UNCLASSIFIED

AD NUMBER
AD862147
LIMITATION CHANGES
TO: **Approved for public release; distribution is unlimited.**
FROM: **Distribution authorized to U.S. Gov't. agencies and their contractors; Critical Technology; JUL 1969. Other requests shall be referred to U.S. Army Aviation Materiel Laboratories , Fort Eustis , Virginia 23604. This document contains export-controlled technical data.**
AUTHORITY
USAAMRDL ltr, 23 Jun 1971

THIS PAGE IS UNCLASSIFIED

AD 862147

AD

USAAVLABS TECHNICAL REPORT 69-45

TRANSVERSE LOADING OF UNIDIRECTIONAL FIBER COMPOSITES

By

Juan Haener

Alberto Puppo

Ming-yuan Feng

July 1969

U. S. ARMY AVIATION MATERIEL LABORATORIES
FORT EUSTIS, VIRGINIA

CONTRACT DAAJ02-68-C-0037

WHITTAKER RESEARCH & DEVELOPMENT/SAN DIEGO

SAN DIEGO, CALIFORNIA

U S ARMY MATERIEL COMMAND

This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of US Army Aviation Materiel Laboratories, Fort Eustis, Virginia 23604.

131

DISCLAIMERS

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission, to manufacture, use, or sell any patented invention that may in any way be related thereto.

DISPOSITION INSTRUCTIONS

Destroy this report when no longer needed. Do not return it to the originator.

DEPARTMENT OF THE ARMY
HEADQUARTERS US ARMY AVIATION MATERIEL LABORATORIES
FORT EUSTIS, VIRGINIA 23604

This program was carried out under Contract DAAJ02-68-C-0037 with Whittaker Research and Development/San Diego.

The data contained in this report are the result of research conducted to find the internal micromechanics of a fiber-reinforced composite under transverse normal loading. The numerical method of finite elements was used to obtain stress distributions.

The report has been reviewed by the U.S. Army Aviation Materiel Laboratories and is considered to be technically sound. It is published for the exchange of information and the stimulation of future research.

Task 1F162204A17002
Contract DAAJ02-68-C-0037
USAAVLABS Technical Report 69-45
July 1969

TRANSVERSE LOADING OF UNIDIRECTIONAL FIBER COMPOSITES

FINAL REPORT

By

Juan Haener

Alberto Puppo

Ming-yuan Feng

Prepared by

Whittaker Research & Development/San Diego
San Diego, California

for

U. S. ARMY AVIATION MATERIEL LABORATORIES
FORT EUSTIS, VIRGINIA

This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of US Army Aviation Materiel Laboratories, Fort Eustis, Virginia 23604.

SUMMARY

The study is a part of an effort being directed toward solving the problem of a composite material subjected to general oblique loading.

This analysis was conducted in order to find the internal micromechanics of a fiber-reinforced composite due to transverse normal loading. Special emphasis has been given to studying the stress distribution near free surfaces, which led to solving a three-dimensional elasticity problem. The numerical method of finite elements has been employed in this analysis. On the other hand, it was necessary to study the behavior far from free surfaces. For this purpose, a two-dimensional program was used. Findings from these two approaches were anticipated, showing that stress conditions become two dimensional a relatively short distance from the end of the composite. Extensive parametric studies have been performed from the combined outputs of these two schemes. Significant diagrams exhibiting elastic properties of different composite materials have been obtained.

FOREWORD

This final report was prepared by Whittaker Research & Development/ San Diego, under U. S. Army Contract DAAJ02-68-C-0037, (Task 1F162204A17002), for the U. S. Army Aviation Materiel Laboratories, Fort Eustis, Virginia. Mr. A. Gustafson was the Army project officer for this program.

This report covers the work accomplished during the period from 26 December 1967 through 26 December 1968.

Special acknowledgement is given to Dr. Gerhard Nowak for his invaluable assistance in the development of the three-dimensional solution and algorithm of the numerical scheme and to Mr. Ted Neff, who performed the computer programming.

