AD-A086 113

AD

CONTRACT REPORT ARBRL-CR-00423

FINITE ELEMENT METHOD OF STRESS ANALYSIS OF NONAXISYMMETRIC CONFIGURATIONS

Prepared by

University of Illinois
Aeronautical & Astronautical
Engineering Dept.
Urbana, IL 61801

April 1980

US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND
BALLISTIC RESEARCH LABORATORY
ABERDEEN PROVING GROUND, MARYLAND

Approved for public release; distribution unlimited.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: CONTRACT REPORT ARBRL-CR-00423	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER: Final Report 3/76 - 12/78	
4. TITLE (and Subtitle): FINITE ELEMENT METHOD OF STRESS ANALYSIS OF NONAXISYMMETRIC CONFIGURATIONS	5. TYPE OF REPORT & PERIOD COVERED: Contract
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): ADAM R. ZAK, JAMES N. CRADDOCK	8. CONTRACT OR GRANT NUMBER(s): Contract DAAD05-76-C-0743
9. PERFORMING ORGANIZATION NAME AND ADDRESS: University of Illinois, Aeronautical & Astronautical Engineering Dept., Urbana, IL. 61801	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: US Army Armament Research & Development Command, US Army Ballistic Research Laboratory (DRDAR-BL), Aberdeen Proving Ground, MD 21005	12. REPORT DATE: April 1980
	13. NUMBER OF PAGES: 140
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Stress Analysis
Finite Element Method
Elasticity
Plasticity
Nonaxisymmetric Configuration
Structural Analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A finite element method of analysis is developed for structural configurations which are derived from axisymmetric geometries but contain definite non-axisymmetric features in the circumferential direction. The purpose of the present analysis is to develop a method which will take into consideration the fact that the stress and strain conditions in these geometries will be related to the corresponding axisymmetric solution. The analysis is developed in terms of a cylindrical coordinate system r, θ and z. As the first step of the

(continued on reverse side)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) analysis, the geometry is divided into several segments in the r-θ plane. Each segment is then divided into a set of quadrilateral elements in the r-z plane. The axisymmetric displacements are obtained for each segment by solving a related axisymmetric configuration. A perturbation analysis is then performed to match the solutions at certain points between the segments, and obtain the perturbation displacements for the total structure. The total displacement is then the axisymmetric displacement plus the perturbation displacement. The stresses and strains are then calculated at any desired point once the total displacements are known. The method is applied to a number of examples to illustrate the accuracy of the method. The results for these examples are presented and discussed. The method of analysis is developed in two versions. The first version is for elastic, orthotropic materials. The second method is for elastic-plastic materials with a kinematic strain hardening model. As in the case of the elastic version, the second version also allows for orthotropic properties and the yield criterion is based on the Hill's yield function which reduces in the limit to von Mises-Hencky for isotropic materials.

TABLE OF CONTENTS

I. INTRODUCTION

There are many problems in structural analysis which have an axisymmetric configuration. An axisymmetric problem has the advantage that a two-dimensional solution is possible. This fact greatly simplifies the analysis of such configurations. There are other problems which have an axisymmetric appearance, but have nonaxisymmetric features which render a purely axisymmetric solution inapplicable. One example of this type of problem occurs in interior ballistics where longitudinal slits or slots are required for sabots. An example of this type of configuration is given in Figure 1.

One method of analyzing these nonaxisymmetric problems would be to use a full three-dimensional finite element approach.[1,2]* However, if some advantage can be taken of the fact that these geometries will have stress and displacement fields somewhat similar to the axisymmetric cases, some efficiency could be gained over the complete three-dimensional method of analysis. Consequently, a method that makes use of the two-dimensional axisymmetric approach, but also yields first order nonaxisymmetric effects, is an appealing alternative.

The objective of this investigation is to develop a method of analysis that will take advantage of the axisymmetric similarity but still give accurate results for stress and displacement fields. To perform this analysis the structure is divided into several segments in the r-θ plane.[3] An axisymmetric solution is obtained for each segment. The segments are joined at a certain number of points. At these points the displacements are forced to match. Since the axisymmetric displacements for adjacent segments won't match, a set of perturbations displacements for each segment is obtained. Then the axisymmetric and perturbation displacements are combined to give the total displacements. Once the displacements are known, it is a simple matter to obtain the stresses and strains for the body.

In addition to elastic loadings, structural components are often loaded only once, but well into the plastic region. Therefore the elastic analysis is expanded to include elastic-plastic strain hardening materials. For this part of the analysis the Hill yield criterion[4] for orthotropic materials is used. If the material is isotropic this reduces to the von Mises-Hencky yield condition. To model the strain hardening behavior the Prager, or kinematic, hardening rule is used.[5]

---

*Superscripts refer to references.

After the method is developed a number of numerical examples are presented. The method is compared to known solutions. Also results are compared to the substructure analysis which resembles the perturbation method presented in this paper.

II. ELASTIC ANALYSIS

The purpose of the present investigation is to develop an approximate method of analysis of elastic configurations which are almost axisymmetric but have geometrical changes in the circumferential direction. The method of analysis which is proposed is based on the finite element approach.

The type of geometries that this method is designed to analyze is represented by Figure 1. If the geometry repeats at regular intervals only one portion of the total structure need be analyzed. Consider one such portion as shown in Figure 2. This configuration is divided into a number of segments in the r-θ plane. Then each segment is divided into quadrilateral elements as shown in Figure 3. On each face of this segment the finite element nodes are assumed to correspond to the nodes in the adjacent segments. Then an axisymmetric analysis is performed on each segment. The segments are joined at four connecting nodes and displacements at these nodes are forced to match.

Axisymmetric Analysis

The first step in the analysis is to obtain an axisymmetric solution for each segment. The axisymmetric solution is obtained from the SAGA I Finite Element Program developed at the University of Illinois.[6,7] This program is capable of performing a stress and displacement analysis of layered, orthotropic bodies of revolution.

The configuration to be analyzed is defined by its cross-section in the r-z plane. Then the geometry is divided into quadrilateral elements as shown in Figure 3. Each quadrilateral element is further divided into four triangular elements as shown in Figure 4. For each triangular element the three components of displacement are defined in terms of linear variation in the r and z coordinates as follows:

$$
\begin{aligned}
u &= a_1 + a_2 r + a_3 z \\
v &= a_4 + a_5 r + a_6 z \\
w &= a_7 + a_8 r + a_9 z
\end{aligned}
\qquad (2.1)
$$

where u, v, w are the displacement components in r, z, and θ directions respectively. Using equation (2.1) it is possible to relate the nine nodal displacements in the triangular element to the unknown coefficients a_1, a_2, etc. and the coordinates of the nodes in the r-z plane.

Let us now consider the appropriate strain-displacement relations:

$$\begin{aligned} \varepsilon_r &= \frac{\partial u}{\partial r} \\ \varepsilon_z &= \frac{\partial v}{\partial z} \\ \varepsilon_\theta &= \frac{u}{r} \\ \gamma_{rz} &= \frac{\partial u}{\partial z} + \frac{\partial v}{\partial r} \\ \gamma_{z\theta} &= \frac{\partial w}{\partial z} \\ \gamma_{\theta r} &= \frac{\partial w}{\partial r} - \frac{w}{r} \end{aligned} \tag{2.2}$$

By solving for a_1, a_2, etc. in terms of the nodal displacements, it is possible to combine equations (2.1) and (2.2) and obtain the matrix equation:

$$\{\varepsilon\} = [B]\ \{\delta\} \tag{2.3}$$

where $\{\varepsilon\}$ represents the six strain components, $\{\delta\}$ is the nine nodal displacements, and [B] is explained in Appendix A.

Next, consider the stress-strain relations. The axisymmetric analysis allowed for an orthotropic material. The three axes of orthotropy are denoted by n, s, and t. The basic stress-strains relations are defined in the n, s, t coordinates as follows:

$$\varepsilon_n = \frac{1}{E_n} (\sigma_n - \nu_{ns}\sigma_s - \nu_{nt}\sigma_t)$$

$$\varepsilon_s = \frac{1}{E_s} (\sigma_s - \nu_{st}\sigma_t - \nu_{sn}\sigma_n)$$

$$\varepsilon_t = \frac{1}{E_t} (\sigma_t - \nu_{tn}\sigma_n - \nu_{ts}\sigma_s)$$

$$\gamma_{ns} = \sigma_{ns}/G_{ns} \tag{2.4}$$

$$\gamma_{st} = \sigma_{st}/G_{st}$$

$$\gamma_{tn} = \sigma_{tn}/G_{tn}$$

where E_n, E_s, E_t, G_{ns}, G_{st}, G_{tn}, ν_{ns}, ν_{st}, and ν_{tn} are nine independent material coefficients. The remaining three coefficients in equation (2.4) are given by:

$$\nu_{sn} = \nu_{ns} (E_s/E_n)$$

$$\nu_{nt} = \nu_{tn} (E_n/E_t) \tag{2.5}$$

$$\nu_{ts} = \nu_{st} (E_t/E_s)$$

Using the angles α and β as defined in Figure 5 the stress-strain relations can be transformed into the r, θ, z system. The relations given in equation (2.4) can also be inverted. This results in the matrix relation:

$$\{\sigma\} = [D] \{\varepsilon\} \tag{2.6}$$

where $\{\sigma\}$ represents the six stress components in the r, θ, z coordinate system, and [D] is further explained in Appendix A.

Employing standard finite element techniques[1] the equilibrium equations for the triangular element are obtained from the principle of virtual work. These equations can be written as:

$$[k]\ \{\delta\} = \{f\} \tag{2.7}$$

where $\{\delta\}$ is the nodal displacement matrix and $\{f\}$ is nodal force matrix. In equation (2.7) [k] is the stiffness matrix and is given by:

$$[k] = \int_V [B]^T\ [D]\ [B]\ d\,V \tag{2.8}$$

where the integration is over the volume of the element.

By satisfying equilibrium at each nodal point, a global matrix equation can be formed as:

$$[K]\ \{q\} = \{F\} \tag{2.9}$$

where [K] is the global stiffness matrix and $\{q\}$ and $\{F\}$ represent the global displacement and global force matrix respectively.

Once this equation is formed it can be solved for the nodal displacements $\{q\}$. Then the stresses and strains can be found from equations (2.3) and (2.6).

Nonaxisymmetric Analysis

While the axisymmetric solution for each segment is being determined, the nonaxisymmetric force and stiffness matrices are calculated and stored. In assembling the nonaxisymmetric stiffness matrix the following displacement functions are used for each triangular element:

$$\begin{aligned} u &= a_1 + a_2 r + a_3 z + \theta(b_1+b_2 r+b_3 z) \\ v &= a_4 + a_5 r + a_6 z + \theta(b_4+b_5 r+b_6 z) \\ w &= a_7 + a_8 r + a_9 z + \theta(b_7+b_8 r+b_9 z) \end{aligned} \tag{2.10}$$

Also the nonaxisymmetric strain displacement equations are used. These are:

$$\varepsilon_r = \frac{\partial u}{\partial r}$$

$$\varepsilon_z = \frac{\partial v}{\partial z}$$

$$\varepsilon_\theta = \frac{u}{r} + \frac{1}{r}\frac{\partial w}{\partial \theta}$$

$$\gamma_{rz} = \frac{\partial u}{\partial z} + \frac{\partial v}{\partial r} \tag{2.11}$$

$$\gamma_{z\theta} = \frac{1}{r}\frac{\partial v}{\partial \theta} + \frac{\partial w}{\partial z}$$

$$\gamma_{\theta r} = \frac{1}{r}\frac{\partial u}{\partial \theta} + \frac{\partial w}{\partial r} - \frac{w}{r}$$

where again u, v and w are the displacements in the r, z and θ directions.

These equations along with equation (2.6) can be combined to form a nonaxisymmetric stiffness matrix as follows:

$$[S_1] = \int_V [B_1]^T \, [D] \, [B_1] \, d\,V \tag{2.12}$$

where $[S_1]$ is the nonaxisymmetric stiffness matrix and $[B_1]$ is the nonaxisymmetric strain-displacement matrix obtained by combining equations (2.10) and (2.11). The matrix $[B_1]$ is defined as:

$$\{\varepsilon\} = [B_1] \, \{\delta\} \tag{2.13}$$

where $\{\delta\}$ is the eighteen nonaxisymmetric displacement components (i.e., 3 displacements u, v, and w at each node on both faces of the segment.) The $[B_1]$ matrix is derived in Appendix A.

In general, the displacements generated by the axisymmetric solution will not be compatible with the displacements generated for an adjacent segment. Consequently, some changes need to be made in these displacements to obtain the real situation. The total displacement is now represented by two components. If we call the total displacement $\{u\}$, we can write:

$$\{u\} = \{u_a\} + \{u_p\} \tag{2.14}$$

where $\{u_a\}$ is the displacement from the axisymmetric solution and $\{u_p\}$ is the additional perturbation displacement.

For each side of a segment the perturbation displacements are expressed as follows:

$$\begin{aligned} u &= a_1 + a_2 r + z(a_3 + a_4 r) \\ v &= a_5 + a_6 r + z(a_7 + a_8 r) \\ w &= a_9 + a_{10} r + z(a_{11} + a_{12} r) \end{aligned} \tag{2.15}$$

where r, z, and θ refer again to coordinates in Figure 1. Similar expressions can be written for the other side of the segment.

If the total displacements are forced to match for two adjacent segments at four nodal points, a sufficient number of equations to be able to solve for the coefficients of the perturbation displacement equations will be obtained. These nodes, where the solutions match are called connecting nodes. Then for each segment:

$$\{u_{cp}\} = [R]\ \{p\} \tag{2.16}$$

where $\{u_{cp}\}$ is the perturbation displacement at the connecting nodes, [R] is a matrix of the connecting nodal coordinates further defined in Appendix A, and $\{p\}$ is the unknown perturbation coefficients. Solving for $\{p\}$ yields:

$$\{p\} = [R]^{-1}\ \{u_{cp}\} \tag{2.17}$$

Also, for each quadrilateral element in a particular segment:

$$\{u_p\} = [G]\ \{p\} \tag{2.18}$$

where [G] is the matrix of nodal coordinates for the element further defined in Appendix A. Substituting equation (2.17) into equation (2.18) yields:

$$\{u_p\} = [G]\ [R]^{-1}\ \{u_{cp}\} \tag{2.19}$$

Also for each element:

$$\{f\} = [S_1]\ (\{u_a\} + \{u_p\}) \tag{2.20}$$

where $[S_1]$ is the nonaxisymmetric stiffness matrix and {f} is the internal force vector. Now, applying the principle of virtual work to equation (2.20) gives:

$$\delta u_p^T\ \{f\} = \delta u_p^T\ [S_1]\ (\{u_a\} + \{u_p\}) \tag{2.21}$$

From equation (2.19), it can be seen that:

$$\delta u_p^T = \delta u_{cp}^T\ [R]^{-1T}\ [G]^T \tag{2.22}$$

Substituting this into equation (2.21) and summing the contribution of all the elements in the segment, results in:

$$\begin{aligned} W_I = {} & \delta u_{cp}^T \Sigma([R]^{-1T}[G]^T[S_1]\{u_a\} \\ & + [R]^{-1T}[G]^T[S_1][G][R]^{-1}\{u_{cp}\}) \end{aligned} \tag{2.23}$$

Also, there are contributions from the external body forces. This can be written as:

$$\delta u_p^T \{f_E\} = \delta u_p^T \{P_1\} \quad (2.24)$$

where $\{P_1\}$ is the external body force vector for an individual element. The same process of substitution and summing over all the elements gives:

$$W_E = \delta u_{cp}^T \Sigma [R]^{-1T}[G]^T\{P_1\} \quad (2.25)$$

Also there will be a contribution from the concentrated loads. This contribution can be stated as:

$$\delta u_p^T \{fc\} = \delta u_p^T \{g\} \quad (2.26)$$

where $\{g\}$ is the concentrated force vector for the element. Then, for the whole segment:

$$W_c = \delta u_{cp}^T \Sigma [R]^{-1T}[A]^T\{g\} \quad (2.27)$$

where [A] is a matrix relating the perturbation displacements at the nodes where the concentrated forces are applied to the perturbation coefficient matrix. That is:

$$\{u_p\} = [A] \{p\} \quad (2.28)$$

at nodes where concentrated forces are applied. The [A] matrix is presented in more detail in Appendix A.

Balancing the virtual work done by all the forces gives:

$$W_E + W_c = W_I \quad (2.29)$$

Here, the following substitutions can be made:

$$\{F_E\} = [R]^{-1T} (\Sigma\ [G]^T \{P_1\} + [A]^T \{g\})$$

$$[S_k] = [R]^{-1T} (\Sigma\ [G]^T [S_1] [G]) [R]^{-1} \qquad (2.30)$$

$$\{F_I\} = [R]^{-1T} (\Sigma\ [G]^T [S_1] \{u_a\})$$

where the $[R]^{-1^T}$ has been brought outside the summation since it is constant for the entire segment, and the summation is over all the elements in the segment. The terms in the summation can be easily calculated during the axisymmetric solution and later be multiplied by the $[R]^{-1}$ terms. Then $\{F_E\}$, $[S_K]$, and $\{F_I\}$ can be stored for each segment. These substitutions give:

$$[S_k] \{u_{cp}\} + \{F_I\} = \{F_E\} \qquad (2.31)$$

Since it is the total displacements that must match at the connecting nodes, equation (2.31) can be written as:

$$[S_k] \{u_{cT}\} - [S_k] \{u_{ca}\} + \{F_I\} = \{F_E\} \qquad (2.32)$$

where $\{u_{ca}\}$ is the axisymmetric displacement at the connecting nodes and $\{u_{cT}\}$ is the total displacement at the connecting nodes. Everything is known except $\{u_{cT}\}$. Combining terms in equation (2.32) yields:

$$[S_k] \{u_{cT}\} = \{F_c\} \qquad (2.33)$$

where $\{F_C\}$ is a combination of $\{F_E\}$, $\{F_I\}$, and $[S_K]$ $\{u_{ca}\}$. There will be one matrix equation for each segment. Now $[S_K]$ can be assembled into a banded global stiffness matrix for the overall structure. Then appropriate boundary conditions can be applied to restrict the rigid body motion, preserve symmetries, or comply with external restraints. Then equation (2.33) can be solved for the $\{u_{cT}\}$ vector, giving the total displacements at all of the connecting nodes.

Now it is a simple matter to work back from this point to obtain the stresses and strains for each element in each segment. For a given segment $\{u_{cT}\}$ and $\{u_{ca}\}$ are known, and therefore, $\{u_{cp}\}$ can be calculated from:

$$\{u_{cp}\} = \{u_{cT}\} - \{u_{ca}\} \tag{2.34}$$

Then knowing $\{u_{cp}\}$ and $[R]^{-1}$, $\{p\}$ can be obtained from equation (2.17). Then for each element the perturbation displacements $\{u_p\}$ can be obtained from equation (2.18). Having determined $\{u_a\}$ and $\{u_p\}$, it is easy to get the total displacement from equation (2.14).

At this point, it is simple to apply the basic finite element relations developed earlier:

$$\begin{aligned} \{\varepsilon\} &= [B_1]\ \{\delta\} \\ \{\sigma\} &= [D]\ \{\varepsilon\} \end{aligned} \tag{2.35}$$

Comparison to Substructure Analysis

This method of analyzing nonaxisymmetric structures is geometrically similar to the substructure analysis. The basic difference between the substructure analysis and the perturbation analysis is in the way total displacements are calculated. In the perturbation analysis the displacements are the sum of the axisymmetric and the perturbation displacements. In an equivalent substructure analysis the internal degrees of freedom are eliminated in each substructure and a set of equations similar to equation (2.15) is obtained except instead of representing only part of the displacements, they represent the total displacements.

With these differences the perturbation analysis outlined previously can be converted into a substructure analysis by modifying equation (2.19) to read:

$$\{u\} = [G]\ [R]^{-1}\ \{u_{cT}\} \tag{2.36}$$

where $\{u\}$ is the displacement matrix for an element and $[G]$ and $[R]$ are defined as before. It is possible to solve for $\{u_{cT}\}$ by using equation (2.33) by replacing $\{F_c\}$ with $\{F_E\}$. That is:

$$[S_k] \{u_{cT}\} = \{F_E\} \tag{2.37}$$

with $\{F_E\}$ and $[S_K]$ defined as in equation (2.30). Although there is major conceptual differences between the perturbation and substructure methods, there are only minor changes required in the computer programs. Therefore, it is possible to obtain solutions for both methods and compare answers.

III. NUMERICAL RESULTS OF ELASTIC ANALYSIS

To test the accuracy of this analysis three examples were used. On each example both a perturbation and a substructure analysis were performed. Whenever possible the results were compared to known solutions.

Axisymmetric Problem

The first test of the perturbation method is that if the configuration is indeed completely axisymmetric, the solution should be identical to the axisymmetric solution. To test this requirement, an example of an axisymmetric disk with a center hole, loaded with internal pressure, was used. Both the perturbation and the substructure analyses were performed.

In both analyses two segments in the circumferential direction were used. Each segment was divided into four elements in the radial direction and one element in the axial direction. The disk had an inner radius of 5 inches and an outer radius of 15 inches, and was 2 inches thick. Thus, each element measured 2.5 inches by 2 inches.

In Table 1 the results for the radial stress, σ_{rr}, and the circumferential stress, $\sigma_{\theta\theta}$, are presented. The axisymmetric results presented were obtained from the SAGA I finite element program[6,7]. These results agree well with analytic results. From this table it can be seen that the perturbation analysis does yield answers that agree well with the axisymmetric solution for both σ_{rr} and $\sigma_{\theta\theta}$. It can also be seen that the substructure analysis gives reasonable answers for $\sigma_{\theta\theta}$ but shows large errors in computing the radial stress, σ_{rr}.

Nonaxisymmetric Disk

The second example tested was a nonaxisymmetric disk under internal pressure as shown in Figure 6. Again both a perturbation and a substructure analysis were performed. For this configuration six segments in the circumferential direction were used. Three of these segments were the same as those described in the previous section, and the other three were of the same thickness but had only two elements in the radial direction. These elements had an outer radius of ten inches.

Since this problem is a plane stress problem in the r-θ plane, the solution obtained by the perturbation analysis can be compared to a plane stress solution. A plane stress solution was obtained using the SAAS Finite Element Program.[9] In Figure 7 results for radial displacement as a function of radial displacement are presented. Results for circumferential stress and radial stress as a function of radial location are presented in Figures 8 and 9 and Tables 2 and 3. It can be seen that the perturbation solution agrees well with the plane stress solution which is known to be accurate. Furthermore, the substructure results disagree with the plane stress results for radial stresses and radial displacements. The circumferential stresses do agree well for the substructure analysis. These results are not shown in Figure 8 for the sake of clarity since the results of all three analyses are very close.

Three Dimensional Tube

The third example considered is a three dimensional tube shown schematically in Figure 10. Again six segments in the θ-direction were used. Four elements were used in the axial or z-direction. The end labelled the "unsymmetric end" in Figure 10 has an end view the same as shown in Figure 6. The "axisymmetric end" has an end view as shown in Figure 11. The tube measured eight inches in the axial direction. The geometric discontinuity is at the mid-point of the tube. As before, the tube is subjected to an internal pressure.

Once again the problem was solved using both the substructure and the perturbation methods. Results for the circumferential stresses are presented in Table 4, and for the radial stresses in Table 5. Since this problem is a true three dimensional problem, no analytic solution was available. However, each end should approach a limit solution as represented by the previous two examples. From Table 4 and 5 it can be seen that the stresses do approach the so-called limit solutions. Also it can be seen that again the substructure method does not predict the radial stresses well. A boundary condition of the problem is that the outer boundary is stress free. The perturbation method fulfills this boundary condition while from Table 5 it is obvious that the substructure method violates it.

Effect of the Choice of Connecting Nodes

One ambiguity in this method is the choice of the connecting nodes. If the accuracy of the solution depends on this choice, the same constraints must be placed on this choice. To see what effect the choice of connecting nodes has example three, the nonaxisymmetric tube, was solved using three sets of connecting nodes. These three sets of nodes are shown in Figure 12, where the circles represent the connecting nodes. The choices ranged from widely spaced as shown in Figure 12a to very close together, in fact all nodes of one element, as shown in Figure 12c. In Table 6 results are shown for these three cases. Results for the other stresses show similar changes. It can be seen that the results are insensitive to the choice of connecting nodes. Thus, this choice can be arbitrary and no restraints need be placed on the generality of the method.

IV. THE ELASTIC-PLASTIC ANALYSIS

Many applications of the perturbation analysis described in Chapter II occur in the field of interior ballistics. In many cases the structures involved must survive an extreme environment only once. Therefore, the components are often loaded well into the plastic range. To analyze such problems the perturbation method was expanded to include elastic-plastic strain hardening material behavior.

Incremental Equation for a General Strain Hardening Material

To describe the plastic work hardening behavior of a material three things are needed:

a) an initial yield condition
b) a flow rule relating plastic strain increments to the stress and the stress increment
c) a hardening rule

For this work the Hill's yield criterion[4] for orthotropic materials was chosen. This reduces to the von Mises-Hencky yield condition if the material is isotropic. The Hill's condition will be discussed more fully later, but it can be represented symbolically as:

$$F\,(\sigma_{ij}) = 1 \tag{4.1}$$

at the point where yielding occurs.

The Prager or the kinematic strain hardening rule was used[5]. This hardening rule assumes that the yield surface retains its initial size and shape but undergoes a translation in the direction of the plastic strain increment. This is illustrated in Figure 13. After plastic flow begins the yield surface can be written as:

$$F\,(\sigma_{ij} - \alpha_{ij}) = 1 \tag{4.2}$$

where the α_{ij}'s represent the translation of yield surface. The assumption that this translation is in the direction of the plastic strain increment can be written as:

$$d\alpha_{ij} = c \, d\varepsilon^p_{ij} \tag{4.3}$$

where c is a constant for the material.

The flow rule chosen was also due to von Mises, namely:

$$d\varepsilon^p_{ij} = \frac{\partial F}{\partial \sigma_{ij}} d\lambda \; ; \; d\lambda \geq 0 \tag{4.4}$$

The scalar $d\lambda$ in equation (4.4) can be determined by the fact that the stresses remain on the yield surface during plastic flow. This condition can be expressed as:

$$(d\sigma_{ij} - d\alpha_{ij}) \frac{\partial F}{\partial \sigma_{ij}} = 0 \tag{4.5}$$

Substituting from equations (4.3)-(4.4) into equation (4.5) gives:

$$(d\sigma_{ij} - c \frac{\partial F}{\partial \sigma_{ij}} d\lambda) \frac{\partial F}{\partial \sigma_{ij}} = 0 \tag{4.6}$$

Solving equation (4.6) for $d\lambda$ gives:

$$d\lambda = \{\frac{1}{c} \frac{\partial F}{\partial \sigma_{ij}} d\sigma_{ij}\} \{\frac{\partial F}{\partial \sigma_{kl}} \frac{\partial F}{\partial \sigma_{kl}}\}^{-1} \tag{4.7}$$

The total strain increment is the sum of the elastic and the plastic strain increments. Thus:

$$d\varepsilon^e_{ij} = d\varepsilon_{ij} - d\varepsilon^p_{ij} \tag{4.8}$$

where $d\varepsilon_{ij}$ is the total strain increment.

Substituting for $d\varepsilon^e_{ij}$ in the elastic stress-strain relationship gives:

$$d\sigma_{ij} = E_{ijkl} \, [d\varepsilon_{kl} - d\varepsilon^p_{kl}] \tag{4.9}$$

where E_{ijkl} is the elastic-stress-strain tensor.

Equation (4.5) can be rewritten as:

$$(d\sigma_{ij} - c\, d\varepsilon^{p}_{ij}) \frac{\partial F}{\partial \sigma_{ij}} = 0 \tag{4.10}$$

Then substituting equations (4.4), (4.7) and (4.9) into equation (4.10) gives:

$$\{E_{ijkl} [d\varepsilon_{kl} - \frac{\partial F}{\partial \sigma_{kl}} d\lambda] - c \frac{\partial F}{\partial \sigma_{ij}} d\lambda\} \frac{\partial F}{\partial \sigma_{ij}} = 0 \tag{4.11}$$

This can be rewritten as:

$$E_{ijkl}\, d\varepsilon_{kl} \frac{\partial F}{\partial \sigma_{ij}} = \{E_{ijkl} \frac{\partial F}{\partial \sigma_{ij}} \frac{\partial F}{\partial \sigma_{kl}} + c \frac{\partial F}{\partial \sigma_{ij}} \frac{\partial F}{\partial \sigma_{ij}} \} d\lambda \tag{4.12}$$

Then defining the bracketed term as 1/D where D is the scalar:

$$D = \{E_{ijkl} \frac{\partial F}{\partial \sigma_{kl}} \frac{\partial F}{\partial \sigma_{ij}} + c \frac{\partial F}{\partial \sigma_{ij}} \frac{\partial F}{\partial \sigma_{ij}} \}^{-1} \tag{4.13}$$

equation (4.12) can be written as:

$$d\lambda = D\, E_{ijkl} \frac{\partial F}{\partial \sigma_{ij}} d\varepsilon_{kl} \tag{4.14}$$

Substituting equation (4.14) into equation (4.4) gives a relation between the total strain increment and the plastic strain increment as:

$$d\varepsilon^{p}_{ij} = D\, E_{mnkl} \frac{\partial F}{\partial \sigma_{mn}} \frac{\partial F}{\partial \sigma_{ij}} d\varepsilon_{kl} \tag{4.15}$$

or:

$$d\varepsilon^{p}_{ij} = C_{ijkl}\, d\,\varepsilon_{kl} \qquad (4.16)$$

where C_{ijkl} is defined from equation (4.15).

Then equation (4.9) can be written as:

$$d\sigma_{kl} = [E_{ijkl} - E_{klmn}\, C_{mnij}]\, d\,\varepsilon_{ij} \qquad (4.17)$$

or:

$$d\sigma_{ij} = A_{ijkl}\, d\,\varepsilon_{kl} \qquad (4.18)$$

where:

$$A_{ijkl} = [E_{ijkl} - E_{ijmn}\, C_{mnkl}] \qquad (4.19)$$

For the isotropic case, c can be evaluated from a uniaxial test. It can be shown that:

$$c = \frac{2}{3}\,\frac{E\cdot E_T}{E-E_T} \qquad (4.20)$$

where E_T is the tangent modulus as shown in Figure 14, and E is Young's Modulus. In general E_T will depend on the stress state at a given time. Thus by knowing the elastic constants and the tangent modulus the stresses can be obtained from the total strains.

