A FINITE DIFFERENCE BASED FINITE STRIP

METHOD FOR THE ANALYSIS OF

TRANSLATIONAL SHELL STRUCTURES

by

R. BARKER

A thesis submitted in partial fulfilment of the requirements for the Degree of Master of Science in the Faculty of Engineering, University of Cape Town.

The copyright of this thesis is held by the University of Cape Town. Reproduction of the whole or any part may be made for study purposes only, and not for publication.

Department of Civil Engineering,
UNIVERSITY OF CAPE TOWN. September, 1976.
The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.
SYNOPSIS

A numerical method for the analysis of translational shell structures is presented. The finite strip concept is utilised together with finite difference approximations to the differential equations along nodal lines. Numerical examples include open and closed translational shells, with various end conditions and continuity over intermediate supports.
DECLARATION OF CANDIDATE

I, Roger Barker, hereby declare that this thesis is my own work and that it has not been submitted for a degree at another University.

[Signature]

September, 1976.
This is my commandment, That ye love one another, as I have loved you.

John 15:12
ACKNOWLEDGEMENTS

The writer wishes to express his appreciation to:

Mr. W.S. Doyle*, under whose guidance this research was conducted, for his enthusiasm, help and interest throughout the thesis period.

Professor J.B. Martint†, for allocating teaching duties in such a manner as to make personal research possible.

Mackintosh, Bergh and Sturgess, Consulting Engineers for financial support.

Scott and de Waal, Consulting Engineers, for financial assistance.

Bridget Spalding for typing the manuscript, and to Charles Basson for producing copies of this thesis.

*Senior Lecturer, Department of Civil Engineering, University of Cape Town.

†Professor and Head of Department of Civil Engineering, University of Cape Town.
### NOTATION

#### UPPER CASE CHARACTERS

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A, B, C, D$</td>
<td>Bending stiffness matrices.</td>
</tr>
<tr>
<td>$D''''$</td>
<td>Fourth derivative of displacement vector $D$.</td>
</tr>
<tr>
<td>$D_x', D_y', D_{xy}$</td>
<td>Flexural rigidities for orthotropic plates.</td>
</tr>
<tr>
<td>$E$</td>
<td>Load vector for bending forces.</td>
</tr>
<tr>
<td>$E_x', E_y$</td>
<td>Young's modulus in $x$ and $y$ directions.</td>
</tr>
<tr>
<td>$G$</td>
<td>Modulus of elasticity in shear.</td>
</tr>
<tr>
<td>$J$</td>
<td>Load vector for in-plane forces.</td>
</tr>
<tr>
<td>$K, L, M$</td>
<td>In-plane stiffness matrices.</td>
</tr>
<tr>
<td>$M_x', M_y$, $M_{xy}$</td>
<td>Bending moments per unit length perpendicular to $x$ and $y$ axes respectively.</td>
</tr>
<tr>
<td>$M_{xy}$</td>
<td>Twisting moment per unit length perpendicular to $x$ axis.</td>
</tr>
<tr>
<td>$M_x', M_y', M_{xy}'$</td>
<td>Bending and twisting forces in local co-ordinate system.</td>
</tr>
<tr>
<td>$T$</td>
<td>Transformation matrix.</td>
</tr>
<tr>
<td>$U_p, U_b$</td>
<td>Total potential energy for in-plane and bending conditions respectively.</td>
</tr>
<tr>
<td>$\delta U_p, \delta U_b$</td>
<td>First variation of Total Potential Energy for in-plane and bending conditions respectively.</td>
</tr>
<tr>
<td>$\Delta U_p, \Delta U_b$</td>
<td>&quot;Part of&quot; expressions given above.</td>
</tr>
<tr>
<td>$X, Y, Z$</td>
<td>Rectangular co-ordinate axes.</td>
</tr>
<tr>
<td>$X_i', Y_i$</td>
<td>Edge forces on strip in $x$ and $y$ directions respectively.</td>
</tr>
</tbody>
</table>

#### LOWER CASE CHARACTERS

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_x', d_y', d_{xy}$</td>
<td>Modified Young's modulus values for in-plane forces.</td>
</tr>
<tr>
<td>$h$</td>
<td>Strip thickness.</td>
</tr>
<tr>
<td>$r_x', r_y', r_{xy}$</td>
<td>Radii of curvature in $x$ and $y$ directions and for twisting moment respectively.</td>
</tr>
<tr>
<td>$u, v, w$</td>
<td>Global displacements in $x$, $y$ and $z$ directions respectively.</td>
</tr>
</tbody>
</table>
GREEK CHARACTERS

\( \gamma_{xy} \)  
Shear strain.

\( \varepsilon_x, \varepsilon_y \)  
Strains in x and y directions respectively.

\( \theta \)  
Global rotation about x axis.

\( \sigma_x, \sigma_y \)  
Direct stresses in x and y direction respectively in global co-ordinates.

\( \sigma'_x, \sigma'_y \)  
Direct stresses in x and y directions respectively in local co-ordinates.

\( \tau_{xy} \)  
Shear stress in global co-ordinates.

\( \tau'_{xy} \)  
Shear stress in local co-ordinates.

\( \nu_x, \nu_y \)  
Poisson's ratio in x and y directions respectively.

\( \phi \)  
Angle strip makes with global axis system.
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synopsis</td>
<td>i</td>
</tr>
<tr>
<td>Declaration</td>
<td>ii</td>
</tr>
<tr>
<td>Dedication</td>
<td>iii</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>iv</td>
</tr>
<tr>
<td>Notation</td>
<td>v</td>
</tr>
</tbody>
</table>

### CHAPTER 1: Introduction

1

### CHAPTER 2: Governing Differential Equations for Finite Strip Analysis

- 2.1 In-plane stiffness of strip 4
- 2.2 Bending stiffness of strip 10
- 2.3 Combination of in-plane and bending forces 14
- 2.4 Transformation matrices 15
- 2.5 Finite difference approximations 17
- 2.6 Extension of equations for computer use 18
- 2.7 Steps in solution of stresses from displacements 21

### CHAPTER 3: Numerical Examples

- 3.1 Numerical examples 22
- 3.2 Discussion of results 35

### CHAPTER 4: Conclusions

- 4.1 Finite difference operator patterns 37
- 4.2 Computer program implications 39
- 4.3 Advantages over Fourier series method 39

### BIBLIOGRAPHY

41

### APPENDIX

A: In-Plane Stiffness Matrices 42
B: Bending Stiffness Matrices 43
C: Combined Matrices 45
D: Finite Difference Operator Patterns 49
E: Computer Program 53
F: User's Manual 100
CHAPTER I

INTRODUCTION

The literature on the analysis of folded plate structures may be considered to fall into seven principal categories with regard to method of analysis. They are:

1. Beam method.
2. Folded plate theory, neglecting relative joint displacements.
3. Folded plate theory considering relative joint displacements.
5. Finite difference method.
7. Finite strip method.

The Beam Method uses conventional beam theory, which requires all cross-sections to remain the same under load. If intermediate transverse stiffening ribs are frequent, this method yields good results, but more rigorous analyses and experiments have shown that this method cannot be applied to more intricate structures.

Methods (2) and (3) both consider the longitudinal supporting action of each plate to be governed by beam theory and the transverse supporting action to be that of a continuous one-way slab. Method (2) assumes that the changes in transverse bending moment and in longitudinal stresses due to relative joint displacements are negligible in comparison with the values of these moments and stresses computed on the basis of no relative joint displacements. For the general case however, these stresses are not negligible and the method is not recommended.

Folded Plate Theory Considering Relative Joint Displacements (3) takes into account the effect of relative displacement of the joints or the transverse moments and membrane stresses. Practical methods of analysis include those by V.Z. Vlasov, which uses a Fourier Series approximation; the Portland Cement Association; Gaafar, which uses the principle of superposition; and Yitzhaki who has also included the application of plasticity to folded plate structures.

The "Elasticity" Method, developed by Goldberg and Leve combines the equations of classical plate theory, for loads normal to the plane of the plates, and
the elasticity equations defining the plane stress problems, for loads in the plane of the plates. Applied loading is approximated by a Fourier Series.

Methods (1) to (4) are generally restricted to single spans with simple loading and edge conditions. (Reference [1] contains a detailed discussion on these methods.)

The application of Finite Difference Equations to shell analysis was performed in the 1950's, the advent of high speed digital computers making it possible to solve the resulting linear algebraic equations.

The method of Finite Elements was developed by many authors, and folded plate problems have been solved using triangular, rectangular, quadrilateral, isoparametric and thick-shell elements. With this method all possible boundary conditions are soluble [14].

A Finite Strip method of analysis was developed by Y.K. Cheung, first for flat plates [20] and then for folded plate structures [4]. In this method the structure is sub-divided into longitudinal strips which extend from one boundary to the other, making the method ideal for the solution of constant cross-section shell structures. Linear displacement functions are used for the in-plane displacements and cubic polynomials for the displacements normal to the strip surface. The displacements along the length of the strip are approximated by Fourier Series harmonics, likewise the strip loading. The Principle of Minimum Potential Energy is utilised to give the strip stiffness matrix in explicit form. The solution of the displacements is achieved in the normal manner.

Since Fourier Series utilisation becomes impractical for structures with more than three spans, and free boundary conditions cannot be used, du Preez [7] developed a general finite strip method which is capable of solving continuous structures with no boundary condition restriction. The interconnection of strips and other elements is also claimed.

Louw [8] subsequently developed a finite strip method of analysis for flat plates using finite difference approximations to the differential equations along the strip edges. Clamped and simple supported boundaries were considered.

This thesis is an extension of Louw's work and includes in-plane forces
enabling constant cross-section folded plates, closed box structures and translational shells to be analysed. Support conditions include simple, fixed and guided ends as well as continuity over interior supports.
CHAPTER 2

GOVERNING DIFFERENTIAL EQUATIONS FOR FINITE STRIP ANALYSIS

Figure 2.1 shows a typical finite strip element. The strip is of constant cross-sectional shape and has four degrees of freedom per edge, (i.e. $u_i$, $v_i$, $w_i$, $\theta_i$). The right-hand axis system, and the corresponding displacements are also shown.

Since in-plane, or membrane stresses, and bending stresses are not coupled, it is convenient to derive the equations for the two cases separately.

2.1 In-plane stiffness of strip

Consider the plate element in Figure 2.2 subjected to in-plane or membrane stresses as shown.
Strains in the \( X \), \( Y \) directions and the shear strain can be given by the following linear functions of the displacement gradients:

\[
\begin{align*}
\varepsilon_x &= \frac{\partial u}{\partial x} \\
\varepsilon_y &= \frac{\partial v}{\partial y} \\
\gamma_{xy} &= \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}
\end{align*}
\] (2.1)

Writing these equations in terms of stresses and including the effect of Poisson's Ratio:

\[
\begin{align*}
\sigma_x &= \frac{E x}{(1 - \nu_x \nu_y)} \left[ \varepsilon_x + \nu_y \varepsilon_y \right] \\
\sigma_y &= \frac{E y}{(1 - \nu_x \nu_y)} \left[ \varepsilon_y + \nu_x \varepsilon_x \right] \\
\tau_{xy} &= G \gamma_{xy}
\end{align*}
\] (2.2)

Consider the strip as shown in Figure 2.1. Let the in-plane displacements at edge \( i \) be \( u_i \) and \( v_i \), and those at edge \( j \) be \( u_j \) and \( v_j \). The displacement at any point on the strip surface in the \( u \) direction can be made up of linear combinations of \( u_i \) and \( u_j \).

Thus \( u = f_1(u_i) + f_2(u_j), \) where \( f_1 \) and \( f_2 \) are linear functions. Since \( u \) is a function of \( x \) and \( y \), it is convenient to separate their dependence so that \( u_i \) and \( u_j \) are functions of \( x \) only and \( f_1 \) and \( f_2 \), functions of \( y \) only.

Hence \( u(x,y) = f_1(y).u_i(x) + f_2(y).u_j(x). \)

At this stage, only \( y \) dependent functions need be given any specific form as the \( x \) dependent functions are dealt with using finite difference approximations.
Since $y$ increases from edge $i$ to edge $j$, and $u(x,y)$ must have the value of $u_i(x)$ at edge $i$ and $u_j(x)$ at edge $j$, $f_1(y)$ can be replaced by $(1 - y/b)$ and $f_2(y)$ by $(y/b)$,

i.e. $u(x,y) = (1 - \frac{y}{b})u_i(x) + \left(\frac{y}{b}\right)u_j(x)$.

By similar reasoning:

$v(x,y) = (1 - \frac{y}{b})v_i(x) + \left(\frac{y}{b}\right)v_j(x)$.

Dropping the $(x,y)$ post-script for $u$ and $v$ and expressing the displacements in matrix form:

$$\begin{bmatrix} u \\ v \end{bmatrix} = C_p \cdot \psi_p$$ (2.3)

where

$$C_p = \begin{bmatrix} (1 - \frac{y}{b}) & 0 & \left(\frac{y}{b}\right) & 0 \\ 0 & (1 - \frac{y}{b}) & 0 & \left(\frac{y}{b}\right) \end{bmatrix}$$ (2.4)

and

$$\psi_p = [u_i(x) v_i(x) u_j(x) v_j(x)]^T$$ (2.5)

(The subscript $p$ indicates in-plane consideration).

Let the equations in (2.1) be represented by one strain matrix:

$$\begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{bmatrix} = \begin{bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \end{bmatrix}$$ (2.6)

Calculating partial derivatives and substituting into (2.6):

$$\varepsilon = \begin{bmatrix} (1 - \frac{y}{b})u'_i(x) + (\frac{y}{b})u'_j(x) \\ (-\frac{1}{b})v'_i(x) + (\frac{1}{b})v'_j(x) \\ ((-\frac{1}{b})u'_i(x) + (\frac{1}{b})u'_j(x)) + ((1 - \frac{y}{b})v'_i(x) + (\frac{y}{b})v'_j(x)) \end{bmatrix}$$ (2.7)
The stresses in (2.2) can be represented by one stress matrix:

\[
\sigma = \begin{bmatrix}
\sigma_x \\
\sigma_y \\
\tau_{xy}
\end{bmatrix} = [D_p][\epsilon]
\]  

(2.8)

Where

\[
[D_p] = \begin{bmatrix}
\frac{E_x}{(1-\nu_{xy})} & \frac{\nu E_x}{(1-\nu_{xy})} & 0 \\
\frac{\nu E_x}{(1-\nu_{xy})} & \frac{E_y}{(1-\nu_{xy})} & 0 \\
0 & 0 & G
\end{bmatrix}
\]  

(2.9)

and \([\epsilon]\) is as given in (2.6).

The Total Potential Energy of a finite strip for in-plane forces can be written:

\[
U_p = \frac{h}{2} \int \int_{X,Y} (\sigma_x\epsilon_x + \sigma_y\epsilon_y + \tau_{xy}\gamma_{xy})dy\cdot dx \quad \text{(Internal Strain Energy)}
\]

\[
- \int_{X} (X_i u_i + Y_i v_i + X_j u_j + Y_j v_j)dx \quad \text{(Edge Forces)}
\]

\[
- \int \int_{X,Y} (X, u + Y, v)dy\cdot dx \quad \text{(Surface Forces)}
\]

(2.10)

In matrix notation:

\[
U_p = \frac{h}{2} \int \int_{X,Y} \epsilon^T [D_p] \epsilon \cdot dy\cdot dx - \int_{X} (u_i v_i u_j v_j) \left[ \begin{array}{c}
X_i \\
Y_i \\
X_j \\
Y_j
\end{array} \right] dx
\]

\[
- \int \int_{X,Y} (u v) \left[ \begin{array}{c}
X \\
Y
\end{array} \right] dx\cdot dy
\]

(2.11)
Equation (2.11) is expanded term by term in its three natural subdivisions. The following substitutions are also made to avoid unnecessary complication:

\[
\begin{align*}
\frac{\partial E}{\partial x} &= \frac{E_x}{(1 - \nu \nu_y)} \\
\frac{\partial E}{\partial y} &= \frac{E_y}{(1 - \nu \nu_y)} \\
\frac{\partial E}{\partial xy} &= \frac{\nu E}{(1 - \nu \nu_y)}
\end{align*}
\]

(2.12)

Expansion of Internal Strain Energy Term

Only an outline of the process is given as it involves excessive arithmetic.

