# FINITE ELEMENT ANALYSIS OF STIFFENED PLATES USING MINDLIN'S THEORY



ANINDYA DEB

# FINITE ELEMENT ANALYSIS OF STIFFENED PLATES USING MINDLIN'S THEORY

BY

© Anindya Deb, B.E. (Hons.)

A thesis submitted to the School of Graduate Studies
in partial fulfillment of the
requirements for the degree of
Master of Engineering

Faculty of Engineering and Applied Science
Memorial University of Newfoundland
November, 1986

St. John's Newfoundland Canada

ISBN 0-315-36977-9

To my parents

## Abstract

Finite element procedures based on Mindlin's theory are computationally advantageous and also have the capability of accounting for transverse shear deformation in plates. Complementary to Mindlin's theory is Timoshenko's theory that accounts for transverse shear deformation in beams. In the present work, both of these shear distortion theories have been applied to the finite element analysis of stiffened plates subjected to lateral loading. Discrete plate-beam formulations, termed FEM(M1) and FEM(M2), have been set up illustrating two major approaches in the finite element analysis of stiffened plate structures. A third orthotropic formulation, named ORTHO, has been presented based on the smeared plate approach, and is applicable to the case of closely spaced torsionally soft stiffeners. The performance of the discrete plate-beam formulations, especially of the second viz. FEM(M2), has been found to be quite satisfactory based on a comparison with a number of the available results. For the first time an attempt has been made to estimate theoretically the errors that are likely to result from the use of an orthotropic formulation. This has been accomplished by comparison between ORTHO and FEM(M2) in the form of a parametric study. Additionally, the orthotropic formulation has been extended to include geometrically

non-linear behaviour. It is recognised that under less demanding conditions this latter formulation may be preferable for the reasons of economy of CPU time and simplicity of input data.

## Acknowledgements

The author wishes to express his heart-felt gratitude to his supervisor Dr. M. Booton for his guidance, encouragement and support.

The author is indebted to Dr. F.A. Aldrich, Dean of Graduate Studies, for awarding a university fellowship without which the study here would not have been possible.

Sincere thanks are due to Dr. G.R. Peters, Dean of Engineering and Applied Science and to Dr. T.R. Chari, Associate Dean of Engineering and Applied Science, for their encouragement, and additionally to the latter for his overall coordination of the author's M.Eng. program.

The author wishes to extend his deep appreciation to Dr. A.S.J. Swamidas for his persistent interest in this work, as well as his encouragement and valuable advice. It is the author's pleasure to express his gratefulness to Dr. M Mukhopadhyay, Professor and Head, Dept. of Naval Architecture, Indian Institute of Technology, Kharagpur, whose communications proved invaluable to this work.

A word of appreciation goes to Michele Walsh for typing the manuscript.

Finally, the author remembers his friends and colleagues whose companionship saw him through the times of stress.

Contents

## LIST OF TABLES

## LIST OF FIGURES

## NOTATION

| | |
|---|---|
| $u, v, w, \theta_x, \theta_y$ | Plate degrees of freedom |
| $u_{sx}, w_{sx}, \theta_{xsx}, \theta_{ysx}$ | X-Stiffener degrees of freedom |
| $v_{sy}, w_{sy}, \theta_{xsy}, \theta_{ysy}$ | Y-Stiffener degrees of freedom |
| $\bar{\varepsilon}_p$ | Plate generalised strain vector |
| $\bar{\sigma}_p$ | Plate generalised stress vector |
| $\underset{\sim}{D}_p$ | Plate constitutive matrix |
| $\bar{\delta}_p$ | Overall plate nodal displacement vector |
| $\bar{\varepsilon}_{sx}, \bar{\varepsilon}_{sy}$ | X-Stiffener and y-stiffener strain vectors respectively in FEM(M1) |
| $\bar{\sigma}_{sx}, \bar{\sigma}_{sy}$ | X-Stiffener and y-stiffener stress vectors respectively in FEM(M1) |
| $\underset{\sim}{D}_{sx}, \underset{\sim}{D}_{sy}$ | X-Stiffener and y-stiffener constitutive matrices respectively in FEM(M1) |
| $\bar{\delta}_{sx}, \bar{\delta}_{sy}$ | Overall x-stiffener and y-stiffener nodal degrees of freedom respectively in FEM(M1) |
| $e_x$ | Eccentricity of an x-stiffener centroid from the plate midsurface in FEM(M1) |
| $e_y$ | Eccentricity of a y-stiffener centroid from the plate mid-surface in FEM(M1) |
| $\underset{\sim}{T}_{sx}, \underset{\sim}{T}_{sy}$ | Transformation matrices corresponding to x- and y-stiffeners respectively in FEM(M1) |

| | |
|---|---|
| $\bar{\delta}_{pxT}$, $\bar{\delta}_{pyT}$ | Vectors of reduced plate degrees of freedom pertaining to x- and y-stiffeners respectively in FEM(M1) |
| $\underset{\sim}{K}_{p}^{(e)}$ | Plate element stiffness matrix |
| $\underset{\sim}{K}_{sx}^{(e)}$, $\underset{\sim}{K}_{sy}^{(e)}$ | X-stiffener and y-stiffener element stiffness matrices respectively in FEM(M1) |
| $\underset{\sim}{N}_{p}$ | Matrix of plate shape functions |
| $\bar{q}_{cl(g)}$ | Global consistent load vector |
| $\bar{P}_{(g)}$ | Global externally applied nodal load vector |
| x,y,z | Global cartesian coordinates |
| $\xi, \eta$ | Element natural coordinates |
| $\underset{\sim}{\Delta}_{sx}$, $\underset{\sim}{\Delta}_{sy}$ | X-stiffener and y-stiffener constitutive matrices respectively in FEM(M2). |
| $\underset{\sim}{k}_{sx}^{(e)}$, $\underset{\sim}{K}_{sy}^{(e)}$ | X-stiffener and y-stiffener element stiffness matrices respectively in FEM(M2) |
| $\underset{\sim}{D}_{o}$ | Orthotropic constitutive matrix in ORTHO |
| $\bar{\varepsilon}$ | Green's strain vector in Lagrangian coordinates |
| $\bar{\varepsilon}_{o}$ | Generalised Green's strain vector in Lagrangian coordinates in ORTHO |
| $\bar{\varepsilon}_{oL}$, $\bar{\varepsilon}_{oNL}$ | Linear, non-linear parts of $\bar{\varepsilon}_{o}$ |
| $\bar{\sigma}_{o}'$ | Generalised Kirchoff's stress vector in ORTHO |
| $\bar{\sigma}_{o}$ | Generalized Eulerian stress vector for finite deformations or Cauchy's stress vector for small deformations in ORTHO |

| | |
|---|---|
| $\bar{\Psi}$ | Residual force vector |
| $\underset{\sim}{K}_T$ | Tangential stiffness matrix |
| $\underset{\sim}{K}_L$, $\underset{\sim}{K}_{NL}$, $\underset{\sim}{K}_\sigma$ | Linear, nonlinear, geometric stiffness parts of $\underset{\sim}{K}_T$ |
| $A_o$ | Initial undeformed plate area |
| TOLER | Tolerance limit for convergence |
| $\eta_w$, $\eta_{stp}$, $\eta_{sts}$ | Orthotropic parameters pertaining to deflection, plate stress and stiffener stress respectively |
| $r_{pt}$ | Fraction of deck plate volume with respect to total volume of deck plate and stiffeners. |
| $\bar{w}$, $S_T$, $S_B$ | Normalized deflection, plate top stress and stiffener bottom stress respectively |
| $\bar{Q}$ | Normalized load |

# CHAPTER 1

## INTRODUCTION

### 1.1 Stiffened Plate Structures

A stiffened plate, like a stiffened or an unstiffened shell, is a structurally efficient form. Ever since man realised the advantages of stiffened plate structures in terms of strength, weight and aesthetics, their applications have grown and made them structures of paramount importance. Today, stiffened plates (see Figs. 1(a) and 1(b)) comprise wholly or partially such vital structures as stiffened bridge decks and box girders, ribbed floor and roof slabs, hulls and decks of ships, and, aircraft bodies and marine structures. A fairly detailed account of the evolution of such structures can be found in Troitsky [12]†.

### 1.2 Methods of Analysis

Problems involving stiffened plates have been investigated principally in the following behavioural domains: (i) bending due to transverse loading, (ii) stability under compressive and combined loadings, and (iii) free and forced vibrations. An extensive list of references pertaining to all these cases is given in Chang [6].

†A number within square brackets indicates a reference number.

The analysis of plates of arbitrary shapes under arbitrary loading and boundary conditions was by itself a formidable problem until the advent of digital computers in the early fifties. The problem of eccentrically stiffened plates, by far the most common of their kind, far exceeds the problem of unstiffened plates in the degree of complexity and inherent redundancy. This is principally so because of the presence of membrane stresses in addition to bending stresses as well as the discrete nature of stiffeners. Different idealizations to the problem were made and were restrictive in their applications. The approaches that have been mainly followed until the development of the powerful finite element method in the last two decades are summarized below:

A method of equivalent grillage was followed by many investigators and consisted in replacing the plate and beam structure by an equivalent gridwork of beams. An effective breadth of plating was included as the flange of a typical beam. Occasionally this method gave good results for deflections and stiffener stresses, but were hardly of any consequence when plate stresses were of interest. Moreover, considerable engineering judgement was required to decide upon the effective widths of stiffener-flanges.

Methods based on the orthotropic plate theory were perhaps among the most extensively used for the solution of composite plates. According to this approach, the stiffened plate system was substituted by an equivalent orthotropic plate of same thickness as the original plating and having an enhanced 'smeared stiffness'. The major problems facing this approach were the inclusion of torsional rigidity of stiffeners and separation of the beam stresses from the plate stresses in the final results. Also, the approach was inapplicable for sparse stiffeners and inaccurate for the case of concentrated loadings.

From the early ninteen-sixties, with the ushering of digital computing machines, more accurate methods requiring solutions of large linear matrix equations began to be developed. A method of this category can be termed as the discrete stiffener approach. According to this approach the displacements of the plate elements and those of the one-way stiffeners were expressed as Fourier series solutions in the direction of stiffeners, satisfying simply supported edge conditions. By substitution of these quantities and the individual plate and beam stresses obtained from them into the continuity and equilibrium conditions at the junctions between plates and stiffeners, a set of linear algebraic equations could be derived. The resultant system was then solved with the aid of electronic computing machines. An excellent investigation of this kind was presented in Smith [2]. However, this approach, other than being tedious, is plagued by its lack of generality. Around this time, some analysts also worked with the development of discrete element and lumped parameter models. For example, models for plates were suggested consisting of rigid bars and springs.

More recently, Chang [6] succeeded in deriving differential equations describing a plate-beam system applying energy principles and using dirac delta functions and methods of operational mathematics. Solutions to certain loading and boundary conditions, and configurations for torsionally soft stiffeners were obtained. Once again, despite being an elegant analytical technique and providing a check to numerical methods and experiments, this approach is not suitable for general loadings, boundary conditions and geometric shapes.

For more information on the above methods, the reader is referred to [6,13].

## 1.3 The Finite Element Method and its Relevance

The finite element method can be singled out as the most powerful tool available to date for the prediction of the complex behaviour of stiffened plates. In this versatile method, the analyst has at his/her disposal the choice of applying (a) concentrated or distributed loads or their combinations at various loading positions, (b) any combination of idealised support conditions, and (c) the most practicable modelling of arbitrary geometric shapes.

Furthermore, finite element techniques have made possible analysis of the stiffened plate problem in the non-linear range. This latter aspect is especially important for an efficient design - in predicting collapse loads and tracing the load-deformation history under geometric and materially non-linear conditions.

## 1.4 Previous Work

Literature on the finite element analysis of stiffened plates started appearing since the late nineteen-sixties and early ninteen-seventies [24, 13, 14, 15, 16, 17] and by now quite an extensive amount of work has been done on the three behavioural problems mentioned in the beginning of Sec. 1.2. Because the present work is only concerned with behaviour of such plates under transverse loading, some of the work pertinent to this category are cited.

At this point it is instructive to direct the discussion on finite element models of eccentrically stiffened plates under the following three approaches:

(i) A commonly used approach is the utilisation of multipoint constraints, that is, the so-called rigid links [18, 19, 20]. The nodes of the stiffener element, modelled as a beam, are made to undergo prescribed displacements corresponding to the displacements of the relevant plate nodes via these links. Most of the current finite element codes are based on this approach. Some of the earliest works in this area are due to McBean [14], Wegmuller [13], Lindberg [16] and Lindberg and Olson [15].

(ii) In a somewhat similar concept, elements can be generated by internally constraining the degrees of freedom of isoparametric beam elements to the displacement field of an isoparametric plate element. An approach of this kind for general applications was introduced by Mukhopadhyay [7] for the case of plane stress elements and later extended to the case of bending by Mukhopadhyay and Satsangi [8].

(iii) Rossow and Ibrahimkhail [1] approached the problem of stiffened plates in a different way from a mathematical viewpoint (although not structurally) and termed their method as the constraint method [19, 21, 22] of stiffened plate analysis. This method permits the use of conforming elements based on complete polynomials of arbitrary order. The problem of stiffened plate analysis is then reduced to a quadratic programming problem with linear equality constraints. O'Leary and Harari [23] attempted to generalise this technique by introducing isoparametric coordinates.

# CHAPTER 2

## SCOPE AND OBJECTIVE

### 2.1 Rationale for Adopting Mindlin's Theory

The application of Mindlin's theory [25] that accounts for transverse shear deformation to the finite element analysis of plates is a recent development [26, 27, 28, 29, 30, 31, 32]. In contrast to a large number of conforming and non-conforming elements based on classical thin plate theory that frequently needed six degrees of freedom per node (transverse displacement, curvatures and twist) [33] or more, elements have been developed based on Mindlin's theory that need only three degrees of freedom per node (transverse displacement and rotations of the normal to plate midsurface). The interpolation polynomials of these latter elements are easily expressed in isoparametric form. As a consequence, the finite element procedures based on this approach turn out to be strikingly advantageous computationally. Furthermore, on account of the inclusion of transverse shear deformation, the range of applicability of Mindlin elements is increased considerably, extending to the domain of thick plates and shells and sandwiched constructions. In the initial stages of their development, however, Mindlin elements had questionable performance because of presence of spurious shear modes and hence 'locking in shear' in the thin plate range. The tangle was overcome through a penalty function approach: here, a prescription of selective and reduced integration rules in the Gaussian integration leading to the generation of element stiffness matrices. As popular as Mindlin's approach is now, it is likely to remain so for a long time to come.

Until today, to the author's knowledge, only one investigation of stiffened plates based on Mindlin's theory has been reported [8] falling under the category (ii) described in Section 1.4. In the present formulations, benefits have been derived from its earlier work and models have been set up (see Sections 2.3 and 2.4) for the first two categories outlined in Section 1.4. Linearly elastic constitutive relations have been assumed for both cases. Additionally, an orthotropic element has been presented (see Section 2.5) and the formulation in this case is extended to include geometrically non-linear behaviour.

## 2.2 Assumptions

The main assumptions involved in the present analyses are stated below:

(a) Consistent with Mindlin's theory, transverse shear distortion is taken into account and is the same for the plate and the stiffeners at the relevant sections.

(b) Consistent with Kirchoff's classical thin plate theory, stresses normal to the plate midsurface are neglected; also, plane transverse sections are assumed to remain plane after bending.

(c) A typical stiffener section is assumed to be symmetric about a vertical plane bisecting the web; consequently, under a vertical loading, the stiffeners deflect vertically.

(d) The in-plane bending and shearing of stiffeners is neglected.

(e) Deformations are assumed to be small permitting a linear elastic analysis.

## 2.3 Formulation FEM(M1)

In this formulation, termed Finite Element Method (Mindlin's 1) and abbreviated FEM(M1), the concept of connecting plate and stiffener nodes with rigid links has been utilized. The stiffeners, modelled as discrete Timoshenko beam elements, are placed along the plate nodal lines. The 8-node isoparametric quadratic bending element [27, 32], the most popular of Mindlin elements, has been adopted. The formulation has been described in Chapter 3.

## 2.4 Formulation FEM(M2)

In this formulation, termed Finite Element Method (Mindlin's 2) and abbreviated as above, the isoparametric plate bending element used in FEM(M1) has been modified to include the effect of stiffeners by internally constraining the stiffener displacement field to the displacement field of the plate element. Orthogonal stiffeners have been considered and may be placed anywhere within a plate element. The description of FEM(M2) is the subject of Chapter 4.

## 2.5 Formulation ORTHO

An orthotropic formulation, abbreviated as ORTHO, has been presented in Chapter 5. According to this formulation, valid under restrictive conditions, an eccentrically stiffened plate system is replaced by an equivalent uniformly thick orthotropic plate of smeared stiffness. To this end the constitutive relation for the isotropic bending element used in FEM(M1) is modified to represent equivalent orthotropic behaviour. Orthotropic thin plate theory, in the premises of classical thin plate propositions, has been a widely applied analytical approach for the solution of the stiffened plate problem. It was therefore deemed as necessary to make the present formulation taking into account transverse shear distortion, and to evaluate its range of validity and limitations through a parametric study.

## 2.6 Computer Software

Special purpose software has been developed in FORTRAN language to implement the finite element procedures. The formulations FEM(M1) and FEM(M2) are contained in the linear Stiffened Plate Analysis Program, SPAP. Any of these formulations may be activated by specifying an option parameter in SPAP. The non-linear orthotropic formulation has been implemented via the program NLORTHO. In these programs, the elements of the system stiffness matrix are stored in a one-dimensional array and solution by Gaussian elimination is performed in the capability of an in-core solver. No special attention was given to use the most efficient numerical algorithms because the aim of the present research has been primarily to test the effectiveness of the finite element models presented.

The listings of SPAP and NLORTHO are included in the Appendices.

# CHAPTER 3

## FORMULATION FEM(M1)

### 3.1 Compatible Plate and Beam Elements

The stiffened plate, in the present model, can be conceived as an assembly of Mindlin plate and Timoshenko beam elements. The quadratic isoparametric bending element [27, 32] together with the corresponding beam elements are shown in Figs. 2(a), 2(b) and 2(c). The stiffener elements are placed along the edges of plate elements wherever necessary. The variations of the plate shape functions $N_i$ along the plate edges ($\xi = \pm 1$, $\eta = \pm 1$) are identical to the variations of the stiffener shape functions $N_{\xi i}$ and $N_{\eta i}$. Hence, the stiffeners are compatible and the convergence of the system is guaranteed.

### 3.2 Sign Conventions

The sign conventions are such that sagging moments $M_{xx}$ and $M_{yy}$ are positive; $M_{yx}$ and $N_{yx}$ are positive as shown in Fig. 3 ($M_{yx} = -M_{xy}$, $N_{yx} = -N_{xy}$); and tensile $N_{xx}$ and $N_{yy}$ are positive. The positive directions of the generalized displacements, henceforth termed only displacements, are also shown in Fig. 3.

### 3.3 Strain-displacement and Stress-strain Relations

The displacement field $\bar{\phi}_p$ for the plate element consists of five degrees of freedom:

$$\bar{\phi}_p = \begin{Bmatrix} u \\ v \\ w \\ \Theta_x \\ \Theta_y \end{Bmatrix} = \sum_{i=1}^{8} N_i \, \underset{\sim}{I} \, \bar{\delta}_{pi} \qquad (3.1)$$

where,

u, v ≡ membrane displacements respectively in the x- and y-directions;

w ≡ transverse displacement in the z-direction;

$\Theta_x, \Theta_y$ ≡ rotations of a normal to the plate midsurface about directions parallel to the y- and x-axes respectively;

$\underset{\sim}{I}$ is a 5 x 5 identity matrix;

and vector $\bar{\delta}_{pi}$ of nodal displacements is given as

$$\bar{\delta}_{pi} = \begin{Bmatrix} u \\ v \\ w \\ \Theta_x \\ \Theta_y \end{Bmatrix}_i , \quad \text{at the ith node;} \qquad (3.1a)$$

The displacement fields $\bar{\phi}_{sx}$ and $\bar{\phi}_{sy}$ for the x- and y-directional stiffeners each consists of four degrees of freedom:

For a typical x-stiffener,

$$\bar{\phi}_{sx} = \begin{Bmatrix} u_{sx} \\ w_{sx} \\ \theta_{xsx} \\ \theta_{ysx} \end{Bmatrix} = \sum_{i=1}^{3} N_{\xi i} \; \underset{\sim}{I} \; \bar{\delta}_{sxi} \tag{3.2}$$

For a typical y-stiffener,

$$\bar{\phi}_{sy} = \begin{Bmatrix} v_{sy} \\ w_{sy} \\ \theta_{xsy} \\ \theta_{ysy} \end{Bmatrix} = \sum_{i=1}^{3} N_{\eta i} \; \underset{\sim}{I} \; \bar{\delta}_{syi} \tag{3.3}$$

where $\underset{\sim}{I}$ in relations (3.2) and (3.3) is a 4x4 identity matrix, and the nodal displacement vectors $\bar{\delta}_{sxi}$ and $\bar{\delta}_{syi}$ are as follows:

$$\bar{\delta}_{sxi} = \begin{Bmatrix} u_{sx} \\ w_{sx} \\ \theta_{xsx} \\ \theta_{ysx} \end{Bmatrix}_i, \text{ at the ith node;} \tag{3.2a}$$

$$\bar{\delta}_{syi} = \begin{Bmatrix} v_{sy} \\ w_{sy} \\ \theta_{xsy} \\ \theta_{ysy} \end{Bmatrix}_i, \text{ at the ith node;} \tag{3,3a}$$

The generalized strain-displacement relation for the plate element is

$$\bar{\varepsilon}_p = \sum_{i=1}^{8} \underset{\sim}{B}_i \, \bar{\delta}_{pi} = \underset{\sim}{B}_p \, \bar{\delta}_p \tag{3.4}$$

where,

$$\bar{\varepsilon}_p^T = \left\{ \begin{array}{l} u_{,x} \\ v_{,y} \\ -(u_{,y} + v_{,x}) \\ -\theta_{x,x} \\ -\theta_{y,y} \\ (\theta_{x,y} + \theta_{y,x}) \\ (\theta_x - w_{,x}) \\ (\theta_y - w_{,y}) \end{array} \right\} \tag{3.4a}$$

$\underset{\sim}{B}_p$ ≡ plate strain-displacement matrix

$\bar{\delta}_p$ ≡ overall nodal displacement vector for a plate element

and,

$$\underset{\sim}{B}_i = \begin{bmatrix} N_{i,x} & 0 & 0 & 0 & 0 \\ 0 & N_{i,y} & 0 & 0 & 0 \\ -N_{i,y} & -N_{i,x} & 0 & 0 & 0 \\ 0 & 0 & 0 & -N_{i,x} & 0 \\ 0 & 0 & 0 & 0 & -N_{i,y} \\ 0 & 0 & 0 & N_{i,y} & N_{i,x} \\ 0 & 0 & -N_{i,x} & N_i & 0 \\ 0 & 0 & -N_{i,y} & 0 & N_i \end{bmatrix} \tag{3.4b}$$

Note that a comma following a subscript indicates a spatial differentiation with respect to the subsequent variable. Also, a superscript T denotes 'transpose'.

The generalized stress-strain relationship for the plate element now follows [29]:

$$\bar{\sigma}_p = D_p \bar{\varepsilon}_p = D_p B_p \bar{\delta}_p \quad \text{(using (3.4))} \tag{3.5}$$

where,

$$\bar{\sigma}_p^T = \{N_{xx}\ N_{yy}\ N_{xy}\ M_{xx}\ M_{yy}\ M_{xy}\ Q_{xz}\ Q_{yz}\} \tag{3.5a}$$

and the non-zero elements $(D_p)_{ij}$ of the 8x8 symmetric constitutive matrix are as follows:

$$(D_p)_{11} = \frac{Et}{1-\nu^2} = (D_p)_{22};\ (D_p)_{12} = \nu(D_p)_{11} = (D_p)_{21};$$

$$(D_p)_{33} = Gt;\ (D_p)_{44} = \frac{Et^3}{12(1-\nu^2)} = (D_p)_{55};\ (D_p)_{45} =$$

$$\nu(D_p)_{44} = (D_p)_{54};$$

$$(D_p)_{66} = \frac{1-\nu}{2}(D_p)_{44};\ (D_p)_{77} = \frac{Gt}{1.2} = (D_p)_{88}.$$

The quantities E, t, $\nu$ and G have the following meanings:

E ≡ Young's modulus for the plate material;

t ≡ Plate thickness;

$\nu$ ≡ Poisson's ratio for the plate material; and,

G ≡ Shear modulus $(= \frac{E}{2(1+\nu)})$

It may be noted that a divisor 1.2 appears in the expression for $(D_p)_{77}$ to account for the non-uniform shear stress distribution across a transverse section.

The generalised stress-strain relations for x- and y-stiffeners follow similarly:

$$\bar{\sigma}_{sx} = \underset{\sim}{D}_{sx} \bar{\varepsilon}_{sx} = \underset{\sim}{D}_{sx} \underset{\sim}{B}_{sx} \bar{\delta}_{sx} \tag{3.6}$$

$$\bar{\sigma}_{sy} = \underset{\sim}{D}_{sy} \bar{\varepsilon}_{sy} = \underset{\sim}{D}_{sy} \underset{\sim}{B}_{sy} \bar{\delta}_{sy} \tag{3.7}$$

where,

$$\bar{\sigma}^T_{sx} = \{N_{sxx} \quad M_{sxx} \quad T_{sx} \quad Q_{sxz}\} \tag{3.6a}$$

$$\bar{\varepsilon}^T_{sx} = \{u_{sx,x} - \theta_{xsx,x} \quad \theta_{ysx,x} \quad (\theta_{xsx} - w_{sx,x})\} \tag{3.6b}$$

$\bar{\delta}_{sx}$ ≡ vector of overall nodal displacements of an x-stiffener

$$\underset{\sim}{B}_{sx} = \sum_{i=1}^{3} \begin{bmatrix} N_{\xi i,x} & 0 & 0 & 0 \\ 0 & 0 & -N_{\xi i,x} & 0 \\ 0 & 0 & 0 & N_{\xi i,x} \\ 0 & -N_{\xi i,x} & N_{\xi i} & 0 \end{bmatrix} \tag{3.6c}$$

$$\bar{\sigma}^T_{sy} = \{N_{syy} \quad M_{syy} \quad T_{sy} \quad Q_{syz}\} \tag{3.7a}$$

$$\bar{\varepsilon}^T_{sy} = \{v_{sy,y} - \theta_{ysy,y} \quad \theta_{xsy,y} \quad (\theta_{ysy} - w_{sy,y})\} \tag{3.7b}$$

$\bar{\delta}_{sy}$ ≡ vector of overall nodal displacements of a y-stiffener

$$\underset{\sim}{B}_{sy} = \sum_{i=1}^{3} \begin{bmatrix} N_{\eta i,x} & 0 & 0 & 0 \\ 0 & 0 & 0 & -N_{\eta i,y} \\ 0 & 0 & N_{\eta i,y} & 0 \\ 0 & -N_{\eta i,y} & 0 & -N_{\eta i} \end{bmatrix} \quad (3.7c)$$

and the non-zero values $(D_{sx})_{ij}$ and $(D_{sy})_{ij}$ of the 4x4 constitutive matrices $\underset{\sim}{D}_{sx}$ and $\underset{\sim}{D}_{sy}$ are as follows:

$$(D_{sx})_{11} = E_{sx}A_{sx};\ (D_{sx})_{22} = E_{sx}I_{sx};\ (D_{sx})_{33} = G_{sx}J_{sxe};\ (D_{sx})_{44} = \frac{G_{sx}A_{sx}}{1.5};$$

$$(D_{sy})_{11} = E_{sy}A_{sy};\ (D_{sy})_{22} = E_{sy}I_{sy};\ (D_{sy})_{33} = G_{sy}J_{sye};\ (D_{sy})_{44} = \frac{G_{sy}A_{sy}}{1.5};$$

where,

$E_{sx}$, $E_{sy}$ ≡ Young's moduli for x- and y-stiffeners

$A_{sx}$, $A_{sy}$ ≡ Cross-sectional areas of x- and y-stiffeners

$I_{sx}$, $I_{sy}$ ≡ Centroidal moments of inertia of x- and y-stiffeners

$G_{sx}$, $G_{sy}$ ≡ Shear moduli of x- and y-stiffeners

$J_{sxe}$, $J_{sye}$ ≡ Equivalent polar moments of inertia of x- and y-stiffeners.

It may be noted that a divisor 1.5 appears in $(D_{sx})_{44}$ and $(D_{y})_{44}$ to account for the warping of stiffener cross-sections.

## 3.4 Global Equilibrium Equations

Assuming that loads are applied over the plate surface and nodes, a global system of linear simultaneous equations can be derived using the virtual work principle (which is an alternative statement of the minimum potential energy theorem):

$$\sum^{NP} \int \partial\bar{\epsilon}_p^T \bar{\sigma}_p \, dA + \sum^{NSX} \int \partial\bar{\epsilon}_{sx}^T \bar{\sigma}_{sx} \, dx + \sum^{NSY} \int \partial\bar{\epsilon}_{sy}^T \bar{\sigma}_{sy} \, dy$$

$$= \sum^{NP} \int \partial\bar{\phi}_p^T \bar{q} \, dA + \sum^{NP} \partial\bar{\delta}_p^T \bar{P} \qquad (3.8)$$

where,

NP, NSX, NSY ≡ Total number of plate, x- stiffener and y- stiffener elements respectively

$$\bar{q} = \begin{Bmatrix} 0 \\ 0 \\ q_z \\ 0 \\ 0 \end{Bmatrix} \equiv \text{Vector of distributed transverse loading}$$

$\bar{P}$ ≡ Vector of externally applied concentrated nodal loads at an element level.

In equation (3.8), '$\partial$' represents first variation and '$\Sigma$' represents summation over elements. Also, it is understood that the integrals are area or line integrals according as the case may be.

The following relations define the multipoint rigid linkages by means of which the slave stiffener degrees of freedom are eliminated in favour of the master plate nodal degrees of freedom from equation (3.8):

$$\bar{\delta}_{sx} = \sum_{i=1}^{3} \begin{bmatrix} 1 & 0 & -e_x & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} u \\ w \\ \theta_x \\ \theta_y \end{Bmatrix}_i = \underset{\sim}{T}_{sx} \bar{\delta}_{pxT} \quad (3.9)$$

$$\bar{\delta}_{sy} = \sum_{i=1}^{3} \begin{bmatrix} 1 & 0 & 0 & -e_y \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} v \\ w \\ \theta_x \\ \theta_y \end{Bmatrix}_i = \underset{\sim}{T}_{sy} \bar{\delta}_{pyT} \quad (3.10)$$

In relations (3.9) and (3.10), $\bar{\delta}_{pxT}$ and $\bar{\delta}_{pyT}$ are the 12x1 vectors of reduced plate nodal degrees of freedom corresponding to x- and y-stiffeners respectively.

Proper substitutions in equation (3.8) now yield

$$\sum^{NP} \int \partial\bar{\delta}_p^T (\underset{\sim}{B}_p \underset{\sim}{D}_p \underset{\sim}{B}_p) \bar{\delta}_p \, dA +$$

$$\sum^{NSX} \int \partial\bar{\delta}_{pxT}^T (\underset{\sim}{T}_{sx}^T \underset{\sim}{B}_{sx}^T \underset{\sim}{D}_{sx} \underset{\sim}{B}_{sx} \underset{\sim}{T}_{sx}) \bar{\delta}_{pxT} \, dx$$

$$+ \sum^{NSY} \int \partial\bar{\delta}_{pyT}^T (\underset{\sim}{T}_{sy}^T \underset{\sim}{B}_{sy}^T \underset{\sim}{D}_{sy} \underset{\sim}{B}_{sy} \underset{\sim}{T}_{sy}) \bar{\delta}_{pyT} \, dy =$$

$$\sum^{NP} \int \partial\bar{\delta}_p^T \underset{\sim}{N}_p \bar{q} \, dA + \sum^{NP} \partial\bar{\delta}_p^T \bar{P} \quad (3.11)$$

The following element stiffness matrices are defined:

$$\text{Plate stiffness matrix, } \underset{\sim}{K}_p^{(e)} = \int \underset{\sim}{B}_p^T \underset{\sim}{D}_p \underset{\sim}{B}_p \, dA \qquad (3.12)$$

Stiffness matrix for an x-stiffener,

$$\underset{\sim}{K}_{sx}^{(e)} = \int \underset{\sim}{T}_{sx}^T \underset{\sim}{B}_{sx}^T \underset{\sim}{D}_{sx} \underset{\sim}{B}_{sx} \underset{\sim}{T}_{sx} \, dx \qquad (3.13)$$

Stiffness matrix for a y-stiffener,

$$\underset{\sim}{K}_{sy}^{(e)} = \int \underset{\sim}{T}_{sy}^T \underset{\sim}{B}_{sy}^T \underset{\sim}{D}_{sy} \underset{\sim}{B}_{sy} \underset{\sim}{T}_{sy} \, dy \qquad (3.14)$$

The superscript (e) indicates that the above matrices are defined at the element level. Substitution of equations (3.12), (3.13) and (3.14) in equation (3.11) yields:

$$\sum^{NP} \partial\bar{\delta}_p^T \underset{\sim}{K}_p^{(e)} \bar{\delta}_p + \sum^{NSX} \partial\bar{\delta}_{pxT}^T \underset{\sim}{K}_{sx}^{(e)} \bar{\delta}_{pxT} + \sum^{NSY} \partial\bar{\delta}_{pyT}^T \underset{\sim}{K}_{sy}^{(e)} \bar{\delta}_{pyT}$$

$$= \sum^{NP} \partial\bar{\delta}_p^T \int \underset{\sim}{N}_p^T \bar{q} \, dA + \sum^{NP} \partial\bar{\delta}_p^T \bar{P} \qquad (3.15)$$

Equation (3.15) is now rewritten by augmenting all the matrices and vectors therein to the global size, with a notation (g) in subscript or superscript indicating 'global':

$$\partial\bar{\delta}_{p(g)}^T \underset{\sim}{K}_p^{(g)} \bar{\delta}_{p(g)} + \partial\bar{\delta}_{p(g)}^T \underset{\sim}{K}_{sx}^{(g)} \bar{\delta}_{p(g)} + \partial\bar{\delta}_{p(g)}^T \underset{\sim}{K}_{sy}^{(g)} \bar{\delta}_{p(g)}$$

$$= \partial\bar{\delta}_{p(g)}^T \bar{q}_{c\ell(g)} + \partial\bar{\delta}_{p(g)}^T \bar{P}_{(g)} \qquad (3.16)$$

where,

global consistent load vector, $\bar{q}_{c\ell(g)} = \sum^{NP} \int N_p^T \bar{q} \, dA$ (3.16a)

Equation (3.16) is finally reduced to

$$\underset{\sim}{K} \, \bar{\delta}_{p(g)} = \bar{R} \quad (3.17)$$

where,

$$\underset{\sim}{K} = \underset{\sim}{K}_p^{(g)} + \underset{\sim}{K}_{sx}^{(g)} + \underset{\sim}{K}_{sy}^{(g)} \quad (3.17a)$$

$$\bar{R} = \bar{q}_{c\ell(g)} + \bar{P}_{(g)} \quad (3.17b)$$

3.5 Numerical Integration and Stress Extrapolation

The element stiffness matrices in equations (3.12), (3.13) and (3.14) are computed by numerical integration in natural coordinates. This involves transforming the differential areas and lengths in global cartesian coordinates to those in the element coordinates via the relevant Jacobian matrices.

For the plate bending element, the mapping from x, y-coordinates to ξ, η-coordinates is defined as:

$$\begin{Bmatrix} x \\ y \end{Bmatrix} = \sum_{i=1}^{8} \begin{bmatrix} N_i & 0 \\ 0 & N_i \end{bmatrix} \begin{Bmatrix} x \\ y \end{Bmatrix}_i \quad (3.18)$$

and the corresponding Jacobian of transformation is

$$\underset{\sim}{J}_{xy} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \qquad (3.19)$$

where,

$$\frac{\partial x}{\partial \xi} = \sum_{i=1}^{8} \frac{\partial N_i}{\partial \xi} x_i, \quad \frac{\partial x}{\partial \eta} = \sum_{i=1}^{8} \frac{\partial N_i}{\partial \eta} x_i, \quad \frac{\partial y}{\partial \xi} = \sum_{i=1}^{8} \frac{\partial N_i}{\partial \xi} y_i \text{ and}$$

$$\frac{\partial y}{\partial \eta} = \sum_{i=1}^{8} \frac{\partial N_i}{\partial \eta} y_i, \ \{(x_i, y_i), \ i = 1 \text{ to } 8\} \text{ being the}$$

element nodal coordinates.

For an x-stiffener,

$$x = \sum_{i=1}^{3} N_{\xi i} x_i \qquad (3.20)$$

and the Jacobian of transformation is

$$J_x = \frac{\partial x}{\partial \xi} \qquad (3.21)$$

where $\frac{\partial x}{\partial \xi} = \sum_{i=1}^{3} \frac{\partial N_{\xi i}}{\partial \xi} x_i$, $x_i$ ($i = 1$ to 3) being the global x-stiffener nodal coordinates.

The Jacobian $J_y$ pertaining to a y-stiffener follows similarly by replacing x and $\xi$ in relations (3.21) and (3.22) by y and $\eta$ respectively. Note that inverses of the Jacobians $\underset{\sim}{J}_{xy}$, $J_x$ and $J_y$ are required in the computation of cartesian derivatives of the shape functions.

