Jerome L Sackman

STRUCTURES AND MATERIALS RESEARCH
DEPARTMENT OF CIVIL ENGINEERING

REPORT NO. 67-6

FINITE ELEMENT ANALYSIS OF AXISYMMETRIC SOLIDS WITH ARBITRARY LOADINGS

by
R. S. DUNHAM
and
R. E. NICKELL

Interim Technical Report
Stanford Research Institute
Menlo Park, California
Subcontract No. B-87010-US

JUNE, 1967

STRUCTURAL ENGINEERING LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY CALIFORNIA

Structures and Materials Research
Department of Civil Engineering
Division of Structural Engineering
and Structural Mechanics

Report Number 67-6

FINITE ELEMENT ANALYSIS OF AXISYMMETRIC SOLIDS WITH ARBITRARY LOADINGS

by

Robert S. Dunham
Graduate Student in Civil Engineering
University of California, Berkeley

and

Robert E. Nickell
Research Engineer, Rohm and Haas Co.
Huntsville, Alabama

Faculty Investigators

Professor Karl S. Pister
Professor Jerome L. Sackman
Professor Robert L. Taylor
Professor Leonard R. Herrmann

Interim Technical Report
Stanford Research Institute
Menlo Park, California

Subcontract No. B-87010-US
Under Prime Contract No. N123(60530)-55641A
with Naval Ordnance Test Station
China Lake, California

Structural Engineering Laboratory
University of California
Berkeley, California

June 1967

ABSTRACT

The finite element method is applied to solids with an axis of material symmetry subjected to Fourier expandable thermal, body force and surface traction loadings. The governing equations are derived for triangular toroidal continuum elements and for thin conical shell elements.

A computer program is described and various results are presented as verification. The program is described and a user's manual is given in the appendicies. A listing of the program source deck is included.

ACKNOWLEDGEMENTS

This report is sponsored by the Stanford Research Institute, Menlo Park, California under Subcontract B-87010-US and the Naval Ordnance Test Station under Prime Contract N123(60530)-55641A.

The authors also acknowledge the critiques and timely inspiration of Dr. Glenn A. Secor and the tireless efforts of Mr. Gerald Goudreau on behalf of the project.

TABLE OF CONTENTS

Symbols and Notation

δ	variational symbol
V	potential energy
U	strain energy
W	strain energy density
B	body (axisymmetric solid)
$f^i, \{f\}^*$	body force vector field
$u_i, \{u\}$	displacement vector field
$d\tau$	element of volume
S_τ	portion of surface on which stresses are prescribed
$p^i, \{p\}$	prescribed surface tractions on S_τ
$d\sigma$	element of surface area
$\tau^{ij}, \{\tau\}$	stress tensor field
$\epsilon^{ij}, \{\epsilon\}$	strain tensor field
τ_t^{ij}	thermal stress tensor field
$C^{ijkl}, [C]$	material constant tensor field
$\alpha_{kl}, \{\alpha\}$	thermal expansion tensor field
$\alpha T_{kl} \{\alpha T\}$	thermal strain tensor field
T	temperature
r, z, θ	cylindrical coordinates
$\{A\}$	generalized coordinates
$[\Phi], \{\phi\}$	displacement expansion tensor field
$\{u_o\}$	nodal point displacement vector

* { } denote vector field (column vector), [ ] denote matrix, < > denote row vector

$[\Phi_o^{(m)}],[\Phi_o],[\phi_o]$	displacement transformation matrix
$[\Phi']$	strain-generalized coordinate tensor field
$[s],[S]$	stiffness matrix in generalized coordinates
$\{f_t\},\{F_t\}$	thermal force vector field
$\{f_b\},\{F_b\}$	body force vector field
$\{f_s\},\{F_s\}$	surface traction force vector field
ΔT	temperature change
ρ	density
ω	angular velocity
a_r	radial acceleration
a_z	axial acceleration
I_i	volume integrals
$\{U\}$	nodal point displacements for entire structure
$[K]$	stiffness matrix for entire structure
$\{F\}$	nodal point forces for entire structure
Σ	shell surface
C	shell contour (circle)
$N^{\alpha\beta}$	shell stress resultant tensor field
$\epsilon_{\alpha\beta}$	shell extensional strain tensor field
$M^{\alpha\beta}$	shell moment tensor field
$\chi_{\alpha\beta}$	shell curvature tensor field
$NT_{\alpha\beta}$	shell extensional thermal stress tensor field
s	longitudinal shell coordinate
R	shell radius of curvature
ξ	transverse shell coordinate

t	shell thickness
ℓ	shell length
u_s	longitudinal shell displacement
v	meridional shell displacement
w	transverse shell displacement
$\{B\}$	shell generalized displacement coordinates
$[\varphi_o],[\varphi_{oo}]$	shell displacement transformation matrix
$[\beta],[\beta_o]$	shell coordinate transformation matrix
$[SS]$	shell stiffness matrix in generalized coordinates
ν	poisson's ratio
λ,μ	Lame's constants

INTRODUCTION

In recent years the stress analysis of axisymmetric solids of arbitrary shapes subjected to arbitrary thermal, mechanical and body force loadings has attracted considerable attention, [1-5]*, especially in application to solid propellant rocket motor grains [6-10]. While the governing field equations have been known for many years, no practicable applications were possible until modern digital computers freed the engineer from extensive hand computations. With this computational tool it is possible for the first time to consider very general geometric configurations in stress analysis problems.

Two numerical methods for stress analysis of arbitrary configurations have received extensive treatment in the technical literature in recent years. The first method is finite difference which seeks to numerically satisfy either the Navier Displacement Equations of Equilibrium or the Compatibility Equations in terms of a stress function. The method is quite general in application, however, it is seriously limited in its ability to handle changes in boundary geometry and to satisfy difficult boundary conditions. It also has the drawback of attempting to satisfy equations involving high order derivatives of the functions.

The second method is Finite Element, which is a Ritz technique of seeking a stationary value of an energy integral [1,6]. In general, a variational theorem which has the governing field equations as Euler Equations is written in terms of discretized displacement variables. Kinematic and/or static assumptions, valid over small subdomains of the body, are made, and the energy is expressed in terms of a summation of integrals over the subdomains.

*Numbers in brackets refer to references.

The method has been used successfully to obtain good approximate solutions to a wide class of problems. Some specific applications to the analysis of motor grains are gravity slump (uniform axial acceleration), internal and external pressurization, shrinkage during the curing stage, non-homogeneous cold-soak and aerodynamic heating [9,10].

This report applies the finite element analysis to axisymmetric solids subjected to arbitrary non-axisymmetric loadings by expanding the various kinematic and forcing functions into Fourier Series [4]. In this class of problems several are of immediate practicable interest. The problems of storage and handling by strap loadings may be treated. Maneuvering and lateral force problems may also be analyzed.

STRESS ANALYSIS OF AXISYMMETRIC SOLIDS WITH ARBITRARY LOADINGS

The basic approach used in this report will be to write the Potential Energy of the axisymmetric solid in terms of a Ritz Series by subdividing the body into a finite element mesh, computing the Potential Energy of each element and summing over all elements. The Potential Energy Theorem is written in terms of the Ritz functions and an absolute minimum is sought.

Theorem of Minimum Potential Energy

The Theorem of Minimum Potential Energy may be written mathematically as

$$\delta V = 0 \tag{1}$$

where*

$$V = U - \iiint_B f^i u_i \, d\tau - \iint_{S_\tau} p^i u_i \, d\sigma \tag{2}$$

$U = \iiint_B W \, d\tau$: strain energy

W: strain energy density

B: body (axisymmetric solid)

S_τ: portion of the surface on which stresses are prescribed

f^i: body force vector field

u_i: displacement vector field

p^i: prescribed surface tractions on S_τ

If thermal forces are present, the strain energy density becomes

---

*Standard covariant and contravariant tensor notation is used throughout (e.g., see Sokolnikoff, Tensor Analysis [13]).

$$W = 1/2\ \tau^{ij}\ \epsilon_{ij} - \tau_t^{ij}\ \epsilon_{ij} \tag{3}$$

where

τ^{ij}: stress tensor field

ϵ_{ij}: strain tensor field

τ_t^{ij}: thermal stress tensor field

Constitutive Equation

For linear elastic solids the constitutive equation takes the form

$$\tau^{ij} = C^{ijkl}\ \epsilon_{kl} \tag{4a}$$

with C^{ijkl} the 4th rank material constant tensor

and $$\tau_t^{ij} = C^{ijkl}\ \alpha T_{kl} \tag{4b}$$

where

$$\alpha T_{kl} = \int_{T_o}^{T} \alpha_{kl}\ dT \tag{4c}$$

For axisymmetric geometries the Potential Energy is written in terms of physical components in the cylindrical coordinate system r, z, θ. Matrix notation is used for convenience. Accordingly

$$V = \iiint_B \langle (\tfrac{1}{2}\{\epsilon(r,z,\theta)\}^T[C]^T - \{\alpha T\}^T[C]^T)\ \{\epsilon(r,z,\theta)\} - \{u(r,z,\theta)\}^T\ \{f(r,z,\theta)\}\rangle\ d\tau$$

$$- \iint_{S_\tau} \{u(r,z,\theta)\}^T\ \{p(r,z,\theta)\}\ d\sigma \tag{5}$$

where

$$\{\tau(r,z,\theta)\}^T = \langle \tau_{rr}, \tau_{zz}, \tau_{\theta\theta}, \tau_{rz}, \tau_{r\theta}, \tau_{z\theta} \rangle \tag{6a}$$

$$\{\epsilon(r,z,\theta)\}^T = \langle \epsilon_{rr}, \epsilon_{zz}, \epsilon_{\theta\theta}, 2\epsilon_{rz}, 2\epsilon_{r\theta}, 2\epsilon_{z\theta} \rangle \tag{6b}$$

and

$$\{\tau\} = [C]\ \{\epsilon\} \tag{7}$$

This report is concerned with only coordinate orthotrophy of the following type

$$[C] = [C]^T = \begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{22} & C_{23} & 0 & 0 & 0 \\ C_{13} & C_{23} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{bmatrix} \tag{8}$$

To apply the Variational Theorem it is necessary to select a displacement field in terms of a set of unknown Ritz parameters (coordinate functions) that satisfy the hypotheses of the theorem. The restrictions on u^i are that it is continuous over the entire body and possess piecewise continuous first partial derivatives.

The method of analysis used herein consists of subdividing the domain into an assemblage of triangular toroids and assuming appropriate kinematic functions within each triangular element such that compatibility of displacements across contiguous element interfaces is maintained.

Kinematic Assumptions

The displacements are first expanded into a Fourier Series as follows:

$$u_r(r,z,\theta) = \sum_{n=o}^{N} u_{rn}(r,z) \cos n\theta \tag{9a}$$

$$u_z(r,z,\theta) = \sum_{n=o}^{N} u_{zn}(r,z) \cos n\theta \tag{9b}$$

$$u_\theta(r,z,\theta) = \sum_{n=o}^{N} u_{\theta n}(r,z) \sin n\theta \tag{9c}$$

Within each triangular element a linear expansion in the r and z coordinates is made*

$$u_{rn} = A_{1n} + A_{2n} r + A_{3n} z \tag{10a}$$

$$u_{zn} = A_{4n} + A_{5n} r + A_{6n} z \tag{10b}$$

$$u_{\theta n} = A_{7n} + A_{8n} r + A_{9n} z \tag{10c}$$

Thus

$$\{u(r,z,\theta)\} = \sum_{n=o}^{N} [\Phi_n(r,z,\theta)] \{A_n\} \tag{11a}$$

where

$$\begin{Bmatrix} u_r(r,z,\theta) \\ u_z(r,z,\theta) \\ u_\theta(r,z,\theta) \end{Bmatrix} = \sum_{n=o}^{N} \begin{bmatrix} \{\phi(r,z)\}^T \cos n\theta & 0 & 0 \\ 0 & \{\phi(r,z)\}^T \cos n\theta & 0 \\ 0 & 0 & \{\phi(r,z)\}^T \sin n\theta \end{bmatrix} \begin{Bmatrix} A_{1n} \\ A_{2n} \\ \vdots \\ A_{9n} \end{Bmatrix} \tag{11b}$$

*Imposed displacement boundary conditions are restricted by the requirement that they be piecewise linear.

and

$$\{\phi(r,z)\}^T = \langle 1, r, z \rangle \tag{11c}$$

Thus the displacement field vector for each Fourier term may be written

$$\{u_n(r,z)\} = [\Phi(r,z)] \{A_n\} \tag{11d}$$

At each nodal point of the triangular element the displacements are given by a Fourier Series and hence it is possible to express the generalized coordinates or Ritz functions, $\{A_n\}$ in terms of the actual physical components of the displacements at the nodal points. And since a linear expansion has been used together with a harmonic expansion it is possible to maintain compatibility of displacements along the contiguous element interfaces by matching their nodal point displacements. The necessity of these steps to the tractability of the method cannot be overemphasized.

Hence, to this end denote (see Fig. 1)

$$\begin{aligned} u_{in} &= u_{rn}(r_i, z_i), \quad u_{jn} = \ldots\ldots \\ w_{in} &= u_{zn}(r_i, z_i), \quad w_{jn} = \ldots\ldots \\ v_{in} &= u_{\theta n}(r_i, z_i), \quad v_{jn} = \ldots\ldots \end{aligned} \tag{12}$$

Thus

$$\{u_{on}\} = [\Phi_o] \{A_n\} \tag{13 a}$$

where

$$\begin{Bmatrix} u_{in} \\ w_{in} \\ \vdots \\ v_{kn} \end{Bmatrix} = \begin{bmatrix} [\phi_o] & 0 & 0 \\ 0 & [\phi_o] & 0 \\ 0 & 0 & [\phi_o] \end{bmatrix} \begin{Bmatrix} A_{1n} \\ \vdots \\ A_{9n} \end{Bmatrix} \tag{13b}$$

and

$$[\phi_o] = \begin{bmatrix} 1 & r_i & z_i \\ 1 & r_j & z_j \\ 1 & r_k & z_k \end{bmatrix} \tag{13c}$$

The generalized coordinates may be expressed in terms of the nodal point displacements

$$\{A_n\} = [\Phi_o^{-1}]\ \{u_{on}\} \tag{13d}$$

where

$$[\Phi_o^{-1}] = \begin{bmatrix} [\phi_o^{-1}] & 0 & 0 \\ 0 & [\phi_o^{-1}] & 0 \\ 0 & 0 & [\phi_o^{-1}] \end{bmatrix} \tag{13e}$$

and

$$[\phi_o^{-1}] = \frac{1}{D} \begin{bmatrix} r_j z_k - r_k z_j & r_k z_i - r_i z_k & r_i z_j - r_j z_i \\ z_j - z_k & z_k - z_i & z_i - z_j \\ r_k - r_j & r_i - r_k & r_j - r_i \end{bmatrix} \tag{13f}$$

and

$$D = 2\ \text{Area} = r_i(z_j - z_k) + r_j(z_k - z_i) + r_k(z_i - z_j) \tag{13g}$$

In a cylindrically coordinate system the strain-displacement equations are

$$\epsilon_{rr} = \frac{\partial u_r}{\partial r}$$

$$\epsilon_{zz} = \frac{\partial u_z}{\partial z}$$

$$\epsilon_{\theta\theta} = \frac{1}{r}\left(\frac{\partial u_\theta}{\partial \theta} + u_r\right) \tag{14a}$$

$$2\,\epsilon_{rz} = \frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r}$$

$$2\,\epsilon_{r\theta} = \frac{1}{r}\left(\frac{\partial u_r}{\partial \theta} - u_\theta\right) + \frac{\partial u_\theta}{\partial r}$$

$$2\,\epsilon_{z\theta} = \frac{\partial u_\theta}{\partial z} + \frac{1}{r}\,\frac{\partial u_z}{\partial \theta}$$

The strain field may be expressed in terms of the generalized coordinates and then in terms of the nodal point displacements

$$\{\epsilon(r,z,\theta)\} = \sum_{n=o}^{N} [\Phi_n'(r,z,\theta)]\ \{A_n\} \tag{14b}$$

where

$$[\Phi_n'(r,z,\theta)] = \begin{bmatrix} 0 & cn\theta & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & cn\theta \\ \frac{cn\theta}{r} & cn\theta & \frac{zcn\theta}{r} & \frac{ncn\theta}{r} & ncn\theta & \frac{nzcn\theta}{r} & 0 & 0 & 0 \\ 0 & 0 & cn\theta & 0 & 0 & 0 & 0 & cn\theta & 0 \\ -\frac{nsn\theta}{r} & -nsn\theta & -\frac{nzsn\theta}{r} & -\frac{sn\theta}{r} & 0 & -\frac{zsn\theta}{r} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & sn\theta & -\frac{nsn\theta}{r} & -nsn\theta & -\frac{nzsn\theta}{r} \end{bmatrix} \tag{14c}$$

denoting

$$cn\theta = \cos n\theta \qquad sn\theta = \sin n\theta$$

thus

$$\{\epsilon(r,z,\theta)\} = \sum_{n=o}^{N} [\Phi_n'(r,z,\theta)] [\Phi_o^{-1}] \{u_{on}\} \tag{14d}$$

On the z-axis, the axis of revolution, there are certain restrictions imposed on the assumed displacement field such that the stresses and strains remain finite at $r = 0$. From Equations (9), (10) and (14a), the condition of finite strain requires $A_{10} = A_{30} = 0$ and $A_{7n} = -A_{1n}$, $A_{9n} = -A_{3n}$, $A_{4n} = A_{6n} = 0$ for $n = 1, 2, \ldots. N$ as $r \to 0$. However, no such restrains need be explicitly applied on the method as the coefficients of these Ritz terms become unbounded as $r \to 0$ resulting in their being calculated as 0 in the minimization process.

The Potential Energy of a single triangular element can be expressed in terms of its nodal point displacements

$$V = \sum_{n=o}^{N} \sum_{m=o}^{N} \Big\langle \iiint_B \Big(\frac{1}{2} \{u_{on}\}^T [\Phi_o^{-1}]^T [\Phi_n'(r,z,\theta)]^T [C] [\Phi_m'(r,z,\theta)] [\Phi_o^{-1}] \{u_{om}\}$$
$$- \{u_{on}\}^T [\Phi_o^{-1}]^T [\Phi_n'(r,z,\theta)]^T [C] \{\alpha T_m(r,z,\theta)\} - \{u_{on}\}^T [\Phi_o^{-1}]^T [\Phi_n(r,z,\theta)]^T \{f_m(r,z,\theta)\} \Big) d\tau$$
$$- \iint_{S_\tau} \{u_{on}\}^T [\Phi_o^{-1}]^T [\Phi_n(r,z,\theta)]^T \{p_m(r,z,\theta)\} \, d\sigma \Big\rangle \tag{15}$$

The temperature, body forces and surface tractions have been taken as Fourier Series

$$\{\alpha T_m(r,z,\theta)\} = \int_{T_o}^{T_m} \{\alpha(r,z)\}\, dT \cos m\theta$$

$$\{f_m(r,z,\theta)\}^T = \langle f_{rm}(r,z) \cos m\theta,\ f_{zm}(r,z) \cos m\theta,\ f_{\theta m}(r,z) \sin m\theta \rangle$$

$$\{p_m(r,z,\theta)\}^T = \langle p_{rm}(r,z) \cos m\theta,\ p_{zm}(r,z) \cos m\theta,\ p_{\theta m}(r,z) \sin m\theta \rangle$$

(16)

Since the θ-dependence is known explicitly in the integrals in Equation (15), the θ-integration may be carried out directly. As a consequence of the orthogonality of the trignometric functions on the interval $0 \leq \theta \leq 2\pi$, the sum in (15) exists only for m=n. Thus

$$V = \sum_{n=o}^{N} \iiint_B \Big(\tfrac{1}{2} \{u_{on}\}^T [\Phi_o^{-1}]^T [s_n(r,z)] [\Phi_o^{-1}] \{u_{on}\} - \{u_{on}\}^T [\Phi_o^{-1}]^T \{f_{tn}(r,z)\}$$

$$- \{u_{on}\}^T [\Phi_o^{-1}]^T \{f_{bn}(r,z)\} \Big) d\tau - \iint_{S_\tau} \{u_{on}\}^T [\Phi_o^{-1}]^T \{f_{sn}(r,z)\}\, d\sigma$$

(17)

where

$$[s_n(r,z)] = [\Phi_n'(r,z)]^T [C] [\Phi_n'(r,z)] \qquad (18a)$$

$$\{f_{tn}(r,z)\} = [\Phi_n'(r,z)]^T [C] \{\alpha T_n(r,z)\} \qquad (18b)$$

$$\{f_{bn}(r,z)\} = [\Phi_n(r,z)]^T \{f_n(r,z)\} \qquad (18c)$$

$$\{f_{sn}(r,z)\} = [\Phi_n(r,z)]^T \{p_n(r,z)\} \qquad (18d)$$

Where Φ_n', Φ_n, αT_n, f_n and p_n are formed by dropping their θ-dependence. It should be noted that the various matrices and vectors in Equations (18) are the only terms in the potential energy expression that are functions of coordinates r and z. This fact will be most useful when the area integrations are performed.

Now it is necessary to assign explicit expressions for the thermal forces, body forces and surface tractions.

For this report the thermal expansion matrix will be taken as

$$\{\alpha(r,z)\}^T = \{\alpha\}^T = \langle \alpha_r, \alpha_z, \alpha_\theta, 0, 0, 0 \rangle \tag{19}$$

and the temperature will be assumed constant within each element. Define

$$\begin{aligned} T_{rn} &= (C_{11}\alpha_r + C_{12}\alpha_z + C_{13}\alpha_\theta)\ \Delta T_n \\ T_{zn} &= (C_{12}\alpha_r + C_{22}\alpha_z + C_{23}\alpha_\theta)\ \Delta T_n \\ T_{\theta n} &= (C_{13}\alpha_r + C_{23}\alpha_z + C_{33}\alpha_\theta)\ \Delta T_n \end{aligned} \tag{20}$$

then

$$\{f_{tn}(r,z)\}^T = \langle \frac{1}{r} T_{\theta n},\ T_{rn}+T_{\theta n},\ \frac{z}{r} T_{\theta n},\ 0,\ 0,\ T_{zn},\ \frac{n}{r} T_{\theta n},\ n\ T_{\theta n},\ \frac{nz}{r} T_{\theta n} \rangle \tag{21}$$

The body force vector will be taken as

$$\{f(r,z,\theta)\}^T = \langle \rho r \omega^2 - \rho a_r \cos\theta,\ -\rho a_z,\ -\rho a_r \sin\theta \rangle \tag{22a}$$

Then

$$\{f_{bn}(r,z)\}^T = \langle \rho r \omega^2, \rho r^2 \omega^2, \rho r z \omega^2, -\rho a_z, -\rho r a_z, -\rho z a_z,\ 0,\ 0,\ 0 \rangle \tag{22b}$$

for n=o

and

$$\{f_{bn}(r,z)\}^T = \langle -\rho a_r, -\rho r a_r, -\rho z a_r, 0,\ 0,\ 0, -\rho a_r, -\rho r a_r, -\rho z a_r \rangle \quad \text{for } n=1 \tag{22c}$$

$$\{f_{bn}(r,z)\} = o \qquad \text{for } n=2,3,\ldots. \tag{22d}$$

The surface traction vector will be taken as arbitrary, so that the surface integration must be performed explicitly for each loading.

The area integration may be performed over the triangular area in the r,z plane. Denote

$$[S_n] = \iiint_B [s_n(r,z)]\, d\tau \tag{23a}$$

$$\{F_{tn}\} = \iiint_B \{f_{tn}(r,z)\}\, d\tau \tag{23b}$$

$$\{F_{bn}\} = \iiint_B \{f_{bn}(r,z)\}\, d\tau \tag{23c}$$

$$\{F_{sn}\} = \iint_{S_\tau} \{f_{sn}(r,z)\}\, d\sigma \tag{23d}$$

The area integration gives rise to the following integrals

$$\begin{aligned}
I_1 &= \iiint_B 1 \cdot d\tau & I_6 &= \iiint_B \frac{z^2}{r^2}\, d\tau \\
I_2 &= \iiint_B \frac{1}{r}\, d\tau & I_7 &= \iiint_B r\, d\tau \\
I_3 &= \iiint_B \frac{1}{r^2}\, d\tau & I_8 &= \iiint_B z\, d\tau \\
I_4 &= \iiint_B \frac{z}{r}\, d\tau & I_9 &= \iiint_B r^2\, d\tau \\
I_5 &= \iiint_B \frac{z}{r^2}\, d\tau & I_{10} &= \iiint_B rz\, d\tau
\end{aligned} \tag{24}$$

and define

$$[S_n] = \begin{bmatrix} [SUU_n] & [SUW_n] & [SUV_n] \\ [SUW_n]^T & [SWW_n] & [SWV_n] \\ [SUV_n]^T & [SWV_n]^T & [SVV_n] \end{bmatrix} \tag{25}$$

where

$$[SUU_n] = \begin{bmatrix} (C_{33}+n^2C_{55})I_3 & (C_{13}+C_{33}+n^2C_{55})I_2 & (C_{33}+n^2C_{55})I_5 \\ & (C_{11}+2C_{13}+C_{33}+n^2C_{55})I_1 & (C_{13}+C_{33}+n^2C_{55})I_4 \\ \text{symmetric} & & (C_{33}+n^2C_{55})I_6 + C_{44}I_1 \end{bmatrix} \tag{26a}$$

$$[SUW_n] = \begin{bmatrix} 0 & 0 & C_{23}I_2 \\ 0 & 0 & (C_{12}+C_{23})I_1 \\ 0 & C_{44}I_1 & C_{23}I_4 \end{bmatrix} \tag{26b}$$

$$[SUV_n] = n\begin{bmatrix} (C_{33}+C_{55})I_3 & C_{33}I_2 & (C_{33}+C_{55})I_5 \\ (C_{13}+C_{33}+C_{55})I_2 & (C_{13}+C_{33})I_1 & (C_{13}+C_{33}+C_{55})I_4 \\ (C_{33}+C_{55})I_5 & C_{33}I_4 & (C_{33}+C_{55})I_6 \end{bmatrix} \tag{26c}$$

$$[SWW_n] = \begin{bmatrix} n^2C_{66}I_3 & n^2C_{66}I_2 & n^2C_{66}I_5 \\ & (C_{44}+n^2C_{66})I_1 & n^2C_{66}I_4 \\ \text{symmetric} & & C_{22}I_1+n^2C_{66}I_6 \end{bmatrix} \tag{26d}$$

$$[SWV_n] = n\begin{bmatrix} 0 & 0 & -C_{66}I_2 \\ 0 & 0 & -C_{66}I_1 \\ C_{23}I_2 & C_{23}I_1 & (C_{33}-C_{66})I_4 \end{bmatrix} \tag{26e}$$

$$[SVV_n] = \begin{bmatrix} (n^2C_{33}+C_{55})I_3 & n^2C_{33}I_2 & (n^2C_{33}+C_{55})I_5 \\ & n^2C_{33}I_1 & n^2C_{33}I_4 \\ \text{symmetric} & & C_{66}I_1+(n^2C_{33}+C_{55})I_6 \end{bmatrix} \tag{26f}$$

and

$$\{F_{tn}\}^T = \langle T_{\theta n}I_2, (T_{rn}+T_{\theta n})I_1, T_{\theta n}I_4, 0, 0, T_{zn}I_1, nT_{\theta n}I_2, nT_{\theta n}I_1, nT_{\theta n}I_4 \rangle \tag{27}$$

$$\{F_{bn}\}^T = \langle \rho\omega^2 I_7, \rho\omega^2 I_9, \rho\omega^2 I_{10}, -\rho a_z I_7, -\rho a_z I_8, 0, 0, 0 \rangle, \quad n=0 \tag{28a}$$

$$\{F_{bn}\}^T = \langle -\rho a_r I_1, -\rho a_r I_7, -\rho a_r I_8, 0, 0, 0, -\rho a_r I_1, -\rho a_r I_7, -\rho a_r I_8 \rangle, \quad n=1 \tag{28b}$$

$$\{F_{bn}\}^T = 0 \qquad n=2,3,\ldots. \tag{28c}$$

For a single finite element the potential energy becomes

$$V = \sum_{n=o}^{N} \left(\frac{1}{2} \{u_{on}\}^T [\Phi_o^{-1}]^T [S_n] [\Phi_o^{-1}] \{u_{on}\} - \{u_{on}\}^T [\Phi_o^{-1}]^T \left\{ \{F_{tn}\} + \{F_{bn}\} + \{F_{sn}\} \right\} \right) \tag{29}$$

Thus for an assemblage of M such elements, the total potential energy of the system is

$$V = \sum_{m=1}^{M} \sum_{n=o}^{N} \left(\frac{1}{2} \{U_n\}^T [\Phi_o^{(m)}]^T [S_n^{(m)}] [\Phi_o^{(m)}] \{U_n\} - \{U_n\}^T [\Phi_o^{(m)}]^T \left\{ \{F_{tn}^{(m)}\} + \{F_{bn}^{(m)}\} + \{F_{sn}^{(m)}\} \right\} \right) \tag{30}$$

where

$\{U_n\}$ discretized displacement vector for the entire assemblage

$\{\Phi_o^{(m)}\}$ generalized coordinate transformation matrix to the displacements in the entire assemblage

The governing equations are obtained by taking the variation of the Potential Energy with respect to the discrete displacement variables

$$\delta V = 0 \tag{31a}$$

$$\sum_{n=o}^{N} \sum_{m=1}^{M} \left([\Phi_o^{(m)}]^T [S_n^{(m)}] [\Phi_o^{(m)}] \{U_n\} = [\Phi_o^{(m)}]^T \{F_n^{(m)}\} \right) \tag{31b}$$

where

$$\{F_n^{(m)}\} = \{F_{tn}^{(m)}\} + \{F_{bn}^{(m)}\} + \{F_{sn}^{(m)}\} \tag{31c}$$

From Eqn. (31) it is now possible to write a set of governing algebraic equations for each Fourier term

$$[K_n] \{U_n\} = \{F_n\} \text{ for } n=0, 1, 2, \ldots, N \tag{32}$$

where

$$[K_n] = \sum_{m=1}^{M} [\Phi_o^{(m)}]^T [S_n^{(m)}] [\Phi_o^{(m)}] \tag{33a}$$

and

$$\{F_n\} = \sum_{m=1}^{M} [\Phi_o^{(m)}]^T \{F_n^{(m)}\} \tag{33b}$$

SHELLS OF REVOLUTION SUBJECTED TO ARBITRARY LOADINGS

Finite element analysis of shells of revolution has been widely used for axisymmetric loadings [2,8,11], and recently has been applied to Fourier expandable loadings [5]. The four references listed above consider only conical shell elements and this represents an important geometric approximation which will greatly influence the kinematic behavior of the shell. Recently curved elements have been studied [12] and the results are promising for shell analyses.