BLANK PAGE

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

LIST OF SYMBOLS

a	fiber radius
a_i	(i = 1...12 constants), defined in equations (1) through (3)
$[A]$	matrix defined in equation (5)
$[A]$, $[B]$, $[C]$	submatrices of a row of the partition
A_m, B_m, C_m, D_m	constants
A', B , C', D', E', F'	represent the corner points of the hexagon
b	half of the distance between two neighboring fibers
c	half of the dimension in x-direction of a basic element
C	elastic stiffness matrix
d	fiber diameter
d_z	height of tetrahedron
D	stiffness matrix
E	modulus of elasticity
E_f	fiber modulus of elasticity
E_m	matrix modulus of elasticity
E_T	transverse modulus of elasticity of a composite
G	shear modulus
i	index number
I	lattice point number
IM(1,1)	defined by equation (48)
IM(2,1)	defined by equations (49) & (52)
IM(3,1)	defined by equations (50) & (53)
IV	matrix indicating nodal points sequences of tetrahedrons
IKO	matrix characterizing points of central symmetry
I_M, I_E, I_B, I_D, I_H	Defined on Figure 7
j	index number
J	lattice point number (equation (46))

LIST OF SYMBOLS (Continued)

k	displacement in x-direction at $x = c$ or defined on Page 23
K^*	total stiffness matrix
K_{ij}	stiffness matrix
$K_1, K_3, K_4, K_5, K_6, K_7,$	constants
K_8, K_9, K_{10}, K_{11}	constants
$\bar{K}_1, \bar{K}_3, \bar{K}_6, \bar{K}_8, \bar{K}_{11}$	constants
ℓ	length of the basic element of a three-dimensional problem
M	total number of force components or represents the center of the fiber
MSK	matrix of unknown displacements in a consecutive order
M_1	number of unknown force components or number of known displacement components
M_1, M_2, P_1, P_2	represent the points in Figure 2
n	$= 3\,(i-1)+j$
$N_1, N_2 N_3, N_4$	node points
p	number of tetrahedrons or number of floors
P	represents a point in the tetrahedron or force
$\{P^*\}$	column vector of the nodal forces for the total system
$P_i^{(k)}$	known force components
$P_i^{(u)}$	unknown force components
q	represents the direction of component at node point
Q	lattice point number (equation (47))
t	represents tetrahedrons
u	displacement vector
$u_j^{(u)}$	unknown displacement components
$u_j^{(k)}$	known displacement components

LIST OF SYMBOLS (Continued)

u_x, u_y, u_z	displacement vextor components in x, y, z directions
$\{u^*\}$	column vector of the nodal displacements for the total system
$\{u^{(t_j)}\}$	column vector of the nodal displacement corresponding to the t_j^{th} tetrahedron
V	volume of tetrahedron
V_f, V_m	fiber and matrix volumetric content
W	strain energy
x, y, z	coordinates (see Figure 4)
α, β, ν	represents topological
α', β', γ'	properties of lattice point of tetrahedrons
$\supset(I) = +1, -1$	or 0 characterizing topological α, β, ν points respectively
ϵ	strain matrix
λ	L'ame constant
ν	Poisson's ratio
ν_f	Poisson's ratio of fiber
ν_m	Poisson's ratio of matrix
θ	Airy's stress function
σ	stress matrix
ξ, η, ζ	local coordinates at nodal point of tetrahedron
α	angle between surface normal and x-coordinates

INTRODUCTION

Only in the ideal case is the loading of a reinforced composite material in the directions of the reinforcements. Normally, oblique loading is encountered by the composite. To obtain the effect of oblique loading, we must combine axial, shear, and transverse loadings. By doing so, we can establish the stresses in the reinforcements and in the matrix as they actually occur; we can then use these stresses for the development of a strength theory based on such reliable data.

This report presents the case of transverse loading of a unidirectional composite. The two different regions considered are (1) the region close to the end of the fibers, where three-dimensional treatment is applied, and (2) regions far from the end of the fibers, where the end perturbations are not more effective and where a plane solution is applied.

In both cases, the finite element method was applied successfully. For the plane problem, we also intended to find a solution by using the so-called point-matching or collocation method which was applied in the solution of the longitudinal shear problem. However, the convergence of this method was very poor for this case of transverse loading. Appendix I gives the equations utilized, along with a description of the form in which the point-matching method was used.

The plane problem was solved in both plane stress and plane strain conditions by using the finite element method. Displacements, stresses, and transverse moduli for different types of composites are given in this report.

THREE-DIMENSIONAL SOLUTION

The numerical approach used to evaluate a specified three-dimensional boundary problem of linear elasticity represents a first-order approximation method. This method is known as the method of finite elements. Its concept is based on the assumption that the stress within small-volume elements of the body is constant. This is precisely true if we consider infinitesimal volume elements. For small-volume elements, this assumption of constant stress proves to be a good model representation of reality.

