FINITE ELEMENT ANALYSIS OF A CONTINUUM UNDERGOING LARGE ELASTIC-PLASTIC DEFORMATION

By

Mohamad Majed Saleh, B.Eng

This thesis is submitted as a fulfilment of the requirement for the award of Master of Engineering (MEng) by research to

DUBLIN CITY UNIVERSITY

Sponsoring Establishment:

Scientific Studies and Research Centre, Damascus, Syria

August 1989

DECLARATION

All the work described in this thesis is my own, and no degree has been obtained on it from any other University or Institution.

Signed: ______________

Mohamad Majed Saleh

August 1989

CONTENTS

A B S T R A C T

NAME: FINITE ELEMENT ANALYSIS OF A CONTINUUM UNDERGOING LARGE ELASTIC-PLASTIC DEFORMATION

M. M. SALEH

In today's society there is a need for engineers to design to the limit of materials, with which they are working, because of industrial demands for more competative designs.

This thesis describes the work carried out to investigate the concept of the finite element method, to gain insight into the theory behind it and to apply this knowledge in developing a computer program to simulate the load mechanisms and boundary conditions, particularly to the ring structure under large elastic-plastic deformation.

Finite element is a method of mathematically modelling a component for stress analysis. It requires large quantities of data which are manipulated by matrix techniques to obtain results. The use of the computer is therefore essential to save time on a complex component.

The finite element program developed in this work is based on a two-dimensional plane elasticity analysis using constant strain triangular elements. Yield is based on Von-Mises' criterion, plastic flow on Prandtl-Reuss relationship and the formulation includes linear strain hardening. The formulation of the elastic-plastic matrix is based on the initial stress method.

The equipment for the experimental work was designed and this included the modification of the hydraulic system of the press machine, the base of testing and the measurement system.

Experimental work was carried out on the ring structure under three different types of loading conditions:

1. between two knife-edges;
2. between two rigid parallel surfaces;
3. between two rigid parallel surfaces and two lateral walls with a gap.

A comparison was made between the output data from the E.P. Program, which was developed in the current work, and the commercial packages. The results of this comparison are in reasonable agreement with each other. A comparison was also carried out between the experimental results and the theoretically predicted results, and reasonable agreement was obtained.

ACKNOWLEDGMENTS

I would like to thank Professor M.S.J. Hashmi for his supervision and guidance.

I would also like to thank the Staff, Mechanical Engineering Workshop; Technical Staff, School of Electronic Engineering; Research Staff, School of Mathematics; Staff, Computer Services; and Al Devine, for their help and assistance.

I am grateful to Mr. Thilaksri Gunawardhana in EOLAS for his assistance in running the PAFEC Finite Element Package.

I also acknowledge financial support from the Scientific Studies and Research Centre, Syria.

Many thanks to Mairead Callan, who kindly typed this thesis.

NOMENCLATURE

The following symbols have been chosen to be consistent with those used in the current work and will be used throughout this thesis:

[A]	Coefficient matrix associated with displacement function
[B]	Matrix relating element strains to element nodal displacements
[D]	Elasticity matrix
$[D]^*_{ep}$	Elastic-plastic matrix
E	Young's modulus of elasticity
F	Yielding function
G	Shear modulus
H'	Slope of equivalent stress/plastic strain curve
[K]	Stiffness matrix
k	Work hardening parameter
L_i, L_j, L_m	Area coordinates
P	External applied load
P_p	External plastic applied load
{P}-	Vector of nodal forces
$\{P'\}$	Vector of equilibriating body forces
q	Uniformaly distributed load
{R}	Vector of residual forces
u,v	Displacements along X and Y axes
$\{\delta'\}$	Indicates displacement vector associated with equilibriating body forces
W_i, W_e	Internal, external work done
X,Y,Z	Rectangular cartesian coordinate system
(x,y)	Indicates quantities are functions of X and Y

x_i, y_i	Indicates displacement associated with nod i
α_1, α_2, etc.	Constants used in displacement function
δ	Displacement
$\{\delta\}$	Vector of nodal displacements
ϵ	Strain
$\epsilon_x, \epsilon_y, \epsilon_z$	Direct strains
$\gamma_{xy}, \gamma_{yz}, \gamma_{zx}$	Shear strains
σ	Stress
σ_y	Yield Stress
σ_m	Hydrostatic stress
σ', ϵ'	Deviatoric stress, strain
$\sigma_{ij}, \epsilon_{ij}$	Stress, strain tensors (shorthand method of referring to array of nine quantities at a point)
$\bar{\sigma}, \bar{\epsilon}$	Equivalent stress, strain
$\{\Delta\sigma\}, \{\Delta\epsilon\}$	Vectors of stress, strain increments
$\tau_{xy}, \tau_{yz}, \tau_{zx}$	Shear stresses
ν	Poisson's ratio
Δ	Area of triangular element
Suffix e,p	Indicates elastic, plastic quantities
[ ]	Indicates a matrix
{ }	Indicates a one-dimensional array, row or column vector
$[^e], \{^e\}$	Matrix, vector relating to a single element
$[\ ]^{-1}$	Inverse of matrix
$[\ ]^T$	Transpose of matrix
M_{ij}	Bending moment
I	Moment of inertia

CHAPTER 1

INTRODUCTION

1.1 PREVIOUS WORK

The finite element method of structural analysis has emerged in the last twenty-five years as the method most widely used by engineers. The popularity of this technique is due to its wide applicability to both static and dynamic structural problems in elastic as well as plastic ranges. The structural analysis here is anything that is fabricated, manufactured or erected that must withstand an imposed load.

The concept of finite element has been in use for 150 years or more[1]. Certainly it is not a new feature in structural analysis. Southwell[2] employed a similar method in his work in 1935. That work was carried out by using beam-type elements. Clough et al[3] in 1956 first derived an element stiffness matrix for triangular element using a linear displacement function. Subsequently many investigators, e.g. Argyris[4], Melosh[5], Gallagher et al[6] and Zienkiewicz[7] have developed elements for different stress conditions with more refinement; covering bending and three-dimensional elements with triangles, rectangles, quadrilateral and tetrahedra. Argyris et al[8] have extended the method to elastic-plastic stress problems by making use of the so-called thermal strain approach, similar to that suggested by Mendelson and Manson[9]. Pope[10] suggested a tangent modulus approach for the solution of elastic-plastic problems by finite element.

Zienkiewcz et al[11] have developed a general formulation of the elastic- plastic matrix for evaluating stress increments. A new 'initial stress' computational process was proposed.

Hibbitt et al[12], have derived their finite element equilibrium equations from the principle of virtual work for large deformation. They identified four stiffness terms, which are called small strain stiffness, initial load stiffness, initial strain stiffness and initial stress stiffness. In elastic-plastic analysis, all of these must be calculated for each increment of deformation.

McMecking and Rice[13] have derived an eulerian finite element formulation for problems of large elastic-plastic flow. The formulation was given in a manner which allows any conventional finite element program, for small strain elastic-plastic analysis, to be simply and rigourously adapted to problems involving arbitrary amounts of deflection and arbitrary levels of stress in comparison to large plastic deformation.

Naylor[14] has analysed a number of problems by finite element using the displacement method to assess the stress accuracy at very low compressibilities. He found that materials which are virtually incompressible can be analysed by the conventional displacement method and accurate stresses obtained.

Weisgerber and Anand[15] have made a comparison between two solution techniques, specifically as they were used with Tresca yield condition, one was a tangent modulus approach for perfectly plastic materials including the strain hardening effects and the other was a modification of the initial stress concept. It was observed that the iterative scheme was significantly more efficient with regard to the needed computer time, but was less accurate than the first one.

Gortemaker and de Pater[16] have shown what the effect was of the different options in computer program upon the computer results. However, they have described numerical and experimental work which has been carried out on elastic-plastic problems involving large deformation. A finite element program has been developed for plane stress and plane strain problems. They have found that, in particular, attention would be paid to the effect that the inclusion of geometric nonlinearity in computations has upon the distribution of displacements, strains and stresses.

Okamotoa et al[17] have developed the theoretical method which gives a solution for non-linear contact problems by finite element method and load incremental theory. They have shown reasonable agreement with experimental data and other solutions.

Deb et al[18] have discussed a computer program of finite element for determining the collapse load under plastic deformation. They have also examined the elastic and plastic stress, strain, distribution and spreading of the plastic zones of a tension specimen with a semi-circular notch of small thickness. They have shown that the accuracy of the solution depends on the mesh pattern and size of the elements for which high speed computer facilities will give a better solution.

Johannes, et al [19] have developed the elastic-plastic finite element method to the so-called state determination which requires the integration of the constitutive equations. This has been examined by systematically applying and modifying algorithms for the solution of ordinary differential equations. As a result it was shown, that methods of higher order can be formulated for elastic-plastic problems, which are much more efficient than the first order algorithms used up to the present. Contro[20] has developed the formulations for elastic-plastic finite element analysis for bending plates. He has incorporated the variational approach into a quadratic programming problem linearly constrained, which he has found would be an efficient numerical tool.

In 1988, Chandra[21] solved the large inelastic deformation of the superplastic sheet metal forming process by a finite element method. The finite element program has been developed to account for the large deformation and strain through the use of a Lagrangian method. Results showed good agreement with available experimental data.

Overall, the finite element method has become the most powerful tool of analysis and is applicable to a wide range of problems.

The following Chapters describe some theoretical analysis of the concept of finite element computer programming supporting the theory and the experiment work which was carried out on the ring structure in different cases.

1.2 CURRENT WORK

Often the solution of engineering problems by conventional analytical methods can prove to be either too difficult or impossible, because the geometry or some other characteristic is irregular or arbitrary. Therefore, numerical techniques, which usually involve a number of repetitive operations making them ideal for solution by computer, are adapted to obtain the approximate solution. The intention of this work is to gain insight into the theory behind the finite element method and applying this knowledge in developing a computer program to solve two-dimensional stress problems in the field of mechanical engineering elastic-plastic design.

Several general purpose finite element packages are available, however, to solve a specialised problem it is often far more costly (in computer time) using these packages. Therefore, the best program for any particular problem usually is one written specifically for that application. Clearly, if a program is designed with a specific design problem in mind, which has most of the generalities of a general system, then this program would be suited to solve either the specific design problem in question or general stress problem.

In the current study the elastic-plastic finite element model is proposed employing constant strain triangular element in two-dimensional plane strain and plane stress formulation of the problem. The incremental displacement or load method can be used for non-linear analysis of the problem. The intention was to simulate the load mechanisms and boundary conditions particularly to the ring structure under large elastic-plastic deformation.

The use of the elastic-plastic finite element method permits one to trace the stress and strain history of each element as the structure is subjected to the load, both the elastic and plastic zones can be clearly identified. The program was tested to identify the accuracy and computer time for running, as well as to compare these with other packages.

The experimental work was carried out in order to establish the load/deflection relationship and the strain at certain point/ deflection relationship and compare these with the results from the program.

An assessment is made as to whether any modification to the current basic model would be justified for future work.

CHAPTER 2

THE FINITE ELEMENT METHOD

It is not intended to give a detailed account of the finite element method, as this is well documented in numerous text books [22,23,24], however, it is felt that a general overview will enhance the reader's understanding of subsequent work.

2.1 INTRODUCTION

It is not possible to obtain closed form mathematical solutions for many engineering problems because the geometry or some other characteristic is irregular or arbitrary. An analytical solution is a mathematical expression that gives the values of the desired unknown quantity at any location in a body, and as a consequence it is valid for an infinite number of locations in the body.

Analytical solutions can be obtained for only certain simplified situations. For this reason it has become necessary to seek approximate, and acceptable, numerical solutions to many problems particularly for components where there are certain non-standard features associated with, for example, the geometry and boundary conditions. The most widely adapted numerical technique in the field of continuim mechanics is probably the finite element method, as it is admirably suited to problems with the non-standard features mentioned above.

Applications of the method fall into three major catagories:

(i) Equilibrium or steady state or time independent problems;
(ii) Eigen value problems;
(iii) Propogation or transient problems.

The most frequent application is the solution of solid mechanic problems of Category (i), in which the steady state displacement or stress distribution is required. Typically in such an anlysis the continuum or region of interest is subdivided into a mesh of elements which have nodal points at their vertices and possibly other positions. In practice, the problem is then solved using a finite element computer program and this requires careful preparation of input data in the form of mesh, boundary conditions, material property and loading specification. The output consists of the unknown nodal quantities, e.g. displacements, stresses.

2.2 GENERAL PROCEDURE OF THE FINITE ELEMENT METHOD

The finite element method involves the division of the structure, into a number of discrete elements interconnected only at specified positions. These positions are referred to as nodes.

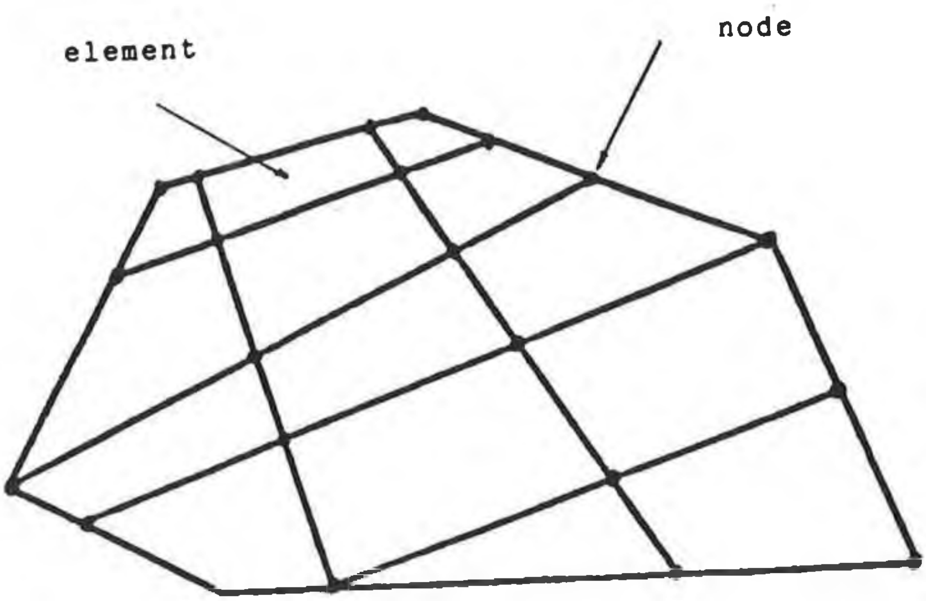
Figure (1) shows a component subdivided into a finite number of elements over its complete domain. The original structure may be assumed, therefore, to be replaced by an assembly of elements and these elements are assumed to be interconnected at a finite number of points, known as nodal points. These nodes occur at the corners of the elements (corner nodes) or along the element boundaries (edge nodes).

The type of element into which a component or structure is subdivided depends upon its geometry and they can be either one, two, or three-dimensional. If it is required to analyse a component with curved sides or faces, elements with curved sides must be used. Isoparametric elements with one or more edge nodes are one family of elements suitable for this purpose. Straight side elements do not necessarily require edge nodes but many have them, they are essential to isopatametric elements to accommodate the curvatures (see Appendix A). The location of the nodal lines on the component, and hence the form of the subdivision into elements depends on three main factors:

(i) Element boundary must coincide with structural discontinuities;

(ii) Nodal points must coincide with the points of application of concentrated loads;

(iii) The nodes must coincide with points on the structure where the displacements are to be calculated.



FIG(1)

component discetised into finite elements

When the component, or structure under consideration, has been subdivided into a finite number of elements, attention is focused, initially, on a single element. The objective is to obtain, for the element, an expression of the form:

$$\{p^e\} = [K^e]\{\delta^e\} \qquad (2.1)$$

relating the forces $\{p^e\}$, and the displacement, at its nodes, $\{\delta^e\}$, by means of its stiffness matrix $[K^e]$.

From the above discussion, it is evident that the choice of suitable displacement function is the most important part of the whole procedure. A good displacement function will lead to an element of high accuracy.

A displacement function is either given as:

(i) a simple polynomial with undetermined coefficients which are subsequently transformed to become the relevant nodal displacement parameters, or

(ii) directly in terms of shape functions which are physically associated with nodal displacement parameters, thus the displacement function can either be given (from (i)) as:

$$\delta = \alpha_1 + \alpha_2 x + \alpha_3 y + \ldots\ldots\ldots \qquad (2.2)$$

in which α_1, α_2, α_3, etc. are undetermined polynomial constants, or from (ii) as:

$$\delta = N_1(x,y)\ \delta_1 + N_2(x,y)\ \delta_2 + \ldots\ldots \qquad (2.3)$$

in which δ_1, δ_2, etc. are the nodal displacement parameters and N_1, N_2, etc. the corresponding shape functions (see Appendix B).

In general, a good displacement function should satisfy the following criteria:

(a) The displacement function, if given in the form of a simple polynomial, must have the same number of polynomial constants as the total number of degrees of freedom at a node.

(b) In most cases, the displacement function should be balanced with respect to all coordinate axes, since most elements are classified as general purpose elements, applicable to all types of problems.

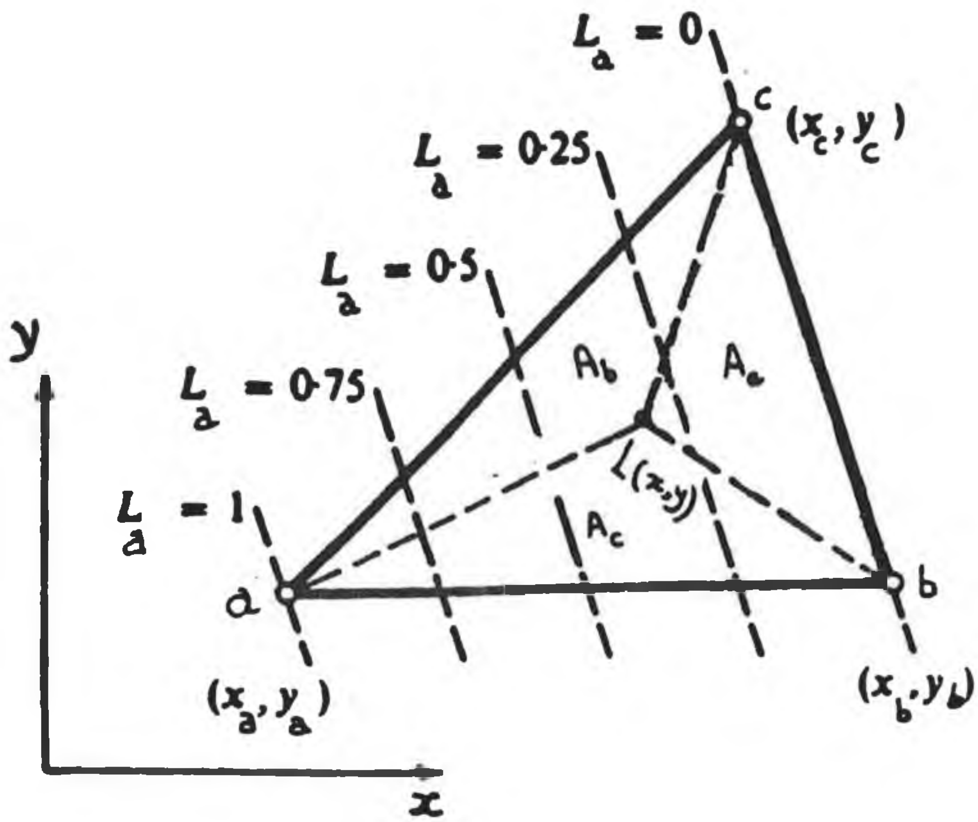
(c) The displacement function must allow the element to undergo rigid body movements.

(d) The displacement function must be able to represent a state of constant strain since this is the expected outcome if the elements are made smaller and smaller.

(e) The displacement function should satisfy the compatibility conditions along common boundaries between adjacent elements. For the current work, the displacement function in terms of the coordinate variables **x**, **y** and the nodal displacement parameter (e.g. U_a, V_a) is chosen to represent the displacement variations within each element and by using the principle of minimum total potential energy or the principle of virtual work.

2.2.1 Area Coordinates

All the previous discussions are concerned with cartesian (x,y) coordinates. For general quadrilateral elements (in Appendix A) with straight or curved sides similar functions can be used, but they should be in curvilinear (ξ,η) coordinates. The cartesian coordinates are also not very convenient for triangular elements, and a special type of coordinate system called area coordinate should be used. Referring to Figure (2) it is seen that the internal point L will divide the triangle (abc) into three small triangles and depending on the position of the point **l** the area of each one of the triangles **lac**, **lcb** and **lba** can vary from zero to Δ, which is the area of the triangle **abe**. In other words, the ratio Aa/Δ, Ab/Δ and Ac/Δ will take up any value between zero and unity. These ratios are called area coordinates, and they are defined by:



FIG(2)

triangle and the area coordinte.

$$\left.\begin{aligned} La &= Aa/\Delta = (a_a + b_a x + c_a y)/2\Delta \\ Lb &= Ab/\Delta = (a_b + b_b x + c_b y)/2\Delta \\ Lc &= Ac/\Delta = (a_c + b_c x + c_c y)/2\Delta \end{aligned}\right] \qquad (2.4)$$

(See Appendix B)

in which:

$$\left.\begin{aligned} a_a &= x_b y_c - x_c y_b \\ b_a &= y_b - y_c \\ c_a &= x_c - x_b \end{aligned}\right] \qquad (2.5)$$

and:

$$2\Delta = \det \begin{vmatrix} 1 & x_a & y_a \\ 1 & x_b & y_b \\ 1 & x_c & y_c \end{vmatrix} = 2\ (\text{area of triangle abc}) \qquad (2.6)$$

(see Appendix C)

x_a, y_a, etc. are the nodal coordinates and a_b, b_b, c_b, etc. can be computed through a cyclic permutation of the subscripts.

From equation (2.4), which in matrix form is:

$$\begin{bmatrix} L_a \\ L_b \\ L_c \end{bmatrix} = \frac{1}{2\Delta} \begin{vmatrix} a_a & b_a & c_a \\ a_b & b_b & c_b \\ a_c & b_c & c_c \end{vmatrix} \begin{bmatrix} 1 \\ x \\ y \end{bmatrix}$$

solving for 1, **x**, **y**

$$\begin{bmatrix} 1 \\ x \\ y \end{bmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ x_a & x_b & x_c \\ y_a & y_b & y_c \end{vmatrix} \begin{bmatrix} L_a \\ L_b \\ L_c \end{bmatrix}$$

or

$$\left.\begin{aligned} L_a + L_b + L_c &= 1 \\ x = L_a x_a + L_b x_b + L_c y_c & \\ y = L_a y_a + L_b y_b + L_c y_c & \end{aligned}\right] \qquad (2.6.1)$$

With the help of the area coordinates, it is now possible to establish a whole family of triangular elements, which is shown in Figure (3).

By using Lagrange polynomials it is a simple matter to construct the shape functions for the elements given in Figure (3) (see Appendix B). Thus, the second and third equations in equation (2.6.1) give the relationship between the cartesian coordinates and the area coordinates.

To every set, L_a, L_b, L_c corresponds a unique set of cartesian coordinates. At point **a**, $L_a = 1$ and $L_b = L_c = 0$, etc. a linear relation between the new and cartesian coordinates implies that contours of L_a are equally place straight lines parallel to side **b-c** on which $L_a = 0$, etc.

2.3 TRIANGULAR FINITE ELEMENT FOR PLANE ELASTICITY

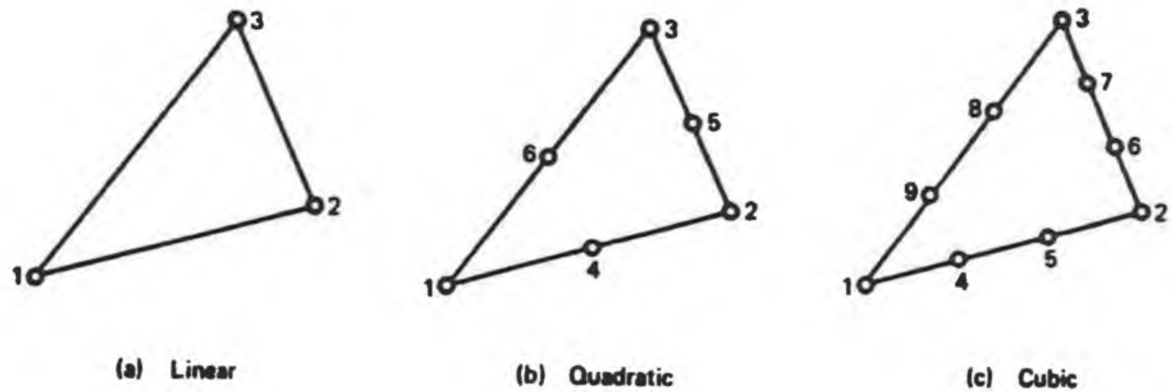
2.3.1 Nodal displacements and forces

One of the simplest types of element used in plane elasticity problems is the 3 noded triangular element as shown in Figure (4): its nodal positions are defined by a cartesian coordinate system.

Each node has two degrees of freedom and therefore, the complete vector of displacement and force

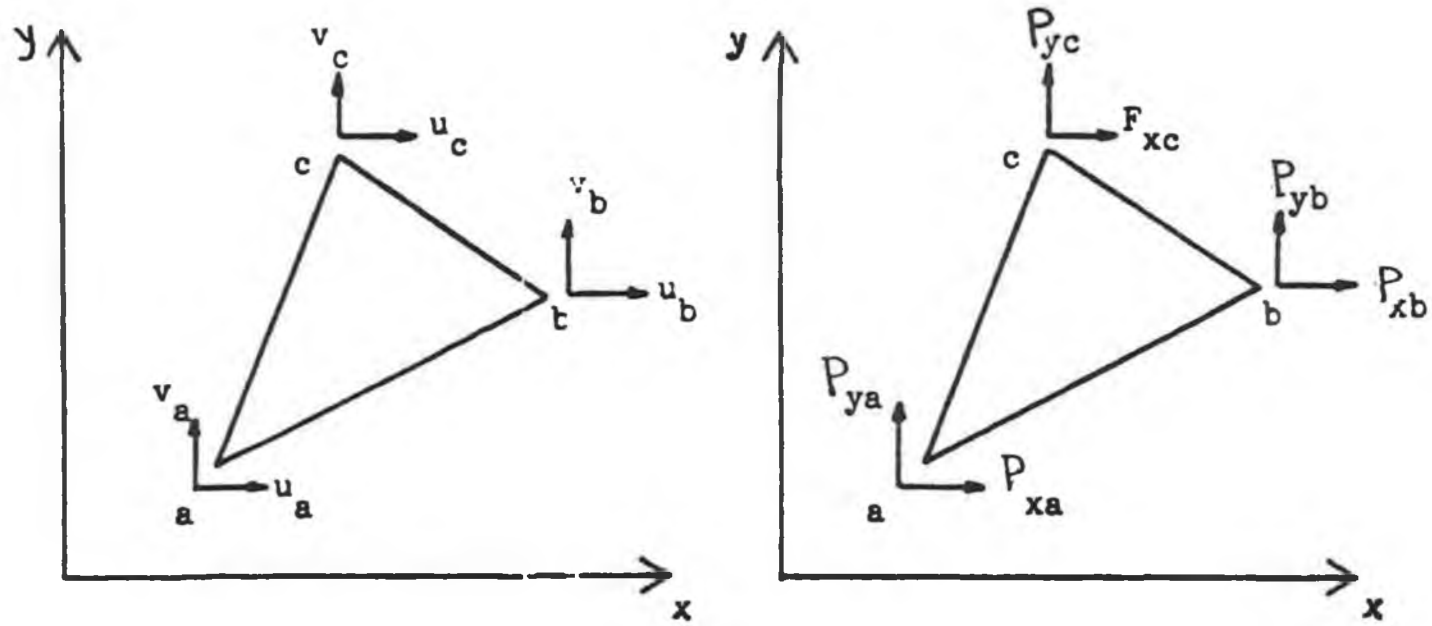
$$\{\delta^e\} = \begin{Bmatrix} u_a \\ v_a \\ u_b \\ v_b \\ u_c \\ v_c \end{Bmatrix} \qquad \{P^e\} = \begin{Bmatrix} P_{xa} \\ P_{ya} \\ P_{xb} \\ P_{yb} \\ P_{xc} \\ P_{yc} \end{Bmatrix} \qquad (2.7)$$

The nodal forces and displacements are related by equation (2.1)



FIG(3)

family of triangular element



FIG(4)

nodal displacement and forces.

2.3.2 Element displacement functions

The first step in the solution of the problem is the choice of a suitable polynomial to represent the displacement of the element under the action of the applied loading system. The simplest representation is given by the two linear polynomials.

$$\left.\begin{aligned} u &= \alpha_1 + \alpha_2 x + \alpha_3 y \\ v &= \alpha_4 + \alpha_5 x + \alpha_6 y \end{aligned}\right\} \qquad (2.8)$$

or in matrix form:

$$\{\delta(x,y)\} = \begin{Bmatrix} u \\ v \end{Bmatrix} = \begin{bmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \end{Bmatrix}$$

i.e.

$$\{\delta(x,y)\} = [f(x,y)]\ \{\alpha\} \qquad (2.9)$$

The coefficients $\{\alpha\}$ in the above equations are obtained by solving the simultaneous equations which result from substituting in turn the coordinates of the nodal points of the elements into the equations, i.e.

$$\begin{aligned} u_a &= \alpha_1 + \alpha_2 x_a + \alpha_3 y_a \\ v_a &= \alpha_4 + \alpha_5 x_a + \alpha_6 y_a \\ u_b &= \alpha_1 + \alpha_2 x_b + \alpha_3 y_b \\ v_b &= \alpha_4 + \alpha_5 x_b + \alpha_6 y_b \\ u_c &= \alpha_1 + \alpha_2 x_c + \alpha_3 y_c \\ v_c &= \alpha_4 + \alpha_5 x_c + \alpha_6 y_c \end{aligned} \qquad (2.10)$$

(See the proof for obtaining the value of coefficient $\{\alpha\}$ in Appendix B) The equation (2.10) may be expressed in matrix form as:

$$\{\delta^e\} = [A]\ \{\alpha\} \qquad (2.11)$$

or

$$\{\alpha\} = [A]^{-1} \{\delta^e\} \qquad (2.12)$$

in which:

$$[A] = \begin{vmatrix} 1 & x_a & y_a & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_a & y_a \\ 1 & x_b & y_b & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_b & y_b \\ 1 & x_c & y_c & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_c & y_c \end{vmatrix} \qquad (2.13)$$

Substitute equation (2.12) into equation (2.9). The displacement at any point in an element can now be determined in terms of the nodal displacements.

$$\{\delta\ (x,y)\} = [f(x,y)]\ [A]^{-1}\ \{\delta^e\} \qquad (2.14)$$

2.3.3 Relate the element strains to displacements

For plane elasticity problems the strain vector is:

$$\{\epsilon\ (x,y)\} = \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \delta_{xy} \end{Bmatrix} = \begin{Bmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial v}{\partial y} \\ \dfrac{\partial v}{\partial x} + \dfrac{\partial u}{\partial y} \end{Bmatrix} \qquad (2.15)$$

and substituting for u and v from equation (2.8) into the strain expressions i.e.

$$\left.\begin{aligned} \epsilon_x &= \alpha_2 \\ \epsilon_y &= \alpha_6 \\ \alpha_{xy} &= \alpha_3 + \alpha_5 \end{aligned}\right\} \qquad (2.16)$$

or in matrix form

$$\{\epsilon\ (x,y)\} = [C]\ \{\alpha\} \qquad (2.17)$$

where :

$$[C] = \begin{vmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{vmatrix} \qquad (2.18)$$

Substituting for $\{\alpha\}$ from equation (2.12)

$$\{\epsilon\ (x,y)\} = [C]\ [A]^{-1}\ \{\delta^e\} \qquad (2.19)$$

which may be written as:

$$\{\epsilon\ (x,y)\} = [B]\ \{\delta^e\} \qquad (2.20)$$

where:

$$[B] = [C]\ [A]^{-1} \qquad (2.21)$$

Equation (2.20) relates the strains $\{\epsilon(x,y)\}$ within the element to the nodal displacements $\{\delta^e\}$. It can be noted that the [B] matrix for a three noded straight sided triangular element consists of fixed values only determined by the nodal coordinates. Thus the strain at any point within the element is constant. For this reason these elements are often referred to as constant strain element (CST) (see Appendix B) for determination of [B] matrix).

2.3.4 Relate the internal stresses to the displacements

The two cases of plane elasticity, (plane stress and plane strain) will be discussed separately since the stress-strain relationships are different for the two cases only isoptropic material will be considered here. Normally, plane stress is applied to members which are relatively thin in comparison to their other dimensions, whereas plane strain is applied to relatively thick members.

(a) plane stress:

In plane stress problems only three stress components ($\sigma_x, \sigma_y, \tau_{xy}$) within the x-y plane are present, the other three components (σ_z, τ_{yz}, τ_{zx}) being equal to zero. The stress-strain equations for three-dimensional elasticity are thus reduced to:

$$\sigma_x = \frac{E}{(1+\nu)(1-2\nu)} [(1-\nu)\epsilon_x + \nu\epsilon_y + \nu\epsilon_z] \quad (2.22)$$

$$\sigma_y = \frac{E}{(1+\nu)(1-2\nu)} [\nu\epsilon_x + (1-\nu)\epsilon_y + \nu\epsilon_z] \quad (2.23)$$

$$\sigma_z = \frac{E}{(1+\nu)(1-2\nu)} [\nu\epsilon_x + \nu\epsilon_y + (1-\nu)\epsilon_z] \quad (2.24)$$

$$\tau_{xy} = \frac{E}{2(1+\nu)} \gamma_{xy} \quad (2.25)$$

from equation (2.24):

$$\epsilon_z = -\frac{\nu}{(1-\nu)} (\epsilon_x + \epsilon_y) \quad (2.26)$$

Substitute the value of equation (2.26) with the value of equation (22):

$$\sigma_x = \frac{E}{1-\gamma^2} (\epsilon_x + \gamma\epsilon_y) \tag{2.27}$$

Substitute the value of equation (2.26) with the value of equation (2.23):

$$\sigma_y = \frac{E}{1-\gamma^2} (\gamma\epsilon_x + \epsilon_y) \tag{2.28}$$

from equations (2.27), (2.28) and (2.25):

$$\left.\begin{aligned} \sigma_x &= \frac{E}{1-\gamma^2} (\epsilon_x + \gamma\epsilon_y) \\ \sigma_y &= \frac{E}{1-\gamma^2} (\gamma\epsilon_y + \epsilon_y) \\ \tau_{xy} &= \frac{E}{2(1-\gamma)} \gamma_{xy} \end{aligned}\right| \tag{2.29}$$

equation (2.29) expressed in matrix form is:

$$\begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix} = \begin{vmatrix} \frac{E}{1-\gamma^2} & \frac{\gamma E}{1-\gamma^2} & 0 \\ \frac{\gamma E}{1-\gamma^2} & \frac{E}{1-\gamma^2} & 0 \\ & & \frac{E}{2(1+\gamma)} \end{vmatrix} \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{bmatrix} \tag{2.30}$$

which may be written as:

$$\{\sigma(x,y)\} = [D] \{\epsilon(x,y)\} \tag{2.31}$$

(b) plane strain:

Unlike plane stress it is ϵ_z and not σ_z which is equal to zero in plane strain. Because of this it is more convenient to express the strains in terms of stresses, i.e.:

$$\epsilon_x = \frac{1}{E}(\sigma_x - \gamma\sigma_y - \gamma\sigma_z) \qquad (2.32)$$

$$\epsilon_y = \frac{1}{E}(-\gamma\sigma_x + \sigma_y - \gamma\sigma_z) \qquad (2.33)$$

$$\epsilon_z = 0 = \frac{1}{E}(-\gamma\sigma_x - \gamma\sigma_y + \sigma_z) \qquad (2.34)$$

$$\gamma_{xy} = \frac{2(1+\gamma)}{E}\tau_{xy} \qquad (2.35)$$

From equation (2.34):

$$\sigma_z = \gamma(\sigma_z + \sigma_y) \qquad (2.36)$$

The equations (2.32), (2.33) and (2.35) may be written in matrix form:

$$\begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} = \frac{1}{E}\begin{bmatrix} (1-\gamma^2) & -\gamma(1+\gamma) & 0 \\ -\gamma(1+\gamma) & (1-\gamma^2) & 0 \\ 0 & 0 & 2(1+\gamma) \end{bmatrix}\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = [D]^{-1}\{\sigma\} \qquad (2.37)$$

Substitute the value of equation (2.36) with the value of equation (2.32):

$$\epsilon_x = \frac{1}{E}[(\sigma_x - \gamma\sigma_y - \gamma^2(\sigma_x + \sigma_y))] \qquad (2.38)$$

Substitute the value of equation (2.36) with the value of equation (2.33):

$$\epsilon_y = \frac{1}{E} [(-y\sigma_x + \sigma_y - y^2\sigma_x - y^2\sigma_y)] \tag{2.39}$$

Equation (2.38) may be written as:

$$\epsilon_x = \frac{1}{E} [(\sigma_x (1-y^2) - \sigma_y(y + y^2)] \tag{2.40}$$

From equation (2.40):

$$\sigma_x = \frac{y}{1-y} \sigma_y + \frac{E}{1-y^2} \epsilon_x \tag{2.41}$$

In the same way, from equation (2.39):

$$\sigma_y = \frac{y}{1-y} \sigma_x + \frac{E}{1-y^2} \epsilon_y \tag{2.42}$$

Substitute the value of equation (2.42) with the value of equation (2.41) and the result is:

$$\sigma_x = \frac{E(1-y)}{(1+y)(1-2y)} \epsilon_x + \frac{E\ y}{(1+y)(1-2y)} \epsilon_y \tag{2.43}$$

Substitute the value of equation (2.41) with the value of equation (2.42). The result is:

$$\sigma_y = \frac{E\ y}{(1+y)(1-2y)} \epsilon_x + \frac{E\ (1-y)}{(1+y)(1-2y)} \epsilon_y \tag{2.44}$$

From equation (2.35) is:

$$\tau_{xy} = \frac{E\gamma_{xy}}{2(1+\nu)} \qquad (2.45)$$

The equations (2.43), (2.44) and (2.45) may be written as:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \begin{bmatrix} \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} & \frac{E\nu}{(1+\nu)(1-2\nu)} & 0 \\ \frac{E\nu}{(1+\nu)(1-2\nu)} & \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} & 0 \\ 0 & 0 & \frac{E}{2(1+\nu)} \end{bmatrix} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} \qquad (2.46)$$

which can be expressed as:

$$\{\sigma(x,y)\} = [D]\ \{\epsilon(x,y)\} \qquad (2.47)$$

and thus the elasticity matrix [D] has been established for both plane stress and plane strain.

Substituting the value of $\{\epsilon(x,y)\}$ from equation (2.20) the following relationship between the element stresses and nodal displacements is obtained:

$$\{\sigma(x,y)\} = [D]\ [B]\ \{\delta^e\} \qquad (2.48)$$

The equation (2.48) is valid for both plane stress and plane strain.

The two elasticity matrices (2.30) and (2.46) have the same form and it is more convenient to present them by common elasticity matrix:

$$[D] = \begin{vmatrix} d_1 & d_2 & 0 \\ d_2 & d_1 & 0 \\ 0 & 0 & d_3 \end{vmatrix}$$

Where for plane stress

$$d_1 = \frac{E}{(1-y^2)} \quad ; d_2 = \frac{yE}{(1-y^2)} \quad ; d_3 = \frac{E}{2(1-y)} \qquad (2.48)$$

and for plane strain:

$$d_1 = \frac{E(1-y)}{(1-y)(1-2y)} \quad ; d_2 = \frac{yE}{(1+y)(1-2y)} \quad ; d_3 = \frac{E}{2(1-y)} \qquad (2.49)$$

2.3.5 Determination of element stiffness matrix

The internal stresses $\{\sigma(x,y)\}$ are now replaced by statically equivalent nodal loads $\{p^e\}$ and hence the nodal loads are related to the nodal displacements $\{\delta^e\}$ thereby defining the required element stiffness matrix $[K^e]$.

The principle of virtual work is used to determine the set of nodal loads that is statically equivalent to the internal stresses. The condition of equivalence may be expressed as follows: during any virtual displacement imposed on the element, the total external work done by the nodal load must equal the total internal work done by stresses, i.e.

$$\Sigma P\delta = \int_{(v)} \sigma \ \epsilon \ d(vol) \qquad (2.50)$$

where:

σ - the internal stress

δ - virtual displacement

ϵ - strain change due to δ

An arbitrary set of virtual nodal displacements is represented by the vector $\{\delta^e\}$. The external work done by the nodal loads W_E is given by:

$$W_E = \{d\delta^e\} \ \{p^e\} \qquad (2.51)$$

If the arbitrarily imposed displacements cause strains $\{d\epsilon(x,y\}$ at a point within the element where the actual stresses are $\{\sigma(x,y)\}$ then the total internal work done W_I is:

$$W_I = \int_{(v)} \{d\epsilon(x,y)\}^T \ \{\sigma(x,y)\} \ d(vol) \qquad (2.52)$$

which is the change of strain energy of the element. Substituting equations (2.20) and (2.47) gives:

$$W_I = \int_{(v)} [B]^T \ \{d\delta^e\} \ [D] \ [B] \ \{\delta^e\} \ d(vol) \qquad (2.53)$$

Assuming that the unit values of nodal displacements are applied, then equating the internal and external work gives:

$$\{P^e\} = [\int_{(v)} [B]^T \ [D] \ [B] \ d(vol)] \ \{\delta^e\} \qquad (2.54)$$

Comparing this with equation (2.1), it follows that the element stiffness matrix $[K^e]$ is the expression inside the square bracket, i.e.

$$[K^e] = \int_{(v)} [B]^T [D] [B] \, d(vol) \qquad (2.55)$$

Since the matrices [B] and [D] contain only constant terms they can be taken outside the integration leaving only $\int_{(v)} d(vol)$ which, in the case of an element of constant thickness which equals the area of the triangle Δ multiplied by its thickness t, thus:

$$[K^e] = [B]^T [D] [B] \Delta . t \qquad (2.56)$$

equation (2.6) gives:

$$\Delta = \frac{1}{2} [(x_b y_c - x_c y_b) - (x_a y_c - x_c y_a) + (x_a y_b - x_b y_a)] \qquad (2.57)$$

Having performed this calculation for each element of the complete structure, the overall stiffness matrix [K] is assembled using the overall load [p] equation (2.58) below, which can solve the unknown displacements $\{\delta\}$ at the structural nodes.

$$\{\delta\} = [K]^{-1} [p] \qquad (2.58)$$

Hence, using equation (2.20) and (2.47), strains and stresses respectively can be obtained.

CHAPTER 3

ELASTIC-PLASTIC PROBLEMS

The complete solution of a general problem in plasticity involves a calculation of the stress and the deformation in both the elastic and plastic regions. It has been shown that in the former the stress is directly connected with total strain by means of the elastic equations. In the latter there is no such unique correspondence and the process of plastic deformation has to be considered mathematically as a succession of small increments of strain. In order to illustrate this it is necessary to introduce elements of plasticity theory.

3.1 BASIC CONCEPTS OF PLASTICITY

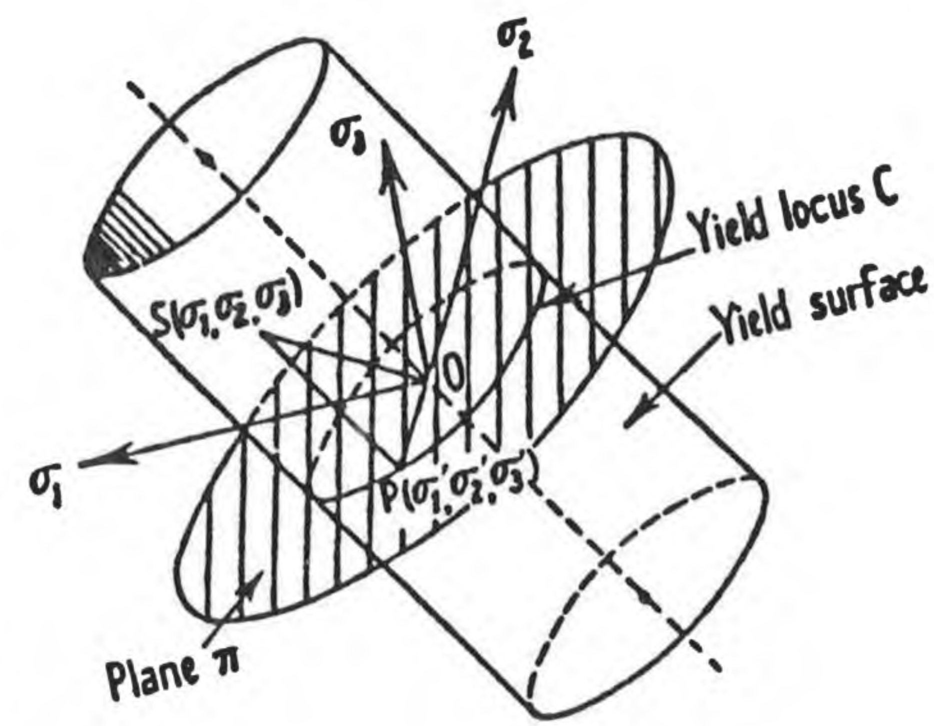
3.1.1 Yield criterion

A yield criterion is a hypothesis concerning the limit of elasticity under any possible combination of stresses.

Figure (5) is a geometrical representation of a three-dimensional stress system. The stress state is represented by a vector in three-dimensional space where the principal stresses are taken as cartesian coordinates. **OS** is the vector $\{\sigma_1, \sigma_2, \sigma_3\}$, while **OP** is the vector representing the deviatoric stress $[\sigma_1', \sigma_2', \sigma_3']$. OP always lies in the plane π whose equation is:

$$\sigma_1 + \sigma_2 + \sigma_3 = 0$$

PS represents the hydrostatic component $\{\sigma_m, \sigma_m, \sigma_m\}$ of the stress. Since yielding is independent of the hydrostatic component stress it is evident that the yield surface in this spatial system is a right cylinder with generators perpendicular to π and cutting it in some curve **C**, the yield locus.



FIG(5)

geometrical representation of three dimentional stress system.

Von Mises' criterion states that the locus is a circle in the π (or deviatoric) plane defined by:

$$\sigma'_{ij}\,\sigma'_{ij} = \text{constant} \qquad (3.1)$$

remembering:

$$\sigma'_{ij} = \sigma_{ij} - \delta_{ij}\,\sigma_m \qquad (3.2)$$

Where the hydrostatic component of stress, $\sigma_m = \sigma_{ii}/3$ and kroneker delta symbol, δ_{ij}, is equal to unity when $i = j$ and zero when $i \neq j$. Hence the yield criterion can be defined as:

$$F(\sigma_{ij},k) = 0 \qquad (3.3)$$

The hypothesis that the radius of Von Mises' circle is a function only of the plastic work, W_p, is written as:

$$\bar{\sigma} = \left[\left(\frac{3}{2}\right)(\sigma'_{ij}\,\sigma'_{ij})\right]^{0.5} = W(W_p) \qquad (3.4)$$

Thus defining the equivalent stress $\bar{\sigma}$, it is convenient to define a yielding function, F, whereby:

$$F = \bar{\sigma} - \bar{\sigma}_y \qquad (3.5)$$

or more fully:

$$F = 0.5\,[(\sigma_x-\sigma_y)^2+(\sigma_y-\sigma_z)^2+(\sigma_z-\sigma_x)^2]+3(\tau_{xy}^2+\tau_{yz}^2+\tau_{zx}^2)]^{0.5}-\bar{\sigma}_y \qquad (3.6)$$

where the numerical factor has been chosen such that $\bar{\sigma}_y = \bar{\sigma}_y(k)$ is the uniaxial stress at yield dependent on the instantaneous value of a hardening parameter k.

On a similar basis the equivalent plastic strain increment can be defined as:

$$d\bar{\epsilon}^p = [(\frac{2}{3}) (d\epsilon'_{ij}{}^p \, d\epsilon'_{ij}{}^p)]^{0.5} \qquad (3.7)$$

where $d\epsilon_{ij}{}^p$ is the increment of plastic strain. It can be shown that the increment of plastic work per unit value is given by:

$$dW_p = \sigma'_{ij} \, d\epsilon_{ij}{}^p \qquad (3.8)$$

The equivalent plastic strain and total plastic work are integrals over the history of deformation.

3.1.2 Flow rule

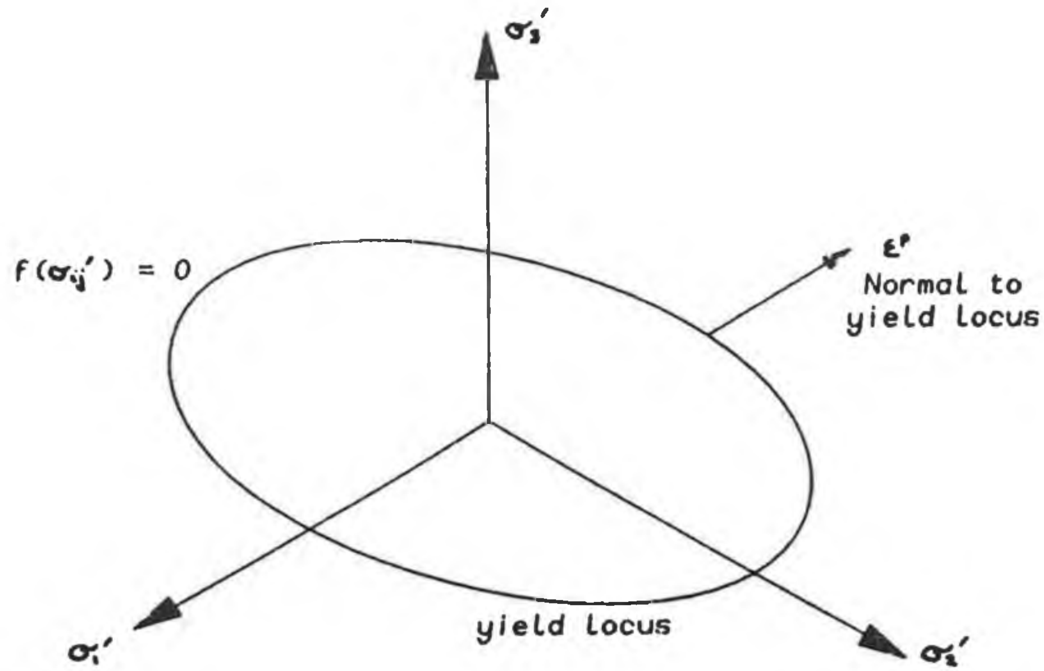
A plastic flow rule defines the direction of the plastic strain increment carrying through the assumption that no plastic work is done by the hydrostatic components of stress. It can be assumed that there is no permanent change in the volume. Hence, since $d\epsilon_1{}^p + d\epsilon_2{}^p + d\epsilon_3{}^p = 0$ the plastic strain increment can be represented by a vector in the π-plane, thus:

$$d\epsilon_{ij}{}^p \, \alpha \, \sigma'_{ij}$$

or:

$$d\epsilon_{ij}{}^p = \lambda \, \sigma'_{ij} \qquad (3.9)$$

where λ is an undetermined constant of proportionality. The rule is known as the normality principal because the relation can be interpreted as requiring the normality of the plastic strain increment to the yield locus. This is illustrated in Figure (6). This restriction of the rule makes it valid only for the particular case known as associated plasticity. However, it has proved to be a good approximation for most metals.



FIG(6)

plastic flow Rule " normality principle"

3.1.3 Hardening law

The hardening law specifies how the yield locus changes with either plastic strain or plastic work. Work hardening has been stated previously in equation (3.4). Strain hardening can be stated as a function of plastic strain thus:

$$\bar{\sigma} = H(\bar{\epsilon}^{p}) \qquad (3.10)$$

For the purpose of this work isotropic strain hardening is assumed. The implication here is that the material is represented by a circular yield locus which expands with strain and stress history, retaining the same shape throughout.

3.1.4 Prandtl-Reuss equations

The Prandtl-Reuss equations are an extension of the earlier Levy-Mises equations which were strictly applicable only to a fictitious material in which the elastic strains are zero. In reality, during an infinitesimal increment of stress, changes of strain are divisible into elastic and plastic parts, thus:

$$d\epsilon'_{ij} = d\epsilon'^{p}_{ij} + d\epsilon'^{e}_{ij} \qquad (3.11)$$

Substituting the flow rule from equation (3.9) and incorporating the basic elastic constitutive equation gives:

$$d\epsilon'_{ij} = \lambda\,\sigma'_{ij} + d\sigma'_{ij}\,/\,(2G)$$

$$d\epsilon_{ii} = 0 + [(1-2\gamma)/E]\,d\sigma_{ii} \qquad (3.12)$$

Considering the simple case of uniaxial tension, using equations (3.4), (3.8) and (3.10)it can easily be shown that:

$$\lambda = 3d\bar{\epsilon}^{p}\,/\,2\bar{\sigma}$$

or:

$$\lambda = 3d\bar{\sigma}\,/\,H' \qquad (3.13)$$

where $H' = d\bar{\sigma}/ d\bar{\epsilon}^p$ is the slope of the equivalent stress/plastic strain curve. Thus, the complete incremental stress-strain equations for an elastic-plastic material can be stated as:

$$d\epsilon'_{ij} = 3\sigma'_{ij}d\bar{\sigma}/2\bar{\sigma} H' + d\sigma'_{ij}/2G$$
$$d\epsilon_{ii} = 1\text{-}2\gamma/E \quad d\sigma_{ii} \qquad (3.14)$$

A typical strain load increment is illustrated graphically in Figure (7).

3.2 THE ELASTIC-PLASTIC MATRIX

It has been shown that there is no direct stress-strain relationship for a material undergoing plastic deformation, i.e.

$$\sigma \neq \sigma(\epsilon) \qquad (3.15)$$

On the basis of the Prandtl-Reuss and the Von Mises yield criterion it can be shown that:

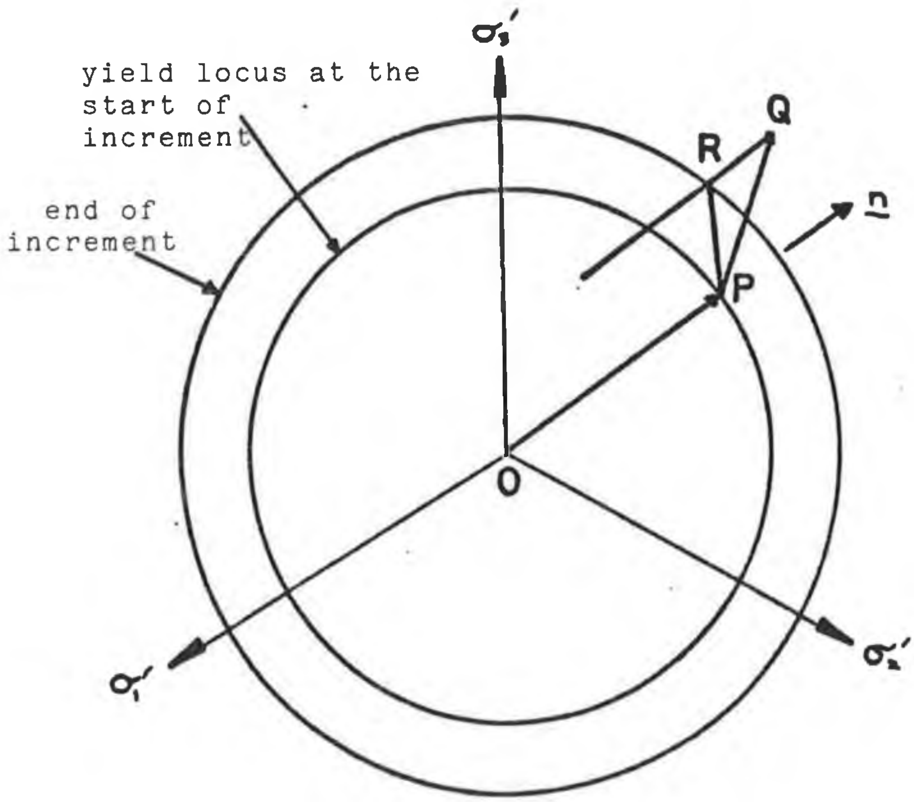
$$F\ (\sigma,k) = 0 \qquad (3.16)$$

$$\delta\epsilon_p = \lambda \frac{\partial F}{\partial \sigma} \qquad (3.17)$$

$$\delta\epsilon = \delta\epsilon^e + \delta\epsilon^p \qquad (3.18)$$

The elastic strain increments are related to stress increments by symetric matrix of constants D known as the 'elastic matrix'. Thus:

$$\delta\epsilon^e = D^{-1}\ \delta\sigma$$



PQ=total strainincrement
RQ=plastic strain increment
PR=elastic strain increment
n =unit vector

FIG(7)

typical strain increment assuming isotropic hardening

Substituting (3.18), i.e.

$$\delta\epsilon = D^{-1}\delta\sigma + \frac{\partial F}{\partial \sigma}\lambda \qquad (3.19)$$

When plastic yield is occuring the stresses are on the yield surface given by (3.16), differentiating thus:

$$0 = \frac{\partial F}{\partial \sigma_1}\delta\sigma_1 + \frac{\partial F}{\partial \sigma_2}\delta\sigma_2 + \ldots\ldots + \frac{\partial F}{\partial k}dk$$

$$0 = \left[\frac{\partial F}{\partial \sigma}\right]^T \delta\sigma - A\lambda \qquad (3.20)$$

where:

$$A = -\frac{\partial F}{\partial K}dk\frac{1}{\lambda} \qquad (3.21)$$

writing (3.19) and (3.20) in single symmetric form as:

$$\begin{bmatrix} \delta\epsilon \\ 0 \end{bmatrix} = \begin{vmatrix} D^{-1} & \dfrac{\partial F}{\partial \sigma} \\ \left(\dfrac{\partial F}{\partial \sigma}\right)^T & -A \end{vmatrix}$$

Multiply first equation (i.e. (3.19) by $\left(\frac{\partial F}{\partial \sigma}\right)^T D$ i.e.

$$\left(\frac{\partial F}{\partial \sigma}\right)^T D\,\delta\epsilon = \left(\frac{\partial F}{\partial \sigma}\right)^T \delta\sigma + \left(\frac{\partial F}{\partial \sigma}\right)^T D\frac{\partial F}{\partial \sigma}\lambda$$

or:

$$\left(\frac{\partial F}{\partial \sigma}\right)^T \delta\sigma = \left(\frac{\partial F}{\partial \sigma}\right)^T D\delta\epsilon - \left(\frac{\partial F}{\partial \sigma}\right)^T D \frac{\partial F}{\partial \sigma} \lambda \qquad (3.22)$$

Substitute the value of equation (3.22) with the value of equation (3.20):

$$\left(\frac{\partial F}{\partial \sigma}\right)^T D\delta\epsilon - \left(\frac{\partial F}{\partial \sigma}\right)^T D \frac{\partial F}{\partial \sigma} \lambda - A\lambda = 0$$

$$\left(\frac{\partial F}{\partial \sigma}\right)^T D\delta\epsilon - \left[\left(\frac{\partial F}{\partial \sigma}\right)^T D \frac{\partial F}{\partial \sigma} + A\right]\lambda = 0 \qquad (3.23)$$

Using equations (3.22) and (3.23) to eliminate λ (i.e. (3.23) into (3.22) :

$$\left(\frac{\partial F}{\partial \sigma}\right)^T \delta\sigma = \left(\frac{\partial F}{\partial \sigma}\right)^T D\delta\epsilon - \left(\frac{\partial F}{\partial \sigma}\right)^T D \frac{\partial F}{\partial \sigma} \cdot \left(\frac{\partial F}{\partial \sigma}\right)^T D\delta\epsilon \left[A + \left(\frac{\partial F}{\partial \sigma}\right)^T D \frac{\partial F}{\partial \sigma}\right]^{-1}$$

$$\delta\sigma = \left(D - D\frac{\partial F}{\partial \sigma} \left(\frac{\partial F}{\partial \sigma}\right)^T D \left[A + \left(\frac{\partial F}{\partial \sigma}\right)^T D \frac{\partial F}{\partial \sigma}\right]^{-1}\right) \delta\epsilon$$

or:

$$\delta\sigma = [D]^*_{ep}\, \delta\epsilon \qquad (3.24)$$

Where the elastic-plastic matrix is:

$$[D]^*_{ep} = [D] - [D] \left\{\frac{\partial F}{\partial \sigma}\right\} \left\{\frac{\partial F}{\partial \sigma}\right\}^T [D] \left[A + \left\{\frac{\partial F}{\partial \sigma}\right\}^T [D] \left\{\frac{\partial F}{\partial \sigma}\right\}\right]^{-1} \quad (3.25)$$

Equations (3.19) and (3.20) can be written in a single symmetric matrix form as:

$$\begin{bmatrix} \delta\epsilon_1 \\ \delta\epsilon_2 \\ \cdot \\ \cdot \\ - \\ 0 \end{bmatrix} = \left[\begin{array}{cc:c} & & \frac{\partial F}{\partial \sigma_1} \\ & [D]^{-1} & \frac{\partial F}{\partial \sigma_2} \\ & & \cdot \\ \hdashline \frac{\partial F}{\partial \sigma_1} & \frac{\partial F}{\partial \sigma_2} & \cdot \quad -A \end{array}\right] \begin{bmatrix} \delta\sigma_1 \\ \delta\sigma_2 \\ \cdot \\ \cdot \\ - \\ \lambda \end{bmatrix} \quad (3.26)$$

3.2.1 Special forms of the elastic-plastic relationship

Most generally the yield criterion is established in the 'six-dimensional' stress space as a function of all the six stress components. When dealing with more restrictive problems such as prescribed by cases of plane strain, plane stress or axial symmetry an appropriate specialization of the yield surface to the more limited freedom has to be made.

Consider the general relationship (3.26) written in terms of the six three-dimensional stress components listed as:

$\sigma_1 = \sigma_x$, $\sigma_2 = \sigma_y$, $\sigma_3 = \sigma_z$, $\sigma_4 = \tau_{xy}$, $\sigma_5 = \tau_{yz}$, $6 = \tau_{zx}$
$\epsilon_1 = \epsilon_x$, ...etc.