<u>Application of General Strain Hardening Relations to an Orthotropic Material</u>

In applying the equations developed in the previous section to an orthotropic material, the Hill's yield condition was used. This yield condition can be expressed as:

$$F(\sigma_{ij}) = \frac{\sigma_{11}^2}{Y_{11}^2} + \frac{\sigma_{22}^2}{Y_{22}^2} + \frac{\sigma_{33}^2}{Y_{33}^2} + \frac{\sigma_{12}^2}{Y_{12}^2} + \frac{\sigma_{23}^2}{Y_{23}^2} + \frac{\sigma_{13}^2}{Y_{13}^2} + \bar{Y}_{11}\sigma_{22}\sigma_{33} + \bar{Y}_{22}\sigma_{11}\sigma_{33} + \bar{Y}_{33}\sigma_{11}\sigma_{22} = 1 \tag{4.21}$$

where Y_{11}, Y_{22} and Y_{33} are the yield stresses in simple tension in the three orthotropic directions and Y_{12}, Y_{23} and Y_{13} are the corresponding yield stresses in simple shear. The $\bar{Y}$ terms are functions of the yield stresses. These functions are:

$$\begin{aligned} \bar{Y}_{11} &= \frac{1}{Y_{11}^2} - \frac{1}{Y_{22}^2} - \frac{1}{Y_{33}^2} \\ \bar{Y}_{22} &= \frac{1}{Y_{22}^2} - \frac{1}{Y_{33}^2} - \frac{1}{Y_{11}^2} \\ \bar{Y}_{33} &= \frac{1}{Y_{33}^2} - \frac{1}{Y_{11}^2} - \frac{1}{Y_{22}^2} \end{aligned} \tag{4.22}$$

After strain hardening has occurred equation (4.21) becomes:

$$\frac{(\sigma_{11}-\alpha_{11})^2}{Y_{11}^2} + \frac{(\sigma_{22}-\alpha_{22})^2}{Y_{22}^2} + \frac{(\sigma_{33}-\alpha_{33})^2}{Y_{33}^2} + \frac{(\sigma_{12}-\alpha_{12})^2}{Y_{12}^2} + \frac{(\sigma_{23}-\alpha_{23})^2}{Y_{23}^2} + \frac{(\sigma_{13}-\alpha_{13})^2}{Y_{13}^2} + \bar{Y}_{11}(\sigma_{22}-\alpha_{22})(\sigma_{33}-\alpha_{33}) + \bar{Y}_{22}(\sigma_{33}-\alpha_{33})(\sigma_{11}-\alpha_{11}) + \bar{Y}_{33}(\sigma_{11}-\alpha_{11})(\sigma_{22}-\alpha_{22}) = 1 \tag{4.23}$$

Then evaluating the $\frac{\partial F}{\partial \sigma_{ij}}$ terms gives:

$$\frac{\partial F}{\partial \sigma_{11}} = \frac{2(\sigma_{11}-\alpha_{11})}{Y_{11}^2} + \overline{Y}_{22}(\sigma_{33}-\alpha_{33}) + \overline{Y}_{33}(\sigma_{22}-\alpha_{22})$$

$$\frac{\partial F}{\partial \sigma_{22}} = \frac{2(\sigma_{22}-\alpha_{22})}{Y_{22}^2} + \overline{Y}_{33}(\sigma_{11}-\alpha_{11}) + \overline{Y}_{11}(\sigma_{33}-\alpha_{33})$$

$$\frac{\partial F}{\partial \sigma_{33}} = \frac{2(\sigma_{33}-\alpha_{33})}{Y_{33}^2} + \overline{Y}_{11}(\sigma_{22}-\alpha_{22}) + \overline{Y}_{22}(\sigma_{11}-\alpha_{11})$$

$$\frac{\partial F}{\partial \sigma_{12}} = \frac{2(\sigma_{12}-\alpha_{12})}{Y_{12}^2}$$

$$\frac{\partial F}{\partial \sigma_{23}} = \frac{2(\sigma_{23}-\alpha_{23})}{Y_{23}^2} \qquad (4.24)$$

$$\frac{\partial F}{\partial \sigma_{13}} = \frac{2(\sigma_{13}-\alpha_{13})}{Y_{13}^2}$$

There still remains the question of evaluating the hardening parameter, c. This was evaluated from a uniaxial test in the isotropic case, but there are three uniaxial tests which will give three different results in the orthotropic case. To solve for the hardening parameter two assumptions were used which reduce the generality of the analysis but increase its simplicity. These assumptions are:

a) transverse isotropy (i.e., $Y_{22} = Y_{33}$)
b) there is one predominant direction

Then by taking a uniaxial tension loading in the preferred direction, such that $\sigma_{11} \neq 0$ and all other stresses are zero, equation (4.24) becomes:

$$\frac{\partial F}{\partial \sigma_{11}} = \frac{2(\sigma_{11}-\alpha_{11})}{Y_{11}^2} - \alpha_{33}\,\overline{Y}_{22} - \alpha_{22}\,\overline{Y}_{33}$$

$$\frac{\partial F}{\partial \sigma_{22}} = -\frac{2\alpha_{22}}{Y_{22}^2} - \alpha_{33}\,\overline{Y}_{11} - \overline{Y}_{33}\,(\sigma_{11}-\alpha_{11})$$

$$\frac{\partial F}{\partial \sigma_{33}} = -\frac{2\alpha_{33}}{Y_{33}^2} - \alpha_{22}\,\overline{Y}_{11} + \overline{Y}_{22}\,(\sigma_{11}-\alpha_{11})$$

$$\frac{\partial F}{\partial \sigma_{12}} = \frac{\partial F}{\partial \sigma_{13}} = \frac{\partial F}{\partial \sigma_{23}} = 0 \tag{4.25}$$

By assuming transverse isotropy equation (4.22) becomes:

$$\overline{Y}_{11} = \frac{1}{Y_{11}^2} - \frac{2}{Y_{22}^2}$$

$$\overline{Y}_{22} = -\frac{1}{Y_{11}^2}$$

$$\overline{Y}_{33} = -\frac{1}{Y_{11}^2} \tag{4.26}$$

Also as a result of transverse isotropy in this uniaxial loading case:

$$\alpha_{22} = \alpha_{33} \tag{4.27}$$

Then, since the plastic flow is incompressible and $d\alpha_{ij}$ is proportional to $d\varepsilon_{ij}^{p}$, it follows that:

$$\alpha_{11} + \alpha_{22} + \alpha_{33} = 0 \tag{4.28}$$

and substituting from equation (4.27) gives:

$$\alpha_{11} = -2\alpha_{22} = -2\alpha_{33} \tag{4.29}$$

By substituting equations (4.26) and (4.29) into equation (4.25):

$$\frac{\partial F}{\partial \sigma_{11}} = \frac{2\sigma_{11} - 3\alpha_{11}}{Y_{11}^2}$$

$$\frac{\partial F}{\partial \sigma_{22}} = -\frac{1}{2}\,\frac{2\sigma_{11} - 3\alpha_{11}}{Y_{11}^2} \tag{4.30}$$

$$\frac{\partial F}{\partial \sigma_{33}} = -\frac{1}{2}\,\frac{2\sigma_{11} - 3\alpha_{11}}{Y_{11}^2}$$

Noting the common factor $(2\sigma_{11} - 3\alpha_{11})$ gives:

$$-\frac{1}{2}\,\frac{\partial F}{\partial \sigma_{11}} = \frac{\partial F}{\partial \sigma_{22}} = \frac{\partial F}{\partial \sigma_{33}} \tag{4.31}$$

Evaluating $d\lambda$ from equation (4.7) gives:

$$d\lambda = \frac{1/c \;\; d\sigma_{11}}{3/2 \;\; \partial F/\partial \sigma_{11}} \tag{4.32}$$

Then from equation (4.4):

$$d\,\varepsilon_{11}^{p} = \frac{2}{3}\,\frac{1}{c}\,d\sigma_{11} \tag{4.33}$$

or:

$$\frac{d\sigma_{11}}{d\varepsilon_{11}^{p}} = \frac{3}{2}\,C \tag{4.34}$$

This result is identical with the isotropic result. Equation (4.21) now becomes:

$$C = \frac{2}{3}\,\frac{E_{11} \cdot E_{11_T}}{E_{11} - E_{11_T}} \tag{4.35}$$

By assuming that this direction is the predominant one the value for "c" given by equation (4.35) can be used. This value is good for isotropic materials and transversely isotropic materials. Therefore, the program is limited in that it cannot have a general orthotropic material. However, the types of materials that are transversely isotropic includes fiber-matrix materials and many others.

Implementation of Elastic-Plastic Analysis in the Finite Element Program

In incorporating the theory for plastic flow into the existing elastic finite element program two simplifications were used.

The first simplification was assuming the uniaxial behavior of the material was bilinear. Figure 14 shows a bilinear material. By doing this E_T and c are reduced to constants independent of the stress state. This obviously leads to much simplification in the programming. The bilinear model is fairly accurate for most materials if the plastic deformation is not too large.

The second simplification is that the stresses were not forced to be on the yield surface during an entire load increment. Thus, the consistency equation given in equation (4.10) is violated. At the start of a load step an element is either plastic or elastic. If it is elastic, it is considered to be elastic for the entire load step. If it is plastic, all terms in equation (4.18) are computed using values of σ_{ij} at the start of the load step. At the end of the load step the stresses are checked for each element to see if it has yielded. The element is treated as either elastic or plastic in the next load step according to this check. Thus, the stresses may overshoot the actual yield surface for a given time step. This effect can be minimized if load steps are kept small. The alternative to this would be to do a solution based on the initial stresses, check the yield condition at the end and iterate to make sure the stress remains on the yield surface. This would cause a substantial increase in the execution time of the program. Since this analysis is by its very nature approximate, the substantial increase in execution time needed to obtain the more exact answers by iteration seems impractical.

V. NUMERICAL RESULTS OF ELASTIC-PLASTIC ANALYSIS

Axisymmetric Perfectly Plastic Problem

The first test of the elastic-plastic numerical analysis is that the numerical solution should match an analytic solution for an axisymmetric problem.

By letting $E_T = .001\ E$ in equation (4.35) a perfectly plastic material was approximated. Using the example of an isotropic axisymmetric disk with a hole in the center under internal pressure, a numerical solution was obtained. This solution was compared to an analytic solution of the same problem[10]. There is one difference between these solutions; namely, the numerical solution uses a von Mises yield condition while the analytic solution uses a Tresca yield condition.

Allowing for the difference in yield conditions the numerical solution agrees well with the analytic solution. In Figure 15 it can be seen that the elastic-plastic boundary for both solutions has the same shape and that yielding at a given radial location occurs at a higher pressure for the numerical solution. This is to be expected from the difference in the yield conditions (i.e., for this example the Tresca condition does predict yielding at a lower pressure than the von Mises condition).

The circumferential strain at the outer surface is shown a function of pressure in Figure 16. Again the analytic and numerical solutions agree well. Since there is more yielding at a given pressure with the Tresca condition, the structure is less stiff. Larger strains are to be expected with the analytic solution at a given pressure than with the numerical solution. This is shown by Figure 16.

Nonaxisymmetric Perfectly Plastic Problems

In doing the numerical solution in the axisymmetric example two segments were used in the θ direction. Each segment had a ratio of outer radius to inner radius of 2. Eight elements in the radial direction and one in the axial direction were used.

Two nonaxisymmetric cases were also studied. Both cases used two segments as before. In each case one segment was the same as the axisymmetric configuration. One configuration removed the outermost element from the other segment leaving seven elements in the radial direction. The second nonaxisymmetric configuration had only the inner four elements in the radial direction. These configurations are shown in Figure 17.

For both these configurations, a numerical solution was obtained. Since the material is perfectly plastic, the solution fails when the elastic-plastic boundary reaches the outer boundary of the structure. The radial displacement of the inner boundary at the interface between the two segments is shown in Figure 18. From this graph it can be seen that as material is removed from one of the segments the structure loses stiffness and there are greater displacements at a given pressure. While there is no analytic solution for comparison in the nonaxisymmetric case, these displacements do exhibit expected trends.

Hardening Cases, Nearly Elastic Examples

The final example involved the configurations denoted as case 1 and case 3 in Figure 17. These examples were solved numerically with $E_T = .9E$ in equation (4.20). For this value of the tangent modulus the numerical solutions would not be expected to vary greatly from the elastic solutions.

For the axisymmetric case, Figure 15 shows how the elastic-plastic boundary grows much slower than in the perfectly plastic case. In Figure 16 it can be seen that the circumferential strain as a function of pressure is very nearly linear. This is consistent with the solution being close to the linear elastic solution.

In Figure 19 the displacements of the inner surface at the interface between the two segments are shown. It can be seen that these displacements are also nearly linear as expected. The elastic solution is shown for comparison.

VI. CONCLUSIONS

The purpose of this investigation was to develop a method that could be used to analyze nonaxisymmetric configurations without performing a full 3-dimensional analysis. The analysis was broken into two parts: elastic and elastic-plastic.

The elastic analysis gives accurate results for a number of examples. These results were verified by comparison to results obtained from another finite element code of proven accuracy. Also these results are vastly superior to results obtained by the substructure method. The substructure method is another method which has certain geometric similarities to the perturbation method presented in this paper. Also the versatility of the method was demonstrated on the tube example. This showed that more complex configurations can be handled by the perturbation method.

For the elastic-plastic version the analysis is designed to handle both isotropic and orthotropic materials. Since the examples presented are all trying to establish the accuracy of the method, only problems with some means of judging the answers are presented. Since there are no analytic, experimental, or numerical results available for orthotropic materials, no orthotropic examples are presented. Obtaining such results experimentally was beyond the scope of this work.

However, for the isotropic elastic-plastic examples the results are accurate. In this case an analytic solution was available for comparison. These examples represent the wide variety of material behavior that can be approximated. Examples range from perfectly plastic to nearly elastic.

In conclusion, the method presented here gives an accurate, efficient means of analyzing a wide class of problems. The method is quite versatile in the geometric configurations it can analyze as well as the material behavior.

REFERENCES

1. O.C. Zienkiewicz, The Finite Element Method in Engineering Science, McGraw-Hill, London (1971).

2. J.S. Przemieniecki, Theory of Matrix Structural Analysis, McGraw-Hill, New York (1968).

3. A.R. Zak, J.N. Craddock, W.H. Drysdale, "Approximate Finite Element Method of Stress Analysis of Non-Axisymmetric Configurations," International Journal of Computers and Structures, vol. 9 (1978).

4. R. Hill, "A Theory of Yielding and Plastic Flow of Anisotropic Metals," Proceedings of the Royal Aeronautical Society, vol. 193 (1948).

5. R. Shield and H. Ziegler, "On Prager's Hardening Rule," Journal of Applied Mathematics and Physics, vol. IX, 1958.

6. A.R. Zak, "Finite Element Computer Program for the Analysis of Layered Orthotropic Structures," BRL Contract Report No. 158 (June 1974). (AD#783412)

7. A.R. Zak, "Numerical Analysis of Laminated Orthotropic Composite Structures," BRL Contract Report No. 272 (Nov. 1975). (AD#A018875)

8. J.S. Przemieniecki, "Matrix Structural Analysis of Substructures," Journal of American Institute of Aeronautics and Astronautics, vol. 1, pp. 138-147 (1963).

9. S.G. Sawyer, "BRLESC Finite Element Program for Axisymmetric, Plane Strain, and Plane Stress, Orthotropic Solids with Temperature-Dependent Material Properties," BRL Report No. 1539 (March 1971).(AD#727702)

10. C.R. Calladine, Engineering Plasticity, Pergamon Press, London (1969).

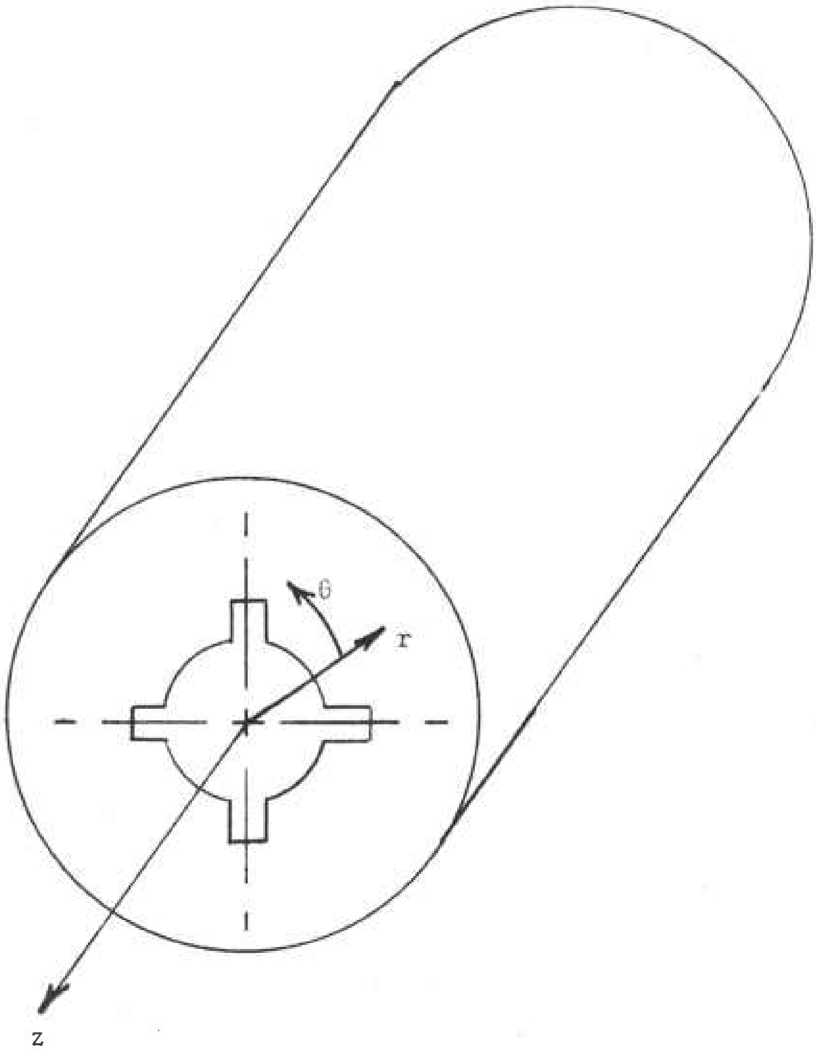


Figure 1. Typical nonaxisymmetric configuration to which the structural analysis method can be applied.

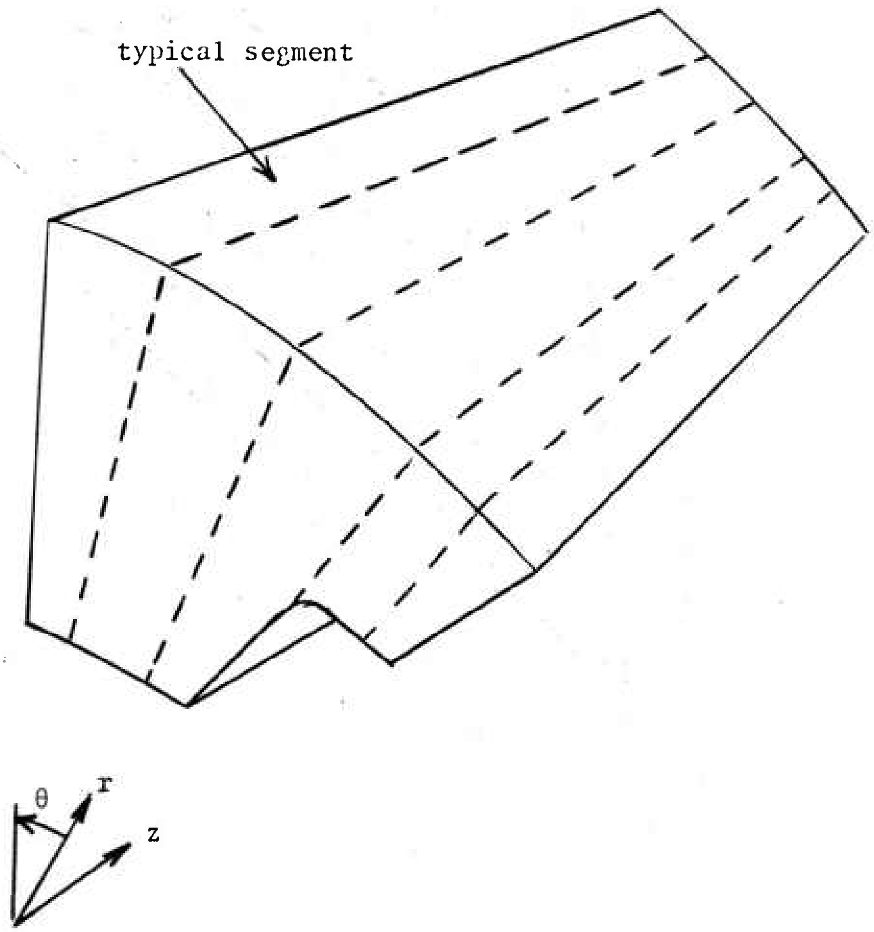


Figure 2. Configuration to be analyzed is divided into segments by plane sections in the r-z plane.

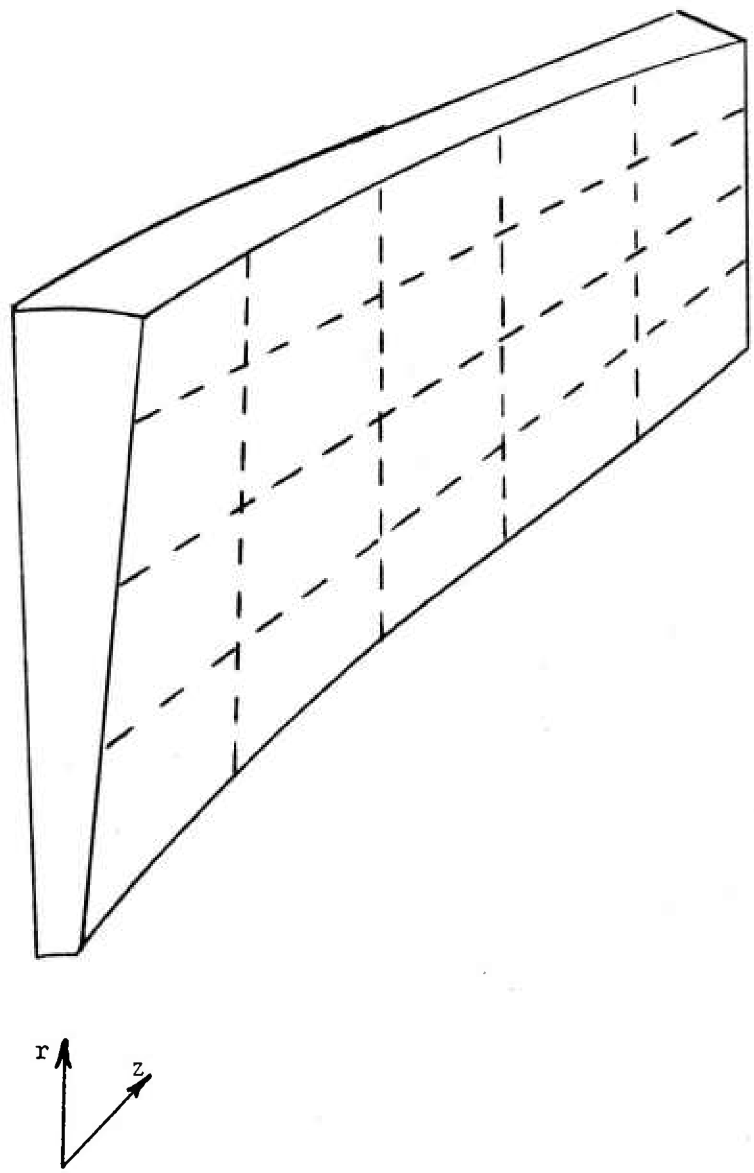


Figure 3. Division of a typical segment into a finite element grid in the r-z plane.

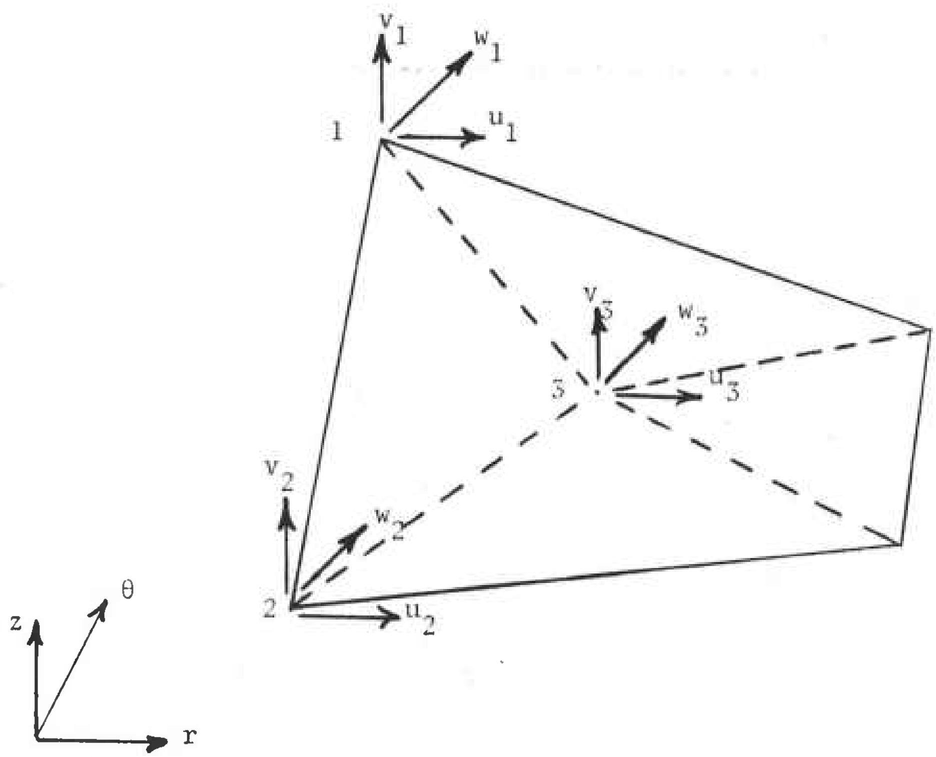


Figure 4. Subdivision of quadrilateral elements into triangular elements and definition of nodal displacements.

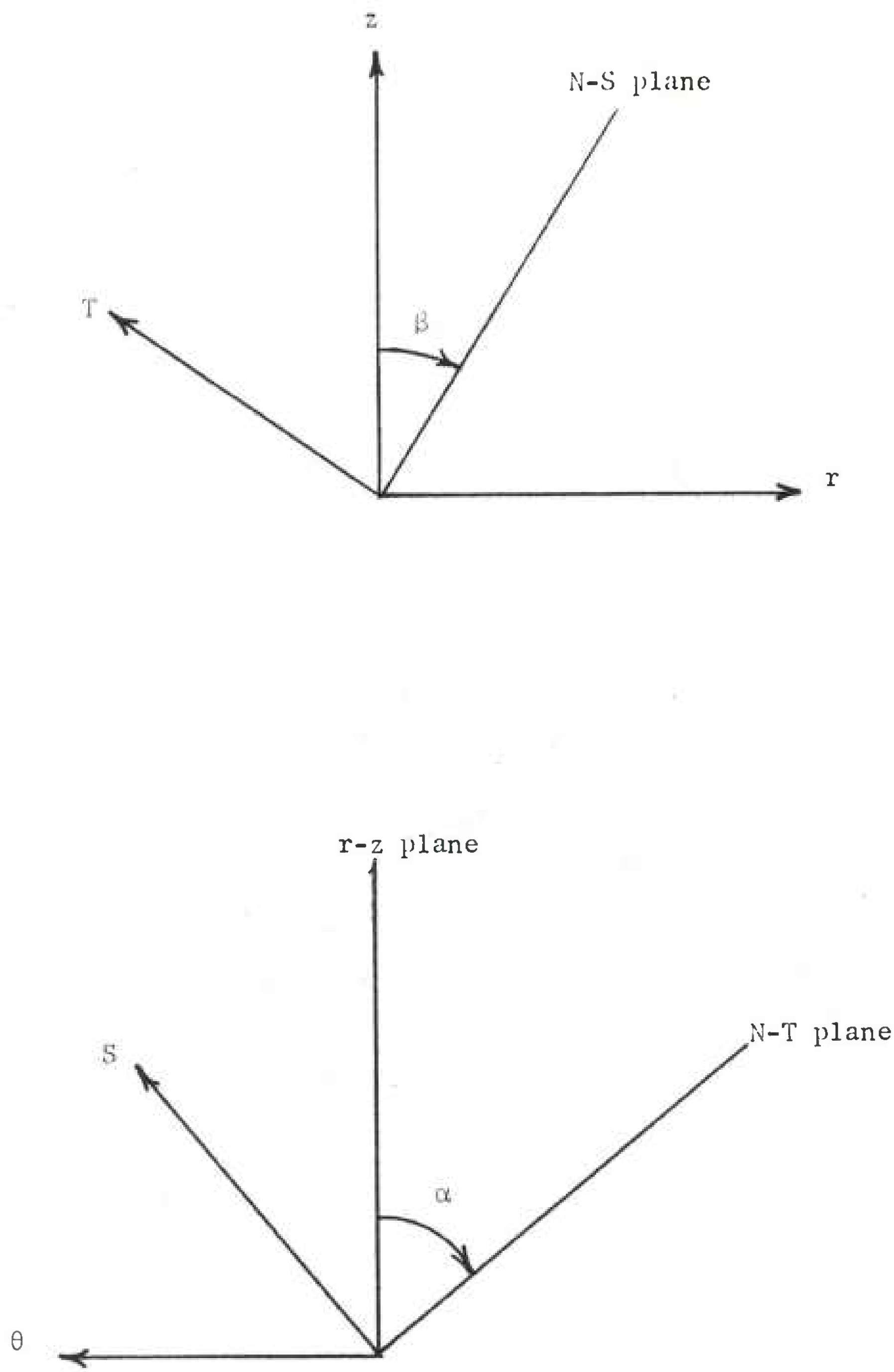


Figure 5. Relation between cylindrical and orthotropic coordinates.

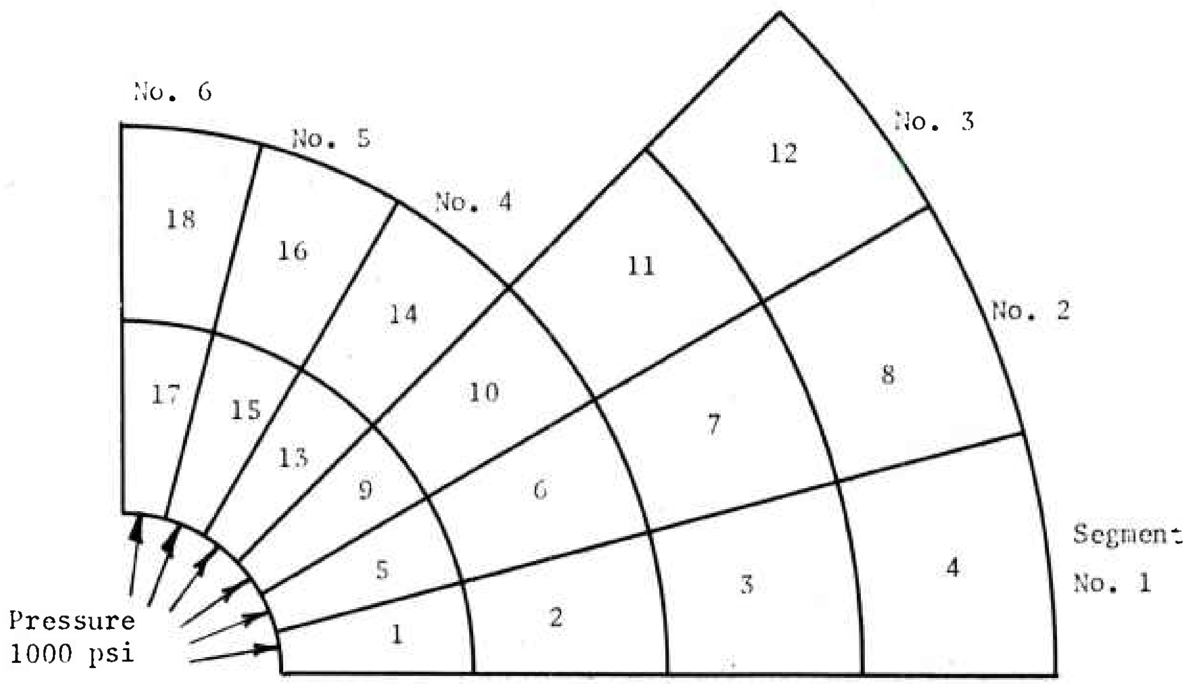


Figure 6. End view of nonaxisymmetric disk showing segment and element numbering systems.

