finite Element Program that is capable of solving two dimensional plane stress or plane strain and plate bending problems. There are several Finite Element programs written to solve engineering problems, but the economical limitations by computer costs restrict the use of such programs. As a result, analysts are developing new equations solution techniques such as the Frontal Technique to reduce the costs of running Finite Element programs.

Several examples performed in sections V and VI compare the accuracy as well as the efficiency of the Frontal Techniques in Finite Element Programming.

The first example is the thick-walled cylindrical pressure vessel subjected to internal pressure. Table (VI.a) shows the results of radial displacement Ur for different angles and radii. The results are then compared with analytical values found by the Theory of Elasticity. The percentage error is less than 2.0%. Table VI.b shows the results of radial and hoop stress for different radii and it is in good agreement with the theory.

Since the integration of the stiffness matrix is done numerically, the stresses are computed at the sampling points (See Figure II.a). For this example the integration is performed using the 3-point rule, thus, there are nine sampling points for an element, as is shown in the computer output.

The second example is the Simply Supported Plate. Table (VI.c) shows the results of maximum deflection and moment, compared with the analytical values. The percentage error is less than 2%, which is in good agreement with the approximated solution. Table VI.f shows the efficiency of the FRONTAL technique in which the execution time is 0.86 seconds while using SAPIV is 2.47 seconds.

The third example is the Free Bar subjected to a temperature distribution. Table VI.d.l shows the results of stresses. The percentage error is increasing as y increases from the center of the bar. This error could be improved by increasing the mesh size in the region remote to the center of the bar.

Table VI.d.2 shows the results of deflections and the percentage error is less than 3.5.

APPENDIX A

PREPARATION OF INPUT DATA

This section describes the preparation of input data for a given problem. As stated earlier this program is capable of solving plane stress or plane strain and plate bending problems. The order of data preparation for each case is listed below.

A.l). **Plane Stress or Strain Program**

I). Problem Card - One Card

Columns	**Variable**	**Entry**
1-5	NPROB	Total number of problems to be solved in one run.

II). Title Card

Columns	**Variable**	**Entry**
1-72	---	Title of the problem

III). Control Card - One Card

Columns	**Variable**	**Entry**
1-5	NPOIN	Total number nodal points.
6-10	NELEM	Total number of elements
11-15	NVFIX	Total number of restrained boundary points, where one or more degrees of freedom are restrained.
16-20	NCASE	Total number of load cases to be analysed.
21-25	TYPE	Problem type parameter: 1 - plane stress 2 - plane strain
26-30	NNODE	Number of nodes per element (=8)
31-35	NDOFN	Number of degrees of freedom per node=2
36-40	NMATS	Total number of different materials.
41-45	NPROP	Number of independent properties per material = 5
46-50	NGAUS	Order of integration formula for numerical integration
51-55	NDIME	Number of Co-ordinate dimensions= 2 .
56-60	NSTRE	Number of independent stress components = 3

Columns Variable Entry

IV). ELEMENT CARD. One card for each element counter-clockwise sequence. For example, consider the element shown in Figure 11.1. The node numbering is (1, 2, 3, 4, 5, 6, 7, 8).

1-5	NUMEL	Element number
6-10	MATNO(NUMEL)	Material property number
11-15	LNODS(NUMEL,1)	1st Nodal connection number
16-20	LNODS(NUMEL,2)	2nd Nodal connection number
"	"	"
"	"	"
"	"	"
46-50	LNODS(NUMEL,8)	8th Nodal connection number

V). NODE CARD. One card for each nodal point. This data card contains a list of nodal co-ordinates. The co-ordinates of highest node should be inputed.

1-5	IPOIN	Nodal point number
6-15	COORD(IPOIN,1)	x-Co-ordinate of node
16-25	COORD(IPOIN,2)	y-Co-ordinate of node.

VI). RESTRAINED NODE CARD. One card for each restrained node should be considered. 1 designates nodal displacement restrained and 0 designated no displacement restrainment.

2-5	NOFIX	Restrained node number
9	IRPRE1	Restraint on x-displacement EQ.0; Free GT.1; Fixed.
10	IFPRE2	Restraint on y-displacement EQ.0; Free GT.1; Fixed
11-20	PRESC1	The prescribed value of x-displacement
21-30	PRESC2	The prescribed value of y-displacement

VII). MATERIAL CARDS. One card for each different material.

1-5	NUMAT	Material Identification Number
6-15	PROPS(NUMAT,1)	Elastic nodulus, E.
16-25	PROPS(NUMAT,2)	Poisson's ratio, v
26-35	PROS(NUMAT,3)	Material thickness, t
36-45	PROS(NUMAT,4)	Mass Density, P
46-55	PROS(NUMAT,5)	Coefficient of thermal expansion

Columns Variable Entry

VIII). LOAD CARD. This data card contains four (4) different kinds of loading, for the plane stress or plain strain case. The input data for each loading is listed below.

VIII.a). TITLE CARD. One card.

1-72		TITLE of the loading

VIII.b). CONTROL CARD. One card.

1-5	IRLOD	Applied point loading 0 No point loading 1 Point loading aplied.
6-10	IGRAV	GRAVITY Loading 0 No gravity loading 1 Gravity loading is considered
11-15	IEDGE	Distributed loading 0 No distributed loading 1 Distributed loading is considered
16-20	ITEMP	Thermal loading 0 No thermal loading 1 Thermal loading is considered

VIII.c). APPLIED LOADING. One card for each nodal loading.

1-5	LODPT	NODE NUMBER
6-15	POINT(1)	Loading in x-direction
16-25	POINT(2)	Loading in y-direction

VIII.d). GRAVITY LOADING. One Card.

1-10	THETA	Angle of gravity axes from the positive y axes.
11-20	GRAVY	GRAVITY Constant

VIII.e). DISTRIBUTED LOADING. This data card is arranged in three different classifications.

a). Control Card

1-5	NEDGE	Number of edge to which loading is applied.

Columns	**Variable**	**Entry**

b). Element Face Topology Card

1-5	NEASS	The element number to which the loading is associated.
6-10	NOPRS(1)	List of nodal points in the
11-15	NOPRS(2)	counter-clockwise order that
16-20	NOPRS(3)	forms the element face on which the loading acts.

c). Distribution Load Card

1-10	PRESS(1,1)	Normal component of distributed load at node NOPRS(1).
11-20	PRESS(2,1)	Normal component of distributed load at node NOPRS(2)
21-30	PRESS(3,1)	Normal component of distributed load at node NOPRS(3)
31-40	PRESS(1,2)	Tangential component at node NOPRS(1)
41-50	PRESS(2,2)	Tangential component at node NOPRS(2)
51-60	PRESS(3,2)	Tangential component at node NOPRS(3)

IX). THERMAL LOADING.

1-5	NODPT	Node number
6-15	TEMPE	Temperature at a node.

A.2). **The Plate Bending Program.** This section prepares the input data for the case of Plate Bending.

I). PROBLEM CARD. One card.

1-5		Total number of problems to be solved in one run.

II). TITLE CARD. One card.

1-72		Title of the program.

III). CONTROL CARD. One Card.

1-5	NPOIN	Total number of nodal points.
6-10	NELEM	Total number of elements
11-15	NVFIX	Total number of restrained points
16-20	NCASE	Total number of load cases.
21-25	NTYPE	BLANK
16-30	NNODE	Number of nodes per element = 8
31-35	NODFN	Number of degrees of freedom per node = 3

Columns	**Variable**	**Entry**
37-40	NMATS	Total number of different materials
41-45	NPROP	Number of independent properties per material = 4
46-50	NGAUS	Order of numerical integration for this case is = 2
51-55	NDIME	Number of Co-ordinate dimensions = 2
56-60	NSTRE	Number of stress components = 5

IV). ELEMENT CARDS. One card for each element.

The node number for each element must be in counter-clockwise order.

1-5	NUMEL	Element number
6-10	NATNO(NUMEL)	Material property number
11-15	LNODS(NUMEL,1)	1st Nodal connection number
16-20	LNODS(NUMEL,2)	2nd Nodal connection number
"	"	"
"	"	"
46-50	LNODS(NUMEL,8)	8th Nodal connection number

V). NODAL CO-ORDINATE CARDS. One card for each node. The co-ordinates of the highest node must be inputed.

1-5	IPOIN	Nodal point number
6-15	COORD(IPOIN,1)	X-Co-ordinate of node
16-25	COORD(IPOIN,2)	Y-Co-ordinate of node

VI). RESTRAINED NODE CARDS. One card for each restrained node.

2-5	NOFIX	Restrained node number
8	IFPRE1	Condition of restraint on nodal displacement, w EQ;0 Free GT;1 Fixed
	IFPRE2	Condition of restraint on nodal rotation, x EQ.0; Free GT.1; Fixed
10	IFPRE3	Condition of restraint on nodal rotation, y EQ.0; Free GT.1; Fixed

Columns	**Variable**	**Entry**
11-20	PRESC 1	The prescribed nodal displacement, w
21-30	PRESC2	The prescribed nodal displacement, θx
31-40	PRESC3	The prescribed nodal displacement, θy.

VII). MATERIAL CARDS. One card for each different material.

1-5	NUMAT	Material identification number
6-15	PROPS(NUMAT,1)	Elastic nodulus, E.
16-25	PROS(NUMAT,2)	Poisson's Ratio, ν
26-35	PROS(NUMAT,3)	Material thickness, t
36-45	PROS(NUMAT,4)	Intensity of uniformly distributed loads.

VIII). TITLE OF LOAD CARD. One card.

1-72		Title of load case.

IX). LOAD CONTROL CARD. One card.

1-5	IPLOD	Applied point load. EQ.0; Free GT.l; Applied nodal load to be input.

X). APLLIED LOAD CARDS. One card for each loaded nodal point. The last card must be for the highest node number.

1-5	LODPT	Node number
6-15	POINT(1)	Load component in Z direction
16-25	POINT(2)	Nodal couple in XZ plane
26-35	POINT(3)	Nodal couple in YZ plane.