(i) plane strain:

In plane strain the components (τ_{yz}, τ_{zx} and ϵ_z) become zero, thus:

$$\delta\epsilon_x = \frac{1}{E}\,\delta\sigma_x - \frac{\gamma}{E}\,\delta\sigma_y - \frac{\gamma}{E}\,\delta\sigma_z + \frac{\partial F}{\partial \sigma}\,\lambda \qquad (3.27)$$

$$\delta\epsilon_y = -\frac{\gamma}{E}\,\delta\sigma_x + \frac{1}{E}\,\delta\sigma_y - \frac{\gamma}{E}\,\delta\sigma_z + \frac{\partial F}{\partial \sigma_y}\,\lambda \qquad (3.28)$$

$$\delta\gamma_{xy} = \frac{2(1+\gamma)}{E}\,\delta\tau_{xy} + \frac{\partial F}{\partial \tau_{xy}}\,\lambda \qquad (3.29)$$

$$\delta\epsilon_z = 0 = -\frac{\gamma}{E}\,\delta\sigma_x - \frac{\gamma}{E}\,\delta\sigma_y + \frac{1}{E}\,\delta\sigma_z + \frac{\partial F}{\partial \sigma_z}\,\lambda \qquad (3.30)$$

$$0 = \frac{\partial F}{\partial \sigma_x}\,\delta\sigma_x + \frac{\partial F}{\partial \sigma_y}\,\delta\sigma_y + \frac{\partial F}{\partial \tau_{xy}}\,\delta\tau_{xy} + \frac{\partial F}{\partial \sigma_z}\,\delta\sigma_z - A\lambda \qquad (3.31)$$

From equation (3.30)

$$\delta\sigma_z = \gamma\delta\sigma_x + \gamma\delta\sigma_y - E\,\frac{\partial F}{\partial \sigma_z}\,\lambda \qquad (3.32)$$

Substitute the value of equation (3.32) with the value of equations (3.27), (3.28) and (3.31):

$$\delta\epsilon_x = \frac{1}{E}\,\delta\sigma_x - \frac{\gamma}{E}\,\delta\sigma_y - \frac{\gamma^2}{E}\,\delta\sigma_x - \frac{\gamma^2}{E}\,\delta\sigma_y + \gamma\,\frac{\partial F}{\partial \sigma_z}\,\lambda + \frac{\partial F}{\partial \sigma_x}\,\lambda$$

$$= \left(\frac{1-y^2}{E}\right)\delta\sigma_x - y\left(\frac{1+y}{E}\right)\delta\sigma_y + \left(\frac{\partial F}{\partial\sigma_x} + y\frac{\partial F}{\partial\sigma_z}\right)\lambda \qquad (3.33)$$

$$\delta\epsilon_y = -y\left(\frac{1+y}{E}\right)\delta\sigma_x + \left(\frac{1-y^2}{E}\right)\delta\sigma_y + \left(\frac{\partial F}{\partial\sigma_y} + y\frac{\partial F}{\partial\sigma_z}\right)\lambda \qquad (3.34)$$

$$0 = \left(\frac{\partial F}{\partial\sigma_x} + y\frac{\partial F}{\partial\sigma_z}\right)\delta\sigma_x + \left(\frac{\partial F}{\partial\sigma_y} + y\frac{\partial F}{\partial\sigma_z}\right)\delta\sigma_y + \frac{\partial F}{\partial\tau_{xy}}\delta\tau_{xy} - \left[E\left(\frac{\partial F}{\partial\sigma_z}\right)^2 + A\right]\lambda \qquad (3.35)$$

or in matrix form:

$$\begin{bmatrix} \delta\epsilon_x \\ \delta\epsilon_y \\ \delta\gamma_{xy} \\ 0 \end{bmatrix} = \begin{bmatrix} 1-y^2/E & -y(1+y)/E & 0 & \left(\frac{\partial F}{\partial\sigma_x} + y\frac{\partial F}{\partial\sigma_z}\right) \\ -y(1+y)/E & 1-y^2/E & 0 & \left(\frac{\partial F}{\partial\sigma_y} + y\frac{\partial F}{\partial\sigma_z}\right) \\ 0 & 0 & 2(1+y)/E & \frac{\partial F}{\partial\tau_{xy}} \\ \left(\frac{\partial F}{\partial\sigma_x} + \frac{y\partial F}{\partial\sigma_z}\right) & \left(\frac{\partial F}{\partial\sigma_y} + \frac{\partial F}{\partial\sigma_z}\right) & \frac{\partial F}{\partial\tau_{xy}} & -\left[A+E\left(\frac{\partial F}{\partial\sigma_z}\right)^2\right] \end{bmatrix} \begin{bmatrix} \delta\sigma_x \\ \delta\sigma_y \\ \delta\tau_{xy} \\ \lambda \end{bmatrix}$$

or:

$$\begin{bmatrix} \delta\epsilon_x \\ \delta\epsilon_y \\ \delta\gamma_{xy} \\ \hline 0 \end{bmatrix} = \left[\begin{array}{ccc|c} & [D]^{-1} & & \text{symmetric} \\ \hline \left(\frac{\partial F}{\partial\sigma_x} + \frac{y\partial F}{\partial\sigma_z}\right) & \left(\frac{\partial F}{\partial\sigma_y} + y\frac{\partial F}{\partial\sigma_z}\right) & \frac{\partial F}{\partial\tau_{xy}} & -\left[A+E\left(\frac{\partial F}{\partial\sigma_z}\right)^2\right] \end{array}\right] \begin{bmatrix} \delta\sigma_x \\ \delta\sigma_y \\ \delta\tau_{xy} \\ \hline \lambda \end{bmatrix} \qquad (3.36)$$

where:

$[D]^{-1}$ is the inverse of the reduced plane strain elastic matrix

(ii) plane stress:

If Z is chosen as the direction normal to the plane, therefore:

$$\sigma_z = \tau_{yz} = \tau_{zx} = 0$$

At the same time the mathematical solution in plane strain, equation (3.36), may be written for plane stress as:

$$\begin{bmatrix} \delta\epsilon_x \\ \delta\epsilon_y \\ \delta\gamma_{xy} \\ \hline 0 \end{bmatrix} = \left[\begin{array}{ccc|c} & [D]^{-1} & & \text{symmetric} \\ \hline \dfrac{\partial F}{\partial \sigma_x} & \dfrac{\partial F}{\partial \sigma_y} & \dfrac{\partial F}{\partial \tau_{xy}} & -A \end{array}\right] \begin{bmatrix} \delta\sigma_x \\ \delta\sigma_y \\ \delta\tau_{xy} \\ \hline \lambda \end{bmatrix} \qquad (3.37)$$

where:

$[D]^{-1}$ is the inverse of the reduced plane stress elastic matrix.

(iii) Axial symmetry

Here the solution is once again more simple as four stress and strain components have non zero values and only two shear stress and strain components vanish. The form of relationship can be written as:

$$\begin{bmatrix} \delta\epsilon_x \\ \delta\epsilon_y \\ \delta\gamma_{xy} \\ \delta\epsilon_z \\ \hline 0 \end{bmatrix} = \left[\begin{array}{c|c} & \dfrac{\partial F}{\partial \sigma_x} \\ & \dfrac{\partial F}{\partial \sigma_y} \\ [D_0]^{-1} & \dfrac{\partial F}{\partial \tau_{xy}} \\ & \dfrac{\partial F}{\partial \sigma_z} \\ \hline \text{symmetric} & -A \end{array}\right] \begin{bmatrix} \delta\sigma_x \\ \delta\sigma_y \\ \delta\tau_{xy} \\ \delta\sigma_z \\ \hline \lambda \end{bmatrix} \qquad (3.38)$$

where:

$[D_0]^{-1}$ is the inverse of the reduced axial symmetry elastic matrix. (See Appendix A)

Clearly for ideal plasticity with no hardening 'A' is simply zero. If hardening is considered, attention must be given to the nature of the parameter (or parameters) **k** on which the shifts of the yield surface depend.

With a 'work hardening' material **k** is taken to be represented by the amount of plastic work done during plastic deformation, thus:

$$dk = \sigma_1 d\epsilon_1{}^p + \sigma_2 d\epsilon_2{}^p \,.. = \{\sigma\}^T d\epsilon_p \qquad (3.39)$$

Substituting the flow rule equation (3.17):

$$dk = \lambda \{\sigma\}^T \frac{\partial F}{\partial \sigma} \qquad (3.40)$$

thus, equation (3.21) can be written as:

$$A = -\frac{\partial F}{\partial k} \{\sigma\}^T \frac{\partial F}{\partial \sigma} \qquad (3.41)$$

The quantity $\bar{\sigma}_y = \bar{\sigma}_y\,(k)$ in equation (3.6), is the uniaxial stress at yield. If a plot of the uniaxial test giving $\bar{\sigma}_y$ verus the plastic uniaxial strain ϵ_{up} is available then

$$dk = \bar{\sigma}_y \, d\epsilon_{up}$$

and

$$-\frac{\partial F}{\partial k} = \frac{\partial \bar{\sigma}_y}{\partial k} = \frac{\partial \bar{\sigma}_y}{\partial \epsilon_{up}} \cdot \frac{1}{\bar{\sigma}_y} = \frac{H'}{\bar{\sigma}_y} \qquad (3.42)$$

in which H' is the slope of the plot at the particular value of $\bar{\sigma}_y$ of substituting into equation (3.41):

$$A = H' \qquad (3.43)$$

According to the above analysis, the quantities $\{\frac{\partial F}{\partial \sigma}\}$ and 'A' in equation (3.25) take up the following substitutions:

(i) plane strain:

From equation (3.36), it can be shown that:

$$\left\{\frac{\partial F}{\partial \sigma}\right\} = \begin{bmatrix} \dfrac{\partial F}{\partial \sigma_x} + y\dfrac{\partial F}{\partial \sigma_z} \\ \dfrac{\partial F}{\partial \sigma_y} + y\dfrac{\partial F}{\partial \sigma_z} \\ \dfrac{F}{\partial \tau_{xy}} \end{bmatrix} \tag{3.44}$$

$$A = H' + E\left(\frac{\partial F}{\partial \sigma_z}\right)^2 \tag{3.45}$$

(ii) plane stress

From equation (3.37), it is valid to write:

$$\left\{\frac{\partial F}{\partial \sigma}\right\} = \begin{bmatrix} \dfrac{\partial F}{\partial \sigma_x} \\ \dfrac{\partial F}{\partial \sigma_y} \\ \dfrac{\partial F}{\partial \tau_{xy}} \end{bmatrix} \tag{3.46}$$

$$A = H' \tag{3.47}$$

On differentation equation (3.6) will be found that:

$$\left.\begin{array}{l} \dfrac{\partial F}{\partial \sigma_x} = \dfrac{3\sigma_x'}{2\bar{\sigma}_y}, \quad \dfrac{\partial F}{\partial \sigma_y} = \dfrac{3\sigma_y'}{2\bar{\sigma}_y}, \quad \dfrac{\partial F}{\partial \sigma_z} = \dfrac{3\sigma_z'}{2\bar{\sigma}_y} \\ \dfrac{\partial F}{\partial \tau_{xy}} = \dfrac{3\tau_{xy}}{\bar{\sigma}_y}, \quad \dfrac{\partial F}{\partial \tau_{yz}} = \dfrac{3\tau_{yz}}{\bar{\sigma}_y}, \quad \dfrac{\partial F}{\partial \tau_{zx}} = \dfrac{3\tau_{zx}}{\bar{\sigma}_y} \end{array}\right] \tag{3.48}$$

$[D]^*_{ep}$ is thus reduced to a three by three matrix and is used in equation (3.24) to calculate the increment of the three stress component in the x-y plane.

The calculation of the plastic increment for the stress in z direction, follows by this:

From equation (3.28), it can be shown:

$$\delta\sigma_z = E(-\gamma/E\ \delta\sigma_x + 1/E\ \delta\sigma_y + \partial F/\partial\sigma_y\,\lambda - \delta\epsilon_y)/\gamma \qquad (3.49)$$

Substitute the value of equation (3.49) with the value of equation (3.27):

$$\lambda = \left[\left(-\frac{1}{E}\delta\sigma_x - \frac{\gamma}{E}\delta\sigma_y + \frac{\gamma}{E}\delta\sigma_x - \frac{1}{E}\delta\sigma_y + \delta\epsilon_y - \delta\epsilon_x\right) \Big/ \left(\frac{\partial F}{\partial\sigma_y} - \frac{\partial F}{\partial\sigma_x}\right)\right] \qquad (3.50)$$

Substitute (3.50) into (3.49):

$$\delta\sigma_z = \left\{\left(-\delta\sigma_x + \frac{1}{\gamma}\delta\sigma_y - \frac{E}{\gamma}\delta\epsilon_y\right) + \frac{1}{E}\frac{\partial F}{\partial\sigma_y}\left[\left(\delta\sigma_x(1+\gamma) - \delta\sigma_y(1+\gamma) + \delta\epsilon_y - \delta\epsilon_x\right) \Big/ \left(\frac{\partial F}{\partial\sigma_y} - \frac{\partial F}{\partial\sigma_x}\right)\right]\right\} \qquad (3.51)$$

It is necessary to keep a record of σ_z in the computation as plastic strains will occur in the normal direction.

CHAPTER 4

THE FINITE ELEMENT SOLUTION OF ELASTIC PLASTIC PROBLEMS

It has been shown that incremental solutions are required for elastic-plastic problems due to the non-linear nature of the constitutive equations. These solutions must simultaneously satisfy throughout the body, the equilibrium of internal and external force and the specified criterion and hardening law. The common method is to repeatedly estimate a stress distribution which satisfies the yield criterion and the hardening law but will not necessarily be in equilibrium with the applied loads, i.e. from the principle of virtual work

$$\{p\} - \int_{(v)} \{\sigma\}[B]^T \{\delta\} \, d(vol) = \{R\} \qquad (4.1)$$

where {R} is a non-zero vector of residual forces if equilibrium is not satisfied. Newton-Raphson iterative techniques were widely used to reduce {R} to zero. A different approach is suggested by Zienkiewicz et al[11] and this 'initial-stress' approach is adopted for the current work.

4.1 THE INITIAL-STRESS METHOD

A full Newton-Raphson solution requires the formation and inversion (or decomposition) of the elastic-plastic stiffness matrix (tangent stiffness matrix) for the structure at each iteration of a particular load increment. Obviously this can be expensive in terms of computer time.

A modified Newton-Raphson solution, whilst generally requiring more iterations, requires a new tangent stiffness matrix to be formed only occasionally, perhaps at the beginning of an increment or after a few iterations. However, savings are not dramatic and the sizing of load increments becomes critical if accuracy is to be maintained.

This is to say, if non-linearity becomes more severe or yielding is spreading rapidly, load increment size should be adapted in order that the tangent stiffness matrix is updated with sufficient frequency as to maintain an acceptable level of accuracy in results.

These problems are overcome to some extent in the 'initial stress' method of solution. By using the fact that even in ideal plasticity increments of strain prescribe uniquely the stress system, an adjustable process is derived in which 'initial stresses' are distributed elastically through the structure. This approach permits the advantage of initial processes (in which the basic elasticity matrix remains unchanged) to be retained and thus only one matrix inversion is required for each load increment. If, however, a single load increment is used it will be found that an approximate lower bound is achieved, the final solution satisfying equilibrium and yield criterion but not necessarily following the current strain development as the appropriate flow rules may be violated. Also the process appears to be the most rapidly convergent.

4.1.1 The initial-stress computational process

The initial-stress process approaches the solution of an elastic-plastic problem as a series of approximations. In the first place during a load increment a purely elastic problem is solved determining an increment of strain $\{\Delta\epsilon'\}$ and of stress $\{\Delta\sigma'\}$ at every point of the continuum (or structure).

The non-linearity implies, however, that for the increment of strain found, the stress increment will, in general, not be correct. If the true increment of stress possible for the given strain is $\{\Delta\sigma\}$ then the solution can only be maintained by a set of body forces equilibriating the 'initial' stress system $\{\Delta\sigma'\} - \{\Delta\sigma\}$.

At the second stage of computation this body force system can be removed by allowing the structure (with unchanged elastic properties) to deform further. An additional set of strain and corresponding stress increments are caused. Once again these are likely to exceed those permissible by the non-linear relationship and the redistribution of equilibrating body forces has to be repeated. The cycling is terminated when these forces reach sufficiently small values. Convergence having been achieved, full non-linear compatibility and equilibrium conditions will be satisfied.

It is convenient to start the increment process only when the first yield has occured and hence in the programme this allows the subsequent load increments to be related to the load at which first yield is noted.

4.1.2 Convergence criteria

The vector of nodal forces corresponding to the equilibrating body forces $\{p'\}$ can never be exactly zero and therefore some convergence conditions must be imposed. This is achieved by testing the values of the applied load vector $\{p\}$ against the values of $\{p'\}$ with respect to some preset tolerance value, say 1%. This is done in the following manner such that convergence is achieved when:

$$\left| \frac{\{P'\}^T \{P'\}}{\{P\}^T \{P\}} \right|^{½} < 1\% \qquad (4.2)$$

When specifying a tolerance value an attempt should be made to balance the computer time available against accuracy. Usually parameters of the elastic field away from the region of yielding are unaffected by incomplete convergence.

4.1.3 Sample calculation

In order to illustrate the steps of the computational process the following sample is considered in Figure (8), where a square plate fixed along one edge is subjected to equal tensile forces applied at

$P = 10^6 N$

$a = 1000$ mm

Thickness $= t$

$= 120$ mm

$E = 0.23E+06 N/mm^2$

$\nu = 0.3$

$\sigma_y = 16.92 N/mm^2$

$H' = 0.1$

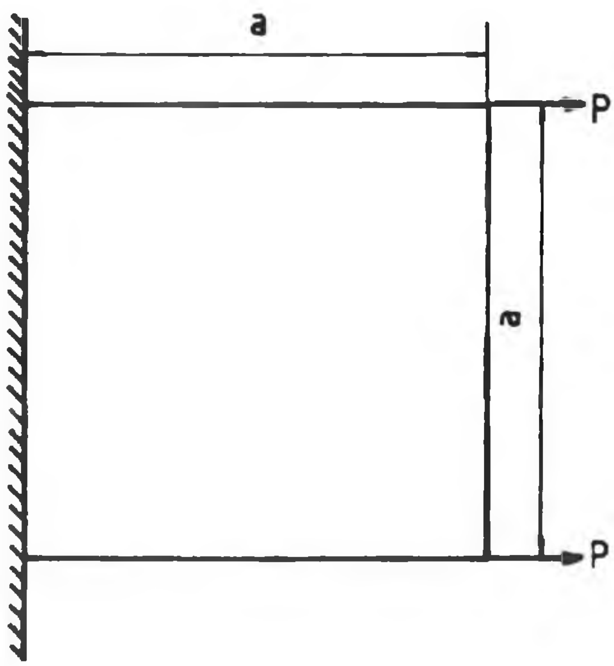


FIG.(8)

the corners of the opposite edge, plane stress is assumed. The plate can be discretised simply as shown in Figure (9). Hence the applied load vector can be started as:

$$\{P\} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ P \\ 0 \\ P \\ 0 \end{bmatrix}$$

Feed input data, material properties, coordinates, number of cycles, load increment, tolerance number of freedom.

For P = 0.1E+07 element 2 is on the point of yielding thus the stiffness matrix of element 1 and 2 are illustrated in Table 1, whereas the overall stiffness matrix is illustrated in Table 2.

The [B] matrix for each element is:

(i) Element (1)

$$[B] = \begin{vmatrix} -1.0E-03 & 0.0 & 1.0E-03 & 0.0 & 0.0 & 0.0 \\ 0.0 & -1.0E-03 & 0.0 & 0.0 & 0.0 & +1.0E-03 \\ -1.0E-03 & -1.0E-03 & 0.0 & 1.0E-03 & 1.0E-03 & 0.0 \end{vmatrix}$$

(ii) Element (2)

$$[B] = \begin{vmatrix} -1.0E-03 & 0.0 & 0.0 & 0.0 & 1.0E-03 & 0.0 \\ 0.0 & 0.0 & 0.0 & -1.0E-03 & 0.0 & 1.0E-03 \\ 0.0 & -1.0E-03 & -1.0E-03 & 0.0 & 1.0E-03 & 1.0E-03 \end{vmatrix}$$

TABLE

$[K]^{e1}$ =

1.780E+07	8571430.	-1.318E+07
8571430.	1.780E+07	-395045.
-1.318E+07	-3956045.	1.318E+07
-4615385.	-4615385.	0.000E+00
-4615385.	-4615385.	0.000E+00
-3956045.	-1.318E+07	3956045.

$[K]^{e2}$ =

1.318E+07	0.000E+00	0.000E+00
0.000E+00	4615385.	4615385.
0.000E+00	4615385.	4615385.
3956045.	0.000E+00	0.000E+00
-1.318E+07	-4615385.	-4615385.
-3956045.	-4615385.	-4615385.

(1)

-4615385.	-4615385.	-3956045.
-4615385.	-4615385.	-1.318E+07
0.000E+00	0.000E+00	3056045.
4615385.	4615385.	0.000E+00
4615385.	4615385.	0.000E+00
0.000E+00	0.000E+00	1.318E+07

3956045	-1.318E+07	-3956045.
0.000E+00	-4615385.	-4615385.
0.000E+00	-4615385.	-4615385.
1.318E+07	-3956045.	-1.31E+07
-3956045.	1.789E+07	8571430.
-1.318E+07	8571430.	1.780E+07

TABLE (2)

[K] =

1.780E+07	8571430	-4615385	-3956045	-1.318E+07	-4615358	0.0	0.0
	1.7803+07	-4615385	-1.318E+07	-3956045	-4615385	0.0	0.0
		1.7803+07	0.0	0.0	8571430	-1.318E+07	-3956045
Symmetric			1.780E+07	8571430	0.0	-4615385	-4615385
				1.780E+07	0.0	-4615385	-4615385
					1.780E+07	-3956045	-1.318E+07
						1.780E+07	8571430
							1.780E+07

The elasticity matrix:

$$[D] = \begin{vmatrix} 219780.2 & 65934.07 & 0.0 \\ 65934.07 & 219780.2 & 0.0 \\ 0.0 & 0.0 & 76923.07 \end{vmatrix}$$

The current state of stress and strain is as follows:

(i) Element (1)

$$\{\epsilon_0\} = \begin{bmatrix} 0.7371E-04 \\ -0.6391E-27 \\ 0.60559E-05 \end{bmatrix}$$

$$\{\sigma_0\} = \begin{bmatrix} 16.20 \\ 4.860 \\ 0.466 \end{bmatrix}$$

(ii) Element (2)

$$\{\epsilon_0\} = \begin{bmatrix} 0.8494E-04 \\ -0.2337E-04 \\ -0.6059E-04 \end{bmatrix} \quad ; \quad \{\sigma_0\} = \begin{bmatrix} 17.13 \\ 0.4667 \\ -0.4659 \end{bmatrix}$$

Apply a load increment of 0.1P, P being the load at the first yield. The solution from this point can be listed as a series of steps as follows:

1. Determine elastic increment of stress and strain.

Element (1)

$$\{\Delta\epsilon'\}_1 = \begin{bmatrix} 7.371E-06 \\ -6.891E-29 \\ 6.058E-07 \end{bmatrix} \quad ; \quad \{\Delta\sigma'\}_1 = \begin{bmatrix} 1.626 \\ 0.486 \\ 4.660E-02 \end{bmatrix}$$

Element (2)

$$\{\Delta\varepsilon'\}_1 = \begin{bmatrix} 8.497E-06 \\ -2.336E-06 \\ -6.058E-06 \end{bmatrix} \quad ; \quad \{\Delta\sigma'\}_1 = \begin{bmatrix} 1.713 \\ 4.664E-02 \\ -4.660E-02 \end{bmatrix}$$

2. Add $\{\Delta\sigma'\}_1$ to the stresses existing at the start of increment $\{\sigma_0\}$ to give $\{\sigma'\}$

Element (1) $\{\sigma'\} = \begin{bmatrix} 17.820 \\ 5.346 \\ 0.5120 \end{bmatrix}$

Element (2) $\{\sigma'\} = \begin{bmatrix} 18.843 \\ 0.513 \\ -0.512 \end{bmatrix}$

3. Check whether $F\{\sigma'\} < 0$. If this is satisfied only elastic strain changes occur and process is stopped. If not, proceed to (4)

Element (1) $F\{\sigma'\} = -1.0556$

Element (2) $F\{\sigma'\} = 1.6928$

The process proceed with Element (2) only.

4. If $F\{\sigma'\} \geq 0$ and also $F\{\sigma_0\} = 0$ (i.e. element was in yield at start of increment) find $\{\sigma\}_1$ by the following equation:

$$\{\Delta\sigma\}_1 = [D]_{ep}^{*} \{\Delta\varepsilon'\}_1$$

Where $[D]^{*}_{ep}$ is the elastic-plastic matrix computed with stresses $\{\sigma'\}$

$$\{\Delta\sigma\}_1 = \begin{vmatrix} 64227.19 & 98505.78 & 5260.431 \\ 98505.78 & 212959.9 & -1101.497 \\ 5620.432 & -1011.497 & 76745.17 \end{vmatrix} \{\Delta\epsilon'\}_1$$

$$= \begin{bmatrix} 0.3123 \\ 0.3400 \\ 0.0007 \end{bmatrix}$$

Evaluate stress which has to be supported by body forces.

$$\{\Delta\sigma''\}_1 = \{\Delta\sigma'\}_1 - \{\Delta\sigma\}_1$$

$$= \begin{bmatrix} 1.401 \\ -0.293 \\ -0.047 \end{bmatrix}$$

Store current stress and strain

$$\{\sigma\} = \{\sigma'\} - \{\Delta\sigma''\}_1$$

$$= \begin{bmatrix} 17.441 \\ 0.806 \\ -0.465 \end{bmatrix}$$

$$\{\epsilon\} = \{\epsilon_0\} + \{\Delta\epsilon'\}_1$$

$$= \begin{bmatrix} 9.3461E-05 \\ -2.570E-05 \\ -6.664E-06 \end{bmatrix}$$

5. If $F\ \{\sigma'\} > 0$ but $F\ \{\sigma_0\} < 0$ find the intermediate stress at which yield begins and compute increment $\{\Delta\sigma\}_1$ as previously from that point then proceed as in (4).

6. Comput nodal forces corresponding to the equilibrating body forces. They are given for any element by

$$\{P'\}^e = \int_{(v)} [B]^T \{\Delta\sigma''\}_1 \, d(vol)$$

(See Figure (10))

$$\{P'\}_2 = \begin{bmatrix} -24069.44 \\ 2843.028 \\ 17603.55 \\ 2843.028 \\ 81226.41 \\ -20446.58 \end{bmatrix}$$

7. Resolve using original elastic properties and the load system $\{P'\}$. Find $\{\Delta\sigma'\}_2$ and $\{\Delta\epsilon'\}_2$

$$\{P'\} = \begin{bmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 17603.55 \\ 2843.028 \\ 81226.41 \\ -20446.58 \end{bmatrix}$$

$$\{\Delta\epsilon'\}_2 = \begin{bmatrix} 7.027E-06 \\ -3.987E-06 \\ 9.393E-07 \end{bmatrix} ; \quad \{\Delta\sigma'\}_2 = \begin{bmatrix} 1.281 \\ -0.413 \\ 0.072 \end{bmatrix}$$

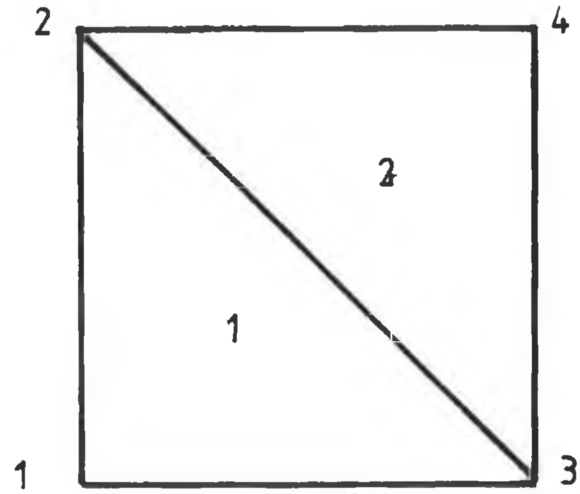

2
4
2
1
1
3

FIG.(9)

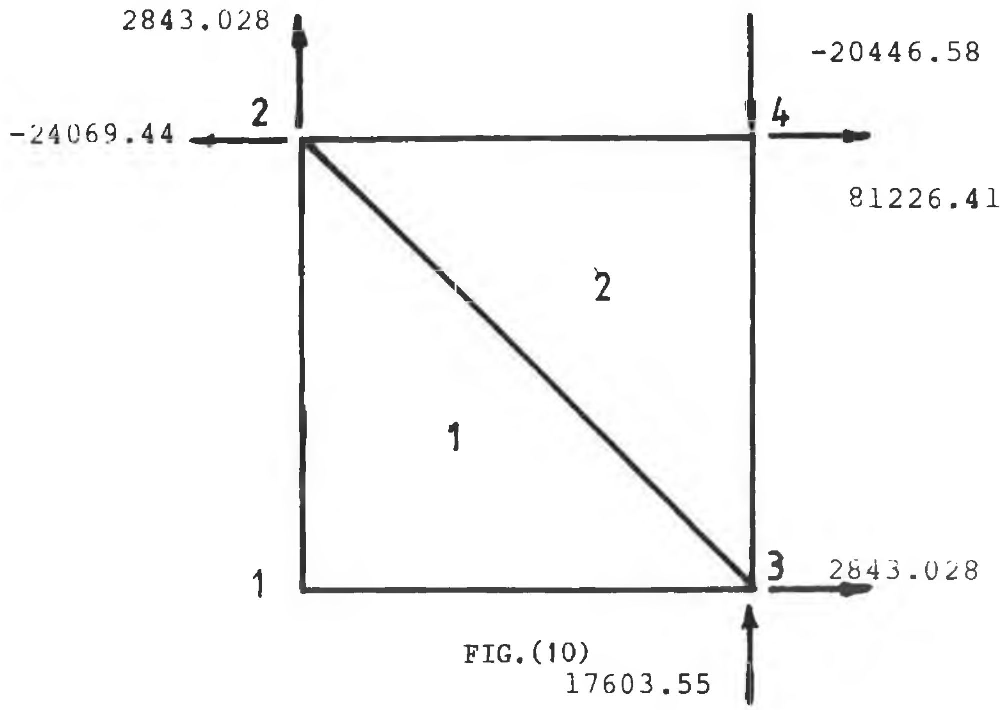

2843.028
-20446.58
2
4
-24069.44
81226.41
2
1
1
3
2843.028
17603.55

FIG.(10)

8. Repeat steps 2 to 7, etc.

The cycling is terminated when the nodal forces of (6) reach sufficiently small values. For a specified tolerance, 1% convergence was achieved after 5 cycles. The method is illustrated graphically in Figure (11). The final output data is listed in Table 3.

The output data in Table 3 is showing that the displacement in **X** direction of point 3 is less than the displacement in **X** direction of point 4. This difference is due to the fact that the problem is simple and consists of only two elements. It is obvious from table 3 that the output data is true because element (2) is fixed in one point in **X** and **Y** directions but element (1) is fixed in two points in **X** and **Y** direction. That is why there is a difference between the displacements which reflect the simplicity of the problem.

4.1.4 Load increment sizing

As mentioned previously, the load increment size is less critical for the initial stress method than other solution techniques. However, smaller increments will ensure greater accuracy, and at the same time faster convergence, as the flow rule is less likely to be violated. It is advisable to use increments less in magnitude that the load required to cause the first yielding and then start the incremental process from this load. Generally speaking, increments in the approximate range 20-50% of the first yield load will provide acceptable results.

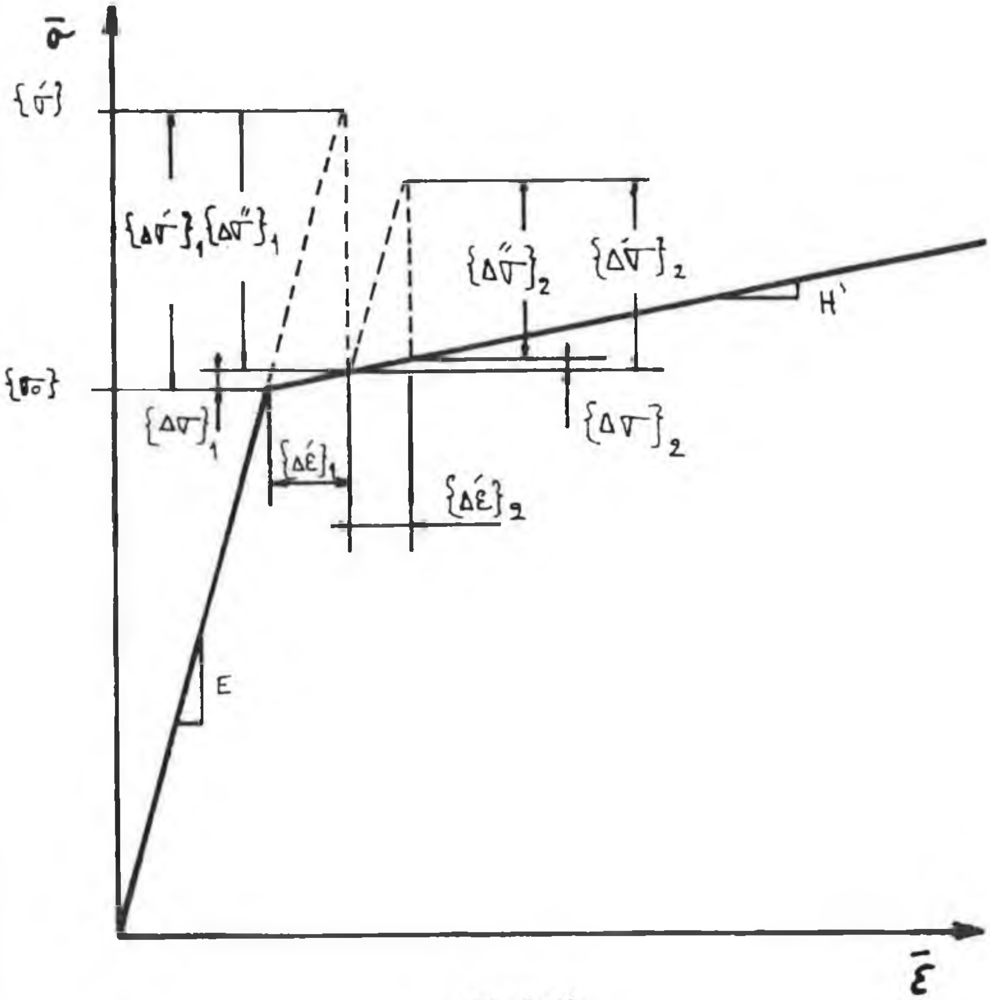


FIG.(11)

TABLE (3)

LOAD INCREMENT = 1.00
CURRENT INCREMENT = 1.00
NO.OF ITERATION WITHIN INCREMENT = 1

NODE	DISPLACEMENT	
1	0.000000	0.000000
2	0.000000	0.000000
3	0.073713	0.006059
4	0.084964	-0.017310

ELEMENT	X-STRESS	Y-STRESS	XY-STRESS	EQ-STRESS	PLASTIC
1	0.1620E+02	0.4860E+01	0.4660E+00	0.1442E+02	0
2	0.1713E+02	0.4667E+00	-0.4659E+00	0.1692E+02	1

ELEMENT	X-STRAIN	Y-STRAIN	XY-STRAIN	EQ-STRAIN
1	0.7371E-04	-0.6391E-27	0.6059E-05	0.7211E-04
2	0.8496E-04	-0.2337E-04	-0.6059E-05	0.8462E-04

NODE	REACTIONS	
1	-0.1000E+07	-0.3196E+06
2	-0.1000E+07	0.3196E+06
3	0.0000E+00	0.0000E+00
4	0.0000E+00	0.0000E+00

LOAD INCREMENT = 1.10
CURRENT INCREMENT = 0.1
NO.OF ITERATION WITHIN INCREMENT = 5

NODE	DISPLACEMENT	
1	0.000000	0.000000
2	0.000000	0.000000
3	0.082796	0.001776
4	0.116710	-0.037159

ELEMENT	X-STRESS	Y-STRESS	XY-STRESS	EQ-STRESS	PLASTIC
1	0.1820E+02	0.5459E+01	0.1366E+00	0.1618E+02	0
2	0.1765E+02	0.3240E+00	-0.1292E+00	0.1749E+02	1

ELEMENT	X-STRAIN	Y-STRAIN	XY-STRAIN	EQ-STRAIN
1	0.8280E-04	-0.6783E-27	0.1776E-05	0.8298E-04
2	0.1167E-03	-0.3894E-04	-0.3245E-05	0.1185E-03

NODE	REACTIONS	
1	-0.1100E+07	-0.3515E+06
2	-0.1100E+07	0.3515E+06
3	0.0000E+00	0.0000E+00
4	0.0000E+00	0.0000E+00

CHAPTER 5

DESIGN AND DEVELOPMENT OF THE E.P. PROGRAM AND COMPARISONS

5.1 OBJECTIVES

There are a number of general purpose programs available for the application of finite element stress analysis in mechanical engineering design. To use a versatile general purpose program for solving specialized problems is often troublesome and far more costly (in computer time) than to write a program expressly for solving the specialised problem[25].

In this study, the objective of the E.P. Program is primarily to solve a specific ring structure design problem under different cases of loading and boundary conditions, and secondly for solving other two-dimensional engineering problems within the scope of the program. The E.P. Program is able to solve these problems in two phases (1. prescribe displacement; 2. prescribe force), for plane elasticity (plane stress/plane strain).

5.2 SOFTWARE DESIGN

To make use of the computer's facilities and to maximise the size of the problem that can be defined, the E.P. Program was divided into a number of modules. These modules appear as subroutines and the main line program is a simple routine whose function is to automate selection and calling of these subroutines. Figure (12) shows the flow chart of these modules as it is in the E.P. Program.

It was necessary to decide how the core storage should be allocated. The general tendency in finite element programs is to fix some maximum dimension on array sizes[26]. The E.P. Program belongs to this catagory. Subsequently, accessed files were used in E.P. Program for back storing data. This was thought to be more favourable than random accessed files. Back storing was used for input data, mesh generation, [B] matrix and graphical output. The

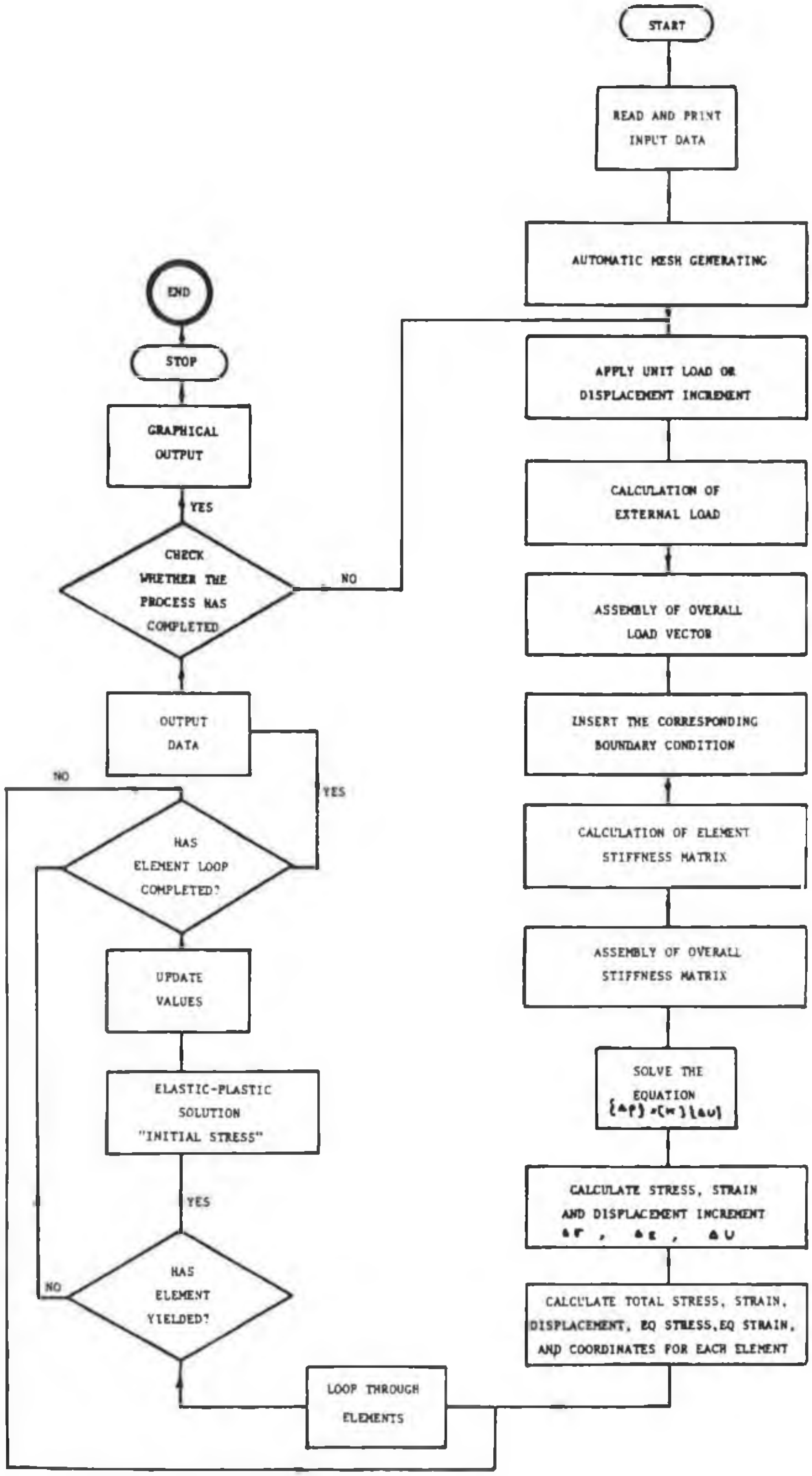

START
READ AND PRINT INPUT DATA
AUTOMATIC MESH GENERATING
APPLY UNIT LOAD OR DISPLACEMENT INCREMENT
CALCULATION OF EXTERNAL LOAD
ASSEMBLY OF OVERALL LOAD VECTOR
INSERT THE CORRESPONDING BOUNDARY CONDITION
CALCULATION OF ELEMENT STIFFNESS MATRIX
ASSEMBLY OF OVERALL STIFFNESS MATRIX
SOLVE THE EQUATION
CALCULATE STRESS, STRAIN AND DISPLACEMENT INCREMENT
CALCULATE TOTAL STRESS, STRAIN, DISPLACEMENT, EQ STRESS, EQ STRAIN, AND COORDINATES FOR EACH ELEMENT
LOOP THROUGH ELEMENTS
HAS ELEMENT YIELDED?
NO
YES
ELASTIC-PLASTIC SOLUTION "INITIAL STRESS"
UPDATE VALUES
HAS ELEMENT LOOP COMPLETED?
NO
YES
OUTPUT DATA
CHECK WHETHER THE PROCESS HAS COMPLETED
NO
YES
GRAPHICAL OUTPUT
STOP
END

FIG.(12)

E.P. Program with its variable names and input data instructions is listed in Appendix (F).

5.3 MAIN PROGRAM

The main program is the principal component whose functions include: the input and storage of data describing the model; the modification of data; loading and saving data and calling the specialist subroutines when needed.

5.4 INPUT DATA

To specify the problem it is necessary to provide the computer with the data that describes the model. This consists of information specifying the title of the problem, selection of options, material properties, conditions for starting and stopping the process including tolerance, load increment and load size. Also the input data includes the boundary condition, the external loads and information about the mesh generation.

The E.P. Program reads the input data from the screen with free format, stores it in separate files and calls it back when needed. These separate files are stored temporarily on a separate disk called "SCRATCH". The idea behind this way is to save time when the program is fed by data. The E.P. Program has two options to create the mesh generation, either the mesh generation is created manually or created automatically by option. The two options are displayed in Appendix (F).

5.5 AUTOMATIC MESH GENERATION

In the last few years, considerable effort has been devoted in developing mesh generation routines in order to eliminate the drudgery of working out the data and to minimize data errors. An early survey of a number of routines was given by Buell and Bush[27].

The input data for the subroutine of generating a triangular mesh, which is presented here, consists of information concerning the total number of generating lines, a weighting factor, and for each generating line the number of intervals required, together with the coordinates of the two end points. The output consists firstly of the **x** and **y** coordinates of each point with each corresponding number, and secondly the element number with each element definition.

The number of divisions in the adjacent generating lines can be equal or can differ by one, so that the mesh can vary according to the specific requirement of the problem in Figure (13). There are two divisions in line **A**, but neighbouring line **B** is allowed to have one, two or three divisions.

Another requirement incorporated into the program is the weighting factor. Depending on whether the weighting is < 1, $= 1$ or > 1, the intervals along a generating line will become progressively shorter, stay equal, or become progressively longer.

From Figure (13(d)) it can be seen that the coordinates of the point **i** along a generating line ab can be computed by:

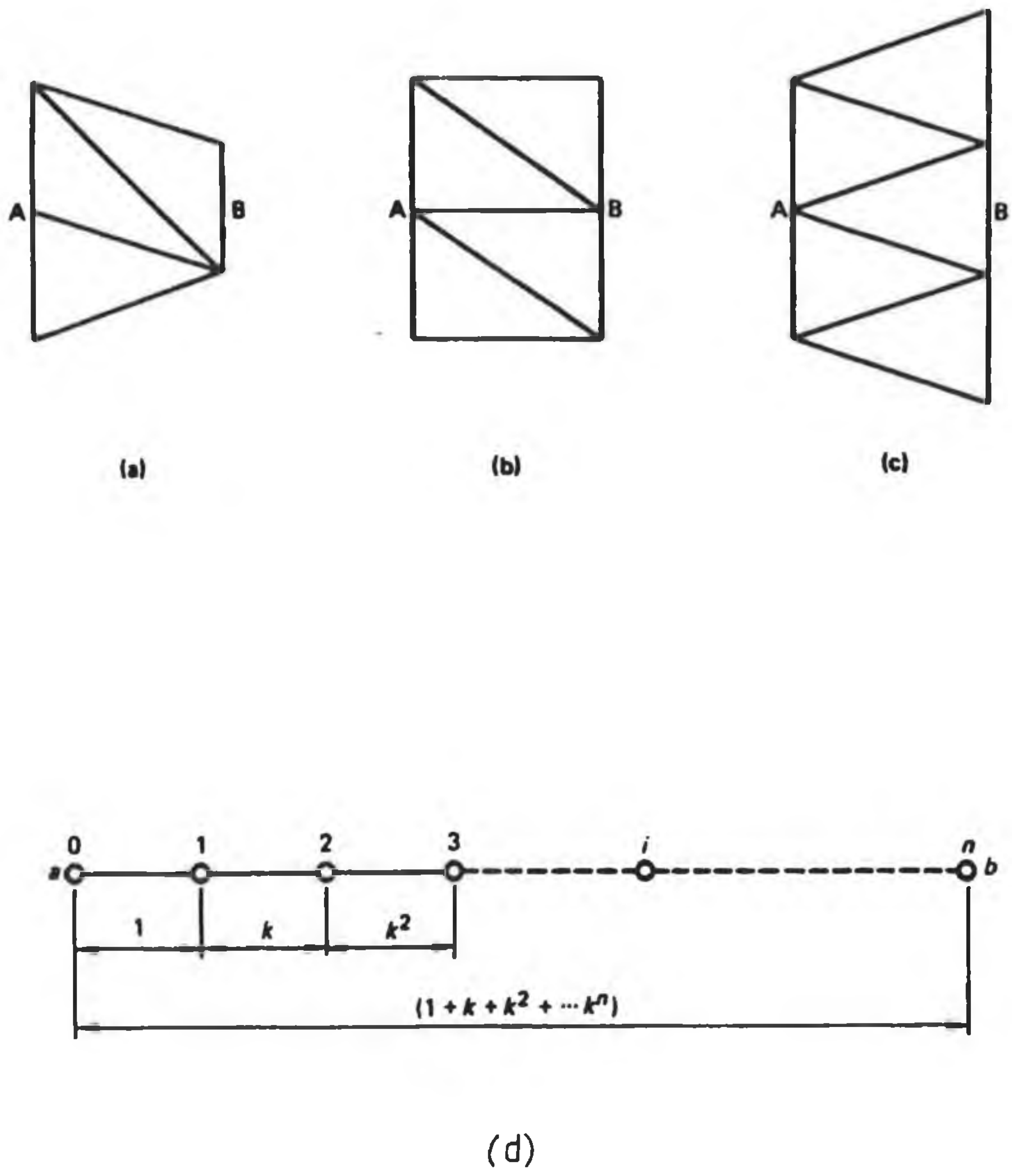


FIG.(13)

$$x_i = (x_b - x_a) \frac{\sum\limits_{j=1} K^{j-1}}{\sum\limits_{j=1}^{n} K^{j-1}} + x_a$$

and (5.1)

$$y_i = (y_b - y_a) \frac{\sum\limits_{j=1}^{i} K^{j-1}}{\sum\limits_{j=1}^{n} K^{j-1}} + y_a$$

The element definitions are worked out by taking the **mth** and the **(m+1)th** generating lines, and forming **n** quadrilaterals along the **jth** line. Referring to Figure (13(a)); in which the **(m+1)th** line has one less division than the **mth** lines, the last **(nth)** quadrilateral becomes a triangle, while for Figure (13(b)) **mth** and **(m+1)th** lines have the same number of intervals and each of the **(nth)** quadrilaterals is simply split into two triangles. For Figure (13(c), however, the **(m+1)th** line has now one more division than the **jth** lines, and one extra triangle must be added to **n** quadrilaterals which have been established. The flow chart of this subroutine is in Figure (14).

5.6 LOAD MATRIX

Two types of loading are applied for the E.P. Program

(1) concentrated loads applied at nodes

(2) uniformally distributed loads over an element.
Each one of these were put in a separate subroutine and called by option.

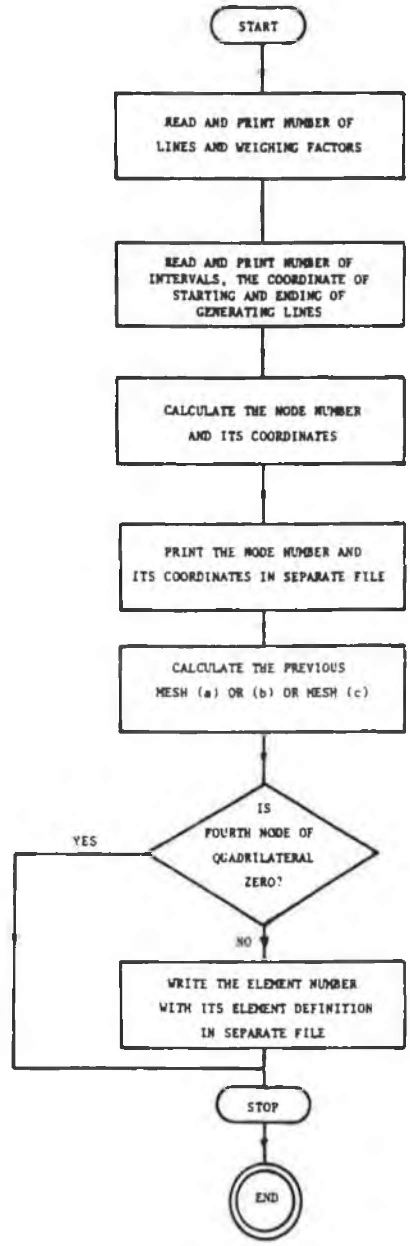

START
READ AND PRINT NUMBER OF LINES AND WEIGHING FACTORS
READ AND PRINT NUMBER OF INTERVALS, THE COORDINATE OF STARTING AND ENDING OF GENERATING LINES
CALCULATE THE NODE NUMBER AND ITS COORDINATES
PRINT THE NODE NUMBER AND ITS COORDINATES IN SEPARATE FILE
CALCULATE THE PREVIOUS MESH (a) OR (b) OR MESH (c)
IS FOURTH NODE OF QUADRILATERAL ZERO?
YES
NO
WRITE THE ELEMENT NUMBER WITH ITS ELEMENT DEFINITION IN SEPARATE FILE
STOP
END

FIG.(14)

5.6.1 Concentrated loads

The load that can be externally applied to a node in the E.P. Program, can be either a static load or a physical restraint. These factors are stored in one dimension array, in which the position of the load, or fastening, follows the equation, where **n** is the number of the node

$$\left.\begin{aligned} x_n &= 2n - 1 \\ y_n &= 2n \end{aligned}\right\} \quad (5.2)$$

To ensure the structure is adequately supported, there must be a minimum of three restraints, with at least one in **x** and **y** direction.

5.6.2 Uniformly distributed load

The magnitude of this load per unit length, and edge over which it is acting is defined, then, the E.P. Program calculates the equivalent nodal force. From Figure (15) the forces in the orient axes x_o-y_o and the global axes **x-y** are related to each other by:

$$\begin{vmatrix} P_x \\ P_y \end{vmatrix} = \begin{vmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{vmatrix} \begin{vmatrix} P_{x_o} \\ P_{y_o} \end{vmatrix} \quad (5.3)$$

or

$$\{P\} = [Q]\,\{P_o\}$$

where:

[Q] is the transformation matrix between the two systems and **θ** is positive as indicated in Figure (15). The direction cosines are easily computed from geometrical considerations.

$$\begin{aligned} \cos\theta &= (x_j - x_i)/l \\ \sin\theta &= (y_j - y_i)/l \end{aligned} \quad (5.4)$$

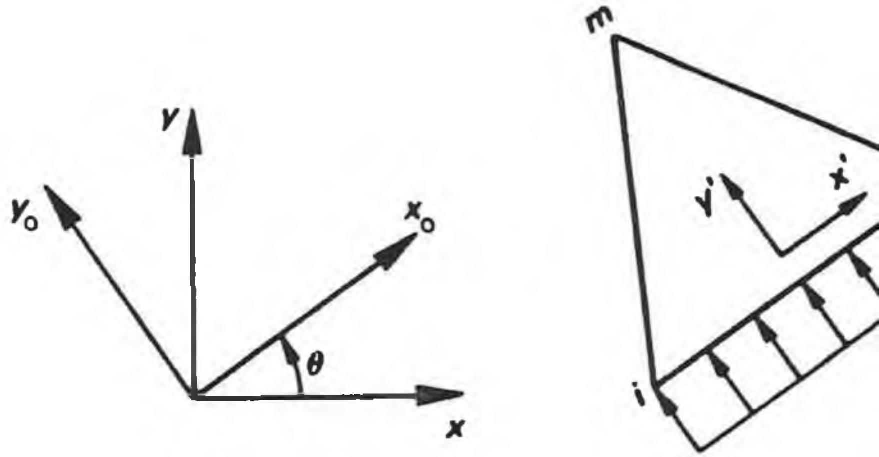
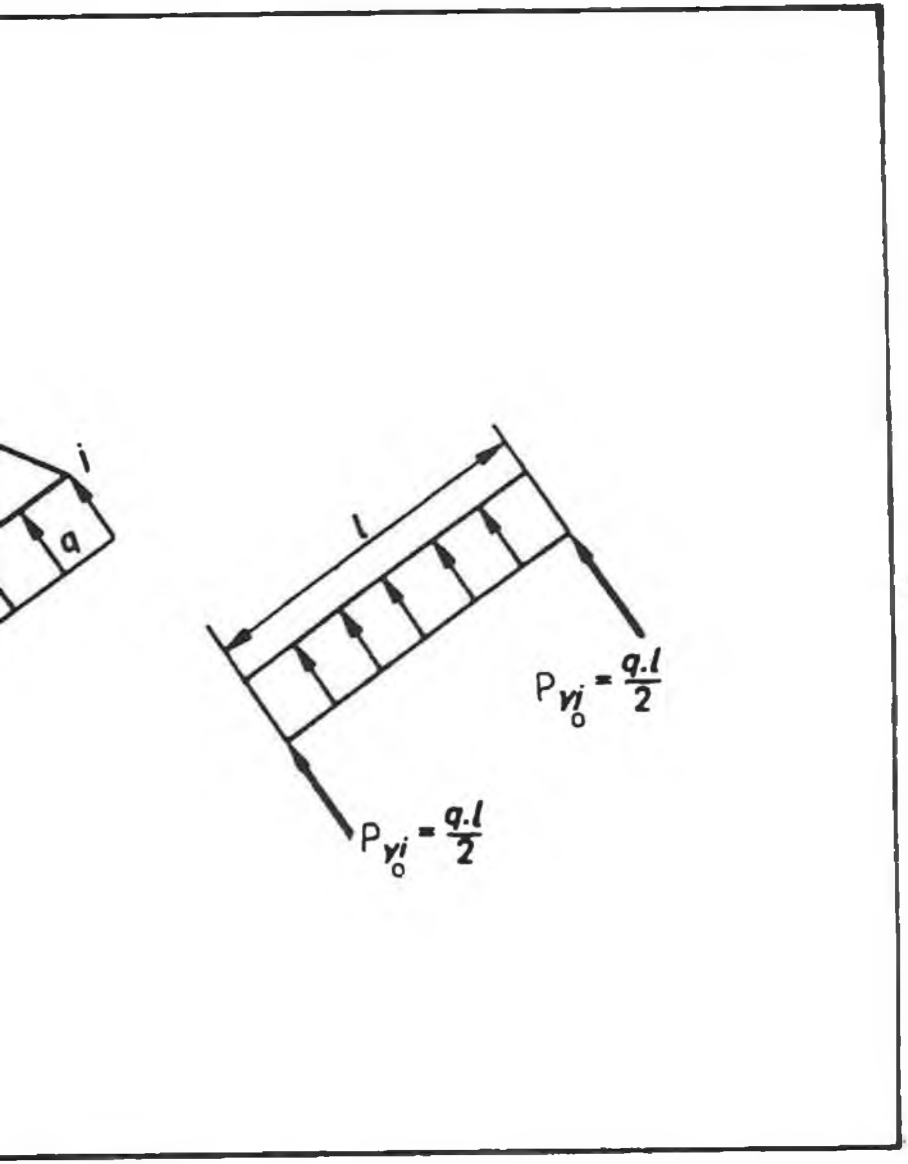


FIG.(15)


i
q
l
$P_{y_o^i} = \frac{q.l}{2}$
$P_{y_o^i} = \frac{q.l}{2}$

Thus the equivalent global components of forces at a node:

$$\begin{bmatrix} P_x \\ P_y \end{bmatrix} = \begin{vmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{vmatrix} \begin{bmatrix} 0 \\ ql/2 \end{bmatrix}$$

$$= ql/2 \begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix} \qquad (5.5)$$

5.7 THE SOLVE PROCEDURES

The function of the solve procedures is basically to derive the complete stiffness structure matrix (overall stiffness structure matrix), from individual element stiffness matrices; and solve the unknown nodal displacement, strain, stress increment and current stress and strain. The mathematical formulas which relate to these are outlined in Chapter (3).

Having calculated the element stiffness matrix, its individual terms are then inserted into the corresponding positions in the complete stiffness matrix. Appendix (E) shows a simple example for assembly of overall stiffness matrix.

The structure is in general restrained at a number of nodes such that various displacements are prevented. In this case the diagonal terms of the stiffness structure corresponding to the fixed displacement is multiplied by a large number. So the value of these prevented displacements are very close to zero.

The current E.P. Program solves the elastic-plastic problem in two phases:

(1) prescribed force
(2) prescribed displacement

Each one of them can be chosen by option described in Appendix (F).

5.7.1 Prescribed force

In prescribed force procedure, the external force would be given to the structure as physical increment loads. Gaussian[28] elimination was chosen to calculate the unknown displacement as it provides a means of solving a set of equations. The stiffness structure matrix is augmented with the force vector and passed to the solution procedure.

The method consists of eliminating the equations one at a time so that there is a corresponding reduction in the size of the modified matrix until finally the matrix is reduced to one equation containing only one variable. The set of eliminated equations forms a triangular matrix and is used for backsubstitution process. By starting at the last equation and working backwards to the first equation, one variable will be determined at each step by using the variables which are already known.

5.7.2 Prescribed displacement

The displacement here is given to the structure in the **x** and **y** directions in incremental way as well. It can be noted that for each node the governing equations can be expressed as:

$$\left|\begin{array}{c|c} [K_{11}] & [K_{12}] \\ \hline [K_{21}] & [K_{22}] \end{array}\right| \begin{bmatrix} \delta_1 \\ \hline \delta_2 \end{bmatrix} = \begin{bmatrix} P_1 \\ \hline P_2 \end{bmatrix} \qquad (5.6)$$

Thus once the overall stiffness matrix has been formed it is partitioned accordingly along with the force and displacement vectors. This leaves the following equations to be solved:

$$\{P_1\} = [K_{11}]\ \{\delta_1\} + [K_{12}]\ \{\delta_2\} \qquad (5.7)$$

$$\{P_2\} = [K_{21}]\ \{\delta_1\} + [K_{22}]\ \{\delta_2\} \qquad (5.8)$$

If $\{\delta_1\}$ is the vector of known displacement then $\{P_2\}$ is a zero vector corresponding to the unrestrained freedoms. This means the unknown displacement vector $\{\delta_2\}$ can be solved from equation (5.8) with only the sub-vector $[K_{22}]$ required to be inverted (see Appendix D for inversion matrix). The unknown vector $\{P_1\}$ can then be calculated by substitution into equation (5.7). In subsequent interations within the same load increment the solution equation becomes:

$$\{\delta_2'\} = [K_{22}]^{-1} \{P_2'\} \qquad (5.9)$$

where $\{P_2'\}$ is the appropriately partitioned vector consisting of equilibrating body forces. Thus the same inverted stiffness matrix is used throughout the increment. The displacements calculated for each interation $\{\delta_2'\}$ are added to the initial displacement vector $\{\delta\}$ at each stage.

Equilibrium of forces is maintained throughout by the following modification at each iteration to the initial vector of unknown forces:

$$\{P_1\} = \{P_1\} - \{P_1'\} + [K_{12}] \{\delta_2'\} \qquad (5.10)$$

$\{P_1'\}$ consisting of the equilibrating body forces acting for static freedoms. The complete computational iteration procedure is illustrated in the flow chart of Figure (16).