Curved elements are beyond the scope of this report. The derivation given herein is similar to reference [5].

Potential Energy

For thin conical shells the Potential Energy may be taken as

$$V = \iint_{\Sigma} \left\{ \frac{1}{2} \left(N^{\alpha\beta} \epsilon_{\alpha\beta} + M^{\alpha\beta} \chi_{\alpha\beta} \right) - NT^{\alpha\beta} \epsilon_{\alpha\beta} \right\} d\sigma \quad - \int_C p^i u_i \, ds \tag{34}$$

where

$N^{\alpha\beta} = \int_{-t/2}^{t/2} \tau^{\alpha\beta} d\xi$ stress resultant tensor field

$\epsilon_{\alpha\beta}$ extensional strain tensor field

$M^{\alpha\beta} = \int_{-t/2}^{t/2} \tau^{\alpha\beta} \xi \, d\xi$ moment tensor field

$\chi_{\alpha\beta}$ curvature tensor field

$NT^{\alpha\beta}$ extensional thermal stress tensor field

It should be noted that no body force terms are included in the analysis and the temperature is assumed constant over the thickness.

Constitutive Equation

The general constitutive equation for the solid must be modified to satisfy the assumption

$$\tau_{\xi\xi} \cong 0$$

Thus

$$\begin{Bmatrix} N_{ss} \\ N_{\theta\theta} \\ N_{s\theta} \end{Bmatrix} = \begin{bmatrix} C^*_{11} & C^*_{12} & 0 \\ C^*_{12} & C^*_{22} & 0 \\ 0 & 0 & C^*_{44} \end{bmatrix} \begin{Bmatrix} \epsilon_{ss} \\ \epsilon_{\theta\theta} \\ 2\epsilon_{s\theta} \end{Bmatrix} \tag{35a}$$

and

$$\begin{Bmatrix} M_{ss} \\ M_{\theta\theta} \\ M_{s\theta} \end{Bmatrix} = \begin{bmatrix} D_{11} & D_{12} & 0 \\ D_{12} & D_{22} & 0 \\ 0 & 0 & D_{44} \end{bmatrix} \begin{Bmatrix} \chi_{ss} \\ \chi_{\theta\theta} \\ 2\chi_{s\theta} \end{Bmatrix} \tag{35b}$$

where

$$C^*_{11} = t\left(C_{11} - \frac{C^2_{13}}{C_{33}}\right), \qquad D_{11} = \frac{t^3}{12}\left(C_{11} - \frac{C^2_{13}}{C_{33}}\right)$$

$$C^*_{12} = t\left(C_{12} - \frac{C_{13}C_{23}}{C_{33}}\right), \qquad D_{12} = \frac{t^3}{12}\left(C_{12} - \frac{C_{13}C_{23}}{C_{33}}\right) \tag{35c}$$

$$C^*_{22} = t\left(C_{22} - \frac{C^2_{23}}{C_{33}}\right), \qquad D_{22} = \frac{t^3}{12}\left(C_{22} - \frac{C^2_{23}}{C_{33}}\right)$$

$$C^*_{44} = t\,C_{44}, \quad D_{44} = \frac{t^3}{12}\,C_{44}$$

For thin conical shells the thermal stresses may be taken as

$$NT_{\alpha\beta} = \sum_{n=o}^{N} C^{*}_{\alpha\gamma} \int_{T_o}^{T_n} \alpha_{\gamma\beta} \, dT \cos n\theta \tag{36}$$

and the coefficient of thermal expansion will be taken as

$$[\alpha] = \begin{bmatrix} \alpha_s & 0 & 0 \\ 0 & \alpha_\theta & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{37}$$

<u>Kinematic Assumptions</u>

Within the scope of the kinematic assumption made for the elasticity element, the displacement field of the conical shell is taken in the form of a Fourier Series as

$$\begin{aligned} u_s(s,\theta) &= \sum_{n=o}^{N} u_{sn}(s) \cos n\theta \\ v(s,\theta) &= \sum_{n=o}^{N} v_n(s) \sin n\theta \\ w(s,\theta) &= \sum_{n=o}^{N} w_n(s) \cos n\theta \end{aligned} \tag{38a}$$

where over the conical shell length the meridinal dependence is assumed as

$$\begin{aligned} u_{sn}(s) &= B_{1n} + B_{2n} \, s \\ v_n(s) &= B_{3n} + B_{4n} \, s \\ w_n(s) &= B_{5n} + B_{6n} \, s + B_{7n} \, s^2 + B_{8n} \, s^3 \end{aligned} \tag{38b}$$

The coordinate functions can be determined in terms of the nodal point displacements and rotations

$$\{u_{osn}\} = [\varphi_o] \; \{B_n\} \tag{39a}$$

or

$$\begin{Bmatrix} u_{sn}(i) \\ w_n(i) \\ v_n(i) \\ u_{sn}(j) \\ w_n(j) \\ v_n(j) \\ \frac{\partial w_n}{\partial s}(j) \\ \frac{\partial w_n}{\partial s}(i) \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & \ell & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \ell & \ell^2 & \ell^3 \\ 0 & 0 & 1 & \ell & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2\ell & 3\ell^2 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{Bmatrix} B_{1n} \\ B_{2n} \\ B_{3n} \\ B_{4n} \\ B_{5n} \\ B_{6n} \\ B_{7n} \\ B_{8n} \end{Bmatrix} \tag{39b}$$

and

$$\{B_n\} = [\varphi_o^{-1}] \; \{u_{osn}\} \tag{39c}$$

where

$$[\varphi_o^{-1}] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{\ell} & 0 & 0 & \frac{1}{\ell} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{\ell} & 0 & 0 & \frac{1}{\ell} & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -\frac{3}{\ell^2} & 0 & 0 & \frac{3}{\ell^2} & 0 & -\frac{1}{\ell} & -\frac{2}{\ell} \\ 0 & \frac{2}{\ell^3} & 0 & 0 & \frac{2}{\ell^3} & 0 & \frac{1}{\ell^2} & \frac{1}{\ell^2} \end{bmatrix} \tag{39d}$$

Transforming from the conical shell coordinate system into the global cylindrical coordinate system

$$\{u_{osn}\} = [\beta_o]\ \{u_{on}\} \tag{40a}$$

where

$$[\beta_o] = \begin{bmatrix} [\beta] & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & [\beta] & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{40b}$$

and

$$[\beta] = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \tag{40c}$$

Thus

$$\{B_n\} = [\varphi_{oo}^{-1}]\ \{u_{on}\} \tag{41a}$$

where

$$[\varphi_{oo}^{-1}] = [\varphi_o^{-1}]\ [\beta_o] \tag{41b}$$

denote

$$a = r_j - r_i \quad , \quad b = z_j - z_i \tag{41c}$$

then

$$[\varphi_{oo}^{-1}] = \begin{bmatrix} \frac{a}{\ell} & \frac{b}{\ell} & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{a}{\ell^2} & -\frac{b}{\ell^2} & 0 & \frac{a}{\ell^2} & \frac{b}{\ell^2} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{\ell} & 0 & 0 & \frac{1}{\ell} & 0 & 0 \\ -\frac{b}{\ell} & \frac{a}{\ell} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{3b}{\ell^3} & -\frac{3a}{\ell^3} & 0 & -\frac{3b}{\ell^3} & \frac{3a}{\ell^3} & 0 & -\frac{1}{\ell} & -\frac{2}{\ell} \\ -\frac{2b}{\ell^4} & \frac{2a}{\ell^4} & 0 & \frac{2b}{\ell^4} & -\frac{2a}{\ell^4} & 0 & \frac{1}{\ell^2} & \frac{1}{\ell^2} \end{bmatrix} \qquad (41d)$$

For a thin conical shell the strain-displacement equations are

$$\epsilon_{ss} = \frac{\partial u_s}{\partial s}$$

$$\epsilon_{\theta\theta} = \frac{1}{R} \left(\frac{\partial v}{\partial \theta} + \sin\phi \, w + \cos\phi \, u_s \right)$$

$$2\,\epsilon_{s\theta} = \frac{\partial v}{\partial s} + \frac{1}{R} \left(\frac{\partial u_s}{\partial \theta} - \cos\phi \, v \right)$$

$$\chi_{ss} = -\frac{\partial^2 w}{\partial s^2} \qquad (42)$$

$$\chi_{\theta\theta} = \frac{1}{R^2} \left(\sin\phi \frac{\partial v}{\partial \theta} - \frac{\partial^2 w}{\partial \theta^2} \right) - \frac{1}{R} \frac{\partial w}{\partial s} \cos\phi$$

$$2\,\chi_{s\theta} = \frac{1}{R^2} \cos\phi \frac{\partial w}{\partial \theta} - \frac{1}{R} \frac{\partial^2 w}{\partial s\,\partial \theta} + \left(\frac{1}{R} \frac{\partial v}{\partial s} - \frac{1}{R^2} \cos\phi \, v \right) \sin\phi$$

The terms in the Potential Energy may be expanded as

$$N^{\alpha\beta}\ \epsilon_{\alpha\beta} = \sum_{n=o}^{N}\sum_{m=o}^{N} \{S_{1n}(s)\}^T [N_{nm}(\theta)] \{S_{1m}(s)\} \tag{43a}$$

$$M^{\alpha\beta}\chi_{\alpha\beta} = \sum_{n=o}^{N}\sum_{m=o}^{N} \{S_{2n}(s)\}^T [M_{nm}(\theta)] \{S_{2m}(s)\} \tag{43b}$$

$$NT^{\alpha\beta}\ \epsilon_{\alpha\beta} = \sum_{n=o}^{N}\sum_{m=o}^{N} \{u_{on}\}^T [\varphi_{oo}^{-1}]^T [\varphi_n'(s)]^T \{NT_m\} \cos n\theta \cos m\theta \tag{43c}$$

and again due to the orthogonality of the trignometric functions on the internal $0 \leq \theta \leq 2\pi$, the double sum exist only for m=n, and hence the double subscripts on the N and M matrices will be dropped.

$$\{S_{1n}(s)\}^T = \langle \frac{\partial u_{sn}}{\partial s}, \frac{\partial v_n}{\partial s}, \frac{u_{sn}}{R}, \frac{v_n}{R}, \frac{w_n}{R} \rangle \tag{44a}$$

$$\{S_{2n}(s)\}^T = \langle \frac{\partial^2 w_n}{\partial s^2}, \frac{1}{R}\frac{\partial v_n}{\partial s}, \frac{1}{R}\frac{\partial w_n}{\partial s}, \frac{v_n}{R^2}, \frac{w_n}{R^2} \rangle \tag{44b}$$

$$\{NT_n\} = \int_{T_o}^{T_n} \begin{Bmatrix} C_{11}^* & \alpha_s \\ C_{22}^* & \alpha_\theta \end{Bmatrix} dT \tag{44c}$$

and

$$[\varphi_n'(s)] = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\cos\phi}{R} & s\frac{\cos\phi}{R} & \frac{n}{R} & \frac{ns}{R} & \frac{\sin\phi}{R} & s\frac{\sin\phi}{R} & s^2\frac{\sin\phi}{R} & s^3\frac{\sin\phi}{R} \end{bmatrix} \tag{44d}$$

Denote $sn = \sin^2 n\theta$, $cn = \cos^2 n\theta$, $s\phi = \sin$, $c\phi = \cos\phi$, then

$$[N_n(\theta)] = \begin{bmatrix} C_{11}^{*}cn & 0 & c\phi C_{12}^{*}cn & nC_{12}^{*}cn & s\phi C_{12}^{*}cn \\ & C_{44}^{*}sn & -nC_{44}^{*}sn & -c\phi C_{44}^{*}sn & 0 \\ & & c^2\phi C_{22}^{*}cn+n^2C_{44}sn & nc\phi(C_{22}^{*}cn+C_{44}^{*}sn) & s\phi c\phi C_{22}^{*}cn \\ & \text{symmetric} & & nC_{22}^{*}cn+c^2\phi C_{44}^{*}sn & ns\phi C_{22}^{*}cn \\ & & & & s^2\phi C_{22}^{*}cn \end{bmatrix}$$

and

$$[M_n(\theta)] = \tag{44e}$$

$$\begin{bmatrix} D_{11}cn & 0 & c\phi D_{12}cn & -ns\phi D_{12}cn & -n^2D_{12}cn \\ & s^2\phi D_{44}sn & ns\phi D_{44}sn & -s^2\phi c\phi D_{44}sn & -ns\phi c\phi D_{44}sn \\ & & c^2\phi D_{22}cn+n^2D_{44}sn & -ns\phi c\phi(D_{22}cn+D_{44}sn) & -n^2c\phi(D_{22}cn+D_{44}sn) \\ & & \text{symmetric} & s^2\phi(n^2D_{22}cn+c\phi D_{44}sn) & ns\phi(n^2D_{22}cn+c^2\phi D_{44}sn) \\ & & & & n^4D_{22}cn+n^2c^2\phi D_{44}sn \end{bmatrix} \tag{44f}$$

The vectors S_1 and S_2 are written in terms of the generalized coordinates as follows

$$\{S_1(s)\} = [X_1(s)]\,\{B_n\} = [X_1(s)]\,[\varphi_{oo}^{-1}]\,\{u_{on}\} \tag{45a}$$

$$\{S_2(s)\} = [X_2(s)]\,\{B_n\} = [X_2(s)]\,[\varphi_{oo}^{-1}]\,\{\dot{u}_{on}\} \tag{45b}$$

where

$$X_1(s)_{ik} = G_{ijk}\, x_j(s)\,, \quad X_2(s)_{ik} = H_{ijk}\, x_j(s) \tag{45c}$$

$$\{x(s)\}^T = \langle\ 1,\ \frac{1}{R},\ \frac{S}{R},\ \frac{S^2}{R},\ \frac{S^3}{R},\ S,\ \frac{1}{R^2},\ \frac{S}{R^2},\ \frac{S^2}{R^2},\ \frac{S^3}{R^2}\ \rangle \tag{45d}$$

and

$$G_{112} = G_{214} = G_{321} = G_{332} = G_{423} = G_{434} = G_{525} = G_{536} = G_{547} = G_{558} = 1$$

$$H_{224} = H_{326} = H_{473} = H_{484} = H_{575} = H_{586} = H_{597} = H_{5\,10\,8} = 1 \tag{45e}$$

$$H_{117} = H_{337} = 2,\ H_{348} = 3,\ H_{168} = 6$$

all other G's H's = o

Again, since the θ-dependence is known explicitly, the θ-integration may be performed directly and the Potential Energy becomes

$$V = \sum_{n=o}^{N} \langle \int_L (\tfrac{1}{2}\{u_{on}\}^T\ [\varphi_{oo}^{-1}]^T\ [[X_1(s)]^T[N_n]\ [X_1(s)] + [X_2(s)]^T[M_n][X_2(s)]][\varphi_{oo}^{-1}]\{u_{on}\}$$

$$- \{u_{on}\}^T\ [\varphi_{oo}^{-1}]^T\ [\varphi_n'(s)]^T\ \{NT_n\})\ ds\ -\ \{u_{on}\}^T\ [\varphi_{oo}^{-1}]^T\ \{F_{sn}\}\rangle \tag{46}$$

Formally the integration over the length of the conical shell element is performed

$$V = \sum_{n=o}^{N} \langle\ \frac{1}{2}\ \{u_{on}\}^T\ [\varphi_{oo}^{-1}]^T\ [SS_n]\ [\varphi_{oo}^{-1}]\ \{u_{on}\}\ -\ \{u_{on}\}^T\ [\varphi_{oo}^{-1}]^T\ \{F_{tn}\}$$

$$-\ \{u_{on}\}^T\ [\varphi_{oo}^{-1}]^T\ \{F_{sn}\}\ \rangle \tag{47a}$$

where

$$[SS_n] = \int_L ([X_1(s)]^T [N_n] [X_1(s)] + [X_2(s)]^T [M_n] [X_2(s)]) \, ds \tag{47b}$$

$$\{F_{tn}\} = \int_L [\varphi_n'(s)]^T ds \, \{NT_n\} \tag{47c}$$

For an assemblage of M such elements, the Total Potential Energy is

$$V = \sum_{m=1}^{M} \sum_{n=o}^{N} \langle \frac{1}{2} \{U_n\}^T [\Phi_o^{(m)}]^T [SS_n^{(m)}] [\Phi_o^{(m)}] \{U_n\} - \{U_n\}^T [\Phi_o^{(m)}]^T \{\{F_{tn}^{(m)}\}+\{F_{sn}^{(m)}\}\} \tag{48}$$

The governing equations (equilibrium equations) are obtained by performing a variation on the Total Potential Energy of the system to seek an absolute minimum. Accordingly

$$\delta V = 0 \tag{49}$$

yields

$$\sum_{n=o}^{N} \sum_{m=1}^{M} ([\Phi_o^{(m)}]^T [SS_n^{(m)}] [\Phi_o^{(m)}] \{U_n\} = [\Phi_o^{(m)}] \{F_n^{(m)}\}) \tag{50a}$$

where

$$\{F_n^{(m)}\} = \{F_{tn}^{(m)}\} + \{F_{sn}^{(m)}\} \tag{50b}$$

For each Fourier Term

$$[K_n] \{U_n\} = \{F_n\}, \quad n=0,1,2,\ldots.N \tag{51a}$$

where

$$[K_n] = \sum_{m=1}^{M} [\Phi_o^{(m)}]^T [SS_n^{(m)}] [\Phi_o^{(m)}] \tag{51b}$$

$$\{F_n\} = \sum_{m=1}^{M} [\Phi_o^{(m)}]^T \{F_n^{(m)}\} \tag{51c}$$

It should be noted that the displacements along the boundary between a shell element and an elasticity element are not continuous. This occurs because the displacement normal to the shell surface is expanded in a cubic while the elasticity element has only a linear expansion. While this is a violation of the Theorem of Minimum Potential Energy, it has been found to give acceptable results. However, engineering judgment must be used in interpreting shell-elasticity element interaction.

COMPUTER PROGRAM

Based on the preceeding derivations, a computer program for the Non-axisymmetrically Loaded Axisymmetric Solid and Shell has been written in FORTRAN IV (Version 13) Source Language for the Direct Coupled System, I.B.M. 7040-7094 Computers, at the Berkeley Campus of the University of California.

As could be expected, the program is quite extensive from a programming standpoint and requires moderately large amounts of computer time to do large problems with many non-zero Fourier Terms. Efforts were made to keep the program manageable in terms of coding and efficient in speed and quantity of input required.

The overlay option in FORTRAN IV and extensive use of tape storage and directly addressable disk storage were used to improve capacity and storage efficiency.

Capabilities

Currently the program can handle 700 nodal points, 600 quadrilateral (or shell) elements, 25 different materials, 50 Fourier Terms (i.e., N=50) for the force, thermal and displacement boundary conditions at a maximum bandwidth of 24 nodal points, that is the maximum nodal point difference in any element is 23.

Coding Techniques

The program consists of a main program, which is a calling sequence, and 14 subroutines. The overlaying option and a description of each routine is presented in Appendix A.

Basically, the program consists of an input and mesh generator, an element stiffness assembler, a quadrilateral elasticity stiffness routine for the solid and shell, a Gaussian elimination and back substitution routine and finally stress calculation and output routines.

Input - Output

Input consists of problem identification and control information, material characterization, nodal point identification with boundary conditions, element identification and Fourier Information. Input is described in detail in Appendix B.

Output is in the form of nodal point displacements and element stress for each non-zero Fourier Term, that is the 3 displacement components and 6 components of stress and strain. There is also an option for computing and outputing displacements and stresses at a maximum of 4 angle stations in the θ-direction. There is an option to store the nodal point displacements on tape if more extensive data reduction is desired.

Timing

Only a crude approximation of timing is possible, but experience has shown

$$t_s \cong .65*NUMNP*[1+.9(NNF-1)] + 3*NS*NNF \qquad (52)$$

where

t_s: execution time in sec.

NUMNP: Number of nodal points ($1 \leq NUMNP \leq 700$)

NNF: Number of non-zero Fourier Terms ($1 \leq NNF \leq 50$)

NS: Number of shell or elasticity elements, whichever is smaller

Thus for the largest possible problem, the timing would be about 7 min per non-zero Fourier Term. The 7040-7094 at U.C. Berkeley requires a 30 sec load time in addition to the execution time.

PROGRAM VERIFICATION

To verify the method of analysis several problems with known solutions are modeled and solutions obtained from the program. The results are compared to the known solutions and in this fashion the validity of the approach is demonstrated.

Harmonic Axisymmetric Plane Strain

Love [14] gives the solutions to the first and second modes of harmonic axisymmetric plane strain.* For a solid cylinder of radius "a" and external pressure p_o the solutions are

$$
\begin{aligned}
\tau_{rr} &= p_o r \cos\theta \\
\tau_{\theta\theta} &= 3\,p_o r \cos\theta \\
\tau_{r\theta} &= p_o r \sin\theta \\
u_r &= p_o \frac{(1-4\nu)}{4\mu} \frac{r^2}{a} \cos\theta \\
u_\theta &= p_o \frac{(5-4\nu)}{4\mu} \frac{r^2}{a} \sin\theta
\end{aligned}
\tag{53a}
$$

for the first mode and

$$
\begin{aligned}
\tau_{rr} &= p_o \cos 2\theta \\
\tau_{\theta\theta} &= p_o \left(\frac{2r^2-a^2}{a^2}\right) \cos 2\theta \\
\tau_{r\theta} &= p_o \left(\frac{r^2-a^2}{r^2}\right) \sin 2\theta \\
u_r &= \frac{p_o}{2\mu}\left(r - \frac{2\nu r^3}{3a^2}\right) \cos 2\theta \\
u_o &= \frac{p_o}{2\mu}\left[(1-\frac{2\nu}{3}) \frac{r^3}{a^2} - r\right] \sin 2\theta
\end{aligned}
\tag{53b}
$$

*These solutions were first given by Mitchell in 1900 [15a,b].

for the second mode.

It should be noted that for the first mode, the condition $\tau_{rr}(a) = p_o \cos\theta$ requires that $\tau_{r\theta}(a) = p_o \sin\theta$ if $u_r(o) = 0$; otherwise $u_r(o) \neq 0$. This is a very important consideration in problems where the body includes the axisymmetric axis.*

The results are presented in Figs. 3 and 4 and show that the finite element solutions are in excellent agreement with the exact solutions.

Concentrated Load on an Infinite Solid Cylinder

Muskhelishvili [16] gives the solution to the problem of a circular region in plane strain with a concentrated load acting on the boundary.** For a circular region of radius "a" and force P acting across the diameter the solution is

$$u_r\Big|_{\theta=0^\circ} = \frac{P}{4\pi\mu}\left[\frac{2(\lambda+2\mu)}{(\lambda+\mu)} \ln\left(\frac{1+\frac{r}{a}}{1-\frac{r}{a}}\right) - \frac{2\mu}{\lambda+\mu}\ \frac{r}{a}\right] \tag{54a}$$

and specializing for $\nu = \frac{1}{4}$

$$u_r\Big|_{\theta=0^\circ} = \frac{P}{4\pi\mu}\left[3 \ln \frac{1+\frac{r}{a}}{1-\frac{r}{a}} - \frac{r}{a}\right] \tag{54b}$$

---

*See the discussion on pg. 10 and pg. B-1.