APPENDIX B

FINITE ELEMENT PROGRAM

B.1 Plate Bending Program

```
C       PROGRAM BOKI(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,
C      *TAPE2,TAPE2,TAPE3,TAPE4)
        DIMENSION TITLE(12)
        COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
       *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
       *NEVAB,ICASE, NCASE,ITEMP, IPROB,NPROB
        COMMON/LGDATA/COORD(80,2),PROPS(10,4),
       *PRESC(40,3),ASDIS(240),ELOAD(25,24),
       *NOFIX(40),IFPRE(40,3),LNODS(25,8),
       *MATNO(25)
        COMMON /WORK/ELCOD(2,8),SHAPE(8),
       *DERIV(2,8),DMATX(5,5),CARTD(2,8),
       *DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
       *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
        READ(5,900) NPROB
  900   FORMAT(I S)
        WRITE(6,905) NPROB
  905   FORMAT(1H0,5X,23HTOTAL NO, OF PROBLEM =,I5)
        DO 20 IPROB=1,NPROB
        REWIND 1
        REWIND 2
        REWIND 3
        REWIND 4
        READ(5,910) TITLE
  910   FORMAT(12A6)
        WRITE(6,915) IPROB,TITLE
  915   FORMAT(/////,6X,12HPROBLEM NO, ,I3,10X,12A6)
C
C*** CALL SUBROUTIN THAT READ MOST OF THE DATA
C
        CALL INPUT
C
C*** CREAT ELEM SSTIFNESS
C
        CALL STIFPE
        DO 10 ICASE=1,NCASE
C
C**** COMPUT LOAD,
C
C
        CALL LOADPB
C
C*** SOLVE THE RESULTING EQ
C
        CALL FRONT
C
C*** COMPUTE STRESS IN. ALL ELEMENTS.
```

```
C
      CALL STREPB
   10 CONTINUE
   20 CONTINUE
      STOP
      END
```

```
      SUBROUTINE INPUT
C
C*** READ THE FIRST DATA CARD, AND ECHO IT IMMEDIATELY
C
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,NEVAB,
     *ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24),
     *NOFIX(40),IFPRE(40,3),LNODS(25,8),
     *MATNO(25)
      COMMON/WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),DBMAT(5,24)
     *,BMATX(5,24),SMATX(5,24,4),POSGP(2),
     *WEIGP(2),GPCOD(2,4),NEROR(24)
      READ(5,900) NPOIN,NELEM,NVFIX,NCASE,NTYPE,
     *NNODE,NDOFN,NMATS,NPROP,NGAUS,NSTRE
  900 FORMAT(12I5)
      NEVAB=NDOFN*NNODE
      WRITE(6,905) NPOIN,NELEM,NVFIX,NCASE,NTYPE,NNODE,NDOFN,NMATS,
     *NPROP,NGAUS,NDIME,NSTRE,NEVAB
  905 FORMAT(//8H NPOIN =,I4,4X,8H NELEW =,I4,
     * 4X,8H NVFIX =,I4,4X,8H NCASE =,I4,4X,
     * 8H NTYPE =,I4,4X,8H NNODE =,I4,4X,
     * 8H NDOFN =,I4// 8H NMATS =,I4,4X,
     * 8H NPROP =,I4,4X,8H NGAUS =,I4,4X,
     * 8H NDIME =,I4,4X,8H NSTRE =,I4,4X,
     * 8H NEVAB =,I4)
      CALL CHECK1
C
C*** READ THE ELEMENT CONNECTION, AND THE PROPERTY NO.
C
C
      WRITE(6,910)
  910 FORMAT(//8H ELEMENT,3X,8HPROPERTY,6X,12HNODE NUMBERS)
      DO 10 IELEM=1,NELEM
      READ(5,900) NUMEL,MATNO(NUMEL),(LNODS(NUMEL,INODE),INODE=1,
     *NNODE)
   10 WHITE(6,915) NUMEL,MATNO(NUMEL),(LNODS(NUMEL,INODE),INODE=1,
     *NNODE)
  915 FORMAT(1X,I5,I9,6X,8I5)
```

```
C*** ZERO ALL NODE COO
C
C
      DO 20 IPOIN=1,NPOIN
      DO 20 IDIME=1,NDIME
   20 COORD(IPOIN,IDIME)=0.0
C
C*** READ SOME NODAL COORD
C
C
      WRITE(6,920)
  920 FORMAT(//25H  NODAL POINT COORDINATES)
      WRITE(6,925)
  925 FORMAT(6H  NODE,7X,1HX,9X,1HY)
   30 READ(5,930) IPOIN,(COORD(IPOIN,IDIME), IDIME=1,NDIME)
  930 FORMAT(I5,5F10.5)
      IF(IPOIN.NE.NPOIN) GO TO 30
C
C*** INTERPOLATE COORD OF MID-SIDE NODES
C
C
      IF(NDIME.EQ.1) GO TO 40
      CALL NODEXY
   40 CONTINUE
      DO 50 IPOIN=1,NPOIN
   50 WRITE(6,935) IPOIN,(COORD(IPOIN,IDIME), IDIME=1,NDIME)
  935 FORMAT(1X,I5,3F10.3)
C
C*** READ THE FIX VALUES,
C
      WHITE(6,940)
  940 FORMAT(//17H RESTRAINED NODES)
      WRITE(6,945)
  945 FORMAT(5H NODE, 1X,4HCODE,6X, 12HFIXED VALUES)
      IF(NODFN.NE.2) GO TO 70
      WRITE(4,999) NDOFN
  999 FORMAT(10X,I S)
      DO 40 IVFIX=1,NVFIX
      READ(5,950) NOFIX(IVFIX),(IFPRE(IVFIX, IDOFN),IDOFN=1,NDOFN),
     * (PRESC(IVFIX,IDOFN),IDOFN=1,NDOFN)
   60 WRITE(6,950) NOFIX(IVFIX),(IFPRE(IVFIX,IDOFN),IDOFN=1,NDOFN),
     *(PRESC(IVFIX,IDOFN),IDOFN=1,NDOFN)
  950 FORMAT(1X,I4,3X,2I1,2F10.6)
      GO TO 90
   70 WRITE(6,888) NDOFN
  888 FORMAT(10X,I5)
```

```
      DO 80 IVFIX=1,NVFIX
      READ(5,955) NOFIX(IVFIX),(IFPRE(IVFIX,IDOFN),IDOFN=1,NDOFN),
     *(PRESC(IVFIX,IDOFN),IDOFN=1,NDOFN)
   80 WRITE(6,955) NOFIX(IVFIX),(IFPRE(IVFIX,IDOFN),IDOFN=1,NDOFN),
     *(PRESC(IVFIX,IDOFN),IDOFN=1,NDOFN)
  955 FORMAT(1X,I4,2X,3I1,3F10.6)
   90 CONTINUE
C
C*** READ THE AVALAELE SELECTION OF ELEMENT PROPERTY
C
C
      WHITE(6,960)
  960 FORMAT(//21H  MATERIAL PROPERTIES)
      WRITE(6,965)
  965 FORMAT(8H NUMBER,7X,10HPROPERTIES)
      DO 100 IMATS=1,NMATS
      READ(5,930) NUMAT,(PROPS(NUMAT,IPROP),IPROP=1,NPROP)
  100 WRITE(6,970) NUMAT,(PROPS(NUMAT,IPROP),IPROP=1,NPROP)
  970 FORMAT(1X,I5,7X,5E14.6)
C
C*** SET UP GAUSSION INTEGRATION CONSTS
C
C
      CALL GAUSSQ
      CALL CHECK2
      RETURN
      END

      SUBROUTINE NODEXY
      COMMON/CONTRO/NPOIN, NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,NEVAB,
     3ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),PRESC(40,3),
     *ASDIS(240),ELOAD(25,24),NOFIX(40),IFPRE(40,3),
     *LNODS(25,8),MANTO(25)
      COMMON/WORK/ELCOD(2,8),SHAPE(8),DERIV(2,8),DMATX(5,5),
     *CARTD(2,8),DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
C
C*** LOOP OVER EACH ELEMENT
C
      DO 30 IELEM=1,NELEM
C
C*** LOOP OVER EACH ELEMENT EDGE
C
C
      DO 20 INODE=1,NNODE,2
C
C**** COMPUT THE NODE NUMBER
C
```

```
      NODST=LNODS(IELEM,INODE)
      IGASH=INODE+2
      IF(IGASH.GT.NNODE) IGASH=1
C
C*** COMPUT THE NODE NUMBER
C
      NODFN=LNODS(IELEM,IGASH)
      MIDPT=INODE+1
C
C*** COMPUT INTEMEDIATE' NODE
C
      NODMD=LNODS(IELEM,MIDPT)
      TOTAL=ABS(COORD(NODMD,1))+ABS(COORD(NODMD,2))
C
C***
C
      IF(TOTAL.GT.0.0) GO TO 20
      KOUNT=1
   10 COORD(NODMD,KOUNT)=(COORD(NODST,KOUNT)+COORD(NODFN,KOUNT))/2.0
      KOUNT=KOUNT+1
      IF(KOUNT.EQ.2) GO TO 10
   20 CONTINUE
   30 CONTINUE
      RETURN
      END


      SUBROUTINE GAUSSQ
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,NEVAB,
     *ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),PRESC(40,3),
     *ASDIS(240),ELOAD(25,24),NOFIX(40),IFPRE(40,3),
     *LNODS(25,8),MANTO(25)
      COMMON/WORK/ELCOD(2,8),SHAPE(8),DERIV(2,8),
     *DMATX(5,5),CARTD(2,8),DBMAT(5,24),BMATX(5,24),
     *SMATX(5,24,4),POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
      IF(NGAUS.GT.2) GO TO 10
      POSGP(1)=-0.577350269189626
      WEIGP(1)=1.0
      GO TO 20
   10 POSGP(1)=-0.774596669241483
      POSGP(2)=0.0
      WEIGP(1)=0.555555555555556
      WEIGP(1)=0.888888888888889
```

```
   20 KGAUS=NGAUS/2
      DO 30 IGASH=1,KGAUS
      JGASH=NGAUS+1-IGASH
      POSGP(JGASH)=-POSGP(IGASH)
      WEIGP(JGASH)=WEIGP(IGASH)
   30 CONTINUE
      RETURN
      END

      SUBROUTINE STIFPB
C
C*** CALCULATES ELEM STIFNESS MATRIX
C
C
      DIMENSION ESTIF(24,24)
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,NSTRE,NTYPE,NGAUS,
     *NPROP,NMATS,NVFIX,NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(30,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24),
     *NOFIX(40),IFPRE(40,3),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
C
C*** LOOP EACH ELEM
C
      DO 70 IELEM=1,NELEM
      LPROP=MATNO(IELEM)
C

C**** CALCUL THE COORD OF THE NODAL POINT
C
      DO 10 INODE=1,NNODE
      LNODE=LNODS(IELEM,INODE)
      DO 10 IDIME=1,NDIME
      ELCOD(IDIME,INODE)=COORD(LNODE,IDIME)
   10 CONTINUE
C
C*** INITIALIZ STINESS MATRIX
C
      DO 20 IEVAB=1,NEVAB
      DO 20 JEVAB=1,NEVAB
      ESTIF(IEVAB,JEVAB)=0.0
   20 CONTINUE
C
C**** CALCUL MATRIX
```

```
C
      CALL MODPB(LPROP)