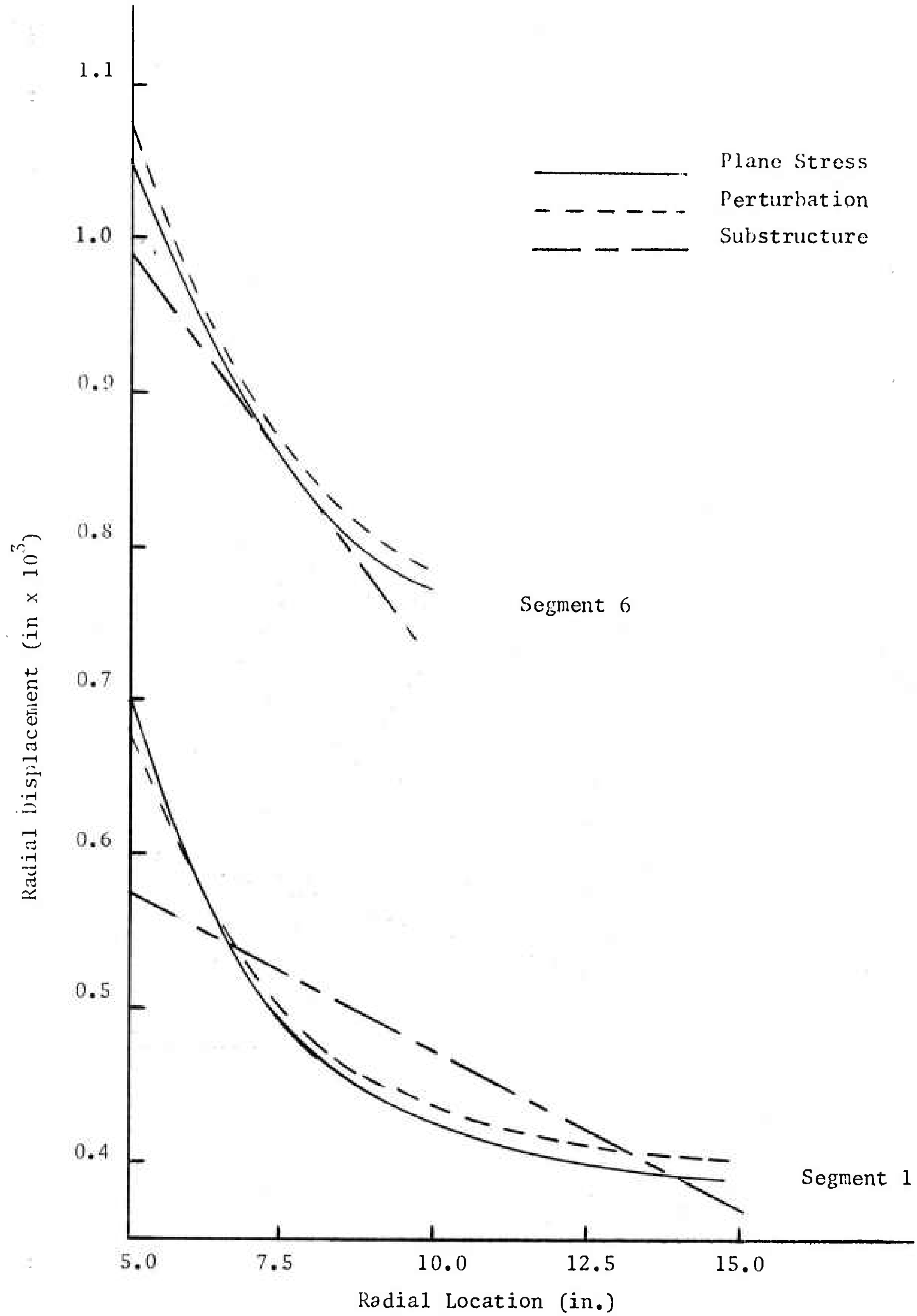


Figure 7. Radial displacement for nonaxisymmetric disk obtained from different methods of analysis.

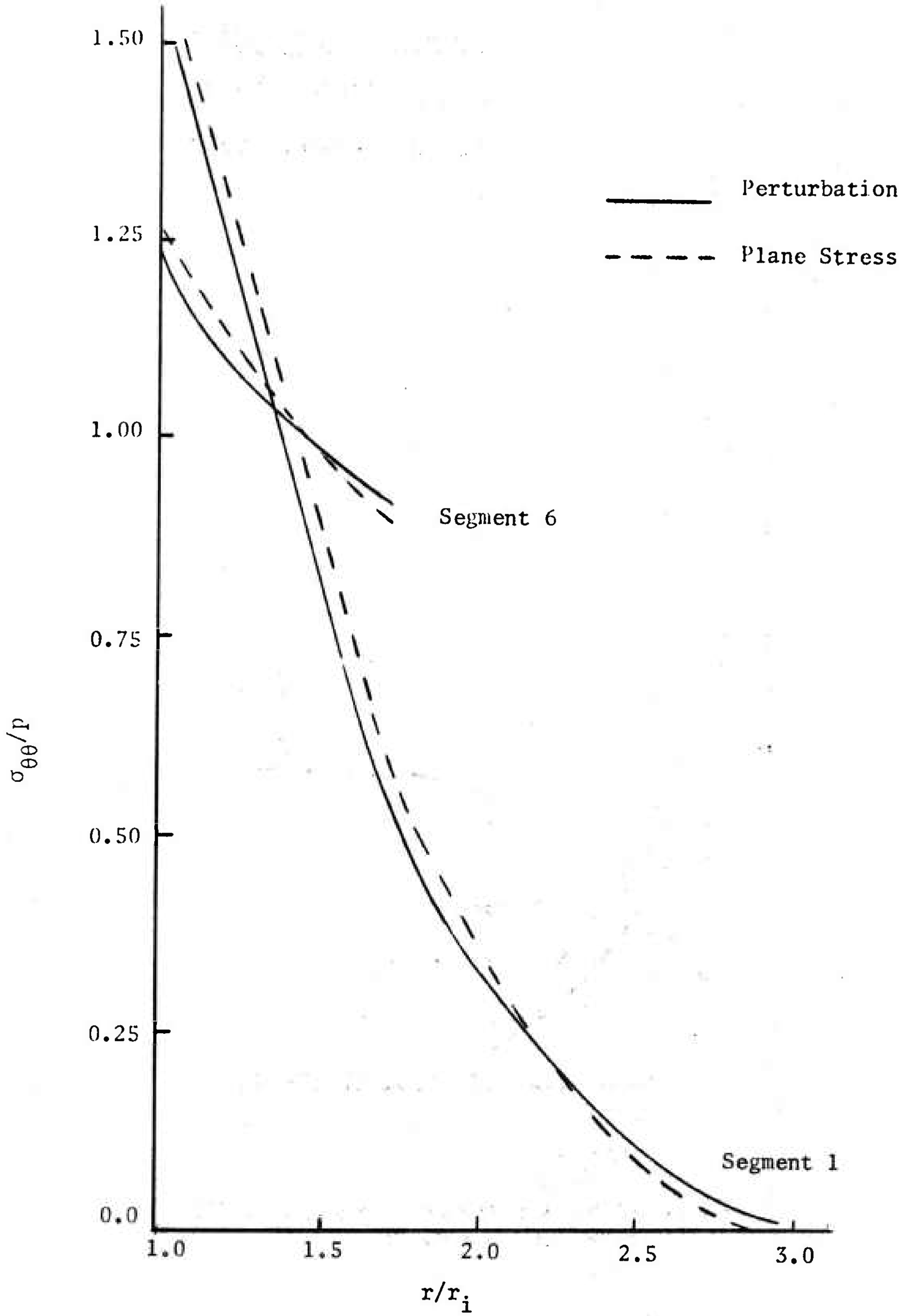


Figure 8. Circumferential stress for nonaxisymmetric disk obtained from different methods of analysis.

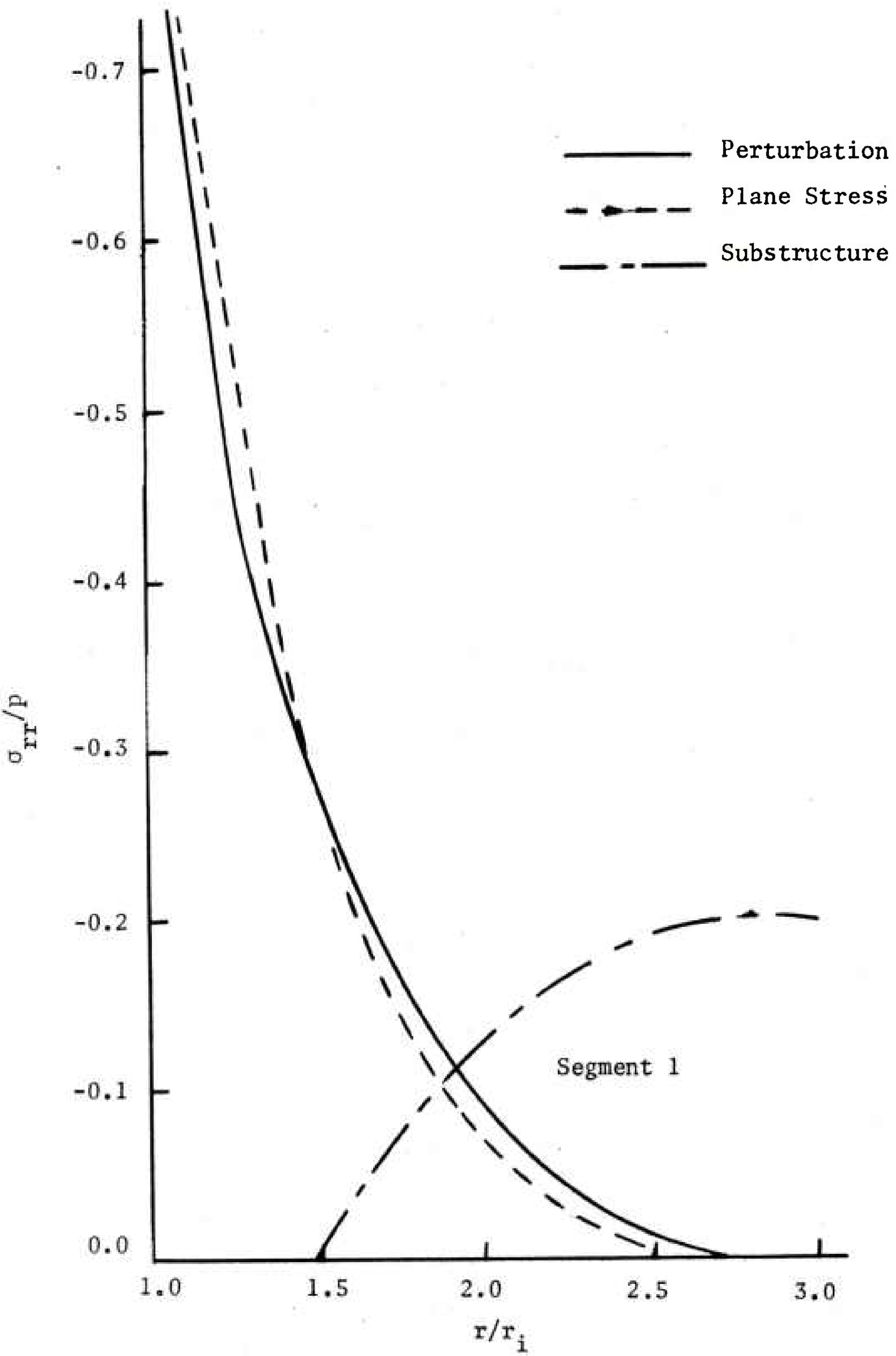


Figure 9. Radial stress for nonaxisymmetric disk obtained from different methods of analysis.

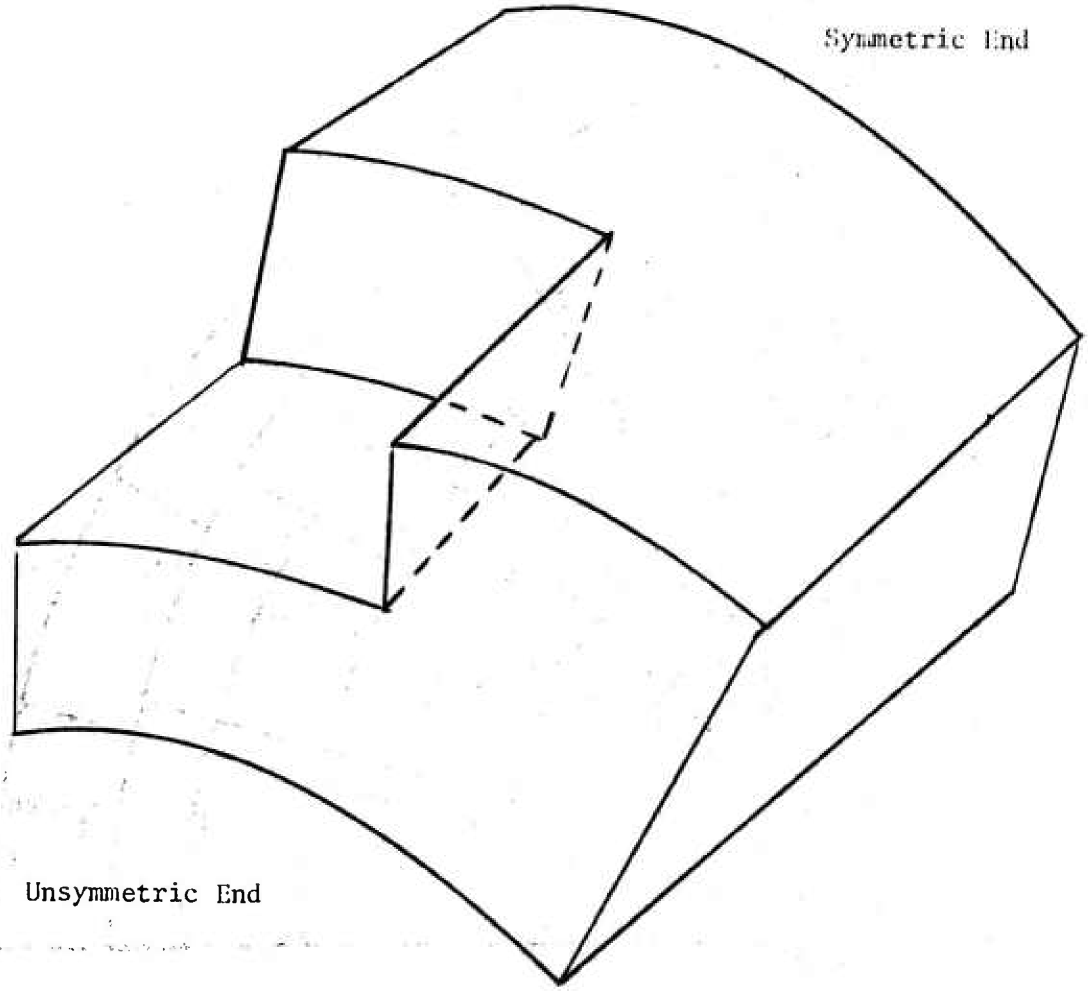


Figure 10. Nonaxisymmetric tube configuration.

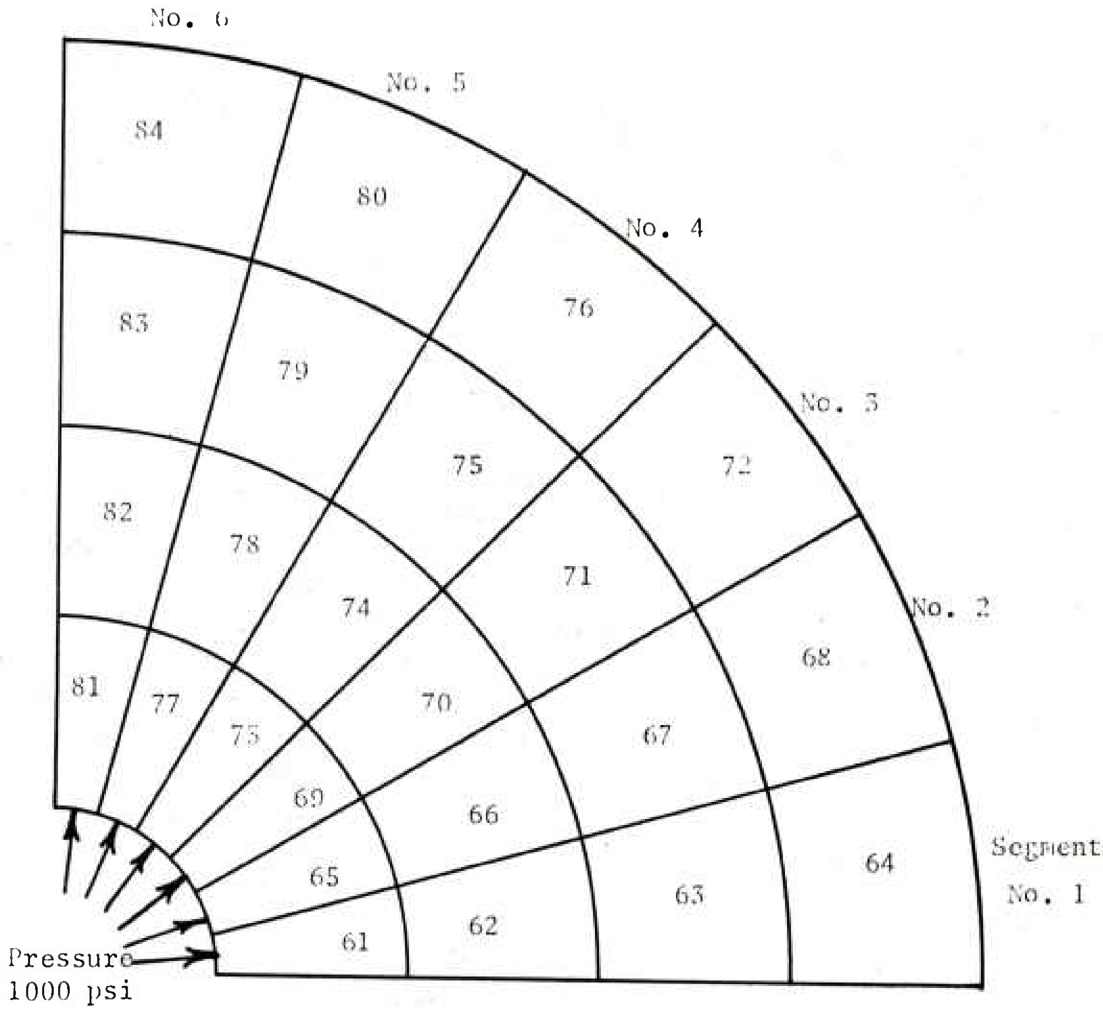


Figure 11. End view of nonaxisymmetric tube configuration showing element and segment numbering systems.

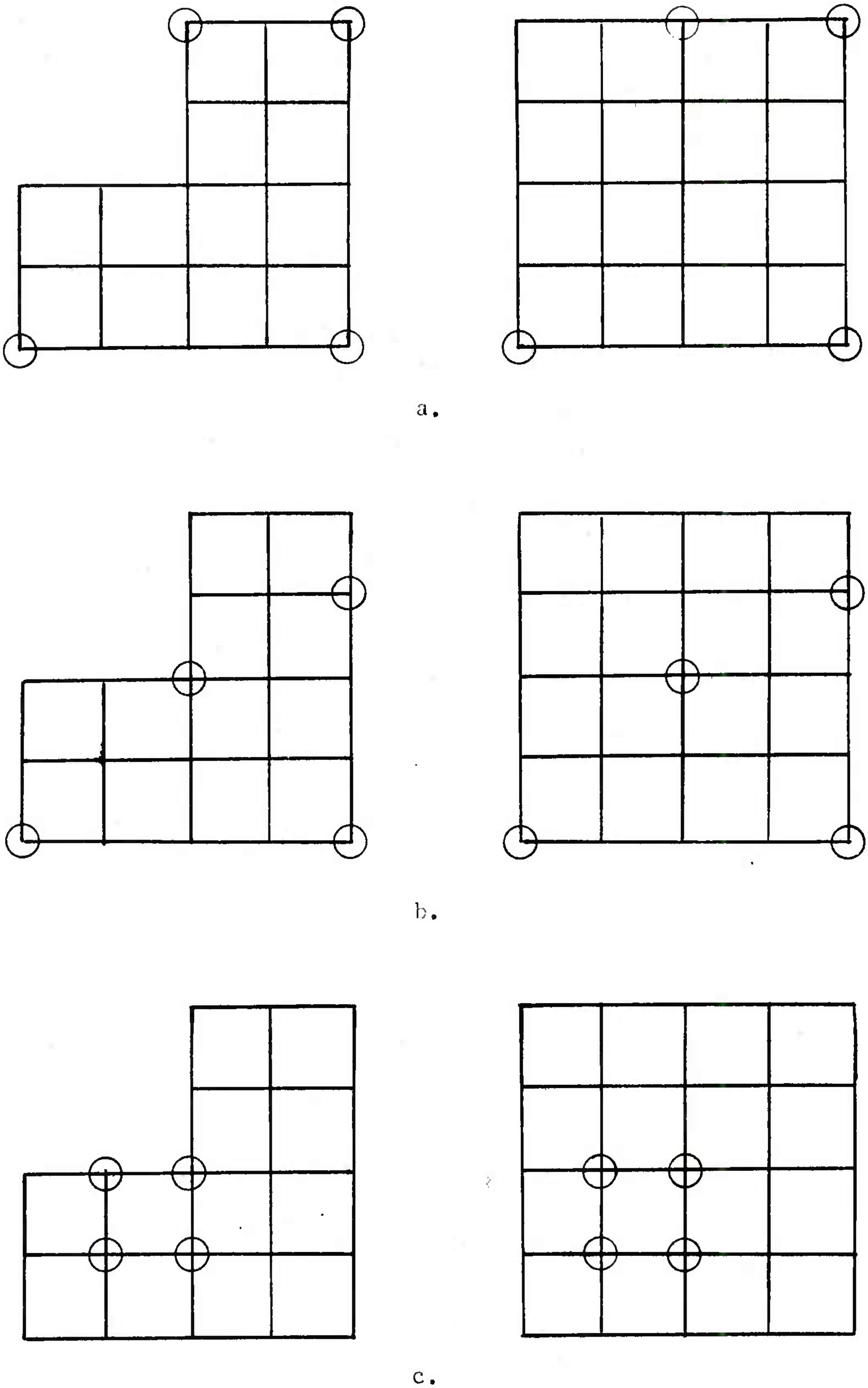


Figure 12. Location of three sets of connecting nodes shown in r-z plane.

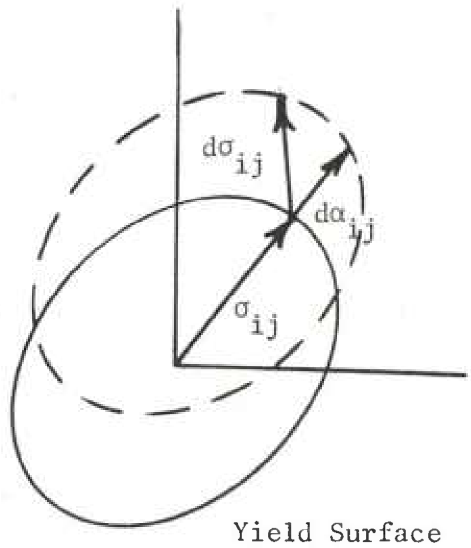


Figure 13. Kinematic or Prager's strain hardening.

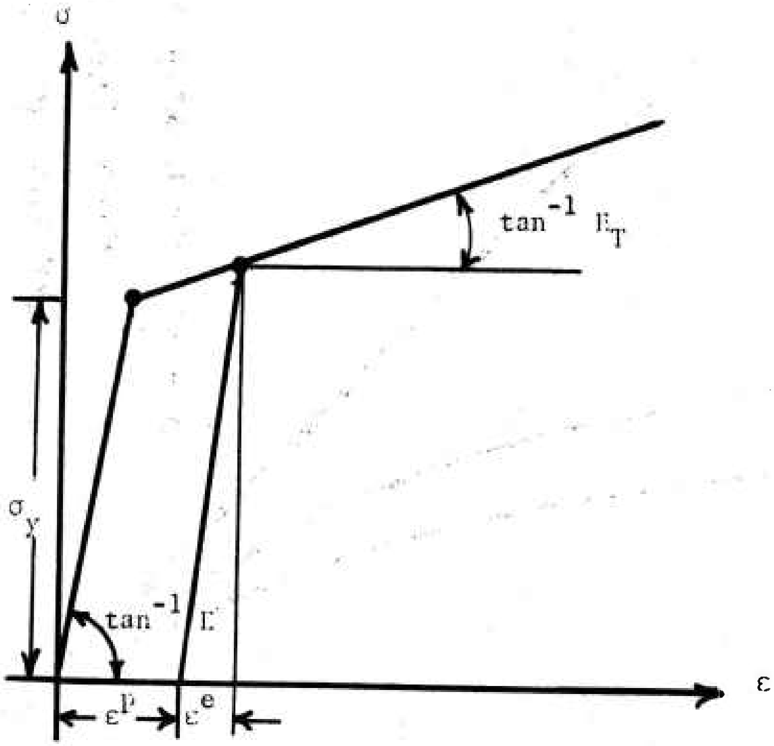


Figure 14. Uniaxial tensile stress-strain curve for a bilinear material.

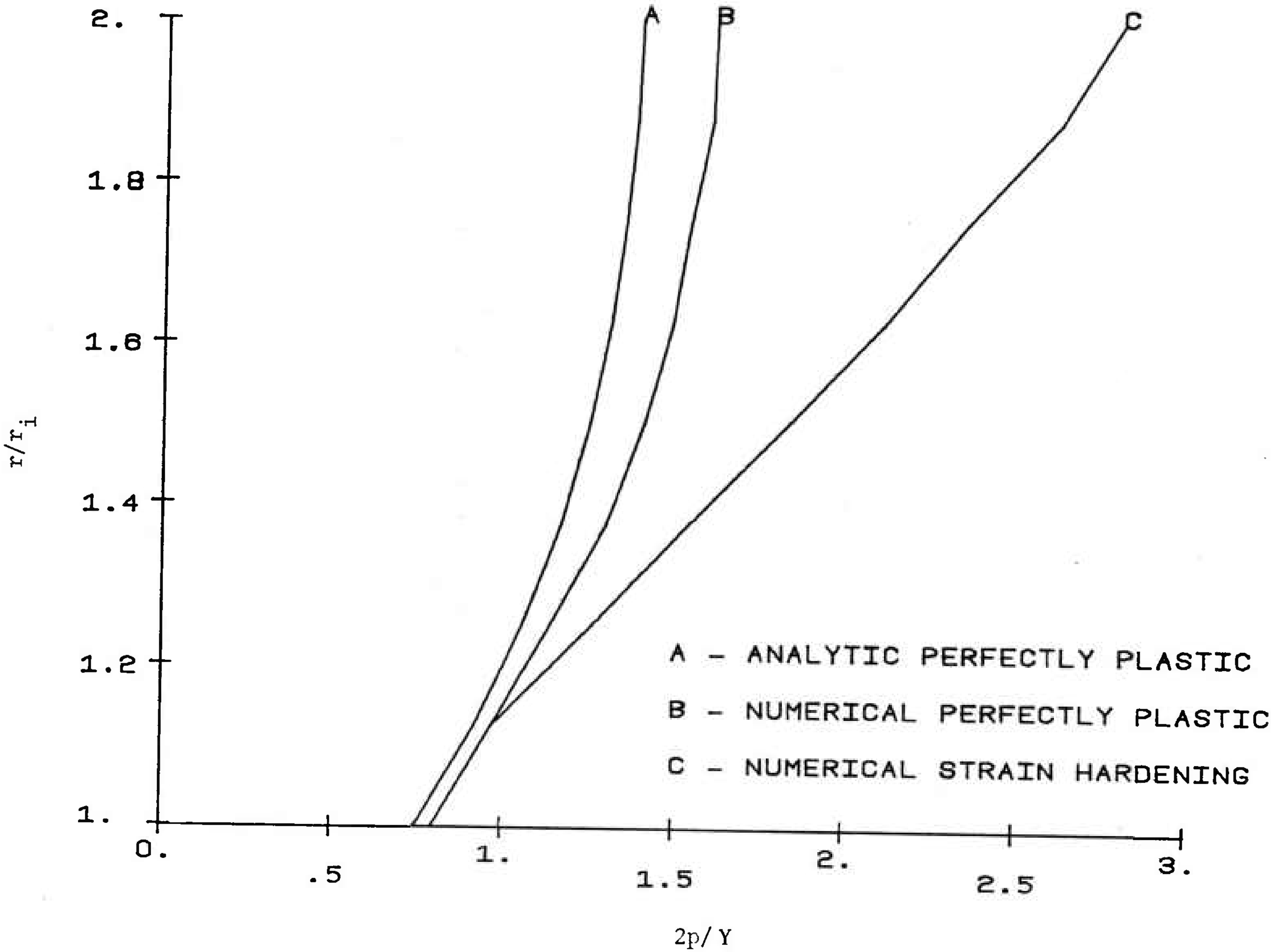


Figure 15. Radial location of elastic-plastic boundary as a function of pressure.

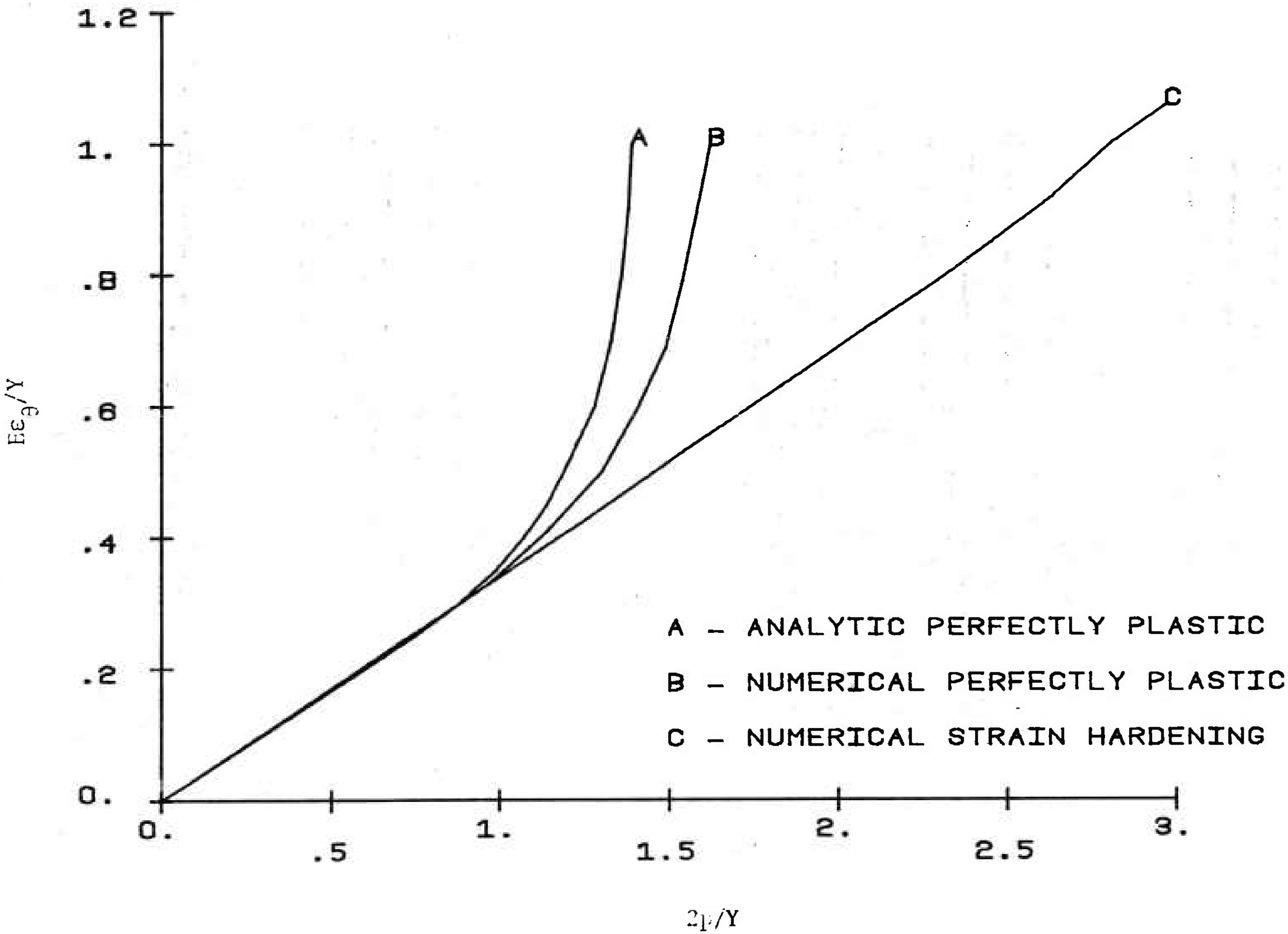


Figure 16. Circumferential strain at outer boundary as a function of pressure.