**This problem was first treated by Hertz in 1883 [17] and Michell in 1901 [15c].

The finite element solution was obtained using the first 8 non-zero terms to approximate the infinite series representing the actual loading

$$P(\theta) = \delta(\theta) + \delta(\theta-\pi) = \sum_{n=o}^{\infty} a_n \cos n\theta \tag{55a}$$

where $\delta(\theta)$ is the Dirac Delta Function. Thus

$$a_o = \frac{1}{2\pi} \int_0^{2\pi} P(\theta)\, d\theta$$

$$a_n = \frac{1}{\pi} \int_0^{2\pi} P(\theta) \cos n\theta \, d\theta, \quad n=1,2,... \tag{55b}$$

hence

$$a_o = \frac{1}{\pi}$$

$$a_n = \begin{cases} 0, & \text{n-odd} \\ \frac{2}{\pi}, & \text{n-even} \end{cases} \tag{55c}$$

The results of the finite element solution are plotted against the exact solution in Fig. 5. The comparison is very good except as $r \to a$ where the finite element displacements fall below the exact displacements and going to a finite limit at $r = a$. Since the exact solution is unbounded as $r \to a$, the finite element solution with only 8 non-zero terms gives quite acceptable results.

Canted Rigid Punch on a Semi-Infinite Half Space

The previous problems have been one-dimensional, and a solution to a two-dimensional problem is needed to verify the method completely. Muki [18] has presented a solution to the problem of a canted rigid punch indenting a semi-infinite half space (see Fig. 6). For τ_{zz} the solution given in [18a] is

$$\tau_{zz} = \frac{4\mu}{\pi(1-\nu)} \frac{\varepsilon}{a} \left\{ \frac{R^{-\frac{1}{2}}}{\rho} [\zeta \cos \frac{\phi}{2} + (1-R) \sin \frac{\phi}{2}] - \rho R^{-\frac{3}{2}} \cos \frac{3\phi}{2} + \frac{R^{-\frac{1}{2}}}{\rho} (\cos \frac{\phi}{2} - \zeta \sin \frac{\phi}{2}) \right\} \cos \theta \tag{56a}$$

where

$$\rho = \frac{r}{a}, \quad \zeta = \frac{z}{a}, \quad R^2 = (\rho^2 + \zeta^2 - 1)^2 + 4\zeta^2$$

$$\tan \phi = \frac{2\zeta}{(\rho^2 + \zeta^2 - 1)} \tag{56b}$$

and for the displacements on the surface of the half space ($\zeta = o$) the solution given in [18b] is

$$u_r = -\frac{\varepsilon(1-2\nu)}{2\pi(1-\nu)} \left[\frac{1}{(1+[1-\rho^2]^{\frac{1}{2}})^2} + \frac{1}{\rho^2} \left\{ 1 - \frac{2}{3\rho^2} \left[1-(1-\rho^2)^{3/2} \right] \right\} -2(1-\rho^2)^{\frac{1}{2}} \right] \cos \theta$$

$$u_\theta = -\frac{\varepsilon(1-2\nu)}{2\pi(1-\nu)} \left[\frac{1}{(1+[1-\rho^2]^{\frac{1}{2}})^2} + \frac{1}{\rho^2} \left\{ 1 - \frac{2}{3\rho^2} \left[1-(1-\rho^2)^{3/2} \right] \right\} +2(1-\rho^2)^{\frac{1}{2}} \right] \sin \theta \tag{57}$$

$$u_z = \varepsilon \rho \quad \cos \theta$$

for $0 \leq \rho \leq 1$

and

$$u_r = -\frac{2\varepsilon(1-2\nu)}{3\pi(1-\nu)} \frac{1}{\rho^2} \cos \theta$$

$$u_\theta = -\frac{2\varepsilon(1-2\nu)}{3\pi(1-\nu)} \frac{1}{\rho^2} \sin \theta \tag{58}$$

$$u_z = \frac{2\varepsilon}{\pi} \left[\rho \sin^{-1} \frac{1}{\rho} - (1 - \frac{1}{\rho^2})^{\frac{1}{2}} \right] \quad \cos \theta$$

for $\rho \geq 1$.

The results for the stresses and displacements are shown in Figs. 6 and 7 respectively. There is some difficulty in interpreting the results as it is not certain that the correct "branch" was chosen for the angle in plotting Muki's formula. Also, at r = z = o, the condition of finite strain requires $u_r = u_\theta$, a condition that is met by the finite element solution but not the plot of Muki's formula.

Thin Shell Cooling Tower on 8 Supports

To verify the shell portion of the theory a problem in the harmonic analysis of shells is given. Flugge [19] has given an approximate solution to a cooling tower resting on 8 columns and supporting its own weight. The approximate solution given was based on a 5 term Fourier expansion of a stress function.

It was assumed that $\nu = 0$ and that at the base of the tower $u_r = u_\theta = 0$. Let P denote the total weight, a the radius, and t the constant thickness. The assumed loading is shown in Fig. 8.

The results are shown in Figs. 9 and 10 and both approximate solutions are in good agreement.

RESULTS

With the validity of the method demonstrated by the preceeding examples, a solution to a more meaningful problem that cannot be solved in closed form is considered.

A problem of considerable interest to the coterie of stress analysts of solid propellant grains is lateral loading on a case bonded grain. For purposes of illustration only, a "simulated lateral load problem" for a hypothetical isotropic, nearly incompressible propellant grain bonded to a thin, isotropic case is considered. The purpose of this problem is to demonstrate the capability of the method of analysis contained herein.

The geometric shape, loading and finite element mesh are shown in Fig. B1. The material properties are taken as

$$\begin{aligned} &\text{Grain:} \quad E = 500, \ \nu = 0.49 \\ &\text{Case:} \quad E = 30\text{x}10^6, \ \nu = 0.30 \end{aligned} \tag{59a}$$

and the loading is taken as

$$p(\theta) = \begin{cases} 200\pi \cos\theta, & -\frac{\pi}{2} < \theta < \frac{\pi}{2} \\ 0, & \frac{\pi}{2} \le \theta \le \frac{3\pi}{2} \end{cases} \tag{59b}$$

hence the Fourier coefficients are

$$a_o = \frac{1}{2\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} 200\pi \cos\theta \; d\theta = 200$$

$$a_1 = \frac{1}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} 200\pi \cos^2\theta \; d\theta = 100\pi$$

$$a_2 = \frac{1}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} 200\pi \cos\theta \cos 2\theta \; d\theta = 133.3 \qquad (59c)$$

$$a_3 = \frac{1}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} 200\pi \cos\theta \cos 3\theta \; d\theta = 0$$

$$a_4 = \frac{1}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} 200\pi \cos\theta \cos 4\theta \; d\theta = -26.67$$

Stress contours are shown for the shear stress at $\theta = 0$ in Fig. 11. The nature of the singularity caused by the radial restraint is clearly demonstrated.

Using elementary beam theory to compute the bending stresses in the shell gives good correlation to the shell stresses obtained from the program. The bending moment is

$$M = \int_0^L \int_0^{2\pi} p(\theta) r \cos\theta \; d\theta \; z dz$$

$$M = \frac{L^2 r}{2} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} 200\pi \cos^2\theta \; d\theta = 50\pi^2 L^2 r \qquad (60a)$$

The moment of inertia of the case alone is

$$I_c = 2\pi r^3 t \tag{60b}$$

hence

$$\sigma_c = \frac{Mr}{I} = \frac{50\pi^2 L^2 r^2}{2\pi r^3 t} = \frac{25\pi L^2}{rt} \tag{60c}$$

From Fig. B1 at z = 15

$$L = 15, \quad r = 10, \quad t = .05$$

thus

$$\sigma_c = 35{,}300$$

The finite element analysis gave a case stress of 52,700 at this station. The local longitudinal bending moment in the case is -20.2 giving a maximum tensile stress of 4800 and a combined tensile stress in the case of 57,500. Thus the finite element solution gives approximately 60% greater stress in the case than does ordinary bending theory.

REFERENCES

1. Wilson, Edward L., "Finite Element Analysis of Two-Dimensional Structures," Structural Engineering Laboratory Report 63-2, University of California, Berkeley (June 1963).

2. Grafton, P.E. and Strome, D.R., "Analysis of Axisymmetric Shells by the Direct Stiffness Method," AIAA Jour., V. 1, No. 10, 2342-2347, (Oct. 1963).

3. Rashid, Yusef R., "Solution of Elasto-Static Boundary Value Problems by the Finite Element Method," Ph.D. Dissertation, University of California, Berkeley (1964).

4. Wilson, Edward L., "Structural Analysis of Axisymmetric Solids," AIAA Jour., V. 3, No. 12, 2269-2274 (Dec. 1965).

5. Percy, J.H., Pian, T.H.H., Klein, S. and Narvaratna, D.R., "Application of the Matrix Displacement Method to Linear Elastic Analysis of Shells of Revolution," AIAA Jour., V. 3, No. 11, 2138-2145 (Nov. 1965).

6. Pister, K.S., Taylor, R.L., and Dill, E.H., "A Computer Program for Axially Symmetric Elasticity Problems," MSC Report No. 65-21-3, Mathematical Sciences Corp., Seattle, Wash. (Dec. 1965).

7. Becker, E.H., and Brisbane, J.J., "Application of the Finite Element Method to Stress Analysis of Solid Propellant Rocker Grains," Rohm and Haas Report S-76 (Nov. 1965).

8. Herrmann, L.R., Taylor, R.L. and Green, D.R., "Finite Element Analysis for Solid Rocket Motor Cases," Structural Engineering Laboratory Report 67-4 (March 1967).

9. Dunham, R.S., Fourney, M.E., Holsapple, K.A., Nickell, R.E., and Taylor, R.L., "Stress Analysis of LPC 156-5 and 156-6 Motors," MSC Report No. 66-21-2, Mathematical Sciences Corp., Seattle, Wash. (May 1966).

10. Dunham, R.S., Fourney, M.E., Green, D.R., Holsapple, K.A., Nickell, R.E., Schmidt, W.F. and Taylor, R.L., "Stress Analysis of Thiokol TU-548 Motor," MSNW Report No. 66-52-1, Mathematical Sciences Northwest, Inc., Seattle, Wash. (June 1966).

11. Popov, E.P., Penzien, J. and Lu, Z.A., "Finite Element Solution for Axisymmetric Shells," ASCE ENG MECH Jour., V. 90, No. 5, 119-145 (Oct. 1964).

12. Jones, R.E. and Strome, D.R., "Direct Stiffness Method Analysis of Shells of Revolution Utilizing Curved Elements," AIAA Jour., V. 4, No. 9, 1519-1525 (Sept. 1966).

13. Sokolinkoff, I.S., Tensor Analysis: Theory and Applications, John Wiley and Sons, Inc., (1951).

14. Love, A.E.H., The Mathematical Theory of Elasticity, Cambridge University Press, 1st ed. 1892, 4th ed. 1927, 270-271 (1927).

15. Michell, J.H., Proc. of the London Mathematical Socity,
    a. "On the Direct Determination of Stress in an Elastic Solid," V. 31 (1899)
    b. "Elementary Distributions of Plane Stress," V. 32, 35-61 (1900).
    c. "Inversion of Plane Stress," V. 34, 134-142 (1901).

16. Muskhelishvili, N.I., Some Basic Problems in the Mathematical Theory of Elasticity, Noordhoff, 324-328 (1953).

17. Hertz, H., "Über die Vertheilung der Druckkräfte in einem elastishen Kreiscylinder," Zeitschrift für Mathematik und Physik, V. 28, 125-128 (1883).

18. Muki, Rokuro,
    a. "Axisymmetric Problems of the Theory of Elasticity for a Semi-Infinite Solid and a Thick Plate," Progress in Solid Mechanics, V. I, 399-439 (1960).
    b. "A Three-Dimensional Problem of a Semi-Infinite Elastic Solid under the Compressive Action of a Rigid Body," Proc. Fac. Engin. Keio Univ., V. 8, No. 30, 8-21 (1955).

19. Flügge, Wilhelm, Stresses in Shells, Springer-Verlag, 234-237 (1962).

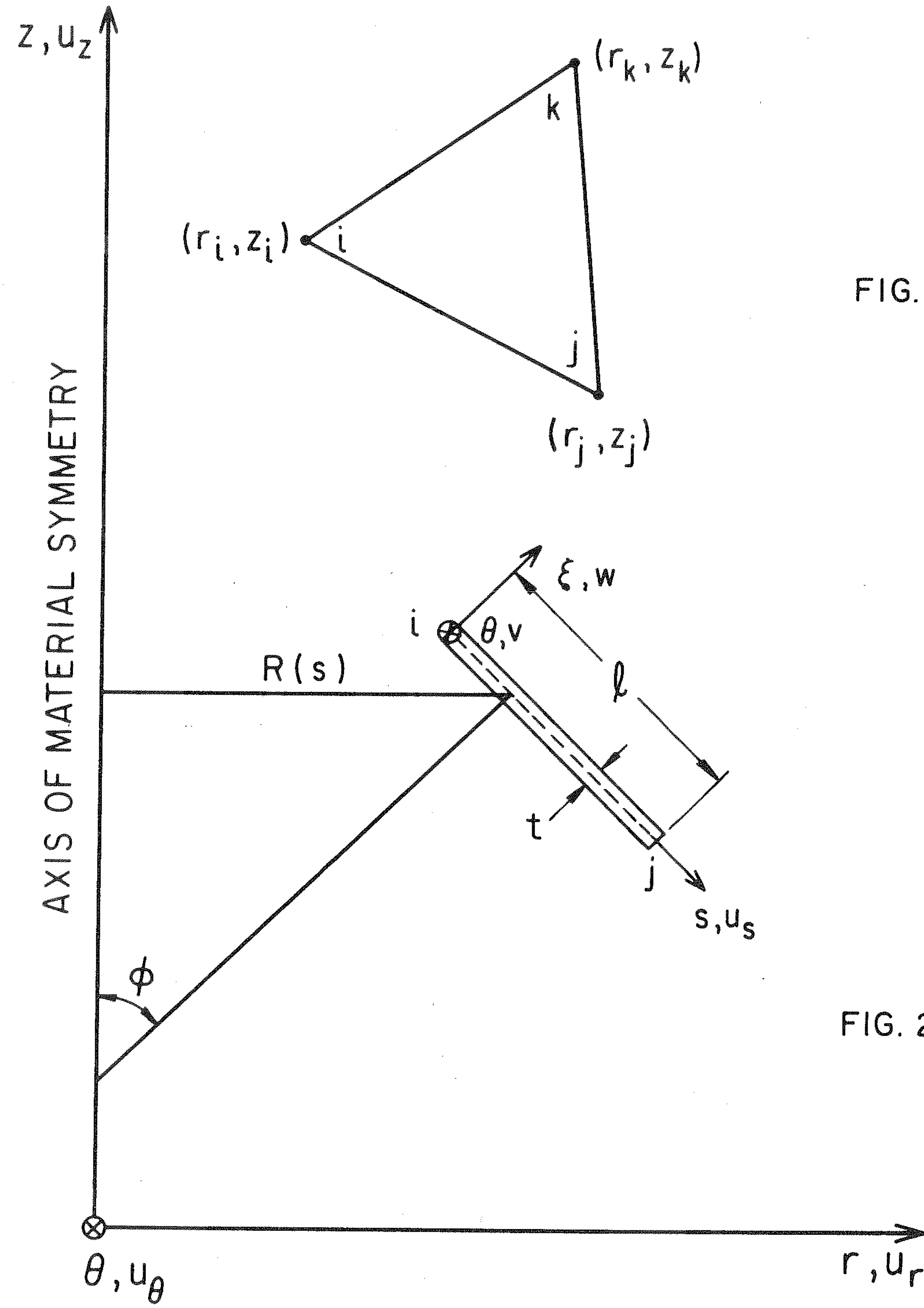
z, u_z
(r_k, z_k)
k
(r_i, z_i)
i
j
(r_j, z_j)
AXIS OF MATERIAL SYMMETRY
ξ, w
i
θ, v
R(s)
ℓ
t
j
s, u_s
φ
θ, u_θ
r, u_r