5.8 E.P. PROGRAM OUTPUT

The output data consists of all current input data, nodal displacements, current coordinates, current stresses, strains and reaction forces and also graphical output which consists of the geometry of the structure before and after loading and the elastic and plastic regions of the structure. These are obtained by linking the E.P. Program with a graphical package called CALCOMP. Figure (17) shows a photograph of a terminal with the CALCOMP plotter

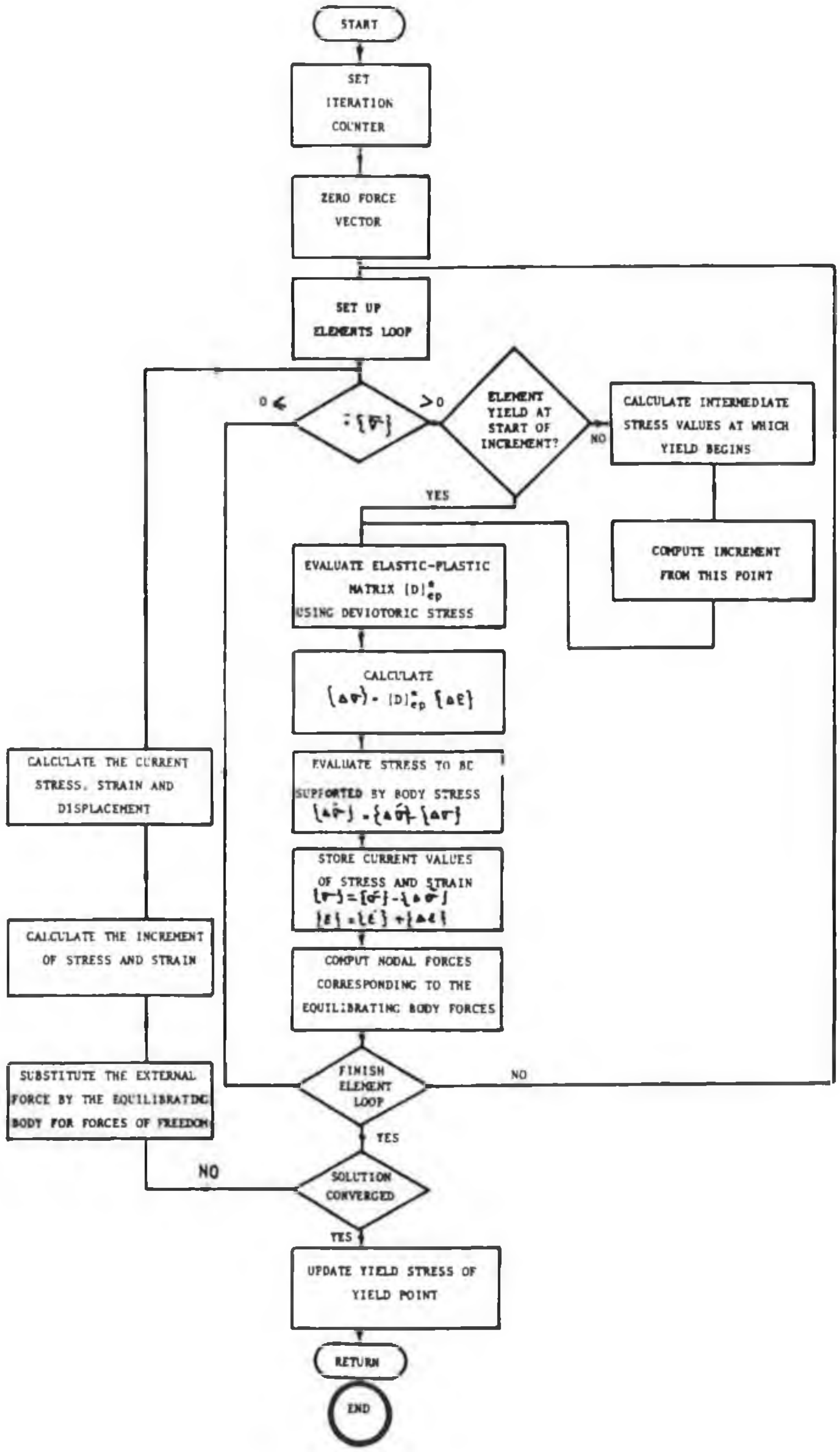
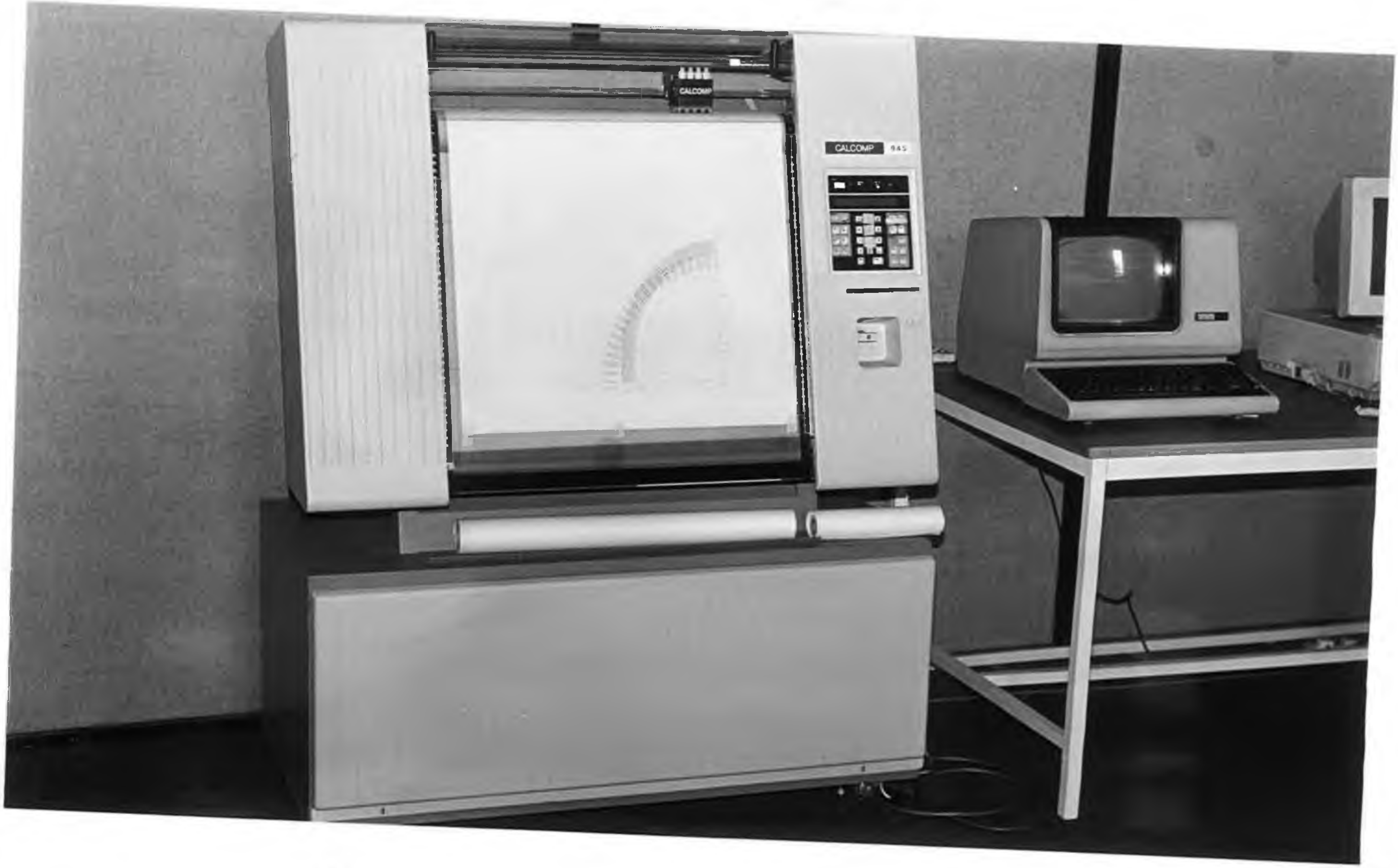

START
SET ITERATION COUNTER
ZERO FORCE VECTOR
SET UP ELEMENTS LOOP
0 ≤
> 0
ELEMENT YIELD AT START OF INCREMENT?
NO
CALCULATE INTERMEDIATE STRESS VALUES AT WHICH YIELD BEGINS
YES
COMPUTE INCREMENT FROM THIS POINT
EVALUATE ELASTIC-PLASTIC MATRIX $[D]^*_{ep}$ USING DEVIOTORIC STRESS
CALCULATE $\{\Delta\sigma\} = [D]^*_{ep}\{\Delta\epsilon\}$
EVALUATE STRESS TO BE SUPPORTED BY BODY STRESS
STORE CURRENT VALUES OF STRESS AND STRAIN
COMPUT NODAL FORCES CORRESPONDING TO THE EQUILIBRATING BODY FORCES
FINISH ELEMENT LOOP
NO
YES
SOLUTION CONVERGED
NO
YES
CALCULATE THE CURRENT STRESS, STRAIN AND DISPLACEMENT
CALCULATE THE INCREMENT OF STRESS AND STRAIN
SUBSTITUTE THE EXTERNAL FORCE BY THE EQUILIBRATING BODY FOR FORCES OF FREEDOM
UPDATE YIELD STRESS OF YIELD POINT
RETURN
END

FIG.(16)



Figure(17)The plotter"calcomp" with the terminal during the execution of the E.P. Program. Before submitting a large problem to be computed it is advisable to obtain the output for the first increment in order that an assessment of the suitability of increment size can be made, thus avoiding wasteful computer time.

5.9 COMPARISON OF THE OUTPUT OF THE E.P. PROGRAM WITH PAFEC FINITE ELEMENT PACKAGE

In order to evaluate the results of the E.P. Program a cantilever beam model was analysed. The results from the analytical formulae were then compared with the results from the developed E.P. Program and PAFEC Finite Element Package for plane stress analysis.

5.10 CALCULATIONS

5.10.1 Elastic Analysis

Figure (18) shows the cantilever beam model with the free end which was subjected to a downward concentrated force. Considering the displacements the following are obtained:

(i) Deflection due to the bending moment:

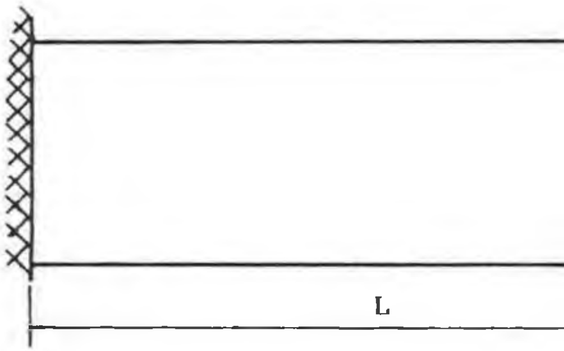
Deflection due to the bending $V_B = \frac{PL^3}{3EI}$

(ii) Deflection due to the shear forces:

$$V_s = \frac{6}{5}\frac{PL}{bdG}$$

Thus, the total deflection at the free end due to bending and shear force is

$$V = V_B + V_s$$

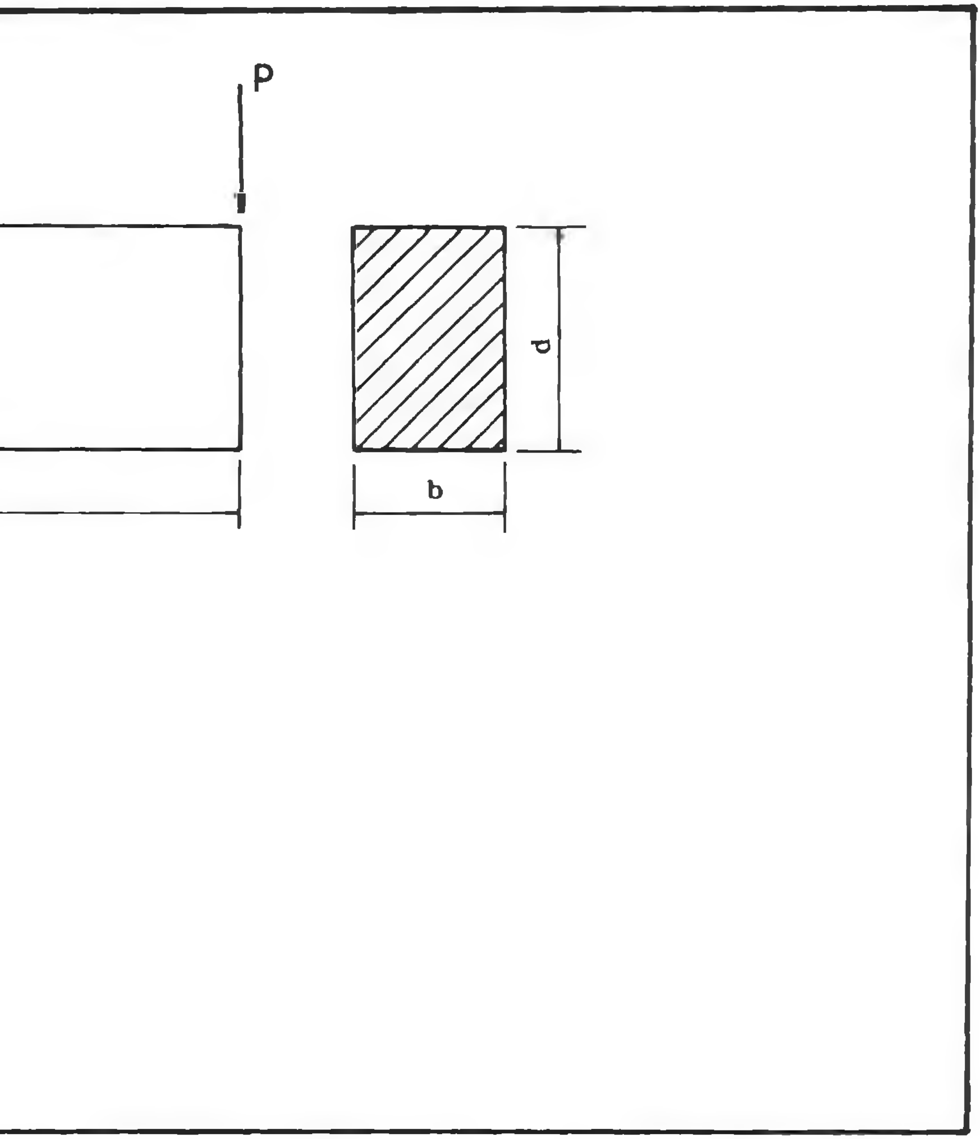


d=30mm

b=20mm

L=100mm

FIG(18)


P
d
b

$$= \frac{1}{3}\frac{PL^3}{EI} + \frac{6}{5}\frac{PL}{bdG} \tag{5.11}$$

The shear or rigidity modulus of elasticity G

$$G = \frac{E}{2\,(1+\gamma)} \tag{5.12}$$

(iii) Material Properties:

$E = 209000\ N/mm^2$

$\gamma = 0.3$

$\sigma_y = 227.53\ N/mm^2$

$H' = 0.00004$

This meant the material was as near as possible to elastic - perfectly plastic

5.10.2 Calculation for yield load

$$\frac{M_y}{I} = \frac{\sigma_y}{Y} \tag{5.13}$$

Where:

$$Y = \frac{d}{2}$$,therefore,

$$M_y = \frac{2I\sigma_y}{d} = \frac{\sigma_y\ .\ bd^2}{6} \tag{5.14}$$

$$M_y = L.P \tag{5.15}$$

From (5.14):

$$My = \frac{227.53 \times 20 \times (30)^2}{6} = 682590 \text{ N.mm}$$

From (5.15):

$$P = \frac{My}{L} = \frac{682590}{100} = 6825.9 \text{ N}$$

This load creates the plastic hinges at the outer of the top and the bottom fibres at the fixed end of the cantilever. The maximum deflection at the free end can be obtained from equation (5.11). The rigidity modulus can be calculated from equation (5.12), i.e.

$$G = \frac{0.209 \times 10^6}{2\ (1 + 0.30)} = 80384.615 \text{ N/mm}^2$$

So, from (4):

$$V = \left(\frac{1\text{x}6825.9 \times (100)^3 \times 12}{3\text{x}0.209\text{x}10^6\text{x}20\text{x}(30)^3}\right) + \left(\frac{6}{5} \cdot \frac{6825.9 \times 100}{20\text{x}30\text{x}80384.615}\right)$$

$$= 0.258 \text{ mm}$$

5.10.3 Calculation for the plastic load

The plastic load is the load which causes a plastic hinge to be formed at the fixed end. After that the cantilever becomes a mechanism, thus, the bending moment for a fully plastic condition is **Mp** and:

$$Mp = \frac{\sigma_y \, bd^4}{4} \qquad (5.16)$$

Also at support bending moment is:

$$M_p = LP_p$$

thus,

$$P_p = \frac{M_p}{L} \qquad (5.17)$$

From (5.16):

$$M_p = \frac{227.53 \times 20 \times (30)^2}{4} = 1023885 \text{ N.mm}$$

From (5.17):

$$P_p = \frac{1023885}{100} = 10238.85 \text{ N}$$

5.10.4 Results of the comparison

The finite element results were obtained for a set of various mesh configurations for the cantilever beam. The output data consists of all current element stress, strain, nodal displacement, final co-ordinate, reaction forces and graphic output. These were done to evaluate the performance in full capacity of the program.

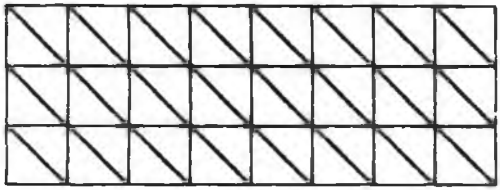
The following Figures and Tables are presented to illustrate the results obtained.

Figure (19) shows the various mesh configurations and the total number of elements and the number of rows as well.

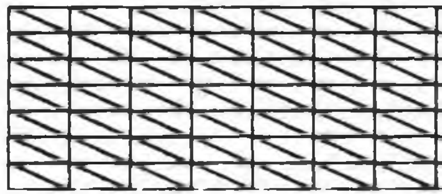
Table (4) shows the results obtained from the PAFEC Finite Element Package and E.P. Finite Element Program which has been developed for this work. These results were obtained for each number of elements in Figure (19), using the same data input. Also the Table shows the yield loads and collapse loads which were obtained from the finite element solution and those which were obtained from the analytical solution. Furthermore, the Table shows the percentage errors in the maximum deflection at the top and bottom of the free end of the cantilever in comparison with the exact solution from the formula. The Table also gives the computing time of the computer which is required to obtain the solution.

Table (5) shows the sequences of the applied loads on the cantilever and the maximum deflection corresponding to the applied load, which was obtained from the PAFEC and E.P. Finite Element Program as well. The results shown in this Table are based on mesh configuration number three which has seven rows as shown in Figure (19). This was chosen from Table (4) in order it to satisfy the accuracy with reasonable time for solution (C.P.U.) time.

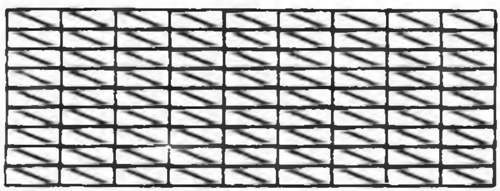
Figure (20) and Figure (21) shows the number of rows against the error in maximun deflection for the E.P. Finite Element Program and the PAFEC Package respectively



3 rows (48 elements)



7 rows (112 elements)



9 rows (162 elements)

FIG.(19)

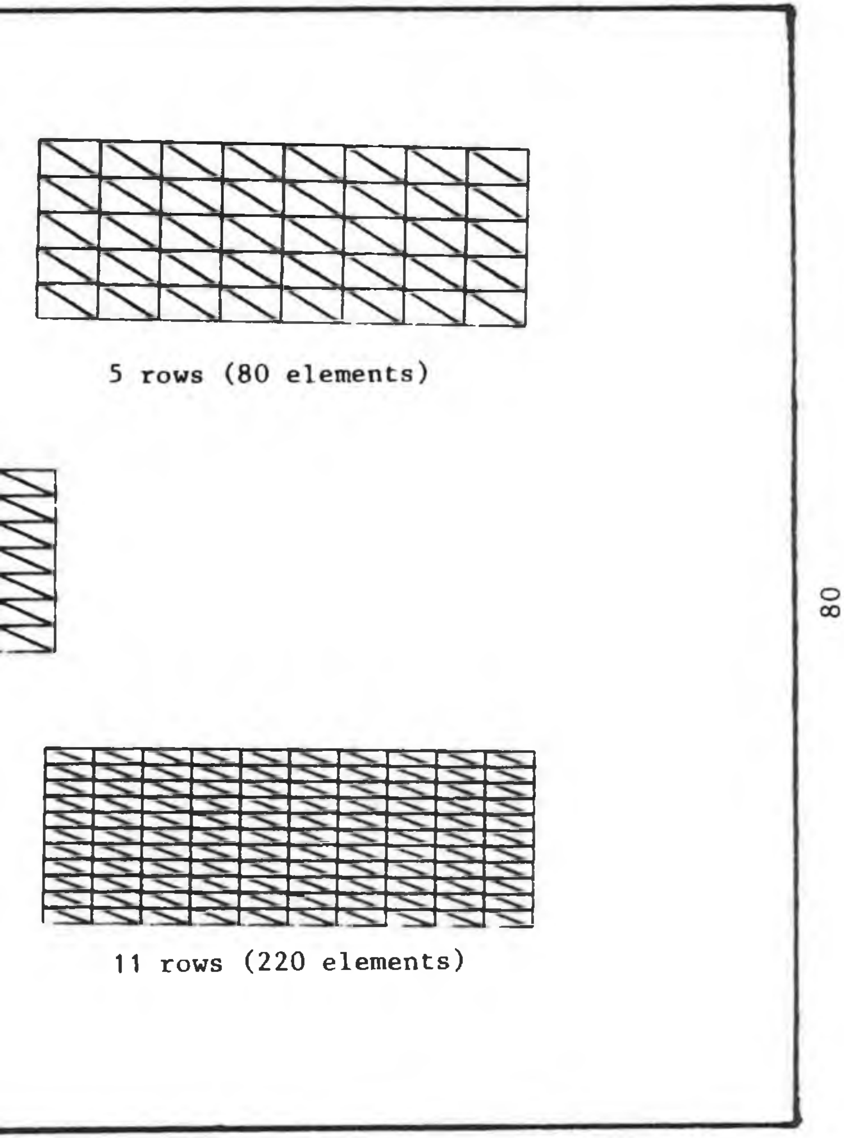

5 rows (80 elements)
11 rows (220 elements)
80

TABLE (4)

No. of rows	Yield Load (N)	Collapse Load (N)	PAFEC					E.P. PROGRAMME				
			Time for Solution C.P.U. (sec.)	Max. deflection at the top (mm)	Error %	Max. deflection at the bottom (mm)	Error %	Time for Solution C.P.U. (sec.)	Max. deflection at the top (mm)	Error %	Max. deflection at the bottom (mm)	Error %
3	10203.4	14284.76	112.43	0.271	5.00	0.265	2.7	117.86	0.271	5.00	0.265	2.7
5	9028.97	14446.35	189.57	0.267	3.4	0.262	1.5	231.18	0.267	3.4	0.262	1.5
7	8684.35	13894.96	256.26	0.265	2.7	0.261	1.2	309.73	0.266	3.1	0.261	1.2
9	8243.84	13190.14	343.29	0.265	2.7	0.260	0.8	797.92	0.264	2.3	0.260	0.8
11	7927.87	13477.37	516.65	0.263	1.9	0.259	0.4	1932.19	0.263	1.9	0.259	0.4
Exact Solution	6825.9	10238.85	---	0.258	---	---	---	---	0.258	---	---	---

No. of Load	Load (N)	Load %
1	8684.35	100
2	9552.78	10
3	10421.22	10
4	11289.65	10
5	12158.09	10
6	13026.52	10
7	13894.96	10

TABLE (5)

PAFEC	E.P. PROGRAMME
Deflection (mm)	Deflection (mm)
0.261	0.261
0.291	0.291
0.323	0.323
0.361	0.362
0.414	0.409
0.493	0.491
0.678	0.661

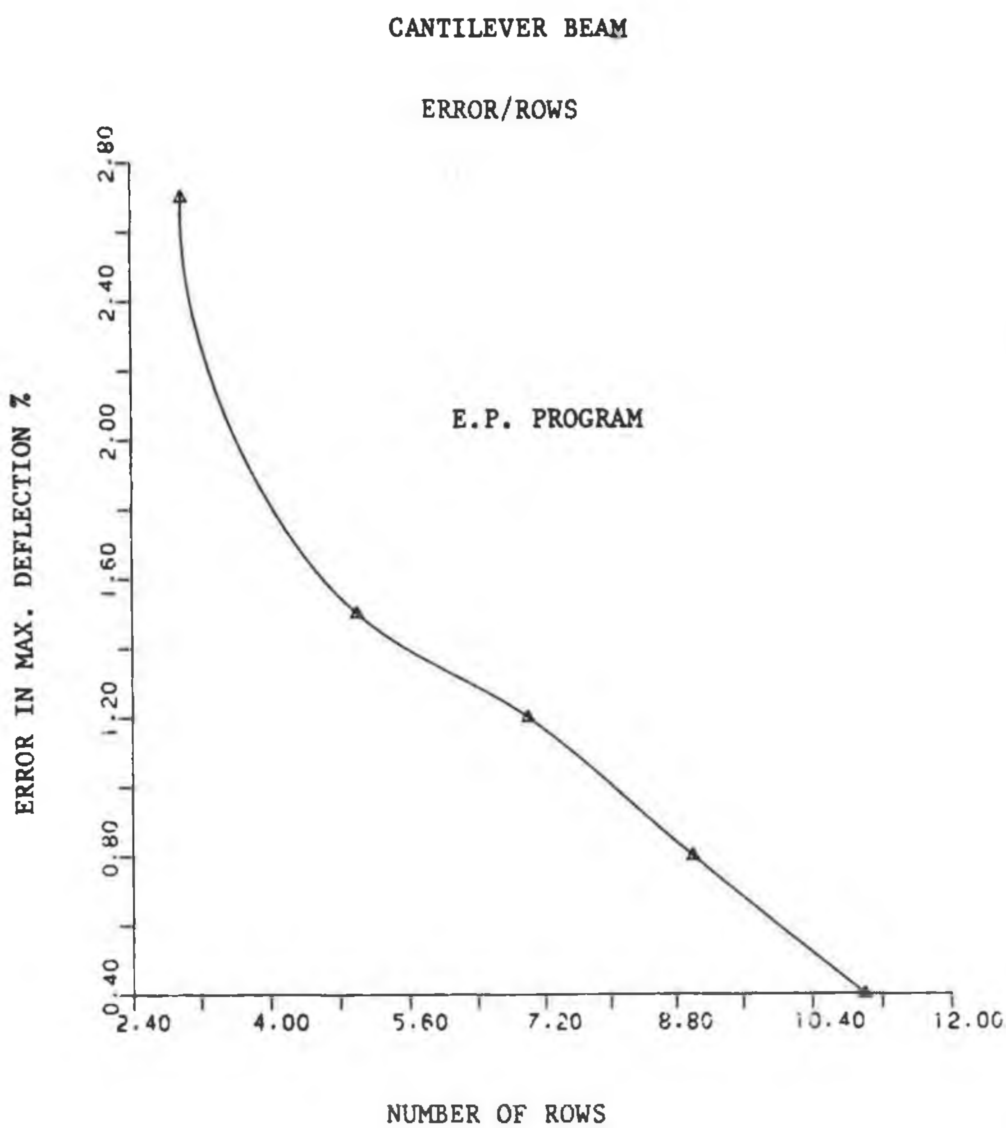

CANTILEVER BEAM
ERROR/ROWS
E.P. PROGRAM
ERROR IN MAX. DEFLECTION %
0.40
0.80
1.20
1.60
2.00
2.40
2.80
2.40
4.00
5.60
7.20
8.80
10.40
12.00
NUMBER OF ROWS

FIG.(20)

CANTILEVER BEAM

ERROR/ROWS

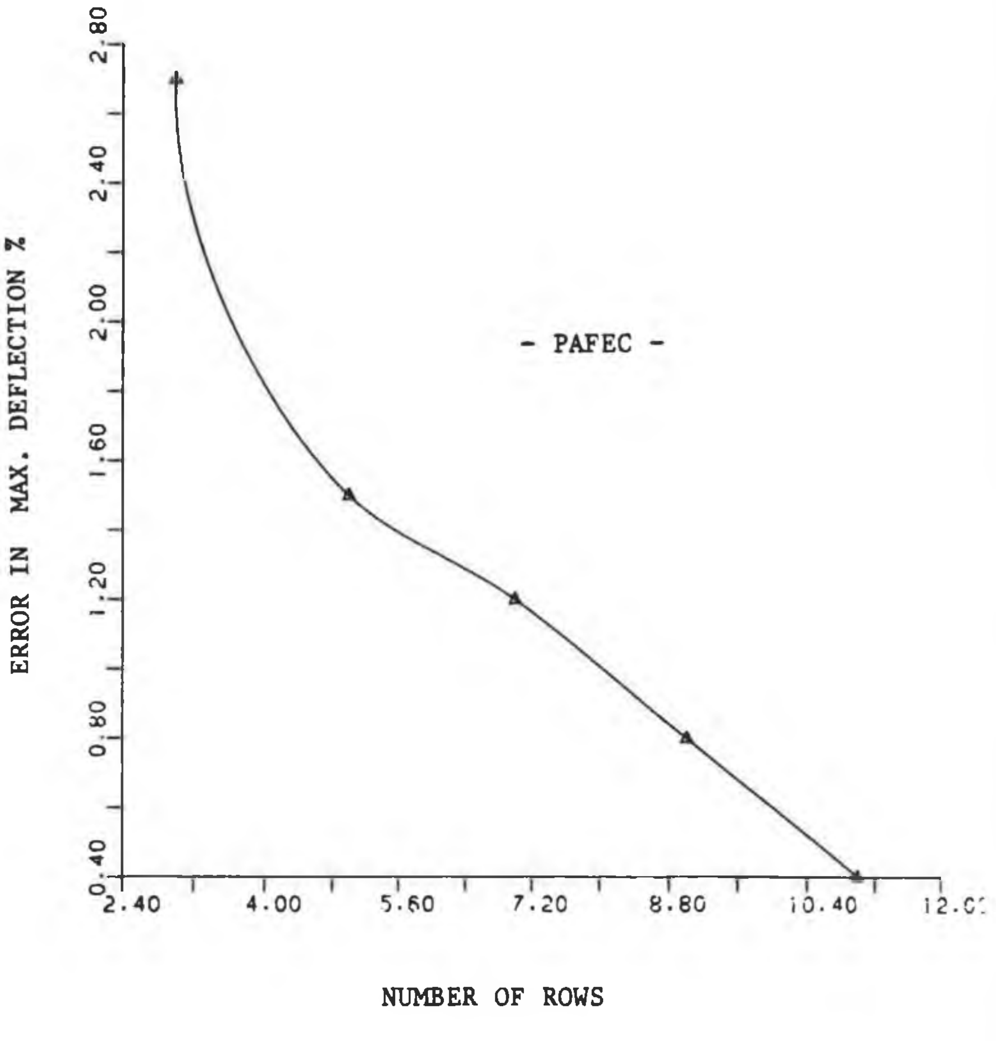


FIG.(21)

Figure (22) and Figure (23) show the number of rows against the time for solution for the E.P. Program and the PAFEC Finite Element, respectively.

Figure (24) shows the deflection against the load which was obtained from the E.P. Finite Element Program and the PAFEC Package for the mesh configuration in Table (5).

Figure (25) is the graphical output of the cantilever beam before and after loading.

5.10.5 Discussion

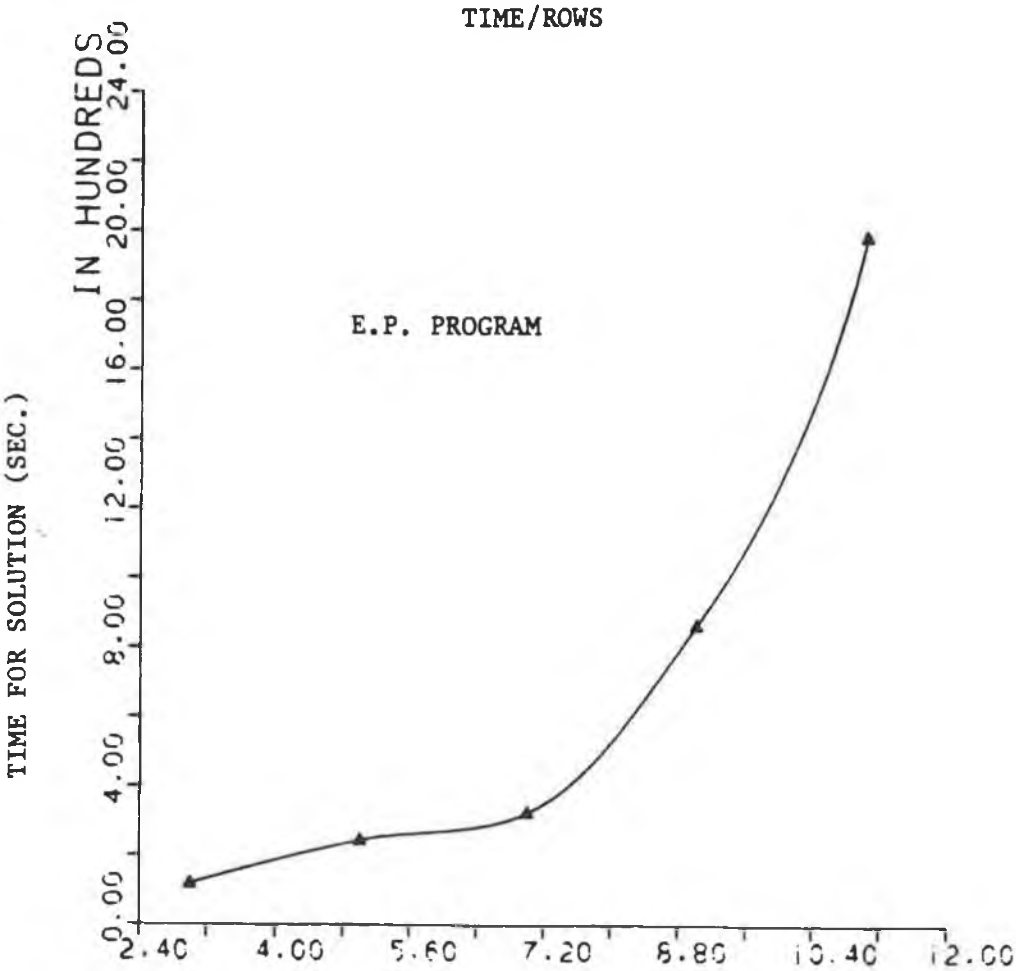
The yield load, collapse load and maximum deflection depending on the yield load were calculated previously by the conventional analytical formulas. However, the yield load, in terms of finite element, is the load causing the first yield. It is convenient to start the incremental process only from that point to relate the subsequent load increment to the load at which the first yield appears.

The material properties and the plane elasticity condition were chosen in order that a direct comparison could be made between the results given by the formula, the PAFEC Finite Element Package and E.P. Finite Element Program. The formula considers the shear forces which need to be considered because the cantilever here is short in comparison to its depth.

From the previous calculation and the results from both the E.P. Finite Element Program and the PAFEC Finite Element Package for different meshes, Table (4) shows that the best results were given by the finer mesh. However, the finer mesh needs more time to complete the solution. So a more realistic mesh, after considering the time for solution and the error as illustrated in Figures (20) and (22), is the mesh which has 7 rows. From the calculation the maximum deflection for that mesh at the free end of the cantilever is:

CANTILEVER BEAM

TIME/ROWS



NUMBER OF ROWS

FIG.(22)

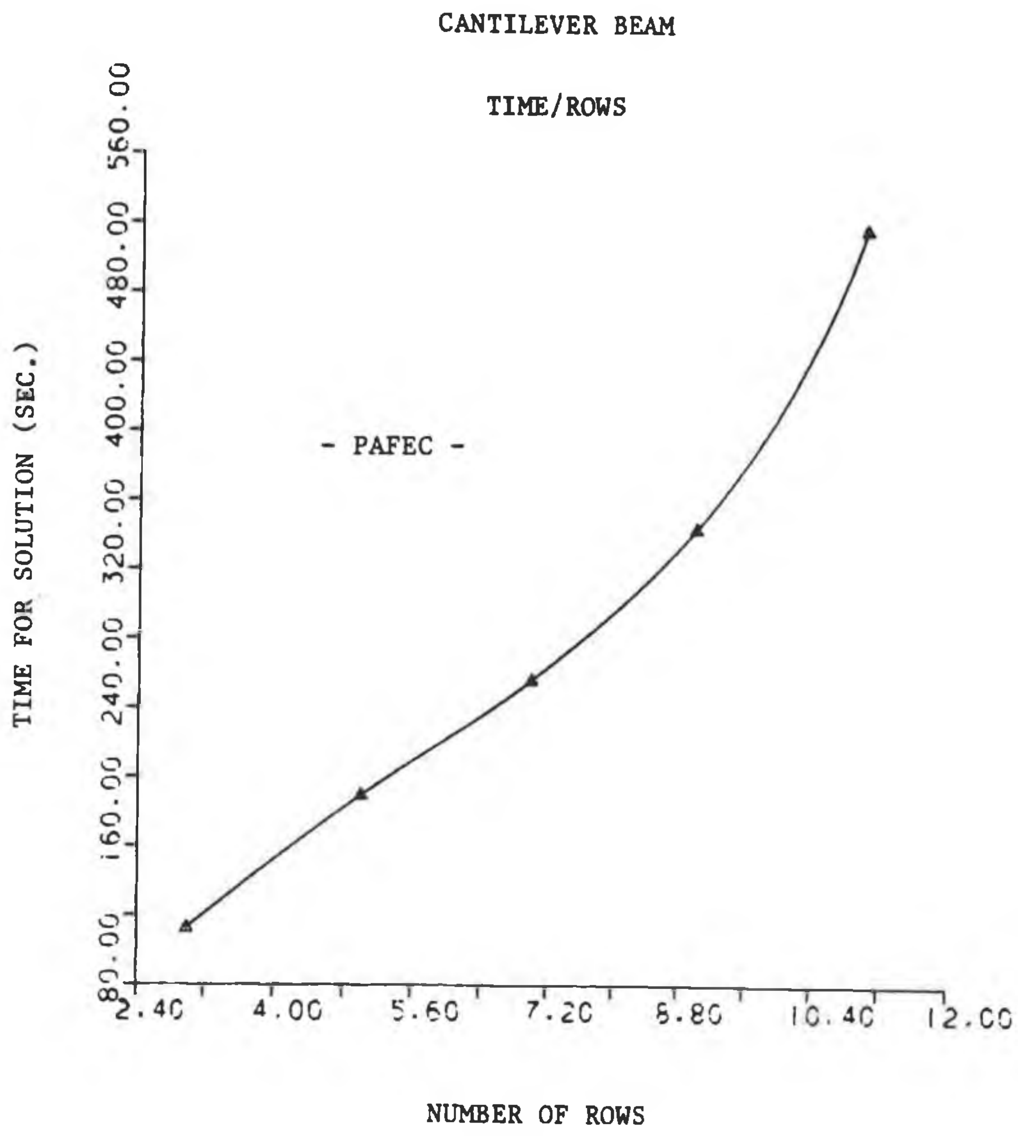
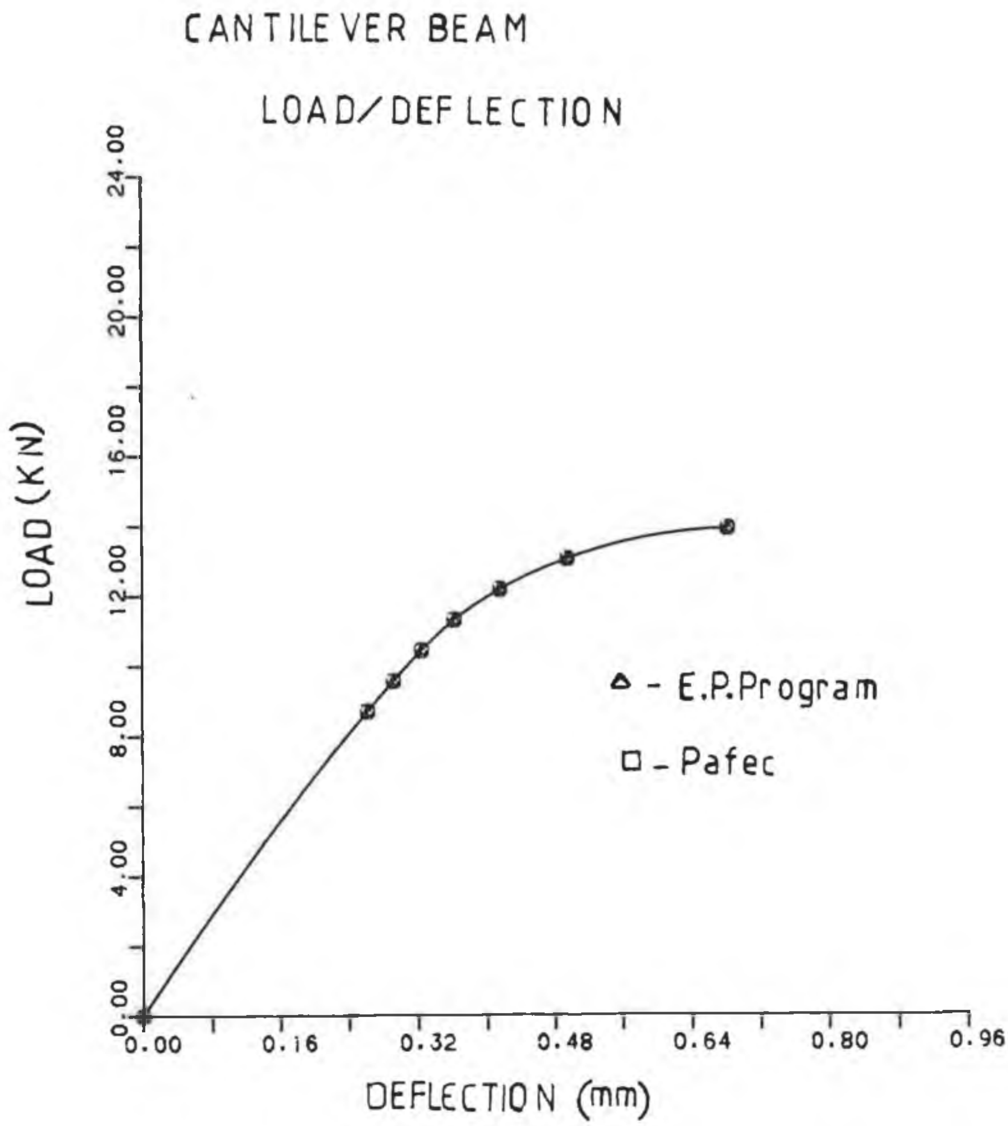

CANTILEVER BEAM
TIME/ROWS
- PAFEC -
TIME FOR SOLUTION (SEC.)
80.00
160.00
240.00
320.00
400.00
480.00
560.00
2.40
4.00
5.60
7.20
8.80
10.40
12.00
NUMBER OF ROWS

FIG. (23)



CANTILEVER BEAM
LOAD/DEFLECTION
LOAD (KN)
24.00
20.00
16.00
12.00
8.00
4.00
0.00
0.00
0.16
0.32
0.48
0.64
0.80
0.96
DEFLECTION (mm)
Δ - E.P.Program
□ - Pafec

FIG.(24)

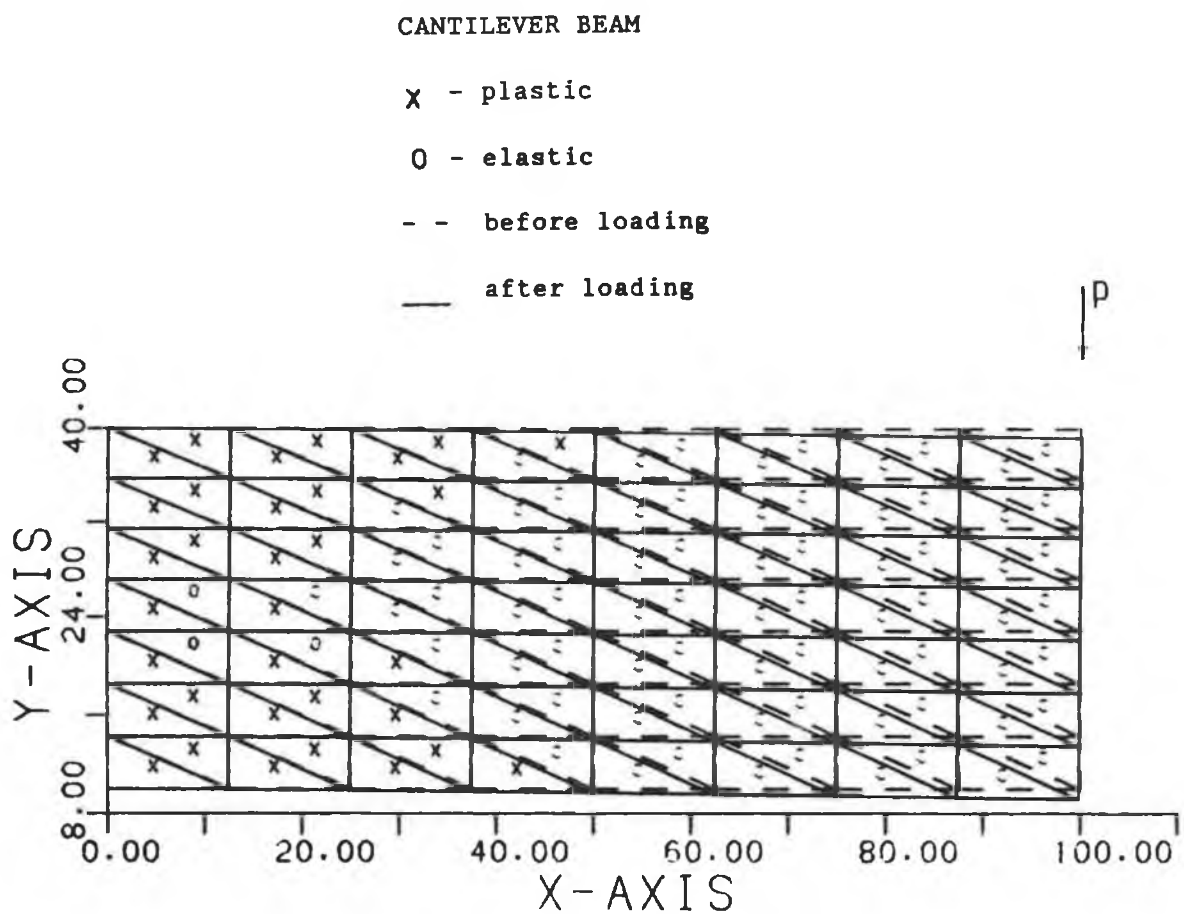

CANTILEVER BEAM
X - plastic
0 - elastic
- - before loading
after loading
P
40.00
24.00
8.00
Y-AXIS
0.00
20.00
40.00
60.00
80.00
100.00
X-AXIS

FIG.(25)

$$V = 0.258 \text{ mm.}$$

The maximum deflection from the results, at the top node of the free end of the cantilever is:

$$V_t = 0.266 \text{ mm}$$

This gives a large error of 3.1%, whereas a more realistic figure of the deflection is given when the node at the bottom of the extreme end is considered. This gives a 1.2% error which is obtained from either the E.P. Finite Element Program or the PAFEC Finite Element Package. The results of maximum deflection from both have very slight differences which perhaps refer to the different format accuracy in both of them. In general the accuracy of the maximum deflection increases by increasing the number of elements. However, increasing the number of elements resulted in an increase of the solution time, which is financially costly. Therefore, a more realistic figure of the meshes is the mesh which takes less computer time, with acceptable accuracy. This can be done by inducting Figure (20) and Figure (22) together.

Figure (24) shows the relationship between the load deflection for the chosen mesh from E.P. Finite Element Program and the PAFEC Finite Element Package. The same data was applied so that the same collapse load was obtained from both. Also the deflections, according to the load increment, gave slight differences which were created by the different format accuracy between both of the programs, as previously mentioned.

However, the major difference in the results between the E.P.Finite Element Program and the PAFEC Finite Element Package was in the solution time. This can be easily seen by the comparison of Figure (22) with Figure (23). The E.P. Finite Element Program requires more time for solution than the PAFEC Finite Element Package , this is because the E.P. Finite Element Program uses the entire overall stiffness structure matrix. Using the banded matrix with the sophisticated subroutine could reduce the solution time significantly. However, this is felt to be research for further study.

Figure (25) shows the graphical output of the cantilever beam of 7 rows before and after loading; the plastic hinges at the two outers fibres of the cantilever at the top and the bottom; and the plastic hinges between the previous ones. This intermediate hinge appears when the loading reachs the collapse. In this case the cantilever becomes a mechanism.

CHAPTER 6

EXPERIMENTAL WORK, EQUIPMENT AND MATERIAL

6.1 INTRODUCTION

In order to establish the effectiveness of the E.P. Program, comparison was made between experimental results and those predicted theoretically. To facilitate such comparison experiments were carried out on ring structures under different loading conditions. These rings were subjected to static loads causing a large elastic-plastic deformation.

A 150 tonne hydraulic press was modified and instrumented to carry out these tests. These modifications are as follows:

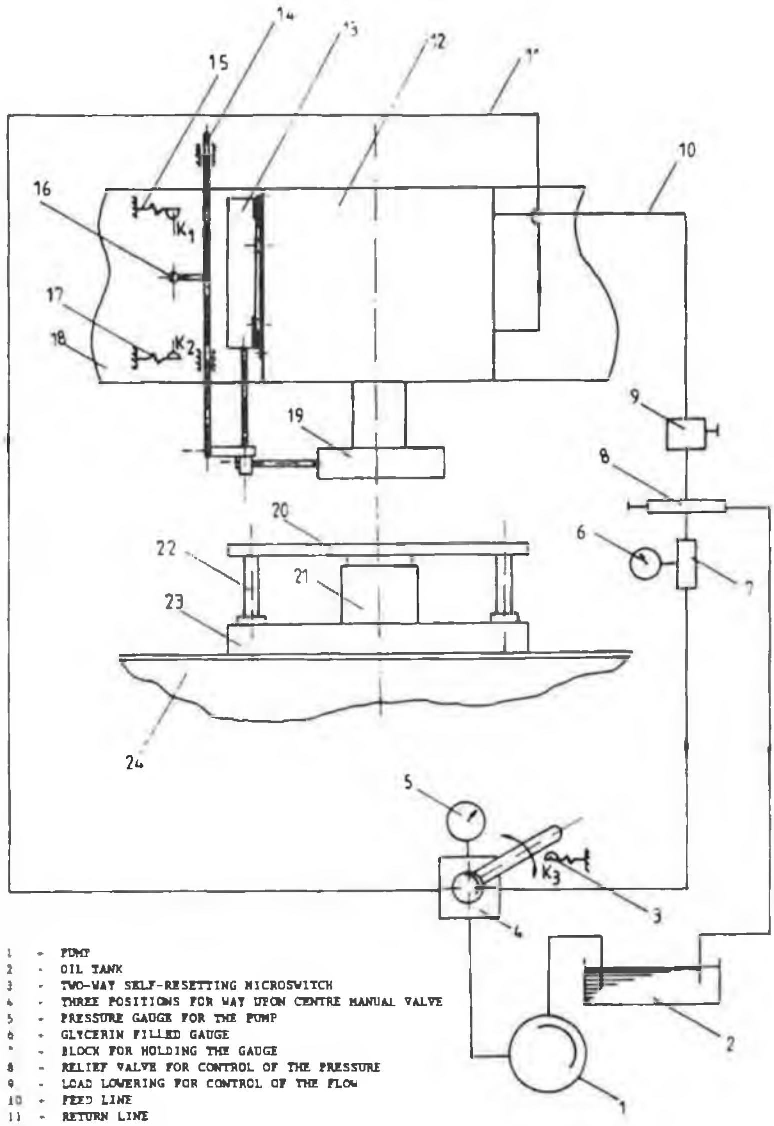
- design and manufacture of the compression platen and loading tools of the machine
- modifying the hydraulic system to control the movement of the piston of the press
- incorporate displacement monitoring system by means of a potentiometer
- instrument the press for measuring load and displacement electronically

Figure (26) shows the press after these modifications.

6.1.1 Modification of the hydraulic system

The purpose of modifying the hydraulic system is to control the downward motion of the hydraulic piston of the press (fast-slow) according to the User's need. Figure (27) shows the schematic diagram of the hydraulic system with its modification. This modification consists of incorporating a relief valve for controlling the pressure, and a glycerin filled gauge for measuring the pressure in the feed line. These components were added to the feed line to control the downward motion of the piston.

Figure(26) The press machine after the modifications



1 - PUMP
2 - OIL TANK
3 - TWO-WAY SELF-RESETTING MICROSWITCH
4 - THREE POSITIONS FOR WAY UPON CENTRE MANUAL VALVE
5 - PRESSURE GAUGE FOR THE PUMP
6 - GLYCERIN FILLED GAUGE
7 - BLOCK FOR HOLDING THE GAUGE
8 - RELIEF VALVE FOR CONTROL OF THE PRESSURE
9 - LOAD LOWERING FOR CONTROL OF THE FLOW
10 - FEED LINE
11 - RETURN LINE
12 - HYDRAULIC CYLINDER
13 - POTENTIOMETER
14 - MICROSWITCH GUIDE
15 - ONE-WAY SELF-RESETTING MICROSWITCH
16 - TOUCHED PIN
17 - ONE-WAY SELF-RESETTING MICROSWITCH
18 - BODY OF THE PRESS MACHINE
19 - TOP PLATFORM
20 - BOTTOM PLATFORM
21 - LOAD CELL
22 - GUIDES
23 - BASE PLATFORM
24 - BODY OF THE PRESS MACHINE

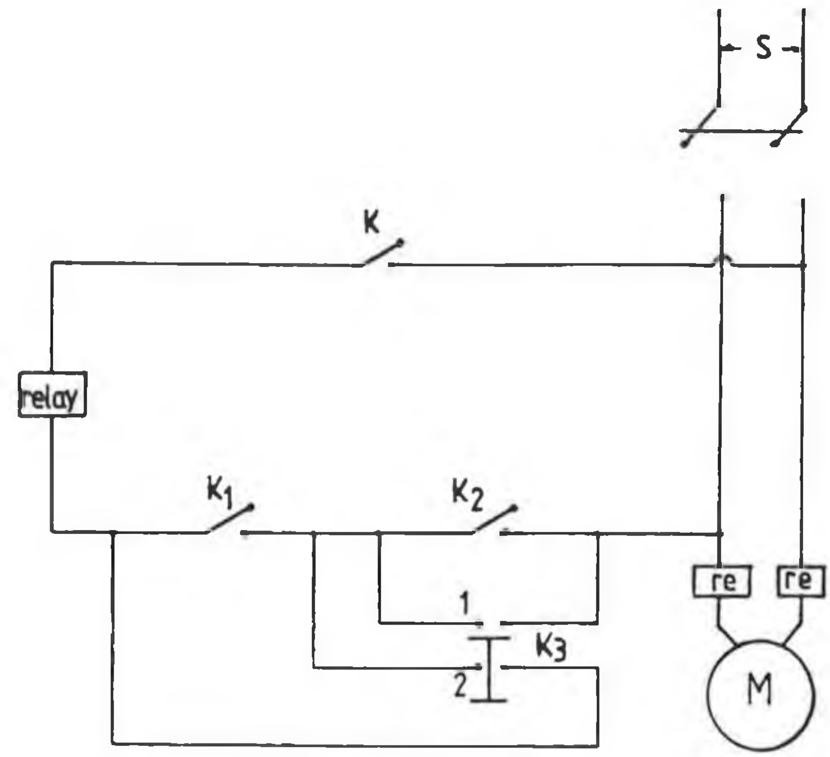
FIG.(27)

6.1.2 Monitor and control the movement of the hydraulic piston with the potentiometer

A potentiometer (10 KV) with a stroke length of 200 mm was mounted on the press to measure the piston's displacement. Physically, the stroke of the piston was longer than the stroke of the potentiometer, and as a result a microswitch system was designed and connected to the electrical circuit of the motor of the hydraulic pump to protect the potentiometer. Figure (27) shows the design of the hydraulic system with control of the stroke which is linked together by an electric circuit shown in Figure (28) which shows the principle of the microswitch system. Firstly, the hydraulic piston at the top point K_1 is opened, K_3 is then moved to point (2) by moving the handle of the "three position for way manual valve" as shown in Figure (27). The direction of the piston movement will then change and the motor will run by closing K. Secondly, when the piston is at the lower point K_2 is opened. However, to close the electrical circuit and to change the direction of the piston, the handle of the manual valve should be moved to the other initial position, so that K_3 comes back to the point (1) and K is closed. Thirdly, the electrical circuit will be closed by closing K if the piston is at a point between K_1 and K_2. Figure (29) shows a photograph of the position of the microswitch K_3 on the machine with the hydraulic system and Figure (30) shows a photograph of the assembly design of the potentiometer with microswitches K_1,K_2 and Figure (31) shows the design of the individual components to the microswitch system of this assembly design.

6.1.3 Design of the platen and tools for the press

Platen of the press and loading tools shown in Figure (32), consist of two knives (upper and lower with clamp); two top platforms (1 large and 1 small); two lateral walls; a bottom plate, a base plate, four guides, and a load cell. Figures (33) to (40) shows the detailed drawing of these components.



THE ELECTRICAL CIRCUIT CONNECTING THE MICROSWITCHES

K - main manual switch

K_1)
) one way self resetting
K_2)

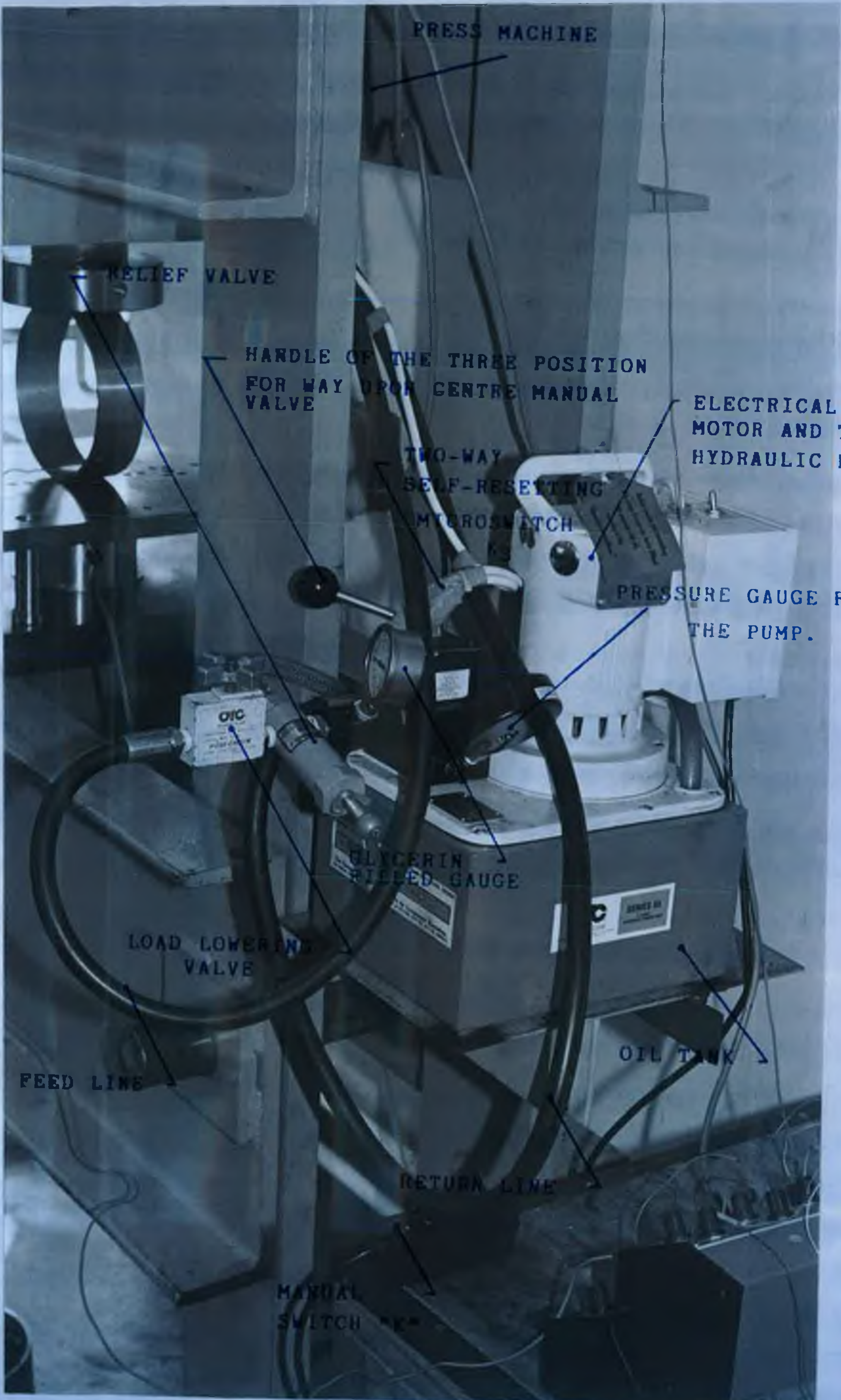
K_3 - two way self resetting

re - relay

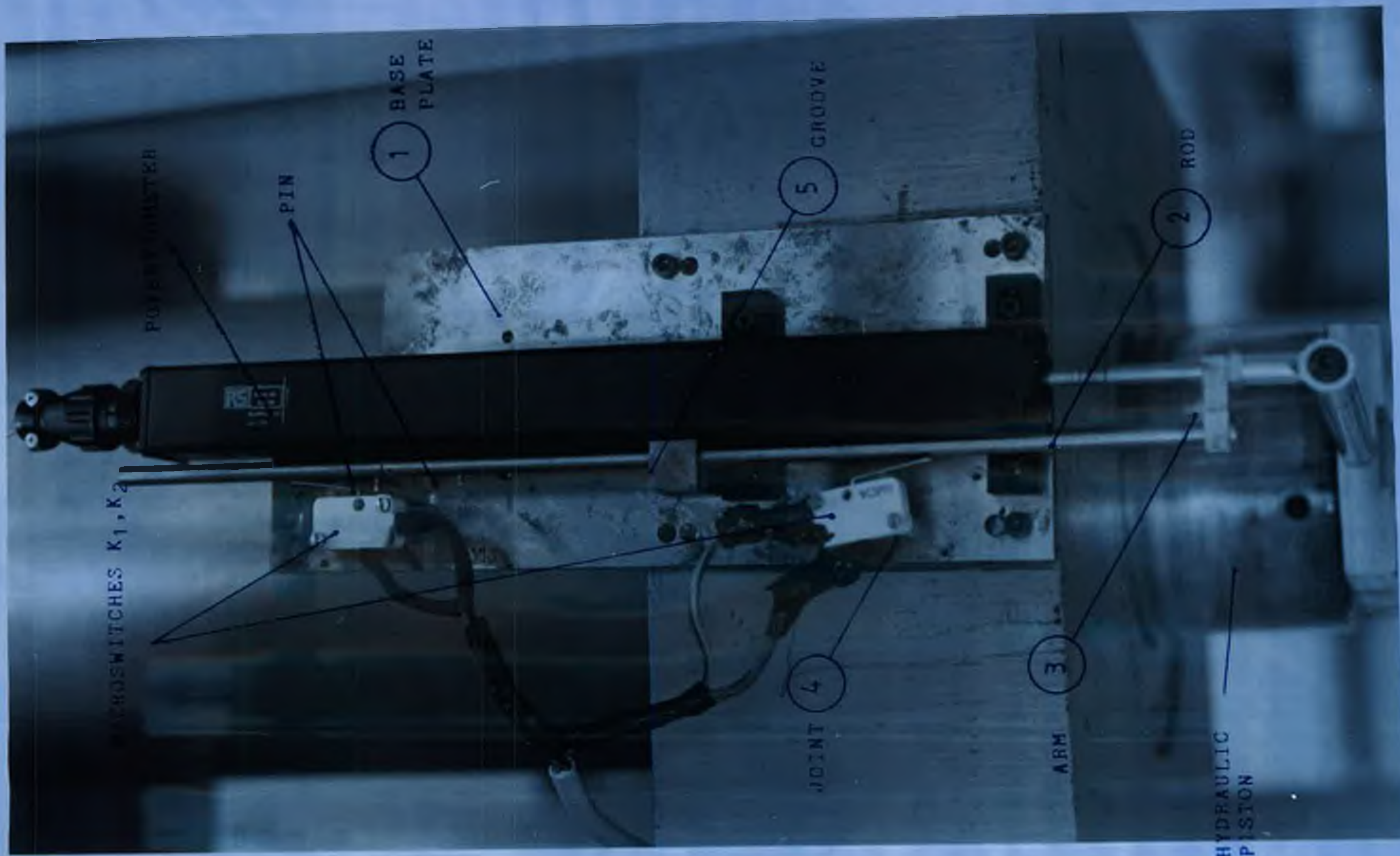
S - source of electricity

M - electrical motor

FIG.(28)



Figure(29) The modification of the hydraulic system with the microswitch k_3



Figure(30) The potentiometer with the microswitches k_1,k_2

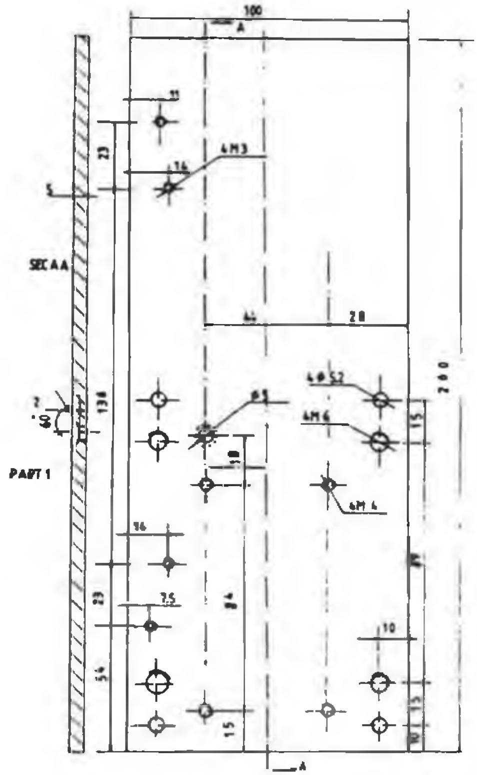
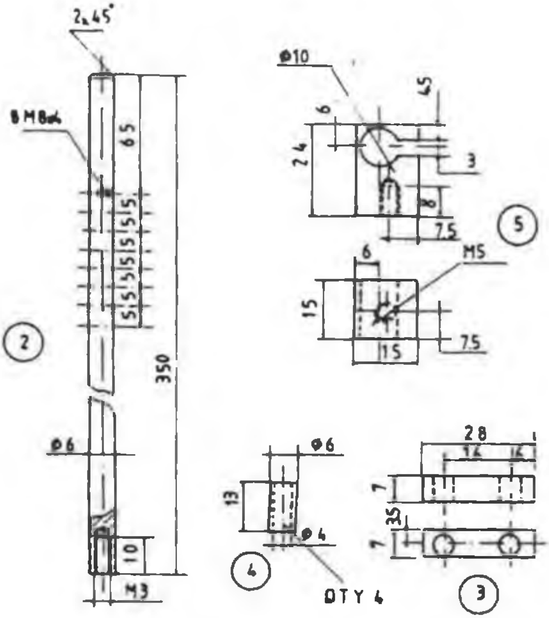