It is convenient to consider small tetrahedrons as volume elements because three-dimensional space can be subdivided into sets of tetrahedrons in a simple way. The assumption of constant stress within each elementary tetrahedron is equivalent to the assumption of linear displacement-vector-distibution within the elementary tetrahedron.

Compatibility and equilibrium conditions introduced at the tetrahedron node points lead to a system of linear algebraic equations whose solutions represent displacements at the node points of the tetrahedrons. From these displacements, we can determine the stresses within each tetrahedron.

THE STIFFNESS MATRIX OF A TETRAHEDRON ELEMENT

In order to obtain the proper working equations, we have to consider an elementary tetrahedron first and find the pertinent relations as far as stress, nodal forces, and nodal displacements are concerned.

Figure 1 shows a general elementary tetrahedron with the nodes N_1, N_2, N_3, and N_4. We assume that the three coordinates ξ_i, η_i, and ζ_i $(i = 1,2,3,4)$ are known for each node point N_i $(i = 1,2,3,4)$.

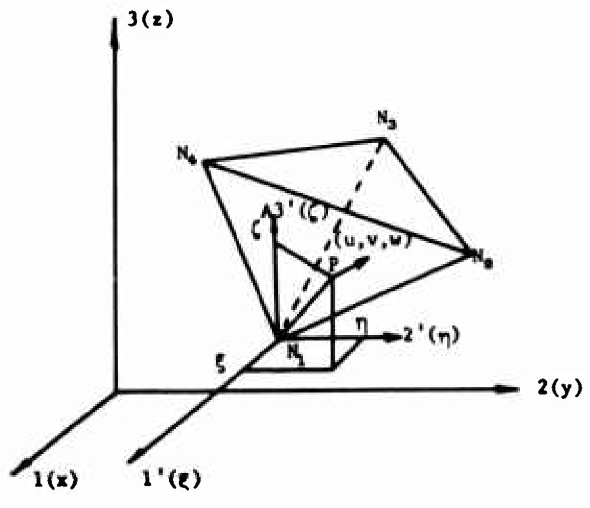


Figure 1. The Elementary Tetrahedron.

Again, we can represent the displacement vector by a linear vector function of the local coordinates ξ , η , and ζ . The origin of the local coordinate system coincides with node N_1 . In this fashion, we put the three displacement components at point $P(\xi,\eta,\zeta)$, as follows:

$$u_x = a_1 + a_2\,\xi + a_3\,\eta + a_4\,\zeta \tag{1}$$

$$u_y = a_5 + a_6\,\xi + a_7\,\eta + a_8\,\zeta \tag{2}$$

$$u_z = a_9 + a_{10}\,\xi + a_{11}\,\eta + a_{12}\,\zeta \tag{3}$$

The coefficients $a_1,\ldots,a_{12}$ are constants and subject to variance with the geometric configuration of the tetrahedron as well as the displacement configuration at the nodes. Let u_{x_i} , u_{y_i} , and u_{z_i} $(i = 1,2,3,4)$ represent displacement components of the four node points. Then, we obtain the following 12 relations from equations (1), (2), and (3):

$$
\begin{aligned}
u_{x_1} &= a_1 + a_2\,\xi_1 + a_3\,\eta_1 + a_4\,\zeta_1 \\
u_{x_2} &= a_1 + a_2\,\xi_2 + a_3\,\eta_2 + a_4\,\zeta_2 \\
u_{x_3} &= a_1 + a_2\,\xi_3 + a_3\,\eta_3 + a_4\,\zeta_3 \\
u_{x_4} &= a_1 + a_2\,\xi_4 + a_4\,\eta_4 + a_4\,\zeta_4 \\
u_{y_1} &= a_5 + a_6\,\xi_1 + a_7\,\eta_1 + a_8\,\zeta_1 \\
u_{y_2} &= a_5 + a_6\,\xi_2 + a_7\,\eta_2 + a_8\,\zeta_2 \\
u_{y_3} &= a_5 + a_6\,\xi_3 + a_7\,\eta_3 + a_8\,\zeta_3 \\
u_{y_4} &= a_5 + a_6\,\xi_4 + a_7\,\eta_4 + a_8\,\zeta_4 \\
u_{z_1} &= a_9 + a_{10}\,\xi_1 + a_{11}\,\eta_1 + a_{12}\,\zeta_1 \\
u_{z_2} &= a_9 + a_{10}\,\xi_2 + a_{11}\,\eta_2 + a_{12}\,\zeta_2 \\
u_{z_3} &= a_9 + a_{10}\,\xi_3 + a_{11}\,\eta_3 + a_{12}\,\zeta_3 \\
u_{z_4} &= a_9 + a_{10}\,\xi_4 + a_{11}\,\eta_4 + a_{12}\,\zeta_4
\end{aligned}
\tag{4}
$$