(a) Expansion of \( D_p \cdot \varepsilon \).

(b) Evaluation of \( \varepsilon^T \).

(c) Multiplication of \( \varepsilon^T \cdot D_p \cdot \varepsilon \).

(d) Integrating with respect to \( y \), across the strip.

(e) Using the Principle of Minimum Potential Energy, the first variation of \( U_p \) with respect to the displacements involved \((u_i, v_i, u_j, v_j; u'_i, v'_i, u'_j, v'_j)\) must be calculated and equated to zero.

The first variation of \( U_p \) with respect to \( u_i \) will be a partial differentiation of \( U_p \) with respect to \( u_i \), which must be multiplied by the first variation of \( u_i \) itself, i.e. \( \delta u_i \).

The process is repeated for each of the displacements mentioned which have first variations \( \delta v_i, \delta u_j, \delta v_j, \delta u'_i, \delta v'_i, \delta u'_j \) and \( \delta v'_j \).

(f) The result of the process culminated in (e) above is then sorted into three matrices; a row vector involving the first variations of the displacements \((1 \times 8)\), a square non-symmetrical matrix involving the elements of equation (2.12), \( G \), and \( h \) as well as the constant \( b \) \((8 \times 8)\) and a column vector involving the displacements \((8 \times 1)\).

(g) The non-symmetrical \((8 \times 8)\) matrix is subdivided into four \((4 \times 4)\) matrices \((K, L, M \text{ and } N^T)\) as given in Appendix 'A'.
(h) Substituting (2.5) and grouping gives: (The \( \Delta \) prefix to \( \delta U_p \) is used to indicate a part of the total \( \delta U_p \)).

\[
\delta U_p = \int_X \left[ \delta \Psi_p^T K \Psi_p + M \Psi_p^T + \delta \Psi_p^T T_p \right] \delta \Psi_p dx
\]

\[(2.13)\]

**Expansion of Edge and Surface Force Terms**

From equation (2.11) the edge force term can be written:

\[
\Delta U_p = - \int_X \Psi_p^T \cdot \left[ \begin{array}{c} Y_i \\ Y_j \\ X_j \\ Y_j \end{array} \right] \cdot dX.
\]

Let the load vector above be represented by \([J]\). When using finite difference approximations to the differential equations \([J]\) becomes:

\[
[J] = \left[ \begin{array}{cccc} X_i & Y_i & X_i & Y_i \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 0 \end{array} \right]^T,
\]

where \(SA\) is the distance between nodes in the longitudinal direction.

The surface force terms are required, for analysis purposes, to be proportioned and placed on the adjoining strip edges. Thus all in-plane surface loads can be included in the \(J\) matrix above. Thus edge and surface forces may be represented by:

\[
\Delta U_p = - \int_X \Psi_p^T J dx.
\]

(2.14)

The first variation of \( \Delta U_p \) with respect to the displacements contained in \(\Psi_p^T\) is:

\[
\delta \Delta U_p = - \int_X \delta \Psi_p^T J dx.
\]

(2.15)

**Integration with Respect to the Variable \(X\)**

Equations (2.13) and (2.15) are added giving the total expression for the first variation of Total Potential Energy for one strip, i.e.

\[
\delta U_p = \int_X \left[ \delta \Psi_p^T K \Psi_p + M \Psi_p^T \right] + \delta \Psi_p^T T_p \Psi_p + L \Psi_p^T - \delta \Psi_p^T J \right] dx
\]

(2.16)

Integration of term involving \(\delta \Psi_p^T\) by parts
\[ \delta U_p = \delta \psi_p^T[M^T \psi_p + L \psi'_p] \bigg|_0^a + \int_0^a [\delta \psi_p^T[-(M^T \psi'_p + L \psi''_p) + (K \psi_p + M \psi'_p) - J] \, dx \]

\( \delta \psi_p \) is an arbitrary displacement vector which can be zero or any small finite quantity depending on the boundary conditions prescribed. For the Total Potential Energy to be a minimum, \( U_p \) is differentiated with respect to the generalised displacements, one at a time and each must be zero, hence:

\[ \delta \psi_p^T[M^T \psi_p + L \psi'_p] \bigg|_0^a = 0 \quad (2.17) \]

and

\[ \int_0^a [\delta \psi_p^T[-(M^T \psi'_p + L \psi''_p) + (K \psi_p + M \psi'_p) - J] \, dx = 0 \quad (2.18) \]

Equation (2.17) is a boundary condition equation which must be satisfied at \( x = 0 \) or \( x = a \). Equation (2.18) represents the strip section away from the boundaries where the displacement vector, \( \delta \psi_p \) is generally non-zero, and hence for equation (2.18) to be satisfied, the integrand must be zero, giving:

\[ \delta \psi_p^T[M^T \psi_p + L \psi'_p] \bigg|_0^a = 0 \quad (2.19) \]

\[ -L \psi''_p + (M - M^T) \psi'_p + K \psi_p - J = 0 \]

2.2 **BENDING STIFFNESS OF STRIP**

Consider a strip cross-section as shown in Figure 2.3 below:

![Figure 2.3](image)

All possible displacements of this cross-section which cause bending or twisting stresses can be represented by combinations of the following displacements. The appropriate cubic polynomial displacement functions are also given: \( (\psi) \)
As with the in-plane displacements, it is convenient to separate the x and y dependence and make the edge displacements functions of x alone. The displacement of any point on the strip surface is then given by the summation of the four displacements. In matrix form:

$$w = C_b u_b,$$  \hspace{1cm} (2.20)

where

$$C_b = \left[ \left( 1 - \frac{3y^2}{b^2} + \frac{2y^3}{b^3} \right) \left( y - \frac{2y^2}{b} + \frac{y^3}{b^2} \right) \left( \frac{3y^2}{b^2} - \frac{2y^3}{b^3} \right) \left( \frac{y}{b} + \frac{y^2}{b^2} \right) \right]$$ \hspace{1cm} (2.21)

and

$$u_b = \left[ w_1(x) \theta_1(x) w_j(x) \theta_j(x) \right]^T$$ \hspace{1cm} (2.22)

The curvature displacement relationship for thin plates may be written:

$$x = \left[ -\frac{\partial^2 w}{\partial x^2} \left| -\frac{\partial^2 w}{\partial y^2} \right| \frac{2\partial^2 w}{\partial x \partial y} \right]^T$$ \hspace{1cm} (2.23)

and the relationship between moment and curvature may be expressed as

$$M = D x$$ \hspace{1cm} (2.24)
where

\[
D = \begin{bmatrix}
D_x & D_y & 0 \\
D_y & 0 & D_{xy} \\
0 & 0 & D_{xy}
\end{bmatrix}
\]  

(2.25)

and has individual elements:

\[
D_x = \frac{E_x h^3}{12(1 - \nu_x \nu_y)} \\
D_y = \frac{E_y h^3}{12(1 - \nu_x \nu_y)} \\
D_{xy} = \frac{G h^3}{12} \\
\]

(2.26)

The Total Potential Energy for bending forces acting on a finite strip can be written: (the subscript b indicates bending consideration)

\[
U_b = \frac{1}{2} \iint_{X \cdot Y} \left( \frac{1}{x} M_x + \frac{1}{y} M_y + 2M_{xy} \right) \, dy \, dx - \iint_{X \cdot Y} w.p. \, dy \, dx - \int_{0}^{a} (M_{xx} \beta_y + M_{yy} \theta_x + Q.w) \, dx
\]

(2.27)

Rewriting equation 2.27 with the surface load expression included in the nodal load expression; and using matrix notation (Shear Strains have been neglected)

\[
U_b = \frac{1}{2} \iint_{X \cdot Y} X^T D \cdot X \, dy \, dx - \iint_{X \cdot Y} C_b u_b \cdot p \, dy \, dx
\]

or

\[
U_b = \frac{1}{2} \iint_{X \cdot Y} X^T D_x X \, dy \, dx - \iint_{X \cdot Y} u_b^T C_b^T p \, dy \, dx
\]

(2.28)

When use is made of finite differences, loading terms become simple manipulations which are performed when filling out the load matrix. Thus, \( C_b^T \cdot p \) can be replaced by a matrix \( E \), which at present requires no further definition.

**Expansion of Matrices**

As with the "in-plane" matrix expansion, the calculations involved in expanding
(2.28) are both tedious and voluminous. An outline of the process is: -

(a) Multiplication of \( D \) and \( x \).
(b) Evaluation of \( X^T \).
(c) Pre-multiplying the result from (a) by that from (b).
(d) Integrating with respect to \( y \) and evaluating the intergrand from 0 to \( b \).
(e) Differentiating with respect to each displacement present \( (w_i, \theta_i, w_j, \theta_j, w_i', \theta_i', w_j', \theta_j') \).
(f) Grouping the result to give:

\[
\delta U_b = \int_X (\delta u^T [A u'' + B u] + \delta u^T [C u'] + \delta u^T [B^T u'' + D u - E]) \, dx \tag{2.29}
\]

The matrices \( A, B, C \) and \( D \) are given in Appendix B.

Integrating by parts:

\[
\delta U_b = \left[ \delta u^T (A u'' + B u) \right]_a^b - \left[ \delta u^T [A u'' + B u'] \right]_a^b - \int_X \delta u^T (A u'' + B u') \, dx \\
+ \left[ \delta u^T C u' \right]_a^b - \int_X \delta u^T C u'' \, dx \\
+ \int_X \delta u^T (B^T u'' + D u - E) \, dx \tag{2.30}
\]

Grouping (2.30) as products of \( \delta u^T \) and \( \delta u^T \)

\[
\delta U_b = \left[ \delta u^T [A u'' + Bu] \right]_a^b - \left[ \delta u^T [Au'' + (B - C) u'] \right]_a^b \\
+ \int_X \delta u^T [Au'' + (B - C + B^T) u'' + Du - E] \, dx \tag{2.31}
\]

Using the same reasoning as in formulating equations (2.19), to satisfy the requirements that Total Potential Energy be a minimum:

\[
\delta u^T [A u'' + B u] \bigg|_o^a = 0 \\
\delta u^T [A u'' + (B - C) u'] \bigg|_o^a = 0 \tag{2.32}
\]

\[ A u'''' + (B - C + B^T) u'' + D u - E = 0 \]
2.3 COMBINATION OF IN-PLANE AND BENDING FORCES

The in-plane and bending stiffness matrices have been developed separately as there is no connection between the two systems of forces. In the combination of these two sets of forces, matrices are simply enlarged with a separate space for each system. This is shown in Figure 2.4 below.

All strip stiffness matrices are now \((8 \times 8)\), compared with \((4 \times 4)\) previously.

The governing differential equation for a single finite strip is composed of the summation of the last equations of (2.19) and (2.32).

In general terms, this summation can be expressed

\[ \alpha D'''' + \beta D''' + \gamma D'' + \varepsilon D' + \zeta D = \lambda \]  

(2.33)
where: $\alpha = A$
$\beta = 0$
$\gamma = (B - C + E^T) - (L)$
$\varepsilon = (M - M^T)$
$\zeta = (D + K)$
$\lambda = (E + J)$ (Appendix B)

The full $\alpha$, $\gamma$, $\varepsilon$, $\zeta$ and $\lambda$ matrices are simply an addition, as indicated in Figure 2.4, and are given in Appendix C.

With $\beta = 0$, (2.33) becomes:

$$\alpha D'''' + \gamma D'' + \varepsilon D' + \zeta D = \lambda$$

(2.35)

2.4 TRANSFORMATION MATRICES

Equation (2.35) is only suitable for solving flat plate problems. The Transformation Matrix derived below enables folded plate problems to be solved as well.
<table>
<thead>
<tr>
<th>No.</th>
<th>Name</th>
<th>Address</th>
<th>Age</th>
<th>Sex</th>
<th>Nationality</th>
<th>Occupation</th>
<th>Year of Study</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>John Smith</td>
<td>123 Main St</td>
<td>18</td>
<td>M</td>
<td>American</td>
<td>Student</td>
<td>2019</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Jane Doe</td>
<td>456 Elm St</td>
<td>19</td>
<td>F</td>
<td>Canadian</td>
<td>Teacher</td>
<td>2020</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Mike Brown</td>
<td>789 Maple Ave</td>
<td>20</td>
<td>M</td>
<td>Australian</td>
<td>Engineer</td>
<td>2021</td>
<td></td>
</tr>
</tbody>
</table>

Note: The table continues with additional entries.

For more information, please contact:

C.P. Renhall

Phone: 123-456-7890

Email: info@school.edu

Address: PO Box 12345, School City, USA
The angle \( \phi \) is measured from the main-axis direction to the local-axis direction. This angle is considered positive if clockwise and negative if anti-clockwise. The Y' axis is parallel to the strip and in the direction of increasing node number.

Consider Figure 2.6; the local axis displacements can be derived from global displacements: (Let local axis displacements be indicated by a prime)

\[
\begin{align*}
\nu' &= \nu \\
\nu' &= \nu \cos \phi + w \sin \phi \\
w' &= -\nu \sin \phi + w \cos \phi \\
\theta' &= \theta.
\end{align*}
\]

(2.36)

Or in matrix form:

\[
\begin{bmatrix}
u' \\ v' \\ w' \\ \theta'
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & \cos \phi & \sin \phi & 0 \\
0 & -\sin \phi & \cos \phi & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
u \\ v \\ w \\ \theta
\end{bmatrix} = R.D
\]

(2.37)

This is the Transformation Matrix for one edge of the strip; for both edges, the matrix is duplicated thus:

\[
T =
\begin{bmatrix}
R & 0 \\
- & - \\
0 & R
\end{bmatrix}
\]

(2.37)

Following the normal transformation procedures, equation (2.35) is written:

\[
T^T \alpha T D'' + T^T \gamma T D'' + T^T \varepsilon T D' + T^T \zeta T D = \lambda
\]

(2.38)

This equation makes possible the solution of folded plate problems. \( \lambda \) is the load matrix where nodal-global loads are substituted. The matrices \( D''', D'' \) and \( D' \) have global finite difference approximations to the differential equation along the length of the strip.
2.5 **FINITE DIFFERENCE APPROXIMATIONS**

The finite difference approximations to the differential equation along a nodal line are derived as for beam elements. All the boundary conditions required in equations (2.19) and (2.32) are satisfied by the use of standard Operator Patterns as given in Appendix D.

The three boundary conditions which occur in this type of structure are; rigidly fixed, simply supported and guided. Each of the four displacements \((u, v, w, \theta)\) must be given an operator pattern for each of the above cases.

The displacements \(v\) and \(w\) become interchanged for a strip on edge, and for any strip which makes an angle to the global co-ordinate system there is the corresponding reciprocation between these two displacements. Thus \(v\) and \(w\) displacements must always have the same operator patterns. The operator pattern for the \(\theta\) displacement has the same form as the \(w\) displacement.

The \(u\) displacement may be either fixed or free. The free operator pattern is derived for a simple rod element having the same values of displacement beyond the boundary as at the boundary node.

The three cases are as follows:

| Rigidly Fixed: | \(u\) | Fixed |
|               | \(v\) | Fixed |
|               | \(w\) | Fixed |
|               | \(\theta\) | Fixed |

| Simply Supported: | \(u\) | Free |
|                  | \(v\) | Simply Supported |
|                  | \(w\) | Simply Supported |
|                  | \(\theta\) | Simply Supported |

| Guided Support: | \(u\) | Fixed |
|                | \(v\) | Guided |
|                | \(w\) | Guided |
|                | \(\theta\) | Guided |

All nodes at the end of a structure are usually given the same boundary condition.
2.6 EXTENSION OF EQUATION (2.39) FOR COMPUTER USE

Consider the following folded plate structure shown in Figure 2.7.

![Figure 2.7](image)

Steps in the solution of nodal displacements are:

(a) Matrices $\alpha$, $\gamma$, $\varepsilon$ and $\zeta$ in equation (2.38) are evaluated.