100
A
11
23
4M3
16
5
SEC A A
44
28
4Φ5.2
Φ5
134
4M6
15
260
2
PART 1
10
4M4
16
23
7.5
84
10
54
15
15
10
A

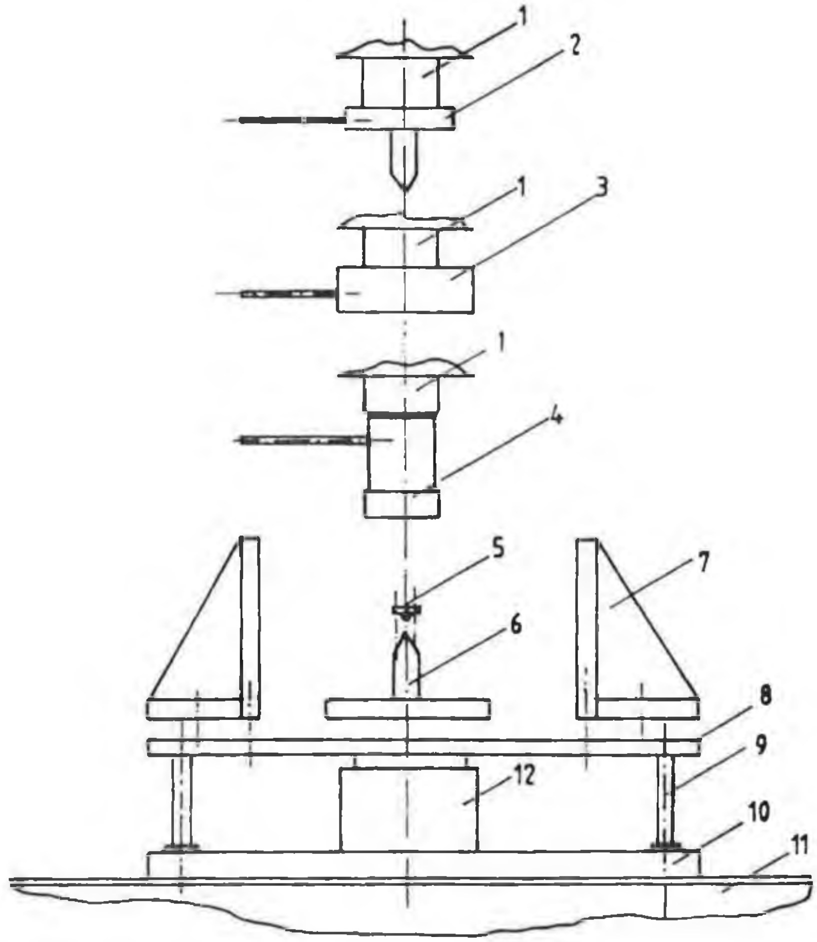
FIG.(31)



MATERIAL: IASI 1014

QUANTITY: 1

DIMENSION: mm



1 - HYDRAULIC PISTON
2 - UPPER KNIFE
3 - TOP PLATFORM (LARGE)
4 - TOP PLATFORM (SMALL)
5 - CLAMP
6 - LOWER KNIFE
7 - LATERAL WALL
8 - BOTTOM PLATFORM
9 - GUIDES
10 - BASE PLATFORM
11 - BODY OF THE PRESS MACHINE

FIG.(32)

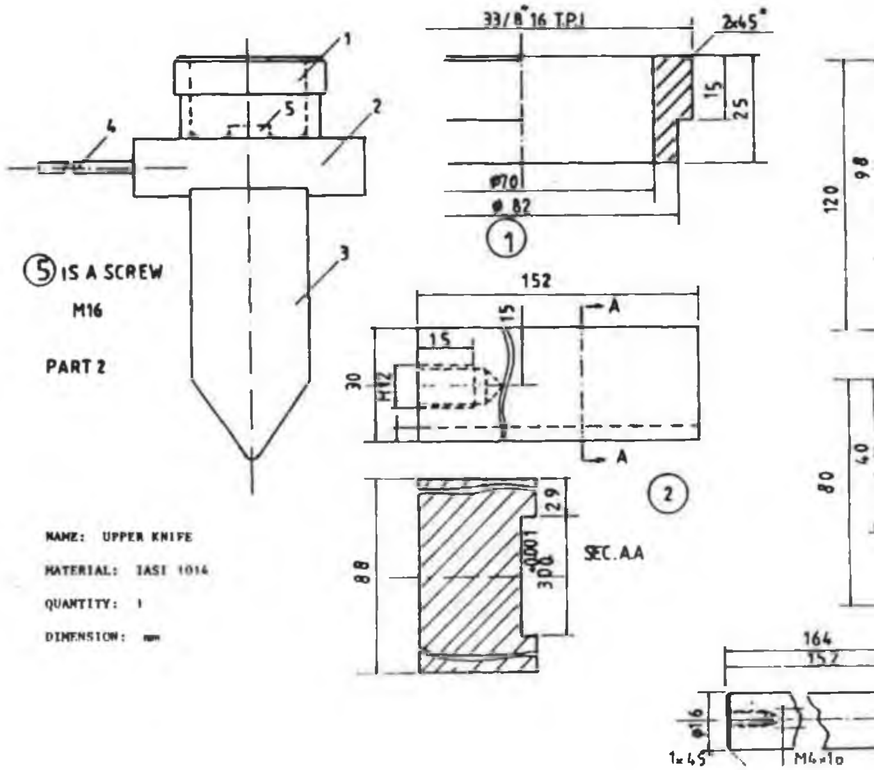
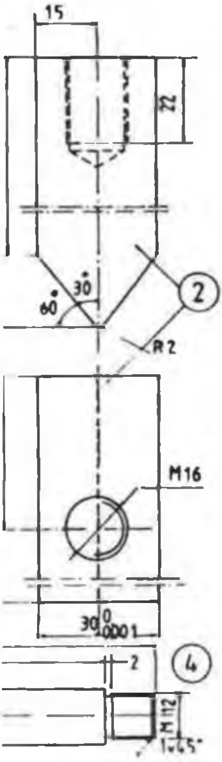

33/8" 16 T.P.I
2x45°
15
25
ø70
ø 82
1
2
3
4
5
(5) IS A SCREW
M16
PART 2
152
A
15
15
30
H12
29
88
SEC.AA
NAME: UPPER KNIFE
MATERIAL: IASI 1014
QUANTITY: 1
DIMENSION: mm
120
98
80
40
164
152
ø16
1x45°
M4x10

FIG.(33)


15
22
30°
60°
2
R2
M16
4
2
M12
1x45°

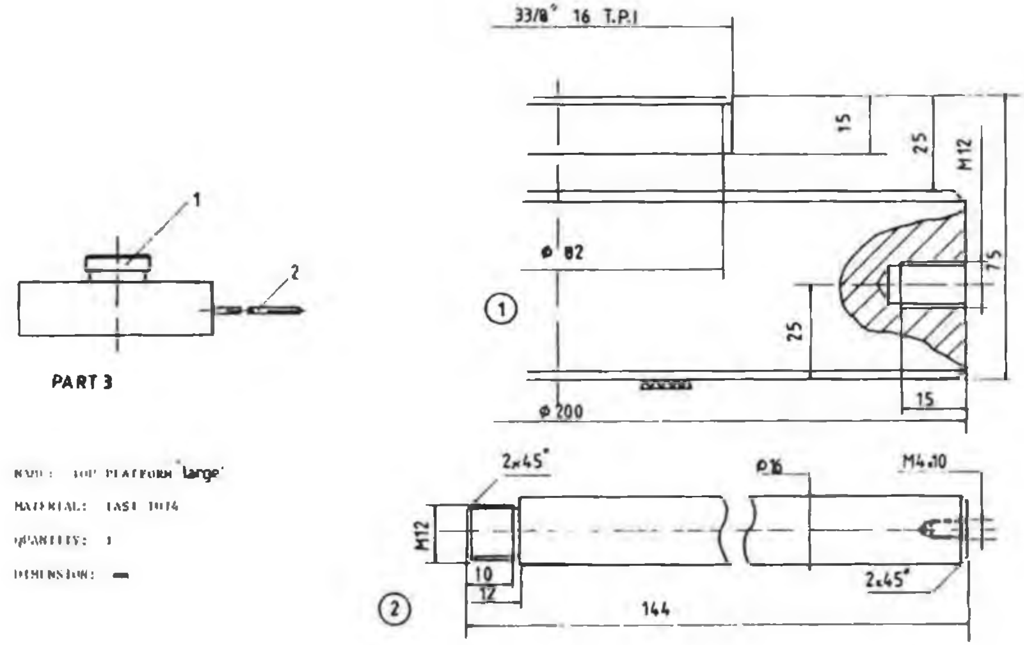
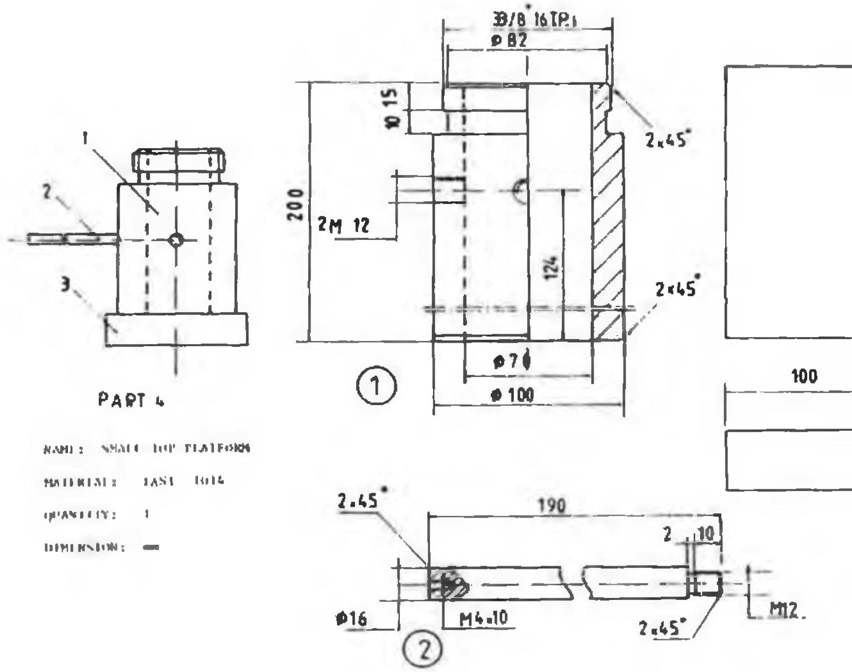
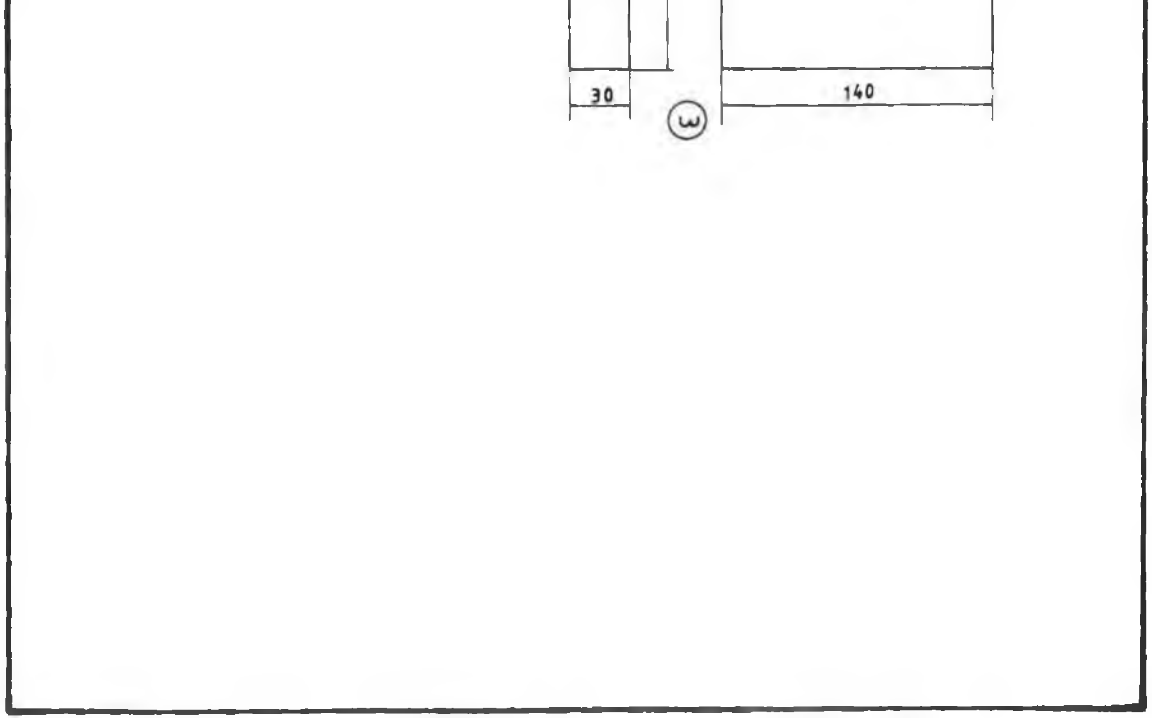

PART 3
1
2
large
33/8" 16 T.P.I
15
25
M12
φ 82
75
25
15
φ 200
2x45°
φ16
M4x10
M12
10
12
2x45°
144

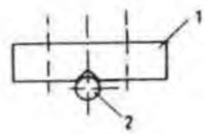
FIG.(34)



1
2
3
PART 4
3/8" 16TP.I
ø 82
ø 15
200
2M 12
124
2x45°
2x45°
ø 70
ø 100
①
100
2.45°
190
2 10
ø16
M4x10
2x45°
M12.
②

FIG.(35)


30
140
3



PART 5

NAME: CLAMP

MATERIAL: CAST IRON

QUANTITY: 1

DIMENSION: —

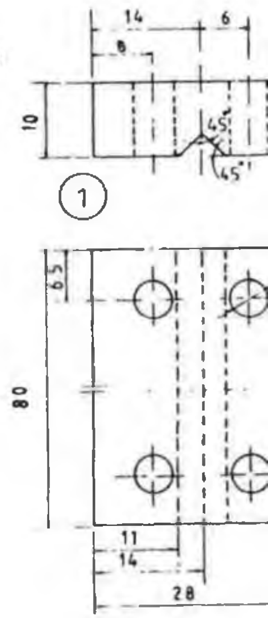
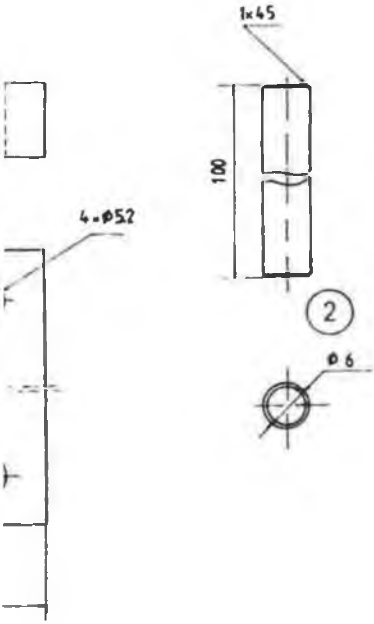


FIG.(36)



1x45
100
4-Ø5.2
2
Ø6

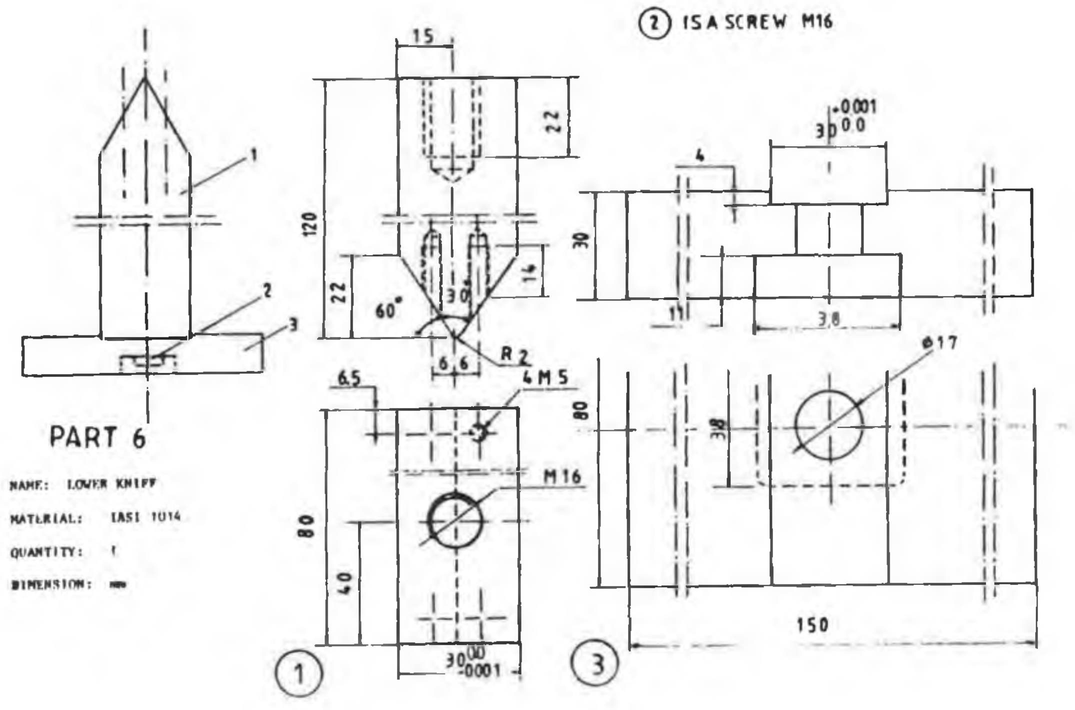
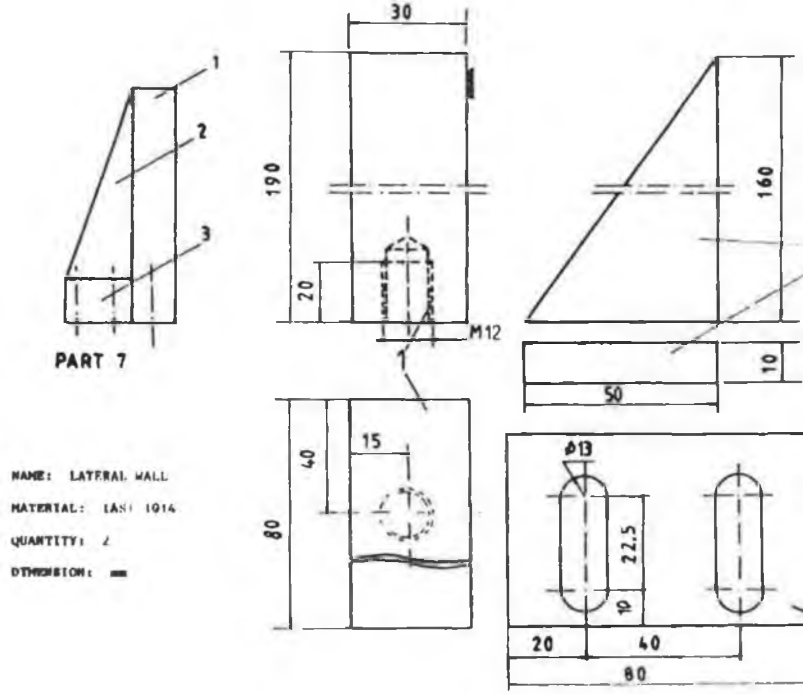
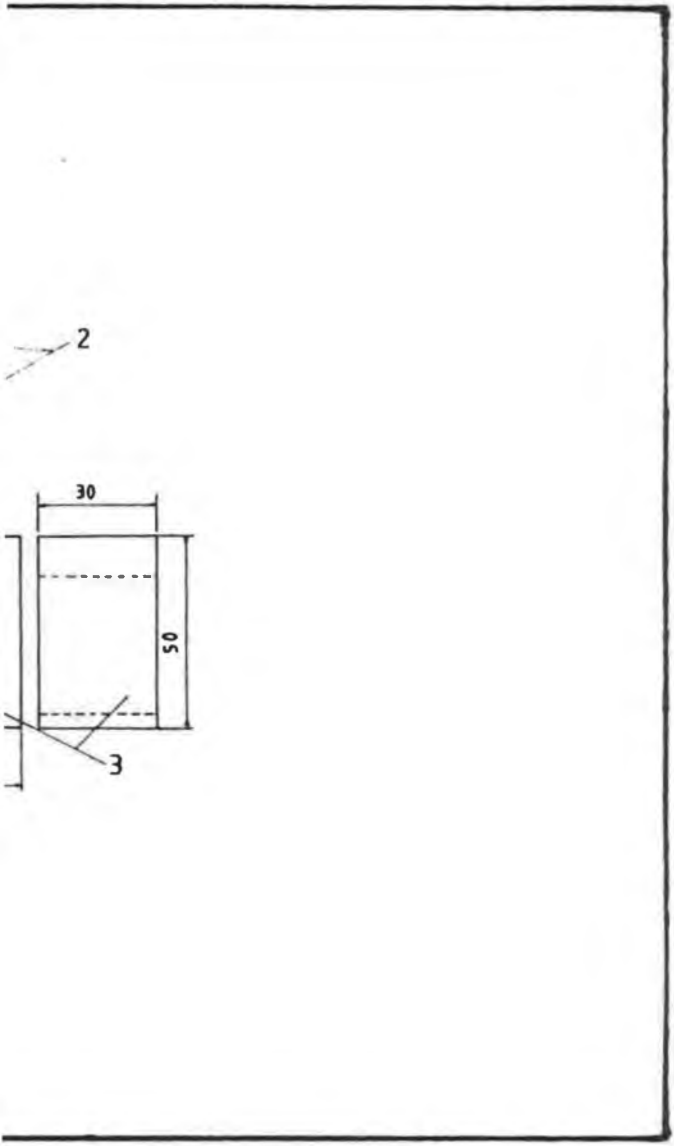

② IS A SCREW M16
PART 6
NAME: LOWER KNIFF
MATERIAL: IASI 1014
QUANTITY: 1
DIMENSION: mm
15
22
120
22
14
60°
30
R 2
6 6
6.5
4 M 5
M 16
80
40
$30^{0.0}_{-0.001}$
$30^{+0.001}_{0.0}$
4
30
11
38
ø17
80
38
150
1
3

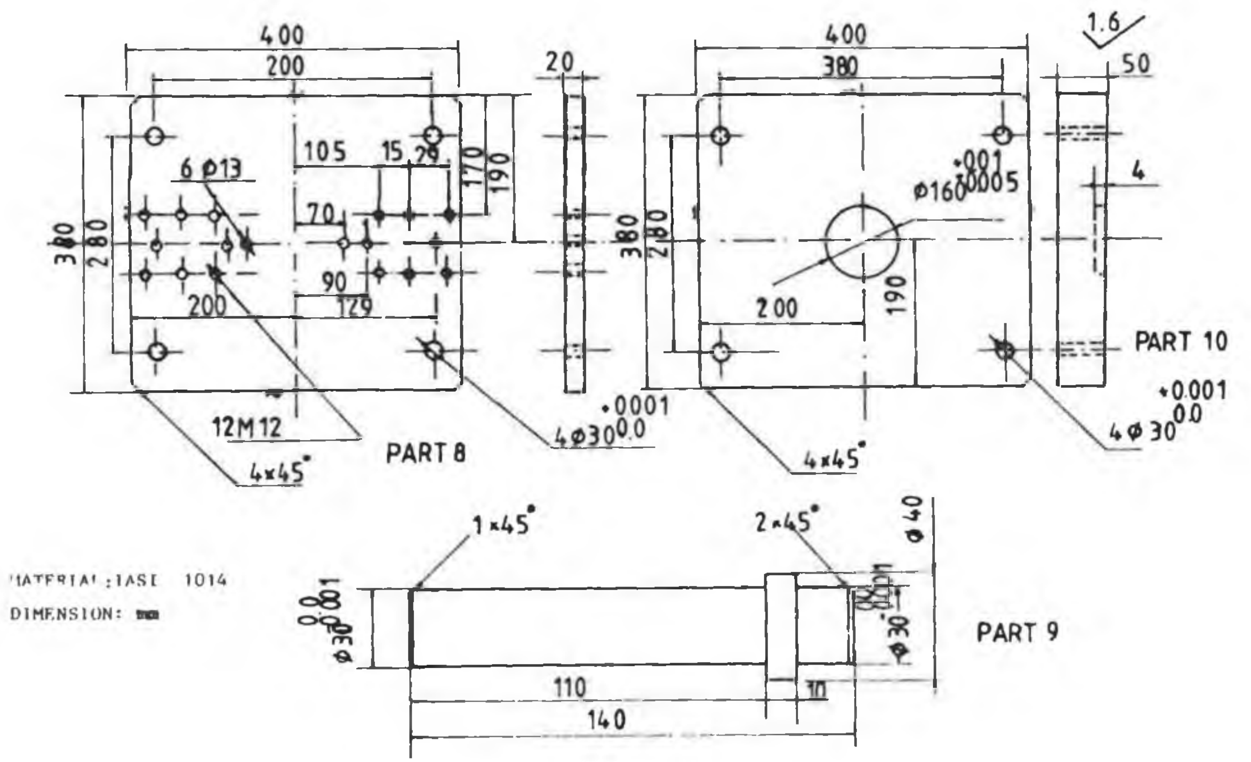
FIG.(37)



PART 7
1
2
3
30
190
20
M12
1
40
15
80
160
10
50
Φ13
22.5
10
20
40
80
NAME: LATERAL WALL
MATERIAL: AISI 1014
QUANTITY: 2
DIMENSION: mm

FIG.(38)


2
30
50
3



PART 8
PART 9
PART 10
12M12
6 Φ13
4×45°
1×45°
2×45°
Φ160
Φ40
MATERIAL: IASI 1014
DIMENSION: mm

FIG.(39)

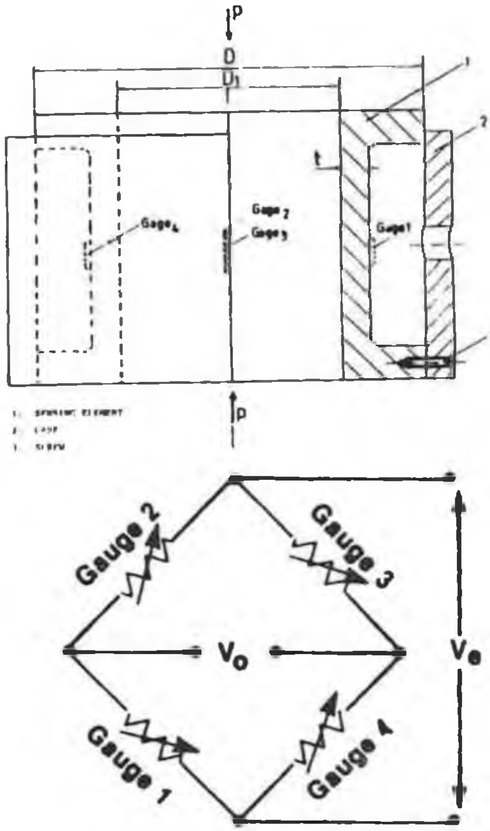
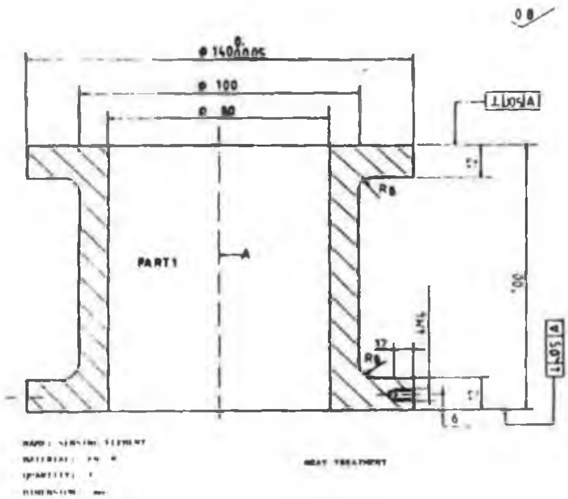
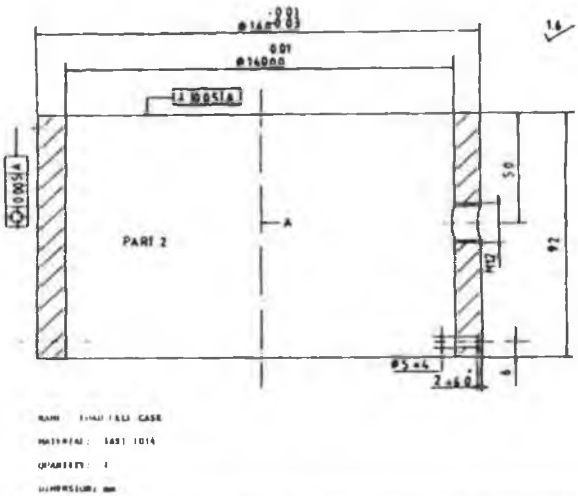

P
D
D1
t
Gage 2
Gage 3
Gage 4
Gage1
P
Gauge 2
Gauge 3
Gauge 1
Gauge 4
Vo
Ve

FIG.(40)


0.8
ø 100
ø 90
R8
PART 1
A
100
HEAT TREATMENT


1.6
PART 2
A
50
92
M12
CASE

Figures (41) and (42) show the calibration curve of the load cell in its "as tested" and "refined" forms respectively.

6.1.4 Instrumentation

Figure (43) shows a photograph of how the instruments are connected to each other.
The potentiometer was connected to the power supply of 10-15 DCV, through the displacement compensator. The output of the potentiometer was plotted on to the **x-direction** of the chart-plotter through the compensator as well. The function of the compensator is to modify the position of the pen of the chart plotter because when the measurement start from a large output of the potentiometer the pen goes outside the scale. The correct measurement starts from the zero **x-y position** of the pen. The circuit of this compensator is shown in Figure (44). The purpose of the condensor C_1 is to reduce the noise of the input whereas C_2 and C_3 are to reduce the noise of the potentiometer. The Ref. is to maintain 10.4 volts as an output if the input from the power supply varies between 10-40 volts, whereas R_1 is to make the ouput of Ref 10 volts exactly.

The load cell was connected to the chart-plotter so that its output is plotted in the **y-direction** through the mini-balance. Also the resistors were used to complete quarter of the bridge, when the strain on the surface of the ring was measured.

6.2 THE EXPERIMENTAL WORK

The E.P. Program has been developed for plane stress and plane strain deformation. The results from this program were compared with the results from the PAFEC Finite Element package.

To check the mathematical and physical model the results were compared with the data obtained from the experimental work. This experiment was carried out on the ring structure subjected to static loads in three different modes:

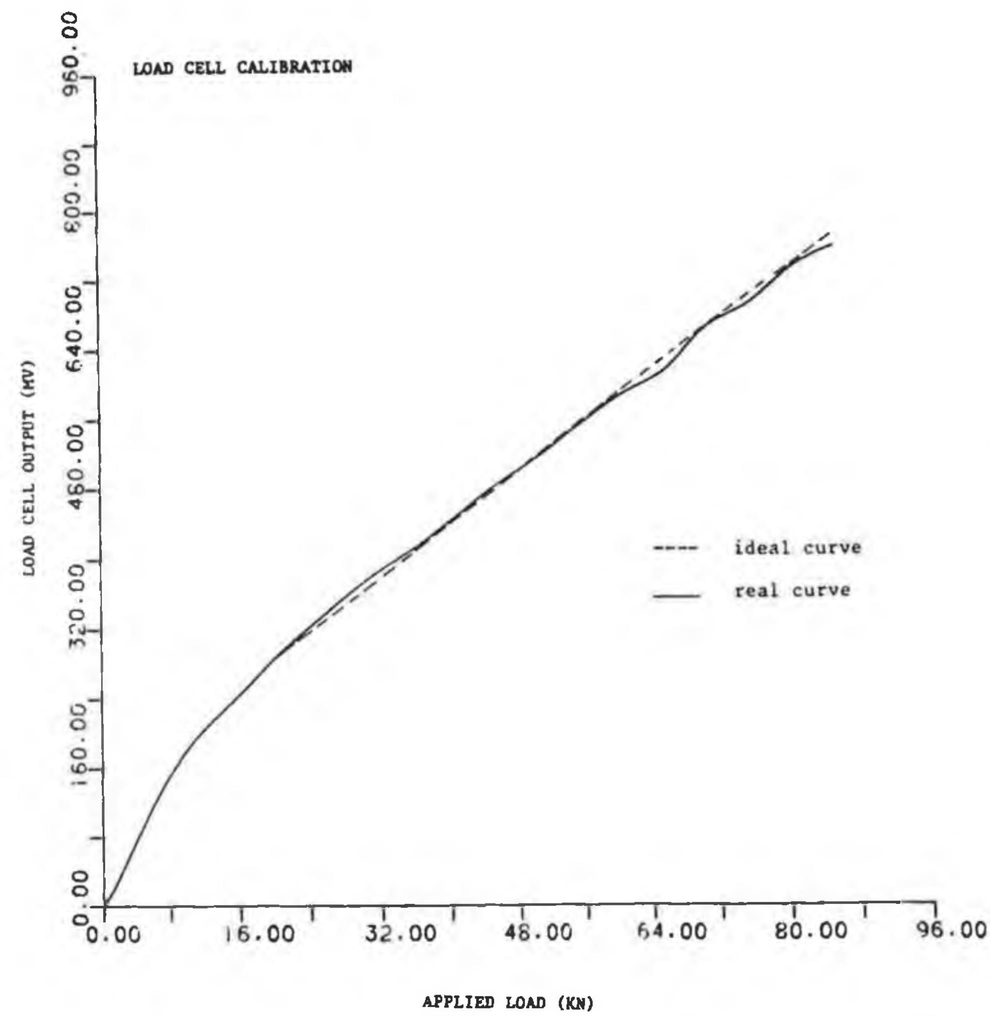

LOAD CELL CALIBRATION
LOAD CELL OUTPUT (MV)
0.00
160.00
320.00
480.00
640.00
800.00
960.00
0.00
16.00
32.00
48.00
64.00
80.00
96.00
APPLIED LOAD (KN)
ideal curve
real curve

FIG.(41)

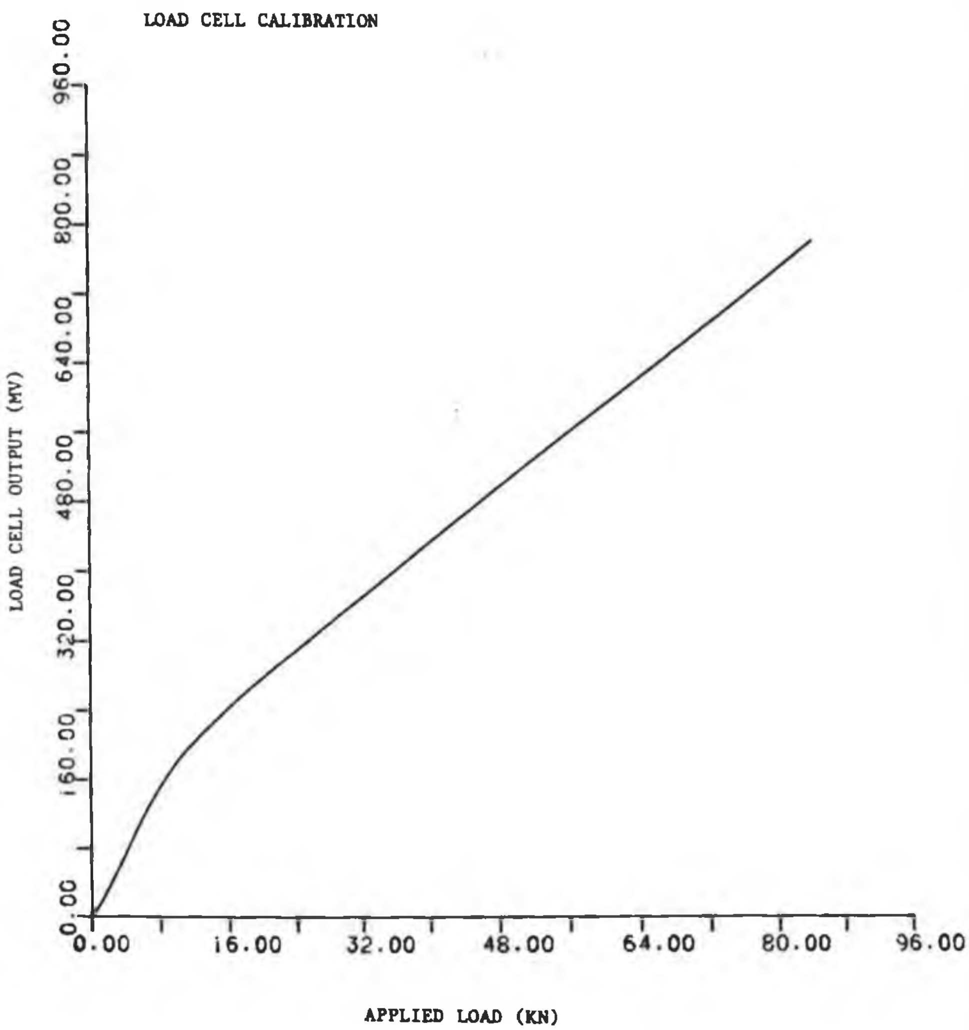
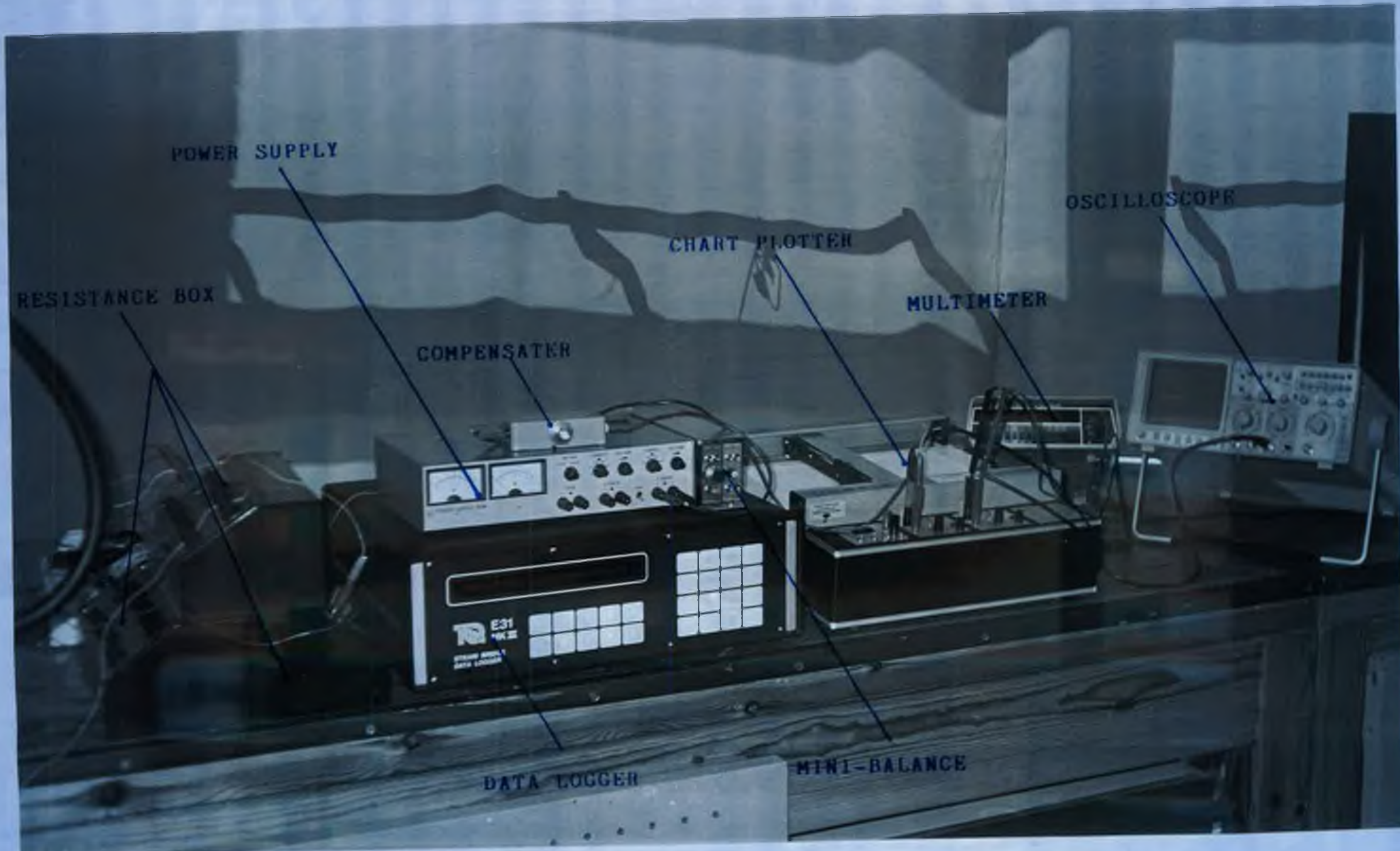
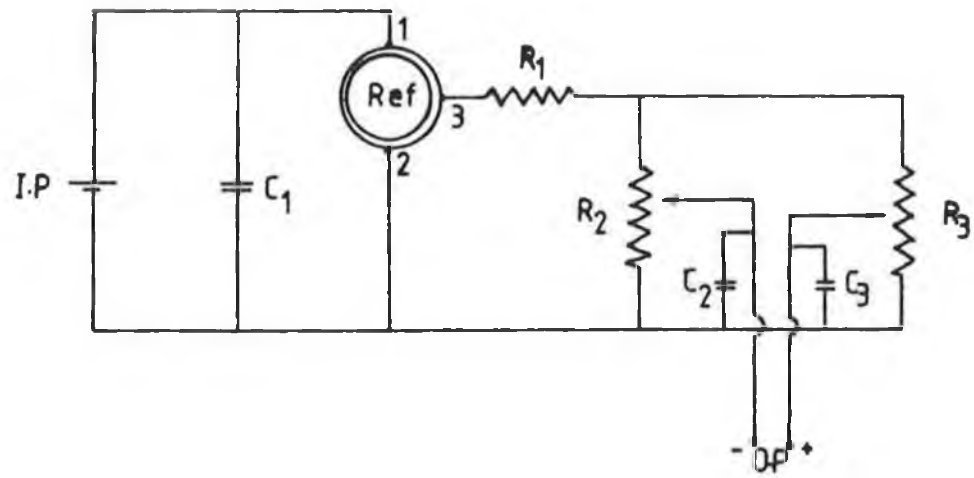

LOAD CELL CALIBRATION
LOAD CELL OUTPUT (MV)
960.00
800.00
640.00
480.00
320.00
160.00
0.00
0.00
16.00
32.00
48.00
64.00
80.00
96.00
APPLIED LOAD (KN)

FIG.(42)



Figure(43)The instrumentation for recording load and displacement



THE ELECTRICAL CIRCUIT OF THE COMPENSATOR

$C_1 = 0.1\ \mu F$

$C_2 = C_3 = 33\ \mu F$

$R_1 = 120\ \Omega$

$R_2 = R_3 = 10\ K\Omega$

Ref: Reference = 10.2 volts when the input
varies between (10 - 40)volts

FIG.(44)

1. between two knife-edges
2. between two rigid parallel surfaces
3. between two rigid parallel surfaces and two lateral walls with an initial gap between the ring and the walls.

Rings were machined from tubes of mild steel. The dimensions of these rings are given in Table (6). The experiment was carried out by using the equipment in Figure (26). The load-deflection and strain-deflection are the results which were compared.

6.2.1 Modes of deformation

6.2.1.1 Between two-knife edges:

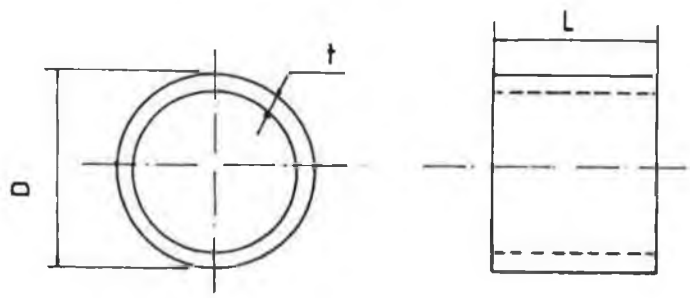
The mode of deformation of the ring between two-knife edges is shown in Figure (45). It is obvious from this Figure that the section of greatest bending moments are at Section A, B,C and D [29]. These moments will continue to increase by increasing the diameter. When the material across the whole of the section at each point has become plastic, yielding occurs and the ring becomes as a mechanism. A photograph of a ring between two-knife edges is shown in Figure (46).

6.2.1.2 Ring between two-rigid parallel surfaces:

A photograph of a compressed ring between two rigid parallel surfaces is shown in Figure (47). DeRuntz et al[30] proceeded to analyse the load deflection relationship on the base of rigid-perfectly plastic theory assuming that the collapse mode consisted of four plastic line hinges as shown in Figure (48(a)). These hinges were assumed to remain stationary relative to the rigid portion of the ring, separation occurring from the outset between the ring and the rigid surfaces in the centre of the contact zone. Others proposed the alternative mode shown in Figure (48(b)) in which the long ring is flattened in the contact zones and conforms to the shape of the plates throughout the loading. The contact zones terminate in hinges **V**, which travel outwards as the deformation proceeds. The horizontal hinges **H** remain stationary.

TABLE (6)

CASE NO.	TEST NO.	RING SIZE (mm)		
		t	D	L
1	1	8	140	60
	2	6	140	60
	3	4	140	60
	4	8	218	60
	5	8	100	60
2	6	8	140	60
	7	6	140	60
	8	4	140	60
	9	8	218	60
	10	8	100	60
3	11	4	140	60



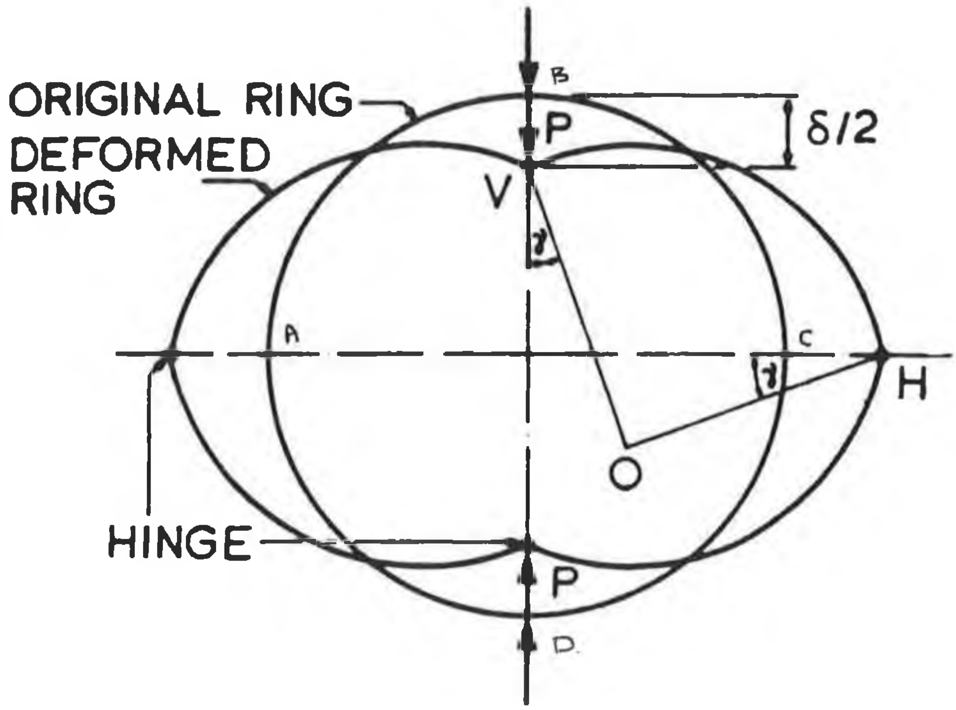
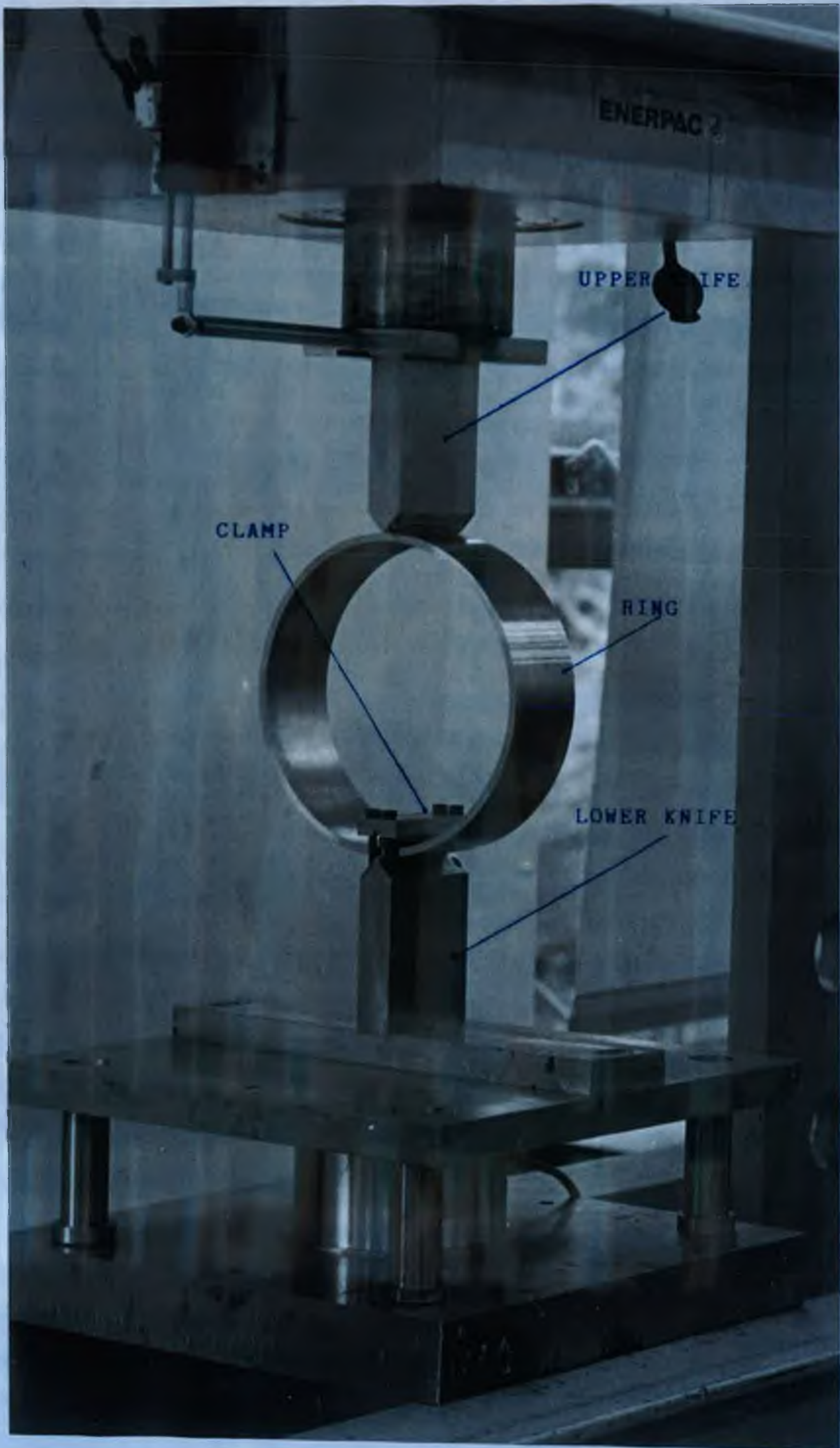
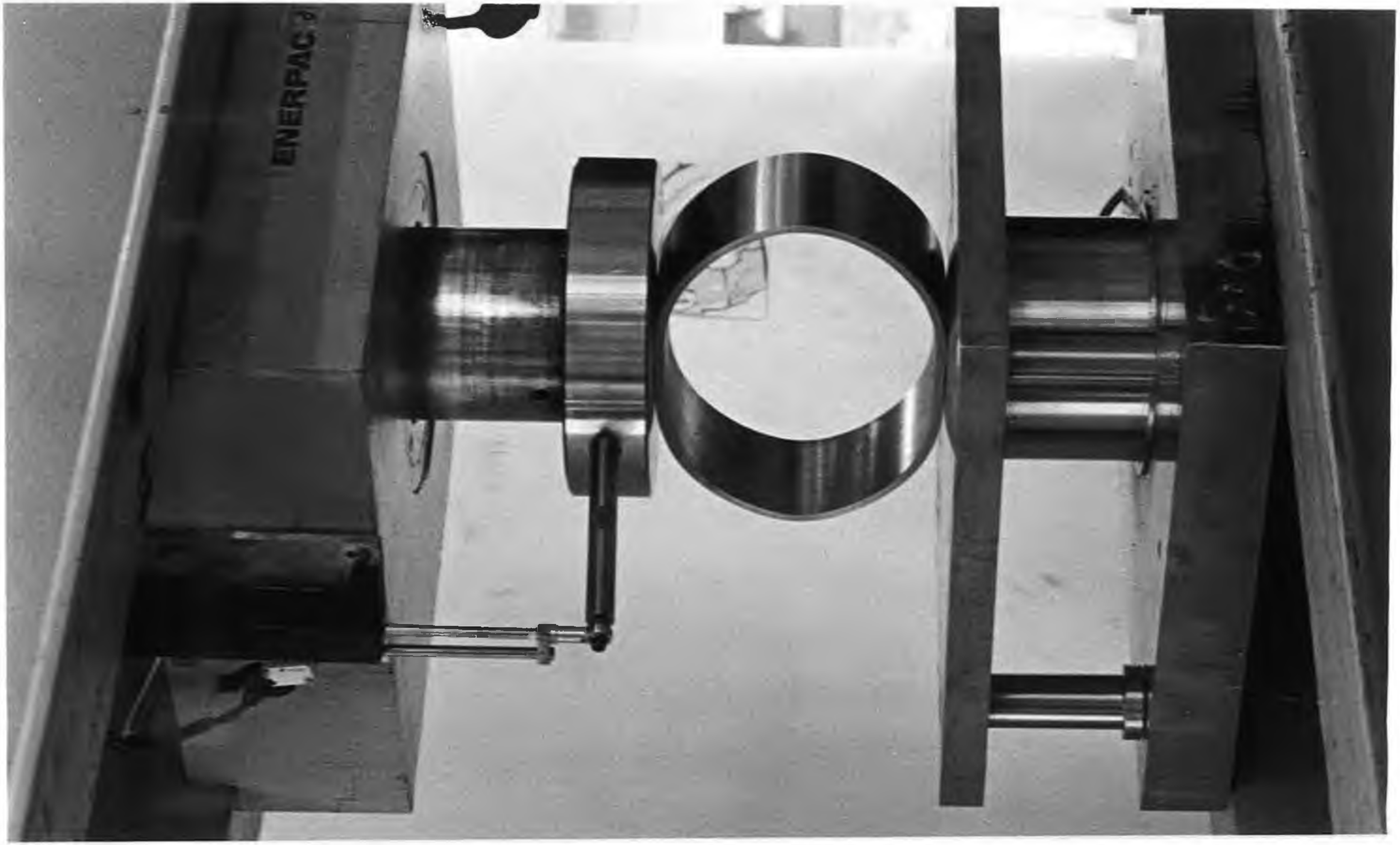


FIG.(45)



Figure(46)A ring between two knife-edges



Figure(47)A ring between two rigid parallel surfaces

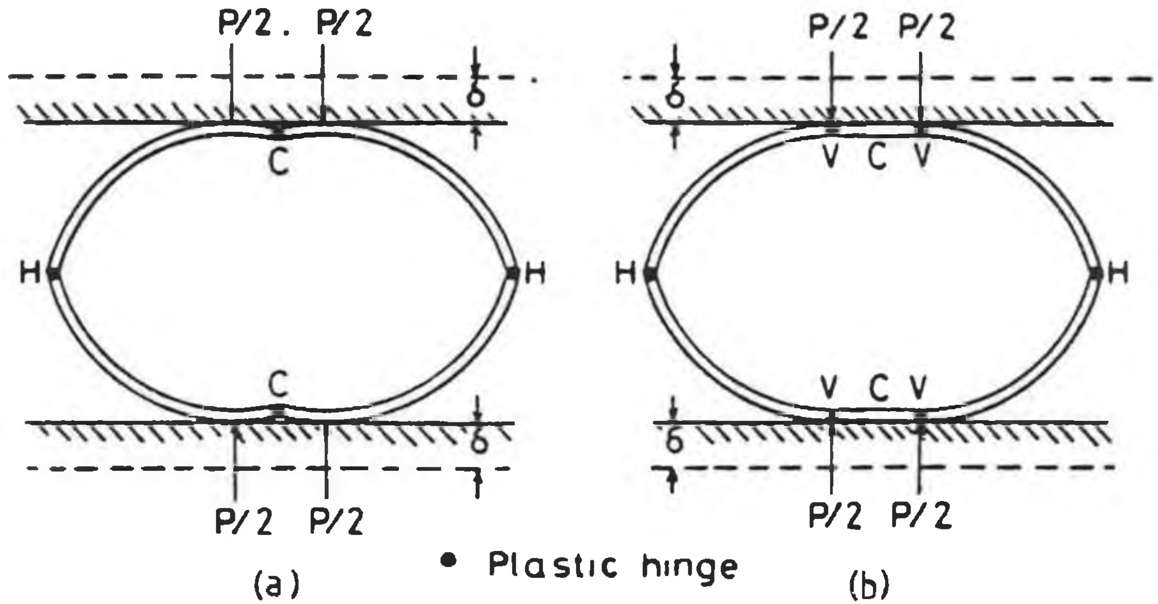


FIG.(48)

Figure 49 shows that increasing the deformation will result in decreasing the arm of the bending moment for two modes of deformation.

6.2.1.3 Ring between two rigid parallel surfaces and two lateral walls with an initial gap between the ring and the walls:

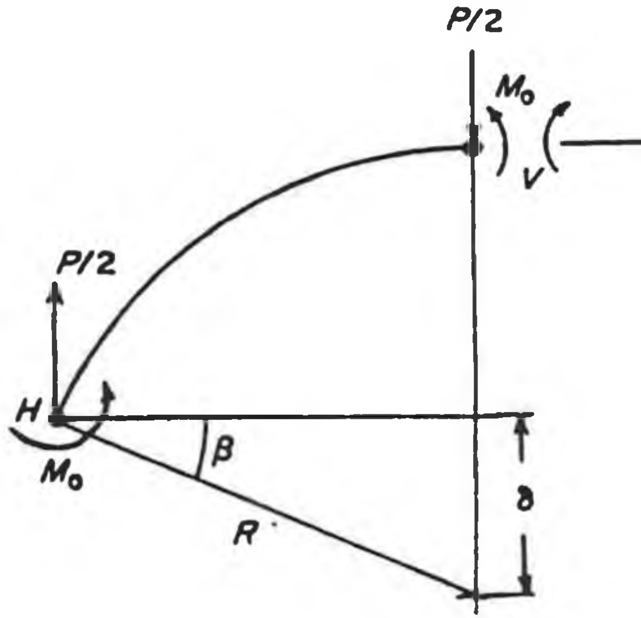
A ring between two lateral walls and two rigid parallel surfaces with an initial gap is shown in Figure (50(a)) and in Figure(50.1). It is obvious that the mode is the same as in the second case (between two rigid parallel surfaces) before the ring came in touch with the walls. After this the mode will change. Reddy et al[31] examined this mode of deformation, starting from the point in which the ring comes in touch with the wall for rigid perfect material.

Figure (50(b)) shows the position of five hinges at the top half of the ring at the point of collapse. The line of action of the resultant force for the left-hand quadrant of the ring is 45 degrees to the horizontal and equidistant from A,E and C.

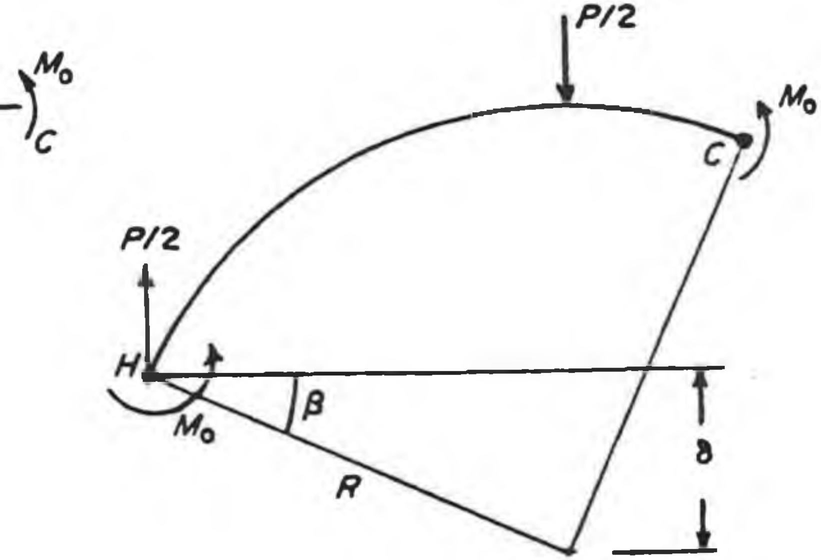
When the load is increased, rotation takes place about the hinges A, E and F, while the hinges at C and D travel through the ring wall which flattens it against the wall, as shown in Figure (50(c)). This mode of deformation persists until the hinges, originally at A and E, reach the same horizontal level at which point the hinge at E begins to travel down through the ring wall, as shown in Figure (50(d)). As can be seen from these diagrams the contact point with the rigid surface moves outwards as the deformation proceeds.

6.3 MATERIAL PROPERTIES

The material properties of the ring structure are approximated as bi-linearly elastic-plastic. These approximated properties were derived from the experimental curve after testing a number of specimens shown in Figure (51). The material properties curve shows the yield point, elasticity modulus and plasticity modulus. A number of assumptions are used in the model:

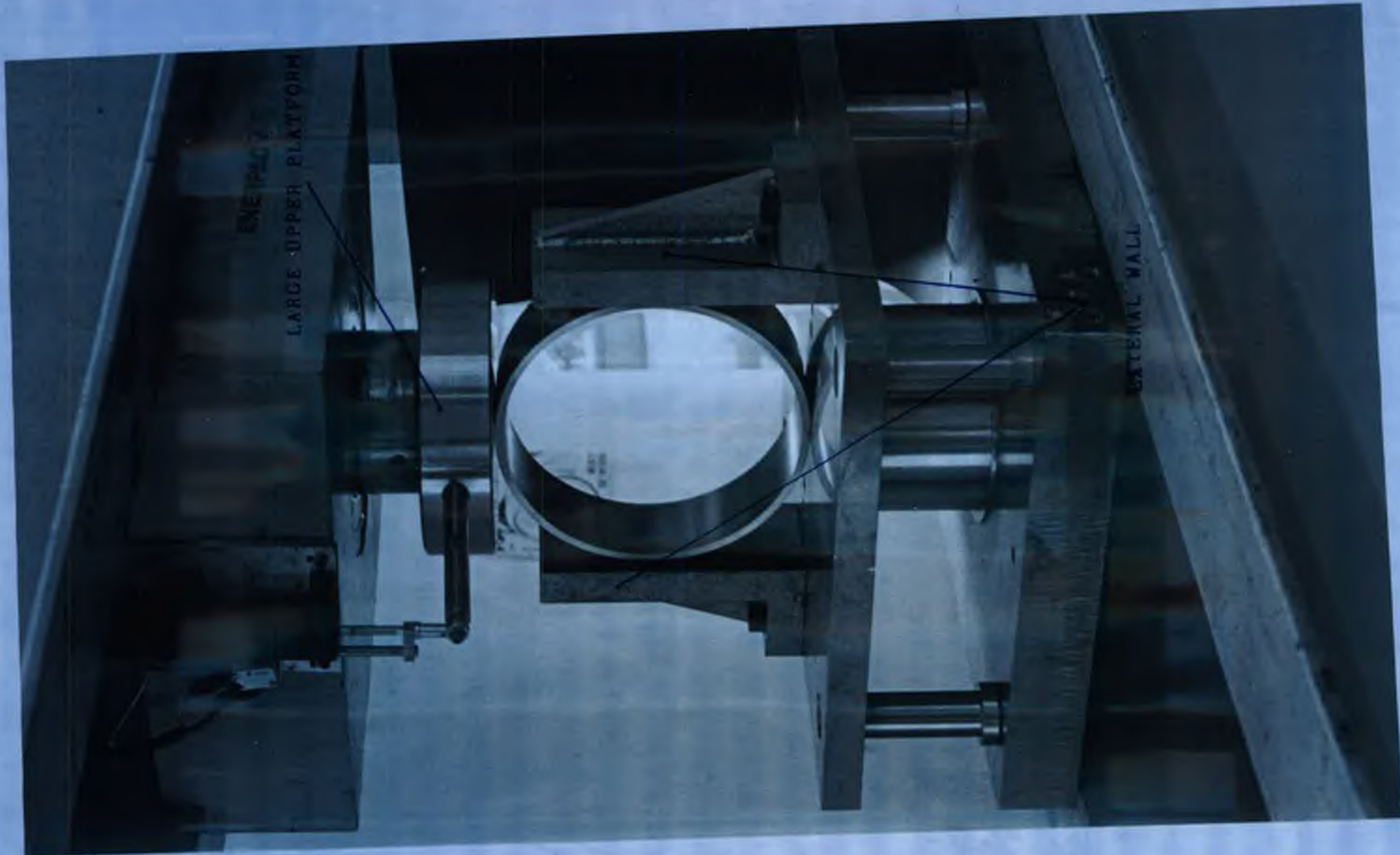


(a)



(b)

FIG.(49)



Figure(50.1) A ring between two rigid parallel surfaces and two lateral walls with an initial gap

(i) The metal is an isoptropic metal;

(ii) The metal has the same properties in tension and compression;

(iii) It is assumed that the friction force between the contact surfaces is zero.