In matrix form, relations (4) can be expressed as follows:

$$
\begin{Bmatrix} u_{x_1} \\ u_{x_2} \\ u_{x_3} \\ u_{x_4} \\ u_{y_1} \\ u_{y_2} \\ u_{y_3} \\ u_{y_4} \\ u_{z_1} \\ u_{z_2} \\ u_{z_3} \\ u_{z_4} \end{Bmatrix}
=
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & \xi_2 & \eta_2 & \zeta_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & \xi_3 & \eta_3 & \zeta_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & \xi_4 & \eta_4 & \zeta_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & \xi_2 & \eta_2 & \zeta_2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & \xi_3 & \eta_3 & \zeta_3 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & \xi_4 & \eta_4 & \zeta_4 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \xi_2 & \eta_2 & \zeta_2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \xi_3 & \eta_3 & \zeta_3 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \xi_4 & \eta_4 & \zeta_4
\end{bmatrix}
\begin{Bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ a_{10} \\ a_{11} \\ a_{12} \end{Bmatrix}
\qquad (5)
$$

Equation (5) can be written symbolically, such as

$$\{u\} = [A] \{a\} \qquad (6)$$

where $\{u\}$ are the 12 displacements at the four nodes, $\{a\}$ the $\{12 \times 1\}$ matrix of the configuration coefficients $a_1, \ldots, a_{12}$, and $[A]$ the $\{12 \times 12\}$ matrix as demonstrated in equation (5).

The six strain components ϵ_x, ϵ_y, ϵ_z, ϵ_{xy}, ϵ_{xz}, and ϵ_{yz} are obtained immediately from equations (1), (2), and (3), as follows:

$$\epsilon_{xx} = \frac{\partial u_x}{\partial \xi} = a_2$$

$$\epsilon_{yy} = \frac{\partial u_y}{\partial \eta} = a_7$$

$$\epsilon_{zz} = \frac{\partial u_z}{\partial \zeta} = a_{12}$$

$$\epsilon_{xy} = \frac{\partial u_x}{\partial \eta} + \frac{\partial u_y}{\partial \xi} = a_3 + a_6$$

$$\epsilon_{xz} = \frac{\partial u_x}{\partial \zeta} + \frac{\partial u_z}{\partial \xi} = a_4 + a_{10}$$

$$\epsilon_{yz} = \frac{\partial u_y}{\partial \zeta} + \frac{\partial u_z}{\partial \eta} = a_8 + a_{11} \qquad (7)$$

We denote $\{\varepsilon\}$ as the $\{6 \times 1\}$ matrix with the elements ϵ_x, ϵ_y, ϵ_z, ϵ_{xy}, ϵ_{xz}, and ϵ_{yz} and write equation (7) symbolically, as follows:

$$\begin{Bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ \epsilon_{xy} \\ \epsilon_{xz} \\ \epsilon_{yz} \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ a_{10} \\ a_{11} \\ a_{12} \end{Bmatrix} \qquad (8)$$

If we use the short form symbol $[D]$ for the $\{6 \times 12\}$ matrix in equation (8), we can write equation (8) in the symbolic short form:

$$\{\epsilon\} = [D]\{a\} \qquad (9)$$

Finally, we can obtain the six stress tensor components σ_x, σ_y, σ_z, σ_{xy}, σ_{xz}, σ_{yz} from Hooke's law:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \sigma_{xy} \\ \sigma_{xz} \\ \sigma_{yz} \end{Bmatrix} = \begin{bmatrix} \lambda+2G & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda+2G & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & \lambda+2G & 0 & 0 & 0 \\ 0 & 0 & 0 & G & 0 & 0 \\ 0 & 0 & 0 & 0 & G & 0 \\ 0 & 0 & 0 & 0 & 0 & G \end{bmatrix} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \epsilon_z \\ \epsilon_{xy} \\ \epsilon_{xz} \\ \epsilon_{yz} \end{Bmatrix} \qquad (10)$$

with the meanings of λ and G ,

$$G = \frac{E}{2(1+\nu)} \tag{11}$$

and

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)} \tag{12}$$

where E is the modulus of elasticity and ν is Poisson's ratio.