(b) The transformation matrices and their transposes are calculated for each strip.

(c) Matrix multiplication yields the following condensed form of equation (2.38) for each strip:

$$(AL) D'' + (FA) D'' + (EP) D' + (ZE) D = \lambda$$

(d) To compile this equation for the complete cross-section the matrices are added where strip boundaries coincide as shown in Figure 2.8.

![Figure 2.8](image)
Notes: 1. Each small block represents a \((4 \times 4)\) matrix with corresponding \(u, v, w\), and \(\theta\) displacements.

2. The \((8 \times 8)\) matrix covering nodes 1 and 2 represents the first strip going from node 1 to 2, the second \((8 \times 8)\) block represents the second strip going from node 2 to 3, and so on.

3. The strip going from node 5 to 7 is broken into \((4 \times 4)\) matrices and is placed as shown by the shaded squares.

(e) The Finite Difference approximations run along the strip boundaries and always represent global displacements. The total equation to be solved is shown in Figure 2.9.

(f) The combined stiffness matrix is found by the addition of all components in the equation under step (c).

(g) The nodal loads are represented in vector form \((\lambda)\).

(h) When compiling the stiffness matrix \([K]\), the cross-section matrices are derived for the unit strip length, only a portion of these matrices must be considered. Thus the value \(S/(\text{Number of cross-sections})\) must multiply all the matrices \((AL, FA, EP, \text{and } ZE)\), where \(S\) is the nodal spacing.
Note: 1. Each shaded block in the AL, FA, EP, and ZE matrices are compiled as shown in Fig. 2.8.
2. The Derivative Matrices are sparsely populated and hence are more compactly stored in the computer analysis. The D matrix is a scalar matrix.
3. When matrix arithmetic is done the form of this equation is $[K][D] = [\lambda]$, where $K$ is a symmetric stiffness matrix.
2.7 STEPS IN SOLUTION OF STRESSES AND MOMENTS FROM DISPLACEMENTS

(a) All displacements are transformed into their respective local axis system:

\[
\begin{bmatrix}
D'_1 \\
D'_2
\end{bmatrix} = \begin{bmatrix}
T_{12} & 0 \\
0 & T_{12}
\end{bmatrix} \cdot \begin{bmatrix}
D_1 \\
D_2
\end{bmatrix}
\]

(2.39)

where the strip is envisaged as going from node 1 to node 2.

(b) The stresses and moments are then calculated by the following formulae:

\[
\sigma'_x = d \frac{\partial u}{\partial x} + d \frac{\partial v}{\partial y}
\]

\[
\sigma'_y = d \frac{\partial u}{\partial y} + d \frac{\partial v}{\partial x}
\]

\[
\tau'_{xy} = g \left( \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right)
\]

\[
M'_x = - \left[ D_x \frac{\partial^2 w}{\partial x^2} + D_1 \frac{\partial^2 w}{\partial y^2} \right]
\]

\[
M'_y = - \left[ D_1 \frac{\partial^2 w}{\partial x^2} + D_y \frac{\partial^2 w}{\partial y^2} \right]
\]

\[
M'_{xy} = D_{xy} 2 \frac{\partial^2 w}{\partial x \partial y}
\]

(2.40)
CHAPTER 3

NUMERICAL EXAMPLES

3.1 Numerical Examples

This chapter contains various numerical examples to illustrate the various displacement, stress and force results as obtained from the computer program.

The results of the first example are compared with those obtained by using conventional beam theory, while the more complicated examples are compared with the results obtained by other researchers in the same field.

EXAMPLE NO. 1:

This example was chosen to illustrate the three different boundary conditions discussed previously, as well as relevant stress and deflection curves.

Ex. 1a

Cross-section: Thick walled box section as shown.
Span: 12,000 m
Boundary conditions: Simply supported - simply supported.
Loading: Uniformly distributed load of 9,96 kN/m.
Young's modulus (X and Y): 31,0E09
Poisson's ratio (X and Y): 0,00
Program C.P.U. time: 1 min 6 sec.

FIG. 31
Shear lag at beam ends

Ex. 1b

Cross-section: As in example 1a.
Span: 12,000 m
Boundary conditions: Simply supported - Guided.
Loading: Uniformly distributed load of 9.96 kN/m.
Young's Modulus (X and Y): 31,0E09.
Poisson's Ratio (X and Y): 0.00.
Program C.P.U. time: 58 sec.

Vertical deflection
### Ex. 1c

<table>
<thead>
<tr>
<th>Cross-section</th>
<th>As in example 1a</th>
</tr>
</thead>
<tbody>
<tr>
<td>Span</td>
<td>12,000 m.</td>
</tr>
<tr>
<td>Boundary conditions</td>
<td>Fixed - Simply supported.</td>
</tr>
<tr>
<td>Loading</td>
<td>Uniformly distributed loading of 9,96 kN/m.</td>
</tr>
<tr>
<td>Young's Modulus (X and Y)</td>
<td>31,0E09.</td>
</tr>
<tr>
<td>Poisson's Ratio (X and Y)</td>
<td>0,00</td>
</tr>
<tr>
<td>Program C.P.U. time</td>
<td>1 min 8 sec.</td>
</tr>
</tbody>
</table>
Vertical deflection

![Diagram showing vertical deflection with data points and labels.](image)

**FIG. 3.8**

σ\_x on centre line of horizontal strips

![Diagram showing σ\_x with data points and labels.](image)

**FIG. 3.9**

u displacement curves (x direction)

![Diagram showing u displacement with data points and labels.](image)

**FIG. 3.10**
EXAMPLE NO. 2:

This example is an open cell, thin walled continuous structure. The ability of the theory and program to cater for an interior support is illustrated. Longitudinal stresses and transverse bending moments are compared with other published work.


Cross-section: As shown in Fig. 3.11.
Span: 24.38 m and 19.507 m.
Boundary conditions: Simply supported - Simply supported with interior support.
Loading: Uniformly distributed load of 3832 N/m².
Young's modulus (X and Y): 2,52443 E 10 N/m².
Poisson's ratio (X and Y): 0.00.
Program C.P.U. time: 2 min 21 sec.

Cross-section

[Diagram of cross-section with labels and dimensions]

Elevation

[Diagram of elevation with labels and dimensions]
Section AB has 2 strips
Section BC has 2 strips
Section CD has 4 strips
Section D to centre line has 2 strips
Only half the cross-section was analysed, the rotations and lateral displacements at the node on the centre line were zeroed.

Longitudinal distribution of transverse moment on fold line C

FIG. 3.13.
Transverse bending moment at section 3

\[ \text{kNm/m} \]

\( \Delta \) Beaufait
\( \bullet \) Scordelis & Lo

\( \sigma_x \) on foldline D

\[ \text{MPa} \]

\( \Delta \) Beaufait
\( \bullet \) Scordelis & Lo
Transverse distribution of longitudinal stress

FIG. 3.16

△ Beaufait
○ Scordelis & Lo
EXAMPLE NO. 3:

The structure shown is an open cell, thin walled translational shell which has members of two different thicknesses.


Cross-section : As shown in Fig. 3.

Span : 30,480 m

Boundary conditions : Simply supported - Simply supported.

Loading : Ridge Line Loading as shown in Fig. 3.17.

Young's modulus (X and Y) : 13,795 kN/m².

Poisson's ratio (X and Y) : 0.00.

Program C.P.U. time : 2 min 38 sec.

All plate elements were divided into 2 strips each. The C.P.U. time given is for the analysis of the complete structure. The program, however, is quite capable of analysing one quarter of such "two way symmetrical" structures. A time of 1 min 29 sec was recorded for such an analysis.
30.48
Elevation

FIG. 3.18

\( \sigma_x \) at midspan

FIG. 3.19

dotted: Elasticity theory
Ref: 16
Transverse moment at midspan

Plate membrane shear at support
Vertical deflection at midspan

FIG. 3.22
3.2 Discussion of Results

In example 1, the box beam webs and flanges were kept relatively thick so as to reduce the effect of secondary stresses and so make the comparison with simple beam theory realistic.

In all cases the deflection and bending stress curves were within acceptable limits. Fig. 3.4 shows that the method of analysis is capable of detecting shear lag, which is important in the analysis of thin walled box structures. Fig. 3.7 shows shear stresses in the beam near the supports. The computer analysis overestimates these stresses at the ends of the structure and it was found that extrapolating the results for nodes near the ends gives acceptable answers. Fig. 3.10 shows the form of the in-plane displacement for the propped cantilever case. It can be seen that the slope of this displacement with respect to the longitudinal axis is zero at the right hand support, fulfilling the requirement of zero stress at the boundary ($\frac{\partial u}{\partial x} = 0$).

Example 2 was compared with published work. It shows that the method is quite capable of dealing with internal supports and gives reasonably accurate results. The values, as plotted, indicate that there is a fairly large range between results obtained by Beaufait and those obtained by Scordelis and Lo.

Fewer nodes were used in the longitudinal direction than in the two references and thus it was not possible to compare results at exactly the same cross-sections. The number of cross-sections was limited by practical considerations and the maximum array size handled by the computer.

Fig. 3.13 shows that Beaufait's solution has a relatively high transverse moment over the internal support. (Cross-section 6). The exact nature of the support is not described in the literature, but Scordelis gives a very small bending moment at this section, indicating a diaphragm type support. The present analysis also makes use of a diaphragm type support, and is capable of simulating any type of internal fixity.

Fig. 3.14 shows the transverse bending moment at Section 3. The results agree reasonably well except at fold line B, where the bending moment is substantially more but still within acceptable design limits.
Example 3 shows essentially the same variables as example 2. The accuracy of the results, compared with published work, is also approximately the same.
CONCLUSIONS

This chapter contains general conclusions in respect of the method of analysis and its applications. The main points of interest which arose with the application of finite differences are discussed, as well as computer implications and advantages over the more usual Fourier Series Method are listed.

4.1 Finite Difference Operator Patterns

The patterns employed in the computer program are given in Appendix D. The following facts governed the choice of patterns and the resulting accuracy.

(a) The half-band width of the stiffness matrix is three times the number of degrees of freedom per cross-section. This is because five point central difference operator patterns were used for the fourth derivative, of which only the central and right hand values were actually used in the analysis. Thus these three values, which determine the stiffness matrix band-width, are the minimum required for the fourth derivative.

(b) The various strip stiffness matrices and their multipliers are as follows:

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Symmetry</th>
<th>Multiplier</th>
<th>Symmetry</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>Symmetrical</td>
<td>D''''</td>
<td>Symmetrical</td>
</tr>
<tr>
<td>γ</td>
<td>Symmetrical</td>
<td>D''</td>
<td>Symmetrical</td>
</tr>
<tr>
<td>ε</td>
<td>Skew-Symmetrical</td>
<td>D'</td>
<td>Skew-Symmetrical</td>
</tr>
<tr>
<td>ζ</td>
<td>Symmetrical</td>
<td>D</td>
<td>Symmetrical</td>
</tr>
</tbody>
</table>

It is noticed that every matrix that is symmetrical must be multiplied by a symmetrical multiplier and vice-versa. This dictates that no odd pattern may be inserted at the end of the structure to increase accuracy, as it would not be compatible with the rest of the multiplier matrix. The maximum number of values that are actually used in these multiplier matrices are limited to three, as more would give values outside the already rather broad band-width.
(c) The values in the scalar matrix \((D)\) may only be different from 1.0 at the extreme ends of the structure. This is because boundary conditions need only be satisfied at the ends of the structure.

(d) The in-plane \(u\) displacement boundary condition is satisfied by using the fact that the longitudinal stress is zero at the free edge. The longitudinal stress is given by:

\[
\sigma_x = d \frac{\partial u}{\partial x} + d \frac{\partial \nu}{\partial y}.
\]

Since the second term makes only a small contribution, and none at all when \(\nu = 0\), the stress may be approximated by:

\[
\sigma_x \approx d \frac{\partial u}{\partial x}.
\]

At the free edge, \(\sigma_x = 0\), which implies that \(\partial u / \partial x\) must be zero. This is satisfied by the following displacement model;

![Diagram showing displacement model](image)

(e) The final stresses are calculated from the displacements obtained, and it was found that small differences in corresponding displacements resulted in unacceptably large differences in stresses. Thus it was absolutely necessary to use operator patterns which gave very good symmetrical displacements.

(f) External reactions and other forces not given in the computer analysis can be readily calculated using the correct finite difference operator pattern together with the displacements given.
4.2 Computer Program Implications

(a) The computer program, as written, uses a large amount of core storage because of the necessarily broad band width of the stiffness matrix. However, the solution time compares very favourably with those given in the references listed. (This may be because the computer used, UNIVAC 1106, is a relatively high speed machine).

(b) A small amount of core storage is required for the CA, CG, CE and CZ matrices, and the finite difference values are stored as efficiently as possible. This method can hence be used on a fairly small core machine if the formation of the stiffness matrix is altered so that blocks of figures are transferred to disc storage and are later recalled as required.

(c) Tapering members can be given an equivalent thickness which corresponds to an identical second moment of area.

4.3 Advantages over Fourier Series Method

The method of using finite differences together with the normal finite strip concept has distinct advantages over the use of Fourier Series. These are:

(a) The global displacements are solved directly, as compared with the Fourier Series Method where substantially smaller calculations are done repeatedly and added, giving the required displacements. Up to seventy, and more, harmonics are used in the papers referred to.

(b) Point loadings are distributed over one nodal length, this will give more accurate answers than using a number of Fourier Series harmonics.

(c) Different boundary conditions are employed without any difficulty and are relatively easily understood.

(d) Interior supports along the length of a structure may be as numerous as desired. Fourier Series becomes impractical for more than three spans.

(e) This method may also be used for strips of non-uniform thickness in the longitudinal direction, i.e. cut-outs and the change of thickness
of members over supports. This simply leads to the necessity of calculating and storing CA, CG, CE and CZ matrices for as many different cross-sections as are required.