FIG. 1

FIG. 2

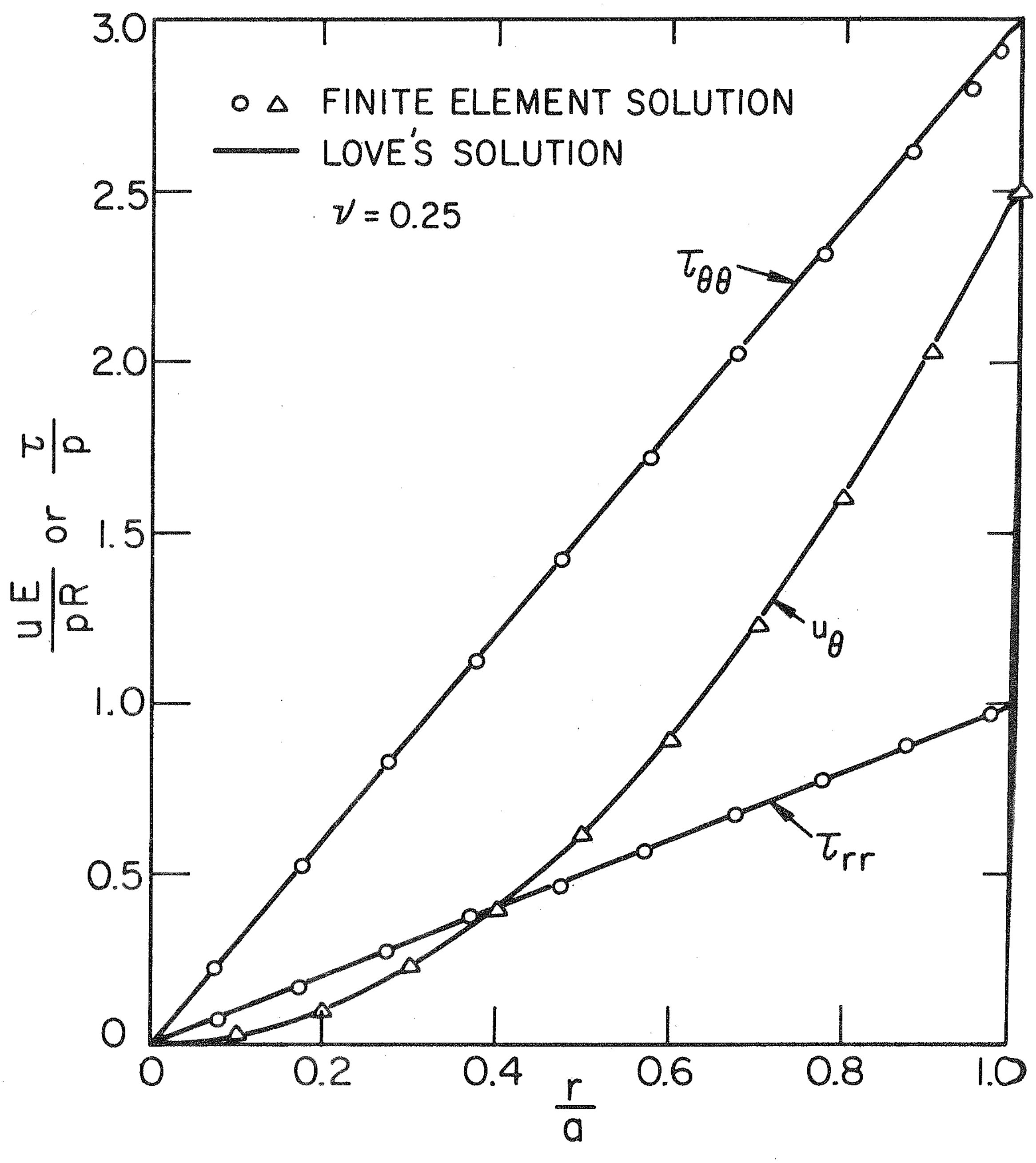

FIG. 3 FIRST MODE SOLID CYLINDER PLANE STRAIN

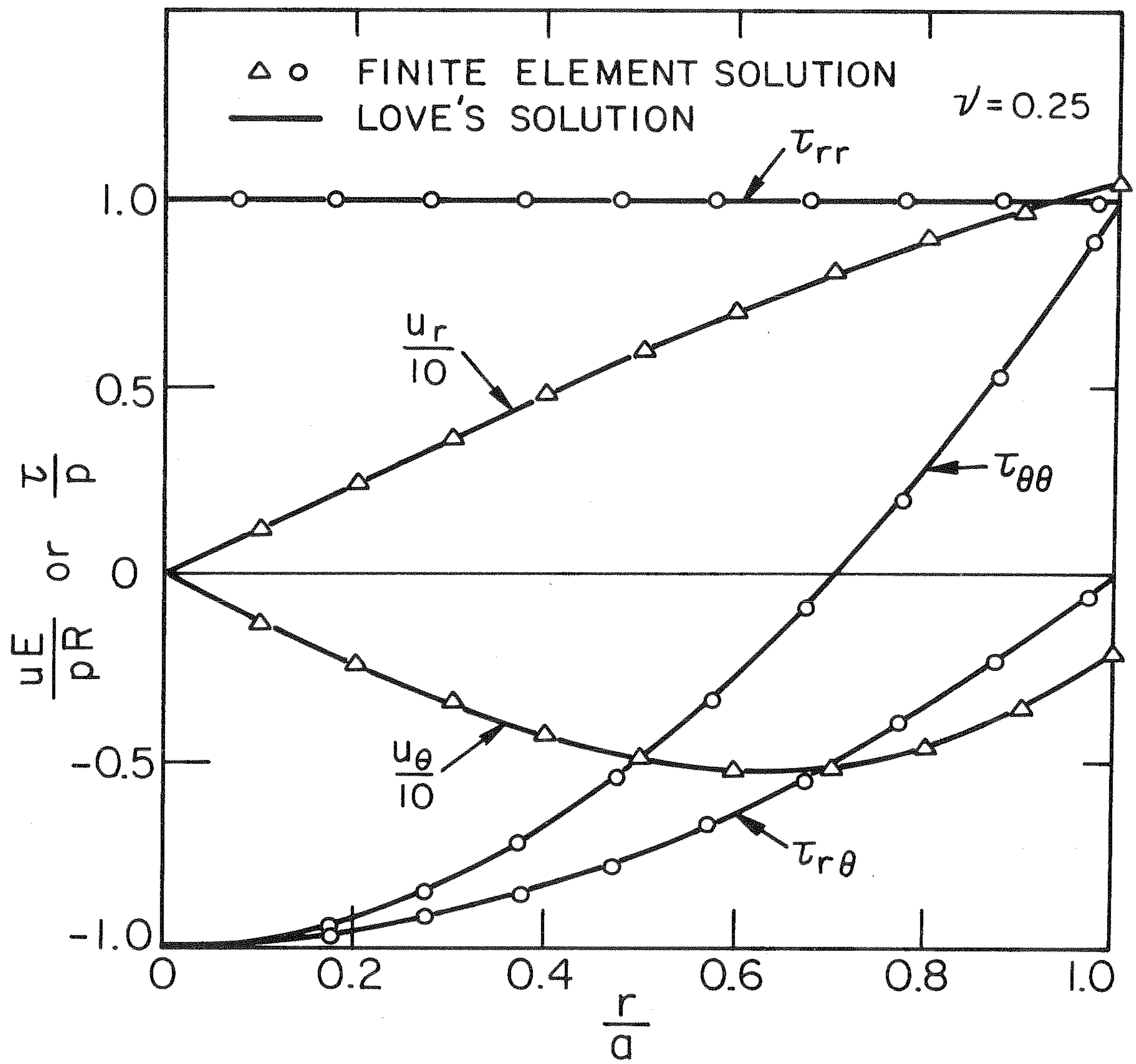

FIG. 4 SECOND MODE SOLID CYLINDER PLAIN STRAIN

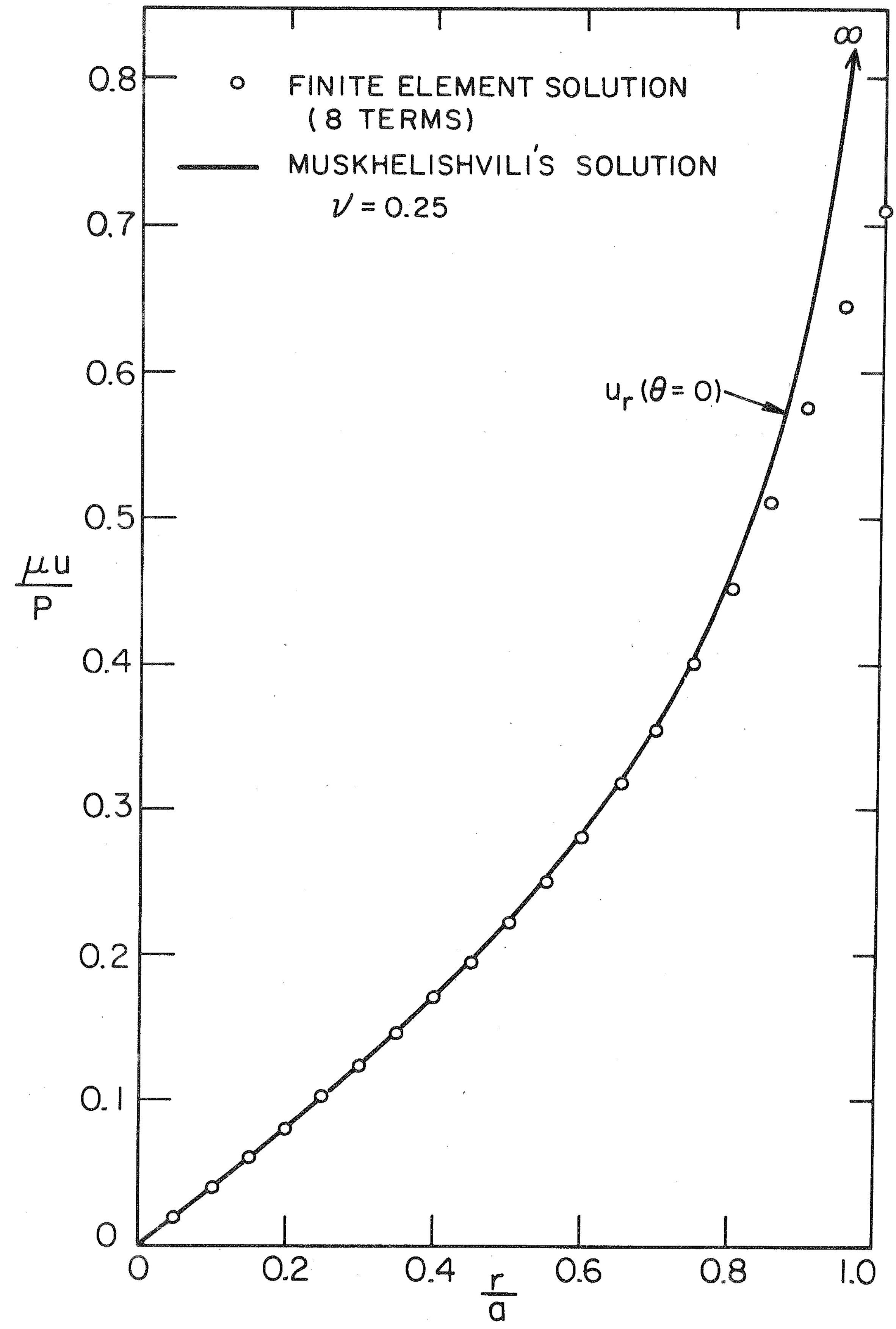

FIG. 5 CONCENTRATED LOAD ON A SOLID CYLINDER

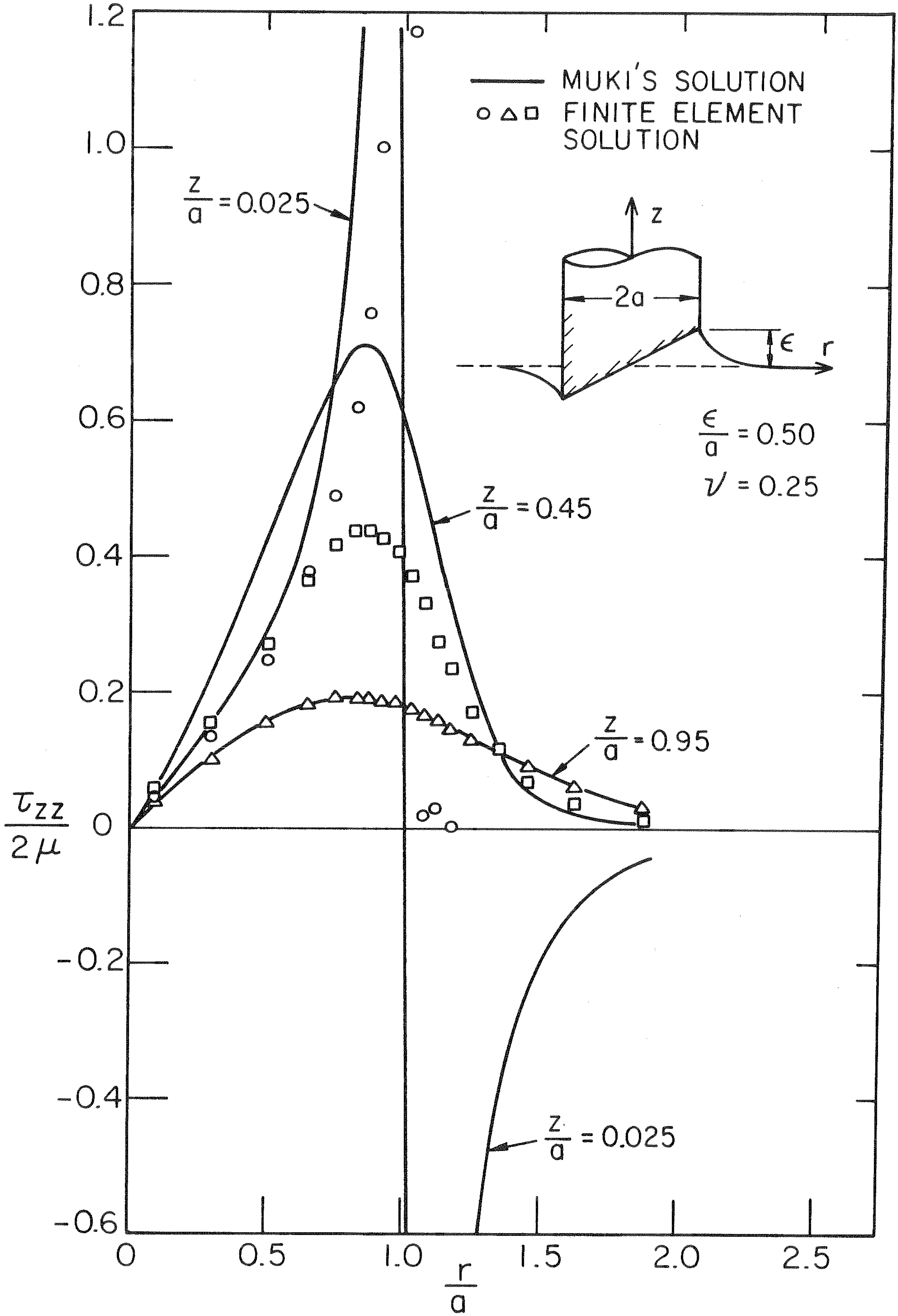

FIG. 6 CANTED RIGID PUNCH ON A SEMI-INFINITE HALF SPACE

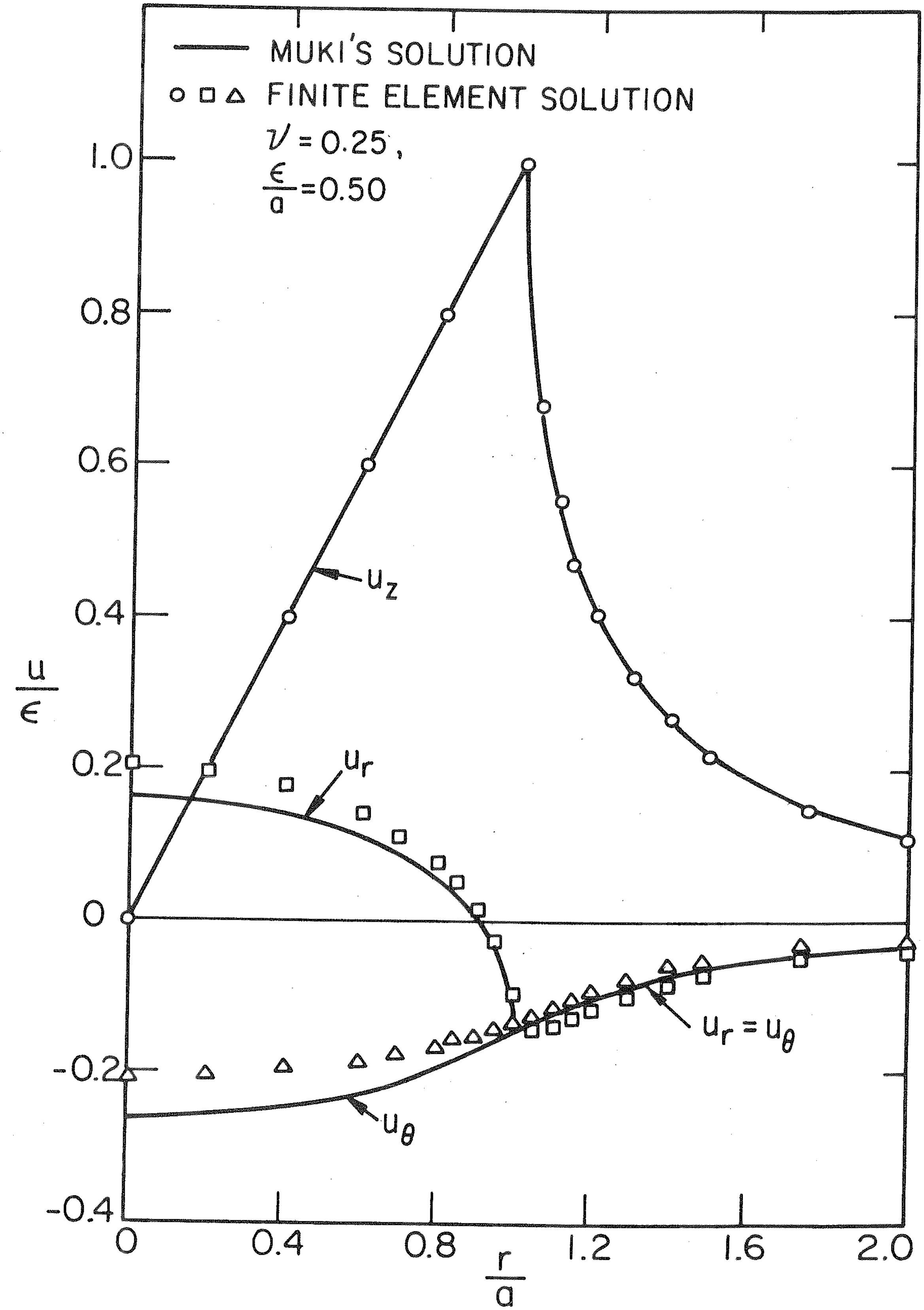

FIG. 7 CANTED RIGID PUNCH ON A SEMI-INFINITE HALF SPACE

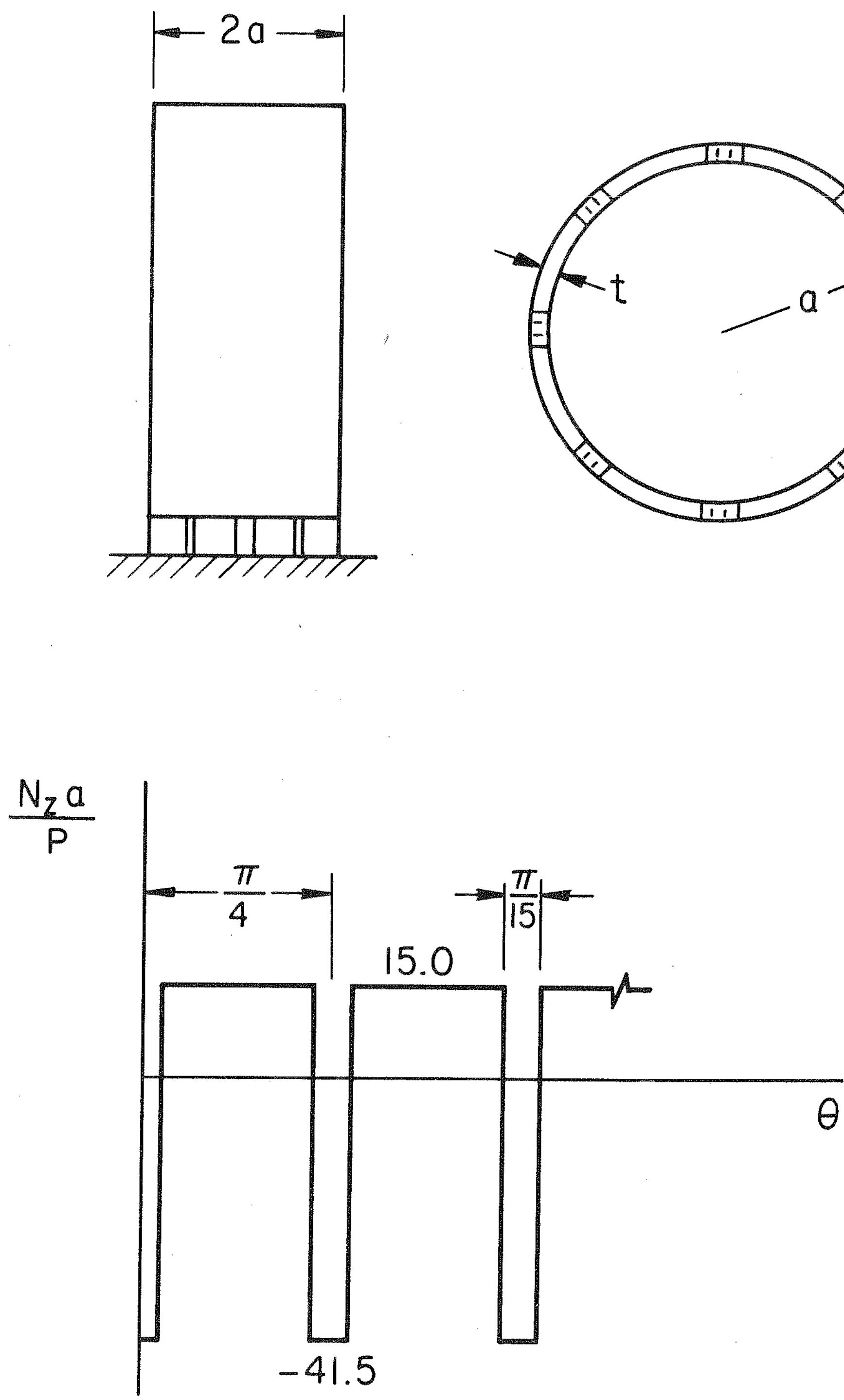

FIG. 8 COOLING TOWER ON 8 SUPPORTS

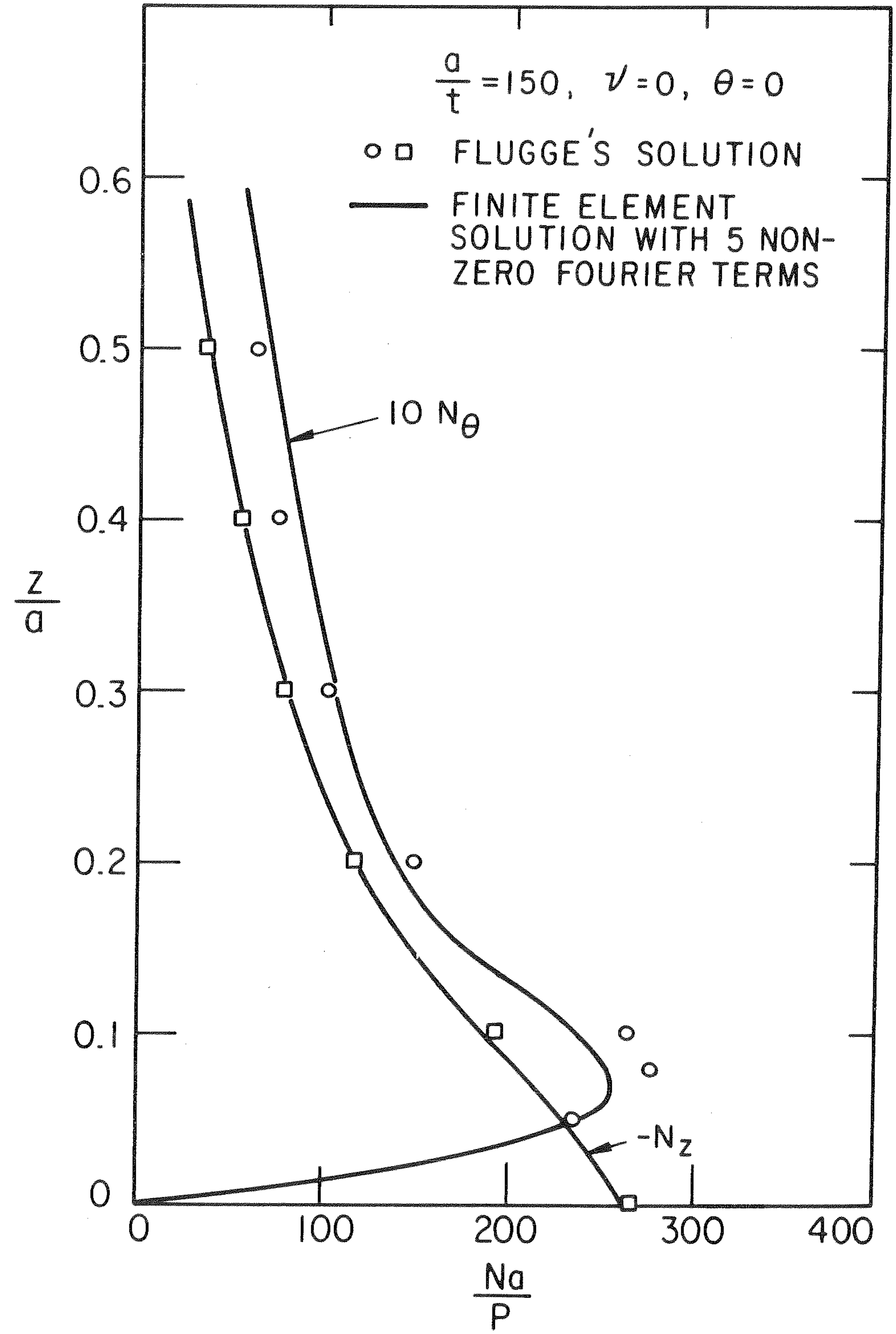

FIG. 9 COOLING TOWER STRESS RESULTANTS

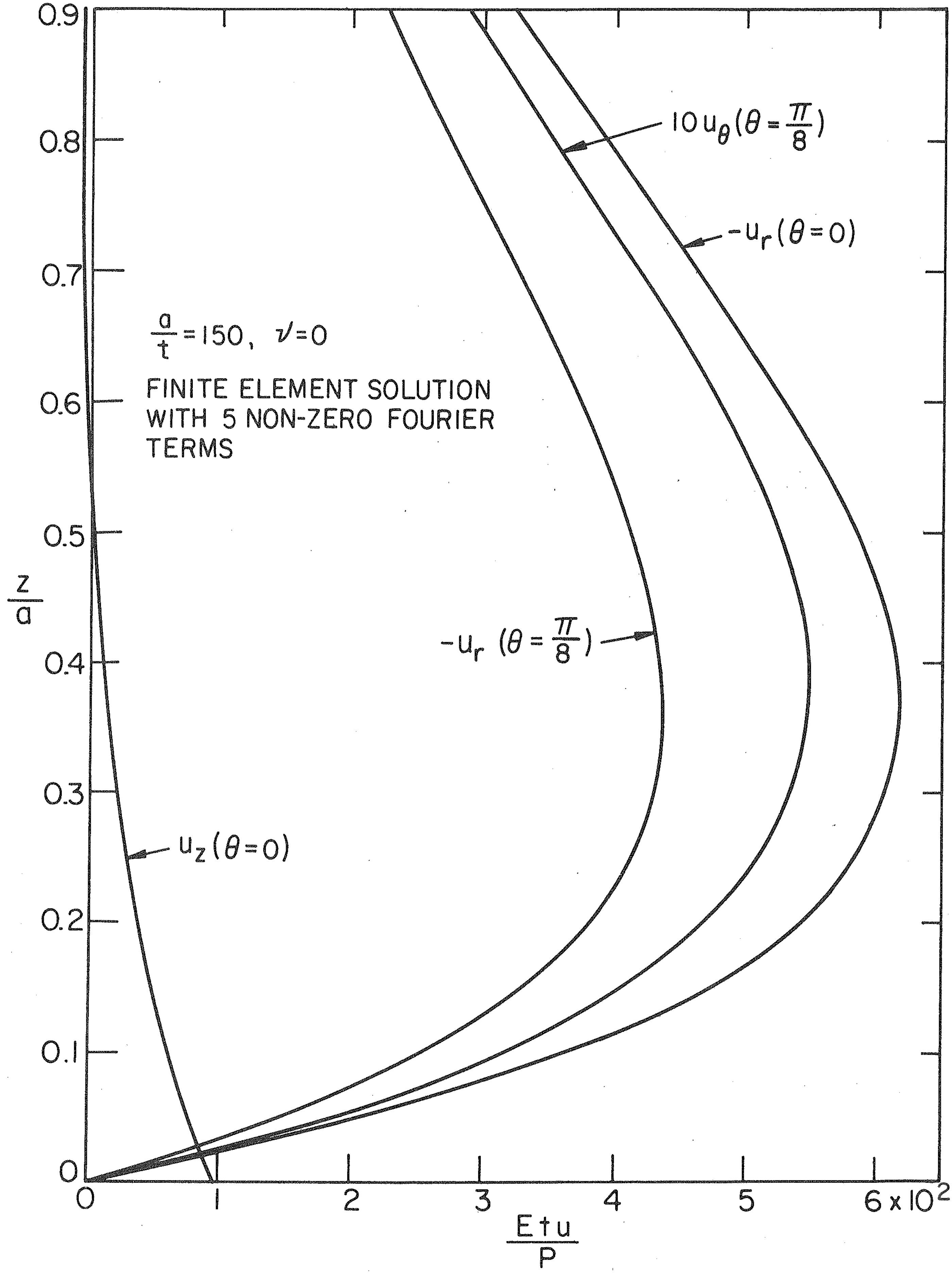

FIG. 10 DISPLACEMENTS FOR COOLING TOWER ON 8 SUPPORTS

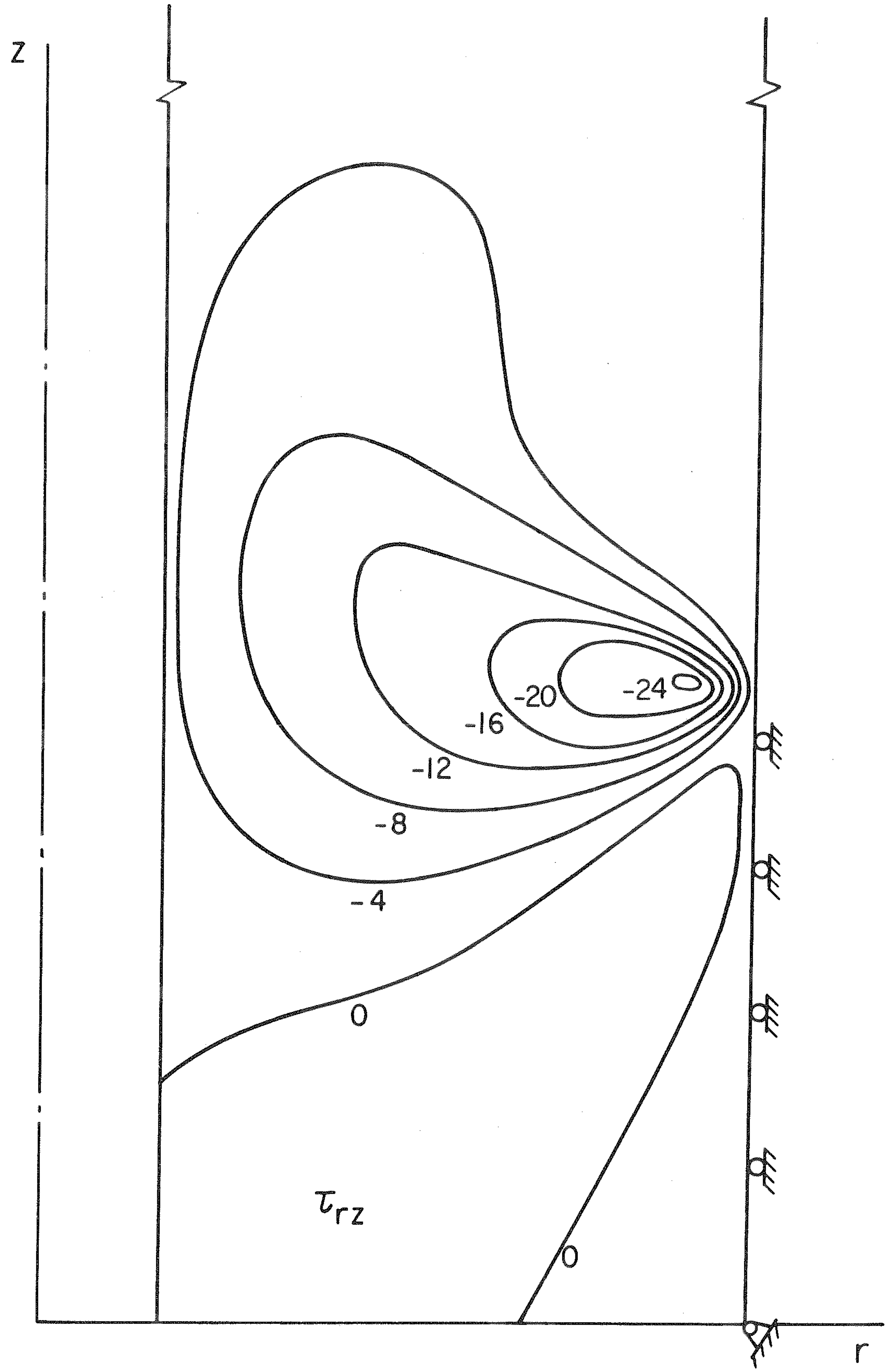

FIG. 11 SIMULATED LATERAL LOAD PROBLEM
SHEAR STRESS AT $\theta = 0$

APPENDIX A

Code Description

The following is a list of the routines described in Fig. A-1 and their functions as referenced by the deck name.

NAOS

Main program and calling sequence. Sets up the surface tractions, body and thermal forces and displacement boundary conditions for each non-zero Fourier term. Stores the necessary boundary condition data on tape, logical unit 8, in the binary mode.

The calling sequence is as follows:

1. LAYOUT (MESH)
2. BLOCKS (BLOK)
3. BACSUB (BCSB)
4. STRESS (NSTR)
5. SHLSTR (NSLS)

   Recycles 2-5 for each Fourier term

6. TSTRES (TSTR)

   Recycles 1-6 for each problem

MESH

Input subroutine. Accepts all input (see Appendix B) and generates a mesh of quadrilateral or triangular elements in the r-z plane. Checks the magnitude of all essential variables against their maximum permissible size. Computes the maximum band width and calculates the nodal point forces from the traction boundary conditions.

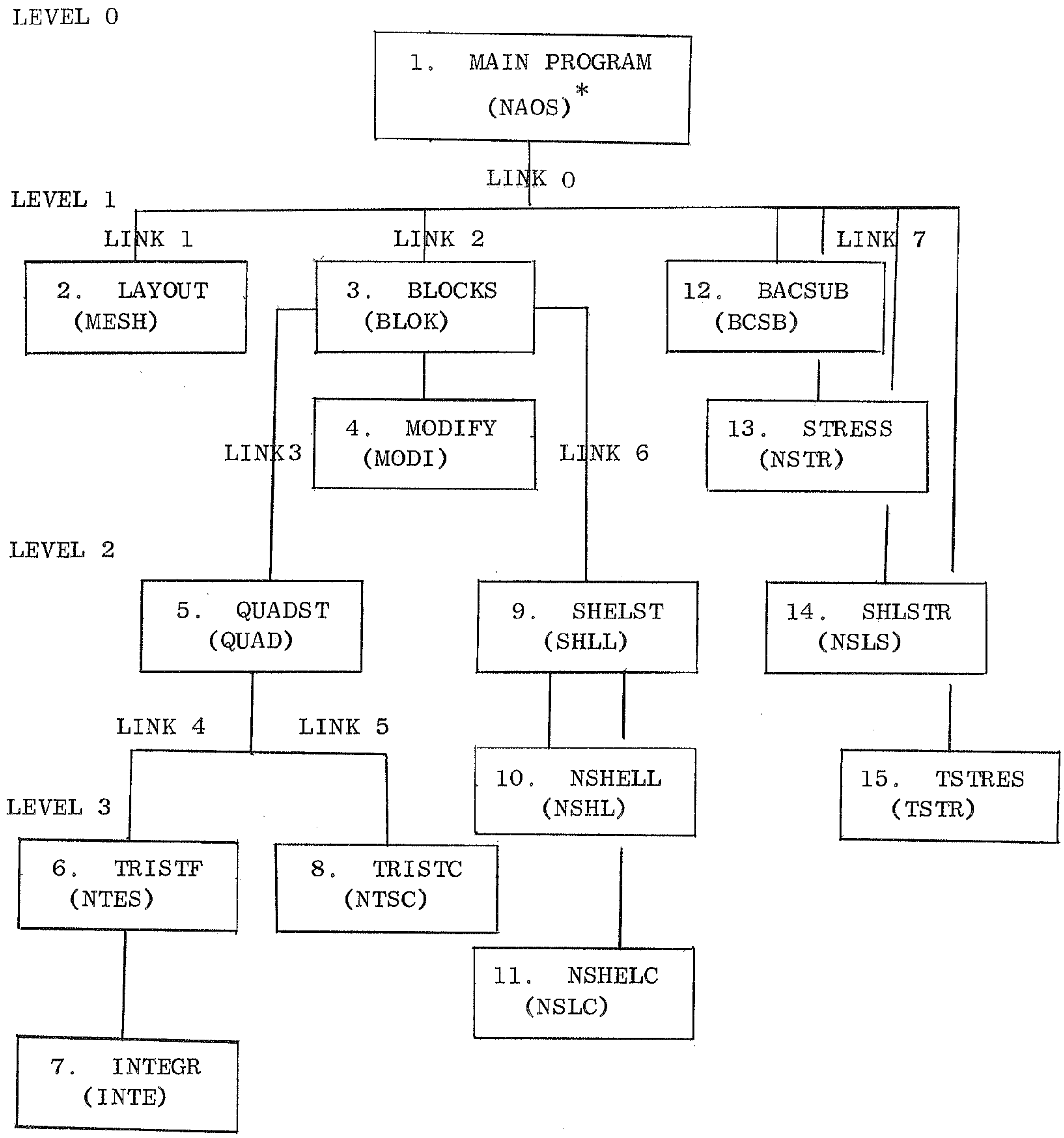

Fig. A-1

*( ) refer to deck names

BLOK

Stiffness assembly subroutine. Forms the structural stiffness matrix in blocks of 24 nodal points using the standard techniques of the direct stiffness method. Calls either the shell or quadrilateral stiffness subroutines and assembles the 12x12 (3 degrees of freedom for each of the 4 nodal points) element stiffness matrix. Sets up any displacement boundary conditions and calls the modification subroutine. Reduces the block of equations by Gaussian elimination and downward substitution (no pivoting). Stores the reduced equation on a directly addressable disk.

