

Strain Energy Release Rates of Composite Interlaminar End-Notch and Mixed Mode Fracture: A Sublaminar/Ply Level Analysis and a Computer Code

(NASA-TM-89827) STRAIN ENERGY RELEASE RATES
OF COMPOSITE INTERLAMINAR END-NOTCH AND
MIXED-MODE FRACTURE: A SUBLAMINAR/PLY LEVEL
ANALYSIS AND A COMPUTER CODE (NASA) 88 p

N87-20389

CSCL 11D G3/24

Unclas
45393

R.R. Valisetty and C.C. Chamis
Lewis Research Center
Cleveland, Ohio

Prepared for the
8th Symposium on Composite Materials Testing and Design
sponsored by the American Society for Testing and Materials
Charleston, South Carolina, April 29-May 1, 1986



CONTENTS

	Page
Summary	1
Introduction	1
Sublaminates/Ply Level Analysis	2
Idealization of Three Point Bend Tests	5
Equations for Solutions	7
Results and Discussion	11
Conclusions	12
Appendix I: Source Code	14
Appendix II: Ply Equations	48
Appendix III: Input and Output Instructions for the Code	51
Appendix IV: Typical Input for ENF Specimen	56
Appendix V: Typical Output for ENF Specimen	59
References	81
Figures	82

STRAIN ENERGY RELEASE RATES OF COMPOSITE INTERLAMINAR END-NOTCH
AND MIXED-MODE FRACTURE: A SUBLAMINATE/PLY LEVEL ANALYSIS
AND A COMPUTER CODE

R.R. Valisetty* and C.C. Chamis
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

SUMMARY

E-3476

A computer code is presented for the sublaminates/ply level analysis of composite structures. This code is useful for obtaining stresses in regions affected by delaminations, transverse cracks, and discontinuities related to inherent fabrication anomalies, geometric configurations, and loading conditions. Particular attention is focussed on those layers or groups of layers (sublaminates) which are immediately affected by the inherent flaws. These layers are analyzed as homogeneous bodies in equilibrium and in isolation from the rest of the laminate. The theoretical model used to analyze the individual layers allows the relevant stresses and displacements near discontinuities to be represented in the form of pure exponential-decay-type functions which are selected to eliminate the exponential-precision-related difficulties in sublaminates/ply level analysis. Thus, sublaminates analysis can be conducted without any restriction on the maximum number of layers, delaminations, transverse cracks, or other types of discontinuities. In conjunction with the strain energy release rate (SERR) concept and composite micromechanics, this computational procedure is used to model select cases of end-notch and mixed mode fracture specimens. The computed stresses are in good agreement with those from a three-dimensional finite element analysis. Also SERRs compare well with limited available experimental data.

INTRODUCTION

Laminated composite structures exhibit a number of different failure modes. These include fiber-matrix debonding within individual layers, delamination or separation of adjacent layers, transverse cracking through one or more layers, and fiber fracture. Often interlaminar delamination initiates at sites where complex states of stress with steep gradients exist. These sites include regions near free edges, ply terminations, cutouts, voids, holes inadvertently damaged areas, and defects resulting from fabrication processes. As a result, an accurate assessment of the interlaminar fracture toughness parameters and stress states which initiate delamination is of critical importance in a composite design.

Several methods are available to determine fracture toughness and can be classified into the following groups: experimental, numerical, and analytical. To begin with, the experimental procedures used to determine fracture toughness are edge delamination, double-cantilever-beam, cracked-lap-shear, biaxial

*National Research Council - NASA Research Associate.

interlaminar fracture, and three-point bend tests for both Mode I (end-notch fracture, ENF) and mixed mode fracture (MMF) including Mode I and Mode II. Because of the test simplicity and the ability to delineate between Mode II and mixed mode fracture, the three-point bend tests are now being investigated at NASA Lewis Research Center. Results for interlaminar fracture toughness of unidirectional graphite/epoxy composites are presented in reference 1.

Next, there are numerical solutions. In general, these are developed using either finite difference, finite element, finite-element boundary integral concepts, boundary layer, energy based approximations, and extended Galerkin procedures. They require extensive idealizations to model the response variables in the critical regions and the numerical accuracy of the results is not always assured. Issues related to this topic are discussed in references 2 to 5.

Conventional laminate theories form the core of the final group, the analytical approaches. These are best suited for obtaining the total strain energy release rates. Fracture-related interlaminar stresses which are necessary for delineating between different modes of crack propagation cannot be obtained from these theories. This is because the theories are based on plane stress and global displacement assumptions. The assumption of plane stress precludes the possibility of considering the interlaminar stresses at the outset of the formulation.

This led to the development of ply/sublaminate level analysis (refs. 2 and 4) of composite laminates. Here, layers or selected groups of layers (treating each group as a homogeneous body) are analyzed with the assumption that they are in equilibrium and independent of each other. Thus, the interlaminar stresses are introduced in the formulation at an earlier stage. Initially, these stresses are assumed to be unknown. Then considering the equilibrium of each layer/sublaminate and compatibility at each interlaminar surface, a set of coupled equations in terms of global deformation variables and interlaminar stresses is generated. Subsequent to a successful solution, the fracture toughness and SERRs are easily computed. This approach was initially applied to the estimation of free-edge stresses (ref. 2), and later, successfully extended to both an edge delamination problem (ref. 6) and the analysis of compression and double-lap fracture specimens (ref. 7).

This report describes the implementation of sublaminate analysis to general composite laminates. This analytical technique has been incorporated into a computer code which is listed in appendix I. The code is written with flexibility in regards to the types of materials and loads, number of layers, and boundary conditions as well as cracks which may be in the plane of the laminate or through the thickness. The versatility of this method and the numerical economy in conducting design-related parametric studies are demonstrated through the simulation of three-point bend tests of ENF and MMF specimens. A brief description of sublaminate analysis and ply level equations are presented and an illustration of the basic steps involved in ply-modeling a given structure and setting up the governing equations is included.

SUBLAMINATE/PLY LEVEL ANALYSIS

This analysis which operates on a ply-by-ply basis may be applied to all layers in a laminate or only to those few layers which surround defects and

where the stresses are adversely affected. The remaining layers can be grouped into sublaminates and may be treated as homogeneous layers. Interlaminar stresses are initially assumed to be unknown. Enforcement of displacement and traction continuity at lamina interfaces leads to the final equations, which are subsequently solved. Thus, a better engineering approximation is achieved concerning the interlaminar stresses. The success of the approach, then, depends on the engineering model that will be used to analyze each layer. The model should be capable of securing "layer equilibrium" as discussed below.

Just as it is possible to combine different layers into a single unit or a sublaminate and treat the resulting unit as a single layer, it is also possible to idealize a single layer as composed of a finite number of thin sublayers. For the purpose of sublaminate/ply level analysis, these sublayers once again can be treated as individual units. This situation is, therefore, similar to employing either a coarse mesh or a fine mesh at selective locations through the thickness in a finite element analysis. The ensuing results for interlaminar stresses depend upon the idealization in terms of sublaminates, layers, and sublayers and to what extent they are affected is discussed in reference 4.

To achieve "layer equilibrium," the model must be capable of dealing with prescribed stress/displacement conditions on transverse interlaminar surfaces and afford specification of at least five boundary conditions (three force resultants or deflections and two moment resultants or rotations) per layer edge. Two such models are available now. The first one, developed by Pagano (ref. 2), is a higher order theory based on Reissner's variational principle. The second one, developed by Valisetty (ref. 8) from an iterative procedure, is a refined homogeneous plate bending theory. The later model was selected to analyze the layers in the present program because it provides explicit distributions through the thickness for stresses and displacements.

A summary of the basic equations for the generic ply shown in figure 1 is given in appendix II. The equations are deduced from the homogeneous plate theory (ref. 8) for the case of cylindrical bending. The basis for these equations and their derivation are presented in references 6 and 8.

Suppose the analysis is to be applied to N layers or a combination of layers and sublaminates. The overall equations of equilibrium and the constitutive relationships will constitute a set of $8N$ equations in terms of like variables ($2N$ displacements, N rotations, $3N$ force resultants, and $2N$ moment resultants). This set is further supplemented by additional $2(N-1)$ equations, which are necessary for the simultaneous solution of $2(N-1)$ interlaminar stresses, when the displacement continuity is enforced at the $(N-1)$ interlaminar surfaces. The transverse shear and normal stresses at the interlaminar surfaces, if they are not prescribed explicitly (for example, as on the top and bottom surfaces of a laminate or on the cracked interlaminar surfaces), are treated as unknowns. From among this set of equations the force and moment resultant variables can be eliminated with the aid of constitutive relations. This leaves a set of $(5N-2)$ coupled ordinary differential equations to be solved for $2N$ displacement variables, N rotations, and $2(N-1)$ interlaminar stresses.

These observations are based upon two requirements: (1) the laminate is in generalized plane strain, and (2) there is no cross-sectional warping. The latter condition is satisfied when cross-ply laminates and symmetric laminates

are used. However, for angle ply laminates and laminates of more general construction, the warping cannot be neglected. In this case, the number of variables increase and likewise the number of equations to be solved.

The final system of differential equations can be represented in the following form:

$$\sum_{k=0}^4 \sum_{j=1}^{4N-2} SK_{ij}^{(k)} f_j^{(k)} = p_i; \quad i = 1, 2, \dots, 4N-2 \quad (1)$$

where p represents the applied loading, f is the vector of governing variables ($2N$ average displacements and $2N-2$ interlaminar stresses), and the superscript k is the order of the in-plane derivative. The coefficient matrices ($[SK]$) depend on layer thickness and material properties. The total solution is constructed by obtaining: (1) a particular solution corresponding to the applied load, and (2) other solutions from the homogeneous part of the differential equations. The analysis shows that there are $(8N-2)$ roots or eigenvalues, therefore an equal number of homogeneous or eigen solutions. An inspection of the eigenvalues reveals the nature of these solutions. There are six zero roots and the remaining are either real or complex. The solution corresponding to the zero roots is identified to be of shear deformation type and the remaining are of local type which are prominent only in the edge regions of the laminate.

There are $(8N-2)$ integration constants which require specification of a like number of boundary conditions at the laminate edges. These are established on the basis of physical reasoning. Since global laminate equilibrium is translated into local layer equilibrium in this analysis, boundary conditions are to be specified at the layer edges insofar as possible. Among such conditions, the engineering boundary conditions appear to be a natural choice. Consequently, one can prescribe either force resultant or an average in-plane displacement, moment resultant or an average rotation and shear resultant or an average transverse displacement at each layer edge. The remaining $(2N-2)$ conditions are specified in terms of either interlaminar transverse shear stress or its in-plane derivative at the in-plane edges of interlaminar surfaces.

By retaining the local solutions at only one laminate edge, the number of boundary conditions can be reduced to $(4N+2)$. The edge at which the local solutions are to be retained depends on the particular problem in question. This procedure does not threaten the laminate equilibrium as it can be secured from the global perspective with the aid of a shear deformation type solution (which is controlled by the six zero roots) alone. However, for this procedure to be successful, the layer-defined force and moment resultants at the edge, where the local solutions are dropped, must be specified to be statically equivalent to the local total loading applied to the whole laminate edge. At the edge where the local solutions are retained the boundary conditions can still be specified on each individual layer.

This maneuver makes the sublaminates/ply level analysis powerful, for it allows equilibrium in any number of layers to be considered independently. Usually, the number of layers to which this type of analysis is applied is restricted by a computer precision requirement. This requirement puts a limit on the magnitude of the real roots and real part of the complex roots. When

the local solutions are obtained for each laminate edge, one at a time, the mode shape of the eigen solutions can be constructed in the form of pure decay type solutions which are valid only at one edge. Thus the exponential-function-related precision problem are effectively eliminated. This is a significant contribution, because now sublaminates/ply level analysis can be extended to laminates with discontinuities in loads, materials, and geometry along the axis of the laminate and without any restriction as to the number of layers.

For cases involving axial discontinuities, the sublaminates analysis procedure essentially remains the same. The first step is to divide the structure into several regions or subunits such that each subunit is free of any discontinuity related to loads, materials, or geometry. Each subunit, then, will be in the form of a laminate loaded by continuously varying loads. Materials may differ for layers within the subunits. The concentrated loads on the original structure translate into shear loads applied to the edges of the subunits. The delaminations can be accounted for by treating the separate portions of the laminate as different sublaminates abutting the uncracked laminate which may be treated as just another sublaminates. This process is described in detail in the next section with an example of ENF and MMF specimens.

Formal solutions in terms of integration constants can then be constructed for each subunit using the aforementioned procedure. The postulation of boundary conditions and continuity conditions between different subunits provides equations for the solution of integration constants. Typical continuity conditions involve specifying axial force resultant, shear resultant, moment resultant, axial displacement, transverse displacement, and cross-sectional rotation for each layer and transverse shear stress and its derivative for each interlaminar surface.

The computer code containing these procedures is presented in appendix I. Instructions for the input and output are given in appendix III. Typical input and output for the ENF specimen are given in appendices IV and V, respectively.

IDEALIZATION OF THREE-POINT BEND TESTS

The sublaminates/ply level analysis is now applied to estimate the interlaminar fracture stresses in the three-point bend test specimens. Figure 2 presents the schematics for the three-point bend tests of the ENF and MMF specimens. For the MMF specimen, the right support is provided on the bottom surface of the top flange instead of the bottom flange. The laminate is made of 24 graphite/epoxy (AS/E) layers and five interlaminar resin layers as shown in figure 3. The manufacturing process gives rise to the resin rich region between adjacent fiber-matrix layers whose thickness usually amounts to a tenth of the fiber diameter and in general depends on the fiber volume ratio (FVR). This region is referred to as the interply resin layer and is considered in order to accurately compute those stresses which induce interply delamination. Two different FVRs, 0.3 and 0.6, are considered in the computations. At these FVRs, the interply resin layer thicknesses are 0.0001854 in. and 0.00004323 in., respectively. The graphite/epoxy layers have the following properties:

At FVR = 0.3

$E_L = 11.96 \text{ msi}$, $E_T = 0.921 \text{ msi}$, $E_{TT} = 1.01 \text{ msi}$, $G_{TT} = 0.245 \text{ msi}$,
 $G_{LT} = 0.368 \text{ msi}$, $\nu_{LT} = 0.348$, $\nu_{TT} = 0.546$
ply thickness = 0.005 in.

At FVR = 0.6

$E_L = 21.15 \text{ msi}$, $E_T = 1.238 \text{ msi}$, $E_{TT} = 1.323 \text{ msi}$, $G_{TT} = 0.362 \text{ msi}$,
 $G_{LT} = 0.623 \text{ msi}$, $\nu_{LT} = 0.282$, $\nu_{TT} = 0.423$
ply thickness = 0.005 in.

where E is the elastic modulus, G is the shear modulus and ν is Poisson's ratios, L represents a direction along the fibers and T , a direction, normal to the fibers.

As inspection reveals that in order to apply sublaminates/ply level analysis the specimen (fig. 2) must be divided into the following four regions:

$$\begin{aligned} \text{Region 1: } & 0 < x_2 < L/2; -H/2 < z < H/2, \\ \text{Region 2: } & L/2 < x_2 < (L-l); -H/2 < z < H/2, \\ \text{Region 3: } & (L-a) < x_2 < L; 0 < z < H/2, \\ \text{Region 4: } & (L-a) < x_2 < L; -H/2 < z < 0. \end{aligned} \tag{2}$$

where a , L and H denote the length of the crack, the length of the specimen and its thickness, respectively. The length of the specimen is 4 in. and (x_2, z) is a global coordinate system. Region 1 is identical to Region 2 and Regions 3 and 4 are identical. Now, the analysis can be applied to each of these four regions obtaining four sets of formal solutions in terms of integration constants. The total number of these constants will be 680.

Before proceeding to setup such a solution, it should be asked whether the analysis can be simplified any further. The answer will be obvious, if it is recognized that the objects of investigation are the stresses in the interlaminar resin layer in the region ahead of the crack. To obtain these stresses, it is not necessary to apply the analysis to all of the layers within each of the four regions. Recognizing this, it can be assumed that the 12 graphite/epoxy and two interlaminar resin layers (fig. 3) can be treated as single units or sublaminates in the third and fourth regions and in the top and bottom halves of the first and second regions. These sublaminates can further be treated as homogeneous orthotropic plates represented by their effective moduli. The resulting idealization is shown in figure 4.

Now, the analysis is applied only to the first and second regions. Since the sublaminates in the third and fourth regions are treated as homogeneous plates, the governing solutions for these regions, which are of the shear deformation type, can be obtained directly. The order of the resulting set of all equations will be 56. This can further be reduced to 32 by retaining only the local solution near the crack in Region 2. This solution yields the fracture related stresses in the resin layer ahead of the crack. The other local solutions, which are suppressed, control the stress variation near the concentrated load and at the left support.

Equations for Solution

The solution procedure for the specimen idealized in figure 4 is described here. For this purpose, the laminate in Region 1 is selected. Let "1," "2," and "3" denote, respectively, the top, middle, and bottom sublaminates. The outer sublaminates each contain 12 graphite/epoxy and two resin layers. The middle one contains just one interply resin layer. For the analysis, these sublaminates are treated as homogeneous plies. It is convenient to use local coordinate systems, $(x_2, z; k=1, 2; \text{ and } 3)$, for presenting stress and displacement distributions. The ply middle surfaces are represented by $Z = 0$ and c denotes ply semithicknesses.

For each layer, stress and displacement distributions through the thickness are available. These were discussed earlier in appendix II along with relevant equilibrium and constitutive equations in terms of engineering kinematic and force variables and interlaminar stresses. As shown in figure 1, these variables are meaningful for only those plies for which they are defined. When it is necessary to indicate such ply dependency, these variables are superscripted with a ply identification number.

Herein, let U_2 and W be the components in the x_2 and z directions, respectively of the displacement of the $z = 0$ surface. Also, let σ_{2z} and σ_{zz} be the transverse shear and normal stresses. Using this notation, the unknown variables can be listed as $U_2^1, U_2^2, U_2^3, W^1, W^2, W^3, \sigma_{2z}^2(x_2, z = c), \sigma_{2z}^3(x_2, z = c), \sigma_{zz}^2(x_2, z = c), \text{ and } \sigma_{zz}^3(x_2, z = c)$. In all, these variables constitute a set of ten unknowns. An identical set of variables also govern the deformation in Region 2. For sublaminates defined in the third and fourth regions, U_2 and W are the only governing variables.

Equations I-2 and I-3 of appendix II may be combined to produce

$$M_{2,22} + c m_{2,2} + q_2 = 0 \quad (3)$$

where

M_2 moment resultant associated with x_2 direction,

c ply semithickness,

m_2 $\sigma_{2z}(x_2, c) + \sigma_{2z}(x_2, -c)$, and

q_2 $\sigma_{zz}(x_2, c) - \sigma_{zz}(x_2, -c)$.

The equilibrium conditions (eq. (3), and eq. (I-1) of appendix II) for each of the three plies and the following continuity requirements:

$$u_2^1 = u_2^2; 0 < x_2 < L/2, z_1 = -c_1,$$

$$u_2^2 = u_2^3; 0 < x_2 < L/2, z_2 = -c_2,$$

(4)

$$w^1 = w^2; 0 < x_2 < L/2, z_1 = -c_1, \text{ and}$$

$$w^2 = w^3; 0 < x_2 < L/2, z_2 = -c_2.$$

for the displacements at the two laminar interfaces make up the desired set of equations for Region 1, where u_2 and w are the components of the displacement in the x_2 and z directions. Equation 4 can be expressed solely in terms of the generalized unknown variables of the problem with the aid of Eqs. I-16 and I-17 of appendix II which are thicknesswise displacement distributions. An identical set of equations follows for the second region. The cracked surfaces in Regions 3 and 4 are free of the transverse stresses. Therefore, the solution for the sublaminates in these regions degenerates into a shear deformation type solution with a shear correction factor of 1.

These four sets of equations are solved separately in a straightforward manner and formal solutions in terms of integration constants are constructed. There are 56 constants which may be determined through the boundary conditions at $x_2 = 0$ and $x_2 = L$ and the continuity conditions between the first and second regions at $x_2 = L/2$, between the second and third regions at $x_2 = (L-a)$, and between the second and fourth regions at $x_2 = (L-a)$. For presenting these conditions, superscripts "4," "5," and "6" are used to denote the top, middle, and bottom layers in Region 2 and "7" and "8" are for the sublaminates in the third and fourth regions. As mentioned before, superscripts "1," "2," and "3" apply similarly to the variables in the first region.

Boundary conditions at $x_2 = 0$:

$$N_2^1 = 0, M_2^1 = 0, Q_2^1 = 0, \sigma_{2z}^1(x_2, -c_1) = 0$$

$$N_2^2 = 0, M_2^2 = 0, Q_2^2 = 0, \sigma_{2z}^2(x_2, -c_2) = 0$$

(5)

$$u_2^3(x_2, -c_3) = 0, w^3(x_2, -c_2) = 0, M_2^3 = 0$$

Continuity conditions at $x_2 = L/2$:

$$\begin{aligned}
 N_2^1 &= N_2^4, M_2^1 = M_2^4, Q_2^1 = Q_2^4 + 60, \sigma_{2z}^1(x_2, -c_1) = \sigma_{2z}^4(x_2, -c_4) \\
 U_2^1 &= U_2^4, \phi_2^1 = \phi_2^4, W^1 = W^4, \sigma_{2z}^1(x_2, -c_1) = \sigma_{2z}^4(x_2, -c_4),_2 \\
 N_2^2 &= N_2^5, M_2^2 = M_2^5, Q_2^2 = Q_2^5, \sigma_{2z}^2(x_2, -c_2) = \sigma_{2z}^5(x_2, -c_2) \\
 U_2^2 &= U_2^5, \phi_2^2 = \phi_2^5, W^2 = W^5, \sigma_{2z}^2(x_2, -c_2) = \sigma_{2z}^5(x_2, -c_2),_2 \\
 N_2^3 &= N_2^6, M_2^3 = M_2^6, Q_2^3 = Q_2^6 + 60, \\
 U_2^3 &= U_2^6, \phi_2^3 = \phi_2^6, W^3 = W^6,
 \end{aligned} \tag{6}$$

Continuity conditions at $x_2 = (L-a)$:

$$\begin{aligned}
 N_2^4 &= N_2^7, M_2^4 = M_2^7, Q_2^4 = Q_2^7 \\
 U_2^4 &= U_2^7, \phi_2^4 = \phi_2^7, W^4 = W^7
 \end{aligned} \tag{7}$$

Continuity conditions at $x_2 = (L-a)$:

$$\begin{aligned}
 N_2^6 &= N_2^8, M_2^6 = M_2^8, Q_2^6 = Q_2^8 \\
 U_2^6 &= U_2^8, \phi_2^6 = \phi_2^8, W^6 = W^8
 \end{aligned} \tag{8}$$

Boundary conditions at $x_2 = (L-a)$:

$$\begin{aligned}
 N_2^5 &= 0, M_2^5 = 0, Q_2^5 = 0, \\
 \sigma_{2z}^4(x_2, -c_4) &= 0, \sigma_{2z}^6(x_2, -c_6) = 0
 \end{aligned} \tag{9}$$

Boundary conditions at $x_2 = L$ for ENF specimen:

$$\begin{aligned}
 N_2^7 &= 0, M_2^7 = 0, Q_2^7 = 0, \\
 N_2^8 &= 0, M_2^8 = 0, w^8(x_2, -c_8) = 0
 \end{aligned} \tag{10}$$

Boundary conditions at $x_2 = L$ for MMF specimen:

$$\begin{aligned}
 N_2^7 &= 0, M_2^7 = 0, w^7(x_2, -c_7) = 0, \\
 N_2^8 &= 0, M_2^8 = 0, Q_2^8 = 0.
 \end{aligned} \tag{11}$$

where c_k ($k = 1, 2, \dots, 8$) are ply semithicknesses, d_2 are cross-sectional rotations, N_2 are force resultants and M_2 are moment resultants.

The applied load of 120 lb is represented as a jump in shear resultants in the outer layers in equation 6. It is assumed that the central interlaminar resin layer is too thin to carry any net shear force and the shear stress in the outer sublaminates away from the point of application of the load is uniform through the thickness. Similarly, the conditions on the transverse shear stress in equation 6 are arrived at under the assumption that the points at which the stress is prescribed are considerably away from the point of application of the load; therefore, the stress is unaffected. Additionally, the crack is modeled as having a finite thickness, the thickness being that of the interply resin layer. The interply resin layer is destroyed as the crack advances so that the crack-tip (idealized as having finite thickness) is stressfree. This is indicated by equation 9.

The approach discussed so far makes use of the complete form of the eigen solutions to represent the response variable in the first two regions. This yields stress and displacement distributions which show not only the effect of the crack-tip but also that of the concentrated load at the left support. Since the objective is the crack-tip stress field, the form of eigen solutions can be modified to include only the decay-type solutions at the crack-tip. This degenerates formal representation of the solution in Region 1 into that of a shear deformation theory, as is also the case with the third and fourth regions. As a result, the total number of integration constants and consequently the number of boundary/continuity conditions are reduced to 36. The conditions at $x_2 = 0$ and $L/2$ are to be modified as shown below:

Boundary conditions at $x_2 = 0$:

$$u_2^2(x_2, 0) = 0 ,$$

$$u_2^2(x_2, 0) = 0 , \tag{12}$$

$$M_2^1 + M_2^2 + M_2^3 + N_2^1 \times c_1 - N_2^3 \times c_3 = 0$$

Continuity conditions at $x_2 = L/2$:

$$u_2^2(x_2, 0) = u_2^5(x_2, 0)$$

$$w_2^2(x_2, 0) = w_2^5(x_2, 0)$$

$$d_2^2(x_2, 0) = d_2^5(x_2, 0)$$

(13)

$$N_2^1 + N_2^2 + N_2^3 + N_2^4 + N_2^5 + N_2^6$$

$$Q_2^1 + Q_2^2 + Q_2^3 + Q_2^4 + Q_2^5 + Q_2^6 + 120$$

$$M_2^1 + M_2^2 + M_2^3 + N_2^1 \times c_1 - N_2^3 \times c_3 = M_2^4 + M_2^5 + M_2^6 + N_2^4 \times c_4 - N_2^6 \times c_6$$

The procedure described so far is programmed into a general purpose code. A description of this code (appendix I) together with instructions for input and output (appendix III) are included.

RESULTS AND DISCUSSION

Figures 5 and 6 display the stress field in the interply layer ahead of the crack-tip for the ENF and MMF specimens, respectively. The crack length is 1 in. and the fiber volume ration (FVR) is 0.6. The ENF specimen yields monotonically decreasing compressive normal stresses, whereas the MMF specimen produces tensile stresses. For both specimens, the transverse shear stress is positive near the crack-tip. Within a distance of 1.5 H from the crack-tip, all stresses approach engineering values. The compressive nature of the transverse normal stress for the ENF specimen indicates that this is a pure Mode II crack. For the MMF specimen, as anticipated, both transverse shear and normal stresses are tensile. Figures 7 and 8 show the behavior of peak stresses as the crack advances. Both types of specimens show monotonically increasing crack-tip stresses.

Strain energy release rates (SERR) are computed by monitoring changes in the total compliance of the structure. Figure 9, which shows the total energy release rate plotted against the crack length, indicates a stable crack growth. Both the ENF and MMF specimens predict the same total SERR. This result was also anticipated since, in the case of the MMF specimen, the unsupported delaminated flange will not pick up any load. However, in the case of the ENF specimen, the possibility for load transfer exists as both delaminated flanges come into contact as the applied load increases. Since this contact is not incorporated into the analysis, the same results are predicted.

In order to study this progressive contact, the relative displacement between the cracked surfaces should be restricted so that cracked sublaminates do not run into one another. If they do the problem becomes nonlinear and is beyond the scope of the present investigation. However, a simple idealization of the practical situation using an ENF specimen assumes complete contact

between the cracked surfaces and that the resulting contact stresses are negligible. Then, the condition on the transverse displacement in equation 10, accordingly, is replaced by the following:

$$w^7(x_2, -c_7) = w^8(x_2, c_8) \quad (14)$$

This model is named ENF-II. Results for the stress field ahead of the crack-tip and total SERR using this model are presented in figures 10 and 11, respectively. The FVR for these results is 0.6. Both normal stresses are zero and in figure 10 the shear stress follows the previously observed trend. In this case, the crack plane becomes a plane of symmetry. This does not permit the normal stresses to develop on the crack plane. This total SERR is only about one-third of that which is available in the first two models. As shown in figure 11, these results compare well with engineering theory which are slightly conservative. Reference 1 presents some experimental data on the total SERR for the ENF specimen. The range of the total SERR shown in figure 11 matches with the experimentally measured range of 3.0 to 3.6 psi-in. The role of the local stress field on crack propagation can be established by measuring individual mode contributions, but that study is deferred for a future investigation.

In order to assess the present study, the interply stresses ahead of the crack-tip are compared in figure 12 with the three-dimensional finite element results of reference 1 for the ENF specimen. With the exception of small differences in peak stresses at the crack-tip, both analyses provide the same trends. The finite element results for the transverse shear stress exhibit an oscillatory behavior. This was attributed, in reference 1, to the particular fashion in which the finite element grid ahead of the crack-tip was updated as the crack length increased.

CONCLUSIONS

The analysis presented herein can be extended easily to cracks through the thickness. In this case the eigen solutions are to be computed each time the crack extends. Discontinuities related to geometry and loads can also be handled in an analogous manner. Similarly, flexibility is available to consider delaminations anywhere through the thickness and along the axis of the laminate. These features make the present analysis ideally suited for design related parametric studies when: (1) different configurations and the effect of materials are to be evaluated, and (2) fracture characteristics need to be established. The following conclusions are reached based on this study:

1. After the eigen solutions are established in each region, the order of the final system of equations representing boundary/symmetry/continuity conditions is reduced to 36 for the specimen idealizations. This results in an analysis which is highly economical.

2. The ability to represent critical local stress fields in terms of pure decay type solutions eliminates the exponential-function-related precision problems. This effectively removes the cap on the number of layers that can be studied on a ply-by-ply basis.

3. Different types of restraints can be handled routinely, as indicated by ENF, ENF-II and MMF models, without incurring any major changes in the final systems of equations.

4. The models also illustrate how the stress field ahead of the crack-tip responds to simple changes in the edge supports. This knowledge is important in designing fracture specimens for different modes of crack propagation.