The element stiffness matrices in relations (3.12)-(3.14) can now be rewritten as:

$$K_p^{(e)} = \int B_p D_p B_p \, dA = \int B_p D_p B_p \, |J_{xy}| \, d\xi d\eta \qquad (3.22)$$

$$K_{sx}^{(e)} = \int T_{sx}^T B_{sx}^T D_{sx} B_{sx} T_{sx} \, dx =$$

$$\int T_{sx}^T B_{sx}^T D_{sx} B_{sx} T_{sx} J_x \, d\xi \qquad (3.23)$$

$$K_{sy}^{(e)} = \int T_{sy}^T B_{sy}^T D_{sy} B_{sy} T_{sy} \, dy =$$

$$\int T_{sy}^T B_{sy}^T D_{sy} B_{sy} T_{sy} J_y \, d\eta \qquad (3.24)$$

The integrations on the extreme right hand sides of the above relations are performed using a reduced 2-point Gaussian quadrature rule. The element stiffness matrices are then assembled somewhat in the style of SAP IV [34], incorporating the geometric boundary conditions in the process, and the resulting matrix equation of the type in equation (3.18) is solved by Gaussian elimination. If distributed loads are present, the consistent nodal load vector is evaluated using an exact 2x2 Gaussian quadrature rule. Finally, the correct values of the stresses/stress-resultants are computed at the Gaussian integration points used in element stiffness matrix generations and bilinearly/linearly extrapolated to the element nodes [28, 35].

# CHAPTER 4

## Formulation FEM (M2)

### 4.1 Description of the Element

In the formulation presented here, the isoparametric plate bending element described in Chapter 3 is transformed into a stiffened plate bending element capable of accomodating internal orthogonal stiffeners (see Fig. 4). This is achieved by constraining the displacement field of a typical stiffener element to that of the plate element to which it is attached [8]. A typical stiffener element may be imagined to have 'pseudo nodes' at its ends whose displacements are expressible in terms of the displacements of the eight nodes of the relevant plate element. Thus the effect of a stiffener element within a plate element is in general felt at all of its (plate's) eight nodes. The stiffness terms of a typical plate element along with contributions from stiffeners are summed up to obtain the resultant stiffness matrix of the stiffened plate bending element under consideration.

### 4.2 Plate-Stiffness Inclusion

The generalised strain-displacement and stress-strain relations are identical to those presented in

Chapter 3 for the 8-noded quadratic element (refer to relations 3.4 and 3.5). Consequently, the plate element stiffness matrix is given by relation (3.22) and is evaluated numerically using a reduced 2x2 Gaussian quadrature rule.

## 4.3 Inclusion of Stiffeners

Incorporation of the stiffness of an x-stiffener is demonstrated below. An analogous procedure may be followed for a y-stiffener.

The following displacement field is assumed for an x-stiffener with respect to the global axes of reference lying on the plate midsurface:

$$\bar{\phi}_{sx} = \begin{Bmatrix} u_{sx} \\ w_{sx} \\ \theta_{xsx} \\ \theta_{ysx} \end{Bmatrix} = \begin{Bmatrix} u - z\theta_x \\ w \\ \theta_x \\ \theta_y \end{Bmatrix} \qquad (4.1)$$

The above notations have been explained previously in Chapter 3.

The relevant strain components (assuming that stiffeners do not contribute to the inplane shearing stiffness of the plate) are:

$$\begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{xz} \end{Bmatrix} = \begin{Bmatrix} u_{sx,z} \\ u_{sx,z} + w_{sx,x} \end{Bmatrix} = \begin{Bmatrix} u_{,x} - z\theta_{x,x} \\ w_{,x} - \theta_x \end{Bmatrix} \qquad (4.2)$$

The linearly elastic stress-strain relation follows from elementary strength of materials:

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{xz} \end{Bmatrix} = \begin{bmatrix} E_{sx} & 0 \\ 0 & \frac{G_{sx}}{1.5} \end{bmatrix} \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{xz} \end{Bmatrix} = \begin{bmatrix} E_{sx} & 0 \\ 0 & \frac{G_{sx}}{1.5} \end{bmatrix} \begin{Bmatrix} u_{,x} - z\theta_{x,x} \\ w_{,x} - \theta_x \end{Bmatrix} \tag{4.3}$$

where $E_{sx}$ and $G_{sx}$ are the Young's and shear moduli respectively for the x-stiffener material and 1.5 is a shear correction factor.

The generalised stresses (stress-resultants) are now evaluated by performing integration of the pertinent quantities over the cross-section of an x-stiffener (following the sign conventions adopted in section 3.2):

$$\begin{Bmatrix} N_{sxx} \\ M_{sxx} \\ Q_{sxz} \end{Bmatrix} = \int \begin{Bmatrix} \sigma_{xx} \\ z\sigma_{xx} \\ -\sigma_{xz} \end{Bmatrix} dA_{sx} = \int \begin{bmatrix} 1 & 0 \\ z & 0 \\ 0 & -1 \end{bmatrix} \begin{Bmatrix} \sigma_{xx} \\ \sigma_{xz} \end{Bmatrix} dA_{sx}$$

$$= \int \begin{bmatrix} 1 & 0 \\ z & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} E_{sx} & 0 \\ 0 & \frac{G_{sx}}{1.5} \end{bmatrix} \begin{Bmatrix} u_{,x} - z\theta_{x,x} \\ w_{,x} - \theta_x \end{Bmatrix} dA_{sx},$$

using relation (4.3).

$$= \int \begin{bmatrix} 1 & 0 \\ z & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} E_{sx} & 0 \\ 0 & \frac{G_{sx}}{1.5} \end{bmatrix} \begin{bmatrix} 1 & z & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{Bmatrix} u_{,x} \\ -\theta_{x,x} \\ \theta_x - w_{,x} \end{Bmatrix} dA_{sx}$$

$$= \int \begin{bmatrix} E_{sx} & zE_{sx} & 0 \\ zE_{sx} & z^2E_{sx} & 0 \\ 0 & 0 & \frac{G_{sx}}{1.5} \end{bmatrix} \begin{Bmatrix} u_{,x} \\ -\theta_{x,x} \\ \theta_x - w_{,x} \end{Bmatrix} dA_{sx} \qquad (4.4)$$

where $N_{sxx}$, $M_{sxx}$ and $Q_{sxz}$ are membrane, bending and transverse shear resultants respectively, and $dA_{sx}$ is the differential of cross-sectional area $A_{sx}$ of a typical x-stiffener.

Incorporating St. Venant's torsion in an approximate way (that is, considering open web sections with ends not restrained against warping so that an equivalent polar moment of inertia or torsional rigidity factor $J_{sxe}$ can be assigned), relation (4.4) can be modified as:

$$\begin{Bmatrix} N_{sxx} \\ M_{sxx} \\ T_{sx} \\ Q_{sxz} \end{Bmatrix} = \begin{bmatrix} (\Delta_{sx})_{11} & & \text{Symmetric} & \\ (\Delta_{sx})_{12} & (\Delta_{sx})_{22} & & \\ 0 & 0 & (\Delta_{sx})_{33} & \\ 0 & 0 & 0 & (\Delta_{sx})_{44} \end{bmatrix} \begin{Bmatrix} u_{,x} \\ -\theta_{x,x} \\ \theta_{y,x} \\ \theta_x - w_{,x} \end{Bmatrix} \qquad (4.5)$$

where the torsional moment $T_{sx}$ has been included; also, $(\Delta_{sx})_{11} = \int E_{sx}\, dA_{sx}$, $(\Delta_{sx})_{12} = \int zE_{sx}\, dA_{sx}$, $(\Delta_{sx})_{22} = \int z^2E_{sx}\, dA_{sx}$, $(\Delta_{sx})_{33} = G_{sx}J_{sxe}$ and $(\Delta_{sx})_{44} = \int \frac{G_{sx}}{1.5}\, dA_{sx}$.

In particular, for a T-section shown in Fig. 4, we have

$$(\Delta_{sx})_{11} = \int E_{sx}\, dA_{sx} = E_{sx}A_{sx} = E_{sx}\,(b_{wx}d_x + b_{fx}t_{fx}) \quad (4.5a)$$

$$(\Delta_{sx})_{12} = \int z\, E_{sx}\, dA_{sx} = E_{sx}b_{wx} \int_{t/2}^{d_x + t/2} dz + E_{sx}b_{fx} \int_{d_x + t/2}^{d_x + t/2 + t_{fx}} z dz$$

$$= \frac{E_{sx}}{2}\,[b_{wx}d_x\,(d_x + t) + b_{fx}t_{fx}\,(2d_x + t + t_{fx})] \quad (4.5b)$$

$$(\Delta_{sx})_{22} = \int z^2 E_{sx}\, dA_{sx} = E_{sx}b_{wx} \int_{t/2}^{d_x + t/2} z^2 dz + E_{sx}b_{fx} \int_{d_x + t/2}^{d_x + t/2 + t_{fx}} z^2 dz$$

$$= \frac{1}{3} E_{sx}\,[b_{wx}\,\{(d_x + \tfrac{t}{2})^3 - (\tfrac{t}{2})^3\} + b_{fx}\,\{(d_x + \tfrac{t}{2} + t_{fx})^3 - (d_x + \tfrac{t}{2})^3\}] \quad (4.5c)$$

$$(\Delta_{sx})_{33} = G_{sx}J_{sxe} = G_{sx}\,[c_1 b_{wx}\, d_x^3 + c_2\, t_{fx}\, b_{fx}^3] \quad (4.5d)$$

where $c_1$ and $c_2$ are correction factors dependent on the ratios $d_x/b_{wx}$ and $b_{fx}/t_{fx}$ (see Table 5.31 of Ref. [42])

$$(\Delta_{sx})_{44} = \int \frac{G_{sx}}{1.5} dA_{sx} = \frac{G_{sx}A_{sx}}{1.5} \tag{4.5e}$$

where

$t$ ≡ plate thickness

$b_{wx}, d_x$ ≡ width and depth respectively of web of an x-stiffener

$b_{fx}, t_{fx}$ ≡ width and thickness respectively of flange of the x-stiffener.

The generalized stress-strain relation expressed by (4.5) is written in the following compact form:

$$\bar{\sigma}_{sx} = \underset{\sim}{\Delta}_{sx} \bar{\varepsilon}_{sx} \tag{4.6}$$

where

$$\bar{\sigma}_{sx}^T = \{N_{sxx}\ M_{sxx}\ T_{sx}\ Q_{sxz}\} \tag{4.6a}$$

$$\bar{\varepsilon}_{sx}^T = \{u_{,x} - \theta_{x,x}\ \theta_{y,x}\ (\theta_x - w_{,x})\} \tag{4.6b}$$

and $\underset{\sim}{\Delta}_{sx}$ is the 4x4 symmetric constitutive matrix with components $(\Delta_{sx})_{ij}$.

Rewriting equation (4.6b), we have,

$$\bar{\varepsilon}_{sx} = \begin{Bmatrix} u_{,x} \\ -\theta_{x,x} \\ \theta_{y,x} \\ \theta_x - w_{,x} \end{Bmatrix} = \sum_{i=1}^{8} \begin{bmatrix} N_{i,x} & 0 & 0 & 0 \\ 0 & 0 & -N_{i,x} & 0 \\ 0 & 0 & 0 & N_{i,x} \\ 0 & -N_{i,x} & N_i & 0 \end{bmatrix} \begin{Bmatrix} u \\ w \\ \theta_x \\ \theta_y \end{Bmatrix}_i$$

$$= \underset{\sim}{B}_{sx} \bar{\delta}_{psx} \tag{4.7}$$

where

$$\underset{\sim}{B}_{sx} = \sum_{i=1}^{8} \begin{bmatrix} N_{i,x} & 0 & 0 & 0 \\ 0 & 0 & -N_{i,x} & 0 \\ 0 & 0 & 0 & N_{i,x} \\ 0 & -N_{i,x} & N_i & 0 \end{bmatrix}, \tag{4.7a}$$

$\bar{\delta}_{psx}$ is a reduced vector of plate nodal displacements pertaining to an x-stiffener element and $N_i$ ($i$ = 1 to 8) are the shape functions of the quadratic element of Fig. 2(a).

The following relations will result for a y-stiffener from proceeding analogously as above:

$$\bar{\sigma}_{sy} = \underset{\sim}{A}_{sy} \bar{\varepsilon}_{sy} \tag{4.8}$$

$$\bar{\varepsilon}_{sy} = \underset{\sim}{B}_{sy} \bar{\delta}_{psy} \tag{4.9}$$

where

$$\bar{\sigma}_{sy}^T = \{N_{syy} \; M_{syy} \; T_{sy} \; Q_{syz}\} \tag{4.8a}$$

$$\bar{\varepsilon}_{sy}^{T} = \{ v_{,y} - \theta_{y,y} \ \theta_{x,y} \ (\theta_y - w_{,y}) \} \tag{4.8b}$$

$$\underset{\sim}{\beta}_{sy} = \sum_{i=1}^{8} \begin{bmatrix} N_{i,y} & 0 & 0 & 0 \\ 0 & 0 & 0 & -N_{i,y} \\ 0 & 0 & N_{i,y} & 0 \\ 0 & -N_{i,y} & 0 & N_i \end{bmatrix}, \tag{4.9a}$$

$\bar{\delta}_{psy}$ is a reduced vector of plate nodal displacements pertaining to a typical y-stiffener and the components of the 4x4 symmetric constitutive matrix $\underset{\sim}{\Delta}_{sy}$ can be obtained from relations (4.5a) through (4.5e) by replacing x by y and attaching similar meanings to the quantities occurring therein.

The element stiffness matrices are now defined:

For an x-stiffener,

$$\underset{\sim}{k}_{sx}^{(e)} = \int \underset{\sim}{\beta}_{sx}^{T} \underset{\sim}{\Delta}_{sx} \underset{\sim}{\beta}_{sx} \, dx = \int \underset{\sim}{\beta}_{sx}^{T} \underset{\sim}{\Delta}_{sx} \underset{\sim}{\beta}_{sx} J_x \, d\xi \tag{4.10}$$

For a y-stiffener,

$$k_{sy}^{(e)} = \int \underset{\sim}{\beta}_{sy}^{T} \underset{\sim}{\Delta}_{sy} \underset{\sim}{\beta}_{sy} \, dy = \int \underset{\sim}{\beta}_{sy}^{T} \underset{\sim}{\Delta}_{sy} \underset{\sim}{\beta}_{sy} J_y \, d\eta \tag{4.11}$$

where $J_x$ and $J_y$ are the Jacobians of transformation in one dimension. Note that the expressions for element stiffness matrices as given by the right hand sides of relations (4.10) and (4.11) can be alternatively obtained by applying virtual work principle (refer to section 3.4) at an element level.

The integrals in relations (4.10) and (4.11) are numerically evaluated following a reduced 2-point Gaussian quadrature rule. Prior to integration, the respective isoparametric coordinates $\eta_r$ and $\xi_r$ of the rth x- and y-stiffeners are to be determined and substituted in all the relevant quantities. For example, for a typical orthogonal rth x-stiffener in a rectangular/square mesh,

$$\eta_r = \frac{y_r - 0.5\ (y_2 + y_6)}{0.5\ (y_2 - y_6)} \tag{4.12}$$

where

$y_r$ ≡ global y- coordinate of the rth x- stiffener

$y_2, y_6$ ≡ global y- coordinates of the 2nd and 6th nodes respectively of the plate element to which the rth x-stiffener is attached.

## 4.4 Global Equilibrium Equations, Solution and Stress Extrapolation

The application of the virtual work principle to the assembly of stiffened plate elements as elaborated in Section 3.4 will lead to the equilibrium equations and will immediately reveal that the total global stiffness of the system is an accumulation of the stiffnesses of the individual stiffened plate elements of proper locations. The solution procedure for the linear system of simultaneous equations is identical to that described in Section 3.5.

Stresses are calculated accurately at the Gaussian integration points of the individual plate elements and extrapolated bilinearly to their nodes where they are averaged. Stresses for stiffeners are also calculated at the Gaussian integration points of the stiffener elements and extrapolated linearly to their pseudo nodes and algebraically averaged.

# CHAPTER 5

## FORMULATION ORTHO

### 5.1 Orthotropic Constitutive Relation

In addition to the general assumptions stated in Sec. 2.2, the following assumptions are made in the present formulation:

(i) Stiffeners are orthogonal and are equally and closely spaced.

(ii) Being of open web and slender type, stiffeners contribute insignificantly to the torsional stiffness of the plate.

The following displacement fields are now considered with reference to the displacements u, v, w, $\theta_x$ and $\theta_y$ at any point in the plate midsurface:

$$\begin{Bmatrix} U \\ V \\ W \end{Bmatrix} = \begin{Bmatrix} u(x,y) - z\theta_x(x,y) \\ v(x,y) - z\theta_y(x,y) \\ w(x,y) \end{Bmatrix} \tag{5.1}$$

The relevant components of the engineering strain tensor are:

$$\bar{\varepsilon}^T = \{\varepsilon_{xx}\ \varepsilon_{yy}\ \varepsilon_{xy}\ \varepsilon_{yz}\ \varepsilon_{xz}\}$$

$$= \{(U_{,x}\ \ V_{,y}\ \ (U_{,y} + V_{,x}),\ (U_{,z} + W_{,x})\ \ (V_{,z} + W_{,y})\}$$

$$= \{(u_{,x} - z\Theta_{x,x})\ (v_{,y} - z\Theta_{y,y})\ [(u_{,y} + v_{,x}) - z(\theta_{x,y} + \theta_{y,x})]$$

$$(w_{,x} - \theta_x)\ (w_{,y} - \theta_y)\}, \qquad (5.2)$$

using relation (5.1).

Assuming isotropic material, stresses in the plate are given as:

$$\begin{Bmatrix} \sigma_{pxx} \\ \sigma_{pyy} \\ \sigma_{pxy} \\ \sigma_{pxz} \\ \sigma_{pyz} \end{Bmatrix} = \begin{bmatrix} D_{11} & D_{12} & 0 & 0 & 0 \\ & D_{22} & 0 & 0 & 0 \\ \text{symmetric} & & D_{33} & 0 & 0 \\ & & & D_{44} & \\ & & & & D_{55} \end{bmatrix} \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{xy} \\ \varepsilon_{xz} \\ \varepsilon_{yz} \end{Bmatrix} \qquad (5.3)$$

where $D_{11} = \frac{E}{1-\nu^2}$, $D_{12} = \frac{\nu E}{1-\nu^2}$, $D_{33} = G$, $D_{44} = \frac{G}{1.2}$ and $D_{55} = \frac{G}{1.2}$ and the elastic constants E, G and $\nu$ have been explained before (following equation (3.5)).

Stresses in the x-stiffeners are:

$$\begin{Bmatrix} \sigma_{sxx} \\ \sigma_{sxz} \end{Bmatrix} = \begin{bmatrix} E_{sx} & 0 \\ 0 & \frac{G_{sx}}{1.5} \end{bmatrix} \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{xz} \end{Bmatrix} \qquad (5.4)$$

where the elastic constants $E_{sx}$ and $G_{sx}$ have been explained before (following relation (3.7c)).

Stresses in the y-stiffeners are:

$$\begin{Bmatrix} \sigma_{syy} \\ \sigma_{syz} \end{Bmatrix} = \begin{bmatrix} E_{sy} & 0 \\ 0 & \frac{G_{sy}}{1.5} \end{bmatrix} \begin{Bmatrix} \varepsilon_{yy} \\ \varepsilon_{yz} \end{Bmatrix} \qquad (5.5)$$

where the elastic constants $E_{sy}$ and $G_{sy}$ have the same meaning as their counterparts for x-stiffeners in relation (5.4).

The generalized stresses are obtained by smearing out the stiffeners over the plate spans and integrating the relevant quantities over the plate thickness and stiffener depths. Typically, the bending moment $M_{xx}$ can be obtained as shown below assuming rectanglar stiffeners (see Fig. 6);

$$M_{xx} = \int_{-t/2}^{t/2} z\sigma_{pxx} dz + \frac{n_{sx} b_{wx}}{L_y} \int_{t/2}^{t/2+d_x} z\sigma_{sxx}\, dz$$

$$= \frac{E}{1-\nu^2} \int_{-t/2}^{t/2} [(zu_{,x} - z^2\Theta_{x,x}) + \nu(zv_{,y} - z^2\Theta_{y,y})]\, dz$$

$$+ \frac{n_{sx} b_{wx}}{Ly} E_{sx} \int_{t/2}^{t/2+d_x} (zu_{,x} - z^2\theta_{u,x})\, dz,$$ using relations (5.3) and (5.4)

$$= \frac{-Et^3}{12(1-\nu^2)} (\Theta_{x,x} + \nu\theta_{y,y}) + r_x E_{sx} [\tfrac{1}{2}\, d_{sx}(t + d_x)\, u_{,x} -$$

$$\frac{1}{3} d_x (d_x^2 + 1.5 d_x t + 0.75 t^2) \Theta_{x,x}\}$$

$$= \frac{1}{2} r_x E_{sx} d_x (t + d_x) u_{,x} - \{\frac{Et^3}{12(1-\nu^2)}$$

$$+ \frac{1}{3} r_x E_{sx} d_x (d_x^2 + 1.5 d_x t + 0.75 t^2)\} \Theta_{x,x} - \frac{\nu E t^3}{12(1-\nu^2)} \Theta_{y,y} \quad (5.6)$$

where, $n_{sx}$ ≡ number of x-stiffeners

$b_{wx}$ ≡ web width of an x-stiffener

$L_y$ ≡ plate span in the y-direction

$r_x = \frac{n_{sx} b_{sx}}{L_y}$ ≡ smearing ratio for x-stiffeners

The remaining stress-resultants can be obtained similarly leading to a constitutive relation of the following type for the linear orthotropic plate:

$$\bar{\sigma}_o = \underset{\sim}{D}_o \bar{\varepsilon}_{oL} \quad (5.7)$$

where,

$\bar{\sigma}_o$ and $\bar{\varepsilon}_{oL}$ are respectively identical to $\bar{\sigma}_p$ and $\bar{\varepsilon}_p$ defined in relations (3.5a) and (3.4a), and the non-zero elements $(D_o)_{ij}$ of the orthotropic constitutive matrix $\underset{\sim}{D}_o$ are given below:

$$(D_o)_{11} = \frac{Et}{1-\nu^2} + r_x E_{sx} d_x \quad (5.7a)$$

$$(D_o)_{12} = \frac{\nu Et}{1-\nu^2} \equiv (D_o)_{21} \quad (5.7b)$$

$$(D_o)_{14} = \frac{1}{2} r_s E_{sx} d_x (t + d_x) = (D_o)_{41} \quad (5.7c)$$

$$(D_o)_{22} = \frac{Et}{1-\nu^2} + r_y E_{sy} d_y \quad (5.7d)$$

$$(D_o)_{25} = \frac{1}{2} r_y E_{sy} d_y (t + d_y) = (D_o)_{52} \quad (5.7e)$$

$$(D_o)_{33} = Gt \tag{5.7f}$$

$$(D_o)_{44} = \frac{Et^3}{12(1-\nu^2)} + \frac{1}{3} r_x E_{sx} d_x (d_x^2 + 1.5\ d_x t + 0.75\ t^2) \tag{5.7g}$$

$$(D_o)_{45} = \frac{\nu Et^3}{12(1-\nu^2)} = (D_o)_{54} \tag{5.7h}$$

$$(D_o)_{55} = \frac{Et^3}{12(1-\nu^2)} + \frac{1}{3} r_y E_{sy} d_y (d_y^2 + 1.5\ d_y t + 0.75\ t^2) \tag{5.7i}$$

$$(D_o)_{66} = \frac{Et^3}{24(1+\nu)} \tag{5.7j}$$

$$(D_o)_{77} = \frac{5}{6} Gt + \frac{2}{3} r_x G_{sx} d_x \tag{5.7k}$$

$$(D_o)_{88} = \frac{5}{6} Gt + \frac{2}{3} r_y G_{sy} d_y \tag{5.7ℓ}$$

## 5.2 The Smeared Plate Element

The quadratic isoparametric plate element described in Sec. 3.1 is used to interpolate the independent degrees of freedom, u, v, w, $\theta_x$ and $\theta_y$ in terms of the corresponding nodal displacements. In conjunction with the generalised stress-strain relation indicated in (5.7), this plate element can be made to behave as a smeared orthotropic plate element.

## 5.3 Incremental Strain-displacement Relationship for Large Deflections

The relevant components of the Green's strain tensor (in Lagrangian coordinates) for finite deformations in an affine space are:

$$\bar{\varepsilon} = \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{xy} \\ \varepsilon_{xz} \\ \varepsilon_{yz} \end{Bmatrix} = \begin{Bmatrix} u_{,x} + \frac{1}{2} u_{,x}^2 + \frac{1}{2} v_{,x}^2 + \frac{1}{2} w_{,x}^2 \\ v_{,y} + \frac{1}{2} u_{,y}^2 + \frac{1}{2} v_{,y}^2 + \frac{1}{2} w_{,y}^2 \\ u_{,y} + v_{,x} + u_{,x}u_{,y} + v_{,x}v_{,y} + w_{,x}w_{,y} \\ u_{,z} + w_{,x} + u_{,x}u_{,z} + v_{,x}v_{,z} + w_{,x}w_{,z} \\ v_{,z} + w_{,y} + u_{,y}u_{,z} + v_{,y}v_{,z} + w_{,y}w_{,z} \end{Bmatrix} \quad (5.8)$$

Introducing von Karman's assumptions which imply that the membrane derivatives are small in comparison with derivatives of w and since the latter is independent of z, equation (5.8) can be written in the following simplified form:

$$\bar{\varepsilon} = \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{xy} \\ \varepsilon_{xz} \\ \varepsilon_{yz} \end{Bmatrix} = \begin{Bmatrix} u_{,x} + \frac{1}{2} w_{,x}^2 \\ v_{,y} + \frac{1}{2} w_{,y}^2 \\ u_{,y} + v_{,x} + w_{,x} w_{,y} \\ u_{,z} + w_{,x} \\ v_{,z} + w_{,y} \end{Bmatrix} \quad (5.9)$$

The generalized Green's strain vector corresponding to the above will be:

$$\bar{\varepsilon}_o = \begin{Bmatrix} u_{,x} \\ v_{,y} \\ -(u_{,y} + v_{,x}) \\ -\theta_{x,x} \\ -\theta_{y,y} \\ \theta_{x,y} + \theta_{y,x} \\ \theta_x - w_{,x} \\ \theta_y - w_{,y} \end{Bmatrix} + \begin{Bmatrix} \frac{1}{2} w_{,x}^2 \\ \frac{1}{2} w_{,y}^2 \\ -w_{,x} w_{,y} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} = \bar{\varepsilon}_{oL} + \bar{\varepsilon}_{oNL} \qquad (5.10)$$

where it is clear that $\bar{\varepsilon}_{oL}$ and $\bar{\varepsilon}_{oNL}$ respectively correspond to the linear and non-linear parts of the total strain vector $\bar{\varepsilon}_o$. Now, $\bar{\varepsilon}_{oL}$ is identical to $\bar{\varepsilon}_p$ given in relation (3.4a). Hence, following relation (3.4),

$$\bar{\varepsilon}_{oL} = \underset{\sim}{B}_{pL} \bar{\delta}_p \qquad (5.11)$$

where $\underset{\sim}{B}_{pL}$ is identical to $\underset{\sim}{B}_p$ in relation (3.4).

Equation (5.10) is re-written on substitution of $\bar{\varepsilon}_{oL}$ from relation (5.11):

$$\bar{\varepsilon}_o = B_{pL} \bar{\delta}_p + \bar{\varepsilon}_{oNL} \qquad (5.12)$$

Considering the first variation in strain field due to a first variation in displacement field, we have

$$\partial\bar{\varepsilon}_o = \underset{\sim}{B}_{pL} \partial\bar{\delta}_p + \partial\bar{\varepsilon}_{oNL} \qquad (5.12)$$

Now,

$$\partial \bar{\varepsilon}_{oNL} = \partial \begin{Bmatrix} \frac{1}{2} w_{,x}^2 \\ \frac{1}{2} w_{,y}^2 \\ -w_{,x} w_{,y} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} = \begin{bmatrix} w_{,x} & 0 \\ 0 & -w_{,y} \\ -w_{,y} & -w_{,x} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \partial \begin{Bmatrix} w_{,x} \\ w_{,y} \end{Bmatrix}$$

$$= \underset{\sim}{A} \, \partial \begin{Bmatrix} w_{,x} \\ w_{,y} \end{Bmatrix} \tag{5.13}$$

where

$$\underset{\sim}{A}^T = \begin{bmatrix} w_{,x} & 0 & -w_{,y} & 0 & 0 & 0 & 0 & 0 \\ 0 & w_{,y} & -w_{,x} & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.13a}$$

Further,

$$\partial \begin{Bmatrix} w_{,x} \\ w_{,y} \end{Bmatrix} = \partial \begin{bmatrix} 0 & 0 & \frac{\partial}{\partial x} & 0 & 0 \\ 0 & 0 & \frac{\partial}{\partial y} & 0 & 0 \end{bmatrix} \begin{Bmatrix} u \\ v \\ w \\ \Theta_x \\ \Theta_y \end{Bmatrix}$$

$$= \sum_{i=1}^{8} \begin{bmatrix} 0 & 0 & N_{i,x} & 0 & 0 \\ 0 & 0 & N_{i,y} & 0 & 0 \end{bmatrix} \partial \bar{\delta}_p, \text{ using}$$

relation (3.1) and $\bar{\delta}_p$ is defined after relation (3.4a).

$$= \underset{\sim}{G} \; \partial\bar{\delta}_p \tag{5.14}$$

where

$$\underset{\sim}{G} = \begin{bmatrix} \sum_{i=1}^{8} & 0 & 0 & N_{i,x} & 0 & 0 \\ & 0 & 0 & N_{i,y} & 0 & 0 \end{bmatrix} \tag{5.14a}$$

Relation (5.13) is now re-written with the help of relation (5.14):

$$\partial\bar{\varepsilon}_{oNL} = \underset{\sim}{A} \; \underset{\sim}{G} \; \partial\bar{\delta}_p \tag{5.15}$$

Substituting $\partial\varepsilon_{oNL}$ from (5.15) in (5.12), we obtain

$$\partial\bar{\varepsilon}_o = \underset{\sim}{B}_{pL} \; \partial\bar{\delta}_p + \underset{\sim}{A} \; \underset{\sim}{G} \; \partial\bar{\delta}_p = (\underset{\sim}{B}_{pL} + \underset{\sim}{B}_{pNL}) \; \partial\bar{\delta}_p$$

$$= \underset{\sim}{B}_{oNL} \; \partial\bar{\delta}_p \tag{5.16}$$

where

$$\underset{\sim}{B}_{pNL} = \underset{\sim}{A} \; \underset{\sim}{G} \tag{5.16a}$$

and,

$$\underset{\sim}{B}_{oNL} = \underset{\sim}{B}_{pL} + \underset{\sim}{B}_{pNL} \tag{5.16b}$$

Equation (5.16) above defines the incremental strain-displacement relation.

## 5.4 Non-linear Equilibrium Equations

The stress tensor that corresponds to the Green's strain tensor is the Kirchoff's stress tensor in material

(Lagrangian) coordinates. Corresponding to the generalized Green's strain vector $\bar{\varepsilon}_o$, we have the generalized Kirchoff's stress vector $\bar{\sigma}'_o$, and the two are connected as follows under elastic conditions:

$$\bar{\sigma}'_o = \underset{\sim}{D}_o \, \bar{\varepsilon}_o \qquad (5.17)$$

where,

$$\bar{\sigma}'_o = \{N'_{xx} \; N'_{yy} \; N'_{xy} \; M'_{xx} \; M'_{yy} \; M'_{xy} \; Q'_{xz} \; Q'_{yz}\}; \qquad (5.17a)$$

$\underset{\sim}{D}_o$ has been defined in relation (5.7) and $\bar{\varepsilon}_o$ has been defined incrementally in relation (5.16). Note that $\bar{\sigma}'_o$ in relation (5.17) is different from $\bar{\sigma}_o$ in relation (5.7) in the sense that $\bar{\sigma}'_o$ is strictly referred to the undeformed configuration whereas $\bar{\sigma}_o$ corresponds to the generalized Cauchy stresses in the deformed state. Physically speaking, the final stresses should be Eulerian, i.e., with respect to the deformed state even for the case of finite deformations. Hence, a conversion from Kirchoff's to Eulerian stresses will be necessary. However, as a first order approximation (since all the displacement derivatives are small in comparison with unity) [36], $\bar{\sigma}_o$ can be approximated by $\bar{\sigma}'_o$. Thus, equation (5.17) is modified as follows:

$$\bar{\sigma}_o \simeq \bar{\sigma}'_o = \underset{\sim}{D}_o \, \bar{\varepsilon}_o \qquad (5.18)$$

As a condition of stable equilibrium, the virtual work principle is invoked at an element level following a total Lagrangian approach:

$$\int_{A_o} \partial \bar{\varepsilon}_o^T \; \bar{\sigma}_o \; dA = \partial \bar{\delta}_p^T \; \bar{R} \qquad (5.19)$$

where $A_o$ is the undeformed plate midsurface area and $\bar{R}$ is equivalent to the expression in (3.17b) but at an element level.

Employing relation (5.16) in equation (5.19), we have,

$$\int_{A_o} \partial \bar{\delta}_p^T \; \underset{\sim}{B}_{oNL}^T \; \bar{\sigma}_o \; dA = \partial \bar{\delta}_p^T \; \bar{R}$$

That is, for a finite virtual displacement field,

$$\int_{A_o} \underset{\sim}{B}_{oNL}^T \; \bar{\sigma}_o \; dA - \bar{R} = \bar{0} \qquad (5.20)$$

In the above equilibrium equations (5.20), $\underset{\sim}{B}_{oNL}^T$ and $\bar{\sigma}_o$ are respectively linear and quadratic functions of $\bar{\delta}_p$. Hence, (5.20) represents a set of non-linear equations in $\bar{\delta}_p$. Although (5.20) has been derived at an element level, it can also represent, in an assembled form, the global equilibrium equations.

## 5.5 Newton-Raphson Solution Algorithm

Equation (5.20) can be solved iteratively starting with an initial guess (at i=1) for $\bar{\delta}_p$ and then improving the value of $\bar{\delta}_p^i$ at every (i+1)th step following a Newton-Raphson algorithm [37]. At any stage of iteration i, equation (5.20) is in general not satisfied and is equal to a residue $\bar{\psi}(\bar{\delta}_p^i)$ computed on the basis of $\bar{\delta}_p^i$, i.e.,

$$\bar{\psi}(\bar{\delta}_p^i) = \int_{A_o} \underset{\sim}{B}_{oNL}^T \, \bar{\sigma}_o \, dA \, (\bar{\delta}_p^i) - \bar{R} \neq 0 \qquad (5.21)$$

The aim is to reduce $\bar{\psi}(\bar{\delta}_p^i)$ in equation (5.21) to a maximum tolerance limit (TOLER) and this is attempted through a linearised Taylor's expansion of $\bar{\psi}$ in the neighbourhood of $\bar{\delta}_p^i$ being equated to zero:

$$\bar{\psi}(\bar{\delta}_p^{i+1}) = \bar{\psi}(\bar{\delta}_p^i) + \underset{\sim}{K}_T \, \Delta\bar{\delta}_p^i = 0 \qquad (5.22)$$

where

$$\underset{\sim}{K}_T(\bar{\delta}_p^i) = \frac{\partial \bar{\psi}}{\partial \bar{\delta}_p}(\bar{\delta}_p^i) \qquad (5.22a)$$

$\underset{\sim}{K}_T$ in relation (5.22a) is called the tangent stiffness matrix.

On solution of $\Delta\bar{\delta}_p^i$ from equation (5.22), $\bar{\delta}_p$ is updated:

$$\bar{\delta}_p^{i+1} = \bar{\delta}_p^i + \Delta\bar{\delta}_p^i \qquad (5.23)$$

Correspondingly, stresses are also updated as in the following relations:

$$\Delta\bar{\sigma}_o^i = \underset{\sim}{D}_o \, \underset{\sim}{B}_{oNL} \, \Delta\bar{\delta}_p^i \qquad (5.24)$$

$$\bar{\sigma}_o^{i+1} = \bar{\sigma}_o^i + \Delta\bar{\sigma}_o^i \qquad (5.25)$$

An updated value of $\bar{\psi}$ is calculated from equation (5.21) and the steps outlined by equations (5.22) through (5.25) are repeated until convergence is attained. In order to improve numerical stability and to obtain intermediate results, the load $\bar{R}$ is usually applied in increments. A modified Newton-Raphson scheme may also be employed according to which the updating of $\underset{\sim}{K}_T$ in relation (5.22a) is carried out at fixed intervals instead of at every iteration step.

## 5.6 Tangent Stiffness Matrix

Assuming that the applied loading $\bar{R}$ is conservative, the tangent stiffness matrix in relation (5.22a) can be evaluated as follows:

Rewriting equation (5.21),

$$\bar{\psi} = \int_{A_o} \underset{\sim}{B}^T \bar{\sigma} \, dA - \bar{R}$$

$$= \int_{A_o} \left\{ \begin{array}{c} \sum_{k=1}^{8} B_{k1} \sigma_k \\ \sum_{k=1}^{8} B_{k2} \sigma_k \\ \cdot \\ \cdot \\ \cdot \\ \sum_{k=1}^{8} B_{km} \sigma_k \end{array} \right\} dA - \bar{R} \quad (5.26)$$

where

$$\underset{\sim}{B} = \underset{\sim}{B}_{oNL} \quad (5.26a)$$

$$\bar{\sigma} = \{\sigma_1\ \sigma_2\ \sigma_3\ \sigma_4\ \sigma_5\ \sigma_6\ \sigma_7\ \sigma_8\}^T = \bar{\sigma}_o \quad (5.26b)$$

and, m is the total number of degrees of freedom per element.