**APPENDIX A:**

**STIFFNESS SUB-MATRICES (In-Plane Forces)**

\[
[K] = \begin{bmatrix}
G \frac{h}{b} & 0 & -G \frac{h}{b} & 0 \\
0 & d_{y} \frac{h}{b} & 0 & -d_{y} \frac{h}{b} \\
-G \frac{h}{b} & 0 & G \frac{h}{b} & 0 \\
0 & -d_{y} \frac{h}{b} & 0 & d_{y} \frac{h}{b}
\end{bmatrix}
\]

\[
[L] = \begin{bmatrix}
d_{x} \frac{bh}{3} & 0 & d_{x} \frac{bh}{6} & 0 \\
0 & G \frac{bh}{3} & 0 & G \frac{bh}{6} \\
d_{x} \frac{bh}{6} & 0 & d_{x} \frac{bh}{3} & 0 \\
0 & G \frac{bh}{6} & 0 & G \frac{bh}{3}
\end{bmatrix}
\]

\[
[M] = \begin{bmatrix}
0 & -G \frac{h}{2} & 0 & -G \frac{h}{2} \\
-d_{xy} \frac{h}{2} & 0 & -d_{xy} \frac{h}{2} & 0 \\
0 & G \frac{h}{2} & 0 & G \frac{h}{2} \\
d_{xy} \frac{h}{2} & 0 & d_{xy} \frac{h}{2} & 0
\end{bmatrix}
\]
APPENDIX B:

STIFFNESS SUB-MATRICES (Bending Forces)

\[
[A] = \begin{bmatrix}
\frac{13b}{x} & \frac{11b^2}{x} & \frac{9b}{x} & -\frac{13b^2}{x} \\
\frac{35}{210} & \frac{210}{70} & \frac{70}{420} & \frac{420}{840} \\
\frac{11b^2}{x} & \frac{b^3}{x} & \frac{13b^2}{x} & -\frac{b^3}{x} \\
\frac{210}{105} & \frac{105}{420} & \frac{420}{35} & \frac{35}{140} \\
\frac{9b}{x} & \frac{13b^2}{x} & \frac{13b}{x} & -\frac{11b^2}{x} \\
\frac{70}{420} & \frac{420}{35} & \frac{35}{210} & \frac{210}{105} \\
\frac{13b^2}{x} & \frac{b^3}{x} & \frac{11b^2}{x} & \frac{b^3}{x} \\
\frac{420}{140} & \frac{140}{210} & \frac{210}{105} & \frac{105}{420}
\end{bmatrix}
\]

\[
[B] = \begin{bmatrix}
-\frac{6D_1}{5b} & \frac{11D_1}{10} & \frac{6D_1}{5b} & \frac{D_1}{10} \\
-\frac{D_1}{10} & -\frac{2bD_1}{15} & \frac{D_1}{10} & \frac{bD_1}{30} \\
\frac{6D_1}{5b} & \frac{D_1}{10} & \frac{6D_1}{5b} & \frac{11D_1}{10} \\
-\frac{D_1}{10} & \frac{bD_1}{30} & \frac{D_1}{10} & -\frac{2bD_1}{15}
\end{bmatrix}
\]

\[
[C] = \begin{bmatrix}
\frac{24D_{xy}}{5b} & \frac{4D_{xy}}{10} & -\frac{24D_{xy}}{5b} & \frac{4D_{xy}}{10} \\
\frac{4D_{xy}}{10} & \frac{8bD_{xy}}{15} & \frac{4D_{xy}}{10} & -\frac{4bD_{xy}}{30} \\
-\frac{24D_{xy}}{5b} & \frac{4D_{xy}}{10} & -\frac{24D_{xy}}{5b} & \frac{4D_{xy}}{10} \\
\frac{4D_{xy}}{10} & \frac{4bD_{xy}}{30} & \frac{4D_{xy}}{10} & \frac{8bD_{xy}}{15}
\end{bmatrix}
\]
\[
[D] = \begin{bmatrix}
\frac{12D_x}{b^3} & \frac{6D_y}{b^2} & -\frac{12D_x}{b^3} & \frac{6D_y}{b^2} \\
\frac{6D_x}{b^2} & \frac{4D_y}{b} & -\frac{6D_x}{b^2} & \frac{2D_y}{b} \\
-\frac{12D_x}{b^3} & -\frac{6D_y}{b^2} & \frac{12D_x}{b^3} & -\frac{6D_y}{b^2} \\
\frac{6D_x}{b^2} & \frac{2D_y}{b} & -\frac{6D_x}{b^2} & \frac{4D_y}{b}
\end{bmatrix}
\]

\[
[A] = \begin{bmatrix}
N & u & \text{(Force)} \\
N/m & v & \text{(Force per unit length)} \\
N/m & w & \text{(Force per unit length)} \\
Nm/m & \theta & \text{(Moment per unit length)}
\end{bmatrix}
\]
APPENDIX C:

\[ \text{\(\alpha, \gamma, \varepsilon\) and \(\zeta\) MATRICES} \]

**THE \(\alpha\) MATRIX**

\[
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{13b^2 D}{35} & \frac{11b^2 D}{210} & 0 & 0 & \frac{9b D}{70} & -\frac{13b^2 D}{420} \\
0 & 0 & \frac{11b^2 D}{210} & \frac{b^3 D}{105} & 0 & 0 & \frac{13b^2 D}{420} & -\frac{b^3 D}{140} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{9b D}{70} & \frac{13b^2 D}{420} & 0 & 0 & \frac{13b D}{35} & -\frac{11b^2 D}{210} \\
0 & 0 & \frac{13b^2 D}{420} & \frac{b^3 D}{140} & 0 & 0 & \frac{11b^2 D}{210} & -\frac{b^3 D}{105} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]
<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>- $\frac{h \cdot b}{\hbar}$</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>- $\frac{h \cdot b}{\hbar}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>- $\frac{h \cdot b}{\hbar}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>$-\frac{6D}{5b}$</td>
<td>$\frac{24D_{xy}}{5b}$</td>
<td>$\frac{11D_{1}}{10}$</td>
<td>$\frac{4D_{xy}}{10}$</td>
<td>0</td>
<td>0</td>
<td>$\frac{6D_{1}}{5b} + \frac{24D_{xy}}{5b}$</td>
</tr>
<tr>
<td>0</td>
<td>$\frac{6D_{1}}{5b}$</td>
<td>$\frac{24D_{xy}}{5b}$</td>
<td>$\frac{11D_{1}}{10}$</td>
<td>$\frac{4D_{xy}}{10}$</td>
<td>0</td>
<td>0</td>
<td>$\frac{6D_{1}}{5b} + \frac{24D_{xy}}{5b}$</td>
</tr>
<tr>
<td>0</td>
<td>$\frac{11D_{1}}{10}$</td>
<td>$\frac{4D_{xy}}{10}$</td>
<td>$\frac{8bD_{xy}}{15}$</td>
<td>0</td>
<td>0</td>
<td>$\frac{D_{1}}{10} + \frac{4D_{xy}}{10}$</td>
<td>$\frac{bD_{1}}{30} + \frac{4bD_{xy}}{30}$</td>
</tr>
<tr>
<td>0</td>
<td>$\frac{6D_{1}}{5b}$</td>
<td>$\frac{24D_{xy}}{5b}$</td>
<td>$\frac{11D_{1}}{10}$</td>
<td>$\frac{4D_{xy}}{10}$</td>
<td>0</td>
<td>0</td>
<td>$\frac{6D_{1}}{5b} + \frac{24D_{xy}}{5b}$</td>
</tr>
<tr>
<td>0</td>
<td>$\frac{11D_{1}}{10}$</td>
<td>$\frac{4D_{xy}}{10}$</td>
<td>$\frac{8bD_{xy}}{15}$</td>
<td>0</td>
<td>0</td>
<td>$\frac{D_{1}}{10} + \frac{4D_{xy}}{10}$</td>
<td>$\frac{bD_{1}}{30} + \frac{4bD_{xy}}{30}$</td>
</tr>
<tr>
<td>0</td>
<td>$\frac{6D_{1}}{5b}$</td>
<td>$\frac{24D_{xy}}{5b}$</td>
<td>$\frac{11D_{1}}{10}$</td>
<td>$\frac{4D_{xy}}{10}$</td>
<td>0</td>
<td>0</td>
<td>$\frac{6D_{1}}{5b} + \frac{24D_{xy}}{5b}$</td>
</tr>
<tr>
<td>0</td>
<td>$\frac{11D_{1}}{10}$</td>
<td>$\frac{4D_{xy}}{10}$</td>
<td>$\frac{8bD_{xy}}{15}$</td>
<td>0</td>
<td>0</td>
<td>$\frac{D_{1}}{10} + \frac{4D_{xy}}{10}$</td>
<td>$\frac{bD_{1}}{30} + \frac{4bD_{xy}}{30}$</td>
</tr>
</tbody>
</table>
## THE \( \varepsilon \) MATRIX

\[
\begin{array}{cccc|cccc}
0 & -\frac{hG}{2} + \frac{h_d}{2} x & 0 & 0 & 0 & \frac{hG}{2} - \frac{h_d}{2} x & 0 & 0 \\
\frac{hG}{2} - \frac{h_d}{2} x & 0 & 0 & 0 & -\frac{hG}{2} + \frac{h_d}{2} x & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{hG}{2} + \frac{h_d}{2} x & 0 & 0 & 0 & \frac{hG}{2} - \frac{h_d}{2} x & 0 & 0 \\
\frac{hG}{2} + \frac{h_d}{2} x & 0 & 0 & 0 & -\frac{hG}{2} + \frac{h_d}{2} x & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]
## THE \( \zeta \) MATRIX

<table>
<thead>
<tr>
<th>( \zeta h )</th>
<th>( 0 )</th>
<th>( 0 )</th>
<th>( 0 )</th>
<th>( - \zeta h )</th>
<th>( 0 )</th>
<th>( 0 )</th>
<th>( 0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \zeta b )</td>
<td>( \frac{d}{b} \frac{h}{b} )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( \frac{d}{b} \frac{h}{b} )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( 0 )</td>
</tr>
<tr>
<td>( \zeta b )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( \frac{12d}{b^3} \frac{y}{b^3} )</td>
<td>( \frac{6d}{b^2} \frac{y}{b^2} )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( - \frac{12d}{b^3} \frac{y}{b^3} )</td>
</tr>
<tr>
<td>( \zeta b )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( \frac{6d}{b^2} \frac{y}{b^2} )</td>
<td>( \frac{4d}{b} \frac{y}{b} )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( - \frac{6d}{b^2} \frac{y}{b^2} )</td>
</tr>
<tr>
<td>( \zeta b )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( \zeta h )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( 0 )</td>
</tr>
<tr>
<td>( \zeta b )</td>
<td>( - \frac{d}{b} \frac{h}{b} )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( \frac{d}{b} \frac{h}{b} )</td>
<td>( 0 )</td>
<td>( 0 )</td>
</tr>
<tr>
<td>( \zeta b )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( - \frac{12d}{b^3} \frac{y}{b^3} )</td>
<td>( - \frac{6d}{b^2} \frac{y}{b^2} )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( \frac{12d}{b^3} \frac{y}{b^3} )</td>
</tr>
<tr>
<td>( \zeta b )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( \frac{6d}{b^2} \frac{y}{b^2} )</td>
<td>( \frac{2d}{b} \frac{y}{b} )</td>
<td>( 0 )</td>
<td>( 0 )</td>
<td>( - \frac{6d}{b^2} \frac{y}{b^2} )</td>
</tr>
</tbody>
</table>
APPENDIX D:

FINITE DIFFERENCE OPERATOR PATTERNS

Note: 1. SA is the spacing between cross-section nodes.
2. Central difference formulae are used.

i.e. \( \frac{dy}{dx} = \frac{1}{2SA} [-1 y_L + 0 y_o + 1 y_R] \)
\( \frac{d^2y}{dx^2} = \frac{1}{SA^2} [1 y_L - 2 y_o + 1 y_R] \)
\( \frac{d^4y}{dx^4} = \frac{1}{SA^4} [1 y_{LL} - 4 y_L + 6 y_o - 4 y_R + 1 y_{RR}] \)

4th DERIVATIVE (+ SA^4)

Simply Supported Ends

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>0 5 -4 1</td>
<td>0 -4 6 -4 1</td>
<td>1 -4 6 -4 1</td>
<td>1 -4 6 -4 1</td>
<td>1 -4 6 -4 1</td>
</tr>
<tr>
<td>0 5 -4 1</td>
<td>0 -4 6 -4 1</td>
<td>1 -4 6 -4 1</td>
<td>1 -4 6 -4 1</td>
<td>0 0 0</td>
<td></td>
</tr>
</tbody>
</table>

Fixed Ends

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>0 7 -4 1</td>
<td>0 -4 6 -4 1</td>
<td>1 -4 6 -4 1</td>
<td>1 -4 6 -4 0</td>
<td>1 -4 7 0</td>
</tr>
<tr>
<td>0 7 -4 1</td>
<td>0 -4 6 -4 1</td>
<td>1 -4 6 -4 1</td>
<td>1 -4 6 -4 0</td>
<td>0 0 0</td>
<td></td>
</tr>
</tbody>
</table>

Guided Ends

| 3 -4 1   | -4 7 -4 1 | 1 -4 6 -4 1 | 1 -4 6 -4 1 | 1 -4 6 -4 1 | 1 -4 3 |
| -4 7 -4 1 | 1 -4 6 -4 1 | 1 -4 6 -4 1 | 1 -4 6 -4 1 | 1 -4 3 |
2nd DERIVATIVE ($\frac{d}{dS}A^2$)

Simply Supported Ends

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-2</td>
</tr>
<tr>
<td>1</td>
<td>-2</td>
</tr>
</tbody>
</table>

Fixed Ends

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-2</td>
</tr>
<tr>
<td>1</td>
<td>-2</td>
</tr>
</tbody>
</table>

Guided Ends

<table>
<thead>
<tr>
<th>-1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-2</td>
</tr>
<tr>
<td>1</td>
<td>-2</td>
</tr>
</tbody>
</table>

Function $u$: Fixed Ends

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-2</td>
</tr>
<tr>
<td>1</td>
<td>-2</td>
</tr>
</tbody>
</table>

Function $u$: Free Ends

<table>
<thead>
<tr>
<th>-1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-2</td>
</tr>
<tr>
<td>1</td>
<td>-2</td>
</tr>
</tbody>
</table>
The following Finite Difference Operator patterns are used in Sub-routine ECHO to determine stresses from deflections.
2nd DERIVATIVE \((\pm SA^2)\)

\[
\begin{array}{ccc}
2 & -5 & 4 & -1 \\
1 & -2 & 1 \\
-1 & 4 & -5 & 2 \\
\end{array}
\]

- Forward Difference
- Central Difference
- Backward Difference

1st DERIVATIVE \((\pm 2 SA)\)

\[
\begin{array}{ccc}
-5 & 4 & -1 \\
-1 & 0 & 1 \\
1 & -4 & 3 \\
\end{array}
\]

- Forward Difference
- Central Difference
- Backward Difference
APENDIX E:

FORTRAN V COMPUTER PROGRAM

*MENAI*

*The strait over which Robert Stephenson built the Britannia Tubular Bridge in 1850.*
MENAI SUBROUTINE SEQUENCE

MAIN
  ↓
DEES
  ↓
FORMA
  ↓
ANGLE
  ↓
CONTAC
  ↓
SILO
  ↓
WHEEL
  ↓
GRAFT
  ↓
ECHO
PROJECT = MENAI(I).HATN

1  
2  COMPI.R(XM=3)
3  
4  **********program to solve translational shell structures**************
5  THE FINITE STRIP METHOD OF ANALYSIS IS USED
6  CUBIC POLYNOMIAL DISPLACEMENT FUNCTIONS ARE USED ACROSS THE STRIPS
7 FOR THE NORMAL DISPLACEMENTS AND THE IN-PLANE DISPLACEMENTS ARE
8 V A R I E D LINEARLY
9 F IN IT E DIFFERENCE APPROXIMATIONS TO THE DIFFERENTIAL EQUATIONS
10 ARE USED ALONG THE LENGTH OF THE STRIPS • UNITS ARE ALWAYS M, N, S
11
12 COMMON /XM/SKU(144:133) * DO4CA(44:45) * SDO2C(44:43) * SDO2E(44)
13 .2CE(44:43) * CA(44:44) * CG(44:44) * CE(44:44)
14 .Z(44:44) * SIGMAX(180) * SIGMAX(180) * SIGMAX(152)
15 .MX(180) * APX(180) * APPX(180)
16 COMMON DOX(30), DOY(30), DO(30), DON(30), DONXY(30)
17 COMMON FA(8,9:3), FG(8,8:3), FE(8,8:3), FZ(8,8:3)
18 COMMON AF(3,9:3), FG(9,9:3), AFZ(3,3:30), AF7(9,9:3), AFZ(9,9:3)
19 COMMON SD(6,5:6), SDO(6,5:6), SDO1(6,5:6), SDO1(6,5:6)
20 COMMON DEX, DEY, DPY, DQ, DNUM, DLOX, DLOXY, DLOY
21 COMMON MHC, MSKB, MST, MSR, MDC (250)
22 COMMON NW27, JENOX, JENUX, MS(31), ME(31)
23 COMMON ANG(31), SA
24 COMMON /82/ATRAN
25 DOUBLE PRECISION ATTRANS(E,8:3)
26 MHC=44
27 MKD=44
28 NW27=132
29 DIMENSION E(72)
30 PRINT 541
31 PRINT 541
32 541 FORMAT(1HI=E(/1,1H*CIX,640F(***))/***
33 .31X**13X*!H*IX,64(**)***/
34 .31X**13X*TRANSLATIONAL SHELL PROGRAM**11X***/
35 .31X**13X*TRANSLATIONAL SHELL PROGRAM**11X***/
36 .31X**13X*GENERAL DESCRIPTION OF STRUCTURE:**************
37 .31X**13X*GENERAL DESCRIPTION OF STRUCTURE:**************
38 .31X**13X*GENERAL DESCRIPTION OF STRUCTURE:**************
39 .31X**13X*GENERAL DESCRIPTION OF STRUCTURE:**************
40 PRINT 567
41 PRINT 567
42 567 FORMAT(*30X**53X***/
43 .31X**3X*GENERAL DESCRIPTION OF STRUCTURE:**************
44 .31X**3X*GENERAL DESCRIPTION OF STRUCTURE:**************
45 .31X**3X*GENERAL DESCRIPTION OF STRUCTURE:**************
46 .31X**3X*GENERAL DESCRIPTION OF STRUCTURE:**************
47 PRINT 566
48 PRINT 566
49 566 FORMAT(*30X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
50 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
51 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
52 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
53 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
54 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
55 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
56 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
57 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
58 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
59 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
60 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
61 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
62 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
63 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
64 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
65 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
66 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
67 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
68 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
69 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
70 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
71 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
72 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
73 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
74 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
75 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
76 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
77 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
78 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
79 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
80 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
81 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
82 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
83 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
84 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
85 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
86 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
87 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
88 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
89 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
90 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
91 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
92 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
93 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
94 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
95 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
96 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
97 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
98 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
99 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
100 .31X**3X*LOADING CASE CONSIDERED:JC(*,*)12X***/
* YOUNG'S MODULUS Y:='*38*E16.6/*
* POISSON RATIO X:'*12X*DPFE.3/*
* POISSON RATIO Y:'*12X*DPFE.3/*
* SHEAR MODULUS G:='*8X*DPFE.6/*