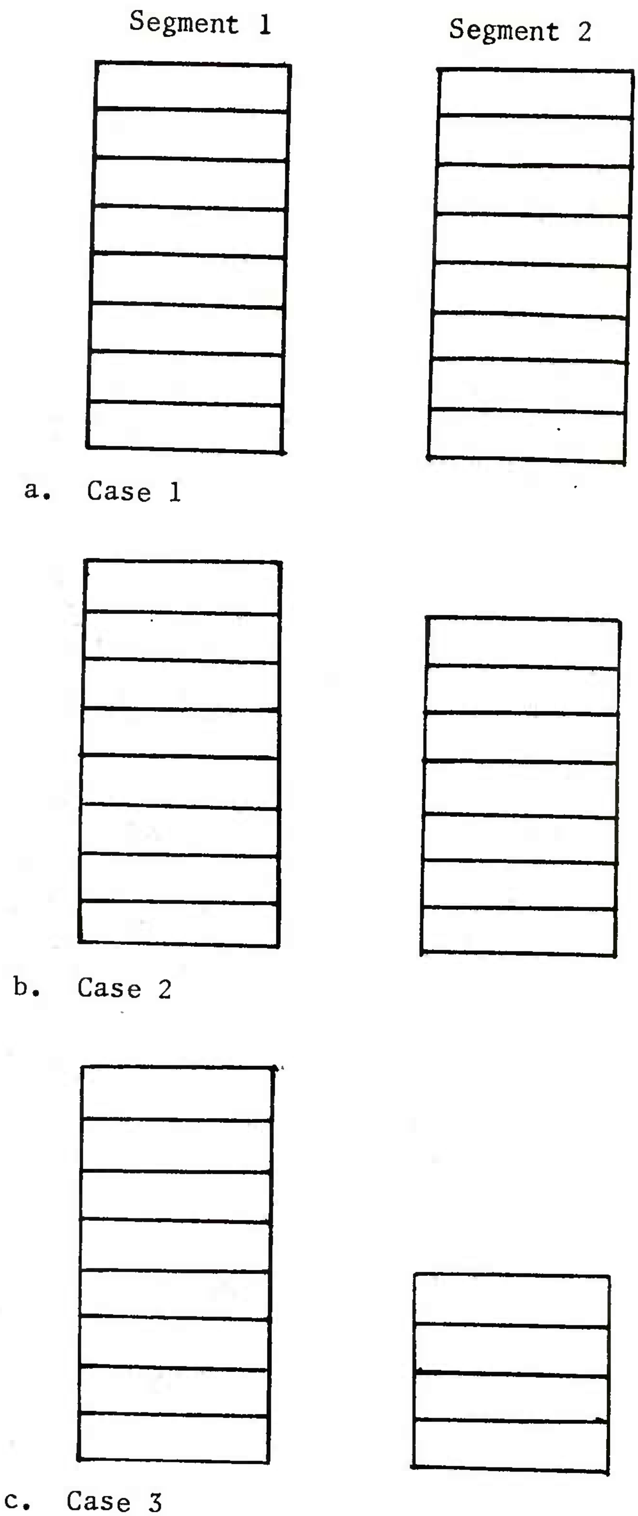


Figure 17. Configurations analyzed shown in r-z plane.

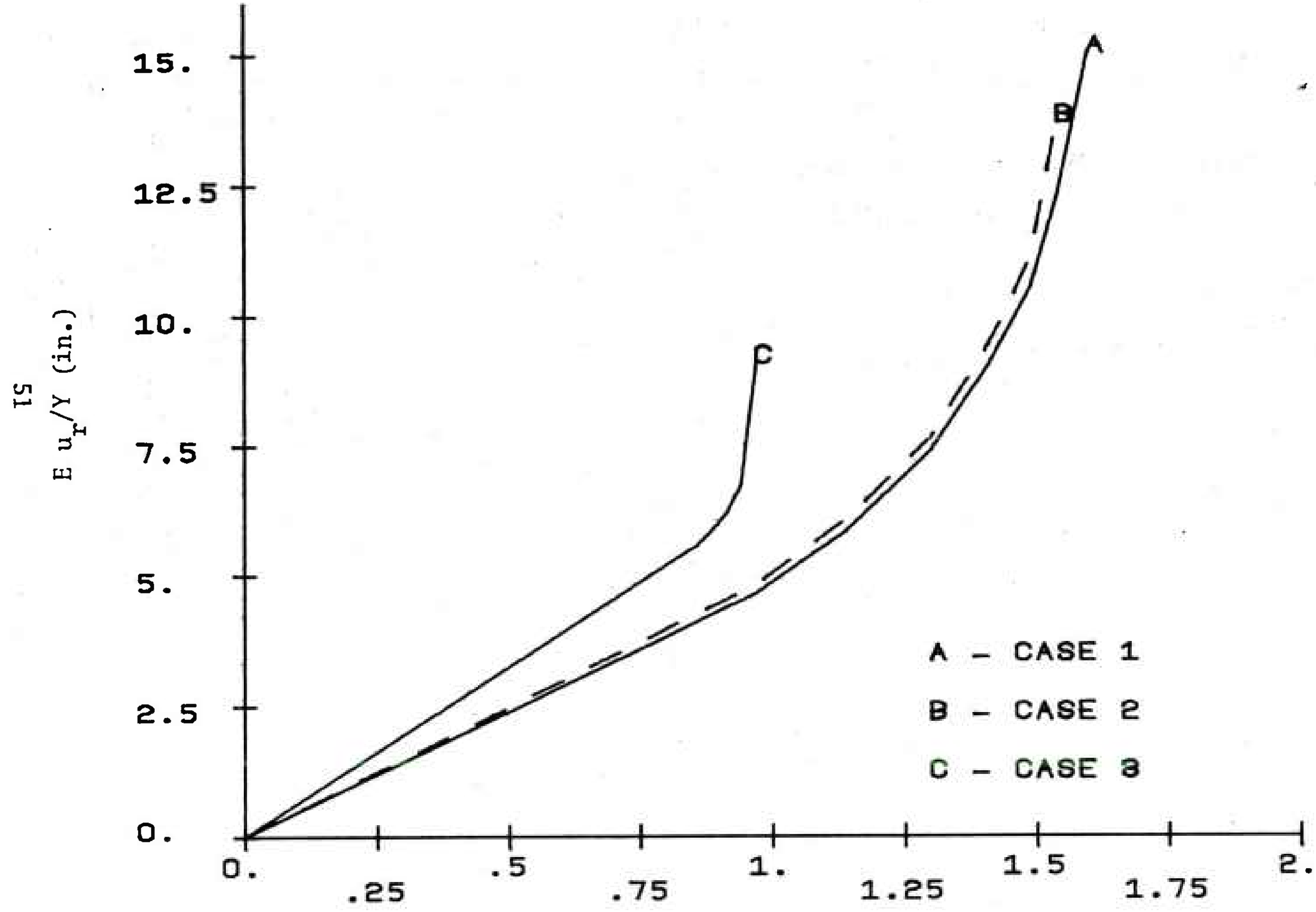


Figure 18. Radial displacement at inner boundary as a function of pressure for perfectly plastic materials.

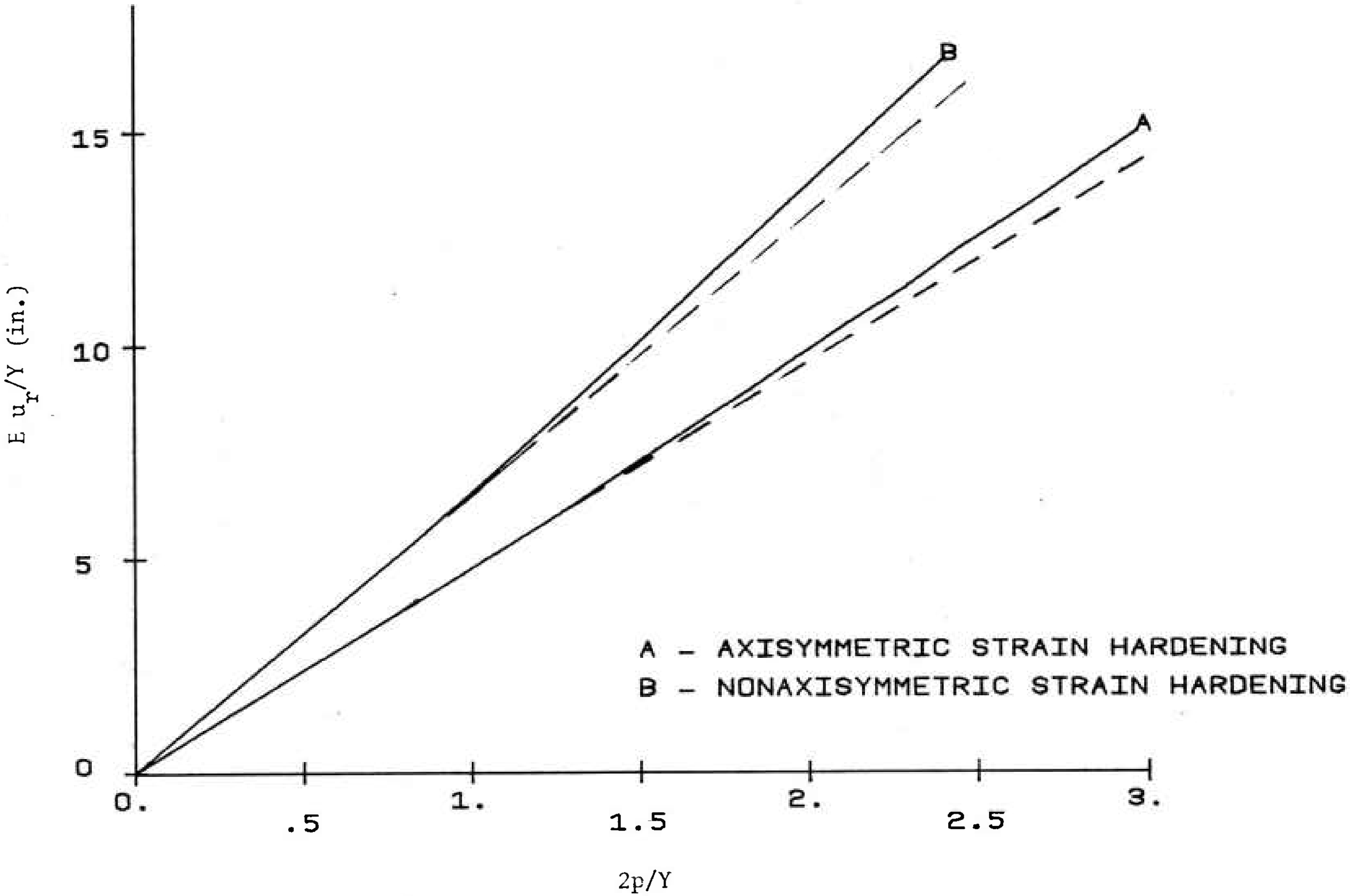


Figure 19. Radial displacement at inner boundary as a function of pressure for strain-hardening materials.

TABLE 1

Stresses from various methods of analysis

for axisymmetric disk under internal pressure of 1000 psi

Radial Stress, σ_{rr}

Element No.	Axisymmetric	Perturbation	Substructure
1	-598	-610	- 39
2	-242	-243	-149
3	- 97	- 94	-199
4	- 24	- 18	-222

Circumferential Stress, $\sigma_{\theta\theta}$

Element No.	Axisymmetric	Perturbation	Substructure
1	857	853	908
2	491	487	527
3	345	345	327
4	271	274	209

TABLE 2

Circumferential stress $\sigma_{\theta\theta}$ from various methods of analysis of a disk with internal pressure of 1000 psi

Element No.	Perturbation	Plane Stress	Substructure	
1	1164	1216	1193	
2	505	540	544	Segment 1
3	208	215	202	
4	46	20	1	
17	1078	1133	1159	Segment 6
18	906	864	829	

TABLE 3

Radial stress σ_{rr} for nonaxisymmetric disk under internal pressure of 1000 psi

Element No.	Perturbation	Plane Stress	Substructure	
1	-446	-519	109	
2	-164	-150	- 79	
3	- 53	- 30	-164	Segment 1
4	6	0	-203	
17	-608	-541	-239	
18	- 92	-116	-325	Segment 6

TABLE 4

Circumferential stress $\sigma_{\theta\theta}$ from perturbation and substructure analysis for nonaxisymmetric tube under internal pressure of 1000 psi

Element No.	Perturbation	Substructure	Limit Solution
1	1010	1061	1164
2	503	544	505
3	284	268	208
4	108	102	46
17	1155	1217	1078
18	811	760	906
61	934	986	857
62	486	528	491
63	301	286	345
64	205	141	271
81	755	956	857
82	479	594	491
83	386	402	345
84	345	288	271

TABLE 5

Radial stress σ_{rr} from perturbation and substructure analysis of nonaxisymmetric tube under internal pressure 1000 psi

Element No.	Perturbation	Substructure	Limit Solution
1	-523	+ 51	-446
2	-200	-104	-164
3	- 71	-178	- 53
4	- 6	-216	+ 6
17	-574	+102	-608
18	-112	+ 6	- 92
61	-570	+ 5	-599
62	-228	-130	-243
63	- 88	-193	- 97
64	- 16	-223	- 24
81	-659	-106	-599
82	-266	-212	-243
83	- 98	-262	- 97
84	- 17	-286	- 29

TABLE 6

Circumferential stress $\sigma_{\theta\theta}$ for nonaxisymmetric tube under internal pressure of 1000 psi with various choices of connecting nodes

Element	Set a	Set b	Set c
1	1010	1011	1011
2	503	503	503
3	284	284	284
4	168	167	168
17	1155	1155	1156
18	811	810	809
61	934	933	934
62	486	486	486
63	301	301	300
64	205	206	205
81	755	756	756
82	479	479	479
83	386	386	385
84	345	345	344

APPENDIX A

DEVELOPMENT OF MATRICES USED IN ANALYSES

A.1 Derivation of Matrix [B]

The [B] matrix introduced in equation (2.3) is obtained by combining equations (2.1) and (2.2). Writing equation (2.1) in matrix form for the nodal displacements of a triangular element gives:

$$\begin{Bmatrix} u_i \\ v_i \\ w_i \end{Bmatrix} = \begin{bmatrix} RZ_i & 0 & 0 \\ 0 & RZ_i & 0 \\ 0 & 0 & RZ_i \end{bmatrix} \begin{Bmatrix} a_j \end{Bmatrix} \tag{A.1}$$

where RZ_i is a row matrix $[1, r_i, z_i]$ and i and j vary from 1 to 3 and from 1 to 9 respectively. The parameters (r_i, z_i) are the nodal coordinates of the triangular element. Equation (A.1) can be expressed as:

$$\{\delta\} = [C]\{\alpha\} \tag{A.2}$$

where $\{\delta\}$ is the nodal displacement, $\{\alpha\}$ is the generalized coordinates and [C] is defined from equation (A.1). Solving equation (A.2) for $\{\alpha\}$ gives:

$$\{\alpha\} = [C]^{-1}\{\delta\} \tag{A.3}$$

Then substituting equation (2.1) into equation (2.2) gives:

$$\begin{Bmatrix} \varepsilon_r \\ \varepsilon_z \\ \varepsilon_\theta \\ \gamma_{rz} \\ \gamma_{z\theta} \\ \gamma_{\theta r} \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \frac{1}{r} & 1 & \frac{z}{r} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{r} & 0 & -\frac{z}{r} \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \end{Bmatrix} \tag{A.4}$$

Equation (A.4) can be expressed as:

$$\{\varepsilon\} = [Q]\ \{\alpha\} \tag{A.5}$$

where [Q] is defined from equation (A.4). Substituting for $\{\alpha\}$ from equation (A.3) gives:

$$\{\varepsilon\} = [Q]\ [C]^{-1}\ \{\delta\} \tag{A.6}$$

Making the substitution:

$$[B] = [Q]\ [C]^{-1} \tag{A.7}$$

gives the result:

$$\{\varepsilon\} = [B]\ \{\delta\} \tag{A.8}$$

which is identical to equation (2.3).

A.2 Derivation of Matrix $[B_1]$

In forming the nonaxisymmetric stiffness matrix the $[B_1]$ matrix was introduced in equation (2.13). This matrix is found in a manner similar to the [B] matrix explained in the previous section. Equation (2.10) can be expanded as:

$$\begin{Bmatrix} u_i \\ v_i \\ w_i \end{Bmatrix} = \begin{bmatrix} C & 0 \\ C & \theta C \end{bmatrix} \{a_j\} \tag{A.9}$$

where matrix C is defined in equation (A.2), θ is a multiplying constant, i varies over integers 1 to 6, and j varies over integers 1 to 18. We define θ = 0 on one face of the segment and $\Theta \equiv \theta$ on the other face. Equation (A.9) can be written in partitioned form as:

$$\begin{Bmatrix} \delta_1 \\ \delta_2 \end{Bmatrix} = \begin{bmatrix} C & 0 \\ C & \theta C \end{bmatrix} \begin{Bmatrix} \alpha \\ \beta \end{Bmatrix} = [C_1] \begin{Bmatrix} \alpha \\ \beta \end{Bmatrix} \tag{A.10}$$

where [C] is identical to the [C] in equation (A.2) and $\{{}^{\alpha}_{\beta}\}$ are the 18 generalized coordinates. Solving for $\{{}^{\alpha}_{\beta}\}$ gives:

$$[C_1]^{-1} \begin{Bmatrix} \delta_1 \\ \delta_2 \end{Bmatrix} = \begin{Bmatrix} \alpha \\ \beta \end{Bmatrix} \tag{A.11}$$

where:

$$[C_1]^{-1} = \begin{bmatrix} C^{-1} & 0 \\ -\frac{1}{\theta} C^{-1} & \frac{1}{\theta} C^{-1} \end{bmatrix} \tag{A.12}$$

Substituting equation (2.10) into (2.11) gives:

$$\begin{Bmatrix} \varepsilon_r \\ \varepsilon_z \\ \varepsilon_\theta \\ \gamma_{rz} \\ \gamma_{z\theta} \\ \gamma_{r\theta} \end{Bmatrix} = \begin{bmatrix} 0,\ 1,\ 8x0,\ \theta,\ 7x0 \\ 5x0,\ 1,\ 8x0,\ \theta,\ 3x0 \\ \frac{1}{r},\ 1,\ \frac{z}{r},\ 6x0,\ \frac{\theta}{r},\ \theta,\ \frac{z\theta}{r},\ 3x0,\ \frac{1}{r},\ 1,\ \frac{z}{r} \\ 2x0,\ 1,\ 0,\ 1,\ 6x0,\ \theta,\ 0,\ \theta,\ 4x0 \\ 8x0,\ 1,\ 3x0,\ \frac{1}{r},\ 1,\ \frac{z}{r},\ 2x0,\ \theta \\ 6x0,\ \frac{-1}{r},\ 0,\ \frac{-z}{r},\ \frac{1}{r},\ 1,\ \frac{z}{r},\ 3x0,\ \frac{-\theta}{r},\ 0,\ \frac{-z\theta}{r} \end{bmatrix} \begin{Bmatrix} a_j \end{Bmatrix} \tag{A.13}$$

where j varies over integers 1 to 18 and the rectangular matrix has been written in a compressed form where nx0 indicates n consecutive elements which are identically equal to zero. Equation (A.13) can be rewritten as:

$$\{\varepsilon\} = [Q_1] \{{}^{\alpha}_{\beta}\} \tag{A.14}$$

Combining equations (A.11) and (A.13) gives:

$$\{\varepsilon\} = [Q_1] [C_1]^{-1} \{\delta\} \tag{A.15}$$

or

$$\{\varepsilon\} = [B_1] \{\delta\} \tag{A.16}$$

where

$$[B_1] = [Q_1]\,[C_1]^{-1} \qquad (A.17)$$

A.3 Description of [D] Matrix

The [D] matrix is defined as:

$$\{\sigma\} = [D]\,\{\varepsilon\} \qquad (A.18)$$

The inverse of equation (A.18) is:

$$\{\varepsilon\} = [D]^{-1}\,\{\sigma\} \qquad (A.19)$$

The $[D]^{-1}$ matrix is defined in equation (2.4). It can be expressed as:

$$[D]^{-1} = \begin{bmatrix} \frac{1}{E_n} & -\frac{\nu_{ns}}{E_n} & -\frac{\nu_{nt}}{E_n} & 0 & 0 & 0 \\ -\frac{\nu_{sn}}{E_s} & \frac{1}{E_s} & -\frac{\nu_{st}}{E_s} & 0 & 0 & 0 \\ -\frac{\nu_{tn}}{E_t} & -\frac{\nu_{ts}}{E_t} & \frac{1}{E_t} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{ns}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{st}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{tn}} \end{bmatrix} \qquad (A.20)$$

The [D] matrix is obtained by inverting equation (A.20).

A.4 Derivation of the Matrices [R] and [G]

If we number the connecting nodes of a segment from 1 to 8 where the nodes 1-4 are on one face and the nodes 5-8 on the other face, the [R] matrix in the equation

$$\{u_{cp}\} = [R]\{p\} \tag{A.21}$$

can be partitioned as:

$$[R] = \left[\begin{array}{c|c} R_1 & 0 \\ \hline 0 & R_2 \end{array}\right] \tag{A.22}$$

where $[R_1]$ has the form:

$$[R_1] = \begin{bmatrix} RR_1 \\ RR_2 \\ RR_3 \\ RR_4 \end{bmatrix} \tag{A.23}$$

where matrices RR_i are defined as follows:

$$[RR_i] = \begin{bmatrix} Rzz_i & 0 & 0 \\ 0 & Rzz_i & Rzz_i \end{bmatrix} \tag{A.24}$$

where Rzz_i is a row matrix $[1, r_i, z_i, r_i z_i]$.

The matrix $[R_2]$ has the same form with r_i replaced by r_{i+4} and z_i replaced by z_{i+4} (i=1,4). Also for each element we have:

$$\{u_p\} = [G]\{p\} \tag{A.25}$$

where [G] can be partitioned as:

$$[G] = \left[\begin{array}{c|c} G_1 & 0 \\ \hline 0 & G_1 \end{array}\right] \tag{A.26}$$

In this case, since the nodes on opposite faces of the segment correspond, their sub-matrices are identical. The matrix G_1 has the same form as R_1 in equation (A.23) except the coordinates apply to the coordinates of the four corners of the quadrilateral element rather than the four connecting points.

A.5 Derivation of the [A] Matrix

The concentrated loads such as pressure loadings are input into the program as occurring between two nodes, i and j. The perturbation displacements at these nodes are related to the perturbation displacements at the connecting nodes by:

$$\{u_{pi}\} = [A]\ \{p\} \tag{A.27}$$

Substituting for {p} from equation (2.17) gives:

$$\{u_{pi}\} = [A]\ [R]^{-1}\ \{u_{cp}\} \tag{A.28}$$

where $\{u_{pi}\}$ represents the perturbation displacements of node "i". A similar expression is then written for node "j".

If {g} is a matrix of applied loads at node "i" then:

$$\{g\} = \begin{Bmatrix} g_r \\ g_z \\ g_\theta \\ g_r \\ g_z \\ g_\theta \end{Bmatrix} \tag{A.29}$$

where g_r, g_z and g_θ are the components of the load in the three directions. These components are repeated to indicate that the load is applied equally to each face of the segment.

The [A] matrix in equation (A.26) can be written in partitioned form as:

$$[A] = \left[\begin{array}{c|c} A^* & 0 \\ \hline 0 & A^* \end{array}\right] \tag{A.30}$$

where [A*] is:

$$[A^*] = [RR_i] \tag{A.31}$$

where $[RR_i]$ was defined in equation (A.24) and where coordinates r_i, z_i refer to points where the load was applied.

Then balancing virtual work gives:

$$\delta u_p^T \{fc\} = \Sigma \, \delta \, u_{pi}^T \{g\} \tag{A.32}$$

but

$$\delta u_{pi}^T = ([A] \, [R]^{-1} \, \delta \, u_{cp})^T \tag{A.33}$$

or

$$\delta u_{pi}^T = \delta \, u_{cp}^T \, [R]^{-1^T} \, [A]^T \tag{A.34}$$

Then equation (A.30) becomes:

$$\delta u_p^T \{fc\} = \delta \, u_{cp}^T \, \Sigma \, [R]^{-1^T} \, [A]^T \{g\} \tag{A.35}$$

where the summation is over all nodes with an external load applied.

APPENDIX B

INPUT CARDS FOR COMPUTER PROGRAMS

1. SEGMENT CONTROL CARD (one for each version)

Elastic Version:

Format (2I10)

Columns	1-10	NTYPS (Number of different types of segments; 4 maximum)
	11-20	NTOTS (Number of total segments; 8 maximum)

Elastic-Plastic Version:

Format (2I10)

Columns	1-10	NTOTS (Number of segments, 8 maximum)
	11-20	NOLINC (Number of load increments)

2. SEGMENT DATA CARDS

One card for each type of segment.

Format (F10.5, I10)

Columns	1-10	THETA (Angle subtended by segment)
	11-20	NST (The number of segments of each type; 5 maximum; not needed in elastic-plastic version)

3. SEGMENT NUMBERING CARDS (not needed in elastic-plastic version)

One card for each type of segment.

Format (5I10)

Columns	1-10	NUMS (1)	(Reference numbers for segments of each type in global numbering system; there can be one to five numbers per card depending on how many segments there are of each type.)
	:	:	
	41-50	NUMS (5)	

4. <u>CONNECTING NODES CARDS</u>

One card for each segment. These cards must be in order according to the global numbering system for segments.

Format (8I10)

Columns	1-10	NPC (1)	(Nodal number for connecting nodes according to the axisymmetric grid for the segment).
	:	:	
	71-80	NPC (8)	

Cards 5-18 must be repeated for each different type of segment. These are the control cards for the axisymmetric solution.

5. <u>TITLE CARD</u>

Format (20A4)

Columns 1-80 TITLE (Title for particular case)

6. <u>CONTROL CARD</u>

Format (6I5, F5.0, 5I5)

Columns	1-5	NNLA (Number of nonlinear approximations; NNLA = 1 for this version of the program)
	6-10	NUMTC (Number of temperature cards; if -2, a constant temperature is specified)
	11-15	NUMMAT (Number of different materials; 6 maximum
	16-20	NUMPC (Number of boundary pressure cards; 200 maximum)

21-25	NUMSC (Number of boundary shear cards; 200 maximum)
26-30	NUMST (Number of boundary shear cards in tangential direction; 200 maximum)
31-35	TREF (Reference temperature)
36-40	INERT (This parameter decides if inertia loads will be present, INERT + 0 means zero values of axial acceleration, and angular acceleration and velocity for each load increment)
51-55	INCF (If INCF = 0, then surface loads for each time increment will be the same as for first increment)
56-60	IPLOT (Plot parameter, 1 if plot required)

7. MESH GENERATION CONTROL CARD

Format (5I5)

Columns		
	1-5	MAXI (Maximum value of I in mesh; 25 maximum)
	6-10	MAXJ (Maximum value of J in mesh; 100 maximum)
	11-15	NSEG (Number of line segment cards)
	16-20	NBC (Number of boundary condition cards)
	21-25	NMTL (Number of material block cards)

8. LINE SEGMENT CARDS

The order of line segment cards is immaterial except when plots are requested; in this case, the line segment cards must define the perimeter of the solid continuously. The order of line segment cards defining internal straight lines is always irrelevant.

4	Circular arc specified by 1st and 2nd points at the ends of the arc with the coordinates of the center of the arc given as the 3rd point (delete I and J for 3rd point)
5	Straight line as boundary diagonal for which I of ist point is minimum for its row and/or I of 2nd point is minimum for its row (input only 1st and 2nd points)
6	Straight line as boundary diagonal for which I of 1st point and/or 2nd point is maximum for its row (input only 1st and 2nd points)

NOTE: In specifying a circular arc, the points are ordered such that a counterclockwise direction about the center is obtained upon moving along the boundary.

9. BOUNDARY CONDITION CARDS

Each card assigns a particular boundary condition to a block of elements bounded by I1, I2, J1, J2. For a line I1 = I2 or J1 = J2. For a point I1 = I2 and J1 = J2.

Format (4I5, I10, 3F10.0)

Columns	1-5	Minimum I
	6-10	Maximum I
	11-15	Minimum J
	16-20	Maximum J
	21-30	Boundary condition code
	31-40	Radial boundary condition code, XR
	41-50	Axial boundary condition, XZ
	51-60	Tangential boundary condition XT

Format (3(2I3, 2F8.3), I5)

Columns	1-3	I coordinate of 1st point
	4-6	J coordinate of 1st point
	7-14	R coordinate of 1st point
	15-22	Z coordinate of 1st point
	23-25	I coordinate of 2nd point
	26-28	J coordinate of 2nd point
	29-36	R coordinate of 2nd point
	37-44	Z coordinate of 2nd point
	45-47	I coordinate of 3rd point
	48-50	J coordinate of 3rd point
	51-58	R coordinate of 3rd point
	59-66	Z coordinate of 3rd point
	67-71	Line segment type parameter

If the number in column 71 is

0	Point (input only 1st point
1	Straight line (input only 1st and 2nd points)
2	Straight line as an internal diagonal (input only 1st and 2nd points)
3	Circular arc specified by 1st and 3rd points at the ends of the arc and 2nd point at the mid-point of the arc

If the number in Columns 21-30 is

0	XR is the specified R-load and XZ is the specified Z-load and XT is the specified θ-load
1	XR is the specified R-displacement and XZ is the specified Z-load and XT is the specified θ-load
2	XR is the specified R-load and XZ is the specified Z-displacement and XT is the specified θ-load
3	XR is the specified R-displacement and XZ is the specified Z-displacement and XT is the specified θ-load
4	XR is the specified R-load and XZ is the specified Z-load and XT is the specified θ-displacement
5	XR is the specified R-displacement and XZ is the specified Z-load and XT is the specified θ-displacement
6	XR is the specified R-load and XZ is the specified Z-displacement and XT is the specified θ-displacement
7	XR is the specified R-displacement and XZ is the specified Z-displacement and XT is the specified θ-displacement

NOTE: All loads are considered to be total forces acting on one radian segment.

10. MATERIAL BLOCK ASSIGNMENT CARD

Each card assigns a material definition number to a block of elements defined by the I, J coordinates.

Format (5I5, 2F10.0, 2I5)

Columns	1-5	Material definition number (1 through 6)
	6-10	Minimum I
	11-15	Maximum I
	16-20	Minimum J
	21-25	Maximum J
	26-35	Material principal property inclination angle BETA which defines N-S plane orientation relative to z direction (see Figure 4)
	36-45	Material principal property inclination angle ALPHA which defines the orientation of N-T plane relative to r-z plane (see Figure 4)
	46-50	IANG (If IANG = 0, then ALPHA is same for total material block. If IANG = 1, the ALPHA varies in sign in the I direction from element to element every NANG elements. This will allow for equal but opposite helical angles.)
	51-55	NANG (Number of elements in the I direction with the same ALPHA).

11. PLOT TITLE CARD*

Format (20A4)

Columns	1-80	Title (Title printed under each plot)

12. PLOT GENERATION INFORMATION CARD*

Format (2F10.0)

Columns	1-10	RMAX (Maximum r coordinate of mesh)
	11-20	ZMAX (Maximum z coordinate of mesh)

*NOTE: Use only if IPLOT = 1 (plot required)

13. TEMPERATURE FIELD INFORMATION CARDS

If NUMTC in columns 6-10 of the CONTROL CARD is greater than 1, the temperature field is given on cards. One card must be supplied for each point for which a temperature is specified.

Format (3F10.0)

Columns	1-10	R coordinate
	11-20	z coordinate
	21-30	Temperature

If NUMTC in columns 6-10 of the CONTROL CARD is -2, a constant temperature field is specified; the value is given on a single card.

Format (F10.0)

Columns 1-10 Temperature

14. MATERIAL PROPERTY INFORMATION CARDS

The following group of cards must be specified for each material (maximum of 6).

a. MATERIAL IDENTIFICATION CARD

Format (2I5, 2F10.0)

Columns	1-5	Material Identification number
	6-10	Number of temperatures for which

properties are given (12 maximum)

11-20	Mass density of material (if required)
21-30	Thermal expansion parameter (If 1, free thermal expansions on the material property cards; otherwise, coefficients of thermal expansion are on the material property cards.)

b. MATERIAL PROPERTY CARDS

Two cards are required for each temperature.

First Card

Format (7F10.0)

Columns	1-10	Temperature
	11-20	Modulus of elasticity, E_N
	21-30	Modulus of elasticity, E_S
	31-40	Modulus of elasticity, E_T
	41-50	Poisson's ratio, ν_{NS}
	51-60	Poisson's ratio, ν_{NT}
	61-70	Poisson's ratio, ν_{ST}

Second Card

Format (6F10.0)

Columns	1-10	Shear Modulus, G_{NS}
	11-20	Shear Modulus, G_{ST}
	21-30	Shear Modulus, G_{TN}
	31-40	$\alpha_N T$ or α_N
	41-50	$\alpha_S T$ or α_S
	51-60	$\alpha_T T$ or α_T

15. YIELD STRESS CARDS (not needed in elastic version)

Format (7F10.0)

Columns		
	1-10	Yield stress in tension in N direction
	11-20	Yield stress in tension in S direction
	21-30	Yield stress in tension in T direction
	31-40	Yield stress in shear in NS direction
	41-50	Yield stress in shear in NT direction
	51-60	Yield stress in shear in TS direction
	61-70	Hardening parameter - C

16. INERTIA LOAD CARD

Format (3F10.0)

Starting with this input card and including the boundary force cards, this data is to be inputted as a block for each load step, that is NLINC times. There are the following exceptions to this:

a) If INERT = 0, then this card is to be omitted completely (no inertia load).

b) If INCI = 0, then this card is not repeated, but appears in first block only (the inertia loads are constant for each load step).

c) If INCF = 0, then the following boundary pressure and shear cards are to be given only for the first block and not repeated again (the pressure and shear loads are constant for each load increment).