      KGASP=0
C
C***  ENTER LOOP FOR NUMERICAL INTIGRATION
C
      DO 50 IGAUS=1,NGAUS
      EXISP=POSGP(IGAUS)
      DO 50 JGAUS=1,NGAUS
      ETASP=POSGP(JGAUS)
      KGASP=KGASP+1
C
C*** EVALUATE SHAPE FN
C
C
      CALL SFR2(EXISP,ETASP)
      CALL JACOB2(IELEN,DJACB,KGASP)
      DAREA=DJACB*WEIGP(IGAUS)*WEIGP(JGAUS)
C
C*** EVALUATE THE B AND DB MATRIX
C
      CALL BMATPB
      CALL DBE
C
C**** CALCUL THE STIFFNESS
C
      DO 30 IEVAB=1,NEVAB
      DO 30 JEVAB=IEVAB,NEVAB
      DO 30 ISTRE=1,NSTRE
      ESTIF(IEVAB,JEVAB)=ESTIF(IEVAB,JEVAB)+BMATX(ISTRE,IEVAB)*DBMAT
     * (ISTRE,JEVAB)*DAREA
   30 CONTINUE
C
C*** STORE COMPONENT OF DB MATRIX
C
      DO 40 ISTRE=1,NSTHE
      DO 40 IEVAB=1,NEVAB
      SMATX(ISTRE,IEVAB,KGASP)=DBMAT(ISTRE,IEVAB)
   40 CONTINUE
   50 CONTINUE
C
C*** CONSTRUCT THE LOWER TRIANGEL
C
      DO 60 IEVAB=1,NEVAB
      DO 60 JEVAB=1,NEVAB
      ESTIF(JEVAB,IEVAB)=ESTIF(IEVAB,JEVAB)
   60 CONTINUE
```

```
C
C**** STORE THE STIFFNESS MATRIX
C
      WRITE(1) ESTIF
      WHITE(3) SMATX,GPCOD
   70 CONTINUE
      RETURN
      END


      SUBROUTINE MODPB(LPROP)
C
C*** CALCULATES MATRIX RIGIDITIES FOR PLATE BENDING
C
C
      COMMON/CONTRO/NPOIN, NELEM, NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP, IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOADI25,24),
     *NOFIX(40),IFPRE(40,3),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
      DO 10 ISTRE=1,NSTRE
      DO 10 JSTRE=1,NSTRE
      DMATX(ISTRE,JSTRE)=0.0
   10 CONTINUE
      YOUNG=PROPS(LPROP,1)
      POISS=PROPS(LPROP,2)
      THICK=PROPS(LPROP,3)
      DMATX(1,1)=YOUNG*THICK*THICK*THICK
     */(12.0*(1.0-POISS*POISS))
      DMATX(1,2)=POISS*DMATX(1,1)
      DMATX(2,2)=DMATX(1,1)
      DMATX(2,1)=DMATX(1,2)
      DMATX(3,3)=(1.0-POISS)*DMATX(1,1)/2.0
     .DMATX(4,4)=YOUNG*THICK/(2.4*(1.0+POISS))
      DMATX(5,5)=DMATX(4,4)
      RETURN
      END
```

```
      SUBROUTINE SFR2(S,T)
C
C*** CALCULATE SHAPE FUNCTION
C
C
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24),
     *NOFIX(40),IFPRE(40,3),LNODS(45,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),DBMAT(5,24),
     *BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
      S2=S*2.0
      T2=T*2.0
      SS=S*S
      TT=T*T
      ST=S*T
      SST=S*S*T
      STT=S*T*T
      ST2=S*T*2.0
C
C*** SHAPE FUNCTOIN
C
      SHAPE(1)=(-1.0+ST+SS+TT-SST-STT)/4.0
      SHAPE(2)=(1.0-T-SS+SST)/2.0
      SHAPE(3)=(-1.0-ST+SS+TT-SST+STT)/4.0
      SHAPE(4)=(1.0+S-TT-STT)/2.0
      SHAPE(5)=(-1.0+ST+SS+TT+SST+STT)/4.0
      SHAPE(6)=(1.0+T-SS-SST)/2.0
      SHAPE(7)=(-1.0-ST+SS+TT+SST-STT)/4.0
      SHAPE(8)=(1.0-S-TT+STT)/2.0
C
C*** SHAPE FUNCTION DERIVATIVES
C
      DERIV(1,1)=(T+S2-ST2-TT)/4.0
      DERIV(1,2)=-S+ST
      DERIV(1,3)=(-T+S2-ST2+TT)/4.0
      DERIV(1,4)=(1.0-TT)/2.0
      DERIV(1,5)=(T+S2+ST2+TT)/4.0
      DERIV(1,6)=-S-ST
      DERIV(1,7)=(-T+S2+ST2-TT)/4.0
      DERIV(1,8)=(-1.0+TT)/2.0
      DERIV(2,1)=(S+T2-SS-ST2)/4.0
```

```
      DERIV(2,2)=(-1.0+SS)/2.0
      DERIV(2,3)=(-S+T2-SS+ST2)/4.0
      DERIV(2,4)=-T-ST
      DERIV(2,5)=(S+T2+SS+ST2)/4.0
      DERIV(2,6)=(1.0-SS)/2.0
      DERIV(2,7)=(-S+T2+SS-ST2)/4.0
      DERIV(2,8)=-T+ST
      RETURN
      END


      SUBROUTINE JACOB2(IELEM,DJACB,KGASP)
C
C*** CALCULATE COORD OF GAUSS AND THE JACOBIAN MATRIX AND IT!:;  DEREM
C
C
      DIMENSION XJACM(2,2),XJACI(2,2)
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25-24),
     *NOFIX(40),IFPRE(40,3),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *DBMAT(5,24),BMATX(5,24),
     *SMATX(5,24,4),POSGP(2),WEIGP(2),
     *GPCOD(2,4),NEROR(24)
C
C*** CALCULATE COORD OF SAMPLING POINT
C
C
      DO 10 IDIME=1,NDIME
      GPCOD(IDIME,KGASP)=0.0
      DO 10 INODE=1,NNODE
     *GPCOD(IDIME,KGASP)=GPCOD(IDIME,KGASP)+
       ELCOD(IDIME,INODE)*SHAPE(INODE)
   10 CONTINUE
C
C**** CPEATE JACOB MATRIX XJACM
C
      DO 20 IDIME=1,NDIME
      DO 20 JDIME=1,NDIME
      XJACM(IDIME,JDIME)=0.0
      DO 20 INODE=1,NNODE
      XJACM(IDIME,JDIME)=XJACM(IDIME,JDIME)+
     * DERIV(IDIME,INODE)*ELCOD(JDIME,INODE)
   20 CONTINUE
```

```
C
C
C*** CAL DETERMINANT AND INVERSE OF JACOB MATRIX
C
      DJACB=XJACM(1,1)*XJACM(2,2)-XJACM(1,2)*
     , XJACM(2,1)
      IF(DJACB.GT.0.0) GO TO 30
      WRITE(6,900) IELEM
      STOP
   30 XJACI(1,1)=XJACM(2,2)/DJACB
      XJACI(2,2)=XJACM(1,1)/DJACB
      XJACI(1,2)=-XJACM(1,2)/DJACB
      XJACI(2,1)=-XJACM(2,1)/DJACB
C
C*** CAL CARTESIAN DERIVATIVES
C
      DO 40 IDIME=1,NDIME
      DO 40 INODE=1,NNODE
      CARTD(IDIME,INODE)=0.0
      DO 40 JDIME=1,NDIME
      CARTD(IDIME,INODE)=CARTD(IDIME,INODE)+
     a XJACI(IDIME,JDIME)*DERIV(JDIME,INODE)
   40 CONTINUE
  900 FORMAT(//,24HPROGRAM HALTED IN JACOB2,
     , /,11X,22H ZERO OR NEGATIVE AREA,/,
     , 10X,16H ELEMENT, NUMBER ,I5)
      RETURN
      END
C
      SUBROUTINE BMATPB
C
C*** CAL STRAIN MATRIX B FOR PLATE BENDING
C
C
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24),
     *NOFIX(40),IFPRE(40,3),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
      DO 10 ISTRE=1,NSTRE
      DO 10 IEVAB=1,NEVAB
```

```
      EMATX(ISTRE,IEVAE)=0.0
   10 CONTINUE
      JGASH=0
      DO 20 INODE=1,NNODE
      IGASH=JGASH+1
      BMATX(4, IGASH)=CARTD(1,INODE)
      EMATX(5,IGASH)=CARTD(2,INODE)
      IGASH=IGASH+1
      JGASH=IGASH+1
      BMATX(1,IGASH)=-CARTD(1,INODE)
      BMATX(3,IGASH)=-CARTD(2,INODE)
      BMATX(4,IGASH)=-SHAPE(INODE)
      BMATX(2,JGASH)=-CARTD(2,INODE)
      EMATX(3,JGASH)=-CARTD(1,INODE)
      BMATX(5,JGASH)=-SHAPE(INODE)
   20 CONTINUE
      RETURN
      END


      SUBROUTINE DBE
C
C**** CAL D * B
C
      COMMON/CONTRO/NPOIN, NELEM, NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE, NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB .
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24),
     *NOFIX(40),IFPRE(40,3),LNODS(25, 8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2, 4),NEROR(24)
      DO 10 ISTRE=1,NSTRE
      DO 10 IEVAB=1,NEVAB
      DBMAT(ISTRE,IEVAB)=0.0
      DO 10 JSTRE=1,NSTRE
      DBMAT(ISTRE,IEVAB)=DBMAT(ISTRE,IEVAB)+
     *DMATX(ISTRE,JSTRE)*BMATX(JSTRE,IEVAB)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE LOADPB
C
C*** CAL 'NODAL FORCE FOR PLATE ELEM
C
C
      DIMENSION TITLE(12),POINT(3)
      COMMON/CONTRO/NPOIN, NELEM, NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24),
     *NOFIX(40),IFPRE(40,3),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
      DO LO IELEM=1,NELEM
      DO 10 IEVAB=1,NEVAB
      ELOAD(IELEM,IEVAB)=0.0
   10 CONTINUE
      READ(5,900) TITLE
  900 FORMAT(12A6)
      WRITE(6,905) TITLE, ICASE
  905 FORMAT(1H0,12A6,5X,12H LOAD CASE =,I3)
C
C**** READ DATA CONTROLING LOADING
C
C
      READ(5,910) IPLOD
  910 FORMAT(I5)
      WRITE(6,915) IPLOD
  915 FORMAT(I5)
C
C**** READ NODAL POINT LOADS
C
C
      IF(IPLOD.EQ.0) GO TO 60
   20 READ(5,920)
     *LODPT,(POINT(IDOFN),IDOFN=1,NDOFN)
      WRITE(6,925)
     *LODPT,(POINT(IDOFN),IDOFN=1,NDOFN)
  920 FORMAT(I5,3F10.5)
  925 FORMAT(I5,3F10.5)
C
C*** ASSOCI4TE THE NODAL POINT LOADS
C
C
C
      DO 30 IELEM=1,NELEM
      DO 30 INODE=1,NNODE
      NLOCA=LNODS(IELEM,INODE)
      IF(LODPT.EQ.NLOCA) GO TO 40
```

```
   30 CONTINUE
   40 DO 50 IDOFN=1,NDOFN
      NGASH=(INODE-1)*NDOFN+IDOFN
      ELOAD(IELEM,NGASH)=POINT(IDOFN)
   50 CONTINUE
      IF(LODPT.NE.NPOIN) GO TO 20
   60 CONTINUE
C
C**** LOOP OVER EACH ELEM
C
      DO 110 IELEM=1,NELEM
      LPROP=MATNO(IELEM)
      UDLOD=PROPS(LPROP,4)
      IF(UDLOD.EQ.0.0) GO TO 110