APPENDIX I: SOURCE CODE

```

0001100      PROGRAM MAIN
0001200 C.....CDABS=CABS
0001300 C.....DCMPLX....REMOVE THIS DECLARATION
0001400 C.....REAL*8
0001500 C.....REAL*4 RCM
0001600 C.....COMPLEX*16
0001700
0001800
0001900 C  E11      S32      CB16      KQ1      CURVTR  2  NZR  12  CRTB  22
0002000 C  E22      S33      CB22      KQ2      NSREG   3  NRR  13  CRTC  23
0002100 C  E33      S44      CB26      KM1      NMT     4  NCR  14  ISGN  24
0002200 C  NU12     S45      CB66      KM2      NSL     5  NBR  15
0002300 C  NU13     S55      SNC1      KP1      NGN     6  CRT  16
0002400 C  NU23     CB1      SNC2      KP2      NGT     7  CHODE 17
0002500 C  G44      CB2      SHM1     KN1      NMTYPE  8  SLN  18
0002600 C  G55      CB6      SHM2     KN2      THICK   9  SK   19
0002700 C  G66      CB11     SHQ1     THETA    THCK    10  RK   20
0002800 C  S31      CB12     SHQ2     EPSLON 1  NSR    11  CRTA  21
0002900
0003000
0003100      IMPLICIT REAL  (A-H,O-Z)
0003200      REAL  RCM
0003300      DIMENSION RCM(150000)
0003400      DATA IN5,IN6,IN7,IN8/5,6,5,6/
0003500      MAXRCM=150000
0003600      IPR=1
0003700      CALL PZERO(RCM,MAXRCM/IPR)
0003800      READ(IN5,2) NSREG,NMT
0003900      WRITE(IN6,9)NSREG,NMT
0004000 9      FORMAT(/1X,2I3,' NO. OF REGIONS AND NO. OF MATERIALS')
0004100 2      FORMAT(20I2)
0004200 92     FORMAT(10E10.3)
0004300      N1=1+39*NMT*IPR
0004400      N2=N1+IPR
0004500      N3=N2+IPR
0004600      N4=N3+1
0004700      N5=N4+1
0004800      N6=N5+NSREG
0004900      N7=N6+NSREG
0005000      N8=N7+NSREG
0005100      N9=N8
0005200      RCM(N3)=NSREG
0005300      RCM(N4)=NMT
0005400      X=1.D 00
0005500 C.....READ EPSLON, CURVTR, NSL, NGN, NGT
0005600      CALL METPRO(RCM,IN5,IN6,IN7,IN8,NSREG,NMT,RCM(N5),RCM(N6),RCM(N7),-
0005700      *1,X,1,1,X,IPR,RCM(N1),RCM(N2),1,1,1,1)
0005800      DO 1 I=1,NSREG
0005900 C.....READ NMTYPE
0006000      CALL METPRO(RCM,IN5,IN6,IN7,IN8,NSREG,NMT,RCM(N5),RCM(N6),RCM(N7),-
0006100      *2,X,I,RCM(N9),X,IPR,X,X,1,I2,1,1)
0006200 1      N9=N9+I2
0006300      N10=N9
0006400      DO 3 I=1,NSREG

```



```

0006500 C.....READ THICK
0006600     CALL METPRO(RCH, IN5, IN6, IN7, IN8, NSREG, NMT, RCH(N5), RCH(N6), RCH(N7), -
0006700     *3, RCH(N10), I, 1, X, IPR, X, X, 1, I2, 1, 1)
0006800 3     N10=N10+I2*IPR
0006900 C.....READ THCK
0007000     CALL METPRO(RCH, IN5, IN6, IN7, IN8, NSREG, NMT, RCH(N5), RCH(N6), RCH(N7), -
0007100     *6, RCH(N9), 1, 1, RCH(N10), IPR, X, X, 1, 1, 1, 1)
0007200     N11=N10+NSREG*IPR
0007300 C.....READ MECHANICAL PROPERTIES
0007400     CALL METPRO(RCH, IN5, IN6, IN7, IN8, NSREG, NMT, 1, 1, 1, 4, X, 1, 1, X, IPR, X, X, -
0007500     *1, 1, 1, 1)
0007600     N12=N11+NSREG
0007700     N13=N12+NSREG
0007800     N14=N13+NSREG
0007900     N15=N14+NSREG
0008000     N16=N15+NSREG
0008100     N17L1=0
0008200     N17L2=0
0008300     N18L=0
0008400     DO 15 I=1, NSREG
0008500     CALL METPRO(RCH, IN5, IN6, IN7, IN8, NSREG, NMT, RCH(N5), RCH(N6), RCH(N7), -
0008600     *5, X, 1, 1, X, IPR, X, X, I, I2, I, I4)
0008700     IF(I4.EQ.0.OR.I4.EQ.1.OR.I4.GT.I) GOTO 16
0008800     GOTO 15
0008900 16    N17L1=N17L1+8*I2-2
0009000     IF(4*I2-2.GT.N;7'.2) N17L2=4*I2-2
0009100     N18L=N18L+4*I2-2
0009200 15    CONTINUE
0009300     N17=N16+N17L1*2*IPR
0009400     N18=N17+N17L1*N17L2*4*2*IPR
0009500     N19=N18+N18L*IPR
0009600     N20=N19+N17L1*N17L1*IPR
0009700     N21=N20+N17L1*IPR
0009800     N22=N21+N17L1*IPR
0009900     N23=N22+N17L1*IPR
0010000     N24=N23+N17L1*IPR
0010100     N25=N24+N17L1
0010200     WRITE(IN6, 17) N1, N2, N3, N4, N5, N6, N7, N8, N9, N10, N11, N12, N13, N14, N15, -
0010300     *N16, N17, N17L1, N17L2, N18, N19, N20, N21, N22, N23, N24, N25
0010400 17    FORMAT(20I5)
0010500     CALL MPRG(IN5, IN6, IN7, IN8, NSREG, NMT, RCH(N5), RCH(N6), RCH(N7), -
0010600     *RCH(N8), RCH(N9), RCH(N10), RCH(N11), RCH(N12), RCH(N13), RCH(N14), -
0010700     *RCH(N15), RCH(N16), RCH(N17), RCH(N18), RCH(N19), RCH(N20), RCH(N21), -
0010800     *RCH(N22), RCH(N23), RCH(N24), N17L1, N17L2, N18L, N25, MAXRCH, RCH, -
0010900     *IPR)
0011000     STOP
0011100     END
0011200     SUBROUTINE METPRO(RCH, IN5, IN6, IN7, IN8, NSREG, N, NSL, NGN, NGT, NXX -
0011300     *, THICK, NR, NMTYPE, THCK, IPR, EPSLON, CURVTR, I1, I2, I3, I4)
0011400     IMPLICIT REAL (A-H, O-Z)
0011500     REAL RCH
0011600     DIMENSION RCH(1), NSL(1), NGN(1), NGT(1), THICK(1), NMTYPE(1), THCK(1)
0011700     GOTO(1, 2, 3, 4, 5, 6), NXX
0011800 1     READ(IN5, 17) EPSLON, CURVTR

```

```

0011900      WRITE(IN6,19) EPSLON,CURVTR
0012000      DO 11 NR=1,NSREG
0012100      READ(IN5,12)  NSL(NR),NGN(NR),NGT(NR)
0012200 11    WRITE(IN6,13)NR,NSL(NR),NGN(NR),NGT(NR)  -
0012300      RETURN
0012400 2     NL=NSL(NR)
0012500      I2=NL
0012600      WRITE(IN6,14) NR
0012700      READ(IN5,12)  (NMTYPE(I),I=1,NL)
0012800      WRITE(IN6,12)  (NMTYPE(I),I=1,NL)
0012900      RETURN
0013000 3     NL=NSL(NR)
0013100      I2=NL
0013200      WRITE(IN6,16) NR
0013300      READ(IN5,17) (THICK(I),I=1,NL)
0013400      WRITE(IN6,17) (THICK(I),I=1,NL)
0013500      RETURN
0013600 6     J=0
0013700      DO 18 K=1,NSREG
0013800      NL=NSL(K)
0013900      THCK(K)=0.D 00
0014000      DO 18 I=1,NL
0014100      J=J+1
0014200      THCK(K)=THCK(K)+THICK(J)
0014300 18    CONTINUE
0014400      WRITE(IN6,172)
0014500      WRITE(IN6,17) (THCK(I),I=1,NSREG)
0014600      RETURN
0014700 4     I=N*IPR
0014800      CALL  MPRO1(RCH(1),RCH(I+1),RCH(2*I+1),RCH(3*I+1),RCH(4*I+1),  -
0014900      *RCH(5*I+1),RCH(6*I+1),RCH(7*I+1),RCH(8*I+1),RCH(9*I+1),RCH(10*I+1)-
0015000      *,RCH(11*I+1),RCH(12*I+1),RCH(13*I+1),RCH(14*I+1),RCH(15*I+1)  -
0015100      *,RCH(16*I+1),RCH(17*I+1),RCH(18*I+1),RCH(19*I+1),RCH(20*I+1)  -
0015200      *,RCH(21*I+1),RCH(22*I+1),RCH(23*I+1),RCH(24*I+1),RCH(25*I+1)  -
0015300      *,RCH(26*I+1),RCH(27*I+1),RCH(28*I+1),RCH(29*I+1),RCH(30*I+1)  -
0015400      *,RCH(31*I+1),RCH(32*I+1),RCH(33*I+1),RCH(34*I+1),RCH(35*I+1)  -
0015500      *,RCH(36*I+1),RCH(37*I+1),RCH(38*I+1)  -
0015600      *,N,IN5,IN6,IN7,IN8)
0015700      RETURN
0015800 5     I2=NSL(I1)
0015900      I4=NGT(I3)
0016000      RETURN
0016100 172   FORMAT(/1X,'SUBLAMINATE THICKNESSES')
0016200 17     FORMAT(8E10.4)
0016300 14     FORMAT(/1X,'REGION = ',I3,', MATERIAL FOR EACH LAYER')
0016400 19     FORMAT(/1X,2E10.4,' EPSLON, CURVTR'/)
0016500 12     FORMAT(20I4)
0016600 13     FORMAT(1X,'REGION = ',I3,', NO. OF LAYERS = ',I3,', NGN =  -
0016700      *',I3,', NGT = ',I3)
0016800 16     FORMAT(/1X,'REGION = ',I3,', PLY THICKNESSES')
0016900      END
0017000      SUBROUTINE MPRG(IN5,IN6,IN7,IN8,NSREG,NMT,NSL,NGN,NGT,  -
0017100      *NMTYPE,THICK,THCK,NSR,NZR,NRR,NCR,  -
0017200      *NBCR,CRT,CMODE,SLN,SK,RK,CRTA,  -

```

```

0017300      *CRTB,CRTC,ISGN,NCN1,NCN2,NCN4,N25,MAXRCH,RCH,IPR)
0017400      IMPLICIT REAL (A-H,O-Z)
0017500      REAL RCH
0017600      DIMENSION NSL(1),NGN(1),NGT(1),NZR(1),NRR(1),NCR(1),NBCR(1),NSR(1)-
0017700      *,CRT(NCN1,1),CMODE(NCN1,NCN2,1),SLN(1),CRTA(1),THCK(1),
0017800      *CRTB(1),CRTC(1),SK(NCN1,1),RK(1),RCH(1),ISGN(1),THICK(1),NMTYPE(1)
0017900      CALL PZERO(CRT,NCN1*2)
0018000      CALL PZERO(CMODE,NCN1*NCN2*4)
0018100      CALL PZERO(SLN,NCN4)
0018200      NG1=0
0018300      NG2=0
0018400      NG4=0
0018500      DO 1244 NR=1,NSREG
0018600      NRGN=NGN(NR)
0018700      NZR(NR)=0
0018800      NRR(NR)=0
0018900      NCR(NR)=0
0019000      NSR(NR)=0
0019100      NL=NSL(NR)
0019200      H=THCK(NR)
0019300      NPLY=NL
0019400      IF(NR.GT.1) NG1=NG1+NSL(NR-1)
0019500      NRT=0
0019600      NCT=0
0019700      IF(NGT(NR).EQ.0) GOTO 12345
0019800      IF(NGT(NR).EQ.NR) GOTO 12345
0019900      NR1=NGT(NR)
0020000      NZR(NR)=NZR(NR1)
0020100      NSR(NR)=NSR(NR1)
0020200      NRR(NR)=NRR(NR1)
0020300      NCR(NR)=NCR(NR1)
0020400      NBCR(NR)=NBCR(NR1)
0020500      GOTO 1244
0020600 12345 DO 1245 NSYM=1,3
0020700      NSTREI=5*NPLY-1
0020800      N25L=4*NSL(NR)-2
0020900      N26=N25+N25L**2*IPR
0021000      N27=N26+N25L**2*IPR
0021100      N28=N27+N25L**2*IPR
0021200      N29=N28+N25L**2*IPR
0021300      N30=N29+N25L**2*IPR
0021400      N31=N30+NSTREI*NSTREI*IPR
0021500      N32=N31+NSTREI*IPR
0021600      N33=N32+(4*NL-2)*IPR
0021700      N34=N33+(4*NL-2)*1
0021800      IF(N34.GT.MAXRCH) WRITE(IN6,39846)N34,MAXRCH
0021900      IF(N34.GT.MAXRCH) STOP
0022000      CALL DEQMAT(NSYM,NL,NEQ,NBC,NG1,NG2,NR,NMT,NSTREI,H,
0022100      *RCH(1+21*NMT),RCH(1+31*NMT),RCH(1+33*NMT),RCH(1+35*NMT),
0022200      *RCH(1+37*NMT),THICK,SLN,CRT,CMODE,RCH(N25),
0022300      *RCH(N26),RCH(N27),RCH(N28),RCH(N29),RCH(N30),RCH(N31),RCH(N32),
0022400      *RCH(N33),RCH,
0022500      *NCN1,NCN2,NSREG,NMTYPE,NG4,N25L,IPR)
0022600      IF(NSYM.GT.1) GOTO 2356

```

```

0022700      NXEQJ=4*NSL(NR)-2
0022800      WRITE(IN6,1459) NR
0022900      WRITE(IN6,1458) (SLN(NG4+I),I=1,NXEQJ)
0023000      IF(NRGN.EQ.0)NBCR(NR)=(8*NSL(NR)-2)
0023100      IF(NRGN.NE.0)NBCR(NR)=(8*NSL(NR)-2-6)/2+6
0023200      NSR(NR)=NXEQJ
0023300      NG4=NG4+1
0023400      I1=NG2+1
0023500      NZT=6
0023600      I2=NG2+NZT
0023700      DO 191 I=I1,I2
0023800      CRT(I,1)=0.D 00
0023900      CRT(I,2)=0.D 00
0024000      WRITE(IN6,1463) I,NR
0024100      DO 191 J=1,NXEQJ
0024200 191   WRITE(IN6,870)(CMODE(I,J,K),K=1,4)
0024300      NZR(NR)=NZR(NR)+NZT
0024400      NG2=NG2+NZT
0024500 2356  IF(NPLY.EQ.1) GOTO 1245
0024600      DO 1369 NIRT=1,2
0024700      WRITE(IN8,2357)NR,NSYM
0024800      READ(IN7,2) NK10
0024900      IF(NIRT.EQ.1) WRITE(IN8,1367)
0025000      IF(NIRT.EQ.2) WRITE(IN8,1370)
0025100      READ(IN7,2) J
0025200      IF(J.NE.0) GOTO 1369
0025300      IF(NIRT.NE.1) GOTO 4884
0025400      N25L=4*NSL(NR)-2
0025500      N26=N25+N25L**2*IPR
0025600      N27=N26+N25L**2*IPR
0025700      N28=N27+N25L**2*IPR
0025800      N29=N28+N25L**2*IPR
0025900      N30=N29+N25L**2*IPR
0026000      N31=N30+N25L *IPR
0026100      N32=N31+N25L*IPR
0026200      N33=N32+N25L**2*IPR
0026300      IF(N33.GT.MAXRCH) WRITE(IN6,39845)N33,MAXRCH
0026400      IF(N33.GT.MAXRCH) STOP
0026500 39845  FORMAT(1X,'N33 = ',I7,' ',MAXRCH = ',I7)
0026600      CALL      RROOT(NSYM,NEQ,NT,NG2,NRGN,NK10,NRT,NCT,IN7,IN8,
0026700      *CRT,CMODE,RCH(N25),RCH(N26),RCH(N27),RCH(N28),RCH(N29),CRTA,
0026800      *RCH(N30),RCH(N31),RCH(N32),NCN1,NCN2,N25L)
0026900      GOTO 4885
0027000 4884  CONTINUE
0027100      N25L=4*NSL(NR)-2
0027200      N26=N25+N25L**2*IPR
0027300      N27=N26+N25L**2*IPR
0027400      N28=N27+N25L**2*IPR
0027500      N29=N28+N25L**2*IPR
0027600      N30=N29+N25L**2*IPR
0027700      N31=N30+N25L*2*IPR
0027800      N32=N31+N25L*2*IPR
0027900      N33=N32+N25L**2*2*IPR
0028000      N34=N33+N25L*IPR

```

```

0028100      IF(N34.GT.MAXRCH) WRITE(IN6,39846)N34,MAXRCH
0028200 39846 FORMAT(1X,'N34 = ',I7,' , MAXRCH = ',I7)
0028300      IF(N34.GT.MAXRCH) STOP
0028400      CPLL      CROOT(NSYM,NEQ,NT,NG2,NRGN,NK10,NRT,NCT,IN7,IN8,
0028500      *CRT,CMODE,RCH(N25),RCH(N26),RCH(N27),RCH(N28),RCH(N29),CRTA,
0028600      *RCH(N30),RCH(N31),RCH(N32),RCH(N33),NCN1,NCN2,N25L)
0028700 4885  IF(NSYM.EQ.3) GOTO 3241
0028800      IF(NIRT.EQ.1.AND.NRT.LT.1) GOTO 3241
0028900      IF(NIRT.EQ.2.AND.NCT.LT.2) GOTO 3241
0029000      NEVN=(NPLY+1)/2
0029100      N2EVN=(4*NPLY-2)/2+2
0029200      N3EVN=(4*NPLY-2)/2+1
0029300      N4EVN=(4*NPLY-2)
0029400      IF(NPLY/2*2.NE.NPLY.AND.NSYM.EQ.1) I3=N3EVN-1
0029500      IF(NPLY/2*2.NE.NPLY.AND.NSYM.EQ.2) I3=N3EVN
0029600      IF(NPLY/2*2.EQ.NPLY.AND.NSYM.EQ.1) I3=N3EVN
0029700      IF(NPLY/2*2.EQ.NPLY.AND.NSYM.EQ.2) I3=N3EVN-1
0029800      DO 7119 I=1,NEVN
0029900      J=4*(NPLY+1-I)
0030000      ISGN(J-3)= (4*I-3)*(-1)**(NSYM)
0030100      ISGN(J-2)= (4*I-2)*(-1)**(NSYM+1)
0030200      ISGN(J-5)= (4*I-1)*(-1)**(NSYM+1)
0030300 7119  ISGN(J-4)= (4*I-0)*(-1)**(NSYM)
0030400      I1=NG2+1
0030500      IF(NIRT.EQ.1) I2=NG2+NRT
0030600      IF(NIRT.EQ.2) I2=NG2+NCT
0030700      DO 424 I=I1,I2
0030800      DO 424 L =1,4
0030900      DO 7126 J=N2EVN,N4EVN
0031000      K=ISGN(J)
0031100      JSGN=K/IABS(K)
0031200      K=IABS(K)
0031300 7126  CMODE(I,J,L)=JSGN*CMODE(I,K,L)
0031400 424  CMODE(I,I3,L)=0.D 00
0031500 3241 IF(NIRT.EQ.1) WRITE(IN8,1323)
0031600      IF(NIRT.EQ.2) WRITE(IN8,1324)
0031700      READ (IN7,2 )J
0031800      IF(J.NE.0) GOTO 1369
0031900      IF(NIRT.EQ.1.AND.NRT.LT.1) GOTO 1369
0032000      IF(NIRT.EQ.2.AND.NCT.LT.2) GOTO 1369
0032100      I1=NG2+1
0032200      IF(NIRT.EQ.1) I2=NG2+NRT
0032300      IF(NIRT.EQ.2) I2=NG2+NCT
0032400      NXELQJ=4*NSL(NR)-2
0032500      DO 880 I3=I1,I2
0032600      WRITE(IN6,8610) I3,CRT(I3,1),CRT(I3,2) ,NR
0032700      DO 880 J=1,NXELQJ
0032800 880  WRITE(IN6,870) (CMODE(I3,J,K),K=1,4)
0032900      IF(NIRT.EQ.1) NG2=NG2+NRT
0033000      IF(NIRT.EQ.2) NG2=NG2+NCT
0033100      IF(NIRT.EQ.1) NRR(NR)=NRR(NR)+NRT
0033200      IF(NIRT.EQ.2) NCR(NR)=NCR(NR)+NCT
0033300      NRT=0
0033400      NCT=0

```

```

0033500 1369 CONTINUE
0033600 WRITE(IN8,483)
0033700 READ(IN7,2)J
0033800 IF(J.EQ.0) GOTO 1245
0033900 NRTNCT=NRT+NCT
0034000 I1=1
0034100 I2=1
0034200 I3=1
0034300 6736 IF(CRT(NG2+I1,2).NE.0.) GOTO 6734
0034400 CRTA(I2+0)=CRT(NG2+I1,1)
0034500 CRTA(I2+1)=CRT(NG2+I1,1)
0034600 I1=I1+2
0034700 I2=I2+2
0034800 GOTO 6735
0034900 6734 DO 6847 I=1,4
0035000 CRTB(I3+I-1)=CRT(NG2+I1,1)
0035100 6847 CRIC(I3+I-1)=CRT(NG2+I1,2)
0035200 I3=I3+4
0035300 I1=I1+4
0035400 6735 IF(I1.GT.NRTNCT) GOTO 6737
0035500 GOTO 6736
0035600 6737 IF(NRT.LT.2) GOTO 6589
0035700 DO 6738 I1=1,NRT
0035800 CRT(NG2+I1,1)=CRTA(I1)
0035900 6738 CRT(NG2+I1,2)=0.D 00
0036000 6589 IF(NCT.LT.4) GOTO 6588
0036100 DO 6739 I1=1,NCT
0036200 CRT(NG2+NRT+I1,1)=CRTB(I1)
0036300 6739 CRT(NG2+NRT+I1,2)=CRIC(I1)
0036400 6588 CONTINUE
0036500 DO 1906 I=1,NRT
0036600 1906 WRITE(IN8,1908) NRT,I,CRT(NG2+I,1),CRT(NG2+I,2)
0036700 DO 1907 I=1,NCT
0036800 1907 WRITE(IN8,1908) NCT,I,CRT(NG2+NRT+I,1),CRT(NG2+NRT+I,2)
0036900 WRITE(IN8,1943)
0037000 READ(IN7,2)NRTXL
0037100 IF(NRTXL.LT.0) NRT=0
0037200 IF(NRTXL.GT.0) NRT=NRTXL
0037300 WRITE(IN8,1944)
0037400 READ(IN7,2)NCTXL
0037500 IF(NCTXL.LT.0) NCT=0
0037600 IF(NCTXL.GT.0) NCT=NCTXL
0037700 WRITE(IN8,1909)
0037800 READ(IN7,2) J
0037900 IF(J.EQ.0) GOTO 2356
0038000 IF(NSYM.EQ.1) NT=((NBC-1)-NCT-NRT-4+2)/2
0038100 IF(NSYM.EQ.2) NT=((NBC-1)-NCT-NRT-2+2)/2
0038200 IF(NSYM.EQ.3) NT=((NBC-0)-NCT-NRT-6+2)/2
0038300 CALL PZERO(CRTA,NCN1)
0038400 NK10=0
0038500 N25L=4*NSL(NR)-2
0038600 N26=N25+N25L**2*IPR
0038700 N27=N26+N25L**2*IPR
0038800 N28=N27+N25L**2*IPR

```

```

0038900      N29=N28+N25L**2*IPR
0039000      N30=N29+N25L**2*IPR
0039100      N31=N30+N25L**2*IPR
0039200      CALL COEFF1(NSYM,CRTA,NEQ,NCT,NRT,NRGN,NT,NG2,RCH(N31),RCH(N25),
0039300      *RCH(N26),RCH(N27),RCH(N28),RCH(N29),CRT,NK10,NCN1,RCH(N30),N25L)
0039400      WRITE(IN8,44844)( (I,CRTA(I)),I=1,NT)
0039500      GOTO 2356
0039600 1245  CONTINUE
0039700 1244  CONTINUE
0039800      WRITE(IN6,1654) (NSL(I),I=1,NSREG)
0039900      WRITE(IN6,1656) (NZR(I),I=1,NSREG)
0040000      WRITE(IN6,1657) (NRR(I),I=1,NSREG)
0040100      WRITE(IN6,1658) (NCR(I),I=1,NSREG)
0040200      WRITE(IN6,1659) (NBCR(I),I=1,NSREG)
0040300      WRITE(IN6,1652) (NSR(I),I=1,NSREG)
0040400      NTBC=0
0040500      DO 1411 I=1,NSREG
0040600 1411  NTBC=NTBC+NZR(I)+NRR(I)+NCR(I)
0040700      WRITE(IN6,8603) NTBC
0040800      CALL PZERO(SK,NCN1*NCN1)
0040900      CALL PZERO(RK,NCN1)
0041000      DO 1413 M1=1,NTBC
0041100      IF(M1.EQ.1) WRITE(IN6,8606)
0041200      M2=1
0041300 14442 READ(IN5,8605)NBCREG,NBCPLY,NBCV,YNBC,ZNBC,PNBC,NBCRG1
0041400 C      IF(M2.EQ.1) WRITE(IN6,6667)
0041500      WRITE(IN6,8605)NBCREG,NBCPLY,NBCV,YNBC,ZNBC,PNBC,NBCRG1
0041600      NR1=NBCREG
0041700      NR=NGT(NR1)
0041800      IF(NR.EQ.0) NR=NR1
0041900      NRGN=NGN(NR)
0042000      NPLY=NBCPLY
0042100      NV=NBCV
0042200      PS=PNBC
0042300      SSSGN=1.
0042400      SSSGN1=1.
0042500      IF(M2.GT.1) SSSGN=PS
0042600      IF(M2.GT.1) SSSGN1=PS
0042700      IF(SSSGN.EQ.0.) SSSGN=1.
0042800      IF(SSSGN1.EQ.0.) SSSGN1=1.
0042900      IF(M2.GT.1) PS=0.D 00
0043000      NV=IABS(NV)
0043100      Y=YNBC
0043200      Z=ZNBC
0043300      H=THCK(NR)
0043400      NL=NSL(NR)
0043500      NV1=NV
0043600      IF(NRGN.NE.1) GOTO 64851
0043700      IF(DABS(Y).LT.1.E-10) GOTO 64851
0043800      IF(NV1.NE.1) GOTO 64852
0043900      NPLY=1
0044000      GOTO 64851
0044100 64852 IF(NV1.NE.3) GOTO 64854
0044200      NPLY=1

```

```

0044300      NV=1
0044400      GOTO 64851
0044500 64853 NPLY=1
0044600      NV=3
0044700      GOTO 64851
0044800 64854 IF(NV1.NE.5) GOTO 64851
0044900      NV=5
0045000      NPLY=1
0045100 64851 NG1=0
0045200      NG2=0
0045300      NG3=0
0045400      NG4=0
0045500      NCT=NCR(NR)
0045600      NZT=NZR(NR)
0045700      NRT=NRR(NR)
0045800      NST=NSR(NR)
0045900      DO 1414 I=1,NR1
0046000      IF(I.EQ.1) GOTO 1414
0046100      IF(I.GT.NR) GOTO 1414
0046200      NG1=NG1+NSL(I-1)
0046300      IF(NGT(I).EQ.0.OR.NGT(I).EQ.NR)NG2=NG2+NZR(I-1)+NRR(I-1)+NCR(I-1)
0046400      IF(NGT(I).EQ.0.OR.NGT(I).EQ.NR)NG4=NG4+NSR(I-1)
0046500      NG3=NG3+NBCR(I-1)
0046600 1414 CONTINUE
0046700      NG1=NG1+NPLY
0046800      IF(NRGN.NE.1) GOTO 66710
0046900      IF(DABS(Y).LT.1.E-10) GOTO 66710
0047000      IF(NV1.NE.3) GOTO 66710
0047100      IF(NV.EQ.3) SSSGN=SSSGN1
0047200      IF(NV.NE.1) GOTO 66710
0047300      ECCN=H/2
0047400      DO 66711 ICFG54=1,NPLY
0047500 66711 ECCN=ECCN-THICK(NG1-NPLY+ICFG54)
0047600      ECCN=ECCN+THICK(NG1)/2
0047700      SSSGN=SSSGN1*ECCN
0047800 66710 I111=NG3+1
0047900      I112=NG3+NBCR(NR)
0048000      DO 1415 N1=I111,I112
0048100      NG2=NG2+1
0048200      IF(CRT(NG2,1).EQ.0..AND.CRT(NG2,2).EQ.0.) MT=1
0048300      IF(CRT(NG2,1).NE.0..AND.CRT(NG2,2).EQ.0.) MT=2
0048400      IF(CRT(NG2,1).NE.0..AND.CRT(NG2,2).NE.0.) MT=3
0048500      NTH=IPR*NMT
0048600      CALL FUN(NL,NPLY,NG1,NG2,NG4,NRGN,NV,MT,NMT,H,Y,Z,F,FPS,
0048700      *RCH(1+18*NTH),RCH(1+19*NTH),RCH(1+21*NTH),RCH(1+25*NTH),
0048800      *RCH(1+27*NTH),RCH(1+29*NTH),RCH(1+31*NTH),RCH(1+33*NTH),
0048900      *RCH(1+35*NTH),RCH(1+37*NTH),RCH(1+39*NTH),RCH(2+39*NTH),
0049000      *THICK ,NMTYPE,
0049100      *SLN,CRT,CMODE,RCH,NCN1,NCN2,IPR)
0049200      FPS1=FPS
0049300      SK(M1,N1)=SK(M1,N1)+SSSGN*F
0049400 C      WRITE(IN6,6666) NR,NPLY,NG1,NG2,MT,M1,N1,NV,Y,Z,F,FPS,SSSGN,PS
0049500 6667  FORMAT(/1X,'NR,NPLY,NG1,NG2,MT,M1,N1,NV,Y,Z,F,FPS,SSSGN,PS')
0049600 6666  FORMAT(1X,8I3,6E9.3)