Columns	1-10	ACELZ (axial acceleration)
	11-20	ANGVEL (angular velocity)
	21-30	ANGACC (angular acceleration)

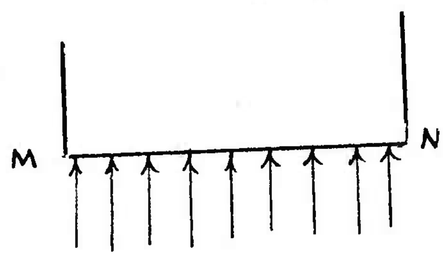
17. BOUNDARY PRESSURE CARDS

One card is required for each boundary element which is subjected to a normal pressure, that is the number of these cards is NUMPC for each load increment.

Format (2I5, F10.0)

Columns	1-5	Nodal point M
	6-10	Nodal point N
	11-20	Normal pressure

As shown in the figure below, the boundary element must be on the left when progressing from M to N. Surface normal tension is input as a negative pressure.



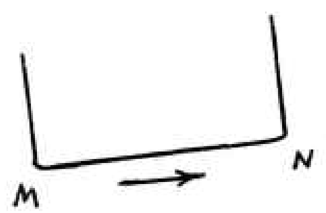
18. BOUNDARY SHEAR CARDS

One card is required for each boundary element which is subjected to surface shear, that is, the number of these cards is NUMSC for each load increment.

Format (2I5, F10.0)

Columns	1-5	Nodal point M
	6-10	Nodal point N
	11-20	Surface shear

As shown in the figure below, the boundary element must be on the left when progressing from M to N. The positive sense of the shear is from M to N.

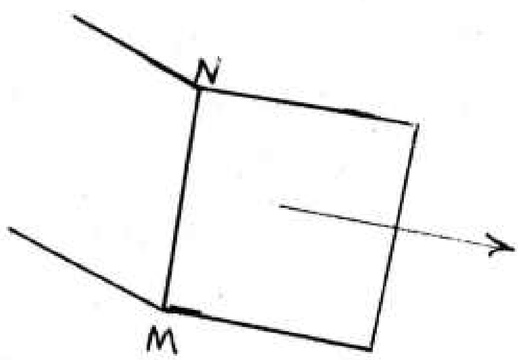


19. BOUNDARY TRANSVERSE SHEAR CARDS

One card is required for each boundary element which is subject to transverse shear, that is, the number of these cards is NUMSC for each load increment.

Format (2I5, F10.0)

Columns	1-5	Nodal point M
	6-10	Nodal point N
	11-20	Surface transverse shear



20. BOUNDARY CONDITION CONTROL CARD

Format (I5)

Columns 1-5 NRDF (This parameter is equal to the total number of displacement components specified at connecting points, for example, if one displacement component was specified at each connecting point then NRDF would be equal to the number of connecting points)

21. BOUNDARY CONDITION CARDS

There are NRDF of these cards.

Format (I10, F10.0)

Columns 1-10 NREQ (The location of the equation to be modified in the assembled matrix relative to the connecting points. For example, if there are 12 connecting points and at the fifth connecting point the second component of displacement is specified then this integer would be equal to 3 x (5-1) + 2 = 14)

11-20 U (The actual boundary condition valve to be specified in position NREQ in the matrix equations)