FORMULATION OF THE BASIC STIFFNESS MATRICES

CALL FORMA

TO COMPILE THE TRANSFORMATION MATRICES FOR INDIVIDUAL STRIPS
AND TO COMPILE GLOBAL STRIP STIFFNESS MATRICES

CALL ANGLE

PRINT 542

543 FORMAT(1HC,*STRIP NO  STRIP WIDTH  STRIP THICKNESS  STRIP ANGLE[RAD]*,/*,
621(-'*)
DO 544 I=1,IDNUM
544 WRITE(I5,545) I,DCB(I),MCH(I),ANC(I)
545 FORMAT(1X*12X*F6.3,12X*F6.3,9X*F10.7)

TO ESTABLISH THE COMBINED STIFFNESS MATRICES FOR THE TOTAL
CROSS-SECTION

CALL CONTAC

TO COMPILE THE FINITE DIFFERENCE EQUATIONS, MULTIPLY WITH CEC2
CEC2 MATRICES AND COMBINE TO FORM UPPER TRIANGULAR MATRIX SKU

CALL SILO

IF(MS(1).EQ.1) PRINT 570
IF(MS(1).EQ.3) PRINT 571
IF(MS(1).EQ.5) PRINT 572
IF(MS(1).EQ.7) PRINT 573
570 FORMAT(1HC,*BOUNDARY CONDITION AT START IS FIXED*)
571 FORMAT(1HC,*BOUNDARY CONDITION AT START IS GUIDED*)
572 FORMAT(1HC,*BOUNDARY CONDITION AT START IS FREE*)
573 FORMAT(1HC,*BOUNDARY CONDITION AT START IS SIMPLY SUPPORTED*)
574 IF(ME(1).EQ.1) PRINT 575
575 FORMAT(1HC,*BOUNDARY CONDITION AT END IS FIXED*)
576 FORMAT(1HC,*BOUNDARY CONDITION AT END IS GUIDED*)
577 FORMAT(1HC,*BOUNDARY CONDITION AT END IS FREE*)
578 FORMAT(1HC,*BOUNDARY CONDITION AT END IS SIMPLY SUPPORTED*)
579 WRITE(5*5301) S
580 FORMAT(1HC,*NODAL SPACING =**F6.3)
581 FORMULATION OF LOAD MATRIX

CALL WHEEL

TO SOLVE DISPLACEMENTS

CALL GRAF

PRINT 555
565 FORMAT(1HC,*CROSS-SECTION**6X,**OCES**29Y,**GLOBAL DISPLACEMENTS
*")

PRINT 546

546 FORMAT(1HC13X*U*19X*V*14X*W*12X*THETA(RAD*)

PRINT 568

568 FORMAT(1H ,8G(16.1))

C NUMBER OF CROSS-SECTIONS

KA10=3+SKD/MKB

C NUMBER OF NODES PER SECTION

KA11=SKB/12

C DEGREES OF FREEDOM PER SECTION

KA16=SKB/3

DO 549 K=1,KA10

WRITE(5,547) K

547 FORMAT(1H *SGXR13)

DO 549 L=1,KA11

KA12=(K-1)*KA16+L*4-3

KA17=KA12+1

KA14=KA13+1

KA18=KA14+1

WRITE(5,548) L,SKU(KA12,NW27),SKU(KA13,NW27),

548 FORMAT(1H ,20X*IZ*7X*PE12*6*6X*E12*6*6X*E12*6)

549 CONTINUE

C

C TO SOLVE STRESSES IN LOCAL AXES CO-ORDINATES

C

CALL ECHO (EDV*JE1)

C

PRINT 560

560 FORMAT(1H ,"CROSS-SECTION SUB-STRUCTURE NODE*35X*LOCAL STRESS")

PRINT 551

551 FORMAT(1H ,41X,"SIGMA-X"*7X,"SIGMA-Y"*6X,"TAU-XY")

552 FORMAT(1H ,12X,"H-X"*9X,"H-Y"*8X,"M-XY")

PRINT 569

569 FORMAT(1H ,11I(-1))

555 I=1,JENOX

WRITE(5,552) I

552 FORMAT(1HC16X*12)

555 J=1,JENUM

WRITE(5,553) J

553 FORMAT(1H ,21X*12)

555 K=1,3

JE201=(I-1)*JENUM+J*3-Z*K-1

WRITE(5,554) K,SIGMAX(JE201),SIGMAY(JE201),TAUXY(JE201),

554 FORMAT(1H ,32X*12*6X*10*11.5*2X*E11.5*2X*E11.5,

*2X*E11.5)

CONTINUE

STOP

END
PROJECT "XENAI(1).DEES

1 COMPLl(CXH=3)
2 SUBROUTINE DEES
3 COMMON/CT/SKl(44,C,133),SD4CA(44,C,5),SD2G(44,C,3),SDGGE(44,C)
4 +4,SDCE(44,C,3),CA(44,C,44),CC(44,C,44),CE(44,C,44)
5 +CZ(44,C,44),SIGMA(180),SIGMAY(180),TAUXY(180)
6 +AMX(180),AMY(180),AMXY(180)
7 COMMON DXY(3D),DYY(3D),D01(3D)+DXY(3D),DNY(3D),DBI(3D)
8 COMMON FAY(8,8,3C),FE(6,6,3C),FF(8,8,3C),FZ(8,8,3C)
9 COMMON AFA(8,8,3C),AFE(8,8,3C),AFZ(8,8,3C)
10 COMMON SD4(E,5),SD2(E+3,6),SC1(E+3,6),SC16(6)
11 COMMON MHC,MSC,MSX,MSZ,MSC(M3),MSC(M3)
12 COMMON VM27,VM31,VM31,VM31,VM31
13 COMMON ANG(ZC),SA
14 COMMON ANG(ZC),SA
15 C READ EX + EY * POISSON X + POISSON Y
16 READ (8,100) DEY + DPY + CPY
17 100 FORMAT ()
18 C READ NUMBER OF STRIPS IN STRUCTURE IDNUM
19 READ(9,101) IDNUM
20 101 FORMAT ()
21 C READ STRIP THICKNESS AND WIDTH
22 DP 102 I = 1, IDNUM
23 102 READ (8,103) DII(I) + DBI(I)
24 103 FORMAT ()
25 C IN ORDER OF CALCULATION*SHEAR MODULUS, LITTLE(RX, DXY, N)
26 C CAPITAL(DX, DY, DZ, DXY)
27 DO = (DEX+DEY)/(4.*1.*(DPX+DPY)/2.*)
28 DLDX = DEX/(1.-CPX*DPY)
29 DLDX = DEx+Dey/(1.-Pc*DPY)
30 DLDY = DEX/DY/(1.-DPX*DPY)
31 D* IC4 I = 1, IDNUM
32 DEX(I) = (DLDX=DII(I)**2.1/12.
33 DDDI(I) = (DLDY=DII(I)**3.3/12.
34 DC1(I) = (DLDY=DII(I)**3.3/12.
35 DXY(I) = (DLDX=DII(I)**2.1/12.
36 CONTINUE
37 RETURN
38 END
FORMULATION OF STIFFNESS MATRICES. THE FA, FG, FE, FZ MATRICES ARE

DO 1000 I=1,IDNUM

C THE ALPHA MATRIX ***********************************************

FA(3,3,I) = 13.*D3(I)*DDX(I)/75.
FA(4,4,I) = DB(I)*DDX(I)/15.
FA(5,5,I) = 13.*D2(I)*DDX(I)/150.
FA(6,6,I) = DB(I)*DDX(I)/15.
FA(7,7,I) = 11.*DB(I)*DDX(I)/210.
FA(8,8,I) = FA(4,3,I).
FA(9,9,I) = -11.*DB(I)*DDX(I)/210.
FA(10,10,I) = FA(6,7,I).
FA(11,11,I) = 13.*D3(I)*DDX(I)/420.
FA(12,12,I) = FA(7,9,I).
FA(13,13,I) = 9.*DB(I)*DDX(I)/75.
FA(14,14,I) = FA(7,7,I).
FA(15,15,I) = -DB(I)*DDX(I)/140.
FA(16,16,I) = FA(9,8,I).
FA(17,17,I) = -13.*DB(I)*DDX(I)/420.
FA(18,18,I) = FA(8,8,I).
FA(19,19,I) = FA(3,4,I).
FA(20,20,I) = -13.*D3(I)*DDX(I)/420.

C THE GAMMA MATRIX ***********************************************

FG(1,1,I) = -CH(I)*DDX*DB(I)/3.
FG(2,2,I) = -CH(I)*DB(I)/3.
FG(3,3,I) = -6.*D1(I)/5.*DB(I)-24.*DDX(I)/5.*DB(I).
FG(4,4,I) = -6.*D1(I)/5.*DB(I).
FG(5,5,I) = -6.*D1(I)/5.*DB(I).
FG(6,6,I) = -6.*D1(I)/5.*DB(I).
FG(7,7,I) = -6.*D1(I)/5.*DB(I).
FG(8,8,I) = -6.*D1(I)/5.*DB(I).
FG(9,9,I) = FD(1,I)/3.
FG(10,10,I) = FD(1,I)/45.
FG(11,11,I) = -6.*D1(I)/5.*DB(I).
FG(12,12,I) = -6.*D1(I)/5.*DB(I).
FG(13,13,I) = -6.*D1(I)/5.*DB(I).
FG(14,14,I) = -6.*D1(I)/5.*DB(I).
FG(15,15,I) = FD(1,I)/3.
FG(16,16,I) = FD(1,I)/45.
FG(17,17,I) = -6.*D1(I)/5.*DB(I).
FG(18,18,I) = -6.*D1(I)/5.*DB(I).
FG(19,19,I) = -6.*D1(I)/5.*DB(I).
FG(20,20,I) = -6.*D1(I)/5.*DB(I).
FE(8,4,I) = DB(I)*DD1(I)/30.+4.*DB(I)*DDX(I)/30.
*+DD(I)*DD1(I)/30.

FE(4,8,I) = FG(8,4,I)

FE(8,3,I) = -DF1(I)/10.-4.*DDX(I)/10.-DD2(I)/10.

FE(3,8,I) = FG(8,3,I)

C THE EPSILON MATRIX

FE(2,1,I) = DH(I)*DDLXY*(-1./2.)-DH(I)*DD*(-1./2.)

FE(1,2,I) = -FE(2,1,I)

FE(5,2,I) = DG*DH(I)/2.-DDLXY*(-1./2.)

FE(2,5,I) = -FE(5,2,I)

FE(6,1,I) = DDLXY*DH(I)/2.-DG*(-1./2.)

FE(1,6,I) = -FE(6,1,I)

FE(6,5,I) = DDLXY*(DH(I)/2.)-DG*(DH(I)/2.)

FE(5,6,I) = -FE(6,5,I)

C THE ZETA MATRIX

FZ(1,1,I) = DG*DH(I)/DB(I)

FZ(2,2,I) = DDLXY*DH(I)/DB(I)

FZ(3,3,I) = 12.*DDY(I)/DB(I)**3.

FZ(4,4,I) = 4.*DDY(I)/DB(I)

FZ(5,5,I) = FZ(1,1,I)

FZ(6,6,I) = FZ(2,2,I)

FZ(7,7,I) = FZ(3,3,I)

FZ(8,8,I) = FZ(4,4,I)

FZ(9,9,I) = E.*DDY(I)/DB(I)**2.

FZ(3,4,I) = FZ(4,3,I)

FZ(8,7,I) = -E.*DDY(I)/DB(I)**2.

FZ(7,8,I) = FZ(8,7,I)

FZ(17,4,I) = -E.*DDY(I)/DB(I)**2.

FZ(4,7,I) = FZ(7,4,I)

FZ(5,1,I) = -DG*DH(I)/DB(I)

FZ(1,5,I) = FZ(5,1,I)

FZ(6,2,I) = DDLXY*DH(I)/DB(I)

FZ(2,6,I) = FZ(6,2,I)

FZ(7,3,I) = -12.*DDY(I)/DB(I)**3.

FZ(3,7,I) = FZ(7,3,I)

FZ(18,4,I) = 2.*DDY(I)/DB(I)

FZ(4,8,I) = FZ(8,4,I)

FZ(18,3,I) = 2.*DDY(I)/DB(I)**2.

FZ(3,8,I) = FZ(8,3,I)

1000 CONTINUE

I000 RETURN

101 END
"PROJECT MENAI(1) ANGLE

1  COMPILE (XM=3)
2  SUBROUTINE ANGLE
3  COMMON /EXT/SKU(144,133)*SD4CA(440,5)*SD2C3(440*3)*SD2CE(440)*
4  *SC1E(440,3)*CA(44,44)*CE(44,44)*CE(44,44)*
5  *CZ(44,44)*SIGMAY(120)*SIGMAY(120)*TAUXY(120)*
6  *AMX(120)*AMY(120)*AMXY(120)*
7  COMMON DX(30)**DY(30)**DD(30)**DXY(30)**DHY(30)**DOB(30)
8  COMMON FA(6,6,30),FO(6,6,30),FE(6,6,30),FZ(6,6,30)
9  COMMON AFA(6,6,30),AFO(6,6,30),AFE(6,6,30),AFZ(6,6,30)
10 COMMON SDF(6,5,6),SD(6,3,6),SD(6,3,6),SD(6,6)
11 COMMON DEX,DEY,DPX,DPY,DG,INDUM,DLOX,DLOXY,DLOY
12 COMMON MHG,MSK4,MSK5,MS1G,MSEGPO(25)
13 COMMON NW7,JENOX,JENUM,MS121,ME(31)
14 COMMON ANG(10)*S0
15 COMMON /B2/ATRANS

16 "DURATION ASUM(9,9)
18 DOUBLE PRECISION ATRANS(8*8+30)
19 DOUBLE PRECISION ATRANS(9*9+30)
20 C READ ANGLE OF STRIP ANGLE IS MEASURED FROM GLOBAL
21 C Y AXIS TO LOCAL Y AXIS FOR STRIP COUNTERCLOCKWISE ANGLE IS CONSIDERED
22 C NEGATIVE ANGLE IS ENTERED IN DEGREES AND FRACTIONS OF A
23 C DEGREE ORDER OF ENTRY IS AS FOR THICKNESS AND WIDTH CARDS
24 C ONE ANGLE PER CARD
25 C DO VCC I=1,INDUM
26 2000 READ (3,2001) ANG(I)
27 2001 FORMAT( )
28 C CHANGE ANGLE TO RADIANS
29 C DO 2002 I=1,INDUM
30 ANG(I) = ANG(I)/57.29577951
31 2002 CONTINUE
32 C FORMULATE TRANSFORMATION MATRICES
33 DO 2003 I = 1,INDUM
34 C ATRANS(1,1,I) = 1.
35 ATRANS(2,2,I) = COS(ANG(I))
36 ATRANS(3,3,I) = ATRANS(2,2,I)
37 ATRANS(4,4,I) = 1.
38 ATRANS(5,5,I) = 1.
39 ATRANS(6,6,I) = ATRANS(2,2,I)
40 ATRANS(7,7,I) = ATRANS(2,2,I)
41 ATRANS(8,8,I) = 1.
42 ATRANS(9,9,I) = ATRANS(2,2,I)
43 ATRANS(10,10,I) = ATRANS(2,2,I)
44 ATRANS(11,11,I) = ATRANS(2,2,I)
45 ATRANS(12,12,I) = ATRANS(2,2,I)
46 ATRANS(13,13,I) = ATRANS(2,2,I)
47 C FORMULATION OF THE TRANSFORMED TRANSFORMATION MATRICES
48 C ATRANS(1,1,I) = 1.
49 ATRANS(2,2,I) = COS(ANG(I))
50 ATRANS(3,3,I) = ATRANS(2,2,I)
51 ATRANS(4,4,I) = 1.
52 ATRANS(5,5,I) = 1.
53 ATRANS(6,6,I) = ATRANS(2,2,I)
54 ATRANS(7,7,I) = ATRANS(2,2,I)
55 ATRANS(8,8,I) = 1.
56 ATRANS(9,9,I) = ATRANS(2,2,I)
57 ATRANS(10,10,I) = ATRANS(2,2,I)
58 ATRANS(11,11,I) = ATRANS(2,2,I)
59 ATRANS(12,12,I) = ATRANS(2,2,I)
CONTINUE TO ESTABLISH THE TRANSFORMED ALFA MATRICES (ATTRAN*FA*ATRANS)

DO 2C03 I1 = 1, IDNUM
C TO CLEAR TEMP. MATRIX ASUM
DO 2C04 I = 1, 8
DO 2C04 J = 1, 8
ASUM(I,J) = 0.