Thus, any typical element of $\bar{\psi}$ is

$$\psi_i = \int_{A_o} \sum_{\ell=1}^{8} B_{\ell i}\, \sigma_\ell\, dA - R_i \quad (5.27)$$

Assume any typical element $(K_T)_{ij}$ of $\underset{\sim}{K}_T$ to be

$$(K_T)_{ij} = \int_{A_o} k_{ij} dA \quad (5.28)$$

From relations (5.22a), (5.27) and (5.28), we have

$k_{ij} = \frac{\partial \psi_i}{\partial \delta_j}$, where $\psi_i$ and $\delta_j$ are typical elements of $\bar{\psi}$ and $\bar{\delta}_p$ respectively.

$$= \sum_{\ell=1}^{8} \left[\frac{\partial B_{\ell i}}{\partial \delta_j} \sigma_\ell + B_{\ell i} \frac{\partial \sigma_\ell}{\partial \delta_j}\right] \qquad (5.28a)$$

That is,

$$\underset{\sim}{k} = \underset{\sim}{B}^T \underset{\sim}{L}(\bar{\sigma}) + \underset{\sim}{M}(\bar{\sigma}) \underset{\sim}{B} = \underset{\sim}{B}_{ONL}^T \underset{\sim}{L}(\bar{\sigma}_o) + M(\bar{\sigma}_o) \underset{\sim}{B}_{ONL} \qquad (5.29)$$

using (5.26a) and (5.26b)

where

$$\underset{\sim}{L}(\bar{\sigma}_o) = \begin{bmatrix} \frac{\partial \sigma_1}{\partial \delta_1} & \frac{\partial \sigma_1}{\partial \delta_2} & \cdot & \cdot & \cdot & \cdot & \frac{\partial \sigma_1}{\partial \delta_m} \\ \frac{\partial \sigma_2}{\partial \delta_1} & \frac{\partial \sigma_1}{\partial \delta_2} & \cdot & \cdot & \cdot & \cdot & \frac{\partial \sigma_2}{\partial \delta_m} \\ \cdot & \cdot & & & & & \cdot \\ \cdot & \cdot & & & & & \cdot \\ \cdot & \cdot & & & & & \cdot \\ \frac{\partial \sigma_8}{\partial \delta_1} & \frac{\partial \sigma_8}{\partial \delta_2} & \cdot & \cdot & \cdot & \cdot & \frac{\partial \sigma_8}{\partial \delta_m} \end{bmatrix} \qquad (5.29a)$$

$$\underset{\sim}{M}(\bar{\sigma}_o) = \begin{bmatrix} \sigma_1 \frac{\partial}{\partial \delta_1} & \sigma_2 \frac{\partial}{\partial \delta_1} & \cdot & \cdot & \cdot & \sigma_8 \frac{\partial}{\partial \delta_1} \\ \sigma_1 \frac{\partial}{\partial \delta_2} & \sigma_2 \frac{\partial}{\partial \delta_2} & \cdot & \cdot & \cdot & \sigma_8 \frac{\partial}{\partial \delta_2} \\ \cdot & \cdot & & & & \cdot \\ \cdot & \cdot & & & & \\ \cdot & \cdot & & & & \\ \cdot & \cdot & & & & \\ \sigma_1 \frac{\partial}{\partial \delta_m} & \sigma_2 \frac{\partial}{\partial \delta_m} & \cdot & \cdot & \cdot & \sigma_8 \frac{\partial}{\partial \delta_m} \end{bmatrix} \qquad (5.29b)$$

From (5.18), we have

$$\partial\bar{\sigma}_o = \underset{\sim}{D}_o \partial\bar{\varepsilon}_o \tag{5.30}$$

Substituting $\partial\bar{\varepsilon}_o$ from (5.16) in (5.30), we obtain

$$\partial\bar{\sigma}_o = \underset{\sim}{D}_o \underset{\sim}{B}_{oNL} \; \partial\bar{\delta}_p \tag{5.31}$$

Comparing relations (5.31) and (5.29a) it is evident that

$$\underset{\sim}{L}(\bar{\sigma}_o) = \underset{\sim}{D}_o \underset{\sim}{B}_{oNL} \tag{5.32}$$

Now, the second term of the right side of equation (5.29),

$$\underset{\sim}{M}(\bar{\sigma}_o) \underset{\sim}{B}_{oNL} = \underset{\sim}{M}(\sigma_o) (\underset{\sim}{B}_{pL} + \underset{\sim}{B}_{pNL}),$$ using relation (5.16b)

$$= \underset{\sim}{M}(\bar{\sigma}_o) \underset{\sim}{B}_{pNL},$$ since $\underset{\sim}{B}_{pL}$ is independent of $\sigma_o$.

$$= \underset{\sim}{M}(\bar{\sigma}_o) \underset{\sim}{A} \underset{\sim}{G}, \tag{5.33}$$

using (5.16a).

Now, it can be shown that

$$\underset{\sim}{M}(\bar{\sigma}_o) \underset{\sim}{A} = \underset{\sim}{G}^T \begin{bmatrix} \sigma_1 & -\sigma_3 \\ -\sigma_3 & \sigma_2 \end{bmatrix}$$

$$= \underset{\sim}{G}^T \begin{bmatrix} N_x & -N_{xy} \\ -N_{xy} & N_y \end{bmatrix} \tag{5.34}$$

Substituting results from (5.32), (5.33) and (5.34) in the right side of equation (5.29), we obtain

$$\underset{\sim}{k} = \underset{\sim}{B}^T_{oNL} \underset{\sim}{D}_o \underset{\sim}{B}_{oNL} + \underset{\sim}{G}^T \begin{bmatrix} N_x & -N_{xy} \\ -N_{xy} & N_y \end{bmatrix} \underset{\sim}{G}$$

$$= \underset{\sim}{k}_L + \underset{\sim}{k}_{NL} + \underset{\sim}{k}_\sigma \tag{5.35}$$

where

$$\underset{\sim}{k}_L = \underset{\sim}{B}_{pL}^T \underset{\sim}{D}_o \underset{\sim}{B}_{pL} \tag{5.35a}$$

$$\underset{\sim}{k}_{NL} = \underset{\sim}{B}_{pL}^T \underset{\sim}{D}_o \underset{\sim}{B}_{pNL} + \underset{\sim}{B}_{pNL}^T \underset{\sim}{D}_o \underset{\sim}{B}_{pL} + \underset{\sim}{B}_{pNL}^T \underset{\sim}{D}_o \underset{\sim}{B}_{pNL} \tag{5.35b}$$

$$\underset{\sim}{k}_\sigma = \underset{\sim}{G}^T \begin{bmatrix} N_x & -N_{xy} \\ -N_{xy} & N_y \end{bmatrix} \underset{\sim}{G} \tag{5.35c}$$

Finally, from equation (5.38), we obtain the tangent stiffness matrix in the following form:

$$\underset{\sim}{K}_T = \int_{A_o} (\underset{\sim}{k}_L + \underset{\sim}{k}_{NL} + \underset{\sim}{k}_\sigma)\, dA$$

$$= \underset{\sim}{K}_L + \underset{\sim}{K}_{NL} + \underset{\sim}{K}_\sigma \tag{5.36}$$

where

$$\underset{\sim}{K}_L = \int_{A_o} \underset{\sim}{k}_L\, dA \tag{5.36a}$$

$$\underset{\sim}{K}_{NL} = \int_{A_o} \underset{\sim}{k}_{NL}\, dA \tag{5.36b}$$

$$\underset{\sim}{K}_\sigma = \int_{A_o} \underset{\sim}{k}_\sigma\, dA \tag{5.36c}$$

In the above, $\underset{\sim}{K}_L$ is the linear stiffness matrix; $\underset{\sim}{K}_{NL}$ is quadratically dependent on $\bar{\delta}_p$; and, $\underset{\sim}{K}_\sigma$ is the initial stress or geometric stiffness matrix.

## 5.7 Convergence Criteria

The following convergence tests were employed by specifying a tolerance limit (TOLER):

$$\frac{[\bar{\psi}(\bar{\delta}_p^i) * \bar{\psi}(\bar{\delta}_p^i)]^{1/2}}{[\bar{R} * \bar{R}]^{1/2}} < \text{TOLER}, \qquad (5.37)$$

which is a residual norm check.

$$\frac{[\bar{\delta}_p^{i+1} * \bar{\delta}_p^{i+1} - \bar{\delta}_p^i * \bar{\delta}_p^i]^{1/2}}{[\bar{\delta}_p^1 * \bar{\delta}_p^1]^{1/2}} < \text{TOLER}, \qquad (5.38)$$

which is a displacement norm check.

Note that the vectors above are assumed as global in size and the symbol * indicates a dot product.

## 5.8 Numerical Integration and Stress Extrapolation

The integrals in equations (5.21), (5.36a), (5.36b) and (5.36c) are numerically evaluated using a reduced 2-point Gaussian quadrature rule. The incremental stress-resultants given by equation (5.24) and the corresponding incremental plate and stiffener stresses are all calculated at the 2x2 Gaussian integration points and bilinearly extrapolated to the nodes. An option is provided in the program NLORTHO by means of which non-linear analysis can be skipped and only a linear analysis can be performed.

# CHAPTER 6

## NUMERICAL STUDIES

### 6.1 Convergence study

The results of a convergence study on a quarter plate for the formulations FEM(M1) and FEM(M2) are presented in Fig. 7. The problem considered is described in Section 6.2 below. It is interesting to note that for this particular case the formulations FEM(M1) and FEM(M2) yield identical results as expected. As revealed in Fig. 7, both the formulations possess good convergence characteristics.

### 6.2 Example 1

A simply supported rectangular steel plate stiffened centrally by two orthogonal stiffeners was originally considered in [6] and used for comparison in [1]. A quarter plate analysis was carried out with a 3x6 mesh using formulations FEM(M1) and FEM(M2). Although matching with the present formulations is excellent for the case of uniformly distributed loading (refer to Figs. 8-10), there is a significant variation for the case of the central point load (refer to Figs. 11-13). A major contributing factor to this deviation is probably the consideration of transverse shear deformation in the present models.

6.3 Example 2

A rectangular plate simply supported at its longer edges and free at its shorter edges, with nine evenly spaced T-bar stiffeners across its shorter spans and subjected to a central concentrated load was analysed in [2]. Results obtained through FEM(M1) and FEM(M2) based on a quarter plate analysis with a 4x4 mesh are presented in Fig. 14. Good agreement with the results given in Fig. 12 of [2] is observed.

6.4 Example 3

A rectangular stiffened steel plate supported at the corners and subjected to a total concentrated load of 10 tons has been analysed with a 5x4 mesh for the entire plate using formulation FEM(M2). This problem is shown in Fig. 1 in [3]. It is observed from Figs. 15, 16 and 18 that deflections obtained herein are on the lower side (within about 14%) as compared to the experimental values but in close agreement with the theoretical results both given in [3]. The computed stresses (refer to Figs. 17 and 19), however, have matched quite well with the experimental values cited in [3].

6.5 Example 4

A slab with edge beams cast in Araldite is shown in Fig. 1 in [5]. Experimental and theoretical values of deflections and stresses for three different cases are given in Table 2 in [5]. With a 5x5 mesh for the whole plate, results obtained through FEM(M2) are presented in Table 1. The computed results agree within a maximum of about 15.5% and less than 6% respectively with the corresponding experimental and theoretical values given in [5].

### 6.6 Comparison Between ORTHO and FEM(M2)

The aim of this investigation is to determine the extent to which the orthotropic formulation may be applied to obtain acceptable results. For this purpose, the various parameters likely to affect the assumptions of orthotrophy are identified and a comparative study with the more accurate formulation FEM(M2) is carried out by varying these parameters. In order to facilitate a systematic study, the simplified problem of a square plate orthogonally stiffened with rectangular stiffeners of identical sectional and material properties is considered. In order to estimate the deviation of maximum deflection and plate as well as stiffener stresses from the orthotropic formulation with respect to those from the discrete plate-beam formulation FEM(M2), the following orthotropic parameters are introduced:

$$\eta_w = \frac{\text{Maximum linear deflection obtained through ORTHO}}{\text{Maximum linear deflection obtained through FEM(M2)}} \tag{6.1}$$

$$\eta_{stp} = \frac{\text{Maximum linear plate stress obtained through ORTHO}}{\text{Maximum linear plate stress obtained through FEM(M2)}} \tag{6.2}$$

$$\eta_{sts} = \frac{\text{Maximum linear stiffener stress obtained through ORTHO}}{\text{Maximum linear stiffener stress obtained through FEM(M2)}} \tag{6.3}$$

Let the parameters $\eta_w$, $\eta_{stp}$ and $\eta_{sts}$, which are all dimensionless, be together denoted as $\eta$. The dependence of $\eta$ on other relevant parameters may ~~be~~ expressed symbolically as follows:

$$\eta = \phi(\ell, t, b_s, d_s, s, E, E_s, \nu, LT, GBC) \tag{6.4}$$

where

$\phi$ ≡ 'function of'

$\ell$ ≡ plate span

$t$ ≡ plate thickness

$b_s$ ≡ stiffener web width

$d_s$ ≡ stiffener depth

$s$ ≡ spacing of stiffeners

$E$ ≡ Young's modulus for the plate material

$E_s$ ≡ Young's modulus for the stiffener material

$\nu$ ≡ Poisson's ratio for the plate

LT ≡ Loading type, concentrated or uniformly distributed

GBC ≡ Geometric boundary conditions, e.g. simply supported and clamped.

In equation(6.4), LT and GBC are discrete conditions for which no numerical values are necessary and it is only to be remembered that these parameters should be varied. A dimensional analysis [38,39] of the remaining variables in equation (6.4) will result in the following seven dimensionless π-parameters:

$$\pi_1 = \ell/t,\ \pi_2 = \ell/b_s,\ \pi_3 = d_s/t\ ,$$
$$\pi_4 = s/\ell,\ \pi_5 = E/E_s,\ \pi_6 = \nu,\ \pi_7 = \eta \qquad (6.5)$$

In the above set of π-parameters, $\pi_1$ identifies if the plate is thin or thick; $\pi_2$ may be utilised to study the effect of the span size of a plate for a given web width of stiffeners; $\pi_3$ is a measure of eccentricity of stiffeners; $\pi_4$ is a parameter of great interest which is used to study the degree of closely-spacedness of stiffeners beyond which the orthotropic formulation will be unacceptable; the parameter $\pi_5$ is modified to a more meaningful rigidity ratio $D/D_s$ as shown later; the parameter $\pi_6$ is not likely to play any role in affecting smeared behaviour and hence has been kept constant at 0.3; and, $\pi_7$ is basically the set of dependent variables $\eta_w$, $\eta_{stp}$ and $\eta_{sts}$ whose numerical differences with unity will indicate the extents of departure from accuracy. It is known from structural mechanics that $E/(1-\nu^2)$ is a more meaningful quantity for plates rather than simply E because of its two-dimensional behaviour. Using this fact and the technique of compounding [38] in dimensional anlaysis, $\pi_5$ is transformed as follows:

$$\pi_5 \rightarrow \frac{E}{1-\nu^2} \cdot \frac{1}{E_s} \cdot \left(\frac{1}{\pi_3}\right)^3 \cdot \pi_4 \cdot \pi_2 = \frac{E}{1-\nu^2} \cdot \frac{1}{E_s} \cdot \frac{t^3}{d_s^3} \cdot \frac{s}{\ell} \cdot \frac{\ell}{b_s}$$

$$= \frac{Et^3}{12(1-\nu^2)} \cdot \frac{12s}{E_s b_s d_s^3}$$

$$= \frac{D}{D_S} \quad (6.5a)$$

where

$$D = \frac{Et^3}{12(1-\nu^2)} \equiv \text{plate rigidity} \quad (6.5b)$$

$$D_s = \frac{E_s b_s d_s^3}{12s} \equiv \text{stiffener rigidity} \quad (6.5c)$$

In the present parametric study, the stiffened plate system is assumed to be made up of one material, i.e. $E = E_s$, and further reflection on the parameter $D/D_s$ is called for. Rewriting $D/D_S$ under the preceding assumption ($E = E_s$), we have,

$$\frac{D}{D_S} = \frac{1}{1-\nu^2} \cdot \frac{\ell}{b_s} \cdot \left(\frac{t}{d_s}\right)^3 \cdot \frac{s}{\ell} = \frac{1}{1-\pi_6^2} \cdot \pi_2 \cdot \frac{1}{\pi_3^3} \cdot \pi_4 \quad (6.6)$$

Since $\pi_6$ (= 0.3) is kept constant, it is evident from equation (6.6) that $\pi_2$ will be determined if $\pi_5$ (= $D/D_s$), $\pi_3$ and $\pi_4$ are known. Hence in the case studies I - VII (Tables 2-8), the parameters that have been varied to study their effects on $\eta$ are as follows:

$$\pi_1 = \frac{\ell}{t}, \pi_3 = \frac{d_s}{t}, \pi_4 = \frac{s}{\ell}, \pi_5 = \frac{D}{D_s}, \text{ LT, GBC} \quad (6.7)$$

## 6.7 Results from the Geometrically Non-linear Plate Analysis

It is known that stiffened plates can take up substantially large loads for which the second order effects in deformation are quite important and yet the material may be stressed much below the yield point. A geometrically non-linear and materially elastic analysis for eccentrically stiffened plates under orthotropic assumptions using the Integral Equations Method was presented in [10]. A quantity $r_{pt}$ defined in [10] is redefined here:

$$r_{pt} = \frac{\text{Volume of deck plate per unit area}}{\text{Total volume of plate and stiffeners per unit area}} \quad (6.8)$$

In light of the discussion in section 6.6, the quantity $r_{pt}$ needs closer examination. Employing the notations described after equation (6.4) and following the definition of $r_{pt}$ in (6.8), we have

$$r_{pt} = \frac{\ell^2 t}{\ell^2 t + 2nb_s d_s \ell} = \frac{\ell/b_s}{\ell/b_s + 2(\frac{\ell}{s} - 1)\frac{d_s}{t}} \qquad (6.9)$$

where, n = number of stiffeners in any direction

$$= \frac{\ell}{s} - 1 \qquad (6.9a)$$

It is seen from relation (6.9) that $r_{pt}$ is a function of three non-dimensional quantities $\ell/b_s$, $s/\ell$ and $d_s/t$ which are all included in the list of $\pi$-parameters in (6.5). If $r_{pt}$ together with $s/\ell$ and $d_s/t$ are used as representative parameters, the quantity $\ell/b_s$ is automatically fixed. Further, if $s/\ell$, $d_s/t$ and $\ell/b_s$ are known, $D/D_s$, which is a relevant parameter affecting orthotropic behaviour, is determinable from equation (6.6). Hence it is sufficient to plot, in the thin plate range, representative values of deflection and plate as well as stiffener stresses against $r_{pt}$, $s/\ell$ and $d_s/t$, given the loading and geometric boundary conditions. In order to attempt a comparison with the results presented in [10], clamped plates under uniformly distributed loading is considered. In accordance with the conventions followed in [10], the representative central deflection and plate as well as stiffener stresses are normalized as follows:

$$\bar{w} = \frac{w_m}{\bar{h}} \equiv \text{Normalized deflection} \qquad (6.10)$$

where,

$$w_m = \text{central deflection} \qquad (6.10a)$$

$$\bar{h} = \frac{\text{Total volume of plate and stiffeners}}{\text{Area of the stiffened plate}}$$

$$= \frac{\ell^2 t + 2n\, b_s d_s \ell}{\ell^2} = t + 2\left(\frac{\ell}{s} - 1\right) \frac{b_s d_s}{\ell} \qquad (6.10b)$$

$$S_T = \sigma_{pm} \frac{(1-\nu^2)\, \ell^2}{E\bar{h}^2}$$

$$\equiv \text{Normalized plate stress} \qquad (6.11)$$

where,

$$\sigma_{pm} = \text{central plate-top stress} \qquad (6.11a)$$

$$S_B = \sigma_{sm} \frac{(1-\nu^2)\, \ell^2}{E\, \bar{h}^2}$$

$$\equiv \text{Normalized stiffener stress} \qquad (6.12)$$

where,

$$\sigma_{sm} = \text{central stiffener-bottom stress} \qquad (6.12a)$$

A normalized loading parameter $\bar{Q}$ is also defined following the authors in [10]:

$$\bar{Q} = \frac{q\ell^4}{\bar{D}\, \bar{h}} \qquad (6.13)$$

where,

$q$ = intensity of uniformly distributed loading (6.13a)

$$\bar{D} = \frac{E\bar{h}^3}{12(1-\nu^2)} \qquad (6.13b)$$

All quantities not defined in relations (6.10) - (6.13) have been defined previously in the present chapter.

It may be noted that a value of $r_{pt}$ equal to unity represents an unstiffened plate. For such a plate, results have been presented in Figs. 21–23 with comparisons from [10]. For stiffened plates, results have been presented for various values of $r_{pt}$ and $d_{s/t}$ in Figs. 24-35, maintaining $s/\ell$ constant at 0.067. The relevant comments on the presentation of these results together with discussions are included in the following chapter.

# CHAPTER 7

## DISCUSSION AND CONCLUDING REMARKS

### 7.1 On Formulations FEM(M1) and FEM(M2)

The main approaches in the finite element analysis of stiffened plates have been demonstrated with the help of the formulations mentioned in the subtitle above. In Example 1 (refer to Sec. 6.2) where the quarter plate mesh lay-out was identical for both FEM(M1) and FEM(M2), practically identical results were obtained. Both the formulations can therefore be said to possess the same degree of theoretical consistence. However, the advantages of FEM(M2), in which stiffeners can be placed within plate elements, becomes apparent when a large number of stiffeners is present. In such a case the use of FEM(M1), in which stiffeners can be placed only along plate nodal lines, would warrant a dense meshing and consequently, an expensive analysis. Further, awkward sizes of plate elements may be necessary when the stiffeners are unequally spaced rendering the output results less reliable. Hence, FEM(M2), in which mesh layout is not dictated by the number and configuration of stiffeners, is a superior formulation as compared to FEM(M1). Consequently, from Section 6.4 onwards in Chapter 6, FEM(M1) has been dropped in favour of FEM(M2) in the numerical studies.

Except in the case of the concentrated load case in Example 1 in Section 6.2, the agreement with the published theoretical and experimental results considered here is quite reasonable. It may be noted that varying support conditions and stiffener-configurations feature in the examples considered. With a fair degree of confidence, it can therefore be said that FEM(M2) is a consistent and reliable formulation.

## 7.2 On Formulation ORTHO and the Parametric Study (Tables 2-8)

The orthotropic theory is of historical importance in the analysis of stiffened plates. Since no single analytical approach of general applicability is available for the analysis of stiffened plates, there is no literature on theoretical evaluation of the accuracy of the orthotropic theory. In the present work, FEM(M2) has been already identified above as an acceptably accurate formulation. Taking FEM(M2) as a standard, a comparison has been made with the orthotropic formulation ORTHO by varying the parameters given in (6.5) of Section 6.6. Such a comparison is rational since the mechanics underlying the formulations FEM(M2) and ORTHO are similar except for the fact that in the latter the stiffeners have been smeared out and the effect of this smearing on the accuracy of deflections and stresses can be evaluated. Fig. 20 is a typical illustration in which it is shown how the deviation in deflection profiles obtained from the two formulations narrows down with increasing number of stiffeners. The observations from the limited parametric study enumerated in Case Studies I - VII (Tables 2-8) are enlisted below:

(i) From Case Study I, it is seen that maximum deflection and plate stresses are estimated within 10% on the safer side for $s/\ell \leq 0.111$ (i.e. 8+8 stiffeners) whereas maximum stiffener stresses are obtained within 20% only for $s/\ell < 0.077$ (i.e. 12+12 stiffeners).

(ii) For the case of a concentrated load as in Case Study II, the smeared-plate assumption yields increasingly erroneous results as indicated by an increasing $\eta_{stp}$ with decreasing plate rigidity (i.e. falling $D/D_s$). Moreover, stiffener stresses are largely overestimated even for a low value of $s/\ell$ as 0.067.

(iii) In Case Study I, all sides were clamped, while in Case Study III, all sides are simply supported. Comparison of the two cases show that the observations made in (i) above for the clamped case are also valid for the simply supported case.

(iv) In order to study the dependence of $\eta$ on span size (in other words, magnitude of stiffener spacing), a larger plate of span 8 m was taken in Case Study IV.

Furthermore, the plate aspect ratio $\ell/t$ was raised to 400. It is observed that for $s/\ell$ ratios of 0.111 and 0.067 and uniformly distributed loading, the $\eta$ parameters have shown no significant changes as compared to the corresponding values in the previous case studies.

(v) The objective of Case Study IV has been maintained in Case Study V. An even larger plate of sides 16 m was considered for analysis. The values of $\eta$ in this case are seen to be about the same as compared to those in Case Study IV for an $s/\ell$ ratio of 0.067 and plate to stiffener rigidity ratio of 0.117.

(vi) On the basis of Case Studies I-V, it is tentatively accepted that roughly for $s/\ell \leq 0.067$, the maximum deflections and plate stresses may be expected to be within 6-10% and the maximum stiffener stresses within 15-20%, all on the safer side, from an orthotropic theory. It is further noted, on the basis of observations in (ii) above, that an orthotropic theory should be avoided for the case of concentrated loadings.

(vii) Case Study VI was undertaken in order to study the effects of the parameters $D/D_s$ and $d_s/t$ on $\eta$. For three values of $D/D_s$ viz. 0.117, 0.469 and 0.938, $d_s/t$ was varied in the range 1.984 to 5.38. In accordance with observations in (vi) above, $s/\ell$ was maintained constant at 0.067. It is observed that for the stated ranges of variation of $D/D_s$ and $d_s/t$, $\eta_w$, $\eta_{stp}$ and $\eta_{sts}$ have remained remarkably consistent throughout.

(viii) In Case Study VII, a rectangular plate stiffened in the transverse direction only with a longitudinal $s/\ell$ ratio of 0.067 was analyzed under a uniformly distributed loading and simply supported edge conditions. The $\eta$ parameters are seen to be in quite the same range as the pertinent values for the orthogonally stiffened square plates analysed previously.

The quantitative observations made above with regard to the Case Studies I-VII seem to corroborate the prevalent qualitative opinions on the orthotropic theory. Huffington [40] stated that the orthotropic theory is applicable provided that the ratios of stiffener spacing to plate boundary dimensions are small enough. Hoppman and Huffington [41] compared their theoretical and experimental results on deflections and strains which they found in close agreement considering an 11in. x 11in. plate stiffened in one direction with 15 stiffeners. The theoretical calculations were based on an orthotropic theory. In the present investigation, deflections and plate stresses have been found to be within 6-10% for $s/\ell$ = .067 i.e., for 14 stiffeners in any direction. This is an interesting correlation with the number of stiffeners, viz., 15 chosen by Hoppman and Huffington [41] probably on the basis of experimental observations. Further, the present limited parameteric study indicates that the actual magnitude of stiffener spacing is inconsequential and the ratio $s/\ell$ (or, the number of stiffeners) governs. Clarkson [3] observed the inaccuracy of the orthotropic theory for the case of concentrated loadings. Case Study II specifically indicates the large overestimation of stiffener stresses that may result from the application of the orthotropic theory to stiffened plates with concentrated loads. Huffington [40] 'tacitly assumed' that his orthotropic analysis was not affected by the plate boundary conditions; in the present numerical study, no significant changes in the values of $\eta$ were found for all-sides-clamped and all-sides- simply-supported conditions. Troitsky [12] remarked that rigorous analysis procedures yield somewhat lower values of stresses as compared to those obtained from Huber's orthotropic theory. The trends in the Case Studies I-VII confirm this statement as in nearly all cases the $\eta$ parameters are greater than unity.

## 7.3 On Results from the Non-linear Orthotropic Analysis via Program NLORTHO

Results for non-dimensional deflection ($\bar{w}$), plate top-stress ($S_T$) and plate bottom-stress ($S_B$) for a clamped unstiffened thin plate ($r_{pt} = 1$) considering geometric non-linear behaviour have been presented in Figs. 21-23 along with values given by Srinivasan and Ramchandran [10]. The latter authors followed an Integral Equations approach. Excellent agreement for $\bar{w}$ and $S_B$, and fairly close agreement for $S_T$ is observed.

The authors Srinivasan and Ramchandran [10] presented their results on stiffened plates for different values of $r_{pt}$. However, $r_{pt}$ by itself is unlikely to be a unique parameter as revealed in equation (6.9) where $r_{pt}$ is shown to be a function of three relevant non-dimensional parameters $\ell/b_s$, $s/\ell$ and $d_s/t$. In fact, for a given value of $r_{pt}$ different combinations of $\ell/b_s$, $s/\ell$ and $d_s/t$ can be chosen. Following the rationale presented in Sec. 6.7, results have been presented in Figs. 24-35 for different values of $r_{pt}$ and $d_s/t$, while maintaining $s/\ell$ constant at 0.067 on account of the dependability of results for this value of $s/\ell$. It is observed that for a given value of $r_{pt}$, the eccentricity parameter $d_s/t$ significantly controls the representative stresses and deflections. With increasing values of $d_s/t$ (for a fixed $r_{pt}$), $\bar{w}$ is increasingly lowered, which is an expected outcome (orthotropic bending rigidity is a cubic function of $d_s$ for a given plate thickness t). A clear trend persists in the $S_T - \bar{Q}$ curves for a given $r_{pt}$ and in general, values of $S_T$ are increased as the parameter $d_s/t$ increases. The same is, however, not seen to be true for $S_B$; particularly, at low $r_{pt}$ values, e.g. 0.67 and 0.5, the differences in the corresponding $S_B-\bar{Q}$ curves (Figs. 32 and 35) appear to narrow down. For a given value of $r_{pt}$, a more efficient design is likely to result from a higher value of $d_s/t$. The curves presented by Srinivasan and Ramchandran [10] have also been reproduced for every value of $r_{pt}$ considered by them and the lack of uniqueness of these curves is apparent. From the $\bar{w}-\bar{Q}$ curves, it is noted that at higher values of the parameter $d_s/t$ (e.g. for $d_s/t = 6.54$ in Fig. 30), the non-linear behaviour is less prominent. This is probably because at higher $d_s/t$ values the stiffened plate system behaves predominantly as a grid structure.

## 7.4 Epilogue

On the basis of the data presented and the preceding discussions, it is believed that the following goals have been fulfilled:

* The application of the computationally advantageous Mindlin's Shear Distortion Theory to the two main approaches in finite element analysis of stiffened plates has been shown via formulations FEM(M1) and FEM(M2).
* An orthotropic formulation ORTHO, also based on Mindlin's theory, has been presented and a quantitative idea on the parameters controlling its applicability has been formed.

* The orthotropic formulation has been extended to the case of geometrically non-linear behaviour since it was recognised that this formulation may require much less computing time than the more rigorous discrete plate-beam formulations, and hence may be preferred for analysis under less demanding conditions.
* Software has been developed which will be of considerable aid in future developmental works on general material and geometric non-linear behaviour under static and dynamic conditions.

References

[1] M.P. Rossow and A.K. Ibrahimkhail, Constrain method analysis of stiffened plates. Comput. Struct., 8, 51-60 (1978).

[2] C.S. Smith, Elastic analysis of stiffened plating under lateral loading. Trans. RINA, 108, 113-130 (1966).

[3] J. Clarkson, Tests of flat pated grillages under concentrated loads. Trans. INA, 101, 129-140 (1959).

[4] J. Clarkson, The behaviour of deck stiffening under concentrated loads; Trans. RINA, 104, 57-65 (1962).

[5] D.N. de G. Allen and R.T. Severn, Composite action of beams and slabs under transverse loading. Struct. Engr., 39, 235-239 (1961).

[6] S.P. Chang, Analysis of eccentrically stiffened plates. Ph.D. Thesis presented to the University of Missouri, Columbia, MO (1973).

[7] M. Mukhopadhyay, Stiffened plate plane stress elements for the analysis of ships' structures. Comput. Struct., 13, 563-573 (1981).

[8] M. Mukhopadhyay and S.K. Satsangi, Isoparametric stiffened plate bending element for the analysis of ships' structures. Trans. RINA, 126, 141-151 (1984).

[9] M. Mukhopadhyay, Analysis of plates using isoparametric quadratic element - shear, reaction, patch loading and some convergence studies. Comput. Struct, 17(4), 587-597 (1983).

[10] R.S. Srinivasan and S.B. Ramchandran, Linear and non-linear analysis of stiffened plates, Int. J. Solids Struct., 13, 897-912 (1977).

[11] R.S. Srinivasan and V. Thiruvenkatachari, Static and dynamic analysis of stiffened plates. Comput. Struct., 21(3), 395-403 (1985).

[12] M.S. Troitsky, Stiffened Plates, Elsevier, Amsterdam (1971).

[13] A.W. Wegmuller, Finite element analysis of elastic-plastic plates and eccentrically stiffened plates. Ph.D. Thesis presented to Lehigh University (1972).

[14] R.P. McBean, Analysis of stiffened plates by the finite element method. Ph.D. Thesis presented to Standford University (1968).

[15] M.D. Olson and G.M. Lindberg, Free vibrations and random response of an integrally-stiffened panel. Aeronautical Report LR-544, National Research Council of Canada, Ottawa.(Oct. 1970).

[16] G.M. Lindberg, Accurate finite element modelling of flat and curved stiffened panels. AGARD Conf. Proc. No. 113 (Sept. 1972).

[17] S.R. Karve, Analysis of stiffened plate systems. Ph.D. Thesis presented to the Indian Institute of Technology, Bombay, India (1974).

[18] L.M. Kutt, Elasto-plastic buckling and post-buckling behaviour of stiffened plates and box-girders. Ph.D. Dissertation, Columbia University (1982).

[19] R.D. Cook, Concepts and Applications of Finite Element Analysis. 2nd Ed., John Wiley & Sons, pp. 158-165 (1981).

[20] J.F. Abel and M.S. Shephard, An algorithm for multipoint constraints in finite element analysis. Int. J. Numer Methods Eng., 14(3), 464-467 (1979).

[21] O.C. Zienkiewicz, The Finite Element Method, Tata McGraw-Hill, New Delhi, pp. 77-92 (1979).

[22] B.A. Szabo and T. Kassos, Linear equality constraints in finite element approximation. Int. J. Numer Methods Eng., 9, 563-580 (1975).

[23] J.R. O'Leary and I. Harari, Finite element analysis of stiffened plates. Comput. Struct., 21(5), 973-985 (1985).

[24] W.C. Gustafson and R.N. Wright, Analysis of skewed composite girder bridges. J. Struct. Div A.S.C.E, 94(ST4), 919-941, (Apr. 1968).

[25] R.D. Mindlin, Influence of rotatory inertia and shear on flexural motion of isotropic elastic plates. J. Appl. Mech. Trans ASME, 73, 13-38 (1951).

[26] T.J.R. Hughes et al, A simple and efficient finite element for plate bending. Int. J. Numer Methods Eng., 11, 1529-1543 (1977).

[27] E. Hinton et al, A simple finite element solution for plates of homogeneous, sandwich and cellular construction. Proc. Instn. Civ. Engrs., Part 2, 59, 43-65 (1975).

[28] E.D.L. Pugh et al, A study of quadratic plate bending elements with reduced integration. Int. J. Numer Methods Eng., 12, 1059-79 (1978).

[29] T.J.R. Huges et al, Reduced and Selective integration techniques in the finite element analysis of plates. Nuclear Engrg. Design, 46, 203-222 (1978).

[30] T.J.R. Huges and M. Cohen, The "heterosis" finite element for plate bending, Comput. Struct, 9, 445-450 (1978).

[31] E. Hinton and N. Bicanic, A comparison of Lagrangian and Serendipity Mindlin plate elements for free vibration, Comput. Struct, 10(3), 483-493 (1979).

[32] D.R.J. Owen and E. Hinton, Finite Elements in Plasticity: Theory and Practice, Pinneridge Press Ltd., Swansea, U.K. (1980).

[33] G.R. Cowper et al, A high precision triangular plate bending element, Aeronautical Report LR-514, National Research Council of Canada, Ottawa (Dec. 1968).

[34] K.J. Bathe and E.L. Wilson, Numerical Methods in Finite Element Analysis. Prentice-Hall, Englewood Cliffs, New Jersey (1976).

[35] J. Barlow, Optimal stress locations in finite element models. Int. J. Numer Methods Eng., 10, 243-251(1976).

[36] Y.C. Fung, Foundations of Solid Mechanics. Prentice-Hall, Englewood Cliffs, New Jersey, pp. 434-470 (1965).

[37] A. Pica et al, Finite element analysis of geometrically non-linear plate behaviour using a Mindlin formulation. Comput. Struct., 11, 203-215 (1980).

[38] A. Deb, M.K. Deb and J.J. Sharp, A comparative study of the techniques in dimensional analysis. Applied Mathematics Notes, 10(4), 17-34 (Dec. 1985).

[39] M.K. Deb and A. Deb, The matrix method: a powerful technique in dimensional analysis. J. Franklin Inst., 321(4), 233-240, (1986).