MODI

Displacement boundary condition modification subroutine. Makes all necessary changes to the stiffness matrix and load vector for zero and non-zero displacement boundary conditions.

QUAD

Quadrilateral element stiffness subroutine. Computes all element properties and the location of the center nodal point using the mean value of the 4 nodes of the quadrilateral or the 3 nodes of the triangle. Calls the appropriate (NTES for the 1st Fourier term or NTSC for all succeeding terms) triangular stiffness subroutine for either the 3 or 4 subtriangles. Stores or reads the necessary data for each triangular element stiffness in the binary mode on tape, logical unit 9. Sums the triangular element stiffnesses into the quadrilateral element stiffness matrix and eliminates the equations on the center node using the same Gaussian elimination as in BLOK. Stores the reduced equations for the center node and element data on a directly addressable disk. Rotates the unknowns for the sloping roller boundary condition into the new coordinate system (r' and z', see Fig. B2).

NTES

Nonaxisymmetric triangular element stiffness subroutine. Forms the 9x9 triangular element stiffness matrix for the first non-zero Fourier term. Calls INTE for the required triangular element volume integrals.

INTE

Triangular element volume integral subroutine. Computes the 10 required volume integrals for a triangular element.

NTSC

Nonaxisymmetric triangular element stiffness continuation subroutine. Computes the triangular element stiffness matrix for succeeding Fourier terms using the stiffness matrix and volume integrals computed for the first term.

SHLL

Shell subroutine. Initializes and calls the appropriate shell stiffness subroutine (NSHL for the first non-zero Fourier term and NSLC for succeeding terms).

NSHL

Nonaxisymmetric shell stiffness subroutine. Computes the shell element properties and shell volume integrals (by a 10 point Gaussian quadrature formula) and forms the 12x12 shell element stiffness matrix for the first non-zero Fourier term. The 6 displacement components are stored in the interior nodal point positions (see Appendix B) and the 2 rotation components are stored in the radial displacement positions of the exterior nodes, the remaining 4 degrees of freedom are left blank. Stores the necessary shell element data on a directly addressable disk.

NSLC

Nonaxisymmetric shell stiffness continuation subroutine. Computes the shell stiffness matrix for succeeding Fourier terms using the element data and shell integrals computed for the first term.

BCSB

Backsubstitution subroutine. Performs backsubstitution on each block of reduced equations read from a directly addressable disk and computes the nodal point displacements and rotations.

NSTR

Nonaxisymmetric element stress subroutine. Outputs the magnitude of the Fourier coefficients of the nodal point displacements. Computes and outputs the magnitude of the Fourier coefficients of the element stresses and strains at the center node of each quadrilateral or triangular element using the element data and reduced equations for the center node stored on a directly addressable disk. Also computes and outputs the principal stresses and strains at $\theta = 0$ and the normal stress and shear stress at the center node acting on a plane parallel to the J-K nodal points. Stores the 3 components of displacement and 6 components of stress on a directly addressable disk.

NSLS

Nonaxisymmetric shell stress subroutine. Computes and outputs the magnitude of the Fourier coefficients of the generalized shell stresses at the shell nodal points using the element data stored on a directly addressable disk.

TSTR

Total stress subroutine. Computes and outputs the total displacements for each of the 3 components at each nodal point for the required θ-stations (see Appendix B) by summing the contributions of each Fourier term at each nodal point. Computes and outputs the total stresses for each of the 6 stress components at each elasticity element at the same θ-stations as the displacements using the same summation procedure. Both displacements and stresses are read from a directly addressable disk.

APPENDIX B

Input User's Manual

As with every problem, the actual body together with its boundary conditions, must be moldeled into a system which can be analyzed by the finite element method. Primarily, this consists of subdividing the axisymmetric solid into a system of quadrilateral and triangular elements interconnected at their verticies (see Fig. B1). This is of course the critical stage at which the analyst must insure that the model captures the actual kinematic behavior of the body.

In the analysis of loading conditions which possess only even Fourier terms, it is readily seen that on the z-axis, $r = 0$, $u_r = u_\theta = 0$. However, for a general loading condition containing both odd and even terms neither u_r nor u_θ need be zero. Therefore, for solids with nodal points located at $r = 0$, no boundary conditions should be specified in general.

Input Data

The following is a description of the input data required to analyze an axisymmetric body subjected to arbitrary non-axisymmetric loads. The description is by card sections, and where appropriate the number of cards procedes the name.

1. 1- Title Card (12A6)*

   Columns 1-72 Arbitrary Problem Identification

2. 1- Control Card (7I5, 4F10.0, I3, I2)

   Columns 1-5 Number of Nodal Points (NUMNP)

   6-10 Number of Elements (NUMEL)

   11-15 Number of Materials (NUMMAT)

   16-20 Number of Pressure Boundary Conditions (NUMPC)

   21-25 Number of Fourier Coefficients (NUMFOU)

   26-30 Number of Print Angles (NANGLE)

   31-35 Number of First Shell Element (NCUT)

   36-45 Lateral Acceleration (ACELR)

   46-55 Axial Acceleration (ACELZ)

   56-65 Angular Velocity (ANGFQ)

   66-75 Reference Temperature (TO)

   78 Displacement Tape Storage Tag (NTAPE)

   80 Mesh Check Tag (ISTOP)

All material numbers greater than or equal to NCUT are taken to be shell elements, and any material number less than NCUT is taken to be an elasticity element. If no shell elements are present NCUT should be set to 26. If no

*Numbers in ( ) indicate FORMAT.

elasticity elements are present NCUT should be set to 0.

If NTAPE > 0, then the displacements for each non-zero Fourier term are stored in binary mode on tape, logical unit 3.

If ISTOP > 0, then a mesh check is obtained, where the mesh is generated and listed, and the values of all essential parameters are checked against their allowable values. The program is stopped in MESH and no further action is taken.

3. 3* NUMMAT - Material Identification Cards

Card 1 (I5, F10.0)

Columns 1-5 Material Number (MTYPE)

6-15 Material Density (RO)

Card 2 (6F10.0) (E-Array)

Columns	Elasticity Element	Shell Element	Isotropic Value
1-10	C_{rr}	C_{ss}	$\lambda+2\mu$
11-20	C_{rz}	$C_{s\theta}$	λ
21-30	$C_{r\theta}$	-	λ
31-40	C_{zz}	$C_{\theta\theta}$	$\lambda+2\mu$
41-50	$C_{z\theta}$	-	λ
51-60	$C_{\theta\theta}$	-	$\lambda+2\mu$

Card 3 (6F10.0) (E-Array Continuation)

1-10	G_{rz}	$G_{s\theta}$	μ
11-20	$G_{r\theta}$	-	μ
21-30	$G_{z\theta}$	-	μ

31-40	α_r	α_s	α
41-50	α_z	α_θ	α
51-60	α_θ	-	α

The values for an elasticity element correspond to the array in Equation (8) and for a shell element to Equation (35). The isotropic value is the value when the material is assumed to be isotropic in terms of the Lamé constants.

4. Nodal Point Cards (I5, F5.0, 6F10.0)

Columns 1-5 Nodal Point Number (N)

6-10 Boundary Condition Code (CODE)

11-20 Radial Coordinate ($R \geq 0$)

21-30 Axial Coordinate (Z)

31-40 Radial Force or Displacement (UR)

41-50 Axial Force or Displacement (UZ)

51-60 Theta Force or Displacement (UT)

61-70 Nodal Point Temperature (T)

In general, every nodal point must be defined but since the program has an automatic mesh generation feature, a minimum of 2 nodal points per row need be input and the intervening points will be assigned coordinates based on a linear interpolation procedure. For example, if nodal point 1 is the first point in a row with coordinates (2.5, 5.4) and nodal point 11 is the next point defined with coordinates (12.5, 10.4), then nodal point 2 will be located at (3.5, 5.9), etc. The boundary condition CODE will be set 0 unless points 1 and 11 have the same CODE, in which case all intervening points will be assigned the same CODE as the two end points. The radial, axial and theta forces or displacements will be set 0 in all cases. The temperature will be interpolated in the same manner as the coordinates.

If N is an "exterior shell nodal point"*, then the temperature field on the nodal point card is interpreted to be the shell thickness at that point and should not be zero in general. Also, an interior shell nodal point must not be a sloping roller (Fig. B2) but may have any other boundary condition code.

The boundary condition code is interpreted in the following manner.

	Specified Quantities		
CODE	Radial (r)	Axial (z)	Theta (θ)
0	Force	Force	Force
1	Displ	Force	Force
2	Force	Displ	Force
3	Displ	Displ	Displ
4	Force	Force	Displ
5	Displ	Displ	Force
6	Displ	Force	Displ
7	Force	Displ	Displ
$-\beta$	Force	Displ	Force

*See paragraph on shell elements in element card section.

If the CODE is specified as a negative quantity, $-\beta$, then the nodal point is assumed on a sloping roller in the r-z plane as shown in Fig. B2. β is taken as the angle the roller makes with the r-axis in radians. For example, if code is -.001 then β is taken as .001 radians. This would closely approximate 2 CODE, if CODE is $-\pi/2$ then $\beta = \pi/2$ which is the same as a 1 CODE. The specified quantities become Radial = $F_{r'}$, Axial = u_z' and Theta = F_θ. A θ-displacement boundary condition may not be used with the sloping roller.

If N is an "exterior shell nodal point," then only two values of CODE may be used. An "0" value means that an external moment is specified in the radial force field. A "1" value means that the slope of the tangent is specified in the radial force field. The axial and theta force fields are not used for exterior shell nodes.

5. Element Cards (6I5)

Columns	1-5	Element Number (N)
	6-10	I^{th} Nodal Point (IX-Array)
	11-15	J^{th} Nodal Point (IX-Array)
	16-20	K^{th} Nodal Point (IX-Array)
	21-25	L^{th} Nodal Point (IX-Array)
	26-30	Material Number (MAT, IX-Array)

In general, every element must be defined but with the automatic mesh generation feature a minimum of 1 element per row need be input. For example if element 10 is read with values I=12, J=13, K=24, L=23 and MAT=1 and the next element read is element 15 with values I=23, J=24, K=35, L=35 and MAT=2, then element 11 would be assigned values 13, 14, 25, 24 and 1, element 12 values 14, 15, 26, 25 and 1, etc. That is the I, J, K, L values are increment by 1 while the material number if held constant.

The nodal point array on the element cards must of course correspond to nodal points on the nodal point cards and the material number must correspond to materials in the material cards.

The I, J, K, L values may be assigned arbitrarly for elasticity elements except that they must be a counter-clockwise permutation (see Fig. B3). If a triangular element is desired then K must equal L, and any node may be chosen for the K^{th} node (see Fig. B3).

When MAT ≥ NCUT then the element is taken as a shell element and the following restrictions hold.

(1) I & J must be the nodes connected to the elasticity elements (interior shell nodal points) and K and L must be the exterior shell nodal points (see Fig. B4).

(2) $K \neq L$

(3) $R(I) = R(L)$, $Z(I) = Z(L)$, $R(J) = R(K)$, and $Z(J) = Z(K)$ without exception.

(4) The thickness of the shell at nodal points K and L should be non-negative. The thickness of a shell element is assumed constant over the shell length and its value is taken as the average of the thickness at K and L.

(5) The material properties for a shell element should correspond to the requirements described in the section on the material cards.

(6) The boundary condition code for nodes K and L must be prescribed in accordance with the procedure outlined in the paragraph on exterior shell nodes in the nodal point card section.

6. <u>NUMPC - Traction Boundary Condition Cards (2I5,2F10.0)</u>

Columns 1-5 I^{th} Nodal Point (IBC)

6-10 J^{th} Nodal Point (JBC)

11-20 Normal Traction (PN)

21-30 Shear Traction (PT)

This section is operable only if NUMPC > 0. Nodes I and J must be chosen such that the body is on the left of the line going from I to J. (This requires that I and J be in the same sequence on the traction cards as on the element cards.) Positive tractions are indicated on the diagram in Fig. B5, and the tractions are assumed constant over the length of the element boundary.

If non-constant tractions are desired, then the surface integral in Equations (18d) or (23d) must be carried out explicitly for the desired loading, and the value of the nodal point forces inserted on the nodal point cards.

The exterior nodes of a shell element should not appear in the traction boundary conditions. If they do appear, they will be ignored.

7. Fourier Coefficient Cards (8F10.0)

There are 3 sets of coefficients in this section; the first section contains the Fourier Force Coefficients of the surface tractions and nodal point forces, the second section contains the Fourier Displacement Coefficients of the non-zero displacement boundary conditions and the third section contains the Fourier Thermal Coefficients of the temperature distribution.

Let

$$p(\theta) = \sum_{n=o}^{N} a_n \cos n\theta$$

then

$$a_o = \frac{1}{2\pi} \int_o^{2\pi} p(\theta)\, d\theta, \quad a_n = \frac{1}{\pi} \int_o^{2\pi} p(\theta) \cos n\theta \, d\theta \qquad n = 1, 2, \ldots\ldots n$$

and denote

$$u(\theta) = \sum_{n=o}^{N} b_n \cos n\theta$$

$$T(\theta) = \sum_{n=o}^{N} c_n \cos n\theta$$

The resulting Fourier coefficient cards are as follows:

Fourier Force Coefficient Cards

Card 1	Columns	1-10	a_0
		11-20	a_1
		21-30	a_2
		31-40	a_3
			etc.

Card 2 (if necessary)	1-10	a_8
	11-20	a_9
		etc.

Fourier Displacement Coefficient Cards

Card 1	1-10	b_0
	11-20	b_1
		etc.
Card 2 (if necessary)	1-10	b_8
	11-20	b_9
		etc.

Fourier Thermal Coefficient Cards

Card 1	1-10	c_0
	11-20	c_1
		etc.

Thus the loads in each Fourier term are the product of the Fourier Force Coefficients for that term and the nodal point forces (including all surface tractions) input on the nodal point and traction boundary condition cards. The non-zero displacement boundary conditions are the product of the Fourier Displacement Coefficients and the nodal point displacements input on the nodal point cards. For most problems this section is inoperable since the displacement boundary conditions are zero. However, there must always be the same number of cards in each set even if they are blank (zeros). The nodal point temperatures are the product of the Fourier Thermal Coefficients and the nodal point temperatures inserted on the nodal point cards. Again, the same number of cards must appear in this set even if they are blank.

It should be noted that since the shell thickness is stored in the nodal point temperature array for exterior shell nodal points a problem occurs when the temperature (thickness) is multiplied by the Fourier Thermal Coefficients. What must be done for problems with shell elements is to set the Fourier Thermal Force Coefficient to 1 for the first non-zero term. If thermal problems are desired on geometries with shell elements, then the nodal point temperatures and Fourier Thermal Coefficients must be modified such that the first non-zero thermal coefficient is 1.

8. 1 or 0 - Angle Station Cards (4F10.0)

This option is operable only under certain values of NANGLE as listed below. The purpose of this section is to define those angle stations in the θ-direction at which displacements and stresses are summed and outputed by TSTRES.

NANGLE	ACTION TAKEN
1. < 0	No cards read. No Fourier summation.
2. 0	No cards read. Summation at $\theta = 0$.
3. 1-4	1 card (1-4 fields) read. Each field is the angle of summation in radians.
4. > 4	1 card (4 fields) read. Same as 3.

Typical Data Input

On the following pages the actual data input for the simulated lateral load problem (see Fig. B1) is listed. The actual modeling and interpretation of results is discussed in the Results Section of the report.

```
SIMULATED LATERAL LOAD PROBLEM
 144  119     2    11     5     4     2         0         0         0         0  0
    1
     17115      16443      16443      17115      16443      17115
       336        336        336
    3
  40300000   17300000   17300000   40300000   17300000   40300000
  11500000   11500000   11500000
    1             2.5          0
    6     5      10.0          0
    7     1      10.0          0                                    0.05
    8             2.5        1.5
   13     1      10.0        1.5
   14     1      10.0        1.5                                    0.05
   15             2.5        3.0
   20     1      10.0        3.0
   21     1      10.0        3.0                                    0.05
   22             2.5        4.5
   27     1      10.0        4.5
   28     1      10.0        4.5                                    0.05
   29             2.5        6.0
   34     1      10.0        6.0
   35     1      10.0        6.0                                    0.05
   36             2.5        7.5
   41     1      10.0        7.5
   42     1      10.0        7.5                                    0.05
   43             2.5        9.0
   48            10.0        9.0
   49            10.0        9.0                                    0.05
   50             2.5       10.5
   55            10.0       10.5
   56            10.0       10.5                                    0.05
   57             2.5       12.0
   62            10.0       12.0
   63            10.0       12.0                                    0.05
   64             2.5       13.5
   69            10.0       13.5
   70            10.0       13.5                                    0.05
   71             2.5       15.0
   76            10.0       15.0
   77            10.0       15.0                                    0.05
   78             2.5       17.5
   83            10.0       17.5
   84            10.0       17.5                                    0.05
   85             2.5       20.0
   90            10.0       20.0
   91            10.0       20.0                                    0.05
   92             2.5       22.5
   97            10.0       22.5
   98            10.0       22.5                                    0.05
   99             2.5       25.0
  104            10.0       25.0
  105            10.0       25.0                                    0.05
  106             2.5       27.5
  111            10.0       27.5
  112            10.0       27.5                                    0.05
  113             2.5       30.0
  118            10.0       30.0
  119            10.0       30.0                                    0.05
```

120			2.5		30.764	
125			9.563		32.924	
126			9.563		32.924	0.05
127			2.5		31.686	
132			8.290		35.592	
133			8.290		35.592	0.05
134			2.5		33.087	
138			6.293		37.772	
139			6.293		37.772	0.05
140			2.5		34.906	
143			4.540		38.910	
144			4.540		38.910	0.05
1	8	1	2	9	1	
6	13	6	7	14	3	
7	15	8	9	16	1	
12	20	13	14	21	3	
13	22	15	16	23	1	
18	27	20	21	28	3	
19	29	22	23	30	1	
24	34	27	28	35	3	
25	36	29	30	37	1	
30	41	34	35	42	3	
31	43	36	37	44	1	
36	48	41	42	49	3	
37	50	43	44	51	1	
42	55	48	49	56	3	
43	57	50	51	58	1	
48	62	55	56	63	3	
49	64	57	58	65	1	
54	69	62	63	70	3	
55	71	64	65	72	1	
60	76	69	70	77	3	
61	78	71	72	79	1	
66	83	76	77	84	3	
67	85	78	79	86	1	
72	90	83	84	91	3	
73	92	85	86	93	1	
78	97	90	91	98	3	
79	99	92	93	100	1	
84	104	97	98	105	3	
85	106	99	100	107	1	
90	111	104	105	112	3	
91	113	106	107	114	1	
96	118	111	112	119	3	
97	120	113	114	121	1	
102	125	118	119	126	3	
103	127	120	121	128	1	
108	132	125	126	133	3	
109	134	127	128	128	1	
111	134	128	129	135	1	
114	138	132	133	139	3	
115	140	134	135	135	1	
116	140	135	136	141	1	
119	143	138	139	144	3	
41	48		1			
48	55		1			
55	62		1			
62	69		1			
69	76		1			

```
 76   83          1
 83   90          1
 90   97          1
 97  104          1
104  111          1
111  118          1
   200.0    314.16    133.33         0    -26.67
       1
       1
     0.0 0.7853957 1.5707914 3.1415927
```