The symbol $[C]$ stands for the $\{6 \times 6\}$ matrix of equation (10), which is the elasticity matrix. Equation (10) therefore is, in short symbolic form,

$$\{\sigma\} = [C] \{\epsilon\} \tag{13}$$

Substitution of $\{\epsilon\}$ by means of equation (9) follows:

$$\{\sigma\} = [C][D] \{a\} \tag{14}$$

From equation (6), we can express $\{a\}$ in terms of

$$\{a\} = [A]^{-1} \{u\} \tag{15}$$

where $[A]^{-1}$ represents the inverse matrix of $[A]$. Therefore, equation (14) is

$$\{\sigma\} = [C][D][A]^{-1} \{u\} \tag{16}$$

Expression (16) is the working equation for calculating the stress tensor $\{\sigma\}$. In order to do this, it is necessary to know all 12 displacement components at the 4 nodes. In general, whether or not each one of them is given input quantities depends on the kind of boundary conditions present. As a matter of fact, we are in general not free to choose a displacement component arbitrarily when the corresponding force component has been fixed. Castigliano's theorem expresses this proposition clearly, stating that in any linear elastic system, the force component acting on the system in a certain direction is equal to the first derivative of the strain energy function with respect to the displacement in the same direction. If we define W as the strain energy function of the system, we can obtain a set of equations which relates the node force component array to the displacement components, as follows:

$$\{P\} = \frac{\partial W}{\partial \{u\}} \tag{17}$$

W proves to be a quadratic form in $\{u\}$, and therefore it is the right-hand side of a linear function in $\{u\}$ representing the desired relationship between forces and displacements. Next, we have to obtain the strain energy function W .

The strain energy of an elastic body is the volume integral over the elastic potential, such as

$$W = \frac{1}{2} \int_V \left(\sigma_x \epsilon_x + \sigma_y \epsilon_y + \sigma_z \epsilon_z + \sigma_{xy} \epsilon_{xy} + \sigma_{xz} \epsilon_{xz} + \sigma_{yz} \epsilon_{yz} \right) dV \tag{18}$$

The integrand can be written symbolically $\{\epsilon\}^T \{\sigma\}$, and since ϵ and σ are constant within the region of integration, we get

$$W = \frac{1}{2} \{\epsilon\}^T \{\sigma\} \int_V dV = \frac{1}{2} \{\epsilon\}^T \{\sigma\} V \tag{19}$$

where

$$V = \frac{1}{6} \left\| \begin{vmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ x_4 - x_1 & y_4 - y_1 & z_4 - z_1 \end{vmatrix} \right\| = \frac{1}{6} \left\| \begin{vmatrix} \xi_2 & \eta_2 & \zeta_2 \\ \xi_3 & \eta_3 & \zeta_3 \\ \xi_4 & \eta_4 & \zeta_4 \end{vmatrix} \right\| \tag{20}$$

is the volume of our elementary tetrahedron. Keeping in mind that the volume must be positive regardless of the selected sequence of the node points, the absolute value of the triple product in equation (20) is indicated.

Relation (19) is the desired function, and all that is left is the substitution of the proper linear expressions in terms of $\{u\}$; namely, expressions (16) and (9) in conjunction with (15). Substitution of $\{a\}$ by (19) issues the modified equation (9).

$$\{\epsilon\} = [D] [A]^{-1} \{u\} \tag{21}$$

The transposed linear matrix $\{\epsilon\}^T$ follows after applying the transposition rule twice, as follows:

$$\{\epsilon\}^T = \{[D][A]^{-1}\{u\}\}^T = \{[A]^{-1}\{u\}\}^T [D]^T$$
$$= \{u\}^T\left([A]^{-1}\right)^T [D]^T \tag{22}$$

With equations (22) and (16), the strain energy function W can be written in the following quadratic form:

$$W = \frac{1}{2} V \{u\}^T\left([A]^{-1}\right)^T [D]^T [C][D][A]^{-1}\{u\} \tag{23}$$