```



```

0049700 1415 CONTINUE
0049800 IF(NRGN.NE.1) GOTO 14140
0049900 IF(DABS(Y).LT.1.E-10) GOTO 14140
0050000 IF(NV1.NE.1 ) GOTO 69941
0050100 NPLY=NPLY+1
0050200 RK(M1)=RK(M1)-SSSGN*FPS1
0050300 IF(M2.EQ.1.AND.NPLY.EQ.NL) RK(M1)=RK(M1)-PS
0050400 IF(NPLY.GT.NL) GOTO 14131
0050500 GOTO 64851
0050600 69941 IF(NV1.NE.3) GOTO 69942
0050700 NPLY=NPLY+1
0050800 IF(NV.NE.1) GOTO 69943
0050900 RK(M1)=RK(M1)-SSSGN*FPS1
0051000 IF(M2.EQ.1.AND.NPLY.EQ.NL) RK(M1)=RK(M1)-PS
0051100 IF(NPLY.GT.NL) GOTO 64853
0051200 GOTO 64851
0051300 69943 RK(M1)=RK(M1)-SSSGN*FPS1
0051400 IF(M2.EQ.1.AND.NPLY.EQ.NL) RK(M1)=RK(M1)-PS
0051500 IF(NPLY.GT.NL) GOTO 14131
0051600 GOTO 64851
0051700 69942 IF(NV1.NE.5) GOTO 14140
0051800 NPLY=NPLY+1
0051900 RK(M1)=RK(M1)-SSSGN*FPS1
0052000 IF(M2.EQ.1.AND.NPLY.EQ.NL) RK(M1)=RK(M1)-PS
0052100 IF(NPLY.GT.NL) GOTO 14131
0052200 GOTO 64851
0052300 14140 RK(M1)=RK(M1)-SSSGN*FPS1
0052400 IF(M2.EQ.1) RK(M1)=RK(M1)-PS
0052500 14131 M2=M2+1
0052600 IF(NBCRG1.EQ.0) GOTO 1413
0052700 GOTO 14442
0052800 1413 CONTINUE
0052900 CALL SKINV(SK,NCN1,SK,NCN1,RK,NCN1,1,SIMUL,NTBC)
0053000 386 READ (IN5,402)NDREG,NDPLY,NDV,NDVT,YND,ZND,STPND,NPTS
0053100 IF(NDREG.EQ.0) GOTO 387
0053200 WRITE(IN6,339)
0053300 Y=YND
0053400 Z=ZND
0053500 N=1
0053600 704 FD=0.D 00
0053700 NR1=NDREG
0053800 NR=NGT(NR1)
0053900 IF(NR.EQ.0) NR=NR1
0054000 NRGN=NGN(NR)
0054100 NPLY=NDPLY
0054200 NV=NDV
0054300 H=THCK(NR)
0054400 NL=NSL(NR)
0054500 NG1=0
0054600 NG2=0
0054700 NG3=0
0054800 NG4=0
0054900 NCT=NCR(NR)
0055000 NZT=NZR(NR)

```

```

0055100      NRT=NRR(NR)
0055200      NST=NSR(NR)
0055300      DO 373 J=1,NR1
0055400      IF(J.EQ.1) GOTO 373
0055500      IF(J.GT.NR) GOTO 373
0055600      NG1=NG1+NSL(J-1)
0055700      IF(NGT(J).EQ.0.OR.NGT(J).EQ.NR)NG2=NG2+NZR(J-1)+NRR(J-1)+NCR(J-1)
0055800      IF(NGT(J).EQ.0.OR.NGT(J).EQ.NR)NG4=NG4+NSR(J-1)
0055900      NG3=NG3+NBCR(J-1)
0056000 373   CONTINUE
0056100      NG1=NG1+NPLY
0056200      I111=NG3+1
0056300      I112=NG3+NBCR(NR)
0056400      DO 703 N1=I111,I112
0056500      NG2=NG2+1
0056600      IF(CRT(NG2,1).EQ.0..AND.CRT(NG2,2).EQ.0.) MT=1
0056700      IF(CRT(NG2,1).NE.0..AND.CRT(NG2,2).EQ.0.) MT=2
0056800      IF(CRT(NG2,1).NE.0..AND.CRT(NG2,2).NE.0.) MT=3
0056900      NTH=IPR*NMT
0057000      CALL FUN(NL,NPLY,NG1,NG2,NG4,NRGN,NV,MT,NMT,H,Y,Z,F,FPS,
0057100      *RCH(1+18*NTH),RCH(1+19*NTH),RCH(1+21*NTH),RCH(1+25*NTH),
0057200      *RCH(1+27*NTH),RCH(1+29*NTH),RCH(1+31*NTH),RCH(1+33*NTH),
0057300      *RCH(1+35*NTH),RCH(1+37*NTH),RCH(1+39*NTH),RCH(2+39*NTH),
0057400      *THICK ,NMTYPE,
0057500      *SLN,CRT,CMODE,RCH,NCN1,NCN2,IPR)
0057600      FPS1=FPS
0057700      IF(N1.EQ.I112) FD=FD+FPS1
0057800 703   FD=FD+RK(N1)*F
0057900      WRITE(IN6,338) Y,FD,Z,NR,NPLY,NV
0058000      N=N+1
0058100      IF(N.GT.1.AND.NDVT.EQ.2) Y=Y+STPND
0058200      IF(N.GT.1.AND.NDVT.EQ.3) Z=Z+STPND
0058300      IF(N.LE.NPTS+1) GOTO 704
0058400      GOTO 386
0058500 387   CONTINUE
0058600 1652  FORMAT('1',/1X,' NO. OF VARIABLES IN EACH REGION      ',8(1X,I3))
0058700 1654  FORMAT(1X,' NO. OF PLYS IN EACH REGION              ',8(1X,I3))
0058800 1656  FORMAT(1X,' NO. OF ZERO ROOTS IN EACH REGION        ',8(1X,I3))
0058900 1657  FORMAT(1X,' NO. OF REAL ROOTS IN EACH REGION       ',8(1X,I3))
0059000 1658  FORMAT(1X,' NO. OF COMP. ROOTS IN EACH REGION      ',8(1X,I3))
0059100 1659  FORMAT(1X,' NO. OF B. CONDITIONS IN EACH REGION',8(1X,I3))
0059200 1909  FORMAT(1X,'GIVE 0 IF POLYNOMIAL IS NOT DESIRED')
0059300 1943  FORMAT(1X,'GIVE NRT FOR POLYNOMIAL....DEFAULT IS   AS IS'/1X,
0059400      *'A NEGATIVE VALUE INITIATES NRT TO 0')
0059500 1944  FORMAT(1X,'GIVE NCT FOR POLYNOMIAL....DEFAULT IS   AS IS'/1X,
0059600      *'A NEGATIVE VALUE INITIATES NCT TO 0')
0059700 339   FORMAT(/1X,'Y,F,Z,NR,NPLY,NV')
0059800 8606  FORMAT('1',/1X,'NBCREG,NBCPLY,NBCV,YNBC,ZNBC,PNBC,NBCREG')
0059900 8603  FORMAT(/1X,I4,' = NTBC')
0060000 483   FORMAT(/1X,'GIVE 0 IF NO POLYNOMIAL DESIRED')
0060100 1324  FORMAT(1X,'GIVE 0 IF COMPLEX ROOTS ARE TO BE STORED')
0060200 1370  FORMAT(1X,'GIVE 0 IF YOU WANT COMPLEX ROOTS')
0060300 8610  FORMAT(/1X,I4,1X,2D15.8,' EIGEN VALUE AND MODE FOR REGION = ',I3)
0060400 870   FORMAT(6(1X,F17.6))

```

```

0060500 1323  FORMAT(1X,'GIVE 0  IF REAL ROOTS ARE TO BE STORED')
0060600 1367  FORMAT(1X,'GIVE 0  IF YOU WANT REAL  ROOTS')
0060700 2357  FORMAT(1X,'GIVE 0  IF YOU DO NOT WANT POLYNOMIAL APPROX.'/'
0060800      *1X,'NR = ',I3,'  NSYM = ',I3)
0060900 1463  FORMAT(/1X,I4,' ZERO MODE,  ',I3,' REGION')
0061000 1459  FORMAT('1',/1X,'PARTICULAR SOLUTION FOR REGION = ',I2)
0061100 8605  FORMAT(3I4,3D12.6,3I4,2D12.6)
0061200 402   FORMAT(4I4,3D12.6,I4,I4)
0061300 338   FORMAT(3(1X,F17.6),3(1X,I3,1X))
0061400 1908  FORMAT(1X,2(I3,1X),2(D15.8,1X))
0061500 44844 FORMAT(I3,2X,D23.16)
0061600 1458  FORMAT(10(1X,D10.4))
0061700 123   FORMAT(8(1X,D9.3))
0061800 12    FORMAT(6(1X,E10.4),/3(1X,E10.4))
0061900 435   FORMAT(8D10.3)
0062000 2     FORMAT(20I2)
0062100      RETURN
0062200      END
0062300      SUBROUTINE  MPRO1(E11,E22,E33,NU12,NU13,NU23,G44,G55,G66,S31,S32, -
0062400      *S33,S44,S45,S55,CB1,CB2,CB6,CB11,CB12,CB16,CB22,CB26,CB66,SNC1, -
0062500      *SNC2,SHM1,SHM2,SHQ1,SHQ2,KQ1,KQ2,KH1,KH2,KP1,KP2,KN1,KN2,THETA, -
0062600      *N,IN5,IN6,IN7,IN8)
0062700      IMPLICIT REAL  (A-H,O-Z)
0062800      REAL  KH1,KH2,KN1,KN2,KQ1,KQ2,KP1,KP2,NU12,NU13,NU23
0062900      DIMENSION E11(1),E22(1),E33(1),NU12(1),NU13(1),NU23(1),G44(1), -
0063000      *G55(1),G66(1),S31(1),S32(1),S33(1),S44(1),S45(1),S55(1),CP1(1), -
0063100      *CB2(1),CB6(1),CB11(1),CB12(1),CB16(1),CB22(1),CB26(1),CB66(1), -
0063200      *SNC1(1),SNC2(1),SHM1(1),SHM2(1),SHQ1(1),SHQ2(1),KQ1(1),KQ2(1), -
0063300      *KH1(1),KH2(1),KP1(1),KP2(1),KN1(1),KN2(1),THETA(1) -
0063400      *,T1(6,6),T2(6,6),T1I(6,6),ST(6,6),SF(6,6),VC6(6)
0063500 C.....LAYER FLEXIBILITI MATRICES
0063600      PI=3.141592653589793
0063700      WRITE(IN6,149)
0063800 149  FORMAT('1')
0063900      WRITE(IN6,5)
0064000      DO 1242 K=1,N
0064100      READ(IN5,4) E11(K),E22(K),E33(K),NU12(K),NU13(K),NU23(K),G44(K), -
0064200      *G55(K),G66(K),THETA(K)
0064300 1242 WRITE(IN6,123)E11(K),E22(K),E33(K),NU12(K),NU13(K),NU23(K),G44(K), -
0064400      *G55(K),G66(K),THETA(K)
0064500 5    FORMAT(/1X,'E11,E22,E33,NU12,NU13,NU23,G44,G55,G66,THETA')
0064600 4    FORMAT(10F8.4)
0064700      DO 101 K=1,N
0064800      CALL PZERO(ST,36)
0064900      ST(1,1)=1./E11(K)
0065000      ST(1,2)=1./E11(K)*NU12(K)*(-1)
0065100      ST(1,3)=1./E11(K)*NU13(K)*(-1)
0065200      ST(2,1)=ST(1,2)
0065300      ST(2,2)=1./E22(K)
0065400      ST(2,3)=1./E22(K)*NU23(K)*(-1)
0065500      ST(3,1)=ST(1,3)
0065600      ST(3,2)=ST(2,3)
0065700      ST(3,3)=1./E33(K)
0065800      ST(4,4)=1./G44(K)

```

```

0065900      ST(5,5)=1./G55(K)
0066000      ST(6,6)=1./G66(K)
0066100      CALL PZERO(T1,36)
0066200      CO=DCOS(THETA(K)*PI/180)
0066300      SI=DSIN(THETA(K)*PI/180)
0066400      T1(1,1)=CO*CO
0066500      T1(1,2)=SI*SI
0066600      T1(1,6)=-2.*CO*SI
0066700      T1(2,1)=SI*SI
0066800      T1(2,2)=CO*CO
0066900      T1(2,6)=2*CO*SI
0067000      T1(3,3)=1
0067100      T1(4,4)=CO
0067200      T1(4,5)=SI
0067300      T1(5,4)=-SI
0067400      T1(5,5)=CO
0067500      T1(6,1)=CO*SI
0067600      T1(6,2)=-CO*SI
0067700      T1(6,6)=CO*CO-SI*SI
0067800      CALL PZERO(T2,36)
0067900      CALL PZERO(SF,36)
0068000      CALL SKINV(T1,6,T2,6,VC6,6,-1,SIMUL,6)
0068100      CALL BIDOT(ST,6,T2,6,SF,6)
0068200      DO 102 I=1,6
0068300      DO 102 J=1,6
0068400 102   T2(I,J)=T1(I,J)
0068500      T2(6,1)=2.*CO*SI
0068600      T2(6,2)=-2.*CO*SI
0068700      T2(1,6)=-CO*SI
0068800      T2(2,6)=CO*SI
0068900      CALL BIDOT(T2,6,SF,6,ST,6)
0069000      WRITE(IN6,105)K
0069100 105   FORMAT(/1X,'FLEXIBILITY MATRIX S FOR ',I2,' MATERIAL')
0069200      DO 1010 I=1,6
0069300 1010  WRITE(IN6,123)ST(I,1),ST(I,2),ST(I,3),ST(I,4),ST(I,5),ST(I,6)
0069400 C.....LAYER STIFFNESS MATRICES....C
0069500      CALL SKINV(ST,6,SF,6,VC6,6,-1,SIMUL,6)
0069600      WRITE(IN6,112) K
0069700 112   FORMAT(/1X,'STIFFNESS MATRIX C FOR ',I2,' MATERIAL')
0069800      DO 108 I=1,6
0069900 108   WRITE(IN6,123)SF(I,1),SF(I,2),SF(I,3),SF(I,4),SF(I,5),SF(I,6)
0070000 C.....PLANESTRESS STIFFNESS MATRIX
0070100      CALL PZERO(T2,36)
0070200      CALL PZERO(T1,36)
0070300      T1(1,1)=ST(1,1)
0070400      T1(1,2)=ST(1,2)
0070500      T1(1,3)=ST(1,6)
0070600      T1(2,1)=ST(2,1)
0070700      T1(2,2)=ST(2,2)
0070800      T1(2,3)=ST(2,6)
0070900      T1(3,1)=ST(6,1)
0071000      T1(3,2)=ST(6,2)
0071100      T1(3,3)=ST(6,6)
0071200      T1(4,4)=1.

```

```

0071300      T1(5,5)=1.
0071400      T1(6,6)=1.
0071500      CALL SKINV(T1,6,T2,6,VC6,6,-1,SIMUL,6)
0071600      CALL PZERO(T1,36)
0071700      T1(1,1)=-ST(1,3)
0071800      T1(2,1)=-ST(2,3)
0071900      T1(3,1)=-ST(6,3)
0072000      CALL BIDOT(T2,6,T1,6,T1I,6)
0072100      WRITE(IN6,115)
0072200 115   FORMAT(/1X,'PLANESTRESS STIFFNESS MATRIX CBB, CB OF ',I2,' LAYER')
0072300      DO 113 I=1,3
0072400 113   WRITE(IN6,123) T2(I,1),T2(I,2),T2(I,3),T1I(I,1)
0072500 C.....MATERIAL PROPERTIES FOR PLY LEVEL ANALYSIS
0072600      S31(K)=ST(3,1)
0072700      S32(K)=ST(3,2)
0072800      S33(K)=ST(3,3)
0072900      S44(K)=ST(4,4)
0073000      S45(K)=ST(4,5)
0073100      S55(K)=ST(5,5)
0073200      CB1(K)=T1I(1,1)
0073300      CB2(K)=T1I(2,1)
0073400      CB6(K)=T1I(3,3)
0073500      CB11(K)=T2(1,1)
0073600      CB12(K)=T2(1,2)
0073700      CB16(K)=T2(1,3)
0073800      CB22(K)=T2(2,2)
0073900      CB26(K)=T2(2,3)
0074000      CB66(K)=T2(3,3)
0074100      SNC1(K)=CB12(K)/CB22(K)*(S31(K)*CB12(K)+S32(K)*CB22(K))+CB12(K)*S4-
0074200      *4(K)-CB1(K)
0074300      SNC2(K)=CB22(K)/CB22(K)*(S31(K)*CB12(K)+S32(K)*CB22(K))+CB22(K)*S4-
0074400      *4(K)-CB2(K)
0074500      KP1(K)=CB1(K)/2
0074600      KP2(K)=CB2(K)/2
0074700      SHQ1(K)=S44(K)*(-.25)
0074800      SHQ2(K)=S44(K)*.75
0074900      KQ1(K)=(.25*CB12(K)*(S31(K)*CB12(K)/CB22(K) +S32(K))-CB12(K)*S44(K)-
0075000      *)+CB1(K))* .6
0075100      KQ2(K)=(.25*CB22(K)*(S31(K)*CB12(K)/CB22(K) +S32(K))-CB22(K)*S44(K)-
0075200      *)+CB2(K))* .6
0075300      KM1(K)=(1.5*CB12(K)*(S31(K)*CB12(K)/CB22(K) +S32(K))-CB12(K)*S44(K)-
0075400      *)+CB1(K))* .1
0075500      KM2(K)=(1.5*CB22(K)*(S31(K)*CB12(K)/CB22(K) +S32(K))-CB22(K)*S44(K)-
0075600      *)+CB2(K))* .1
0075700      KN1(K)=(CB12(K)*(S31(K)*CB12(K)/CB22(K) +S32(K))+CB12(K)*S44(K)+2*-
0075800      *CB1(K))/12
0075900 101   KN2(K)=(CB22(K)*(S31(K)*CB12(K)/CB22(K) +S32(K))+CB22(K)*S44(K)+2*-
0076000      *CB2(K))/12
0076100      WRITE(IN6,125)
0076200 125   FORMAT(/1X,'CB11, CB12, CB16, CB22, CB26, CB66, CB1, CB2, CB6')
0076300 123   FORMAT(1X,8E10.4)
0076400      DO 126 K=1,N
0076500 126   WRITE(IN6,123)CB11(K),CB12(K),CB16(K),CB22(K),CB26(K),CB66(K),
0076600      *CB1(K),CB2(K),CB6(K)

```

```

0076700      WRITE(IN6,129)
0076800 129   FORMAT(/1X,'S31,S32,S33,S44,S45,S55,SNC1,SNC2,SHM2,SHQ2')
0076900      DO 130 K=1,N
0077000 130   WRITE(IN6,123)S31(K),S32(K),S33(K),S44(K),S45(K),S55(K),SNC1(K), -
0077100      *SNC2(K),SHM2(K),SHQ2(K)
0077200      WRITE(IN6,119)
0077300 119   FORMAT(/1X,'KN1,KN2,KP1,KP2,KM1,KM2,KQ1,KQ2')
0077400      DO 122 K=1,N
0077500 122   WRITE(IN6,123) KN1(K),KN2(K),KP1(K),KP2(K),KM1(K),KM2(K),KQ1(K), -
0077600      *KQ2(K)
0077700      RETURN
0077800      END
0077900      FUNCTION WVD(N,I,Z,J,RCM,IPR)
0078000      IMPLICIT REAL (A-H,O-Z)
0078100      REAL RCM
0078200      DIMENSION RCM(1)
0078300      M=J*IPR
0078400      WVD= WVD1(N,I,Z, RCM(1),RCM(1+M),RCM(1+2*M),RCM(1+3*M),RCM(1+4*M)-
0078500      *,RCM(1+5*M),RCM(1+6*M),RCM(1+7*M),RCM(1+8*M),RCM(1+9*M) -
0078600      *,RCM(1+10*M),RCM(1+11*M),RCM(1+12*M),RCM(1+13*M),RCM(1+14*M) -
0078700      *,RCM(1+15*M),RCM(1+16*M),RCM(1+17*M),RCM(1+18*M),RCM(1+19*M) -
0078800      *,RCM(1+20*M),RCM(1+21*M),RCM(1+22*M),RCM(1+23*M),RCM(1+24*M) -
0078900      *,RCM(1+25*M),RCM(1+26*M),RCM(1+27*M),RCM(1+28*M),RCM(1+29*M) -
0079000      *,RCM(1+30*M),RCM(1+31*M),RCM(1+32*M),RCM(1+33*M),RCM(1+34*M) -
0079100      *,RCM(1+35*M),RCM(1+36*M),RCM(1+37*M),RCM(1+39*M),RCM(2+39*M))
0079200      RETURN
0079300      END
0079400      FUNCTION WVD1(N,I,Z, E11,E22,E33,NU12,NU13,NU23,G44,G55,G66,S31, -
0079500      *S32,S33,S44,S45,S55,CB1,CB2,CB6,CB11,CB12,CB16,CB22,CB26,CB66,SNC1-
0079600      *,SNC2,SHM1,SHM2,SHQ1,SHQ2,KQ1,KQ2,KM1,KM2,KP1,KP2,KN1,KN2,EPSLON, -
0079700      *CURVTR)
0079800      IMPLICIT REAL (A-H,O-Z)
0079900      REAL KM1,KM2,KN1,KN2,KQ1,KQ2,KP1,KP2,NU12,NU13,NU23
0080000      DIMENSION E11(1),E22(1),E33(1),NU12(1),NU13(1),NU23(1),G44(1), -
0080100      *G55(1),G66(1),S31(1),S32(1),S33(1),S44(1),S45(1),S55(1),CB1(1), -
0080200      *CB2(1),CB6(1),CB11(1),CB12(1),CB16(1),CB22(1),CB26(1),CB66(1), -
0080300      *SNC1(1),SNC2(1),SHM1(1),SHM2(1),SHQ1(1),SHQ2(1),KQ1(1),KQ2(1), -
0080400      *KM1(1),KM2(1),KP1(1),KP2(1),KN1(1),KN2(1)
0080500      GOTO(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18),N
0080600 1      WVD1=1
0080700      RETURN
0080800 2      WVD1 =-(S31(I)*CB12(I)+S32(I)*CB22(I))/2*Z*Z
0080900      RETURN
0081000 3      WVD1 =(S31(I)*CB11(I)+S32(I)*CB12(I))*(Z*EPSLON-Z*Z/2*CURVTR)
0081100 C.....HERE Z SHOULD HAVE A DIMENSION OF A LENGTH AND W,XX SHOULD
0081200 C.....BE GIVEN AS IS WITHOUT ANY NONDIMENSIONALIZATION
0081300      RETURN
0081400 4      WVD1 =(S31(I)*CB12(I)+S32(I)*CB22(I))*Z
0081500      RETURN
0081600 5      WVD1 =(S31(I)*SNC1(I)+S32(I)*SNC2(I))/12*(Z*Z*Z-Z) -
0081700      *-S33(I)/12*(Z**3-3*Z)+(S31(I)*KN1(I)+S32(I)*KN2(I))*Z
0081800      RETURN
0081900 6      WVD1 =(S31(I)*SNC1(I)+S32(I)*SNC2(I))/4*(Z**4/4-3./10*Z*Z) -
0082000      *-S33(I)/4*(Z**4/4-Z*Z/2)+(S31(I)*KM1(I)+S32(I)*KM2(I))/2*Z*Z

```

```

0082100      RETURN
0082200  7    WVD1   =(S31(I)*KP1(I)+S32(I)*KP2(I))*Z+S33(I)*Z/2
0082300      RETURN
0082400  8    WVD1   =(S31(I)*KQ1(I)+S32(I)*KQ2(I))/2*Z*Z-S33(I)/4*(Z**4/4-Z*Z/2) -
0082500      *+(S31(I)*SNC1(I)+S32(I)*SNC2(I))/4*(Z**4/4-Z*Z**3/10)+S33(I)/4*Z*Z
0082600      RETURN
0082700  9    WVD1   =-Z
0082800      RETURN
0082900 10    WVD1   =(S31(I)*CB12(I)+S32(I)*CB22(I))/6*Z**3+S44(I)*CB22(I)/6*(-
0083000      *Z**3-3*Z)
0083100      RETURN
0083200 11    WVD1   =1
0083300      RETURN
0083400 12    WVD1   =-S31(I)*CB12(I)/2*Z*Z
0083500      RETURN
0083600 13    WVD1   =(S44(I)+S32(I))/4*Z*Z
0083700      RETURN
0083800 14    WVD1   =-S31(I)*KN1(I)/2*Z*Z
0083900      *-(S31(I)*SNC1(I)+S32(I)*SNC2(I))/12*(Z**4/4-Z*Z/2)
0084000      *+S33(I)/12*(Z**4/4-3./2*Z*Z)+0*S44(I)/4*SNC2(I)*(-Z**4/12+Z**2/6)
0084100      RETURN
0084200 15    WVD1   =S44(I)/2*Z
0084300      RETURN
0084400 16    WVD1   =-(S31(I)*KM1(I)+S32(I)*KM2(I))/6*Z**3
0084500      *
0084600      *      -S44(I)*KM2(I)/6*(Z**3-3*Z)
0084700      *
0084800      *      -(S31(I)*SNC1(I)+S32(I)*SNC2(I))/4*(Z**5/20-Z**3/10)
0084900      *
0085000      *      +S33(I)/4*(Z**5/20-Z**3/6)
0085100      *
0085200      *      +0*S44(I)/4*SNC2(I)*(-Z**5/20+Z**3/10-Z/20)
0085300      RETURN
0085400 17    WVD1   =-S31(I)*KP1(I)*Z*Z/2-S33(I)*Z*Z/4
0085500      RETURN
0085600 18    WVD1   =-(S31(I)*KQ1(I)+S32(I)*KQ2(I))/6*Z**3
0085700      *
0085800      *      -S44(I)*KQ2(I)/6*(Z**3-3*Z)
0085900      *
0086000      *      -(S31(I)*SNC1(I)+S32(I)*SNC2(I))/4*(Z**5/20-Z**3/10)
0086100      *
0086200      *      -S33(I)*Z**3/12
0086300      *
0086400      *      +S33(I)*(Z**5/20-Z**3/6)/4
0086500      *
0086600      *      +0*S44(I)/4*SNC2(I)*(-Z**5/20+Z**3/10-Z/20)
0086700      RETURN
0086800      END
0086900      SUBROUTINE CDET(NSYM,COEF,X,F,NSDFG,CVC30,NPIVOT,NK10,
0087000      *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,WA,NCN1,N25L)
0087100      IMPLICIT REAL (A-H,O-Z)
0087200      COMPLEX A(N25L,1),CVC30(1),X,F,X2347,DCMPLX
0087300      DIMENSION WA(1),COEF(1),SK0(N25L,1),SK1(N25L,1),SK2(N25L,1),
0087400      *SK3(N25L,1),SK4(N25L,1),CRT(NCN1,1)
0087500      IF(NSDFG.NE.0) GOTO 8
0087600      IF(NK10.EQ.0) GOTO 8
0087700      F=(0.D 00,0.D 00)
0087800      DO 80 I=1,NT
0087900      F=F+COEF(I)*X**(2*I-2)
0088000      RETURN
0088100      DO 2 I=1,N
0088200      DO 2 J=1,N
0088300      A(I,J)=SK0(I,J)+SK1(I,J)*X+SK2(I,J)*X*X +SK3(I,J)*X*X*X

```

```

0087500      *+SK4(I,J)*X*X*X*X
0087600      IF(NSDFG.NE.0) GOTO 3
0087700      CALL LEQT1C(A,N,N25L,CVC30,1,N25L,0,WA,IER)
0087800      F=(1.D 00,0.D 00)
0087900      DO 1 I=1,N
0088000      IPVT=WA(I)
0088100      IF(IPVT.NE.I) F=-F
0088200 1      F=F*A(I,I)
0088300      IF(NPIVOT.EQ.11000) RETURN
0088400      Y=1.D 00
0088500      NRTNCT=NRT+NCT
0088600      IF(NRTNCT.LI.2) GOTO 232
0088700      I1=1
0088800 215    IF(CRT(NG2+I1,2).NE.0.) GOTO 231
0088900      Y=Y*(X**2-CRT(NG2+I1,1)**2)
0089000      I1=I1+2
0089100      GOTO 214
0089200 231    Y=Y*((X-CRT(NG2+I1,1))**2+CRT(NG2+I1,2)**2)
0089300      Y=Y*((X+CRT(NG2+I1,1))**2+CRT(NG2+I1,2)**2)
0089400      I1=I1+4
0089500 214    IF(I1.GT.NRTNCT) GOTO 232
0089600      GOTO 215
0089700 232    F=F/Y
0089800      IF(NSYM.EQ.1) F=F/X**5
0089900      IF(NSYM.EQ.2) F=F/X**3
0090000      IF(NSYM.EQ.3) F=F/X**6
0090100      RETURN
0090200 3      CVC30(NPIVOT)=(1.D 00,0.D 00)
0090300      A(NPIVOT,NPIVOT)=(1.D 00,0.D 00)
0090400      DO 4 I=1,N
0090500      IF(I.NE.NPIVOT) CVC30(I)=-A(I,NPIVOT)
0090600      IF(I.NE.NPIVOT) A(NPIVOT,I)=(0.D 00,0.D 00)
0090700 4      IF(I.NE.NPIVOT) A(I,NPIVOT)=(0.D 00,0.D 00)
0090800      CALL LEQT1C(A,N,N25L,CVC30,1,N25L,0,WA,IER)
0090900      F=(1.D 00,0.D 00)
0091000      DO 71 I=1,N
0091100      IPVT=WA(I)
0091200      IF(IPVT.NE.I) F=-F
0091300 71     F=F*A(I,I)
0091400      RETURN
0091500      END
0091600      SUBROUTINE FDET(NSYM,COEF,X,F,NSDFG,VC30,NPIVOT,NK10,NCN1,N,NRT,
0091700      *NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,N25L)
0091800      IMPLICIT REAL (A-H,O-Z)
0091900      DIMENSION CRT(NCN1,1),SK0(N25L,1),SK1(N25L,1),SK2(N25L,1),
0092000      *SK3(N25L,1),SK4(N25L,1),A(N25L,1),COEF(1),VC30(1)
0092100      IF(NSDFG.NE.0) GOTO 8
0092200      IF(NK10.EQ.0) GOTO 8
0092300      F=0.D 00
0092400      DO 80 I=1,NT
0092500 80     F=F+COEF(I)*X**(2*I-2)
0092600      RETURN
0092700 8      DO 2 I=1,N
0092800      DO 2 J=1,N