APPENDIX C

PROGRAM LISTING

```
      PROGRAM NONAXI(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,
     1 TAPE2,TAPE3,TAPE22,TAPE23,TAPE24,
     2TAPE21,TAPE25,TAPE26)
      INTEGER CODE
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
     1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200)
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/NXMESH/THETAN(4),NST(4),NUMS(4,5),NPC(8,8)
      COMMON/ANS1/NUMELS(4),NUMNPS(4)
      COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS,GTS1G(24,24,4)
      COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
      COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/CONVRG/IDONE
      COMMON/PLANE/NPP
      COMMON/RESULT/BS(6,15),D(6,6),C(6,6),AR,BB(6,9),CNS(6,6)
      COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
      DIMENSION TITLE(20)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE CONTROL INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      READ(5,3000) NTYPS,NTOTS
      DO 150 I=1,NTYPS
  150 READ(5,3001) THETAN(I),NST(I)
       DO 151 I=1,NTYPS
      J2 =NST(I)
  151 READ(5,3000) (NUMS(I,J),J=1,J2)
      DO 152  I= 1,NTOTS
  152 READ(5,3002) (NPC(I,J),J=1,8)
 3000 FORMAT(8I10)
 3001 FORMAT(F10.5,I10)
 3002 FORMAT(8I10)
       DO 60 I=21,26
   60  REWIND I
       WRITE(6,3010)
 3010 FORMAT("1","SEGMENT DATA FOR NONAXISYMMETRIC PROBLEM")
       WRITE(6,3011) NTYPS,NTOTS
 3011 FORMAT (" "," NUMBER OF TYPES OF SEGMENTS = ",I5,//,
     1             " NUMBER OF TOTAL SEGMENTS     =",I5)
       DO 153 I=1,NTYPS
      WRITE(6,3012) I,THETAN(I),NST(I)
 3012 FORMAT(" ",///," SEGMENT TYPE = ",I5/,"      THETA = ",F10.5/,
     1  "  NUMBER OF SEGMENTS OF THIS TYPE = ",I5)
       J2 = NST(I)
```

```
      WRITE(6,3013)(NUMS(I,J),J=1,J2)
 3013 FORMAT(" ","  SEGMENT NUMBERS IN GLOBAL SYSTEM ARE ",5I5)
  153 CONTINUE
       DO 154  I=1,NTOTS
  154  WRITE(6,3014)I,(NPC(I,J),J=1,8)
 3014 FORMAT("  ","CONNECTING NODES FOR SEGMENT",I5," ARE",8I5)
      DO 950 NTP = 1,NTYPS
       THETA= THETAN(NTP)      /57.295780
   50 READ(5,1000 )TITLE,NNLA,NUMTC,NUMMAT,NUMPC,NUMSC,NUMST,TREF
     1,INERT,NLINC,INCI,INCF,IPLOT
      WRITE(6,2000)TITLE,NNLA,NUMTC,NUMMAT,NUMPC,NUMSC,NUMST,TREF,INERT,
     1NLINC
      NPP=0
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     GENERATE FINITE ELEMENT MESH
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  100 CALL MESH
       NUMELS(NTP) = NUMEL
       NUMNPS(NTP) = NUMNP
      IF (IPLOT.EQ.1) CALL MPLOT
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE T5MPERATURE DATA
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  103 IF(NUMTC.EQ.0) GO TO 440
      IF(NUMTC.GT.0) READ(5,1001) (X(I),Y(I),TEM(I),I=1,NUMTC)
      IF(NUMTC.EQ.-2) CALL TEM2(NUMNP)
      IF(NUMTC.EQ.-2) GO TO 440
      MPRINT=0
      DO 210 I=1,NUMTC
      IF(MPRINT.NE.0) GO TO 200
      WRITE(6,2001)
      MPRINT=59
  200 MPRINT=MPRINT-1
  210 WRITE(6,2002) X(I),Y(I),TEM(I)
      MPRINT=0
      DO 230 N=1,NUMNP
      IF(MPRINT.NE.0) GO TO 220
      WRITE(6,2003)
      MPRINT=59
  220 MPRINT=MPRINT-1
      CALL TEMP(R(N),Z(N),T(N))
  230 WRITE(6,2004) N,R(N),Z(N),T(N)
  440 MPRINT=0
      DO 460 N=1,NUMEL
      IF(MPRINT.NE.0) GO TO 450
      WRITE(6,2008)
      MPRINT=59
  450 MPRINT=MPRINT-1
      II=IX(N,1)
      JJ=IX(N,2)
```

```
      KK=IX(N,3)
      LL=IX(N,4)
C
C     TEM IS TEMPORARY STORAGE FOR ELEMENT TEMPERATURES
C
      TEM(N)=(T(II)+T(JJ)+T(KK)+T(LL))/4.00
  460 WRITE(6,2009) N,(IX(N,I),I=1,5),BETA(N),ALPHA(N),TEM(N)
      DO 470 K=1,NUMEL
  470 T(K)=TEM(K)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE MATERIAL PROPERTIES
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  500 CONTINUE
      DO 510 M=1,NUMMAT
      READ(5,1004)  MTYPE,(NT,RO(MTYPE),AOFTS(MTYPE))
      WRITE(6,2010) MTYPE,NT,RO(MTYPE)
      READ(5,1005)((E(I,J,MTYPE),J=1,14),I=1,NT)
      IF(AOFTS(MTYPE).NE.1.) WRITE(6,2011)((E(I,J,MTYPE),J=1,13),I=1,NT)
      IF(AOFTS(MTYPE).EQ.1.) WRITE(6,2012)((E(I,J,MTYPE),J=1,13),I=1,NT)
      DO 510 I=NT,12
      DO 510 J=1,16
  510 E(I,J,MTYPE)=E(NT,J,MTYPE)
      DO 900 NL=1,NLINC
      WRITE(6,2030) NL
      ACELZ=0.00
      ANGVEL=0.00
      ANGACC=0.00
      IF(INERT .EQ. 0) GO TO 511
      IF(NL .NE. 1 .AND. INCI .EQ. 0) GO TO 511
C*********************************************************************
C     READ AND WRITE DYNAMIC FORCES
C*********************************************************************
      READ(5,1030) ACELZ, ANGVEL, ANGACC
      WRITE(6,2031) ACELZ, ANGVEL, ANGACC
  511 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND WRITE PRESSURE AND SHEAR BOUNDARY CONDITIONS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(NL .NE. 1 .AND. INCF .EQ. 0) GO TO 700
  600 IF(NUMPC.EQ.0) GO TO 630
      MPRINT=0
      DO 620 L=1,NUMPC
      IF(MPRINT.NE.0) GO TO 610
      WRITE(6,2013)
      MPRINT=58
  610 MPRINT=MPRINT-1
      READ(5,1006)  IP(L),JP(L),PR(L)
  620 WRITE(6,2014) IP(L),JP(L),PR(L)
  630 IF(NUMSC.EQ.0) GO TO 701
      MPRINT=0
```

```
      DO 650 L=1,NUMSC
      IF(MPRINT.NE.0) GO TO 640
      WRITE(6,2015)
      MPRINT=58
  640 MPRINT=MPRINT-1
      READ(5,1006)  IS(L),JS(L),SH(L)
  650 WRITE(6,2014) IS(L),JS(L),SH(L)
  701 IF(NUMST.EQ.0) GO TO 700
      MPRINT=0
      DO 680 L=1,NUMST
      IF(MPRINT.NE.0) GO TO 670
      WRITE(6,2025)
      MPRINT=58
  670 MPRINT=MPRINT-1
      READ(5,1006) IT(L),JT(L),ST(L)
  680 WRITE(6,2014)IT(L),JT(L),ST(L)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     DETERMINE BANDWIDTH, INITIALIZE ELASTIC-PLASTIC RATIO,
C     AND CONVERT BETA FROM DEGREES TO RADIANS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  700 J=0
      DO 710 N=1,NUMEL
      IX(N,5)=IABS(IX(N,5))
      DO 710 I=1,4
      DO 710 L=1,4
      KK=IABS(IX(N,I)-IX(N,L))
      IF(KK.GE.J) J=KK
  710 CONTINUE
      MBAND=3*J+3
      IF(NL.GT.1) GO TO 721
      DO 720 N=1,NUMEL
      EPR(N)=1.
      ALPHA(N)=ALPHA(N)/57.295780
  720 BETA(N)=BETA(N)/57.295780
  721 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     SOLVE NONLINEAR PROBLEM BY SUCCESSIVE APPROXIMATIONS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 800 NNN=1,NNLA
C
C     FORM STIFFNESS MATRIX
C
      CALL STIFF
C
C     SOLVE FOR DISPLACEMENTS
C
      CALL SOLV
C
C     COMPUTE STRESSES
C
```

```
      CALL STRESS
      CALL STORE
      IF(IDONE.NE.1) GO TO 800
  799 NITER=NNN
      IF(IDONE.EQ.1) GO TO 810
  800 CONTINUE
  810 IF(IDONE.EQ.1) WRITE(6,2016) NITER
      IF(IDONE.NE.1) WRITE(6,2017) NITER
  900 CONTINUE
  950 CONTINUE
       CALL ASEMBL
       CALL ANSWER
  910 CONTINUE
 1000 FORMAT(20A4/6I5,F5.0,5I5)
 1001 FORMAT(3F10.0)
 1004 FORMAT (2I5,2F10.0)
 1005 FORMAT(7F10.0)
 1006 FORMAT (2I5,F10.0)
 1030 FORMAT(3F10.0)
 2000 FORMAT (2H1 ,20A4/
     1 33H0  NUMBER OF APPROXIMATIONS------I4/
     2 33H0  NUMBER OF TEMPERATURE CARDS---I4/
     3 33H0  NUMBER OF MATERIALS-----------I4/
     4 33H0  NUMBER OF PRESSURE CARDS------I4/
     5 33H0  NUMBER OF SHEAR CARDS---------I4/
     6 33H0  NUMBER OF TORSION CARDS-------I4/
     7 33H0  REFERENCE TEMPERATURE---------E12.4/
     8 33H0  NUMBER OF INERTIA CARDS-------I4/
     9 33H0  NUMBER OF LOAD INCREMENTS-----I4/)
 2001 FORMAT (1H1,13X,1HR,14X,1HZ,14X,1HT)
 2002 FORMAT (3F15.3)
 2003 FORMAT (35H1    N      R         Z         T)
 2004 FORMAT (I5,2F10.4,F10.0)
 2008 FORMAT (74H1  EL   I   J   K  L    MATERIAL   ANGLE BETA   ANGLE A
     1LPHA    TEMPERATURE)
 2009 FORMAT (I5,4I4,I8,F11.1,2F13.3)
 2010 FORMAT (1H1,"MATERIAL IDENTIFICATION NUMBER =",I2/
     11H ,"NO. OF MATERIAL TEMPERATURE CARDS =",I2/
     21H ,"MASS DENSITY =",E15.7)
 2011 FORMAT (1H ,"TEMPERATURE =",E15.7/
     11H ,"MODULUS OF ELASTICITY-EN =",E15.7/
     21H ,"MODULUS OF ELASTICITY-ES =",E15.7/
     31H ,"MODULUS OF ELASTICITY-ET =",E15.7/
     41H ,"POISSON RATIO-NUNS =",E15.7/
     51H ,"POISSON RATIO-NUNT =",E15.7/
     61H ,"POISSON RATIO-NUST =",E15.7/
     71H ,"SHEAR MODULUS-GNS =",E15.7/
     81H ,"SHEAR MODULUS-GST =",E15.7/
     91H ,"SHEAR MODULUS-GTN =",E15.7/
     11H ,"COEFFICIENT OF THERMAL EXPANSION-AN =",E15.7/
```

```
     21H ,"COEFFICIENT OF THERMAL EXPANSION-AS =",E15.7/
     31H ,"COEFFICIENT OF THERMAL EXPANSION-AT =",E15.7/)
 2012 FORMAT (1H ,"TEMPERATURE =",E15.7/
     11H ,"MODULUS OF ELASTICITY-EN =",E15.7/
     21H ,"MODULUS OF ELASTICITY-ES =",E15.7/
     31H ,"MODULUS OF ELASTICITY-ET =",E15.7/
     41H ,"POISSON RATIO-NUNS =",E15.7/
     51H ,"POISSON RATIO-NUNT =",E15.7/
     61H ,"POISSON RATIO-NUST =",E15.7/
     71H ,"SHEAR MODULUS-GNS =",E15.7/
     81H ,"SHEAR MODULUS-GST =",E15.7/
     91H ,"SHEAR MODULUS-GTN =",E15.7/
     11H ,"FREE THERMAL STRAIN-FN =",E15.7/
     21H ,"FREE THERMAL STRAIN-FS =",E15.7/
     31H ,"FREE THERMAL STRAIN-FT =",E15.7/)
 2013 FORMAT (30H1 PRESSURE BOUNDARY CONDITIONS/20H    I    J  PRESSURE)
 2014 FORMAT (2I5,F10.1)
 2015 FORMAT (27H1 SHEAR BOUNDARY CONDITIONS/17H    I    J  SHEAR)
 2016 FORMAT (26H  THE SYSTEM CONVERGED IN I2,11H ITERATIONS)
 2017 FORMAT (33H  THE SYSTEM DID NOT CONVERGE IN I2,11H ITERATIONS)
 2024 FORMAT (43H0 THE AXISYMMETRIC OPTION NAS BEEN SELECTED)
 2025 FORMAT(30H1  TORSION BOUNDARY CONDITIONS/17H    I    J  SHEAR)
 2030 FORMAT(1H1,"LOAD STEP=",I4)
 2031 FORMAT(1H0 ,"AXIAL ACCELERATION  =",E12.4/
     11H0 ,"ANGULAR VELOCITY    =",E12.4/
     21H0 ,"ANGULAR ACCELERATION=",E12.4)
  920 STOP
      END
      SUBROUTINE ANGLE (R,Z,RC,ZC,ANG)
C     FIND ANGLE OF INCLINATION BETWEEN 0 AND 2*PI
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      PI=3.1415927
      D1=(Z-ZC)
      D2=(R-RC)
      IF(ABS(R-RC).GT.1.E-8) GO TO 100
      ANG=PI/2.
      IF(D1.GT.1.E-8) RETURN
      ANG=-ANG
      RETURN
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ALLOW CIRCLE TO CROSS AXIS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  100 ANG=ATAN2(D1,D2)
      RETURN
      END
       SUBROUTINE ANSWER
      INTEGER CODE
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200)
```

```
      COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
     1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N8
      COMMON/NXSOLV/SKG(132,24),FTG(132),ITOT
      COMMON/ANS2/ UT1(24), G(24,24), GR1(24,24),DUMM(24,24)
      COMMON/ANS1/NUMELS(4),NUMNPS(4)
      COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
      COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS,GTS1G(24,24,4)
      COMMON/NXMESH/THETAN(4),NST(4),NUMS(4,5),NPC(8,8)
      COMMON/ARG1/SIG1(18),EPS1(18)
      COMMON/SOLVE/B(162),A(162,81),NUMBLK,MBAND
        DIMENSION UT(24),UC1(24),UC(24),R1(24,24)
       REWIND 25
      REWIND 26
         KOLD=1
       DO 100 K=1,NTYPS
       KNEW=K
        J1=  NST(K)
      NUMNP = NUMNPS(K)
       NUMNP3 = 3*NUMNP
       NUMEL = NUMELS(K)
       K20 = K+20
      READ(26) (B(I),I=1,NUMNP3)
      READ(26) ((IX(I,J),J=1,4),I=1,NUMEL)
       DO 100 L=1,J1
       NS=NUMS(K,L)
       REWIND K20
       WRITE(6,1200) K,NS
       READ(25)((R1(I,J),J=1,24),I=1,24)
       DO 110 KK=1,4
       NP1 = NPC(NS,KK)
       NP2 = NPC(NS,KK+4)
       DO 110 I=1,3
      UC(3*(KK-1)+I) = B(3*NP1-3+I)
       UC(3*(KK-1)+I+12) = B(3*NP2-3+I)
  110  CONTINUE
      DO 115 KK=1,24
  115  UT(KK) = FTG(KK+(NS-1)*12)
      WRITE(6,900)
  900  FORMAT(" "," EL    SIGMAR    SIGMAZ    SIGMAC   SIGMARZ  SIGMAZC"
     1   ," SIGMACR       SIGEFF",/"          EPSR      ERSZ      EPSC",
     2     "        EPSRZ       ERPSZC      EPSCR")
C       IF(KOLD.EQ.KNEW) REWIND 21
C       IF(KOLD.NE.KNEW) KOLD=KNEW
       DO 120 N=1,NUMEL
       READ(K20)((CRZ(I,J),J=1,6),I=1,6)
       READ(K20)((BS1(I,J),J=1,30),I=1,6)
       READ(K20)((  G(I,J),J=1,24),I=1,24)
       DO 125 I=1,24
      DO 125 J=1,24
         GR1(I,J) = 0.00
```

```
      DO 125 M=1,24
  125 GR1(I,J) = GR1(I,J) + G(I,M)*R1(M,J)
      DO 126 I=1,24
      UC1(I) =0.00
      UT1(I) =0.00
      DO 126 J=1,24
      UC1(I) = UC1(I) + GR1(I,J)*UC(J)
  126 UT1(I) = UT1(I) + GR1(I,J)*UT(J)
      DO 130 I=1,4
      II=3*I
      JJ=3*IX(N,I)
      P1(II-2) = B(JJ-2)
      P1(II-1) = B(JJ-1)
      P1(II  ) = B(JJ  )
      P1(II+10) = B(JJ-2)
      P1(II+11) = B(JJ-1)
      P1(II+12) = B(JJ)
  130 CONTINUE
      DO 135 I=1,24
  135 P1(I) = P1(I) -UC1(I)+UT1(I)
      DO 136  I=1,3
      P1(I+24)= (P1(I)+P1(I+3)+P1(I+6)+P1(I+9))/4.00
  136 P1(I+27) = (P1(I+12)+P1(I+15)+P1(I+18)+P1(I+21))/4.00
      DO 140 I=1,6
      EPS1(I) = 0.00
      DO 140 J=1,30
  140 EPS1(I) = EPS1(I)+BS1(I,J)*P1(J)
      DO 150 I=1,6
      SIG1(I) = 0.00
      DO 150  J=1,6
  150 SIG1(I) = SIG1(I) +  CRZ(I,J)*EPS1(J)
      SIGEFF=(SIG1(1)-SIG1(2))**2+(SIG1(2)-SIG1(3))**2+(SIG1(3)-
     1    SIG1(1))**2+6.*(SIG1(4)**2+SIG1(5)**2+SIG1(6)**2)
      SIGEFF=SQRT(.5*SIGEFF)
      DO 141 J=1,6
  141  EPS1(J) = EPS1(J)*100.0
      WRITE(6,1000)N,(SIG1(I),I=1,6),SIGEFF
      WRITE(6,1100)(EPS1(I) ,I=1,6)
  120 CONTINUE
  100 CONTINUE
 1000 FORMAT(" ",I5,6F9.0,3X,F9.0)
 1100 FORMAT(" " ,5X,6F9.5)
 1200 FORMAT("1","SEGMENT TYPE",I5,//," ","SEGMENT NUMBER = ",I5)
      RETURN
      END
      SUBROUTINE ASEMBL
      COMMON/GLBSEG/FI(24,8),FE(24,8),UC(24,8),SK(24,24,8)
      COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS,GTS1G(24,24,4)
      COMMON/NXSOLV/SKG(132,24),FTG(132),ITOT
      COMMON/ANS2/FC(24),G(24,24),GR1(24,24),DUMM(24,24)
```

```
      ITOT= 24 + 12*(NTOTS-1)
       DO 10 I=1,ITOT
       FTG(I) = 0.00
       DO 10 J = 1,24
   10  SKG(I,J) = 0.00
       DO 100 M=1,NTOTS
C     WRITE(6,1000)(FE(I,M),I=1,24)
C     WRITE(6,1000)(FI(I,M),I=1,24)
C     WRITE(6,1000)((SK(I,J,M),J=1,24),I=1,24)
C     WRITE(6,1000)(UC(I,M),I=1,24)
 1000 FORMAT(" ",12E10.3)
C **************************************************************
C   COMBINE FI, FE, AND SK*UC INTO A TOTAL FORCE VECTOR  FC
C **************************************************************
       DO 55 I=1,24
       FC(I) = 0.00
        DO 55 J=1,24
   55  FC(I) = FC(I) + SK(I,J,M)* UC(J,M)
       DO  60 I=1,24
   60  FC(I) = FC(I) +FE(I,M) -FI(I,M)
C **********************************************************
C   NOW FILL GLOBAL FORCE AND STIFFNESS MATRICES
C **********************************************************
       DO 70 I=1,24
        I1 = I+(M-1)*12
         FTG(I1) = FTG(I1)  + FC(I)
       DO 70 J=I,24
       SKG(I1,J+1-I) = SKG (I1,J+1-I) + SK(I,J,M)
   70  CONTINUE
  100  CONTINUE
C  READ THE TOTAL NUMBER OF RESTRAINED DEGREES OF FREEDOM
       READ(5,1200)  NRDF
       WRITE(6,1255) NRDF
C    IMPOSE BOUNDARY CONDITIONS ON RESTRAINED D-O-F
       DO 150  NBC=1,NRDF
C  READ THE EQUATION NUMBER AND THE IMPOSED BOUNDARY CONDITION
       READ(5,1250) NREQ,U
       WRITE(6,1260)NREQ,U
       CALL XMODFY(U,NREQ)
  150 CONTINUE
 1200  FORMAT(I5)
 1250 FORMAT(I5,F10.0)
 1255 FORMAT(1H1,"NUMBER OF RESTRAINED DEGREES OF FREEDOM =",I10/
     1     " EQUATION NUMBER   VALUE  ")
 1260 FORMAT (" ",5X,I5,5X,F10.2)
      CALL XSOLVE
       WRITE(6,1050)
        WRITE(6,1100)(FTG(I),I=1,ITOT)
 1050 FORMAT("1","TOTAL DISPLACEMENTS AT CONNECTING NODES"/
     1             18X,2HUR,18X,2HUZ,18X,2HUT)
```

```
 1100  FORMAT(" ",3E20.7)
       RETURN
       END
       SUBROUTINE CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
       INTEGER CODE
       COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
       COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
      1NPNUM(25,80),T(1000),XT(1000)
       DIMENSION AR(25,80),AZ(25,80)
       EQUIVALENCE (R(1),AR),(Z(1),AZ)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      FIND INTERSECTION OF LINE AND CIRCLE = NEW R AND Z
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
       ANG1=ANG1+DELPHI
       RR=SQRT((RSTRT-RC)**2+(ZSTRT-ZC)**2)
       AR(I,J)=RC+RR*COS(ANG1)
       AZ(I,J)=ZC+RR*SIN(ANG1)
       RETURN
       END
       SUBROUTINE INTER
       COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
      1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
       COMMON/PLANE/NPP
       DIMENSION XM(7),R(7),Z(7),XX(9)
       DATA XX/3*.1259391805448,3*.1323941527884,.225,
      1 .696140478028,.410426192314/
       R(7)=(RR(1)+RR(2)+RR(3))/3.0
       Z(7)=(ZZ(1)+ZZ(2)+ZZ(3))/3.0
       DO 100 I=1,3
       J=I+3
       R(I)=XX(8)*RR(I)+(1.00-XX(8))*R(7)
       R(J)=XX(9)*RR(I)+(1.00-XX(9))*R(7)
       Z(I)=XX(8)*ZZ(I)+(1.00-XX(8))*Z(7)
  100  Z(J)=XX(9)*ZZ(I)+(1.00-XX(9))*Z(7)
       DO 200 I=1,7
  200  XM(I)=XX(I)*R(I)
       DO 300 I=1,10
  300  XI(I)=0.00
       AREA=.50*(RR(1)*(ZZ(2)-ZZ(3))+RR(2)*(ZZ(3)-ZZ(1))+RR(3)*(ZZ(1)
      1 -ZZ(2)))
       IF(NPP.NE.0) GO TO 600
       DO 400 I=1,7
       XI(1)=XI(1)+XM(I)
       XI(2)=XI(2)+XM(I)/R(I)
       XI(3)=XI(3)+XM(I)/(R(I)**2)
       XI(4)=XI(4)+XM(I)*Z(I)/R(I)
       XI(5)=XI(5)+XM(I)*Z(I)/(R(I)**2)
       XI(6)=XI(6)+XM(I)*(Z(I)**2)/(R(I)**2)
       XI(7)=XI(7)+XM(I)*R(I)
       XI(8)=XI(8)+XM(I)*Z(I)
```

```
      XI(9)=XI(9)+XM(I)*(R(I)**2)
  400 XI(10)=XI(10)+XM(I)*R(I)*Z(I)
      DO 500 I=1,10
  500 XI(I)=XI(I)*AREA
      RETURN
  600 XI(1)=AREA
      XI(7)=R(7)*AREA
      XI(8)=Z(7)*AREA
      RETURN
      END
      SUBROUTINE MESH
      INTEGER CODE
      DIMENSION AR(25,80),AZ(25,80),NCODE(25,80)
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200)
      EQUIVALENCE (R(1),AR),(Z(1),AZ),(IX(1,1),NCODE)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     MESH CONTROL INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      READ (5,1000) MAXI,MAXJ,NSEG,NBC,NMTL
      WRITE(6,2000) MAXI,MAXJ,NSEG,NBC,NMTL
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      ISEG=-1
      PI=3.1415927
      DO 110 J=1,100
      DO 100 I=1,25
      NCODE(I,J)=0
      AR(I,J)=0.
      AZ(I,J)=0.
      JMAX(I)=0
  100 JMIN(I)=MAXI
      IMIN(J)=MAXJ
  110 IMAX(J)=0
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     LINE SEGMENT CARDS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  150 ISEG=ISEG+1
  159 IF(ISEG.EQ.NSEG) GO TO 400
      READ(5,1001) I1,J1,R1,Z1,I2,J2,R2,Z2,I3,J3,R3,Z3,IPTION
      WRITE(6,2001)I1,J1,R1,Z1,I2,J2,R2,Z2,I3,J3,R3,Z3,IPTION
      IPTION=IPTION+1
      AR(I1,J1)=R1
      AZ(I1,J1)=Z1
      NCODE(I1,J1)=1
```

```
      CALL MNIMX(I1,J1)
      GO TO (150,200,200,300,300,200,200), IPTION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     GENERATE STRAIGHT LINES ON BOUNDARY
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
200   DI= ABS(FLOAT(I2-I1))
      DJ= ABS(FLOAT(J2-J1))
      AR(I2,J2)=R2
      AZ(I2,J2)=Z2
      NCODE(I2,J2)=1
      CALL MNIMX(I2,J2)
      ISTRT=I1
      ISTP=I2
      JSTRT=J1
      JSTP=J2
      DIFF=MAX1(DI,DJ)
      ITER=DIFF-1.
      IINC=0
      JINC=0
      IF(I2.NE.I1) IINC=(I2-I1)/IABS(I2-I1)
      IF(J2.NE.J1) JINC=(J2-J1)/IABS(J2-J1)
      KAPPA=1
      IF(I2.NE.I1.AND.J2.NE.J1.AND.IPTION.NE.3) KAPPA=2
      IF(KAPPA.EQ.2) DIFF=2.*DIFF
      RINC=(R2-R1)/DIFF
      ZINC=(Z2-Z1)/DIFF
      WRITE(6,2002) DI,DJ,DIFF,RINC,ZINC,ITER,IINC,JINC,KAPPA
C
C     CHECK FOR INPUT ERROR
C
      IF(KAPPA.NE.2.OR.DI.EQ.DJ) GO TO 210
      WRITE(6,2003)
      GO TO 150
C
C     INTERPOLATE
C
  210 I=I1
      J=J1
      WRITE(6,2004)
      DO 230 M=1,ITER
      IF(ITER.EQ.0.AND.IPTION.EQ.2) GO TO 230
      IF(ITER.EQ.0.AND.IPTION.EQ.6) GO TO 230
      IF(ITER.EQ.0.AND.IPTION.EQ.7) GO TO 230
      IF(KAPPA.EQ.2) GO TO 220
      IOLD=I
      I=I+IINC
      JOLD=J
      J=J+JINC
      AR(I,J)=AR(IOLD,JOLD)+RINC
      AZ(I,J)=AZ(IOLD,JOLD)+ZINC
```

```
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      NCODE(I,J)=1
      GO TO 230
  220 CONTINUE
      IF(I1.GT.I2.AND.IPTION.EQ.7) GO TO 221
      IF(I1.LT.I2.AND.IPTION.EQ.6) GO TO 221
      IOLD=I
      I=I+IINC
      AR(I,J)=AR(IOLD,J)+RINC
      AZ(I,J)=AZ(IOLD,J)+ZINC
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      JOLD=J
      J=J+JINC
      AR(I,J)=AR(I,JOLD)+RINC
      AZ(I,J)=AZ(I,JOLD)+ZINC
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      GO TO 230
  221 JOLD=J
      J=J+JINC
      AR(I,J)=AR(I,JOLD)+RINC
      AZ(I,J)=AZ(I,JOLD)+ZINC
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      IOLD=I
      I=I+IINC
      AR(I,J)=AR(IOLD,J)+RINC
      AZ(I,J)=AZ(IOLD,J)+ZINC
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
  230 CONTINUE
      IF(KAPPA.EQ.1) GO TO 150
      IF(I1.GT.I2.AND.IPTION.EQ.7) GO TO 231
      IF(I1.LT.I2.AND.IPTION.EQ.6) GO TO 231
      IOLD=I
      I=I+IINC
      AR(I,J)=AR(IOLD,J)+RINC
      AZ(I,J)=AZ(IOLD,J)+ZINC
      GO TO 232
  231 CONTINUE
      JOLD=J
      J=J+JINC
      AR(I,J)=AR(I,JOLD)+RINC
      AZ(I,J)=AZ(I,JOLD)+ZINC
```

```
  232 CONTINUE
      NCODE(I,J)=1
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      CALL MNIMX(I,J)
      GO TO 150
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     GENERATE CIRCULAR ARCS ON BOUNDARY
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  300 AR(I2,J2)=R2
      AZ(I2,J2)=Z2
      NCODE(I2,J2) = 1
      CALL MNIMX(I2,J2)
      IF(IPTION.EQ.5) GO TO 320
C
C     FIND CENTER OF CIRCLE
C
      AR(I3,J3)=R3
      AZ(I3,J3)=Z3
      NCODE(I3,J3)=1
      CALL MNIMX(I3,J3)
      SLAC=(Z2-Z1)/(R2-R1)
      SLBF=-1./SLAC
      SLCE=(Z3-Z2)/(R3-R2)
      SLDF=-1./SLCE
C
C     CHECK FOR INPUT ERROR
C
      IF(ABS(SLAC-SLCE).GT..001) GO TO 310
      WRITE(6,2006) R1,Z1,R2,Z2,R3,Z3,SLAC,SLCE
      GO TO 150
  310 R4=R1+(R2-R1)/2.
      Z4=Z1+(Z2-Z1)/2.
      R5=R2+(R3-R2)/2.
      Z5=Z2+(Z3-Z2)/2.
      BBF=Z4-SLBF*R4
      BDF=Z5-SLDF*R5
      RC=(BBF-BDF)/(SLDF-SLBF)
      ZC=SLBF*RC+BBF
      WRITE(6,2007) RC,ZC
      KAPPA=1
      GO TO 330
  320 KAPPA=2
      RC=R3
      ZC=Z3
  330 ISTRT=I1
      ISTP=I2
      JSTRT=J1
      JSTP=J2
      RSTRT=R1
      RSTP=R2
```

```
      ZSTRT=Z1
      ZSTP=Z2
  340 CALL ANGLE(RSTRT,ZSTRT,RC,ZC,ANG1)
      CALL ANGLE(RSTP,ZSTP,RC,ZC,ANG2)
      IF(ANG2.LE.ANG1)  ANG2=2.0*PI+ANG2
C
C     FIND ANGULAR INCREMENT
C
      DI= ABS(FLOAT(ISTP-ISTRT))
      DJ= ABS(FLOAT(JSTP-JSTRT))
      IINC=0
      JINC=0
      IF(ISTRT.NE.ISTP) IINC=(ISTP-ISTRT)/IABS(ISTP-ISTRT)
      IF(JSTRT.NE.JSTP) JINC=(JSTP-JSTRT)/IABS(JSTP-JSTRT)
      LAMDA=1
      IF(IINC.NE.0.AND.JINC.NE.0) LAMDA=2
      DIFF=MAX1(DI,DJ)
      ITER=DIFF-1.
      IF(LAMDA.EQ.2) DIFF=2.*DIFF
      DELPHI=(ANG2-ANG1)/DIFF
      WRITE(6,2008) ANG1,ANG2,DIFF,DELPHI
C
C     CHECK FOR INPUT ERROR
C
      IF(LAMDA.NE.2.OR.DI.EQ.DJ) GO TO 350
      WRITE(6,2003)
      GO TO 150
  350 IO=ISTRT
      JO=JSTRT
      WRITE(6,2004)
C
C     INTERPOLATE
C
      NPT=IABS(I2-I1)+IABS(J2-J1)-1
      DO 380 M=1,ITER
  359 IF(LAMDA.EQ.2) GO TO 360
      I=IO+IINC
      J=JO+JINC
      CALL MNIMX(I,J)
      NCODE(I,J)=1
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      GO TO 370
  360 I=IO+IINC
      J=JO
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
      J=JO+JINC
```

```
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
  370 IO=I
  380 JO=J
      IF(LAMDA.NE.2) GO TO 390
      I=IO+IINC
      NCODE(I,J)=1
      CALL MNIMX(I,J)
      CALL CIRCLE(ANG1,DELPHI,RSTRT,ZSTRT,RC,ZC,I,J)
      WRITE(6,2005) I,J,AR(I,J),AZ(I,J)
  390 IF(KAPPA.EQ.2) GO TO 150
      ISTRT=I2
      ISTP=I3
      JSTRT=J2
      JSTP=J3
      RSTRT=R2
      RSTP=R3
      ZSTRT=Z2
      ZSTP=Z3
      KAPPA=2
  399 GO TO 340
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     CALCULATE COORDINATES OF INTERIOR POINTS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  400 IF(MAXJ.LE.2) GO TO 430
      J2=MAXJ-1
      DO 420 N=1,500
      RESID=0.
      DO 410 J=2,J2
      I1=IMIN(J)+1
      I2=IMAX(J)-1
      DO 410 I=I1,I2
      IF(NCODE(I,J).EQ.1) GO TO 410
      DR=(AR(I+1,J)+AR(I-1,J)+AR(I,J+1)+AR(I,J-1))/4.-AR(I,J)
      DZ=(AZ(I+1,J)+AZ(I-1,J)+AZ(I,J+1)+AZ(I,J-1))/4.-AZ(I,J)
      RESID=RESID+ABS(DR)+ABS(DZ)
      AR(I,J)=AR(I,J)+1.8*DR
      AZ(I,J)=AZ(I,J)+1.8*DZ
  410 CONTINUE
      IF(N.EQ.1) RES1=RESID
      IF(N.EQ.1.AND.RESID.EQ.0.)GO TO 430
      IF(RESID/RES1.LT.1.E-5) GO TO 430
  420 CONTINUE
  430 WRITE(6,2009) N
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      CALL POINTS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 1000 FORMAT (5I5)
```

```
1001 FORMAT (3(2I3,2F8.3),I5)
2000 FORMAT (30H1  MESH GENERATION INFORMATION//
    1 41H0  MAXIMUM VALUE OF I IN THE MESH---------I3/
    2 41H0  MAXIMUM VALUE OF J IN THE MESH---------I3/
    3 41H0  NUMBER OF LINE SEGMENT CARDS-----------I3/
    4 41H0  NUMBER OF BOUNDARY CONDITION CARDS----I3/
    5 41H0  NUMBER OF MATERIAL BLOCK CARDS---------I3///)
2001 FORMAT (//88H  INPUT   I1  J1    R1      Z1    I2  J2    R2      Z
    12    I3  J3    R3      Z3     IPTION/8X,3(2I4,2F8.4),I6)
2002 FORMAT (5H  DI=F4.0,5H  DJ=F4.0,7H  DIFF=F4.0,7H  RINC=F8.3,7H  ZI
    1NC=F8.3,7H  ITER=I3,7H  IINC=I3,7H  JINC=I3,8H  KAPPA=I1)
2003 FORMAT(1X,38H**BAD INPUT--THIS LINE IS NOT DIAGONAL)
2004 FORMAT (30H    I    J         AR        AZ)
2005 FORMAT (2I5,2F11.6)
2006 FORMAT (51H ** BAD INPUT - THESE POINTS DO NOT DEFINE A CIRCLE,/,
    13X,6F12.4,10X,2E20.8)
2007 FORMAT(19H  CENTER COORDINATE,(F11.6,1X,F11.6,1X))
2008 FORMAT (7H  ANG1=F9.6,7H  ANG2=F9.6,7H  DIFF=F3.0,9H  DELPHI=F9.6)
2009 FORMAT (//30H  COORDINATES CALCULATED AFTER I3,11H ITERATIONS)
     RETURN
     END
C
C
C        SUBROUTINE MINV
C
C        PURPOSE
C           INVERT A MATRIX
C
C        USAGE
C           CALL MINV(A,N,D,L,M)
C
C        DESCRIPTION OF PARAMETERS
C           A - INPUT MATRIX, DESTROYED IN COMPUTATION AND REPLACED BY
C               RESULTANT INVERSE.
C           N - ORDER OF MATRIX A
C           D - RESULTANT DETERMINANT
C           L - WORK VECTOR OF LENGTH N
C           M - WORK VECTOR OF LENGTH N
C
C        REMARKS
C           MATRIX A MUST BE A GENERAL MATRIX
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           THE STANDARD GAUSS-JORDAN METHOD IS USED. THE DETERMINANT
C           IS ALSO CALCULATED. A DETERMINANT OF ZERO INDICATES THAT
C           THE MATRIX IS SINGULAR.
C
```

```
C     ...........................................................
C
      SUBROUTINE MINV(A,N,D,L,M)
      DIMENSION A(1),L(1),M(1)
C
C        ........................................................
C
C        IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE
C        C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION
C        STATEMENT WHICH FOLLOWS.
C
C     DOUBLE PRECISION A,D,BIGA,HOLD
C
C        THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS
C        APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS
C        ROUTINE.
C
C        THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO
C        CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  ABS IN STATEMENT
C        10 MUST BE CHANGED TO DABS.
C
C        ........................................................
C
C        SEARCH FOR LARGEST ELEMENT
C
      D=1.0
      NK=-N
      DO 80 K=1,N
      NK=NK+N
      L(K)=K
      M(K)=K
      KK=NK+K
      BIGA=A(KK)
      DO 20 J=K,N
      IZ=N*(J-1)
      DO 20 I=K,N
      IJ=IZ+I
   10 IF(ABS(BIGA)-ABS(A(IJ))) 15,20,20
   15 BIGA=A(IJ)
      L(K)=I
      M(K)=J
   20 CONTINUE
C
C        INTERCHANGE ROWS
C
      J=L(K)
      IF(J-K) 35,35,25
   25 KI=K-N
      DO 30 I=1,N
      KI=KI+N
```

```
      HOLD=-A(KI)
      JI=KI-K+J
      A(KI)=A(JI)
   30 A(JI) =HOLD
C
C        INTERCHANGE COLUMNS
C
   35 I=M(K)
      IF(I-K) 45,45,38
   38 JP=N*(I-1)
      DO 40 J=1,N
      JK=NK+J
      JI=JP+J
      HOLD=-A(JK)
      A(JK)=A(JI)
   40 A(JI) =HOLD
C
C        DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS
C        CONTAINED IN BIGA)
C
   45 IF(BIGA) 48,46,48
   46 D=0.0
      RETURN
   48 DO 55 I=1,N
      IF(I-K) 50,55,50
   50 IK=NK+I
      A(IK)=A(IK)/(-BIGA)
   55 CONTINUE
C
C        REDUCE MATRIX
C
      DO 65 I=1,N
      IK=NK+I
      HOLD=A(IK)
      IJ=I-N
      DO 65 J=1,N
      IJ=IJ+N
      IF(I-K) 60,65,60
   60 IF(J-K) 62;65,62
   62 KJ=IJ-I+K
      A(IJ)=HOLD*A(KJ)+A(IJ)
   65 CONTINUE
C
C        DIVIDE ROW BY PIVOT
C
      KJ=K-N
      DO 75 J=1,N
      KJ=KJ+N
      IF(J-K) 70,75,70
   70 A(KJ)=A(KJ)/BIGA
```

```
   75 CONTINUE
C
C        PRODUCT OF PIVOTS
C
      D=D*BIGA
C
C        REPLACE PIVOT BY RECIPROCAL
C
      A(KK)=1.0/BIGA
   80 CONTINUE
C
C        FINAL ROW AND COLUMN INTERCHANGE
C
      K=N
  100 K=(K-1)
      IF(K) 150,150,105
  105 I=L(K)
      IF(I-K) 120,120,108
  108 JQ=N*(K-1)
      JR=N*(I-1)
      DO 110 J=1,N
      JK=JQ+J
      HOLD=A(JK)
      JI=JR+J
      A(JK)=-A(JI)
  110 A(JI) =HOLD
  120 J=M(K)
      IF(J-K) 100,100,125
  125 KI=K-N
      DO 130 I=1,N
      KI=KI+N
      HOLD=A(KI)
      JI=KI-K+J
      A(KI)=-A(JI)
  130 A(JI) =HOLD
      GO TO 100
  150 RETURN
      END
      SUBROUTINE MNIMX(I,J)
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      IF(J.LT.JMIN(I)) JMIN(I)=J
      IF(J.GT.JMAX(I)) JMAX(I)=J
      IF(I.LT.IMIN(J)) IMIN(J)=I
      IF(I.GT.IMAX(J)) IMAX(J)=I
      RETURN
      END
      SUBROUTINE MODIFY(NEQ,N,U)
      COMMON/SOLVE/B(162),A(162,81),NUMBLK,MBAND
      DO 10 M=2,MBAND
      K=N-M+1
```

```
      IF(K.LE.0) GO TO 5
      B(K)=B(K)-A(K,M)*U
      A(K,M)=0.00
    5 K=N+M-1
      IF(NEQ.LT.K) GO TO 10
      B(K)=B(K)-A(N,M)*U
      A(N,M)=0.00
   10 CONTINUE
      A(N,1)=1.00
      B(N)=U
      RETURN
      END
      SUBROUTINE MPLOT
      INTEGER CODE
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      REAL X(100),Y(100),TX(2),TY(2),TITLE(20),ZMAX
      READ (5,1000) TITLE,RMAX,ZMAX
C     CALL CCP2SY (0.7,0.2,0.2,TITLE,0.0,80)
C     CALL CCP1PL (0.7,0.7,-3)
      TX(1)=0.00
      TY(1)=0.00
      TX(2)=RMAX/9.0
      TY(2)=RMAX/9.0
      ZMAX=ZMAX*TY(2)+2.0
      IF (ZMAX.LT.17.0) ZMAX=17.0
      DO 100 J=1,MAXJ
      NSTART=IMIN(J)
      NSTOP=IMAX(J)
      N=0
      DO 101 I=NSTART,NSTOP
      N=N+1
      NP=NPNUM(I,J)
      Y(N)=R(NP)
  101 X(N)=Z(NP)
C     CALL CCP6LN (X,Y,N,1,TX,TY)
  100 CONTINUE
      DO 102 I=1,MAXI
      NSTART=JMIN(I)
      NSTOP=JMAX(I)
      N=0
      DO 103 J=NSTART,NSTOP
      N=N+1
      NP=NPNUM(I,J)
      Y(N)=R(NP)
  103 X(N)=Z(NP)
C     CALL CCP6LN (X,Y,N,1,TX,TY)
  102 CONTINUE
C     CALL CCP1PL (ZMAX,-0.7,-3)
```

```
1000 FORMAT (20A4/2F10.0)
     RETURN
     END
     SUBROUTINE NAXSTF(II,JJ,KK)
     INTEGER CODE
     COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
     COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
    1NUMST
     COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
    1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
     COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
    1NPNUM(25,80),T(1000),XT(1000)
     COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
    1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
    2ST(200)
     COMMON/NXQUAD/AR1
     COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
     DIMENSION C(18,18),B(18,18),B1(6,18),B2(6,18),B3(6,18),B4(6,18),
    1          B5(6,18),B6(6,18),B1A(6,18),B1B(6,18),B2A(6,18),B2B(6,18
    2          ),B3A(6,18),B3B(6,18),B4A(6,18),B4B(6,18),B5A(6,18),
    3          B5B(6,18),B6A(6,18),B6B(6,18)
C    ZERO MATRICES
     DO 100  I=1,18
     DO 100  J=1,18
 100 C(I,J)= 0.0
     DO 110 I=1,6
     DO 110 J=1,18
     B1(I,J) =0.0
     B2(I,J) =0.0
     B3(I,J) =0.0
     B4(I,J) =0.0
     B5(I,J) =0.0
 110 B6(I,J) =0.0
     RR(1) = RRR(II)
     RR(2) = RRR(JJ)
     RR(3) = RRR(KK)
     ZZ(1) = ZZZ(II)
     ZZ(2) = ZZZ(JJ)
     ZZ(3) = ZZZ(KK)
     COMM=RR(2)*(ZZ(3)-ZZ(1))+RR(1)*(ZZ(2)-ZZ(3))+RR(3)*(ZZ(1)-ZZ(2))
C   FILL  C INVERSE
     C(1,1)= ( RR(2)*ZZ(3) -RR(3)* ZZ(2)) / COMM
     C(1,4)= ( RR(3)*ZZ(1) -RR(1)* ZZ(3)) / COMM
     C(1,7)= ( RR(1)*ZZ(2) -RR(2)* ZZ(1)) / COMM
     C(2,1)= ( ZZ(2) - ZZ(3)) / COMM
     C(2,4)= ( ZZ(3) - ZZ(1)) / COMM
     C(2,7)= ( ZZ(1) - ZZ(2)) / COMM
     C(3,1)= ( RR(3) - RR(2)) / COMM
     C(3,4)= ( RR(1) - RR(3)) / COMM
     C(3,7)= ( RR(2) - RR(1)) / COMM
```

```
      C(4,2)= C(1,1)
      C(4,5)= C(1,4)
      C(4,8)= C(1,7)
      C(5,2)= C(2,1)
      C(5,5)= C(2,4)
      C(5,8)= C(2,7)
      C(6,2)= C(3,1)
      C(6,5)= C(3,4)
      C(6,8)= C(3,7)
      C(7,3)= C(1,1)
      C(7,6)= C(1,4)
      C(7,9)= C(1,7)
      C(8,3)= C(2,1)
      C(8,6)= C(2,4)
      C(8,9)= C(2,7)
      C(9,3)= C(3,1)
      C(9,6)= C(3,4)
        C(9,9) = C(3,7)
      DO 120 I=10,18
      DO 120 J= 1,9
      I1 = I-9
      J1=J+9
      C(I,J) =(-1./THETA) * C(I1,J)
      C(I,J1)=( 1./THETA) * C(I1,J)
  120 CONTINUE
C   FILL  B MATRICES
C   B1  CONSTANT TERMS
C   B2  THETA TERMS
C   B3  1/R  TERMS
C   B4  THETA/ R  TERMS
C   B5  Z/R TERMS
C   B6  THETA *Z/R TERMS
      DO  130 J=1,18
      B1(1,J) = C(2,J)
      B1(2,J) = C(6,J)
      B1(3,J) = C(2,J)+C(17,J)
      B1(4,J) = C(3,J)+C(5,J)
      B1(5,J) = C(9,J) +C(14,J)
      B1(6,J) = C(11,J)
      B2(1,J) = C(11,J)
      B2(2,J) = C(15,J)
      B2(3,J) = C(11,J)
      B2(4,J) = C(12,J)+C(14,J)
      B2(5,J) = C(18,J)
      B3(3,J) = C(1,J)+ C(16,J)
      B3(5,J) = C(13,J)
      B3(6,J) = C(10,J) - C(7,J)
      B4(3,J) = C(10,J)
      B4(6,J) = -C(16,J)
      B5(3,J) =  C(3,J) +C(18,J)
```

```
      B5(5,J) =  C(15,J)
      B5(6,J) =  C(12,J)-C(9,J)
      B6(3,J) =  C(12,J)
      B6(6,J) = -C(18,J)
  130 CONTINUE
C NOW CALCULATE  BT * D * B
      CALL INTER
      THETA2 = (THETA **2)/2.0
      THETA3 = (THETA **3)/3.0
      DO 140 I=1,6
      DO 140 J=1,18
      B1A(I,J)=(B1(I,J)*XI(1) +B3(I,J)* XI(2) + B5(I,J)* XI(4))* THETA +
     1         (B2(I,J)*XI(1) +B4(I,J)* XI(2) + B6(I,J)* XI(4))* THETA2
      B2A(I,J)=(B1(I,J)*XI(1) +B3(I,J)* XI(2) + B5(I,J)* XI(4))* THETA2
     1    +    (B2(I,J)*XI(1) +B4(I,J)* XI(2) + B6(I,J)* XI(4))* THETA3
      B3A(I,J)=(B1(I,J)*XI(2) +B3(I,J)* XI(3) + B5(I,J)* XI(5))* THETA
     1        +(B2(I,J)*XI(2) +B4(I,J)* XI(3) + B6(I,J)* XI(5))* THETA2
      B4A(I,J)=(B1(I,J)*XI(2) +B3(I,J)* XI(3) + B5(I,J)* XI(5))* THETA2
     1        +(B2(I,J)*XI(2) +B4(I,J)* XI(3) + B6(I,J)* XI(5))* THETA3
      B5A(I,J)=(B1(I,J)*XI(4) +B3(I,J)* XI(5) + B5(I,J)* XI(6))* THETA
     1       + (B2(I,J)*XI(4) +B4(I,J)* XI(5) + B6(I,J)* XI(6))* THETA2
      B6A(I,J)=(B1(I,J)*XI(4) +B3(I,J)* XI(5) + B5(I,J)* XI(6))* THETA2
     1       + (B2(I,J)*XI(4) +B4(I,J)* XI(5) + B6(I,J)* XI(6))* THETA3
  140 CONTINUE
      DO 150 I=1,6
      DO 150 K=1,18
      B1B(I,K)= 0.0
      B2B(I,K)= 0.0
      B3B(I,K)= 0.0
      B4B(I,K)= 0.0
      B5B(I,K)= 0.0
      B6B(I,K)= 0.0
      DO 150 J=1,6
      B1B(I,K) = B1B(I,K) + CRZ(I,J) * B1A(J,K)
      B2B(I,K) = B2B(I,K) + CRZ(I,J) * B2A(J,K)
      B3B(I,K) = B3B(I,K) + CRZ(I,J) * B3A(J,K)
      B4B(I,K) = B4B(I,K) + CRZ(I,J) * B4A(J,K)
      B5B(I,K) = B5B(I,K) + CRZ(I,J) * B5A(J,K)
      B6B(I,K) = B6B(I,K) + CRZ(I,J) * B6A(J,K)
  150 CONTINUE
      DO 160  I=1,18
      DO 160  K=1,18
       B(I,K)=0.0
      DO 160  J=1,6
      B(I,K) = B(I,K) + B1(J,I)* B1B(J,K)+B2(J,I)*B2B(J,K)+B3(J,I)*
     1     B3B(J,K)+B4(J,I)*B4B(J,K)+B5(J,I)*B5B(J,K)+B6(J,I)*B6B(J,K)
  160 CONTINUE
  250   CONTINUE
C  B(I,K) NOW CONTAINS THE STIFFNESS MATRIX FOR ONE TRIANGULAR ELEMENT
      AR1 = AR1 + XI(1) *THETA
```

```
      DO 235 K=1,6
       DO 235 I=1,3
      BS1(K,3*II-3+I) = BS1(K,3*II-3+I) +B1A(K,I    )
      BS1(K,3*JJ-3+I) = BS1(K,3*JJ-3+I) +B1A(K,I+3 )
      BS1(K,3*KK-3+I) = BS1(K,3*KK-3+I) +B1A(K,I+6 )
      BS1(K,3*II+I+12)= BS1(K,3*II+12+I)+B1A(K,I+9 )
      BS1(K,3*JJ+I+12)= BS1(K,3*JJ+12+I)+B1A(K,I+12)
      BS1(K,3*KK+I+12)= BS1(K,3*KK+12+I)+B1A(K,I+15)
235    CONTINUE
      IIM = 3* II -3
      JJM = 3* JJ -3
      KKM = 3* KK -3
      DO 170 K=1,4
      DO 170 I=1,3
      DO 170 J=1,3
      IF(K.EQ.1 .OR. K.EQ.2) I1=I
      IF(K.EQ.3 .OR. K.EQ.4) I1=I +9
      IF(K.EQ.1 .OR. K.EQ.3) J1=J
      IF(K.EQ.2 .OR. K.EQ.4) J1=J    +9
      IF(K.EQ.1 .OR. K.EQ.2) K1=0
      IF(K.EQ.3 .OR. K.EQ.4) K1=15
      IF(K.EQ.1 .OR. K.EQ.3) K2=0
      IF(K.EQ.2 .OR. K.EQ.4) K2=15
182   KK2=KKM
      II2=IIM
      JJ2=JJM
180   KK1=KKM
      JJ1=JJM
      II1=IIM
       S1(II1+I+K1,II2+J+K2) = S1(II1+I+K1,II2+J+K2) +B(I1,J1)
       S1(II1+I+K1,JJ2+J+K2) = S1(II1+I+K1,JJ2+J+K2) +B(I1,J1+3)
       S1(II1+I+K1,KK2+J+K2) = S1(II1+I+K1,KK2+J+K2) +B(I1,J1+6)
       S1(JJ1+I+K1,II2+J+K2) = S1(JJ1+I+K1,II2+J+K2) +B(I1+3,J1)
       S1(JJ1+I+K1,JJ2+J+K2) = S1(JJ1+I+K1,JJ2+J+K2) +B(I1+3,J1+3)
       S1(JJ1+I+K1,KK2+J+K2) = S1(JJ1+I+K1,KK2+J+K2) +B(I1+3,J1+6)
       S1(KK1+I+K1,II2+J+K2) = S1(KK1+I+K1,II2+J+K2) + B(I1+6,J1)
       S1(KK1+I+K1,JJ2+J+K2) = S1(KK1+I+K1,JJ2+J+K2) + B(I1+6,J1+3)
       S1(KK1+I+K1,KK2+J+K2) = S1(KK1+I+K1,KK2+J+K2) + B(I1+6,J1+6)
170    CONTINUE
       RETURN
       END
      FUNCTION NODE(I,J)
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      NODE=0
      DO 100 JJ=1,J
      NSTART=IMIN(JJ)
      NSTOP=IMAX(JJ)
      DO 100 II=NSTART,NSTOP
      NODE=NODE+1
      IF(JJ.EQ.J.AND.II.EQ.I) RETURN
```

```
  100 CONTINUE
      RETURN
      END
      SUBROUTINE POINTS
      INTEGER CODE
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200)
      COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
      COMMON/TD/IMIN(100),IMAX(100),JMIN(25),JMAX(25),MAXI,MAXJ,NMTL,NBC
      COMMON/PLANE/NPP
      DIMENSION AR(25,80),AZ(25,80),MATRIL(100,5),BLKANG(100),BLKALF(1
     100)
      DIMENSION IBNG(100),NBNG(100)
      EQUIVALENCE (R(1),AR),(Z(1),AZ)
C     ESTABLISH NODAL POINT INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      NEL=0
      NODSUM=0
      DO 100 J=1,MAXJ
      NSTART=IMIN(J)
      NSTOP=IMAX(J)