Equation (23) leads to the desired force displacement relationship through the second Castigliano theorem (17). Applying the differentiation rules yields

$$\{P\} = \frac{V}{2}\left([A]^{-1}\right)^T [D]^T [C][D][A]^{-1}\{u\}$$
$$+ \frac{V}{2}\{u\}^T\left([A]^{-1}\right)^T [D]^T [C][D][A]^{-1} \tag{24}$$

Repeated application of the transposition rule and observation that $[C]^T = [C]$ (the elastic matrix is a symmetric one) shows that the last right-hand term is equal to the first term. We, therefore, get the system of linear equations

$$\{P\} = V\left([A]^{-1}\right)^T [D]^T [C][D][A]^{-1}\{u\} \tag{25}$$

which contains the set of working equations to obtain the unknown displacements for computing the stress tensor according to equation (16). We call

$$[K] = V\left([A]^{-1}\right)^T [D]^T [C][D][A]^{-1} \tag{26}$$

the stiffness matrix. The stiffness matrix [K] is a symmetric one since the matrix of elastic constants [C] is symmetric, as pointed out above. The matrix $[A]^{-1}$ depends only on the local coordinates of the tetrahedron nodes. Matrix (26) therefore depends only on the local coordinates of the tetrahedron nodes and the elastic constants of the elastic medium within the tetrahedron.

The physical significance of each K-matrix element can be demonstrated by considering special load conditions at the nodes of the tetrahedron.

Let us assume that we want to interpret the meaning of the ij^{th} element of the K-matrix on the element in the i^{th} row and j^{th} column. If we set, in equation (25), all displacement components but the j^{th} one equal to zero and the j^{th} displacement component equal to one (single displacement condition), then the 12 force components become equal to the 12 coeff_ ients in the j^{th} column. The value of the coefficients in the i^{th} row and j^{th} column is therefore the i^{th} force component which must be present to maintain all displacement components at zero except the j^{th} displacement component, which must be kept at unity. In general, we must apply 12 force components to maintain this special strain condition of the tetrahedron in corresponding with the 12 coefficients in the j^{th} column.

The j^{th} force in particular acts in the same direction as the j^{th} unity displacement. Sinc the force is doing work on the system, the orientation of force and lisplacement in the j^{th} direction must correspond. This means that all diagonal elements of the K-matrix must be positive.

In many cases, it is the single unity displacement condition that helps us to visualize intuitively the physical significance of certain matrix elements.

THE COMPOUND STIFFNESS MATRIX OF A LATTICE FORMED BY A SYSTEM OF TETRAHEDRONS

Any space region can be subdivided into systems of parallelepipeds and each parallelepiped into six tetrahedrons. In this way, the space region can be built up by a system of tetrahedrons in the same way that a two-dimensional region can be covered by a net of triangles.

If we consider an interior tetrahedron — that is, one completely surrounded by four other tetrahedrons and whose node points are not boundary points of the space regions — then it can be shown that each node point must be common with node points of 17 or 23 other tetrahedrons in the neighborhood of the tetrahedron being considered. This means that the forces generated at a lattice point of interior space are the sum of all forces generated by the 18 or 24 tetrahedrons, each having one of its node points coinciding with the lattice point being considered.

Consequently, we can now see that, in general, the force at a lattice point will be affected by the displacements at each node of each of the 18 or 24 connected tetrahedrons. This is evident since all proper force equations from each of the 18 or 24 contributing tetrahedrons must be summed together in order to get the resultant force at the lattice point. In this manner, we get, in general, a linear relationship between the compound force at the lattice point and all pertinent displacement components appearing in all the 18 or 24 tetrahedrons connected at the lattice point.

Let us suppose that the i^{th} lattice point is connected to p tetrahedron, having the numerical sequence $t_1(i), t_2(i), \ldots, t_p(i)$. The elements of this sequence are functions of the lattice point, as indicated by the argument. The force vector at the i^{th} lattice point is represented by its three components

$$\begin{Bmatrix} P_1^{(i)} \\ P_2^{(i)} \\ P_3^{(i)} \end{Bmatrix} \tag{27}$$

The stiffness matrix of the tetrahedron t shall be expressed by

$$\left[K^{(t)}\right] \tag{28}$$

and the line elements of this matrix belonging to the force components 1,2,3 at the lattice point i are

$$\begin{bmatrix} \left[K_1^{(t,i)}\right] \\ \left[K_2^{(t,i)}\right] \\ \left[K_3^{(t,i)}\right] \end{bmatrix} \tag{29}$$