```



```

0092900 2      A(I,J)=SK0(I,J)+SK1(I,J)*X+SK2(I,J)*X*X+SK3(I,J)*X*X*X
0093000      *+SK4(I,J)*X*X*X*X
0093100      IF(NSDFG.NE.0) GOTO 3
0093200      CALL SKINV(A,N25L,A,N25L,VC30,1,-1,F,N)
0093300      IF(NPIVOT.EQ.11000) RETURN
0093400      Y=1.D 00
0093500      NRTNCT=NRT+NCT
0093600      IF(NRTNCT.LT.2) GOTO 232
0093700      I1=1
0093800 215    IF(CRT(NG2+I1,2).NE.0.) GOTO 231
0093900      Y=Y*(X**2-CRT(NG2+I1,1)**2)
0094000      I1=I1+2
0094100      GOTO 214
0094200 231    Y=Y*((X-CRT(NG2+I1,1))**2+CRT(NG2+I1,2)**2)
0094300      Y=Y*((X+CRT(NG2+I1,1))**2+CRT(NG2+I1,2)**2)
0094400      I1=I1+4
0094500 214    IF(I1.GT.NRTNCT) GOTO 232
0094600      GOTO 215
0094700 232    F=F/Y
0094800      IF(NSYM.EQ.1) F=F/X**5
0094900      IF(NSYM.EQ.2) F=F/X**3
0095000      IF(NSYM.EQ.3) F=F/X**6
0095100      RETURN
0095200 3      VC30(NPIVOT)=1.D 00
0095300      A(NPIVOT,NPIVOT)=1.D 00
0095400      DO 4 I=1,N
0095500      IF(I.NE.NPIVOT) VC30(I)=-A(I,NPIVOT)
0095600      IF(I.NE.NPIVOT) A(NPIVOT,I)=0.D 00
0095700 4      IF(I.NE.NPIVOT) A(I,NPIVOT)=0.D 00
0095800      CALL SKINV(A,N25L,A,N25L,VC30,1,0,F,N)
0095900      RETURN
0096000      END
0096100      SUBROUTINE COEFF1(LJ,COEF,N,NCT,NRT,NRGN,NT,NG2,SK,SK0,
0096200      *SK1,SK2,SK3,SK4,CRT,NK10,NCN1,A,N25L)
0096300      IMPLICIT REAL (A-H,O-Z)
0096400      DIMENSION SK(N25L,1),COEF(1),SK0(N25L,1),SK1(N25L,1),SK2(N25L,1),
0096500      *SK3(N25L,1),SK4(N25L,1),CRT(NCN1,1),A(N25L,1)
0096600      DO 6 I=1,NT
0096700      Y=DFLOAT(I)
0096800      X=1.D 00
0096900      IF(LJ.EQ.2) X=X*Y*Y*Y
0097000      IF(LJ.EQ.1) X=X*Y*Y*Y*Y*Y
0097100      IF(LJ.EQ.3) X=X*Y*Y*Y*Y*Y*Y
0097200      CALL FDET(LJ,COEF,Y,FI,0,COEF,11000,NK10,NCN1,
0097300      *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,N25L)
0097400      NRTNCT=NRT+NCT
0097500      IF(NRTNCT.LT.2) GOTO 232
0097600      I1=1
0097700 215    IF(CRT(NG2+I1,2).NE.0.) GOTO 231
0097800      X=X*(Y**2-CRT(NG2+I1,1)**2)
0097900      I1=I1+2
0098000      GOTO 214
0098100 231    X=X*((Y-CRT(NG2+I1,1))**2+CRT(NG2+I1,2)**2)
0098200      X=X*((Y+CRT(NG2+I1,1))**2+CRT(NG2+I1,2)**2)

```

```

0098300      I1=I1+4
0098400 214  IF(I1.GT.NRTNCT) GOTO 232
0098500      GOTO 215
0098600 232  COEF(I)=FI/X
0098700      DO 6 J=1,NT
0098800 6    SK(I,J)= Y**(2*J-2)
0098900      CALL SKINV(SK,N25L,SK,N25L,COEF,1,0,SIMUB,NT)
0099000      RETURN
0099100      END
0099200      SUBROUTINE CROOT(NSYM,N,NT,NG2,NRGN,NK10,NRT,NCT,IN7,IN8, -
0099300      *CRT,CMODE,SK0,SK1,SK2,SK3,SK4,COEF,CVC31,CVC32,A,WA,NCN1,NCN2,N25L-
0099400      *)
0099500      IMPLICIT REAL (A-H,O-Z)
0099600      DIMENSION CRT(NCN1,1),CMODE(NCN1,NCN2,1),SK0(N25L,1),SK1(N25L,1), -
0099700      *SK2(N25L,1),SK3(N25L,1),SK4(N25L,1),COEF(1),WA(1)
0099800      COMPLEX   XF12,FF1,FF2,XI,XF,FI,CVC31(1),CVC32(1),XRL,XIM, -
0099900      *A(N25L,1), DCMLPX
0100000 C.....LIMIT MAX. NO. OF ITERATIONS TO 100, AND ROOTS TO 100
0100100      XRL=(1.,0.)
0100200      XIM=(0.,-1.)
0100300      DO 4 I=1,100
0100400 4011 WRITE(IN8,5)
0100500 5    FORMAT(1X,'2F10.6....XI..INPUT INITIAL COMPLEX ROOT VALUE')
0100600      READ(IN7,6) XI
0100700      IF(CABS(XI).LE.1.D-20) GOTO 14
0100800 6    FORMAT(10F10.6)
0100900      NITER=0
0101000 9    CALL CDET(NSYM,COEF,XI,FI,0,CVC31,1,NK10, -
0101100      *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,WA,NCN1,N25L)
0101200      RFI=FI*XRL
0101300      RGI=FI*XIM
0101400      UI=XI*XRL
0101500      VI=XI*XIM
0101600      XI=CMPLX( 1.00000000001*UI,VI)
0101700      CALL CDET(NSYM,COEF,XI,FI,0,CVC31,1,NK10, -
0101800      *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,WA,NCN1,N25L)
0101900      RFU=FI*XRL
0102000      RGU=FI*XIM
0102100      RFU=(RFU-RFI)/.00000000001/UI
0102200      RGU=(RGU-RGI)/.00000000001/UI
0102300      XI=CMPLX(UI,1.00000000001*VI)
0102400      CALL CDET(NSYM,COEF,XI,FI,0,CVC31,1,NK10, -
0102500      *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,WA,NCN1,N25L)
0102600      RFV=FI*XRL
0102700      RGV=FI*XIM
0102800      RFV=(RFV-RFI)/.00000000001/VI
0102900      RGV=(RGV-RGI)/.00000000001/VI
0103000      UF=UI-(RFI*RGV-RFV*RGI)/(RFU*RGV-RFV*RGU)
0103100      VF=VI-(RFU*RGI-RFI*RGU)/(RFU*RGV-RFV*RGU)
0103200      NITER=NITER+1
0103300      WRITE(IN8,12)NITER,XI,FI
0103400      IF(CABS(XI/CMPLX(UF,VF))-1.).LE..000000001) GOTO 7
0103500      IF(NITER.GT.100) GOTO 4011
0103600      XI=CMPLX(UF,VF)

```

```

0103700      GOTO 9
0103800 7     WRITE(IN8,12)I,XI,FI
0103900      WRITE(IN8,76)
0104000      READ(IN7,17) J
0104100 17    FORMAT(20I2)
0104200      IF(J.NE.0) GOTO 77
0104300      GOTO 4011
0104400 12    FORMAT(1X,I3,1X,4D15.8,' I,XI,FI')
0104500 77    DO 7712 IIII=1,4
0104600      CRT(NG2+NRT+NCT+IIII,1)=UF
0104700 7712 CRT(NG2+NRT+NCT+IIII,2)=VF
0104800      NCT=NCT+4
0104900 76    FORMAT(1X,'I2...INPUT...J=1 TO STORE ROOT')
0105000 4     CONTINUE
0105100 14    WRITE(IN8,21)
0105200 21    FORMAT(1X,'I2...INPUT J=0 TO FIND COMPLEX EIGEN MODES')
0105300      READ(IN7,17) J
0105400      IF(J.NE.0) GOTO 86
0105500      III=0
0105600      DO 82 I=1,NCT,4
0105700      III=III+1
0105800      X1=CRT(NG2+I,1)
0105900      X2=CRT(NG2+I,2)
0106000      DO 821 J1=1,4,2
0106100      IF(J1.NE.1) GOTO 832
0106200      XI=CMPLX( X1, X2)
0106300      CALL CDET(NSYH,COEF,XI,FI,1,CVC31,1,NK10,
0106400      *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,WA,NCN1,N25L)
0106500 861   FORMAT(/3(1X,I4),1X,2D15.8,1X,'COMPLEX EIGEN VALUE AND MODE')
0106600 832   IF(J1.NE.3) GOTO 821
0106700      XI=CMPLX(-X1, X2)
0106800      CALL CDET(NSYH,COEF,XI,FI,1,CVC32,1,NK10,
0106900      *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,WA,NCN1,N25L)
0107000 87   FORMAT(10(1X,D10.4))
0107100 821   CONTINUE
0107200      NCT12=NCT/2
0107300      IF(NRGN.EQ.0) WRITE(IN8,861) NG2,NCT,I,XI
0107400      IF(NRGN.EQ.1) WRITE(IN8,861) NG2,NCT12,III,XI
0107500      IF(NRGN.EQ.1) CRT(NG2+2*III-1,1)=X1
0107600      IF(NRGN.EQ.1) CRT(NG2+2*III-1,2)=X2
0107700      IF(NRGN.EQ.1) CRT(NG2+2*III-0,1)=X1
0107800      IF(NRGN.EQ.1) CRT(NG2+2*III-0,2)=X2
0107900      DO 7634 J=1,N
0108000 7634 WRITE(IN8,87) CVC31(J),CVC32(J)
0108100      DO 823 J=1,N
0108200      RCVC31=CVC31(J)*XRL
0108300      ACVC31=CVC31(J)*XIM
0108400      RCVC32=CVC32(J)*XRL
0108500      ACVC32=CVC32(J)*XIM
0108600      IF(NRGN.EQ.1) GOTO 76341
0108700      CMODE(NG2+I+0,J,1)=( ACVC31+ACVC32)/2.
0108800      CMODE(NG2+I+0,J,2)=( RCVC31-RCVC32)/2.
0108900      CMODE(NG2+I+0,J,3)=( ACVC31-ACVC32)/2.
0109000      CMODE(NG2+I+0,J,4)=( RCVC31+RCVC32)/2.

```

```

0109100      CMODE(NG2+I+1,J,1)=( ACVC31-ACVC32)/2.
0109200      CMODE(NG2+I+1,J,2)=( RCVC31+RCVC32)/2.
0109300      CMODE(NG2+I+1,J,3)=( ACVC31+ACVC32)/2.
0109400      CMODE(NG2+I+1,J,4)=( RCVC31-RCVC32)/2.
0109500      CMODE(NG2+I+2,J,1)=( RCVC31+RCVC32)/2.
0109600      CMODE(NG2+I+2,J,2)=(-ACVC31+ACVC32)/2.
0109700      CMODE(NG2+I+2,J,3)=( RCVC31-RCVC32)/2.
0109800      CMODE(NG2+I+2,J,4)=(-ACVC31-ACVC32)/2.
0109900      CMODE(NG2+I+3,J,1)=( RCVC31-RCVC32)/2.
0110000      CMODE(NG2+I+3,J,2)=(-ACVC31-ACVC32)/2.
0110100      CMODE(NG2+I+3,J,3)=( RCVC31+RCVC32)/2.
0110200      CMODE(NG2+I+3,J,4)=(-ACVC31+ACVC32)/2.
0110300      GOTO 823
0110400 76341 CONTINUE
0110500      CMODE(NG2+2*III-1,J,1)= RCVC31
0110600      CMODE(NG2+2*III-1,J,2)=-ACVC31
0110700      CMODE(NG2+2*III-1,J,3)=0.D 00
0110800      CMODE(NG2+2*III-1,J,4)=0.D 00
0110900      CMODE(NG2+2*III-0,J,1)= ACVC31
0111000      CMODE(NG2+2*III-0,J,2)= RCVC31
0111100      CMODE(NG2+2*III-0,J,3)=0.D 00
0111200      CMODE(NG2+2*III-0,J,4)=0.D 00
0111300 823  CONTINUE
0111400 82  CONTINUE
0111500      IF(NRGN.EQ.1) NCT=NCT/2
0111600 86  RETURN
0111700      END
0111800      SUBROUTINE RROOT(NSYH,N,NT,NG2,NRGN,NK10,NRT,NCT,IN7,IN8,
0111900      *CRT,CMODE,SK0,SK1,SK2,SK3,SK4,COEF,VC31,VC32,A,NCN1,NCN2,N25L)
0112000      IMPLICIT REAL (A-H,O-Z)
0112100      DIMENSION CRT(NCN1,1),CMODE(NCN1,NCN2,1),SK0(N25L,1),SK1(N25L,1),
0112200      *SK2(N25L,1),SK3(N25L,1),SK4(N25L,1),COEF(1),VC31(1),VC32(1)
0112300      *,A(N25L,1)
0112400 C6390  FORMAT(1X,I2,14F9.4)
0112500 C      NQQ=N
0112600 C      DO 6391 I=1,NQQ
0112700 C6391  WRITE(6,6390) I,(SK0(I,J),J=1,NQQ)
0112800 C      DO 6392 I=1,NQQ
0112900 C6392  WRITE(6,6390) I,(SK1(I,J),J=1,NQQ)
0113000 C      DO 6393 I=1,NQQ
0113100 C6393  WRITE(6,6390) I,(SK2(I,J),J=1,NQQ)
0113200 C      DO 6394 I=1,NQQ
0113300 C6394  WRITE(6,6390) I,(SK3(I,J),J=1,NQQ)
0113400 C      DO 6395 I=1,NQQ
0113500 C6395  WRITE(6,6390) I,(SK4(I,J),J=1,NQQ)
0113600 C.....LIMIT MAX. NO. OF ROOTS TO 100, AND ITERATIONS TO 100
0113700      DO 4 I=1,100
0113800 4011  WRITE(IN8,5)
0113900 5      FORMAT(1X,'10F10.6....XI..INPUT INITIAL ROOT VALUE')
0114000      READ(IN7,6) XI
0114100      IF(DABS(XI).LE.1.D-20) GOTO 14
0114200 6      FORMAT(10F10.6)
0114300      NITER=0
0114400 9      CALL FDET(NSYH,COEF,XI,FI,0,VC31,1,NK10,NCN1,

```

```

0114500 *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,N25L)
0114600 XF1=(1.0+0.0000000001)*XI
0114700 CALL FDET(NSYM,COEF,XF1,FF1,0,VC31,1,NK10,NCN1,
0114800 *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,N25L)
0114900 XF2=(1.0-0.0000000001)*XI
0115000 CALL FDET(NSYM,COEF,XF2,FF2,0,VC31,1,NK10,NCN1,
0115100 *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,N25L)
0115200 DERXI=(FF1-FF2)/(XF1-XF2)
0115300 XF=XI-FI/DERXI
0115400 NITER=NITER+1
0115500 WRITE(IN8,12)NITER,XI,FI
0115600 IF(DABS(XI/XF-1.)).LE..000000001) GOTO 7
0115700 IF(NITER.GT.100) GOTO 4011
0115800 XI=XF
0115900 GOTO 9
0116000 7 WRITE(IN8,12)I,XI,FI
0116100 12 FORMAT(1X,I3,1X,2(D15.8,1X),' I,XI,FI')
0116200 WRITE(IN8,76)
0116300 READ(IN7,17) J
0116400 17 FORMAT(20I2)
0116500 IF(J.NE.0) GOTO 77
0116600 GOTO 4011
0116700 77 DO 7712 IIII=1,2
0116800 CRT(NG2+NRT+NCT+IIII,2)=0.D 00
0116900 7712 CRT(NG2+NRT+NCT+IIII,1)=XI
0117000 NRT=NRT+2
0117100 76 FORMAT(1X,'I2...INPUT...J=1 TO STORE ROOT')
0117200 4 CONTINUE
0117300 14 WRITE(IN8,21)
0117400 21 FORMAT(1X,'I2...INPUT J=0 TO FIND REAL EIGEN MODES')
0117500 READ(IN7,17) J
0117600 IF(J.NE.0) GOTO 86
0117700 III=0
0117800 DO 82 I=1,NRT,2
0117900 III=III+1
0118000 XI=-CRT(NG2+I,1)
0118100 CALL FDET(NSYM,COEF,XI,FI,1,VC32,1,NK10,NCN1,
0118200 *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,N25L)
0118300 XI=CRT(NG2+I,1)
0118400 CALL FDET(NSYM,COEF,XI,FI,1,VC31,1,NK10,NCN1,
0118500 *N,NRT,NCT,CRT,NRGN,NT,NG2,SK0,SK1,SK2,SK3,SK4,A,N25L)
0118600 DO 83 I1=1,N
0118700 IF(NRGN.EQ.1) GOTO 24850
0118800 CMODE(NG2+I ,I1,1)=(VC31(I1)+VC32(I1))/2
0118900 CMODE(NG2+I ,I1,2)=(VC31(I1)-VC32(I1))/2
0119000 CMODE(NG2+I+1,I1,1)=(VC31(I1)-VC32(I1))/2
0119100 CMODE(NG2+I+1,I1,2)=(VC31(I1)+VC32(I1))/2
0119200 GOTO 83
0119300 24850 CMODE(NG2+III,I1,1)=VC31(I1)
0119400 CMODE(NG2+III,I1,2)=0.D 00
0119500 83 CONTINUE
0119600 IF(NRGN.NE.1) GOTO 8612
0119700 CRT(NG2+III,1)=XI
0119800 NRT12=NRT/2

```

```

0119900      WRITE(IN8,861) NG2,NRT12,III,CRT(NG2+III,1)
0120000      WRITE(IN8,87) (CMODE(NG2+III,J,1),J=1,N)
0120100      WRITE(IN8,87) (CMODE(NG2+III,J,2),J=1,N)
0120200 8612  IF(NRGN.NE.0) GOTO 82
0120300 861  FORMAT(/3(1X,I4),1X,D15.8,1X,'REAL  EIGEN VALUE AND MODE')
0120400      WRITE(IN8,861) NG2,NRT, I,CRT(NG2+I,1)
0120500      WRITE(IN8,87) (CMODE(NG2+I ,J,1),J=1,N)
0120600      WRITE(IN8,87) (CMODE(NG2+I ,J,2),J=1,N)
0120700      WRITE(IN8,87) (CMODE(NG2+I+1,J,1),J=1,N)
0120800      WRITE(IN8,87) (CMODE(NG2+I+1,J,2),J=1,N)
0120900 87   FORMAT(10(1X,D10.4))
0121000 82   CONTINUE
0121100      IF(NRGN.EQ.1) NRT=NRT/2
0121200 86   RETURN
0121300      END
0121400      FUNCTION HPX(N,I,P,A,B,YN,Y,MT,NRGN)
0121500      IMPLICIT REAL      (A-H,O-Z)
0121600      DIMENSION P(6,4)
0121700      CH0(X)=DCOSH(X)*(1-NRGN)+DEXP(A*X)*NRGN
0121800      SH0(X)=DSINH(X)*(1-NRGN)+DEXP(A*X)*NRGN
0121900      CH1(X)=SH0(X)*A
0122000      SH1(X)=CH0(X)*A
0122100      CH2(X)=CH0(X)*A*A
0122200      SH2(X)=SH0(X)*A*A
0122300      CH3(X)=SH0(X)*A*A*A
0122400      SH3(X)=CH0(X)*A*A*A
0122500      CH4(X)=CH0(X)*A*A*A*A
0122600      SH4(X)=SH0(X)*A*A*A*A
0122700      CH5(X)=SH0(X)*A*A*A*A*A
0122800      SH5(X)=CH0(X)*A*A*A*A*A
0122900      CH6(X)=CH0(X)*A*A*A*A*A*A
0123000      SH6(X)=SH0(X)*A*A*A*A*A*A
0123100      SHC0(X)=SH0(X)*DCOS(B*X)
0123200      SHS0(X)=SH0(X)*DSIN(B*X)
0123300      CHS0(X)=CH0(X)*DSIN(B*X)
0123400      CHC0(X)=CH0(X)*DCOS(B*X)
0123500      SHC1(X)=A*CHC0(X)-B*SHS0(X)
0123600      SHS1(X)=A*CHS0(X)+B*SHC0(X)
0123700      CHS1(X)=A*SHS0(X)+B*CHC0(X)
0123800      CHC1(X)=A*SHC0(X)-B*CHS0(X)
0123900      SHC2(X)=(A*A-B*B)*SHC0(X)-2*A*B*CHS0(X)
0124000      SHS2(X)=(A*A-B*B)*SHS0(X)+2*A*B*CHC0(X)
0124100      CHS2(X)=(A*A-B*B)*CHS0(X)+2*A*B*SHC0(X)
0124200      CHC2(X)=(A*A-B*B)*CHC0(X)-2*A*B*SHS0(X)
0124300      SHC3(X)=(A**3-3*A*B*B)*CHC0(X)-(3*A*A*B-B**3)*SHS0(X)
0124400      SHS3(X)=(A**3-3*A*B*B)*CHS0(X)+(3*A*A*B-B**3)*SHC0(X)
0124500      CHS3(X)=(A**3-3*A*B*B)*SHS0(X)+(3*A*A*B-B**3)*CHC0(X)
0124600      CHC3(X)=(A**3-3*A*B*B)*SHC0(X)-(3*A*A*B-B**3)*CHS0(X)
0124700      SHC4(X)=(A**4-6*A*A*B*B+B**4)*SHC0(X)-(4*A**3*B-4*A*B**3)*CHS0(X)
0124800      SHS4(X)=(A**4-6*A*A*B*B+B**4)*SHS0(X)+(4*A**3*B-4*A*B**3)*CHC0(X)
0124900      CHS4(X)=(A**4-6*A*A*B*B+B**4)*CHS0(X)+(4*A**3*B-4*A*B**3)*SHC0(X)
0125000      CHC4(X)=(A**4-6*A*A*B*B+B**4)*CHC0(X)-(4*A**3*B-4*A*B**3)*SHS0(X)
0125100      SHC5(X)=(A**4-6*A*A*B*B+B**4)*SHC1(X)-(4*A**3*B-4*A*B**3)*CHS1(X)
0125200      SHS5(X)=(A**4-6*A*A*B*B+B**4)*SHS1(X)+(4*A**3*B-4*A*B**3)*CHC1(X)

```

```

0125300      CHS5(X)=(A**4-6*A*A*B*B+B**4)*CHS1(X)+(4*A**3*B-4*A*B**3)*SHC1(X)
0125400      CHC5(X)=(A**4-6*A*A*B*B+B**4)*CHC1(X)-(4*A**3*B-4*A*B**3)*SHS1(X)
0125500      SHC6(X)=(A**4-6*A*A*B*B+B**4)*SHC2(X)-(4*A**3*B-4*A*B**3)*CHS2(X)
0125600      SHS6(X)=(A**4-6*A*A*B*B+B**4)*SHS2(X)+(4*A**3*B-4*A*B**3)*CHC2(X)
0125700      CHS6(X)=(A**4-6*A*A*B*B+B**4)*CHS2(X)+(4*A**3*B-4*A*B**3)*SHC2(X)
0125800      CHC6(X)=(A**4-6*A*A*B*B+B**4)*CHC2(X)-(4*A**3*B-4*A*B**3)*SHS2(X)
0125900      N1=N+1
0126000      GOTO(1,2,3,4),N1
0126100 1     IF(MT.EQ.1)HPX=(P(I,1)+P(I,2)*Y+P(I,3)*Y*Y+P(I,4)*Y*Y*Y)
0126200      IF(MT.EQ.2)HPX=(P(I,1)*CH0(Y)+P(I,2)*SH0(Y))
0126300      IF(MT.EQ.3)HPX=(P(I,1)*CHC0(Y)+P(I,2)*CHS0(Y)+P(I,3)*SHC0(Y)+P(I,4-
0126400      *)*SHS0(Y))
0126500      RETURN
0126600 2     IF(MT.EQ.1)HPX=(P(I,2)*1 +P(I,3)*2*Y+P(I,4)*3*Y*Y)*YN
0126700      IF(MT.EQ.2)HPX=(P(I,1)*CH1(Y)+P(I,2)*SH1(Y))*YN
0126800      IF(MT.EQ.3)HPX=(P(I,1)*CHC1(Y)+P(I,2)*CHS1(Y)+P(I,3)*SHC1(Y)+P(I,4-
0126900      *)*SHS1(Y))*YN
0127000      RETURN
0127100 3     IF(MT.EQ.1)HPX=(      P(I,3)*2 +P(I,4)*3*2*Y)*YN**2
0127200      IF(MT.EQ.2)HPX=(P(I,1)*CH2(Y)+P(I,2)*SH2(Y))*YN**2
0127300      IF(MT.EQ.3)HPX=(P(I,1)*CHC2(Y)+P(I,2)*CHS2(Y)+P(I,3)*SHC2(Y)+P(I,4-
0127400      *)*SHS2(Y))*YN**2
0127500      RETURN
0127600 4     IF(MT.EQ.1)HPX=(      +P(I,4)*3*2*1)*YN**3
0127700      IF(MT.EQ.2)HPX=(P(I,1)*CH3(Y)+P(I,2)*SH3(Y))*YN**3
0127800      IF(MT.EQ.3)HPX=(P(I,1)*CHC3(Y)+P(I,2)*CHS3(Y)+P(I,3)*SHC3(Y)+P(I,4-
0127900      *)*SHS3(Y))*YN**3
0128000      RETURN
0128100      END
0128200      SUBROUTINE FUN(NL,N,I1,N2,N4,NRGN,NV,MT,NMT,H,Y1,Z1,F,FPS,      -
0128300      *CB11,CB12,CB22,SNC2,SH12,SHQ2,KQ2,KM2,KP2,KN2,EPSLON,CURVTR,THICK,-
0128400      *NMTYPE,SLN,CRT,CHODE,RCH,NCN1,NCN2,IPR)
0128500      IMPLICIT REAL (A-H,0-Z)
0128600      REAL RCH
0128700      REAL KM1,KM2,KN1,KN2,KQ1,KQ2,KP1,KP2,NU12,NU13,NU23
0128800      DIMENSION CB11(1),CB12(1),CB22(1),SNC2(1),SH12(1),SHQ2(1),KQ2(1), -
0128900      *KM2(1),KP2(1),KN2(1),THICK(1),NMTYPE(1), -
0129000      *SLN(1),CRT(NCN1,1),CHODE(NCN1,NCN2,1),RCH(1),P(6,4)
0129100      DATA Z01,Z02,Z03,Z04,Z05,Z06/0.,0.,0.,0.,0.,0./
0129200 C.....NV=1  N2 (X1,X2)      NV=2  U2(X1,X2,Z)
0129300 C.....NV=3  M2 (X1,X2)      NV=4  PHI(X1,X2,0)
0129400 C.....NV=5  Q2 (X1,X2)      NV=6  W (X1,X2,Z)
0129500 C.....NV=7  SYX(X1,X2,Z)    NV=8  SYZ(X1,X2,Z)      NV=9  SZZ(X1,X2,Z)
0129600 C.....NV=10  WB(X1,X2) THICKNESS STRETCH PARAMETER
0129700 C.....NV=11  SYZ(X1,X2,-1),X2 SHEAR STRESS DERIVATIVE AT PLY BOTTOM
0129800      N1=NMTYPE(I1)
0129900      CALL PZERO(P,24)
0130000      CN=THICK(I1)/2.
0130100      YN=CN/(H/2.)
0130200      Y=Y1/(H/2.)
0130300      Z=Z1/CN
0130400      A=CRT(N2,1)
0130500      B=CRT(N2,2)
0130600      IF(4*N-5.LT.3.OR.4*N-5.GT.4*NL-2) GOTO 8101

```

```

0130700      Z05=SLN(N4+4*N-5)
0130800      Z06=SLN(N4+4*N-4)
0130900 8101  Z01=SLN(N4+4*N-3)
0131000      Z02=SLN(N4+4*N-2)
0131100      IF(4*N-1.LT.3.OR.4*N-1.GT.4*NL-2) GOTO 8201
0131200      Z03=SLN(N4+4*N-1)
0131300      Z04=SLN(N4+4*N-0)
0131400 8201  IF(4*N-5.LT.3.OR.4*N-5.GT.4*NL-2) GOTO 9905
0131500      DO 8945 I=1,4
0131600      P(5,I)=CMODE(N2,4*N-5,I)
0131700 8945  P(6,I)=CMODE(N2,4*N-4,I)
0131800 9905  DO 8946 I=1,4
0131900      P(1,I)=CMODE(N2,4*N-3,I)
0132000 8946  P(2,I)=CMODE(N2,4*N-2,I)
0132100      IF(4*N-1.LT.3.OR.4*N-1.GT.4*NL-2) GOTO 9904
0132200      DO 8947 I=1,4
0132300      P(3,I)=CMODE(N2,4*N-1,I)
0132400 8947  P(4,I)=CMODE(N2,4*N-0,I)
0132500 9904  GOTO(1,2,3,5,5,6,7,5,9,6,11),NV
0132600 1      FPS=2*CN*(KP2(N1)*(Z06+Z04)+CB12(N1)*EPSLON)
0132700      H11=HPX(1,1,P,A,B,YN,Y,MT,NRGN)
0132800      H15=HPX(1,5,P,A,B,YN,Y,MT,NRGN)
0132900      H13=HPX(1,3,P,A,B,YN,Y,MT,NRGN)
0133000      H06=HPX(0,6,P,A,B,YN,Y,MT,NRGN)
0133100      H04=HPX(0,4,P,A,B,YN,Y,MT,NRGN)
0133200      F=2*CN*(H11*CB22(N1)+(H15-H13)*KN2(N1)+(H06+H04)*KP2(N1))
0133300      RETURN
0133400 3      FPS=(KQ2(N1)/CN*(Z06-Z04)-CB12(N1)*CURVTR)*2./3*CN**3
0133500      H22=HPX(2,2,P,A,B,YN,Y,MT,NRGN)
0133600      H15=HPX(1,5,P,A,B,YN,Y,MT,NRGN)
0133700      H13=HPX(1,3,P,A,B,YN,Y,MT,NRGN)
0133800      H06=HPX(0,6,P,A,B,YN,Y,MT,NRGN)
0133900      H04=HPX(0,4,P,A,B,YN,Y,MT,NRGN)
0134000      F=2./3*CN*CN*(-H22*CB22(N1)+(H15+H13)*KM2(N1)+(H06-H04)*KQ2(N1))
0134100      RETURN
0134200 5      FPS=CN*(Z05+Z03)
0134300      H32=HPX(3,2,P,A,B,YN,Y,MT,NRGN)
0134400      H25=HPX(2,5,P,A,B,YN,Y,MT,NRGN)
0134500      H23=HPX(2,3,P,A,B,YN,Y,MT,NRGN)
0134600      H16=HPX(1,6,P,A,B,YN,Y,MT,NRGN)
0134700      H14=HPX(1,4,P,A,B,YN,Y,MT,NRGN)
0134800      H05=HPX(0,5,P,A,B,YN,Y,MT,NRGN)
0134900      H03=HPX(0,3,P,A,B,YN,Y,MT,NRGN)
0135000      F=2./3*CN*(-H32*CB22(N1)+(H25+H23)*KM2(N1)+(H16-H14)*KQ2(N1))
0135100      **+(H05+H03)*CN
0135200      IF(NV.EQ.4) GOTO 4
0135300      IF(NV.EQ.8) GOTO 8
0135400      RETURN
0135500 4      FPS=(SHI2(N1)+SHQ2(N1))*(Z05+Z03)
0135600      H12=HPX(1,2,P,A,B,YN,Y,MT,NRGN)
0135700      F=SHQ2(N1)/CN*F-H12+SHI2(N1)*(H05+H03)
0135800      RETURN
0135900 6      ZZZZ=.000000001D 00
0136000      IF(NV.EQ.10) Z=ZZZZ