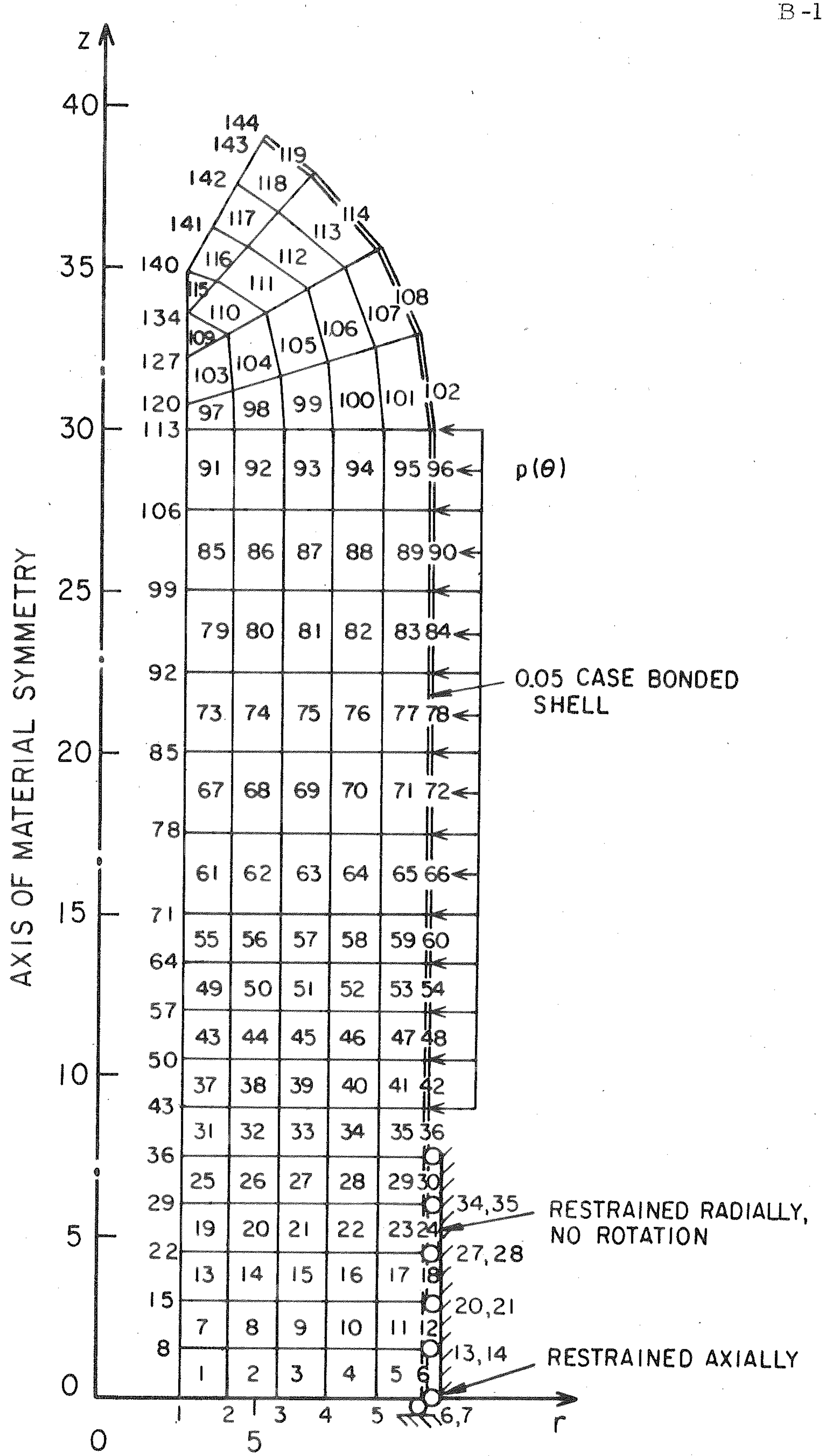

FIG. B1 MESH FOR SIMULATED LATERAL LOAD PROBLEM

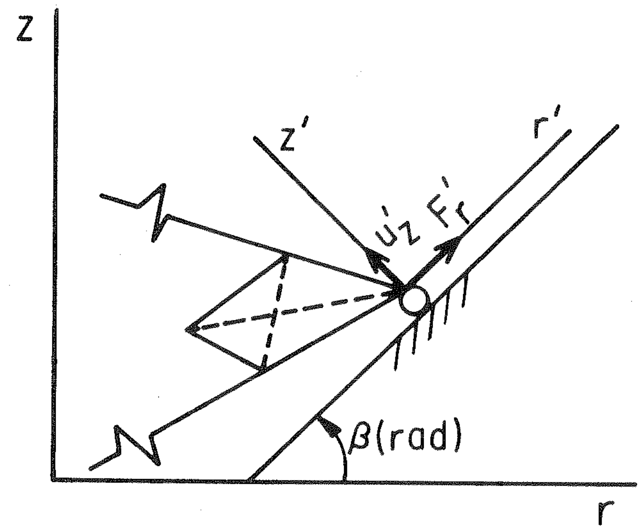

FIG. B2 SLOPING ROLLER

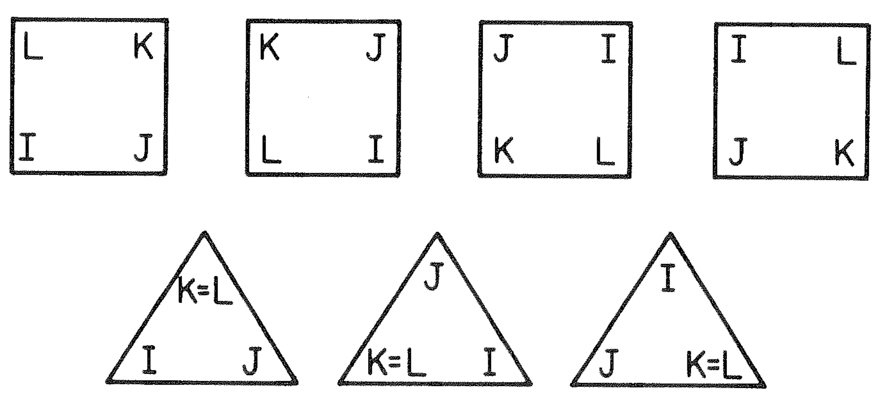

FIG. B3 ELEMENT ARRAY

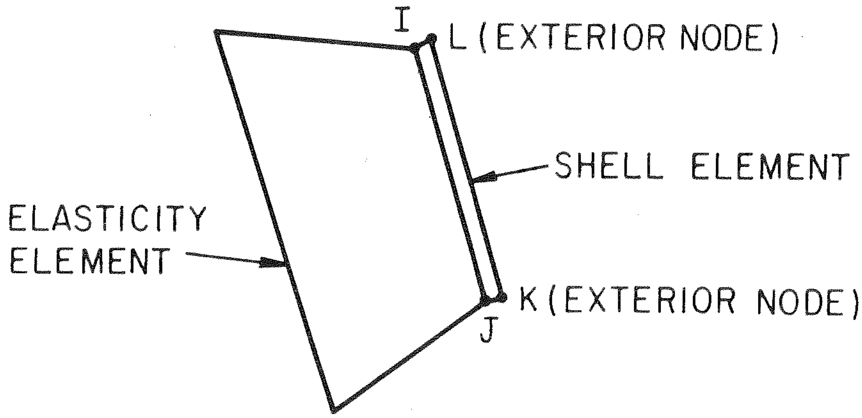

FIG. B4 SHELL-ELASTICITY INTERFACE

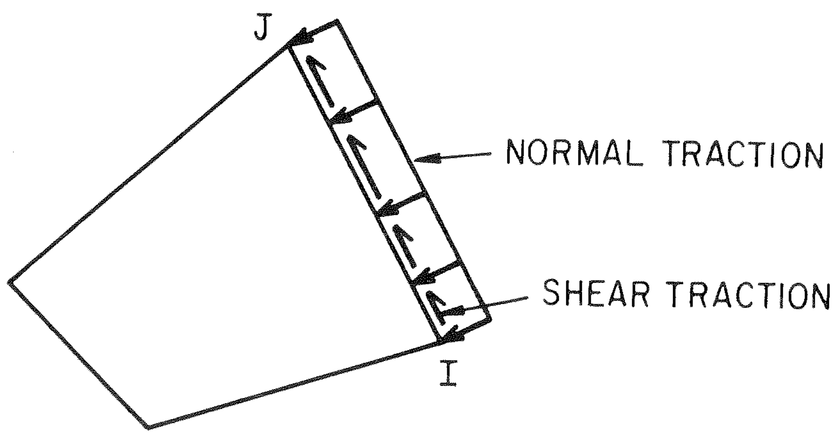

FIG. B5 PRESSURE BOUNDARY CONDITION

APPENDIX C

PROGRAM LISTINGS

DECKS

1. NAOS
2. MESH
3. BLOK
4. MODI
5. QUAD
6. NTES
7. INTE
8. NTSC
9. SHLL
10. NSHL
11. NSLC
12. BCSB
13. NSTR
14. NSLS
15. TSTR

```
$IBFTC NAOS     DECK,LIST,REF
C
C*****    NON-AXISYMMETRIC ORTHOTROPIC PROGRAM WITH SHELL
C
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /BANARG/
     1 A(72,144), B(144), NNP, ND, ND2, NCOUNT, MBAND, NUMBLK, NCONTD,
     2 NCONTS
      COMMON /FORIER/
     1 NFOUR, NF, NF2, XNF, NFT, NUMFOU, FORCOF(50,3), HED(12)
      REAL
     1 NF, NF2
C
C*****    INITIALIZATION
C
      NNP=24
      ND=72
      ND2=144
      NCOUNT=5184
C
C*****    SET UP THE MESH
C
   10 CALL LAYOUT
      TEMP=ACELR
      XNF=0.0
      ACELR=0.0
      NCONTD=3*NUMNP
      NCONTS=6*NUMEL
      IF (NTAPE.GT.0) REWIND 3
C
C*****    STORE BOUNDARY CONDITION DATA ON DISK
C
      REWIND 8
      WRITE (8) (UR(N),UZ(N),UT(N),T(N),N=1,NUMNP)
      NFOUR=1
C
C*****    BEGIN FOURIER LOOP
C
   20 X=FORCOF(NFOUR,1)
      IF (X.EQ.0.0) GO TO 300
      Y=FORCOF(NFOUR,2)
      Q=FORCOF(NFOUR,3)
      REWIND 9
C
C*****    MODIFY BOUNDARY LOADS FOR FOURIER COEFFICIENTS
C
      DO 200 N=1,NUMNP
      C=CODE(N)
      IF (C) 120,100,50
   50 IC=IFIX(C)
      GO TO (110,120,130,140,150,160,170), IC
  100 UR(N)=X*UR(N)
      UZ(N)=X*UZ(N)
      UT(N)=X*UT(N)
      GO TO 190
```

```
  110 UR(N)=Y*UR(N)
      UZ(N)=X*UZ(N)
      UT(N)=X*UT(N)
      GO TO 190
  120 UR(N)=X*UR(N)
      UZ(N)=Y*UZ(N)
      UT(N)=X*UT(N)
      GO TO 190
  130 UR(N)=Y*UR(N)
      UZ(N)=Y*UZ(N)
      UT(N)=Y*UT(N)
      GO TO 190
  140 UR(N)=X*UR(N)
      UZ(N)=X*UZ(N)
      UT(N)=Y*UT(N)
      GO TO 190
  150 UR(N)=Y*UR(N)
      UZ(N)=Y*UZ(N)
      UT(N)=X*UT(N)
      GO TO 190
  160 UR(N)=Y*UR(N)
      UZ(N)=X*UZ(N)
      UT(N)=Y*UT(N)
      GO TO 190
  170 UR(N)=X*UR(N)
      UZ(N)=Y*UZ(N)
      UT(N)=Y*UT(N)
  190 IF (NFOUR.EQ.1) UT(N)=0.0
  200 T(N)=Q*T(N)
      IF (NFOUR.EQ.2) ACELR=TEMP
      NF=NFOUR-1
      NF2=NF*NF
C
C*****    SOLVE STRUCTURE FOR EACH NON-ZERO FOURIER TERM
C
      CALL BLOCKS
      CALL BACSUB
      IF (NTAPE.GT.0) WRITE (3) NFOUR,(UR(N),N=1,NCONTD)
      CALL STRESS
      IF (NCUT.NE.26) CALL SHLSTR
C
C*****    RESET BOUNDARY CONDITIONS FOR NEXT TERM
C
      IF (NFOUR+1.GT.NUMFOU) GO TO 400
      REWIND 8
      READ (8) (UR(N),UZ(N),UT(N),T(N),N=1,NUMNP)
  300 NFOUR=NFOUR+1
      ACELZ=0.0
      ACELR=0.0
      ANGFQ=0.0
      XNF=1.0
      IF (NFOUR.GT.NUMFOU) GO TO 400
      GO TO 20
C
C*****    SUM THE FOURIER COMPONENTS
C
  400 IF (NANGLE.GE.0) CALL TSTRES
      IF (NTAPE.GT.0) REWIND 3
```

```
      WRITE (6,2000)
C
C*****     DO THE NEXT PROBLEM
C
      GO TO 10
 2000 FORMAT (15H1END OF PROBLEM)
      END
```

```
$IBFTC MESH    DECK,LIST,REF
      SUBROUTINE LAYOUT
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /BANARG/
     1 A(72,144), B(144), NNP, ND, ND2, NCOUNT, MBAND, NUMBLK, NCONTD,
     2 NCONTS
      COMMON /FORIER/
     1 NFOUR, NF, NF2, XNF, NFT, NUMFOU, FORCOF(50,3), HED(12)
      DIMENSION
     1 IBC(500), JBC(500), PN(500), PT(500)
      EQUIVALENCE
     1 (IBC(1),A(1,1))
      DATA
     1 MAXNP /800/, MAXEL /700/, MAXMAT /25/, MAXFOU /50/
      REAL
     1 NF, NF2
C
C*****    READ AND PRINT OF CONTROL DATA
C
      READ (5,1000) HED,NUMNP,NUMEL,NUMMAT,NUMPC,NUMFOU,NANGLE,NCUT,
     1 ACELR, ACELZ, ANGFQ, TO, NTAPE, ISTOP
      WRITE(6,2000) HED,NUMNP,NUMEL,NUMMAT,NUMPC,NUMFOU,NANGLE,NCUT,
     1 ACELR, ACELZ, ANGFQ, TO, NTAPE
      IF (ISTOP.EQ.0) GO TO 5
      WRITE (6,2038)
    5 IF (NUMNP.LE.MAXNP) GO TO 10
      WRITE (6,2034) MAXNP
      ISTOP=1
   10 IF (NUMEL.LE.MAXEL) GO TO 20
      WRITE (6,2035) MAXEL
      ISTOP=1
   20 IF (NUMFOU.LE.MAXFOU) GO TO 30
      WRITE (6,2036) MAXFOU
      ISTOP=1
C
C*****    READ AND PRINT OF MATERIAL PROPERTIES
C
   30 DO 50 M=1, NUMMAT
      READ (5,1001) MTYPE, RO(MTYPE)
      IF (MTYPE.LE.MAXMAT.AND.MTYPE.GT.0) GO TO 40
      WRITE (6,2037) MAXMAT
      ISTOP=1
   40 WRITE (6,2001) MTYPE, RO(MTYPE)
      READ (5,1002) (E(J,MTYPE),J=1,12)
   50 WRITE(6,2002) (E(J,MTYPE),J=1,12)
C
C*****    READ AND PRINT OF NODAL POINT DATA
C
      WRITE(6,2013)
      L=0
   60 READ (5,1003) N, CODE(N), R(N), Z(N), UR(N), UZ(N), UT(N), T(N)
      NNL=L+1
      ZX=N-L
      DR=(R(N)-R(L))/ZX
```

```
      DZ=(Z(N)-Z(L))/ZX
      DT=(T(N)-T(L))/ZX
   70 L=L+1
      IF (N-L) 100, 90, 80
   80 CODE(L)=0.
      IF (CODE(NNL-1).EQ.CODE(N)) CODE(L)=CODE(N)
      R(L)=R(L-1)+DR
      Z(L)=Z(L-1)+DZ
      UR(L)=0.
      UZ(L)=0.
      UT(L)=0.0
      T(L)=T(L-1)+DT
      GO TO 70
   90 WRITE(6,2003) (K,CODE(K),R(K),Z(K),UR(K),UZ(K),UT(K),T(K),K=NNL,N)
      IF (NUMNP-N) 100, 110, 60
  100 WRITE (6,2030) N
      ISTOP=1
  110 CONTINUE
C
C*****    READ AND PRINT OF ELEMENT PROPERTIES
C
      WRITE(6,2014)
      N=0
  130 READ(5,1004) M,(IX(M,I),I=1,5)
  140 N=N+1
      IF (M-N) 185, 170, 150
  150 IX(N,1)=IX(N-1,1)+1
      IX(N,2)=IX(N-1,2)+1
      IX(N,3)=IX(N-1,3)+1
      IX(N,4)=IX(N-1,4)+1
      IX(N,5)=IX(N-1,5)
  170 WRITE (6,2004) N,(IX(N,I),I=1,5)
      IF (M-N) 185, 180, 140
  180 IF (NUMEL-N) 190, 190, 130
  185 WRITE (6,2031) N
      ISTOP=1
  190 CONTINUE
C
C*****    READ PRESSURE AND/OR SHEAR BOUNDARY STRESSES
C
      IF (NUMPC.LE.0) GO TO 210
      WRITE (6,2015)
      DO 200 L=1,NUMPC
      READ (5,1005) IBC(L), JBC(L), PN(L), PT(L)
  200 WRITE (6,2005) IBC(L), JBC(L), PN(L), PT(L)
C
C*****    DETERMINE THE BANDWIDTH AND COMPUTE THE SURFACE INTEGRALS
C
  210 MBAND=0
      DO 250 N=1,NUMEL
      DO 250 I=1,4
      IF (NUMPC.LE.0) GO TO 240
      J=I+1
      IF (I.EQ.4) J=1
      K=IX(N,I)
      L=IX(N,J)
      DO 230 MM=1,NUMPC
      IF (K.NE.IBC(MM).OR.L.NE.JBC(MM)) GO TO 230
```

```
      CK=CODE(K)
      CL=CODE(L)
      X=R(L)-R(K)
      Y=Z(L)-Z(K)
      XX=X/6.0
      YY=0.5*R(K)+XX
      XX=1.0+XX/YY
      FRK=YY*(PT(MM)*X-PN(MM)*Y)
      FRL=XX*FRK
      FZK=YY*(PT(MM)*Y+PN(MM)*X)
      FZL=XX*FZK
      IF (CK.LT.0.0) UR(K)=UR(K)+FRK*COS(CK)-FZK*SIN(CK)
      IF (CL.LT.0.0) UR(L)=UR(L)+FRL*COS(CL)-FZL*SIN(CL)
      IF (CK.EQ.1.0) UZ(K)=UZ(K)+FZK
      IF (CL.EQ.1.0) UZ(L)=UZ(L)+FZL
      IF (CK.EQ.2.0) UR(K)=UR(K)+FRK
      IF (CL.EQ.2.0) UR(L)=UR(L)+FRL
      IF (CK.NE.0.0) GO TO 220
      UR(K)=UR(K)+FRK
      UZ(K)=UZ(K)+FZK
  220 IF (CL.NE.0.0) GO TO 230
      UR(L)=UR(L)+FRL
      UZ(L)=UZ(L)+FZL
  230 CONTINUE
  240 DO 250 J=1,4
      K=IABS(IX(N,I)-IX(N,J))
      IF (K.GT.MBAND) MBAND=K
  250 CONTINUE
      MBAND=3*MBAND+3
      IF (MBAND.LE.ND) GO TO 260
      WRITE (6,2032) MBAND
      ISTOP=1
C
C*****    READ FOURIER COEFFICIENTS
C
  260 WRITE (6,2016)
      READ (5,1006) (FORCOF(N,1),N=1,NUMFOU)
      READ (5,1006) (FORCOF(N,2),N=1,NUMFOU)
      READ (5,1006) (FORCOF(N,3),N=1,NUMFOU)
      DO 320 N=1,NUMFOU
      M=N-1
  320 WRITE (6,2006) M, (FORCOF(N,I),I=1,3)
C
C*****    READ ANGLES OF STRESS OUTPUT
C
      IF (NANGLE.GT.0) GO TO 350
      XANG(1)=0.0
      GO TO 380
  350 IF (NANGLE.GT.4) NANGLE=4
      READ (5,1006) (XANG(N),N=1,NANGLE)
      WRITE (6,2007) (XANG(N),N=1,NANGLE)
C
C*****    COMPUTE FIRST TAPE STORAGE OF STIFFNESS ARRAYS
C
  380 DO 400 N=1,NUMFOU
      IF (FORCOF(N,1).EQ.0.0) GO TO 400
      NFT=N
      GO TO 900
```

```
  400 CONTINUE
      WRITE (6,2033)
      ISTOP=1
C
C*****    DIAGNOSTICS - TRANSFER
C
  900 IF (ISTOP.EQ.0) RETURN
      STOP
C
C*****    FORMAT STATEMENTS
C
 1000 FORMAT (12A6/7I5,4F10.0,I3,I2)
 1001 FORMAT (1I5,1F10.0)
 1002 FORMAT (6F10.0/6F10.0)
 1003 FORMAT(I5,F5.0,6F10.0)
 1004 FORMAT(6I5)
 1005 FORMAT (2I5,2F10.0)
 1006 FORMAT (8F10.0)
 2000 FORMAT (1H1  12A6/
     1   30HONUMBER OF NODAL POINTS------ I3/
     2   30HONUMBER OF ELEMENTS---------- I3/
     3   30HONUMBER OF DIFF. MATERIALS--- I3/
     4   30HONUMBER OF PRES./SHEAR CARDS- I3/
     5   30HONUMBER OF FOURIER TERMS----- I3/
     X   30HONUMBER OF PRINT ANGLES------ I3/
     6   30HOSHELL CUTOFF NUMBER--------- I3/
     X   30HORADIAL ACCELERATION--------- E12.4/
     7   30HOAXIAL ACCELERATION---------- E12.4/
     8   30HOANGULAR VELOCITY------------ E12.4/
     9   30HOREFERENCE TEMPERATURE------- F12.2/
     X   30HOSTRESS TAPE STORAGE TAG----- I3//)
 2001 FORMAT (17HOMATERIAL NUMBER=I3,15H, MASS DENSITY=E12.4//)
 2002 FORMAT (6X,4HC-RR 6X,4HC-RZ 6X,4HC-RT 6X,4HC-ZZ 6X,4HC-ZT 6X,
     1 4HC-TT 6X,4HG-RR 6X,4HG-ZZ 6X,4HG-TT 3X,7HALPHA-R 3X,7HALPHA-Z
     2 3X,7HALPHA-T/(1P12E10.3))
 2013 FORMAT(132H1NODAL POINT         TYPE  R-ORDINATE  Z-ORDINATE  R LOA
     1D OR DISPLACEMENT  Z LOAD OR DISPLACEMENT  T LOAD OR DISPLACEMENT
     2TEMPERATURE//)
 2003 FORMAT(I12,F12.2,2F12.3,3E24.7,F12.2)
 2014 FORMAT(52H1     ELEMENT     I     J     K     L     MATERIAL  )
 2004 FORMAT (1I13, 4I6, 1I13)
 2015 FORMAT (42H1PRESSURE AND/OR SHEAR BOUNDARY CONDITIONS//36H      I
     1  J    PRESSURE        SHEAR//)
 2005 FORMAT (2I6,2F12.3)
 2016 FORMAT (97H1 FOURIER NUMBER            FORCE COEFFICIENT   DISPLACEM
     1ENT COEFFICIENT          THERMAL COEFFICIENT//)
 2006 FORMAT (I16,3E27.8/)
 2007 FORMAT (38H1ANGLES OF STRESS PRINTOUTS IN RADIANS///(F20.4/))
 2030 FORMAT (26HONODAL POINT CARD ERROR N= I5)
 2031 FORMAT (23HOELEMENT CARD ERROR, N=I5)
 2032 FORMAT (27H1BANDWIDTH EXCEEDED, MBAND=I5)
 2033 FORMAT (25H3BAD FOURIER COEFFICIENTS)
 2034 FORMAT (46H3NUMBER OF NODAL POINTS EXCEEDS THE MAXIMUM OFI4)
 2035 FORMAT (42H3NUMBER OF ELEMENTS EXCEEDS THE MAXIMUM OFI4)
 2036 FORMAT (54H3NUMBER OF FOURIER COEFFICIENTS EXCEEDS THE MAXIMUM OF
     1 I3)
 2037 FORMAT (44H3A MATERIAL NUMBER IS OUTSIDE THE RANGE 0 TOI3)
 2038 FORMAT (16H MESH CHECK ONLY//)
      END
```

```
$IBFTC BLOK     DECK,LIST,REF
      SUBROUTINE BLOCKS
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /BANARG/
     1 A(72,144), B(144), NNP, ND, ND2, NCOUNT, MBAND, NUMBLK, NCONTD,
     2 NCONTS
      COMMON /FORIER/
     1 NFOUR, NF, NF2, XNF, NFT, NUMFOU, FORCOF(50,3), HED(12)
      COMMON /QUADST/
     1 S(15,15), P(15)
      DIMENSION
     1 LM(4)
      REAL
     1 NF, NF2
C
C*****    INITIALIZATION
C
      ISTOP=0
      NUMBLK=0
      DO 50 N=1,ND2
      B(N)=0.0
      DO 50 M=1,ND
   50 A(M,N)=0.0
C
C*****    FORM STIFFNESS MATRIX IN BLOCKS
C
   60 NUMBLK=NUMBLK+1
      NH=NNP*(NUMBLK+1)
      NM=NH-NNP
      NL=NM-NNP+1
      KSHIFT=3*NL-3
C
C*****    SELECT ELEMENT IN BLOCK
C
      DO 210 N=1, NUMEL
      IF (IX(N,5).LE.0) GO TO 210
      DO 80 I=1,4
      IF (IX(N,I).LT.NL) GO TO 80
      IF (IX(N,I).LE.NM) GO TO 90
   80 CONTINUE
      GO TO 210
   90 IF (IX(N,5).LT.NCUT) GO TO 100
      CALL SHELST(N,JSTOP)
      GO TO 110
  100 CALL QUADSF(N,JSTOP)
  110 IF(JSTOP.EQ.0) GO TO 120
      ISTOP=1
  120 IF(ISTOP.EQ.1) GO TO 210
C
C*****   ADD STIFFNESS AND FORCE VECTOR
C
      DO 140 I=1,4
  140 LM(I)=3*IX(N,I)-3
      DO 200 I=1,4
```

```
      DO 200 K=1,3
      II=LM(I)+K-KSHIFT
      KK=3*I-3+K
      B(II)=B(II)+P(KK)
      DO 200 J=1,4
      DO 200 L=1,3
      JJ=LM(J)+L-II+1-KSHIFT
      LL=3*J-3+L
      IF(JJ.LE.0) GO TO 200
      IF (ND.GE.JJ) GO TO 195
      WRITE (6,2002) N
      ISTOP=1
      GO TO 210
  195 A(JJ,II)=A(JJ,II)+S(KK,LL)
  200 CONTINUE
  210 CONTINUE
      IF (ISTOP.EQ.1) GO TO 390
C
C*****    ADD CONCENTRATED FORCES, MODIFY FOR DISPLACEMENT B. C.
C
      DO 300 N=NL,NH
      IF (N.GT.NUMNP) GO TO 310
      K=3*N-KSHIFT-2
      U=UR(N)
      C=CODE(N)
      IF (C) 270,295,240
  240 IC=IFIX(C)
      GO TO (260,270,230,280,290,250,265), IC
  230 K=K+1
      U=UZ(N)
      CALL MODIFY(K,U)
      K=K-1
      U=UR(N)
  250 CALL MODIFY(K,U)
      K=K+2
      U=UT(N)
      CALL MODIFY(K,U)
      K=K-2
      GO TO 295
  260 CALL MODIFY(K,U)
      GO TO 295
  265 K=K+2
      U=UT(N)
      CALL MODIFY(K,U)
      K=K-2
      GO TO 295
  270 U=UZ(N)
      K=K+1
      CALL MODIFY(K,U)
      K=K-1
      GO TO 295
  280 U=UT(N)
      K=K+2
      CALL MODIFY(K,U)
      K=K-2
      GO TO 295
  290 CALL MODIFY(K,U)
      K=K+1
```

```
      U=UZ(N)
      CALL MODIFY(K,U)
      K=K-1
  295 IF (N.GT.NM) GO TO 300
      B(K)=B(K)+UR(N)
      B(K+1)=B(K+1)+UZ(N)
      B(K+2)=B(K+2)+UT(N)
  300 CONTINUE
C
C*****    REDUCE BLOCK OF EQUATIONS
C
  310 DO 350 N=1,ND
      IF(A(1,N).EQ.0.) GO TO 350
      B(N)=B(N)/A(1,N)
      DO 340 L=2,MBAND
      IF(A(L,N).EQ.0.) GO TO 340
      C=A(L,N)/A(1,N)
      I=N+L-1
      J=0
      DO 330 K=L,MBAND
      J=J+1
  330 A(J,I)=A(J,I)-C*A(K,N)
      B(I)=B(I)-A(L,N)*B(N)
      A(L,N)=C
  340 CONTINUE
  350 CONTINUE
C
C*****    WRITE BLOCK OF REDUCED EQUATIONS ON DISK
C
  390 IF (NM.GE.NUMNP) GO TO 900
      IF (ISTOP.EQ.1) GO TO 60
      NTRACK=40*(NUMBLK-1)
      CALL WRDISK(NTRACK,A,NCOUNT)
      NTRACK=NTRACK+39
      CALL WRDISK(NTRACK,B,ND)
C
C*****    SHIFT BLOCK OF EQUATIONS UP FOR NEXT BLOCK
C
      DO 400 N=1,ND
      MM=ND+N
      B(N)=B(MM)
      B(MM)=0.
      DO 400 M=1,MBAND
      A(M,N)=A(M,MM)
  400 A(M,MM)=0.0
      GO TO 60
  900 IF (ISTOP.EQ.0) RETURN
      STOP
 2002 FORMAT (29H0BAND WIDTH EXCEEDS ALLOWABLE I4)
      END
```

```
$IBFTC MODI    DECK,LIST,REF
      SUBROUTINE MODIFY(N,X)
      COMMON/BANARG/
     1 A(72,144), B(144), NNP, ND, ND2, NCOUNT, MBAND, NUMBLK, NCONTD,
     2 NCONTS
      DO 250 M=2,ND
      K=N-M+1
      IF (K.LE.0) GO TO 235
      B(K)=B(K)-A(M,K)*X
      A(M,K)=0.0
  235 K=N+M-1
      IF (K.GT.ND2) GO TO 250
      B(K)=B(K)-A(M,N)*X
      A(M,N)=0.0
  250 CONTINUE
      A(1,N)=1.0
      IF (X.EQ.0.0) A(1,N)=0.0
      B(N)=0.0
      RETURN
      END
```

```
$IBFTC QUAD    DECK,LIST
      SUBROUTINE QUADSF(N,ISTOP)
C
C*****    THIS SUBROUTINE FORMS THE QUADRILATERAL STIFFNESS MATRIX
C
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /FORIER/
     1 NFOUR, NF, NF2, XNF, NFT, NUMFOU, FORCOF(50,3), HED(12)
      COMMON /QUADST/
     1 SK(15,15), F(15)
      COMMON /QADTRI/
     1 RRR(6), ZZZ(6), BF(9), XI(10), S(9,9), A(3,3), EE(56)
      DIMENSION
     1 D(12,3), CF(3), LM(9), C(3,3), TAPSTF(100)
      EQUIVALENCE
     1 (D(1,1),EE(18)), (CF(1),EE(54)), (XI(1),TAPSTF(1))
      DATA
     1 NCOUNT /56/, LM(3) /13/, LM(6) /14/, LM(9) /15/, NCONT /100/,
     2 MCOUNT /17/
      REAL
     1 NF, NF2
C
C*****    INITIALIZATION
C
      NTRACK=1319+N
      ISTOP=0
      DO 100 I=1,15
      F(I)=0.0
      DO 100 J=1,15
  100 SK(I,J)=0.0
C
C*****    CALCULATE ELEMENT VARIABLES OR READ THEM FROM THE DISK
C
      I=IX(N,1)
      J=IX(N,2)
      K=IX(N,3)
      L=IX(N,4)
      MTYPE=IX(N,5)
      IX(N,5)=-MTYPE
      IF (NFOUR.GT.NFT) GO TO 140
      IF (K.EQ.L) GO TO 110
      RRR(3)=0.25*(R(I)+R(J)+R(K)+R(L))
      ZZZ(3)=0.25*(Z(I)+Z(J)+Z(K)+Z(L))
      GO TO 120
  110 RRR(3)=(R(I)+R(J)+R(K))/3.0
      ZZZ(3)=(Z(I)+Z(J)+Z(K))/3.0
  120 EE(16)=RRR(3)
      EE(17)=ZZZ(3)
      DO 130 MM=1,9
  130 EE(MM)=E(MM,MTYPE)
      GO TO 150
  140 CALL RDDISK(NTRACK,EE,MCOUNT)
      RRR(3)=EE(16)
      ZZZ(3)=EE(17)
```

```
      TEMP=0.25*(T(I)+T(J)+T(K)+T(L))
      IF (K.EQ.L) TEMP=(T(I)+T(J)+T(K))/3.
  150 DO 170 MM=10,12
  170 EE(MM)=(TEMP-TO)*E(MM,MTYPE)
      C1=EE(1)*EE(10)+EE(2)*EE(11)+EE(3)*EE(12)
      C2=EE(2)*EE(10)+EE(4)*EE(11)+EE(5)*EE(12)
      EE(12)=EE(3)*EE(10)+EE(5)*EE(11)+EE(6)*EE(12)
      EE(11)=C2
      EE(10)=C1
      EE(13)=RO(MTYPE)*ANGFQ*ANGFQ
      EE(14)=-RO(MTYPE)*ACELR
      EE(15)=-RO(MTYPE)*ACELZ
C
C*****    FORM STIFFNESSES OF FOUR SUBTRIANGLES
C
      NM=4