CONTINUE TO DO ATRAN*ASUM=AFG
DO 2C07 K2 = 1, 8
DO 2C07 K3 = 1, 8
DO 2C07 K4 = 1, 8
AFG(K2,K3,I1) = AFG(K2,K3,I1)*ATRANS(K2,K4,I1)*ASUM(K4,K3)

DO 2C07 K2 = 1, 8
DO 2C07 K3 = 1, 8
DO 2C07 K4 = 1, 8
AFG(K2,K3,I1) = AFG(K2,K3,I1)*ATRANS(K2,K4,I1)*ASUM(K4,K3)

CONTINUE TO DO 2C10 I1 = 1, IDNUM
C TO CLEAR FG( * , I1) SO SAME STORAGE MAY BE USED FOR AFG( * , I1)
DO 2C11 L = 1, 8
DO 2C11 M = 1, 8
FG(L,M,I1) = 0.

CONTINUE TO DO 2C12 K2 = 1, 8
DO 2C12 K3 = 1, 8
DO 2C12 K4 = 1, 8
AFG(K2,K3,I1) = AFG(K2,K3,I1)*ATRANS(K2,K4,I1)*ASUM(K4,K3)

CONTINUE TO ESTABLISH THE TRANSFORMED EPSILON MATRICES (ATTRAN*FE*ATRANS)
DO 2C13 I1 = 1, IDNUM
C TO CLEAR TEMP. MATRIX ASUM

Project: MENAI(1)\ CONTACT

1. Compile(XM=3)
2. SUBROUTINE CONTACT
3. COMMON EXT/SKU(44C+132), S74CA[44C,+51, SDC2G(44C+3), SDCZE(44D),
   4. SDCZE(44D), S74CA(44C+4),CG(44,44), CE(44,44),
   5. CZ(44,44), SIGMAX(18C), SIGAX(18C), TAUXY(18C),
   6. AMX(18C), AMY(18C), AMXY(18D)
7. COMMON DDY(30), DDY(31), DDY(31), DDY(31), DB(30)
8. COMMON FA(30), FA(30), FA(30), FA(30), FA(30)
9. COMMON AF(30), AF(30), AF(30), AF(30), AF(30)
10. COMMON FC(30), FC(30), FC(30), FC(30), FC(30)
11. COMMON DE(30), DE(30), DE(30), DE(30), DE(30)
12. COMMON MH(30), MH(30), MH(30), MH(30)
13. COMMON NW(30), NW(30), NW(30), NW(30)
14. COMMON ANG(30), ANG(30)
15. C
16. C READ A CARD TELLING WHICH AF, AS, E OR Z (IDNUM) TO PICK (LC3) AND
17. C ROW OR COLUMN NUMBER WHERE IT MUST START IN (A, G, F OR Z)
18. C MATRIX (LC4); ONE CARD PER STRIP THE MATRICES THAT
19. C DONT REQUIRE SUBDIVISION ARE DONE FIRST
20. 3005 READ (8,3C2) LC3, LC4
21. 3002 FORMAT ( )
22. C TEST FOR LAST CARD; MUST BE C + 0 TO STOP PROCESS
23. IF (LC3.EQ.0 .AND. LC4 .EQ. 0 ) GO TO 3366
24. 30 LD = 1 + LC2 + LD = 1 + LC2
25. 30 LC3 = (LC4-1+LC1)
26. 30 LC4 = (LC4-1+LC2)
27. 30 CA(LC3,LC4) = CA(LC3,LC4) + AFA(LC1,LC2,LC3)
28. 30 CE(LC3,LC4) = CE(LC3,LC4) + AFZ(LC1,LC2,LC3)
29. 30 CZ(LC3,LC4) = CZ(LC3,LC4) + AFZ(LC1,LC2,LC3)
30. 30 CONTINUE: GO TO 3366
31. 30 CONTINUE: GO TO 3366
32. 30 CONTINUE: GO TO 3366
33. 30 CONTINUE: GO TO 3366
34. 30 CONTINUE: GO TO 3366
35. 30 CONTINUE: GO TO 3366
36. 30 CONTINUE: GO TO 3366
37. 30 CONTINUE: GO TO 3366
38. 30 CONTINUE: GO TO 3366
39. 30 CONTINUE: GO TO 3366
40. 30 CONTINUE: GO TO 3366
41. 30 CONTINUE: GO TO 3366
42. 30 CONTINUE: GO TO 3366
43. 30 CONTINUE: GO TO 3366
44. 30 CONTINUE: GO TO 3366
45. 30 CONTINUE: GO TO 3366
46. 30 CONTINUE: GO TO 3366
47. 30 CONTINUE: GO TO 3366
48. 30 CONTINUE: GO TO 3366
49. 30 CONTINUE: GO TO 3366
50. 30 CONTINUE: GO TO 3366
51. 30 CONTINUE: GO TO 3366
52. 30 CONTINUE: GO TO 3366
53. 30 CONTINUE: GO TO 3366
54. 30 CONTINUE: GO TO 3366
55. 30 CONTINUE: GO TO 3366
56. 30 CONTINUE: GO TO 3366
57. 30 CONTINUE: GO TO 3366
58. 30 CONTINUE: GO TO 3366
59. 30 CONTINUE: GO TO 3366

C
CA(LC15+LC16) = CA(LC15*LC16)*AF(A(LC17+LC18*LC6)
CG(LC15*LC16) = CG(LC15*LC16)*AF(G(LC17+LC18+LC6)
CE(LC15+LC16) = CE(LC15+LC16)*AF(E(LC17,LC18+LC6)
CZ(LC15*LC16) = CZ(LC15*LC16)*AF(Z(LC17+LC18+LC6)
3009 CONTINUE
GO TO 3010
3007 CONTINUE
RETURN
END

PRT's HEMAI.SIL0
<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>SD4(5,5,3)=1</td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>SD4(6,5,3)=6</td>
<td></td>
</tr>
<tr>
<td>62</td>
<td>SD4(6,4,3)=4</td>
<td></td>
</tr>
<tr>
<td>63</td>
<td>SD4(6,5,3)=1</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td><strong>C</strong></td>
<td><strong>FOURTH FACE</strong></td>
</tr>
<tr>
<td>65</td>
<td>SD4(3,3,4)=6</td>
<td></td>
</tr>
<tr>
<td>66</td>
<td>SD4(3,4,4)=4</td>
<td></td>
</tr>
<tr>
<td>67</td>
<td>SD4(4,3,4)=6</td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>SD4(4,4,4)=4</td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>SD4(5,4,4)=4</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>SD4(5,5,4)=6</td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>SD4(5,4,4)=4</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>SD4(5,5,4)=1</td>
<td></td>
</tr>
<tr>
<td>73</td>
<td>SD4(6,5,4)=6</td>
<td></td>
</tr>
<tr>
<td>74</td>
<td>SD4(6,4,4)=4</td>
<td></td>
</tr>
<tr>
<td>75</td>
<td><strong>C</strong></td>
<td><strong>FIFTH FACE</strong></td>
</tr>
<tr>
<td>76</td>
<td>SD4(3,3,5)=7</td>
<td></td>
</tr>
<tr>
<td>77</td>
<td>SD4(4,3,5)=5</td>
<td></td>
</tr>
<tr>
<td>78</td>
<td>SD4(4,4,5)=2</td>
<td></td>
</tr>
<tr>
<td>79</td>
<td>SD4(5,3,5)=7</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>SD4(5,4,5)=4</td>
<td></td>
</tr>
<tr>
<td>81</td>
<td>SD4(6,3,5)=5</td>
<td></td>
</tr>
<tr>
<td>82</td>
<td><strong>C</strong></td>
<td><strong>SIXTH FACE</strong></td>
</tr>
<tr>
<td>83</td>
<td>SD4(4,3,6)=1</td>
<td></td>
</tr>
<tr>
<td>84</td>
<td>SD4(5,3,6)=5</td>
<td></td>
</tr>
<tr>
<td>85</td>
<td><strong>C</strong></td>
<td><strong>SECOND DERIVATIVE</strong></td>
</tr>
<tr>
<td>86</td>
<td><strong>C</strong></td>
<td><strong>FIRST FACE</strong></td>
</tr>
<tr>
<td>87</td>
<td><strong>C</strong></td>
<td><strong>SECOND FACE</strong></td>
</tr>
<tr>
<td>88</td>
<td><strong>C</strong></td>
<td><strong>THIRD FACE</strong></td>
</tr>
</tbody>
</table>

**Notes:**
- Each entry represents a specific derivative value.
- **C** denotes a corner or edge, indicating a different type of derivative.
- The values are given in the format SD4(x,y,z)=value, where x, y, and z are the dimensions.
<table>
<thead>
<tr>
<th>Page</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>120</td>
<td>S_n(1,2,4)=-2.</td>
</tr>
<tr>
<td>121</td>
<td>S_2(1,3,4)=1.</td>
</tr>
<tr>
<td>122</td>
<td>S_2(2,2,4)=1.</td>
</tr>
<tr>
<td>123</td>
<td>S_2(2,3,4)=1.</td>
</tr>
<tr>
<td>124</td>
<td>S_n(3,2,4)=-2.</td>
</tr>
<tr>
<td>125</td>
<td>S_3(3,3,4)=1.</td>
</tr>
<tr>
<td>126</td>
<td>S_2(4,2,4)=1.</td>
</tr>
<tr>
<td>127</td>
<td>S_2(4,3,4)=1.</td>
</tr>
<tr>
<td>128</td>
<td>S_2(5,2,4)=1.</td>
</tr>
<tr>
<td>129</td>
<td>S_2(5,3,4)=1.</td>
</tr>
<tr>
<td>130</td>
<td>S_2(5,2,4)=-2.</td>
</tr>
<tr>
<td>131</td>
<td>S_2(6,3,4)=1.</td>
</tr>
<tr>
<td>132</td>
<td>C FIFTH FACE</td>
</tr>
<tr>
<td>133</td>
<td>S_2(1,2,5)=-2.</td>
</tr>
<tr>
<td>134</td>
<td>S_2(2,2,5)=-2.</td>
</tr>
<tr>
<td>135</td>
<td>S_2(2,3,5)=1.</td>
</tr>
<tr>
<td>136</td>
<td>S_2(3,2,5)=1.</td>
</tr>
<tr>
<td>137</td>
<td>S_2(4,2,5)=1.</td>
</tr>
<tr>
<td>138</td>
<td>S_2(4,3,5)=1.</td>
</tr>
<tr>
<td>139</td>
<td>S_2(5,2,5)=1.</td>
</tr>
<tr>
<td>140</td>
<td>S_2(5,3,5)=-1.</td>
</tr>
<tr>
<td>141</td>
<td>S_2(6,2,5)=-1.</td>
</tr>
<tr>
<td>142</td>
<td>S_2(6,3,5)=-1.</td>
</tr>
<tr>
<td>143</td>
<td>C SIXTH FACE</td>
</tr>
<tr>
<td>144</td>
<td>S_2(2,2,6)=1.</td>
</tr>
<tr>
<td>145</td>
<td>S_2(5,2,6)=1.</td>
</tr>
<tr>
<td>146</td>
<td>C FIRST DERIVATIVE</td>
</tr>
<tr>
<td>147</td>
<td>C FIRST FACE</td>
</tr>
<tr>
<td>148</td>
<td>C SECOND FACE</td>
</tr>
<tr>
<td>149</td>
<td>C THIRD FACE</td>
</tr>
<tr>
<td>150</td>
<td>C FOURTH FACE</td>
</tr>
<tr>
<td>151</td>
<td>C FIFTH FACE</td>
</tr>
<tr>
<td>152</td>
<td>C SIXTH FACE</td>
</tr>
</tbody>
</table>
300  SD2CC(MS5:L)=SD2(MRS*L,J)
301  SD1CE(MS5:L)=SD1(MRS*L,J)
302  4C09  CONTINUE
303  SD2CE(MS5)=SD1(MRS*J)
304  4C07  CONTINUE
305  C  COPY IN V VALUES
306  DO 4C18 J=1,3
307  MS7=(I*4-2)+(J-1)*MS4
308  DO 4019 K=1,5
309  SD4CA(MS7*K)=SD4(MS6*K,J)
310  4019  CONTINUE
311  DO 4020 L=1,3
312  SD2CE(MS7*L)=SD2(MS6*L,J)
313  SD1CE(MS7*L)=SD1(MS6*L,J)
314  4C20  CONTINUE
315  SD2CE(MS7)=SD(MS6*J)
316  4C18  CONTINUE
317  C  COPY IN W VALUES
318  DO 4C28 J=1,3
319  MS1C=(I*4-1)+(J-1)*MS4
320  DO 4C29 K=1,5
321  SD4CA(MS1C*K)=SD4(MS9*K,J)
322  4C29  CONTINUE
323  DO 4030 L=1,3
324  SD2CE(MS1C*L)=SD2(MS9*L,J)
325  SD1CE(MS1C*L)=SD1(MS9*L,J)
326  4C30  CONTINUE
327  SD2CE(MS1C)=SD(MS9*J)
328  4C28  CONTINUE
329  C  COPY IN THETA VALUES
330  DO 4C33 J=1,3
331  MS13=(I*4)+(J-1)*MS4
332  DO 4C34 K=1,5
333  SD4CA(MS13*K)=SD4(MS12*K,J)
334  4C34  CONTINUE
335  DO 4035 L=1,3
336  SD2CE(MS13*L)=SD2(MS12*L,J)
337  SD1CE(MS13*L)=SD1(MS12*L,J)
338  4C35  CONTINUE
339  SD2CE(MS13)=SD(MS12*J)
340  4C33  CONTINUE
341  4C33  CONTINUE
342  C  END OF STARTING CONDITIONS
343  C
344  C  END CONDITIONS
345  DO 4C36 I=1,31
346  C  TEST FOR LAST ENTRY
347  IF (MEI1).EQ.0 GO TO 4036
348  C
349  IF(MEI1).EQ.1 GO TO 4037
350  IF(MEI1).EQ.2 GO TO 4039
351  IF(MEI1).EQ.3 GO TO 4044
352  IF(MEI1).EQ.7 GO TO 4046
353  4C37  MS14=1
354  MS16=7
355  MS22=3
356  MS26=3
357  GO TO 4C48
358  4C39  MS14=1
359  MS18=5
CONTINUE
DO 4068 L=1,3
SD2CE(MS27+L)=SD2(MS25+L)+MS28)
SD1CE(MS27+L)=SD1(MS26+L)+MS28)
CONTINUE
SD2CE(MS27+L)=SD(MS26+MS28)
CONTINUE
END CONDITIONS.