```



```

0136100 61   FPS=WVD(1,N1,Z,NMT,RCH,IPR)*CN*Z02+WVD(3,N1,Z*CN,NMT,RCH,IPR) -
0136200      *+(WVD(7,N1,Z,NMT,RCH,IPR)+WVD(8,N1,Z,NMT,RCH,IPR))*Z06 -
0136300      * +(WVD(7,N1,Z,NMT,RCH,IPR)-WVD(8,N1,Z,NMT,RCH,IPR))*Z04)*CN
0136400      H02=HPX(0,2,P,A,B,YN,Y,MT,NRGN)
0136500      H22=HPX(2,2,P,A,B,YN,Y,MT,NRGN)
0136600      H11=HPX(1,1,P,A,B,YN,Y,MT,NRGN)
0136700      H15=HPX(1,5,P,A,B,YN,Y,MT,NRGN)
0136800      H13=HPX(1,3,P,A,B,YN,Y,MT,NRGN)
0136900      H06=HPX(0,6,P,A,B,YN,Y,MT,NRGN)
0137000      H04=HPX(0,4,P,A,B,YN,Y,MT,NRGN)
0137100      F=CN*(H02*WVD(1,N1,Z,NMT,RCH,IPR)+H22*WVD(2,N1,Z,NMT,RCH,IPR)+ -
0137200      *   H11*WVD(4,N1,Z,NMT,RCH,IPR)+ -
0137300      *   H15*(WVD(5,N1,Z,NMT,RCH,IPR)+WVD(6,N1,Z,NMT,RCH,IPR))+ -
0137400      *   H13*(-WVD(5,N1,Z,NMT,RCH,IPR)+WVD(6,N1,Z,NMT,RCH,IPR))+ -
0137500      *   H06*(WVD(7,N1,Z,NMT,RCH,IPR)+WVD(8,N1,Z,NMT,RCH,IPR))+ -
0137600      *   H04*(WVD(7,N1,Z,NMT,RCH,IPR)-WVD(8,N1,Z,NMT,RCH,IPR)))
0137700      IF(NV.EQ.6) RETURN
0137800      IF(Z.NE.ZZZZ) GOTO 62
0137900      FPS1=FPS
0138000      F1=F
0138100      Z=-ZZZZ
0138200      GOTO 61
0138300 62   FPS=(FPS1-FPS)/2/CN/ZZZZ
0138400      F=(F1-F)/2/CN/ZZZZ
0138500      RETURN
0138600 11   FPS=0
0138700      H13=HPX(1,3,P,A,B,YN,Y,MT,NRGN)
0138800      F=H13/CN
0138900      RETURN
0139000 2   FPS=Z01*CN*WVD(11,N1,Z,NMT,RCH,IPR)+Z05*CN*(WVD(13,N1,Z,NMT, -
0139100      *RCH,IPR)+WVD(15,N1,Z,NMT,RCH,IPR))+ -
0139200      *Z03*CN*(-WVD(13,N1,Z,NMT,RCH,IPR)+WVD(15,N1,Z,NMT,RCH,IPR))
0139300      H12=HPX(1,2,P,A,B,YN,Y,MT,NRGN)
0139400      H32=HPX(3,2,P,A,B,YN,Y,MT,NRGN)
0139500      H01=HPX(0,1,P,A,B,YN,Y,MT,NRGN)
0139600      H21=HPX(2,1,P,A,B,YN,Y,MT,NRGN)
0139700      H05=HPX(0,5,P,A,B,YN,Y,MT,NRGN)
0139800      H25=HPX(2,5,P,A,B,YN,Y,MT,NRGN)
0139900      H03=HPX(0,3,P,A,B,YN,Y,MT,NRGN)
0140000      H23=HPX(2,3,P,A,B,YN,Y,MT,NRGN)
0140100      H16=HPX(1,6,P,A,B,YN,Y,MT,NRGN)
0140200      H14=HPX(1,4,P,A,B,YN,Y,MT,NRGN)
0140300      F=(H12*WVD(9,N1,Z,NMT,RCH,IPR)+H32*WVD(10,N1,Z,NMT,RCH,IPR)+ -
0140400      *   H01*WVD(11,N1,Z,NMT,RCH,IPR)+H21*WVD(12,N1,Z,NMT,RCH,IPR)+ -
0140500      *   H05*(WVD(13,N1,Z,NMT,RCH,IPR)+WVD(15,N1,Z,NMT,RCH,IPR))+ -
0140600      *   H03*(-WVD(13,N1,Z,NMT,RCH,IPR)+WVD(15,N1,Z,NMT,RCH,IPR))+ -
0140700      *   H25*(WVD(14,N1,Z,NMT,RCH,IPR)+WVD(16,N1,Z,NMT,RCH,IPR))+ -
0140800      *   H23*(-WVD(14,N1,Z,NMT,RCH,IPR)+WVD(16,N1,Z,NMT,RCH,IPR))+ -
0140900      *   H16*(WVD(17,N1,Z,NMT,RCH,IPR)+WVD(18,N1,Z,NMT,RCH,IPR))+ -
0141000      *   H14*(WVD(17,N1,Z,NMT,RCH,IPR)-WVD(18,N1,Z,NMT,RCH,IPR)))*CN
0141100      RETURN
0141200 8   FPS=Z05*(+Z/2+.5)+Z03*(-Z/2+.5)
0141300      H05=HPX(0,5,P,A,B,YN,Y,MT,NRGN)
0141400      H03=HPX(0,3,P,A,B,YN,Y,MT,NRGN)

```

```

0141500      F=H05*(Z/2+.5)+H03*(-Z/2+.5)+(1-Z**2)/2*F
0141600      RETURN
0141700 9     FPS=(.5+Z/2-(Z**3-Z)/4)*Z06+(.5-Z/2+(Z**3-Z)/4)*Z04
0141800      H15=HPX(1,5,P,A,B,YN,Y,MT,NRGN)
0141900      H13=HPX(1,3,P,A,B,YN,Y,MT,NRGN)
0142000      H06=HPX(0,6,P,A,B,YN,Y,MT,NRGN)
0142100      H04=HPX(0,4,P,A,B,YN,Y,MT,NRGN)
0142200      F=H06*(.5+Z/2-(Z**3-Z)/4)+H04*(.5-Z/2+(Z**3-Z)/4)+
0142300      *H15*(-(Z**2-1)/4-(Z**3-Z)/4)+H13*((Z**2-1)/4-(Z**3-Z)/4)
0142400      RETURN
0142500 7     FPS=Z06*(KP2(N1)+Z*KQ2(N1)+(Z**3-.6*Z)/4*SNC2(N1))+Z04*(KP2(N1)-Z*-
0142600      *KQ2(N1)-(Z**3-.6*Z)/4*SNC2(N1))+CB12(N1)*(EPSLN-CURVTR*CN*Z)
0142700      H15=HPX(1,5,P,A,B,YN,Y,MT,NRGN)
0142800      H13=HPX(1,3,P,A,B,YN,Y,MT,NRGN)
0142900      H06=HPX(0,6,P,A,B,YN,Y,MT,NRGN)
0143000      H04=HPX(0,4,P,A,B,YN,Y,MT,NRGN)
0143100      H11=HPX(1,1,P,A,B,YN,Y,MT,NRGN)
0143200      H22=HPX(2,2,P,A,B,YN,Y,MT,NRGN)
0143300      F= H11      *CB22(N1)-H22      *CB22(N1)*Z
0143400      *+H15      *( KN2(N1)+Z*KM2(N1)+SNC2(N1)*( Z**2-1./3+Z**3-.6*Z)/4)
0143500      *+H13      *(-KN2(N1)+Z*KM2(N1)+SNC2(N1)*(-Z**2+1./3+Z**3-.6*Z)/4)
0143600      *+H06      *( KP2(N1)+Z*KQ2(N1)+.25*SNC2(N1)*(+(Z**3-.6*Z)))
0143700      *+H04      *( KP2(N1)-Z*KQ2(N1)+.25*SNC2(N1)*(-(Z**3-.6*Z)))
0143800      RETURN
0143900      END
0144000      SUBROUTINE DEQMAT(NSYM2,NPLY,NEQ,NBC,NG1,NG2,NR,NMT,NSTREI,H,
0144100      *CB22,KQ2,KM2,KP2,KN2,THICK,      SLN,CRT,CMODE,
0144200      *SK0,SK1,SK2,SK3,SK4,SX,VC45,FSLN,ISGN,RCH,
0144300      *NCN1,NCN2,NSREG,NMTYPE,NG4,N25L,IPR)
0144400      IMPLICIT REAL (A-H,O-Z)
0144500      REAL RCH
0144600      REAL KM1,KM2,KN1,KN2,KQ1,KQ2,KP1,KP2,NU12,NU13,NU23
0144700      DIMENSION CB22(1),KQ2(1),KM2(1),KP2(1),KN2(1),THICK(1),SLN(1),
0144800      *CRT(NCN1,1),CMODE(NCN1,NCN2,1),SX(NSTREI,1),
0144900      *VC45(1),FSLN(1),ISGN(1),NMTYPE(1),SK0(N25L,1),
0145000      *SK1(N25L,1),SK2(N25L,1),SK3(N25L,1),SK4(N25L,1),RCH(1)
0145100 C.....U1,U2,W AND Z, ARE NONDIMENSIONALIZED W.R.T CN
0145200 C.....X1,X2 ARE NONDIMENSIONALIZED W.R.T. LAMINATE SEMITHICKNESS, H/2
0145300 C.....EPSLN=UNIFORM STRAIN IN X1-DIRECTION, U1,1
0145400 C.....CURVTR=NONDIMENSIONALIZED CURVATURE IN X1-DIRECTION, W,XX
0145500      CALL PZERO(SK0,N25L*N25L)
0145600      CALL PZERO(SK1,N25L*N25L)
0145700      CALL PZERO(SK2,N25L*N25L)
0145800      CALL PZERO(SK3,N25L*N25L)
0145900      CALL PZERO(SK4,N25L*N25L)
0146000      CALL PZERO(FSLN,N25L)
0146100 C.....CREATING OF THE DIFFERENTIAL EQUATION OPERATOR MATRIX
0146200 C..... SK4 *LM**4+ SK3 *LM**3+ SK2 *LM**2+ SK1 *LM1+ SK0
0146300      DO 101 N=1,NPLY
0146400      L=NMTYPE(N+NG1)
0146500      YN=THICK(N+NG1)/2/(H/2)
0146600      CN=THICK(N+NG1)/2
0146700      IF(N.LT.NPLY) L1=NMTYPE(N+NG1+1)
0146800      IF(N.LT.NPLY) YN1=THICK(N+NG1+1)/2/(H/2)

```

```

0146900      IF(N.LT.NPLY) CN1=THICK(N+NG1+1)/2
0147000      SK2(4*N-3,4*N-3)=2.*YN*YN*CB22(L)
0147100      IF(4*N-5.GT.2.AND.4*N-5.LT.4*NPLY-4) SK2(4*N-3,4*N-5)=2.*YN*YN*KN2-
0147200      *(L)
0147300      IF(4*N-1.GT.2.AND.4*N-1.LT.4*NPLY-4) SK2(4*N-3,4*N-1)=-2*YN*YN*KN2-
0147400      *(L)
0147500      IF(4*N-5.GT.2.AND.4*N-5.LT.4*NPLY-4) SK0(4*N-3,4*N-5)=1
0147600      IF(4*N-1.GT.2.AND.4*N-1.LT.4*NPLY-4) SK0(4*N-3,4*N-1)=-1.
0147700      IF(4*N-4.GT.3.AND.4*N-4.LT.4*NPLY-3) SK1(4*N-3,4*N-4)=2.*YN*KN2(L)
0147800      IF(4*N .GT.3.AND.4*N .LT.4*NPLY-3) SK1(4*N-3,4*N)=2.*YN*KN2(L)
0147900      SK4(4*N-2,4*N-2)=-2./3*CB22(L)*YN*YN*YN*YN
0148000      IF(4*N-5.GT.2.AND.4*N-5.LT.4*NPLY-4) SK3(4*N-2,4*N-5)=2./3*KN2(L)*-
0148100      *YN*YN*YN
0148200      IF(4*N-1.GT.2.AND.4*N-1.LT.4*NPLY-4) SK3(4*N-2,4*N-1)=2./3*KN2(L)*-
0148300      *YN*YN*YN
0148400      IF(4*N-4.GT.3.AND.4*N-4.LT.4*NPLY-3) SK2(4*N-2,4*N-4)=2./3*KQ2(L)*-
0148500      *YN*YN
0148600      IF(4*N .GT.3.AND.4*N .LT.4*NPLY-3) SK2(4*N-2,4*N )=-2./3*KQ2(L)-
0148700      **YN*YN
0148800      IF(4*N-5.GT.2.AND.4*N-5.LT.4*NPLY-4) SK1(4*N-2,4*N-5)=YN
0148900      IF(4*N-1.GT.2.AND.4*N-1.LT.4*NPLY-4) SK1(4*N-2,4*N-1)=YN
0149000      IF(4*N-4.GT.3.AND.4*N-4.LT.4*NPLY-3) SK0(4*N-2,4*N-4)=1
0149100      IF(4*N .GT.3.AND.4*N .LT.4*NPLY-3) SK0(4*N-2,4*N )=-1
0149200      IF(N.EQ.NPLY) GOTO 101
0149300      SK0(4*N-1,4*N-2)=WVD(1,L,-1.,NMT,RCH,IPR)*CN
0149400      SK2(4*N-1,4*N-2)=WVD(2,L,-1.,NMT,RCH,IPR)*YN*YN*CN
0149500      SK0(4*N-1,4*N+2)=-WVD(1,L1,1.0,NMT,RCH,IPR)*CN1
0149600      SK2(4*N-1,4*N+2)=-WVD(2,L1,1.0,NMT,RCH,IPR)*YN1*YN1*CN1
0149700      SK1(4*N-1,4*N-3)=WVD(4,L,-1.,NMT,RCH,IPR)*YN*CN
0149800      SK1(4*N-1,4*N+1)=-WVD(4,L1,1.0,NMT,RCH,IPR)*YN1*CN1
0149900      IF(4*N-5.GT.2.AND.4*N-5.LT.4*NPLY-4) SK1(4*N-1,4*N-5)=WVD(5,L,-1.,-
0150000      *NMT,RCH,IPR)*YN*CN+ WVD(6,L,-1.,NMT,RCH,IPR)*CN*YN
0150100      IF(4*N-1.GT.2.AND.4*N-1.LT.4*NPLY-4) SK1(4*N-1,4*N-1)= -
0150200      *- WVD(5,L,-1.,NMT,RCH,IPR)*CN*YN- WVD(5,L1,1.,NMT,RCH,IPR)*CN1*YN1-
0150300      **+ WVD(6,L,-1.,NMT,RCH,IPR)*CN*YN- WVD(6,L1,1.,NMT,RCH,IPR)*CN1*YN1
0150400      IF(4*N+3.GT.2.AND.4*N+3.LT.4*NPLY-4) SK1(4*N-1,4*N+3)= WVD(5,L1,1.-
0150500      *0,NMT,RCH,IPR)*CN1*YN1- WVD(6,L1,1.0,NMT,RCH,IPR)*CN1*YN1
0150600      IF(4*N-4.GT.3.AND.4*N-4.LT.4*NPLY-3) SK0(4*N-1,4*N-4)= WVD(7,L,-1.-
0150700      *0,NMT,RCH,IPR)*CN+ WVD(8,L,-1.,NMT,RCH,IPR)*CN
0150800      IF(4*N .GT.3.AND.4*N .LT.4*NPLY-3) SK0(4*N-1,4*N )= WVD(7,L,-1.-
0150900      *0,NMT,RCH,IPR)*CN- WVD(8,L,-1.,NMT,RCH,IPR)*CN- WVD(7,L1,1.0,NMT, -
0151000      *RCH,IPR)*CN1- WVD(8,L1,1.0,NMT,RCH,IPR)*CN1
0151100      IF(4*N+4.GT.3.AND.4*N+4.LT.4*NPLY-3)SK0(4*N-1,4*N+4)=- WVD(7,L1,1.-
0151200      *0,NMT,RCH,IPR)*CN1+ WVD(8,L1,1.0,NMT,RCH,IPR)*CN1
0151300      FSLN(4*N-1)=- WVD(3,L,-CN,NMT,RCH,IPR)+WVD(3,L1,CN1,NMT,RCH,IPR)
0151400      SK1(4*N,4*N-2)= WVD(9,L,-1.,NMT,RCH,IPR)*YN*CN
0151500      SK3(4*N,4*N-2)= WVD(10,L,-1.,NMT,RCH,IPR)*YN*YN*YN*CN
0151600      SK1(4*N,4*N+2)=- WVD(9,L1,1.0,NMT,RCH,IPR)*YN1*CN1
0151700      SK3(4*N,4*N+2)=- WVD(10,L1,1.0,NMT,RCH,IPR)*YN1*YN1*YN1*CN1
0151800      SK0(4*N,4*N-3)= WVD(11,L,-1.,NMT,RCH,IPR)*CN
0151900      SK2(4*N,4*N-3)= WVD(12,L,-1.,NMT,RCH,IPR)*YN*YN*CN
0152000      SK0(4*N,4*N+1)=- WVD(11,L1,1.0,NMT,RCH,IPR)*CN1
0152100      SK2(4*N,4*N+1)=- WVD(12,L1,1.0,NMT,RCH,IPR)*YN1*YN1*CN1
0152200      IF(4*N-5.GT.2.AND.4*N-5.LT.4*NPLY-4) SK0(4*N,4*N-5)= WVD(13,L,-1.,-

```

```

0152300 *NMT,RCH,IPR)*CN+ WVD(15,L,-1.,NMT,RCH,IPR)*CN
0152400 IF(4*N-1.GT.2.AND.4*N-1.LT.4*NPLY-4) SK0(4*N,4*N-1)=- WVD(13,L,-1.-
0152500 *,NMT,RCH,IPR)*CN+ WVD(15,L,-1.,NMT,RCH,IPR)*CN- WVD(13,L1,1.0,NMT,-
0152600 *RCH,IPR)*CN1-WVD(15,L1,1.0,NMT,RCH,IPR)*CN1
0152700 IF(4*N+3.GT.2.AND.4*N+3.LT.4*NPLY-4) SK0(4*N,4*N+3)= WVD(13,L1,1.0-
0152800 *,NMT,RCH,IPR)*CN1- WVD(15,L1,1.0,NMT,RCH,IPR)*CN1
0152900 IF(4*N-5.GT.2.AND.4*N-5.LT.4*NPLY-4) SK2(4*N,4*N-5)=( WVD(14,L,-1.-
0153000 *,NMT,RCH,IPR)+WVD(16,L,-1.,NMT,RCH,IPR))*YN*YN*CN
0153100 IF(4*N-1.GT.2.AND.4*N-1.LT.4*NPLY-4) SK2(4*N,4*N-1)= -
0153200 *(- WVD(14,L,-1.,NMT,RCH,IPR)+ WVD(16,L,-1.,NMT,RCH,IPR))*YN*YN*CN--
0153300 *(WVD(14,L1,1.0,NMT,RCH,IPR)+WVD(16,L1,1.,NMT,RCH,IPR))*YN1*YN1*CN1
0153400 IF(4*N+3.GT.2.AND.4*N+3.LT.4*NPLY-4) SK2(4*N,4*N+3)=( WVD(14,L1,1.-
0153500 *0,NMT,RCH,IPR)-WVD(16,L1,1.0,NMT,RCH,IPR))*CN1*YN1*YN1
0153600 IF(4*N-4.GT.3.AND.4*N-4.LT.4*NPLY-3) SK1(4*N,4*N-4)=( WVD(17,L,-1.-
0153700 *,NMT,RCH,IPR)+WVD(18,L,-1.,NMT,RCH,IPR))*YN*CN
0153800 IF(4*N .GT.3.AND.4*N .LT.4*NPLY-3) SK1(4*N,4*N )= -
0153900 *( WVD(17,L,-1.,NMT,RCH,IPR)- WVD(18,L,-1.,NMT,RCH,IPR))*YN*CN-( -
0154000 *WVD(17,L1,1.0,NMT,RCH,IPR)+WVD(18,L1,1.0,NMT,RCH,IPR))*YN1*CN1
0154100 IF(4*N+4.GT.3.AND.4*N+4.LT.4*NPLY-3) SK1(4*N,4*N+4)=- (WVD(17,L1,1.-
0154200 *0,NMT,RCH,IPR)-WVD(18,L1,1.0,NMT,RCH,IPR))*YN1*CN1
0154300 101 CONTINUE
0154400 C NQQ=8
0154500 C DO 6491 I=1,NQQ
0154600 C6491 WRITE(6,6390) I,(SK0(I,J),J=1,NQQ)
0154700 C DO 6492 I=1,NQQ
0154800 C6492 WRITE(6,6390) I,(SK1(I,J),J=1,NQQ)
0154900 C DO 6493 I=1,NQQ
0155000 C6493 WRITE(6,6390) I,(SK2(I,J),J=1,NQQ)
0155100 C DO 6494 I=1,NQQ
0155200 C6494 WRITE(6,6390) I,(SK3(I,J),J=1,NQQ)
0155300 C DO 6495 I=1,NQQ
0155400 C6495 WRITE(6,6390) I,(SK4(I,J),J=1,NQQ)
0155500 IF(NSYM2.GT.1.AND.NPLY.EQ.1) GOTO 142
0155600 IF(NSYM2.GT.1) GOTO 7104
0155700 SLN(NG4+2)=0.D 00
0155800 DO 5534 I=2,NPLY
0155900 IF(NPLY.LT.2) GOTO 5534
0156000 SLN(NG4+4*I-2)=(FSLN(4*I-5)-SK0(4*I-5,4*I-6))*SLN(NG4+ 4*I-6))/ -
0156100 *SK0(4*I-5,4*I-2)
0156200 5534 CONTINUE
0156300 DO 5536 I=1,NPLY
0156400 5536 SLN(NG4+ 4*I-2)=SLN(NG4+ 4*I-2)/(THICK(NG1+I)/2.)
0156500 IF(NPLY.GT.1) GOTO 7113
0156600 CHODE(NG2+5,1,1)=1
0156700 CHODE(NG2+2,2,1)=1
0156800 CHODE(NG2+1,2,2)=1
0156900 CHODE(NG2+6,1,2)=1
0157000 CHODE(NG2+4,2,3)=1
0157100 CHODE(NG2+3,2,4)=1
0157200 7113 IF(NPLY.LE.1) GOTO 142
0157300 CHODE(NG2+5,1,1)=1
0157400 CHODE(NG2+2,2,1)=1
0157500 CHODE(NG2+1,2,2)=1
0157600 CHODE(NG2+6,1,2)=1

```

```

0157700      CMODE(NG2+4,2,3)=1
0157800      DO 7114 I=2,NPLY
0157900      CMODE(NG2+5,4*I-3,1)=1
0158000      CMODE(NG2+2,4*I-2,1)=1
0158100      CMODE(NG2+1,4*I-2,2)=-1./SK0(4*I-5,4*I-2)*(
0158200      *                               SK0(4*I-5,4*I-6)*CMODE(NG2+1,4*I-6,2))
0158300      CMODE(NG2+1,4*I-3,1)=-1./SK0(4*I-4,4*I-3)*(
0158400      *                               SK0(4*I-4,4*I-7)*CMODE(NG2+1,4*I-7,1)+
0158500      *                               SK1(4*I-4,4*I-6)*CMODE(NG2+1,4*I-6,2)+
0158600      *                               SK1(4*I-4,4*I-2)*CMODE(NG2+1,4*I-2,2))
0158700      CMODE(NG2+6,4*I-3,2)=-1./SK0(4*I-4,4*I-3)*(
0158800      *                               SK0(4*I-4,4*I-7)*CMODE(NG2+6,4*I-7,2))
0158900      CMODE(NG2+6,4*I-2,1)=-1./SK0(4*I-5,4*I-2)*(
0159000      *                               SK1(4*I-5,4*I-7)*CMODE(NG2+6,4*I-7,2)+
0159100      *                               SK0(4*I-5,4*I-6)*CMODE(NG2+6,4*I-6,1)+
0159200      *                               SK1(4*I-5,4*I-3)*CMODE(NG2+6,4*I-3,2))
0159300      CMODE(NG2+4,4*I-2,3)=-1./SK0(4*I-5,4*I-2)*(
0159400      *                               SK0(4*I-5,4*I-6)*CMODE(NG2+4,4*I-6,3))
0159500      CMODE(NG2+4,4*I-3,2)=-1./SK0(4*I-4,4*I-3)*(
0159600      *                               2.*SK1(4*I-4,4*I-6)*CMODE(NG2+4,4*I-6,3)+
0159700      *                               2.*SK1(4*I-4,4*I-2)*CMODE(NG2+4,4*I-2,3)+
0159800      *                               SK0(4*I-4,4*I-7)*CMODE(NG2+4,4*I-7,2))
0159900 7114 CMODE(NG2+4,4*I-2,1)=-1./SK0(4*I-5,4*I-2)*(
0160000      *                               SK0(4*I-5,4*I-6)*CMODE(NG2+4,4*I-6,1)+
0160100      *                               2.*SK2(4*I-5,4*I-6)*CMODE(NG2+4,4*I-6,3)+
0160200      *                               2.*SK2(4*I-5,4*I-2)*CMODE(NG2+4,4*I-2,3)+
0160300      *                               SK1(4*I-5,4*I-7)*CMODE(NG2+4,4*I-7,2)+
0160400      *                               SK1(4*I-5,4*I-3)*CMODE(NG2+4,4*I-3,2))
0160500      CALL PZERO(SX,NSTREI)
0160600      CALL PZERO(VC45,NSTREI)
0160700      SX(1,1)=1
0160800      SX(2,3)=1
0160900      SX(3,4)=1
0161000      VC45(3)=1
0161100      DO 7116 I=1,NPLY
0161200      SX(3+5*I-4,5*I-3)=2.*SK2(4*I-3,4*I-3)
0161300      IF(5*I-5.GT.4.AND.5*I-5.LT.NSTREI+1)
0161400      *SX(3+5*I-4,5*I-5)=SK0(4*I-3,4*I-5)
0161500      IF(5*I .GT.4.AND.5*I .LT.NSTREI+1)
0161600      *SX(3+5*I-4,5*I )=SK0(4*I-3,4*I-1)
0161700      IF(I.EQ.NPLY) GOTO 7116
0161800      SX(3+5*I-3,5*I-1)=SK0(4*I-1,4*I-2)
0161900      SX(3+5*I-3,5*I+4)=SK0(4*I-1,4*I+2)
0162000      SX(3+5*I-2,5*I-1)=6.*SK2(4*I-1,4*I-2)
0162100      SX(3+5*I-2,5*I-2)= SK0(4*I-1,4*I-2)
0162200      SX(3+5*I-2,5*I-3)=2.*SK1(4*I-1,4*I-3)
0162300      SX(3+5*I-2,5*I+4)=6.*SK2(4*I-1,4*I+2)
0162400      SX(3+5*I-2,5*I+3)= SK0(4*I-1,4*I+2)
0162500      SX(3+5*I-2,5*I+2)=2.*SK1(4*I-1,4*I+1)
0162600      SX(3+5*I-1,5*I-1)=3.*SK1(4*I-0,4*I-2)
0162700      SX(3+5*I-1,5*I-3)= SK0(4*I-0,4*I-3)
0162800      SX(3+5*I-1,5*I+4)=3.*SK1(4*I-0,4*I+2)
0162900      SX(3+5*I-1,5*I+2)= SK0(4*I-0,4*I+1)
0163000      SX(3+5*I-0,5*I-1)=6.*SK3(4*I-0,4*I-2)