(iv) It is assumed that the model of deformation is two-dimensional (x-y) and so the ring length enters the problem as unit length.

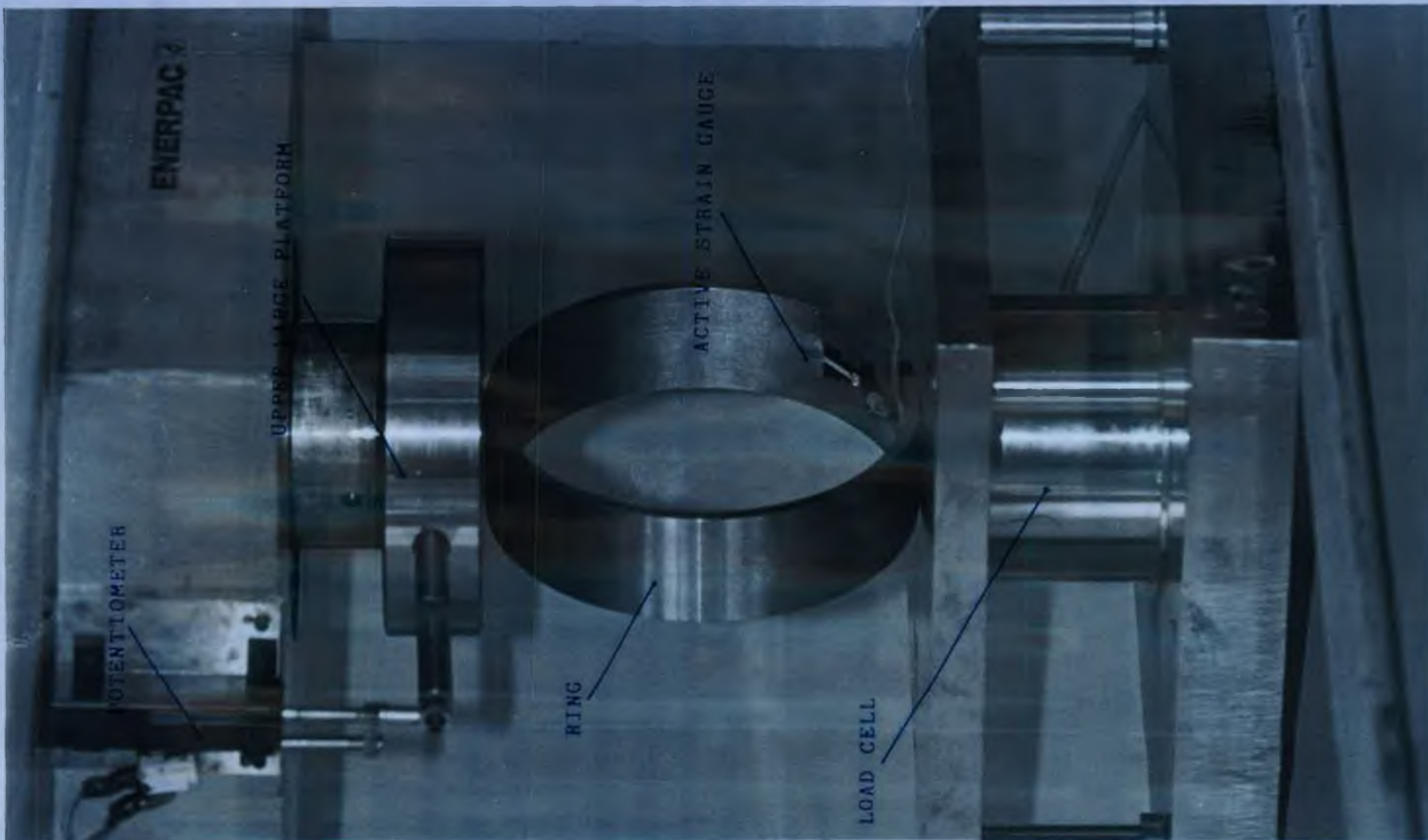
(v) Plane strain conditions are justified when the ratio of the ring length to the wall thickness of the ring exceed 5, i.e. there is no deformation in Z direction.

6.4 STRAIN MEASUREMENTS

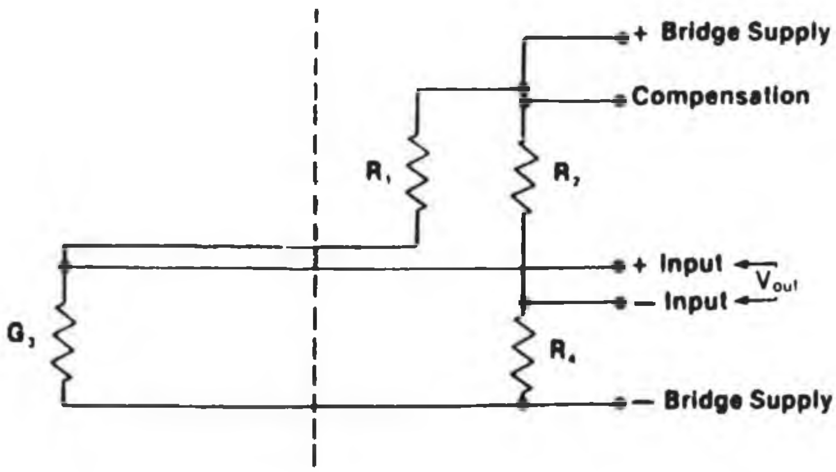
Strain was measured experimentally during the deformation of a ring using electrical resistance strain gauge technique and the measured results were compared with those predicted theoretically. Test number 6 was chosen for measuring the strain on the inner surface of the ring. For the plane strain condition $\epsilon_z = 0$ and γ_{xy} on the surface is also equal to zero. From the components ϵ_x and ϵ_y the normal strain was obtained. Figure (52) shows a photograph illustrating the mounting of the strain gauge on the inner surface of the ring. This strain gauge was linked into a quarter bridge as shown in Figure (53). After converting the output signal from the bridge the strain can be calculated from:

$$\epsilon = \frac{4 \cdot V_{out}}{V \cdot K'} \qquad (6.1)$$

where



Figure(52) The mounting of stain gauge on the ring



R_2, R_4, R_1 dummy gauge of same resistance as G_3.

FIG.(53)

V_{out} - the output signal from the bridge (Volt)

V - the bridge voltage (usually 6V)

K' - the gauge factor (2.1 in this case)

6.5 RESULTS

Results were obtained for different dimensions of a ring structure and different loading cases as shown in Table (6). The material properties which were used in the theory were taken from the stress/strain properties in Figure (51). In each case of loading the ring is symmetrical about **x** and **y** axes and hence, half of the ring was analysed to study the general deformation and to reduce the computer time. The starting mesh of 240 elements and 160 nodes was used for each case.

Prescribed displacements were used here to estimate the load/deflection or the strain/deflection relationship. The size of the increment was 0.7 of the yield load and in total about 350 increments were applied to reach 50-75 mm of deflection at the point which the load is applied. The following graphs are presented to illustrate the results obtained from the theory and experimental work:

Figure (54): Graph of applied load on the ring against deflection of loaded point. This was obtained from the experiment for three different wall thicknesses of the ring with constant lengths and outer diameters for knife-edge loads.

Figure (55): Illustrates the load/deflection data of the loaded point. This was obtained from the experiment for different outer diameters with constant wall thickness and length of the ring for knife-edge loads.

Figure (56): Shows the load/deflection relationship which was obtained from the experiment and the theory for a loaded ring between two-knife edges.

Figure (57): Shows a photograph of a ring after being squashed between two knife-edges during the experiment.

Figure (58): Graphical output from the E.P. Program after loading the ring between two knife-edges.

Figure (59): Illustrates the relationships between load and deflection of the loaded point. The data in this figure was obtained from the experiment for the loaded ring between two rigid parallel surfaces for constant outer diameters and lengths with different wall thicknesses of the ring.

Figure (60): Presents also the experimental load/deflection relationship data for the loaded point of the ring between two rigid parallel surfaces for different outer diameters with constant lengths and wall thicknesses of the ring.

Figure (61): Shows the load against deflection of the loaded point of a ring between two rigid parallel surfaces. The data was obtained from experimental work and the theory.

Figure (62): Shows a photograph of a ring from the experiment after being squashed between two rigid parallel surfaces.

Figure (63): Graphical output from the E.P. Program after loading the ring between two rigid parallel surfaces.

Figure (64): Shows the strain on the inner surface of the ring (at 90° upwards of the axes which passes the centre of the ring) against the deflection of loaded points. Date was obtained from the experimental work and theory.

Figure (65): Shows the experimental load/deflection relationship for the loaded point of a ring between two rigid parallel surfaces and two lateral walls for a given dimension of the ring.

Figure (66): Shows also the load against deflection of the loaded point of a ring between two rigid parallel surfaces and two lateral walls. Data was obtained from experimental work and the theory.

Figure (67): Shows a photograph of a ring from the experimental work after being squashed between two rigid parallel surfaces and two lateral walls.

Figure (68): Graphical output from the E.P. Program after loading the ring between two rigid parallel surface and two lateral walls.

6.6 RESULTS AND DISCUSSIONS

The results presented in the previous Section were obtained from different loading cases of the ring as given in Table (6).

6.6.1 Ring between two knife-edges:

According to other researchers the results of squashing the ring between two knife-edges suggests that the load increased over a certain deflection range and then fell by increasing the deflection. This was for the rigid perfectly plastic material, whereas the strain hardening material predicted that the load increased with increasing deflection. In both theories the moment arm of applied load about the hinge on the horizontal diameter of the ring increased with deflection, therefore the material strain hardening properties play a significant role in defining the shape of the load/deflection curve[32]. The experimental results from the present study suggest that the load increases for up to a certain deflection and then remained fairly constant for loading of ring between two knife-edges as shown in Figure (54). This Figure shows load-deflection curves for rings of three different wall thicknesses.

Figure (55) shows the effect of changing the outer diameter of the ring on the shape of the load/deflection curve. In this Figure the experimental observation was made on test number 5 which shows that the load increased for a certain deflection and then remainded constant to certain level of deflection and after that increased again sharply.

Figure (56) shows a comparison between the load/deflection curve obtained from the experiment and the theory. The discrepancy between the two curves refers perhaps to:

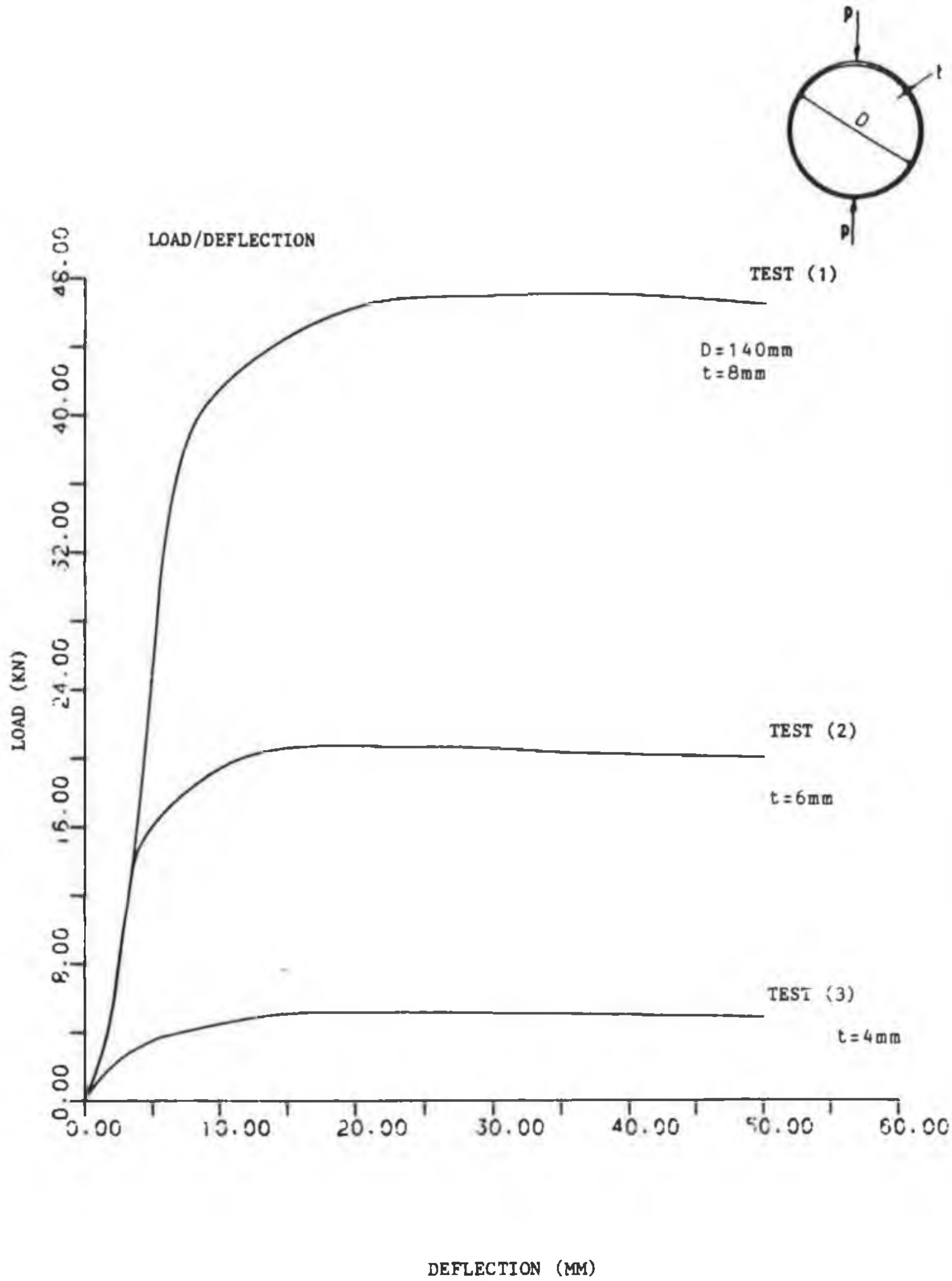
(i) the load cells: When the load is less than about 16 KN, it is obvious that from the load cell calibration curve in Figure (42) the reading of the load is not sufficiently accurate because of the non-linearity of the load cell in this region.

(ii) the experimental model: This means theoretically it was assumed that the material was annealed, whereas in the experiment the rings were machined and hence became work hardened. These rings were tested without having been annealed. Thus some error will result.

(iii) inaccurate simulation of the theory: The ring was mounted between the knives as shown in Figure (46) and also a line marking was made on the surface of the ring where it touched the knives. There was very slight friction during the process of deformation, whereas the theory ignored it.

(iv) the lack of accuracy in the vertical alignment of the loading piston of the press: This means the centre line of the hydraulic piston is not exactly perpendicular to the bed of the press. Therefore, during the deformtion process the centre line of the upper knife may travel in a different plane to the centre line of the lower knife. This may have caused an non-uniform distribution of load on the load cell and hence may have given inaccurate reading.

Figures (57) and (58) show the results of deforming a ring between two knife-edges. These results were obtained from the experiemental work and the computer. Subsequently, Figure (58) shows a very slight difference between the deformation of the upper half of the ring and the lower half. This is perhaps because the mesh generation in both halves are not perfectly symmetrical.



LOAD/DEFLECTION
TEST (1)
D=140mm
t=8mm
TEST (2)
t=6mm
TEST (3)
t=4mm
LOAD (KN)
DEFLECTION (MM)

FIG.(54)

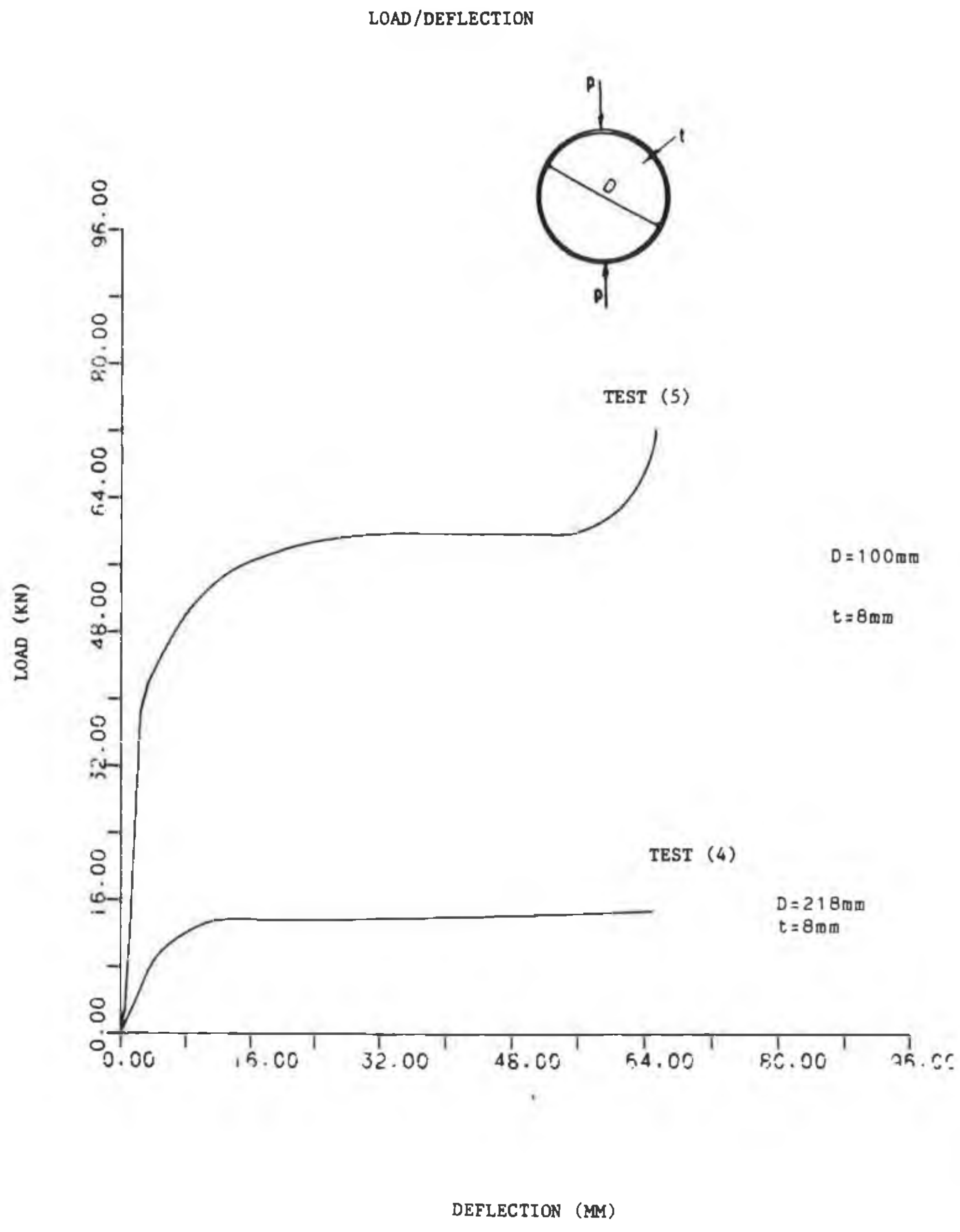
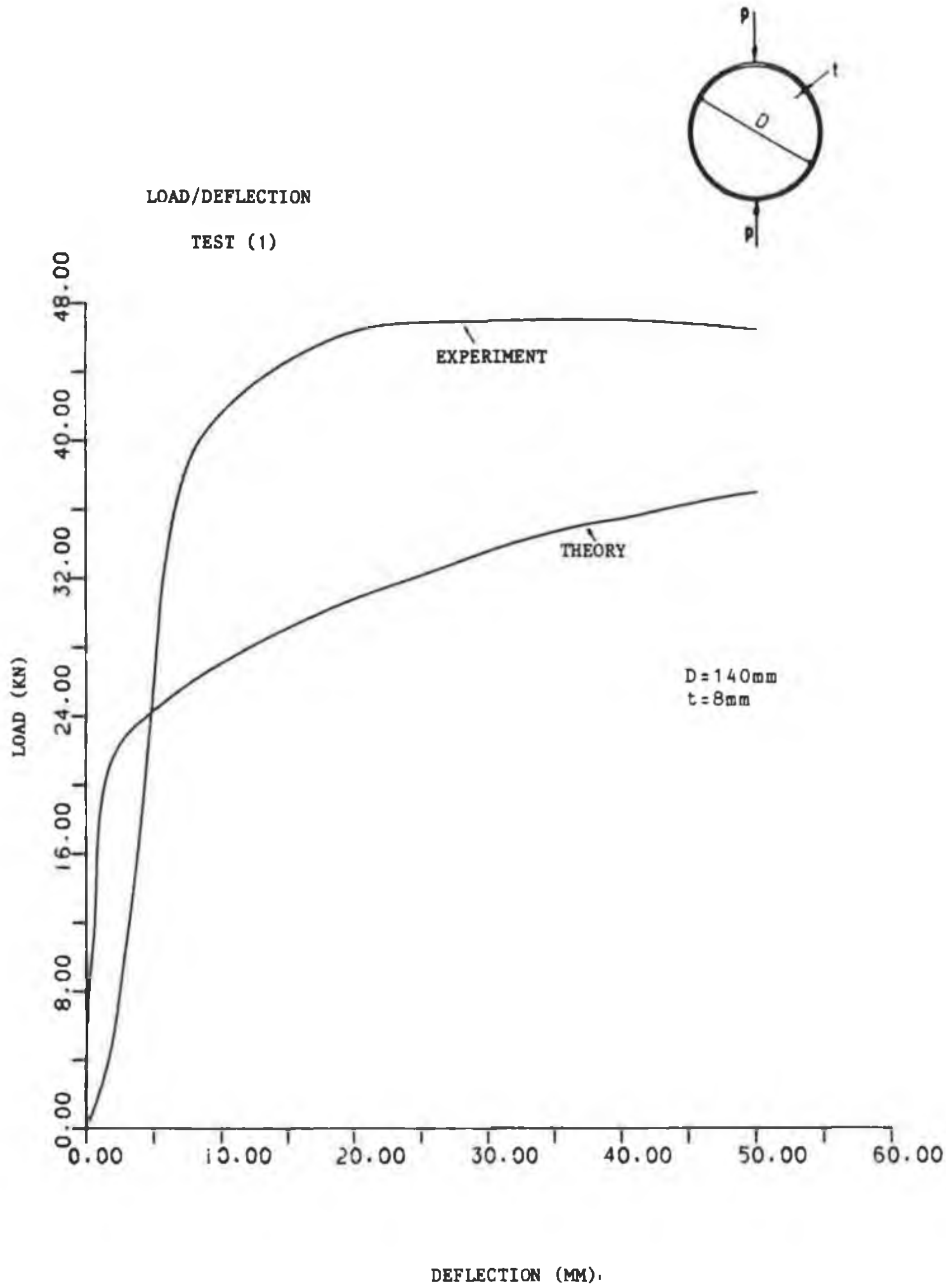

LOAD/DEFLECTION
P
t
D
P
TEST (5)
D=100mm
t=8mm
TEST (4)
D=218mm
t=8mm
LOAD (KN)
0.00
16.00
32.00
48.00
64.00
80.00
96.00
DEFLECTION (MM)

FIG.(55)



P
t
D
P
LOAD/DEFLECTION
TEST (1)
EXPERIMENT
THEORY
D=140mm
t=8mm
LOAD (KN)
0.00
8.00
16.00
24.00
32.00
40.00
48.00
0.00
10.00
20.00
30.00
40.00
50.00
60.00
DEFLECTION (MM).

FIG.(56)

Figure(57)The deformation of a ring between two knife-edges

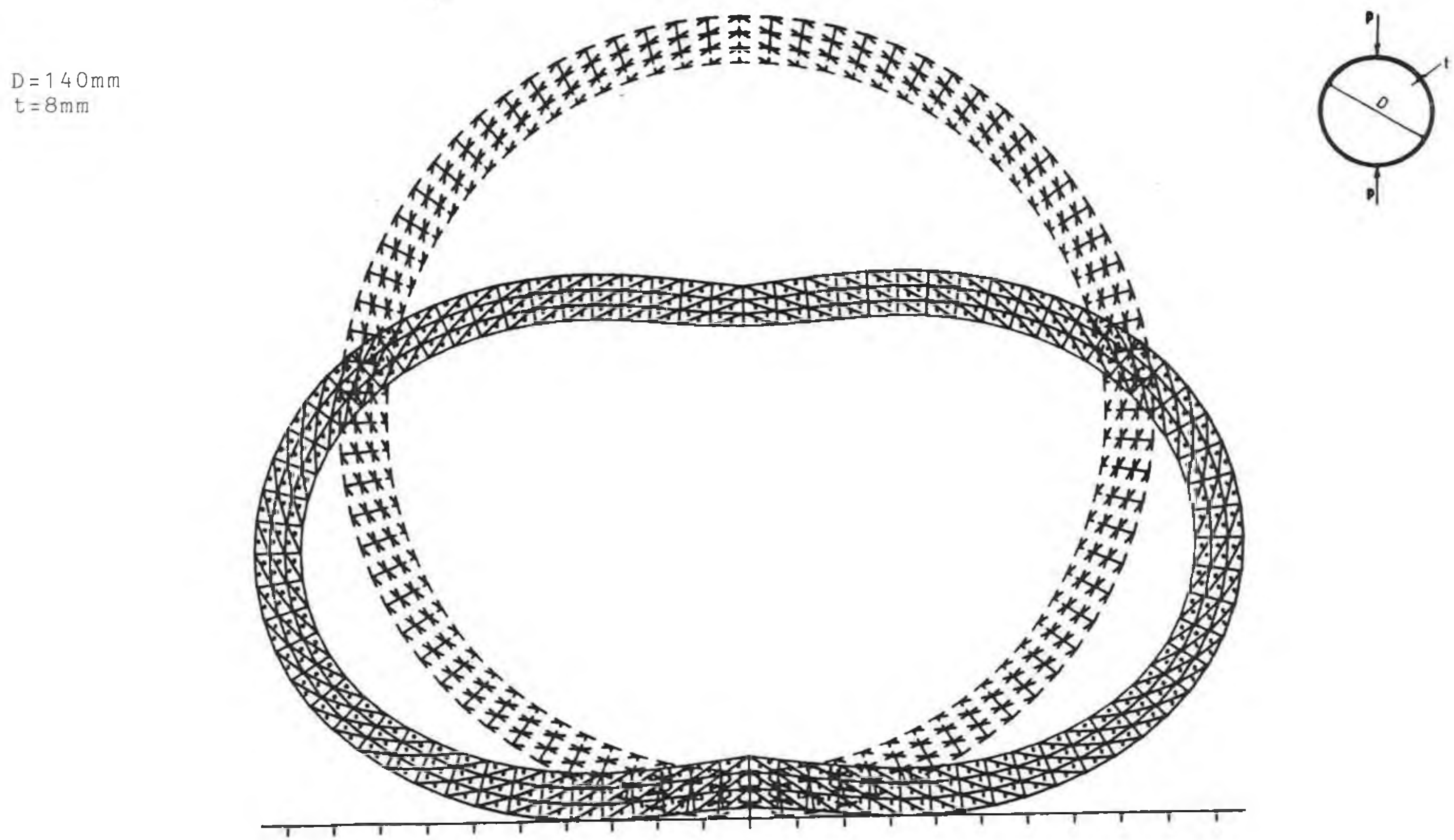


FIG.(58)

6.6.2 Ring between two rigid parallel surfaces:

The experimental results predicted in Figure (59) shows the shape of the load/deflection curve of rings under loading between two rigid parallel surfaces, with different wall thickness dimension. It is obvious that the load will increase sharply with increasing deflection of between 5-10 mm and then slowly for deflection for higher than these. This means, physically, during the deformation process the contact lines with the rigid surfaces split and move outwards as the deformation continues. Therefore, the moment arm of the applied load about the hinge on the horizontal diameter of a ring will decrease. This requires an increase in the applied load to maintain the deformation. The influence of the wall thickness on the load/deflection curve is clearly illustrated in this Figure. In Figure (60) the effect of the outer diameter of the rings on the load/deflection curve is illustrated.

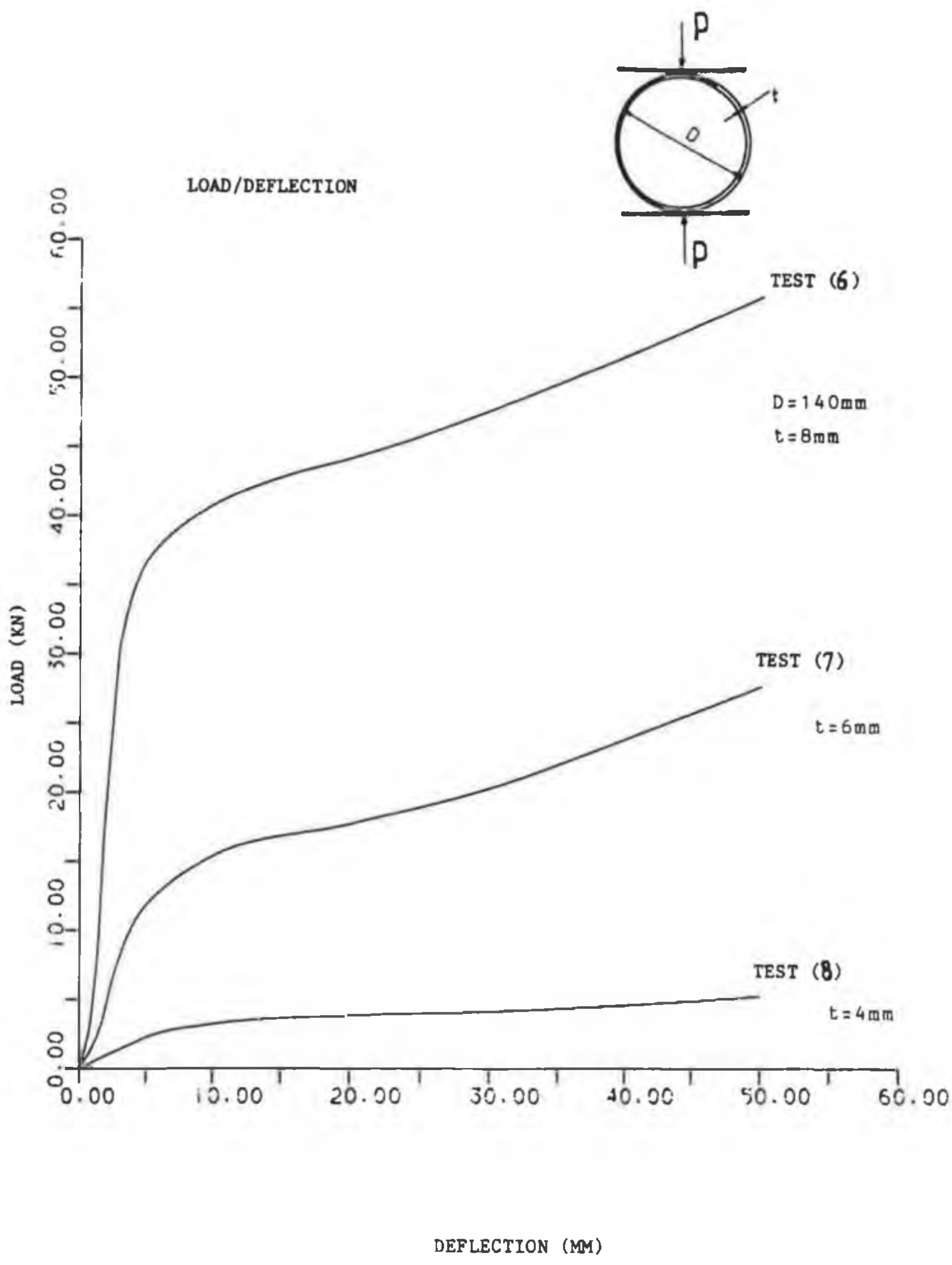
Figure (61) shows the theoretical load/deflection curve and the experimental one. It is obvious that the theoretical curve is in reasonable agreement with the trend of the experimental one and the theoretical model is perhaps more acceptable than the one between the two knife-edges. The discrepancy between the theoretical and experimental results, as shown in Figure (61), may be attributed to the friction between the material surfaces in the experiment, which has not been taken into account in the theory, and to the inaccuracy of the theoretical models. The discrepancy could also be from the reasons, previously mentioned in Section (6.6.2).

Figure (62) shows a photograph of a ring which has been deformed between two rigid parallel surfaces, whereas Figure (63) shows the one predicted from the theory. The latter shows reasonable agreement of simulation of the experiment. Also, it shows a very slight difference between the deformation of the upper half of the ring and the lower half. Once again, this slight difference is attributed to non-symmetrical mesh configuration.

Figure (64) shows typical results from the experimental strain measurements on the inner surface of the ring loaded between two rigid parallel surfaces and the strain predicted from the theory at point S. It can be seen from both curves in Figure (64) that the discrepancy in strain between the experimental and theoretical results is poor for increasing deformation. According to the experimental observation, the strain gauge became detached from the surface during the process. The agreement between the measured and predicted strains are good in the elastic range, although, as seen in Figure (64), the measurement strains are lower than the predicted ones. The disagreement increasing with deflection. This can perhaps be attributed to the fact that the measured strains are averaged over the length of the gauge. Theoretically, strains are constant over the element.

6.6.3 Ring between rigid parallel surfaces and two lateral walls with an initial gap between the surfaces:

Figure (65) shows the experimental load/deflection curve of a ring being deformed between two rigid parallel surfaces and two lateral walls with an initial gap. The results from the theory and experiment for this case are shown in Figure (66). The load levels when the ring touches the walls, according to the experimental observation and theoretical prediction, are denoted by L_e and L_t respectively. In the theory the lower half of the ring is a mirror image of the upper half during and after the loading, as shown in Figure (68), because the friction forces were ignored. Whereas the friction forces at the contact lines with the walls causes the upper half to deform more than the lower half, as shown in Figure (67), the load level at which the lower half begins to deform, according to the experimental observation, is denoted by L_{ep}. It is obvious from Figure (67) that the experiment gives the collapse of the ring before the theory and also that the theoretical model is more acceptable. The results from the theory are in good agreement with the experimental results.



LOAD/DEFLECTION
P
t
D
P
TEST (6)
D=140mm
t=8mm
TEST (7)
t=6mm
TEST (8)
t=4mm
LOAD (KN)
0.00
10.00
20.00
30.00
40.00
50.00
60.00
DEFLECTION (MM)

FIG.(59)

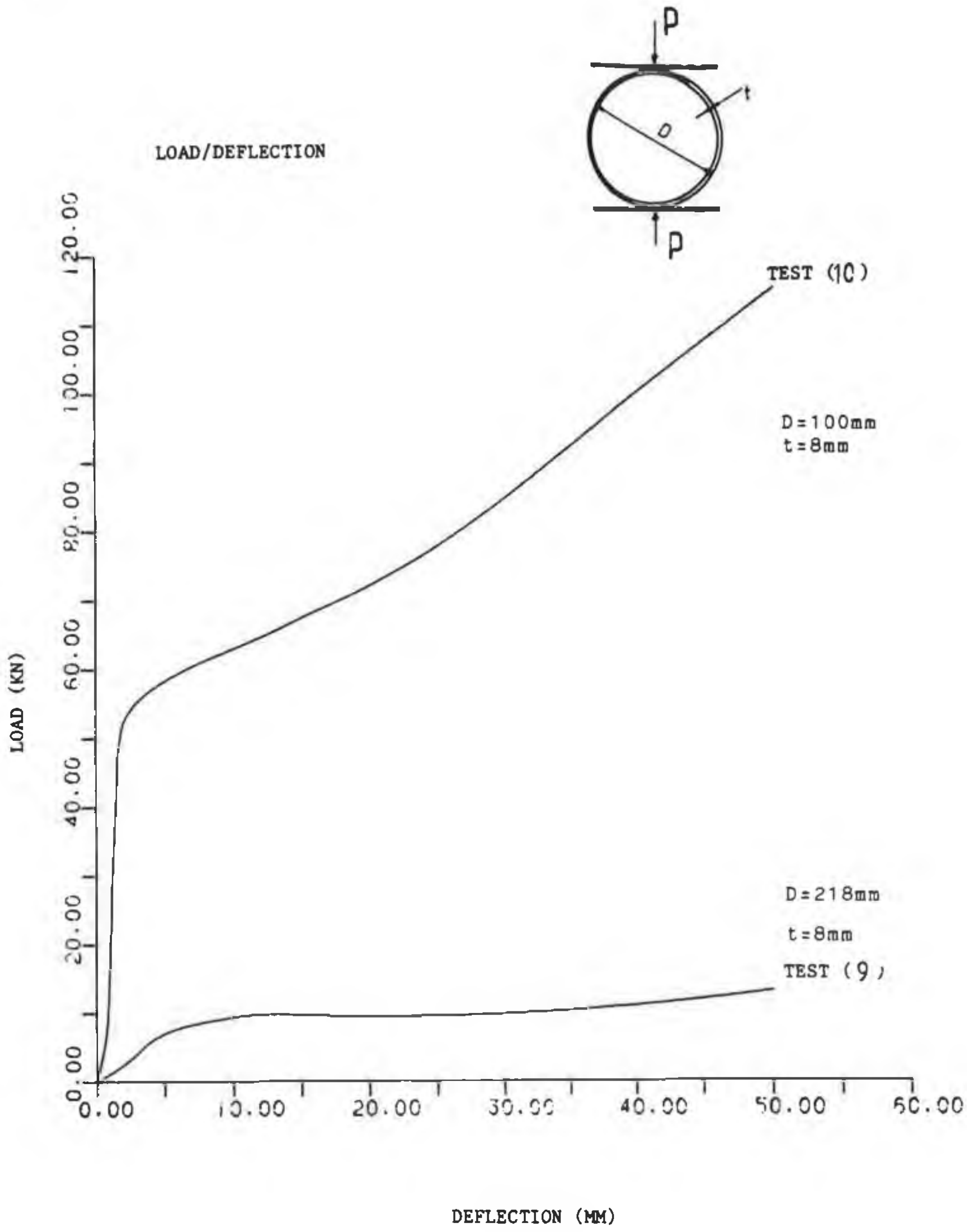

LOAD/DEFLECTION
P
t
D
P
TEST (10)
D=100mm
t=8mm
D=218mm
t=8mm
TEST (9)
LOAD (KN)
0.00
20.00
40.00
60.00
80.00
100.00
120.00
0.00
10.00
20.00
30.00
40.00
50.00
60.00
DEFLECTION (MM)

FIG.(60)

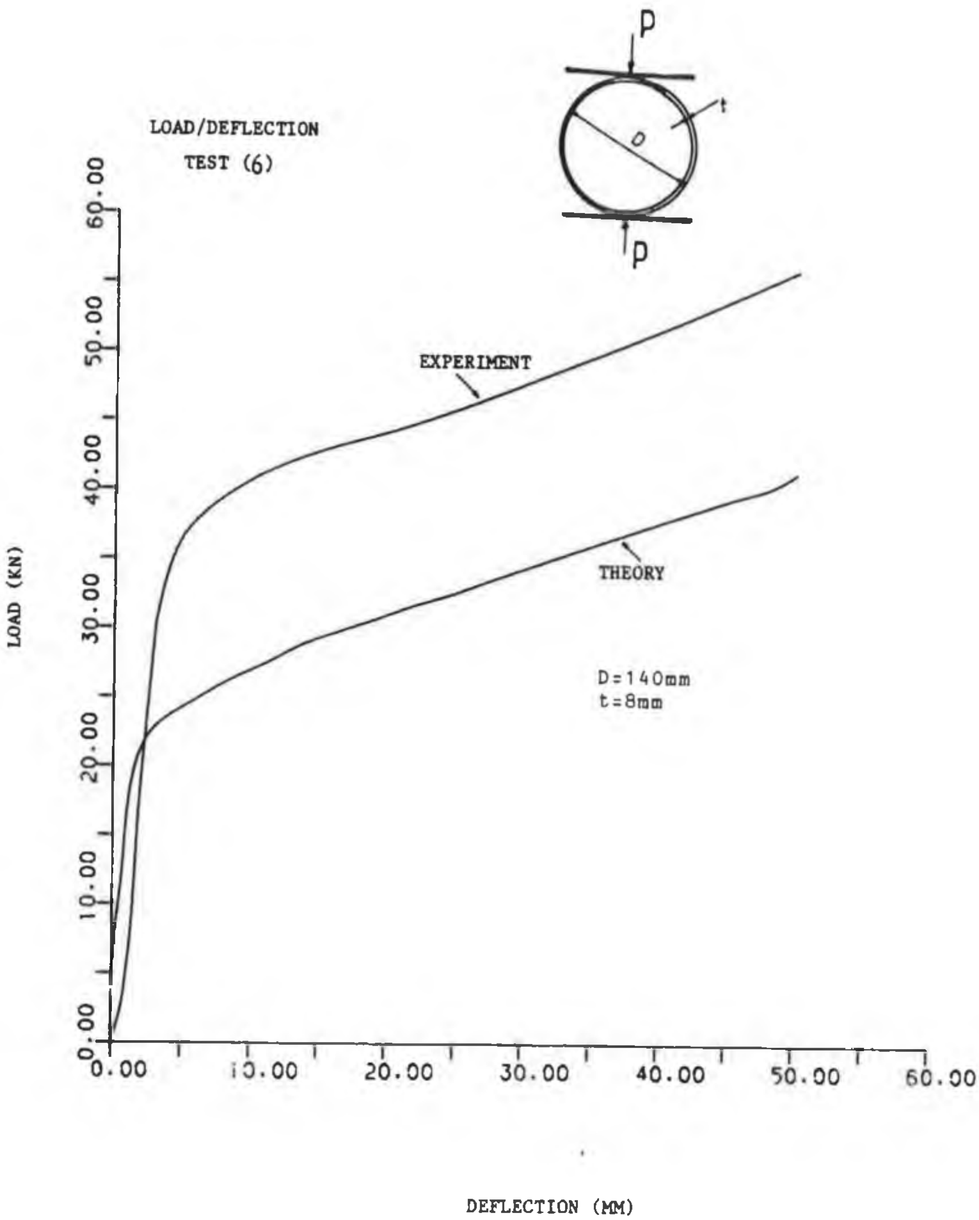
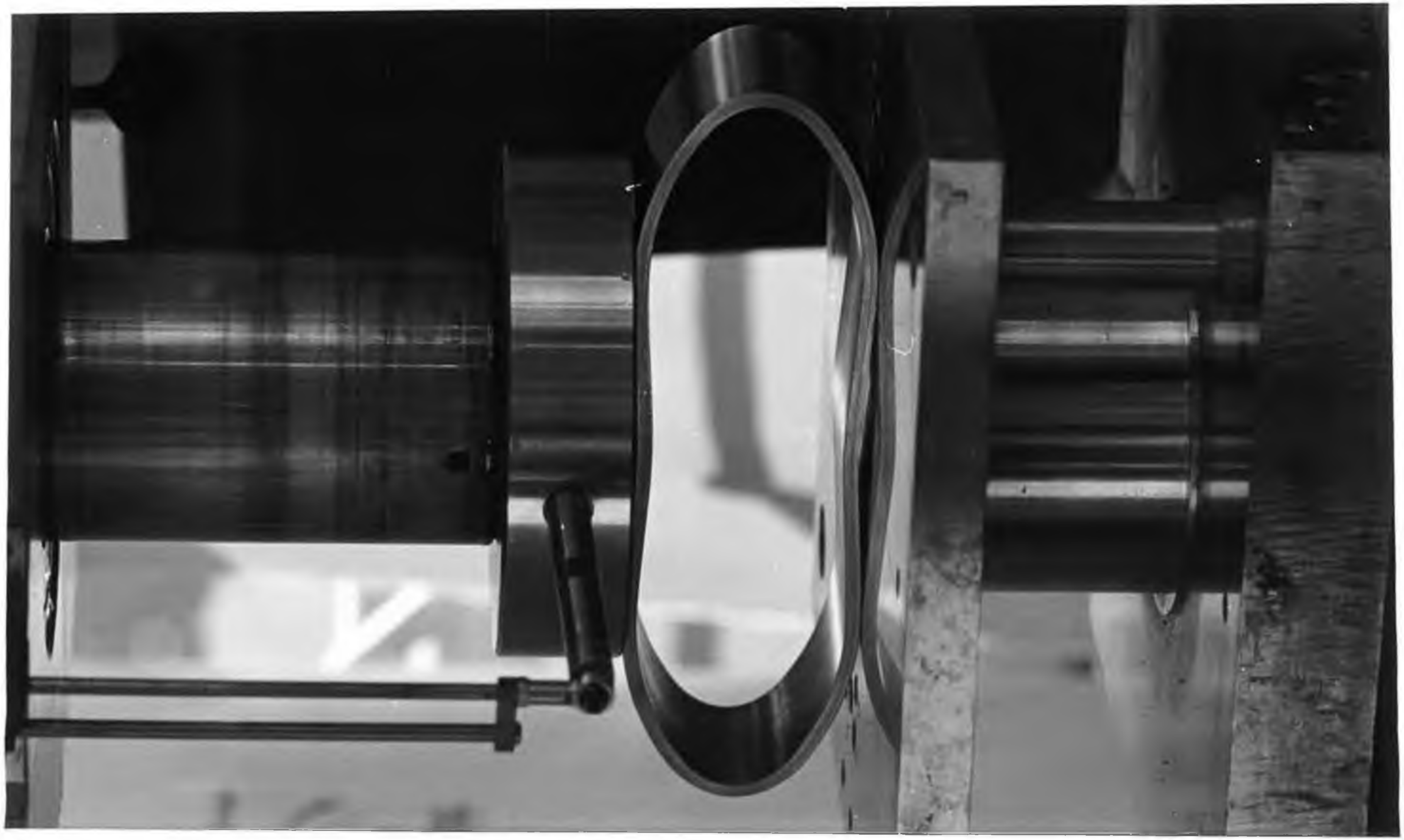

LOAD/DEFLECTION
TEST (6)
P
t
D
P
EXPERIMENT
THEORY
D=140mm
t=8mm
LOAD (KN)
0.00
10.00
20.00
30.00
40.00
50.00
60.00
0.00
10.00
20.00
30.00
40.00
50.00
60.00
DEFLECTION (MM)

FIG.(61)



Figure(62)The deformation of a ring between two rigid parallel surfaces

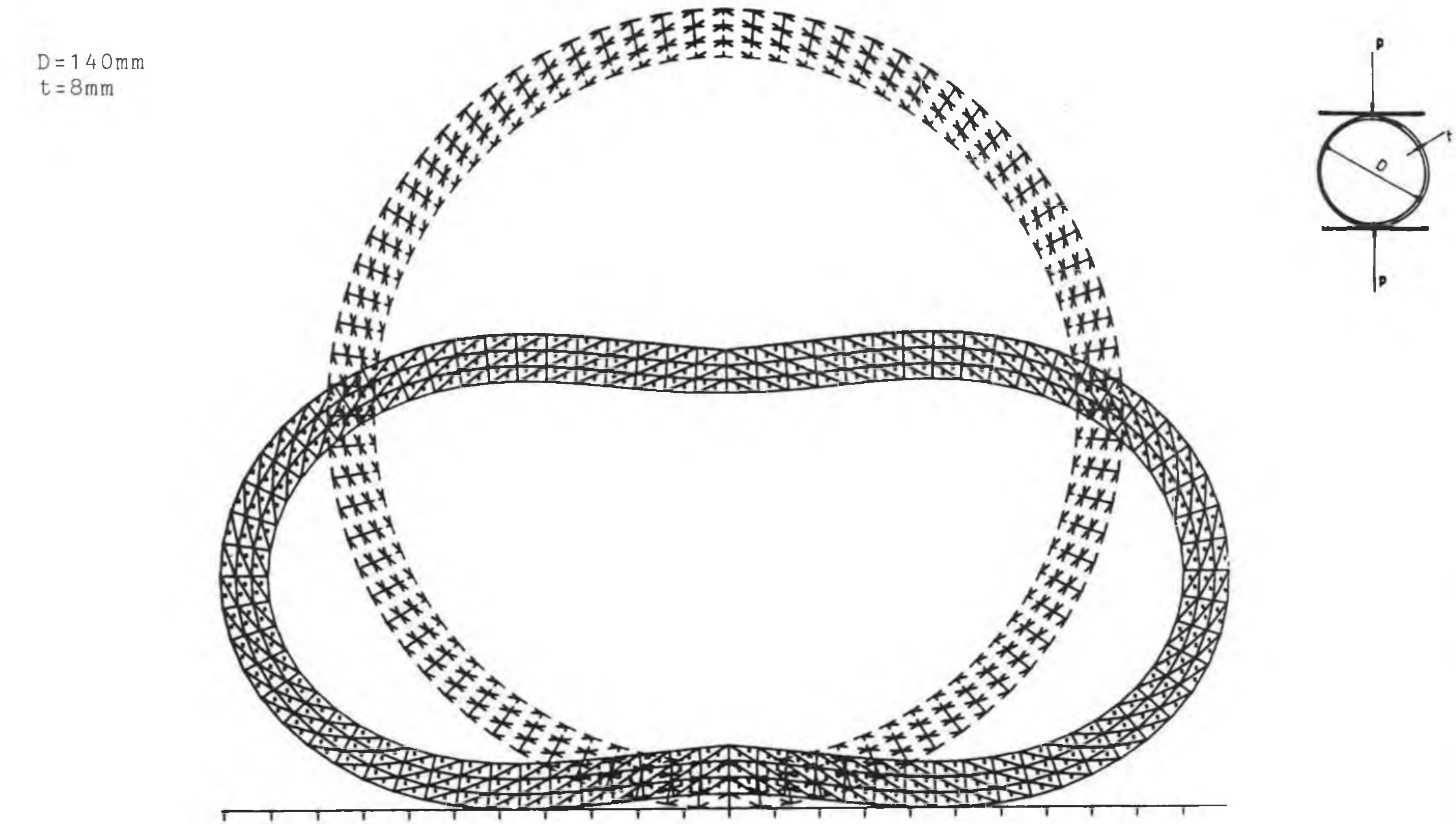


FIG.(63)

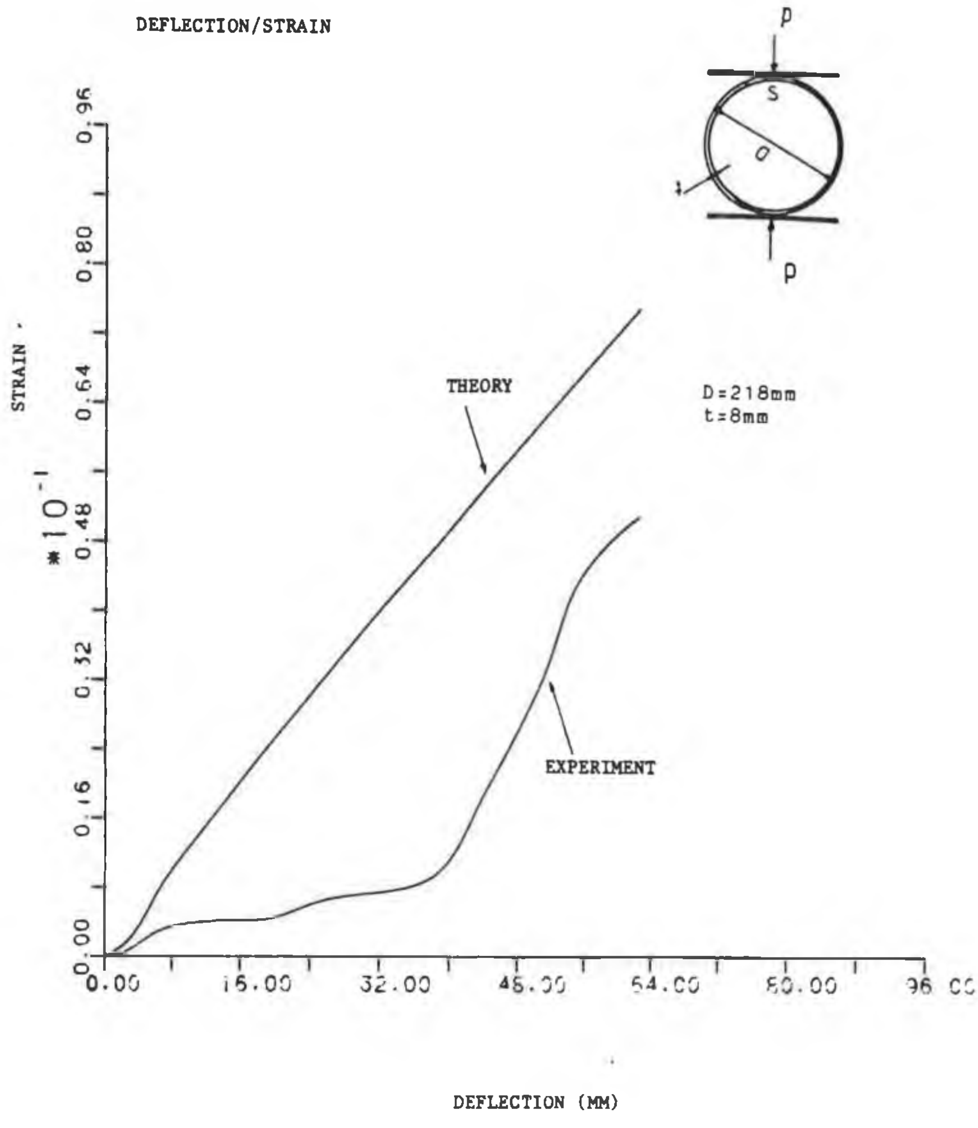
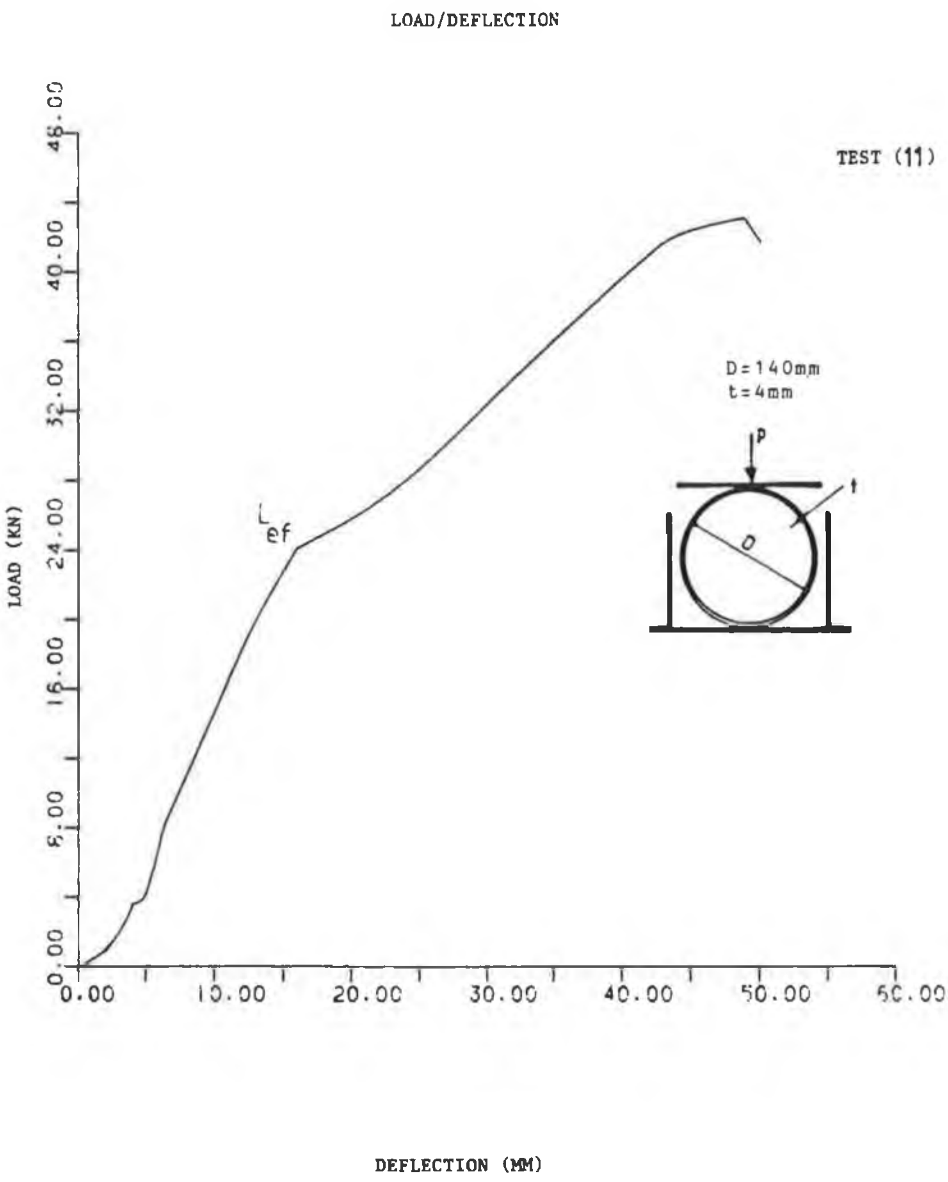
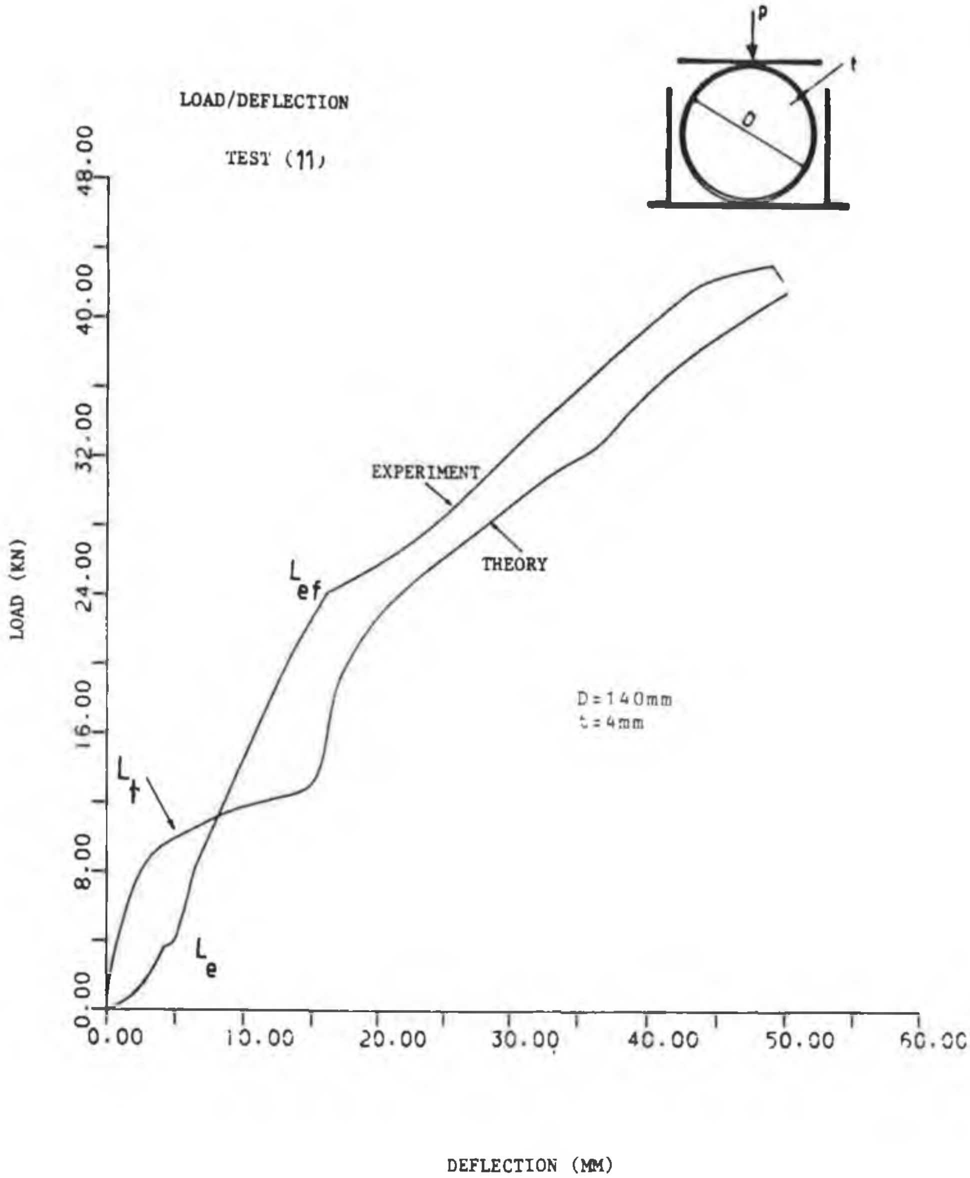

DEFLECTION/STRAIN
STRAIN
*10⁻¹
0.00
0.16
0.32
0.48
0.64
0.80
0.96
0.00
16.00
32.00
48.00
64.00
80.00
96.00
DEFLECTION (MM)
THEORY
EXPERIMENT
D=218mm
t=8mm
P
S
D
t
P

FIG.(64)



LOAD/DEFLECTION
TEST (11)
LOAD (KN)
0.00
8.00
16.00
24.00
32.00
40.00
48.00
0.00
10.00
20.00
30.00
40.00
50.00
60.00
DEFLECTION (MM)
L_{ef}
D=140mm
t=4mm
P
t
D

FIG.(65)



LOAD/DEFLECTION
TEST (11)
P
t
D
LOAD (KN)
EXPERIMENT
THEORY
L_{ef}
L_t
L_e
D=140mm
t=4mm
0.00
8.00
16.00
24.00
32.00
40.00
48.00
0.00
10.00
20.00
30.00
40.00
50.00
60.00
DEFLECTION (MM)

FIG.(66)

Figure(67)The deformation of a ring between two rigid parallel surfaces and two lateral walls with an initial gap

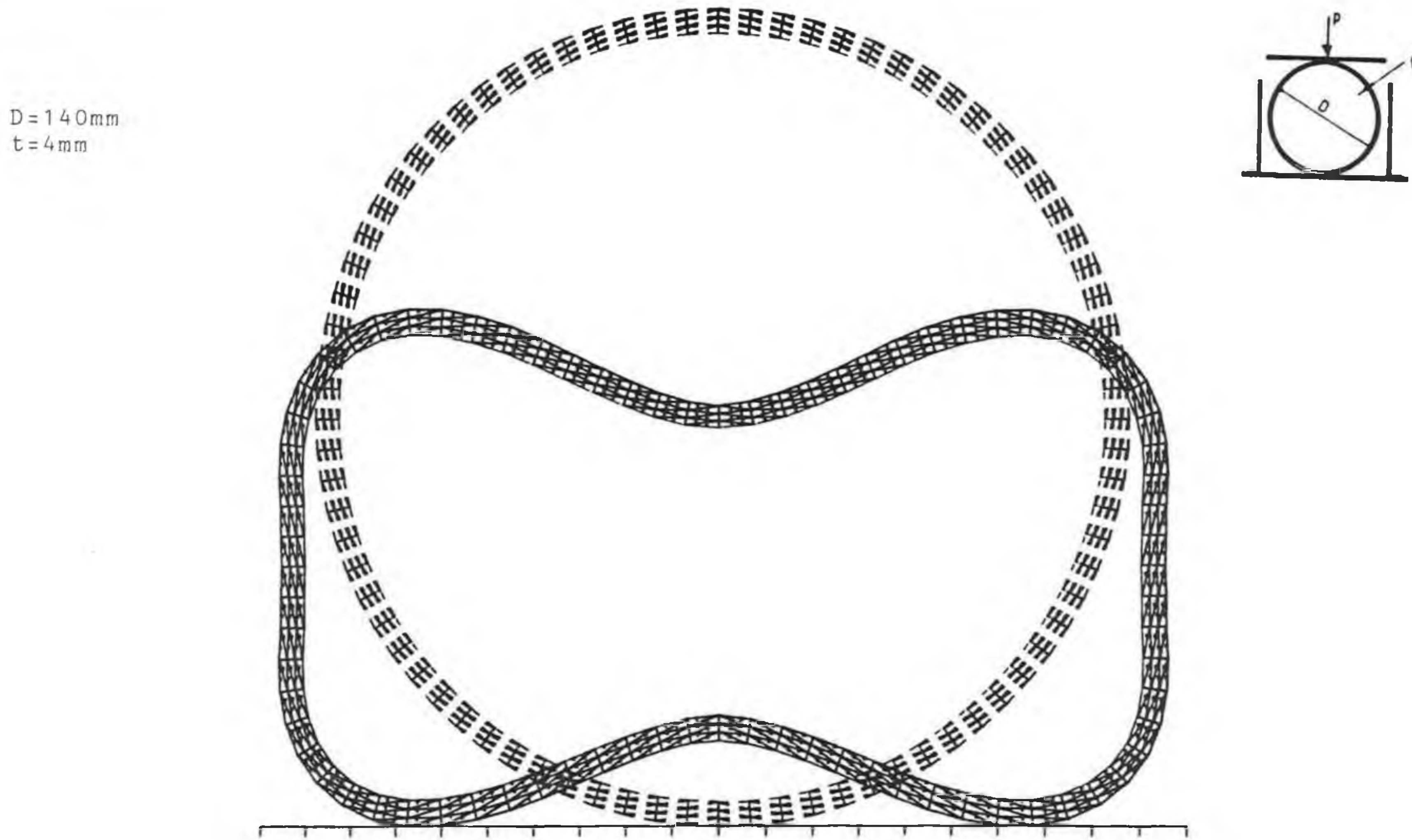


FIG.(68)

CHAPTER 7

CONCLUSION

An elastic-plastic finite element software has been developed and applied to general mechanical engineering problems. This software covers two types of solutions: (a. prescribed displacement; b. prescribed force) for plane elastic-plastic deformation (plane stress and plane strain) with two kinds of loading: concentrated loading and uniformly distributed load. The outcome of the computer program has been compared theoretically and experimentally and from the foregoing it may be concluded that:

1. The method provides details of the stress and strain distribution in the structure.

2. The method shows the extent and development of the plastic zone with the increase in deformation of the structure.

3. The finer mesh gives the better results in terms of the accuracy.

4. Computer time is extensive by using the finer mesh.

5. The material strain hardening properties play a significant role on the shape of the load-deflection curve.

7.1 FURTHER WORK

It is suggested that the following additional work could be undertaken in a future study to enhance the applicability of the program:

1. Reduce the computer time for solutions by using a banded matrix and more sophisticated solution routines.

2. Incorporate the unloading process in the program.

3. Refine the program with several load conditions such as gravity loads, centrifugal loads, temperature loads, non-uniformly distributed loads, etc.