C
C**** EVALUATE COORD POINTS
C
      DO 70 INODE=1,NNODE
      LNODE=LNODS(IELEM,INODE)
      DO 70 IDIME=1,NDIME
      ELCOD(IDIME, INODE)=COORD(LNODE,IDIME)
   70 CONTINUE
      DO 80 IEVAB=1,NEVAE
      ELOAD(IELEM,IEVAE)=0.0
   80 CONTINUE
      KGASP=0
C
C*** ENTER LOOP FOR INTEGRATION
C
      DO 100 IGAUS=1,NGAUS
      EXISP=POSGP(IGAUS)
      DO 100 JGAUS=1,NGAUS
      ETASP=POSGP(JGAUS)
      KGASP=KGASP+1
C
C**** EVALUATE THE SHAPE FUNCTION
C
      CALL SFR2(EXISP,ETASP)
      CALL JACOB2(IELEM,DJACB,KGASP)
      DAREA=DJACB*WEIGP(IGAUS)*WEIGP(JGAUS)
C
C**** CAL LOAD AND NODAL POINTS
C
      DO 90 INODE=1,NNODE
      NPOSN=(INODE-1)*NDOFN+1
      ELOAD(IELEM,NPOSN)=ELOAD(IELEM,NPOSN)+
     *SHAPE(INODE)*UDLOD*DAREA
```

```
   90 CONTINUE
  100 CONTINUE
  110 CONTINUE
      WRITE(6,930)
  930 FORMAT(1H0,5X,
     *36H TOTAL NODAL FORCES FOR EACH ELEMENT)
      DO 120 IELEM=1,NELEM
      WRITE(6,935) IELEM,
     *(ELOAD(IELEM,IEVAB),IEVAB=1,NEVAB)
  120 CONTINUE
  935 FORMAT(1X,I4,5X,8E12.4/(10X,8E12.4)/
     *(10X,8E12.4))
      RETURN
      END

      SUBROUTINE FRONT
      DIMENSION FIXED(160),EQUAT(60),VECRV(160),
     *GLOAD(60),GSTIF(1830),ESTIF(24,24),
     *IFFIX(160),NACVA(60),LOCEL(24),NDEST(24)
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24),
     *NOFIX(40),IFPRE(40,3),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
      NFUNC(I,J)=(J*J-J)/2+I
      MFRON=60
      MSTIF=1830
C
C*** INTERPERT FIXITYDATA IN VECTOR FORM
C
      NTOTV=NPOIN*NDOFN
      DO 100 ITOTV=1,NTOTV
      IFFIX(ITOTV)=0
  100 FIXED(ITOTV)=0.0
      DO 110 IVFIX=1,NVFIX
      NLOCA=(NOFIX(IVFIX)-1)*NDOFN
      DO 110 IDOFN=1,NDUFN
      NGASH=NLOCA+IDOFN
      IFFIX(NGASH)=IFPRE(IVFIX,IDOFN)
  110 FIXED(NGASH)=PRESC(IVFIX,IDOFN)
C
C*** CHANGE THE SIGN
C
      DO 140 IPOIN=1,NPOIN
      KLAST=0
      DO 130 IELEM=1,NELEM
      DO 120 INODE=1,NNODE
```

```
      IF(LNODS(IELEM,INODE).NE.IPOIN) GO TO 120
      KLAST=IELEM
      NLAST=INODE
  120 CONTINUE
  130 CONTINUE
      IF(KLAST.NE.0) LNODS(KLAST,NLAST)=-IPOIN
  140 CONTINUE
C
C*** START BY INITIALIZING EVERYTHING THAT MATTERS TO ZERO
C
C
      DO 150 ISTIF=1,MSTIF
  150 GSTIF(ISTIF)=0.0
      DO 160 IFRON=1,MFRON
      GLOAD(IFRON)=0.0
      EQUAT(IFRON)=0.0
      VECRV(IFRON)=0.0
  160 NACVA(IFRON)=0
C
C*** AND PREPARE FOR DISC READING
C
C
      REWIND 1
      REWIND 2
      REWIND 3
      REWIND 4
C
C*** ENTER MAIN ELEMENT ASSEMBLY REDUCTION
C
      NFRON=0
      KELVA=0
      DO 380 IELEM=1,NELEM
      KEVAB=0
      READ(1) ESTIF
      DO 170 INODE=1,NNODE
      DO 170 IDOFN=1,NDOFN
      NPOSI=(INODE-1)*NDOFN+IDOFN
      LOCNO=LNODS(IELEM,INODE)
      IF(LOCNO.GT.0) LOCEL(NPOSI)=(LOCNO-1)*
     * NDOFN+IDOFN
      IF(LOCNO.LT.0) LOCEL(NPOSI)=(LOCNO+1)*
     * NDOFN-IDOFN
  170 CONTINUE
C
C***  START BY LOOKING FOR EXISTING DISTINATION
C
      DO 210 IEVAB=1,NEVAB
      NIKNO=IABS(LOCEL(IEVAB))
```

```
      KEXIS=0
      DO 180 IFRON=1,NFRON
      IF(NIKNO.NE.NACVA(IFRON)) GO TO 180
      KEVAB=KEVAB+1
      KEXIS=1
      NDEST(KEVAB)=IFRON
  180 CONTINUE
      IF(KEXIS.NE.0) GO TO 210
C
C*** WE SEEK FOR NEW EMPTY PLACE FOR DESTINATION VECTOR
C
C
      DO 190 IFRON=1,MFRON
      IF(NACVA(IFRON).NE.0) GO TO 190
      NACVA(IFRON)=NIKNO
      KEVAB=KEVAB+1
      NDEST(KEVAB)=IFRON
      GO TO 200
  190 CONTINUE
C
C*** THE NEW PLACE MAY DEMAND AN INCREASE
C
  200 IF(NDEST(EEVAB).GT.NFRON) NFRON=NDEST(KEVAB)
  a10 CONTINUE
C
C*** ASSEMBEL ELEM LOAD
C
      DO 240 IEVAB=1,NEVAB

      IDEST=NDEST(IEVAB)
      GLOAD(IDEST)=GLOAD(IDEST)+ELOAD(IELEM,IEVAB)
C
C*** ASSEMBLE ELEM STIFNESS BUT NOT IN SOLUTION
C
C
      IF(ICASE.GT.1) GO TO 230
      DO 220 JEVAB=1,IEVAB
      JDEST=NDEST(JEVAB) .
      NGASH=NFUNC(IDEST,JDEST)
      NGISH=NFUNC(JDEST,IDEST)
      IF(JDEST.GE.IDEST) GSTIF(NGASH)=
     * GSTIF(NGASH)+ESTIF(IEVAB, JEVAB)
      IF(JDEST.LT.IDEST) GSTIF(NGISH)=
     *GSTIF(NGISH)+ESTIF(IEVAB,JEVAB)
  220 CONTINUE
  230 CONTINUE
  240 CONTINUE
C
_C*** RE-EXAHIN EACH ELEMENT
```

```
C
      DO 370 IEVAB=1,NEVAB
      NIKNO=-LOCEL(IEVAB)
      IF(NIKNO.LE.0) GO TO 370
C
C*** FIND PSITION OF VARIABLEFOR ELIMINATION
C
      DO 350 IFRON=1,NFRON
      IF(NACVA(IFRON).NE.NIKNO) GO TO 350
C C*** EXTPAT THE COEFF
C
      IF(ICASE.GT.1) GO TO 260
      DO 250 JFRON=1,MFRON
      IF(IFRON.LT. JFRON) NLOCA=NFUNC(IFRON,JFRON)
      IF(IFRON.GE.JFRON) NLOCA=NFUNC(JFRON,IFRON)
      EQUAT(JFRON)=GSTIF(NLOCA)
  250 GSTIF(NLOCA)=0.0
  260 CONTINUE
C
C*** AND EXTRACT THE RIGHT HAND SIDE
C
      EQRHS=GLOAD(IFRON)
      GLOAD(IFRON)=0.0
      KELVA=KELVA+1
C
C*** WRITE EQ TO DIEC OR TAPE
C
      IF(ICASE.GT.1) GO TO 270
      WRITE(2) EQUAT,EQRHS,IFRON,NIKNO
      GO TO 280
  270 WRITE(4) EQRHS
      READ(2) EQUAT,DUMMY,IDUMM,NIKNO
  280 CONTINUE
C
C*** DEAL WITH PIVOT
C
      PIVOT=EQUAT(IFRON)
      EQUAT(IFRON)=0.0
C
C*** ENQUIRE WHETHER PRESENT VARIABLE IS FREE OR PRESCRIBED
C
      IF(IFFIX(NIKNO).EQ.0) GO TO 300
C
C*** DEAL WITH PRESCRIBE DEFLACTION
C
      DO 290 JFRON=1,NFRON
  290 GLOAD(JFRON)=GLOAD(JFRON)-FIXED(NIKNO)*
     * EQUAT(JFRON)
      GO TO 340
```

```
C
C&&& ELIMINAT A FREE VARIABLE
C
C
  300 DO 330 JFRON=1,NFRON
      GLOAD(JFRON)=GLOAD(JFRON)-EQUAT(JFRON)*
     * EQRHS/PIVOT
C
C*** NOW DEAL EITH COEFF
C
      IF(ICASE.GT.1) GO TO 320
      IF(EQUAT(JFRON).EQ.0.0) GO TO 330
      NLOCA=NFUNC(0,JFRON)
      DO 310 LFRON=1,JFRON
      NGASH=LFRON+NLOCA
  310 GSTIF(NGASH)=GSTIF(NGASH)-EQUAT(JFRON)*
     * EQUAT(LFRON)/PIVOT
  320 CONTINUE
  330 CONTINUE
  340 EQUAT(IFRON)=PIVOT
C
C*** RECORED THE NEW VACANT SPACE ,AND REDUCE
C     FRONTWITH IF POSSIBLE
C
      NACVA(IFRON)=0
      GO TO 360
C
C*** COMPLETE THE ELEMENT LOOP IN THE FORWARED
C      ELIMINATIQN
C
  350 CONTINUE
  360 IF(NACVA(NFRON).NE.0) GO TO 370
      NFRON=NFRON-1
      IF(NFRON.GT.0) GO TO 360
  370 CONTINUE
  380 CONTINUE
C*** ENTER BACK SUBSTITUTION
C
      DO 410 IELVA=1,KELVA
C
C*** READ A NEW EQ
C
      BACKSPACE 2
      READ(2) EQUAT,EQRHS,IFRON,NIKNO
      BACKSPACE 2
      IF(ICASE.EQ.1) GO TO 390
      BACKSPACE 4
      READ(4) EQRHS
```

```
      BACKSPACE 4
  390 CONTINUE
C
C*** PREPARE TO BACK-SUBSTITUTION
C
      PIVOT=EQUAT(IFRON)
      IF(IFFIX(NIKNO).EQ.1) VECRV(IFRON)=
     * FIXED(NIKNO)
      IF(IFFIX(NIKNO).EQ.0) EQUAT(IFRON)=0.0
C
C*** BACK-SUBSTITUTION
C
      DO 400 JFRON=1,MFRON
  400 EQRHS=EQRHS-VECRV(JFRON)*EQUAT(JFRON)
C
C***  PUT THE FINAL VALUES WHERE THEY BELONG
C
      IF(IFFIX(NIKNO).EQ.0) VECRV(IFRON)=
     * EQRHS/PIVOT
      IF(IFFIX(NIKNO).EQ.1) FIXED(NIKNO)=-EQRHS
      ASDIS(NIKNO)=VECRV(IFRON)
  410 CONTINUE
      WRITE(6,900)
  900 FORMAT(1H0,5X,13HDISPLACEMENTS)
      IF(NDOFN.NE.2) GO TO 430
      IF(NDIME.NE.1) GO TO 420
      WRITE(6,905)
  905 FORMAT(1H0,5X,4HNODE,6X,5HDISP.,7X,
     * 8HROTATION)
      GO TO 440
  420 WRITE(6,910)
  910 FORMAT(1H0,5X,4HNODE,5X,7HX-DISP.,
     * 7X,7HY-DISP.)
      GO TO 440
  430 WRITE(6,915)
  915 FORMAT(1H0,5X,4HNODE,6X,5HDISP.,8X,
     * 7HXZ-ROT.,7X,7HYZ-ROT.)
  440 CONTINUE
      DO 450 IPOIN=1,NPOIN
      NGASH=IPOIN*NDOFN
      NGISH=NGASH-NDOFN+1
  450 WRITE(6,920) IPOIN,(ASDIS(IGASH),IGASH=
     * NGISH,NGASH)
  920 FORMAT(I10,3E14.6)
      WRITE(6,925)
  925 FORMAT(1H0,5X,9HREACTIONS)
      IF(NDOFN.NE.2) GO TO 470
      IF(NDIME.NE.1) GO TO 460
```

```
      WRITE(6,930)
  930 FORMAT(1H0,5X,4HNODE,6X,5HFORCE,8X,6HMOMENT)
      GO TO 480
  460 WRITE(6,935)
  935 FORMAT(1H0,5X,4HNODE,5X,7HX-FORCE,7X,