      IF (K.EQ.L) NM=3
      DO 300 M=1,NM
      MM=M+1
      IF (M.EQ.NM) MM=1
      LM(7)=3*M
      LM(4)=LM(7)-1
      LM(1)=LM(4)-1
      LM(8)=3*MM
      LM(5)=LM(8)-1
      LM(2)=LM(5)-1
      I=IX(N,M)
      J=IX(N,MM)
      RRR(1)=R(I)
      RRR(2)=R(J)
      ZZZ(1)=Z(I)
      ZZZ(2)=Z(J)
      IF (NFOUR.GT.NFT) GO TO 250
      CALL TRISTF
      IF (XI(2).LE.0.0) ISTOP=1
      WRITE (9) (TAPSTF(K),K=1,NCONT)
      IF (ISTOP.EQ.0) GO TO 260
      WRITE (6,2000) N
      RETURN
  250 READ (9) (TAPSTF(K),K=1,NCONT)
      CALL TRISTC
  260 DO 200 I=1,3
      K=I+3
      DO 200 J=7,9
      S(I,J)=NF*S(I,J)
  200 S(K,J)=NF*S(K,J)
      DO 270 I=1,9
      II=LM(I)
      F(II)=F(II)+BF(I)
      DO 270 J=1,9
      JJ=LM(J)
      S(J,I)=S(I,J)
  270 SK(II,JJ)=SK(II,JJ)+S(I,J)
  300 CONTINUE
C
C*****    SOLVE FOR UNKNOWNS AT CENTER NODE
C
      C1=SK(14,14)*SK(15,15)-SK(14,15)*SK(14,15)
```

```
      C2=SK(13,13)*SK(15,15)-SK(13,15)*SK(13,15)
      C3=SK(13,13)*SK(14,14)-SK(13,14)*SK(13,14)
      C4=SK(13,15)*SK(14,15)-SK(13,14)*SK(15,15)
      C5=SK(13,14)*SK(14,15)-SK(13,15)*SK(14,14)
      C6=SK(13,15)*SK(13,14)-SK(13,13)*SK(14,15)
      DEN=SK(13,13)*C1+SK(13,14)*C4+SK(13,15)*C5
      IF (DEN.NE.0.0) GO TO 320
      WRITE (6,2001) N
      ISTOP=1
      RETURN
  320 CC=1.0/DEN
      C(1,1)=CC*C1
      C(1,2)=CC*C4
      C(1,3)=CC*C5
      C(2,2)=CC*C2
      C(2,3)=CC*C6
      C(3,3)=CC*C3
      C(2,1)=C(1,2)
      C(3,1)=C(1,3)
      C(3,2)=C(2,3)
      DO 330 I=1,12
      DO 330 J=1,3
      D(I,J)=0.0
      DO 330 K=1,3
      L=12+K
  330 D(I,J)=D(I,J)+SK(I,L)*C(K,J)
      DO 340 I=1,12
      DO 340 K=1,3
      L=12+K
      F(I)=F(I)-D(I,K)*F(L)
      DO 340 J=I,12
  340 SK(I,J)=SK(I,J)-D(I,K)*SK(J,L)
      DO 350 I=1,3
      CF(I)=0.0
      DO 350 J=1,3
      L=12+J
  350 CF(I)=CF(I)+C(I,J)*F(L)
      DO 360 I=1,12
      DO 360 J=I,12
  360 SK(J,I)=SK(I,J)
C
C*****     ROTATE UNKNOWNS IF REQUIRED
C
      DO 400 M=1,4
      MM=IX(N,M)
      CC=-CODE(MM)
      IF (CC.LE.0.0) GO TO 400
      DX=COS(CC)
      DY=SIN(CC)
      K=3*M
      J=K-1
      I=J-1
      F(I)=F(I)*DX+F(J)*DY
      F(J)=0.0
      DO 390 II=1,12,3
      IF (II.EQ.I) GO TO 390
      JJ=I+1
      KK=JJ+1
      C1=DX*SK(I,II)+DY*SK(J,II)
```

```
      C2=DX*SK(I,JJ)+DY*SK(J,JJ)
      C3=DX*SK(I,KK)+DY*SK(J,KK)
      SK(J,II)=-DY*SK(I,II)+DX*SK(J,II)
      SK(II,J)=SK(J,II)
      SK(J,JJ)=-DY*SK(I,JJ)+DX*SK(J,JJ)
      SK(JJ,J)=SK(J,JJ)
      SK(J,KK)=-DY*SK(I,KK)+DX*SK(J,KK)
      SK(KK,J)=SK(J,KK)
      SK(I,II)=C1
      SK(II,I)=C1
      SK(I,JJ)=C2
      SK(JJ,I)=C2
      SK(I,KK)=C3
      SK(KK,I)=C3
  390 CONTINUE
      TEMP=SK(I,I)*DX*DX+2.0*DX*DY*SK(I,J)+SK(J,J)*DY*DY
      SK(I,J)=DX*DY*(SK(J,J)-SK(I,I))+SK(I,J)*(DX*DX-DY*DY)
      SK(I,I)=TEMP
      SK(J,I)=SK(I,J)
      SK(J,J)=1.0
      SK(I,K)=SK(I,K)*DX+SK(J,K)*DY
      SK(J,K)=-SK(K,I)*DY+SK(K,J)*DX
      SK(K,I)=SK(I,K)
      SK(K,J)=SK(J,K)
  400 CONTINUE
      CALL WRDISK(NTRACK,EE,NCOUNT)
      RETURN
 2000 FORMAT (33H3ZERO OR NEGATIVE AREA, ELEMENT =I4)
 2001 FORMAT (27H3SINGULAR MATRIX, ELEMENT =I4)
      END
```

```
$IBFTC NTES    DECK,LIST
      SUBROUTINE TRISTF
C
C*****     THIS SUBROUTINE FORMS THE TRIANGULAR ELEMENT STIFFNESS MATRIX
C*****     AND BODY FORCE VECTOR
C
      COMMON /QADTRI/
     1 R(6), Z(6), BF(9), XI(10), S(9,9), A(3,3), E(56)
      COMMON /FORIER/
     1 NFOUR, NF, NF2, NFT, NUMFOU, FORCOF(50,2), HED(12)
      DIMENSION
     1  SUU(3,3), SUW(3,3), SUT(3,3), SWW(3,3), SWT(3,3),
     2 STT(3,3), TUU(3,3), TUW(3,3), TUT(3,3), TWW(3,3), TWT(3,3),
     3 TTT(3,3), BTFR(3), BTFZ(3), BTFT(3)
      EQUIVALENCE
     1 (E(1),C11), (E(2),C12), (E(3),C13), (E(4),C22), (E(5),C23),
     2 (E(6),C33), (E(7),C44), (E(8),C55), (E(9),C66), (E(10),TLR),
     3 (E(11),TLZ), (E(12),TLT), (E(13),BFR), (E(14),BFC),
     4 (E(15),BFZ), (S(1,1),SUU), (S(1,7),SUW), (S(1,8),SUT),
     5 (S(1,2),SWW), (S(1,9),SWT), (S(1,3),STT), (S(7,4),BTFR),
     6 (S(7,5),BTFZ), (S(7,6),BTFT)
      REAL
     1 NF, NF2
C
C*****     FORM THE VOLUME INTEGRALS
C
      CALL INTEGR
C
C*****     FORM THE DISPLACEMENT TRANSFORMATION MATRIX
C
      IF (XI(2).LE.0.0) RETURN
      C=0.5/XI(2)
      A(1,1)=C*(R(2)*Z(3)-R(3)*Z(2))
      A(1,2)=C*(R(3)*Z(1)-R(1)*Z(3))
      A(1,3)=C*(R(1)*Z(2)-R(2)*Z(1))
      A(2,1)=C*(Z(2)-Z(3))
      A(2,2)=C*(Z(3)-Z(1))
      A(2,3)=C*(Z(1)-Z(2))
      A(3,1)=C*(R(3)-R(2))
      A(3,2)=C*(R(1)-R(3))
      A(3,3)=C*(R(2)-R(1))
C
C*****     FORM THE STIFFNESS MATRICIES IN THE GENERALIZED COORDINATES
C
      X=C33+NF2*C55
      SUU(1,1)=X*XI(3)
      SUU(1,2)=(C13+X)*XI(2)
      SUU(1,3)=X*XI(5)
      SUU(2,1)=SUU(1,2)
      SUU(2,2)=(C11+2.0*C13+X)*XI(1)
      SUU(2,3)=(C13+X)*XI(4)
      SUU(3,1)=SUU(1,3)
      SUU(3,2)=SUU(2,3)
      SUU(3,3)=X*XI(6)+C44*XI(1)
      Y=NF2*C66
      SWW(1,1)=Y*XI(3)
      SWW(1,2)=Y*XI(2)
```

```
      SWW(1,3)=Y*XI(5)
      SWW(2,1)=SWW(1,2)
      SWW(2,2)=(C44+Y)*XI(1)
      SWW(2,3)=Y*XI(4)
      SWW(3,1)=SWW(1,3)
      SWW(3,2)=SWW(2,3)
      SWW(3,3)=Y*XI(6)+C22*XI(1)
      X=C55+NF2*C33
      Y=C33*NF2
      STT(1,1)=X*XI(3)
      STT(1,2)=Y*XI(2)
      STT(1,3)=X*XI(5)
      STT(2,1)=STT(1,2)
      STT(2,2)=Y*XI(1)
      STT(2,3)=Y*XI(4)
      STT(3,1)=STT(1,3)
      STT(3,2)=STT(2,3)
      STT(3,3)=X*XI(6)+C66*XI(1)
      X=C33
      C=C55
      Y=X+C13
      SUT(2,2)=Y*XI(1)
      Y=Y+C
      C=X+C
      SUT(1,1)=C*XI(3)
      SUT(1,2)=X*XI(2)
      SUT(1,3)=C*XI(5)
      SUT(2,1)=Y*XI(2)
      SUT(2,3)=Y*XI(4)
      SUT(3,1)=C*XI(5)
      SUT(3,2)=X*XI(4)
      SUT(3,3)=C*XI(6)
      SUW(1,1)=0.0
      SUW(1,2)=0.0
      SUW(1,3)=C23*XI(2)
      SUW(2,1)=0.0
      SUW(2,2)=0.0
      SUW(2,3)=(C12+C23)*XI(1)
      SUW(3,1)=0.0
      SUW(3,2)=C44*XI(1)
      SUW(3,3)=C23*XI(4)
      X=C66
      Y=C23
      SWT(1,1)=0.0
      SWT(1,2)=0.0
      SWT(1,3)=-X*XI(2)
      SWT(2,1)=0.0
      SWT(2,2)=0.0
      SWT(2,3)=-X*XI(1)
      SWT(3,1)=Y*XI(2)
      SWT(3,2)=Y*XI(1)
      SWT(3,3)=(Y-X)*XI(4)
C
C*****    FORM BODY FORCE VECTOR
C
      BTFR(1)=TLT*XI(2)+BFR*XI(7)+BFC*XI(1)
      BTFR(2)=(TLR+TLT)*XI(1)+BFR*XI(9)+BFC*XI(7)
      BTFR(3)=TLT*XI(4)+BFR*XI(10)+BFC*XI(8)
```

```
      BTFZ(1)=BFZ*XI(1)
      BTFZ(2)=BFZ*XI(7)
      BTFZ(3)=TLZ*XI(1)+BFZ*XI(8)
      X=TLT*NF
      BTFT(1)=X*XI(2)+BFC*XI(1)
      BTFT(2)=X*XI(1)+BFC*XI(7)
      BTFT(3)=X*XI(4)+BFC*XI(8)
C
C*****    TRANSFORM THE STIFFNESS MATRIX AND BODY FORCE VECTOR TO
C*****    GLOBAL COORDINATES
C
      DO 130 I=1,3
      DO 130 J=1,3
      TUU(I,J)=0.0
      TUW(I,J)=0.0
      TUT(I,J)=0.0
      TWW(I,J)=0.0
      TWT(I,J)=0.0
      TTT(I,J)=0.0
      DO 130 K=1,3
      C=A(K,J)
      TUU(I,J)=TUU(I,J)+SUU(I,K)*C
      TUW(I,J)=TUW(I,J)+SUW(I,K)*C
      TUT(I,J)=TUT(I,J)+SUT(I,K)*C
      TWW(I,J)=TWW(I,J)+SWW(I,K)*C
      TWT(I,J)=TWT(I,J)+SWT(I,K)*C
  130 TTT(I,J)=TTT(I,J)+STT(I,K)*C
      DO 140 I=1,3
      IW=I+3
      IT=IW+3
      BF(I)=0.0
      BF(IW)=0.0
      BF(IT)=0.0
      DO 140 J=1,3
      S(I,J)=0.0
      JW=J+3
      JT=JW+3
      S(I,JW)=0.0
      S(I,JT)=0.0
      S(IW,JW)=0.0
      S(IW,JT)=0.0
      S(IT,JT)=0.0
      C=A(J,I)
      BF(I)=BF(I)+C*BTFR(J)
      BF(IW)=BF(IW)+C*BTFZ(J)
      BF(IT)=BF(IT)+C*BTFT(J)
      DO 140 K=1,3
      C=A(K,I)
      S(I,J)=S(I,J)+C*TUU(K,J)
      S(I,JW)=S(I,JW)+C*TUW(K,J)
      S(I,JT)=S(I,JT)+C*TUT(K,J)
      S(IW,JW)=S(IW,JW)+C*TWW(K,J)
      S(IW,JT)=S(IW,JT)+C*TWT(K,J)
  140 S(IT,JT)=S(IT,JT)+C*TTT(K,J)
      RETURN
      END
```

```
$IBFTC INTE    DECK,LIST
      SUBROUTINE INTEGR
C
C*****     THIS SUBROUTINE CALCULATES THE TRIANGULAR VOLUME INTEGRALS
C*****     FOR THE STIFFNESS MATRIX AND BODY FORCE VECTOR.
C
      COMMON /QADTRI/
     1 R(6), Z(6), BF(9), XI(10), S(9,9), A(3,3), E(56)
      DIMENSION
     1 XM(6)
      DO 100 I=3,10
  100 XI(I)=0.0
C
C*****     FORM THE AREA AND VOLUME INTEGRALS USING THE EXACT EXPRESSION
C
      XI(2)=0.5*(R(1)*(Z(2)-Z(3))+R(2)*(Z(3)-Z(1))+R(3)*(Z(1)-Z(2)))
      C=XI(2)/3.0
      XI(1)=C*(R(1)+R(2)+R(3))
      XI(4)=C*(Z(1)+Z(2)+Z(3))
      C=0.25*C
      R(4)=0.5*(R(1)+R(2))
      R(5)=0.5*(R(2)+R(3))
      R(6)=0.5*(R(3)+R(1))
      Z(4)=0.5*(Z(1)+Z(2))
      Z(5)=0.5*(Z(2)+Z(3))
      Z(6)=0.5*(Z(3)+Z(1))
      D=3.0*C
      DO 200 I=1,3
      J=I+3
      XM(I)=C*R(I)
  200 XM(J)=D*R(J)
      DO 300 I=1,6
      A=XM(I)
      B=R(I)
      C=Z(I)
      D=A*B
      XI(7)=XI(7)+D
      XI(8)=XI(8)+A*C
      XI(9)=XI(9)+D*B
      XI(10)=XI(10)+D*C
      IF (B.EQ.0.0) B=0.00001
      D=A/(B*B)
      XI(3)=XI(3)+D
      XI(5)=XI(5)+D*C
      XI(6)=XI(6)+D*C*C
  300 CONTINUE
      RETURN
      END
```

```
$IBFTC NTSC    DECK,LIST
      SUBROUTINE TRISTC
C
C*****     THIS SUBROUTINE FORMS THE TRIANGULAR ELEMENT STIFFNESS MATRIX
C*****     AND BODY FORCE VECTOR
C
      COMMON /QADTRI/
     1 R(6), Z(6), BF(9), XI(10), S(9,9), A(3,3), E(56)
      COMMON /FORIER/
     1 NFOUR, NF, NF2, NFT, NUMFOU, FORCOF(50,2), HED(12)
      DIMENSION
     1  SUU(3,3), SUW(3,3), SUT(3,3), SWW(3,3), SWT(3,3),
     2 STT(3,3), TUU(3,3), TUW(3,3), TUT(3,3), TWW(3,3), TWT(3,3),
     3 TTT(3,3), BTFR(3), BTFZ(3), BTFT(3)
      EQUIVALENCE
     1 (E(1),C11), (E(2),C12), (E(3),C13), (E(4),C22), (E(5),C23),
     2 (E(6),C33), (E(7),C44), (E(8),C55), (E(9),C66), (E(10),TLR),
     3 (E(11),TLZ), (E(12),TLT), (E(13),BFR), (E(14),BFC),
     4 (E(15),BFZ), (S(1,1),SUU), (S(1,7),SUW), (S(1,8),SUT),
     5 (S(1,2),SWW), (S(1,9),SWT), (S(1,3),STT), (S(7,4),BTFR),
     6 (S(7,5),BTFZ), (S(7,6),BTFT)
      REAL
     1 NF, NF2
C
C*****     FORM THE STIFFNESS MATRICIES IN THE GENERALIZED COORDINATES
C
      X=C33+NF2*C55
      SUU(1,1)=X*XI(3)
      SUU(1,2)=(C13+X)*XI(2)
      SUU(1,3)=X*XI(5)
      SUU(2,1)=SUU(1,2)
      SUU(2,2)=(C11+2.0*C13+X)*XI(1)
      SUU(2,3)=(C13+X)*XI(4)
      SUU(3,1)=SUU(1,3)
      SUU(3,2)=SUU(2,3)
      SUU(3,3)=X*XI(6)+C44*XI(1)
      Y=NF2*C66
      SWW(1,1)=Y*XI(3)
      SWW(1,2)=Y*XI(2)
      SWW(1,3)=Y*XI(5)
      SWW(2,1)=SWW(1,2)
      SWW(2,2)=(C44+Y)*XI(1)
      SWW(2,3)=Y*XI(4)
      SWW(3,1)=SWW(1,3)
      SWW(3,2)=SWW(2,3)
      SWW(3,3)=Y*XI(6)+C22*XI(1)
      X=C55+NF2*C33
      Y=C33*NF2
      STT(1,1)=X*XI(3)
      STT(1,2)=Y*XI(2)
      STT(1,3)=X*XI(5)
      STT(2,1)=STT(1,2)
      STT(2,2)=Y*XI(1)
      STT(2,3)=Y*XI(4)
      STT(3,1)=STT(1,3)
      STT(3,2)=STT(2,3)
      STT(3,3)=X*XI(6)+C66*XI(1)
```

```
C
C*****    FORM BODY FORCE VECTOR
C
      BTFR(1)=TLT*XI(2)
      BTFR(2)=(TLR+TLT)*XI(1)
      BTFR(3)=TLT*XI(4)
      BTFZ(1)=0.0
      BTFZ(2)=0.0
      BTFZ(3)=TLZ*XI(1)
      X=TLT*NF
      BTFT(1)=X*XI(2)
      BTFT(2)=X*XI(1)
      BTFT(3)=X*XI(4)
C
C*****    TRANSFORM THE STIFFNESS MATRIX AND BODY FORCE VECTOR TO
C*****    GLOBAL COORDINATES
C
      DO 130 I=1,3
      DO 130 J=1,3
      TUU(I,J)=0.0
      TWW(I,J)=0.0
      TTT(I,J)=0.0
      DO 130 K=1,3
      C=A(K,J)
      TUU(I,J)=TUU(I,J)+SUU(I,K)*C
      TWW(I,J)=TWW(I,J)+SWW(I,K)*C
  130 TTT(I,J)=TTT(I,J)+STT(I,K)*C
      DO 140 I=1,3
      IW=I+3
      IT=IW+3
      BF(I)=0.0
      BF(IW)=0.0
      BF(IT)=0.0
      DO 140 J=1,3
      JW=J+3
      JT=JW+3
      S(I,J)=0.0
      S(IW,JW)=0.0
      S(IT,JT)=0.0
      C=A(J,I)
      BF(I)=BF(I)+C*BTFR(J)
      BF(IW)=BF(IW)+C*BTFZ(J)
      BF(IT)=BF(IT)+C*BTFT(J)
      DO 140 K=1,3
      C=A(K,I)
      S(I,J)=S(I,J)+C*TUU(K,J)
      S(IW,JW)=S(IW,JW)+C*TWW(K,J)
  140 S(IT,JT)=S(IT,JT)+C*TTT(K,J)
      RETURN
      END
```

```
$IBFTC SHLL    DECK,LIST
      SUBROUTINE SHELST(NN,ISTOP)
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /FORIER/
     1 NFOUR, NF, NF2, XNF, NFT, NUMFOU, FORCOF(50,3), HED(12)
      COMMON /QUADST/
     1 SK(15,15), F(15)
      COMMON /NSHELN/
     1 IG(10), JG(10), KG(10), IH(12), JH(12), KH(12), H(12), XS(10),
     2 XW(10), Y(10), ST(8,10), B1(5,5), B2(5,5), A(8,11), XI(10,10),
     3 NCOUNT, NTRACK
      REAL
     1 NF, NF2
      DATA
     1 IG /1,2,3,3,4,4,5,5,5,5/, JG /1,1,2,3,2,3,2,3,4,5/,
     2 KG /2,4,1,2,3,4,5,6,7,8/, IH /1,1,2,3,3,3,4,4,5,5,5,5/,
     3 JH /1,6,2,2,3,4,7,8,7,8,9,10/, KH /7,8,4,6,7,8,3,4,5,6,7,8/,
     4 H /2.,6.,1.,1.,2.,3.,1.,1.,1.,1.,1.,1./, A /80*0./, Y(1) /1./,
     5 XS /-.97390653,-.86506337,-.67940957,-.43339539,-.14887434,
     6 .14887434,.43339539,.67940957,.86506337,.97390653/,
     7 XW /.06667134,.14945135,.21908636,.26926672,.29552422,.29552422,
     8 .26926672,.21908636,.14945135,.06667134/,
     9 NCOUNT /188/
C
C*****    INITIALIZATION
C
      ISTOP=0
      NTRACK=1319+NN
      DO 100 I=1,12
      F(I)=0.0
      DO 100 J=1,12
  100 SK(I,J)=0.0
C
C*****    CALL APPROPIATE SHELL STIFFNESS SUBROUTINE
C
      IF (NFOUR.GT.NFT) CALL NSHELC
      IF (NFOUR.EQ.NFT) CALL NSHELL(NN,ISTOP)
      IX(NN,5)=-IX(NN,5)
      RETURN
      END
```

```
$IBFTC NSHL    DECK,LIST
      SUBROUTINE NSHELL(NN,ISTOP)
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /FORIER/
     1 NFOUR, NF, NF2, XNF, NFT, NUMFOU, FORCOF(50,3), HED(12)
      COMMON /QUADST/
     1 SK(15,15), F(15)
      COMMON /NSHELN/
     1 IG(10), JG(10), KG(10), IH(12), JH(12), KH(12), H(12), XS(10),
     2 XW(10), Y(10), ST(8,10), B1(5,5), B2(5,5), A(8,11), XI(10,10),
     3 NCOUNT, NTRACK
      DIMENSION
     1 DISSTF(188)
      EQUIVALENCE
     1 (DISSTF(1),A(1,1)), (DISSTF(81),CS), (DISSTF(82),SS),
     2 (DISSTF(83),XT), (DISSTF(84),XL), (DISSTF(85),C11),
     3 (DISSTF(86),C12), (DISSTF(87),C22), (DISSTF(88),C44),
     4 (Y(2),RS), (Y(6),S)
      REAL
     1 NF, NF2
C
C*****    INITIALIZATION, GEOMETRY, AND MATERIAL PROPERTIES
C
      I=IX(NN,1)
      J=IX(NN,2)
      K=IX(NN,3)
      L=IX(NN,4)
      MTYPE=IX(NN,5)
      IF (R(I).EQ.ABS(R(L)).AND.Z(I).EQ.Z(L)) GO TO 10
      ISTOP=1
   10 IF (R(J).EQ.ABS(R(K)).AND.Z(J).EQ.Z(K)) GO TO 20
      ISTOP=1
   20 IF (R(I).LE.0.0.OR.R(J).LE.0.0) ISTOP=1
      R(K)=-ABS(R(K))
      R(L)=-ABS(R(L))
      XT=0.5*(T(K)+T(L))
      IF (XT.LE.0.0) ISTOP=1
      IF (ISTOP.EQ.0) GO TO 30
      WRITE (6,2000) NN
      RETURN
   30 X=1./E(6,MTYPE)
      C11=XT*(E(1,MTYPE)-E(3,MTYPE)*E(3,MTYPE)*X)
      C12=XT*(E(2,MTYPE)-E(3,MTYPE)*E(5,MTYPE)*X)
      C22=XT*(E(4,MTYPE)-E(5,MTYPE)*E(5,MTYPE)*X)
      T44=XT*E(7,MTYPE)
      DR=R(J)-R(I)
      DZ=Z(J)-Z(I)
      XL=SQRT(DR*DR+DZ*DZ)
      X=0.5*XL
      XB=0.5*DR
      RI=R(I)
      SS=-DZ/XL
      CS=DR/XL
C
```

```
C*****    COMPUTE THE VOLUME INTEGRALS
C
      DO 50 I=1,10
      DO 50 J=I,10
   50 XI(I,J)=0.
      DO 100 K=1,10
      XX=1.+XS(K)
      S=X*XX
      RS=RI+XB*XX
      RS=1./RS
      Y(3)=S*RS
      Y(4)=S*Y(3)
      Y(5)=S*Y(4)
      Y(7)=RS*RS
      Y(8)=S*Y(7)
      Y(9)=S*Y(8)
      Y(10)=S*Y(9)
      XX=X/RS
      DO 100 I=1,10
      DO 100 J=I,10
  100 XI(I,J)=XI(I,J)+XX*XW(K)*Y(I)*Y(J)
      DO 150 I=2,10
      DO 150 J=1,I
  150 XI(I,J)=XI(J,I)
C
C*****    FORM THE STIFFNESS MATRIX IN GENERALIZED COORDINATES
C
      C44=XNF*T44
      B1(1,1)=C11
      B1(1,2)=0.
      B1(1,3)=CS*C12
      B1(1,4)=NF*C12
      B1(1,5)=SS*C12
      B1(2,2)=C44
      B1(2,3)=-NF*C44
      B1(2,4)=-CS*C44
      B1(2,5)=0.
      B1(3,3)=CS*CS*C22+NF2*C44
      B1(3,4)=NF*CS*(C22+C44)
      B1(3,5)=SS*CS*C22
      B1(4,4)=NF2*C22+CS*CS*C44
      B1(4,5)=NF*SS*C22
      B1(5,5)=SS*SS*C22
      X=XT*XT/12.
      C11=X*C11
      C12=X*C12
      C22=X*C22
      C44=X*C44
      B2(1,1)=C11
      B2(1,2)=0.
      B2(1,3)=CS*C12
      B2(1,4)=-NF*SS*C12
      B2(1,5)=-NF2*C12
      B2(2,2)=SS*SS*C44
      B2(2,3)=NF*SS*C44
      B2(2,4)=-SS*SS*CS*C44
      B2(2,5)=-NF*SS*CS*C44
      B2(3,3)=CS*CS*C22+NF2*C44
```

```
      B2(3,4)=-NF*SS*CS*(C22+C44)
      B2(3,5)=-NF2*CS*(C22+C44)
      B2(4,4)=SS*SS*(NF2*C22+CS*CS*C44)