4. Incorporate the friction force in the solution of the program.

5. Improve the mesh generating subroutine by incorporating several kinds of meshes with more flexibility in use.

6. Improve the graphical output of the program by using sophisticated graphical packages and colour monitor to display the contours of stress and other variables.

7. Improve the design of experimental work to make it more up-to date for use in the future.

8. Develop three-dimensional solution capabilities.

9. Apply the program for analysing three-dimensional engineering components for extending its application to die-design and to complex geometry components and complex metal flow problems.

REFERENCES:

1 - Sergerline, L.J., "Applied Finite Element Analysis" John Wiley and Sons, New York, 1984

2 - Southwell, R.V., "Stress-Calculation in Framework by the Method of 'Systematic Relaxation of Constraints' I and II" Proceeding of The Royal Society A, No.151, 1935, pp 59-95

3 - Clough, R.W., et al., "Stiffness and Deflection Analysis of Complex Structures", Journal of Aerospace Science, Vol. 23, No. 9, 1956, pp 805-823

4 - Argyris, J.H., Ing. Arch. Vol. 34, No. 1, (1965)

5 - Melosh, R.J., J.ARSC Struct.Div. Vol. 89, No. 574 (1963)

6 - Gallagher, R.H., Padlog, J. and et al, J.Am. Rocket Soc. Vol. 32, No. 700 (1962)

7 - Zienkiewicz, O.C., "Stress Analysis", edited by O.C. Zienkiewicz and G.S. Hollister, Chapter 8, Pergamon Press (1965)

8 - Argyris, J.H., Kelsey, S. and Kamel, W.H., "Matrix Method of Structural Analysis. A Precis of Recent Developments" edited by F. de Veubeke. Pergamon Press (1963)

9 - Mendelson, A. and Manson, S.S., "Practical Solution of Plastic Deformation Problems in the Elastic-plastics Range" NASA, T.R., R28 (1959)

10 - Pope, G., "A Discrete Element Method for Analysis of Plane Elastic-plastic Stress Problems", R.A.E., T.R. 65028 (1965)

11 - Zienkiewicz, O.C., Valliappan, S. and King, I.P., "Elasto-plastic Solution of Engineering Problems 'Initial Stress' Finite Element Approach" Int. J. Vol. 1 (1968)

12 - Hibbitt, H.D., Marcal, P.V. and Rice, J.R., "A Finite Element Formulation for Problems of Large Strain and Large Displacement" Int.J. Solids Struct., 1069 (1970)

13 - McMeeking, R.M., and Rice, J.R., "Finite-Element Formulations for Problems of Large Elastic-Plastic Deformation" Int.J. Solids Structures, 1975 vol. 11

14 - Naylor, D.J., "Stresses in Nearly Incompressible Materials by Finite Elements with Application to the Calculation of Excess Pore Pressures" Int. J.* 8, 443-460 (1974)

15 - Weisgerber, Frank E., Anand,Subhash C.,"Iterpolative Versus Iterative Solution Scheme for Trasca Yield Condition in Elastic-Plastic Finite Element Analysis" Int.J.* 12, 765-777 (1978)

16 - Gortemaker, P.C.M., de Pater, C., "A Finite Element Formulation for Large Elastic-plastic Deformations" Int. Conf. Struct. Mech. React. Technol 5th, V.L. Berlin, Ger. Aug. 13-17 1979

17 - Okamoto, Noriaki and Nakazawa, Masaru, "Finite Element Incremental Contact Analysis with Various Frictional Conditions" Int. J.* 14, 337-357 (1979)

18 - Deb., Dr. S.R., Barman, S., Chatterji, Dr. T.K., "Analysis of Elastic-Plastic Problems by Finite Element Method" J. Mech. Eng. V.60, 3, 1979

19 - Wissmann, Johannes W. and Hauck, Christian, "Efficient Elastic-Plastic Finite Element Analysis with Higher Order Stress-point Algorithms" Comput Struct, 17, 1, 1983 (89-95)

20 - Contro, Roberto, "An 'Equilibrium' Finite Element Model for Elastic-Plastic Plates" Int. J.** V. 2, pp 87-98, 1986

21 - Chandara, N., "Analysis of Superplastic Metal Forming by a Finite Element Method" Int. J. 26, pp 1925-1944, (1988)

22 - Zienkiewicz, O.O., "The Finite Element Method" McGraw-Hill Book Co. (UK) Ltd. (3rd rev.ed.) 1977

23 - Cheung, Y.K. and Yeo, M.F., "A Practical Introduction to Finite Element Analysis" Pitman Publishing Limited, London 1979

24 - Dhatt, G. and Touzot, G., "The Finite Element Method Displayed" John Wiley & Sons, New York, 1984

25 - Heubner, K.H., "Finite Element Method for Engineers" John Reilly & Sons, New York, 1975, Chapter 1,6,10

26 - Henrywood, R.K., "The Design, Development, Documentation and Support of a Major Finite Element System" Computer Aided Design, Vol. 5, No. 3, 1973

27 - Buell, W.R. and Bush, B.A., "Mesh Generation - A Survey" Journal of Engineering for Industry. Transactions of the American Society of Mechanical Engineering, Series B, Vol. 95, No. 1, 1973, pp332-338

28 - Faddeeva, V.N., "Computational Method of Linear Algebra" New York, Dover Publications, 1956

29 - Johnson, W., "The Compression of Circular Rings" Journal of the Royal Aeronautical Society, Vol. 60, 1956

30 - Reid, S.R. and Reddy, T. Y., "Effect of Strain Hardening on the Lateral Compression of Tubes between Rigid Plates", Int. J. Solid Structures 1978, Vol. 14

31 - Reddy, T.Y. and Reid, S.R., "Lateral Compression of Tubes and Tube-systems with Side Constraints" Int. Journal of Mechanical Sciences, 21, 187 (1979)

32 - Reid, S.R. and Bell, W.W., "Influence of Strain Hardening on the Deformation of Thin Rings Subjected to Opposed Concentrated Load" Int. Journal Solids Structures Vol. 18 No.8, 1982

* int. J. numer methods Eng

** int. J. plast

BIBLIOGRAPHY

- Benham P.P., Warnock, F.V., "Mechanics of Solids and Structures" Pitman Paperbakcs, UK, 1973

- Hill, R., "The Mathematical Theory of Plasticity" Oxford University Press, 1950

- Johnson, W. and Mellor, P.B., "Engineering Plasticity" Van Nostrand Reinhold Co. Ltd., 1973

- Graham, Aledander, "Matrix Theory and Applications for Engineers and Mathematicians" John Wiley & Sons, New York, 1979

- Hinton, E. and Owne, D.R.J., "Finite Elements in Plasticity" Pineridge Press Limited, Swansea, U.K., 1980

- Cheung, Y.K., and Yeo, M.F., "A Practical Introduction to Finite Element Analysis" Pitman Publishing Limited, London, 1979

- Schallert, W.F. and Clarke, C.R., "Programming in Fortran", Addison Wesley Pub. Co., 1979

- Crosslan, N., "Transfer Report" Sheffield City Polytechnic, 1987

- Jones, N. and Weirzbicki, T., "Structural Crashworthiness" Butterworth & Co. (Publishers) Ltd. 1983

APPENDIX A

EXAMPLES OF FINITE ELEMENTS

This Appendix includes the formulation of some examples of finite elements into which a component or structure is subdividied.

1.0 BAR ELEMENT

A bar element is primarily used in Trusses, and by definition, it is assumed to be acted upon only by axial forces. Consider the bar element of Figure (A.1) which has two nodes and one degree of freedom equal to two, in accordance with criterion (a) in Section 2.2 there should only be two polynomial constants. Therefore, the following polynomial is assumed:

$$U = \alpha_1 + \alpha_2 x \qquad (A.1)$$

Or in matrix form:

$$\{\delta(x)\} = \{U\} = [1 \quad x] \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \qquad (A.2)$$

$$\{\delta(x)\} = [f(x)]\,\{\alpha\} \qquad (A.3)$$

The nodal displacement parameter-polynomial constant relationship substituting the nodal coordinates **x = 0 , x = 1** into the equation (A.3) one after the other, i.e.

$$\begin{aligned} u_1 &= \alpha_1 + \alpha_2(o) \\ u_2 &= \alpha_1 + \alpha_2(1) \end{aligned} \qquad (A.4)$$

The equations (A.4) may be expressed in matrix form as:

$$\{\delta^e\} = [A]\,\{\alpha\} \qquad (A.5)$$

or:

$$\{\alpha\} = [A]^{-1}\,\{\delta^e\} \qquad (A.6)$$

in which:

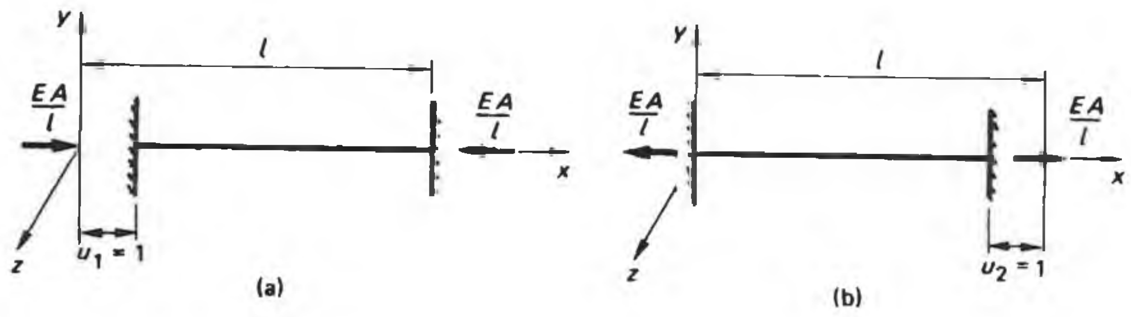


FIG.(A.1)

$$[A] = \begin{bmatrix} 1 & 0 \\ 1 & l \end{bmatrix}, \quad [A]^{-1} = \frac{1}{l}\begin{bmatrix} l & 0 \\ -1 & 1 \end{bmatrix} \qquad (A.7)$$

Substitute the value of equation (A.6) into the value of equation (A.3), i.e.

$$\{\delta(x)\} = \{U\} = [f(x)]\,[A]^{-1}\,\{\delta^e\} \qquad (A.8)$$

or

$$\{U\} = \begin{bmatrix} (1-\frac{x}{l}) & \frac{x}{l} \end{bmatrix} \{\delta^e\} \qquad (A.9)$$

or

$$\{U\} = \begin{bmatrix} (1-\frac{x}{l}) & \frac{x}{l} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \qquad (A.10)$$

1.1 RELATE THE ELEMENT STRAINS TO DISPLACEMENT

Only axial strain is present in this case, i.e.

$$\{\epsilon\} = \epsilon_x = \frac{du}{dx} = \begin{bmatrix} -\frac{1}{l} & \frac{1}{l} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix}$$

$$\{\epsilon\} = [B]\,\{\delta^e\} \qquad (A.11)$$

Therefore

$$[B] = \begin{bmatrix} -\frac{1}{l} & \frac{1}{l} \end{bmatrix} \qquad (A.12)$$

1.2 THE STRESS STRAIN RELATIONSHIP

The relationship is simply given by Hook's law and

$$\{\sigma\} = \sigma_x = E\epsilon_x \qquad (A.13)$$

therefore

$$[D] = E$$

1.3 ELEMENT STIFFNESS MATRIX

From equation (2.55), Section (2.3.5) the stiffness matrix is

$$[K^e] = \int_{(v)} [B]^T \, [D] \, [B] \, d(vol)$$

$$= \int_{Ar} d(area) \int_0^l [B]^T \, [D] \, [B] \, dx$$

$$= Ar \int_0^l \begin{bmatrix} -\frac{1}{l} \\ \frac{1}{l} \end{bmatrix} [E] \begin{vmatrix} -\frac{1}{l} & \frac{1}{l} \end{vmatrix} dx$$

$$= \frac{EAr}{l} \begin{vmatrix} 1 & -1 \\ -1 & 1 \end{vmatrix} \qquad (A.14)$$

According to equation (2.1), Section (2.2) the load matrix is:

$$\{p^e\} = [K^e] \{\delta^e\} \qquad (A.15)$$

or

$$\begin{bmatrix} p_{x1} \\ p_{x2} \end{bmatrix} = \frac{EAr}{l} \begin{vmatrix} 1 & -1 \\ -1 & 1 \end{vmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

2.0 BEAM ELEMENT

A beam element is concerned with bending action as shown in Figure (A.2). The element has two nodes each having two degrees of freedom (vertical displacement and rotation).

Since the element has four degrees of freedom ($V_1, \theta_{21}, V_1, \theta_{22}$), there must be four unknown coefficients in the polynomial representing displacement. Therefore, the following polynomial is assumed:

$$V = \alpha_1 + \alpha_2 x + \alpha_3 x^2 + \alpha_4 x^3 \tag{A.16}$$

Hence

$$\theta_2 = \frac{dv}{dx} = \alpha_2 + 2\alpha_3 x + 3\alpha_4 x^2 \tag{A.17}$$

The displacement at any point within the element is defined by **v** and θ_2, thus the displacement function in matrix form:

$$\{\delta(\mathbf{x})\} = \begin{Bmatrix} v \\ \theta_2 \end{Bmatrix} = \begin{vmatrix} 1 & x & x^2 & x^3 \\ 0 & 1 & 2x & 3x^2 \end{vmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{Bmatrix} \tag{A.18}$$

or

$$\{\delta(x)\} = [f(x)] \{\alpha\} \tag{A.19}$$

If the nodal coordinates are now substituted with equations (A.16) and (A.17) (at nodal 1: x = 0; at nodal 2: x = 1), it is possible to established the following equations:

$$\begin{Bmatrix} v_1 \\ \theta_{21} \\ v_2 \\ \theta_{22} \end{Bmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1^2 & 1^3 \\ 0 & 1 & 21 & 31^2 \end{vmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{Bmatrix} \tag{A.20}$$

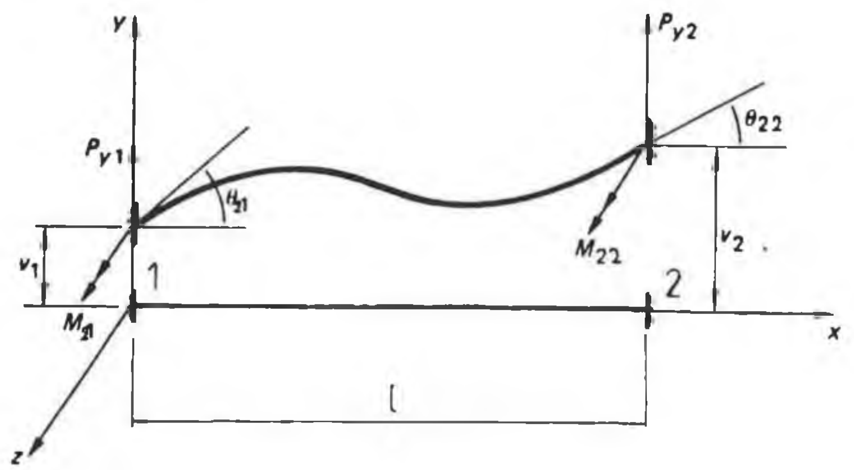


FIG.(A.2)

or:

$$\{\delta^e\} = [A]\ \{\alpha\} \tag{A.21}$$

Therefore,

$$\{\alpha\} = [A]^{-1}\ \{\delta^e\} \tag{A.22}$$

in which:

$$[A] = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & l & l^2 & l^3 \\ 0 & 1 & 2l & 3l^2 \end{vmatrix} \tag{A.23}$$

$$[A]^{-1} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -3/l^2 & -2/l & 3/l^2 & -1/l \\ 2/l^3 & 1/l^2 & -2/l^3 & 1/l^2 \end{vmatrix}$$

Substitute the value of equation (A.22) with the value of equation (A.19), i.e.

$$\{\delta(x)\} = [f(x)]\ [A]^{-1}\ \{\delta^e\} \tag{A.24}$$

$$\begin{bmatrix} v \\ \\ \theta_2 \end{bmatrix} = \begin{vmatrix} (1-\frac{3x^2}{l^2}+\frac{2x^3}{l^2}) & (x-\frac{2x^2}{l}+\frac{x^3}{l^2}) & (\frac{3x^2}{l^2}\;\frac{2x^3}{l^2}) & (\frac{-x^2}{l}+\frac{x^3}{l^2}) \\ \\ (-\frac{6x}{l^2}+\frac{6x^2}{l^2}) & (1-\frac{4x}{l}+\frac{3x^2}{l^2}) & (\frac{6x}{l^2}-\frac{6x^2}{l^2}) & (-\frac{3x}{l}+\frac{3x^2}{l^2}) \end{vmatrix} \begin{bmatrix} v_1 \\ \theta_{21} \\ \\ v_2 \\ \theta_{22} \end{bmatrix} \tag{A.25}$$

2.1 STRAIN-DISPLACEMENT RELATIONSHIP

The only strain for a beam element is the curvature about the Z axis

$$\{\epsilon\} = -y \frac{d^2v}{dx^2}$$

$$= y \left| \left(\frac{6}{l^2} - \frac{12x}{l^3}\right), \left(-\frac{4}{l} - \frac{6x}{l^2}\right), \left(\frac{-6}{l^2} + \frac{12x}{l^3}\right), \left(-\frac{2}{l} - \frac{6x}{l^2}\right) \right| \begin{bmatrix} v_1 \\ \theta_{21} \\ v_2 \\ \theta_{22} \end{bmatrix}$$

or

$$\{\epsilon\} = [B]\ \{\delta^e\} \qquad (A.26)$$

2.2. THE STRESS STRAIN RELATIONSHIP

For a beam element the stress corresponds to σ and the strain corresponds to $-y \frac{d^2v}{dx^2}$, therefore

$$\{\sigma\} = E\ (-y \frac{d^2v}{dx^2}) \qquad (A.27)$$

or:

$$\{\sigma\} = [E]\ \{\epsilon\}$$
$$= [B]\ \{\epsilon\}$$
$$[B] = [E] \qquad (A.28)$$

2.3 ELEMENT STIFFNESS MATRIX

$$[K^e] = \int_{(v)} [B]^T\ [D]\ [B]\ d(vol)$$

For the beam element d(vol) is replaced by **(bdydx) b-** is the third dimension of the beam, therefore

$$[K^e] = \int_o^R \int_{y_1}^{y_2} y \begin{vmatrix} \frac{6}{l^2} - \frac{12x}{l^3} \\ \frac{4}{l} - \frac{6x}{l^2} \\ -\frac{6}{l^2} + \frac{12x}{l^3} \\ \frac{2}{l} - \frac{6x}{l^2} \end{vmatrix} [E]\, y \begin{vmatrix} \frac{6}{l^2} - \frac{12x}{l^3}, & \frac{4}{l} - \frac{6x}{l^2} + \frac{12x}{l^3}, & \frac{2}{l} - \frac{6x}{l^2} \end{vmatrix} dbydx$$

but:

$$\int_{y_1}^{y_2} y^2\, d\, b\, y = I$$

Therefore

$$[K^e] = \frac{EI}{l^3} \begin{vmatrix} 12 & 6l & -12 & 6l \\ 6l & 4l^2 & -6l & 2l^2 \\ -12 & -6l & 12 & -6l \\ 6l & 2l^2 & -6l & 4l^2 \end{vmatrix} \qquad (A.29)$$

The nodal loads of form:

$$\{p^e\} = [K^e]\{\delta^e\} \qquad (A.30)$$

$$\{p^e\} = \begin{bmatrix} F_{y1} \\ M_{21} \\ F_{y2} \\ M_{22} \end{bmatrix}, \quad \{\delta^e\} = \begin{bmatrix} v_1 \\ \theta_{21} \\ v_2 \\ \theta_{22} \end{bmatrix}$$

M_2 - is the moment about the z axis

Θ_2 - the rotation

3.0 THREE NODE TRIANGULAR AXI-SYMMETRIC ELEMENT

The three node triangular axi-symmetric element is used in the problem of stress distribution in bodies of revolution (axi-symmetric solids) under axi-symmetric loading. Figure (A.3) shows that **OZ** is the axis of symmetry, **OR** is the radial axis and the three node tirangular axi-symmetric element. Each one of the nodes has two degrees of freedom.

This gives a total of six degrees of freedom $(u_1, v_1, u_2, v_2, u_3, v_3)$. The forces are $(F_{r1} . F_{z1}, F_{r2}, F_{z2}, F_{r3}, F_{z3})$

3.1 DISPLACEMENT FUNCTION

The displacement function is given by two linear polynomials as:

$$u = \alpha_1 + \alpha_{2r} + \alpha_3 Z \qquad \text{(A.31)}$$
$$v = \alpha_4 + \alpha_{5r} + \alpha_6 Z$$

or in matrix form:

$$\{\delta(r,z)\} = \begin{Bmatrix} u \\ v \end{Bmatrix} = \begin{vmatrix} 1 & r & z & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & r & z \end{vmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \end{Bmatrix}$$

i.e.

$$\{\delta(r,Z)\} = [f(r,Z)]\ \{\alpha\} \qquad \text{(A.32)}$$

At node 1: $r = r_1,\ Z = Z_1,\ u = u_1,\ v = v_1$
At node 2: $r = r_2,\ Z = Z_2,\ u = u_2,\ v = v_2$
At node 3: $r = r_3,\ Z = Z_3,\ u = u_3,\ v = v_3$

Therefore

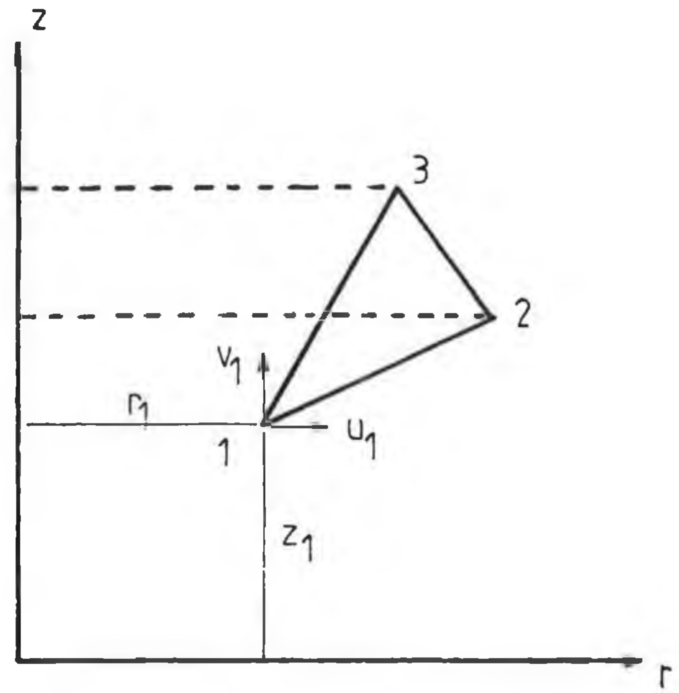


FIG.(A.3)

$$\{\delta^e\} = \begin{Bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{Bmatrix} = \begin{bmatrix} 1 & r_1 & Z_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & r_1 & Z_1 \\ 1 & r_2 & Z_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & r_2 & Z_2 \\ 1 & r_3 & Z_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & r_3 & Z_3 \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \end{Bmatrix}$$

or:

$$\{\delta^e\} = [A]\,\{\alpha\} \qquad \text{(A.33)}$$

thus:

$$\{\alpha\} = [A]^{-1}\,\{\delta^e\} \qquad \text{(A.34)}$$

Substitute the value of equation (A.34) with the value of equation (A.32). The displacement function in terms of nodal displacement is:

$$\{\delta(r,Z\} = [F(r,Z)]\,[A]^{-1}\,\{\delta^e\} \qquad \text{(A.35)}$$

in which:

$$[A]^{-1} = \frac{1}{2\Delta}$$

$$\begin{bmatrix} r_2Z_3-r_3Z_2 & 0 & -r_1Z_3+r_3Z_1 & 0 & r_1Z_2-r_2Z_1 & \\ Z_2 - Z_3 & 0 & Z_3 - Z_1 & 0 & Z_1 - Z_2 & \\ r_3 - r_2 & 0 & r_1 - r_3 & 0 & r_2 - r_1 & \\ 0 & r_2Z_3-r_3Z_2 & 0 & -r_1Z_3+r_3Z_1 & 0 & r_1Z_2-r_2Z_1 \\ 0 & Z_2 - Z_3 & 0 & Z_3 - Z_1 & 0 & Z_1 - Z_2 \\ 0 & r_3 - r_2 & 0 & r_1 - r_3 & 0 & r_2 - r_1 \end{bmatrix}$$

(A.36)

where:

Δ = area of triangle

3.2 STRAIN DISPLACEMENT RELATIONSHIP

There are four components of strain have now to be considered.

$$\{\epsilon(r,Z)\} = \begin{Bmatrix} \epsilon_r \\ \epsilon_Z \\ \epsilon_\theta \\ \gamma_{rZ} \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial u}{\partial r} \\ \dfrac{\partial v}{\partial Z} \\ \dfrac{u}{r} \\ \dfrac{\partial u}{\partial Z} + \dfrac{\partial v}{\partial r} \end{bmatrix} \qquad (A.37)$$

and substituting for **u** and **v** from equation (A.31) into the strain expressions, i.e.

$$\epsilon_r = \alpha_2$$

$$\epsilon_Z = \alpha_6$$

$$\epsilon_\theta = \frac{\alpha_1}{r} + \alpha_2 + \frac{\alpha_3}{r} Z$$

$$\gamma_{rZ} = \alpha_3 + \alpha_5$$

or in matrix form:

$$\{\epsilon(r.Z)\} = [C]\{\alpha\} \qquad (A.38)$$

$$[C] = \begin{vmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \dfrac{1}{r} & 1 & \dfrac{Z}{r} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{vmatrix}$$

Substitute the value of $\{\alpha\}$ from equation (A.34):

$$\{\epsilon(r,Z)\} = [C]\,[A]^{-1}\,\{\delta^e\} \qquad (A.39)$$

which may be written as:

$$\{\epsilon(r,Z)\} = [B]\,\{\delta^e\} \qquad (A.40)$$

where:

$$[B] = [C]\,[A]^{-1} \qquad (A.41)$$

Substitute the value of $[A]^{-1}$ from equation (A.36):

$$[B] = \frac{1}{2\Delta}\left|\begin{matrix} Z_2 - Z_3 & 0 & Z_3 - Z_1 & 0 & Z_1 - Z_2 & 0 \\ 0 & r_3 - r_2 & & r_1 - r_3 & 0 & r_2 - r_1 \\ \dfrac{r_2Z_3 - r_3Z_2}{r} & 0 & \dfrac{-r_1Z_3 + r_3Z_1}{r} & 0 & \dfrac{r_1Z_2 - r_2Z_1}{r} & 0 \\ \genfrac{}{}{0pt}{}{+(Z_2-Z_3)}{Z} & 0 & \genfrac{}{}{0pt}{}{+(Z_3-Z_1)}{Z} & 0 & \genfrac{}{}{0pt}{}{+(Z_1-Z_2)}{Z} & 0 \\ \genfrac{}{}{0pt}{}{+-(r_3-r_2)}{r} & 0 & \genfrac{}{}{0pt}{}{+-(r_1-r_2)}{r} & 0 & \genfrac{}{}{0pt}{}{+-(r_2-r_1)}{r} & 0 \\ r_3 - r_2 & Z_2 - Z_3 & r_1 - r_3 & Z_3 - Z_1 & r_2 - r_1 & Z_1 - Z_2 \end{matrix}\right|$$

(A.42)

3.3 STRESS-STRAIN RELATIONSHIP

There are four components of stress that have to be considered here as well. Figure (A.4) illustrates and defines these strains and the associated stresses. The stress-strain equations for three-dimensional elasticity are presented to:

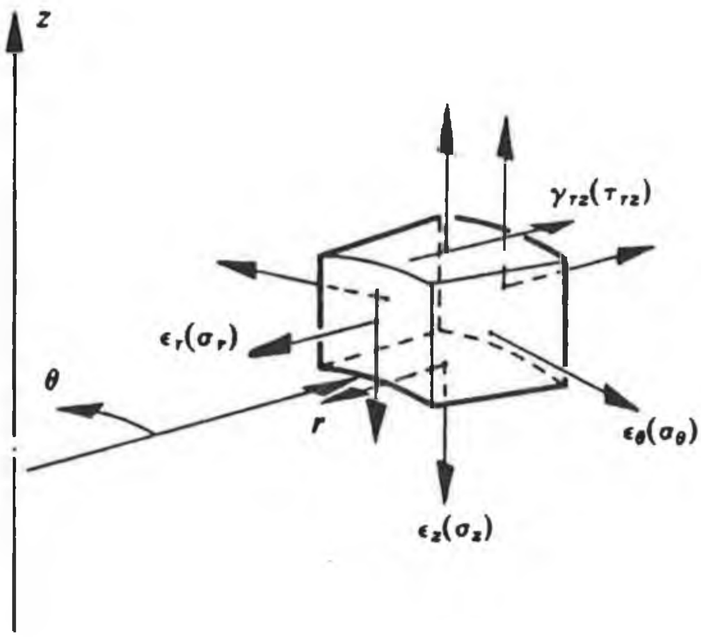

z
$\gamma_{rz}(\tau_{rz})$
$\epsilon_r(\sigma_r)$
θ
r
$\epsilon_\theta(\sigma_\theta)$
$\epsilon_z(\sigma_z)$

FIG.(A.4)

$$\epsilon_r = \frac{\sigma_r}{E} - \gamma\frac{\sigma_z}{E} - \gamma\frac{\sigma_\theta}{E}$$

$$\epsilon_z = \frac{\sigma_z}{E} - \gamma\frac{\sigma_z}{E} - \gamma\frac{\sigma_\theta}{E} \qquad (A.43)$$

$$\epsilon_\theta = \frac{\sigma_\theta}{E} - \gamma\frac{\sigma_z}{E} - \gamma\frac{\sigma_z}{E}$$

$$\gamma_{rz} = \frac{\tau_{rz}}{G} = \frac{\tau_{rz}\,2(1+\gamma)}{E}$$

By solving for σ_r, σ_z, σ_θ and $_{rz}$, therefore

$$\begin{Bmatrix} \sigma_r \\ \sigma_z \\ \sigma_\theta \\ \tau_{rz} \end{Bmatrix} = \frac{E(1-\gamma)}{(1+v)(1-2\gamma)} \begin{bmatrix} 1 & \frac{\gamma}{1-\gamma} & \frac{\gamma}{1-\gamma} & 0 \\ \frac{\gamma}{1-\gamma} & 1 & \frac{\gamma}{1-\gamma} & 0 \\ \frac{\gamma}{1-\gamma} & \frac{\gamma}{1-\gamma} & 1 & 0 \\ 0 & 0 & 0 & \frac{(1-2\gamma)}{2(1-\gamma)} \end{bmatrix} \begin{Bmatrix} \epsilon_r \\ \epsilon_z \\ \epsilon_\theta \\ \gamma_{rz} \end{Bmatrix} \qquad (A.44)$$

or:

$$\{\sigma\} = [D_o]\{\epsilon\} \qquad (A.45)$$

$[D_o]$ - the elasticity matrix.

3.4 ELEMENT STIFFNESS MATRIX

By relating to equation (2.55), Section (2.3.5) the stiffness matrix is:

$$[K^e] = \int_{(v)} [B]^T \, [D] \, [B] \, d(vol)$$

but for a body of revolution

$$d(vol) = 2\pi r dr \, dZ$$

thus:

$$[K^e] = \iint [B]^T \, [D] \, [B] \, 2\pi \, r \, dr \, dZ \qquad (A.46)$$

As [B] depends on r and Z, the equation (A.45) has to be integrated with respect to r and Z. To avoid this $[K^e]$ can be obtained approximately by evaluating $[\bar{B}]$ for a centreoidal point defined by the coordinates.

$$\bar{r} = \frac{1}{3}(r_1 + r_2 + r_3) \quad \text{and} \quad \bar{Z} = \frac{1}{3}(Z_1 + Z_2 + Z_3)$$

The stiffness matrix

$$[K^e] = 2\pi[\bar{B}]^T \, [D] \, [\bar{B}] \, \bar{r} \, A \qquad (A.47)$$

where:

A is again the area of the triangle.

4.0 FOUR NODE RECTANGULAR ELEMENT

Figure (A.5) shows a four node rectangular element. Each one of the nodes has two degrees of freedom. This gives a total of eight degrees of freedom $(u_1, v_1, u_2, v_2, u_3, v_3, u_4, v_4)$. The nodal forces are $(F_{x1}, F_{y1}, F_{x2}, F_{y2}, F_{x3}, F_{y3}, F_{x4}, F_{y4})$.

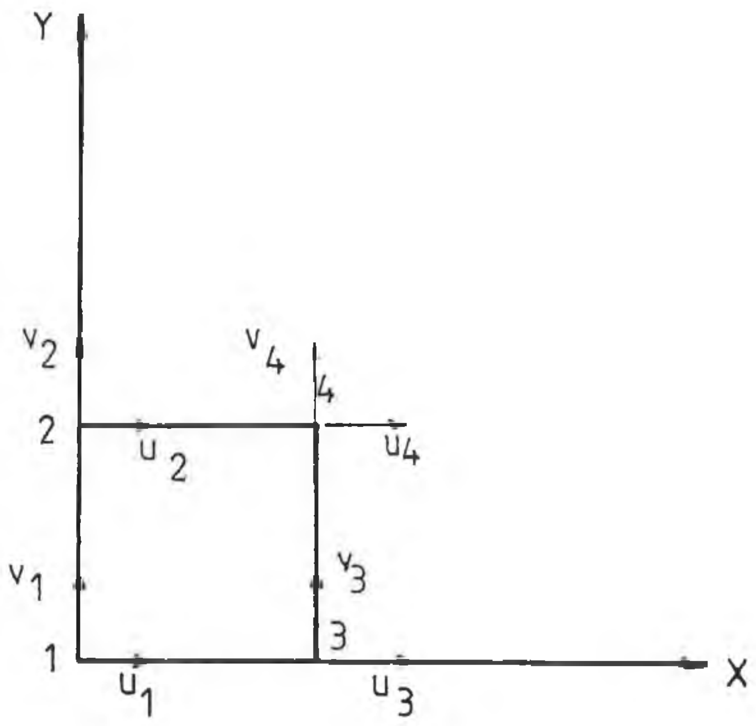


FIG.(A.5)

4.1 DISPLACEMENT FUNCTION

Since there are eight degrees of freedom, eight coefficients are required in the polynomial describing displacement.

$$\begin{aligned} u &= \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 xy \\ v &= \alpha_5 + \alpha_6 x + \alpha_7 y + \alpha_8 xy \end{aligned} \qquad (A.48)$$

or in matrix form:

$$\{\delta(x,y)\} = \begin{bmatrix} u \\ \\ v \end{bmatrix} = \begin{bmatrix} 1 & x & y & xy & 0 & 0 & 0 & 0 \\ \\ 0 & 0 & 0 & 0 & 1 & x & y & xy \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \\ \alpha_7 \\ \alpha_8 \end{bmatrix}$$

i.e.

$$\{\delta(x,y) = [f(x,y]\ \{\alpha\} \qquad (A.49)$$

At node 1: $x = 0$, $y = 0$, $u = u_1$, $v = v_1$

At node 2: $x = 0$, $y = b$, $u = u_2$, $v = v_2$

At node 3: $x = a$, $y = 0$, $u = u_3$, $v = v_3$

At node 4: $x = a$, $y = b$, $u = u_4$, $v = v_4$

Therefore,

$$\{\delta^e\} = \begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \\ u_4 \\ v_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & b & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & a & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & a & 0 & 0 \\ 1 & a & b & ab & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & a & b & ab \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \\ \alpha_7 \\ \alpha_8 \end{bmatrix}$$

or:

$$\{\delta_e\} = [A]\{\alpha\} \tag{A.50}$$

Thus:

$$\{\alpha\} = [A]^{-1}\{\delta^e\} \tag{A.51}$$

Therefore, the displacement function in terms of nodal displacement is:

$$\{\delta(x,y)\} = [f(x,y)]\,[A]^{-1}\,\{\delta^e\} \tag{A.52}$$

in which:

$$[A]^{-1} = \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1/a & 0 & 0 & 0 & 1/a & 0 & 0 & 0 \\ -1/b & 0 & 1/b & 0 & 0 & 0 & 0 & 0 \\ 1/ab & 0 & -1/ab & 0 & -1/ab & 0 & 1/ab & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1/a & 0 & 0 & 0 & 1/a & 0 & 0 \\ 0 & -1/b & 0 & 1/b & 0 & 0 & 0 & 0 \\ 0 & 1/ab & 0 & -1/ab & 0 & -1/ab & 0 & 1/ab \end{vmatrix} \tag{A.53}$$

4.2 STRAIN-DISPLACEMENT RELATIONSHIP

For the plane elasticity the strain vector is:

$$\{\epsilon(x,y)\} = \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial v}{\partial y} \\ \dfrac{\partial u}{\partial y} + \dfrac{\partial v}{\partial x} \end{bmatrix} \tag{A.54}$$

By substituting the value of u and v from equation (A.48) with the value of equation (A.54) i.e.

$$\begin{aligned} \epsilon_x &= \alpha_2 + \alpha_4 y \\ \epsilon_y &= \alpha_7 + \alpha_8 x \\ xy &= \alpha_3 + \alpha_4 x + \alpha_6 + \alpha_8 y \end{aligned} \qquad (A.55)$$

It can be seen that the strain varies lineraly along the edge of an element. It can be deducted that the stresses will vary in a similar way.

The equation (A.55) above can be written in matrix form as:

$$\{\epsilon(x,y) = [C]\ \{\alpha\} \qquad (A.56)$$

Where:

$$[C] = \begin{vmatrix} 0 & 1 & 0 & y & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & x \\ 0 & 0 & 1 & x & 0 & 1 & 0 & y \end{vmatrix} \qquad (A.57)$$

Substitute the value of equation (A.55) with the value of equation (A.56), i.e.

$$\{\epsilon(x,y)\} = [C]\ [A]^{-1}\ \{\epsilon^e\} \qquad (A.58)$$

or:

$$\{\epsilon(x,y)\} = [B]\ \{\delta^e\} \qquad (A.59)$$

where:

$$[B] = [C]\ [A]^{-1} \qquad (A.60)$$

By substituting the value of $[A]^{-1}$ and $[C]$ with the value of equation (A.60) i.e.

$$
[B]=\begin{bmatrix}
-\frac{1}{a}+\frac{y}{ab} & 0 & \frac{-y}{ab} & 0 & \frac{1}{a}-\frac{y}{ab} & 0 & \frac{y}{ab} & 0 \\
0 & -\frac{1}{b}+\frac{x}{ab} & 0 & \frac{1}{b}+\frac{x}{ab} & 0 & \frac{-x}{ab} & 0 & \frac{x}{ab} \\
-\frac{1}{b}+\frac{x}{ab} & -\frac{1}{a}+\frac{y}{ab} & \frac{1}{b}-\frac{x}{ab} & -\frac{y}{ab} & -\frac{x}{ab} & \frac{1}{a}-\frac{y}{ab} & \frac{x}{ab} & \frac{y}{ab}
\end{bmatrix}
$$

(A.61)

4.3 STRESS-STRAIN RELATIONSHIP

The stress-strain relationship for four node rectangular elements is the same for constant strain triangular elements, therefore, the stress-strain can be expressed as:

$$\{\sigma(x,y)\} = [D] \{\epsilon(x,y)\} \qquad (A.62)$$

where:

[D] - the elasticity matrix which has the same elements in Section (2.3.4)

4.4 ELEMENT STIFFNESS MATRIX

In the same way in Section (2.3.5), the stiffness matrix is:

$$[K^e] = \int_{(v)} [B]^T [D] [B] \, d(vol) \qquad (A.63)$$

but for element of constant thickness t :

$$[K^e] = t \iint [B]^T [D] [B] \; dxdy$$

or

$$[K^e] = t \iint [B]^T [D] [B] \, dx \, dy \qquad (A.64)$$

where:

t : the thickness of element

5.0 TETRAHEDRAL ELEMENT

The simplest two-dimensional continuum element was a triangle. In three-dimensional problems of stress analysis its equivalent is a tetrahedron, an element with four nodal corners.

5.1 DISPLACEMENT FUNCTION

Figure (A.6) illustrates a tetrahedral element **a,b,c,p** in space defined by the **x,y** and **z** co-ordinates. The state of displacement of a point is defined by three displacement components **u,v,** and **w** in the direction of three co-ordinates **x,y** and **z**. Thus:

$$\{\delta(x,y,z)\} = \begin{vmatrix} u \\ v \\ w \end{vmatrix} \qquad (A.65)$$

Just as in a plane triangle, where a linear variation of a quantity was defined by its three nodal values, here a linear variation will be defined by the four nodal values, therefore,

$$u = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 z \qquad (A.66)$$

Equating the values of displacement at the nodes we have four equations of the type:

$$u_a = \alpha_1 + \alpha_2 x_a + \alpha_3 y_a + \alpha_4 z_a \text{ etc.} \qquad (A.67)$$

from which α_1 to α_4 can be evaluated.

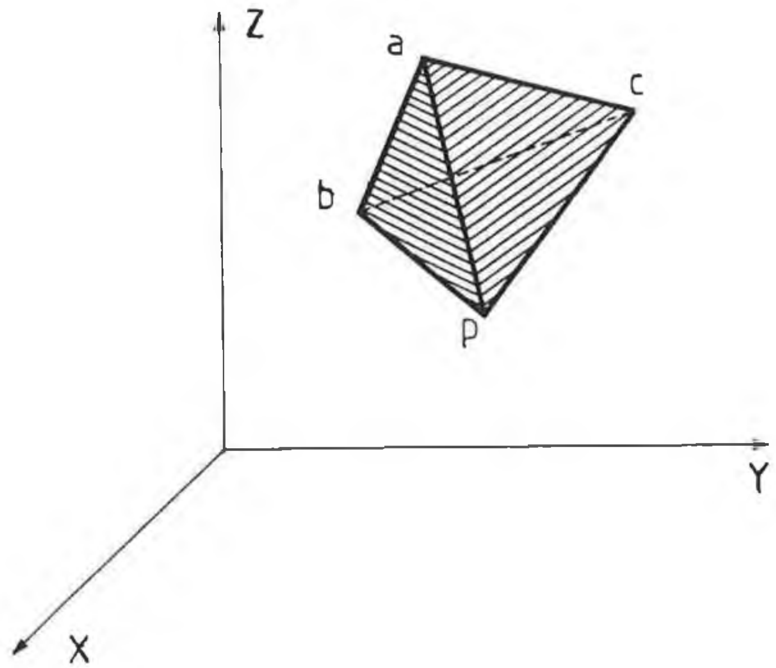


FIG.(A.6)

Again, it is possible to write this solution in a form similar to that in the constant strain triangular element (see Section 2.2.1 and Appendix B). By using a determinant form, i.e.

$$u = \frac{1}{6V}\Big[(a_a + b_a x + c_a y + d_a z)\, u_a + (a_b + b_b x + c_b y + d_b z)\, u_b$$

$$+(a_c + b_c x + c_c y + d_c z)\, u_c + (a_p + b_p x + c_p y + d_p z) u_p\Big] \quad (A.68)$$

with:

$$6V = \det \begin{vmatrix} 1 & x_a & y_a & z_a \\ 1 & x_b & y_b & z_b \\ 1 & x_c & y_c & z_c \\ 1 & x_p & y_p & z_p \end{vmatrix}$$

in which, incidentially, the value **V** represents the volume of the tetrahedron. By expanding the other relevant determinants into their co-factors, i.e.

$$a_a = \det \begin{vmatrix} x_b & y_b & z_b \\ x_c & y_c & z_c \\ x_p & y_p & z_p \end{vmatrix} \;;\quad b_a = \det \begin{vmatrix} 1 & y_b & z_b \\ 1 & y_c & z_c \\ 1 & y_p & z_p \end{vmatrix}$$

$$c_a = \det \begin{vmatrix} x_b & 1 & z_b \\ x_c & 1 & z_c \\ x_p & 1 & z_p \end{vmatrix} \;;\quad d_a = \det \begin{vmatrix} x_b & y_b & 1 \\ x_c & y_c & 1 \\ x_p & y_p & 1 \end{vmatrix} \quad (A.69)$$

With the other constants defined by cyclic interchange of the subscripts in the order of **p,a,b,c.**
The element displacement is defined by the twelve displacement components of the nodes as:

$$\{\delta^e\} = \begin{bmatrix} \delta_a \\ \delta_b \\ \delta_c \\ \delta_p \end{bmatrix} \qquad (A.70)$$

with:

$$\{\delta_a\} = \begin{bmatrix} u_a \\ v_a \\ w_a \end{bmatrix} \text{ etc.} \qquad (A.71)$$

The displacements of an arbitrary point of the form:

$$\{\delta(x,y,z)\} = [f(x,y,z)]\,[A]^{-1}\,\{\delta^e\} \qquad (A.72)$$

5.2 STRAIN-DISPLACEMENT RELATIONSHIP

Six strain components are relevant in full three-dimensional analysis. The strain matrix can now be defined as:

$$\{\epsilon(x,y,z)\} = \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \epsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial v}{\partial y} \\ \dfrac{\partial w}{\partial z} \\ \dfrac{\partial u}{\partial y} + \dfrac{\partial v}{\partial x} \\ \dfrac{\partial v}{\partial z} + \dfrac{\partial w}{\partial y} \\ \dfrac{\partial x}{\partial x} + \dfrac{\partial u}{\partial z} \end{bmatrix} \qquad (A.73)$$

Therefore, the strain-displacement relationship can be written as:

$$\{\epsilon(x,y,z)\} = [B]\ \{\delta^{e}\} \qquad \text{(A.74)}$$

5.3 STRESS-STRAIN RELATIONSHIP

Six stress components are presented in the stress matrix, which is of the form:

$$\{\sigma(x,y,z\} = \begin{bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{bmatrix} = [D]\ \{\epsilon(x,y,z)\} \qquad \text{(A.75)}$$

in which, the elasticity matrix [D] for isotropic material can be written as:

$$[D] = \frac{E(1-\gamma)}{(1+\gamma)(1-2\gamma)} \begin{vmatrix} 1 & \gamma/(1-\gamma) & \gamma/(1-\gamma) & 0 & 0 & 0 \\ & 1 & \gamma/(1-\gamma) & 0 & 0 & 0 \\ & & 1 & 0 & 0 & 0 \\ & & & \dfrac{(1-2\gamma)}{2(1-\gamma)} & 0 & 0 \\ & \text{Symmetric} & & & \dfrac{(1-2\gamma)}{2(1-\gamma)} & 0 \\ & & & & & \dfrac{(1-2\gamma)}{2(1-\gamma)} \end{vmatrix} \qquad \text{(A.76)}$$

5.4 ELEMENT STIFFNESS MATRIX

The stiffness matrix defined by the general relationship can now be explicity integrated since the strain and stress components are constant within the element. Thus, the stiffness matrix can be defined as:

$$[K^e] = [B]^T [D] [B] V \qquad (A.77)$$

where V represents the volume of the elementary tetrahedron.

6.0 ISOPARAMETRIC ELEMENTS

To ensure that a small number of elements can represent a relatively complex shape of type which is liable to occur in real engineering components, simple rectangles and triangles no longer suffice. It

is necessary to distort these simple shapes into others of more complex shape. For example, two-dimensional elements can be mapped into distorted forms as shown in Figure (A.7).

It can be seen that it is possible to use a set of curvilinear coordinates ξ, η, which will take up unit values along the element edges.

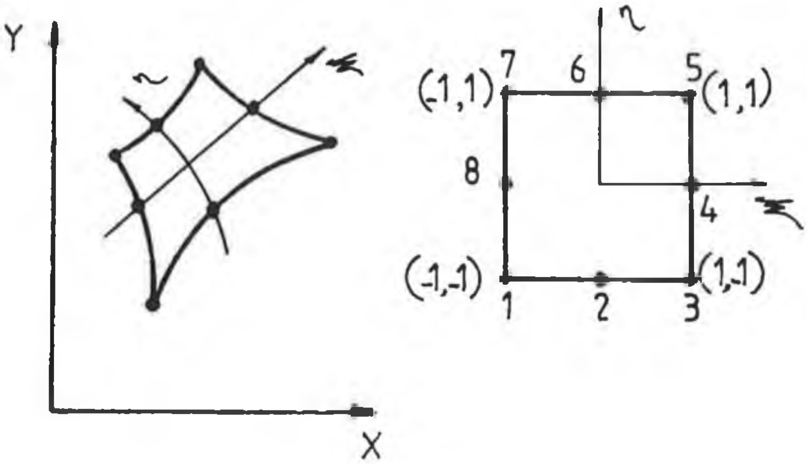
It has been shown previously that for a two-dimensional element, the displacement function in terms of shape functions (see Appendix B) is given by:

$$\begin{aligned} U &= N_1u_1 + N_2u_2 + N_3u_3 \quad + \quad N_ru_r \\ &= \Sigma N_iu_i \qquad (A.78) \\ V &= N_1v_1 + N_2v_2 + N_3v_3 \quad + \quad N_rv_r \\ &= \Sigma N_iv_i \end{aligned}$$

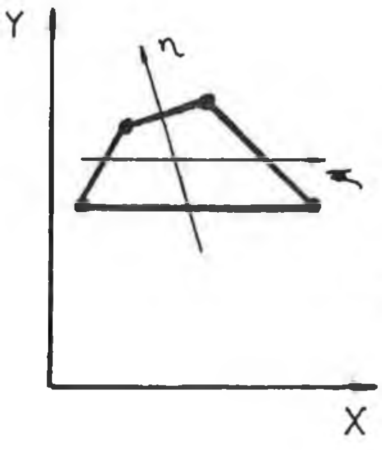
in which N_i is a shape function of the curvilinear coordinates ξ, η and r is the number of nodes. The coordinates x and y inside the element domain can be described in a similar way by:

$$\begin{aligned} x &= M_1x_1 + M_2x_2 + \text{--} \quad M_rx_r \\ &= \Sigma M_iy_i \qquad (A.79) \\ y &= M_1y_1 + M_2y_2 + \quad M_ry_r \\ &= \Sigma M_iy_i \end{aligned}$$

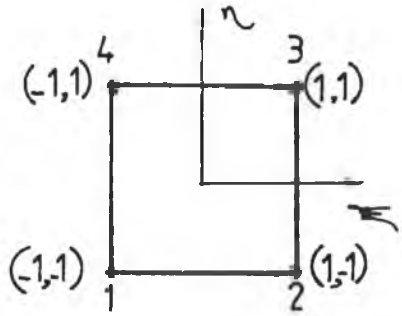
in which M_i is also a function of ξ, η



(a) quadratic rectangular element



(b) linear quadrilateral element



(c) linear rectangular element

FIG.(A.7)

CARTESIAN AND CURVILINEAR COORDINATES

For the particular case in which N_i and M_i are identical, i.e. the shape functions defining the displacement fields and geometry are the same, the element is termed isoparametric. Equation (A.79) provides the relationship between the cartesian and the curvilinear coordinate systems, so that appropriate transformations can be carried out in the stiffness formulation.

6.1 QUADRILATERAL ELEMENTS

6.1.1 Linear element

This element in Figure (A.7(b)) is a more general form of the rectangular element. The displacement functions in simple polynomial form are of the form:

$$u = \alpha_1 + \alpha_2 \xi + \alpha_3 \eta + \alpha_4 \xi \eta \qquad \text{(A.80)}$$
$$v = \alpha_5 + \alpha_6 \xi + \alpha_7 \eta + \alpha_8 \xi \eta$$

The shape functions can be constructed directly as products of the linear Lagrange polynomials (see Appendix B).

At Node 1 : $\xi = -1$, $\eta = -1$ and $N_1 = ¼(1-\xi)\ (1-\eta)$

Since $N_1 = 1$ where the cordinates of node 1 are substituted into the expressions.

Similarly for Node 2: $\xi = 1$, $= 1$ and $N_2 = ¼(1+\xi)(1-\eta)$

Node 3: $\xi = 1$, $\eta = 1$ and $N_3 = ¼(1+\xi)(1+\eta)$

Node 4: $\xi = -1$, $\eta = 1$ and $N_4 = ¼(1-\xi)(1+\eta)$

The four functions (N_1, N_2, N_3, N_4) can be written as a single equation:

$$N_i = ¼(1+\xi\xi_i)\ (1+\eta\eta_i) \qquad \text{(A.81)}$$

where ξ_i and η_i are the coordinates of Node i.

6.1.2 Quadratic element (Figure A.7(a))

There are altogether eight nodes for this element with only three nodes along one edge. The displacement functions in polynomial form are:

$$u = \alpha_1 + \alpha_2\xi + \alpha_3\eta + \alpha_4\xi^2 + \alpha_5\xi\eta + \alpha_6\eta^2 + \alpha_7\xi^2\eta + \alpha_8\xi\eta^2 \qquad (A.82)$$
$$v = \alpha_9 + \alpha_{10}\xi + \alpha_{11}\eta + \alpha_{12}\xi^2 + \alpha_{13}\xi\eta + \alpha_{14}\eta^2 + \alpha_{15}\xi^2\eta + \alpha_{16}\xi\eta^2$$

As far as shape functions are concerned for corner nodes they should vary as parabolas in ξ and η directions and by definition, they should always have zero values at the midside nodes. For midside nodes the construction is fairly straightforward and using Node 2:

$$N_2 = \tfrac{1}{2}(1-\xi^2)\ (1-\eta) \qquad (A.83)$$

In general for midside nodes with $\xi_i = 0$,

$$N_i = \tfrac{1}{2}(1-\xi^2)\ (1-\eta\eta_i) \qquad (A.84)$$

and for midside nodes with $\eta_i = 0$

$$N_i = \tfrac{1}{2}(1-\xi\xi_i)\ (1-\eta^2) \qquad (A.85)$$

The shape function for node 1 is:

$$N_1 = \tfrac{1}{4}(1-\xi)(1-\eta)(-\xi-1-\eta) \qquad (A.86)$$

In general, the shape function for corner node is given by:

$$N_i = \tfrac{1}{4}(1+\xi\xi_i)(1+\eta\eta_i)(\xi\xi_i+\eta\eta_i-1) \qquad (A.87)$$

6.2 STIFFNESS MATRIX FORMULATION

It has already been shown in Section (2.3.5) that the stiffness matrix of an element can be derived through a virtual work approach. Therefore, the general form of the stiffness matrix is:

$$[K] = \int_{(v)} [B]^T [D] [B] \, d(vol) \qquad (A.88)$$

for two-dimensional problems:

$$[K] = t\int [B]^T [D] [B] \, d(area) \qquad (A.89)$$

The [B] matrix in equation (A.89) expresses strain $\epsilon_x, \epsilon_y, \gamma_{xy}$. i.e.

$$\begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \begin{Bmatrix} u \\ v \end{Bmatrix}$$

or

$$\{\epsilon(x,y)\} = [L] \{\delta(x,y) \qquad (A.90)$$

Substitute the value of equation (2.14) in Chapter 2 with the value of equation (A.90), i.e.

$$\{\epsilon(x,y)\} = [L] [f(x,y)] [A]^{-1} \{\delta^e\} \qquad (A.91)$$

or:

$$\{\epsilon(x,y)\} = [B] \{\delta^e\} \qquad (A.92)$$

Therefore:

$$[B] = [L] [f(x,y)] [A]^{-1} \qquad (A.93)$$

The equation (A.93) can be written in terms of shape function as:

$$[B] = [L]\,[N]$$

i.e.

$$[B] = \begin{vmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{vmatrix} \begin{vmatrix} N_1 & 0 & N_2 & 0 & \ldots \\ \\ 0 & N_1 & 0 & N_2 & \ldots \end{vmatrix} \qquad (A.94)$$

$$= \begin{vmatrix} \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & 0 & \frac{\partial N_8}{\partial x} & 0 \\ 0 & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & 0 & \frac{\partial N_8}{\partial y} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial x} & \frac{\partial N_8}{\partial y} & \frac{\partial N_8}{\partial x} \end{vmatrix} \qquad (A.95)$$

However, at this point a problem arises as the shape functions for an isoparametric element are defined in terms of the curvilinear coordinates ξ and η and, therefore, cannot be differentiated directly with respect to **x** and **y**.

In order to overcome this problem it is necessary to obtain a relationship between the derivatives of the two sets of coordinates. This is obtained by using the chain rule of partial differentiation, i.e.

$$\frac{\partial N}{\partial \xi} = \frac{\partial N}{\partial x}\frac{\partial x}{\partial \xi} + \frac{\partial N}{\partial y}\frac{\partial y}{\partial \xi} \qquad (A.96)$$

$$\frac{\partial N}{\partial \eta} = \frac{\partial N}{\partial x}\frac{\partial x}{\partial \eta} + \frac{\partial N}{\partial y}\frac{\partial y}{\partial \eta}$$

which in matrix form gives:

$$\begin{Bmatrix} \frac{\partial N}{\partial \xi} \\ \frac{\partial N}{\partial \eta} \end{Bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \begin{Bmatrix} \frac{\partial N}{\partial x} \\ \frac{\partial N}{\partial y} \end{Bmatrix} = [J] \begin{Bmatrix} \frac{\partial N}{\partial x} \\ \frac{\partial N}{\partial y} \end{Bmatrix} \qquad (A.97)$$

The matrix [J] relating the derivatives of the two systems is called the Jacobian matrix and its coefficients can be obtained by differentiating equation (A.79), which is written:

$$x = N_1x_1 + N_2x_2 + \text{---}\ N_8x_8$$
$$y = N_1y_1 + N_2y_2 + \text{---}\ N_8y_8$$

Therefore,

$$[J] = \begin{bmatrix} \frac{\partial N_1}{\partial \xi} & \frac{\partial N_2}{\partial \xi} & \cdots\cdots & \frac{\partial N_8}{\partial \xi} \\ \frac{\partial N_1}{\partial \eta} & \frac{\partial N_2}{\partial \eta} & & \frac{\partial N_8}{\eta} \end{bmatrix} \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ x_8 & y_8 \end{bmatrix} \qquad (A.98)$$

or:

$$[J] = \begin{vmatrix} \dfrac{\partial \Sigma N_i x_i}{\partial \xi} & \dfrac{\partial \Sigma N_i y_i}{\partial \xi} \\ \\ \dfrac{\partial \Sigma N_i x_i}{\partial \eta} & \dfrac{\partial \Sigma N_i y_i}{\partial \eta} \end{vmatrix} \qquad (A.99)$$

With [J] determined it is possible to express $\dfrac{\partial N}{\partial x}$ and $\dfrac{\partial N}{\partial y}$ in terms of $\dfrac{\partial N}{\partial \xi}$ and $\dfrac{\partial N}{\partial \eta}$, i.e.

$$\begin{vmatrix} \dfrac{\partial N}{\partial x} \\ \\ \dfrac{\partial N}{\partial y} \end{vmatrix} = [J]^{-1} \begin{vmatrix} \dfrac{\partial N}{\partial \xi} \\ \\ \dfrac{\partial N}{\partial \eta} \end{vmatrix} \qquad (A.100)$$

To complete the transformation between the two coordinate systems it is necessary to express d(area) in terms of $d\xi$ and $d\eta$. It can be shown that:

$$\vec{d\xi} = \begin{bmatrix} \dfrac{\partial x}{\partial \xi} \\ \\ \dfrac{\partial y}{\partial \xi} \end{bmatrix} . d\xi \text{ and } \vec{d\eta} = \begin{bmatrix} \dfrac{\partial x}{\partial \eta} \\ \\ \dfrac{\partial y}{\partial \eta} \end{bmatrix} . d\eta \qquad (A.101)$$

The cross-product of the two vectors is equal to the area of the elemental parallelogram concerned and thus:

$$d(\text{area}) = \vec{d\xi} \times \vec{d\eta}$$

or:

$$d(\text{area}) = \det \begin{vmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial x}{\partial \eta} \\ \dfrac{\partial y}{\partial \xi} & \dfrac{\partial y}{\partial \eta} \end{vmatrix} (d\xi \; d\eta)$$

$$= \det [J] \, d\xi \, d\eta \qquad (A.102)$$

It follows that the integration limits should be changed to ± 1 and the equation (A.89) can now be written as:

$$[K] = t \int_{-1}^{+1} \int_{-1}^{+1} [B]^T [D] [B] \det [J] \, d\xi \, d\eta \qquad (A.103)$$

where $[B]^T$ [D] [B] is a function of ξ and η only.

APPENDIX B

SOME FORMULATIONS OF FINITE ELEMENT

This Appendix involves some basic mathematical formulations of finite elements, such as a way of choosing the displacement functions and a way of obtaining the polynomial constants.

1.0 THE PASCAL TRIANGLE

For displacement functions of two-dimensional elements given in terms of simple polynomials, the Pascal Triangle Figure (B.1) is a useful aid for determining the combination of terms which should be used. The terminology used here is somewhat different from the one used in mathematics which deals with the coefficients of the binomial theorem.