[40] N.J. Huffington, Jr., Theoretical determination of rigidity properties of orthogonally stiffened plates, J. Appl. Mech Trans ASME, 15-20 (Mar. 1956).

[41] W.H. Hoppmann (II) and N.J. Huffington, Jr, A study of orthogonally stiffened plates, J. Appl. Mech. Trans. ASME, 343-350 (Sept. 1956).

[42] A.P. Boresi and O.M. Sidebottom, Advanced Mechanics of Materials. 4th Ed., John Wiley & Sons (1985).

Table 1 Comparison of Theoretical and Experimental Results from Ref. [5] with Formulation FEM(M2)

| | | Deflection 1"(2.54 cm) from beam-centre (in) | | | Stress($lb/in^2$) at bottom of beam-centre | | |
|---|---|---|---|---|---|---|---|
| d (in) | load on each beam (lb) | Theoretical | Experimental | FEM(M2) | Theoretical | Photoelastic measurement | FEM(M2) |
| 0.752 (1.91 cm) | 39.7 (177 N) | .016 (.406 mm) | .0178 (.452 mm) | .0152 (.386 mm) | 1929 (13.3 MPa) | 2224 (15.3 MPa) | 1879 (13.0 MPa) |
| 0.600 (1.52 cm) | 36.2 (161 N) | .0279 (.709 mm) | .0255 (.648 mm) | .0263 (.668 mm) | 2667 (18.4 MPa) | 2909 (20.1 MPa) | 2648 (18.3 MPa) |
| 0.450 (1.14 cm) | 13.8 (61.4 N) | .0233 (.592 mm) | .0206 (.523 mm) | .0223 (.566 mm) | 1665 (11.5 MPa) | 1882 (13.0 MPa) | 1667 (11.5 MPa) |

**Case Study I:** 2m x 2m plate, 0.02 m thick;
0.01 m x 0.1 m stiffeners
LT: udl of 60,000 N/m$^2$
GBC: fixed on all sides
$\ell/t$: 100

Table 2

| $s/\ell$ | $D/D_S$ | $d_s/t$ | FEM(M2) | | | ORTHO | | | $\eta_w$ | $\eta_{stp}$ | $\eta_{sts}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-8}$ | $\sigma_{sm} \times 10^{-8}$ | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-8}$ | $\sigma_{sm} \times 10^{-8}$ | | | |
| .067 | .117 | 5 | .291 | .121 | -0.475 | .307 | .125 | -0.547 | 1.05 | 1.04 | 1.15 |
| .077 | .135 | 5 | .326 | .129 | -0.525 | .347 | .133 | -0.629 | 1.06 | 1.03 | 1.20 |
| .091 | .160 | 5 | .385 | .143 | -0.594 | .403 | .144 | -0.743 | 1.05 | 1.00 | 1.25 |
| .111 | .195 | 5 | .463 | .160 | -0.715 | .485 | .159 | -0.912 | 1.05 | 0.99 | 1.28 |
| .143 | .251 | 5 | .560 | .182 | -0,903 | .617 | .183 | -1.19 | 1.10 | 1.01 | 1.31 |
| .200 | .352 | 5 | .812 | .261 | -1.16 | .867 | .229 | -1.71 | 1.07 | 0.88 | 1.48 |

**Case Study II:** 2m x 2m plate, 0.02 m thick;
0.01 m x 0.1 m stiffeners
LT: Central point load of 24,000N
GBC: fixed on all sides
$\ell/t$: 100

Table 3

| $s/\ell$ | $D/D_S$ | $d_s/t$ | FEM (M2) | | | ORTHO | | | $\eta_w$ | $\eta_{stp}$ | $\eta_{sts}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-8}$ | $\sigma_{sm} \times 10^{-8}$ | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-8}$ | $\sigma_{sm} \times 10^{-8}$ | | | |
| .067 | .117 | 5 | .553 | -.400 | 1.06 | .584 | -.467 | 1.72 | 1.05 | 1.17 | 1.62 |
| .077 | .135 | 5 | .622 | -.434 | 1.15 | .658 | -.497 | 1.97 | 1.06 | 1.15 | 1.71 |
| .091 | .160 | 5 | .746 | -.505 | 1.29 | .760 | -.538 | 2.32 | 1.02 | 1.07 | 1.80 |

**Case Study III:** 2m x 2m plate, 0.02 m thick; 0.01 m x 0.1 m stiffeners

LT: udl of 60,000 N/m$^2$
GBC: All sides simply supported
$\ell/t$: 100

Table 4

| $s/\ell$ | $D/D_S$ | $d_s/t$ | FEM(M2) | | | ORTHO | | | $\eta_w$ | $\eta_{stp}$ | $\eta_{sts}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-8}$ | $\sigma_{sm} \times 10^{-8}$ | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-8}$ | $\sigma_{sm} \times 10^{-8}$ | | | |
| .067 | .117 | 5 | 1.29 | −.159 | .608 | 1.36 | −.163 | .711 | 1.06 | 1.02 | 1.17 |
| .077 | .135 | 5 | 1.45 | −.170 | .689 | 1.55 | −.175 | .818 | 1.07 | 1.03 | 1.19 |
| .091 | .160 | 5 | 1.68 | −.185 | .795 | 1.81 | −.192 | .967 | 1.08 | 1.04 | 1.22 |
| .111 | .195 | 5 | 2.01 | −.207 | .952 | 2.20 | −.217 | 1.18 | 1.09 | 1.05 | 1.24 |
| .143 | .251 | 5 | 2.48 | −.239 | 1.19 | 2.81 | −.257 | 1.54 | 1.13 | 1.08 | 1.29 |
| .200 | .352 | 5 | 3.37 | −.350 | 1.57 | 3.93 | −.329 | 2.17 | 1.17 | 0.94 | 1.38 |

**Case Study IV:** 8m x 8m plate, 0.02 m thick;
0.02 m x 0.126 m stiffeners

LT: udl of 60 $N/m^2$

GBC: All sides simply supported

$\ell/t$: 400

Table 5

| $s/\ell$ | $D/D_S$ | $d_s/t$ | FEM(M2) | | | ORTHO | | | $\eta_w$ | $\eta_{stp}$ | $\eta_{sts}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-6}$ | $\sigma_{sm} \times 10^{-6}$ | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-6}$ | $\sigma_{sm} \times 10^{-6}$ | | | |
| .067 | .117 | 6.3 | .330 | −.235 | 1.25 | .350 | −.243 | 1.48 | 1.06 | 1.03 | 1.18 |
| .111 | .195 | 6.3 | .519 | −.312 | 1.98 | .575 | −.335 | 2.47 | 1.11 | 1.07 | 1.25 |

**Case Study V:** 16m x 16m plate, 0.04 m thick; 0.03 m x 0.2774 m stiffeners
LT: udl of 10 $N/m^2$
GBC: All sides simply supported
$\ell/t$: 400

Table 6

| $s/\ell$ | $D/D_S$ | $d_s/t$ | FEM(M2) | | | ORTHO | | | $\eta_w$ | $\eta_{stp}$ | $\eta_{sts}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $w_m x\ 10^3$ | $\sigma_{pm} x\ 10^{-5}$ | $\sigma_{sm} x\ 10^{-5}$ | $w_m x\ 10^3$ | $\sigma_{pm} x\ 10^{-5}$ | $\sigma_{sm} x\ 10^{-5}$ | | | |
| .067 | .117 | 6.94 | .110 | -.379 | 2.30 | .118 | -.394 | 2.47 | 1.06 | 1.04 | 1.19 |

**Case Study VI:** 2m x 2m plate, 0.02 m thick;
LT: udl of 60,000 $N/m^2$
GBC: All sides simply supported
$\ell/t$: 100

Table 7

| $s/\ell$ | $D/D_S$ | $d_s/t$ | FEM(M2) | | | ORTHO | | | $\eta_w$ | $\eta_{stp}$ | $\eta_{sts}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-8}$ | $\sigma_{sm} \times 10^{-8}$ | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-8}$ | $\sigma_{sm} \times 10^{-8}$ | | | |
| .067 | .117 | 3.97 | 1.28 | -.174 | 0.470 | 1.35 | -.177 | 0.550 | 1.06 | 1.02 | 1.17 |
| .067 | .117 | 4.37 | 1.28 | -.167 | 0.523 | 1.36 | -.171 | 0.612 | 1.06 | 1.02 | 1.17 |
| .067 | .117 | 5 | 1.29 | -.159 | 0.608 | 1.36 | -.163 | 0.711 | 1.06 | 1.02 | 1.17 |
| .067 | .117 | 5.38 | 1.29 | -.155 | 0.659 | 1.37 | -.159 | 0.722 | 1.06 | 1.02 | 1.17 |
| .067 | .469 | 2.5 | 3.62 | -.363 | 0.928 | 3.82 | -.372 | 1.08 | 1.06 | 1.02 | 1.16 |
| .067 | .469 | 2.75 | 3.70 | -.361 | 1.05 | 3.90 | -.369 | 1.22 | 1.05 | 1.02 | 1.16 |
| .067 | .469 | 3.15 | 3.80 | -.355 | 1.23 | 4.00 | -.364 | 1.42 | 1.05 | 1.02 | 1.16 |
| .067 | .469 | 3.39 | 3.86 | -.353 | 1.34 | 4.07 | -.362 | 1.55 | 1.05 | 1.02 | 1.16 |
| .067 | .938 | 1.98 | 5.68 | -.513 | 1.22 | 5.97 | -.525 | 1.42 | 1.05 | 1.02 | 1.16 |
| .067 | .938 | 2.50 | 6.08 | -.518 | 1.63 | 6.40 | -.531 | 1.89 | 1.05 | 1.03 | 1.16 |
| .067 | .938 | 2.69 | 6.18 | -.517 | 1.77 | 6.49 | -.531 | 2.05 | 1.05 | 1.03 | 1.16 |

Case Study VII: 3.6m x 16m plate, .02m thick; transversely stiffened by 14 equally spaced stiffeners (i.e. $s/\ell$ = .067, where $\ell$ = longitudinal span = 16m)

LT: ud$\ell$ of 1000 $N/m^2$

GBC: All sides simple supported

$\ell/t$: 800

Table 8

| $D/D_s$ | $d_s/t$ | FEM(M2) | | | ORTHO | | | $\eta_w$ | $\eta_{stp}$ | $\eta_{sts}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-7}$ | $\sigma_{sm} \times 10^{-7}$ | $w_m \times 10^3$ | $\sigma_{pm} \times 10^{-7}$ | $\sigma_{sm} \times 10^{-7}$ | | | |
| .117 | 6.93 | 0.420 | -.153 | 0.849 | 0.437 | -.155 | 1.01 | 1.04 | 1.02 | 1.19 |
| .117 | 7.94 | 0.419 | -.144 | 0.979 | 0.437 | -.146 | 1.17 | 1.04 | 1.02 | 1.19 |
| .117 | 8.74 | 0.418 | -.138 | 1.08 | 0.436 | -.140 | 1.29 | 1.04 | 1.02 | 1.19 |
| .938 | 3.97 | 2.25 | -.501 | 2.91 | 2.37 | -.515 | 3.39 | 1.05 | 1.03 | 1.16 |
| .938 | 4.36 | 2.29 | -.500 | 3.26 | 2.41 | -.512 | 3.78 | 1.05 | 1.02 | 1.16 |
| .938 | 5.00 | 2.34 | -.494 | 3.79 | 2.48 | -.511 | 4.41 | 1.06 | 1.03 | 1.17 |

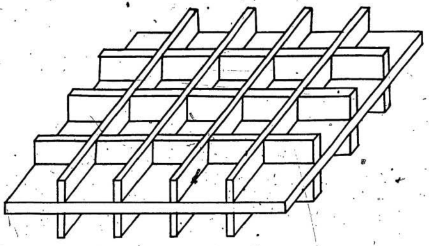


Fig.1(a). A Symmetrically Stiffened Plate

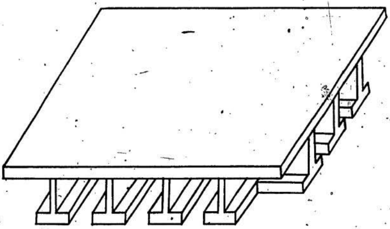
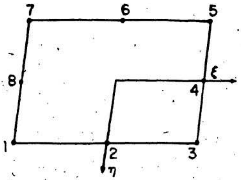


Fig.1(b). An Eccentrically Stiffened Plate



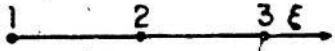

$$N_i = \frac{1}{4}(1+\xi\xi_i)(1+\eta\eta_i)(\xi\xi_i+\eta\eta_i-1)$$

$$, \quad i = 1,3,5,7$$

$$= \frac{\xi_i^2}{2}(1+\xi\xi_i)(1-\eta^2) + \frac{\eta_i^2}{2}(1+\eta\eta_i)(1-\xi^2),$$

$$i = 2,4,6,8$$

Fig.2(a). An 8-Node Serendipity Plate Bending Element




$$N_{\xi i} = \frac{\xi\xi_i}{2}(1+\xi\xi_i),$$

$$i = 1,3$$

$$= (1-\xi^2), \quad i = 2$$

Fig.2(b). An X-Stiffener Element


$$N_{\eta i} = \frac{\eta\eta_i}{2}(1+\eta\eta_i),$$

$$i = 1,3$$

$$= (1-\eta^2), \quad i = 2$$

Fig.2(c) A Y-Stiffener Element

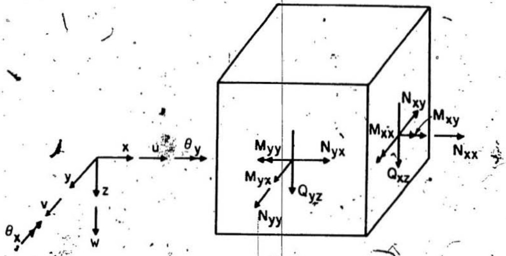



Fig.3. A Unit Cube Of The Continuum Depicting Generalised Degrees Of Freedom And Stress-Resultants

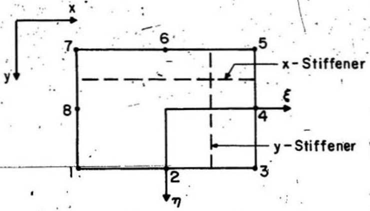



Fig.4. An Orthogonally Stiffened Isoparametric Plate Bending Element

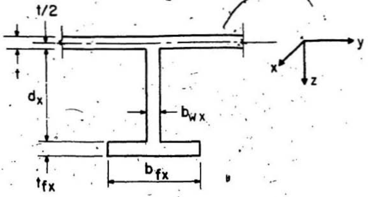



Fig.5. Cross-section Of A Typical X-Stiffener

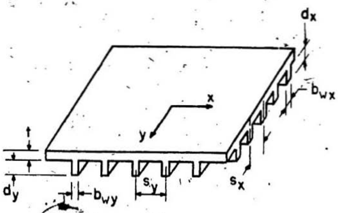



Fig.6. Geometry Of An Orthotropic Plate

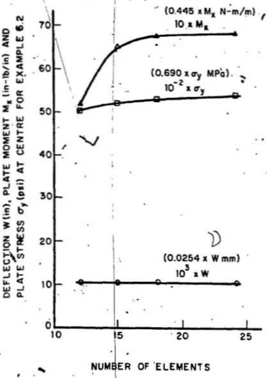



Fig.7. Convergence Study

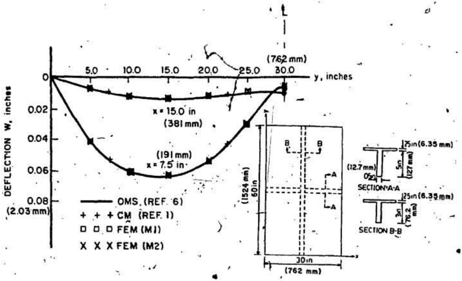



Fig.8. Deflection Patterns Under Uniformly Distributed Load

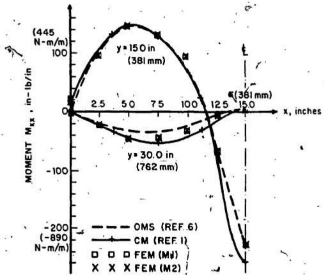



Fig.9. Plate Moment Variation Under Uniformly Distributed Load

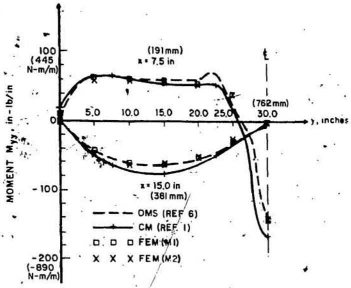



Fig.10. Plate Moment Variation Under Uniformly Distributed Load

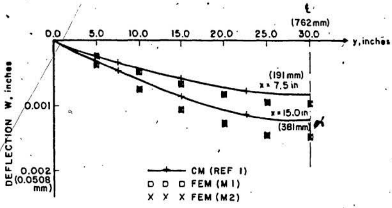



Fig.11. Deflection Patterns Under Concentrated Load

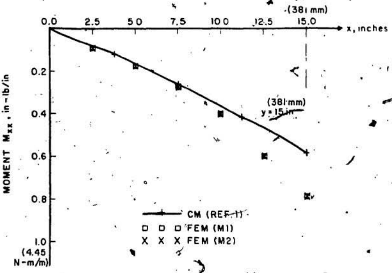



Fig.12. Plate Moment Variation Under Concentrated Load

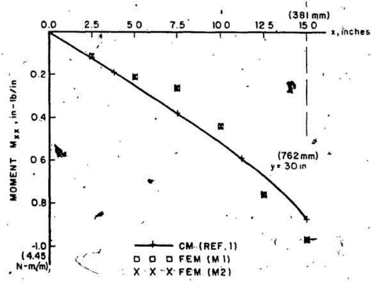



Fig.13. Plate Moment Variation Under Concentrated Load

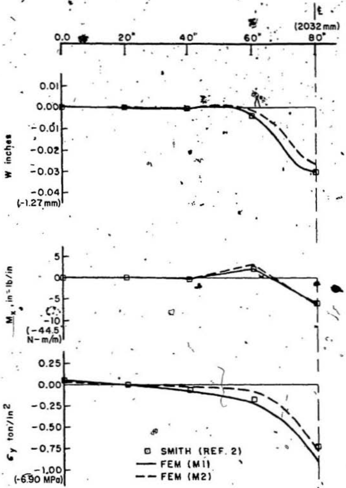



Fig.14. Variation Of Deflection, Plate Moment And Plate Stress Along The Centre Line In The Transverse Direction

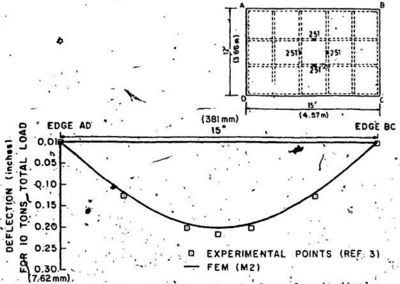



Fig.15. Deflection Patterns For An Outer Longitudinal Beam

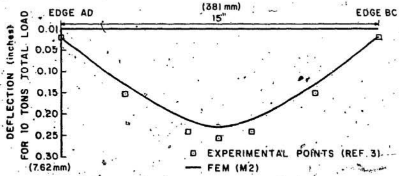



Fig.16. Deflection Patterns For An Inner Longitudinal Beam

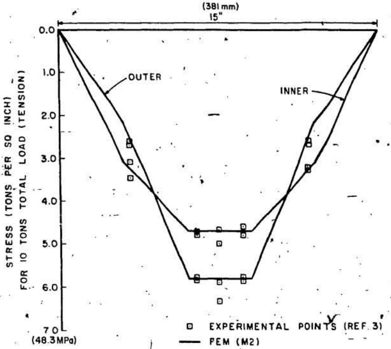



Fig.17. Stresses In Longitudinal Beams

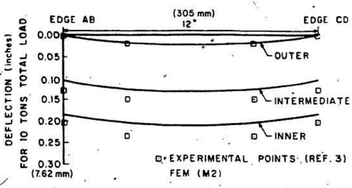



Fig.18. Deflection Patterns For Transverse Beams

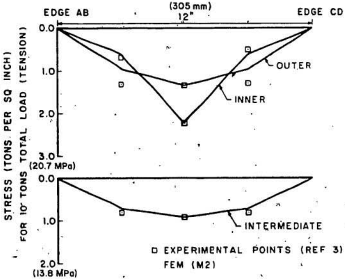



Fig.19. Stresses In Transverse Beams

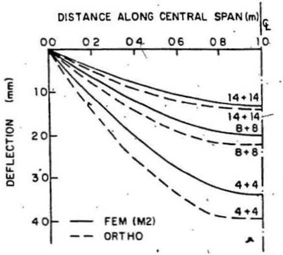



Fig.20. Progressive Decrease In Deviation In Deflection Profiles With Increasing Number Of Stiffeners (2m*2m*0.02m S-S-S-S Plate; 0.01m*0.1m Stiffeners; UDL = 60,000 N/(m*m))

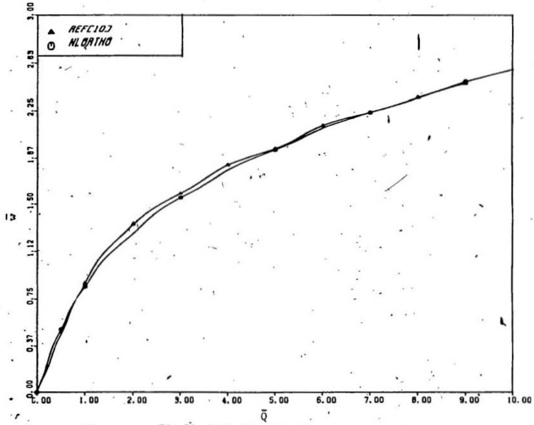



Fig.21. Load vs Deflection ($r_{pt}$=1.0, s/1=0.067)

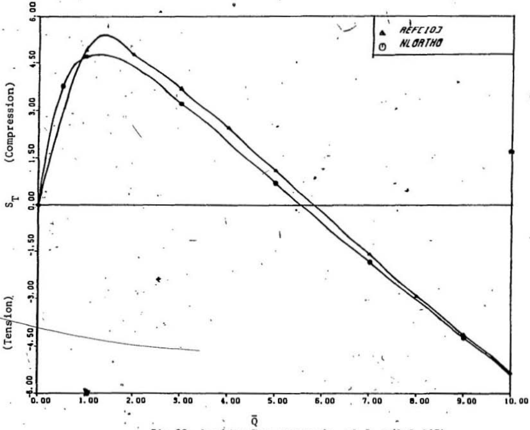



Fig.22. Load vs Top-stress ($r_{pt}$=1.0, s/1=0.067)

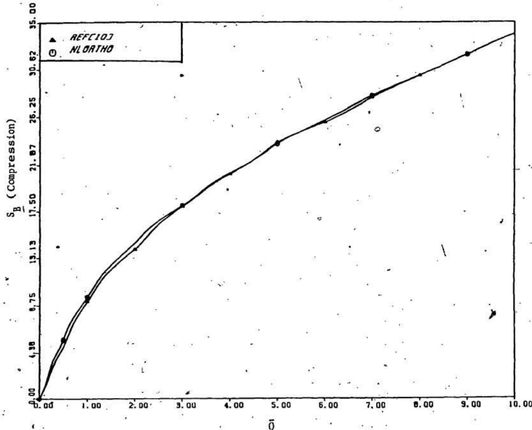



Fig.23. Load vs Bottom-stress ($r_{pt}$=1.0, s/l=0.067)

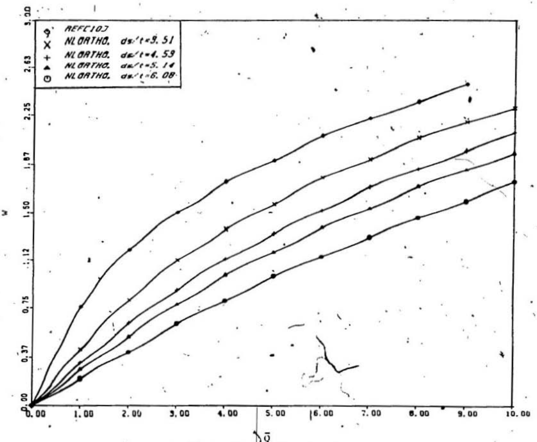



Fig.24. Load vs Deflection ($r_{pt}$=0.9, s/1=0.067).

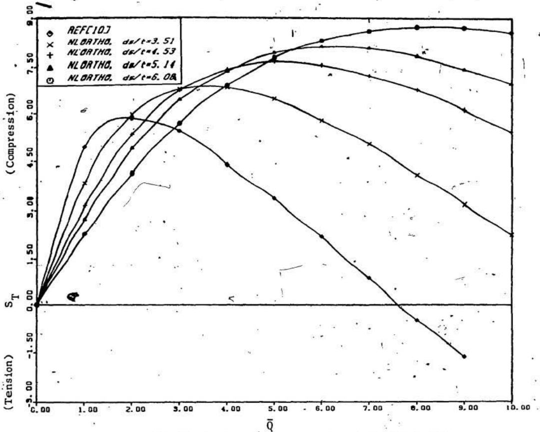



Fig.25. Load vs Top-stress ($r_{pt}$=0.9, ṡ/1=0.067)

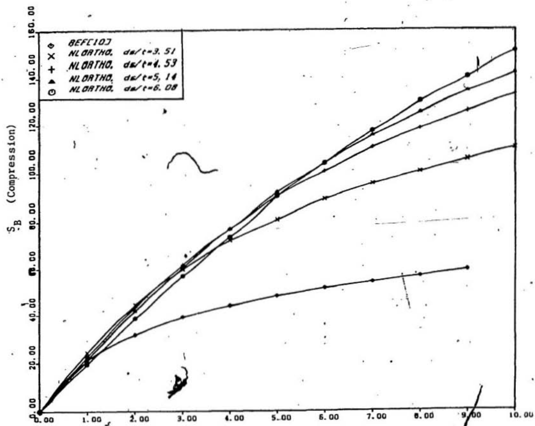



Fig.26. Load vs Bottom-stress ($r_{pt}$=0.9, s/1=0.067)

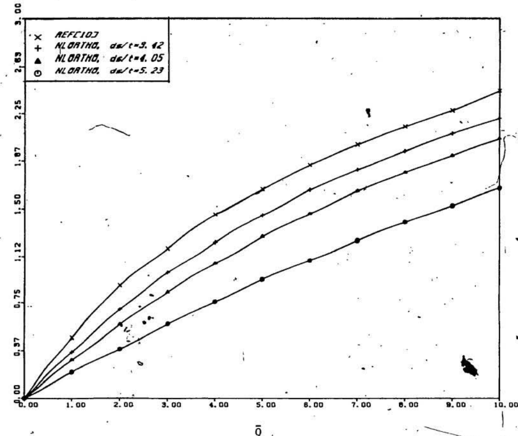



Fig.27. Load vs Deflection ($r_{pt}$=0.8, s/l=0.067)

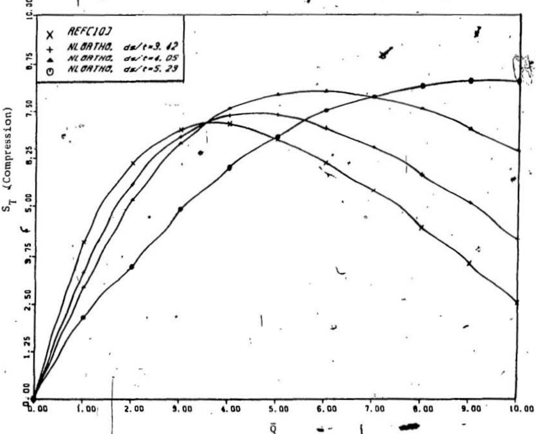



Fig.28. Load vs Top-stress ($r_{pt}$=0.67, s/l=0.067)

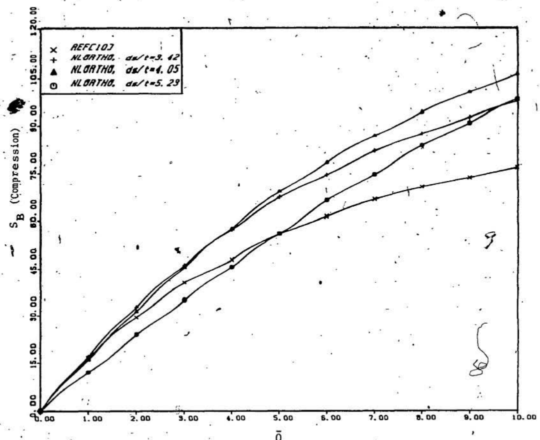



Fig.29. Load vs Bottom-stress ($r_{pt}$=0.8, s/1=0.067)

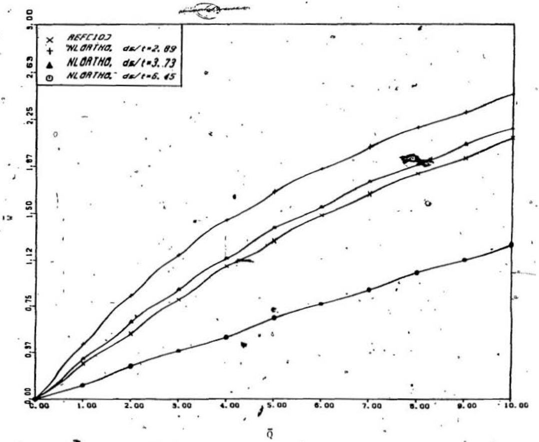



Fig. 30. Load vs Deflection ($r_{pt}$=0.67, s/1=0.067)

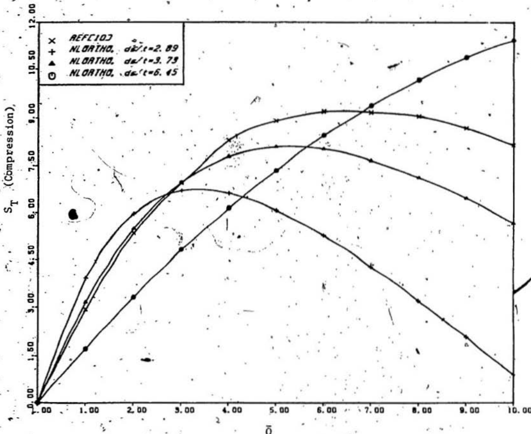



Fig. 31. Load vs Top-stress ($r_{pt}$=0.67, s/l=0.067)

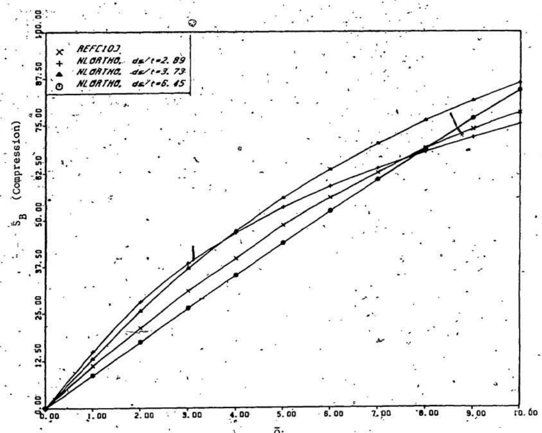



Fig.32. Load vs Bottom-stress ($r_{pt}$=0.67, s/1=0.067)

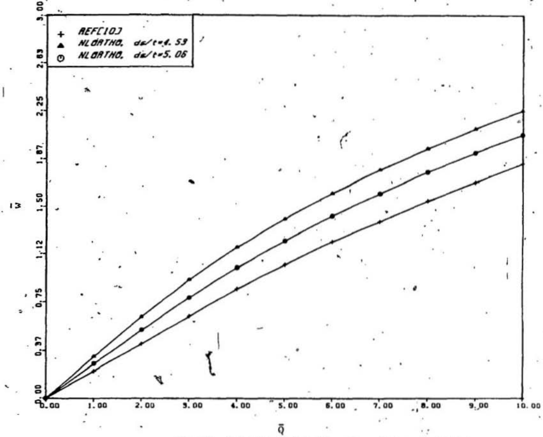



Fig.33. Load vs Deflection ($r_{pt}$=0.5, s/1=0.067)

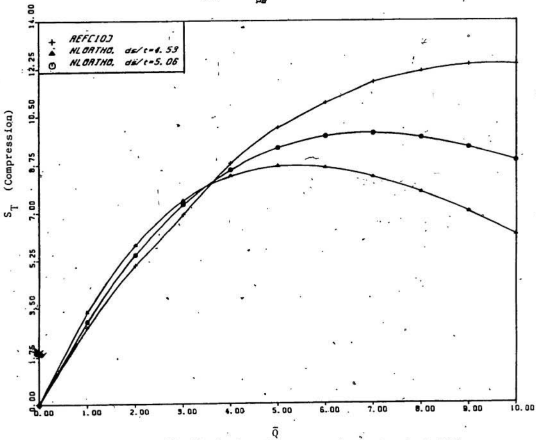



Fig.34. Load vs Top-stress ($r_{pt}$=0.5, s/1=0.067)

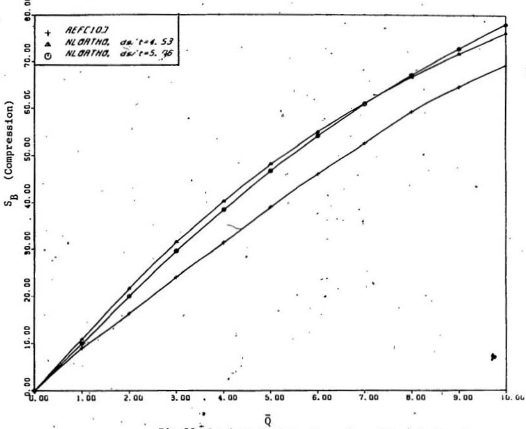



Fig.35. Load vs Bottom-stress ($r_{pt}$=0.5, s/l=0.067)