The compound force at the i^{th} lattice point is therefore

$$\begin{Bmatrix} P_1^{(i)} \\ P_2^{(i)} \\ P_3^{(i)} \end{Bmatrix} = \sum_{j=1}^{p} \begin{bmatrix} \left[K_1^{(t_j(i),\, i)}\right] \\ \left[K_2^{(t_j(i),\, i)}\right] \\ \left[K_3^{(t_j(i),\, i)}\right] \end{bmatrix} \left\{u^{(t_j)}\right\} \tag{30}$$

where $\left\{u^{(t_j)}\right\}$ is the column nodal displacement vector corresponding to the t_j tetrahedron.

The total matrix that is obtained in this fashion for all N lattice points is denoted by $[K^*]$, which is

$$[K^*] = \sum_{j=1}^{p} \begin{bmatrix} \left[K_1^{(t_j(i),\, i)}\right] \\ \left[K_2^{(t_j(i),\, i)}\right] \\ \left[K_3^{(t_j(i),\, i)}\right] \end{bmatrix} \tag{31}$$

where $i = 1,2,3,..N$.

It is obvious that the configuration of tetrahedrons connected to a lattice point i will be different when i is a boundary point itself. The column vector $\left\{t_j^{(i)}\right\}$ will contain lesser elements than the corresponding matrix for interior lattice points.

We have distinguished between the force components belonging to a lattice point i and in a certain direction (1, 2, or 3). This scheme has the advantage of allowing us to know with which element we are dealing. In order to use one index number to identify a force (or displacement) component but still contain the information of the lattice number i and the type of direction (J = 1, 2, 3), we introduce the index number scheme:

$$n = 3(i - 1) + j$$

($i = 1,2...N$; $j = 1,2,3$) . In this way, the indices of components form a sequence of natural numbers, and each number still contains the information of being an x, y, or z component as well as the lattice index number i . This information can be extracted easily, since

$$j = \mathrm{mod}(n,3) + 1$$

where $\mathrm{mod}(n,3)$ is the remainder of division, n divided by 3, and

$$\left.\begin{array}{ll} i = \left[\frac{n}{3}\right] + 1 & \text{if } \mathrm{mod}(n,3) \neq 0 \\ \text{or} & \\ i = \left[\frac{n}{3}\right] & \text{if } \mathrm{mod}(n,3) = 0 \end{array}\right\} \tag{32}$$

where $\left[\frac{n}{3}\right]$ means the largest integer contained in the quotient $\frac{n}{3}$.

With matrix (31), the system of force components acting upon the given configuration of lattice points can be expressed symbolically by

$$\{P^*\} = [K^*] \{u^*\} \tag{33}$$

Since we use indexing of forces and displacements, the system of equations (33) can be written in analytical form, such as

$$P_i^* = \sum_{j=1}^{M} K_{ij}^* \; u_j^* \tag{34}$$

where $i = 1,2,3...M$. We assume now that there is M, the total number of force components, as well as displacement components. The matrix $[K^*]$ will be one of $M \times M$.

Because of applied forces and displacements introduced to the system, there are some force components known and the remaining force components unknown. The same is true for the displacement components in a complementary sense. If there are M_1 unknown force components, then there are $M - M_1$ known force components and there will be $M - M_1$ unknown displacement components while there are M_1 known displacements. Let us arrange the equations (34) such that the first M_1 equations express the equilibrium of the unknown forces and the rest express the equilibrium of the known forces. In the same fashion, we arrange the terms in each equation such that the first M_1 terms contain known displacements and the remaining terms contain the unknown displacements. Furthermore, we distinguish by superindexing the known and unknown displacements and forces with (k) and (u). Then we get the following two systems of equations:

$$P_i^{(u)} = \sum_{j=1}^{M_1} K^*_{ij} u_j^{(k)} + \sum_{j=M_1+1}^{M} K^*_{ij} u_j^{(u)} \tag{35}$$

where $i = 1,2...M_1$, and

$$P_i^{(k)} = \sum_{j=1}^{M_1} K^*_{ij} u_j^{(k)} + \sum_{j=M_1+1}^{M} K^*_{ij} u_j^{(u)} \tag{36}$$

where $i = M_1 + 1, M_1 + 2,..M$.