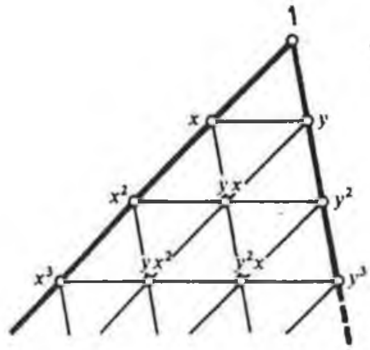
Consider a series of triangles generated on a pattern indicated in Figure (B.2). The number of nodes in each member of the family is now such that a complete polynomial expansion, of the order needed for inter-element compatibility, is ensured. This particular feature puts the triangle family in a special privileged position, in which the inversion of the matrix [C] will always exist.

2.0 LAGRANGE POLYNOMIAL

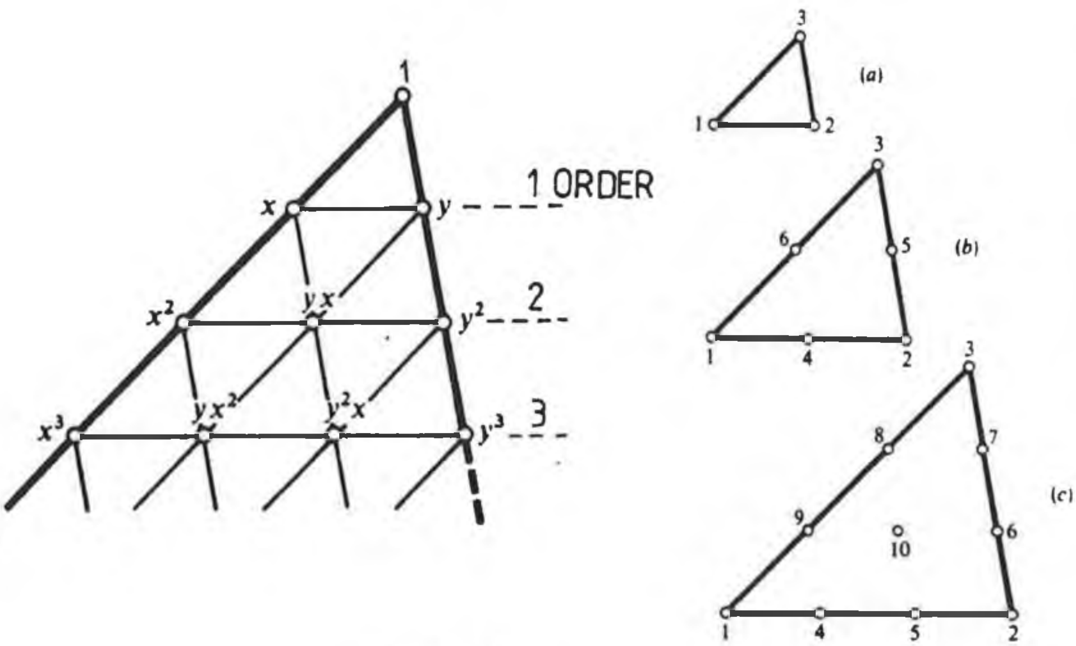
Lagrange polynomials are often used for the construction of shape functions of elements in which only function values but not derivatives are specified at the nodes. The basic form of the Lagrange polynomial in a single coordinate system with **n** nodes is:

$$\delta(x) = \sum_{i=0}^{n} L_i^n(x)\,\delta_i \qquad \text{(B.1)}$$

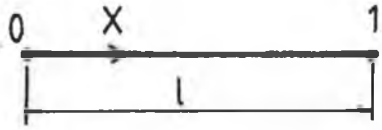
where $L_i^n(x)$ is called the Lagrange multiplier function and is given by



FIG(B.1)



FIG(B.2)



FIG(B.3)

$$L_i^n(x) = \frac{(x-x_0)(x-x_1)\ldots(x-x_{i-1})(x-x_{i+1})\ldots(x-x_n)}{(x_i-x_0)(x_i-x_1)\ldots(x_i-x_{i-1})(x_i-x_{i+1})\ldots(x_i-x_n)} \qquad (B.2)$$

Example (B-1):

Figure (B.3) shows the single coordinate from equation (B.2), i.e.

$$L_0^1(x) = \frac{x - x_1}{0 - x_1} = 1 - \frac{x}{l}$$

$$L_1^1(x) = \frac{x}{x_1} = \frac{x}{l}$$

It is also possible to apply the Lagrange polynomials to shape functions involving two or even three coordinates. Thus the shape function for a two-dimensional problem would be:

$$\delta(x,y) = \sum_{i=0}^{n} \sum_{J=0}^{m} L_i^n(x)\, L_J^m(y)\,\delta_{iy} \qquad (B.3)$$

where **n** and **m** stand for the number of subdivisions or argeements in the **n** and **y** directions respectively. Obviously the Lagrange polynomials in example (B-1) can be utilized in the construction of shape functions for isoparametric elements although it is necessry to shift the coordinate origin from the left end to the centre and change the variables from x/l to $(1+\xi)/2$ in the expressions.

3.0 NODAL DISPLACEMENT PARAMETER POLYNOMIAL CONSTANT RELATIONSHIPS

Since both **u** and **v** have the same shape function it is only necessary to find the relationship. From equation (2.10), i.e.

$$u_a = \alpha_1 + \alpha_2 x_a + \alpha_3 y_a \quad \text{(B.4)}$$

$$u_b = \alpha_1 + \alpha_2 x_b + \alpha_3 y_b \quad \text{(B.5)}$$

$$u_c = \alpha_1 + \alpha_2 x_c + \alpha_3 y_c \quad \text{(B.6)}$$

The polynomial constants can be expressed in terms of the nodal displacements by carrying out Gaussian elimination as follows:

(i) Eliminate α_1 by performing equation (B.5) - equation (B.4) and equation (B.6) - equation (B.4),

$$u_b - u_a = (x_b - x_a)\,\alpha_2 + (y_b - y_a)\,\alpha_3 \quad \text{(B.7)}$$

$$u_c - u_a = (x_c - x_a)\,\alpha_2 + (y_c - y_a)\,\alpha_3 \quad \text{(B.8)}$$

(ii) Eliminate α_2 by performing the calculations of equation (B.8) x $(x_b - x_a)$ - equation (B.7) x $(x_c - x_a)$.

$$(x_c - x_b)\,u_a + (x_a - x_c)\,u_b + (x_b - x_a)\,u_c$$
$$= \{(x_b - x_a)(y_c - y_a) - (x_c - x_a)(y_b - y_a)\}\,\alpha_3$$

or:

$$\alpha_3 = \frac{1}{2\Delta}(x_{cb}u_a + x_{ac}u_b + x_{ba}u_c) \quad \text{(B.9)}$$

Where $x_{cb} = (x_c - x_b)$ and $y_{ba} = (y_b - y_c)$, etc. and Δ is in fact, the area of triangle **abc** and can be computed (see Appendix C) and equation (2.6) in Chapter 2.

(iii) Compute α_2 by substituting equation (B.9) with equation (B.7),

$$\alpha_2 = \frac{1}{2\Delta}(y_{bc}\,u_a + y_{ca}\,u_b + y_{ab}\,u_c) \quad \text{(B.10)}$$

(iv) Compute α_1 by substituting equation (B.9) and equation (B.10) with equation (B.4),

$$\alpha_1 = \frac{1}{2\Delta}[(x_b y_c - x_c y_b)u_a + (x_c y_a - x_a y_c)u_b + (x_a y_b - x_a y - x_b y_a)u_c] \quad (B.11)$$

Writing equations (B.11),(B.10) and (B.9) in matrix form:

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \frac{1}{2\Delta} \begin{vmatrix} x_b y_c - x_c y_a & x_c y_a - x_a y_c & x_a y_b - x_b y_a \\ y_{bc} & y_{ca} & y_{ab} \\ x_{cb} & x_{ac} & x_{ba} \end{vmatrix} \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} \quad (B.12)$$

The expressions in equation (B.12) can be considered by introducing the notations given in equation (2.5), Section (2.2.1) and equation (B.12) now becomes:

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \frac{1}{2\Delta} \begin{vmatrix} a_a & a_b & a_c \\ b_a & b_b & b_c \\ c_a & c_b & c_c \end{vmatrix} \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} \quad (B.13)$$

and a similar equation can be written for v displacements

$$\begin{bmatrix} \alpha_4 \\ \alpha_5 \\ \alpha_6 \end{bmatrix} = \frac{1}{2\Delta} \begin{vmatrix} a_a & a_b & a_c \\ b_a & b_b & b_c \\ c_a & c_b & c_c \end{vmatrix} \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} \quad (B.14)$$

The $[A]^{-1}$ matrix, unlike in Section (2.3.2) in Chapter 2, must now be constructed from equation (B.13) and (B.14)

$$[A]^{-1} = \frac{1}{2\Delta} \begin{vmatrix} a_a & 0 & a_b & 0 & a_c & 0 \\ b_a & 0 & b_b & 0 & b_c & 0 \\ c_a & 0 & c_b & 0 & c_c & 0 \\ 0 & a_a & 0 & a_c & 0 & a_c \\ 0 & b_a & 0 & b_c & 0 & b_c \\ 0 & c_a & 0 & c_c & 0 & c_c \end{vmatrix} \quad (B.15)$$

4.0 DISPLACEMENT FUNCTIONS IN SHAPE FUNCTION FORM

From the previous cases it was shown that

$$\{\delta(x,y)\} = [f(x,y)]\,[A]^{-1}\,\{\delta^e\}$$

$$= [N]\,\{\delta_3\} \qquad (B.15)$$

where:

[N] is the shape function

$$[N] = [f(x,y)]\,[A]^{-1} \qquad (B.16)$$

For the constant strain triangle:

$$[N] = \frac{1}{2\Delta}\begin{vmatrix} a_a+b_ax+c_ay & 0 & a_b+b_bx+c_by & 0 & a_c+b_cx+c_cy & 0 \\ 0 & a_a+b_ax+c_ay & 0 & a_a+b_bx+c_by & 0 & a_c+b_cx+c_cy \end{vmatrix}$$

$$[N] = \begin{vmatrix} L_a & 0 & L_b & 0 & L_c & 0 \\ 0 & L_a & 0 & L_b & 0 & L_c \end{vmatrix} \qquad (B.17)$$

Where:

L_a, L_b, L_c are the area coordinates and the shape functions are simply the area coordinates, thus:

$$N_a = L_a, \quad N_b = L_b, \quad N_c = L_c \qquad (B.18)$$

and the displacement equations can be written in this form:

$$u = N_a u_a + N_b u_b + N_c u_c \qquad \text{(B.19)}$$
$$v = N_a v_a + N_b v_b + N_c v_c$$

This is obvious as each individually gives unity at one node, zero at others and varies linearly everywhere.

APPENDIX C

AN INTEGRATION FORMULAE FOR A TRIANGLE (FIGURE 2)

Let a triangle be defined in the **x-y** plane by three points (x_a, y_a), (x_b, y_b), (x_c, y_c) with the origin at the coordinates taken at the centreoid, i.e.

$$\frac{x_a + x_b + x_c}{3} = \frac{y_a + y_b + y_c}{3} = 0$$

Then integrating over the triangle area:

$$\int d_x d_y = \frac{1}{2} \begin{vmatrix} 1 & x_a & y_a \\ 1 & x_b & y_b \\ 1 & x_c & y_c \end{vmatrix} = \Delta = \text{area of triangle}$$

APPENDIX (D)

1.0 MATRICES

A matrix is an array of terms as shown in equation (D.1) below. The terms may be pure numbers, constants or variables

$$[A] = \begin{vmatrix} a_{11} & a_{12} & a_{13} & \cdots\cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & & a_{2n} \\ a_{31} & a_{32} & a_{32} & & a_{3n} \\ \cdot & & & & \\ \cdot & & & & \\ \cdot & & & & \\ a_{m1} & a_{m2} & a_{m3} & & a_{mn} \end{vmatrix} \qquad (D.1)$$

The matrix has **m** rows and **n** columns and is said to be a **mxn** matrix. If **m** = **n**, the matrix is called a square matrix. If **n=1** the matrix consists of single column and is usually called a column vector. If **m=1** the matrix has a single row and is called a row vector. The element a_{ii} lies on the leading diagonal.

A set of linear equations can be represented in matrix form as follows:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3 \end{aligned}$$

or:

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

5.2 MATRIX OPERATIONS

5.2.1 Transpose matrix

The transposition of matrix [A] is denoted by $[A]^T$ and is obtained by interchanging rows and columns. Using equation (D.1) as an example:

$$[A]^T = \begin{vmatrix} a_{11} & a_{21} & a_{31} & \cdots & a_{m1} \\ a_{12} & a_{22} & a_{32} & \cdots & a_{m2} \\ a_{13} & a_{23} & a_{33} & \cdots & a_{m3} \\ \cdot & & & & \\ \cdot & & & & \\ \cdot & & & & \\ a_{in} & a_{2n} & a_{3n} & & a_{mn} \end{vmatrix}$$

5.2.2 Addition and subtraction

Matrices can only be added or subtracted if they are of the same order. The process consists of adding or subtracting corresponding terms, e.g.

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \pm \begin{vmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{vmatrix} = \begin{vmatrix} (a_{11} \pm b_{11}) & (a_{12} \pm b_{12}) \\ (a_{21} \pm b_{21}) & (a_{22} \pm b_{22}) \end{vmatrix}$$

5.2.3 Multiplication

Two matrices may be multiplied by each other only if the number of columns in the first is equal to the number of rows in the second. The terms in the product matrix are obtained by taking the solar product of each row of matrix [A] with each column in matrix [B], e.g.

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \begin{vmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{vmatrix} = \begin{vmatrix} (a_{11}b_{11}+a_{12}b_{21}) & (a_{11}b_{12}+a_{12}b_{22}) \\ (a_{21}b_{11}+a_{22}b_{21}) & (a_{21}b_{12}+a_{22}b_{22}) \end{vmatrix}$$

5.2.4 Transposition of product

The transposition of a matrix product of matrix [A] and [B] is equal to the product of $[B]^T$ and $[A]^T$, i.e.

$$([A]\ [B])^T = [B]^T\ [A]^T$$

5.2.5 Determinant

The determinant of matrix [A] is denoted det [A]. For n=1 and n=2 we have these definitions:

$$\det [a] = a \quad ; \quad \det \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

5.2.6 The cofactors and minors

The minor of the element in the ith row and jth column is the determinant of the (n-1) by (n-1) matrix when row i and column j are deleted. The cofactor of a_{ij} is denoted by A_{ij} and it is $(-1)^{i+j}$ times the minor of a_{ij}.

5.2.7 Matrix inversion of a square matrix

The inversion of a matrix [A] is denoted by $[A]^{-1}$ and satisfies the relation

$$[A]^{-1}\ [A] = [I]$$

where [I] is an identify matrix. An identify matrix consists of unity values down the leading diagonal with zeros elsewhere, e.g. for 3 x 3 matrix

$$[I] = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

Matrix inversion is analogous process to the division of a scalar quantity. To invert a matrix the adjoint of the matrix adj[A] has to be defined.

adj[A] = transposed matrix of cofactors

The inverse matrix is given by

$$[A]^{-1} = \frac{adj[A]}{det[A]}$$

If det[A] = 0 the matrix is said to be singular.

The inverse does not exist.

To find the det[A] for a matrix [A] of order 3, each element of the first row should be multiplied by its cofactor and added together.

5.3 Symmetric matrix

A square matrix in which $a_{ij} = a_{ji}$ is called a symmetric matrix.

APPENDIX (E)

ASSEMBLY OF OVERALL STIFFNESS MATRIX

Consider the following structure shown in Figure (E.1(a)) in which the definitions of the triangle are:

Element No.	Node i	Node J	Node m
1	1	2	3
2	2	4	3
3	2	5	4

If the structure is split up into its component elements and the external forces are divided between the appropriate elements in Figure (E.1(b)), then it is obvious that from joint equilibrium:

$$\text{Node 1} \begin{cases} p'^{1}_{x1} = P_{x1} \\ p'^{1}_{y1} = 0 \end{cases}$$

$$\text{Node 2} \begin{cases} p'^{1}_{x2} + p'^{2}_{x2} + p'^{3}_{x2} = 0 \\ p'^{1}_{y2} + p'^{2}_{y2} + p'^{3}_{y2} = P_{y2} \end{cases}$$

$$\text{Node 3} \begin{cases} p'^{1}_{x3} + p'^{2}_{x3} = P_{x3} \\ p'^{1}_{y3} + p'^{2}_{y3} = P_{y3} \end{cases} \qquad (E.1)$$

$$\text{Node 4} \begin{cases} p'^{2}_{x4} + p'^{3}_{x4} = P_{x4} \\ p'^{2}_{y4} + p'^{3}_{y4} = 0 \end{cases}$$

$$\text{Node 5} \begin{cases} p'^{3}_{x5} = P_{x5} \\ p'^{3}_{y5} = 0 \end{cases}$$

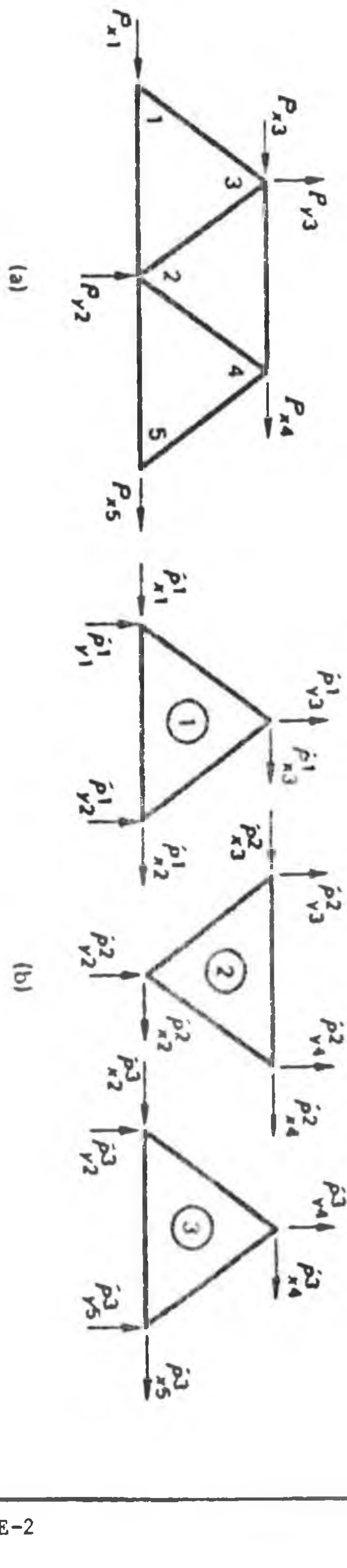


FIG.(E.1)

From element equilibrium and displacement it can be written:

$$\begin{vmatrix} k^1{}_{ii} & k^1{}_{ij} & k^1{}_{im} \\ k^1{}_{ji} & k^1{}_{jj} & k^1{}_{jm} \\ k^1{}_{mi} & k^1{}_{mj} & k^1{}_{mm} \end{vmatrix} \begin{Bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{Bmatrix} = \begin{Bmatrix} p^1{}_1 \\ p^1{}_2 \\ p^1{}_3 \end{Bmatrix} \text{ for element 1}$$

$$\begin{vmatrix} k^2{}_{ii} & k^2{}_{ij} & k^2{}_{im} \\ k^2{}_{ji} & k^2{}_{jj} & k^2{}_{jm} \\ k^2{}_{mi} & k^2{}_{mj} & k^2{}_{mm} \end{vmatrix} \begin{Bmatrix} \delta_2 \\ \delta_4 \\ \delta_3 \end{Bmatrix} = \begin{Bmatrix} p^2{}_2 \\ p^2{}_4 \\ p^2{}_3 \end{Bmatrix} \text{ for element 2} \qquad (E.2)$$

$$\begin{vmatrix} k^3{}_{ii} & k^3{}_{ij} & k^3{}_{im} \\ k^3{}_{ji} & k^3{}_{jj} & k^3{}_{jm} \\ k^3{}_{mi} & k^3{}_{mm} & k^3{}_{mm} \end{vmatrix} \begin{Bmatrix} \delta_2 \\ \delta_5 \\ \delta_4 \end{Bmatrix} = \begin{Bmatrix} p^3{}_2 \\ p^3{}_5 \\ p^3{}_4 \end{Bmatrix} \text{ for element 3}$$

In which the superscripts refer to the element numbers, and $K^1{}_{ii}$, δ_1, $p^1{}_1$, etc. are submatrices involving **x** and **y** components. By substituting the value of equation (E.2) with the value of equation (E.1) gives

$$k^1{}_{ii}\delta_1 + k^1{}_{ij}\delta_2 + k^1{}_{im}\delta_3 = p^1{}_1 = \begin{Bmatrix} P_{x1} \\ 0 \end{Bmatrix}$$

$$k^1{}_{ji}\delta_1 + (k_1jj + k^2{}_{ii} + k^3{}_{ii})\delta_2 + (k^1{}_{jm} + k^2{}_{im})\delta_3 + (k^2{}_{ij} + k^3{}_{im})\delta_4$$

$$+ k^3{}_{ij}\delta_5 = P^1{}_2 + P^2{}_2 + P^3{}_2 = \begin{Bmatrix} 0 \\ P_{y2} \end{Bmatrix}$$

$$k^1{}_{mi}\delta_1 + (k^1{}_{mj} + k^2{}_{mi})\delta_2 + (k^1{}_{mm} + k^2{}_{mm})\delta_3 + k^2{}_{mj}\delta_4 = P^1{}_3 + P^2{}_3 =$$

$$\begin{Bmatrix} P_{x3} \\ P_{y3} \end{Bmatrix}$$

$$(k^2{}_{ji} + k^3{}_{mi})\delta_2 + k^2{}_{jm}\delta a43 + k^2{}_{jj} + k^3{}_{mm})\delta_4 + k^3{}_{mj}\delta_5 = P^2{}_4 + P^3{}_4 = \begin{bmatrix} P_{x4} \\ 0 \end{bmatrix}$$

$$k^3{}_{ji}\delta_2 + k^3{}_{jm}\delta_4 + k^3{}_{jj}\delta_5 = P^3{}_5 = \begin{bmatrix} P_{x5} \\ 0 \end{bmatrix}$$

or

$$\left|\begin{array}{ccccc}
k^1{}_{ii} & k^1{}_{ij} & k^1{}_{im} & & \\
\hline
k^1{}_{ji} & k^1{}_{jj} + k^2{}_{ii} + k^3{}_{ii} & k^1{}_{jm} + k^2{}_{im} & k^2{}_{ij} + k^3{}_{im} & k^3{}_{ij} \\
\hline
k^1{}_{mi} & k^1{}_{mj} + k^2{}_{mi} & k^1{}_{mm} + k^2{}_{mm} & k^2{}_{mj} & \\
\hline
 & k^2{}_{ji} + k^3{}_{mi} & k^2{}_{jm} & k^2{}_{jj} + k^3{}_{mm} & k^3{}_{mj} \\
\hline
 & k^3{}_{ji} & & k^3{}_{jm} & k^3{}_{jj}
\end{array}\right| \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \\ \delta_5 \end{bmatrix} = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \end{bmatrix} \qquad (E.3)$$

Where:

- $\{P\}$ the overall load matrix
- P'_{x1}, P'_{x2}. P'_{x3}. P'_{xy}, P'_{x}. – are the equilibriating forces.

The above process describes the theoretical basis of combining the individual element equations to form the overall structure equations.

APPENDIX (F)

LISTING OF E.P. PROGRAM WITH THE USER'S MANUAL

1.0 PROGRAM VARIABLE NAMES

The following are some computer program variable names used in connection with E.P. Program:

1.1 MAIN PROGRAM

Command block

TITLE (12	Title of the problem
IIF	Number of restricted points
FTx, FTy FX1, of FY1	The spans of X and Y axes which refer to the graph structure before and after loading
NIH	Temporary store for IPR and IPI
NP	Number of points
NE	Number of elements
NSZF	Number of degrees of freedom in structure
TOL	Solution convergence tolerance
DY	Current increment
YTOT	Load increment
IUZ	Temporary store of the lowest point of the outer ring
NROWS	Number of rows in the total structure-displacement vector which do not contain prescribed displacement
FN	The total force exerted at the anvil/workpiece interface resulting from the displacement value prescribed
G	Shear modulus
CE	The first load increment size
BE	The final load increment size
IH	The number of the highest point in "Y" direction (reference coordinate)

LF	Number of the highest point in "Y" direction (reference coordinate)
IPR,IPI	Number of the longest point in "X" direction (reference coordinate)
NX(50)	Number of intervals
JJF	Number of the loaded points
JIO	Print control variable: JIO = 0 print input data JIO = 1 does not print input data
NIO	Option control variable for plane elasticity: NIO = 1 plane strain NIO = 2 plane stress
MIO	Option control variable for load condition MIO = 1 concentrating load MIO = 2 ring case (2) MIO = 3 ring case (3) MIO = 4 uniform distributed load
$P_I(3,3)$	Elasticity matrix for plane stress
ISC	Option control variable for drawing ISC = 1 draw the case studied before and after loading beside each other ISC = 2 draw the case studied before and after loading at each other
DISPL(600)	Vector of nodal displacement
BOD(600)	Vector of temporary store of displacement
REC(600)	Vector of temporary store of reaction
IOPT	Option control variable for solution: IOPT = 1 "prescribed force" IOPT = 2 prescribed displacement
DSY	Option control variable for drawing: DSY = 1. draw structure before loading DSY > 1. draw structure after loading (This can be done automatically inside the program)
IZO	Option control variable for automatic mest variable: IZO = 1 automatic mesh generating for nodal points and element definition

	IZO = 2 read number of points, number of elements, nodal coordinates and element definition from input command
IOT	Option control variable for output
	IOT = 1 print output of the displacement, the final coordinate, stress, strain, and reaction force
	IOT = 2 print output of displacement and reaction force
	IOT = 3 print output of stress and strain
	IOT = 4 print output of the normal strain on the inner surface of the ring and the displacement
NV	Number of pairs of values giving information of the number and size of load increment.
TG(12)	The title of graph
KF	Stop variable:
	KF = 0 program is continuing
	KF = 1 program is stopping
NIN(10)	Vector containing the number of increments associated with a particular increment size
XIN(10)	Vector of different increment size to be applied. This vector is used in association with NIN(10).
LPIN	Temporary store of the number of increments
INLP	Counter of LPIN
CORD(600,2)	Stores of x and y coordinates
NOP(600,3)	Stores of the elements definition
PROP (5)	Material propoeties: vector:
	PROP(1): Young's modulus
	PROP(2): Poison's modulus
	PROP(3): material yield stress
	PROP(4): slope of equivalent stress/strain curve
	PROP(5): thickness of the material
D (3,3)	Elasticity matrix for plane strain
SK(600,600)	Stiffness matrix of the structure

BOUN(600)	Vector of boundary condition: BOUN ( ) = 1.0 for freedom BOUN ( ) = 0.0 for restrained
TAREA(600)	Vector of value of element area multiplied by element thickness
JROW (600)	Vector containing the number of the rows of the total displacement vector
FORC (600)	Overall load vector
ETA (600,4)	Total strain vector
SIG (600,4)	Total stress vector
JCOL (600)	Vector containing the number of the rows of the total displacement vector
SEQI (600)	Total equivalent stress vector at the start of a load increment
SEQF (600)	Total equivalent stress vector at the end of a load increment
ELETA (600,4)	Vector of elastic strain increment
ELSIG (600,4)	Vector of elastic stress increment
DIS (600,2)	Stores of x and y displacements
LYD (400)	Vector has expressed the situation of the elements in terms of elastic or plastic: LYD ( ) = 1 the element has yielded LYD ( ) = 0 the element still elastic
TION (600,2)	Vector containing the x and y reaction for at each node
EEQ (600)	Equivalent strain vector when IOT = 1,3
EEQ (600)	Normal strain on the inner surface of the ring when IPT = 4
ESTIFM (6,6)	Individual element stiffness matrix
Unit = 3, Unit = 7, Unit = 8	Logical channel numbers

1.2 SUBROUTINE GDAT

YL (50)	Vector containing the y coordinates of starting points of generating lines

XL (50)	Vector containing the x coordinates of starting points of generating lines
XF (50)	Vector containing the x coordinates of the ending points of generating lines
YF (50)	Vector containing the y coordinates of the ending points of generating lines
NOD (4)	Vector containing a temporary store of element definition
SUM1 (40)	Vector containing a temporary store of coordinate factors

1.3 SUBROUTINE STIFT(N)

B (3,6)	Matrix relating element strains to element nodal displacement
H (3,6)	Matrix resulting from multiplication of D (3,3) x B (3,6)

1.4 SUBROUTINE SOLV 3

SUBK (600,600)	Transition matrix enabling the overall stiffness matrix to be re-formed in partitions
UPRES (600)	Vector containing the values of prescribed displacements UPRES ( ) = 00 at restrained UPRES ( ) = 1 at non-restrained

1.5 SUBROUTINE GAUSS

Lp (600)

Lq (600,2)

R (600) are: Transition matrices enabling the inversion of appropriate partition of the overall stiffness matrix

1.6 SUBROUTINE SOLV 4

U (600)	Vector of unknown displacement
R (6)	Vector of element displacement value

1.7 SUBROUTINE SOLV 1

SS (600)	Vector containing a temporary store of the applied load
SL (600,600)	Temporary store of stiffness of structure

1.8 SUBROUTINE TOTAL

X_1 (2500), Y_1 (2500), X_2 (2500), Y_2 (2500), X_3 (2500), Y_3 (2500)	The temporary store of the nodal coordinate before loading
DSEQ	Incremental value of equivalent stress for each interation
Unit = 12, Unit = 14	The logical channel numbers

1.9 SUBROUTINE PLAST

FMAT (4)	Corresponds to vector $\{\frac{\partial F}{\partial \sigma}\}$
GFMT (4)	Vector of differentation
DEVX, DEVY, DEVZ	Deviatoric stress values

FTM (3), STM (3,3), TTM (3,3)	Sub-matrices used in calculation of elastic-plastic matrix
DELPL (3,3)	The elastic-plastic matrix
SINC (4)	Vector of plastic stress increment
SBF (4)	Corresponds to vector of stress to be supported by body forces
BF (6)	Vector of equilibrating body forces for an element
REAC (600)	Load in which all body forces are assembled
X_{11} (2500), Y_{11} (2500), X_{22} (2500), Y_{22} (2500), X_{33} (2500), Y_{33} (2500)	The temporary store of nodal coordinates after loading

1.10 SUBROUTINE GRAPH

X (2500), Y (2500)	Temporary store of x y coordinate of the nodal points
Xc (2500), Yc (2500), Xce (2500), Yce (2500), Xc1 (2500), Yc1 (2500)	Temporary store of the x-y coordinates of the centre of the elements before and after loading

2.0 INSTRUCTIONS FOR PREPARING DATA

The instructions are provided here for the E.P. Computer program, which is developed in the current work. The input data for running the program is put in input command with free format. This was done to save the time for preparing data. The input command consists of:

CARD SET 1 one card (18A4)

Title: Title of the problem

CARD SET 2 one card (I5, 2F10.4, 4I5, 8I3)

IIF	Number of restrcited points
FTX	The span of the x-axix for drawing the case studied
FTY	The span of the y-axix for drawing the case studied
IH	The number of the highest point in y direction
LF	The number of the highest point in y direction
IPR	The number of the longest point in x direction
IPI	The number of the longest point in x direction
JJF	Number of the loaded poind
JIO	Print control variable: O - print input data I - does not print input data
NIO	Option control variable for plane elasticity: 1 - plane strain 2 - plane stress
MIO	Option control variable for load condition: 1 - concentrating load 2 - ring case (2) 3 - ring case (3) 4 - uniform distributed load
ISC	Option control variable for drawing: 1 - draw the case studied before and after loading, beside each other 2 - draw the case studied before and after loading, at each other

IOPT	Option control variable for solution:
	1 - prescribed force
	2 - prescribed displacement
IZO	Option control variable for mesh variable:
	1 - automatic mesh generating
	2 - read number of points, number of elements, nodal coordinates and element definition from the input command
IOT	Option control variable for output
	IOT = 1 print output of the displacement, the final coordinate, stress, strain and reaction force
	IOT = 2 print output of displacement and reaction force
	IOT = 3 print output of stress and strain
	IOT = 4 print output of the normal strain on the inner surface of the ring and the displacement

<u>CARD SET 3</u> one card (E10.4, F10.2, E10.4, 2F10.4)

PROP (1)	Young's modulus
PROP (2)	Poison's modulus
PROP (3)	Material yield stress
PROP (4)	Slope of equivalent stress/strain curve
PROP (5)	Thickness of the material

<u>CARD SET 4</u> one card (2F10.4)

CE	The first load increment size
BE	The final load increment size

<u>CARD SET 5</u> one card (I5, F10.4)

NV	Number of pairs of value giving in formation of the number and size of load increment
TOL	Solution convergence tolerance

CARD SET 6	one card for each NV in Card Set 5 (I5, F10.4)
NIN	Vector containing the number of increments associated with a particular increment size
XIN	Vector of different increment sizes to be applied, this vector is used in association with NIN (10)

CARD SET 7	one card for each IIF in Card Set 1 (I5, 2F10.2)
K	Number of restricted node
U	Boundary condition in x direction
V	Boundary condition in y direction

NOTE:

U = 0.0	The point is restricted in x direction
U = 1.0	The point is free in x direction
V = 0.0	The point is restricted in y direction
V = 1.0	The point is free in y direction

CARD SET 8	one card for each JJF in Card Set 1 (I5,2E16.8)
K	The number of loaded points
Fx	The external applied load in x direction
Fy	The external applied load in y direction

NOTE: If the signal of Fx or Fy positive,that means tension force
If the signal of Fx or Fy negative, that means compression force

CARD SET 9 one card (I5, F10.5)

NY	The number of generating lines
CON1	A weighting factor

CARD SET 10 one card for each NY in Card Set 9 (I5, 4F10.5)

NX	Number of intervals
XF	The x coordinates of the starting point of the generating line
YF	The y coordinates of the starting point of the generating line
XL	The x coordinates of the final point of the generating line
YL	The y coordinates of the final point of the generating line

CARD SET 11 one card (18A4)

TG	Title of the graph

NOTES:

1. If MIO, in Card Set 1 equals four, then Card Set 9 becomes one card for each JJF in Card Set 1 (2I5, E16.8)

K	Number of first loaded point by uniform distributed load
NY	Number of the following loaded points by uniform distributed load
ZY	The uniform distributed load between the previous two points, K and NY

2. When MIO = 4 in last note, JJR in Card Set 1 becomes the number of the loaded edges by external uniform distributed load.

3. If IOPT = 2, in Card Set 1, then Card Set 3 becomes one card for each number of loaded point JJF in Card Set 1 (I5, 2E16.8) with:

   Fx The prescribed displacement in x direction

   Fy The prescribed displacement in y direction

4. If IZO = 2 in Card Set 1, then Card Set 9 becomes one card (2I5)

   NP Number of the total points in the structure

   NE Number of the elements in the structure

   CARD SET 10 one card for each NP in the previous card, Card Set 9 (I10, 2F16.8)

   N Number of points

   CORD The x and y coordinates for each point in the structure

   CARD SET 11 one card for each NE in Card Set 9 (I5, 3I10)

   L Number of elements

   NOP The element definition for each element in the structure

   CARD SET 11 one card (18A4)

   TG Title of the graph

5. For the graph representation of the structure, the first nodal coordinate should be taken greater than zero. In others words, the graph should be started from a point in x - y plane not from the original coordinates.

6. When IOPT = 3 the variable NIN(3) should have different values during the execution, i.e. NIN(1) $\neq$ NIN(2) $\neq$ NI(3).

(Table (F.1) shows two different types of input data)