# Appendix 1

## Listing of the Stiffened Plate Analysis Program SPAP

```
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C       THIS IS THE LISTING OF PROGRAM SPAP (STIFFENED PLATE ANALYSIS
C       PROGRAM). SPAP IS SELF-CONTAINED.
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        DIMENSION X(100),Y(100),NODES(100,8),NODESX1(100,3),NODESY1(100,
     1 3),NODESX2(100,2),NODESY2(100,2),ID(5,100),LM(100,40),LMX(100,15
     2 ),OSTF(100000),LMY(100,15),QZ(100),ESTF(40,40),PG(100),ESTFX1(15
     3 ,15),ESTFY1(15,15),ESTFX2(40,40),ESTFY2(40,40),P(400),XD(400),
     5 DISP(600),INBN(70),ID1(5,100),NSTOR(15),P1(10)
        DIMENSION IFR(100),NST(100,10),IFRX(100),IFRY(100),NSTX(100,5),
     1 NSTY(100,5),IPX(50),IPY(50),XS(60),YS(60)
        COMMON NODET,NELEM,NNODE,NDOFN,NBN,LTYPE,IANT
        COMMON IETPX,IETPY
        COMMON NELEMX,NNODEX,NDOFNX
        COMMON NELEMY,NNODEY,NDOFNY
        COMMON NODETX,NODETY
        COMMON/ELPROP/YNG(3),POS,TH
        COMMON/ELPROPX/IDELX1,IDELX2,BRX(2),DPX(2),BFX(2),TFX(2),
     1 CWX(2),CFX(2)
        COMMON/ELPROPY/IDELY1,IDELY2,BRY(2),DPY(2),BFY(2),TFY(2),
     1 CWY(2),CFY(2)
        COMMON IER1,IER2,IER3,IER4,IER5,IER6
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C       INPUT DATA IN FILE SPAP1.DAT
C       FOR CHECKING CORRECTNESS OF INPUT DATA AND ERROR DIAGNOSTIC
C       MESSAGES, REFER TO FILE SPAP2.DAT
C       OUTPUT DATA WILL BE CONTAINED IN FILE SPAP3.DAT
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        OPEN(UNIT=5,FILE='SPAP1.DAT',TYPE='OLD')
        OPEN(UNIT=2,FILE='SPAP2.DAT',TYPE='NEW')
        OPEN(UNIT=6,FILE='SPAP3.DAT',TYPE='NEW')
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C       NODET=TOTAL NUMBER OF NODES IN THE MESH
C       NELEM=TOTAL NUMBER OF PLATE ELEMENTS IN THE MESH
C       NNODE=NUMBER OF NODES PER ELEMENT
C       NDOFN=NUMBER OF DEGREES OF FREEDOM PER NODE
C       NBN=NUMBER OF BOUNDARY NODES
C       LTYPE=1 FOR POINT LOAD; ANY OTHER INTEGER FOR UDL
C       IANT=1 IMPLIES FEM(M1) IS ACTIVATED; FEM(M2) OTHERWISE
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        READ(5,*)NODET,NELEM,NNODE,NDOFN,NBN,LTYPE,IANT
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C       IETPX=1 OR 2 ACCORDING AS THERE ARE ONE OR TWO TYPES OF
C       X-STIFFENERS
```

```
C       IETPY=1 OR 2 ACCORDING AS THERE ARE ONE OR TWO TYPES OF
C       Y-STIFFENERS
C       X()=GLOBAL X-COORDINATE OF A PLATE NODE
C       Y()=GLOBAL Y-COORDINATE OF A PLATE NODE
C       NODES(,)=ELEMENT NODE NUMBER FROM THE FIRST ELEMENT TO THE LAST
C       ELEMENT
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        READ(5,*)IETPX,IETPY
        DO 10 NODE=1,NODET
 10     READ(5,*)X(NODE),Y(NODE)
        DO 20 IE=1,NELEM
        READ(5,*)(NODES(IE,I),I=1,NNODE)
 20     CONTINUE
        NDEF=NODET*NDOFN
        NDEFE=NNODE*NDOFN
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        IF (IANT.EQ.1)THEN
        CONTINUE
        ELSE
        GO TO 6001
        END IF
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C       INPUT STIFFENER CONFIGURATION DATA FOR FEM(M1)
C       NELEMX OR NELEMY=TOTAL NUMBER OF X- OR Y-STIFFENERS
C       NDOFNX OR NDOFNY=NUMBER OF DEGREES OF FREEDOM PER NODE FOR
C       X- OR Y-STIFFENERS
C       NNODEX OR NNODEY=NUMBER OF NODES PER ELEMENT FOR X- OR Y-
C       STIFFENERS
C       NODESX1(IEX,3) OR NODESY1(IEY,3)=NODE NUMBER FOR X- OR Y-
C       STIFFENERS FROM THE FIRST TO THE LAST ELEMENT
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        READ(5,*)NELEMX,NDOFNX,NNODEX
        IF(NELEMX.EQ.0)GO TO 561
        DO 21 IEX=1,NELEMX
        READ(5,*)(NODESX1(IEX,I),I=1,NNODEX)
 21     CONTINUE
 561    READ(5,*)NELEMY,NDOFNY,NNODEY
        IF(NELEMY.EQ.0)GO TO 562
        DO 22 IEY=1,NELEMY
        READ(5,*)(NODESY1(IEY,I),I=1,NNODEY)
 22     CONTINUE
        NDEFEX=NNODEX*NDOFNX
        NDEFEY=NNODEY*NDOFNY
        GO TO 562
```

```
 6001   CONTINUE
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C       INPUT STIFFENER CONFIGURATION DATA FOR FEM(M2)
C       NODETX,NODETY=TOTAL NUMBER OF PSEUDO X-STIFFNER AND
C       Y-STIFFENER NODES
C       NELEMX,NELEMY=TOTAL NUMBER OF X-STIFFENER AND Y-STIFFENER
C       ELEMENTS
C       NODESX2(IEX,1 OR 2)=PSEUDO NODE NUMBER OF AN X-STIFFENER
C       IPX(IEX)=NUMBER OF THE PLATE ELEMENT TO WHICH AN X-STIFFENER
C       ELEMENT IS ATTACHED
C       YS(IEX)=GLOBAL Y-COORDINATE OF AN ORTHOGONAL X-STIFFENER
C       NODESY2(IEY,1 OR 2), IPY(IEY) & XS(IEY) HAVE SIMILAR MEANINGS
C       FOR A Y-STIFFENER AS EXPLAINED ABOVE FOR AN X-STIFFENER
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        READ(5,*)NODETX,NODETY
        READ(5,*)NELEMX,NELEMY
        IF(NELEMX.EQ.0)GO TO 3001
        DO 3002 IEX=1,NELEMX
 3002   READ(5,*)(NODESX2(IEX,I),I=1,2)
        DO 3003 IEX=1,NELEMX
 3003   READ(5,*)IPX(IEX),YS(IEX)
 3001   IF(NELEMY.EQ.0)GO TO 562
        DO 3005 IEY=1,NELEMY
 3005   READ(5,*)(NODESY2(IEY,I),I=1,2)
        DO 3006 IEY=1,NELEMY
 3006   READ(5,*)IPY(IEY),XS(IEY)
C!!!!!!!INITIALIZE ID ARRAY!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 562    DO 101 J=1,NODET
        DO 101 I=1,NDOFN
 101    ID(I,J)=0
        WRITE(2,2007)
 2007   FORMAT(2X,16HCONSTRAINT NODES,3X,18HCONSTRAINT INDICES)
C!!!!!!!INPUT GEOMETRIC BOUNDARY CONDITIONS!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C       NODEB=A BOUNDARY NODE NUMBER
C       ID(K,NODEB)=1 MEANS K TH. DEGREE OF FREEDOM IS FIXED AT NODEB
C       ID(K,NODEB)=0 MEANS K TH. DEGREE OF FREEDOM IS FREE AT NODEB
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        DO 201 I=1,NBN
        READ(5,*)NODEB,((ID(K,NODEB)),K=1,NDOFN)
        INBN(I)=NODEB
        DO 202 K=1,NDOFN
        ID1(K,NODEB)=ID(K,NODEB)
 202    CONTINUE
 201    CONTINUE
```

```
C!!!!!!!MODIFYING THE ID ARRAY!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      KOUNT=0
      DO 501 J=1,NODET
      DO 501 I=1,NDOFN
      IF(ID(I,J).EQ.1)GO TO 601
      KOUNT=KOUNT+1
      ID(I,J)=KOUNT
      NADF=ID(I,J)
      GO TO 501
 601  ID(I,J)=0
 501  CONTINUE
      CALL CAPE(NELEM,NNODE,NDOFN,NODES,LM,ID,NDEFE)
C!!!!!!!READ THE ELEMENT PROPERTIES!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     YNG(1)=YOUNG'S MODULUS FOR THE PLATE MATERIAL
C     POS=POISSON'S RATIO FOR THE X-STIFFENER MATERIAL
C     TH=PLATE THICKNESS
C     YNG(2),YNG(3)=YOUNG'S MODULI FOR THE X-STIFFENER AND Y-STIFFENER
C     MATERIALS RESPECTIVELY
C     BRX(),BRY()=WEB WIDTHS OF AN X-STIFFENER AND A Y-STIFFENER
C     RESPECTIVELY
C     DPX(),DPY()=DEPTHS OF WEB OF AN X-STIFFENER AND A Y-STIFFENER
C     RESPECTIVELY
C     BFX(),BFY()=FLANGE WIDTHS OF AN X-STIFFENER AND A Y-STIFFENER
C     RESPECTIVELY
C     TFX(),TFY()=FLANGE THICKNESSES OF AN X-STIFFENER AND A
C     Y-STIFFENER RESPECTIVELY
C     CWX(),CWY()=TORSIONAL RIGIDITY CONSTANTS FOR AN X-STIFFENER AND
C     A Y-STIFFENER RESPECTIVELY
C     IDELX1,IDELX2=LIMITS OF X-STIFFENER ELEMENTS OF SECOND TYPE OF
C     GEOMETRY WHEN IETPX=2
C     IDELY1,IDELY2=LIMITS OF Y-STIFFENER ELEMENTS OF SECOND TYPE OF
C     GEOMETRY WHEN IETPY=2
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      READ(5,*)YNG(1),POS,TH
      IF(NELEMX.GT.0)THEN
      READ(5,*) YNG(2)
      READ(5,*)BRX(1),DPX(1),BFX(1),TFX(1),CWX(1),CFX(1)
      IF(IETPX.EQ.1)GO TO 1800
      READ(5,*)IDELX1,IDELX2
      READ(5,*)BRX(2),DPX(2),BFX(2),TFX(2),CWX(2),CFX(2)
 1800 IF(IANT.EQ.1)THEN
      CALL CASEX(NELEMX,NNODEX,NDOFNX,NODESX1,LMX,ID,NDEFEX)
      END IF
      END IF
```

```
      IF(NELEMY.GT.0)THEN
      READ(5,*) YNG(3)
      READ(5,*)BRY(1),DPY(1),BFY(1),TFY(1),CWY(1),CFY(1)
      IF(IETPY.EQ.1)GO TO 1801
      READ(5,*)IDELY1,IDELY2
      READ(5,*)BRY(2),DPY(2),BFY(2),TFY(2),CWY(2),CFY(2)
 1801 IF(IANT.EQ.1)THEN
      CALL CASEY(NELEMY,NNODEY,NDOFNY,NODESY1,LMY,ID,NDEFEY)
      END IF
      END IF
      IF(LTYPE-1)99,990,99
C!!!!!!!READ UNIFORMLY DISTRIBUTED LOAD DATA!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     QZ()=UNIFORMLY DISTRIBUTED LOAD FOR THE WHOLE PLATE
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 99   READ(5,*)QZ(1)
      DO 505 I=2,NELEM
 505  QZ(I)=QZ(1)
      GO TO 992
C!!!!!!!READ CONCENTRATED LOAD DATA!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     NNCL=TOTAL NUMBER OF NODES AT WHICH CONCENTRATED LOAD IS APPLIED
C     NODEC=NUMBER OF A NODE CARRYING POINT LOAD/LOADS
C     P()=A NODAL POINT LOAD IN THE DIRECTION OF THE CORRESPONDING
C     NODAL DEGREE OF FREEDOM
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 990  READ(5,*)NNCL
      DO 901 INCL=1,NNCL
      READ(5,*)NODEC,(P(ID(I,NODEC)),I=1,NDOFN)
      NSTOR(INCL)=NODEC
 901  P1(INCL)=P(ID(3,NODEC))
      GO TO 993
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 992  CALL GCLVA(NELEM,NODES,QZ,P,X,Y,LM)
 993  CALL CONS(AE,G,D,SR)
      DO 30 IE=1,NELEM
      CALL ESMB(NODES,X,Y,IE,
     1 TH,POS,AE,G,D,SR,NNODE,NDEFE,ESTF,IER1)
      IF(IER1.EQ.1)GO TO 575
      CALL GSMB(LM,ESTF,IE,NADF,NDEFE,OSTF)
 30   CONTINUE
      IF(NELEMX.EQ.0)GO TO 41
      DO 40 IEX=1,NELEMX
      IF(IANT.EQ.1)THEN
      CALL ESMBX1(NNODEX,NODESX1,X,IEX,ESTFX1,IER2)
      IF(IER2.EQ.1)GO TO 575
```

```
      CALL GSMBX1(LMX,ESTFX1,IEX,NADF,NDEFEX,OSTF)
      ELSE
      K1=IPX(IEX)
      CALL ESMBX2(NNODE,IEX,IPX,YS,X,Y,NODES,ESTFX2,IER4)
      IF(IER4.EQ.1)GO TO 575
      CALL GSMB(LM,ESTFX2,K1,NADF,NDEFE,OSTF)
      END IF
40    CONTINUE
41    IF(NELEMY.EQ.0)GO TO 670
      DO 50 IEY=1,NELEMY
      IF(IANT.EQ.1)THEN
      CALL ESMBY1(NNODEY,NODESY1,Y,IEY,ESTFY1,IER3)
      IF(IER3.EQ.1)GO TO 575
      CALL GSMBY1(LMY,ESTFY1,IEY,NADF,NDEFEY,OSTF)
      ELSE
      K2=IPY(IEY)
      CALL ESMBY2(NNODE,IEY,IPY,XS,X,Y,NODES,ESTFY2,IER5)
      IF(IER5.EQ.1)GO TO 575
      CALL GSMB(LM,ESTFY2,K2,NADF,NDEFE,OSTF)
      END IF
50    CONTINUE
670   CALL SOLV(NADF,OSTF,P,XD,IER6)
      IF(IER6.EQ.1)GO TO 575
      D1=0.0
      K=0
      KOUNT=0
      DO 661 J=1,NODET
      DO 661 I=1,NDOFN
      IF(ID(I,J).EQ.0)THEN
      KOUNT=KOUNT+1
      DISP(KOUNT)=D1
      ELSE
      K=K+1
      KOUNT=KOUNT+1
      DISP(KOUNT)=XD(K)
      END IF
661   CONTINUE
      WRITE(6,657)
657   FORMAT(2X,8HNODE NO.,6X,1HU,13X,1HV,13X,1HW,10X,6HTHETAX,
     1 7X,6HTHETAY)
      DO 658 NN=1,NODET
      WRITE(6,659)NN,(DISP(5*NN+MM-5),MM=1,5)
659   FORMAT(4X,I3,3X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4)
658   CONTINUE
```

```
      DO 559 K=1,NODET
      DO 559 IM=1,NELEM
      DO 559 IE=1,NNODE
      IF(NODES(IM,IE).EQ.K)THEN
      IFR(K)=IFR(K)+1
      NST(K,IFR(K))=IM
      ELSE
      END IF
559   CONTINUE
      CALL STRESS(POS,TH,AE,G,D,SR,NELEM,NNODE,X,Y,NODES,DISP,
     1 NODET,IFR,NST)
      IF(NELEMX.EQ.0)GO TO 565
      IF(IANT.EQ.1)THEN
      DO 560 K=1,NODET
      DO 560 IEX=1,NELEMX
      DO 560 IN=1,NNODEX
      IF(NODESX1(IEX,IN).EQ.K)THEN
      IFRX(K)=IFRX(K)+1
      NSTX(K,IFRX(K))=IEX
      ELSE
      END IF
560   CONTINUE
      ELSE
      DO 5611 K=1,NODET
      DO 5611 IEX=1,NELEMX
      DO 5611 IN=1,2
      IF(NODESX2(IEX,IN).EQ.K)THEN
      IFRX(K)=IFRX(K)+1
      NSTX(K,IFRX(K))=IEX
      ELSE
      END IF
5611  CONTINUE
      END IF
      IF(IANT.EQ.1)THEN
      CALL STRESSX1(NELEMX,NODET,NODESX1,NNODEX,X,DISP,IFRX,NSTX)
      ELSE
      CALL STRESSX2(NELEMX,NNODE,NODETX,NODESX2,X,Y,YS,DISP,
     1 NODES,IFRX,IPX,NSTX)
      END IF
565   IF(NELEMY.EQ.0)GO TO 575
      IF(IANT.EQ.1)THEN
      DO 570 K=1,NODET
      DO 570 IEY=1,NELEMY
      DO 570 IN=1,NNODEY
```

```
      IF(NODESY1(IEY,IN).EQ.K)THEN
      IFRY(K)=IFRY(K)+1
      NSTY(K,IFRY(K))=IEY
      ELSE
      END IF
570   CONTINUE
      ELSE
      DO 571 K=1,NODET
      DO 571 IEY=1,NELEMY
      DO 571 IN=1,2
      IF(NODESY2(IEY,IN).EQ.K)THEN
      IFRY(K)=IFRY(K)+1
      NSTY(K,IFRY(K))=IEY
      ELSE
      END IF
571   CONTINUE
      END IF
      IF(IANT.EQ.1)THEN
      CALL STRESSY1(NELEMY,NODET,NODESY1,NNODEY,Y,DISP,IFRY,NSTY)
      ELSE
      CALL STRESSY2(NELEMY,NNODE,NODETY,NODESY2,X,Y,XS,DISP,
     1 NODES,IFRY,IPY,NSTY)
      END IF
575   CALL DOCTOR(X,Y,NODES,NODESX1,NODESX2,NODESY1,NODESY2,
     1 IPX,YS,IPY,XS,INBN,ID1,NNCL,NSTOR,P1,QZ)
      STOP
      END


      SUBROUTINE GAUSQ2(GSPX,GSPY,W1)
      DIMENSION XG(2),YG(2),W(2),W1(4),GSPX(4),GSPY(4)
      XG(1)=0.5773502692
      XG(2)=-0.5773502692
      YG(1)=XG(1)
      YG(2)=XG(2)
      DO 10 K=1,4
      W1(K)=1.0
10    CONTINUE
      GSPX(1)=-XG(1)
      GSPX(2)=XG(1)
      GSPX(3)=XG(1)
      GSPX(4)=-XG(1)
      GSPY(1)=XG(1)
```

```
      GSPY(2)=XG(1)
      GSPY(3)=-XG(1)
      GSPY(4)=-XG(1)
      RETURN
      END


      SUBROUTINE GAUSQ21(GSPX,GSPY,W1)
      DIMENSION GSPX(2),GSPY(2)
      GSPX(1)=-0.5773502692
      GSPX(2)=0.5773502692
      GSPY(1)=GSPX(1)
      GSPY(2)=GSPX(2)
      W1=1.0
      RETURN
      END


      SUBROUTINE SHAPE(S,T,SH,ADERIV)
      DIMENSION SH(8),ADERIV(2,8)
      SH(1)=0.25*(1.-S)*(1.+T)*(-S+T-1)
      SH(2)=0.5*(1.+T)*(1.-S**2)
      SH(3)=0.25*(1.+S)*(1.+T)*(S+T-1)
      SH(4)=0.5*(1.+S)*(1.-T**2)
      SH(5)=0.25*(1.+S)*(1.-T)*(S-T-1)
      SH(6)=0.5*(1.-T)*(1.-S**2)
      SH(7)=0.25*(1.-S)*(1.-T)*(-S-T-1.)
      SH(8)=0.5*(1.-S)*(1.-T**2)
      S2=2.*S
      T2=2.*T
      ST2=2.*S*T
      ADERIV(1,1)=0.25*(S2-T+ST2-T*T)
      ADERIV(1,2)=0.5*(-S2-ST2)
      ADERIV(1,3)=0.25*(S2+T+ST2+T*T)
      ADERIV(1,4)=0.5*(1.-T*T)
      ADERIV(1,5)=0.25*(S2-T-ST2+T*T)
      ADERIV(1,6)=0.5*(-S2+ST2)
      ADERIV(1,7)=0.25*(S2+T-ST2-T*T)
      ADERIV(1,8)=0.5*(-1.+T*T)
      ADERIV(2,1)=0.25*(T2-S+S*S-ST2)
```

```
      ADERIV(2,2)=0.5*(1.-S*S)
      ADERIV(2,3)=0.25*(T2+S+S*S+ST2)
      ADERIV(2,4)=0.5*(-T2-ST2)
      ADERIV(2,5)=0.25*(T2-S-S*S+ST2)
      ADERIV(2,6)=0.5*(-1.+S*S)
      ADERIV(2,7)=0.25*(T2+S-S*S-ST2)
      ADERIV(2,8)=0.5*(-T2+ST2)
      RETURN
      END



      SUBROUTINE JACOB(IE,X,Y,NODES,
     1 ADERIV,XDJAC,CARTD,IER1)
      DIMENSION X(100),Y(100),NODES(100,8),EJAC(2,2),EJINV(2,2),
     1 ADERIV(2,8),CARTD(2,8)
      DO 9 I=1,2
      DO 9 J=1,2
      EJAC(I,J)=0.
      EJINV(I,J)=0.
9     CONTINUE
      DO 10 I=1,8
      EJAC(1,1)=EJAC(1,1)+ADERIV(1,I)*X(NODES(IE,I))
      EJAC(1,2)=EJAC(1,2)+ADERIV(1,I)*Y(NODES(IE,I))
      EJAC(2,1)=EJAC(2,1)+ADERIV(2,I)*X(NODES(IE,I))
      EJAC(2,2)=EJAC(2,2)+ADERIV(2,I)*Y(NODES(IE,I))
10    CONTINUE
      XDJAC=EJAC(1,1)*EJAC(2,2)-EJAC(1,2)*EJAC(2,1)
      IF(XDJAC.LE.0.0) GO TO 52
      EJINV(1,1)=EJAC(2,2)/XDJAC
      EJINV(1,2)=-EJAC(1,2)/XDJAC
      EJINV(2,1)=-EJAC(2,1)/XDJAC
      EJINV(2,2)=EJAC(1,1)/XDJAC
      DO 20 I=1,8
      CARTD(1,I)=EJINV(1,1)*ADERIV(1,I)+EJINV(1,2)*ADERIV(2,I)
      CARTD(2,I)=EJINV(2,1)*ADERIV(1,I)+EJINV(2,2)*ADERIV(2,I)
20    CONTINUE
      GO TO 53
52    IER1=1
53    RETURN
      END
```

```
      SUBROUTINE CONS(AE,G,D,SR)
      COMMON/ELPROP/YNG(3),POS,TH
      AE=YNG(1)*TH/(1.-POS**2)
      G=YNG(1)*TH/(2.*(1.+POS))
      D=YNG(1)*TH**3/(12.*(1.-POS**2))
      SR=YNG(1)*TH/(2.4*(1.+POS))
      RETURN
      END
```

```
      SUBROUTINE CONS11(BRX,DPX,BFX,TFX,ASX,DSX,SSX,EX,
     1 CWX,CFX,TRX)
      COMMON/ELPROP/YNG(3),POS,TH
      COMMON CSX,XI,GX
      CSX=BRX*DPX+BFX*TFX
      ASX=YNG(2)*(BRX*DPX+BFX*TFX)
      GX=(BRX*DPX*(DPX/2.+TFX)+BFX*TFX*TFX/2.)/
     1 (BRX*DPX+BFX*TFX)
      XI=BRX*DPX**3/12.+BRX*DPX*(DPX/2.+TFX-GX)**2+
     1 BFX*TFX**3/12.+BFX*TFX*(GX-TFX/2.)**2
      DSX=YNG(2)*XI
      EX=DPX+TFX+TH/2.-GX
      SSX=YNG(2)*(BRX*DPX+BFX*TFX)/(3.0*(1.+POS))
      TRX=YNG(2)*(CWX*DPX*BRX**3+CFX*BFX*TFX**3)/(2.*(1.+POS))
      RETURN
      END
```

```
      SUBROUTINE CONS21(BRY,DPY,BFY,TFY,ASY,DSY,SSY,EY,
     1 CWY,CFY,TRY)
      COMMON/ELPROP/YNG(3),POS,TH
      COMMON CSY,YI,GY
      CSY=BRY*DPY+BFY*TFY
      ASY=YNG(3)*(BRY*DPY+BFY*TFY)
      GY=(BRY*DPY*(DPY/2.+TFY)+BFY*TFY*TFY/2.)/
     1 (BRY*DPY+BFY*TFY)
      YI=BRY*DPY**3/12.+BRY*DPY*(DPY/2.+TFY-GY)**2+
     1 BFY*TFY**3/12.+BFY*TFY*(GY-TFY/2.)**2
      DSY=YNG(3)*YI
      EY=DPY+TFY+TH/2.-GY
```

```
      SSY=YNG(3)*(BRY*DPY+BFY*TFY)/(3.0*(1.+POS))
      TRY=YNG(3)*(CWY*DPY*BRY**3+CFY*BFY*TFY**3)/(2.*(1.+POS))
      RETURN
      END


      SUBROUTINE CONS12(BRX,DPX,BFX,TFX,ASX1,ASX2,DSX,SSX,
     1 CWX,CFX,TRX)
      COMMON/ELPROP/YNG(3),POS,TH
      ASX1=YNG(2)*(BRX*DPX+BFX*TFX)
      ASX2=0.5*YNG(2)*(BRX*DPX*(DPX+TH)+BFX*TFX*(2.*DPX+TH+TFX))
      C1=(DPX+0.5*TH)**3-(0.5*TH)**3
      C2=(DPX+0.5*TH+TFX)**3-(DPX+0.5*TH)**3
      DSX=YNG(2)*(BRX*C1+BFX*C2)/3.
      SSX=YNG(2)*(BRX*DPX+BFX*TFX)/(3.0*(1.+POS))
      TRX=YNG(2)*(CWX*DPX*BRX**3+CFX*BFX*TFX**3)/(2.*(1.+POS))
      RETURN
      END


      SUBROUTINE CONS22(BRY,DPY,BFY,TFY,ASY1,ASY2,DSY,SSY,
     1 CWY,CFY,TRY)
      COMMON/ELPROP/YNG(3),POS,TH
      ASY1=YNG(3)*(BRY*DPY+BFY*TFY)
      ASY2=0.5*YNG(3)*(BRY*DPY*(DPY+TH)+BFY*TFY*(2.*DPY+TH+TFY))
      C1=(DPY+0.5*TH)**3-(0.5*TH)**3
      C2=(DPY+0.5*TH+TFY)**3-(DPY+0.5*TH)**3
      DSY=YNG(3)*(BRY*C1+BFY*C2)/3.
      SSY=YNG(3)*(BRY*DPY+BFY*TFY)/(3.0*(1.+POS))
      TRY=YNG(3)*(CWY*DPY*BRY**3+CFY*BFY*TFY**3)/(2.*(1.+POS))
      RETURN
      END


      SUBROUTINE SHAPE1(S,SH,DERIV)
      DIMENSION SH(3),DERIV(3)
      SH(1)=-0.5*S*(1.-S)
      SH(2)=1.-S**2
```

```
      SH(3)=0.5*S*(1.+S)
      DERIV(1)=0.5*(-1.+2.*S)
      DERIV(2)=-2.*S
      DERIV(3)=0.5*(1.+2.*S)
      RETURN
      END


      SUBROUTINE JACOBX1(IE,X,NODESX1,DERIV,DJAC,CARTD,IER2)
      DIMENSION X(100),NODESX1(100,3),DERIV(3),CARTD(3)
      EJAC=DERIV(1)*X(NODESX1(IE,1))+DERIV(2)*X(NODESX1(IE,2))
     1 +DERIV(3)*X(NODESX1(IE,3))
      DJAC=EJAC
      IF(DJAC.LE.0.)GO TO 10
      EJINV=1./DJAC
      CARTD(1)=EJINV*DERIV(1)
      CARTD(2)=EJINV*DERIV(2)
      CARTD(3)=EJINV*DERIV(3)
      GO TO 20
 10   IER2=1
 20   RETURN
      END


      SUBROUTINE JACOBY1(IE,Y,NODESY1,DERIV,DJAC,CARTD,IER3)
      DIMENSION Y(100),NODESY1(100,3),DERIV(3),CARTD(3)
      EJAC=DERIV(1)*Y(NODESY1(IE,1))+DERIV(2)*Y(NODESY1(IE,2))
     1 +DERIV(3)*Y(NODESY1(IE,3))
      DJAC=EJAC
      IF(DJAC.LE.0.)GO TO 10
      EJINV=1./DJAC
      CARTD(1)=EJINV*DERIV(1)
      CARTD(2)=EJINV*DERIV(2)
      CARTD(3)=EJINV*DERIV(3)
      GO TO 20
 10   IER3=1
 20   RETURN
      END
```

```
      SUBROUTINE JACOBX2(IE,NNODE,X,NODES,ADERIV,XDJAC,CARTD,IER4)
      DIMENSION X(100),NODES(100,8),ADERIV(2,8),CARTD(2,8)
      EJAC=0.0
      DO 10 I=1,NNODE
      EJAC=EJAC+ADERIV(1,I)*X(NODES(IE,I))
10    CONTINUE
      XDJAC=EJAC
      IF(XDJAC.LE.0.0)GO TO 30
      EJINV=1./XDJAC
      DO 20 I=1,NNODE
      CARTD(1,I)=EJINV*ADERIV(1,I)
20    CONTINUE
      GO TO 40
30    IER4=1
40    RETURN
      END
```

```
      SUBROUTINE JACOBY2(IE,NNODE,Y,NODES,ADERIV,YDJAC,CARTD,IER5)
      DIMENSION Y(100),NODES(100,8),ADERIV(2,8),CARTD(2,8)
      EJAC=0.0
      DO 10 I=1,NNODE
      EJAC=EJAC+ADERIV(2,I)*Y(NODES(IE,I))
10    CONTINUE
      YDJAC=EJAC
      IF(YDJAC.LE.0.0)GO TO 30
      EJINV=1./YDJAC
      DO 20 I=1,NNODE
      CARTD(2,I)=EJINV*ADERIV(2,I)
20    CONTINUE
      GO TO 40
30    IER5=1
40    RETURN
      END
```

```
      SUBROUTINE ESMB(NODES,X,Y,IE,
     1 TH,POS,AE,G,D,SR,NNODE,NDEFE,ESTF)
      DIMENSION TEMP(25),CARTD(2,8),SH(8),ESTF(40,40),ADERIV(2,8)
      DIMENSION GSPX(4),GSPY(4),W1(4),NODES(100,8),X(100),Y(100)
```

```
C!!!!!!!ASSEMBLES PLATE STIFFNESS MATRIX BY REDUCED INTEGRATION!!!!!!!!!
      CALL GAUSQ2(GSPX,GSPY,W1)
      DO 991 I=1,NNODE
      DO 991 J=1,NNODE
      DO 70 I1=1,25
 70   TEMP(I1)=0.0
      DO 80 K=1,4
      S=GSPX(K)
      T=GSPY(K)
      CALL SHAPE(S,T,SH,ADERIV)
      CALL JACOB(IE,X,Y,NODES,ADERIV,
     1 XDJAC,CARTD,IER1)
      CST=W1(K)*XDJAC
      TEMP(1)=TEMP(1)+CST*(AE*CARTD(1,I)*CARTD(1,J)+G*CARTD(2,I)*
     1 CARTD(2,J))
      TEMP(2)=TEMP(2)+CST*(POS*AE*CARTD(1,I)*CARTD(2,J)+
     1 G*CARTD(2,I)*CARTD(1,J))
      TEMP(6)=TEMP(6)+CST*(POS*AE*CARTD(2,I)*CARTD(1,J)+
     1 G*CARTD(1,I)*CARTD(2,J))
      TEMP(7)=TEMP(7)+CST*(AE*CARTD(2,I)*CARTD(2,J)+
     1 G*CARTD(1,I)*CARTD(1,J))
      TEMP(13)=TEMP(13)+CST*(SR*CARTD(1,I)*CARTD(1,J)+