      DO 100 I=NSTART,NSTOP
  100 NODSUM=NODSUM+1
      NELSUM=0
      JJMAX=MAXJ-1
      DO 110 JJ=1,JJMAX
      NSTOP=MIN0(IMAX(JJ),IMAX(JJ+1))-1
      NSTART=MAX0(IMIN(JJ),IMIN(JJ+1))
      DO 110 II=NSTART,NSTOP
  110 NELSUM=NELSUM+1
      NUMNP=NODSUM
      NUMEL=NELSUM
      DO 120 J=1,MAXJ
      NSTART=IMIN(J)
      NSTOP=IMAX(J)
      DO 120 I=NSTART,NSTOP
      NPNUM(I,J)=NODE(I,J)
      NP=NPNUM(I,J)
      R(NP)=AR(I,J)
  120 Z(NP)=AZ(I,J)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     READ AND ASSIGN BOUNDARY CONDITIONS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZE
```

```
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 130 I=1,NUMNP
      CODE(I)=0
      IF(R(I).EQ.0..AND.NPP.EQ.0) CODE(I)=1.
      XR(I)=0.
      XZ(I)=0.
      XT(I)=0.0
  130 T(I)=0.
      IF(NBC.EQ.0) GO TO 210
      DO 200 IBCON=1,NBC
      READ(5,1002) I1,I2,J1,J2,ICN,RCON,ZCON,TCON
      DO 200 I=I1,I2
      DO 200 J=J1,J2
      NP=NPNUM(I,J)
      CODE(NP)=ICN
      XR(NP)=RCON
      XT(NP)=TCON
  200 XZ(NP)=ZCON
  210 MPRINT=0
      DO 230 J=1,MAXJ
      NSTART=IMIN(J)
      NSTOP=IMAX(J)
      DO 230 I=NSTART,NSTOP
      NP=NPNUM(I,J)
      IF(MPRINT.NE.0) GO TO 220
      WRITE(6,2000)
      MPRINT=59
  220 MPRINT=MPRINT-1
  230 WRITE(6,2001) I,J,NP,CODE(NP),R(NP),Z(NP),XR(NP),XZ(NP),XT(NP)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ASSIGN MATERIALS IN BLOCKS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 300 M1=1,NUMEL
  300 IX(M1,5)=0
      DO 310 IMTL=1,NMTL
      READ (5,1000) MTL,(MATRIL(IMTL,IM),IM=2,5),BLKANG(IMTL),BLKALF(IMT
     1L),IBNG(IMTL),NBNG(IMTL)
  310 MATRIL(IMTL,1)=MTL
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ESTABLISH ELEMENT INFORMATION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      JJMAX=MAXJ-1
      N=0
      MTL=1
      KTL=1
      DO 440 JJ=1,JJMAX
      NSTOP=MIN0(IMAX(JJ),IMAX(JJ+1))-1
      NSTART=MAX0(IMIN(JJ),IMIN(JJ+1))
      DO 440 II=NSTART,NSTOP
      NEL=NEL+1
```

```
      DO 400 IMTL=1,NMTL
      IF(II.LT.MATRIL(IMTL,2)) GO TO 400
      IF(II.GE.MATRIL(IMTL,3)) GO TO 400
      IF(JJ.LT.MATRIL(IMTL,4)) GO TO 400
      IF(JJ.GE.MATRIL(IMTL,5)) GO TO 400
      KAT=IMTL
      MAT=MATRIL(IMTL,1)
  400 CONTINUE
      IF(KAT.EQ.KTL) GO TO 410
      KTL=KAT
      MTL=MAT
      GO TO 420
  410 IF(II.EQ.NSTART) GO TO 420
      IF(JJ.NE.JJMAX.OR.II.NE.NSTOP) GO TO 440
      M=NEL+1
      IANG=ICNG
      NANG=NCNG
      GO TO 421
  420 I=NPNUM(II,JJ)
      J=I+1
      K=NPNUM(II+1,JJ+1)
      L=K-1
      M=NEL
      IX(M,1)=I
      IX(M,2)=J
      IX(M,3)=K
      IX(M,4)=L
      IX(M,5)=MTL
      BETA(M)=BLKANG(KTL)
      ALPHA(M)=BLKALF(KTL)
      IANG=ICNG
      NANG=NCNG
      ICNG=IBNG(KTL)
      NCNG=NBNG(KTL)
421   NC=2
  430 N=N+1
      IF(M.LE.N) GO TO 440
      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)
      BETA(N)=BETA(N-1)
      IF(IANG.EQ.1) GO TO 442
      ALPHA(N)=ALPHA(N-1)
      GO TO 443
  442 CONTINUE
      IF(NC.GT.NANG) GO TO 444
      ALPHA(N)=ALPHA(N-1)
      GO TO 443
```

```
  444 NC=1
      ALPHA(N)=-ALPHA(N-1)
  443 CONTINUE
      NC=NC+1
      IF(M.GT.N) GO TO 430
  440 CONTINUE
      IF(NUMNP.GT.2000) WRITE(6,2002)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     SET NODAL POINT TEMPERATURE TO REFERENCE TEMPERATURE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(NUMTC.NE.0) RETURN
      DO 500 N=1,NUMNP
  500 T(N)=TREF
 1000 FORMAT (5I5,2F10.0,2I5)
 1002 FORMAT(4I5,I10,3F10.0)
 2000 FORMAT (128H1   I     J     NP          TYPE     R-ORDINATE     Z-ORDINA
     1TE   R LOAD OR DISPLACEMENT  Z LOAD OR DISPLACEMENT  T LOAD OR DISP
     2LACEMENT)
 2001 FORMAT (2I5,I6,I12,F13.6,F14.6,E26.7,E24.7,E24.7)
 2002 FORMAT (35H  BAD INPUT - TOO MANY NODAL POINTS)
      RETURN
      END
      SUBROUTINE QUAD
      INTEGER CODE
      REAL NUSN,NUTN,NUTS,NUNS,NUNT,NUST
      DIMENSION DUMMY(6,6),DUMMY1(6,6)
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/NXQUAD/AR1
      COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
      COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200)
      COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
     1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
      COMMON/RESULT/BS(6,15),D(6,6),C(6,6),AR,BB(6,9),CNS(6,6)
      COMMON/PLANE/NPP
      COMMON/DUM1/S1TEM(3,30),S1T(24,24),TS(6,24)
       DIMENSION S2T(24,6)
       DIMENSION BS1T(6,3)   ,P1T(3)
      I1=IX(N,1)
      J1=IX(N,2)
      K1=IX(N,3)
      L1=IX(N,4)
      MTYPE=IX(N,5)
      IX(N,5)=-IX(N,5)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
C     INTERPOLATE MATERIAL PROPERTIES
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 100 I=1,12
  100 EE(I)=E(1,I+1,MTYPE)
      DO 110 I=1,6
      DO 110 J=1,6
      CNS(I,J)=0.00
      C(I,J)=0.00
  110 D(I,J)=0.00
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FORM STRESS-STRAIN RELATIONSHIP IN N-S-T SYSTEM
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      NUNS=EE(4)
      NUNT=EE(5)
      NUST=EE(6)
      NUSN=(EE(2)*NUNS)/EE(1)
      NUTN=(EE(3)*NUNT)/EE(1)
      NUTS=(EE(3)*NUST)/EE(2)
      DIV=1.00-NUNS*NUSN-NUST*NUTS-NUNT*NUTN-NUSN*NUNT*NUTS
     1-NUNS*NUTN*NUST
      CNS(1,1)=EE(1)*(1.00-NUST*NUTS)/DIV
      CNS(1,2)=EE(2)*(NUNS+NUNT*NUTS)/DIV
      CNS(1,3)=EE(3)*(NUNT+NUNS*NUST)/DIV
      CNS(2,1)=CNS(1,2)
      CNS(2,2)=EE(2)*(1.00-NUNT*NUTN)/DIV
      CNS(2,3)=EE(3)*(NUST+NUSN*NUNT)/DIV
      CNS(3,1)=CNS(1,3)
      CNS(3,2)=CNS(2,3)
      CNS(3,3)=EE(3)*(1.00-NUNS*NUSN)/DIV
      CNS(4,4)=EE(7)
      CNS(5,5)=EE(8)
      CNS(6,6)=EE(9)
C     SET UP STRAIN TRANSFORM TO N-S-T SYSTEM
      SINA=SIN(ALPHA(N))
      COSA=COS(ALPHA(N))
      S2=SINA**2
      C2=COSA**2
      SC=SINA*COSA
      D(1,1)=C2
      D(1,3)=S2
      D(1,6)=-SC
      D(2,1)=S2
      D(2,3)=C2
      D(2,6)=SC
      D(3,2)=1.00
      D(4,1)=2.00*SC
      D(4,3)=-2.00*SC
      D(4,6)=C2-S2
      D(5,4)=SINA
      D(5,5)=COSA
```

```
      D(6,4)=COSA
      D(6,5)=-SINA
C     SET UP STRAIN TRANSFORMATION TO R-Z-T SYSTEM
      SINB=SIN(BETA(N))
      COSB=COS(BETA(N))
      S2=SINB**2
      C2=COSB**2
      SC=SINB*COSB
      C(1,1)=S2
      C(1,2)=C2
      C(1,4)=SC
      C(2,1)=C2
      C(2,2)=S2
      C(2,4)=-SC
      C(3,3)=1.00
      C(4,1)=-2.00*SC
      C(4,2)=2.00*SC
      C(4,4)=S2-C2
      C(5,5)=SINB
      C(5,6)=-COSB
      C(6,5)=COSB
      C(6,6)=SINB
C     CALCULATE CRZ MATRIX
      DO 120 I=1,6
      DO 120 J=1,6
      DUMMY(I,J)=0.00
      DO 120 K=1,6
  120 DUMMY(I,J)=DUMMY(I,J)+D(I,K)*C(K,J)
      DO 130 I=1,6
      DO 130 J=1,6
      DUMMY1(I,J)=0.00
      DO 130 K=1,6
  130 DUMMY1(I,J)=DUMMY1(I,J)+CNS(I,K)*DUMMY(K,J)
      DO 140 I=1,6
      DO 140 J=1,6
      DUMMY(I,J)=0.00
      DO 140 K=1,6
  140 DUMMY(I,J)=DUMMY(I,J)+D(K,I)*DUMMY1(K,J)
      DO 160 I=1,6
      DO 150 J=1,6
      CRZ(I,J)=0.00
      DO 150 K=1,6
  150 CRZ(I,J)=CRZ(I,J)+C(K,I)*DUMMY(K,J)
      TT(I)=0.00
      DO 160 M=1,6
      P(M)=0.00
      DO 161 II=1,3
      IF(AOFTS(MTYPE).EQ.1.) P(M)=CNS(M,II)*EE(II+9)
  161 P(M)=P(M)+(T(N)-TREF)*CNS(M,II)*EE(II+9)
      DO 160 K=1,6
```

```
  160 TT(I)=TT(I)+C(K,I)*D(M,K)*P(M)
C
C     FORM QUADRILATERAL STIFFNESS MATRIX
      RRR(5)=(R(I1)+R(J1)+R(K1)+R(L1))/4.
      ZZZ(5)=(Z(I1)+Z(J1)+Z(K1)+Z(L1))/4.
      DO 200 M=1,4
      MM=IX(N,M)
      IF(NPP.NE.0) GO TO 190
      IF(R(MM).EQ.0..AND.CODE(MM).EQ.0.)CODE(MM)=1.
  190 RRR(M)=R(MM)
  200 ZZZ(M)=Z(MM)
      DO 220 II=1,15
      P1(II)=0.0
      P1(II+15) =0.0
      P(II)=0.00
      DO 220 JJ=1,15
  220 S(II,JJ)=0.00
      VOL=0.
      DO 90 I=1,6
      DO 90 J=1,15
      BS1(I,J)=0.0
       BS1(I,J+15) = 0.0
90    BS(I,J)=0.00
      AR=0.00
  240 CALL TRISTF(4,1,5)
      CALL TRISTF(1,2,5)
      CALL TRISTF(2,3,5)
      CALL TRISTF(3,4,5)
      DO 91 I=1,6
      DO 91 J=1,15
91    BS(I,J)=BS(I,J)/AR
       DO 300 I=1,30
       DO 300 J=1,30
  300   S1(I,J)=0.0
       AR1 =0.0
       CALL NAXSTF(4,1,5)
       CALL NAXSTF(1,2,5)
       CALL NAXSTF(2,3,5)
       CALL NAXSTF(3,4,5)
       DO 310 I=1,6
       DO 310 J=1,30
  310  BS1(I,J)= BS1(I,J)/AR1
      DO 320 I=1,6
       DO 320 J=1,3
  320 BS1T(I,J) = BS1(I,J+12)
      DO 325 I=1,6
      DO 325 J=1,12
  325 BS1(I,J+12) = BS1(I,J+15)
      DO 330 I=1,6
      DO 330 J=1,3
```

```
330 BS1(I,J+24) = BS1T(I,J)
    DO 340 I=1,3
340 P1T (I) = P1(I+12)
    DO 341 I=1,12
341 P1(I+12) = P1(I+15)
    DO 342 I=1,3
342 P1(I+24) = P1T(I)
     DO 149  I=1,3
     DO 149  J=1,30
149   S1TEM(I,J) = S1(I+12,J)
    DO 151 I=1,12
    DO 151 J=1,30
151 S1(I+12,J) = S1(I+15,J)
    DO 152 I=1,3
    DO 152 J=1,30
152 S1(I+24,J) = S1TEM(I,J)
    DO 153 I=1,3
    DO 153 J=1,30
153 S1TEM(I,J) = S1(J,I+12)
    DO 154 J=1,12
    DO 154 I=1,30
154 S1(I,J+12) = S1(I,J+15)
    DO 155 I=1,3
    DO 155 J=1,30
155 S1( J,I+24) = S1TEM(I,J)
     DO 251 I=1,6
     DO 251 J=1,24
251 TS(I,J) = 0.0
     DO 252 I=1,3
     DO 252 J=1,4
     TS(I,I+(J-1)*3) = 0.250
252 TS(I+3,I+12+(J-1)*3) = 0.250
     DO 253 I=1,24
     DO 253 J=1,24
     S1T(I,J) = 0.00
     DO 253 K=1,6
253  S1T(I,J) = S1T(I,J) + S1(I,24+K)*TS(K,J)
     DO 254 I=1,24
     DO 254 J=1,24
254  S1(I,J) = S1(I,J) + S1T(I,J) + S1T(J,I)
     DO 255 I=1,24
     DO 255 J=1,6
     S2T(I,J) = 0.0
     DO 255  K=1,6
255  S2T(I,J) = S2T(I,J) +TS(K,I)*S1(K+24,J+24)
     DO 256  I=1,24
     DO 256  J=1,24
      S1T(I,J)  = 0.0
      DO 256  K=1,6
256  S1T(I,J) = S1T(I,J) + S2T(I,K)*TS(K,J)
```

```
      DO 257  I=1,24
      DO 257  J =1,24
  257   S1(I,J)  =S1(I,J)+S1T(I,J)
      RETURN
      END
      SUBROUTINE SOLV
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200)
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/NONAX1/S1(30,30),P1(30),THETA,BS1(6,30)
      COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS,GTS1G(24,24,4)
      COMMON/SOLVE/B(162),A(162,81),NUMBLK,MBAND
      MM=MBAND
      NN=81
      NL=NN+1
      NH=NN+NN
      REWIND 1
      REWIND 2
      NB=0
      GO TO 150
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     REDUCE EQUATIONS BY BLOCKS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C     1. SHIFT BLOCK OF EQUATIONS
C
  100 NB=NB+1
      DO 125 N=1,NN
      NM=NN+N
      B(N)=B(NM)
      B(NM)=0.00
      DO 125 M=1,MM
      A(N,M)=A(NM,M)
  125 A(NM,M)=0.00
C
C     2. READ NEXT BLOCK OF EQUATIONS INTO CORE
C
      IF(NUMBLK.EQ.NB) GO TO 200
  150 READ(2) (B(N),(A(N,M),M=1,MM),N=NL,NH)
      IF(NB.EQ.0) GO TO 100
C
C     3. REDUCE BLOCK OF EQUATIONS
C
  200 DO 300 N=1,NN
      IF(A(N,1).EQ.0.00) GO TO 300
      B(N)=B(N)/A(N,1)
      DO 275 L=2,MM
      IF(A(N,L).EQ.0.00) GO TO 275
```

```
      C=A(N,L)/A(N,1)
      I=N+L-1
      J=0
      DO 250 K=L,MM
      J=J+1
  250 A(I,J)=A(I,J)-C*A(N,K)
      B(I)=B(I)-A(N,L)*B(N)
      A(N,L)=C
  275 CONTINUE
  300 CONTINUE
C
C     4. WRITE BLOCK OF REDUCED EQUATIONS ON FORTRAN UNIT 1
C
      IF(NUMBLK.EQ.NB) GO TO 400
      WRITE (1) (B(N),(A(N,M),M=2,MM),N=1,NN)
      GO TO 100
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     BACK-SUBSTITUTION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  400 DO 450 M=1,NN
      N=NN+1-M
      DO 425 K=2,MM
      L=N+K-1
  425 B(N)=B(N)-A(N,K)*B(L)
      NM=N+NN
      B(NM)=B(N)
  450 A(NM,NB)=B(N)
      NB=NB-1
      IF(NB.EQ.0) GO TO 500
      BACKSPACE 1
      READ (1) (B(N),(A(N,M),M=2,MM),N=1,NN)
      BACKSPACE 1
      GO TO 400
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ORDER FORMER UNKNOWNS IN B ARRAY
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  500 K=0
      DO 600 NB=1,NUMBLK
      DO 600 N=1,NN
      NM=N+NN
      K=K+1
  600 B(K)=A(NM,NB)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     WRITE SOLUTION
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
       NN12 = 3*NUMNP
 1500 FORMAT(" ",5I10)
       WRITE(26) (B(I),I=1,NN12)
       WRITE(26)((IX(I,J),J=1,4),I=1,NUMEL)
      MPRINT=0
```

```
      DO 710 N=1,NUMNP
      IF(MPRINT.NE.0) GO TO 700
      WRITE (6,2000)
      MPRINT=59
  700 MPRINT=MPRINT-1
  710 WRITE (6,2001) N,B(3*N-2),B(3*N-1),B(3*N)
 2000 FORMAT (13H1 NODAL POINT,18X,2HUR,18X,2HUZ,18X,2HUT)
 2001 FORMAT (I13,3E20.7)
      RETURN
      END
      SUBROUTINE STIFF
      INTEGER CODE
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200)
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/SOLVE/B(162),A(162,81),NUMBLK,MBAND
      COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS,GTS1G(24,24,4)
       COMMON/ANS4/FT(24,4),GTS1U(24),GTS1UT(24,4)
      COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
     1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
      COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
      COMMON/PLANE/NPP
      COMMON/ANS2/GTP1(24),G(24,24),GTS1(24,24),GTS1GE(24,24)
      COMMON/DUM1/S1TEM(3,30),S1T(24,24),TS(6,24)
      DIMENSION LM(4),S2(12,3),S3(3,12),S4(3,3),S5(12,3),S6(12,12)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZATION
      REWIND 2
        REWIND 3
      NB=27
      ND=3*NB
      ND2=2*ND
      STOP=0.
      NUMBLK=0
      DO 100 N=1,ND2
      B(N)=0.00
      DO 100 M=1,ND
  100 A(N,M)=0.00
       DO 50 I=1,24
       FT(I,NTP) = 0.0
       GTS1UT(I,NTP)=0.0
       DO 50 J=1,24
   50  GTS1G(I,J,NTP) = 0.0
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FORM STIFFNESS MATRIX IN BLOCKS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
200 NUMBLK=NUMBLK+1
    NH=NB*(NUMBLK+1)
    NM=NH-NB
    NL=NM-NB+1
    KSHIFT=3*NL-3
    DO 340 N=1,NUMEL
    IF(IX(N,5).LE.0) GO TO 340
    DO 210 I=1,4
    IF(IX(N,I).LT.NL) GO TO 210
    IF(IX(N,I).LE.NM) GO TO 220
210 CONTINUE
    GO TO 340
220 CALL QUAD
    IF(VOL.GT.0.) GO TO 230
    WRITE(6,2000) N
    STOP=1.
230 IF(IX(N,3).EQ.IX(N,4)) GO TO 300
    DO 231 II=1,3
    DO 231 JJ=1,3
231 S4(II,JJ)=S(II+12,JJ+12)
    CALL SYMINV(S4,3)
    DO 232 II=1,12
    DO 232 JJ=1,3
232 S2(II,JJ)=S(II,JJ+12)
    DO 233 II=1,3
    DO 233 JJ=1,12
233 S3(II,JJ)=S(II+12,JJ)
    DO 240 I=1,12
    DO 240 J=1,3
    S5(I,J)=0.00
    DO 240 K=1,3
240 S5(I,J) = S5(I,J) + S2(I,K) * S4(K,J)
    DO 241 I=1,12
    DO 241 J=1,12
    S6(I,J)=0.00
    DO 241 K=1,3
241 S6(I,J) = S6(I,J) + S5(I,K) * S3(K,J)
    DO 234 II=1,12
    DO 234 JJ=1,3
234 P(II)=P(II)-S5(II,JJ)*P(JJ+12)
    DO 235 II=1,12
    DO 235 JJ=1,12
235 S(II,JJ)=S(II,JJ)-S6(II,JJ)
    DO 259 I=1,24
    DO 259 J=1,24
259 G(I,J) = 0.0
    DO 260 K=1,4
    DO 260 I=1,3
     G(K*3-3+I,I*4-3) =1.0
     G(K*3-3+I,I*4-2) = RRR(K)
```

```
      G(K*3-3+I,I*4-1) = ZZZ(K)
  260 G(K*3-3+I,I*4  ) = ZZZ(K)   *RRR(K)
       DO 262 I=1,12
       DO 262 J=1,12
  262 G(I+12,J+12) = G(I,J)
      NTP20 = NTP+20
       WRITE(NTP20) (( CRZ(I,J),J=1,6),I=1,6)
       WRITE(NTP20)((BS1(I,J),J=1,30),I=1,6)
       WRITE(NTP20)((G(I,J),J=1,24),I=1,24)
      DO 280 I=1,24
       GTP1(I)=0.0
      DO 280 K=1,24
      GTS1(I,K) = 0.0
      GTP1(I)= GTP1(I)+ G(K,I)*P1(K)
      DO 280 J=1,24
  280 GTS1(I,K) = GTS1(I,K) + G(J,I) * S1(J,K)
       WRITE(3) ((GTS1(I,J),J=1,24),I=1,24)
      DO 281 I=1,24
       FT(I,NTP) =FT(I,NTP) + GTP1(I)
      DO 281 J=1,24
      GTS1GE(I,J) = 0.0
      DO 281 K=1,24
  281 GTS1GE(I,J) = GTS1GE(I,J)+ GTS1(I,K) *G(K,J)
       DO 282 I=1,24
       DO 282 J=1,24
  282 GTS1G(I,J,NTP) = GTS1G(I,J,NTP) + GTS1GE(I,J)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ADD ELEMENT STIFFNESS MATRIX TO BODY STIFFNESS MATRIX
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  300 DO 310 I=1,4
310   LM(I)=3*IX(N,I)-3
      DO 330 I=1,4
      DO 330 K=1,3
      II=LM(I)+K-KSHIFT
      KK=3*I-3+K
      B(II)=B(II)+P(KK)
      DO 330 J=1,4
      DO 330 L=1,3
      JJ=LM(J)+L-II+1-KSHIFT
      LL=3*J-3+L
      IF(JJ.LE.0) GO TO 330
      IF(ND.GE.JJ) GO TO 320
      WRITE(6,2001) N
      STOP=1.
      GO TO 340
  320 A(II,JJ)=A(II,JJ)+S(KK,LL)
  330 CONTINUE
  340 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ADD CONCENTRATED FORCES
```

```
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 400 N=NL,NM
      IF(N.GT.NUMNP) GO TO 500
      K=3*N-KSHIFT
      B(K)=B(K)+XT(N)
      B(K-1)=B(K-1)+XZ(N)
  400 B(K-2)=B(K-2)+XR(N)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ADD PRESSURE BOUNDARY CONDITIONS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  500 IF(NUMPC.EQ.0) GO TO 600
      DO 540 L=1,NUMPC
      I=IP(L)
      J=JP(L)
      PP=PR(L)/6.
      DR=(R(J)-R(I))*PP
      DZ=(Z(I)-Z(J))*PP
      RX=2.*R(I)+R(J)
      ZX=R(I)+2.*R(J)
      II=3*I-KSHIFT-1
      JJ=3*J-KSHIFT-1
      IF(II.LE.0.OR.II.GT.ND) GO TO 520
      SINA=0.
      COSA=1.
  510 B(II-1)=B(II-1)+RX*(COSA*DZ+SINA*DR)
       GR=RX*(COSA*DZ+SINA*DR)*THETA/2.0
       FT(1,NTP) = FT(1,NTP)+GR
       FT(2,NTP) = FT(2,NTP) +R(I)*GR
       FT(3,NTP) = FT(3,NTP) +Z(I)*GR
       FT(4,NTP) = FT(4,NTP) + R(I)*Z(I)*GR
       FT(14,NTP)= FT(14,NTP) +R(I)*GR
       FT(13,NTP)  = FT(13,NTP) +GR
       FT(15,NTP) = FT(15,NTP)+Z(I)*GR
       FT(16,NTP) = FT(16,NTP) +Z(I)*R(I)*GR
      B(II)=B(II)-RX*(SINA*DZ-COSA*DR)
      GZ=-RX*(SINA*DZ-COSA*DR) *THETA/2.0
       FT(5,NTP)=FT(5,NTP)+GZ
       FT(6,NTP)=FT(6,NTP)+R(I)*GZ
       FT(7,NTP)=FT(7,NTP)+ Z(I)*GZ
       FT(8,NTP)=FT(8,NTP)+Z(I)*R(I)*GZ
       FT(17,NTP)=FT(17,NTP)+GZ
       FT(18,NTP)=FT(18,NTP)+R(I)*GZ
       FT(19,NTP)=FT(19,NTP)+Z(I)*GZ
       FT(20,NTP)=FT(20,NTP)+Z(I)*R(I)*GZ
  520 IF(JJ.LE.0.OR.JJ.GT.ND) GO TO 540
      SINA=0.
      COSA=1.
  530 B(JJ-1)=B(JJ-1)+ZX*(COSA*DZ+SINA*DR)
       GR= ZX *(COSA*DZ+SINA*DR)  *THETA/2.0
       FT(1,NTP)=FT(1,NTP)+GR
```

```
      FT(2,NTP)=FT(2,NTP)+R(J)*GR
      FT(3,NTP)=FT(3,NTP)+Z(J)*GR
      FT(4,NTP)=FT(4,NTP)+Z(J)*R(J)*GR
      FT(13,NTP)=FT(13,NTP)+GR
      FT(14,NTP)=FT(14,NTP)+R(J)*GR
      FT(15,NTP)=FT(15,NTP)+Z(J)*GR
      FT(16,NTP)=FT(16,NTP)+Z(J)*R(J)*GR
     B(JJ)=B(JJ)-ZX*(SINA*DZ-COSA*DR)
       GZ= -ZX*(SINA*DZ-COSA*DR) *THETA/2.0
      FT(5,NTP)=FT(5,NTP)+GZ
      FT(6,NTP)=FT(6,NTP)+R(J)*GZ
      FT(7,NTP)=FT(7,NTP) +Z(J)*GZ
      FT(8,NTP)=FT(8,NTP)+Z(J)*R(J)*GZ
      FT(17,NTP)=FT(17,NTP)+GZ
      FT(18,NTP)=FT(18,NTP)+R(J)*GZ
      FT(19,NTP)=FT(19,NTP)+Z(J)*GZ
      FT(20,NTP)=FT(20,NTP)+Z(J)*R(J)*GZ
  540 CONTINUE
 1100  FORMAT(" ",12E10.3)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ADD SHEAR BOUNDARY CONDITIONS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  600 IF(NUMSC.EQ.0) GO TO 701
      DO 640 L=1,NUMSC
      I=IS(L)
      J=JS(L)
      SS=SH(L)/6.
      DZ=(Z(I)-Z(J))*SS
      DR=(R(J)-R(I))*SS
      RX=2.*R(I)+R(J)
      ZX=R(I)+2.*R(J)
      II=3*I-KSHIFT-1
      JJ=3*J-KSHIFT-1
      IF(II.LE.0.OR.II.GT.ND) GO TO 620
      SINA=0.
      COSA=1.
  610 B(II-1)=B(II-1)+RX*(SINA*DZ+COSA*DR)
       GR= RX*(SINA*DZ+COSA*DR) *THETA/2.0
       FT(1,NTP)=FT(1,NTP)+GR
       FT(2,NTP)=FT(2,NTP)+R(I)*GR
       FT(3,NTP) =FT(3,NTP)+Z(I)*GR
       FT(4,NTP)=FT(4,NTP)+Z(I)*R(I)*GR
       FT(13,NTP)=FT(13,NTP)+GR
       FT(14,NTP)=FT(14,NTP)+R(I)*GR
       FT(15,NTP)=FT(15,NTP)+Z(I)*GR
       FT(16,NTP)=FT(16,NTP)+Z(I)*R(I)*GR
      B(II)=B(II)-RX*(COSA*DZ-SINA*DR)
       GZ= -RX*(COSA*DZ-SINA*DR) *THETA/2.0
       FT(5,NTP)=FT(5,NTP)+GZ
        FT(6,NTP)=FT(6,NTP)+R(I)*GZ
```

```
      FT(7,NTP)=FT(7,NTP)+Z(I)*GZ
      FT(8,NTP)=FT(8,NTP)+Z(I)*R(I)*GZ
      FT(17,NTP)=FT(17,NTP)+GZ
      FT(18,NTP)=FT(18,NTP)+R(I)*GZ
      FT(19,NTP)=FT(19,NTP)+Z(I)*GZ
      FT(20,NTP)=FT(20,NTP)+Z(I)*R(I)*GZ
620   IF(JJ.LE.0.OR.JJ.GT.ND) GO TO 640
      SINA=0.
      COSA=1.
630   B(JJ-1)=B(JJ-1)+ZX*(SINA*DZ+COSA*DR)
      GR= ZX*(SINA*DZ+COSA*DR)*THETA/2.0
      FT(1,NTP)=FT(1,NTP)+GR
      FT(2,NTP)=FT(2,NTP)+R(J)*GR
      FT(3,NTP)=FT(3,NTP)+Z(J)*GR
      FT(4,NTP)=FT(4,NTP)+Z(J)*R(J)*GR
      FT(13,NTP)=FT(13,NTP)+GR
      FT(14,NTP)=FT(14,NTP)+R(J)*GR
      FT(15,NTP)=FT(15,NTP)+Z(J)*GR
      FT(16,NTP)=FT(16,NTP)+Z(J)*R(J)*GR
      B(JJ)=B(JJ)-ZX*(COSA*DZ-SINA*DR)
      GZ= -ZX*(COSA*DZ-SINA*DR)*THETA/2.0
      FT(5,NTP)=FT(5,NTP)+GZ
      FT(6,NTP)=FT(6,NTP)+R(J)*GZ
      FT(7,NTP)=FT(7,NTP)+Z(J)*GZ
      FT(8,NTP)=FT(8,NTP)+Z(J)*R(J)*GZ
      FT(17,NTP)=FT(17,NTP)+GZ
      FT(18,NTP)=FT(18,NTP)+R(J)*GZ
      FT(19,NTP)=FT(19,NTP)+Z(J)*GZ
      FT(20,NTP)=FT(20,NTP)+Z(J)*R(J)*GZ
640   CONTINUE
701   IF(NUMST.EQ.0) GO TO 700
      DO 680 L=1,NUMST
      I=IT(L)
      J=JT(L)
      RT=ST(L)/6.
      RX=2.*R(I)+R(J)
      ZX=R(I)+2.*R(J)
      XX=SQRT((R(J)-R(I))**2+(Z(J)-Z(I))**2)
      II=3*I-KSHIFT
      JJ=3*J-KSHIFT
      IF(II.LE.0.OR.II.GT.ND) GO TO 670
      B(II)=B(II)+RT*RX*XX
      GT=RT*RX*XX*THETA/2.0
      FT(9,NTP)=FT(9,NTP)+GT
      FT(10,NTP)=FT(10,NTP)+R(I)*GT
      FT(11,NTP)=FT(11,NTP)+Z(I)*GT
      FT(12,NTP)=FT(12,NTP)+Z(I)*R(I)*GT
      FT(21,NTP)=FT(21,NTP)+GT
      FT(22,NTP)=FT(22,NTP)+R(I)*GT
      FT(23,NTP)=FT(23,NTP)+Z(I)*GT
```

```
      FT(24,NTP)=FT(24,NTP)+Z(I)*R(I)*GT
  670 IF(JJ.LE.0.OR.JJ.GT.ND) GO TO 680
      B(JJ)=B(JJ)+RT*ZX*XX
      GT=RT*ZX*XX*THETA/2.0
      FT(9,NTP)=FT(9,NTP)+GT
      FT(10,NTP)=FT(10,NTP)+R(J)*GT
      FT(11,NTP)=FT(11,NTP)+Z(J)*GT
      FT(12,NTP)=FT(12,NTP)+Z(J)*R(J)*GT
      FT(21,NTP)=FT(21,NTP)+GT
      FT(22,NTP)=FT(22,NTP)+R(J)*GT
      FT(23,NTP)=FT(23,NTP)+Z(J)*GT
      FT(24,NTP)=FT(24,NTP)+Z(J)*R(J)*GT
  680 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     ADD DISPLACEMENT BOUNDARY CONDITIONS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  700 DO 750 M=NL,NH
      IDM=0
      IF(M.GT.NUMNP) GO TO 750
      IF(CODE(M).GT.3) GO TO 751
      U=XR(M)
      N=3*M-2-KSHIFT
  752 IF(CODE(M)) 740,750,710
  710 IF(CODE(M).EQ.1) GO TO 720
      IF (CODE(M).EQ.2) GO TO 740
      IF(CODE(M).EQ.3) GO TO 730
      GO TO 740
  720 CALL MODIFY(ND2,N,U)
      CODE(M)=CODE(M)+IDM
      GO TO 750
  730 CALL MODIFY(ND2,N,U)
  740 U=XZ(M)
      N=N+1
      CALL MODIFY(ND2,N,U)
      CODE(M)=CODE(M)+IDM
      GO TO 750
  751 IDM=IDM+4
      U=XT(M)
      N=3*M-KSHIFT
      CALL MODIFY(ND2,N,U)
      U=XR(M)
      N=3*M-2-KSHIFT
      IF(CODE(M).EQ.4) GO TO 750
      CODE(M)=CODE(M)-4
      GO TO 752
  750 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     WRITE BLOCK OF EQUATIONS ON FORTRAN UNIT AND SHIFT UP LOWER BLOCK
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      WRITE (2) (B(N),(A(N,M),M=1,MBAND),N=1,ND)
```

```
      DO 800 N=1,ND
      K=N+ND
      B(N)=B(K)
      B(K)=0.00
      DO 800 M=1,ND
      A(N,M)=A(K,M)
  800 A(K,M)=0.00
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     CHECK FOR LAST BLOCK
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(NM.LT.NUMNP) GO TO 200
      IF(STOP.NE.0.) STOP
 2000 FORMAT (27H  NEGATIVE AREA ELEMENT NO.,I4)
 2001 FORMAT (46H  BAND WIDTH EXCEEDS ALLOWABLE FOR ELEMENT NO.,I4)
      RETURN
      END
      SUBROUTINE STORE
      INTEGER CODE
      COMMON/ANS4/ FT(24,4),GTS1U(24),GTS1UT(24,4)
      COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS,GTS1G(24,24,4)
      COMMON/NXMESH/THETAN(4),NST(4),NUMS(4,5),NPC(8,8)
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/SOLVE/B(162),A(162,81),NUMBLK,MBAND
      COMMON/GLBSEG/FI(24,8),FE(24,8),UC(24,8),SK(24,24,8)
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/ANS2/LW(24),R1(24,24),SK1(24,24),DUMM(24,24)
      DIMENSION MW(24)
      JJ =NST(NTP)
      DO 100 L=1,JJ
       DO 50 I=1,24
        DO 50 J=1,24
   50  R1(I,J) = 0.0
      NS = NUMS(NTP,L)
      DO 110 KK = 1,4
      NP1 = NPC(NS,KK)
      NP2 = NPC(NS,KK+4)
      DO 110 I= 1,3
      R1(3*(KK-1)+I   ,I*4-3 ) = 1.0
      R1(3*(KK-1)+I   ,I*4-2 ) = R(NP1)
      R1(3*(KK-1)+I   ,I*4-1 ) = Z(NP1)
      R1(3*(KK-1)+I   ,I*4   ) = R(NP1) * Z(NP1)
      R1(3*(KK-1)+I+12,I*4+9 ) = 1.0
      R1(3*(KK-1)+I+12,I*4+10) = R(NP2)
      R1(3*(KK-1)+I+12,I*4+11) = Z(NP2)
      R1(3*(KK-1)+I+12,I*4+12) = R(NP2)* Z(NP2)
      UC(3*(KK-1)+I,NS) = B(3*NP1-3+I)
      UC(3*(KK-1)+I+12,NS) = B(3*NP2-3+I)
  110 CONTINUE
```

```
       CALL MINV(R1,24,D1,LW,MW)
      WRITE(25) ((R1(I,J),J=1,24),I=1,24)
      DO 115 I=1,24
      FE(I,NS)=0.0
       FI(I,NS)=0.0
      DO 115 J=1,24
       FE(I,NS) = FE(I,NS)+ R1(J,I)*FT(J,NTP)
       FI(I,NS) = FI(I,NS) + R1(J,I)*GTS1UT(J,NTP)
       SK1(I,J) = 0.00
      DO 115 K=1,24
      SK1(I,J) = SK1(I,J) + R1(K,I) * GTS1G(K,J,NTP)
  115 CONTINUE
      DO 120  I=1,24
      DO 120  J=1,24
      SK(I,J,NS) =0.0
      DO 120  K=1,24
      SK(I,J,NS) = SK(I,J,NS) + SK1(I,K) * R1(K,J)
  120 CONTINUE
C     WRITE(6,1000)(FE(I,NS),I=1,24)
C     WRITE(6,1000)(FI(I,NS),I=1,24)
C     WRITE(6,1000)((SK(I,J,NS),J=1,24),I=1,24)
C     WRITE(6,1000)(UC(I,NS),I=1,24)
  100 CONTINUE
 1000  FORMAT(" ",12E10.3)
      RETURN
      END
      SUBROUTINE STRESS
      INTEGER CODE
       COMMON/ANS4/FT(24,4),GTS1U(24) ,GTS1UT(24,4)
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200)
      COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
     1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
      COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
      COMMON/NXDATA/NTP,NTYPS,NTS,NTOTS,GTS1G(24,24,4)
      COMMON/NXMESH/THETAN(4),NST(4),NUMS(4,5),NPC(8,8)
      COMMON/ARG1/SIG1(18),EPS1(18)
      COMMON/SOLVE/B(162),A(162,81),NUMBLK,MBAND
      COMMON/CONVRG/IDONE
      COMMON/PLANE/NPP
      COMMON/RESULT/BS(6,15),D(6,6),C(6,6),AR,BB(6,9),CNS(6,6)
      DIMENSION LM(4),TP(6),TR(3,3),Q(3)
      DIMENSION QQ(3)
      COMMON/DUM1/S1TEM(3,30),GTS1(24,24),TS(6,24)
```

```
       DIMENSION P11(24)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
       REWIND 3
      XKE=0.
      XPE=0.
      MPRINT=0
      ERROR=.005
      IDONE=1
      DO 200 N=1,NUMEL
      IX(N,5)=IABS(IX(N,5))
C     CALL QUAD
      MTYPE=IABS(IX(N,5))
      DO 100 I=1,4
      II=3*I
      JJ=3*IX(N,I)
      P1(II-2) = B(JJ-2)
      P1(II-1) = B(JJ-1)
      P1(II  ) = B(JJ  )
      P1(II+10) = B(JJ-2)
      P1(II+11) = B(JJ-1)
      P1(II+12) = B(JJ)
      P(II-2)=B(JJ-2)
      P(II-1)=B(JJ-1)
100   P(II)  =B(JJ)
       READ(3)((GTS1(I,J),J=1,24),I=1,24)
       DO 115 I=1,24
       GTS1U(I)=0.0
       DO 115 J=1,24
  115  GTS1U(I) = GTS1U(I)+ GTS1(I,J)*P1(J)
       DO 116 I=1,24
  116  GTS1UT(I,NTP)=GTS1UT(I,NTP) + GTS1U(I)
       GO TO 200
      DO 110 I=1,3
110   Q(I)=P(I+12)
      DO 120 I=1,3
      DO 120 J=1,3
120   TR(I,J)=S(I+12,J+12)
      CALL SYMINV(TR,3)
      DO 125 J=1,3
      QQ(J)=0.00
      DO 125 K=1,12
      QQ(J)=QQ(J)+S(J+12,K)*P(K)
 125  CONTINUE
      DO 130 I=1,3
      P(I+12)=0.00
      DO 130 J=1,3
 130  P(I+12)=P(I+12)+TR(I,J)*(Q(J)-QQ(J))
 500  CONTINUE
```

```
C
C     MATRIX P NOW CONTAINS 15 DISPLACEMENTS FOR QUADRILATERAL ELEMENT
C
C     CALCULATE AVERAGE STRAINS
C
      DO 140 I=1,6
      EPS(I)=0.00
      DO 140 J=1,15
140   EPS(I)=EPS(I)+BS(I,J)*P(J)
C
C     CALCULATE AVERAGE STRESSES
C
      DO 151 I=1,6
      SIG(I)=0.00
      DO 151 J=1,6
151   SIG(I)=SIG(I)+CRZ(I,J)*EPS(J)
      DO 152 I=1,6
 152  SIG(I)=SIG(I)-TT(I)
C
C     CALCULATE STRAINS  IN N-S-T COORDINATES
C
      DO 150 I=1,6
      EPS(I+6)=0.00
      DO 150 J=1,6
      DO 150 K=1,6
150   EPS(I+6)=EPS(I+6)+D(I,J)*C(J,K)*EPS(K)
C
C     CALCULATE STRESSES IN N-S-T COORDIATES
C
      DO 160 I=1,6
      SIG(I+6)=0.00
      DO 160 J=1,6
160   SIG(I+6)=SIG(I+6)+CNS(I,J)*EPS(J+6)
      DO 161 M=1,6
      P(M)=0.00
      DO 161 II=1,3
      IF(AOFTS(MTYPE).EQ.1.) P(M)=CNS(M,II)*EE(II+9)
  161 P(M)=P(M)+(T(N)-TREF)*CNS(M,II)*EE(II+9)
      DO 162 I=1,6
 162  SIG(I+6)=SIG(I+6)-P(I)
C
C
      DO 300 I=1,12
300   EPS(I)=100.0*EPS(I)
      IF(MPRINT.NE.0) GO TO 210
      WRITE(6,2000)
      WRITE(6,2002)
      MPRINT=19
  210 MPRINT=MPRINT-1
      WRITE(6,2001) N,RRR(5),ZZZ(5),(SIG(I),I=1,12)
```

```
      WRITE(6,2003) T(N),(EPS(I),I=1,12)
  200 CONTINUE
 2000 FORMAT(129H1    EL      R      Z      SIGMAR    SIGMAZ    SIGMAC    SIGMA
     1RZ  SIGMAZC  SIGMACR  SIGMAN   SIGMAS   SIGMAT   SIGMANS  SIGMAST
     2 SIGMATN)
 2001 FORMAT(1H0,I5,1X,2F7.4,12F9.0)
 2002 FORMAT(128H0        TEMPERATURE     EPSR      EPSZ      EPSC      EPSR
     1Z     EPSZC     EPSCR     EPSN      EPSS      EPST      EPSNS     EPSST
     2  EPSTN)
 2003 FORMAT(6X,F13.0,2X,12F9.5)
      RETURN
      END
      SUBROUTINE SYMINV(A,NMAX)
      DIMENSION A(NMAX,NMAX)
      DO 300 N=1,NMAX
      D=A(N,N)
      DO 100 J=1,NMAX
  100 A(N,J)=-A(N,J)/D
      DO 210 I=1,NMAX
      IF(N.EQ.I) GO TO 210
      DO 200 J=1,NMAX
      IF(N.NE.J) A(I,J)=A(I,J)+A(I,N)*A(N,J)
  200 CONTINUE
  210 A(I,N)=A(I,N)/D
  300 A(N,N)=1.00/D
      RETURN
      END
      SUBROUTINE TEMP(R,Z,T)
      COMMON/SOLVE/X(4428),Y(4428),TEM(4428),NUMTC,MBAND
      DIMENSION SMALL(20),ISM(20)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     INITIALIZE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      J=1
      JMAX=16
      IF(NUMTC.LT.JMAX) JMAX=NUMTC
      DO 10 I=1,JMAX
      SMALL(I)=0.
   10 ISM(I)=0
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND THE JMAX CLOSEST POINTS
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DO 50 I=1,NUMTC
      DSQ=(X(I)-R)**2+(Y(I)-Z)**2
      IF(DSQ.GT..1E-4) GO TO 20
      T=TEM(I)
      RETURN
   20 IF(I.EQ.1) SMALL(1)=DSQ
      IF(I.EQ.1) ISM(1)=1
      IF(I.EQ.1) GO TO 50
```

```
      IF(SMALL(J).LE.DSQ.AND.J.LT.JMAX) SMALL(J+1)=DSQ
      IF(SMALL(J).LE.DSQ.AND.J.LT.JMAX) ISM(J+1)=I
      IF(SMALL(J).LE.DSQ) GO TO 40
      DO 30 K=1,J
      JB=J-K +1
      IF(JB.EQ.0) GO TO 40
      SMALL(JB+1)=SMALL(JB)
      ISM(JB+1)=ISM(JB)
      SMALL(JB)=DSQ
      ISM(JB)=I
      IF(JB.EQ.1) GO TO 40
      IF(SMALL(JB-1).LE.DSQ) GO TO 40
   30 CONTINUE
   40 IF(J.LT.JMAX) J=J+1
   50 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND THE THIRD TEMPERATURE POINT BY AREA TEST
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      JCHK=JMAX-2
      J=0
      I1=ISM(1)
      I2=ISM(2)
   60 I3=ISM(J+3)
      AREA=.50*(Y(I1)*X(I3)-Y(I3)*X(I1)+Y(I3)*X(I2)-Y(I2)*X(I3)+
     1       Y(I2)*X(I1)-Y(I1)*X(I2))
      D1=(X(I2)-X(I1))**2+(Y(I2)-Y(I1))**2
C     IF D1 IS APPROXIMATELY 0. IT IS ASSUMED THAT THERE EXISTS A
C     DUPLICATION OF INPUT
      IF(D1.GT..1E-3) GO TO 70
      I2=I3
      J=J+1
      GO TO 60
   70 IF(AREA**2.GT..1*D1*SMALL(1)) GO TO 80
      J=J+1
      IF(J.LT.JCHK) GO TO 60
      WRITE(6,2000) I1,I2,I3,J
      T=TEM(I1)
      RETURN
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     FIND TEMPERATURE INTERCEPT
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
   80 DETA=Y(I1)*(TEM(I3)-TEM(I2))+Y(I2)*(TEM(I1)-TEM(I3))
     1     +Y(I3)*(TEM(I2)-TEM(I1))
      DETB=X(I1)*(TEM(I2)-TEM(I3))+X(I2)*(TEM(I3)-TEM(I1))
     1     +X(I3)*(TEM(I1)-TEM(I2))
      DETC=TEM(I1)*(X(I2)*Y(I3)-X(I3)*Y(I2))+TEM(I2)*(X(I3)*Y(I1)-X(I1)*
     1Y(I3))+TEM(I3)*(X(II)*Y(I2)-X(I2)*Y(II))
      T=(DETA*R+DETB*Z+DETC)/(2.*AREA)
 2000 FORMAT (28H  ERROR IN TEMPERATURE INPUT,5H  I1=I4,5H  I2=I4,
     15H  I3=I4,4H  J=I4)
```

```
      RETURN
      END
      SUBROUTINE TEM2(NUMNP)
      INTEGER CODE
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      READ(5,1000) TCONST
      DO 100 N=1,NUMNP
  100 T(N)=TCONST
 1000 FORMAT(F10.0)
      RETURN
      END
      SUBROUTINE TRISTF (II,JJ,KK)
      INTEGER CODE
      COMMON/MATP/RO(6),E(12,16,6),EE(16),AOFTS(6)
      COMMON/BASIC/ACELZ,ANGVEL,ANGACC,TREF,VOL,NUMNP,NUMEL,NUMPC,NUMSC,
     1NUMST
      COMMON/ARG/RRR(5),ZZZ(5),RR(4),ZZ(4),S(15,15),P(15),TT(6),
     1H(6,15),CRZ(6,6),XI(10),ANGLE(4),SIG(18),EPS(18),N
      COMMON/NPDATA/R(1000),CODE(1000),XR(1000),Z(1000),XZ(1000),
     1NPNUM(25,80),T(1000),XT(1000)
      COMMON/ELDATA/BETA(1000),EPR(1000),PR(200),SH(200),IX(1000,5),
     1IP(200),JP(200),IS(200),JS(200),ALPHA(1000),IT(200),JT(200),
     2ST(200)
      COMMON/NONAXI/S1(30,30),P1(30),THETA,BS1(6,30)
      COMMON/RESULT/BS(6,15),D(6,6),C(6,6),AR,BB(6,9),CNS(6,6)
      DIMENSION B1A(6,9),B1B(6,9),B2A(6,9),B2B(6,9),B3A(6,9),B3B(6,9)
      DIMENSION B1(6,9),B2(6,9),B3(6,9),F(6,9),G(9,6),V(9,9)
      DIMENSION BF(3),BFR(3),BFZ(3),TP(9),B(9,9)
      MTYPE=IABS(IX(N,5))
      RR(1)=RRR(II)
      RR(2)=RRR(JJ)
      RR(3)=RRR(KK)
      ZZ(1)=ZZZ(II)
      ZZ(2)=ZZZ(JJ)
      ZZ(3)=ZZZ(KK)
      CALL INTER
      VOL=VOL+XI(1)
      COMM=RR(2)*(ZZ(3)-ZZ(1))+RR(1)*(ZZ(2)-ZZ(3))+RR(3)*(ZZ(1)-ZZ(2))
       DO 10 I=1,6
       DO 10 J=1,9
      B1(I,J)=0.00
      B2(I,J)=0.00
   10 B3(I,J)=0.00
C     FILL B1 MATRIX-CONSTANT TERMS
      B1(1,1)=(ZZ(2)-ZZ(3))/COMM
      B1(1,4)=(ZZ(3)-ZZ(1))/COMM
      B1(1,7)=(ZZ(1)-ZZ(2))/COMM
      B1(3,1)=B1(1,1)
      B1(3,4)=B1(1,4)
```

```
      B1(3,7)=B1(1,7)
      B1(2,2)=(RR(3)-RR(2))/COMM
      B1(2,5)=(RR(1)-RR(3))/COMM
      B1(2,8)=(RR(2)-RR(1))/COMM
      B1(4,1)=B1(2,2)
      B1(4,4)=B1(2,5)
      B1(4,7)=B1(2,8)
      B1(4,2)=B1(1,1)
      B1(4,5)=B1(1,4)
      B1(4,8)=B1(1,7)
      B1(5,3)=B1(4,1)
      B1(5,6)=B1(4,4)
      B1(5,9)=B1(4,7)
C     FILL B2 MATRIX-1/R TERMS
      B2(3,1)=(1/COMM)*((ZZ(3)-ZZ(2))*RR(2)+(RR(2)-RR(3))*ZZ(2))
      B2(3,4)=(1/COMM)*((ZZ(1)-ZZ(3))*RR(3)-(RR(1)-RR(3))*ZZ(3))
      B2(3,7)=(1/COMM)*((ZZ(2)-ZZ(1))*RR(1)+(RR(1)-RR(2))*ZZ(1))
      B2(6,3)=-B2(3,1)
      B2(6,6)=-B2(3,4)
      B2(6,9)=-B2(3,7)
C     FILL B3 MATRIX-Z/R TERMS
      B3(3,1)=(RR(3)-RR(2))/COMM
      B3(3,4)=(RR(1)-RR(3))/COMM
      B3(3,7)=(RR(2)-RR(1))/COMM
      B3(6,3)=(RR(2)-RR(3))/COMM
      B3(6,6)=(RR(3)-RR(1))/COMM
      B3(6,9)=(RR(1)-RR(2))/COMM
      AR=AR+XI(1)
      DO 80 I=1,6
      DO 80 J=1,9
80    BB(I,J)=B1(I,J)*XI(1)+B2(I,J)*XI(2)+B3(I,J)*XI(4)
      DO 81 K=1,6
      DO 81 I=1,3
      BS(K,3*JJ-3+I)=BB(K,I+3)+BS(K,3*JJ-3+I)
      BS(K,3*II-3+I)=BB(K,I)+BS(K,3*II-3+I)
81    BS(K,3*KK-3+I)=BB(K,I+6)+BS(K,3*KK-3+I)
      DO 220  I=1,6
       DO 220 J=1,9
      B1A(I,J)=B1(I,J)*XI(1)+B2(I,J)*XI(2)+B3(I,J)*XI(4)
      B2A(I,J)=B1(I,J)*XI(2)+B2(I,J)*XI(3)+B3(I,J)*XI(5)
      B3A(I,J)=B1(I,J)*XI(4)+B2(I,J)*XI(5)+B3(I,J)*XI(6)
  220  CONTINUE
      DO 200 I=1,6
      DO 200 K=1,9
      B1B(I,K)=0.0
      B2B(I,K)=0.0
      B3B(I,K)=0.0
      DO 200 J=1,6
      B1B(I,K)=B1B(I,K)+CRZ(I,J)*B1A(J,K)
      B2B(I,K)=B2B(I,K)+CRZ(I,J)*B2A(J,K)
```

```
      B3B(I,K)=B3B(I,K)+CRZ(I,J)*B3A(J,K)
  200  CONTINUE
      DO 230 I=1,9
      DO 230 K=1,9
      B(I,K)=0.0
      DO 230 J=1,6
      B(I,K)=B(I,K)+B1(J,I)*B1B(J,K)+B2(J,I)*B2B(J,K)+B3(J,I)*B3B(J,K)
  230   CONTINUE
C     ASSEMBLE QUADRILATERAL STIFFNESS MATRIX, S, FROM TRIANGULAR
C     STIFFNESS MATRIX, B.
      IIM=3*II-3
      JJM=3*JJ-3
      KKM=3*KK-3
      DO 120 I=1,3
      DO 120 J=1,3
      S(IIM+I,IIM+J)=B(I   ,J   )+S(IIM+I,IIM+J)
      S(IIM+I,JJM+J)=B(I   ,J+3)+S(IIM+I,JJM+J)
      S(IIM+I,KKM+J)=B(I   ,J+6)+S(IIM+I,KKM+J)
      S(JJM+I,IIM+J)=B(I+3,J   )+S(JJM+I,IIM+J)
      S(JJM+I,JJM+J)=B(I+3,J+3)+S(JJM+I,JJM+J)
      S(JJM+I,KKM+J)=B(I+3,J+6)+S(JJM+I,KKM+J)
      S(KKM+I,IIM+J)=B(I+6,J   )+S(KKM+I,IIM+J)
      S(KKM+I,JJM+J)=B(I+6,J+3)+S(KKM+I,JJM+J)
      S(KKM+I,KKM+J)=B(I+6,J+6)+S(KKM+I,KKM+J)
  120 CONTINUE
C     ASSEMBLE BODY FORCES MATRIX
      BF(1)=(ZZ(3)*RR(2)-RR(3)*ZZ(2))/COMM
      BF(2)=(ZZ(1)*RR(3)-RR(1)*ZZ(3))/COMM
      BF(3)=(ZZ(2)*RR(1)-RR(2)*ZZ(1))/COMM
      BFR(1)=(ZZ(2)-ZZ(3))/COMM
      BFR(2)=(ZZ(3)-ZZ(1))/COMM
      BFR(3)=(ZZ(1)-ZZ(2))/COMM
      BFZ(1)=(RR(3)-RR(2))/COMM
      BFZ(2)=(RR(1)-RR(3))/COMM
      BFZ(3)=(RR(2)-RR(1))/COMM
C     BODY FORCE IN Z-DIRECTION
      COMM=-ACELZ*RO(MTYPE)
      DO 140 I=1,3
      IIK=3*I-1
  140 TP(IIK)=COMM*(BF(I)*XI(1)+BFR(I)*XI(7)+BFZ(I)*XI(8))
C     BODY FORCE IN R-DIRECTION
      COMM=ANGVEL**2*RO(MTYPE)
      DO 150 I=1,3
      L=3*I-2
  150 TP(L)=COMM*(BF(I)*XI(7)+BFR(I)*XI(9)+BFZ(I)*XI(10))
C     BODY FORCES IN TANG. DIRECTION
      COMM=-ANGACC*RO(MTYPE)
      DO 160 I=1,3
      IIM=3*I
  160 TP(IIM)=COMM*(BF(I)*XI(7)+BFR(I)*XI(9)+BFZ(I)*XI(10))
```

```
C     ADD THERMAL EFFECTS
      DO 161 J=1,9
      DO 161 K=1,6
  161 TP(J)=TP(J)+(XI(1)*B1(K,J)+XI(2)*B2(K,J)
     1+XI(4)*B3(K,J))*TT(K)
C     REARRANGE TP INTO P-MATRIX,THE BODY FORCES MATRIX
      K=3*II-2
      L=3*JJ-2
      M=3*KK-2
      DO 170 I=1,3
      J=I-1
      P1(K+J   ) = P1(K+J   )+TP(I  )*THETA/2.0 +TP(I+6)*THETA/4.0
      P1(K+J+15) = P1(K+J+15)+TP(I  )*THETA/2.0 +TP(I+6)*THETA/4.0
      P1(L+J   ) = P1(L+J   )+TP(I+3)*THETA/2.0 +TP(I+6)*THETA/4.0
      P1(L+J+15) = P1(L+J+15)+TP(I+3)*THETA/2.0 +TP(I+6)*THETA/4.0
      P1(M+J   ) = P1(M+J   )+TP(I+6)*THETA/2.0
      P1(M+J+15) = P1(M+J+15)+TP(I+6)*THETA/2.0
      P(K+J)=P(K+J)+TP(I)
      P(L+J)=P(L+J)+TP(I+3)
  170 P(M+J)=P(M+J)+TP(I+6)
      RETURN
      END
      SUBROUTINE XMODFY(U,N)
      COMMON/NXSOLV/SK (132,24),R1 (132),NSZF
       NBAND=24
      DO 10 M=2,NBAND
      K=N-M+1
      IF(K.LE.0) GO TO 5
      R1(K)=R1(K)-SK(K,M)*U
      SK(K,M)=0.
    5 K=N+M-1
      IF(NSZF.LT.K) GO TO 10
      R1(K)=R1(K)-SK(N,M)*U
      SK(N,M)=0.
   10 CONTINUE
      SK(N,1)=1.
      R1(N)=U
      RETURN
      END
       SUBROUTINE XSOLVE
      COMMON/NXSOLV/SK (132,24),R1 (132),NSZF
       NBAND=24
      DO 300 N=1,NSZF
      I=N
      DO 290 L=2,NBAND
      I=I+1
      IF(SK(N,L)) 240,290,240
  240 AC=SK(N,L)/SK(N,1)
      J=0
      DO 270 K=L,NBAND
```

```
      J=J+1
      IF(SK(N,K)) 260,270,260
  260 SK(I,J)=SK(I,J)-AC*SK(N,K)
  270 CONTINUE
  280 SK(N,L)=AC
C
      R1(I)=R1(I)-AC*R1(N)
  290 CONTINUE
  300 R1(N)=R1(N)/SK(N,1)
C
      N=NSZF
  350 N=N-1
      IF(N) 500,500,360
  360 L=N
      DO 400 K=2,NBAND
      L=L+1
      IF(SK(N,K)) 370,400,370
  370 R1(N)=R1(N)-SK(N,K)*R1(L)
  400 CONTINUE
      GO TO 350
C
  500 RETURN
      END
      END
^N
```