STRUCTURE MUST HAVE AT LEAST 7 NODES (6 DIVISIONS) ALONG LENGTH.
READ NODAL SPACING AND DIVIDE BY SA**4, SA**2, 62* SA.

READ (6*4069) SA

FORMAT(
SAN=SA**4
SA2=SA**2
SA3=SA**2.
DO 4070 I=1,MSK0
SD4CA(I,J)=SD4CA(I,J)/SA4
CONTINUE
DO 4072 K=1,3
SD2CG(I*K)=SD2CG(I*K)/SA2
CONTINUE
DO 4073 L=1,3
SD1CE(I*L)=SD1CE(I*L)/SA3
CONTINUE

ROWS AND COLUMNS ARE REDUCED TO ZERO IN THE SKY
MATRIX TO ALLOW FOR SUPPORT CONDITIONS.

READ NUMBER OF ROWS TO BE REDUCED TO ZERO.

READ (8,4400) MS31

FORMAT(
ROW NUMBERS TO BE ZEROED ARE ENTERED TO A CARD IN FREE FORMAT
FIRST INDICATE HOW MANY CARDS TO BE READ. (ZEROS TO BE USED
TO FILL FIELD)
READ NUMBER OF CARDS TO BE READ(MS4C1)
READ(9,4400) MS401
MN=1
GO TO 4402

4401 I=NN,ST+MS401 GO TO 4402
MS4C3=10+NN-9
MS402=10+NN
READ(8,4400) (MS4C1(I),I=MS4C3,MS4C2)
NN=NN+1
GO TO 4401

CONTINUE

TO FACILITATE MATRIX MULTIPLICATION THE SD4CA, SD2CG, SD1CE
MATRICES HAVE COLUMNS SHIFTED UP ON LH. SIDE OF MAIN
DIAGONAL. AND DOWN ON RH SIDE OF MAIN DIAGONAL.

4TH DERIVATIVE
MS35=2*MS4
MS36=1*MS4
MS28=MS2D-MS35
DO 4074 I=1,MS38
MS37=MS36
SD4CA(I,J)=SD4CA(MS37+1)
600  4317  CONTINUE
601       DO 4323  K=1,MS310
602  4322  MS324=MS2ZERO(K)
603  4323  SKU(MS324*17)=1.
604       CONTINUE
605  4327  RETURN
606       END
60  SKUN(H5+NW27)=W(J)
61  SKUN(H6+NW27)=WTHEA(J1
62  SC02  CONTINUE
63  C
64  C  END
65  NW7=M*KD-NW1+1
66  DO 5003  K=1,NW1
67  NW8=NW7+K
68  SKUN(NW8+NW27)=SKU(K,NW27)
69  5003  CONTINUE
70  C
71  C  CENTRAL SECTION
72  C
73  C  DOUBLE END VALUES OF LOADING.
74  DO 5004  L=1,NW2
75  WU(L)=2*WU(L)
76  WV(L)=2*WV(L)
77  WU(L)=2*WU(L)
78  WU(L)=2*WU(L)
79  5004  CONTINUE
80  C
81  C  SUBSTITUTION INTO CENTRAL SECTION OF SKU(NNW27)
82  NW9=M*SKD-2*NW1/NW1
83  DO 5005  M=1,NW9
84  C
85  C  STARTING ADDRESSES IN SKU (NW27)
86  NW10=NW1+1
87  DO 5006  N=1,NW2
88  NW15=NW15*(M-1)+4*NW2*(N-1)+4
89  NW16=NW15+1
90  NW17=NW15+2
91  NW18=NW15+3
92  SKU(NW15+NW27)=WU(N)
93  SKU(NW16+NW27)=WV(N)
94  SKU(NW17+NW27)=WU(N)
95  SKU(NW18+NW27)=WU(N)
96  5005  CONTINUE
97  C
98  C  LIVE LOADS
99  C
100  C  READ NUMBER OF NODES HAVING LIVE LOADS.
101  READ(3,5007) NW19
102  5007  FORMAT(  
103  C  NW20 IS NODE NUMBER. NW21 IS CSS-SECTION NUMBER; OTHERS ARE
104  C  GLOBAL LIVE LOADS.
105  DO 5011  I=1,NW19
106  READ(18,5009)NW20(I),NW21(I),WLU(I),WLW(I),WLW(I),WTHE(I)
107  5009  FORMAT(  
108  C  5011  CONTINUE
109  C  COPY INTO SKU (NW27)
110  I=1
111  SC16  NW22=(4*NW20(I)-3)+(NW21(I)-1)*NW1
112  NW23=NW22+1
113  NW24=NW22+2
114  NW25=NW22+3
115  SKU(NW22+NW27)=SKU(NW22+NW27)+WLW(I)
116  SKU(NW23+NW27)=SKU(NW23+NW27)+WV(I)
117  SKU(NW24+NW27)=SKU(NW24+NW27)+WU(I)
118  SKU(NW25+NW27)=SKU(NW25+NW27)+WU(I)
119  I=I+1
120  IF(I.GT.NW19) GO TO 510
120  GC TO 5016
121  5016  CONTINUE
122  C
123  C  PUT LOADS AT SUPPORT TO ZERO.
124  C
125  DO 5015 K=1,MS310
126      NW28=MSZERO(K)
127      SKU(NW28,NW27)=0.
128  5015  CONTINUE
129  RETURN
130  END

&PRTS MENAI.GRAFT
COMMON/EXT/SKU(44C,133)*SD4CA(44C,51)*SD2CG(44C,3)*SDC2E(44C)*
+SD1CE(44C,3)*CA(44,44)*CG(44C,44)*CE(44,44),
+CS(44,44)*SIGMAX(16C)*SIGMAY(16C)*TAUxy(16C)*
+AMX(180),AMY(180)*AMXY(180),
+COMMON DX(30),DY(30),DD(30),DH(30)*GD(30),
+COMMON FA(30,8,30),FG(30)*FE(30,8,30),FZ(30,8,30),
+COMMON AF(30,8,30),AFG(30,8,30),AFE(30,30),AFZ(30,30),
+COMMON SD(30,6),SDZ(30,6),SD1(30,6),SD0(30,6),
+COMMON DXDYOPY,DPY,DPY,GNUM,DLDY,DLDX,DLDY,
+COMMON *HCG,MSKJ,MSKB,*SMID,*MSF*01253,*
+COMMON NW2,NJ3,NJ1,*ME(31),ME(31),
+COMMON ANG(30)*S,
+COMMON/2ATRANS
+DIMENSION JETRIO(16),JETO(16),JEMID(16),JENDO(16),
+DIMENSION EDVJ(16),
+DOUBLE PRECISION ATRIMS(8B,8J3),
+DIMENSION EDSIS(8),ELDIS(8),
+DOUBLE PRECISION EDUX(3),EDVY(3),EDUS(3),EDUX(3),
+DOUBLE PRECISION ED2WY(3),ED2WXY(3),
+READ(8,7001)JETRIO(I),JETO(I),JEMID(I),JENDO(I),
+READ(8,7001)JNUM,JE1,
+7000 FORMAT(1)
+FOR EACH SUB-STRUCTURE READ:ONE STRIP NUMBER WITH SAME
+ANGLE(JETRIO),NUMBER IN SKU ON FIRST CROSS-SECTION WHERE
+FIRST NODE DISPLACEMENTS BEGIN (JETO)*CONTINUE(JEMID),
+AND END(JENDO),
+DO 7001 I=1,JEN
+READ(8,7001)JETRIO(I),JETO(I),JEMID(I),JENDO(I),
+CONTINUE
+7001 FORMAT(1)
+C NUMBER OF CROSS-SECTIONS IN STRUCTURE(JENOX)
+JENOX=3*MSK3/MSK8,
+IF FICTITIOUS NODES USED,JENOX IS REDUCED
+IF(MS(1),FO.G) JENOX=JENOX-1
+IF(ME(1),EQ.GE.1) JENOX=JENOX-1,
+SPACE TO ALLOW IN NEW DISPLACEMENT VECTOR EDV IF
+4+5-NUMBER OF SUB-STRUCTURES*NUMBER OF CROSS-SECTIONS
+IN COMPLETE STRUCTURE (THIS IS CALCULATED AND PLACED
+IN LIST IN MAIN PROGRAM, NUMBER IS JEL)
+C NUMBER OF DEGREES OF FREEDOM PER CROSS-SECTION IN NEW EDV
+JE2=JEL,JENOX
+C NUMBER OF DEGREES OF FREEDOM IN CROSS-SECTION IN OLD SKU
+JC=MSK3/
+JAC=0
+IF(MS(1),EQ.G) JA=1
+PERFORM TRANSFORMATIONS (CALCULATE LOCAL DISPLACEMENTS)
+C FOR NUMBER OF CROSS-SECTIONS IN STRUCTURE(JENOX)
+DO 7002 J=1,JENOX
+C FOR NUMBER OF TWO STRIP SUB-STRUCTURES(JNUM)
+DO 7003 K=1,JNUM
+C LOAD GLOBAL DISPLACEMENTS INTO 7005
+C FIRST FOR
+DO 7004 L=1,4
+JE3L
6C  JE4=JEST(K)+(J-1+JA)*JE9*Z-1
61  EGDIS(JE3)=<ku(JE4,NW27)
62  70C4 CONTINUE
63  0 C LAST FOUR
64  DO 70C5 M=1,4
65  JE5=4+M
66  JE6=JEMID(K)+(J-1+JA)*JE9*M-1
67  EGDIS(JE6)=<ku(JE6,NW27)
68  70C5 CONTINUE
69  0 C MULTIPLY: ATRANS*EGDIS
70  DO 70C6 N=1,8
71  JE8=(J-1)+JE2+(12*K-11)+I2-1
72  EDV(JE8)=ELDIS(I2)
73  70C6 CONTINUE
74  0 C COPY INTO EDV
75  DO 70C7 I2=1,3
76  JE9=(J-1)+JE2+(12*K-11)+I2-1
77  EDV(JE8)=ELDIS(I2)
78  70C7 CONTINUE
79  0 C CLEAR ELDIS AND EGDIS
80  DO 70C8 JC=1,8
81  ELDIS(JC)=C.
82  EGDIS(JC)=C.
83  70C8 CONTINUE
84  0 C LOAD LAST OF SET OF THE THREE INTO EGDIS
85  DO 70C9 K2=1,4
86  JE10=JECND(K)+(J-1+JA)*JE9+K2-1
87  EGDIS(K2)=<ku(JE10,NW27)
88  70C9 CONTINUE
89  0 C MULTIPLY: ATRANS*EGDIS
90  DO 70C10 L2=1,4
91  DO 70C11 M2=1,4
92  JE11=JE10D(K)
93  ELDIS(L2)=ELDIS(L2)+ATRANS*(L2+M2*JE11)*EGDIS(M2)
94  70C10 CONTINUE
95  0 C COPY INTO EDV
96  DO 70C12 N2=1,4
97  JE12=(J-1)+JE2+12*K-3+N2-1
98  EDV(JE12)=ELDIS(N2)
99  70C11 CONTINUE
100  0 C CLEAR ELDIS AND EGDIS
101  DO 70C13 JC=1,8
102  ELDIS(JC)=C.
103  EGDIS(JC)=C.
104  70C12 CONTINUE
105  0 C MULTIPLY: ATRANS*EGDIS
106  DO 70C14 L2=1,4
107  DO 70C15 M2=1,4
108  JE15=JE10D(K)
109  ELDIS(L2)=ELDIS(L2)+ATRANS*(L2+M2*JE15)*EGDIS(M2)
110  70C13 CONTINUE
111  0 C NUMBER OF CROSS-SECTIONS EXCLUDING FIRST AND LAST
112  JC16=JENOX-7
113  0 C STRESS PACKAGES REQUIRED IN CENTRAL SECTION
114  JE17=JE16+JENUM
115  0 C STRESS CALCULATIONS FOR FIRST CROSS-SECTION
116  DO 70C18 I1=1,JENUM
117  0 C TO GET TO STARTING DISPLACEMENT IN EDV
118  JE7DV=12*I1-11
119  0 C L.H. NODE
180 C  CALCULATE EC2WX2(2)
181 JF272=JESTDV*6
182 JE272=JESTDV+6+JE2
183 JF274=JESTDV+6+2*JE2
184 JE275=JESTDV+6+2*JE2
185 ED2WX2(2)=12.*EDV(JE272)-5.*EDV(JF273)+9.*EDV(JE274)
186 *=1.*EDV(JE275)/(5A**2.)*
187 C  CALCULATE ED2WX2(2)
188 JF274=JESTDV+2
189 JF275=JESTDV+6
190 JF276=JESTDV+10
191 ED2WX2(2)=12.*EDV(JE47)-2.*EDV(JE48)+1.*EDV(JF79)/103*(JE23)**2)
192 C  CALCULATE ED2WX(2)
193 JF277=JESTDV+7
194 JF278=JESTDV+7+2*JE2
195 JF279=JESTDV+7+2*JE2
196 ED2WX(2)=(-3.*EDV(JE63)+4.*EDV(JE51)+1.*EDV(JE52))/
197 *(2.*SA)
198 C  R.H. NODE
201 C  CALCULATE EDUX(3)
202 JF278=JESTDV+9
203 JF279=JESTDV*9
204 JF279=JESTDV*9
205 EDUX(3)=(-3.*EDV(JE54)+4.*EDV(JF54)-1.*EDV(JE56))/(2.*SA)
206 C  CALCULATE EDVY(3)
207 JF279=JESTDV+9
208 JF278=JESTDV+9
209 JF277=JESTDV+9
210 EDVY(3)=1.*EDV(JE57)+4.*EDV(JE58)+3.*EDV(JE59)/(2.*CB(JE23))
211 C  CALCULATE EDVY(3)
212 JF278=JESTDV+9
213 JF277=JESTDV+9
214 JF278=JESTDV+9
215 EDVX(3)=(-3.*EDV(JE63)+4.*EDV(JF61)-1.*EDV(JE62))/(2.*SA)
216 C  CALCULATE EDVX(3)
217 JF277=JESTDV+9
218 JF278=JESTDV*9
219 JF278=JESTDV+9
220 EDVX(3)=(-3.*EDV(JE63)+4.*EDV(JE64)+3.*EDV(JE65))/(2.*CB(JE23))
221 C  CALCULATE ED2WX(3)
222 JF278=JESTDV+10
223 JF277=JESTDV+10+JE2
224 JF277=JESTDV+10+2*JE2
225 JF277=JESTDV+10+3*JE2
226 ED2WX(3)=(-3.*EDV(JE276)+5.*EDV(JF277)+4.*EDV(JE278)
227 *=1.*EDV(JE279)/(5A**2.)*
228 C  CALCULATE ED2WX(3)
229 JF277=JESTDV+10
230 JF277=JESTDV+10
231 JF277=JESTDV+10
232 ED2WX(3)=(-3.*EDV(JE253)+4.*EDV(JF254)+3.*EDV(JE255))
233 *=CB(JE23)**2.*
234 C  CALCULATE EDVY(3)
235 JF277=JESTDV+11
236 JF277=JESTDV+11+JE2
237 JF277=JESTDV+11+2*JE2
238 ED2WX(3)=(-3.*EDV(JE72)+4.*EDV(JF73)-1.*EDV(JE74))")
360  J E 1 2 2 = J E C D V + 4
361  J E 1 2 3 = J E C D V + 8
362  E D V Y ( 3 ) = ( 1 + E D V ( J E 1 2 1 ) - 4 + E D V ( J E 1 2 2 ) + 3 + E D V ( J E 1 2 3 ) ) / ( 2 + D B ( J E 8 2 ) )
363  C  C A L C U L A T E  E D 2 W X Z 1 ( ? )
364  J E 1 2 4 = J E C D V + 1 0 - J E 2
365  J E 1 2 5 = J E C D V + 1 0
366  J E 1 2 6 = J E C D V + 1 0 - J E 2
367  E D 2 W X Z 2 ( 3 ) = ( 1 + E D V ( J E 1 2 4 ) - 2 + E D V ( J E 1 2 5 ) + 1 + E D V ( J E 1 2 6 ) ) / ( S A + 2 )
368  C  C A L C U L A T E  E D 2 W Y 2 ( 3 )
369  J E 2 5 8 = J E C D V + 3
370  J E 2 6 0 = J E C D V + 7
371  J E 2 6 1 = J E C D V + 1 1
372  E D 2 W Y 2 ( 3 ) = ( 1 + E D V ( J E 1 2 5 9 ) - 4 + E D V ( J E 2 6 0 ) + 3 + E D V ( J E 2 6 1 ) )
373  * ( O B ( J E 8 2 ) )^{2}
374  C  C A L C U L A T E  E D 2 W X Y ( 3 )
375  J E 1 3 7 = J E C D V + 1 1 - J E 2
376  J E 1 3 1 = J E C D V + 1 1 - J E 2
377  E D 2 W X Y ( 3 ) = ( 1 - E D V ( J E 1 3 0 ) + 1 - E D V ( J E 1 3 1 ) ) / ( 2 + S A )
378  C  C A L C U L A T E  S T R E S S E S
380  7 C 1 7  C O N T I N U E
381  C  C L E A R  D E R I V A T I V E  V E C T O R S
382  D O  T C 1 7 . M = 1 + 3
383  K E 1 3 3 = J E N U M + 3 * ( L S - 1 ) + J E N U M + 3 * ( L - 1 ) + M
384  S G M A X ( J E 1 3 3 ) = E D V Y = E D U X ( M ) + D L D X + E D V Y ( M )
385  S G M A Y ( J E 1 3 3 ) = E D V X - E D U X ( M ) + D L D Y + E D V Y ( M )
386  C  A M X ( J E 1 3 3 ) = ( E D V ( J E 8 2 ) + E D 2 W X 2 ( M ) + C 2 ( J E 8 2 ) + E D 2 W Y 2 ( M ) )
387  A M Y ( J E 1 3 3 ) = ( E D 2 W X 2 ( M ) + E D V ( J E 8 2 ) + E D 2 W Y 2 ( M ) )
388  A M X ( J E 1 3 3 ) = E D V ( J E 8 2 ) + E D 2 W X 2 ( M )
389  A M Y ( J E 1 3 3 ) = E D 2 W Y 2 ( M )
390  7 C 1 8  C O N T I N U E
391  C  C L E A R  D E R I V A T I V E  V E C T O R S
392  D O  T C 1 8 . N = 1 + 3
393  E D U X ( N ) = 0
394  E D V Y ( N ) = 0
395  E D V Y ( N ) = 0
396  E D 2 W X 2 ( N ) = 0
397  E D 2 W Y 2 ( N ) = 0
398  E D 2 W Y 2 ( N ) = 0
399  7 C 1 9  C O N T I N U E
400  C  S T R E S S  C A L C U L A T I O N S  F O R  L A S T  C R O S S  S E C T I O N
401  C  D O  T C 1 9 . I = 1 + J E N U M
402  C  T O  G E T  T O  S T A R T I N G  A D D R E S S
403  J E E D V = J E - 1 + J - 2 + 1 + I + 1 a + 1 2 - 1 1 - 1
404  C  L . H .  N O D E
405  C  C A L C U L A T E  E D U X ( I )
407  J E 1 3 5 = J E E D V - J E 2
408  J E 1 2 6 = J E E D V
409  E D U X ( N ) = ( 1 + E D V ( J E 1 3 4 ) - 4 + E D V ( J E 1 3 5 ) + 3 + E D V ( J E 1 3 6 ) ) / ( 2 + S A )
410  C  C A L C U L A T E  E D V Y ( I )
411  J E 1 3 7 = J E E D V + 1
412  J E 1 3 8 = J E E D V + 5
**ALL UNITS USE 49C N.M.**