      B2(4,5)=NF*SS*(NF2*C22+CS*CS*C44)
      B2(5,5)=NF2*(NF2*C22+CS*CS*C44)
      DO 200 I=2,5
      DO 200 J=1,I
      B1(I,J)=B1(J,I)
  200 B2(I,J)=B2(J,I)
      DO 300 NG=1,10
      I=KG(NG)
      K=JG(NG)
      M=IG(NG)
      DO 300 MG=1,10
      J=KG(MG)
      L=JG(MG)
      N=IG(MG)
  300 SK(I,J)=SK(I,J)+B1(M,N)*XI(K,L)
      DO 400 NG=1,12
      I=KH(NG)
      K=JH(NG)
      M=IH(NG)
      DO 400 MG=1,12
      J=KH(MG)
      L=JH(MG)
      N=IH(MG)
  400 SK(I,J)=SK(I,J)+H(NG)*B2(M,N)*H(MG)*XI(K,L)
C
C*****    FORM DISPLACEMENT TRANSFORMATION MATRIX
C
      C44=X*T44
      XL=1./XL
      DR=DR*XL
      DZ=DZ*XL
      X=XL*XL
      A(1,1)=DR
      A(1,2)=DZ
      A(2,1)=-DR*XL
      A(2,2)=-DZ*XL
      A(2,4)=-A(2,1)
      A(2,5)=-A(2,2)
      A(3,3)=1.
      A(4,3)=-XL
      A(4,6)=XL
      A(5,1)=-DZ
      A(5,2)=DR
      A(6,10)=1.
      A(7,1)=3.*DZ*X
      A(7,2)=-3.*DR*X
      A(7,4)=-A(7,1)
      A(7,5)=-A(7,2)
      A(7,7)=-XL
      A(7,10)=2.*A(7,7)
      A(8,1)=-2.*DZ*X*XL
      A(8,2)=2.*DR*X*XL
      A(8,4)=-A(8,1)
      A(8,5)=-A(8,2)
      A(8,7)=X
```

```
      A(8,10)=X
      DO 500 I=1,8
      DO 500 J=1,10
      ST(I,J)=0.
      DO 500 K=1,8
  500 ST(I,J)=ST(I,J)+SK(I,K)*A(K,J)
      DO 600 I=1,10
      DO 600 J=1,10
      SK(I,J)=0.
      DO 600 K=1,8
  600 SK(I,J)=SK(I,J)+A(K,I)*ST(K,J)
C
C*****    STORE INFORMATION FOR STRESS CALCULATION
C
      XL=1./XL
      CALL WRDISK(NTRACK,DISSTF,NCOUNT)
      RETURN
 2000 FORMAT (23H3BAD SHELL ELEMENT, NO. I4)
      END
```

```
$IBFTC NSLC    DECK,LIST
      SUBROUTINE NSHELC
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /FORIER/
     1 NFOUR, NF, NF2, XNF, NFT, NUMFOU, FORCOF(50,3), HED(12)
      COMMON /QUADST/
     1 SK(15,15), F(15)
      COMMON /NSHELN/
     1 IG(10), JG(10), KG(10), IH(12), JH(12), KH(12), H(12), XS(10),
     2 XH(10), Y(10), ST(8,10), B1(5,5), B2(5,5), A(8,11), XI(10,10),
     3 NCOUNT, NTRACK
      DIMENSION
     1 DISSTF(188)
      EQUIVALENCE
     1 (DISSTF(1),A(1,1)), (DISSTF(81),CS), (DISSTF(82),SS),
     2 (DISSTF(83),XT), (DISSTF(84),XL), (DISSTF(85),C11),
     3 (DISSTF(86),C12), (DISSTF(87),C22), (DISSTF(88),C44),
     4 (Y(2),RS), (Y(6),S)
      REAL
     1 NF, NF2
C
C*****    INITIALIZATION
C
      CALL RDDISK(NTRACK,DISSTF,NCOUNT)
C
C*****    FORM NEW STIFFNESS MATRIX
C
      B2(1,1)=C11
      B2(1,2)=0.
      B2(1,3)=CS*C12
      B2(1,4)=-NF*SS*C12
      B2(1,5)=-NF2*C12
      B2(2,2)=SS*SS*C44
      B2(2,3)=NF*SS*C44
      B2(2,4)=-SS*SS*CS*C44
      B2(2,5)=-NF*SS*CS*C44
      B2(3,3)=CS*CS*C22+NF2*C44
      B2(3,4)=-NF*SS*CS*(C22+C44)
      B2(3,5)=-NF2*CS*(C22+C44)
      B2(4,4)=SS*SS*(NF2*C22+CS*CS*C44)
      B2(4,5)=NF*SS*(NF2*C22+CS*CS*C44)
      B2(5,5)=NF2*(NF2*C22+CS*CS*C44)
      X=12./(XT*XT)
      C11=X*C11
      C12=X*C12
      C22=X*C22
      C44=X*C44
      B1(1,1)=C11
      B1(1,2)=0.
      B1(1,3)=CS*C12
      B1(1,4)=NF*C12
      B1(1,5)=SS*C12
      B1(2,2)=C44
      B1(2,3)=-NF*C44
```

```
      B1(2,4)=-CS*C44
      B1(2,5)=0.
      B1(3,3)=CS*CS*C22+NF2*C44
      B1(3,4)=NF*CS*(C22+C44)
      B1(3,5)=SS*CS*C22
      B1(4,4)=NF2*C22+CS*CS*C44
      B1(4,5)=NF*SS*C22
      B1(5,5)=SS*SS*C22
      DO 200 I=2,5
      DO 200 J=1,I
      B1(I,J)=B1(J,I)
  200 B2(I,J)=B2(J,I)
      DO 300 NG=1,10
      I=KG(NG)
      K=JG(NG)
      M=IG(NG)
      DO 300 MG=1,10
      J=KG(MG)
      L=JG(MG)
      N=IG(MG)
  300 SK(I,J)=SK(I,J)+B1(M,N)*XI(K,L)
      DO 400 NG=1,12
      I=KH(NG)
      K=JH(NG)
      M=IH(NG)
      DO 400 MG=1,12
      J=KH(MG)
      L=JH(MG)
      N=IH(MG)
  400 SK(I,J)=SK(I,J)+H(NG)*B2(M,N)*H(MG)*XI(K,L)
      DO 500 I=1,8
      DO 500 J=1,10
      ST(I,J)=0.
      DO 500 K=1,8
  500 ST(I,J)=ST(I,J)+SK(I,K)*A(K,J)
      DO 600 I=1,10
      DO 600 J=1,10
      SK(I,J)=0.
      DO 600 K=1,8
  600 SK(I,J)=SK(I,J)+A(K,I)*ST(K,J)
      RETURN
      END
```

```
$IBFTC BCSB    DECK,LIST,REF
      SUBROUTINE BACSUB
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /BANARG/
     1 A(72,144), B(144), NNP, ND, ND2, NCOUNT, MBAND, NUMBLK, NCONTD,
     2 NCONTS
      NU=ND*NUMBLK+1
      NB=NUMBLK
  400 DO 450 M=1,ND
      N=ND+1-M
      DO 425 K=2,MBAND
      L=N+K-1
  425 B(N)=B(N)-A(K,N)*B(L)
      NM=N+ND
      B(NM)=B(N)
      NU=NU-1
  450 UR(NU)=B(N)
      NB=NB-1
      IF (NB.LE.0) RETURN
      NTRACK=40*(NB-1)
      CALL RDDISK(NTRACK,A,NCOUNT)
      NTRACK=NTRACK+39
      CALL RDDISK (NTRACK,B,ND)
      GO TO 400
      END
```

```
$IBFTC NSTR    DECK,LIST
      SUBROUTINE STRESS
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /BANARG/
     1 O(72,144), W(144), NNP, ND, ND2, NCOUNT, MBAND, NUMBLK, NCONTD,
     2 NCONTS
      COMMON /FORIER/
     1 NFOUR, NF, NF2, XNF, NFT, NUMFOU, FORCOF(50,3), HED(12)
      DIMENSION
     1 D(12,3), U(12), EE(56), SIG(6,700), ORD(2,700)
      EQUIVALENCE
     1 (EE(18),D(1,1)), (EE(54),UK), (EE(55),WK), (EE(56),TK),
     2 (EE(16),RK), (EE(17),ZK), (SIG(1,1),O(1,1)), (ORD(1,1),O(1,100))
      DATA
     1 NCONT /56/, PI /3.1415927/
      REAL
     1 NF, NF2
C
C*****    PRINT OF NODAL POINT VARIABLES
C
      NFOUR=NFOUR-1
      MPRINT=0
      DO 50 N=1,NUMNP
      IF (R(N).LT.0.0) GO TO 50
      C=-CODE(N)
      IF (C.LE.0.0) GO TO 30
      DX=COS(C)
      DY=SIN(C)
      A=UR(3*N-2)*DX-UR(3*N-1)*DY
      UR(3*N-1)=UR(3*N-2)*DY+UR(3*N-1)*DX
      UR(3*N-2)=A
   30 M=3*N
      IF (MPRINT.GT.0) GO TO 40
      MPRINT=50
      WRITE (6,2000) HED,NFOUR
   40 WRITE (6,2001) N, R(N), Z(N), UR(M-2), UR(M-1), UR(M)
   50 MPRINT =MPRINT-1
      MPRINT=0
C
C*****    CALCULATION OF CENTER NODE VARIABLES
C
      DO 300 N=1,NUMEL
      MTYPE=IABS(IX(N,5))
      IX(N,5)=MTYPE
      IF (MTYPE.GE.NCUT) GO TO 300
      NTRACK=1319+N
      CALL RDDISK(NTRACK,EE,NCONT)
      IF (NFOUR+1.LT.NUMFOU) GO TO 60
      ORD(1,N)=RK
      ORD(2,N)=ZK
   60 I=IX(N,1)
      J=IX(N,2)
      K=IX(N,3)
      L=IX(N,4)
```

```
      C=0.25
      IF (K.EQ.L) C=1./3.
      II=3*I
      JJ=3*J
      KK=3*K
      LL=3*L
      U(1)=UR(II-2)
      U(2)=UR(II-1)
      U(3)=UR(II)
      U(4)=UR(JJ-2)
      U(5)=UR(JJ-1)
      U(6)=UR(JJ)
      U(7)=UR(KK-2)
      U(8)=UR(KK-1)
      U(9)=UR(KK)
      U(10)=UR(LL-2)
      U(11)=UR(LL-1)
      U(12)=UR(LL)
      DO 70 JJ=1,12
      UK=UK-D(JJ,1)*U(JJ)
      WK=WK-D(JJ,2)*U(JJ)
   70 TK=TK-D(JJ,3)*U(JJ)
C
C*****    CALCULATION OF STRESSES AT CENTER NP
C
      ERR=0.0
      EZZ=0.0
      ERZ=0.0
      ERT=0.0
      EZT=0.0
C
C*****    COMPUTE STRESS AT NODE K OF TRIANGLE
C
      DO 200 NN=1,4
      GO TO (130,140,150,160), NN
  130 II=I
      JJ=J
      UI=U(1)
      WI=U(2)
      TI=U(3)
      UJ=U(4)
      WJ=U(5)
      TJ=U(6)
      GO TO 170
  140 II=J
      JJ=K
      UI=U(4)
      WI=U(5)
      TI=U(6)
      UJ=U(7)
      WJ=U(8)
      TJ=U(9)
      GO TO 170
  150 IF (K.EQ.L) GO TO 200
      II=K
      JJ=L
      UI=U(7)
      WI=U(8)
```

```
      TI=U(9)
      UJ=U(10)
      WJ=U(11)
      TJ=U(12)
      GO TO 170
  160 II=L
      JJ=I
      UI=U(10)
      WI=U(11)
      TI=U(12)
      UJ=U(1)
      WJ=U(2)
      TJ=U(3)
  170 RI=R(II)
      RJ=R(JJ)
      ZI=Z(II)
      ZJ=Z(JJ)
      AJ=RJ-RI
      AK=RK-RI
      BJ=ZJ-ZI
      BK=ZK-ZI
      A=AJ-AK
      B=BJ-BK
      D=AJ*BK-AK*BJ
      TRR=B*UI+BK*UJ-BJ*UK
      TZZ=-A*WI-AK*WJ+AJ*WK
      TRZ=-A*UI+B*WI-AK*UJ+BK*WJ+AJ*UK-BJ*WK
      TZT=-A*TI-AK*TJ+AJ*TK
      TRT=-((RJ*ZK-RK*ZJ)*TI+(RK*ZI-RI*ZK)*TJ+(RI*ZJ-RJ*ZI)*TK+ZK*TZT)/
     1 RK
      ERR=TRR/D+ERR
      EZZ=TZZ/D+EZZ
      ERZ=TRZ/D+ERZ
      ERT=TRT/D+ERT
      EZT=TZT/D+EZT
  200 CONTINUE
      ERR=C*ERR
      EZZ=C*EZZ
      ETT=(UK+NF*TK)/RK
      ERZ=C*ERZ
      ERT=C*ERT-NF*UK/RK
      EZT=C*EZT-NF*WK/RK
C
C*****    CALCULATE STRESSES
C
      SIGRR=EE(1)*ERR+EE(2)*EZZ+EE(3)*ETT-EE(10)
      SIGZZ=EE(2)*ERR+EE(4)*EZZ+EE(5)*ETT-EE(11)
      SIGTT=EE(3)*ERR+EE(5)*EZZ+EE(6)*ETT-EE(12)
      SIGRZ=EE(7)*ERZ
      SIGRT=EE(8)*ERT
      SIGZT=EE(9)*EZT
C
C*****    CALCULATE PRINCIPAL STRESSES AND STRAINS
C
      CC=(SIGRR+SIGZZ)*0.5
      DD=(ERR+EZZ)*0.5
      CR=SQRT((0.5*(SIGZZ-SIGRR))**2+SIGRZ**2)
      DR=0.5*SQRT((EZZ-ERR)**2+ERZ**2)
```

```
      SIG1=CC+CR
      EPS1=DD+DR
      SIG2=CC-CR
      EPS2=DD-DR
      ANG=0.0
      IF (SIGRZ.EQ.0.0.AND.SIGRR.EQ.SIGZZ) GO TO 250
      ANG=28.648*ATAN2(2.*SIGRZ,(SIGRR-SIGZZ))
C
C*****    CALCULATE STRESSES PARALLEL TO LINE J-K
C
  250 J=IX(N,2)
      K=IX(N,3)
      ANGLE=2.0*ATAN2(Z(K)-Z(J),R(K)-R(J))-PI
      COS2A=COS(ANGLE)
      SIN2A=SIN(ANGLE)
      CX=0.5*(SIGRR-SIGZZ)
      SIGJK=CX*COS2A+SIGRZ*SIN2A+CC
      TAUJK=-CX*SIN2A+SIGRZ*COS2A
C
C*****    STORE STRESSES
C
      SIG(1,N)=SIGRR
      SIG(2,N)=SIGZZ
      SIG(3,N)=SIGTT
      SIG(4,N)=SIGRZ
      SIG(5,N)=SIGRT
      SIG(6,N)=SIGZT
C
C*****    PRINT STRESSES FOR EACH FOURIER COEFFICIENT
C
      IF (MPRINT.GT.0) GO TO 290
      WRITE (6,2002) HED,NFOUR
      MPRINT=25
  290 WRITE (6,2003)
     1 N,RK,(SIG(I,N),I=1,6),SIG1,SIG2,ANG,SIGJK,TAUJK,ZK,ERR,EZZ,ETT ,
     2 ERZ,ERT,EZT,EPS1,EPS2
  300 MPRINT=MPRINT-1
C
C*****    STORE STRESSES AND DISPLACEMENTS ON DISK
C
      IF (NANGLE.LT.0) GO TO 350
      NTRACK=2040+40*NFOUR
      CALL WRDISK(NTRACK,SIG,NCONTS)
      NTRACK=NTRACK+25
      CALL WRDISK(NTRACK,UR,NCONTD)
  350 NFOUR=NFOUR+1
      RETURN
C
C*****    FORMATS
C
 2000 FORMAT (1H1,12A6, 39H DISPLACEMENTS FOR FOURIER TERM, NUMBERI3/
     1 82H    NP    R-ORDINATE    Z-ORDINATE    R-DISPLACEMENT    Z-DISPLACEM
     2ENT    T-DISPLACEMENT//)
 2001 FORMAT (I5,2F13.2,3E17.5)
 2002 FORMAT (1H1,12A6, 34H STRESSES FOR FOURIER TERM, NUMBERI3//132H   E
     1LM    R-ORD    R-STRESS    Z-STRESS    T-STRESS   RZ-STRESS   RT-STRESS
     2 ZT-STRESS MAX-STRESS MIN-STRESS ANGLE IN  JK-STRESS   JK-SHEAR/
     38X,102HZ-ORD    R-STRAIN    Z-STRAIN    T-STRAIN  RZ-STRAIN   RT-STRAI
```

```
     4N  ZT-STRAIN MAX-STRAIN MIN-STRAIN RZ-PLANE//)
2003 FORMAT (I5,0P1F8.2,1P8E11.2,0P1F9.2,1P2E11.2/5X,0P1F8.2,1P8E11.2)
     END
```

```
$IBFTC NSLS    DECK,LIST
      SUBROUTINE SHLSTR
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /FORIER/
     1 NFOUR, NF, NF2, XNF, NFT, NUMFOU, FORCOF(50,3), HED(12)
      DIMENSION
     1 A(8,11), AA(8), U(10), DISSTF(88), SIG(16)
      EQUIVALENCE
     1 (DISSTF(1),A(1,1)), (DISSTF(81),CS), (DISSTF(82),SS),
     2 (DISSTF(83),XT), (DISSTF(84),XL), (DISSTF(85),C11),
     3 (DISSTF(86),C12), (DISSTF(87),C22), (DISSTF(88),C44),
     4 (UI,AA(1)), (ESS,AA(2)), (VI,AA(3)), (ETT,AA(4)), (WI,AA(5)),
     5 (TI,AA(6)), (SIG(1),RSI), (SIG(9),RSJ)
      DATA
     1 NCOUNT /88/, U(8) /0.0/, U(9) /0.0/
      REAL
     1 NF, NF2
C
C*****     INITIALIZATION
C
      NFOUR=NFOUR-1
      MPRINT=0
      DO 500 N=1,NUMEL
      MTYPE=IX(N,5)
      IF (MTYPE.LT.NCUT) GO TO 500
      NTRACK=1319+N
      CALL RDDISK(NTRACK,DISSTF,NCOUNT)
      I=IX(N,1)
      J=IX(N,2)
      K=IX(N,3)
      L=IX(N,4)
      RSI=1./R(I)
      RSJ=1./R(J)
      SIG(2)=Z(I)
      SIG(10)=Z(J)
      I=3*I
      J=3*J
      K=3*K-2
      L=3*L-2
      U(3)=UR(I)
      U(6)=UR(J)
      I=I-1
      J=J-1
      U(2)=UR(I)
      U(5)=UR(J)
      I=I-1
      J=J-1
      U(1)=UR(I)
      U(4)=UR(J)
      U(7)=UR(K)
      U(10)=UR(L)
      DO 100 K=1,8
      AA(K)=0.0
      DO 100 L=1,10
```

```
  100 AA(K)=AA(K)+A(K,L)*U(L)
C
C*****     COMPUTE STRAINS
C
      UJ=UI+ESS*XL
      VJ=VI+ETT*XL
      WJ=WI+XL*(TI+XL*(AA(7)+XL*AA(8)))
      TJ=TI+XL*(2.0*AA(7)+3.0*XL*AA(8))

      ETTI=RSI*(NF*VI+SS*WI+CS*UI)
      ETTJ=RSJ*(NF*VJ+SS*WJ+CS*UJ)
      GAMI=ETT-RSI*(CS*VI+NF*UI)
      GAMJ=ETT-RSJ*(CS*VJ+NF*UJ)
      XKSSI=-2.0*AA(7)
      XKSSJ=XKSSI-6.0*AA(8)*XL
      XKTTI=RSI*(RSI*(NF2*WI+SS*NF*VI)-CS*TI)
      XKTTJ=RSJ*(RSJ*(NF2*WJ+SS*NF*VJ)-CS*TJ)
      XKSTI=RSI*(NF*TI+RSI*NF*CS*WI+SS*(ETT-RSI*CS*VI))
      XKSTJ=RSJ*(NF*TJ+RSJ*NF*CS*WJ+SS*(ETT-RSJ*CS*VJ))
C
C*****     COMPUTE STRESSES
C
      SIG(3)=C11*XKSSI+C12*XKTTI
      SIG(11)=C11*XKSSJ+C12*XKTTJ
      SIG(4)=C12*XKSSI+C22*XKTTI
      SIG(12)=C12*XKSSJ+C22*XKTTJ
      SIG(5)=C44*XKSTI
      SIG(13)=C44*XKSTJ
      X=12.0/(XT*XT)
      C11=X*C11
      C12=X*C12
      C22=X*C22
      C44=X*C44
      SIG(6)=C11*ESS+C12*ETTI
      SIG(14)=C11*ESS+C12*ETTJ
      SIG(7)=C12*ESS+C22*ETTI
      SIG(15)=C12*ESS+C22*ETTJ
      SIG(8)=C44*GAMI
      SIG(16)=C44*GAMJ
      RSI=1./RSI
      RSJ=1./RSJ
C
C*****     PRINT SHELL STRESSES FOR EACH FOURIER COEFFICIENT
C
      IF (MPRINT.GT.0) GO TO 400
      WRITE (6,2000) HED,NFOUR
      MPRINT=25
  400 MPRINT=MPRINT-1
      WRITE (6,2001) N,(SIG(K),K=1,16)
  500 CONTINUE
      NFOUR=NFOUR+1
      RETURN
 2000 FORMAT (1H1,12A6,40H SHELL STRESSES FOR FOURIER TERM, NUMBERI3//
     1 111H SHELL R-ORD(I) Z-ORD(I)  LONG MOM(I)  CIRC MOM(I) CROSS MOM(
     2I)  LONG STRESS(I)  CIRC STRESS(I) SHEAR STRESS(I)/
     3 111H   ELM R-ORD(J) Z-ORD(J)  LONG MOM(J)  CIRC MOM(J) CROSS MOM(
     4J)  LONG STRESS(J)  CIRC STRESS(J) SHEAR STRESS(J)//)
 2001 FORMAT (1I6,2F9.2,3E13.3,3E16.3/6X,2F9.2,3E13.3,3E16.3)
      END
```

```
$IBFTC TSTR    DECK,LIST,REF
      SUBROUTINE TSTRES
      COMMON /NPELD/
     1 UR(800), UZ(800), UT(800), CODE(800), T(800), R(800), Z(800),
     2 IX(700,5), E(12,25), RO(25), NUMNP, NUMEL, TO, ANGFQ, ACELR,
     3 ACELZ, NCUT, NANGLE, XANG(4), NTAPE
      COMMON /BANARG/
     1 A(72,144), B(144), NNP, ND, ND2, NCOUNT, MBAND, NUMBLK, NCONTD,
     2 NCONTS
      COMMON /FORIER/
     1 NFOUR, NF, NF2, XNF, NFT, NUMFOU, FORCOF(50,3), HED(12)
      DIMENSION
     1 U(2400), SIG(6,700), TEMP(6,700), ORD(2,700)
      EQUIVALENCE
     1 (U,SIG,A), (TEMP,UR), (ORD,A(1,100))
      REAL
     1 NF, NF2
C
C*****    SUM THE DISPLACEMENTS
C
      DO 200 NA=1,NANGLE
      DO 50 N=1,NCONTD
   50 U(N)=0.0
      DO 100 NFOUR=1,NUMFOU
      IF (FORCOF(NFOUR,1).EQ.0.0) GO TO 100
      NTRACK=2065+40*(NFOUR-1)
      CALL RDDISK(NTRACK,UR,NCONTD)
      NF=NFOUR-1
      CS=COS(NF*XANG(NA))
      SS=SIN(NF*XANG(NA))
      DO 90 N=1,NUMNP
      M=3*N
      U(M)=U(M)+SS*UR(M)
      M=M-1
      U(M)=U(M)+CS*UR(M)
      M=M-1
   90 U(M)=U(M)+CS*UR(M)
  100 CONTINUE
C
C*****    PRINT THE DISPLACEMENTS
C
      MPRINT=0
      DO 170 N=1,NUMNP
      IF (R(N).LT.0.0) GO TO 160
      M=3*N
      IF (MPRINT.GT.0) GO TO 150
      MPRINT=50
      WRITE(6,2000) HED,XANG(NA)
  150 WRITE (6,2001) N,R(N),Z(N),U(M-2),U(M-1),U(M)
  160 MPRINT=MPRINT-1
  170 CONTINUE
  200 CONTINUE
C
C*****    SUM THE STRESSES
C
      DO 500 NA=1,NANGLE
      DO 220 N=1,NUMEL
      DO 220 M=1,6
  220 SIG(M,N)=0.0
```

```
      DO 300 NFOUR=1,NUMFOU
      IF (FORCOF(NFOUR,1).EQ.0.0) GO TO 300
      NTRACK=2040+40*(NFOUR-1)
      CALL RDDISK(NTRACK,TEMP,NCONTS)
      NF=NFOUR-1
      CS=COS(NF*XANG(NA))
      SS=SIN(NF*XANG(NA))
      DO 270 N=1,NUMEL
      IF (IX(N,5).GE.NCUT) GO TO 270
      DO 250 M=1,4
  250 SIG(M,N)=SIG(M,N)+CS*TEMP(M,N)
      DO 260 M=5,6
  260 SIG(M,N)=SIG(M,N)+SS*TEMP(M,N)
  270 CONTINUE
  300 CONTINUE
C
C*****    PRINT THE STRESSES
C
      MPRINT=0
      DO 400 N=1,NUMEL
      IF (IX(N,5).GE.NCUT) GO TO 400
      IF (MPRINT.GT.0) GO TO 350
      MPRINT=50
      WRITE (6,2002) HED,XANG(NA)
  350 WRITE (6,2003) N,ORD(1,N),ORD(2,N),(SIG(M,N),M=1,6)
  400 MPRINT=MPRINT-1
  500 CONTINUE
      RETURN
 2000 FORMAT (1H1,12A6,31H TOTAL FOURIER DISPLACEMENTS AT F6.3,8H RADIAN
     1S//82H   NP   R-ORDINATE   Z-ORDINATE   R-DISPLACEMENT   Z-DISPLAC
     2EMENT   T-DISPLACEMENT//)
 2001 FORMAT (1I5,2F13.2,3E17.5)
 2002 FORMAT (1H1,12A6,26H TOTAL FOURIER STRESSES AT F6.3,8H RADIANS//93
     1H  ELM   R-ORD   Z-ORD   RR-STRESS   ZZ-STRESS   TT-STRESS   RZ-ST
     2RESS   RT-STRESS   ZT-STRESS//)
 2003 FORMAT (1I5,0P2F8.2,1P6E12.3)
      END
```