     * 7HY-FORCE)
      GO TO 480
  470 WRITE(6,940)
  940 FORMAT(1H0,5X,4HNODE,6X,5HFORCE,6X,
     * 9HXZ-MOMENT,5X,9HYZ-MOMENT)
  480 CONTINUE
      DO 510 IPOIN=1,NPOIN
      NLOCA=(IPOIN-1)*NDOFN
      DO 490 IDOFN=1,NDOFN
      NGUSH=NLOCA+IDOFN
      IF(IFFIX(NGUSH).GT.0) GO TO 500
  490 CONTINUE
      GO TO 510
  500 NGASH=NLOCA+NDOFN
      NGISH=NLOCA+1
      WRITE(6,945) IPOIN (FIXED(IGASH),IGASH=
     * NGISH,NGASH)
  510 CONTINUE
  945 FORMAT(I10,3E14.6)
C
C*** POST FRONT SET ALL ELEM CONNECTION TO POSITIVE VALUE
C*** FOR USE IN STRESS CALCULATION
C
      DO 520 IELEM=1,NELEM
      DO 520 INODE=1,NNODE
  540 LNODS(IELEM,INODE)=IABS(LNODS(IELEM,INODE))
      RETURN
      END
      SUBROUTINE STREPB
C*** CAL STRESS RESULTANTS AT GAUSS POINTS
C
      DIMENSION ELDIS(3,8),STRSG(5)
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24),
     *NOF[X(40).[FPRE(40,3),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *BBMPI(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
```

```
      WRITE(6,900)
      WRITE(6,905)
C
C*** LOOP OVER EACH ELEM
C
      DO 40 IELEM=1,NELEM
C
C*** READ STRESS MATRIX
C
      READ(3) SMATX,GPCOD
      WRITE(6,910) IELEM
C
C*** IDENTIFY THE DISP OF THE ELEM NODAL POINT
C
      DO 10 INODE=1,NNODE
      LNODE=LNODS(IELEM,INODE)
      NPOSN=(LNODE-1)*NDOFN
      DO 10 IDOFN=1,NDOFN
      NPOSN=NPOSN+1
      ELDIS(IDOFN,INODE)=ASDIS(NPOSN)
   10 CONTINUE
      KGASP=0
C
C*** ENTER LOOP OVER EACH POINT
C
      DO 30 IGAUS=1,NGAUS
      DO 30 JGAUS=1,NGAUS
      KGASP=KGASP+1
      DO 20 ISTRE=1,NSTRE
      STRSG(ISTRE)=0.0
      KGASH=0
C
C*** COMPUTE THE STRESS RESULTANTS
C
C
      DO 20 INODE=1,NNODE
      DO 20 IDOFN=1,NDOFN
      KGASH=KGASH+1
      STRSG(ISTRE)=STRSG(ISTRE)+
     *SMATX(ISTRE,KGASH,KGASP)*ELDIS(IDOFN,INODE)
   20 CONTINUE
C
C*** OUTPUT THE STRESS RESULTANTS
C
      WRITE(6,915) KGASP,
     *(GPCOD(IDIME,KGASP),IDIME=1,NDIME),
     *(STRSG(ISTRE),ISTRE=1,NSTRE)
   30 CONTINUE
```

```
   40 CONTINUE
  900 FORMAT(/,10X,8HSTRESSES,/)
  5    FORMAT(1H0,4HG,P.,2X,8HX-COORD.,2X,
     *8HY-COORD.,3X,8HX-MOMENT,4X,8HY-MOMENT,
     *3X,9HXY-MOMENT,2X,10HXZ-S.FORCE,2X,
     *10HYZ-S.FORCE)
  910 FORMAT(/,5X,12HELEMENT NO.=,I5)
  915 FORMAT(I5,2F10.4,5E12.5)
      RETURN
      END
```

```
      SUBROUTINE CHECK1
C
C*** TO CRITICIZE THE DATA AND PRINT ANY DIAGNOSTICS
C
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24),
     *NOFIX(40),IFPRE(40,3),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
      DO 10 IEROR=1,24
   10 NEROR(IEROR)=0
C
C*** CREATE THE DIGNOSTIC MESSAGES
C
      IF (NPOIN.LE.0) NEROR(1)=1
      IF(NELEM*NNODE.LT.NPOIN) NEROR(2)=1
      IF(NVFIX.LT.1.OR.NVFIX.GT.NPOIN) NEROR(3)=1
      IF(NCASE.LE.0) NEROR(4)=1
      IF(NTYPE.LT.0.OR.NTYPE.GT.2) NEROR(5)=1
      IF(NNODE.LT.3.OR.NNODE.GT.8) NEROR(6)=1
      IF(NDOFN.LT.2.OR.NDOFN.GT.3) NEROR(7)=1
      IF(NMATS.LE.0.OR.NMATS.GT.NELEM) NERQR(8)=1
      IF(NPROP.LT.3.OR.NPROP.GT.5) NEROR(9)=1
      IF(NGAUS.LT.2.OR.NGAUS.GT.3) NEROR(10)=1
      IF(NDIME.LT.1.OR.NDIME.GT.2) NEROR(11)=1
      IF(NSTRE.LT.2.OR.NSTRE.GT.5) NEROR(12)=1
C
C*** EITHER RETURN, OR ELSE PRINT THE ERRORS
C
      KEROR=0
      DO 20 IEROR=1,12
```

```
      IF(NEROR(IEROR).EQ.0) GO TO 20
      KEROR=1
      WRITE(6,900) IEROR
  900 FORMAT(//25H *** DIAGNOSIS BY CHECK1,
     * 6H ERROR,I3)
   20 CONTINUE
      IF(KEROR.EQ.0) RETURN
C
C*** OTHERWISE ECHO ALL THE REMAINING DATA
C
      CALL ECHO
      END

      SUBROUTINE ECHO
      DIMENSION NTITL(80)
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24).
     *NOFIX(40).IFPRE(40,3).LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2).WEIGP(2),GPCOD(2,4),NEROR(24)
      WRITE(6,900)
  900 FORMAT(//25H NOW FOLLOWS A LISTING OF,
     * 25H POST-DISASTER DATA CARDS/)
   10 READ(5,905) NTITL
  905 FORMAT(80A1)
      WRITE(6,910) NTITL
  910 FORMAT(20X,:?()&: )
      GO TO 10
      END

      SUBROUTINE CHECK2
C
C*** TO CRITICIZE THE DATA FROM SUBROUTIN INPUT
C
C
      DIMENSION NDFRO(60)
      COMMON/CONTRO/NPOIN.NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,4),
     *PRESC(40,3),ASDIS(240),ELOAD(25,24),
     *NOFIX(40),IFPRE(40,3),LNODS(25,8),
     *MATNO(25)
```

```
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(5,5),CARTD(2,8),
     *DBMAT(5,24),BMATX(5,24),SMATX(5,24,4),
     *POSGP(2),WEIGP(2),GPCOD(2,4),NEROR(24)
      MFRON=200
C
C**** CHECK AGAINST TWO IDENTICAL NONZERO
C
      DO 10 IELEM=1,NELEM
   10 NDFRO(IELEM)=0
      DO 40 IPOIN=2,NPOIN
      KPOIN=IPOIN-1
      DO 30 JPOIN=1,KPOIN
      DO 20 IDIME=1,NDIME
      IF(COORD(IPOIN,IDIME).NE. COORD(JPOIN,
     * IDIME)) GO TO 30
   20 CONTINUE
      NEROR(13)=NEROR(13)+1
   30 CONTINUE
   40 CONTINUE
C
C*** CHECK THE LIST OF ELEM PROPERTY
C
      DO 50 IELEM=1,NELEM
   50 IF(MATNO(IELEM).LE.0.OR.MATNO(IELEM).GT.
     * NMATS) NEROR(14)=NEROR(14)+1
C
C*** CHECK FOR IMPOSSIBLE NODE
C
      DO 70 IELEM=1,NELEM
      DO 60 INODE=1,NNODE
      IF(LNODS(IELEM,INUDE).EQ.0) NEROR(15)=
     * NEROR(15)+1
   60 IF(LNODS(IELEM,INODE).LT. 0.OR. LNODS(IELEM,
     * INODE).GT.NPOIN) NEROR(16)=NEROR(16)+1
   70 CONTINUE
C
C*** CHECK FOR ANY REPETITION OF A NODE
C
      DO 140 IPOIN=1,NPOIN
      KSTAR=0
      DO 100 IELEM=1,NELEM
      KZERO=0
      DO 90 INODE=1,NNODE
      IF(LNODS(IELEM,INODE).NE.IPOIN) GO TO 90
      KZERO=KZERO+1
      IF(KZERO.GT.1) NEROR(17)=NEROR(17)+1
```

```
C
C****
C
      IF(KSTAR.NE.0) GO TO 80
      KSTAR=IELEM
C
C**** CALCUL INCREASE OR DECREASE IN FRONTWIDTH
C
      NDFRO(IELEM)=NDFRO(IELEM)+NDOFN
   80 CONTINUE
C
C*** AND CHANGE THE SIGN OF THE LAST NODE
C
      KLAST=IELEM
      NLAST=INODE
   90 CONTINUE
  100 CONTINUE
      IF(KSTAR.EQ.0) GO TO 110
      IF(KLAST.LT.NELEM) NDFRO(KLAST+1)=
     * NDFRO(KLAST+1)-NDOFN
      LNODS(KLAST,NLAST)=-IPOIN
      GO TO 140
C
C**** CHECK THE COORD FOR AN UNUSED NODE
C
  110 WRITE(6,900) IPOIN
  900 FORMAT(/15H CHECK WHY NODE,I4,
     * 14H NEVER APPEARS)
      NEROR(18)=NEROR(18)+1
      SIGMA=0.0
      DO 120 IDIME=1,NDIME
  120 SIGMA=SIGMA+ABS(COORD(IPOIN,IDIME))
      IF(SIGMA.NE.0.0) NEROR(19)=NEROR(19)+1
C
C*** CHECK THAT UNUSED NUMBERT IS NOT A RESTRAINED NODE
C
C
      DO 130 IVFIX=1,NVFIX
  130 IF(NOFIX(IVFIX).EQ.IPOIN) NEROR(20)=
     * NEROR(20)+1
  140 CONTINUE
C
C**** CALCUL THE LARGEST FRONTWIDTH
C
      NFRON=0
      KFRON=0
      DO 150 IELEM=1,NELEM
      NFRON=NFRON+NDFRO(IELEM)
  150 IF(NFRON.GT.KFRON) KFRON=NFRON
```

```
      WRITE(6,905) KFRON
  905 FORMAT(//28HMAX FRONTWIDTH ENCOUNTERED =,I5)
      IF(KFRON.GT.MFRON) NEROR(21)=1
C
C*** CONTINUE CHECKING THE DATA FOR THE FIXED VALUE
C
      DO 170 IVFIX=1,NVFIX
      IF(NOFIX(IVFIX).LE.0.OR.NOFIX(IVFIX).
     * GT.NPOIN) NEROR(22)=NEROR(22)+1
      KOUNT=0
      DO 160 IDOFN=1,NDOFN
  160 IF(IFPHE(IVFIX,IDOFN).GT.0) KOUNT=1
      IF (KOUNT.EQ.0) NEROR(23)=NEROR(23)+1
      KVFIX=IVFIX-1
      DO 170 JVFIX=1,KVFIX
  170 IF(IVFIX.NE.1.AND.NOFIX(IVFIX).EQ.
     * NOFIX(JVFIX)) NEROR(24)=NEROR(24)+1
      KEROR=0
      DO 180 IEROR=13,24
      IF(NEROR(IEROR).EQ.0) GO TO 180
      KEROR=1
      WRITE(6,910) IEROR,NEROR(IEROR)
  910 FORMAT(//30H*** DIAGNOSIS BY CHECK2, ERROR,
     * I3,6X,18H ASSOCIATED NUMBER,I5)
  180 CONTINUE
      IF(KEROR.NE.0) GO TO 200
C
C*** RETURN ALL NODAL CONNECTION NUMBERS TO POSITIVE VALUE
C
C
      DO 190 IELEM=1,NELEM
      DO 190 INODE=1,NNODE
  190 LNODS(IELEM,INODE)=ABS(LNODS(IELEM,INODE))
      RETURN
  200 CALL ECHO
      END
```