| YOUNG'S MODULUS X | 2.52443E+10 |
| YOUNG'S MODULUS Y | 2.52443E+15 |
| POISSON RATIO X | 0.36 |
| POISSON RATIO Y | 0.3 |
| SHEAR MODULUS G | 1.262215E+10 |

<table>
<thead>
<tr>
<th>STRIP NO</th>
<th>STRIP WIDTH</th>
<th>STRIP THICKNESS</th>
<th>STRIP ANGLE (RAD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.162</td>
<td>0.031</td>
<td>0.010</td>
</tr>
<tr>
<td>2</td>
<td>0.162</td>
<td>0.031</td>
<td>0.010</td>
</tr>
<tr>
<td>3</td>
<td>0.162</td>
<td>0.031</td>
<td>0.010</td>
</tr>
<tr>
<td>4</td>
<td>0.162</td>
<td>0.031</td>
<td>0.010</td>
</tr>
<tr>
<td>5</td>
<td>0.162</td>
<td>0.031</td>
<td>0.010</td>
</tr>
<tr>
<td>6</td>
<td>0.162</td>
<td>0.031</td>
<td>0.010</td>
</tr>
<tr>
<td>7</td>
<td>0.162</td>
<td>0.031</td>
<td>0.010</td>
</tr>
<tr>
<td>8</td>
<td>0.162</td>
<td>0.031</td>
<td>0.010</td>
</tr>
<tr>
<td>9</td>
<td>0.162</td>
<td>0.031</td>
<td>0.010</td>
</tr>
</tbody>
</table>

BOUNDARY CONDITION AT START IS SIMPLY SUPPORTED

BOUNDARY CONDITION AT END IS SIMPLY SUPPORTED

NORMAL FLEXURE = 4.07E

CROSS-SECTION NODE GLOBAL DISPLACEMENTS

<table>
<thead>
<tr>
<th>U</th>
<th>V</th>
<th>W</th>
<th>THETA (RAD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**B1**
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>-2.17G23+05</td>
<td>0.0000</td>
<td>-3.0000</td>
<td>0.0000</td>
<td>1.2058+05</td>
<td>2.8830+05</td>
<td>1.0000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2.20670+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>2.2950+05</td>
<td>2.7663+05</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>-1.23355+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>6.7633+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-3.17654+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>1.2058+05</td>
<td>2.8830+05</td>
<td>1.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2.20670+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>2.2950+05</td>
<td>2.7663+05</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>-1.23355+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>6.7633+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-3.17654+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>1.2058+05</td>
<td>2.8830+05</td>
<td>1.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2.20670+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>2.2950+05</td>
<td>2.7663+05</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>-1.23355+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>6.7633+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-3.17654+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>1.2058+05</td>
<td>2.8830+05</td>
<td>1.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2.20670+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>2.2950+05</td>
<td>2.7663+05</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>-1.23355+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>6.7633+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-3.17654+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>1.2058+05</td>
<td>2.8830+05</td>
<td>1.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2.20670+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>2.2950+05</td>
<td>2.7663+05</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>-1.23355+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>6.7633+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-3.17654+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>1.2058+05</td>
<td>2.8830+05</td>
<td>1.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2.20670+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>2.2950+05</td>
<td>2.7663+05</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>-1.23355+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>6.7633+05</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
</tbody>
</table>
APPENDIX F:

MENAI - USERS' MANUAL

The program is liberally documented and the READ statements are explained in the various sub-routines. All data is entered in FREE FORMAT, units being combinations of Newtons and metres.

MAIN PROGRAM: Calls the sub-routine in a specific sequence, and monitors all the output. No data is read by this program.

Dimension parameters indicating the size of large matrices must be changed to suit each structure:

<table>
<thead>
<tr>
<th>Sequence No.</th>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>MHC</td>
</tr>
<tr>
<td>26</td>
<td>MSKD</td>
</tr>
<tr>
<td>27</td>
<td>MSKB</td>
</tr>
<tr>
<td>28</td>
<td>NW27</td>
</tr>
<tr>
<td>29</td>
<td>JE1</td>
</tr>
</tbody>
</table>

where:

- **MHC** = Size of "Stiffness" matrix for 1 cross-section (= Degrees of freedom per section).
- **MSKD** = Size of Diagonal in total Stiffness matrix (= Degrees of freedom in complete structure).
- **MSKB** = Half Band Width in Stiffness matrix (= 3 x MHC)
- **NW27** = Column in Stiffness matrix for placing the load vector. (= MSKB + 1).
- **JE1** = Size of displacement vector to accommodate transformed global displacements. (=(Number of 2-strip sub-structures) x 12 x (Number of cross-sections in Structure)).

The common storage block "EXT" must be changed for each different structure as indicated:

```
COMMON/EXT/SKU(MSKD,NW27),SD4CA(MSKD,5),SD2CG(MSKD,3),SDOZE(MSKD),
.SD1CE(MSKD,3),CA(MHC,MHC),CG(MHC,MHC),CE(MHC,MHC),
```
• CZ(MHC,MHC),SIGMA(X(JE1/4)),SIGMA(Y(JE1/4)),TAUXY(JE1/4),
• AMX(JE1/4),AMY(JE1/4),AMXY(JE1/4)

**SUB-ROUTINE DEES:** Formulates the elastic constants.

<table>
<thead>
<tr>
<th>Sequence No.</th>
<th>Variables Read</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>DEX, DEY, DPX, DPY</td>
</tr>
<tr>
<td>19</td>
<td>IDNUM</td>
</tr>
<tr>
<td>23</td>
<td>DH(I), DB(I)</td>
</tr>
</tbody>
</table>

where:

- **DEX** = Young’s Modulus in the X axis direction.
- **DEY** = " " " Y " " .
- **DPX** = Poisson's Ratio in the X axis direction.
- **DPY** = " " " Y " " .
- **IDNUM** = Number of strips in structure (Max. = 30).
- **DH(I)** = Strip thickness.
- **DB(I)** = Strip widths.

**SUB-ROUTINE FORMA:** Formulates the basic strip stiffness matrices. No data is read by this sub-program.

**SUB-ROUTINE ANGLE:** Compiles the transformation matrices for individual strips and produces global stiffness matrices.

<table>
<thead>
<tr>
<th>Sequence No.</th>
<th>Variable Read</th>
</tr>
</thead>
<tbody>
<tr>
<td>27</td>
<td>ANG(I)</td>
</tr>
</tbody>
</table>

where:

- **ANG(I)** = Strip angle to global co-ordinate directions. One strip angle per card, entry sequence same as thickness and width cards.

**SUB-ROUTINE CONTAC:** Formulates the combined cross-section stiffness matrices.

<table>
<thead>
<tr>
<th>Sequence No.</th>
<th>Variable Read</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>LC3, LC4</td>
</tr>
<tr>
<td>38</td>
<td>LC5</td>
</tr>
<tr>
<td>49</td>
<td>LC6, LC7, LC8, LC11, LC12</td>
</tr>
</tbody>
</table>
where:

LC3 = Strip number.
LC4 = Row number where the L.H. top corner of the strip stiffness matrices belong in the combined matrices. Repeat for all 'normal' strips. Place 0,0 after last entry to stop process.
LC5 = Number of strip stiffness matrices that require 'split' placing. (If split placing is not required, enter 0 and program automatically skips this process).
LC6 = Strip number.
LC7 = Row number in combined matrices where strip stiffness sub-matrices belong.
LC8 = Column number (ditto).
LC11 = Row number in sub-matrices where the copy process starts.
LC12 = Column number in sub-matrices where copy process starts.

Repeat step 49 four (4) times for every strip stiffness matrix that requires 'split' placing. Information given first for L.H. top section (4 x 4), then R.H. top section, then L.H. bottom section, and finally R.H. bottom section (4 x 4). Place 0,0,0,0,0 after last entry to stop process.

SUB-ROUTINE SILO: Formulates the Finite Difference equations, multiplies them with the combined cross-sectional stiffness matrices to give the upper half of the banded stiffness matrix.

<table>
<thead>
<tr>
<th>Sequence No.</th>
<th>Variable Read</th>
</tr>
</thead>
<tbody>
<tr>
<td>247</td>
<td>MS(I) (I = 1,16)</td>
</tr>
<tr>
<td>249</td>
<td>MS(I) (I = 17,31)</td>
</tr>
<tr>
<td>254</td>
<td>ME(I) (I = 1,16)</td>
</tr>
<tr>
<td>256</td>
<td>ME(I) (I = 17,31)</td>
</tr>
<tr>
<td>432</td>
<td>SA (Min No. nodes in longitudinal direction = 7)</td>
</tr>
<tr>
<td>453</td>
<td>MS 310</td>
</tr>
<tr>
<td>459</td>
<td>MS 401</td>
</tr>
<tr>
<td>464</td>
<td>MSZERO(I)</td>
</tr>
</tbody>
</table>

where:

MS(I) = Single digit numbers indicating boundary conditions at start of structure. 1 = Fixed, 3 = Guided, 7 = Simply Supported. (Zeroes to be used to fill field).
ME(I) = Ditto ... at end of structure.
SA = Longitudinal nodal spacing (metres).
MS310 = Number of rows to be reduced to zero in stiffness and Load Matrix (Columns are automatically done as well)
MS401 = Number of cards to be read with ten row numbers per card. (Zeroes are used to fill field).
MSZERO(I) = Vector of numbers entered on MS401 cards (Max. = 250)

SUB-Routine WHEEL: Formulates the load matrix.

Sequence No. | Variable Read
-------------|------------------
46           | WU(I), WV(I), WW(I), WTHETA(I) (I = 1, to number of nodes per cross-section)
100          | NW19
105          | NW10(I), NW21(I), WLU(I), WLW(I), WLW(I), WLTHE(I) (I = 1, NW19)

where:

WU(I) = Cross-section dead load in global X direction.
WV(I) = Half cross-section dead load (N/m) in global Y direction.
WW(I) = Half cross-section dead load (N/m) in global Z direction.
WTHETA(I) = Half cross-section twisting dead load about X axis (Nm/m).

One card is read for every nodal point on cross-section.

NW19 = Number of nodes having global live loads. (Max. number = 50)
If no live loads - use dummy.
NW20 = Node number for load.
NW21 = Cross-section number for load.
WLU(I) = Global live load in X direction (N).
WLW(I) = " " " " Y " (N/m).
WLW(I) = " " " " Z " (N/m).
WLTHE(I) = Global twisting live load about X axis (Nm/m).

One card read for each node having global live loads.
**SUB-ROUTINE GRAFT:** Solves the linear algebraic equations to give global displacements. No data is read by this sub-program.

**SUB-ROUTINE ECHO:** Solves stresses in local axes co-ordinates. The structure is now thought of as a combination of two-strip sub-structures to facilitate the application of the Finite Difference operator patterns. Sections may have an odd number of strips; this merely leads to a duplication of stress and moment calculations for a particular node or the cross-section. Strips belonging to the same sub-structure must have the same thickness and width.

<table>
<thead>
<tr>
<th>Sequence No.</th>
<th>Variable Read</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>JENUM</td>
</tr>
<tr>
<td>30</td>
<td>JETRID(I), JEST(I), JEMID(I), JEEND(I) (I = 1, JENUM)</td>
</tr>
</tbody>
</table>

where:

- **JENUM** = Number of two-strip sub-structures.
- **JETRID(I)** = Strip number for transformation of displacements into local axis directions.
- **JEST(I)** = Row number in SKU( ,NW27) matrix, on first cross-section, where first node displacements begin.
- **JEMID(I)** = Row number in SKU( ,NW27) matrix, on first cross-section, where second node displacements begin.
- **JEEND(I)** = Row number in SKU( ,NW27) matrix, on first cross-section, where third node displacements begin.

**SIGN CONVENTION FOR OUTPUT**

**STRESSES:** (In local co-ordinate directions)

- **SIGMA - X:** Positive indicates tensile stress. \((N/m^2)\)
- **SIGMA - Y:** Positive indicates tensile stress. \((N/m^2)\)
- **TAU - XY:** Positive as indicated.
- **M - X:** Positive induces sagging of strip.
- **M - Y:** Positive induces sagging of strip.
- **M - XY:** Positive as indicated \((= - (M - YX))\).