```

```

0163100      SX(3+5*I-0,5*I-2)=  SK1(4*I-0,4*I-2)
0163200      SX(3+5*I-0,5*I-3)=2.*SK2(4*I-0,4*I-3)
0163300      SX(3+5*I-0,5*I-4)=  SK0(4*I-0,4*I-3)
0163400      SX(3+5*I-0,5*I+4)=6.*SK3(4*I-0,4*I+2)
0163500      SX(3+5*I-0,5*I+3)=  SK1(4*I-0,4*I+2)
0163600      SX(3+5*I-0,5*I+2)=2.*SK2(4*I-0,4*I+1)
0163700      SX(3+5*I-0,5*I+1)=  SK0(4*I-0,4*I+1)
0163800      IF(5*I-5.GT.4.AND.5*I-5.LT.NSTREI+1)
0163900      *SX(3+5*I-0,5*I-5)=  SK0(4*I-0,4*I-5)
0164000      IF(5*I .GT.4.AND.5*I .LT.NSTREI+1)
0164100      *SX(3+5*I-0,5*I )=  SK0(4*I-0,4*I-1)
0164200      IF(5*I+5.GT.4.AND.5*I+5.LT.NSTREI+1)
0164300      *SX(3+5*I-0,5*I+5)=  SK0(4*I-0,4*I+3)
0164400 7116  CONTINUE
0164500      CALL SKINV(SX,NSTREI,SX,NSTREI,VC45,NSTREI,1,SIMUB,NSTREI)
0164600      DO 7117 I=1,NPLY
0164700      CMODE(NG2+3,4*I-3,1)=VC45(5*I-4)
0164800      CMODE(NG2+3,4*I-3,3)=VC45(5*I-3)
0164900      CMODE(NG2+3,4*I-2,2)=VC45(5*I-2)
0165000      CMODE(NG2+3,4*I-2,4)=VC45(5*I-1)
0165100      IF(I.EQ.NPLY) GOTO 7117
0165200      CMODE(NG2+3,4*I-1,1)=VC45(5*I )
0165300 7117  CONTINUE
0165400 7104  IF(NSYM2.EQ.3) GOTO 142
0165500      NEVN=(NPLY+1)/2
0165600      NEQ=4*NEVN-2
0165700      IF(NPLY/2*2.EQ.NPLY) NEQ=4*NEVN
0165800      N2EVN=(4*NPLY-2)/2+2
0165900      N3EVN=(4*NPLY-2)/2+1
0166000      N4EVN=(4*NPLY-2)
0166100      IF(NPLY/2*2.NE.NPLY.AND.NSYM2.EQ.1) I1=N3EVN-1
0166200      IF(NPLY/2*2.NE.NPLY.AND.NSYM2.EQ.1) I2=N3EVN-1
0166300      IF(NPLY/2*2.NE.NPLY.AND.NSYM2.EQ.2) I1=N3EVN
0166400      IF(NPLY/2*2.NE.NPLY.AND.NSYM2.EQ.2) I2=N3EVN
0166500      IF(NPLY/2*2.EQ.NPLY.AND.NSYM2.EQ.1) I1=N3EVN
0166600      IF(NPLY/2*2.EQ.NPLY.AND.NSYM2.EQ.1) I2=N3EVN-1
0166700      IF(NPLY/2*2.EQ.NPLY.AND.NSYM2.EQ.2) I1=N3EVN-1
0166800      IF(NPLY/2*2.EQ.NPLY.AND.NSYM2.EQ.2) I2=N3EVN
0166900      IF(NPLY/2*2.NE.NPLY.AND.NSYM2.EQ.1) NBC = 8*(NEVN-1)+1+4
0167000      IF(NPLY/2*2.EQ.NPLY.AND.NSYM2.EQ.1) NBC = 8*(NEVN-1)+1+4+2+2
0167100      IF(NPLY/2*2.NE.NPLY.AND.NSYM2.EQ.2) NBC = 8*(NEVN-1)+1+2
0167200      IF(NPLY/2*2.EQ.NPLY.AND.NSYM2.EQ.2) NBC = 8*(NEVN-1)+1+4+2
0167300      DO 7119 I=1,NEVN
0167400      J=4*(NPLY+1-I)
0167500      ISGN(J-3)= (4*I-3)*(-1.)**(NSYM2)
0167600      ISGN(J-2)= (4*I-2)*(-1.)**(NSYM2+1)
0167700      ISGN(J-5)= (4*I-1)*(-1.)**(NSYM2+1)
0167800 7119  ISGN(J-4)= (4*I-0)*(-1.)**(NSYM2)
0167900      DO 7121 I=1,N3EVN
0168000      IF(I.EQ.I2) GOTO 7121
0168100      DO 71212 J=N2EVN,N4EVN
0168200      K=ISGN(J)
0168300      JSGN=K/IABS(K)
0168400      K=IABS(K)

```

```

0168500      SK0(I,K)=SK0(I,K)+SK0(I,J)*JSGN
0168600      SK1(I,K)=SK1(I,K)+SK1(I,J)*JSGN
0168700      SK2(I,K)=SK2(I,K)+SK2(I,J)*JSGN
0168800      SK3(I,K)=SK3(I,K)+SK3(I,J)*JSGN
0168900 71212 SK4(I,K)=SK4(I,K)+SK4(I,J)*JSGN
0169000 7121  CONTINUE
0169100      DO 7129 J=1,N3EVN
0169200      SK0(I2,J)=0.D 00
0169300      SK1(I2,J)=0.D 00
0169400      SK2(I2,J)=0.D 00
0169500      SK3(I2,J)=0.D 00
0169600 7129  SK4(I2,J)=0.D 00
0169700      CALL PZERO(SK0(1,I1),N3EVN)
0169800      CALL PZERO(SK1(1,I1),N3EVN)
0169900      CALL PZERO(SK2(1,I1),N3EVN)
0170000      CALL PZERO(SK3(1,I1),N3EVN)
0170100      CALL PZERO(SK4(1,I1),N3EVN)
0170200      SK1(I2,I1)=1.
0170300 6390  FORMAT(1X,I2,14F9.4)
0170400 C      NQQ=8
0170500 C      DO 6391 I=1,NQQ
0170600 C6391  WRITE(6,6390) I,(SK0(I,J),J=1,NQQ)
0170700 C      DO 6392 I=1,NQQ
0170800 C6392  WRITE(6,6390) I,(SK1(I,J),J=1,NQQ)
0170900 C      DO 6393 I=1,NQQ
0171000 C6393  WRITE(6,6390) I,(SK2(I,J),J=1,NQQ)
0171100 C      DO 6394 I=1,NQQ
0171200 C6394  WRITE(6,6390) I,(SK3(I,J),J=1,NQQ)
0171300 C      DO 6395 I=1,NQQ
0171400 C6395  WRITE(6,6390) I,(SK4(I,J),J=1,NQQ)
0171500      RETURN
0171600 142   NEQ=4*NPLY-2
0171700      NBC=8*NPLY-2
0171800      RETURN
0171900      END
0172000      SUBROUTINE PZERO(A,N)
0172100      IMPLICIT REAL (A-H,O-Z)
0172200      DIMENSION A(1)
0172300      DO 1 I=1,N
0172400 1     A(I)=0.D 00
0172500      RETURN
0172600      END
0172700      SUBROUTINE BIDOT(A,I1,B,I2,C,I3)
0172800      IMPLICIT REAL (A-H,O-Z)
0172900      DIMENSION A(I1,1),B(I2,1),C(I1,1)
0173000      DO 1 I=1,I1
0173100      DO 1 K=1,I3
0173200      C(I,K)=0.D 00
0173300      DO 1 J=1,I2
0173400 1     C(I,K)=C(I,K)+A(I,J)*B(J,K)
0173500      RETURN
0173600      END
0173700      SUBROUTINE SKINV(SK,NSK,SKI,NSKI,X,NX,INDIC,SIMUL,N)
0173800      IMPLICIT REAL (A-H,O-Z)

```

```

0173900 C.....INDIC =0 GIVES DETE. (SIMUL), INVERSE (SKI) AND SOLUTION(X)
0174000 C.....INDIC -VE GIVES DETERMINANT AND INVERSE
0174100 C.....INDIC +VE GIVES DETERMINANT AND SOLUTION
0174200     DIMENSION IROW(251),JCOL(251),JORD(251),Y(251),A(251,251),SK(NSK,1-
0174300     *),SKI(NSKI,1),X(NX)
0174400     EPS=1.D-70
0174500     NRC=251
0174600     MAX=N
0174700     IF(INDIC.GE.0) MAX=N+1
0174800     DO 101 I=1,N
0174900     DO 101 J=1,N
0175000 101  A(I,J)=SK(I,J)
0175100     IF(INDIC.LT.0) GO TO 103
0175200     DO 102 I=1,N
0175300 102  A(I,MAX)=X(I)
0175400 103  CONTINUE
0175500     IF(N.LT.NRC) GOTO 5
0175600     WRITE(6,200)
0175700 200  FORMAT(/1X,'MATRIX IS TOO BIG IN "SKINV" '/')
0175800     RETURN
0175900 5    CONTINUE
0176000     DETER=1.
0176100     DO 18 K=1,N
0176200     KM1=K-1
0176300     PIVOT=0.
0176400     DO 11 I=1,N
0176500     DO 11 J=1,N
0176600     IF(K.EQ.1) GOTO 9
0176700     DO 8 ISCAN=1,KM1
0176800     DO 8 JSCAN=1,KM1
0176900     IF(I.EQ.IROW(ISCAN)) GOTO 11
0177000     IF(J.EQ.JCOL(JSCAN)) GOTO 11
0177100 8    CONTINUE
0177200 9    IF(DABS(A(I,J)).LE.DABS(PIVOT)) GOTO 11
0177300     PIVOT=A(I,J)
0177400     IROW(K)=I
0177500     JCOL(K)=J
0177600 11   CONTINUE
0177700     IF(DABS(PIVOT).GT.EPS) GOTO 13
0177800     SIMUL=0.
0177900     WRITE(6,1)
0178000 1    FORMAT(1X,'STOPPED IN SUBROUTINE SKINV..... DETERMINANT = 0')
0178100     RETURN
0178200 13   IROWK=IROW(K)
0178300     JCOLK=JCOL(K)
0178400     DETER=DETER*PIVOT
0178500     DO 14 J=1,MAX
0178600 14   A(IROWK,J)=A(IROWK,J)/PIVOT
0178700     A(IROWK,JCOLK)=1./PIVOT
0178800     DO 18 I=1,N
0178900     AIJCK=A(I,JCOLK)
0179000     IF(I.EQ.IROWK) GOTO 18
0179100     A(I,JCOLK)=-AIJCK/PIVOT
0179200     DO 17 J=1,MAX

```



```

0179300 17   IF(J.NE.JCOLK) A(I,J)=A(I,J)-AIJCK*A(IROWK,J)
0179400 18   CONTINUE
0179500     DO 20 I=1,N
0179600     IROWI=IROW(I)
0179700     JCOLI=JCOL(I)
0179800     JORD(IROWI)=JCOLI
0179900 20   IF(INDIC.GE.0) X(JCOLI)=A(IROWI,MAX)
0180000     INTCH=0
0180100     NM1=N-1
0180200     DO 22 I=1,NM1
0180300     IP1=I+1
0180400     DO 22 J=IP1,N
0180500     IF(JORD(J).GE.JORD(I)) GOTO 22
0180600     JTEMP=JORD(J)
0180700     JORD(J)=JORD(I)
0180800     JORD(I)=JTEMP
0180900     INTCH=INTCH+1
0181000 22   CONTINUE
0181100     IF(INTCH/2*2.NE.INTCH) DETER=-DETER
0181200     IF(INDIC.LE.0) GOTO 26
0181300     SIMUL=DETER
0181400     RETURN
0181500 26   DO 28 J=1,N
0181600     DO 27 I=1,N
0181700     IROWI=IROW(I)
0181800     JCOLI=JCOL(I)
0181900 27   Y(JCOLI)=A(IROWI,J)
0182000     DO 28 I=1,N
0182100 28   A(I,J)=Y(I)
0182200     DO 30 I=1,N
0182300     DO 29 J=1,N
0182400     IROWJ=IROW(J)
0182500     JCOLJ=JCOL(J)
0182600 29   Y(IROWJ)=A(I,JCOLJ)
0182700     DO 30 J=1,N
0182800 30   A(I,J)=Y(J)
0182900     SIMUL=DETER
0183000     DO 209 I=1,N
0183100     DO 209 J=1,N
0183200 209  SKI(I,J)=A(I,J)
0183300     RETURN
0183400     END

```

APPENDIX II: PLY EQUATIONS

The derivation of these equations is reported in reference 8. These equations encompass the following main categories: overall equilibrium, constitutive relations, stress distribution, and displacement distribution.

Overall Equilibrium:

$$N_{2,2}^k + n_2^k = 0 \quad (I-1)$$

$$M_{2,2}^k + c_k m_2^k - Q_2^k = 0 \quad (I-2)$$

$$Q_{2,2}^k + q^k = 0 \quad (I-3)$$

where

$$n_2^k = \sigma_{2z}^k(x_2, c_k) - \sigma_{2z}^k(x_2, -c_k) \quad (I-4)$$

$$m_2^k = \sigma_{2z}^k(x_2, c_k) - \sigma_{2z}^k(x_2, -c_k) \quad (I-5)$$

$$q^k = \sigma_{zz}^k(x_2, c_k) - \sigma_{zz}^k(x_2, -c_k) \quad (I-6)$$

where N_2 , M_2 and Q_2 are, respectively, the force, moment, and shear resultants associated with the x_2 -coordinate direction, and n_2 , m_2 , and q are related to the interfacial stresses. The semithickness of the k -th ply is c_k . Superscript k which identifies the generic ply will be dropped in the subsequent equations for convenience.

Constitutive Relations:

$$N_{\beta}/h = \bar{C}_{\beta 1} \epsilon + \bar{C}_{\beta 2} U_{2,2} + K_{n\beta} c n_{2,2} + K_{p\beta} p ; \beta = 1,2 \quad (I-7)$$

$$M_{\beta}/I = -\bar{C}_{\beta 1} W_{,11} - \bar{C}_{\beta 2} W_{,22} + K_{m\beta} m_{2,2} + K_{q\beta} q/c \quad (I-8)$$

$$\phi_2 + W_{,2} = (Q_2 - c m_{2,2}/3) S_{44} c^2 / (2I) \quad (I-9)$$

where

$$K_{m\beta} = (3 \bar{C}_{\beta 2} S_{3\alpha} \bar{C}_{\alpha 2} / \bar{C}_{22} - 2 \bar{C}_{\beta 2} S_{44} + 2 \bar{C}_{\beta}) / 20 ; \alpha, \beta = 1,2$$

$$K_{q\beta} = (3 \bar{C}_{\beta 2} S_{3\alpha} \bar{C}_{\alpha 2} / \bar{C}_{22} - 12 \bar{C}_{\beta 2} S_{44} + 12 \bar{C}_{\beta}) / 20$$

$$K_{n\beta} = (\bar{C}_{\beta 2} S_{3\alpha} \bar{C}_{\alpha 2} / \bar{C}_{22} + \bar{C}_{\beta 2} S_{44} + 2 \bar{C}_{\beta}) / 12$$

$$K_{p\beta} \bar{C}_{\beta} / 2$$

$$p = \sigma_{zz}(x_2, c) + \sigma_{zz}(x_2, -c) , h = 2c , I = 2 c^3 / 3 \quad (I-10)$$

where ϵ is the applied strain in the x_1 -direction and $W_{,11}$ is the curvature about x_2 -axis. $[\bar{C}]$ is the plane stress stiffness matrix and $[S]$

is the general flexibility matrix. The constants \bar{C}_1 , \bar{C}_2 , and \bar{C}_6 introduce the influence of the transverse normal stress in the computation on the refined surface parallel stresses. For a given lamina with arbitrary orientation of the fibers σ_1 and σ_2 are surface parallel normal stresses, σ_3 is transverse normal stress and σ_6 the in-plane shear stress. The corresponding engineering strains are denoted by ϵ_1 , ϵ_2 , ϵ_3 , and T_6 . Using the generalized Hooke's law, the equations for ϵ_1 , ϵ_2 and T_6 can be inverted to obtain

$$\sigma_i = \bar{C}_{ij}\epsilon_j + \bar{C}_i\sigma_3 \quad ; \quad i, j = 1, 2, \text{ and } 6 \quad (I-11)$$

where ϵ_6 is an alternative notation for T_6 . The expressions for \bar{C}_1 , \bar{C}_2 , and \bar{C}_6 can be obtained in terms of flexibilities and later in terms of lamina elastic moduli with relative ease.

The axial displacement U_2 and the transverse displacement W are associated with reference surface $z = 0$. Here ϕ_2 is the rotation of a normal to the reference surface.

Stress Distribution:

$$\sigma_\beta = N_\beta/h + M_\beta z/I + K_\beta n_{2,2}(z^2 - c^2/3)/(2h) + K_\beta(c m_{2,2} + q)(z^3 - 3c^2z/5)/(6I) \quad (I-12)$$

$$\sigma_{2z} = n_{2z}z/I + c m_{2z}(3z^2 - c^2)/(6I) - Q_2(z^2 - c^2)/(2I) \quad (I-13)$$

$$\sigma_{zz} = p/2 - n_{2,2}(z^2 - c^2)/(2h) + q z/h - (c m_{2,2} + q)(z^3 - c^2z)/(6I) \quad (I-14)$$

where

$$K_\beta = \bar{C}_{\beta 2} S_{3\alpha} \bar{C}_{\alpha 2} / \bar{C}_{22} + \bar{C}_{\beta 2} S_{44} - \bar{C}_\beta \quad ; \quad \alpha, \beta = 1 \text{ and } 2 \quad (I-15)$$

σ_1 = surface parallel stress in x_1 direction,

σ_2 = surface parallel stress in x_2 direction,

σ_{2z} = shear stress in x_2 - z plane, and

σ_{zz} = transverse normal stress in z direction.

Displacement Distribution:

$$\begin{aligned} w = W + S_{3\alpha}(N_\alpha z/h + M_\alpha z^2/(2I)) + S_{3\alpha}K_\alpha n_{2,2}(z^3 - c^2z)/(6h) \\ + S_{3\alpha}K_\alpha(c m_{2,2} + q)(z^4/4 - 3c^2z^2/10)/(6I) \\ + S_{33}(p z/2 + q z^2/(2h)) - S_{33}n_{2,2}(z^3 - 3c^2z)/(6h) \\ + S_{33}(c m_{2,2} + q)(z^4/4 - c^2z^2/2)/(6I) \end{aligned} \quad (I-16)$$

$$\begin{aligned}
u = & U_2 - z W_{,2} - S_{3\alpha} (N_\alpha z^2 / (2h) + M_\alpha z^3 / (6I))_{,2} \\
& - S_{3\alpha} K_\alpha \{ n_{2,22} (z^4 / 4 - c^2 z^2 / 2) / (6h) \\
& + (c m_{2,22} + q_{,2}) (z^5 / 20 - c^2 z^3 / 10) / (6I) \} + S_{33} \{ p_{,2} z^2 / 4 + q_{,2} z^3 / (6h) \\
& - n_{2,22} (z^4 / 4 - 3 c^2 z^2 / 2) / (6h) - (c m_{2,22} + q_{,2}) (z^5 / 20 - c^2 z^3 / 6) / (6I) \} \\
& + S_{44} \{ z^2 n_2 / (2h) + c m_2 (z^3 - c^2 z) / (6I) - Q_2 (z^3 - 3 c^2 z) / (6I) \} \quad (I-17)
\end{aligned}$$

The Greek subscripts in the foregoing expressions assume values of 1 and 2. The repeated Greek indices indicate a summation over the appropriate range. Finally, u_2 and w are the displacement components in the x_2 and z coordinate directions.

APPENDIX III: INPUT AND OUTPUT INSTRUCTIONS FOR THE CODE

This code contains a MAIN program and 16 subroutines. Additionally, it also needs a subroutine to find the determinant and inverse of a complex matrix. The IMSL mathematical subroutine library is used for this purpose. The program is restricted to cylindrical bending problems in x_2 - z plane. The laminate can be analyzed in stretching in x_1 -direction or in bending in x_1 - z plane.

The MAIN program controls input. The subroutine MPRG controls construction of eigen solutions for each region, construction and solution of boundary/continuity conditions matrix, and finally computation of response variables through the thickness and along the axis of the laminate. The core memory can be controlled by the dimension of RCM vector in MAIN program. Also, set the value of the variable MAXRCM to the same value as the dimension of RCM vector.

The input is divided into three parts. The first part given by Data cards 1 through 5 describes the sublaminates, ply layup, and material properties. The second part generates the eigenvalues and eigen modes. Finally, boundary/continuity conditions are input through Data card 6. The distribution of the response variables is controlled by Data card 7. The first part of the input is explained below:

Data 1: NSREG, NMT

Format(I2)...These variables, respectively, indicate the number of sublaminates which the laminate is divided into and the number of materials used. A different ply angle indicates a different material.

Data 2: EPSLON, CURVTR

Format(8E10.4)...EPSLON is the uniform strain in x_1 -direction and CURVTR is the quotient of curvature in x_1 - z plane.

Data 3: NL, NGN, NGT

Format(20I4)...NL is the number of layers independently analyzed in a sublaminate. A "0" for NGN constructs full solutions. A "1" gives decay type solutions for the current sublaminate. If NGT is either "0" or the current sublaminate identification number, then the full eigenvalue problem is solved for the current sublaminate. If NGT is initiated to a number less than the current sublaminate identification number, then its eigen solutions are not computed. Instead these are assumed to be the same as those of the NGT sublaminate. This means that the present sublaminate is identically the same as the NGT sublaminate. The data for NSREG sublaminates is given here.

Data 4: (THICK(K), K=1,NL)

Format(8E10.4)...Thicknesses for the NL layers of the sublaminate. The data for all the NSREG sublaminates are entered here.

Data 5: $E_1, E_2, E_3, \nu_{12}, \nu_{13}, \nu_{23}, G_{23}, G_{13}, G_{12}, \theta$

Format(10f8.4)...Material properties along ply principal directions. θ is the angle of orientation. The data for all the NMT materials is entered here.

The second part of the data, to compute eigen modes, is to be given interactively. This procedure can be automated if a reasonable approximation of the eigen values is known. It begins with the following series of prompts:

Prompt 1: "GIVE 0 IF YOU DO NOT WANT POLYNOMIAL APPROX.
NR = xx, nsym = xxx"

A default ("0") uses complete eigen matrix to evaluate real roots. Any other number uses a polynomial representation. Control goes to Prompt 2. NR indicates the current sublaminates for which real roots are sought. There is a do loop on NR covering all sublaminates. Sublaminates are skipped for which NGN is not initiated to "0" or their proper identification number. There is a do loop on NSYM. Pure stretching modes are constructed if NGN is 1. Pure bending modes are computed when it is 2. Both are computed when NSYM = 3. These do loops end when the reply to Prompt 5 is a default.

Prompt 2: "GIVE 0 IF YOU WANT REAL ROOTS"

A default computes real roots by going to Prompt 9. Any other number transfers to Prompt 3.

Prompt 3: "GIVE 0 IF YOU DO NOT WANT POLYNOMIAL APPROX.
NR = xx, nsym = xxx"

A default ("0") used complete eigen matrix to evaluate complex roots. Any other number uses a polynomial representation. Control goes to Prompt 4.

Prompt 4: "GIVE 0 IF YOU WANT COMPLEX ROOTS"

A default computes complex roots by going to Prompt 13. Any other number transfers to Prompt 5.

Prompt 5: "GIVE 0 IF POLYNOMIAL IS NOT DESIRED"

A default ends do loops on NR and NSYM. Any other number initiates the process for construction of a polynomial suppressing the real and complex roots constructed thus far by transferring control to Prompt 6. Here, it also shows the roots computed thus far.

Prompt 6: "GIVE NRT FOR POLYNOMIAL...DEFAULT IS AS IS A NEGATIVE VALUE
INITIATES NRT TO 0"

NRT indicates the number of real roots computed thus far for the particular cases of NR and NSYM. A negative value initiates NRT to 0. Any other number less than the current NRT initiates NRT to that value. This helps controlling the number of real roots in the polynomial evaluation. Control next goes to Prompt 7.

Prompt 7: "GIVE NCT FOR POLYNOMIAL...DEFAULT IS AS IS A NEGATIVE VALUE
INITIATES NCT TO 0"

NCT indicates the number of complex roots computed thus far for the particular cases of NR and NSYM. A negative value initiates NCT to 0. Any other number less than the current NCT initiates NCT to that value. This helps

controlling the number of complex roots in the polynomial evaluation. Control goes to Prompt 8.

Prompt 8: "GIVE 0 IF POLYNOMIAL IS NOT DESIRED"

A default transfers the control to Prompt 1 without changing either NR or NSYM. Any other number computes the polynomial. Control again shifts to Prompt 1.

Prompt 9: "10F10.6...XI...INITIAL ROOT VALUE"

This commences the process for finding real roots. This is based on Newton-Raphson technique. XI is the initial root value. A default transfers control to Prompt 11. If there is convergence, the root is displayed and the control goes to Prompt 10. If there is no convergence after 100 iterations the control goes to Prompt 9 displaying the fact.

Prompt 10: "12...INPUT...J=1 TO STORE ROOT"

After the converged root is displayed, it asks whether the root is to be saved (input "1") or not (default) for the purpose of a later storage. If allowed, it stores two values for the root, +xxx and -xxx, and appends NRT by 2. If NGN is not initiated to "0" only one root is stored and NRT is increased by 1. The choice x_2 -origin in relation to the edge where decay type solutions are required determines whether to store positive or negative roots. Control goes to Prompt 9. NRT starts at 0 for each NR and NSYM. At the behest of Prompt 6 this parameter can be changed to drop unwanted roots.

Prompt 11: "12...INPUT...J=0 TO FIND REAL EIGEN MODES"

A default computes the eigen modes. Next is Prompt 12.

Prompt 12: "GIVE 0 IF REAL ROOTS ARE TO BE STORED"

A default stores real roots and real modes and sets NRT back to zero. Once the roots are stored, then the polynomial at Prompt 5 cannot be constructed by suppressing these roots. Control goes to Prompt 3.

Prompts 13 through 16:

The "in" control comes from Prompt 4 and the "out" control goes to Prompt 5. These prompts do the same job as described for Prompts 9 through 12, but for complex roots. NCT keeps track of the number of complex roots to be stored. Four roots...positive, positive complex conjugate, negative, and negative complex conjugate versions will be saved and NCT will be appended by 4 if NGN is "0." Otherwise, only positive or negative roots (depends on the discretion of the user) will be saved and NCT will be padded up by two.

The logic outlined by these prompts is elementary and is useful for finding the eigen solutions. User ends this process after determining that all roots are found and stored. Once a root is stored at the instance of either Prompt 10 or 14, it will be suppressed so that the next iteration will not converge on to this root. The next set of data concerns the boundary/continuity conditions and printing of response variables.

Data 6: NBCREG, NBCPLY, NBCV, YNBC, ZNBC, PNBC, NBCREGL

Format(3I4,3E12.6,I4)...Repeat this data to cover all boundary and continuity conditions. NBCREG is the sublaminate, NBCPLY is the ply, NBCV is the variable, YNBC is the local x_2 -coordinate ZNBC is the local z coordinate. The origin for the local z axis is always in the local ply middle plane. Origin for x_2 -coordinate can be placed anywhere but care should be taken when using pure decay type solutions. For each sublaminate, the plies are numbered 1, 2, 3,...locally from top to bottom.

- NBCV = 1 indicates $N_2(x_2)$
- NBCV = 2 indicates $u_2(x_2, z)$
- NBCV = 3 indicates $M_2(x_2)$
- NBCV = 4 indicates $\phi_2(x_2)$
- NBCV = 5 indicates $Q_2(x_2)$
- NBCV = 6 indicates $w(x_2, z)$
- NBCV = 7 indicates $\sigma_{22}(x_2, z)$
- NBCV = 8 indicates $\sigma_{2z}(x_2, z)$
- NBCV = 9 indicates $\sigma_{zz}(x_2, z)$
- NBCV = 10 indicates transverse stretch parameter
- NBCV = 11 indicates $\sigma_{z,2}(x_2, z)$

The tenth variable, transverse thickness stretch parameter, denotes the coefficient of z in the thicknesswise distribution of the transverse displacement. N_2 , M_2 , and Q_2 are defined layerwise when full solutions are used. At edges where decay type solutions are present, this definition is again retained. At far edges, however, these are defined over entire sublaminate thickness preserving local force equilibrium. For conditions involving several variables (NV's), this data continues on to the next card and is terminated by specifying NBCREGL = 0 on the last card.

For example,

```

2  1  1 .100000d 01 .000000d 00 .120000d 03  8
1  3  3-.100000d 01 .000000d 00-.100000d 01  8
2  2  2 .200000d 01-.200000d 00 .900000d 01  8
4  7  5-.100000d 01-.200000d 00 .200000d 01  0

```

indicates the following:

$$N_2 - M_2 + 9U_2 + 2Q_2 + 120 = 0$$

where,

N_2 is defined for the second sublaminate and first ply at $x_2 = 1$,
 M_2 is defined for the first sublaminate and third ply at $x_2 = -1$,
 u_2 is defined for the second sublaminate and second ply at $x_2 = 2, z = -2$,
 Q_2 is defined for the fourth sublaminate and seventh ply at $x_2 = -1$.
 PS in the first card carries the inhomogeneous part of the condition. In subsequent cards, PS's carry the coefficients of the second, third, and fourth variables, respectively. NBCREGL is an arbitrary number in the first three cards.

Data 7: NDREG, NDPLY, NDV, NDVT, YND, ZND, STPND, NDP

Format(4I4,3d12.6,I4)...A value "2" for NDVT gives distribution of NDV variable of the NDPLY ply in the NDREG sublamine with respect to x_2 -coordinate starting at $x_2 = YND$ computed NDP times at in interval of STPND. A value "3" gives the distribution with respect to z coordinate. Repeat Data 7 as many times as is required. A blank card terminates the computations of all distributions.

The source code is given in appendix I. Typical input and output for a ENF specimen are shown in the next two appendices. The output is self-explanatory. The number of plys, real and complex roots, eigen modes and boundary conditions for each region are printed. The input boundary/continuity conditions are also printed.

[CBB] is the plane stress stiffness matrix. CB11, CB12,...are its elements. {CB} is the vector referred to in equation I-11 of appendix II. SNC1 and SNC2 denote K_{n1} and K_{n2} of equation I-12. SHM2 and SHQ2 denote the coefficients of Q_2 and m_2 in equation I-9. KN1, KN2, KM1, KM2, KQ1, KQ2, KP1, and KP2 are defined in equation I-11.

The amplitudes of the governing variables are given as rows in four columns under ZERO MODE, REGION = xx and EIGENVALUE and MODE FOR REGION = xx. The governing variables in each region are assumed in the order of U_2 , W , $\sigma_{2z}(x_2, z = -c)$ and $\sigma_{zz}(x_2, z = -c)$ for each layer in that region starting from top layer to the bottom layer. The different columns indicate the mode shape. In the case of zero modes, the first, second, third, and fourth columns denote, respectively, the coefficients of x_2^0 , x_2^1 , x_2^2 and x_2^3 . In the case of real roots only the first two columns are meaningful. They represent the coefficients of $\cosh(a*x_2^2/H)$ and $\sinh(a*x_2^2/h)$, in that order. If NRGN is not "0," then the second column will also become zero and the first column represents the coefficient of $\exp(a*x_2^2/H)$. Symbols, a and H, denote the real root and the thickness of the sublamine respectively. In the case of complex roots, these columns represent the coefficients of $\cosh(a*x_2^2/H)*\cos(b*x_2^2/H)$, $\cosh(a*x_2^2/H)*\sin(b*x_2^2/H)$, $\sinh(a*x_2^2/H)*\cos(b*x_2^2/H)$ and $\sinh(a*x_2^2/H)*\sin(b*x_2^2/H)$, respectively. The first and second columns would represent the coefficients of $\exp(a*x_2^2/H)*\cos(b*x_2^2/H)$ and $\exp(a*x_2^2/H)*\sin(b*x_2^2/H)$ if NRGN is not initiated to zero. $a+ib$ denotes a complex root.

The distributions are output with Y, F, Z, NR, NPLY, and NV denoting x_2 -coordinate, value of the variable NV, z-coordinate, sublamine, ply, and the variable NV, in that order.