DISTRIBUTION LIST

No. of Copies	Organization
12	Commander Defense Technical Info Center ATTN: DDC-DDA Cameron Station Alexandria, VA 22314
1	Director Defense Advanced Research Projects Agency ATTN: Tech Info 1400 Wilson Boulevard Arlington, VA 22209
1	Deputy Assistant Secretary of the Army (R&D) Department of the Army Washington, DC 20310
1	HQDA (DAMA-ARP) Washington, DC 20310
1	HQDA (DAMA-MS) Washington, DC 20310
1	Commander US Army BMD Advanced Technology Center ATTN: BMDATC-M, Mr. P. Boyd P.O. Box 1500 Huntsville, AL 35809
1	Commander US Army Materiel Development & Readiness Command ATTN: DRCDMD-ST 5001 Eisenhower Avenue Alexandria, VA 22333
2	Commander US Army Armament Research & Development Command ATTN: DRDAR-TSS (2 cys) Dover, NJ 07801
6	Commander US Army Armament Research & Development Command ATTN: DRDAR-LCA, Mr. C.W. Larson DRDAR-LCU, Mr. R. Davitt DRDAR-LC, Dr.J.T. Frasier DRDAR-LCN, MR. G. Demitrak DRDAR-LCU, Mr. D. Robertson DRDAR-LCU, LTC J. Houle Dover, NJ 07801
1	Director US Army ARRADCOM Benet Weapons Laboratory ATTN: DRDAR-LCB-TL Watervliet, NY 12189
1	Commander US Army Armament Materiel Readiness Command ATTN: DRSAR-LEP-L, Tech Lib Rock Island, IL 61299
1	Commander US Army Aviation Research & Development Command ATTN: DRSAV-E 12th and Spruce Streets St. Louis, MO 63166
1	Director US Army Air Mobility Research & Development Laboratory Ames Research Center Moffett Field, CA 94035
1	Commander US Army Communication Research & Development Command ATTN: DRDCO-PPA-SA Fort Monmouth, NJ 07703

DISTRIBUTION LIST

No. of Copies	Organization
1	Commander US Army Electronics Research & Development Command Technical Support Activity ATTN: DELSD-L Fort Monmouth, NJ 07703
3	Commander US Army Missile Command ATTN: DRDMI-R DRDMI-YBL DRDMI-YDL Redstone Arsenal, AL 35809
1	Commander US Army Tank Automotive Research & Development Command ATTN: DRDTA-UL Warren, MI 48090
4	Commander US Army Materials and Mechanics Research Center ATTN: DRXMR-T, Dr.D. Roylance DRXMR-T, Dr.A.F. Wilde Dr. J. Mescall DRXMR-ATL Watertown, MA 02172
1	Commander US Army Research Office ATTN: Dr. E. Saibel P.O. Box 12211 Research Triangle Park NC 27709
1	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-SL (Tech Lib) White Sands Missile Range NM 88002

No. of Copies	Organization
1	Office of Naval Research Department of the Navy ATTN: Code ONR 439, N. Perrone 800 North Quincy Street Arlington, VA 22217
3	Commander Naval Air Systems Command ATTN: AIR-604 Washington, DC 20360
3	Commander Naval Ordnance Systems Command ATTN: ORD-9132 Washington, DC 20360
2	Commander Naval Air Development Center Johnsville Warminster, PA 18974
1	Commander Naval Missile Center Point Mugu, CA 93041
1	Commander & Director David W. Taylor Naval Ship Research & Development Ctr Bethesda, MD 20084
2	Commander Naval Surface Weapons Center Silver Springs, MD 20910
1	Commander Naval Surface Weapons Center ATTN: Code TX, Dr.W.G. Soper Dahlgren, VA 22448
3	Commander Naval Weapons Center ATTN: Code 4057 Code 5114, E. Lundstrom Code 3813, M. Backman China Lake, CA 93555

DISTRIBUTION LIST

No. of Copies	Organization
4	Commander Naval Research Laboratory ATTN: Mr. W.J. Ferguson Dr. C. Sanday Dr. H. Pusey Dr.F. Rosenthal Washington, DC 20375
1	Superintendent Naval Postgraduate School ATTN: Dir of Lib Monterey, CA 93940
2	ADTC/DLJW (MAJ D. Matuska LTC J. Osborn) Eglin AFB, FL 32542
1	AFFDL (FDT) Wright-Patterson AFB, OH 45433
1	AFML/LLN (Dr.T. Nicholas) Wright-Patterson AFB, OH 45433
1	ASD/XROT (Gerald Bennett) Wright-Patterson AFB, OH 45433
1	ASD/ENFTV (Matt Kolleck) Wright-Patterson AFB, OH 45433
1	Director Lawrence Livermore Laboratory ATTN: Dr. R.H. Toland, L-424 P.O. Box 808 Livermore, CA 94550
1	Director Lawrence Livermore Laboratory ATTN: Dr.M.L. Wilkins P.O. Box 808 Livermore, CA 94550
1	Director Lawrence Livermore Laboratory ATTN: Dr.R. Werne P.O. Box 808 Livermore, CA 94550

No. of Copies	Organization
1	Headquarters National Aeronautics and Space Administration Washington, DC 20546
1	Director Jet Propulsion Laboratory ATTN: Lib (TD) 4800 Oak Grove Drive Pasadena, CA 91103
4	Director National Aeronautics and Space Administration Langley Research Center Langley Station Hampton, VA 23365
1	Director National Aeronautics and Space Administration Manned Spacecraft Center ATTN: Lib Houston, TX 77058
1	Aeronautical Research Assoc. of Princeton Inc. 50 Washington Road P.O. Box 2229 Princeton, NJ 08540
2	Aerospace Corporation ATTN: Mr. L. Rubin Mr.L.G. King 2350 E. El Segundo Blvd. El Segundo, CA 90245
1	Boeing Aerospace Division ATTN: Mr. R.G. Blaisdell (M.S. 40-25) P.O. Box 3703 Seattle, WA 98124

DISTRIBUTION LIST

No. of Copies	Organization
1	Effects Technology, Inc. 5385 Hollister Avenue P.O. Box 30400 Santa Barbara, CA 93105
1	Falcon R&D ATTN: Mr.R. Miller 109 Inverness Dr. East Englewood, CO 80112
2	Falcon R&D Thor Facility ATTN: Mr. D. Malick Mr. J. Wilson 696 Fairmount Avenue Baltimore, MD 21204
1	FMC Corporation Ordnance Engineering Div San Jose, CA 95114
1	General Electric Company Armament Systems Dept. Burlington, VT 05401
1	President General Research Corporation ATTN: Lib McLean, VA 22101
1	Goodier Aerospace Corp. 1210 Massillon Road Akron, OH 44315
1	H.P. White Laboratory Bel Air, MD 21014
3	Honeywell, Inc. Government & Aerospace Prod. Division ATTN: Mr.J. Blackburn Dr. G. Johnson Mr. R. Simpson 600 Second Street, NE Hopkins, MN 55343

No. of Copies	Organization
1	Kaman Sciences Corporation ATTN: Dr.P. Snow P.O. Box 7463 Colorado Springs, CO 80933
1	Lockheed Corporation ATTN: Dr.C.E. Vivian Sunnyvale, CA 94087
1	Lockheed Huntsville ATTN: Dr. L.A. Picklesimer P.O. Box 1103 Huntsville, AL 35809
1	Materials Research Laboratory, Inc. 1 Science Road Glenwood, IL 60427
1	McDonnell Douglas Astronautics ATTN: Dr. J. Wall, Mail Stop 21-2 5301 Bolsa Avenue Huntington Beach, CA 92647
1	Pacific Technical Corporation ATTN: Dr. F.K. Feldman 460 Ward Drive Santa Barbara, CA 93105
2	Physics International Company ATTN: Dr.D. Orphal Dr.E.T. Moore 2700 Merced Street San Leandro, CA 94577
1	Rockwell International Autometics Missile Systems Div. ATTN: Dr.M. Chawla 4300 E. 5th Avenue Columbus, OH 43216
3	Sandia Laboratories ATTN: Dr.W.Herrmann Dr.L. Bertholf Dr.J.W. Nunziatio Albuquerque, NM 87115

DISTRIBUTION LIST

No. of Copies	Organization
1	Science Applicatons, Inc. 101 Continental Blvd, Suite 310 El Segundo, CA 90245
1	Science Applications, Inc. ATTN: G. Burghart 201 W. Dyer Rd (Unit B) Santa Ana, CA 92707
2	Systems, Science & Software ATTN: Dr.R. Sedgwick Ms. L. Hageman P.O. Box 1620 La Jolla, CA 92038
1	US Steel Corporation Research Center 125 Jamison Lane Monroeville, PA 15146
1	Drexel University Dept of Mechanical Engineering ATTN: Dr.P.C. Chou 32nd and Chestnut Streets Philadelphia, PA 19104
1	New Mexico Institute of Mining and Technology Terra Group Socorro, NM 87801
1	Forrestal Research Center Aeronautical Engineering Lab Princeton University ATTN: Dr. Eringen Princeton, NJ 08540
2	Southwest Research Institute Dept of Mechanical Sciences ATTN: Dr. U. Lindholm Dr. W. Baker 8500 Culebra Road San Antonio, TX 78228

No. of Copies	Organization
1	Illinois Institute of Technology Dept of Mechanics and Aerospace Eng. ATTN: Dr.P.H. Francis Chicago, IL 60616
3	SRI International ATTN: Dr.L. Seaman Dr.D. Curran Dr.D. Shockey 333 Ravenswood Avenue Menlo Park, CA 94025
2	University of Arizona Civil Engineering Department ATTN: Dr.D.A. DaDeppo Dr. R. Richard Tucson, AZ 85721
4	University of California Los Alamos Scientific Lab ATTN: Dr. R. Karpp Dr.J. Dienes Dr.L. Germain Dr.B. Germain P.O. Box 808 Livermore, CA 94550
1	University of Dayton University of Dayton Research Institute ATTN: Mr.H.F. Swift Dayton, OH 45405
2	University of Delaware Department of Mechanical Engineering ATTN: Prof J. Vinson Dean J. Greenfield Newark, DE 19711
1	University of Denver Denver Research Institute ATTN: Mr. R.F. Recht 2390 South University Boulevard Denver, CO 80210

DISTRIBUTION LIST

No. of Copies	Organization
1	University of Illinois Aeronautical and Astronautical Engineering Department ATTN: Prof. A.R. Zak 101 Transportation Bldg Urbana, IL 61801

Aberdeen Proving Ground

Dir, USAMSAA
ATTN: DRXSY-D
DRXSY-MP, H.Cohen

Cdr, USATECOM
ATTN: DRSTE-TO-F
Mr.W. Pless
Mr.S. Keithley

Dir, Wpns Sys Concepts Team
Bldg E3516, EA
ATTN: DRDAR-ACW

USER EVALUATION OF REPORT

Please take a few minutes to answer the questions below; tear out this sheet and return it to Director, US Army Ballistic Research Laboratory, ARRADCOM, ATTN: DRDAR-TSB, Aberdeen Proving Ground, Maryland 21005. Your comments will provide us with information for improving future reports.

1. BRL Report Number ______________________________

2. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)

____

____

____

3. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.) ______________________________

____

____

4. Has the information in this report led to any quantitative savings as far as man-hours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate.

____

____

5. General Comments (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.) ______________

____

____

____

6. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

Name: ______________________________

Telephone Number: ______________________________

Organization Address: ______________________________

____

____