     1 SR*CARTD(2,I)*CARTD(2,J))
      TEMP(14)=TEMP(14)-CST*SR*CARTD(1,I)*SH(J)
      TEMP(15)=TEMP(15)-CST*SR*CARTD(2,I)*SH(J)
      TEMP(18)=TEMP(18)-CST*SR*SH(I)*CARTD(1,J)
      TEMP(19)=TEMP(19)+CST*(D*CARTD(1,I)*CARTD(1,J)+
     1 D*(1.-POS)*CARTD(2,I)*CARTD(2,J)/2.0+SR*SH(I)*SH(J))
      TEMP(20)=TEMP(20)+CST*(D*POS*CARTD(1,I)*CARTD(2,J)+
     1 D*(1.-POS)*CARTD(2,I)*CARTD(1,J)/2.0)
      TEMP(23)=TEMP(23)-CST*SR*SH(I)*CARTD(2,J)
      TEMP(24)=TEMP(24)+CST*(POS*D*CARTD(2,I)*CARTD(1,J)+
     1 D*(1.-POS)*CARTD(1,I)*CARTD(2,J)/2.0)
      TEMP(25)=TEMP(25)+CST*(D*CARTD(2,I)*CARTD(2,J)+
     1 D*(1.-POS)*CARTD(1,I)*CARTD(1,J)/2.0+SR*SH(I)*SH(J))
 80   CONTINUE
      L=0
      DO 991 II=5*I-4,5*I
      DO 991 JJ=5*J-4,5*J
      L=L+1
      ESTF(II,JJ)=TEMP(L)
 991  CONTINUE
      RETURN
      END
```

```
      SUBROUTINE GSMB(LM,E,IE,NADF,NDEFE,O)
      DIMENSION LM(100,40),E(40,40),O(100000)
      DO 10 I=1,NDEFE
      LM1=LM(IE,I)
      DO 10 J=I,NDEFE
      LM2=LM(IE,J)
      IF(LM2.EQ.0)GO TO 10
      IF(LM1.EQ.0)GO TO 10
      IF(LM1.LE.LM2)GO TO 20
      GO TO 30
20    IF(LM1.EQ.1)THEN
      K=LM1+LM2-1
      O(K)=O(K)+E(I,J)
      ELSE IF (LM1.GT.1)THEN
      LS=0
      DO 25 II=1,LM1-1
25    LS=(NADF-II+1)+LS
      K=LS+(LM2-LM1+1)
      O(K)=O(K)+E(I,J)
      END IF
      GO TO 10
30    LS=0
      DO 35 JJ=1,LM2-1
35    LS=(NADF-JJ+1)+LS
      K=LS+(LM1-LM2+1)
      O(K)=O(K)+E(I,J)
10    CONTINUE
      RETURN
      END


      SUBROUTINE ESMBX1(NNODEX,NODESX1,X,IE,ESTFX1,IER2)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     COMPUTES X-STIFFENER ELEMENT MATRIX IN FEM(M1) BY REDUCED
C     INTEGRATION
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION NODESX1(100,3),X(100),ESTFX1(15,15),SH(3),DERIV(3),
     1 CARTD(3),TEMP(25),GSPX(2),GSPY(2)
      COMMON/ELPROP/YNG(3),POS,TH
      COMMON/ELPROPX/IDELX1,IDELX2,BRX(2),DPX(2),BFX(2),TFX(2),
     1 CWX(2),CFX(2)
      IF(IDELX1.EQ.0)THEN
```

```
      BRXX=BRX(1)
      DPXX=DPX(1)
      BFXX=BFX(1)
      TFXX=TFX(1)
      CWXX=CWX(1)
      CFXX=CFX(1)
      GO TO 5
      ELSE IF(IE.GE.IDELX1.AND.IE.LE.IDELX2)THEN
      BRXX=BRX(2)
      DPXX=DPX(2)
      BFXX=BFX(2)
      TFXX=TFX(2)
      CWXX=CWX(2)
      CFXX=CFX(2)
      GO TO 5
      ELSE
      BRXX=BRX(1)
      DPXX=DPX(1)
      BFXX=BFX(1)
      TFXX=TFX(1)
      CWXX=CWX(1)
      CFXX=CFX(1)
      END IF
5     CALL CONS11(BRXX,DPXX,BFXX,TFXX,ASX,DSX,SSX,EX,
     1 CWXX,CFXX,TRX)
      CALL GAUSQ21(GSPX,GSPY,W1)
      DO 10 I=1,NNODEX
      DO 10 J=1,NNODEX
      DO 20 II=1,25
20    TEMP(II)=0.0
      DO 30 K=1,2
      S=GSPX(K)
      CALL SHAPE1(S,SH,DERIV)
      CALL JACOBX1(IE,X,NODESX1,DERIV,DJAC,CARTD,IER2)
      IF(IER2.EQ.1)GO TO 40
      CST=W1*DJAC
      TEMP(1)=TEMP(1)+ASX*CARTD(I)*CARTD(J)*CST
      TEMP(4)=TEMP(4)-CST*EX*ASX*CARTD(I)*CARTD(J)
      TEMP(13)=TEMP(13)+CST*SSX*CARTD(I)*CARTD(J)
      TEMP(14)=TEMP(14)-CST*SSX*CARTD(I)*SH(J)
      TEMP(16)=TEMP(16)-CST*EX*ASX*CARTD(I)*CARTD(J)
      TEMP(18)=TEMP(18)-CST*SSX*SH(I)*CARTD(J)
      TEMP(19)=TEMP(19)+CST*(EX**2*ASX*CARTD(I)*CARTD(J)+
     1. DSX*CARTD(I)*CARTD(J)+SSX*SH(I)*SH(J))
```

```
      TEMP(25)=TEMP(25)+CST*TRX*CARTD(I)*CARTD(J)
30    CONTINUE
      L=0
      DO 10 II=5*I-4,5*I
      DO 10 JJ=5*J-4,5*J
      L=L+1
      ESTFX1(II,JJ)=TEMP(L)
10    CONTINUE
40    RETURN
      END


      SUBROUTINE ESMBY1(NNODEY,NODESY1,Y,IE,ESTFY1,IEY3)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     COMPUTES Y-STIFFENER ELEMENT MATRIX IN FEM(M1) BY REDUCED
C     INTEGRATION
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION NODESY1(100,3),Y(100),ESTFY1(15,15),SH(3),DERIV(3),
    1 CARTD(3),TEMP(25),GSPX(2),GSPY(2)
      COMMON/ELPROP/YNG(3),POS,TH
      COMMON/ELPROPY/IDELY1,IDELY2,BRY(2),DPY(2),BFY(2),TFY(2),
    1 CWY(2),CFY(2)
      IF(IDELY1.EQ.0)THEN
      BRYY=BRY(1)
      DPYY=DPY(1)
      BFYY=BFY(1)
      TFYY=TFY(1)
      CWYY=CWY(1)
      CFYY=CFY(1)
      GO TO 5
      ELSE IF(IE.GE.IDELY1.AND.IE.LE.IDELY2)THEN
      BRYY=BRY(2)
      DPYY=DPY(2)
      BFYY=BFY(2)
      TFYY=TFY(2)
      CWYY=CWY(2)
      CFYY=CFY(2)
      GO TO 5
      ELSE
      BRYY=BRY(1)
      DPYY=DPY(1)
      BFYY=BFY(1)
```

```
      TFYY=TFY(1)
      CWYY=CWY(1)
      CFYY=CFY(1)
      END IF
5     CALL CONS21(BRYY,DPYY,BFYY,TFYY,ASY,DSY,SSY,EY,
     1 CWYY,CFYY,TRY)
      CALL GAUSQ21(GSPX,GSPY,W1)
      DO 10 I=1,NNODEY
      DO 10 J=1,NNODEY
      DO 20 II=1,25
20    TEMP(II)=0.0
      DO 30 K=1,2
      S=GSPY(K)
      CALL SHAPE1(S,SH,DERIV)
      CALL JACOBY1(IE,Y,NODESY1,DERIV,DJAC,CARTD,IEY3)
      IF(IEY3.EQ.1)GO TO 40
      CST=W1*DJAC
      TEMP(7)=TEMP(7)+ASY*CARTD(I)*CARTD(J)*CST
      TEMP(10)=TEMP(10)-CST*EY*ASY*CARTD(I)*CARTD(J)
      TEMP(13)=TEMP(13)+CST*SSY*CARTD(I)*CARTD(J)
      TEMP(15)=TEMP(15)-CST*SSY*CARTD(I)*SH(J)
      TEMP(19)=TEMP(19)+CST*TRY*CARTD(I)*CARTD(J)
      TEMP(22)=TEMP(22)-CST*EY*ASY*CARTD(I)*CARTD(J)
      TEMP(23)=TEMP(23)-CST*SSY*SH(I)*CARTD(J)
      TEMP(25)=TEMP(25)+CST*(EY**2*ASY*CARTD(I)*CARTD(J)+
     1 DSY*CARTD(I)*CARTD(J)+SSY*SH(I)*SH(J))
30    CONTINUE
      L=0
      DO 10 II=5*I-4,5*I
      DO 10 JJ=5*J-4,5*J
      L=L+1
      ESTFY1(II,JJ)=TEMP(L)
10    CONTINUE
40    RETURN
      END


      SUBROUTINE ESMBX2(NNODE,IEX,IPX,YS,X,Y,NODES,ESTFX2,IER4)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     COMPUTES X-STIFFENER ELEMENT MATRIX IN FEM(M2) BY REDUCED
C     INTEGRATION
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
      DIMENSION NODES(100,8),ESTFX2(40,40),SH(8),ADERIV(2,8),
     1 CARTD(2,8),TEMP(25),GSPX(2),GSPY(2),IPX(50),YS(50),
     2 X(100),Y(100),GSPX3(3),GSPY3(3),W3(3)
      COMMON/ELPROP/YNG(3),POS,TH
      COMMON/ELPROPX/IDELX1,IDELX2,BRX(2),DPX(2),BFX(2),TFX(2),
     1 CWX(2),CFX(2)
      IF(IDELX1.EQ.0)THEN
      BRXX=BRX(1)
      DPXX=DPX(1)
      BFXX=BFX(1)
      TFXX=TFX(1)
      CWXX=CWX(1)
      CFXX=CFX(1)
      GO TO 5
      ELSE IF(IEX.GE.IDELX1.AND.IEX.LE.IDELX2)THEN
      BRXX=BRX(2)
      DPXX=DPX(2)
      BFXX=BFX(2)
      TFXX=TFX(2)
      CWXX=CWX(2)
      CFXX=CFX(2)
      GO TO 5
      ELSE
      BRXX=BRX(1)
      DPXX=DPX(1)
      BFXX=BFX(1)
      TFXX=TFX(1)
      CWXX=CWX(1)
      CFXX=CFX(1)
      END IF
5     CALL CONS12(BRXX,DPXX,BFXX,TFXX,ASX1,ASX2,DSX,SSX,
     1 CWXX,CFXX,TRX)
      K1=IPX(IEX)
      A=0.5*(Y(NODES(K1,2))-Y(NODES(K1,6)))
      B=YS(IEX)-0.5*(Y(NODES(K1,2))+Y(NODES(K1,6)))
      XETA=B/A
      CALL GAUSQ21(GSPX,GSPY,W1)
      DO 10 I=1,NNODE
      DO 10 J=1,NNODE
      DO 20 II=1,25
20    TEMP(II)=0.0
      DO 30 K=1,2
      S=GSPX(K)
      T=XETA
```

```
      CALL SHAPE(S,T,SH,ADERIV)
      CALL JACOBX2(K1,NNODE,X,NODES,ADERIV,XDJAC,CARTD,IER4)
      IF(IER4.EQ.1)GO TO 40
      CST=W1*XDJAC
      TEMP(1)=TEMP(1)+CST*ASX1*CARTD(1,I)*CARTD(1,J)
      TEMP(4)=TEMP(4)-CST*ASX2*CARTD(1,I)*CARTD(1,J)
      TEMP(13)=TEMP(13)+CST*SSX*CARTD(1,I)*CARTD(1,J)
      TEMP(14)=TEMP(14)-CST*SSX*CARTD(1,I)*SH(J)
      TEMP(16)=TEMP(4)
      TEMP(18)=TEMP(18)-CST*SSX*SH(I)*CARTD(1,J)
      TEMP(19)=TEMP(19)+CST*(DSX*CARTD(1,I)*CARTD(1,J)+
     1 SSX*SH(I)*SH(J))
      TEMP(25)=TEMP(25)+CST*TRX*CARTD(1,I)*CARTD(1,J)
 30   CONTINUE
      L=0
      DO 10 II=5*I-4,5*I
      DO 10 JJ=5*J-4,5*J
      L=L+1
      ESTFX2(II,JJ)=TEMP(L)
 10   CONTINUE
 40   RETURN
      END


      SUBROUTINE ESMBY2(NNODE,IEY,IPY,XS,X,Y,NODES,ESTFY2,IER5)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     COMPUTES X-STIFFENER ELEMENT MATRIX IN FEM(M1) BY REDUCED
C     INTEGRATION
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION NODES(100,8),ESTFY2(40,40),SH(8),ADERIV(2,8),CARTD(2,8
     1 ),TEMP(25),GSPX(2),GSPY(2),IPY(50),XS(60),X(100),Y(100),
     2 GSPX3(3),GSPY3(3),W3(3)
      COMMON/ELPROP/YNG(3),POS,TH
      COMMON/ELPROPY/IDELY1,IDELY2,BRY(2),DPY(2),BFY(2),TFY(2),
     1 CWY(2),CFY(2)
      IF(IDELY1.EQ.0)THEN
      BRYY=BRY(1)
      DPYY=DPY(1)
      BFYY=BFY(1)
      TFYY=TFY(1)
      CWYY=CWY(1)
      CFYY=CFY(1)
```

```
      GO TO 5
      ELSE IF(IEY.GE.IDELY1.AND.IEY.LE.IDELY2)THEN
      BRYY=BRY(2)
      DPYY=DPY(2)
      BFYY=BFY(2)
      TFYY=TFY(2)
      CWYY=CWY(2)
      CFYY=CFY(2)
      GO TO 5
      ELSE
      BRYY=BRY(1)
      DPYY=DPY(1)
      BFYY=BFY(1)
      TFYY=TFY(1)
      CWYY=CWY(1)
      CFYY=CFY(1)
      END IF
5     CALL CONS22(BRYY,DPYY,BFYY,TFYY,ASY1,ASY2,DSY,SSY,
     1 CWYY,CFYY,TRY)
      K1=IPY(IEY)
      A=0.5*(X(NODES(K1,4))-X(NODES(K1,8)))
      B=XS(IEY)-0.5*(X(NODES(K1,4))+X(NODES(K1,8)))
      EETA=B/A
      CALL GAUSQ21(GSPX,GSPY,W1)
      DO 10 I=1,NNODE
      DO 10 J=1,NNODE
      DO 20 II=1,25
20    TEMP(II)=0.0
      DO 30 K=1,2
      S=EETA
      T=GSPY(K)
      CALL SHAPE(S,T,SH,ADERIV)
      CALL JACOBY2(K1,NNODE,Y,NODES,ADERIV,YDJAC,CARTD,IER5)
      IF(IER5.EQ.1)GO TO 40
      CST=W1*YDJAC
      TEMP(7)=TEMP(7)+CST*ASY1*CARTD(2,I)*CARTD(2,J)
      TEMP(10)=TEMP(10)-CST*ASY2*CARTD(2,I)*CARTD(2,J)
      TEMP(13)=TEMP(13)+CST*SSY*CARTD(2,I)*CARTD(2,J)
      TEMP(15)=TEMP(15)-CST*SSY*CARTD(2,I)*SH(J)
      TEMP(19)=TEMP(19)+CST*TRY*CARTD(2,I)*CARTD(2,J)
      TEMP(22)=TEMP(10)
      TEMP(23)=TEMP(23)-CST*SSY*SH(I)*CARTD(2,J)
      TEMP(25)=TEMP(25)+CST*(DSY*CARTD(2,I)*CARTD(2,J)+
     1 SSY*SH(I)*SH(J))
```

```
30      CONTINUE
        L=0
        DO 10 II=5*I-4,5*I
        DO 10 JJ=5*J-4,5*J
        L=L+1
        ESTFY2(II,JJ)=TEMP(L)
10      CONTINUE
40      RETURN
        END

        SUBROUTINE GSMBX1(LMX,E1,IE,NADF,NDEFEX,O1)
        DIMENSION LMX(100,15),E1(15,15),O1(60000)
        DO 10 I=1,NDEFEX
        LM1=LMX(IE,I)
        DO 10 J=I,NDEFEX
        LM2=LMX(IE,J)
        IF(LM2.EQ.0)GO TO 10
        IF(LM1.EQ.0)GO TO 10
        IF(LM1.LE.LM2)GO TO 20
        GO TO 30
20      IF(LM1.EQ.1)THEN
        K=LM1+LM2-1
        O1(K)=O1(K)+E1(I,J)
        ELSE IF (LM1.GT.1)THEN
        LS=0
        DO 25 II=1,LM1-1
25      LS=(NADF-II+1)+LS
        K=LS+(LM2-LM1+1)
        O1(K)=O1(K)+E1(I,J)
        END IF
        GO TO 10
30      LS=0
        DO 35 JJ=1,LM2-1
35      LS=(NADF-JJ+1)+LS
        K=LS+(LM1-LM2+1)
        O1(K)=O1(K)+E1(I,J)
10      CONTINUE
        RETURN
        END
```

```
      SUBROUTINE GSMBY1(LMY,E2,IE,NADF,NDEFEY,02)
      DIMENSION LMY(100,15),E2(15,15),02(60000)
      DO 10 I=1,NDEFEY
      LM1=LMY(IE,I)
      DO 10 J=I,NDEFEY
      LM2=LMY(IE,J)
      IF(LM2.EQ.0)GO TO 10
      IF(LM1.EQ.0)GO TO 10
      IF(LM1.LE.LM2)GO TO 20
      GO TO 30
20    IF(LM1.EQ.1)THEN
      K=LM1+LM2-1
      02(K)=02(K)+E2(I,J)
      ELSE IF (LM1.GT.1)THEN
      LS=0
      DO 25 II=1,LM1-1
25    LS=(NADF-II+1)+LS
      K=LS+(LM2-LM1+1)
      02(K)=02(K)+E2(I,J)
      END IF
      GO TO 10
30    LS=0
      DO 35 JJ=1,LM2-1
35    LS=(NADF-JJ+1)+LS
      K=LS+(LM1-LM2+1)
      02(K)=02(K)+E2(I,J)
10    CONTINUE
      RETURN
      END


      SUBROUTINE CAPE(NELEM,NNODE,NDOFN,NODES,LM,ID,NDEFE)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     GENERATES CONNECTIVITY ARRAY FOR PLATE ELEMENTS
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION LM(100,40),ID(5,100),NODES(100,8)
      DO 70 IE=1,NELEM
      IJ=0
      DO 70 I=1,NNODE
      DO 70 J=1,NDOFN
      IJ=IJ+1
      LM(IE,IJ)=ID(J,NODES(IE,I))
```

```
 70     CONTINUE
        RETURN
        END


        SUBROUTINE CASEX(NELEMX,NNODEX,NDOFNX,NODESX1,LMX,ID,NDEFEX)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C       GENERATES CONNECTIVITY ARRAY FOR X-STIFFENER ELEMENTS
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        DIMENSION LMX(100,15),ID(5,100),NODESX1(100,3)
        DO 70 IE=1,NELEMX
        IJ=0
        DO 70 I=1,NNODEX
        DO 70 J=1,NDOFNX
        IJ=IJ+1
        IF(J.EQ.1)THEN
        J1=1
        ELSE IF(J.EQ.2)THEN
        J1=2
        ELSE IF(J.EQ.3)THEN
        J1=3
        ELSE IF(J.EQ.4)THEN
        J1=4
        ELSE IF(J.EQ.5)THEN
        J1=5
        END IF
        LMX(IE,IJ)=ID(J1,NODESX1(IE,I))
 70     CONTINUE
        RETURN
        END


        SUBROUTINE CASEY(NELEMY,NNODEY,NDOFNY,NODESY1,LMY,ID,NDEFEY)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C       GENERATES CONNECTIVITY ARRAY FOR Y-STIFFENER ELEMENTS
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        DIMENSION LMY(100,15),ID(5,100),NODESY1(100,3)
        DO 70 IE=1,NELEMY
        IJ=0
        DO 70 I=1,NNODEY
        DO 70 J=1,NDOFNY
        IJ=IJ+1
        IF(J.EQ.1)THEN
```

```
      J1=1
      ELSE IF(J.EQ.2)THEN
      J1=2
      ELSE IF(J.EQ.3)THEN
      J1=3
      ELSE IF(J.EQ.4)THEN
      J1=4
      ELSE IF(J.EQ.5)THEN
      J1=5
      END IF
      LMY(IE,IJ)=ID(J1,NODESY1(IE,I))
 70   CONTINUE
      RETURN
      END


      SUBROUTINE SOLV(NADF,OSTF,P,XD,IER6)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     GAUSSIAN SOLUTION SUBROUTINE
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION IDIG(400),OSTF(100000),P(400),XD(400)
      NN=NADF*(NADF+1)/2
      IDIG(1)=1
      DO 10 I=2,NADF
      DO 20 K=1,I-1
 20   IDIG(I)=IDIG(I)+(NADF-K+1)
      IDIG(I)=IDIG(I)+1
 10   CONTINUE
      DO 5 K=1,NADF
      IF(OSTF(IDIG(K)).LE.0.0)GO TO 80
 5    CONTINUE
C!!!!!!FORWARD ELIMINATION!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DO 30 I=1,NADF-1
      L=0
      DO 35 J=I,NADF-1
      L=L+1
      DO 40 K=J,NADF-1
 40   OSTF(IDIG(J+1)+K-J)=OSTF(IDIG(J+1)+K-J)-
     1 OSTF(IDIG(I)+L)*OSTF(IDIG(I)+K-J+L)/OSTF(IDIG(I))
      P(I+L)=P(I+L)-P(I)*OSTF(IDIG(I)+L)/OSTF(IDIG(I))
 35   CONTINUE
 30   CONTINUE
C!!!!!!BACK SUBSTITUTION!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
      MI=0
      LI=0
      DO 50 I=1,NADF
      LI=LI+1
      IF(I.GT.1)GO TO 60
      XD(NADF)=P(NADF)/OSTF(IDIG(NADF))
      GO TO 50
60    SUM=0
      MI=MI+1
      DO 70 K=1,LI-1
70    SUM=SUM+OSTF(IDIG(NADF-I+1)+K)*XD(NADF-MI+K)
      XD(NADF-I+1)=(P(NADF-I+1)-SUM)/OSTF(IDIG(NADF-I+1))
50    CONTINUE
      GO TO 90
80    IER6=1
90    RETURN
      END


      SUBROUTINE STRESS(POS,TH,AE,G,D,SR,NELEM,NNODE,X,Y,NODES,DISP,
     1 NODET,IFR,NST)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     CALCULATES STRESSES AT ELEMENT NODES BY BILINEAR EXTRAPOLATION
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION GSPX(4),GSPY(4),WW(4),CARTD(2,8),ADERIV(2,8),
     1 SH(8),X1(4),X2(4),X3(4),X4(4),X5(4),X6(4),X7(4),X8(4),
     1 NODES(100,8),DISP(600),XNX(4),XNY(4),XNXY(4),XMX(4),
     1 XMY(4),XMXY(4),XQXZ(4),XQYZ(4),ELNX(80,100),ELNY(80,100),
     1 ELNXY(80,100),ELMX(80,100),ELMY(80,100),ELMXY(80,100),
     1 ELQXZ(80,100),ELQYZ(80,100),X(100),Y(100),W(8,4),N(8)
      DIMENSION IFR(100),NST(100,10),S1(100),S2(100),S3(100),S4(100),
     1 S5(100),S6(100),S7(100),S8(100)
      CALL GAUSQ2(GSPX,GSPY,WW)
      DO 5 K=1,NODET
      DO 5 IE=1,NELEM
      S1(K)=0.
      S2(K)=0.
      S3(K)=0.
      S4(K)=0.
      S5(K)=0.
      S6(K)=0.
      S7(K)=0.
      S8(K)=0.
```

```
      ELNX(IE,K)=0.
      ELNY(IE,K)=0.
      ELNXY(IE,K)=0.
      ELMX(IE,K)=0.
      ELMY(IE,K)=0.
      ELMXY(IE,K)=0.
      ELQXZ(IE,K)=0.
      ELQYZ(IE,K)=0.
5     CONTINUE
      DO 40 IE=1,NELEM
      DO 10 IG=1,4
      X1(IG)=0.
      X2(IG)=0.
      X3(IG)=0.
      X4(IG)=0.
      X5(IG)=0.
      X6(IG)=0.
      X7(IG)=0.
      X8(IG)=0.
      S=GSPX(IG)
      T=GSPY(IG)
      CALL SHAPE(S,T,SH,ADERIV)
      CALL JACOB(IE,X,Y,NODES,ADERIV,XDJAC,CARTD,IER1)
      DO 20 I=1,NNODE
      X1(IG)=X1(IG)+CARTD(1,I)*DISP(5*NODES(IE,I)-4)
      X2(IG)=X2(IG)+CARTD(2,I)*DISP(5*NODES(IE,I)-3)
      X3(IG)=X3(IG)-(CARTD(2,I)*DISP(5*NODES(IE,I)-4)
     1 +CARTD(1,I)*DISP(5*NODES(IE,I)-3))
      X4(IG)=X4(IG)-CARTD(1,I)*DISP(5*NODES(IE,I)-1)
      X5(IG)=X5(IG)-CARTD(2,I)*DISP(5*NODES(IE,I))
      X6(IG)=X6(IG)+(CARTD(2,I)*DISP(5*NODES(IE,I)-1)
     1 +CARTD(1,I)*DISP(5*NODES(IE,I)))
      X7(IG)=X7(IG)-(CARTD(1,I)*DISP(5*NODES(IE,I)-2)
     1 -SH(I)*DISP(5*NODES(IE,I)-1))
      X8(IG)=X8(IG)-(CARTD(2,I)*DISP(5*NODES(IE,I)-2)
     1 -SH(I)*DISP(5*NODES(IE,I)))
20    CONTINUE
      XNX(IG)=AE*(X1(IG)+POS*X2(IG))
      XNY(IG)=AE*(POS*X1(IG)+X2(IG))
      XNXY(IG)=G*X3(IG)
      XMX(IG)=D*(X4(IG)+POS*X5(IG))
      XMY(IG)=D*(POS*X4(IG)+X5(IG))
      XMXY(IG)=0.5*(1.-POS)*D*X6(IG)
      XQXZ(IG)=SR*X7(IG)
```

```
      XQYZ(IG)=SR*X8(IG)
10    CONTINUE
      DO 15 I=1,NNODE
15    N(I)=NODES(IE,I)
      P=0.5773502692
      Q=0.5773502692
      W1=0.25*(1.+1./P)*(1.+1./Q)
      W2=0.25*(1.-1./P)*(1.+1./Q)
      W3=0.25*(1.-1./P)*(1.-1./Q)
      W4=0.25*(1.+1./P)
      W5=0.25*(1.-1./P)
      W(1,1)=W1
      W(2,1)=W4
      W(3,1)=W2
      W(4,1)=W5
      W(5,1)=W3
      W(6,1)=W5
      W(7,1)=W2
      W(8,1)=W4
      W(1,2)=W2
      W(2,2)=W4
      W(3,2)=W1
      W(4,2)=W4
      W(5,2)=W2
      W(6,2)=W5
      W(7,2)=W3
      W(8,2)=W5
      W(1,3)=W3
      W(2,3)=W5
      W(3,3)=W2
      W(4,3)=W4
      W(5,3)=W1
      W(6,3)=W4
      W(7,3)=W2
      W(8,3)=W5
      W(1,4)=W2
      W(2,4)=W5
      W(3,4)=W3
      W(4,4)=W5
      W(5,4)=W2
      W(6,4)=W4
      W(7,4)=W1
      W(8,4)=W4
      DO 60 J=1,NNODE
```

```
      DO 60 K=1,4
      ELNX(IE,N(J))=W(J,K)*XNX(K)+ELNX(IE,N(J))
      ELNY(IE,N(J))=W(J,K)*XNY(K)+ELNY(IE,N(J))
      ELNXY(IE,N(J))=W(J,K)*XNXY(K)+ELNXY(IE,N(J))
      ELMX(IE,N(J))=W(J,K)*XMX(K)+ELMX(IE,N(J))
      ELMY(IE,N(J))=W(J,K)*XMY(K)+ELMY(IE,N(J))
      ELMXY(IE,N(J))=W(J,K)*XMXY(K)+ELMXY(IE,N(J))
      ELQXZ(IE,N(J))=W(J,K)*XQXZ(K)+ELQXZ(IE,N(J))
      ELQYZ(IE,N(J))=W(J,K)*XQYZ(K)+ELQYZ(IE,N(J))
60    CONTINUE
40    CONTINUE
      DO 110 K=1,NODET
      DO 110 I=1,IFR(K)
      S1(K)=S1(K)+ELNX(NST(K,I),K)
      S2(K)=S2(K)+ELNY(NST(K,I),K)
      S3(K)=S3(K)+ELNXY(NST(K,I),K)
      S4(K)=S4(K)+ELMX(NST(K,I),K)
      S5(K)=S5(K)+ELMY(NST(K,I),K)
      S6(K)=S6(K)+ELMXY(NST(K,I),K)
      S7(K)=S7(K)+ELQXZ(NST(K,I),K)
      S8(K)=S8(K)+ELQYZ(NST(K,I),K)
110   CONTINUE
      DO 120 K=1,NODET
      S1(K)=S1(K)/REAL(IFR(K))
      S2(K)=S2(K)/REAL(IFR(K))
      S3(K)=S3(K)/REAL(IFR(K))
      S4(K)=S4(K)/REAL(IFR(K))
      S5(K)=S5(K)/REAL(IFR(K))
      S6(K)=S6(K)/REAL(IFR(K))
      S7(K)=S7(K)/REAL(IFR(K))
      S8(K)=S8(K)/REAL(IFR(K))
120   CONTINUE
      WRITE(6,130)
130   FORMAT(3X,8HNODE NO.,7X,3HNXX,10X,3HNYY,10X,3HNXY,10X,3HMXX,
     1 10X,3HMYY,10X,3HMXY,10X,3HQXZ,10X,3HQYZ)
      DO 140 K=1,NODET
      WRITE(6,150)K,S1(K),S2(K),S3(K),S4(K),S5(K),S6(K),S7(K),S8(K)
150   FORMAT(5X,I3,4X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4,2X,
     1 E11.4,2X,E11.4,2X,E11.4,2X,E11.4)
140   CONTINUE
      WRITE(6,*)'NORMAL PLATE STRESSES AT TOP AND BOTTOM FIBRES'
      WRITE(6,160)
160   FORMAT(3X,8HNODE NO.,3X,11HSIGMAX(TOP),3X,11HSIGMAX(BOT),3X,
     1 11HSIGMAY(TOP),3X,11HSIGMAY(BOT))
```

```
      DO 170 K=1,NODET
      SIG1=S1(K)/TH-6.*S4(K)/(TH**2)
      SIG2=S1(K)/TH+6.*S4(K)/(TH**2)
      SIG3=S2(K)/TH-6.*S5(K)/(TH**2)
      SIG4=S2(K)/TH+6.*S5(K)/(TH**2)
      WRITE(6,180)K,SIG1,SIG2,SIG3,SIG4
180   FORMAT(5X,I3,6X,E11.4,3X,E11.4,3X,E11.4,3X,E11.4)
170   CONTINUE
      RETURN
      END


      SUBROUTINE GCLVA(NELEM,NODES,QZ,P,X,Y,LM)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     CALCULATES CONSISTENT LOAD VECTOR BY 2x2 GAUSSIAN INTEGRATION
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION NODES(100,8),QZ(100),P(400),X(100),Y(100),
     1 LM(100,40),QC(8),GSPX(4),GSPY(4),W1(4),SH(8),ADERIV(2,8),
     2 CARTD(2,8)
      CALL GAUSQ2(GSPX,GSPY,W1)
      DO 10 IE=1,NELEM
      DO 20 I=1,8
20    QC(I)=0.0
      DO 30 K=1,4
      S=GSPX(K)
      T=GSPY(K)
      CALL SHAPE(S,T,SH,ADERIV)
      CALL JACOB(IE,X,Y,NODES,ADERIV,XDJAC,CARTD,IER1)
      CST=W1(K)*XDJAC
      DO 40 I=1,8
40    QC(I)=QC(I)+SH(I)*QZ(IE)*CST
30    CONTINUE
      IF(LM(IE,3).EQ.0)THEN
      GO TO 50
      ELSE
      P(LM(IE,3))=P(LM(IE,3))+QC(1)
      END IF
50    IF(LM(IE,8).EQ.0)THEN
      GO TO 60
      ELSE
      P(LM(IE,8))=P(LM(IE,8))+QC(2)
      END IF
60    IF(LM(IE,13).EQ.0)THEN
```

```
      GO TO 70
      ELSE
      P(LM(IE,13))=P(LM(IE,13))+QC(3)
      END IF
70    IF(LM(IE,18).EQ.0)THEN