APPENDIX IV: TYPICAL INUT FOR ENF SPECIMEN

```

4 2
0.0000 0.0000
1
3 1
1 0 0
1
1
1 2 1
1
1
1.200D-01
0.600D-01 0.400D-04 0.600D-01
0.600D-01
0.600D-01
18.8000 1.1930 1.1930 0.2600 0.2600 0.4230 0.3620 0.6230 0.6230 90.0000
0.5000 0.5000 0.5000 0.3500 0.3500 0.3500 0.1852 0.1852 0.1852 90.0000
ONK10
OREAL
3.75
1STORE
1.05
1STORE
0 EXIT
0 EIGEN
0STORE
ONK10
0COMPLEX
808.000000764.
1STORE
EXIT
EIGEN
STORE
0POLY
ONK10
OREAL
2.5
1STORE
.59
1STORE
0 EXIT
0EIGEN
0DO NOT STORE
ONK10
0COMPLEX
23.200000023.3
1STORE
EXIT
0EIGEN
0DO NOT STORE
0POLY
ONK10
1REAL
ONK10
1COMPLEX

```

OPOLY

1	1	2-.100000D 01	0.000000	0.000000	0
1	1	3-.100000D 01	0.000000	0.000000	0
1	1	6-.100000D 01	0.000000	0.000000	0
1	1	10.100000D 01	0.000000	0.000000	2
2	1	1-.100000D 01	0.000000	-1.000000	0
1	1	20.100000D 01	0.000000	0.000000	2
2	1	2-.100000D 01-.300000D-01		-1.000000	0
1	1	30.100000D 01	0.000000	0.000000	2
2	1	3-.100000D 01	0.000000	-1.000000	0
1	1	40.100000D 01	0.000000	0.000000	2
2	1	4-.100000D 01-.300000D-01		-1.000000	0
1	1	50.100000D 01	0.000000	.120000D 03	2
2	1	5-.100000D 01	0.000000	-1.000000	0
1	1	60.100000D 01	0.000000	0.000000	2
2	1	6-.100000D 01-.300000D-01		-1.000000	0
2	1	1	0.000000	0.000000	3
3	1	1	0.000000	0.000000	0
2	1	2	0.000000	0.000000	3
3	1	2	0.000000	0.000000	0
2	1	3	0.000000	0.000000	3
3	1	3	0.000000	0.000000	0
2	1	4	0.000000	0.000000	3
3	1	4	0.000000	0.000000	0
2	1	5	0.000000	0.000000	3
3	1	5	0.000000	0.000000	0
2	1	6	0.000000	0.000000	3
3	1	6	0.000000	0.000000	0
2	1	10	0.000000-.300000D-01	0.000000	3
3	1	10	0.000000-.300000D-01	0.000000	0
2	3	10	0.000000 .300000D-01	0.000000	4
4	1	10	0.000000 .300000D-01	0.000000	0
2	2	1	0.000000	0.000000	0
2	2	3	0.000000	0.000000	0
2	2	5	0.000000	0.000000	0
2	3	1	0.000000	0.000000	4
4	1	1	0.000000	0.000000	0
2	3	2	0.000000	0.000000	4
4	1	2	0.000000	0.000000	0
2	3	3	0.000000	0.000000	4
4	1	3	0.000000	0.000000	0
2	3	4	0.000000	0.000000	4
4	1	4	0.000000	0.000000	0
2	3	5	0.000000	0.000000	4
4	1	5	0.000000	0.000000	0
2	3	6	0.000000	0.000000	4
4	1	6	0.000000	0.000000	0
3	1	11.000000D 00	0.000000	0.000000	0
3	1	31.000000D 00	0.000000	0.000000	0
3	1	51.000000D 00	0.000000	0.000000	0
4	1	11.000000D 00	0.000000	0.000000	0
4	1	31.000000D 00	0.000000	0.000000	0
4	1	61.000000D 00	0.000000	0.000000	0
1	1	6	2-.100000D 01-.600000D-01	11.000000D 00	2

1	1	1	2	-.100000D	01	-.000000D	-011.000000D	00	2
1	1	3	2	-.100000D	01	-.000000D	-011.000000D	00	2
1	1	5	2	-.100000D	01	-.000000D	-011.000000D	00	2
2	1	1	2	-.100000D	01	-.000000D	-010.500000D	00	2
2	1	3	2	-.100000D	01	-.000000D	-010.500000D	00	2
2	1	5	2	-.100000D	01	-.000000D	-010.500000D	00	2
2	2	1	2	-.100000D	01	-.000000D	-010.500000D	00	2
2	2	3	2	-.100000D	01	-.000000D	-010.500000D	00	2
2	2	5	2	-.100000D	01	-.000000D	-010.500000D	00	2
2	3	1	2	-.100000D	01	-.000000D	-010.500000D	00	2
2	3	3	2	-.100000D	01	-.000000D	-010.500000D	00	2
2	3	5	2	-.100000D	01	-.000000D	-010.500000D	00	2
3	1	1	2	-.000000D	01	-.000000D	-010.500000D	00	2
3	1	3	2	-.000000D	01	-.000000D	-010.500000D	00	2
3	1	5	2	-.000000D	01	-.000000D	-010.500000D	00	2
4	1	1	2	-.000000D	01	-.000000D	-010.500000D	00	2
4	1	3	2	-.000000D	01	-.000000D	-010.500000D	00	2
4	1	5	2	-.000000D	01	-.000000D	-010.500000D	00	2
2	2	7	2	0.000000		0.000000	-.625000D	-02	160
2	2	8	2	0.000000		0.000000	-.625000D	-02	160
2	2	9	2	0.000000		0.000000	-.625000D	-02	160

/EOF

APPENDIX V: TYPICAL OUTPUT FOR ENF SPECIMEN

4 2 NO. OF REGIONS AND NO. OF MATERIALS

0.0000E+00.0000E+00 EPSILON, CURVTR

REGION = 1, NO. OF LAYERS = 1, NGN = 0, NGT = 0
 REGION = 2, NO. OF LAYERS = 3, NGN = 1, NGT = 0
 REGION = 3, NO. OF LAYERS = 1, NGN = 0, NGT = 0
 REGION = 4, NO. OF LAYERS = 1, NGN = 0, NGT = 0

REGION = 1, MATERIAL FOR EACH LAYER

REGION = 2, MATERIAL FOR EACH LAYER

REGION = 3, MATERIAL FOR EACH LAYER

REGION = 4, MATERIAL FOR EACH LAYER

REGION = 1, PLY THICKNESSES
 0.1200E+00

REGION = 2, PLY THICKNESSES
 0.6000E-010.4000E-040.6000E-01

REGION = 3, PLY THICKNESSES
 0.6000E-01

REGION = 4, PLY THICKNESSES
 0.6000E-01

SUBLAMINATE THICKNESSES
 0.1200E+00.1200E+00.6000E-010.6000E-01
 1

E11,E22,E33,NU12,NU13,NU23,G44,G55,G66,THETA
 0.1880E+020.1193E+010.1193E+010.2600E+000.2600E+000.4230E+000.3620E+000.6230E+00
 0.6230E+000.9000E+02
 0.5000E+000.5000E+000.5000E+000.3500E+000.3500E+000.3500E+000.1852E+000.1852E+00
 0.1852E+000.9000E+02

FLEXIBILITY MATRIX S FOR 1 MATERIAL
 0.8382E+00-.1383E-01-.3546E+000.0000E+000.0000E+000.0000E+000.1697E-15

- .1383E-010 .5319E-01- .1383E-010 .0000E+000 .0000E+000 .2523E-14
- .3546E+000 - .1383E-010 .8382E+000 .0000E+000 .0000E+000 - .1169E-14
0 .0000E+000 .0000E+000 .0000E+000 .1605E+010 .1985E-140 .0000E+000
0 .0000E+000 .0000E+000 .0000E+000 .1985E-140 .2762E+010 .0000E+000
0 .1697E-150 .2523E-14- .1169E-140 .0000E+000 .0000E+000 .1605E+01

STIFFNESS MATRIX C FOR 1 MATERIAL

0 .1469E+010 .5457E+000 .6302E+000 .0000E+000 .0000E+000 - .5542E-15
0 .5457E+000 .1908E+020 .5457E+000 .0000E+000 .0000E+000 - .2966E-13
0 .6302E+000 .5457E+000 .1469E+010 .0000E+000 .0000E+000 .1450E-15
0 .0000E+000 .0000E+000 .0000E+000 .6230E+000 .4476E-150 .0000E+000
0 .0000E+000 .0000E+000 .0000E+000 - .4476E-150 .3620E+000 .0000E+000
- .5542E-15- .2966E-130 .1450E-150 .0000E+000 .0000E+000 .6230E+000

PLANE STRESS STIFFNESS MATRIX CBB, CB OF

0 .1198E+010 .3115E+000 - .6164E-150 .4291E+000
0 .3115E+000 .1888E+02- .2971E-130 .3716E+00
- .6164E-15- .2971E-130 .6230E+000 .9872E-16

FLEXIBILITY MATRIX S FOR 2 MATERIAL

0 .2000E+01- .7000E+00- .7000E+000 .0000E+000 .0000E+000 .7409E-18
- .7000E+000 .2000E+01- .7000E+000 .0000E+000 .0000E+000 - .7409E-18
- .7000E+00- .7000E+000 .2000E+010 .0000E+000 .0000E+000 .0000E+000
0 .0000E+000 .0000E+000 .0000E+000 .5400E+010 .0000E+000 .0000E+000
0 .0000E+000 .0000E+000 .0000E+000 .0000E+000 .5400E+010 .0000E+000
0 .7409E-18- .7409E-180 .0000E+000 .0000E+000 .0000E+000 .5400E+01

STIFFNESS MATRIX C FOR 2 MATERIAL

0 .8025E+000 .4321E+000 .4321E+000 .0000E+000 .0000E+000 - .5082E-19
0 .4321E+000 .8025E+000 .4321E+000 .0000E+000 .0000E+000 .5082E-19
0 .4321E+000 .4321E+000 .8025E+000 .0000E+000 .0000E+000 - .4636E-33
0 .0000E+000 .0000E+000 .0000E+000 .1852E+000 .0000E+000 .0000E+000
0 .0000E+000 .0000E+000 .0000E+000 .0000E+000 .1852E+000 .0000E+000
- .5082E-190 .5082E-19- .4636E-330 .0000E+000 .0000E+000 .1852E+000

PLANE STRESS STIFFNESS MATRIX CBB, CB OF

0 .5698E+000 .1994E+000 - .5082E-190 .5385E+000
0 .1994E+000 .5698E+000 .5082E-190 .5385E+000
- .5082E-190 .5082E-190 .1852E+000 - .5778E-33

CB11, CB12, CB22, CB26, CB66, CB1, CB2, CB6

0 .1198E+010 .3115E+000 - .6164E-150 .1888E+02- .2971E-130 .6230E+000 .4291E+000 .3716E+00
0 .0000E+000
0 .5698E+000 .1994E+000 - .5082E-190 .5698E+000 .5082E-190 .1852E+000 .5385E+000 .5385E+000
0 .0000E+000

S31, S32, S33, S44, S45, S55, SNC1, SNC2, SHM2, SHQ2

- .3546E+000 - .1383E-010 .8382E+000 .1605E+010 .1985E-140 .2762E+010 .6477E-010 .2956E+02
- .4013E+000 .1204E+01
- .7000E+000 - .7000E+000 .2000E+010 .5400E+010 .0000E+000 .5400E+010 .3499E+000 .2000E+01
- .1350E+010 .4050E+01

KN1, KN2, KP1, KP2, KM1, KM2, KQ1, KQ2

0 .1127E+000 .2557E+010 .2146E+000 .1858E+000 - .8009E-02- .3049E+01- .4346E-01- .1802E+02

0.1638E+00 .3013E+00 .2692E+00 .2692E+00-.8211E-01-.3346E+00-.3513E+00-.1604E+01
 79 80 81 82 83 87 91 95 101 107 111 115 119 123 127 131 211 40 10 3411
 3427 5027 5067 5107 5147 5187 5227

1 PARTICULAR SOLUTION FOR REGION = 1
 0.0000E+00 0.0000E+00

1 ZERO MODE, 1 REGION
 0.000000 0.000000
 0.000000 1.000000

2 ZERO MODE, 1 REGION
 0.000000 0.000000
 1.000000 0.000000

3 ZERO MODE, 1 REGION
 0.000000 0.000000
 0.000000 0.000000

4 ZERO MODE, 1 REGION
 0.000000 0.000000
 0.000000 1.000000

5 ZERO MODE, 1 REGION
 1.000000 0.000000
 0.000000 0.000000

6 ZERO MODE, 1 REGION
 0.000000 1.000000
 0.000000 0.000000

1 PARTICULAR SOLUTION FOR REGION = 2
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00

7 ZERO MODE, 2 REGION
 0.000000 0.000000
 0.000000 1.000000
 0.000000 0.000000
 0.000000 0.000000
 750.249917 0.000000
 0.000000 1500.000000
 0.000000 0.000000
 0.000000 0.000000
 1.000333 0.000000
 0.000000 1.000000

8 ZERO MODE, 2 REGION
 0.000000 0.000000
 1.000000 0.000000
 0.000000 0.000000
 0.000000 0.000000
 0.000000 0.000000
 1.000000 0.000000
 0.000000 0.000000

0.000000
0.000000
0.000000

0.000000
0.000000
0.000000

0.000000
0.000000
0.000000

0.000000
1.000000
0.000000
0.000000
0.000000
1500.000000
0.000000
0.000000
0.000000
0.000000
1.000000

-1.500500
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
1.500500
0.000000

2 REGION
0.000000
0.000000
-28.312051
0.000000
62492.292815
0.000000
-418.299411
0.000000
0.000000
0.000000
83.323057
0.000000
0.000000

0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000

0.000000
1.000000
0.000000
0.000000
0.000000
0.000000
1500.000000
0.000000
0.000000
0.000000
0.000000
2.000666
0.000000
1.000000

2 REGION
0.000000
0.000000
0.000000
0.000000
1500.499833
0.000000
0.000000
0.000000
0.000000
0.000000
2.000666
0.000000
0.000000

0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000

0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000

2 REGION
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000

0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000

0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000
0.000000

2 REGION
1.000000
0.000000
0.000000
0.000000
1500.000000
0.000000
0.000000
0.000000
0.000000
0.000000
1.000000
0.000000
0.000000

GIVE 0 IF YOU DO NOT WANT POLYNOMIAL APPROX.
NR = 2 NSYM = 1
GIVE 0 IF YOU WANT REAL ROOTS


```

10F10.6.....XI..INPUT INITIAL ROOT VALUE
1 0.3750000E+01 0.4731999E-07 I,XI,FI
2 0.3748037E+01 0.2284049E-10 I,XI,FI
3 0.3748036E+01 0.9602490E-13 I,XI,FI
4 0.3748036E+01 0.9651383E-17 I,XI,FI
1 0.3748036E+01 0.9651383E-17 I,XI,FI
I2...INPUT...J=1 TO STORE ROOT
10F10.6.....XI..INPUT INITIAL ROOT VALUE
1 0.1050000E+01 0.4297109E-08 I,XI,FI
2 0.1041734E+01 0.1542848E-10 I,XI,FI
3 0.1041704E+01 -0.8610974E-14 I,XI,FI
4 0.1041704E+01 -0.1960541E-17 I,XI,FI
2 0.1041704E+01 -0.1960541E-17 I,XI,FI
I2...INPUT...J=1 TO STORE ROOT
10F10.6.....XI..INPUT INITIAL ROOT VALUE
I2...INPUT J=0 TO FIND REAL EIGEN MODES

```

```

12 2 1 0.3748036E+01 REAL EIGEN VALUE AND MODE
0.1000E+01 -.5208E+00 0.6995E+01 -.8737E-02 0.0000E+00 0.2607E+04
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00

```

```

12 2 2 0.1041704E+01 REAL EIGEN VALUE AND MODE
0.1000E+01 0.1080E+01 0.4290E+01 -.1489E-02 0.0000E+00 0.2479E+04
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
GIVE 0 IF REAL ROOTS ARE TO BE STORED

```

```

13 0.3748036E+01 0.0000000E+00 EIGEN VALUE AND MODE FOR REGION = 2
1.000000 0.000000 0.000000 0.000000
-0.520829 0.000000 0.000000 0.000000
6.995287 0.000000 0.000000 0.000000
-0.008737 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000
2607.393034 0.000000 0.000000 0.000000
6.995287 0.000000 0.000000 0.000000
0.008737 0.000000 0.000000 0.000000
-1.000000 0.000000 0.000000 0.000000
-0.520829 0.000000 0.000000 0.000000

```

```

14 0.1041704E+01 0.0000000E+00 EIGEN VALUE AND MODE FOR REGION = 2
1.000000 0.000000 0.000000 0.000000
1.080237 0.000000 0.000000 0.000000
4.290226 0.000000 0.000000 0.000000
-0.001489 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000
2478.639427 0.000000 0.000000 0.000000
4.290226 0.000000 0.000000 0.000000
0.001489 0.000000 0.000000 0.000000
-1.000000 0.000000 0.000000 0.000000
1.080237 0.000000 0.000000 0.000000

```

```

GIVE 0 IF YOU DO NOT WANT POLYNOMIAL APPROX.
NR = 2 NSYM = 1
GIVE 0 IF YOU WANT COMPLEX ROOTS
2F10.6.....XI..INPUT INITIAL COMPLEX ROOT VALUE
1 0.8080000E+03 0.7640000E+03-0.12036548E+04-0.61901902E+03 I,XI,FI

```

```

2 0.80895690E+03 0.76561962E+03 0.13470563E+04 -0.33421342E+03 I,XI,FI
*** TERMINAL ERROR
3 0.80590462E+03 0.76837678E+03 0.17262976E+04 0.53075139E+04 I,XI,FI
*** TERMINAL ERROR
4 0.80557780E+03 0.76824207E+03 0.12990180E+04 0.55361773E+04 I,XI,FI
*** TERMINAL ERROR
5 0.80799898E+03 0.76479830E+03 -0.33968396E+03 0.32976972E+02 I,XI,FI
*** TERMINAL ERROR
6 0.80814428E+03 0.76491454E+03 -0.94234759E+02 -0.29204353E+02 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
7 0.80814522E+03 0.76503220E+03 -0.34344867E+02 -0.66588807E+02 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
8 0.80816312E+03 0.76498867E+03 -0.17958008E+01 -0.11299457E+02 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
9 0.80816635E+03 0.76497798E+03 0.71629596E+01 0.10026871E+01 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
10 0.80816691E+03 0.7649884E+03 0.14287519E+01 -0.23833712E+01 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
11 0.80816873E+03 0.76498343E+03 -0.88597704E+00 -0.67733064E+00 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
12 0.80816902E+03 0.76498272E+03 -0.17799826E+00 0.67669535E-02 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
13 0.80816889E+03 0.76498256E+03 0.96409432E-01 -0.25189725E-02 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
14 0.80816897E+03 0.76498263E+03 -0.42219723E-01 0.19949809E-01 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
15 0.80816890E+03 0.76498256E+03 0.82382746E-01 0.68086943E-02 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
16 0.80816893E+03 0.76498261E+03 0.64987188E-02 -0.94443887E-03 I,XI,FI
*** TERMINAL ERROR
*** TERMINAL ERROR
17 0.80816895E+03 0.76498261E+03 0.29928882E-02 -0.18379697E-02 I,XI,FI
*** TERMINAL ERROR
1 0.80816893E+03 0.76498261E+03 0.29928882E-02 -0.18379697E-02 I,XI,FI
I2...INPUT...J=1 TO STORE ROOT
2F10.6...XI...INPUT INITIAL COMPLEX EIGEN MODES
I2...INPUT J=0 TO FIND COMPLEX ROOT VALUE

```

14 2 1 -0.80816893E+03 0.76498261E+03 COMPLEX EIGEN VALUE AND MODE

0.1000E+01 0.0000E+00 0.1000E+01 0.0000E+00
 0.2503E-02 -0.2370E-02 -0.2503E-02 -0.2370E-02
 0.7625E+01 -0.6344E-05 0.7625E+01 0.6344E-05
 0.1333E+04 0.1262E+04 -0.1333E+04 0.1262E+04
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 -0.3856E+06 -0.3388E+06 0.3856E+06 -0.3388E+06
 GIVE 0 IF COMPLEX ROOTS ARE TO BE STORED

15 0.80816893E+03 0.76498261E+03 EIGEN VALUE AND MODE FOR REGION = 2

1.000000 0.000000 0.000000 0.000000
 0.002503 0.002370 0.000000 0.000000
 7.625240 0.000000 0.000000 0.000000
 1332.934249 -1261.588821 0.000000 0.000000
 0.000000 0.000000 0.000000 0.000000
 -385552.250163 338805.724230 0.000000 0.000000
 7.625240 1261.588821 0.000000 0.000000
 -1332.934249 -1.000000 0.000000 0.000000
 -1.000000 0.002503 0.002370 0.000000

16 0.80816893E+03 0.76498261E+03 EIGEN VALUE AND MODE FOR REGION = 2

0.000000 1.000000 0.000000 0.000000
 -0.002370 0.002503 0.000000 0.000000
 -0.000006 7.625240 0.000000 0.000000
 1261.588821 1332.934249 0.000000 0.000000
 0.000000 0.000000 0.000000 0.000000
 -338805.724230 -385552.250163 0.000000 0.000000
 -0.000006 7.625240 0.000000 0.000000
 -1261.588821 -1332.934249 0.000000 0.000000
 0.000000 -1.000000 0.000000 0.000000
 -0.002370 0.002503 0.002370 0.000000

GIVE 0 IF NO POLYNOMIAL DESIRED
 GIVE 0 IF YOU DO NOT WANT POLYNOMIAL APPROX.

NR = 2 NSYM = 2
 GIVE 0 IF YOU WANT REAL ROOTS
 10F10.6...XI..INPUT INITIAL ROOT VALUE
 1 0.25000000E+01 0.56359421E-06 I,XI,FI
 2 0.25038045E+01 -0.26547414E-08 I,XI,FI
 3 0.25037867E+01 0.68529545E-12 I,XI,FI
 4 0.25037867E+01 0.27345368E-15 I,XI,FI
 1 0.25037867E+01 0.27345368E-15 I,XI,FI
 I2...INPUT...J=1 TO STORE ROOT
 10F10.6...XI..INPUT INITIAL ROOT VALUE
 1 0.59000000E+00 0.48992205E-07 I,XI,FI
 2 0.59822567E+00 -0.33901159E-09 I,XI,FI
 3 0.59816954E+00 -0.77504131E-13 I,XI,FI
 4 0.59816953E+00 0.10187731E-16 I,XI,FI
 2 0.59816953E+00 0.10187731E-16 I,XI,FI
 I2...INPUT...J=1 TO STORE ROOT
 10F10.6...XI..INPUT INITIAL ROOT VALUE
 I2...INPUT J=0 TO FIND REAL EIGEN MODES

16 2 1 0.25037867E+01 REAL EIGEN VALUE AND MODE
 0.1000E+01 -.7295E+02 0.3782E-01 -.1265E+03 0.4788E+05 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00

16 2 2 0.59816953E+00 REAL EIGEN VALUE AND MODE
 0.1000E+01 -.2226E+02 0.5873E-03 -.3038E+02 0.1181E+06 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 GIVE 0 IF REAL ROOTS ARE TO BE STORED

17 0.25037867E+01 0.00000000E+00 EIGEN VALUE AND MODE FOR REGION = 2
 1.000000 0.000000 0.000000 0.000000
 -72.950252 0.000000 0.000000 0.000000
 0.037818 0.000000 0.000000 0.000000
 -126.451326 0.000000 0.000000 0.000000
 47877.904548 0.000000 0.000000 0.000000
 0.000000 0.000000 0.000000 0.000000
 -0.037818 0.000000 0.000000 0.000000
 -126.451326 0.000000 0.000000 0.000000
 1.000000 0.000000 0.000000 0.000000
 72.950252 0.000000 0.000000 0.000000

18 0.59816953E+00 0.00000000E+00 EIGEN VALUE AND MODE FOR REGION = 2
 1.000000 0.000000 0.000000 0.000000
 -22.257227 0.000000 0.000000 0.000000
 0.000587 0.000000 0.000000 0.000000
 -30.377292 0.000000 0.000000 0.000000
 118078.602044 0.000000 0.000000 0.000000
 0.000000 0.000000 0.000000 0.000000
 -0.000587 0.000000 0.000000 0.000000
 -30.377292 0.000000 0.000000 0.000000
 1.000000 0.000000 0.000000 0.000000
 22.257227 0.000000 0.000000 0.000000

GIVE 0 IF YOU DO NOT WANT POLYNOMIAL APPROX.

NR = 2 NSYM = 2

GIVE 0 IF YOU WANT COMPLEX ROOTS

2F10.6...XI..INPUT INITIAL COMPLEX ROOT VALUE
 1 0.23200000E+02 0.23300000E+02 -0.76523440E-01 0.10615611E+00 I,XI,FI
 2 0.2302721E+02 0.2332779E+02 -0.12220121E-02 0.48153943E-02 I,XI,FI
 3 0.23021458E+02 0.23331900E+02 0.55482352E-04 -0.36123335E-03 I,XI,FI
 *** TERMINAL ERROR (IER = 129) FROM IMSL ROUTINE LEQTTIC
 *** TERMINAL ERROR (IER = 129) FROM IMSL ROUTINE LEQTTIC
 *** TERMINAL ERROR (IER = 129) FROM IMSL ROUTINE LEQTTIC
 4 0.23021859E+02 0.23331572E+02 0.38146148E-05 0.37013959E-05 I,XI,FI
 *** TERMINAL ERROR (IER = 129) FROM IMSL ROUTINE LEQTTIC
 *** TERMINAL ERROR (IER = 129) FROM IMSL ROUTINE LEQTTIC
 *** TERMINAL ERROR (IER = 129) FROM IMSL ROUTINE LEQTTIC
 5 0.23021859E+02 0.23331580E+02 0.59870490E-06 -0.52050829E-06 I,XI,FI
 *** TERMINAL ERROR (IER = 129) FROM IMSL ROUTINE LEQTTIC
 *** TERMINAL ERROR (IER = 129) FROM IMSL ROUTINE LEQTTIC
 *** TERMINAL ERROR (IER = 129) FROM IMSL ROUTINE LEQTTIC
 6 0.23021860E+02 0.23331580E+02 -0.81667991E-08 -0.25195860E-09 I,XI,FI
 1 0.23021860E+02 0.23331580E+02 -0.81667991E-08 -0.25195860E-09 I,XI,FI
 I2...INPUT...J=1 TO STORE ROOT
 2F10.6...XI..INPUT INITIAL COMPLEX ROOT VALUE

I2...INPUT J=0 TO FIND COMPLEX EIGEN MODES

18 2 1 -0.23021860E+02 0.23331580E+02 COMPLEX EIGEN VALUE AND MODE
0.1000E+01 0.0000E+00 0.1000E+01 0.0000E+00
0.3914E-01 -0.3640E-01 -0.3914E-01 -0.3640E-01
0.7545E+01 0.8589E-02 0.7545E+01 -0.8589E-02
0.2475E+02 0.2607E+02 -0.2475E+02 0.2607E+02
-0.1761E+04 0.1109E+06 -0.1761E+04 -0.1109E+06
0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
GIVE 0 IF COMPLEX ROOTS ARE TO BE STORED

19 0.23021860E+02 0.23331580E+02 EIGEN VALUE AND MODE FOR REGION = 2
1.000000 0.000000 0.000000 0.000000
0.039140 0.036398 0.000000 0.000000
7.544946 -0.008589 0.000000 0.000000
24.748159 -26.073365 0.000000 0.000000
-1760.793992 -110928.639935 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000
-7.544946 0.008589 0.000000 0.000000
24.748159 -26.073365 0.000000 0.000000
1.000000 0.000000 0.000000 0.000000
-0.039140 -0.036398 0.000000 0.000000

20 0.23021860E+02 0.23331580E+02 EIGEN VALUE AND MODE FOR REGION = 2
1.000000 0.000000 0.000000 0.000000
-0.036398 0.039140 0.000000 0.000000
0.008589 7.544946 0.000000 0.000000
26.073365 24.748159 0.000000 0.000000
110928.639935 -1760.793992 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000
-0.008589 -7.544946 0.000000 0.000000
26.073365 24.748159 0.000000 0.000000
0.000000 1.000000 0.000000 0.000000
0.036398 -0.039140 0.000000 0.000000

GIVE 0 IF NO POLYNOMIAL DESIRED
GIVE 0 IF YOU DO NOT WANT POLYNOMIAL APPROX.
NR = 2 NSYM = 3
GIVE 0 IF YOU WANT REAL ROOTS
GIVE 0 IF YOU DO NOT WANT POLYNOMIAL APPROX.
NR = 2 NSYM = 3
GIVE 0 IF YOU WANT COMPLEX ROOTS

GIVE 0 IF NO POLYNOMIAL DESIRED
1 PARTICULAR SOLUTION FOR REGION = 3
0.0000E+00 0.0000E+00

21 ZERO MODE, 3 REGION 0.000000 0.000000
0.000000 1.000000 0.000000 0.000000
22 ZERO MODE, 3 REGION 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000

1.000000 0.000000 0.000000 0.000000

23 ZERO MODE, 3 REGION 0.000000 0.000000 0.000000 1.000000

24 ZERO MODE, 3 REGION 0.000000 0.000000 0.000000 0.000000

25 ZERO MODE, 3 REGION 1.000000 0.000000 0.000000 0.000000

26 ZERO MODE, 3 REGION 0.000000 1.000000 0.000000 0.000000

1 PARTICULAR SOLUTION FOR REGION = 4
 0.0000E+00 0.0000E+00

27 ZERO MODE, 4 REGION 0.000000 0.000000 0.000000 0.000000

28 ZERO MODE, 4 REGION 0.000000 0.000000 0.000000 0.000000

29 ZERO MODE, 4 REGION 0.000000 0.000000 0.000000 1.000000

30 ZERO MODE, 4 REGION 0.000000 0.000000 0.000000 0.000000

31 ZERO MODE, 4 REGION 1.000000 0.000000 0.000000 0.000000

32 ZERO MODE, 4 REGION 0.000000 1.000000 0.000000 0.000000

NO. OF PLYS IN EACH REGION 1 3 1 1
 NO. OF ZERO ROOTS IN EACH REGION 6 6 6 6
 NO. OF REAL ROOTS IN EACH REGION 0 4 0 0
 NO. OF COMP. ROOTS IN EACH REGION 0 4 0 0
 NO. OF B. CONDITIONS IN EACH REGION 6 14 6 6