B.2 Plane Stress or Plane Strain Program

Subroutines INPUT, GAUSSA, Node XY, SFR2, JACOB2, DBE, FRONT, CHECK1, ECHO, and CHECK2, which are used in plate bending program are identical to the plane stress or plane strain program except for the common blocks.

```
C     PROGRAM BOK1(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,
C    *TAPE2,TAPE2,TAPE3,TAPE4)
      DIMENSION TITLE(12)
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,5),
     *PRESC(40,2),ASDIS(160),ELOAD(25,16),STRIN(3,225),
     *NOFIX(40),IFPRE(40,2),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(3,3),CARTD(2,8),
     *DBMAT(3,16),BMATX(3,16),SMATX(3,16,9),
     *POSGP(3),WEIGP(3),GPCOD(2,9),NEROR(24)
      READ(5,900) NPROB
  900 FORMAT(I5)
      WRITE(6,905) NPROB
  905 FORMAT(1HO,5X,23HTOTAL NO, OF PROBLEM =,I5)
      DO 20 IPROB=1,NPROB
      REWIND 1
      REWIND 2
      REWIND 3
      REWIND 4
      READ(5,910) TITLE
  910 FORMAT(12A6)
      WRITE(6,915) IPROB,TITLE
  915 FORMAT(/////,6X,12HPROBLEM NO, ,I3,10X,12A6)
C
C*** CALL SUBROUTIN THAT READ MOST OF THE DATA
C
      CALL INPUT
C
C*** CREAT ELEM SSTIFNESS
C
      CALL STIFFS
      DO 10 ICASE=1,NCASE
C
C**** COMPUT LOAD,
C
C
      CALL LOADPS
C
C*** SOLVE THE RESULTING EQ
C
      CALL FRONT
C
C*** COMPUT STRESS IN ALL ELEMENTS.
C
      CALL STREPS
   10 CONTINUE
   20 CONTINUE
      STOP
      END
```

```
t stifps fortran

      SUBROUTINE STIFPS
      DIMENSION ESTIF(16,16)
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,NEVAB,
     *ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,5),PRESC(40,2),
     *ASDIS(160),ELOAD(25,16),STRIN(4,225),NOFIX(40),IFPRE(40,2),
     *LNODS(25,8),MATNO(25)
      COMMON/WORK/ELCOD(2,8),SHAPE(8),DERIV(2,8),
     *DMATX(3,3),CARTD(2,8),DBMAT(3,16),BMATX(3,16),
     *SMATX(3,16,9),POSGP(3),WEIGP(3),GPCOD(2,9),NEROR(24)
C
C**** LOOP OVER EACH ELEMENT
C
      DO 70 IELEM=1,NELEM
      LPROP=MATNO(IELEM)
C  C
C ***** EVALUATE THE COOR OF ELEMENT
C C
      DO 10 INODE=1,NNODE
      LNODE=LNODS(IELEM,INODE)
      DO 10 IDIME=1,NDIME
   10 ELCOD(IDIME,INODE)=COORD(LNODE,IDIME)
C
C**** EVALUATE D----MATRIX
C
      CALL MODPS(LPROP)
      THICK=PROPS(LPROP,3)
C
C**** INITIALIZE THE STIFFNESS MATRIX
C
      DO 20 IEVAB=1,NEVAB
      DO 20 JEVAB=1,NEVAE
   20 ESTIF(IEVAB,JEVAB)=0.0
      KGASP=0
C
C*** LOOP AREA INTIGRUTION
C
      DO 50 IGAUS=1,NGAUS
      DO 50 JGAUS=1,NGAUS
      KGASP=KGASP+1
      EXISP=POSGP(IGAUS)
      ETASP=POSGP(JGAUS)
C
C**** EVALUATE THE SHAPE FUNCTION
C
```

```
      CALL SFR2(EXISP,ETASP)
      CALL JACOB2(IELEM,DJACB,KGASP)
      DVOLU=DJACB*WEIGP(IGAUS)*WEIGP(JGAUS)
      IF(THICK.NE.0.0) DVOLU=DVOLU*THICK
C
C**** EVALUATE B AND D*B MATRIX
C
      CALL BMATPS
      CALL DEE
C C*****
C****** EVALUATE THE ELEMENT STIFFNESSES
C
      DO 30 IEVAB=1,NEVAB
      DO 30 JEVAB=IEVAB,NEVAB
      DO 30 ISTRE=1,NSTRE
   30 ESTIF(IEVAB,JEVAB)=ESTIF(IEVAB,JEVAB)+
     * BMATX(ISTRE,IEVAB)*DBMAT(ISTRE,JEVAB)*DVOLU
C
C*** STORE THE ELEMENT OF DB MGTRIX
C
      DO 40 ISTRE=1,NSTRE
      DO 40 IEVAB=1,NEVAB
   40 SMATX(ISTRE,IEVAB,KGASP)=DBMAT(ISTRE,IEVAB)
   50 CONTINUE
C
C*** CONSTRUCT THE,LOWER TRIANGLE OF
C**** THE STIFFNESS MATRIX
C
      DO 60 IEVAB=1,NEVAB
      DO 60 JEVAB=1,NEVAB
   60 ESTIF(JEVAB,IEVAB)=ESTIF(IEVAB,JEVAB)
C
C**** STORE STIFFNESS MATRIX
C
      WRITE(1) ESTIF
      WRITE(3) SMATX,GPCOD
   70 CONTINUE
      RETURN
      END
```

```
 t modps fortran

      SUBROUTINE MODPS(LPROP)
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,5),
     *PRESC(40,2),ASDIS(160),ELOAD(25,16),
     *STRIN(4,225),NOFIX(40),IFPRE(40,2),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(3,3),CARTD(2,8),
     *DBMAT(3,16),BMATX(3,16),SMATX(3,16,9),
     *POSGP(3),WEIGP(3),GPCOD(2,9),NEROR(24)
      YOUNG=PROPS(LPROP,1)
      POISS=PROPS(LPROP,2)
      DO 10 ISTRE=1,NSTRE
      DO 10 JSTRE=1,NSTRE
      DHATX(ISTRE,JSTRE)=0.0
   10 CONTINUE
      IF(NTYPE.NE.1) GO TO 20
C
C**** D MATRIX FOR PLAIN STRESS CASE
C
      CONST=YOUNG/(1.0-POISS*POISS)
      DMATX(1,1)=CONST
      DMATX(2,2)=CONST
      DMATX(1,2)=CONST*POISS
      DMATX(2,1)=CONST*POISS
      DMATX(3,3)=CONST*(1.0-POISS)/2.0
      GO TO 30
   20 IF(NTYPE.NE.2) GO TO 30
C
C***** D MATRIX FOR PLAIN STRIAN CASE
C
      CONST=YOUNG*(1.0-POISS)/((1.0+POISS)*
     * (1.0-2.0*POISS))
      DMATX(1,1)=CONST
      DMATX(2,2)=CONST
      DMATX(1,2)=CONST*POISS/(1.0-POISS)
      DMATX(2,1)=CONST*POISS/(1.0-POISS)
      DMATX(3,3)=CONST*(1.0-2.0*POISS)/
     * (2.0*(1.0-POISS))
   30 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE BMATPS
      COMMON/CONTRO/NPOIN, NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX'
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,5),
     *PRESC(40,2),ASDIS(160),ELOAD(25,16),
     *STRIN(4,225),NOFIX(40),IFPRE(40,3),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2.8),DMATX(3,3),CARTD(2,8),
     *DBMAT(3,16),BMATX(3,16),SMATX(3,16,9),
     *POSGP(3),WEIGP(3),GPCOD(2,9),NEROR(24)
      NGASH=0
      DO 10 INODE=1,NNODE
      MGASH=NGASH+1
      NGASH=MGASH+1
      EMATX(1,MGASH)=CARTD(1,INODE)
      BMATX(1,NGASH)=0.0
      BMATX(2, MGASH)=0.0
      BMATX(2,NGASH)=CARTD(2,INODE)
      BMATX(3,NGWSH)=CARTD(2,INODE)
      BMATX(3,NGASH)=CARTD(1,INODE)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE LOADPS
      DIMENSION TITLE(12),POINT(2),PRESS(3,2),PGASH(2),
     *DGASH(2),TEMPE(80),STRAN(3),STRES(3),NOPRS(3)
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,5),