3.0 LISTING OF THE E.P. PROGRAM

```
C *****************************************************
C =====================================================
C *****************************************************
C *                                                   *
C *                                                   *
C * COMPILER :FOR FILENAME * LINK FILENAME,CALCOMP/LIB *
C *                                                   *
C *                                                   *
C *            *** THE MASTER PROGRAM ***             *
C *                                                   *
C *                                                   *
C * DATE: 10-5-1989       WRITTEN BY: MOHAMAD M. SALEH *
C *                                                   *
C *****************************************************
C =====================================================
C *****************************************************
C       CONTROL MAIN PROGRAM
        COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1  TOL,DY,YTOT,IUZ,
     2  NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3  MIO,D1(3,3),
     4  FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5  DSY,IZO,IOT
        COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1  LPIN,INLP
        COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1  FX1,FY1,NIH,
     2  D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3  FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4  SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5  LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
        KF=0
C
C                    READ INPUT DATA
C
            CALL GDAT
        OPEN(UNIT=3,FILE='SCRATCH:FINITE.DAT',STATUS='OLD')
            CALL GDATA
        CLOSE(UNIT=3,STATUS='KEEP')
C
C                  CREATE MESH
C
        IF(IZO.NE.1) GO TO 20
        OPEN(UNIT=7,FILE='SCRATCH:COOR.DAT',STATUS='OLD')
        CALL COORGEN
        CLOSE(UNIT=7,STATUS='KEEP')
        OPEN(UNIT=8,FILE='SCRATCH:ELGEN.DAT',STATUS='OLD')
        CALL ELGEN
        CLOSE(UNIT=8,STATUS='KEEP')
20      CONTINUE
        NSZF=NP*2
C
```

```
C                    LOOP THROUGH INCREMENTS
        DO 600 N=1,NV
        LPIN=NIN(N)
        DY=XIN(N)
        DO 500 INLP=1,LPIN
C
C                  DISPLACE WORKPIECE UNDER PRESSURE
C
        OPEN(UNIT=3,FILE='SCRATCH:FINITE.DAT',STATUS='OLD')
        CALL LOAD
        CLOSE(UNIT=3,STATUS='KEEP')
C
C                  FORM AND SOLVE EQUATIONS
C
        CALL FORMK
        IF(IOPT.NE.1) GO TO 60
        OPEN(UNIT=4,FILE='SCRATCH:MATFIN.DAT',STATUS='OLD')
        CALL SOLV1
        CLOSE(UNIT=4,STATUS='KEEP')
        CALL TOTAL
        OPEN(UNIT=4,FILE='SCRATCH:MATFIN.DAT',STATUS='OLD')
        CALL PLAST
        CLOSE(UNIT=4,STATUS='KEEP')
        GO TO 65
C
C                 CALCULATE STRESSES
C
60      CONTINUE
        CALL SOLV3
        CALL GAUSS
        OPEN(UNIT=4,FILE='SCRATCH:MATFIN.DAT',STATUS='OLD')
        CALL SOLV4
        CLOSE(UNIT=4,STATUS='KEEP')
        CALL TOTAL
C
C
C               ELASTIC-PLASTIC ANALYSIS
C
C
        OPEN(UNIT=4,FILE='SCRATCH:MATFIN.DAT',STATUS='OLD')
        CALL PLAST
        CLOSE(UNIT=4,STATUS='KEEP')
65      CONTINUE
C
C
C                 STORE OUTPUT
C
C
        CALL OUTPUT
        IF(ABS(YTOT).LT.BE) GO TO 300
        CALL GRAPH
300     CONTINUE
        IF(KF.EQ.1) GO TO 700
500     CONTINUE
600     CONTINUE
```

```
700     STOP
        END
C *******************************************************
C *                       SUBROUTINE GDAT
C *******************************************************
        SUBROUTINE GDAT
        COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1  TOL,DY,YTOT,IUZ,
     2  NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3  MIO,D1(3,3),
     4  FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5  DSY,IZO,IOT
        COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1  LPIN,INLP
        COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1  FX1,FY1,NIH,
     2  D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3  FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4  SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5  LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
        DIMENSION YL(50),XF(50),YF(50),NOD(4),SUM1(40),
     1  XL(50)
C
C                       READ AND PRINT TITLE
C
        WRITE(6,4)
        READ(5,3)TITLE
3       FORMAT(18A4)
        READ(5,*) IIF,FTX,FTY,IH,LF,IPR,IPI,JJF,JIO,NIO,MIO,
     1  ISC,IOPT,IZO,IOT
C
C                       READ AND PRINT MATERIAL DATA
C
        READ(5,*) (PROP(J),J=1,5)
C
C                       READ AND PRINT INCREMENT DATA
C
        READ(5,*) CE,BE
        READ(5,*) NV,TOL
        READ(5,*) (NIN(J),XIN(J),J=1,NV)
C
C                       READ AND PRINT OUTPUT CONTROL DATA
C
        OPEN(UNIT=3,FILE='SCRATCH:FINITE.DAT',STATUS='NEW')
        WRITE(3,5) TITLE
        WRITE(3,8) IIF,FTX,FTY,IH,LF,IPR,IPI,JJF,JIO,NIO,MIO,
     1  ISC,IOPT,IZO,IOT
        WRITE(3,7) (PROP(J),J=1,5)
        WRITE(3,13) CE,BE
        WRITE(3,6) NV,TOL
        WRITE(3,11) (NIN(J),XIN(J),J=1,NV)
        IF(MIO.EQ.4) GO TO 18
        DO 16 NC=1,IIF
        READ(5,*) K,U,V
16      WRITE(3,21) K,U,V
```

```
      DO 17 NB=1,JJF
      READ(5,*) K,FX,FY
      WRITE(3,14) K,FX,FY
17    CONTINUE
      GO TO 19
18    DO 30 I=1,IIF
      READ(5,*) K,U,V
30    WRITE(3,21) K,U,V
      DO 32 JJ=1,JJF
      READ(5,*) K,NY,ZY
32    WRITE(3,22) K,NY,ZY
19    CONTINUE
      IF(IZO.EQ.1) GO TO 28
      DSY=1.
      READ(5,*) NP,NE
      WRITE(3,1006) NP,NE
      DO 105 J=1,NP
      READ(5,*) N,(CORD(J,M),M=1,2)
      WRITE(3,1002) N,(CORD(J,M),M=1,2)
105   CONTINUE
      DO 110 K=1,NE
      READ(5,*) L,(NOP(K,MM),MM=1,3)
      WRITE(3,1003) L,(NOP(K,MM),MM=1,3)
110   CONTINUE
      READ(5,119) TG
28    CONTINUE
      CLOSE(UNIT=3,STATUS='KEEP')
      IF(IZO.NE.1) GO TO 450
C
C                      READ AND PRINT COORDINATE DATA
C
      READ(5,*) NY,CON1
      DO 100 I=1,NY
      READ(5,*) NX(I),XF(I),YF(I),XL(I),YL(I)
100   CONTINUE
      READ(5,119) TG
      OPEN(UNIT=7,FILE='SCRATCH:COOR.DAT',STATUS='NEW')
      WRITE(7,1001) NY,CON1
      DO 101 I=1,NY
101   WRITE(7,1000) NX(I),XF(I),YF(I),XL(I),YL(I)
C
C                      GENERATE POINTS COORDINATE
C
      N=0
      DO 350 I=1,NY
      NXI=NX(I)+1
      SUM1(1)=0.0
      SUM1(2)=1.0
      SUM=1.0
      IF(NXI-2) 190,291,190
190   DO 250 K=3,NXI
      SUM1(K)=SUM1(K-1)*CON1
      SUM=SUM+SUM1(K)
250   CONTINUE
291   CONTINUE
```

```
      X=XF(I)
      Y=YF(I)
      DO 300 J=1,NXI
      N=N+1
      X=(XL(I)-XF(I))*SUM1(J)/SUM+X
      Y=(YL(I)-YF(I))*SUM1(J)/SUM+Y
      WRITE(7,1002) N,X,Y
300   CONTINUE
350   CONTINUE
      NP=N
      CLOSE(UNIT=7,STATUS='KEEP')
C
C                     GENERATE AND PRINT ELEMENTS
C
      OPEN(UNIT=8,FILE='SCRATCH:ELGEN.DAT',STATUS='NEW')
      M=0
      NSUM=0
      NYI=NY-1
      DO 600 I=1,NYI
      NXI=NX(I)
      DO 500 J=1,NXI
      IF(J-NXI) 379,371,379
371   IF(NX(I+1)-NX(I)) 380,379,401
379   NOD(1)=J+NSUM
      NOD(2)=NOD(1)+1
      NOD(3)=NOD(2)+NXI+1
      NOD(4)=NOD(3)-1
      GO TO 412
380   NOD(1)=NOD(2)
      NOD(2)=NOD(1)+1
      NOD(4)=0
      GO TO 412
401   NOD(1)=J+NSUM
      NOD(2)=NOD(1)+1
      NOD(3)=NOD(2)+NXI+1
      NOD(4)=NOD(3)-1
      M=M+1
      WRITE(8,1003) M,NOD(1),NOD(4),NOD(2)
      M=M+1
      WRITE(8,1003) M,NOD(2),NOD(4),NOD(3)
      NOD(1)=NOD(2)
      NOD(2)=NOD(3)+1
      NOD(4)=0
412   M=M+1
      WRITE(8,1003) M,NOD(1),NOD(4),NOD(2)
      IF(NOD(4)) 433,434,433
433   M=M+1
      WRITE(8,1003) M,NOD(2),NOD(4),NOD(3)
434   CONTINUE
500   CONTINUE
      NSUM=NSUM+NXI+1
600   CONTINUE
      NE=M
      WRITE(8,1004) NP
      WRITE(8,1005) NE
```

```
      CLOSE(UNIT=8,STATUS='KEEP')
450   CONTINUE
4     FORMAT(/,23H *** INPUT NEW DATA ***,/)
5     FORMAT(1H ,18A4)
6     FORMAT(I5,F10.4)
7     FORMAT(1H ,E10.4,F10.2,2X,E10.4,2F10.4)
8     FORMAT(I5,2F10.4,4I5,8I3)
11    FORMAT(I5,F10.4)
13    FORMAT(2F10.4)
14    FORMAT(I5,2E16.8)
21    FORMAT(I5,2F10.2)
22    FORMAT(2I5,E16.8)
119   FORMAT(18A4)
1000  FORMAT(I5,4F10.5)
1001  FORMAT(I5,5X,F10.5)
1002  FORMAT(I10,2F16.8)
1003  FORMAT(I5,3I10)
1004  FORMAT(I5)
1005  FORMAT(I5)
1006  FORMAT(2I5)
      RETURN
      END
C ********************************************************
C *                    SUBROUTINE GDATA
C ********************************************************
      SUBROUTINE GDATA
      COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1 TOL,DY,YTOT,IUZ,
     2 NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3 MIO,D1(3,3),
     4 FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5 DSY,IZO,IOT
      COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1 LPIN,INLP
      COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1 FX1,FY1,NIH,
     2 D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3 FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4 SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5 LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
      REWIND(3)
C
C              READ AND PRINT TITLE
C
C
      READ(3,5) TITLE
      WRITE(6,5) TITLE
      READ(3,8) IIF,FTX,FTY,IH,LF,IPR,IPI,JJF,JIO,NIO,MIO,
     1 ISC,IOPT,IZO,IOT
      WRITE(6,54)
      WRITE(6,64) IIF,FTX,FTY,IH,LF,IPR,IPI,JJF,JIO,NIO,MIO,
     1 ISC,IOPT,IZO,IOT
      NIH=IPI
      IUZ=1
C
```

```
C              READ AND PRINT MATERIAL DATA
C
       READ(3,7) (PROP(J),J=1,5)
       IF(NIO.EQ.2) GO TO 95
C
C              FORM STRESS-STRAIN MATRIX "plane strain"
C
       G=PROP(1)/(2.*(1.+PROP(2)))
       C1=PROP(1)*PROP(2)/((1.+PROP(2))*(1.-PROP(2)*2.))
       D(1,1)=(C1*(1.-PROP(2)))/PROP(2)
       D(1,2)=C1
       D(1,3)=0.0
       D(2,1)=D(1,2)
       D(2,2)=D(1,1)
       D(2,3)=0.0
       D(3,1)=0.0
       D(3,2)=0.0
       D(3,3)=G
       WRITE(6,52)
       WRITE(6,7) (PROP(J),J=1,5)
       GO TO 97
C
C              FORM STRESS-STRAIN MATRIX "plane stress"
C
95     CONTINUE
       G1=PROP(1)/(2.*(1.+PROP(2)))
       C2=PROP(1)/(1.-(PROP(2)*PROP(2)))
       D1(1,1)=C2
       D1(1,2)=C2*PROP(2)
       D1(1,3)=0.0
       D1(2,1)=D1(1,2)
       D1(2,2)=D1(1,1)
       D1(2,3)=0.0
       D1(3,1)=0.0
       D1(3,2)=0.0
       D1(3,3)=G1
       WRITE(6,52)
       WRITE(6,7) (PROP(J),J=1,5)
97     CONTINUE
C
C              READ AND PRINT INCREMENT DATA
C
       READ(3,63) CE,BE
       WRITE(6,63) CE,BE
       READ(3,6) NV,TOL
       WRITE(6,6) NV,TOL
       READ(3,11) (NIN(J),XIN(J),J=1,NV)
       WRITE(6,59)
       WRITE(6,11) (NIN(J),XIN(J),J=1,NV)
       IF(MIO.EQ.4) GO TO 82
       WRITE(6,65)
       DO 77 I=1,IIF
       READ(3,68) K,U,V
       WRITE(6,69) K,U,V
77     CONTINUE
```

```
      WRITE(6,70)
      DO 78 NB=1,JJF
      READ(3,74) K,FX,FY
78    WRITE(6,73) K,FX,FY
      GO TO 90
82    WRITE(6,65)
      DO 83 J=1,IIF
      READ(3,68) K,U,V
83    WRITE(6,69) K,U,V
      WRITE(6,70)
      DO 75 I=1,JJF
      READ(3,71) K,NY,ZY
75    WRITE(6,72) K,NY,ZY
90    CONTINUE
      IF(IZO.EQ.1) GO TO 94
      READ(3,150) NP,NE
      WRITE(6,160)
      DO 300 J=1,NP
      READ(3,165) K,(CORD(J,M),M=1,2)
      WRITE(6,165) K,(CORD(J,M),M=1,2)
300   CONTINUE
      WRITE(6,175)
      DO 310 I=1,NE
      READ(3,180) L,(NOP(I,MM),MM=1,3)
      WRITE(6,180) L,(NOP(I,MM),MM=1,3)
310   CONTINUE
      WRITE(6,155) NP,NE
      WRITE(6,185)
      WRITE(6,190)
94    CONTINUE
5     FORMAT(18A4)
6     FORMAT(I5,F10.4)
7     FORMAT(1H ,E10.4,F10.2,2X,E10.4,2F10.4)
8     FORMAT(I5,2F10.4,4I5,8I3)
11    FORMAT(I5,F10.4)
12    FORMAT(I5)
51    FORMAT(1H ,20H NUMBER OF ELEMENTS=,I5)
52    FORMAT(/,1H ,22H MATERIAL PROPERTIES :,/)
54    FORMAT(/,1H ,18H PRESSURING DATA :,/)
59    FORMAT(/,1H ,17H INCREMENT DATA :,/ )
62    FORMAT(E10.4)
63    FORMAT(2F10.4)
64    FORMAT(/,5H IIF=,I5,3X,5H FTX=,F10.4,3X,
     1 5H FTY=,F10.4,3X,4H IH=,I5,//,
     2 4H LF=,I5,4X,5H IPR=,I5,8X,5H IPI=,I5,8X,
     2 5H JJF=,I3,//,
     4 5H JIO=,I3,5X,5H NIO=,I3,10X,5H MIO=,I3,10X,
     5 5H ISC=,I3,//,
     6 6H IOPT=,I3,4X,5H IZO=,I2,11X,5H IOT=,I3)
65    FORMAT(/,21H BOUNDARY CONDITION :,/)
68    FORMAT(I5,2F10.2)
69    FORMAT(I5,2X,F10.2,2X,F10.2)
70    FORMAT(/,15H NODAL FORCES :,/)
71    FORMAT(2I5,E16.8)
72    FORMAT(I5,2X,I5,2X,E16.8)
```

```
73        FORMAT(I5,2X,E16.8,2X,E16.8)
74        FORMAT(I5,2E16.8)
150       FORMAT(2I5)
155       FORMAT(/,18H NUMBER OF POINTS=,I5,/,
     1    20H NUMBER OF ELEMENTS=,I5)
160       FORMAT(/,14H NODAL POINT :,/)
165       FORMAT(I10,2F16.8)
175       FORMAT(/,11H ELEMENTS :,/)
180       FORMAT(I5,3I10)
185       FORMAT(/,24H ** FINISH INPUT DATA **,/)
190       FORMAT(/,17H ** NEW OUTPUT **,//)
420       FORMAT(I3,3F8.3,I3,2F8.3,7I3)
          RETURN
          END
C ***********************************************************
C *                         SUBROUTINE COORGEN
C ***********************************************************
          SUBROUTINE COORGEN
          COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1    TOL,DY,YTOT,IUZ,
     2    NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3    MIO,D1(3,3),
     4    FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5    DSY,IZO,IOT
          COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1    LPIN,INLP
          COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1    FX1,FY1,NIH,
     2    D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3    FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4    SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5    LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
          DIMENSION YL(50),XF(50),YF(50),NOD(4),SUM1(40),
     1    XL(50)
          REWIND(7)
C
C                           GENERATE COORDINATES
C
          WRITE(6,1006)
          READ(7,1005) NY,CON1
          WRITE(6,1005) NY,CON1
          DO 15 I=1,NY
          READ(7,1007) NX(I),XF(I),YF(I),XL(I),YL(I)
          WRITE(6,1007) NX(I),XF(I),YF(I),XL(I),YL(I)
15        CONTINUE
          WRITE(6,1008)
          DO 16 J=1,NP
          READ(7,1009) N,(CORD(N,M),M=1,2)
          WRITE(6,1009) N,(CORD(N,M),M=1,2)
16        CONTINUE
1005      FORMAT(I5,5X,F10.5)
1006      FORMAT(/,20H CONTROL MESH DATA :,/)
1007      FORMAT(I5,6F10.5)
1008      FORMAT(/,14H NODAL POINT :,/)
1009      FORMAT(I10,2F16.8)
```

```
      RETURN
      END
C *******************************************************
C *                     SUBROUTINE ELGEN
C *******************************************************
      SUBROUTINE ELGEN
      COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1 TOL,DY,YTOT,IUZ,
     2 NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3 MIO,D1(3,3),
     4 FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5 DSY,IZO,IOT
      COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1 LPIN,INLP
      COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1 FX1,FY1,NIH,
     2 D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3 FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4 SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5 LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
      DIMENSION YL(50),XF(50),YF(50),NOD(4),SUM1(50),
     1 XL(50)
      REWIND(8)
C
C
C
C                       GENERATE ELEMENT TOPOLOGY
C
      READ(8,100) (N,(NOP(N,M),M=1,3),N=1,NE)
C
C                       INITIALIZE  VALUES
C
      DO 310 J=1,NP
      DO 310 M=1,2
      TION(J,M)=0.0
310   CONTINUE
      DO 340 I=1,NE
      EEQ(I)=0.0
      SEQI(I)=0.0
      DO 320 J=1,3
      ETA(I,J)=0.0
320   SIG(I,J)=0.0
      SIG(I,4)=0.0
340   CONTINUE
      DSY=1.
C
C
      IF(JIO.NE.0) GO TO 400
C
C               PRINT ELEMENT TOPOLOGY
C
      WRITE(6,58)
      WRITE(6,100) (N,(NOP(N,M),M=1,3),N=1,NE)
      WRITE(6,59)NP
      WRITE(6,60)NE
```

```
      WRITE(6,102)
      WRITE(6,103)
400   CONTINUE
8     FORMAT(I10,2F10.4)
58    FORMAT(/,11H ELEMENTS :,/)
59    FORMAT(/,20H NUMBER OF POINTS  =,I5)
60    FORMAT(20H NUMBER OF ELEMENTS=,I5)
100   FORMAT(I5,3I10)
102   FORMAT(/,24H ** FINISH INPUT DATA **,/)
103   FORMAT(//,17H ** NEW OUTPUT **,//)
      RETURN
      END
C **********************************************************
C *                      SUBROUTINE LOAD
C **********************************************************
      SUBROUTINE LOAD
      COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1 TOL,DY,YTOT,IUZ,
     2 NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3 MIO,D1(3,3),
     4 FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5 DSY,IZO,IOT
      COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1 LPIN,INLP
      COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1 FX1,FY1,NIH,
     2 D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3 FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4 SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5 LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
C
C
C                       CHOOSE THE CASE LOAD
C
C
      IF(MIO.EQ.4) GO TO 16
      IF(MIO.EQ.3) GO TO 17
      IF(MIO.EQ.2) GO TO 18
      GO TO 19
16    CALL LOAD4
      GO TO 420
17    CALL LOAD3
      GO TO 420
18    CALL LOAD2
      GO TO 420
19    REWIND(3)
      FN=0.0
      DO 22 I=1,NE
22    LYD(I)=0
      DO 15 JJ=1,NSZF
15    FORC(JJ)=0.0
C
C                       ENTER VALUE OF FORCE
C
C
```

```
      AD=CE*DY
      YTOT=YTOT+AD
      IF(ABS(YTOT).LT.BE) GO TO 25
      AD=AD-(YTOT-BE)
      YTOT=BE
      KF=1
      WRITE(6,40)
C
C                    INITIALIZE BOUNDARY VECTOR
C
25    DO 50 J=1,NSZF
50    BOUN(J)=1.
C
C                    LOOP FOR EACH NODE UPDATING COORDINATES
C                    APPLYING BOUNDARY CONDITIONS
C
      READ(3,5) TITLE
      READ(3,8) IIF
      READ(3,7) (PROP(J),J=1,5)
      READ(3,63) CE,BE
      READ(3,6) NV,TOL
      READ(3,11)(NIN(J),XIN(J),J=1,NV)
      DO 380 NC=1,IIF
      READ(3,12) K,U,V
      M=K*2
      BOUN(M)=V
      M=M-1
      BOUN(M)=U
380   CONTINUE
      DO 410 NB=1,JJF
      READ(3,100) K,FX,FY
      LL=K*2
      FORC(LL)=FY*AD
      LL=LL-1
      FORC(LL)=FX*AD
410   CONTINUE
      JIM=NX(1)+1
      IG=IH-1
      IGT=LF-IG
      FORC(IGT*2)=-FORC(LF*2)
      JU=IPI-JIM
      DO 190 IJ=JU,IUZ,-JIM
      IF(CORD(IJ,2).LE.CORD(IUZ,2)) GO TO 192
190   CONTINUE
      GO TO 170
192   CONTINUE
      IUZ=IJ
170   CONTINUE
      IF(IOPT.NE.1) GO TO 500
      DO 499 I=1,NSZF
      FN=FN+FORC(I)
499   CONTINUE
500   CONTINUE
      IF(CORD(NP,2).LE.0.0) GO TO 1000
5     FORMAT(18A4)
```

```
6       FORMAT(I5,F10.4)
7       FORMAT(1H ,E10.4,F10.2,2X,E10.4,2F10.4)
8       FORMAT(I5)
11      FORMAT(I5,F10.4)
12      FORMAT(I5,2F10.2)
40      FORMAT(//,48H THE PROCESS WILL COMPLETE AFTER
     1  NEXT INCREMENT  )
63      FORMAT(2F10.4)
100     FORMAT(I5,2E16.8)
110     FORMAT(//,34H     CRITICAL SIZE OF DEFLECTION .,//)
420     CONTINUE
        RETURN
1000    WRITE(6,110)
        STOP
        END
C ************************************************************
C *                         SUBROUTINE LOAD2
C ************************************************************
        SUBROUTINE LOAD2
        COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1  TOL,DY,YTOT,IUZ,
     2  NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3  MIO,D1(3,3),
     4  FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5  DSY,IZO,IOT
        COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1  LPIN,INLP
        COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1  FX1,FY1,NIH,
     2  D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3  FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4  SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5  LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
        REWIND(3)
        FN=0.0
        DO 22 I=1,NE
22      LYD(I)=0
        DO 15 JJ=1,NSZF
15      FORC(JJ)=0.0
C
C                         ENTER VALUE OF FORCE
C
C
        AD=CE*DY
        YTOT=YTOT+AD
        IF(ABS(YTOT).LT.BE) GO TO 25
        AD=AD-(YTOT-BE)
        YTOT=BE
        KF=1
        WRITE(6,40)
C
C                         INITIALIZE BOUNDARY VECTOR
C
```

```
25      DO 50 J=1,NSZF
50      BOUN(J)=1.
C
C                       LOOP FOR EACH NODE UPDATING COORDINATES
C                       APPLYING BOUNDARY CONDITIONS
C
        READ(3,5) TITLE
        READ(3,8) IIF
        READ(3,7) (PROP(J),J=1,5)
        READ(3,63) CE,BE
        READ(3,6) NV,TOL
        READ(3,11)(NIN(J),XIN(J),J=1,NV)
        DO 410 NC=1,IIF
        READ(3,12) K,U,V
        M=K*2
        BOUN(M)=V
        I=M-1
        BOUN(I)=U
410     CONTINUE
        DO 420 LC=1,JJF
        READ(3,100) K,FX,FY
        MM=K*2
        FORC(MM)=FY*AD
        MM=MM-1
        FORC(MM)=FX*AD
420     CONTINUE
        JIM=NX(1)+1
        MH=NP-NX(1)
        IL=MH-JIM
        DO 10 II=IPI,IL,JIM
        IF(CORD(II,2).LT.CORD(IH,2)) GO TO 10
        GO TO 30
10      CONTINUE
        GO TO 41
30      CONTINUE
        DO 31 JJ=II,MH,JIM
        NN=JJ*2
        FORC(NN)=FORC(LF*2)
        NN=NN-1
        FORC(NN)=FORC(LF*2-1)
31      CONTINUE
        IM=II+JIM
        DO 90 IJ=IM,MH,JIM
        IF(CORD(IJ,2).LE.CORD(IH,2)) GO TO 92
90      CONTINUE
        GO TO 101
92      CONTINUE
        DO 35 JJ=IM,MH,JIM
        MM=JJ*2
        FORC(MM)=0.0
        MM=MM-1
        FORC(MM)=0.0
35      CONTINUE
101     CONTINUE
        IH=II
```

```
41      CONTINUE
        IG=IH-1
        IGT=LF-IG
        FORC(IGT*2)=-FORC(IH*2)
        IUZ=IGT
        IF(CORD(NP,2).LE.0.0) GO TO 1000
110     CONTINUE
        IF(IOPT.NE.1) GO TO 500
        DO 499 I=1,NSZF
        FN=FN+FORC(I)
499     CONTINUE
500     CONTINUE
5       FORMAT(18A4)
6       FORMAT(I5,F10.4)
7       FORMAT(1H ,E10.4,F10.2,2X,E10.4,2F10.4)
8       FORMAT(I5)
11      FORMAT(I5,F10.4)
12      FORMAT(I5,2F10.2)
40      FORMAT(//,48H THE PROCESS WILL COMPLETE AFTER
     1  NEXT INCREMENT  )
63      FORMAT(2F10.4)
100     FORMAT(I5,2E16.8)
900     FORMAT(//,34H     CRITICAL SIZE OF DEFLECTION .,//)
        RETURN
1000    CONTINUE
        CORD(NP,2)=0.0
        WRITE(6,900)
        STOP
        END
C ***********************************************************
C *                        SUBROUTINE LOAD3
C ***********************************************************
        SUBROUTINE LOAD3
        COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1  TOL,DY,YTOT,IUZ,
     2  NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3  MIO,D1(3,3),
     4  FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5  DSY,IZO,IOT
        COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1  LPIN,INLP
        COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1  FX1,FY1,NIH,
     2  D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3  FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4  SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5  LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
        REWIND(3)
        FN=0.0
        DO 22 I=1,NE
22      LYD(I)=0
        DO 15 JJ=1,NSZF
15      FORC(JJ)=0.0
C
C                        ENTER VALUE OF FORCE
```

```
C
C
      AD=CE*DY
      YTOT=YTOT+AD
      IF(ABS(YTOT).LT.BE) GO TO 25
      AD=AD-(YTOT-BE)
      YTOT=BE
      KF=1
      WRITE(6,40)
C
C                       INITIALIZE BOUNDARY VECTOR
C
25    DO 50 J=1,NSZF
50    BOUN(J)=1.
C
C                       LOOP FOR EACH NODE UPDATING COORDINATES
C                       APPLYING BOUNDARY CONDITIONS
C
      READ(3,5) TITLE
      READ(3,8) IIF
      READ(3,7) (PROP(J),J=1,5)
      READ(3,63) CE,BE
      READ(3,6) NV,TOL
      READ(3,11)(NIN(J),XIN(J),J=1,NV)
      DO 410 NC=1,IIF
      READ(3,12) K,U,V
      M=K*2
      BOUN(M)=V
      I=M-1
      BOUN(I)=U
410   CONTINUE
      DO 420 LC=1,JJF
      READ(3,100) K,FX,FY
      MM=K*2
      FORC(MM)=FY*AD
      MM=MM-1
      FORC(MM)=FX*AD
420   CONTINUE
      JIM=NX(1)+1
      MH=NP-NX(1)
      IL=MH-JIM
      IF(LPIN.EQ.NIN(3)) GO TO 43
      IF(LPIN.NE.NIN(2)) GO TO 46
      IF(INLP.LT.2) GO TO 46
      IF(INLP.GT.2) GO TO 43
      XTC=CORD(NIH,1)
      WRITE(6,110) XTC
43    CONTINUE
      BOUN(IPR*2-1)=0.0
      DIS(IPR,1)=0.0
      JAK=JIM+IPI
      DO 51 KK=MH,JAK,-JIM
      IF(CORD(KK,1).LT.CORD(IPR,1)) GO TO 51
      GO TO 44
51    CONTINUE
```

```
         GO TO 46
 44      CONTINUE
         DO 45 LL=KK,IPI,-JIM
         BOUN(LL*2-1)=0.0
         DIS(LL,1)=0.0
 45      CONTINUE
         IPR=KK
         JOK=IPI-JIM
         DO 150 JZ=1,JOK,JIM
         IF(CORD(JZ,1).LT.CORD(NIH,1)) GO TO 150
         GO TO 155
 150     CONTINUE
         GO TO 46
 155     CONTINUE
         DO 160 LV=JZ,IPI,JIM
         BOUN(LV*2-1)=0.0
         DIS(LV,1)=0.0
 160     CONTINUE
         NIH=JZ
 46      CONTINUE
         DO 10 II=IPI,IL,JIM
         IF(CORD(II,2).LT.CORD(IH,2)) GO TO 10
         GO TO 30
 10      CONTINUE
         GO TO 41
 30      CONTINUE
         DO 31 IK=II,MH,JIM
         NN=II*2
         FORC(NN)=FORC(LF*2)
         NN=NN-1
         FORC(NN)=FORC(LF*2-1)
 31      CONTINUE
         IH=II
         IM=II+JIM
         DO 90 IJ=IM,MH,JIM
         IF(CORD(IJ,2).LE.CORD(IH,2)) GO TO 92
 90      CONTINUE
         GO TO 101
 92      CONTINUE
         DO 35 JJ=IM,NP,JIM
         MM=JJ*2
         FORC(MM)=0.0
         MM=MM-1
         FORC(MM)=0.0
 35      CONTINUE
 101     CONTINUE
 41      CONTINUE
         IG=IH-1
         IGT=LF-IG
         FORC(IGT*2)=-FORC(IH*2)
         IUZ=IGT
 800     CONTINUE
 170     CONTINUE
         IF(IOPT.NE.1) GO TO 500
         DO 499 I=1,NSZF
```

```
          FN=FN+FORC(I)
499       CONTINUE
500       CONTINUE
          IF(CORD(NP,2).LE.CORD(JIM,2)) GO TO 1000
5         FORMAT(18A4)
6         FORMAT(I5,F10.4)
7         FORMAT(1H ,E10.4,F10.2,2X,E10.4,2F10.4)
8         FORMAT(I5)
11        FORMAT(I5,F10.4)
12        FORMAT(I5,2F10.2)
40        FORMAT(//,48H THE PROCESS WILL COMPLETE AFTER
     1    NEXT INCREMENT)
63        FORMAT(2F10.4)
100       FORMAT(I5,2E16.8)
110       FORMAT(/,' THE COORDINATES OF THE WALL ARE:',//,
     1    6H Y=0.0,6X,3H X=,F16.8,/)
900       FORMAT(//,34H     CRITICAL SIZE OF DEFLECTION .,//)
          RETURN
1000      CONTINUE
          CORD(NP,2)=CORD(JIM,2)
          WRITE(6,900)
          STOP
          END
C ********************************************************
C *                       SUBROUTINE LOAD4
C ********************************************************
          SUBROUTINE LOAD4
          COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1    TOL,DY,YTOT,IUZ,
     2    NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3    MIO,D1(3,3),
     4    FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5    DSY,IZO,IOT
          COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1    LPIN,INLP
          COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1    FX1,FY1,NIH,
     2    D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3    FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4    SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5    LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
          REWIND(3)
          FN=0.0
          DO 10 I=1,NE
10        LYD(I)=0
          DO 22 JJ=1,NSZF
22        FORC(JJ)=0.0
C
C                       ENTER VALUE OF FORCE
C
C
          AD=CE*DY
          YTOT=YTOT+AD
          IF(ABS(YTOT).LT.BE) GO TO 25
          AD=AD-(YTOT-BE)
```

```
      YTOT=BE
      KF=1
      WRITE(6,40)
C
C                      INITIALIZE BOUNDARY VECTOR
C
25    DO 50 J=1,NSZF
50    BOUN(J)=1.
C
C             LOOP FOR EACH NODE UPDATING COORDINATES
C                      APPLYING BOUNDARY CONDITIONS
C
      READ(3,5) TITLE
      READ(3,8) IIF,FTX
      READ(3,7) (PROP(J),J=1,5)
      READ(3,63) CE,BE
      READ(3,6) NV,TOL
      READ(3,11)(NIN(J),XIN(J),J=1,NV)
      DO 390 II=1,IIF
      READ(3,15) K,U,V
      NN=K*2
      BOUN(NN)=V
      NN=NN-1
      BOUN(NN)=U
390   CONTINUE
      JAN=JJF-1
      DO 410 NC=1,JAN
      READ(3,12) K,NY,ZY
      AMI=CORD(K,1)-CORD(NY,1)
      BMI=CORD(NY,2)-CORD(K,2)
      EQL=(AMI**2.+BMI**2.)**0.5
      COS=AMI/EQL
      SIN=BMI/EQL
      NN=K*2
      FORC(NN)=(((ZY*EQL*PROP(5))/2.)*COS)*AD
      NN=NN-1
      FORC(NN)=-(((ZY*EQL*PROP(5))/2.)*SIN)*AD
      MM=NY*2
      FORC(MM)=(((ZY*EQL*PROP(5))/2.)*COS)*AD
      MM=MM-1
      FORC(NN)=-(((ZY*EQL*PROP(5))/2.)*SIN)*AD
410   CONTINUE
      IF(IOPT.NE.1) GO TO 500
      DO 499 I=1,NSZF
      FN=FN+FORC(I)
499   CONTINUE
500   CONTINUE
5     FORMAT(18A4)
6     FORMAT(I5,F10.4)
7     FORMAT(1H ,E10.4,F10.2,2X,E10.4,2F10.4)
8     FORMAT(I5,F10.4)
11    FORMAT(I5,F10.4)
12    FORMAT(2I5,E16.8)
15    FORMAT(I5,2F10.2)
40    FORMAT(//48H THE PROCESS WILL
```

```
     1   COMPLETE AFTER NEXT INCREMENT  )
63      FORMAT(2F10.4)
        RETURN
        END
C ***********************************************************
C *                        SUBROUTINE STIFT(N)
C ***********************************************************
        SUBROUTINE STIFT(N)
        COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1  TOL,DY,YTOT,IUZ,
     2  NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3  MIO,D1(3,3),
     4  FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5  DSY,IZO,IOT
        COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1  LPIN,INLP
        COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1  FX1,FY1,NIH,
     2  D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3  FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4  SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5  LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
        DIMENSION B(3,6),H(3,6)
C
C
C                        DETERMINE ELEMENT CONNECTIONS
C
C
        I=NOP(N,1)
        J=NOP(N,2)
        K=NOP(N,3)
C
C                        SET UP LOCAL COORDINATE SYSTEM
C
        AJ=CORD(J,1)-CORD(I,1)
        AK=CORD(K,1)-CORD(I,1)
        BJ=CORD(J,2)-CORD(I,2)
        BK=CORD(K,2)-CORD(I,2)
        DET=(AJ*BK-AK*BJ)/2.0
        IF(DET.LE.0.0) GO TO 200
        IF(NIO.EQ.1) PROP(5)=1.
        TAREA(N)=DET*PROP(5)
C
C
C                        FORM STRAIN-DISPLACEMENT MATRIX
C                        AND WRITE TO FILE NUMBER
C
C
        B(1,1)=BJ-BK
        B(1,2)=0.0
        B(1,3)=BK
        B(1,4)=0.0
        B(1,5)=-BJ
        B(1,6)=0.0
        B(2,1)=0.0
```

```
      B(2,2)=AK-AJ
      B(2,3)=0.0
      B(2,4)=-AK
      B(2,5)=0.0
      B(2,6)=AJ
      B(3,1)=AK-AJ
      B(3,2)=BJ-BK
      B(3,3)=-AK
      B(3,4)=BK
      B(3,5)=AJ
      B(3,6)=-BJ
      DO 20 I=1,3
      DO 20 J=1,6
20    B(I,J)=B(I,J)/(DET*2.)
      DO 25 I=1,3
      WRITE(4,5) N,(B(I,J),J=1,6)
5     FORMAT(I3,2X,6F12.4)
25    CONTINUE
C
C             H IS STRESS BACK-SUBSTITUTION MATRIX
C
      IF(NIO.EQ.2) GO TO 65
      DO 60 I=1,3
      DO 60 J=1,6
      H(I,J)=0.0
      DO 60 K=1,3
60    H(I,J)=H(I,J)+D(I,K)*B(K,J)
      GO TO 67
65    CONTINUE
      DO 63 I=1,3
      DO 63 J=1,6
      H(I,J)=0.0
      DO 63 K=1,3
63    H(I,J)=H(I,J)+D1(I,K)*B(K,J)
67    CONTINUE
C
C                     ESTIFM IS STIFFNESS MATRIX
C
      DO 80 I=1,6
      DO 80 J=1,6
      ESTIFM(I,J)=0.0
      DO 80 K=1,3
80    ESTIFM(I,J)=ESTIFM(I,J)+(H(K,I)*B(K,J)*TAREA(N))
      RETURN
C
C                     ERROR EXIT FOR BAD CONNECTIONS
C
200   WRITE(6,300) N
300   FORMAT(33H ZERO OR NEGATIVE AREA ELEMENT NO,I4
     1 /26H *** EXECUTION TERMINATED  )
      STOP
      END
```

```
C ********************************************************
C *                       SUBROUTINE FORMK
C ********************************************************
        SUBROUTINE FORMK
        COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1  TOL,DY,YTOT,IUZ,
     2  NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3  MIO,D1(3,3),
     4  FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5  DSY,IZO,IOT
        COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1  LPIN,INLP
        COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1  FX1,FY1,NIH,
     2  D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3  FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4  SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5  LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
C
C
C                       ZERO STIFFNESS MATRIX
C
        DO 100 N=1,NSZF
        DO 100 M=1,NSZF
100     SK(N,M)=0.0
C
C                       SCAN ELEMENTS
C
        OPEN(UNIT=4,FILE='SCRATCH:MATFIN.DAT',STATUS='NEW')
        DO 500 N=1,NE
        CALL STIFT(N)
C
C                       ASSEMBLE ELEMENT STIFFNESS MATRICES
C                       IN GLOBAL STIFFNESS MATRIX
C
C                       FIRST ROWS
C
        DO 400 JJ=1,3
        NROWB=(NOP(N,JJ)-1)*2
        DO 400 J=1,2
        NROWB=NROWB+1
        I=(JJ-1)*2+J
C
C                       THEN COLUMNS
C
        DO 200 KK=1,3
        NCOLB=(NOP(N,KK)-1)*2
        DO 200 K=1,2
        L=(KK-1)*2+K
        NCOLB=NCOLB+1
        SK(NROWB,NCOLB)=SK(NROWB,NCOLB)+ESTIFM(I,L)
200     CONTINUE
400     CONTINUE
500     CONTINUE
        CLOSE(UNIT=4,STATUS='KEEP')
```

```
      RETURN
      END
C *******************************************************
C *                     SUBROUTINE SOLV3
C *******************************************************
      SUBROUTINE SOLV3
      COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1 TOL,DY,YTOT,IUZ,
     2 NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3 MIO,D1(3,3),
     4 FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5 DSY,IZO,IOT
      COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1 LPIN,INLP
      COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1 FX1,FY1,NIH,
     2 D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3 FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4 SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5 LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
      DIMENSION SUBK(600,600),UPRES(600)
      NPRES=0
      L=1
      M=1
C
C                      ESTABLISH PRESCRIBED FREEDOMS
C
      DO 6 I=1,NSZF
      IF(FORC(I).NE.0.0) BOUN(I)=FORC(I)
6     CONTINUE
      DO 100 J=1,NSZF
      IF(BOUN(J).EQ.1.) GO TO 200
      NPRES=NPRES+1
      UPRES(NPRES)=BOUN(J)
      JCOL(M)=J
      M=M+1
      GO TO 100
200   JROW(L)=J
      L=L+1
100   CONTINUE
      NROWS=NSZF-NPRES
      NST=NROWS+1
C
C                RE-FORM STIFFNESS MATRIX IN PARTITIONS
C
      DO 300 J=1,NROWS
      LC=JROW(J)
      DO 150 M=1,NROWS
      MC=JROW(M)
150   SUBK(J,M)=SK(LC,MC)
      DO 250 N=1,NPRES
      NC=JCOL(N)
      SUBK(J,M)=SK(LC,NC)
      M=M+1
250   CONTINUE
```

```
300     CONTINUE
        DO 500 J=NST,NSZF
        K=J-NROWS
        KC=JCOL(K)
        DO 350 L=1,NROWS
        LC=JROW(L)
350     SUBK(J,L)=SK(KC,LC)
        DO 400 N=1,NPRES
        MC=JCOL(N)
        SUBK(J,L)=SK(KC,MC)
        L=L+1
400     CONTINUE
500     CONTINUE
C
C                         FORM FORCE VECTOR
C
        DO 550 K=1,NSZF
550     FORC(K)=0.0
        DO 600 I=1,NROWS
        L=1
        DO 600 J=NST,NSZF
        FORC(I)=FORC(I)-SUBK(I,J)*UPRES(L)
        L=L+1
600     CONTINUE
        DO 700 I=NST,NSZF
        L=1
        DO 700 J=NST,NSZF
        FORC(I)=FORC(I)+SUBK(I,J)*UPRES(L)
        L=L+1
700     CONTINUE
        DO 900 I=1,NSZF
        DO 900 J=1,NSZF
900     SK(I,J)=SUBK(I,J)
        RETURN
        END
C *******************************************************
C *                        SUBROUTINE GAUSS
C *******************************************************
        SUBROUTINE GAUSS
        COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1  TOL,DY,YTOT,IUZ,
     2  NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3  MIO,D1(3,3),
     4  FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5  DSY,IZO,IOT
        COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1  LPIN,INLP
        COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1  FX1,FY1,NIH,
     2  D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3  FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4  SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5  LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
        DIMENSION LP(600),LQ(600,2),R(600)
C
```

```
C                               INVERT STIFFNESS MATRIX
C
        DO 12 I=1,NROWS
12      LP(I)=0
        DO 150 K=1,NROWS
        CON=0.0
        DO 50 I=1,NROWS
        IF(LP(I).EQ.1) GO TO 50
        DO 40 J=1,NROWS
        IF(LP(J)-1) 30,40,200
30      IF(ABS(CON).GE.ABS(SK(I,J))) GO TO 40
        IR=I
        IC=J
        CON=SK(I,J)
40      CONTINUE
50      CONTINUE
        LP(IC)=LP(IC)+1
        IF(IR.EQ.IC) GO TO 90
        DO 60 I=1,NROWS
        CON=SK(IR,I)
        SK(IR,I)=SK(IC,I)
60      SK(IC,I)=CON
90      LQ(K,1)=IR
        LQ(K,2)=IC
        R(K)=SK(IC,IC)
        SK(IC,IC)=1.0
        DO 100 I=1,NROWS
100     SK(IC,I)=SK(IC,I)/(R(K))
        DO 150 I=1,NROWS
        IF(I.EQ.IC) GO TO 150
        CON=SK(I,IC)
        SK(I,IC)=0.0
        DO 130 J=1,NROWS
130     SK(I,J)=SK(I,J)-SK(IC,J)*CON
150     CONTINUE
        DO 170 I=1,NROWS
        J=NROWS-I+1
        IF(LQ(J,1).EQ.LQ(J,2)) GO TO 170
        IR=LQ(J,1)
        IC=LQ(J,2)
        DO 160 K=1,NROWS
        CON=SK(K,IR)
        SK(K,IR)=SK(K,IC)
        SK(K,IC)=CON
160     CONTINUE
170     CONTINUE
        RETURN
200     WRITE(2,999)
999     FORMAT(//33H SINGULAR MATRIX CANNOT BE SOLVED,
     1  /26H *** EXECUTION TERMINATED  )
        STOP
        END
```

```
C ***********************************************************
C *                          SUBROUTINE SOLV4
C ***********************************************************
      SUBROUTINE SOLV4
      COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1 TOL,DY,YTOT,IUZ,
     2 NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3 MIO,D1(3,3),
     4 FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5 DSY,IZO,IOT
      COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1 LPIN,INLP
      COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1 FX1,FY1,NIH,
     2 D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3 FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4 SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5 LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
      DIMENSION U(600),B(3,6),R(6)
      REWIND(4)
C
C                          CALCULATE UNKNOWN DISPLACEMENTS
C
      DO 10 I=1,NROWS
      U(I)=0.0
      DO 10 J=1,NROWS
      U(I)=U(I)+SK(I,J)*FORC(J)
10    CONTINUE
C
C                          CALCULATE REACTIONS
C
      NST=NROWS+1
      DO 50 I=NST,NSZF
      DO 50 J=1,NROWS
      FORC(I)=FORC(I)+SK(I,J)*U(J)
50    CONTINUE
C
C
C                  DETERMINE FORCE APPLIED BY PRESSURING
C
C     IF(FN.NE.0.0) GO TO 80
      N=NROWS
      DO 70 J=1,NSZF
      IF(BOUN(J).EQ.1.) GO TO 70
      N=N+1
      IF(BOUN(J).EQ.0.) GO TO 70
      FN=FN+FORC(N)
70    CONTINUE
80    CONTINUE
C
C                          FORM DISPLACEMENT VECTOR
C
      DO 90 J=1,NROWS
      NC=JROW(J)
      BOUN(NC)=U(J)
```

```
90      CONTINUE
C
C                       CALCULATE ELASTIC INCREMENT OF
C                       STRAIN AND STRESS
C
        IF(NIO.EQ.2) GO TO 101
        DO 500 NC=1,NE
        DO 7 I=1,3
7       READ(4,9) N,(B(I,J),J=1,6)
9       FORMAT(I3,2X,6F12.3)
        DO 100 I=1,3
        II=2*NOP(N,I)-1
        JJ=2*NOP(N,I)
        R(2*I-1)=BOUN(II)
        R(2*I)=BOUN(JJ)
100     CONTINUE
        DO 300 I=1,3
        ELETA(N,I)=0.0
        DO 300 J=1,6
300     ELETA(N,I)=ELETA(N,I)+B(I,J)*R(J)
        DO 400 I=1,3
        ELSIG(N,I)=0.0
        DO 400 J=1,3
400     ELSIG(N,I)=ELSIG(N,I)+D(I,J)*ELETA(N,J)
        ELSIG(N,4)=(ELSIG(N,1)+ELSIG(N,2))*PROP(2)
500     CONTINUE
        GO TO 499
101     CONTINUE
        DO 501 NC=1,NE
        DO 71 I=1,3
71      READ(4,91) N,(B(I,J),J=1,6)
91      FORMAT(I3,2X,6F12.3)
        DO 103 I=1,3
        II=2*NOP(N,I)-1
        JJ=2*NOP(N,I)
        R(2*I-1)=BOUN(II)
        R(2*I)=BOUN(JJ)
103     CONTINUE
        DO 301 I=1,3
        ELETA(N,I)=0.0
        DO 301 J=1,6
301     ELETA(N,I)=ELETA(N,I)+B(I,J)*R(J)
        ELETA(N,4)=-(ELSIG(N,1)+ELSIG(N,2))*
     1  (PROP(2)/(PROP(1)))
        DO 401 I=1,3
        ELSIG(N,I)=0.0
        DO 401 J=1,3
        ELSIG(N,I)=ELSIG(N,I)+D1(I,J)*ELETA(N,J)
401     CONTINUE
501     CONTINUE
499     CONTINUE
        RETURN
        END
```

```
C *********************************************************
C *                      SUBROUTINE SOLV1
C *********************************************************
      SUBROUTINE SOLV1
      COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1 TOL,DY,YTOT,IUZ,
     2 NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3 MIO,D1(3,3),
     4 FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5 DSY,IZO,IOT
      COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1 LPIN,INLP
      COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1 FX1,FY1,NIH,
     2 D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3 FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4 SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5 LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
      DIMENSION SS(600),SBF(4),BF(6),
     1 SL(600,600),B(3,6),R(6)
      REWIND(4)
      NPRES=0
      L=1
      M=1
C
C                       CARRY OUT GAUSSIAN ELIMINATION
C
      DO 901 I=1,NSZF
      SS(I)=0.0
      SS(I)=FORC(I)
      DISPL(I)=0.0
      REC(I)=0.0
      DO 901 J=1,NSZF
      SL(I,J)=0.0
901   SL(I,J)=SK(I,J)
      I=0
      DO 151 J=1,NSZF
      I=I+1
      IF(BOUN(J).EQ.1.) GO TO 151
      SL(I,I)=SL(I,I)+1.0E+30
151   CONTINUE
      M1=NSZF
      M2=M1-1
      DO 301 I=1,M2
      II=I+1
      DO 290 K=II,M1
      FACT=SL(K,I)/SL(I,I)
      DO 280 J=II,M1
      SL(K,J)=SL(K,J)-FACT*SL(I,J)
280   CONTINUE
      SL(K,I)=0.0
      SS(K)=SS(K)-FACT*SS(I)
290   CONTINUE
301   CONTINUE
C
```

```
C                          BACK SUBSTITUTE
C
         DO 402 I=1,M1
         II=M1-I+1
         PIVOT=SL(II,II)
         SL(II,II)=0.0
         DO 351 J=II,M1
         SS(II)=SS(II)-SL(II,J)*DISPL(J)
351      CONTINUE
         DISPL(II)=(SS(II)/PIVOT)
402      CONTINUE
C
C                          ESTABLISH PRESCRIBED FREEDOMS
C
         DO 100 J=1,NSZF
         IF(BOUN(J).EQ.1.) GO TO 200
         NPRES=NPRES+1
         JCOL(M)=J
         M=M+1
         GO TO 100
200      JROW(L)=J
         L=L+1
100      CONTINUE
         NROWS=NSZF-NPRES
         NST=NROWS+1
C
C                          FORM DISPLACEMENT VECTOR
C
         DO 90 J=1,NSZF
         BOD(J)=0.0
         IF(BOUN(J).EQ.0.0) GO TO 90
         BOD(J)=DISPL(J)
90       CONTINUE
C
C                          CALCULATE ELASTIC INCREMENT OF
C                          STRAIN AND STRESS
C
         IF(NIO.EQ.2) GO TO 101
         DO 500 NC=1,NE
         DO 7 I=1,3
7        READ(4,9) N,(B(I,J),J=1,6)
9        FORMAT(I3,2X,6F12.3)
         DO 107 I=1,3
         II=2*NOP(N,I)-1
         JJ=2*NOP(N,I)
         R(2*I-1)=BOD(II)
         R(2*I)=BOD(JJ)
107      CONTINUE
         DO 300 I=1,3
         ELETA(N,I)=0.0
         DO 300 J=1,6
300      ELETA(N,I)=ELETA(N,I)+B(I,J)*R(J)
         DO 400 I=1,3
         ELSIG(N,I)=0.0
         DO 400 J=1,3
```

```
400     ELSIG(N,I)=ELSIG(N,I)+D(I,J)*ELETA(N,J)
        ELSIG(N,4)=(ELSIG(N,1)+ELSIG(N,2))*PROP(2)
C
C                     CALCULATE REACTION
C
        DO 558 I=1,3
558     SBF(I)=ELSIG(N,I)
        DO 650 I=1,6
        BF(I)=0.0
        DO 600 J=1,3
600     BF(I)=BF(I)+B(J,I)*SBF(J)
650     BF(I)=BF(I)*TAREA(N)
        LP=0
        DO 700 I=1,3
        DO 700 K=1,2
        J=(NOP(N,I)-1)*2+K
        LP=LP+1
        REC(J)=BF(LP)+REC(J)
700     CONTINUE
500     CONTINUE
        GO TO 499
101     CONTINUE
        DO 501 NC=1,NE
        DO 71 I=1,3
71      READ(4,91) N,(B(I,J),J=1,6)
91      FORMAT(I3,2X,6F12.3)
        DO 103 I=1,3
        II=2*NOP(N,I)-1
        JJ=2*NOP(N,I)
        R(2*I-1)=BOD(II)
        R(2*I)=BOD(JJ)
103     CONTINUE
        DO 304 I=1,3
        ELETA(N,I)=0.0
        DO 304 J=1,6
304     ELETA(N,I)=ELETA(N,I)+B(I,J)*R(J)
        ELETA(N,4)=-(ELSIG(N,1)+ELSIG(N,2))*
     1  (PROP(2)/(PROP(1)))
        DO 401 I=1,3
        ELSIG(N,I)=0.0
        DO 401 J=1,3
        ELSIG(N,I)=ELSIG(N,I)+D1(I,J)*ELETA(N,J)
401     CONTINUE
C
C                     CALCULATE REACTION
C
        DO 559 I=1,3
559     SBF(I)=ELSIG(N,I)
        DO 653 I=1,6
        BF(I)=0.0
        DO 607 J=1,3
607     BF(I)=BF(I)+B(J,I)*SBF(J)
653     BF(I)=BF(I)*TAREA(N)
        LP=0
        DO 707 I=1,3
```

```
      DO 707 K=1,2
      J=(NOP(N,I)-1)*2+K
      LP=LP+1
      REC(J)=BF(LP)+REC(J)
707   CONTINUE
501   CONTINUE
499   CONTINUE
      RETURN
      END
C ********************************************************
C *                     SUBROUTINE SOLV2
C ********************************************************
      SUBROUTINE SOLV2
      COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1 TOL,DY,YTOT,IUZ,
     2 NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3 MIO,D1(3,3),
     4 FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5 DSY,IZO,IOT
      COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1 LPIN,INLP
      COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1 FX1,FY1,NIH,
     2 D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3 FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4 SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5 LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
      DIMENSION SS(600),
     1 SL(600,600),B(3,6),R(6)
      REWIND(4)
      NPRES=0
      L=1
      M=1
C
C                     CARRY OUT GAUSSIAN ELIMINATION
C
      DO 901 I=1,NSZF
      SS(I)=0.0
      SS(I)=FORC(I)
      DISPL(I)=0.0
      DO 901 J=1,NSZF
      SL(I,J)=0.0
901   SL(I,J)=SK(I,J)
      I=0
      DO 151 J=1,NSZF
      I=I+1
      IF(BOUN(J).EQ.1.) GO TO 151
      SL(I,I)=SL(I,I)+1.0E+30
151   CONTINUE
      M1=NSZF
      M2=M1-1
      DO 301 I=1,M2
      II=I+1
      DO 290 K=II,M1
      FACT=SL(K,I)/SL(I,I)
```

```
      DO 280 J=II,M1
      SL(K,J)=SL(K,J)-FACT*SL(I,J)
280   CONTINUE
      SL(K,I)=0.0
      SS(K)=SS(K)-FACT*SS(I)
290   CONTINUE
301   CONTINUE
C
C                    BACK SUBSTITUTE
C
      DO 402 I=1,M1
      II=M1-I+1
      PIVOT=SL(II,II)
      SL(II,II)=0.0
      DO 351 J=II,M1
      SS(II)=SS(II)-SL(II,J)*DISPL(J)
351   CONTINUE
      DISPL(II)=(SS(II)/PIVOT)
402   CONTINUE
C
C                    FORM DISPLACEMENT VECTOR
C
      DO 90 J=1,NSZF
      BOD(J)=0.0
      IF(BOUN(J).EQ.0.0) GO TO 90
      BOD(J)=DISPL(J)
90    CONTINUE
C
C                    CALCULATE ELASTIC INCREMENT OF
C                    STRAIN AND STRESS
C
      IF(NIO.EQ.2) GO TO 101
      DO 500 NC=1,NE
      DO 7 I=1,3
7     READ(4,9) N,(B(I,J),J=1,6)
9     FORMAT(I3,2X,6F12.3)
      DO 107 I=1,3
      II=2*NOP(N,I)-1
      JJ=2*NOP(N,I)
      R(2*I-1)=BOD(II)
      R(2*I)=BOD(JJ)
107   CONTINUE
      DO 300 I=1,3
      ELETA(N,I)=0.0
      DO 300 J=1,6
300   ELETA(N,I)=ELETA(N,I)+B(I,J)*R(J)
      DO 400 I=1,3
      ELSIG(N,I)=0.0
      DO 400 J=1,3
      ELSIG(N,I)=ELSIG(N,I)+D(I,J)*ELETA(N,J)
400   CONTINUE
      ELSIG(N,4)=(ELSIG(N,1)+ELSIG(N,2))*PROP(2)
500   CONTINUE
      GO TO 499
101   CONTINUE
```

```
      DO 501 NC=1,NE
      DO 71 I=1,3
71    READ(4,91) N,(B(I,J),J=1,6)
91    FORMAT(I3,2X,6F12.3)
      DO 103 I=1,3
      II=2*NOP(N,I)-1
      JJ=2*NOP(N,I)
      R(2*I-1)=BOD(II)
      R(2*I)=BOD(JJ)
103   CONTINUE
      DO 304 I=1,3
      ELETA(N,I)=0.0
      DO 304 J=1,6
304   ELETA(N,I)=ELETA(N,I)+B(I,J)*R(J)
      ELETA(N,4)=-(ELSIG(N,1)+ELSIG(N,2))*
     1 (PROP(2)/(PROP(1)))
      DO 401 I=1,3
      ELSIG(N,I)=0.0
      DO 401 J=1,3
      ELSIG(N,I)=ELSIG(N,I)+D1(I,J)*ELETA(N,J)
401   CONTINUE
501   CONTINUE
499   CONTINUE
      RETURN
      END
C ********************************************************
C *                    SUBROUTINE TOTAL
C ********************************************************
      SUBROUTINE TOTAL
      COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1 TOL,DY,YTOT,IUZ,
     2 NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3 MIO,D1(3,3),
     4 FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5 DSY,IZO,IOT
      COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1 LPIN,INLP
      COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1 FX1,FY1,NIH,
     2 D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3 FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4 SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5 LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
      DIMENSION X1(2500),Y1(2500),X2(2500),Y2(2500),
     1 X3(2500),Y3(2500)
C
C             DETERMINE AND PRINT ELEMENT COORDINATES
C                                 BEFORE LOADING
C
      IF(DSY.NE.1.) GO TO 30
      FX1=FTX
      FY1=FTY
      DO 10 N=1,NE
      I=NOP(N,1)
      J=NOP(N,2)
```

```
      K=NOP(N,3)
      X1(N)=CORD(I,1)
      Y1(N)=CORD(I,2)
      X2(N)=CORD(J,1)
      Y2(N)=CORD(J,2)
      X3(N)=CORD(K,1)
      Y3(N)=CORD(K,2)
10    CONTINUE
      OPEN(UNIT=12,FILE='MESH.DAT',STATUS='NEW')
      DO 15 N=1,NE
      WRITE(12,210) X1(N),Y1(N),X2(N),Y2(N),X3(N),
     1 Y3(N),X1(N),Y1(N),X2(N),Y2(N)
15    CONTINUE
      CLOSE(UNIT=12,STATUS='KEEP')
      DSY=DSY+1.
      ZO=CORD(IH,2)/CORD(IPR,1)
      IF(ZO.LT.1.) GO TO 32
      IF(ZO.EQ.1.) GO TO 30
      IF(ZO.GT.1.) GO TO 33
      GO TO 30
32    CONTINUE
      IF(ZO.EQ.0.0) GO TO 999
      FTX=FTX/ZO
      GO TO 30
33    FTY=FTY*ZO
30    CONTINUE
C
C                     UPDATE COORDINATES
C
      IF(IOPT.NE.1) GO TO 220
      K=0
      DO 40 J=1,NP
      DO 40 M=1,2
      K=K+1
      DIS(J,M)=DIS(J,M)+BOD(K)
      CORD(J,M)=CORD(J,M)+BOD(K)
40    CONTINUE
      GO TO 225
220   CONTINUE
      K=0
      DO 42 J=1,NP
      DO 42 M=1,2
      K=K+1
      DIS(J,M)=DIS(J,M)+BOUN(K)
      CORD(J,M)=CORD(J,M)+BOUN(K)
42    CONTINUE
      NST=NROWS+1
      DO 45 J=NST,NSZF
      L=J-NROWS
      KC=JCOL(L)
      BOUN(KC)=0.0
45    CONTINUE
225   CONTINUE
C
C             UPDATE CURRENT VALUES OF STRESS AND STRAIN
```

```
C
      IF(NIO.EQ.2) GO TO 101
      DO 100 N=1,NE
      DO 50 I=1,3
50    ETA(N,I)=ETA(N,I)+ELETA(N,I)
      DO 70 I=1,4
70    SIG(N,I)=SIG(N,I)+ELSIG(N,I)
C
C              CALCULATE EQUIVALENT STRESS AND STRAIN
C
      DSEQ=((((ELSIG(N,1)-ELSIG(N,2))**2)*0.5)+
     1 (((ELSIG(N,2)-ELSIG(N,4))**2)*0.5)+(((ELSIG(N,4)-
     2 ELSIG(N,1))**2)*0.5)+ (((ELSIG(N,3))**2)*3.0))**0.5
      SEQF(N)=((((SIG(N,1)-SIG(N,2))**2)*0.5)+
     1 (((SIG(N,2)-SIG(N,4))**2)
     2 *0.5)+(((SIG(N,4)-SIG(N,1))**2)*0.5)+
     3 (((SIG(N,3))**2)*3.0))**0.5
      IF(SEQF(N).LT.SEQI(N)) GO TO 80
      EEQ(N)=EEQ(N)+DSEQ/PROP(1)
      GO TO 100
80    EEQ(N)=EEQ(N)-DSEQ/PROP(1)
100   CONTINUE
      GO TO 201
101   CONTINUE
      DO 110 N=1,NE
      DO 51 I=1,4
      ETA(N,I)=ETA(N,I)+ELETA(N,I)
51    CONTINUE
      DO 71 I=1,3
      SIG(N,I)=SIG(N,I)+ELSIG(N,I)
71    CONTINUE
C
C              CALCULATE EQUIVALENT STRESS AND STRAIN
C
      DSEQ=((((ELSIG(N,1)-ELSIG(N,2))**2)*0.5)+
     1 (((ELSIG(N,2)-ELSIG(N,4))**2)*0.5)+(((ELSIG(N,4)-
     2 ELSIG(N,1))**2)*0.5)+ (((ELSIG(N,3))**2)*3.0))**0.5
      SEQF(N)=((((SIG(N,1)-SIG(N,2))**2)*0.5)+
     1 (((SIG(N,2)-SIG(N,4))**2)
     2 *0.5)+(((SIG(N,4)-SIG(N,1))**2)*0.5)+
     3 (((SIG(N,3))**2)*3.0))**0.5
      IF(SEQF(N).LT.SEQI(N)) GO TO 81
      EEQ(N)=EEQ(N)+DSEQ/PROP(1)
      GO TO 110
81    EEQ(N)=EEQ(N)-DSEQ/PROP(1)
110   CONTINUE
201   CONTINUE
210   FORMAT(2E16.8,/,2E16.8,/,2E16.8,/,2E16.8,/,2E16.8)
      RETURN
999   WRITE(6,1000)
1000  FORMAT(/,' CHECK THE COORDINATE OF THE TOP POINT ',/)
      STOP
      END
```

```
C *********************************************************
C *                       SUBROUTINE PLAST
C *********************************************************
      SUBROUTINE PLAST
      COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1 TOL,DY,YTOT,IUZ,
     2 NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3 MIO,D1(3,3),
     4 FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5 DSY,IZO,IOT
      COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1 LPIN,INLP
      COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1 FX1,FY1,NIH,
     2 D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3 FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4 SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5 LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
      DIMENSION B(3,6),FMAT(4),GFMT(4),FTM(3),STM(3,3),
     1 DELPL(3,3),SINC(4),SBF(4),BF(6),REAC(600),TTM(3,3),
     2 X11(2500),Y11(2500),X22(2500),Y22(2500),X33(2500),
     1 Y33(2500)
C
C                       SET ITERATION COUNTER
C
      ITER=1
10    REWIND(4)
C
C                       ZERO FORCE VECTOR
C
      IF(IOPT.NE.1) GO TO 12
      DO 50 I=1,NSZF
      FORC(I)=0.0
50    REAC(I)=0.0
      GO TO 14
12    CONTINUE
      DO 51 I=1,NSZF
51    REAC(I)=0.0
14    CONTINUE
C
C                       SET UP ELEMENT LOOP
C
      DO 800 NC=1,NE
      DO 11 I=1,3
11    READ(4,19) N,(B(I,J),J=1,6)
19    FORMAT(I3,2X,6F12.3)
C
C                       BRANCH FOR YIELDED ELEMENTS
C
      IF(NIO.EQ.2) GO TO 107
      BR1=SEQF(N)-PROP(3)
      IF(BR1) 750,750,70
70    BR2=SEQI(N)-PROP(3)
      IF(BR2) 90,100,100
90    RAT=(SEQF(N)-PROP(3))/(SEQF(N)-SEQI(N))
```

```
      DO 120 I=1,3
120   ELETA(N,I)=ELETA(N,I)*RAT
      DO 130 I=1,4
130   ELSIG(N,I)=ELSIG(N,I)*RAT
      GO TO 108
107   CONTINUE
      BR1=SEQF(N)-PROP(3)
      IF(BR1) 750,750,71
71    BR2=SEQI(N)-PROP(3)
      IF(BR2) 91,100,100
91    RAT=(SEQF(N)-PROP(3))/(SEQF(N)-SEQI(N))
      DO 121 I=1,4
121   ELETA(N,I)=ELETA(N,I)*RAT
      DO 131 I=1,3
131   ELSIG(N,I)=ELSIG(N,I)*RAT
108   CONTINUE
C
C                    EVALUATE ELASTO-PLASTIC MATRIX
C
100   LYD(N)=1
      TOTS=(SIG(N,1)+SIG(N,2)+SIG(N,4))/3.0
      DEVX=SIG(N,1)-TOTS
      DEVY=SIG(N,2)-TOTS
      DEVZ=SIG(N,4)-TOTS
      FMAT(4)=(DEVZ*3.)/(SEQF(N)*2.)
      FMAT(1)=(DEVX*3.)/(SEQF(N)*2.)+PROP(2)*FMAT(4)
      FMAT(2)=(DEVY*3.)/(SEQF(N)*2.)+PROP(2)*FMAT(4)
      FMAT(3)=(SIG(N,3)*3.)/SEQF(N)
      GFMT(1)=(DEVX*3.)/(SEQF(N)*2.)
      GFMT(2)=(DEVY*3.)/(SEQF(N)*2.)
      GFMT(3)=(SIG(N,3)*3.)/SEQF(N)
      IF(NIO.EQ.2) GO TO 201
      DO 150 I=1,3
      FTM(I)=0.0
      DO 150 J=1,3
150   FTM(I)=FTM(I)+D(I,J)*FMAT(J)
      DO 200 I=1,3
      DO 200 J=1,3
200   STM(I,J)=FTM(I)*FMAT(J)
      DO 250 I=1,3
      DO 250 J=1,3
      TTM(I,J)=0.0
      DO 250 L=1,3
250   TTM(I,J)=TTM(I,J)+STM(I,L)*D(L,J)
      BRKT=0.0
      DO 300 I=1,3
300   BRKT=BRKT+FMAT(I)*FTM(I)
      TIMES=1./(BRKT+((PROP(4)*PROP(1))+
     1((FMAT(4)**2)*PROP(1))))
      DO 400 I=1,3
      DO 400 J=1,3
400   DELPL(I,J)=D(I,J)-TTM(I,J)*TIMES
      GO TO 402
201   CONTINUE
      DO 151 I=1,3
```

```
      FTM(I)=0.0
      DO 151 J=1,3
151   FTM(I)=FTM(I)+D1(I,J)*GFMT(J)
      DO 202 I=1,3
      DO 202 J=1,3
202   STM(I,J)=FTM(I)*GFMT(J)
      DO 251 I=1,3
      DO 251 J=1,3
      TTM(I,J)=0.0
      DO 251 L=1,3
251   TTM(I,J)=TTM(I,J)+STM(I,L)*D1(L,J)
      BRKT=0.0
      DO 301 I=1,3
301   BRKT=BRKT+GFMT(I)*FTM(I)
      TIMES=1./(BRKT+(PROP(4)*PROP(1)))
      DO 401 I=1,3
      DO 401 J=1,3
      DELPL(I,J)=D1(I,J)-TTM(I,J)*TIMES
401   CONTINUE
402   CONTINUE
C
C             USE MATRIX TO CALCULATE STRESS INCREMENT
C
      DO 450 I=1,3
      SINC(I)=0.0
      DO 450 J=1,3
      SINC(I)=SINC(I)+DELPL(I,J)*ELETA(N,J)
450   CONTINUE
      IF(NIO.NE.2) GO TO 501
      GO TO 552
501   CONTINUE
C
C                     CALCULATE Z-STRESS INCREMENT
C
      SINC(4)=(PROP(2)*(SINC(1)+SINC(2)))-
     1 (PROP(1)*FMAT(4)*
     2 (((PROP(1)*(ELETA(N,2)-ELETA(N,1)))+SINC(1)-
     3 (PROP(2)*(SINC(2)-
     2 SINC(1)))-SINC(2))/PROP(1))/(GFMT(2)-GFMT(1)))
C
C             CALCULATE STRESS DUE TO THE BODY FORCES
C
      DO 500 I=1,4
500   SBF(I)=ELSIG(N,I)-SINC(I)
      DO 550 I=1,4
550   SIG(N,I)=SIG(N,I)-SBF(I)
      GO TO 559
552   CONTINUE
      DO 503 I=1,3
      SBF(I)=ELSIG(N,I)-SINC(I)
503   CONTINUE
C
C                     UPDATE CURRENT STRESS VALUE
C
      DO 551 I=1,3
```

```
      SIG(N,I)=SIG(N,I)-SBF(I)
551   CONTINUE
559   CONTINUE
      SEQF(N)=((((SIG(N,1)-SIG(N,2))**2)*0.5)+
     1 (((SIG(N,2)-SIG(N,4))**2)
     2 *0.5)+(((SIG(N,4)-SIG(N,1))**2)*0.5)+
     3 (((SIG(N,3))**2)*3.0))**0.5
C
C                    CALCULATE BODY FORCES
C
      DO 650 I=1,6
      BF(I)=0.0
      DO 600 J=1,3
600   BF(I)=BF(I)+B(J,I)*SBF(J)
650   BF(I)=BF(I)*TAREA(N)
C
C             ASSEMBLE BODY FORCES IN LOAD VECTOR
C
      LP=0
      DO 700 I=1,3
      DO 700 K=1,2
      J=(NOP(N,I)-1)*2+K
      LP=LP+1
      REAC(J)=BF(LP)+REAC(J)
      IF(IOPT.NE.1) GO TO 700
      IF(BOUN(J).EQ.0.0) REAC(J)=0.0
      FORC(J)=REAC(J)
700   CONTINUE
750   SEQI(N)=SEQF(N)
800   CONTINUE
      IF(IOPT.NE.1) GO TO 801
      DO 820 NN=1,NE
      IF(LYD(NN).EQ.1) GO TO 840
820   CONTINUE
      GO TO 950
C
C                    TEST FOR CONVERGENCE
C
840   CONTINUE
      RN=0.0
      DO 875 J=1,NSZF
875   RN=RN+FORC(J)
      CONV=(ABS(RN)/ABS(FN))**0.5
C
C                    UPDATE REACTION FORCES
C
      DO 890 J=1,NPRES
      KC=JCOL(J)
      REC(KC)=REC(KC)-REAC(KC)
890   CONTINUE
      IF(CONV.LT.TOL) GO TO 950
C
C                    ITERATION COUNTER
C
      ITER=ITER+1
```

```
      IF(ITER.GT.80) GO TO 900
C
C                     LOOP THROUGH SOLUTION ROUTINES
C
      CALL SOLV2
      CALL TOTAL
      GO TO 10
900   WRITE(6,996)
      GO TO 1000
950   CONTINUE
C
C
C             DETERMINE AND PRINT ELEMENT COORDINATES
C                                  AFTER LOADING
C
      IF(ABS(YTOT).LT.BE) GO TO 142
C
C
C                     CORRECT COORDINATE
C
      IF(MIO.NE.3) GO TO 1201
      DO 1211 J=1,NP
      CORD(J,2)=CORD(J,2)-CORD(IUZ,2)
1211  CONTINUE
1201  CONTINUE
      DO 140 N=1,NE
      I=NOP(N,1)
      J=NOP(N,2)
      K=NOP(N,3)
      X11(N)=CORD(I,1)
      Y11(N)=CORD(I,2)
      X22(N)=CORD(J,1)
      Y22(N)=CORD(J,2)
      X33(N)=CORD(K,1)
      Y33(N)=CORD(K,2)
140   CONTINUE
      OPEN(UNIT=14,FILE='MESHP.DAT',STATUS='NEW')
      DO 160 N=1,NE
160   WRITE(14,210) X11(N),Y11(N),X22(N),Y22(N),
     1 X33(N),Y33(N),X11(N),Y11(N),X22(N),Y22(N)
      CLOSE(UNIT=14,STATUS='KEEP')
      ZO=CORD(IH,2)/CORD(IPR,1)
      IF(ZO.LT.1.) GO TO 32
      IF(ZO.EQ.1.) GO TO 30
      IF(ZO.GT.1.) GO TO 33
      GO TO 30
32    FX1=FX1/ZO
      GO TO 30
33    FY1=FY1*ZO
30    CONTINUE
142   CONTINUE
C
C                     SUMMATION OF REACTION FORCES
C
      DO 990 I=NST,NSZF
```

```
      N=I-NROWS
      NC=JCOL(N)
      JJ=NC
      M=1
      LP=0
960   LP=LP+1
      NC=NC-2
      IF(NC) 980,970,960
970   M=M+1
980   J=LP
990   TION(J,M)=TION(J,M)+REC(JJ)
      GO TO 809
801   CONTINUE
      NST=NROWS+1
      NPRES=NSZF-NROWS
      DO 821 NN=1,NE
      IF(LYD(NN).EQ.1) GO TO 841
821   CONTINUE
      GO TO 951
C
C                     TEST FOR CONVERGENCE
C
841   CONTINUE
      DO 851 J=1,NROWS
      NC=JROW(J)
      FORC(J)=REAC(NC)
851   CONTINUE
      RN=0.0
      DO 876 J=1,NROWS
876   RN=RN+FORC(J)
      CONV=(ABS(RN)/ABS(FN))**0.5
C
C                     UPDATE REACTION FORCES
C
      I=NST
      DO 891 J=1,NPRES
      KC=JCOL(J)
      FORC(I)=FORC(I)-REAC(KC)
      I=I+1
891   CONTINUE
      IF(CONV.LT.TOL) GO TO 951
C
C                     ITERATION COUNTER
C
      ITER=ITER+1
      IF(ITER.GT.80) GO TO 901
C
C             LOOP THROUGH SOLUTION ROUTINES
C
      CALL SOLV4
      CALL TOTAL
      GO TO 10
901   WRITE(6,996)
      GO TO 1000
951   CONTINUE
```

```
C
C
C             DETERMINE AND PRINT ELEMENT COORDINATES
C                                  AFTER LOADING
C
      IF(ABS(YTOT).LT.BE) GO TO 144
C
C
C                       CORRECT COORDINATE
C
      IF(MIO.NE.3) GO TO 1200
      DO 1210 J=1,NP
      CORD(J,2)=CORD(J,2)-CORD(IUZ,2)
1210  CONTINUE
1200  CONTINUE
      DO 146 N=1,NE
      I=NOP(N,1)
      J=NOP(N,2)
      K=NOP(N,3)
      X11(N)=CORD(I,1)
      Y11(N)=CORD(I,2)
      X22(N)=CORD(J,1)
      Y22(N)=CORD(J,2)
      X33(N)=CORD(K,1)
      Y33(N)=CORD(K,2)
146   CONTINUE
      OPEN(UNIT=14,FILE='MESHP.DAT',STATUS='NEW')
      DO 148 N=1,NE
148   WRITE(14,210) X11(N),Y11(N),X22(N),Y22(N)
     1 ,X33(N),Y33(N),X11(N),Y11(N),X22(N),Y22(N)
      CLOSE(UNIT=14,STATUS='KEEP')
      ZO=CORD(IH,2)/CORD(IPR,1)
      IF(ZO.LT.1.) GO TO 34
      IF(ZO.EQ.1.) GO TO 35
      IF(ZO.GT.1.) GO TO 36
      GO TO 30
34    FX1=FX1/ZO
      GO TO 35
36    FY1=FY1*ZO
35    CONTINUE
144   CONTINUE
C
C                       SUMMATION OF REACTION FORCES
C
      DO 991 I=NST,NSZF
      N=I-NROWS
      NC=JCOL(N)
      M=1
      LP=0
961   LP=LP+1
      NC=NC-2
      IF(NC) 981,971,961
971   M=M+1
981   J=LP
991   TION(J,M)=TION(J,M)+FORC(I)
```

```
809     CONTINUE
        WRITE(6,997) YTOT
        WRITE(6,998) DY
        WRITE(6,999) ITER
210     FORMAT(2E16.8,/,2E16.8,/,2E16.8,/,2E16.8,/,2E16.8)
996     FORMAT(//45H CONVERGENCE NOT ACHIEVED
     1  WITHIN INCREMENT  )
997     FORMAT(//18H LOAD INCREMENT = ,F10.4)
998     FORMAT(/20H CURRENT INCREMENT = ,F10.4)
999     FORMAT(/36H NO.OF ITERATION WITHIN INCREMENT =
     1   ,I5)
        RETURN
1000    STOP
        END
C *******************************************************
C *                     SUBROUTINE OUTPUT
C *******************************************************
        SUBROUTINE OUTPUT
        COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1  TOL,DY,YTOT,IUZ,
     2  NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3  MIO,D1(3,3),
     4  FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5  DSY,IZO,IOT
        COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1  LPIN,INLP
        COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1  FX1,FY1,NIH,
     2  D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3  FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4  SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5  LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
C
C                       OPTIONAL OUTPUT
C
        IF(IOT.EQ.1) GO TO 500
        IF(IOT.EQ.2) GO TO 500
        IF(IOT.EQ.3) GO TO 520
        IF(IOT.EQ.4) GO TO 500
        GO TO 590
500     CONTINUE
C
C                       PRINT DISPLACEMENTS
C
        WRITE(6,99)
        DO 2 J=1,NP
        WRITE(6,100) J,(DIS(J,M),M=1,2)
2       CONTINUE
        IF(IOT.EQ.2) GO TO 510
        IF(IOT.EQ.4) GO TO 530
C
C                       PRINT COORDINATES
        WRITE(6,101)
        DO 3 KK=1,NP
        WRITE(6,100) KK,(CORD(KK,M),M=1,2)
```

```
3       CONTINUE
C
C
C                       PRINT STRESSES
C
520     CONTINUE
        WRITE(6,199)
        DO 250 N=1,NE
        WRITE(6,200) N,(SIG(N,I),I=1,4),SEQF(N),LYD(N)
250     CONTINUE
C
C                       PRINT STRAINS
C
        WRITE(6,299)
        DO 300 N=1,NE
        WRITE(6,399) N,(ETA(N,I),I=1,4),EEQ(N)
300     CONTINUE
        IF(IOT.EQ.3) GO TO 590
        IF(IOT.EQ.1) GO TO 510
530     CONTINUE
C
C               PRINT NORMAL STRAIN ON THE SURFACE
C
        JTM=NX(1)+1
        LT=JTM+JTM
        IT=0
        JZ=JTM+4
        DO 540 NI=LT,NP,JTM
        IB=NI-JTM
        YT=CORD(NI,2)-CORD(IB,2)
        XT=CORD(IB,1)-CORD(NI,1)
        TAT=YT/XT
        THI=ATAND(TAT)
        BO1=(ETA(JZ,1)+ETA(JZ,2))/2.
        BO2=(ETA(JZ,1)-ETA(JZ,2))/2.
        THI1=2.*THI
        BO3=COSD(THI1)
        BO4=SIND(THI1)
        BO5=0.5*(BO4)*(ETA(JZ,3))
        BO6=BO2*BO3
        IT=IT+1
        EEQ(IT)=BO1+BO5+BO6
        JZ=JZ+10
540     CONTINUE
        WRITE(6,555)
        DO 550 L=1,IT
        WRITE(6,560) L,EEQ(L)
550     CONTINUE
        IF(IOT.EQ.4) GO TO 590
510     CONTINUE
C
C                       PRINT REACTIONS
C
        WRITE(6,350)
        DO 21 J=1,NP
```

```
      WRITE(6,400) J,(TION(J,M),M=1,2)
21    CONTINUE
590   CONTINUE
99    FORMAT(///7H   NODE,8X,13H DISPLACEMENTS     )
100   FORMAT(I10,2F15.6)
101   FORMAT(///7H   NODE,8X,13H COORDINATES       )
299   FORMAT(/'   ELEMENT    X-STRAIN   Y-STRAIN
     1 XY-STRAIN    Z-STRAIN   EQ-STRAIN')
399   FORMAT(I10,5E12.4)
199   FORMAT(/'   ELEMENT    X-STRESS   Y-STRESS
     1 XY-STRESS    Z-STRESS  EQ-STRESS   PLASTIC')
200   FORMAT(I10,5E12.4,I5)
350   FORMAT(/7H   NODE,8X,10H REACTIONS )
400   FORMAT(I10,2E12.4)
555   FORMAT(/,'     ELEMENT         NORMAL STRAIN')
560   FORMAT(I10,11X,E12.4)
      RETURN
      END
C *********************************************************
C *                         SUBROUTINE GRAPH
C *********************************************************
      SUBROUTINE GRAPH
      COMMON /BLOCK01/ TITLE(12),IIF,FTX,NP,NE,NSZF,
     1 TOL,DY,YTOT,IUZ,
     2 NROWS,FN,G,CE,BE,IH,LF,IPR,NX(50),JJF,JIO,NIO,
     3 MIO,D1(3,3),
     4 FTY,IPI,ISC,DISPL(600),BOD(600),REC(600),IOPT,
     5 DSY,IZO,IOT
      COMMON /BLOCK02/ NV,TG(12),KF,NIN(10),XIN(10),
     1 LPIN,INLP
      COMMON /BLOCK03/ CORD(600,2),NOP(600,3),PROP(5),
     1 FX1,FY1,NIH,
     2 D(3,3),SK(600,600),BOUN(600),TAREA(600),JROW(600),
     3 FORC(600),ETA(600,4),SIG(600,4),JCOL(600),SEQI(600),
     4 SEQF(600),ELETA(600,4),ELSIG(600,4),DIS(600,2),
     5 LYD(600),TION(600,2),EEQ(600),ESTIFM(6,6)
      DIMENSION X(2500),Y(2500),XCE(2500),YCE(2500),
     1 XC(2500),YC(2500),XC1(2500),YC1(2500),XX(2500),
     2 YY(2500)
C
C                          DRAW ELEMENTS MESH BEFORE
C                               APPLYING LOAD
C
      M=NE*5
      FD=FTY+5.
      FD1=FD-1.
      FD2=FD1-1.
      FD3=FD2-1.
      FD4=FD3-1.
      OPEN(UNIT=12,FILE ='MESH.DAT',STATUS='OLD')
      READ(12,24) (X(N),Y(N),N=1,M)
      CLOSE(UNIT=12,STATUS='KEEP')
      CALL PLOTS(0.0,0.0,6)
      CALL SCALE(X,FTX,M,1)
      CALL SCALE(Y,FTY,M,1)
```

```
      CALL AXIS(0.0,0.0,'X-AXIS',-6,FTX,0.0,X(M+1),X(M+2))
      CALL AXIS(0.0,0.0,'Y-AXIS',6,FTY,90.0,Y(M+1),Y(M+2))
      IF(ISC.EQ.2) GO TO 167
      CALL LINE(X,Y,M,1,0,0)
      GO TO 45
167   CONTINUE
      CALL DASHL(X,Y,M,1)
      CALL NEWPEN(4)
      CALL SYMBOL(2.0,FD1,0.20,'x _ plastic',,0.0,11)
      CALL NEWPEN (3)
      CALL SYMBOL(2.0,FD2,0.20,'0 _ elastic',,0.0,11)
      CALL SYMBOL(2.0,FD3,0.20,'- - - before loading',,0.0,20)
      CALL SYMBOL(2.0,FD4,0.20,'----- after loading',,0.0,19)
      GO TO 46
45    CONTINUE
      CALL NEWPEN (3)
      CALL SYMBOL(2.0,FD,0.3,%DESCR(TG),,0.0,18)
      CALL SYMBOL(2.0,FD1,0.21,'before loading',,0.0,14)
      CALL NEWPEN (4)
      CALL SYMBOL(2.0,FD2,0.20,'x _ plastic',,0.0,11)
      CALL NEWPEN (3)
      CALL SYMBOL(2.0,FD3,0.20,'0 _ elastic',,0.0,11)
      CALL PLOT(0.0,0.0,3)
      DO 166 N=1,M,5
      XCE(N)=(X(N)+X(N+1)+X(N+2))/3.
      YCE(N)=(Y(N)+Y(N+1)+Y(N+2))/3.
      XCE(N)=(XCE(N)-X(M+1))/X(M+2)
      YCE(N)=(YCE(N)-Y(M+1))/Y(M+2)
      CALL NEWPEN (3)
      CALL SYMBOL(XCE(N),YCE(N),0.20,'0',,0.0,1)
166   CONTINUE
46    CONTINUE
C
C                        DRAW ELEMENTS MESH AFTER
C                             APPLYING LOAD
C
C
      SD=FY1+5.
      SD1=SD-1.
      CALL NEWPEN (1)
      CALL PLOT(0.0,0.0,3)
      OPEN(UNIT=14,FILE ='MESHP.DAT',STATUS='OLD')
      READ(14,24) (X(N),Y(N),N=1,M)
      CLOSE(UNIT=14,STATUS='KEEP')
      IF(ISC.EQ.2) GO TO 168
      FE=FTX+2.
      GO TO 169
168   CONTINUE
      CALL PLOT(0.0,0.0,-3)
      GO TO 43
169   CONTINUE
      CALL PLOT(FE,0.0,-3)
43    CONTINUE
      CALL SCALE(X,FX1,M,1)
      CALL SCALE(Y,FY1,M,1)
```

```
      IF(ISC.EQ.2) GO TO 42
      CALL AXIS(0.0,0.0,'X-AXIS',-6,FX1,0.0,X(M+1),X(M+2))
      CALL AXIS(0.0,0.0,'Y-AXIS',6,FY1,90.0,Y(M+1),Y(M+2))
42    CONTINUE
      CALL PLOT(0.0,0.0,3)
      JJ=1
      X(JJ)=(X(JJ)-X(M+1))/X(M+2)
      Y(JJ)=(Y(JJ)-Y(M+1))/Y(M+2)
      CALL PLOT(X(JJ),Y(JJ),2)
140   JJ=JJ+1
      X(JJ)=(X(JJ)-X(M+1))/X(M+2)
      Y(JJ)=(Y(JJ)-Y(M+1))/Y(M+2)
      CALL PLOT(X(JJ),Y(JJ),2)
      JJ=JJ+1
      X(JJ)=(X(JJ)-X(M+1))/X(M+2)
      Y(JJ)=(Y(JJ)-Y(M+1))/Y(M+2)
      CALL PLOT(X(JJ),Y(JJ),2)
      JJ=JJ+1
      X(JJ)=(X(JJ)-X(M+1))/X(M+2)
      Y(JJ)=(Y(JJ)-Y(M+1))/Y(M+2)
      CALL PLOT(X(JJ),Y(JJ),2)
      JJ=JJ+1
      X(JJ)=(X(JJ)-X(M+1))/X(M+2)
      Y(JJ)=(Y(JJ)-Y(M+1))/Y(M+2)
      CALL PLOT(X(JJ),Y(JJ),2)
      IF(JJ.EQ.M) GO TO 155
      JJ=JJ+1
      X(JJ)=(X(JJ)-X(M+1))/X(M+2)
      Y(JJ)=(Y(JJ)-Y(M+1))/Y(M+2)
      CALL PLOT(X(JJ),Y(JJ),3)
      GO TO 140
155   CONTINUE
      CALL NEWPEN (3)
      CALL SYMBOL(2.0,SD,0.3,%DESCR(TG),,0.0,18)
      IF(ISC.EQ.2) GO TO 47
      CALL SYMBOL(2.0,SD1,0.21,'after loading',,0.0,13)
47    CONTINUE
      CALL PLOT(0.0,0.0,3)
C
C                         DRAW CENTRE OF THE ELEMENTS
C
      N=1
      J=1
40    IF(LYD(J).NE.1) GO TO 50
      XC(N)=(X(N)+X(N+1)+X(N+2))/3.
      YC(N)=(Y(N)+Y(N+1)+Y(N+2))/3.
      CALL NEWPEN (4)
      CALL SYMBOL(XC(N),YC(N),0.20,'x',,0.0,1)
50    J=J+1
      N=N+5
      IF(J.EQ.(NE+1)) GO TO 60
      GO TO 40
60    CONTINUE
      CALL PLOT(0.0,0.0,3)
      CALL NEWPEN (3)
```

```
      L=1
      KK=1
70    IF(LYD(KK).EQ.1) GO TO 80
      XC1(L)=(X(L)+X(L+1)+X(L+2))/3.
      YC1(L)=(Y(L)+Y(L+1)+Y(L+2))/3.
      CALL SYMBOL(XC1(L),YC1(L),0.20,'0',,0.0,1)
80    KK=KK+1
      L=L+5
      IF(KK.EQ.(NE+1)) GO TO 90
      GO TO 70
90    CONTINUE
24    FORMAT(2F16.8)
      CALL NEWPEN(1)
      CALL PLOT(0.0,0.0,999)
      RETURN
      END
C ____________________________________________________________
C ************************************************************
C *                    END OF THE PROGRAM                    *
C ____________________________________________________________
C ************************************************************
```

INPUT DATA FOR AUTOMATIC MESH GENERATING

```
$          RUN TTER
 CASE(1)  UNIFORM LOADING
2,3.,3.,4,2,3,1,2,0,2,1,2,1,1
0.2E+06,0.30,0.1692E+02,0.10,120.0
1.,1.1
3,0.1
1,1.
130,0.1
2,0.8
1,0.0,0.1
2,0.0,0.0
3,0.1E+07,0.0
4,0.1E+07,0.0
2,1.
1,0.0,0.0,0.0,1000.0
1,1000.0,0.0,1000.0,1000.0
TEST
$ EXIT
```

INPUT DATA FOR MANUAL MESH GENERATING

```
$     RUN TTER
 CASE(1) UNIFORM LOADING
2,5.,5.,4,4,1,1,2,0,2,1,2,1,2,2
0.2E+06,0.30,0.1692E+02,0.10,120.0
1.,1.1
3,0.1
1,1.
130,0.1
2,0.8
1,0.0,0.0
2,0.0,0.0
4,0.1001E+07,0.0
5,0.1001E+07,0.0
5,4
1,10.0,10.0
2,10.0,1010.0
3,510.0,510.0
4,1010.0,10.0
5,1010.0,1010.0
1,1,3,2
2,1,4,3
3,3,5,2
4,3,4,5
TEST
$ EXIT
```