      GO TO 80
      ELSE
      P(LM(IE,18))=P(LM(IE,18))+QC(4)
      END IF
80    IF(LM(IE,23).EQ.0)THEN
      GO TO 90
      ELSE
      P(LM(IE,23))=P(LM(IE,23))+QC(5)
      END IF
90    IF(LM(IE,28).EQ.0)THEN
      GO TO 100
      ELSE
      P(LM(IE,28))=P(LM(IE,28))+QC(6)
      END IF
100   IF(LM(IE,33).EQ.0)THEN
      GO TO 110
      ELSE
      P(LM(IE,33))=P(LM(IE,33))+QC(7)
      END IF
110   IF(LM(IE,38).EQ.0)THEN
      GO TO 10
      ELSE
      P(LM(IE,38))=P(LM(IE,38))+QC(8)
      END IF
10    CONTINUE
      RETURN
      END


      SUBROUTINE STRESSX1(NELEMX,NODET,NODESX1,NNODEX,X,DISP,IFRX,
     1 NSTX)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     CALCULATES X-STIFFENER STRESSES IN FEM(M1) BY LINEAR
C     EXTRAPOLATION
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION NODESX1(100,3),X(100),SH(3),DERIV(3),CARTD(3),GSPX(2),
     1 GSPY(2),NSTX(100,5),IFRX(100),X1(2),X2(2),X3(2),X4(2),DISP(600)
     2 ,XNSX(2),XMSX(2),XTSX(2),XQSXZ(2),ELNSX(80,100),ELMSX(80,100),
```

```
     3 ELTSX(80,100),ELQSXZ(80,100),S1(100),S2(100),S3(100),S4(100)
       COMMON/ELPROP/YNG(3),POS,TH
       COMMON/ELPROPX/IDELX1,IDELX2,BRX(2),DPX(2),BFX(2),TFX(2),
     1 CWX(2),CFX(2)
       COMMON CSX,XI,GX
       CALL GAUSQ21(GSPX,GSPY,W)
       DO 10 IEX=1,NELEMX
       N1=NODESX1(IEX,1)
       N2=NODESX1(IEX,2)
       N3=NODESX1(IEX,3)
       IF(IDELX1.EQ.0)THEN
       BRXX=BRX(1)
       DPXX=DPX(1)
       BFXX=BFX(1)
       TFXX=TFX(1)
       CWXX=CWX(1)
       CFXX=CFX(1)
       GO TO 5
       ELSE IF(IEX.GE.IDELX1.AND.IEX.LE.IDELX2)THEN
       BRXX=BRX(2)
       DPXX=DPX(2)
       BFXX=BFX(2)
       TFXX=TFX(2)
       CWXX=CWX(2)
       CFXX=CFX(2)
       GO TO 5
       ELSE
       BRXX=BRX(1)
       DPXX=DPX(1)
       BFXX=BFX(1)
       TFXX=TFX(1)
       CWXX=CWX(1)
       CFXX=CFX(1)
       END IF
 5     CALL CONS11(BRXX,DPXX,BFXX,TFXX,ASX,DSX,SSX,EX,
     1 CWXX,CFXX,TRX)
       DO 20 IG=1,2
       X1(IG)=0.
       X2(IG)=0.
       X3(IG)=0.
       X4(IG)=0.
 20    CONTINUE
       DO 30 IG=1,2
       S=GSPX(IG)
```

```
      CALL SHAPE1(S,SH,DERIV)
      CALL JACOBX1(IEX,X,NODESX1,DERIV,DJAC,CARTD,IER2)
      DO 40 I=1,NNODEX
      X1(IG)=X1(IG)+CARTD(I)*(DISP(5*NODESX1(IEX,I)-4)-
     1 EX*DISP(5*NODESX1(IEX,I)-1))
      X2(IG)=X2(IG)-CARTD(I)*DISP(5*NODESX1(IEX,I)-1)
      X3(IG)=X3(IG)+CARTD(I)*DISP(5*NODESX1(IEX,I))
      X4(IG)=X4(IG)+(-CARTD(I)*DISP(5*NODESX1(IEX,I)-2)+SH(I)*
     1 DISP(5*NODESX1(IEX,I)-1))
40    CONTINUE
      XNSX(IG)=ASX*X1(IG)
      XMSX(IG)=DSX*X2(IG)
      XTSX(IG)=TRX*X3(IG)
      XQSXZ(IG)=SSX*X4(IG)
30    CONTINUE
      P=0.5773502692
      W1=0.5*(1.+1./P)
      W2=0.5*(1.-1./P)
      ELNSX(IEX,N1)=W1*XNSX(1)+W2*XNSX(2)
      ELNSX(IEX,N3)=W2*XNSX(1)+W1*XNSX(2)
      ELNSX(IEX,N2)=0.5*XNSX(1)+0.5*XNSX(2)
      ELMSX(IEX,N1)=W1*XMSX(1)+W2*XMSX(2)
      ELMSX(IEX,N3)=W2*XMSX(1)+W1*XMSX(2)
      ELMSX(IEX,N2)=0.5*XMSX(1)+0.5*XMSX(2)
      ELTSX(IEX,N1)=W1*XTSX(1)+W2*XTSX(2)
      ELTSX(IEX,N3)=W2*XTSX(1)+W1*XTSX(2)
      ELTSX(IEX,N2)=0.5*XTSX(1)+0.5*XTSX(2)
      ELQSXZ(IEX,N1)=W1*XQSXZ(1)+W2*XQSXZ(2)
      ELQSXZ(IEX,N3)=W2*XQSXZ(1)+W1*XQSXZ(2)
      ELQSXZ(IEX,N2)=0.5*XQSXZ(1)+0.5*XQSXZ(2)
10    CONTINUE
      DO 70 K=1,NODET
      IF(IFRX(K).EQ.0)GO TO 70
      DO 70 I=1,IFRX(K)
      S1(K)=S1(K)+ELNSX(NSTX(K,I),K)
      S2(K)=S2(K)+ELMSX(NSTX(K,I),K)
      S3(K)=S3(K)+ELTSX(NSTX(K,I),K)
      S4(K)=S4(K)+ELQSXZ(NSTX(K,I),K)
70    CONTINUE
      WRITE(6,*)'NORMAL X-STIFFENER STRESSES AT TOP & BOTTOM FIBRES'
      WRITE(6,*)
      WRITE(6,130)
130   FORMAT(3X,8HNODE NO.,3X,10HSIGSX(TOP),3X,10HSIGSX(BOT))
      DO 75 K=1,NODET
```

```
      IF(IFRX(K).EQ.0)GO TO 75
      S1(K)=S1(K)/REAL(IFRX(K))
      S2(K)=S2(K)/REAL(IFRX(K))
      S3(K)=S3(K)/REAL(IFRX(K))
      S4(K)=S4(K)/REAL(IFRX(K))
      SIG1=S1(K)/CSX-(DPXX+TFXX-GX)*S2(K)/XI
      SIG2=S1(K)/CSX+GX*S2(K)/XI
      WRITE(6,120)K,SIG1,SIG2
120   FORMAT(4X,I3,6X,E11.4,3X,E11.4)
75    CONTINUE
      RETURN
      END


      SUBROUTINE STRESSY1(NELEMY,NODET,NODESY1,NNODEY,Y,DISP,IFRY,
     1 NSTY)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     CALCULATES Y-STIFFENER STRESSES IN FEM(M1)BY LINEAR
C     EXTRAPOLATION
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION NODESY1(100,3),Y(100),SH(3),DERIV(3),CARTD(3),GSPX(2),
     1 GSPY(2),NSTY(100,5),IFRY(100),Y1(2),Y2(2),Y3(2),Y4(2),DISP(600)
     2 ,YNSY(2),YMSY(2),YTSY(2),YQSYZ(2),ELNSY(80,100),ELMSY(80,100),
     3 ELTSY(80,100),ELQSYZ(80,100),S1(100),S2(100),S3(100),S4(100)
      COMMON CSY,YI,GY
      COMMON/ELPROP/YNG(3),POS,TH
      COMMON/ELPROPY/IDELY1,IDELY2,BRY(2),DPY(2),BFY(2),TFY(2),
     1 CWY(2),CFY(2)
      CALL GAUSQ21(GSPY,GSPY,W)
      DO 10 IEY=1,NELEMY
      N1=NODESY1(IEY,1)
      N2=NODESY1(IEY,2)
      N3=NODESY1(IEY,3)
      IF(IDELY1.EQ.0)THEN
      BRYY=BRY(1)
      DPYY=DPY(1)
      BFYY=BFY(1)
      TFYY=TFY(1)
      CWYY=CWY(1)
      CFYY=CFY(1)
      GO TO 5
      ELSE IF(IEY.GE.IDELY1.AND.IEY.LE.IDELY2)THEN
```

```
      BRYY=BRY(2)
      DPYY=DPY(2)
      BFYY=BFY(2)
      TFYY=TFY(2)
      CWYY=CWY(2)
      CFYY=CFY(2)
      GO TO 5
      ELSE
      BRYY=BRY(1)
      DPYY=DPY(1)
      BFYY=BFY(1)
      TFYY=TFY(1)
      CWYY=CWY(1)
      CFYY=CFY(1)
      END IF
5     CALL CONS21(BRYY,DPYY,BFYY,TFYY,ASY,DSY,SSY,EY,
     1 CWYY,CFYY,TRY)
      DO 20 IG=1,2
      Y1(IG)=0.
      Y2(IG)=0.
      Y3(IG)=0.
      Y4(IG)=0.
20    CONTINUE
      DO 30 IG=1,2
      S=GSPY(IG)
      CALL SHAPE1(S,SH,DERIV)
      CALL JACOBY1(IEY,Y,NODESY1,DERIV,DJAC,CARTD,IEY3)
      DO 40 I=1,NNODEY
      Y1(IG)=Y1(IG)+CARTD(I)*(DISP(5*NODESY1(IEY,I)-3)-
     1 EY*DISP(5*NODESY1(IEY,I)))
      Y2(IG)=Y2(IG)-CARTD(I)*DISP(5*NODESY1(IEY,I))
      Y3(IG)=Y3(IG)+CARTD(I)*DISP(5*NODESY1(IEY,I)-1)
      Y4(IG)=Y4(IG)+(-CARTD(I)*DISP(5*NODESY1(IEY,I)-2)+SH(I)*
     1 DISP(5*NODESY1(IEY,I)))
40    CONTINUE
      YNSY(IG)=ASY*Y1(IG)
      YMSY(IG)=DSY*Y2(IG)
      YTSY(IG)=TRY*Y3(IG)
      YQSYZ(IG)=SSY*Y4(IG)
30    CONTINUE
      P=0.5773502692
      W1=0.5*(1.+1./P)
      W2=0.5*(1.-1./P)
      ELNSY(IEY,N1)=W1*YNSY(1)+W2*YNSY(2)
```

```
      ELNSY(IEY,N3)=W2*YNSY(1)+W1*YNSY(2)
      ELNSY(IEY,N2)=0.5*YNSY(1)+0.5*YNSY(2)
      ELMSY(IEY,N1)=W1*YMSY(1)+W2*YMSY(2)
      ELMSY(IEY,N3)=W2*YMSY(1)+W1*YMSY(2)
      ELMSY(IEY,N2)=0.5*YMSY(1)+0.5*YMSY(2)
      ELTSY(IEY,N1)=W1*YTSY(1)+W2*YTSY(2)
      ELTSY(IEY,N3)=W2*YTSY(1)+W1*YTSY(2)
      ELTSY(IEY,N2)=0.5*YTSY(1)+0.5*YTSY(2)
      ELQSYZ(IEY,N1)=W1*YQSYZ(1)+W2*YQSYZ(2)
      ELQSYZ(IEY,N3)=W2*YQSYZ(1)+W1*YQSYZ(2)
      ELQSYZ(IEY,N2)=0.5*YQSYZ(1)+0.5*YQSYZ(2)
 10   CONTINUE
      DO 70 K=1,NODET
      IF(IFRY(K).EQ.0)GO.TO 70
      DO 70 I=1,IFRY(K)
      S1(K)=S1(K)+ELNSY(NSTY(K,I),K)
      S2(K)=S2(K)+ELMSY(NSTY(K,I),K)
      S3(K)=S3(K)+ELTSY(NSTY(K,I),K)
      S4(K)=S4(K)+ELQSYZ(NSTY(K,I),K)
 70   CONTINUE
      WRITE(6,*)'NORMAL Y-STIFFENER STRESSES AT TOP & BOTTOM FIBRES'
      WRITE(6,*)
      WRITE(6,130)
 130  FORMAT(3X,8HNODE NO.,3X,10HSIGSY(TOP),3X,10HSIGSY(BOT))
      DO 75 K=1,NODET
      IF(IFRY(K).EQ.0)GO TO 75
      S1(K)=S1(K)/REAL(IFRY(K))
      S2(K)=S2(K)/REAL(IFRY(K))
      S3(K)=S3(K)/REAL(IFRY(K))
      S4(K)=S4(K)/REAL(IFRY(K))
      SIG1=S1(K)/CSY-(DPYY+TFYY-GY)*S2(K)/YI
      SIG2=S1(K)/CSY+GY*S2(K)/YI
      WRITE(6,120)K,SIG1,SIG2
 120  FORMAT(4X,I3,6X,E11.4,3X,E11.4)
 75   CONTINUE
      RETURN
      END


      SUBROUTINE STRESSX2(NELEMX,NNODE,NODETX,NODESX2,X,Y,YS,DISP,
     1 NODES,IFRX,IPX,NSTX)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C       CALCULATES X-STIFFENER STRESSES IN FEM(M2) BY LINEAR
```

```
C      EXTRAPOLATION
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
       DIMENSION NODESX2(100,2),X(100),SH(8),ADERIV(2,8),CARTD(2,8),
     1 GSPX(2),GSPY(2),NSTX(100,5),IFRX(100),X1(2),X2(2),DISP(600),
     2 SIGX(2),SIGXN(50,70),SIGMA(100),NODES(100,8),YS(60),IPX(50),
     3 Y(100)
       COMMON/ELPROP/YNG(3),POS,TH
       COMMON/ELPROPX/IDELX1,IDELX2,BRX(2),DPX(2),BFX(2),TFX(2),
     1 CWX(2),CFX(2)
       CALL GAUSQ21(GSPX,GSPY,W)
       DO 10 IEX=1,NELEMX
       N1=NODESX2(IEX,1)
       N2=NODESX2(IEX,2)
       IF(IDELX1.EQ.0)THEN
       DPXX=DPX(1)
       TFXX=TFX(1)
       GO TO 5
       ELSE IF(IEX.GE.IDELX1.AND.IEX.LE.IDELX2)THEN
       DPXX=DPX(2)
       TFXX=TFX(2)
       GO TO 5
       ELSE
       DPXX=DPX(1)
       TFXX=TFX(1)
       END IF
 5     P=0.5*TH+DPXX+TFXX
       K1=IPX(IEX)
       A=0.5*(Y(NODES(K1,2))-Y(NODES(K1,6)))
       B=YS(IEX)-0.5*(Y(NODES(K1,2))+Y(NODES(K1,6)))
       EETA=B/A
       DO 20 IG=1,2
       X1(IG)=0.0
       X2(IG)=0.0
       DO 20 IN=1,NNODE
       S=GSPX(IG)
       T=EETA
       CALL SHAPE(S,T,SH,ADERIV)
       CALL JACOBX2(K1,NNODE,X,NODES,ADERIV,XDJAC,CARTD,IER4)
       X1(IG)=X1(IG)+CARTD(1,IN)*DISP(5*NODES(K1,IN)-4)
       X2(IG)=X2(IG)+CARTD(1,IN)*DISP(5*NODES(K1,IN)-1)
       SIGX(IG)=YNG(2)*(X1(IG)-P*X2(IG))
 20    CONTINUE
       XG=0.5773502692
       W1=0.5*(1.+1./XG)
```

```
      W2=0.5*(1.-1./XG)
      SIGXN(IEX,N1)=W1*SIGX(1)+W2*SIGX(2)
      SIGXN(IEX,N2)=W2*SIGX(1)+W1*SIGX(2)
10    CONTINUE
      DO 30 K=1,NODETX
      DO 30 I=1,IFRX(K)
      SIGMA(K)=SIGMA(K)+SIGXN(NSTX(K,I),K)
30    CONTINUE
      DO 40 I=1,NODETX
40    SIGMA(I)=SIGMA(I)/REAL(IFRX(I))
      WRITE(6,50)
50    FORMAT(3X,27HPSEUDO X-STIFFENER NODE NO.,4X,34HNORMAL STRESS
     1 AT BOTTOM-MOST FIBRE)
      DO 60 K=1,NODETX
      WRITE(6,70)K,SIGMA(K)
70    FORMAT(13X,I3,30X,E11.4)
60    CONTINUE
      RETURN
      END


      SUBROUTINE STRESSY2(NELEMY,NNODE,NODETY,NODESY2,X,Y,XS,DISP,
     1 NODES,IFRY,IPY,NSTY)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     CALCULATES Y-STIFFENER STRESSES IN FEM(M2) BY LINEAR
C     EXTRAPOLATION
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION NODESY2(100,2),X(100),SH(8),ADERIV(2,8),CARTD(2,8),
     1 GSPX(2),GSPY(2),NSTY(100,5),IFRY(100),X1(2),X2(2),DISP(600),
     2 SIGY(2),SIGYN(50,70),SIGMA(100),NODES(100,8),XS(60),IPY(50),
     3 Y(100)
      COMMON/ELPROP/YNG(3),POS,TH
      COMMON/ELPROPY/IDELY1,IDELY2,BRY(2),DPY(2),BFY(2),TFY(2),
     1 CWY(2),CFY(2)
      CALL GAUSQ21(GSPX,GSPY,W)
      DO 10 IEY=1,NELEMY
      N1=NODESY2(IEY,1)
      N2=NODESY2(IEY,2)
      IF(IDELY1.EQ.0)THEN
      DPYY=DPY(1)
      TFYY=TFY(1)
      GO TO 5
      ELSE IF(IEY.GE.IDELY1.AND.IEY.LE.IDELY2)THEN
```

```
      DPYY=DPY(2)
      TFYY=TFY(2)
      GO TO 5
      ELSE
      DPYY=DPY(1)
      TFYY=TFY(1)
      END IF
5     P=0.5*TH+DPYY+TFYY
      K1=IPY(IEY)
      A=0.5*(X(NODES(K1,4))-X(NODES(K1,8)))
      B=XS(IEY)-0.5*(X(NODES(K1,4))+X(NODES(K1,8)))
      XI=B/A
      DO 20 IG=1,2
      X1(IG)=0.0
      X2(IG)=0.0
      DO 20 IN=1,NNODE
      S=XI
      T=GSPY(IG)
      CALL SHAPE(S,T,SH,ADERIV)
      CALL JACOBY2(K1,NNODE,Y,NODES,ADERIV,YDJAC,CARTD,IER5)
      X1(IG)=X1(IG)+CARTD(2,IN)*DISP(5*NODES(K1,IN)-3)
      X2(IG)=X2(IG)+CARTD(2,IN)*DISP(5*NODES(K1,IN))
      SIGY(IG)=YNG(3)*(X1(IG)-P*X2(IG))
20    CONTINUE
      YG=0.5773502692
      W1=0.5*(1.+1./YG)
      W2=0.5*(1.-1./YG)
      SIGYN(IEY,N1)=W1*SIGY(1)+W2*SIGY(2)
      SIGYN(IEY,N2)=W2*SIGY(1)+W1*SIGY(2)
10    CONTINUE
      DO 30 K=1,NODETY
      DO 30 I=1,IFRY(K)
      SIGMA(K)=SIGMA(K)+SIGYN(NSTY(K,I),K)
30    CONTINUE
      DO 40 I=1,NODETY
40    SIGMA(I)=SIGMA(I)/REAL(IFRY(I))
      WRITE(6,50)
50    FORMAT(3X,27HPSEUDO Y-STIFFENER NODE NO.,4X,34HNORMAL STRESS
     1 AT BOTTOM-MOST FIBRE)
      DO 60 K=1,NODETY
      WRITE(6,70)K,SIGMA(K)
70    FORMAT(13X,I3,30X,E11.4)
60    CONTINUE
      RETURN
```

```
      END


      SUBROUTINE DOCTOR(X,Y,NODES,NODESX1,NODESX2,NODESY1,NODESY2,
     1 IPX,YS,IPY,XS,INBN,ID1,NNCL,NSTOR,P1,QZ)
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C     PRINTS INPUT DATA AND DIAGNOSES ERRORS
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      DIMENSION X(100),Y(100),NODES(100,8),NODESX1(100,3),NODESY1(100,
     1 3),NODESX2(100,2),NODESY2(100,2),IPX(50),YS(60),IPY(50),XS(60),
     2 INBN(70),ID1(5,100),NSTOR(15),P1(10),QZ(100)
      COMMON NODET,NELEM,NNODE,NDOFN,NBN,LTYPE,IANT
      COMMON IETPX,IETPY
      COMMON NELEMX,NNODEX,NDOFNX
      COMMON NELEMY,NNODEY,NDOFNY
      COMMON NODETX,NODETY
      COMMON/ELPROP/YNG(3),POS,TH
      COMMON/ELPROPX/IDELX1,IDELX2,BRX(2),DPX(2),BFX(2),TFX(2),
     1 CWX(2),CFX(2)
      COMMON/ELPROPY/IDELY1,IDELY2,BRY(2),DPY(2),BFY(2),TFY(2),
     1 CWY(2),CFY(2)
      COMMON IER1,IER2,IER3,IER4,IER5,IER6
      IF(IANT.EQ.1)THEN
      WRITE(2,*)'THIS CHECK FILE IS FOR FORMULATION FEM(M1)'
      ELSE
      WRITE(2,*)'THIS CHECK FILE IS FOR FORMULATION FEM(M2)'
      END IF
      WRITE(2,*)
      WRITE(2,10)NODET,NELEM,NNODE,NDOFN,NBN,LTYPE,IANT
10    FORMAT(3X,8HNODET = ,I3,3X,8HNELEM = ,I3,3X,8HNNODE = ,I3,
     1 3X,8HNDOFN = ,I3,3X,6HNBN = ,I3,3X,8HLTYPE = ,I3,3X,
     2 7HIANT = ,I3)
      WRITE(2,*)
      WRITE(2,20)IETPX,IETPY
20    FORMAT(3X,8HIETPX = ,I1,3X,8HIETPY = ,I1)
      WRITE(2,*)
      WRITE(2,30)
30    FORMAT(3X,8HNODE NO.,5X,8HX-COORD.,4X,8HY-COORD.)
      DO 40 I=1,NODET
      WRITE(2,50)I,X(I),Y(I)
50    FORMAT(6X,I3,6X,F8.4,4X,F8.4)
40    CONTINUE
```

```
      WRITE(2,*)
      WRITE(2,60)
60    FORMAT(3X,17HPLATE ELEMENT NO.,15X,9HNODE NOS.)
      DO 70 IE=1,NELEM
70    WRITE(2,80)IE,(NODES(IE,I),I=1,NNODE)
80    FORMAT(10X,I3,7X,8(2X,I3))
      IF(IANT.EQ.1)THEN
      CONTINUE
      ELSE
      GO TO 90
      END IF
      WRITE(2,*)
      WRITE(2,100)NELEMX,NDOFNX,NNODEX
100   FORMAT(3X,9HNELEMX = ,I5,9HNDOFNX = ,I5,9HNNODEX = ,I5)
      IF(NELEMX.EQ.0)GO TO 110
      WRITE(2,*)
      WRITE(2,120)
120   FORMAT(3X,17HX-STIFFENER ELEM.,7X,9HNODE NOS.)
      DO 130 IEX=1,NELEMX
      WRITE(2,140)IEX,(NODESX1(IEX,I),I=1,NNODEX)
140   FORMAT(9X,I4,10X,2(I4,3X))
130   CONTINUE
110   WRITE(2,*)
      WRITE(2,150)NELEMY,NDOFNY,NNODEY
150   FORMAT(3X,9HNELEMY = ,I5,9HNDOFNY = ,I5,9HNNODEY = ,I5)
      IF(NELEMY.EQ.0)GO TO 160
      WRITE(2,*)
      WRITE(2,170)
170   FORMAT(3X,17HY-STIFFENER ELEM.,7X,9HNODE NOS.)
      DO 180 IEY=1,NELEMY
      WRITE(2,190)IEY,(NODESY1(IEY,I),I=1,NNODEY)
190   FORMAT(9X,I4,10X,2(I4,3X))
180   CONTINUE
160   GO TO 200
90    WRITE(2,*)
      WRITE(2,210)NODETX,NODETY
210   FORMAT(3X,9HNODETX = ,I3,3X,9HNODETY = ,I3)
      WRITE(2,*)
      WRITE(2,220)NELEMX,NELEMY
220   FORMAT(3X,9HNELEMX = ,I3,3X,9HNELEMY = ,I3)
      IF(NELEMX.EQ.0)GO TO 230
      WRITE(2,*)
      WRITE(2,240)
240   FORMAT(3X,17HX-STIFFENER ELEM.,6X,9HNODE NOS.)
```

```
      DO 250 IEX=1,NELEMX
      WRITE(2,260)IEX,(NODESX2(IEX,I),I=1,2)
260   FORMAT(9X,I4,10X,2(I4,3X))
250   CONTINUE
      WRITE(2,*)
      WRITE(2,270)
270   FORMAT(3X,3HIPX,6X,2HYS)
      DO 280 IEX=1,NELEMX
      WRITE(2,290)IPX(IEX),YS(IEX)
290   FORMAT(3X,I3,3X,F8.4)
280   CONTINUE
230   IF(NELEMY.EQ.0)GO TO 300
      WRITE(2,*)
      WRITE(2,310)
310   FORMAT(3X,17HY-STIFFENER ELEM.,6X,9HNODE NOS.)
      DO 320 IEY=1,NELEMY
      WRITE(2,330)IEY,(NODESY2(IEY,I),I=1,2)
330   FORMAT(9X,I4,10X,2(I4,3X))
320   CONTINUE
      WRITE(2,*)
      WRITE(2,340)
340   FORMAT(3X,3HIPY,6X,2HXS)
      DO 350 IEY=1,NELEMY
      WRITE(2,360)IPY(IEY),XS(IEY)
360   FORMAT(3X,I3,3X,F8.4)
350   CONTINUE
200   CONTINUE
300   CONTINUE
      WRITE(2,*)
      WRITE(2,370)
370   FORMAT(3X,13HBOUNDARY NODE,2X,13HID NOS. INPUT)
      DO 380 I=1,NBN
      WRITE(2,390)INBN(I),(ID1(K,INBN(I)),K=1,NDOFN)
390   FORMAT(8X,I3,5X,5(2X,I1))
380   CONTINUE
      WRITE(2,*)
      WRITE(2,400)YNG(1)
400   FORMAT(3X,24HPLATE YOUNG'S MODULUS = ,E12.4)
      WRITE(2,410)POS
410   FORMAT(3X,24HPLATE POISSON'S RATIO = ,F6.4)
      WRITE(2,420)TH
420   FORMAT(3X,18HPLATE THICKNESS = ,F8.4)
      IF(NELEMX.GT.0)THEN
      WRITE(2,*)
```

```
      WRITE(2,430)YNG(2)
430   FORMAT(3X,47HYOUNG'S MODULUS FOR THE X-STIFFENER MATERIAL = ,
     1 E12.4)
      IF(IETPX.EQ.1)THEN
      WRITE(2,*)'THE FOLLOWING ARE THE X-STIFFENER DETAILS :'
      ELSE
      WRITE(2,*)'THE FOLLOWING DETAILS ARE FOR THE FIRST TYPE OF'
      WRITE(2,*)'X-STIFFENERS :'
      END IF
      WRITE(2,440)BRX(1),DPX(1)
440   FORMAT(3X,12HWEB WIDTH = ,F10.5,3X,12HWEB DEPTH = ,F10.5)
      WRITE(2,450)BFX(1),TFX(1)
450   FORMAT(3X,15HFLANGE WIDTH = ,F10.5,3X,19HFLANGE THICKNESS = ,
     1 F10.5)
      WRITE(2,460)CWX(1),CFX(1)
460   FORMAT(3X,29HTORSIONAL CONSTANT FOR WEB = ,F7.4,3X,
     1 32HTORSIONAL CONSTANT FOR FLANGE = ,F7.4)
      IF(IETPX.EQ.1)GO TO 470
      WRITE(2,*)
      WRITE(2,480)IDELX1,IDELX2
480   FORMAT(3X,16HX-STIFFENER NO. ,I2,3HTO ,I2,16H ARE OF 2ND TYPE)
      WRITE(2,*)
      WRITE(2,*)'THE FOLLOWING DETAILS ARE FOR THE 2ND TYPE OF X-STI
     1 FFENERS'
      WRITE(2,490)BRX(2),DPX(2)
490   FORMAT(3X,12HWEB WIDTH = ,F10.5,3X,12HWEB DEPTH = ,F10.5)
      WRITE(2,500)BFX(2),TFX(2)
500   FORMAT(3X,15HFLANGE WIDTH = ,F10.5,3X,19HFLANGE THICKNESS = ,
     1 F10.5)
      WRITE(2,510)CWX(2),CFX(2)
510   FORMAT(3X,29HTORSIONAL CONSTANT FOR WEB = ,F7.4,3X,
     1 32HTORSIONAL CONSTANT FOR FLANGE = ,F7.4)
      END IF
470   IF(NELEMY.GT.0)THEN
      WRITE(2,*)
      WRITE(2,520)YNG(3)
520   FORMAT(3X,47HYOUNG'S MODULUS FOR THE Y-STIFFENER MATERIAL = ,
     1 E12.4)
      IF(IETPY.EQ.1)THEN
      WRITE(2,*)'THE FOLLOWING ARE THE Y-STIFFENER DETAILS :'
      ELSE
      WRITE(2,*)'THE FOLLOWING DETAILS ARE FOR THE FIRST TYPE OF'
      WRITE(2,*)'Y-STIFFENERS :'
      END IF
```

```
      WRITE(2,530)BRY(1),DPY(1)
530   FORMAT(3X,12HWEB WIDTH = ,F10.5,3X,12HWEB DEPTH = ,F10.5)
      WRITE(2,540)BFY(1),TFY(1)
540   FORMAT(3X,15HFLANGE WIDTH = ,F10.5,3X,19HFLANGE THICKNESS = ,
     1 F10.5)
      WRITE(2,550)CWY(1),CFY(1)
550   FORMAT(3X,29HTORSIONAL CONSTANT FOR WEB = ,F7.4,3X,
     1 32HTORSIONAL CONSTANT FOR FLANGE = ,F7.4)
      IF(IETPY.EQ.1)GO TO 560
      WRITE(2,*)
      WRITE(2,570)IDELY1,IDELY2
570   FORMAT(3X,16HY-STIFFENER NO. ,I2,3HTO ,I2,16H ARE OF 2ND TYPE)
      WRITE(2,*)
      WRITE(2,*)'THE FOLLOWING DETAILS ARE FOR THE 2ND TYPE OF Y-STI
     1 FFENERS :'
      WRITE(2,580)BRY(2),DPY(2)
580   FORMAT(3X,12HWEB WIDTH = ,F10.5,3X,12HWEB DEPTH = ,F10.5)
      WRITE(2,590)BFY(2),TFY(2)
590   FORMAT(3X,15HFLANGE WIDTH = ,F10.5,3X,19HFLANGE THICKNESS = ,
     1 F10.5)
      WRITE(2,600)CWY(2),CFY(2)
600   FORMAT(3X,29HTORSIONAL CONSTANT FOR WEB = ,F7.4,3X,
     1 32HTORSIONAL CONSTANT FOR FLANGE = ,F7.4)
      END IF
560   IF(LTYPE.EQ.1)THEN
      WRITE(2,*)
      WRITE(2,610)
610   FORMAT(3X,8HNODE NO.,3X,18HLATERAL POINT LOAD)
      DO 620 I=1,NNCL
      WRITE(2,630)NSTOR(I),P1(I)
630   FORMAT(5X,I3,10X,E12.4)
620   CONTINUE
      ELSE
      WRITE(2,*)
      WRITE(2,640)QZ(1)
640   FORMAT(3X,32HUNIFORMLY DISTRIBUTED LOADING = ,E12.4)
      END IF
      IF(IER1.EQ.1)THEN
      WRITE(2,*)
      WRITE(2,*)'DETERMINANT OF THE JACOBIAN PERTAINING TO'
      WRITE(2,*)'A PLATE ELEMENT ZERO OR NEGATIVE'
      WRITE(2,*)
      WRITE(2,*)'ERROR IN PLATE NODAL COORDINATES DATA SUSPECTED'
      WRITE(2,*)
```

```
      WRITE(2,*)'PROGRAM TERMINATED'
      GO TO 650
      END IF
      IF(IER2.OR.IER4.EQ.1)THEN
      WRITE(2,*)
      WRITE(2,*)'DETERMINANT OF THE JACOBIAN PERTAINIG TO''
      WRITE(2,*)'AN X-STIFFENER ELEMENT ZERO OR NEGATIVE'
      WRITE(2,*)
      WRITE(2,*)'ERROR IN X-STIFFENER LOCATION DATA SUSPECTED'
      WRITE(2,*)
      WRITE(2,*)'PROGRAM TERMINATED'
      GO TO 650
      END IF
      IF(IER3.OR.IER5.EQ.1)THEN
      WRITE(2,*)
      WRITE(2,*)'DETERMINANT OF THE JACOBIAN PERTAINIG TO'
      WRITE(2,*)'AN Y-STIFFENER ELEMENT ZERO OR NEGATIVE'
      WRITE(2,*)
      WRITE(2,*)'ERROR IN Y-STIFFENER LOCATION DATA SUSPECTED'
      WRITE(2,*)
      WRITE(2,*)'PROGRAM TERMINATED'
      GO TO 650
      END IF
      IF(IER6.EQ.1)THEN
      WRITE(2,*)
      WRITE(2,*)'GLOBAL STIFFNESS MATRIX NOT POSITIVE-DEFINITE'
      WRITE(2,*)
      WRITE(2,*)'CHECK DATA ON MATERIAL PROPERTIES AND SECTIONAL'
      WRITE(2,*)'DETAILS'
      WRITE(2,*)
      WRITE(2,*)'PROGRAM TERMINATED'
      GO TO 650
      END IF
650   STOP
      END
```