     *PRESC(40,2),ASDIS(160),ELOAD(25,16),
     *STRIN(4,225),NOFIX(40),IFPRE(40,2),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(3,3),CARTD(2,8),
     *DBMAT(3,16),BMATX(3,16),SMATX(3,16,9),
     *POSGP(3),WEIGP(3),GPCOD(2,9),NEROR(24)
      DO 10 IELEM=1,NELEM
      DO 10 IEVAB=1,NEVAB
   10 ELOAD(IELEM,IEVAB)=0.0
      READ(5,900) TITLE
  900 FORMAT(12A6)
      WRITE(6,905) TITLE,ICASE
  905 FORMAT(1HO,12A6,5X,12H LOAD CASE =,I3)
C
C**** READ DATA LOADING
C
      READ(5,910) IPLOD,IGRAV,IEDGE,ITEMP
      WRITE(6,910) IPLOD,IGRAV,IEDGE,ITEMP
  910 FORMAT(4I5)
C
C****   READ NODAL POIT LOADS
C
      IF(IPLOD.EQ.0) GO TO 500
   20 READ(5,915) LODPT,(POINT(IDOFN),IDOFN=1,NDOFN)
      WRITE(6,915) LODPT,(POINT(IDOFN),IDOFN=1,NDOFN)
  915 FORMAT(I5,2F10.3)
C
C**** ASSOCIAT THE NODAL POINT LOADS WITH AN ELEMENT
C
      DO 30 IELEM=1,NELEM
      DO 30 INODE=1,NNODE
      NLOCA=LNODS(IELEM,INODE)
      IF(LODPT.EQ.NLOCA) GO TO 40
   30 CONTINUE
   40 DO 50 IDOFN=1,NDOFN
      NGASH=(INODE-1)*NDOFN+IDOFN
   50 ELOAD(IELEM,NGASH)=POINT(IDOFN)
      IF(LODPT.LT.NPOIN) GO TO 20
  500 CONTINUE
      IF(IGRAV.EQ.0) GO TO 600
```

```
C
C*** GRAVITY LOADING SECTION
C
C**** READ GRAVITY ANGLE AND GRAVITATIONAL CONSTANT
C
      READ(5,920) THETA,GRAVY
  920 FORMAT(2F10.3)
      WRITE(6,925) THETA,GRAVY
  925 FORMAT(1H0,16H GRAVITY ANGLE =,F10.3,19H GRAVITY CONSTANT =,
     *F10.3)
      THETA=THETA/57.295779514
C
C**** LOOP OVER EACH ELEM
C
      DO 90 IELEM=1,NELEH
C
C******* SET UP PRELIMINARY CONST
C
      LPROP=MATNO(IELEM)
      THICK=PROPS(LPROP,3)
      DENSE=PROPS(LPROP,4)
      IF(DENSE.EQ.0.0) GO TO 90
      GXCOM=DENSE*GRAVY*SIN(THETA)
      GYCOM=-DENSE*GRAVY*COS(THETA)
C
C***** COMPUTE COORD OF THE ELEM NODAL POINT
C
      DO 60 INODE=1,NNODE
      LNODE=LNODS(IELEM,INODE)
      DO GO IDIME=1,NDIME
   60 ELCOD(IDIME,INODE)=COORD(LNODE,IDIME)
C
C****** ENTER LOOPS FOR AREA NUMERICAL INTEGRATION
C
      DO 80 IGAUS=1,NGAUS
      00 80 JGAUS=1,NGAUS
      EXISP=POSGP(IGAUS)
      ETASP=POSGP(JGAUS)
C
C***** COMPUTE SHAPE FUNCTION AT SAMPLING POINT AND ELEM VOLUME
C
      CALL SFR2(EXISP,ETASP)
      KGASP=1
      CALL JACOB2(IELEM,DJACB,KGASP)
      DVOLU=DJACB*WEIGP(IGAUS)*WEIGP(JGAUS)
      IF(THICK.NE.0.3) DVOLU=DVOLU*THICK
C
C***** CALCULATE LOADS AND ASSOCIATE WITH ELEM NODAL POINT
C
```

```
C
      DO 70 INODE-1,NNODE
      NGASH=(INODE-1)*NDOFN+1
      MGASH=(INODE-1)*NDOFN+2
      ELOAD(IELEM,NGASH)=ELOAD(IELEM,NGASH)+GXCOM*SHAPE(INODE)*DVOLU
   70 ELOAD(IELEM,MGASH)=ELOAD(IELEM,MGASH)+GYCOM*SHAPE(INODE)*
     * DVOLU
   80 CONTINUE
   90 CONTINUE
  600 CONTINUE
      IF(IEDGE.EQ.O) GO TO 700
C
C***** DISTRIBUTED LOAD EDGE
C
      READ(5,930) NEDGE
  930 FORMAT(IS)
      WRITE(6,935) NEDGE
  935 FORMAT(1HO,5X,21HNO, OF LOADED EDGES=,I5)
      WRITE(6,940)
  940 FORMAT(1HO,5X,38HLIST OF LOADED EDGES AND APPLIED LOADS)
      NODEG=3
C
C******* LOOP OVER EACH ELEMENT
C
      DO 160 IEDGE=1,NEDGE
C
C***** READ DATA LOCATING THE LOADED EDGE AND APPLIED LOAD
CD
      READ(5,945) NEASS,(NOPRS(IODEG),IODEG=1,NODEG)
  945 FORMAT!4IS)
      WRITE(6,950) NEASS,(NOPRS(IODEG),IODEG=1,NODEG)
  950 FORMAT(I10,5X,3I5)
      READ(5,955) ((PRESS(IODEG,IDOFN),IODEG=1,NODEG),
     * IDOFN=1,NDOFN)
      WRITE(6,955) ((PRESS(IODEG,IDOFN),IODEG=1,NODEG),
     * IDOFN=1,NDOFN)
  955 FORMAT(6F10.3)
      ETASP=-1.0
C
C****** CALCULATE COORD OF THE NODES OF THE ELEH EDGE
C
      DO 100 IODEG=1,NODEG
      LNODE=NOPRS(IODEG)
      DO 100 IDIME=1,NDIME
  100 ELCOD(IDIME,IODEG)=COORD(LNODE,IDIME)
C
C**** ENTER LOOP FOR NUMERICAL INTIGRATION
C
```

```
      DO 150 IGAUS=1,NGAUS
      EXISP=POSGP(IGAUS)
C
C***** EVALUATE SHAPE FUNCTION AT THE SAMPLING POINT
C
      CALL SFR2(EXISP,ETASP)
C
C**** CALCULATE COMPONENTS OF THE EQUIVALENT NODAL LOADS
C
      DO 110 IDOFN=1,NDOFN
      PGASH(IDOFN)=0.0
      DGASH(IDOFN)=0.0
      DO 110 IODEG=1,NODEG
      PGASH(IDOFN)=PGASH(IDOFN)+PRESS(IODEG,IDOFN)*SHAPE(IODEG)
  110 DGASH(IDOFN)=DGASH(IDOFN)+ELCOD(IDOFN,IODES;)*DERIV(1,IODEG)
      DVOLU=WEIGP(IGAUS)
      PXCOM=DGASH(1)*PGASH(2)-DGASH(2)*PGASH(1)
      PYCOM=DGASH(1)*PGASH(1)+DGASH(2)*PGASH(2)
C
C**** ASSOCIATE THE EQUVALENT NODAL EDGE LOAD WITH AN ELEM
C
      DO 120 INODE=1,NNODE
      NLOCA=LNODS(NEASS,INODE)
      IF(NLOCA.EQ.NOPRS(1)) GO TO 130
  120 CONTINUE
  130 JNODE=INODE+NODEG-1
      KOUNT=0
      DO 140 KNODE=INODE,JNODE
      KOUNT=KOUNT+1
      NGASH=(KNODE-1)*NDOFN+1
      MGASH=(KNODE-1)*NDOFN+2
      IF(KNODE.GT.NNODE) NGASH=1
      IF(KNODE.GT.NNODE) MGASH=2
      ELOAD(NEASS,NGASH)=ELOAD(NEASS,NGASH)+
     * SHAPE(KOUNT)*PXCOM*DVOLU
  140 ELOAD(NEASS,MGASH)=ELOAD(NEASS,MGASH)+
     * SHAPE(KOUNT)*PYCOM*DVOLU
  150 CONTINUE
  100 CONTINUE
  700 CONTINUE
      IF(ITEMP.EQ.0) GO TO 300
      DIMENSION ELCODD(3,8)
C
C**** THERMAL LOAD SECTION
C
C
C**** INITIALIZE AND INPUT NODAL TEMPERATURE
C
```

```
C
      DO 170 IPOIN=1,NPOIN
  170 TEMPE(IPOIN)=0.0
      WRITE(6,960)
  960 FORMAT(1H0,5X,29HPRESCRIBED NODAL TEMPERATURES)
  180 READ(5,945) NODPT,TEMPE(NODPT)
      WRITE(6,965) NODPT,TEMPE(NODPT)
  965 FORMAT(I5,F10.3)
      IF(NODPT.LT.NPOIN)GO TO 180
      MDIME=NDIME+1
      KGAST=0
C
C***** LOOP OVER EACH ELEM
C
      DO 280 IELEM=1,NELEM
      LPROP=MATNO(IELEM)
      DO 200 INODE=1,NNODE
      LNODE=LNODS(IELEM,INODE)
C
C***** IDENTIFY THE COORD AND TEMP OF EACH ELEM
C
      DO 190 IDIME=1,NDIME
  190 ELCOD(IDIME,INODE)=COORD(LNODE,IDIME)
  200 ELCODD(MDIME,INODE)=TEMPE(LNODE)
C
C***** SET UP MATERIAL PROPERTIES