1 NO. OF VARIABLES IN EACH REGION 2 10 2 2

32 = NTBC

NBCREG	NBCPLY	NBCV	YNBC	ZNBC	PNBC	NBCREG
1	1	2	100000E+010	000000E+000	0000000E+00	0
1	1	3	100000E+010	000000E+000	0000000E+00	0
1	1	6	100000E+010	000000E+000	0000000E+00	0
1	1	10	100000E+010	000000E+000	0000000E+00	2
2	1	1	100000E+010	000000E+000	1000000E+01	0
1	1	20	100000E+010	000000E+000	0000000E+00	2
2	1	2	100000E+01	300000E-01	1000000E+01	0
1	1	30	100000E+010	000000E+000	0000000E+00	2
2	1	3	100000E+010	000000E+000	1000000E+01	0
1	1	40	100000E+010	000000E+000	0000000E+00	2
2	1	4	100000E+01	300000E-01	1000000E+01	0
1	1	50	100000E+010	000000E+000	1200000E+03	2
2	1	5	100000E+010	000000E+000	1000000E+01	0
1	1	60	100000E+010	000000E+000	0000000E+00	2
2	1	6	100000E+01	300000E-01	1000000E+01	0
2	1	10	000000E+000	000000E+000	0000000E+00	3
3	1	10	000000E+000	000000E+000	0000000E+00	0
2	1	20	000000E+000	000000E+000	0000000E+00	3
3	1	20	000000E+000	000000E+000	0000000E+00	0
2	1	30	000000E+000	000000E+000	0000000E+00	3
3	1	30	000000E+000	000000E+000	0000000E+00	0
2	1	40	000000E+000	000000E+000	0000000E+00	3
3	1	40	000000E+000	000000E+000	0000000E+00	0
2	1	50	000000E+000	000000E+000	0000000E+00	3
3	1	50	000000E+000	000000E+000	0000000E+00	0
2	1	60	000000E+000	000000E+000	0000000E+00	3
3	1	60	000000E+000	000000E+000	0000000E+00	0
2	1	100	000000E+000	300000E-010	0000000E+00	3
3	1	100	000000E+000	300000E-010	0000000E+00	0
2	3	100	000000E+000	300000E-010	0000000E+00	4
4	1	100	000000E+000	300000E-010	0000000E+00	0
2	2	10	000000E+000	000000E+000	0000000E+00	0
2	2	30	000000E+000	000000E+000	0000000E+00	0
2	2	50	000000E+000	000000E+000	0000000E+00	0
2	3	10	000000E+000	000000E+000	0000000E+00	4
2	3	10	000000E+000	000000E+000	0000000E+00	0
2	3	20	000000E+000	000000E+000	0000000E+00	4
4	1	20	000000E+000	000000E+000	0000000E+00	0
2	3	30	000000E+000	000000E+000	0000000E+00	4
4	2	30	000000E+000	000000E+000	0000000E+00	0
2	3	40	000000E+000	000000E+000	0000000E+00	4
4	1	40	000000E+000	000000E+000	0000000E+00	0
2	3	50	000000E+000	000000E+000	0000000E+00	4
4	2	50	000000E+000	000000E+000	0000000E+00	0
2	3	60	000000E+000	000000E+000	0000000E+00	4
4	1	60	000000E+000	000000E+000	0000000E+00	0
3	1	30	100000E+010	000000E+000	0000000E+00	0
3	1	50	100000E+010	000000E+000	0000000E+00	0
3	1	10	100000E+010	000000E+000	0000000E+00	0
4	1	10	100000E+010	000000E+000	0000000E+00	0
4	1	30	100000E+010	000000E+000	0000000E+00	0
4	1	60	100000E+010	000000E+000	0000000E+00	0

-0.743750	749.801436	0.000000	2	2	8
-0.750000	749.797690	0.000000	2	2	8
-0.756250	749.794330	0.000000	2	2	8
-0.762500	749.791315	0.000000	2	2	8
-0.768750	749.788610	0.000000	2	2	8
-0.775000	749.786183	0.000000	2	2	8
-0.781250	749.784005	0.000000	2	2	8
-0.787500	749.782052	0.000000	2	2	8
-0.793750	749.780299	0.000000	2	2	8
-0.800000	749.778726	0.000000	2	2	8
-0.806250	749.777315	0.000000	2	2	8
-0.812500	749.776049	0.000000	2	2	8
-0.818750	749.774913	0.000000	2	2	8
-0.825000	749.773894	0.000000	2	2	8
-0.831250	749.772980	0.000000	2	2	8
-0.837500	749.772160	0.000000	2	2	8
-0.843750	749.771424	0.000000	2	2	8
-0.850000	749.770764	0.000000	2	2	8
-0.856250	749.770171	0.000000	2	2	8
-0.862500	749.769640	0.000000	2	2	8
-0.868750	749.769163	0.000000	2	2	8
-0.875000	749.768735	0.000000	2	2	8
-0.881250	749.768351	0.000000	2	2	8
-0.887500	749.768007	0.000000	2	2	8
-0.893750	749.767698	0.000000	2	2	8
-0.900000	749.767420	0.000000	2	2	8
-0.906250	749.767172	0.000000	2	2	8
-0.912500	749.766948	0.000000	2	2	8
-0.918750	749.766748	0.000000	2	2	8
-0.925000	749.766569	0.000000	2	2	8
-0.931250	749.766407	0.000000	2	2	8
-0.937500	749.766263	0.000000	2	2	8
-0.943750	749.766133	0.000000	2	2	8
-0.950000	749.766017	0.000000	2	2	8
-0.956250	749.765912	0.000000	2	2	8
-0.962500	749.765818	0.000000	2	2	8
-0.968750	749.765734	0.000000	2	2	8
-0.975000	749.765659	0.000000	2	2	8
-0.981250	749.765591	0.000000	2	2	8
-0.987500	749.765531	0.000000	2	2	8
-0.993750	749.765476	0.000000	2	2	8
-1.000000	749.765427	0.000000	2	2	8
Y, F, Z, NR, MPLX, MV					
0.000000	-24253.389799	0.000000	2	2	9
-0.006250	-8819.215527	0.000000	2	2	9
-0.012500	-6727.635173	0.000000	2	2	9
-0.018750	-4590.680653	0.000000	2	2	9
-0.025000	-2976.740149	0.000000	2	2	9
-0.031250	-1771.874917	0.000000	2	2	9
-0.037500	-874.399832	0.000000	2	2	9
-0.043750	-212.588805	0.000000	2	2	9
-0.050000	269.486656	0.000000	2	2	9
-0.056250	614.757490	0.000000	2	2	9

-0.062500	856.201076	0.000000	2	2	9
-0.068750	1019.131162	0.000000	2	2	9
-0.075000	1122.962117	0.000000	2	2	9
-0.081250	1182.568348	0.000000	2	2	9
-0.087500	1209.331329	0.000000	2	2	9
-0.093750	1211.946039	0.000000	2	2	9
-0.100000	1197.042016	0.000000	2	2	9
-0.106250	1169.661612	0.000000	2	2	9
-0.112500	1133.628231	0.000000	2	2	9
-0.118750	1091.829804	0.000000	2	2	9
-0.125000	1046.436994	0.000000	2	2	9
-0.131250	999.071100	0.000000	2	2	9
-0.137500	950.933232	0.000000	2	2	9
-0.143750	902.903648	0.000000	2	2	9
-0.150000	855.618126	0.000000	2	2	9
-0.156250	809.526632	0.000000	2	2	9
-0.162500	764.938377	0.000000	2	2	9
-0.168750	722.056386	0.000000	2	2	9
-0.175000	681.003994	0.000000	2	2	9
-0.181250	641.845131	0.000000	2	2	9
-0.187500	604.599841	0.000000	2	2	9
-0.193750	569.256110	0.000000	2	2	9
-0.200000	535.778888	0.000000	2	2	9
-0.206250	504.116930	0.000000	2	2	9
-0.212500	474.207978	0.000000	2	2	9
-0.218750	445.982665	0.000000	2	2	9
-0.225000	419.367435	0.000000	2	2	9
-0.231250	394.286722	0.000000	2	2	9
-0.237500	370.664553	0.000000	2	2	9
-0.243750	348.425708	0.000000	2	2	9
-0.250000	327.496561	0.000000	2	2	9
-0.256250	307.805658	0.000000	2	2	9
-0.262500	289.284104	0.000000	2	2	9
-0.268750	271.865818	0.000000	2	2	9
-0.275000	255.487668	0.000000	2	2	9
-0.281250	240.089533	0.000000	2	2	9
-0.287500	225.614304	0.000000	2	2	9
-0.293750	212.007846	0.000000	2	2	9
-0.300000	199.218928	0.000000	2	2	9
-0.306250	187.199126	0.000000	2	2	9
-0.312500	175.902726	0.000000	2	2	9
-0.318750	165.286607	0.000000	2	2	9
-0.325000	155.310120	0.000000	2	2	9
-0.331250	145.934975	0.000000	2	2	9
-0.337500	137.125114	0.000000	2	2	9
-0.343750	128.846599	0.000000	2	2	9
-0.350000	121.067496	0.000000	2	2	9
-0.356250	113.757765	0.000000	2	2	9
-0.362500	106.889151	0.000000	2	2	9
-0.368750	100.435086	0.000000	2	2	9
-0.375000	94.370590	0.000000	2	2	9
-0.381250	88.672180	0.000000	2	2	9
-0.387500	83.317779	0.000000	2	2	9
-0.393750	78.286639	0.000000	2	2	9

REFERENCES

1. Murthy, P.L.N., and Chamis, C.C., "Interlaminar Fracture Toughness: Three-Dimensional Finite-Element Modeling for End-Notch and Mixed-Mode Flexure," NASA TM-87138, National Aeronautics and Space Administration, Washington, D.C., 1985.
2. Pagano, N.J., International Journal of Solids and Structures, Vol. 14, No. 5, 1978, pp. 385-400.
3. Raju, I.S., Whitcomb, J.D., and Goree, J.G., "A New Look at the Numerical Analyses of Free-Edge Stresses in Composite Laminates," NASA TP-1751, National Aeronautics and Space Administration, Washington, D.C., 1980.
4. Pagano, N.J. and Soni, S.R., International Journal of Solids and Structures, Vol. 19, No. 3, 1983, pp. 207-228.
5. Salamon, N.J., Journal of Composite Materials Supplement, Vol. 14, No. 1, 1980, pp. 177-194.
6. Valisetty, R.R. and Rehfield, L.W., in Delamination and Debonding of Materials, ASTM STP-876, W.S. Johnson, ED., American Society for Testing and Materials, Philadelphia, 1985, pp. 52-68.
7. Rehfield, L.W., Erian, A.A., and Valisetty, R.R., Computers and Structures, Vol. 20, No. 1-3, 1985, pp. 401-411.
8. Valisetty, R.R., "Bending of Beams, Plates, and Laminates: Refined Theories and Comparative Studies," Ph.D. Thesis, Georgia Institute of Technology, Atlanta, GA., 1983.

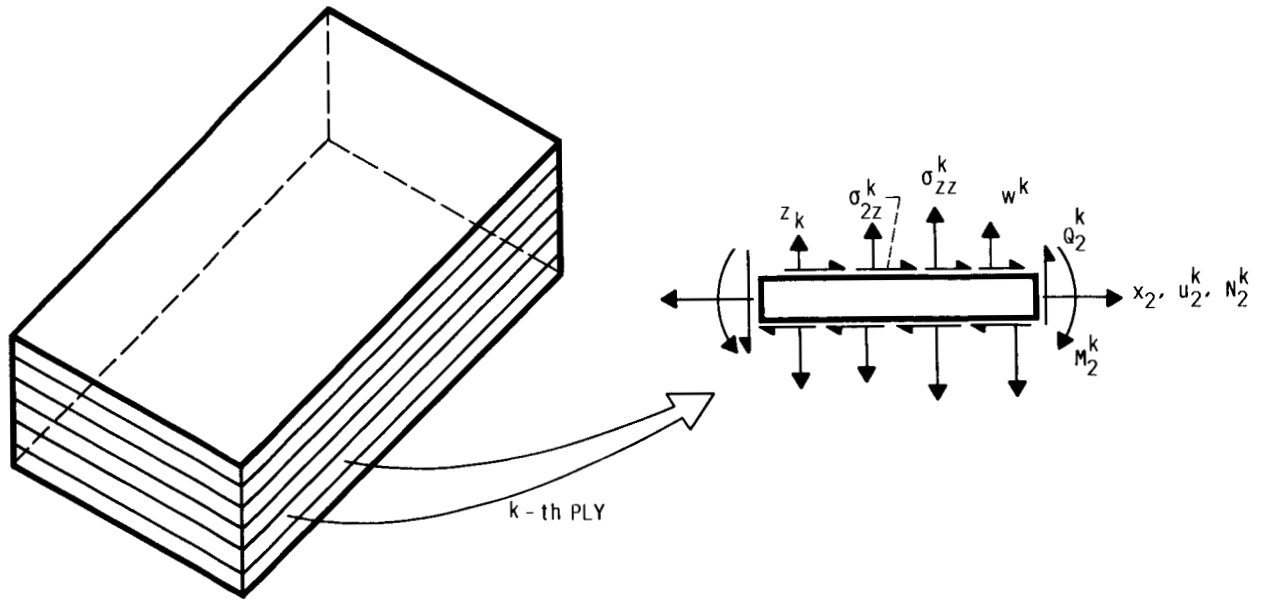


FIGURE 1. - EQUILIBRIUM OF A PLY UNDER INTERLAMINAR STRESSES AND EDGE FORCES.

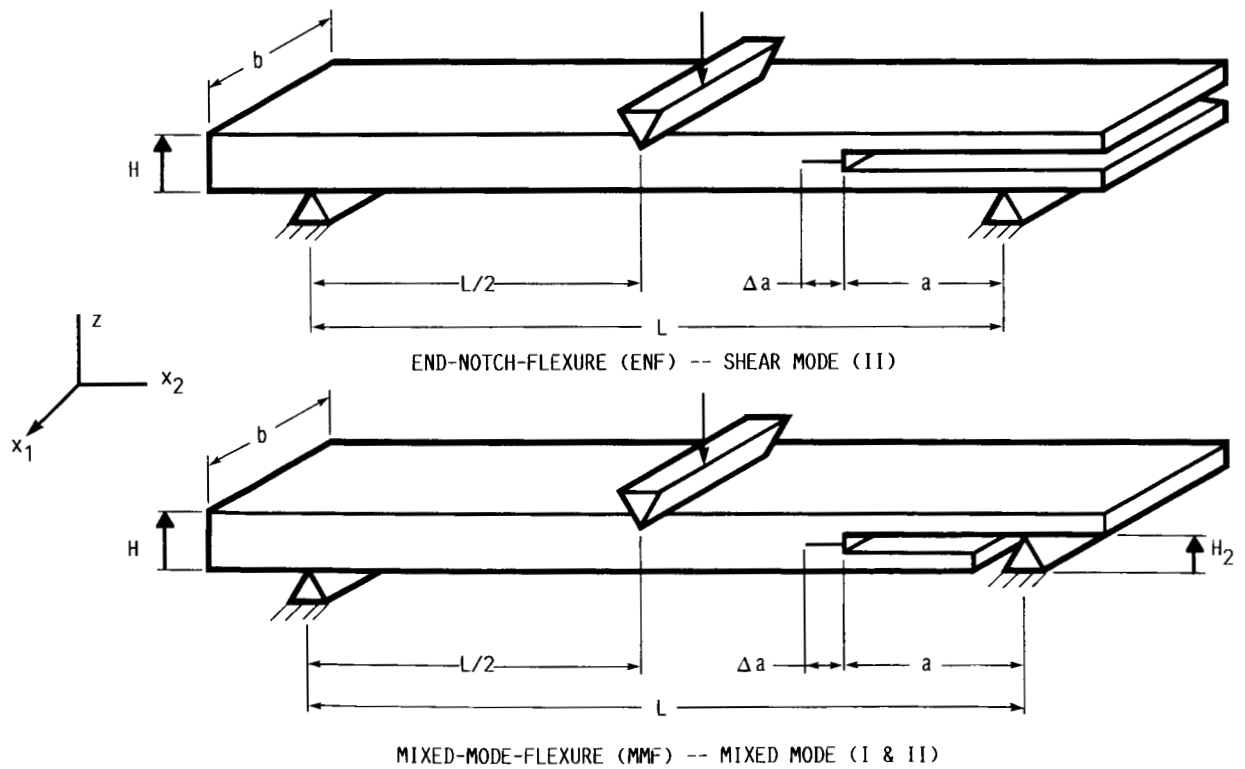


FIGURE 2. - SCHEMATIC OF FLEXURAL TEST FOR INTERLAMINAR FRACTURE MODE TOUGHNESS.

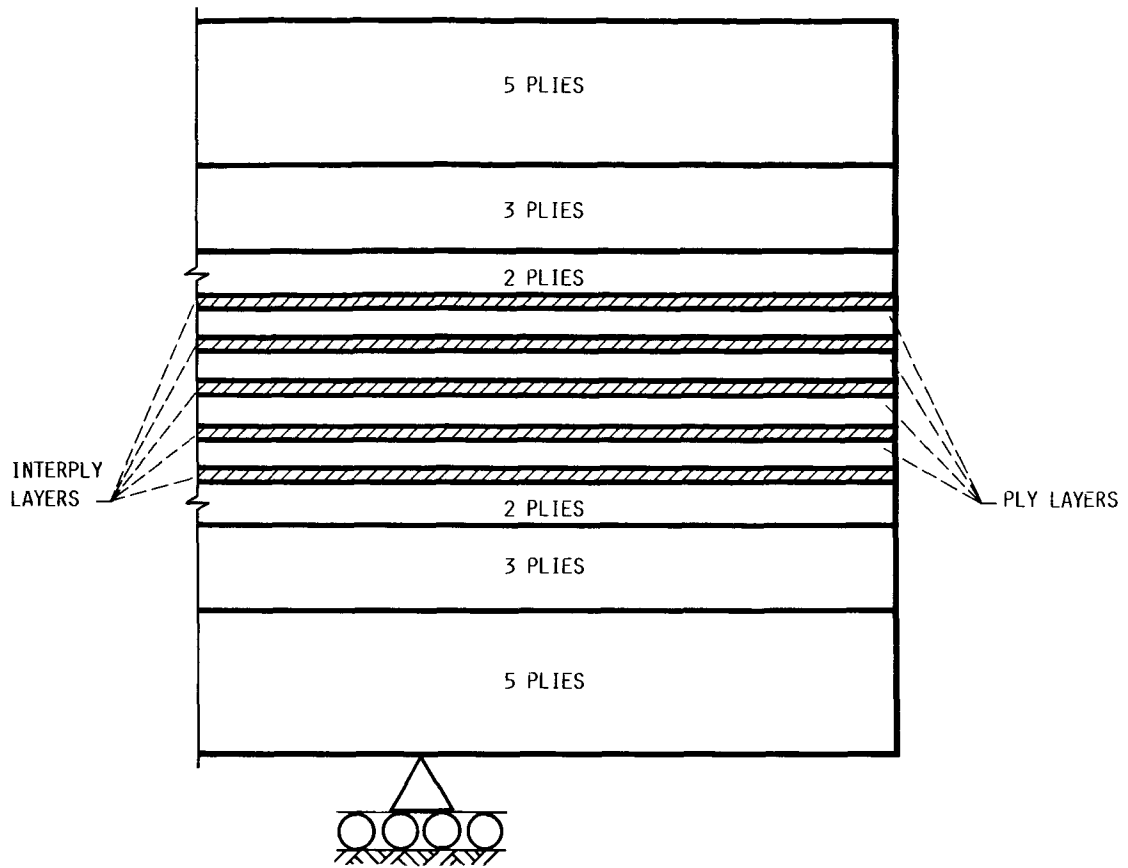


FIGURE 3. - PLY STACKING SEQUENCE.

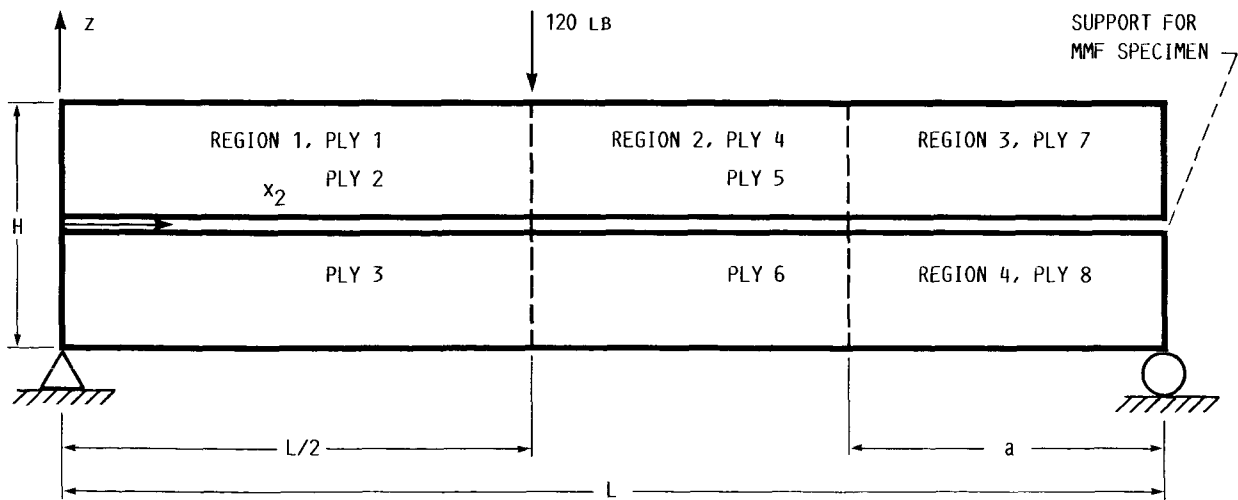


FIGURE 4. - IDEALIZATION OF THE ENF SPECIMEN FOR SUBLAMINATE ANALYSIS.

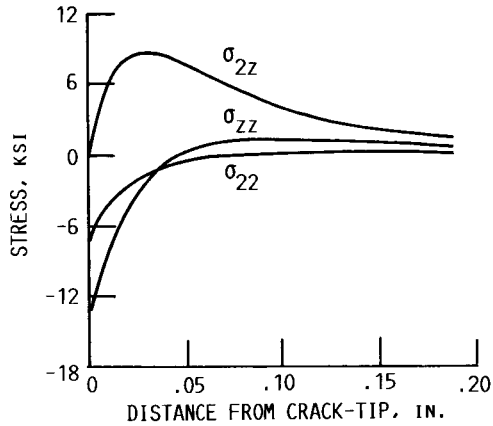


FIGURE 5. - STRESS DISTRIBUTION IN THE INTERPLY LAYER AHEAD OF THE CRACK-TIP FOR THE ENF SPECIMEN, CRACK LENGTH, $a = 1$ IN.

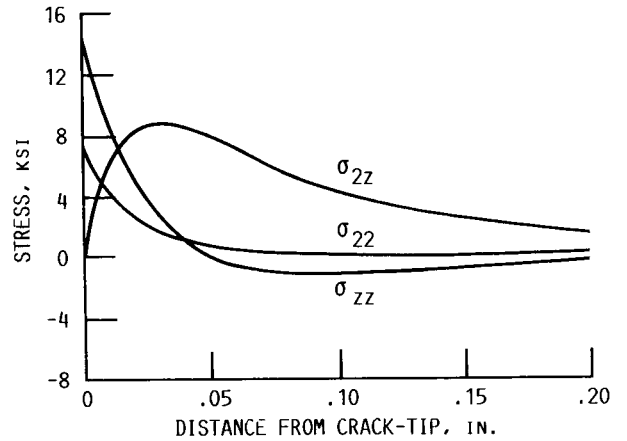


FIGURE 6. - STRESS DISTRIBUTION IN THE INTERPLY LAYER AHEAD OF THE CRACK-TIP FOR THE MMF SPECIMEN, CRACK LENGTH, $a = 1$ IN.

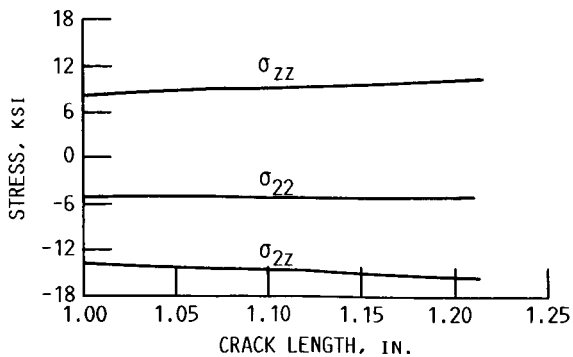


FIGURE 7. - PEAK STRESS BEHAVIOR IN INTERPLY LAYER FOR THE ENF SPECIMEN.

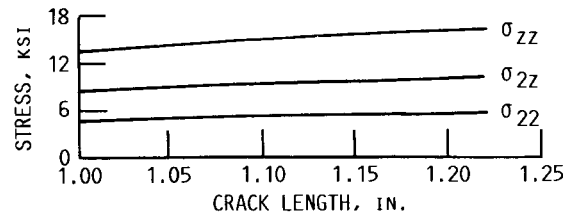


FIGURE 8. - PEAK STRESS BEHAVIOR IN INTERPLY LAYER FOR THE MMF SPECIMEN.

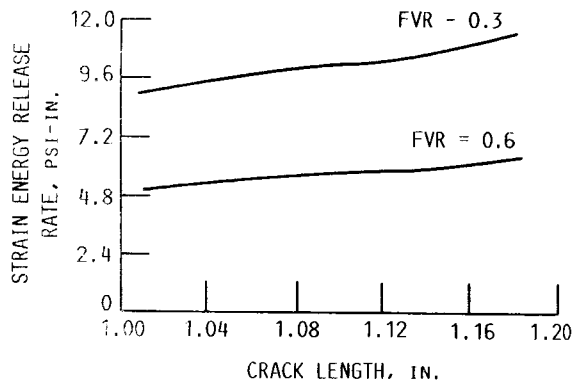


FIGURE 9. - TOTAL STRAIN ENERGY RELEASE RATES FOR ENF AND MMF SPECIMENS.

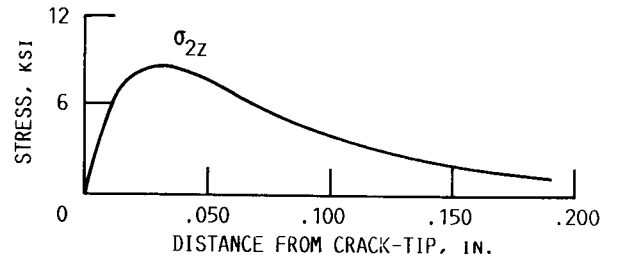


FIGURE 10. - STRESS FIELD IN INTERPLY LAYER AHEAD OF CRACK-TIP FOR ENF-II IDEALIZATION.

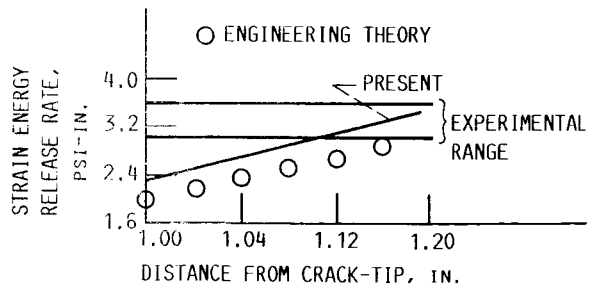


FIGURE 11. - TOTAL STRAIN ENERGY RELEASE RATE FOR ENF-II IDEALIZATION.

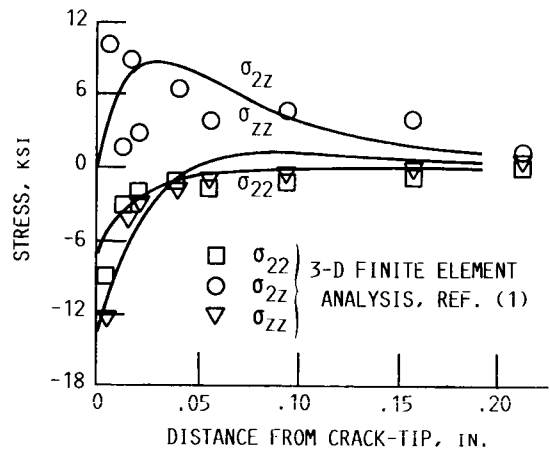


FIGURE 12. - STRESS DISTRIBUTION IN THE INTERPLY LAYER AHEAD OF THE CRACK-TIP FOR THE ENF SPECIMEN, CRACK LENGTH, $a = 1$ IN.

1. Report No. NASA TM-89827		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle Strain Energy Release Rates of Composite Interlaminar End-Notch and Mixed-Mode Fracture: A Sublaminar/Ply Level Analysis and a Computer Code				5. Report Date April 1987	
				6. Performing Organization Code 505-63-11	
7. Author(s) R.R. Valisetty and C.C. Chamis				8. Performing Organization Report No. E-3476	
				10. Work Unit No.	
9. Performing Organization Name and Address National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135				11. Contract or Grant No.	
				13. Type of Report and Period Covered Technical Memorandum	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546				14. Sponsoring Agency Code	
15. Supplementary Notes Prepared for the 8th Symposium on Composite Materials Testing and Design, sponsored by the American Society for Testing and Materials, Charleston, South Carolina, April 29 - May 1, 1986. R.R. Valisetty, National Research Council - NASA Research Associate; C.C. Chamis, NASA Lewis Research Center.					
16. Abstract A computer code is presented for the sublaminar/ply level analysis of composite structures. This code is useful for obtaining stresses in regions affected by delaminations, transverse cracks, and discontinuities related to inherent fabrication anomalies, geometric configurations, and loading conditions. Particular attention is focussed on those layers or groups of layers (sublaminates) which are immediately affected by the inherent flaws. These layers are analyzed as homogeneous bodies in equilibrium and in isolation from the rest of the laminate. The theoretical model used to analyze the individual layers allows the relevant stresses and displacements near discontinuities to be represented in the form of pure exponential-decay-type functions which are selected to eliminate the exponential-precision-related difficulties in sublaminar/ply level analysis. Thus, sublaminar analysis can be conducted without any restriction on the maximum number of layers, delaminations, transverse cracks, or other types of discontinuities. In conjunction with the strain energy release rate (SERR) concept and composite micromechanics, this computational procedure is used to model select cases of end-notch and mixed-mode fracture specimens. The computed stresses are in good agreement with those from a three-dimensional finite element analysis. Also SERRs compare well with limited available experimental data.					
17. Key Words (Suggested by Author(s)) Sublaminar analysis; Ply level analysis; Composite laminate analysis; Strain energy release rate; Interlaminar delamination; Mode I fracture; Mode II fracture; Interply layer; Fracture toughness; Graphite fiber; Epoxy matrix; Crack closure; Interlaminar fracture; Interply layer; Mixed mode fracture; Graphite/epoxy laminates			18. Distribution Statement Unclassified - unlimited STAR Category 24		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of pages 86	22. Price* A05