Appendix 2

Listing of the Non-linear Orthotropic Analysis Program NLORTHO

```
C*************************************************************************
C       THIS IS LISTING OF PROGRAM NLORTHO FOR THE NON-LINEAR
C       ORTHOTROPIC PLATE ANALYSIS
C*************************************************************************
        DIMENSION X(100),Y(100),NODES(100,8),
     1  ID(5,100),LM(100,40),
     2  QZ(100),ESTF(40,40),
     3  OSTF(100000),PG(100),P(600),XD(600),
     4  DISP(600),IFR(100),NST(100,10),OSTF1(100000)
        DIMENSION DELP(600),XI(600),XXI(600),XDI(600),CHI(5),CH3(5),
     1  DELXD(600),DELXNX(50,4),DELXNY(50,4),DELXNXY(50,4),
     2  DELXMX(50,4),DELXMY(50,4),DELXMXY(50,4),DELXQXZ(50,4),
     3  DELXQYZ(50,4),XD1(600),XII(600),DELXSIGPT(50,4),DELYSIGPT(50,4)
     4  ,DELXSIGSB(50,4),DELYSIGSB(50,4)
        COMMON/CONS/POS,C13,C1,C2,C15,C4,C1D,D12,D1,D13,SR1,SR2,SRP
        COMMON/STRES/XNX(50,4),XNY(50,4),XNXY(50,4),XMX(50,4),XMY(50,4),
     1  XMXY(50,4),XQXZ(50,4),XQYZ(50,4),XSIGPT(50,4),YSIGPT(50,4),
     3  XSIGSB(50,4),YSIGSB(50,4)
        OPEN(UNIT=5,FILE='NL.DAT',TYPE='OLD')
        OPEN(UNIT=2,FILE='NL.CHEK',TYPE='NEW')
        OPEN(UNIT=6,FILE='NL.SOL',TYPE='NEW')
C*************************************************************************
        READ(5,*)IANTP !!EQUAL TO UNITY FOR A LINEAR ANALYSIS!!
        IF(IANTP.EQ.1)GO TO 5
        READ(5,*)ISME !!NO. OF TIMES STIFFNESS MATRIX TO BE UPDATED!!
        READ(5,*)TOLER !!% TOLERANCE FOR CONVERGENCE!!
        READ(5,*)NSTEP !!NO. OF LOAD STEPS!!
 5      READ(5,*)NODET,NELEM,NNODE,NDOFN,NBN,LTYPE
        READ(5,*)NSOP !!NODAL STRESS OR DEFLECTION OPTION PARAMETER!!
C       IF NSOP IS NON-ZERO, THEN DEFLECTIONS AND STRESSES WILL BE
C       PRINTED ONLY FOR NODE NUMBER NSOP
        WRITE(2,*)'NODET,NELEM,NNODE,NDOFN,NBN,LTYPE'
        WRITE(2,1001)NODET,NELEM,NNODE,NDOFN,NBN,LTYPE
 1001   FORMAT(1X,7I5)
        READ(5,*)NSTX,SNX
        WRITE(2,*)'NSTX,SNX'
        WRITE(2,*)NSTX,SNX
        READ(5,*)NSTY,SNY
        WRITE(2,*)'NSTY,SNY'
        WRITE(2,*)NSTY,SNY
C*************************************************************************
        DO 10 NODE=1,NODET
        READ(5,*)X(NODE),Y(NODE)
        WRITE(2,2000)
 2000   FORMAT(5X,4HNODE,5X,7HX(NODE),5X,7HY(NODE))
```

```
      WRITE(2,1002)NODE,X(NODE),Y(NODE)
1002  FORMAT(5X,I5,5X,F10.5,5X,F10.5)
10    CONTINUE
C*****************************************************************
      WRITE(2,2001)
2001  FORMAT(3X,8H ELEMENT,12X,9HNODE NOS.)
      DO 20 IE=1,NELEM
      READ(5,*) (NODES(IE,I),I=1,NNODE)
      WRITE(2,1003) IE,(NODES(IE,I),I=1,NNODE)
1003  FORMAT(7X,I4,5X,4(I4,2X))
20    CONTINUE
C*****************************************************************
      NDEF=NODET*NDOFN
      NDEFE=NNODE*NDOFN
C******INITIALIZE ID ARRAY*****************************************
562   DO 30 J=1,NODET
      DO 30 I=1,NDOFN
30    ID(I,J)=0
      WRITE(2,2007)
2007  FORMAT(2X,16HCONSTRAINT NODES,3X,18HCONSTRAINT INDICES)
      DO 40 I=1,NBN
      READ(5,*)NODE,((ID(K,NODE)),K=1,NDOFN)
      WRITE(2,2008)NODE,((ID(K,NODE)),K=1,NDOFN)
2008  FORMAT(8X,I4,9X,5(I2,2X))
40    CONTINUE
C***********MODIFY ID ARRAY****************************************
      KOUNT=0
      DO 50 J=1,NODET
      DO 50 I=1,NDOFN
      IF(ID(I,J).EQ.1)GO TO 60
      KOUNT=KOUNT+1
      ID(I,J)=KOUNT
      NADF=ID(I,J)
      GO TO 50
60    ID(I,J)=0
50    CONTINUE
      NOSTF=NADF*(NADF+1)/2
      CALL CAPE(NELEM,NNODE,NDOFN,NODES,LM,ID,NDEFE)
C****READING ELEMENT PROPERTIES************************************
      READ(5,*)YNG,POS,TH1
      WRITE(2,2009)
2009  FORMAT(2X,15HYOUNG'S MODULUS,3X,15HPOISSON'S RATIO,
     1 3X,15HPLATE THICKNESS)
      WRITE(2,2010)YNG,POS,TH1
2010  FORMAT(5X,E12.4,6X,E12.4,6X,E12.4)
```

```
      WRITE(2,2011)
 2011 FORMAT(2X,21HWIDTH OF X-STIFFENERS,3X,21H
     1 DEPTH OF X-STIFFENERS)
      READ(5,*)BX,TH2
      WRITE(2,2012)BX,TH2
 2012 FORMAT(6X,E12.4,10X,E12.4)
      WRITE(2,2013)
 2013 FORMAT(2X,21HWIDTH OF Y-STIFFENERS,3X,21H
     1 DEPTH OF Y-STIFFENERS)
      READ(5,*)BY,TH3
      WRITE(2,2014)BY,TH3
 2014 FORMAT(6X,E12.4,10X,E12.4)
      IF(LTYPE-1)70,80,70
C*****************READING DISTRIBUTED LOAD DATA**********************
 70   READ(5,*)QZ(1)
      DO 90 I=2,NELEM
 90   QZ(I)=QZ(1)
      GO TO 100
C****READING CONCENTRATED LOAD DATA***********************************
 80   READ(5,*)NNCL
      DO 110 INCL=1,NNCL
 110  READ(5,*)NODE,(P(ID(I,NODE)),I=1,NDOFN)
      GO TO 120
C********************************************************************
 100  CALL GCLVA(NELEM,NODES,QZ,P,X,Y,LM)
 120  CALL CONS(YNG,POS,BX,BY,NSTX,NSTY,SNX,SNY,TH1,
     1 TH2,TH3,D1,D2,D3,C1,C1D,C2,C3,C4,C5,SR1,SR2,D12,
     2 D13,C13,C15,SRP)
      IF(IANTP.EQ.1)NSTEP=1
      DO 130 I=1,NADF
 130  DELP(I)=P(I)/REAL(NSTEP)
C****BEGIN LOAD-INCREMENT********************************************
      DO 140 ISTEP=1,NSTEP
      DO 150 I=1,NADF
      P(I)=REAL(ISTEP)*DELP(I)
      XI(I)=DELP(I)+XI(I)
 150  CONTINUE
C****BEGIN ITERATION LOOP********************************************
      IF(IANTP.EQ.1)ISME=1
      ITER=0
 220  ITER=ITER+1
      IF(ITER.EQ.1)THEN
      DO 158 I=1,NADF
 158  XII(I)=XI(I)
      END IF
```

```
      IF(ITER.LE.ISME)THEN
      DO 162 I=1,NOSTF
162   OSTF(I)=0.0
      DO 170 IE=1,NELEM
      CALL ESMB(IANTP,NODES,X,Y,IE,NNODE,NDEFE,NELEM,DISP,ESTF)
      CALL GSMB(LM,ESTF,IE,NADF,NDEFE,OSTF)
170   CONTINUE
      ELSE
      END IF
      CALL SOLV(NADF,OSTF,XI,XD)
      XD11=0.0
      K=0
      KOUNT=0
      DO 180 J=1,NODET
      DO 180 I=1,NDOFN
      IF(ID(I,J).EQ.0)THEN
      KOUNT=KOUNT+1
      DELXD(KOUNT)=XD11
      ELSE
      K=K+1
      KOUNT=KOUNT+1
      DELXD(KOUNT)=XD(K)
      END IF
180   CONTINUE
      DO 190 I=1,NDEF
      DISP(I)=DISP(I)+DELXD(I)
190   CONTINUE
C***********************************************************************
      CALL STRESS(YNG,TH1,TH2,TH3,IANTP,NELEM,NNODE,X,Y,NODES,
     1 NODET,DELXD,DISP,DELXNX,DELXNY,DELXNXY,DELXMX,DELXMY,DELXMXY,
     2 DELXQXZ,DELXQYZ,DELXSIGPT,DELYSIGPT,DELXSIGSB,DELYSIGSB,NSTX,
     3 NSTY)
      DO 195 IE=1,NELEM
      DO 195 IG=1,4
      XNX(IE,IG)=XNX(IE,IG)+DELXNX(IE,IG)
      XNY(IE,IG)=XNY(IE,IG)+DELXNY(IE,IG)
      XNXY(IE,IG)=XNXY(IE,IG)+DELXNXY(IE,IG)
      XMX(IE,IG)=XMX(IE,IG)+DELXMX(IE,IG)
      XMY(IE,IG)=XMY(IE,IG)+DELXMY(IE,IG)
      XMXY(IE,IG)=XMXY(IE,IG)+DELXMXY(IE,IG)
      XQXZ(IE,IG)=XQXZ(IE,IG)+DELXQXZ(IE,IG)
      XQYZ(IE,IG)=XQYZ(IE,IG)+DELXQYZ(IE,IG)
      XSIGPT(IE,IG)=XSIGPT(IE,IG)+DELXSIGPT(IE,IG)
      YSIGPT(IE,IG)=YSIGPT(IE,IG)+DELYSIGPT(IE,IG)
      XSIGSB(IE,IG)=XSIGSB(IE,IG)+DELXSIGSB(IE,IG)
```

```
      YSIGSB(IE,IG)=YSIGSB(IE,IG)+DELYSIGSB(IE,IG)
195   CONTINUE
C*****************************************************************
      IF(IANTP.EQ.1)GO TO 137
      CALL RESIDUE(NADF,ITER,NELEM,NNODE,X,Y,NODES,DISP,LM,XXI)
      DO 200 I=1,NADF
200   XI(I)=P(I)-XXI(I)
      CH1=CH
      CALL CONV(NADF,ITER,ISTEP,NODET,TOLER,P,XI,INDEX,CH)
      IF(INDEX.EQ.1)THEN
      GO TO 138
      ELSE
      END IF
      IF(CH.GT.CH1.AND.ITER.GT.50)THEN
      GO TO 250
      ELSE
      GO TO 220
      END IF
138   WRITE(6,2019)ISTEP
2019  FORMAT(///,2X,12HLOAD STEP = ,I3)
137   WRITE(6,2020)
2020  FORMAT(2X,8HNODE NO.,6X,1HU,13X,1HV,13X,1HW,10X,6HTHETAX,
    1 7X,6HTHETAY)
      DO 139 MK=1,NODET
      IF(NSOP.GT.0)THEN
      JK=NSOP
      WRITE(6,2021)JK,(DISP(5*JK+KK-5),KK=1,5)
2021  FORMAT(4X,I3,3X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4)
      GO TO 135
      ELSE
      END IF
      WRITE(6,2022)MK,(DISP(5*MK+KK-5),KK=1,5)
2022  FORMAT(4X,I3,3X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4)
139   CONTINUE
135   CALL STRESSF(NELEM,NNODE,NODET,NODES,NSOP)
      IF(IANTP.EQ.1)GO TO 240
140   CONTINUE
240   GO TO 260
250   WRITE(6,*)'ITERATIONS DIVERGING OR 100 ITERATIONS ALREADY'
      WRITE(6,*)'ITER=',ITER,'CH=',CH,'CH1=',CH1
260   STOP
      END
```

```
      SUBROUTINE SHAPE(S,T,SH,ADERIV)
      DIMENSION SH(8),ADERIV(2,8)
C***CALCULATES SHAPE FUNCTIONS AND THEIR ADERIVATIVES***
      SH(1)=0.25*(1.-S)*(1.+T)*(-S+T-1)
      SH(2)=0.5*(1.+T)*(1.-S**2)
      SH(3)=0.25*(1.+S)*(1.+T)*(S+T-1)
      SH(4)=0.5*(1.+S)*(1.-T**2)
      SH(5)=0.25*(1.+S)*(1.-T)*(S-T-1)
      SH(6)=0.5*(1.-T)*(1.-S**2)
      SH(7)=0.25*(1.-S)*(1.-T)*(-S-T-1.)
      SH(8)=0.5*(1.-S)*(1.-T**2)
      S2=2.*S
      T2=2.*T
      ST2=2.*S*T
      ADERIV(1,1)=0.25*(S2-T+ST2-T*T)
      ADERIV(1,2)=0.5*(-S2-ST2)
      ADERIV(1,3)=0.25*(S2+T+ST2+T*T)
      ADERIV(1,4)=0.5*(1.-T*T)
      ADERIV(1,5)=0.25*(S2-T-ST2+T*T)
      ADERIV(1,6)=0.5*(-S2+ST2)
      ADERIV(1,7)=0.25*(S2+T-ST2-T*T)
      ADERIV(1,8)=0.5*(-1.+T*T)
      ADERIV(2,1)=0.25*(T2-S+S*S-ST2)
      ADERIV(2,2)=0.5*(1.-S*S)
      ADERIV(2,3)=0.25*(T2+S+S*S+ST2)
      ADERIV(2,4)=0.5*(-T2-ST2)
      ADERIV(2,5)=0.25*(T2-S-S*S+ST2)
      ADERIV(2,6)=0.5*(-1.+S*S)
      ADERIV(2,7)=0.25*(T2+S-S*S-ST2)
      ADERIV(2,8)=0.5*(-T2+ST2)
      RETURN
      END

      SUBROUTINE JACOB(IE,X,Y,NODES,
     1 ADERIV,XDJAC,CARTD)
      DIMENSION X(100),Y(100),NODES(100,8),EJAC(2,2),EJINV(2,2),
     1 ADERIV(2,8),CARTD(2,8)
C*******EVALUATES THE JACOBIAN AND ITS INVERSE**********************
      DO 9 I=1,2
      DO 9 J=1,2
      EJAC(I,J)=0.
      EJINV(I,J)=0.
    9 CONTINUE
      DO 10 I=1,8
```

```
      EJAC(1,1)=EJAC(1,1)+ADERIV(1,I)*X(NODES(IE,I))
      EJAC(1,2)=EJAC(1,2)+ADERIV(1,I)*Y(NODES(IE,I))
      EJAC(2,1)=EJAC(2,1)+ADERIV(2,I)*X(NODES(IE,I))
      EJAC(2,2)=EJAC(2,2)+ADERIV(2,I)*Y(NODES(IE,I))
10    CONTINUE
      XDJAC=EJAC(1,1)*EJAC(2,2)-EJAC(1,2)*EJAC(2,1)
      IF(XDJAC.LE.0.0) GO TO 52
      EJINV(1,1)=EJAC(2,2)/XDJAC
      EJINV(1,2)=-EJAC(1,2)/XDJAC
      EJINV(2,1)=-EJAC(2,1)/XDJAC
      EJINV(2,2)=EJAC(1,1)/XDJAC
      DO 20 I=1,8
      CARTD(1,I)=EJINV(1,1)*ADERIV(1,I)+EJINV(1,2)*ADERIV(2,I)
      CARTD(2,I)=EJINV(2,1)*ADERIV(1,I)+EJINV(2,2)*ADERIV(2,I)
20    CONTINUE
      GO TO 53
53    RETURN
52    WRITE(6,*) 'DETERMINANT OF JACOBIAN LESS OR EQUAL TO ZERO'
      WRITE(6,*)'PROGRAM TERMINATED'
      END

      SUBROUTINE CONS(YNG,POS,BX,BY,NSTX,NSTY,SNX,SNY,TH1,
     1 TH2,TH3,D1,D2,D3,C1,C1D,C2,C3,C4,C5,SR1,SR2,D12,
     2 D13,C13,C15,SRP)
C***EVALUATES CONSTANTS FOR PLATE-STIFFENER SYSTEM***
      DENX=NSTX*BX/SNY
      DENY=NSTY*BY/SNX
      D1=YNG*TH1**3/(12.*(1.-POS*POS))
      D2=YNG*DENX*TH2*(TH2*TH2+1.5*TH1*TH2+0.75*TH1*TH1)/3.
      D3=YNG*DENY*TH3*(TH3*TH3+1.5*TH1*TH3+0.75*TH1*TH1)/3.
      C1=YNG*TH1/(1.-POS*POS)
      C1D=YNG*TH1/(2.*(1.+POS))
      C2=YNG*DENX*TH2*(TH2+TH1)/2.
      C3=YNG*DENX*TH2
      C4=YNG*DENY*TH3*(TH1+TH3)/2.
      C5=YNG*DENY*TH3
      SR1=YNG*(TH1+TH2*DENX)/(2.4*(1.+POS))
      SR2=YNG*(TH1+TH3*DENY)/(2.4*(1.+POS))
      SRP=YNG*TH1/(2.4*(1.+POS))
      D12=D1+D2
      D13=D1+D3
      C13=C1+C3
      C15=C1+C5
      RETURN
```

```
      END


      SUBROUTINE ESMB(IANTP,NODES,X,Y,IE,NNODE,NDEFE,NELEM,DISP,
     1 ESTF)
C*******CALCULATES ELEMENT STIFFNESS MATRIX*************************
      DIMENSION TEMP(25),CARTD(2,8),SH(8),ESTF(40,40),ADERIV(2,8)
      DIMENSION GSPX(4),GSPY(4),W1(4),NODES(100,8),X(100),Y(100),
     1 DISP(600)
      COMMON/CONS/POS,C13,C1,C2,C15,C4,C1D,D12,D1,D13,SR1,SR2,SRP
      COMMON/STRES/XNX(50,4),XNY(50,4),XNXY(50,4),XMX(50,4),XMY(50,4),
     1 XMXY(50,4),XQXZ(50,4),XQYZ(50,4)
      CALL GAUSQ2(GSPX,GSPY,W1)
      DO 10 I=1,NNODE
      DO 10 J=1,NNODE
      DO 20 I1=1,25
      TEMP(I1)=0.0
 20   CONTINUE
C*******2-POINT REDUCED INTEGRATION ********************************
C*******GENERATE LINEAR STIFFNESS MATRIX****************************
      DO 30 K=1,4
      S=GSPX(K)
      T=GSPY(K)
      CALL SHAPE(S,T,SH,ADERIV)
      CALL JACOB(IE,X,Y,NODES,ADERIV,XDJAC,CARTD)
      CST=W1(K)*XDJAC
      TEMP(1)=TEMP(1)+CST*(CARTD(1,I)*CARTD(1,J)*C13+
     1 CARTD(2,I)*CARTD(2,J)*C1D)
      TEMP(2)=TEMP(2)+CST*(POS*C1*CARTD(1,I)*CARTD(2,J)+
     1 C1D*CARTD(2,I)*CARTD(1,J))
      TEMP(4)=TEMP(4)-CST*C2*CARTD(1,I)*CARTD(1,J)
      TEMP(6)=TEMP(6)+CST*(POS*C1*CARTD(2,I)*CARTD(1,J)+
     1 C1D*CARTD(1,I)*CARTD(2,J))
      TEMP(7)=TEMP(7)+CST*(C15*CARTD(2,I)*CARTD(2,J)+
     1 C1D*CARTD(1,I)*CARTD(1,J))
      TEMP(10)=TEMP(10)-CST*C4*CARTD(2,I)*CARTD(2,J)
      TEMP(13)=TEMP(13)+CST*(SR1*CARTD(1,I)*CARTD(1,J)+
     1 SR2*CARTD(2,I)*CARTD(2,J))
      TEMP(14)=TEMP(14)-CST*SR1*CARTD(1,I)*SH(J)
      TEMP(15)=TEMP(15)-CST*SR2*CARTD(2,I)*SH(J)
      TEMP(18)=TEMP(18)-CST*SR1*SH(I)*CARTD(1,J)
      TEMP(16)=TEMP(16)-CST*C2*CARTD(1,I)*CARTD(1,J)
      TEMP(19)=TEMP(19)+CST*(D12*CARTD(1,I)*CARTD(1,J)+
     1 D1*(1.-POS)*CARTD(2,I)*CARTD(2,J)/2.0+SR1*SH(I)*SH(J))
```

```
      TEMP(20)=TEMP(20)+CST*(D1*POS*CARTD(1,I)*CARTD(2,J)+
     1 D1*(1.-POS)*CARTD(2,I)*CARTD(1,J)/2.0)
      TEMP(22)=TEMP(22)-CST*C4*CARTD(2,I)*CARTD(2,J)
      TEMP(23)=TEMP(23)-CST*SR2*SH(I)*CARTD(2,J)
      TEMP(24)=TEMP(24)+CST*(POS*D1*CARTD(2,I)*CARTD(1,J)+
     1 D1*(1.-POS)*CARTD(1,I)*CARTD(2,J)/2.0)
      TEMP(25)=TEMP(25)+CST*(D13*CARTD(2,I)*CARTD(2,J)+
     1 D1*(1.-POS)*CARTD(1,I)*CARTD(1,J)/2.0+SR2*SH(I)*SH(J))
   30 CONTINUE
      IF(IANTP.EQ.1)GO TO 90
C*******BEGIN DISPLACEMENT-DEPENDENT STIFFNESS TERMS*****************
      DO 40 K=1,4
      S=GSPX(K)
      T=GSPY(K)
      CALL SHAPE(S,T,SH,ADERIV)
      CALL JACOB(IE,X,Y,NODES,ADERIV,XDJAC,CARTD)
      CST=W1(K)*XDJAC
      XM1=0.0
      XM2=0.0
      DO 50 JK=1,NNODE
      XM1=XM1+DISP(5*NODES(IE,JK)-2)*CARTD(1,JK)
      XM2=XM2+DISP(5*NODES(IE,JK)-2)*CARTD(2,JK)
   50 CONTINUE
C***********************BL-D-BNL**********************************
      TEMP(3)=TEMP(3)+CST*(CARTD(1,I)*(XM1*CARTD(1,J))*C13+
     1 CARTD(1,I)*(XM2*CARTD(2,J))*POS*C1+CARTD(2,I)*(XM2*CARTD(1,J)+
     2 XM1*CARTD(2,J))*C1D)
      TEMP(8)=TEMP(8)+CST*(CARTD(2,I)*(XM1*CARTD(1,J))*POS*C1+
     1 CARTD(2,I)*(XM2*CARTD(2,J))*C15+CARTD(1,I)*(XM2*CARTD(1,J)+
     2 XM1*CARTD(2,J))*C1D)
      TEMP(18)=TEMP(18)-CST*CARTD(1,I)*(XM1*CARTD(1,J))*C2
      TEMP(23)=TEMP(23)-CST*CARTD(2,I)*(XM2*CARTD(2,J))*C4
C***********************BNL-D-BL**********************************
      TEMP(11)=TEMP(11)+CST*((XM1*CARTD(1,I))*CARTD(1,J)*C13+
     1 (XM2*CARTD(2,I))*CARTD(1,J)*POS*C1+(XM2*CARTD(1,I)+XM1*
     2 CARTD(2,I))*CARTD(2,J)*C1D)
      TEMP(12)=TEMP(12)+CST*((XM1*CARTD(1,I))*CARTD(2,J)*POS*C1+
     1 (XM2*CARTD(2,I))*CARTD(2,J)*C15+(XM2*CARTD(1,I)+XM1*CARTD(2,I))*
     2 CARTD(1,J)*C1D)
      TEMP(14)=TEMP(14)-CST*(XM1*CARTD(1,I))*CARTD(1,J)*C2
      TEMP(15)=TEMP(15)-CST*(XM2*CARTD(2,I))*CARTD(2,J)*C4
C***********************BNL-D-BNL*********************************
      TEMP(13)=TEMP(13)+CST*((XM1*CARTD(1,I))*(XM1*CARTD(1,J))*C13+
     1 (XM1*CARTD(1,I))*(XM2*CARTD(2,J))*POS*C1+(XM2*CARTD(2,I))*
     2 (XM1*CARTD(1,J))*POS*C1+(XM2*CARTD(2,I))*(XM2*CARTD(2,J))*C15+
```

```
     3  (XM2*CARTD(1,I)+XM1*CARTD(2,I))*(XM2*CARTD(1,J)+XM1*CARTD(2,J))
     4   *C1D)
C****************GEOMETRIC STIFFNESS TERMS***************************
 80      TEMP(13)=TEMP(13)+CST*(XNX(IE,K)*CARTD(1,I)*CARTD(1,J)-
     1   XNXY(IE,K)*(CARTD(2,I)*CARTD(1,J)+CARTD(1,I)*CARTD(2,J))+
     2   XNY(IE,K)*CARTD(2,I)*CARTD(2,J))
 40      CONTINUE.
C*********************************************************************
 90      L=0
         DO 10 II=5*I-4,5*I
         DO 10 JJ=5*J-4,5*J
         L=L+1
         ESTF(II,JJ)=TEMP(L)
 10      CONTINUE
         DO 60 I=1,NDEFE
         IF(ESTF(I,I).LE.0.) GO TO 61
 60      CONTINUE
         RETURN
 61      WRITE(6,*)'A DIAGONAL  ELEMENT OF AN ELEMENT STIFFNESS MATRIX '
         WRITE(6,*)'LESS OR EQUAL TO ZERO; PROGRAM TERMINATED.'
         END


         SUBROUTINE GSMB(LM,E,IE,NADF,NDEFE,O)
         DIMENSION LM(100,40),E(40,40),O(100000)
         DO 10 I=1,NDEFE
         LM1=LM(IE,I)
         DO 10 J=I,NDEFE
         LM2=LM(IE,J)
         IF(LM2.EQ.0)GO TO 10
         IF(LM1.EQ.0)GO TO 10
         IF(LM1.LE.LM2)GO TO 20
         GO TO 30
 20      IF(LM1.EQ.1)THEN
         K=LM1+LM2-1
         O(K)=O(K)+E(I,J)
         ELSE IF (LM1.GT.1)THEN
         LS=0
         DO 25 II=1,LM1-1
 25      LS=(NADF-II+1)+LS
         K=LS+(LM2-LM1+1)
         O(K)=O(K)+E(I,J)
         END IF
         GO TO 10
```

```
30      LS=0
        DO 35 JJ=1,LM2-1
35      LS=(NADF-JJ+1)+LS
        K=LS+(LM1-LM2+1)
        O(K)=O(K)+E(I,J)
10      CONTINUE
        RETURN
        END


        SUBROUTINE GAUSQ2(GSPX,GSPY,W1)
C*******TWO-POINT GAUSS INTREGATION INITIALISATION****
        DIMENSION XG(2),YG(2),W(2),W1(4),GSPX(4),GSPY(4)
        XG(1)=0.5773502692
        XG(2)=-0.5773502692
        YG(1)=XG(1)
        YG(2)=XG(2)
        W(1)=1.0
        W(2)=1.0
        K=0
        DO 10 I=1,2
        DO 10 J=1,2
        K=K+1
        W1(K)=W(I)*W(J)
10      CONTINUE
        GSPX(1)=-XG(1)
        GSPX(2)=XG(1)
        GSPX(3)=XG(1)
        GSPX(4)=-XG(1)
        GSPY(1)=XG(1)
        GSPY(2)=XG(1)
        GSPY(3)=-XG(1)
        GSPY(4)=-XG(1)
        RETURN
        END


        SUBROUTINE GAUSQ3(GSPX3,GSPY3,W3)
C*******THREE-POINT GAUSS INTEGRATION IN 2-D INITIALISATION******
        DIMENSION XG(3),YG(3),W(3),W3(9),GSPX3(9),GSPY3(9)
        XG(1)=0.77459667
        XG(2)=-0.77459667
        XG(3)=0.0
```

```
      YG(1)=XG(1)
      YG(2)=XG(2)
      YG(3)=XG(3)
      W(1)=0.55555556
      W(2)=0.55555556
      W(3)=0.88888889
      K=0
      DO 10 I=1,3
      DO 10 J=1,3
      K=K+1
      GSPX3(K)=XG(I)
      GSPY3(K)=YG(J)
      W3(K)=W(I)*W(J)
10    CONTINUE
      RETURN
      END


      SUBROUTINE GAUSQ21(GSPX,GSPY,W1)
C*******TWO-POINT GAUSS INTEGRATION IN ONE-DIMENSION INITIALISATION*****
      DIMENSION GSPX(2),GSPY(2)
      GSPX(1)=-0.5773502692
      GSPX(2)=0.5773502692
      GSPY(1)=GSPX(1)
      GSPY(2)=GSPX(2)
      W1=1.0
      RETURN
      END


      SUBROUTINE CAPE(NELEM,NNODE,NDOFN,NODES,LM,ID,NDEFE)
      DIMENSION LM(100,40),ID(5,100),NODES(100,8)
C*******GENERATES CONNECTIVITY ARRAY FOR PLATE ELEMENTS*****************
      DO 70 IE=1,NELEM
      IJ=0
      DO 70 I=1,NNODE
      DO 70 J=1,NDOFN
      IJ=IJ+1
      LM(IE,IJ)=ID(J,NODES(IE,I))
70    CONTINUE
      RETURN
      END

      SUBROUTINE SOLV(NADF,OSTF,P,XD)
```

```
      DIMENSION IDIG(600),OSTF(100000),P(600),XD(600),P1(600)
C*****IDENTIFYING THE LOCATIONS OF DIAGONAL ELEMENTS************
      NN=NADF*(NADF+1)/2
      DO 3 I=1,NADF
 3    P1(I)=P(I)
      DO 4 I=1,NADF
 4    IDIG(I)=0
      IDIG(1)=1
      DO 10 I=2,NADF
      DO 20 K=1,I-1
 20   IDIG(I)=IDIG(I)+(NADF-K+1)
      IDIG(I)=IDIG(I)+1
 10   CONTINUE
      DO 5 K=1,NADF
      IF(OSTF(IDIG(K)).LE.0.0)GO TO 81
5     CONTINUE
C*******FORWARD ELIMINATION****************************************
      DO 30 I=1,NADF-1
      L=0
      DO 35 J=I,NADF-1
      L=L+1
      DO 40 K=J,NADF-1
 40   OSTF(IDIG(J+1)+K-J)=OSTF(IDIG(J+1)+K-J)-
     1 OSTF(IDIG(I)+L)*OSTF(IDIG(I)+K-J+L)/OSTF(IDIG(I))
      P1(I+L)=P1(I+L)-P1(I)*OSTF(IDIG(I)+L)/OSTF(IDIG(I))
 35   CONTINUE
 30   CONTINUE
C************BACK SUBSTITUTION**************************************
      MI=0
      LI=0
      DO 50 I=1,NADF
      LI=LI+1
      IF(I.GT.1)GO TO 60
      XD(NADF)=P1(NADF)/OSTF(IDIG(NADF))
      GO TO 50
 60   SUM=0
      MI=MI+1
      DO 70 K=1,LI-1
 70   SUM=SUM+OSTF(IDIG(NADF-I+1)+K)*XD(NADF-MI+K)
      XD(NADF-I+1)=(P1(NADF-I+1)-SUM)/OSTF(IDIG(NADF-I+1))
 50   CONTINUE
      RETURN
 81   WRITE(6,*)'GLOBAL STIFFNESS MATRIX NOT POSITIVE DEFINITE'
      WRITE(6,*)'PROGRAM TERMINATED'
      END
```

```
      SUBROUTINE STRESS(YNG,TH1,TH2,TH3,IANTP,NELEM,NNODE,X,Y,NODES,
     1 NODET,DELXD,DISP,XNX,XNY,XNXY,XMX,XMY,XMXY,XQXZ,XQYZ,
     2 XSIGPT,YSIGPT,XSIGSB,YSIGSB,NSTX,NSTY)
C*******CALCULATES STRESSES AT ELEMENT NODES BY BILINEAR EXTRAPOLATION**
      DIMENSION GSPX(4),GSPY(4),W(4),CARTD(2,8),ADERIV(2,8),X(100),
     1 SH(8),X1(4),X2(4),X3(4),X4(4),X5(4),X6(4),X7(4),X8(4),Y(100),
     2 NODES(100,8),DISP(600),XNX(50,4),XNY(50,4),XNXY(50,4),XMX(50,4),
     3 XMY(50,4),XMXY(50,4),XQXZ(50,4),XQYZ(50,4),DELXD(600),
     4 XSIGPT(50,4),YSIGPT(50,4),XSIGSB(50,4),YSIGSB(50,4)
      COMMON/CONS/POS,C13,C1,C2,C15,C4,C1D,D12,D1,D13,SR1,SR2,SRP
      NDEF=NODET*5
      CALL GAUSQ2(GSPX,GSPY,W)
      DO 8 IE=1,NELEM
      DO 10 IG=1,4
      X1(IG)=0.
      X2(IG)=0.
      X3(IG)=0.
      X4(IG)=0.
      X5(IG)=0.
      X6(IG)=0.
      X7(IG)=0.
      X8(IG)=0.
      S=GSPX(IG)
      T=GSPY(IG)
      CALL SHAPE(S,T,SH,ADERIV)
      CALL JACOB(IE,X,Y,NODES,ADERIV,XDJAC,CARTD)
      XM1=0.0
      XM2=0.0
      DO 12 JK=1,NNODE
      XM1=XM1+DISP(5*NODES(IE,JK)-2)*CARTD(1,JK)
   12 XM2=XM2+DISP(5*NODES(IE,JK)-2)*CARTD(2,JK)
      IF(IANTP.EQ.1)XM1=0.0
      IF(IANTP.EQ.1)XM2=0.0
      DO 20 I=1,NNODE
      X1(IG)=X1(IG)+CARTD(1,I)*DELXD(5*NODES(IE,I)-4)+
     1 XM1*CARTD(1,I)*DELXD(5*NODES(IE,I)-2)
      X2(IG)=X2(IG)+CARTD(2,I)*DELXD(5*NODES(IE,I)-3)+
     1 XM2*CARTD(2,I)*DELXD(5*NODES(IE,I)-2)
      X3(IG)=X3(IG)-(CARTD(2,I)*DELXD(5*NODES(IE,I)-4)
     1 +CARTD(1,I)*DELXD(5*NODES(IE,I)-3))-(XM2*CARTD(1,I)+
     2 XM1*CARTD(2,I))*DELXD(5*NODES(IE,I)-2)
      X4(IG)=X4(IG)-CARTD(1,I)*DELXD(5*NODES(IE,I)-1)
      X5(IG)=X5(IG)-CARTD(2,I)*DELXD(5*NODES(IE,I))
      X6(IG)=X6(IG)+(CARTD(2,I)*DELXD(5*NODES(IE,I)-1)
     1 +CARTD(1,I)*DELXD(5*NODES(IE,I)))
```

```
      X7(IG)=X7(IG)-(CARTD(1,I)*DELXD(5*NODES(IE,I)-2)
     1 -SH(I)*DELXD(5*NODES(IE,I)-1))
      X8(IG)=X8(IG)-(CARTD(2,I)*DELXD(5*NODES(IE,I)-2)
     1 -SH(I)*DELXD(5*NODES(IE,I)))
20    CONTINUE
      XNX(IE,IG)=C13*X1(IG)+POS*C1*X2(IG)+C2*X4(IG)
      XNY(IE,IG)=POS*C1*X1(IG)+C15*X2(IG)+C4*X5(IG)
      XNXY(IE,IG)=C1D*X3(IG)
      XMX(IE,IG)=C2*X1(IG)+D12*X4(IG)+POS*D1*X5(IG)
      XMY(IE,IG)=C4*X2(IG)+POS*D1*X4(IG)+D13*X5(IG)
      XMXY(IE,IG)=0.5*(1.-POS)*D1*X6(IG)
      XQXZ(IE,IG)=SR1*X7(IG)
      XQYZ(IE,IG)=SR2*X8(IG)
      XSIGPT(IE,IG)=YNG*(X1(IG)+POS*X2(IG)-0.5*TH1*X4(IG)-
     1 POS*0.5*TH1*X5(IG))/(1.-POS**2)
      YSIGPT(IE,IG)=YNG*(POS*X1(IG)+X2(IG)-POS*0.5*TH1*X4(IG)-
     1 0.5*TH1*X5(IG))/(1.-POS**2)
      IF(NSTX.GT.0)THEN
      XSIGSB(IE,IG)=YNG*(X1(IG)+(0.5*TH1+TH2)*X4(IG))/(1.-POS**2)
      END IF
      IF(NSTY.GT.0)THEN
      YSIGSB(IE,IG)=YNG*(X2(IG)+(0.5*TH1+TH3)*X5(IG))/(1.-POS**2)
      END IF
10    CONTINUE
8     CONTINUE
      RETURN
      END


      SUBROUTINE STRESSF(NELEM,NNODE,NODET,NODES,NSOP)
      DIMENSION ELNX(80,100),ELNY(80,100),ELNXY(80,100),ELMX(80,100),
     1 ELMY(80,100),ELMXY(80,100),ELQXZ(80,100),ELQYZ(80,100),X(100),
     2 Y(100),NODES(100,8),ELXSIGPT(80,100),ELYSIGPT(80,100),
     3 ELXSIGSB(80,100),ELYSIGSB(80,100),W(8,4),N(8)
      DIMENSION IFR(100),NST(100,10),S1(100),S2(100),S3(100),S4(100),
     1 S5(100),S6(100),S7(100),S8(100),ST1(100),ST2(100),ST3(100),
     2 ST4(100)
      COMMON/STRES/XNX(50,4),XNY(50,4),XNXY(50,4),XMX(50,4),XMY(50,4),
     1 XMXY(50,4),XQXZ(50,4),XQYZ(50,4),XSIGPT(50,4),YSIGPT(50,4),
     2 XSIGSB(50,4),YSIGSB(50,4)
C******CALCULATES ELEMENT STRESSES BY BILINEAR EXTRAPOLATION***********
      DO 5 K=1,NODET
      DO 5 IE=1,NELEM
      S1(K)=0.0
```

```
      S2(K)=0.0
      S3(K)=0.0
      S4(K)=0.0
      S5(K)=0.0
      S6(K)=0.0
      S7(K)=0.0
      S8(K)=0.0
      ST1(K)=0.0
      ST2(K)=0.0
      ST3(K)=0.0
      ST4(K)=0.0
      IFR(K)=0.0
      ELNX(IE,K)=0.0
      ELNY(IE,K)=0.0
      ELNXY(IE,K)=0.0
      ELMX(IE,K)=0.0
      ELMY(IE,K)=0.0
      ELMXY(IE,K)=0.0
      ELQXZ(IE,K)=0.0
      ELQYZ(IE,K)=0.0
      ELXSIGPT(IE,K)=0.0
      ELYSIGPT(IE,K)=0.0
      ELXSIGSB(IE,K)=0.0
5     ELYSIGSB(IE,K)=0.0
      DO 20 K=1,NODET
      DO 20 IM=1,NELEM
      DO 20 IK=1,NNODE
      IF(NODES(IM,IK).EQ.K)THEN
      IFR(K)=IFR(K)+1
      NST(K,IFR(K))=IM
      ELSE
      END IF
20    CONTINUE
      DO 10 IE=1,NELEM
      DO 15 I=1,8
      N(I)=NODES(IE,I)
15    CONTINUE
      P=0.5773502692
      Q=0.5773502692
      W1=0.25*(1.+1./P)*(1.+1./Q)
      W2=0.25*(1.-1./P)*(1.+1./Q)
      W3=0.25*(1.-1./P)*(1.-1./Q)
      W4=0.25*(1.+1./P)
      W5=0.25*(1.-1./P)
      W(1,1)=W1
```

```
W(2,1)=W4
W(3,1)=W2
W(4,1)=W5
W(5,1)=W3
W(6,1)=W5
W(7,1)=W2
W(8,1)=W4
W(1,2)=W2
W(2,2)=W4
W(3,2)=W1
W(4,2)=W4
W(5,2)=W2
W(6,2)=W5
W(7,2)=W3
W(8,2)=W5
W(1,3)=W3
W(2,3)=W5
W(3,3)=W2
W(4,3)=W4
W(5,3)=W1
W(6,3)=W4
W(7,3)=W2
W(8,3)=W5
W(1,4)=W2
W(2,4)=W5
W(3,4)=W3
W(4,4)=W5
W(5,4)=W2
W(6,4)=W4
W(7,4)=W1
W(8,4)=W4
DO 60 J=1,8
DO 60 K=1,4
ELNX(IE,N(J))=W(J,K)*XNX(IE,K)+ELNX(IE,N(J))
ELNY(IE,N(J))=W(J,K)*XNY(IE,K)+ELNY(IE,N(J))
ELNXY(IE,N(J))=W(J,K)*XNXY(IE,K)+ELNXY(IE,N(J))
ELMX(IE,N(J))=W(J,K)*XMX(IE,K)+ELMX(IE,N(J))
ELMY(IE,N(J))=W(J,K)*XMY(IE,K)+ELMY(IE,N(J))
ELMXY(IE,N(J))=W(J,K)*XMXY(IE,K)+ELMXY(IE,N(J))
ELQXZ(IE,N(J))=W(J,K)*XQXZ(IE,K)+ELQXZ(IE,N(J))
ELQYZ(IE,N(J))=W(J,K)*XQYZ(IE,K)+ELQYZ(IE,N(J))
ELXSIGPT(IE,N(J))=W(J,K)*XSIGPT(IE,K)+ELXSIGPT(IE,N(J))
ELYSIGPT(IE,N(J))=W(J,K)*YSIGPT(IE,K)+ELYSIGPT(IE,N(J))
ELXSIGSB(IE,N(J))=W(J,K)*XSIGSB(IE,K)+ELXSIGSB(IE,N(J))
ELYSIGSB(IE,N(J))=W(J,K)*YSIGSB(IE,K)+ELYSIGSB(IE,N(J))
```

```
60      CONTINUE
10      CONTINUE
        DO 30 K=1,NODET
        DO 30 I=1,IFR(K)
        S1(K)=S1(K)+ELNX(NST(K,I),K)
        S2(K)=S2(K)+ELNY(NST(K,I),K)
        S3(K)=S3(K)+ELNXY(NST(K,I),K)
        S4(K)=S4(K)+ELMX(NST(K,I),K)
        S5(K)=S5(K)+ELMY(NST(K,I),K)
        S6(K)=S6(K)+ELMXY(NST(K,I),K)
        S7(K)=S7(K)+ELQXZ(NST(K,I),K)
        S8(K)=S8(K)+ELQYZ(NST(K,I),K)
        ST1(K)=ST1(K)+ELXSIGPT(NST(K,I),K)
        ST2(K)=ST2(K)+ELYSIGPT(NST(K,I),K)
        ST3(K)=ST3(K)+ELXSIGSB(NST(K,I),K)
        ST4(K)=ST4(K)+ELYSIGSB(NST(K,I),K)
30      CONTINUE
        DO 40 K=1,NODET
        S1(K)=S1(K)/REAL(IFR(K))
        S2(K)=S2(K)/REAL(IFR(K))
        S3(K)=S3(K)/REAL(IFR(K))
        S4(K)=S4(K)/REAL(IFR(K))
        S5(K)=S5(K)/REAL(IFR(K))
        S6(K)=S6(K)/REAL(IFR(K))
        S7(K)=S7(K)/REAL(IFR(K))
        S8(K)=S8(K)/REAL(IFR(K))
        ST1(K)=ST1(K)/REAL(IFR(K))
        ST2(K)=ST2(K)/REAL(IFR(K))
        ST3(K)=ST3(K)/REAL(IFR(K))
        ST4(K)=ST4(K)/REAL(IFR(K))
40      CONTINUE
        IF(NSOP.GT.0)THEN
        GO TO 90
        ELSE
        END IF
        WRITE(6,2000)
2000    FORMAT(3X,8HNODE NO.,7X,3HNXX,10X,3HNYY,10X,3HNXY,10X,3HMXX,
     1  10X,3HMYY,10X,3HMXY,10X,3HQXZ,10X,3HQYZ)
        DO 50 K=1,NODET
        WRITE(6,2010)K,S1(K),S2(K),S3(K),S4(K),S5(K),S6(K),S7(K),S8(K)
2010    FORMAT(5X,I3,4X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4,2X,
     1  E11.4,2X,E11.4,2X,E11.4,2X,E11.4)
50      CONTINUE
        WRITE(6,2020)
2020    FORMAT(///,3X,102HSTRESSES AT PLATE TOP (XSIGPT & YSIGPT) AND
```

```
     1  STIFFENER BOTTOM (XSIGSB & YSIGSB) IN X- AND Y-DIRECTIONS)
        WRITE(6,2030)
2030    FORMAT(/,3X,8HNODE NO.,3X,6HXSIGPT,7X,6HYSIGPT,7X,6HXSIGSB,7X,
     1  6HYSIGSB)
        DO 70 K=1,NODET
        WRITE(6,2040)K,ST1(K),ST2(K),ST3(K),ST4(K)
2040    FORMAT(5X,I3,2X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4)
70      CONTINUE
        GO TO 100
90      WRITE(6,2050)
2050    FORMAT(3X,8HNODE NO.,7X,3HNXX,10X,3HNYY,10X,3HNXY,10X,3HMXX,
     1  10X,3HMYY,10X,3HMXY,10X,3HQXZ,10X,3HQYZ)
        K=NSOP
        WRITE(6,2060)K,S1(K),S2(K),S3(K),S4(K),S5(K),S6(K),S7(K),S8(K)
2060    FORMAT(5X,I3,4X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4,2X,
     1  E11.4,2X,E11.4,2X,E11.4,2X,E11.4)
        WRITE(6,2070)
2070    FORMAT(///,3X,102HSTRESSES AT PLATE TOP (XSIGPT & YSIGPT) AND
     1  STIFFENER BOTTOM (XSIGSB & YSIGSB) IN X- AND Y-DIRECTIONS)
        WRITE(6,2080)
2080    FORMAT(/,3X,8HNODE NO.,3X,6HXSIPT,7X,6HYSIGPT,7X,6HXSIGSB,7X,
     1  6HYSIGSB)
        WRITE(6,2090)K,ST1(K),ST2(K),ST3(K),ST4(K)
2090    FORMAT(6X,I3,2X,E11.4,2X,E11.4,2X,E11.4,2X,E11.4)
100     RETURN
        END



        SUBROUTINE GCLVA(NELEM,NODES,QZ,P,X,Y,LM)
        DIMENSION NODES(100,8),QZ(100),P(600),X(100),Y(100),
     1  LM(100,40),QC(8),GSPX(4),GSPY(4),W1(4),SH(8),ADERIV(2,8),
     2  CARTD(2,8),GSPX3(3),GSPY3(3),W3(9)
        CALL GAUSQ2(GSPX,GSPY,W1)
        DO 10 IE=1,NELEM
        DO 20 I=1,8
20      QC(I)=0.0
        DO 30 K=1,4
        S=GSPX(K)
        T=GSPY(K)
        CALL SHAPE(S,T,SH,ADERIV)
        CALL JACOB(IE,X,Y,NODES,ADERIV,XDJAC,CARTD)
        CST=W1(K)*XDJAC
```

```
      DO 40 I=1,8
40    QC(I)=QC(I)+SH(I)*QZ(IE)*CST
30    CONTINUE
      IF(LM(IE,3).EQ.0)THEN
      GO TO 50
      ELSE
      P(LM(IE,3))=P(LM(IE,3))+QC(1)
      END IF
50    IF(LM(IE,8).EQ.0)THEN
      GO TO 60
      ELSE
      P(LM(IE,8))=P(LM(IE,8))+QC(2)
      END IF
60    IF(LM(IE,13).EQ.0)THEN
      GO TO 70
      ELSE
      P(LM(IE,13))=P(LM(IE,13))+QC(3)
      END IF
70    IF(LM(IE,18).EQ.0)THEN
      GO TO 80
      ELSE
      P(LM(IE,18))=P(LM(IE,18))+QC(4)
      END IF
80    IF(LM(IE,23).EQ.0)THEN
      GO TO 90
      ELSE
      P(LM(IE,23))=P(LM(IE,23))+QC(5)
      END IF
90    IF(LM(IE,28).EQ.0)THEN
      GO TO 100
      ELSE
      P(LM(IE,28))=P(LM(IE,28))+QC(6)
      END IF
100   IF(LM(IE,33).EQ.0)THEN
      GO TO 110
      ELSE
      P(LM(IE,33))=P(LM(IE,33))+QC(7)
      END IF
110   IF(LM(IE,38).EQ.0)THEN
      GO TO 10
      ELSE
      P(LM(IE,38))=P(LM(IE,38))+QC(8)
      END IF
10    CONTINUE
      RETURN
```

```
      END

      SUBROUTINE RESIDUE(NADF,ITER,NELEM,NNODE,X,Y,NODES,DISP,LM,XXI)
      DIMENSION GSPX(4),GSPY(4),W1(4),X(100),Y(100),NODES(100,8),
     1 SH(8),ADERIV(2,8),CARTD(2,8),DISP(600),LM(100,40),XXI(600)
      COMMON/STRES/XNX(50,4),XNY(50,4),XNXY(50,4),XMX(50,4),XMY(50,4),
     1 XMXY(50,4),XQXZ(50,4),XQYZ(50,4)
C*******THIS IS A PREREQUISITE FOR THE RESIDUAL FORCE VECTOR************
      DO 5 I=1,NADF
    5 XXI(I)=0.0
      CALL GAUSQ2(GSPX,GSPY,W1)
      DO 10 IE=1,NELEM
      DO 20 IN=1,NNODE
      DO 30 KT=1,5
      INT=5*IN+KT-5
      DO 40 K=1,4
      S=GSPX(K)
      T=GSPY(K)
      CALL SHAPE(S,T,SH,ADERIV)
      CALL JACOB(IE,X,Y,NODES,ADERIV,XDJAC,CARTD)
      CST=W1(K)*XDJAC
      XM1=0.0
      XM2=0.0
      DO 50 JK=1,NNODE
      XM1=XM1+DISP(5*NODES(IE,JK)-2)*CARTD(1,JK)
   50 XM2=XM2+DISP(5*NODES(IE,JK)-2)*CARTD(2,JK)
C     XM1=0.0
C     XM2=0.0
      IF(LM(IE,INT).GT.0.AND.KT.EQ.1)THEN
      XXI(LM(IE,INT))=XXI(LM(IE,INT))+CST*(XNX(IE,K)*CARTD(1,IN)-
     1 XNXY(IE,K)*CARTD(2,IN))
      END IF
      IF(LM(IE,INT).GT.0.AND.KT.EQ.2)THEN
      XXI(LM(IE,INT))=XXI(LM(IE,INT))+CST*(XNY(IE,K)*CARTD(2,IN)-
     1 XNXY(IE,K)*CARTD(1,IN))
      END IF
      IF(LM(IE,INT).GT.0.AND.KT.EQ.3)THEN
      XXI(LM(IE,INT))=XXI(LM(IE,INT))+CST*(XNX(IE,K)*XM1*CARTD(1,IN)+
     1 XNY(IE,K)*XM2*CARTD(2,IN)-XNXY(IE,K)*XM2*CARTD(1,IN)-
     2 XNXY(IE,K)*XM1*CARTD(2,IN)-XQXZ(IE,K)*CARTD(1,IN)-XQYZ(IE,K)*
     3 CARTD(2,IN))
      END IF
      IF(LM(IE,INT).GT.0.AND.KT.EQ.4)THEN
```

```
      XXI(LM(IE,INT))=XXI(LM(IE,INT))+CST*(-XMX(IE,K)*CARTD(1,IN)+
     1 XMXY(IE,K)*CARTD(2,IN)+XQXZ(IE,K)*SH(IN))
      END IF
      IF(LM(IE,INT).GT.0.AND.KT.EQ.5)THEN
      XXI(LM(IE,INT))=XXI(LM(IE,INT))+CST*(-XMY(IE,K)*CARTD(2,IN)+
     1 XMXY(IE,K)*CARTD(1,IN)+XQYZ(IE,K)*SH(IN))
      END IF
40    CONTINUE
30    CONTINUE
20    CONTINUE
10    CONTINUE
      RETURN
      END


      SUBROUTINE CONV(NADF,ITER,ISTEP,NODET,TOLER,P,XI,INDEX,CH)
      DIMENSION XI(600),P(600)
C*****MODULE REQUIRED FOR CHECKING CONVERGENCE OF ITERATIONS***********
      S1=0.0
      S2=0.0
      DO 10 I=1,NADF
      S1=S1+XI(I)**2
      S2=S2+P(I)**2
10    CONTINUE
      S1=SQRT(S1)
      S2=SQRT(S2)
      CH=S1/S2
      IF(CH.LE.TOLER)THEN
      INDEX=1
      ELSE
      INDEX=0
      END IF
      RETURN
      END
```