C
      CALL MODPS(LPROP)
      YOUNG=PROPS(LPROP,1)
      POISS=PROPS(LPROP,2)
      THICK=PROPS(LPROP,3)
      ALPHA=PROPS(LPROP,5)
C
C***** LOOP FOR AREA NUMERICAL INTIGRATION
C
      DO 270 IGAUS=1,NGAUS
      DO 270 JGAUS=1,NGAJJS
      KGAST=KGAST+1
      EXISP=POSGP(IGAUS)
      ETASP=POSGP(JGAUS)
C
C***** EVALUATE SHAPE FUNCTION AND TEMP AT THE SAMPLING POINT
C
      CALL SFR2(EXISP,ETASP)
      KGCSP=1
      CALL JACOB2(IELEM,DJACB,KGASP)
      THERM=0.0
      DO 210 INODE=1,NNODE
```

```
  210 THERM=THERM+ELCODD(MDIME,INODE)*SHAPE(INODE)
      DVOLU=DJACB*WEIGP(IGAUS)*WEIGP(JGAUS)
      IF(THICK.NE.0.0) DVOLU=DVOLU*THICK
C
C***** EVALUATE THE INITIAL THERMAL STRAIN
C
      EIGEN=THERM*ALPHA
      IF(NTYPE.EQ.2) GQ TO 220
      STRAN(1)=-EIGEN
      STRAN(2)=-EIGEN
      STRAN(3)=0.0
      GO TO 230
  220 STRAN(1)=-(1.0+POISS)*EIGEN
      STRAN(2)=-(1.0+POISS)*EIGEN
      STRAN(3)=0.0
C
C***** AND THE CORRESPOUNDING INITIAL STRESS
C
  230 DO 250 ISTRE=1,NSTRE
      STRES(ISTRE)=0.0
      DO 240 JSTRE=1,NSTRE
  240 STRES(ISTRE)=STRES(ISTRE)+DMATX(ISTRE,JSTRE)*STRAN(JSTRE)
  250 STRIN(ISTRE,KGAST)=STRES(ISTRE)
      IF(NTYPE.EQ.2) STRIN(4,KGAST)=-YOUNG*EIGEN
      IF(NTYPE.EQ.1) STRIN(4,KGAST)=0.0
C
C****** EVALUATE THE EQUIVALENT NODAL FORCES
C
      DO 260 INODE=1,NNODE
      NGASH=(INODE-1)*NDOFN+1
      MGASH=(INODE-1)*NDOFN+2
      ELOAD(IELEM,NGASH)=ELOAD(IELEM,NGASH)
     * -(CPRTD(1,INODE)*STRES(1)+CARTD(2,INODE)*STRES(3))*DVOLU
  240 ELOAD(IELEM,MGASH)=ELOAD(IELEM,MGASH)
     * -(CARTD(1,INODE)*STRES(3)+CARTD(2,INODE)*STRES(2))*DVOLU
  270 CONTINUE
  280 CONTINUE
  800 CONTINUE
      WRITE(6,970)
  770 FORMAT(1H0,5X,36H TOTAL NODAL FORCES FOR EACH ELEMENT)
      DO 290 IELEM=1,NELEM
  290 WRITE(6,975) IELEM,(ELOAD(IELEM,IEVAB),IEVAB=1,NEVAB)
  975 FORMAT(1X,I4,5X,8E12.4/(10X,8E12.4))
      RETURN
      END
```

```
      SUBROUTINE STREPS
      DIMENSION STRSP(3),ELDIS(2,8),STRSG(4)
      COMMON/CONTRO/NPOIN,NELEM,NNODE,NDOFN,NDIME,
     *NSTRE,NTYPE,NGAUS,NPROP,NMATS,NVFIX,
     *NEVAB,ICASE,NCASE,ITEMP,IPROB,NPROB
      COMMON/LGDATA/COORD(80,2),PROPS(10,5),
     *PRESC(40,2),ASDIS(160),ELOAD(25,16),STRIN(4,225),
     *NOFIX(40),IFPRE(40,2),LNODS(25,8),
     *MATNO(25)
      COMMON /WORK/ELCOD(2,8),SHAPE(8),
     *DERIV(2,8),DMATX(3,3),CARTD(2,8),
     *DBMAT(3,16),BMATX(3,16),SMATX(3,16,9),
     *POSGP(3),WEIGP(3),GPCOD(2,9),NEROR(24)
      NSTR1=NSTRE+1
      WRITE(6,900)
      WRITE(6,905)
  905 FORMAT(1H0,4HG,P.,2X,8HX-COORD.,2X,8HY-COORD.,3X,8HX-STRESS,
     * 4X,8HY-STRESS,3X,9HXY-STRESS,3X,8HZ-STRESS,4X,
     * SHMAX P.S.,4X,8HMIN P.S.,6X,5HANGLE)
      KGAST=0
C
C***** LOOP FOR EACH ELEM
C
      DO 60 IELEM=1,NELEM
      LPROP=MATNO(IELEM)
      POISS=PROPS(LPROP,2)
C
C***** READ THE STRESS MATRIX, SAMPLING POINT COORD FOR ELEM
C
      READ(3) SMATX,GPCOD
      WRITE(6,910) IELEM
C
C***** IDENTIFY THE DISPLACEMENT OF THE ELEM NODE
C
      DO 10 INODE=1,NNODE
      LNODE=LNODS(IELEM,INODE)
      NPOSN=(LNODE-1)*NDOFN
      DO 10 IDOFN=1, NDOFN
      NPOSN=NPOSN+1
      ELDIS(IDOFN,INODE)=ASDIS(NPOSN)
   10 CONTINUE
      KGASP=0
C
C***** ENTER LOOP FOR EACH SAMPLING POINT
C
      DO 50 IGAUS=1,NGAUS
      DO 50 JGAUS=1,NGAUS
      KGAST=KGAST+1
      KGASP=KGASP+1
```

```
C
C**** COMPUTE THE CARTESIAN STRESS COMPONENT
C
      DO 20 ISTRE=1,NSTRE
      STRSG(ISTRE)=0.0
      KGASH=0
      DO 20 INODE=1,NNODE
      DO 20 IDOFN=1,NDOFN
      KGASH=KGASH+1
      STRSG(ISTRE)=STRSG(ISTRE)+SMATX(ISTRE,
     * KGASH,KGASP)*ELDIS(IDOFN,INODE)
   20 CONTINUE
C
C**** COMPUTE OUT OF PLANE NORMAL STRESS COMPONENT
C
      IF(NTYPE.EQ.2) STRSG(4)=POISS*(STRSG(1)+STRSG(2))
      IF(NTYPE.EQ.1) STRSG(4)=0.0
C
C****** FOR THERMAL LOADING ADD AN INITIAL
C------THERMAL STRESSES
C
      IF(ITEMP.EQ.0) GO TO 40
      DO 30 ISTR1=1,NSTR1
      STRSG(ISTR1)=STRSG(ISTR1)+STRIN(ISTR1,KGAST)
   30 CONTINUE
C
C***** COMPUTE THE PRINCIPAL STRESSES
C
   40 XGASH=(STRSG(1)+STRSG(2))*0.5
      XGISH=(STRSG(1)-STRSG(2))*0.5
      XGESH=STRSG(3)
      XGOSH=SQRT(XGISH*XGISH+XGESH*XGESH)
      STRSP(1)=XGASH+XGOSH
      STRSP(2)=XGASH-XGOSH
      IF(XGISH.EQ.0.0) XGISH=0.1E-20
      STRSP(3)=ATAN(XGESH/XGISH)*28.647889757
C
C**** OUTPUT THE STRESS
C
      WRITE(6,915) KGASP,(GPCOD(IDIME,KGASP),
     *IDIME=1,NDIME),(STRSG(ISTR1),ISTR1=1,NSTR1),
     *(STRSP(ISTRE),ISTRE=1,NSTRE)
   50 CONTINUE
   60 CONTINUE
  900 FORMAT(/,10X,8HSTRESSES,/)
  910 FORMAT(/,5X,12HELEMENT NO.=,I5)
  915 FORMAT(I5,2F10.it,6E12.5,F10.4)
      RETURN
      END
```

BIBLIOGRAPHY

Boresi, A. P. and P. O. Lynn. Elasticity in Engineering Mechanics. Englewood Cliffs, New Jersey: Prentice-Hall, 1974.

D'Isa, Frank A. Mechanics of Metals. Reading, Massachusetts: Academic Press, 1977.

Gallagher, Richard H. Finite Element Analysis Fundaementals. Englewood Cliffs, New Jersy: Prentice-Hall, 1975.

Huebner, Kenneth L. The Finite Element Method for Engineers, New York: Wiley, 1975.

Irons, Bruce M. "A Frontal Solution Program for Finite Element and Analysis," International Journal for Numerical Methods in Engineering. Vol. 2, 1970, pp. 2-12.

Ural, Oktay. Finite Element Method. New York: Intext Press, 1973.

Zienkiewicz, O. C. The Finite Element Method. London: McGraw-Hill, 1977.

REFERENCES

Bathe, Klaue-Jurgen. _Numerical Methods in Finite Element Analysis_. Englewood Cliffs, New Jersey: Prentice-Hall, 1974.

Hinton, E., and D. R. J. Owen. _Finite Element Programming_. Reading, Massachusetts: Academic Press, 1977.