The last equation is a system of linear equations for the $M - M_1$ unknown displacements $u_j^{(u)}$, where $j = M_1 + 1, M_1 + 2,..M$.

With the solution of equation (36), the unknown forces can be computed from equation (35). Equation (36) is a nonhomogeneous linear system of $(M - M_1)$ equations:

$$\sum_{j=M_1+1}^{M} K^*_{ij} u_j^{(u)} = P_i^{(k)} - \sum_{j=1}^{M_1} K^*_{ij} u_j^{(k)} \tag{37}$$

where $i = M_1 + 1, M_1 + 2,..M$.

In order to apply numerical methods to solve this system of equations, it is necessary to re-index the unknowns by consecutive natural numbers. This can be easily done by generating a number sequence (MSK matrix) that contains the indices of the unknown displacements in a consecutive order. The arguments of this sequence are now consecutive, natural numbers which are the working indices for solving equations (37). By means of the MSK matrix, it is now possible to identify each solution element with the original index. The main purpose of this scheme is to provide the capability to extract the information of location (lattice point index) and direction (x, y, or z) of the displacement from the equation working index. This capability is required in many instances; e.g., to satisfy the polar symmetry conditions and to select the proper sequence of the 12 displacement elements for each tetrahedron in order to compute the stresses. Stress within a tetrahedron is computed according to equation (16). The stress distribution for the three-dimensional region of elastic medium is then obtained by applying certain boundary conditions.

In the following, the boundary problem will be specified and the scheme of the numerical approach described.

DESCRIPTION OF THE NUMERICAL SCHEME

Definition of the Boundary Problem

The main objective of this study is to investigate the effect of free surfaces on the stress distribution within a fiber matrix composite under loads transverse to the fiber.

Figure 2 shows this configuration schematically. The stress distribution problem here is three dimensional. From this figure, we can observe some symmetry conditions if we assume that the fibers form a periodic network of hexagonals with one fiber axis intersecting at each apex and centroid of the hexagonals (hexagonal A,B,C,D,E,F is one of them).

Around each fiber cross section, there is a hexagonal such that another network of hexagonals is formed. This hexagonal is called the basic fiber-resin element of the system. In Figure 2, the hexagonal A',B',C',D',E',F' is such an element.

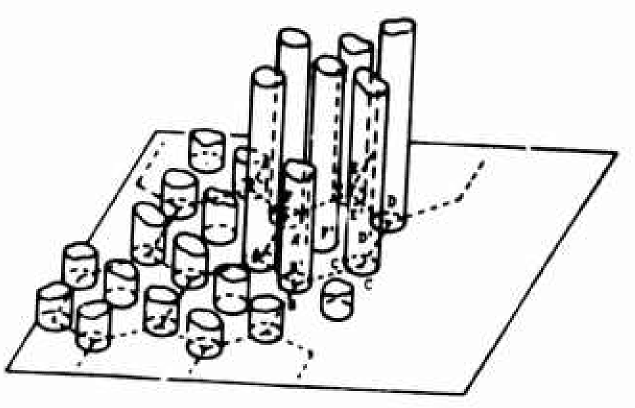


Figure 2. Basic Fiber-Resin Element.

Figure 3 depicts the top view of the intersection of the fiber-resin element.

We consider now an external force acting in a plane normal to line $\overline{B',E'}$ but at a distance which is a high multiple of distance $\overline{B',E'}$ away from M . We call this the transverse load upon one system. Because of symmetry of load and configuration conditions, the displacements at points within the fiber axial-parallel planes going through lines M_1 , M_2 and P_1 , P_2 are constant and in the direction of the external force. They are oriented opposite each other but are of equal magnitude. All points of the plane normal to line M,P_1 and containing point P_1 must have displacements along the plane they generate. The same displacement conditions exist for points in the plane parallel to the one described above but containing point M . Both planes are planes of symmetry for the displacement contribution. In this way, there is a periodic repetition of the displacement pattern which exists within the prism above the rectangular cell M,N,M_1,P_1 in the direction of the four planes of symmetry which are the side planes of the prism. Therefore, we can study this transverse load problem by the following boundary conditions, which are based on the new coordinate convention for the basic prism shown in Figure 4.

$$\sigma_z(x,y,o) = 0 \tag{38}$$

$$\sigma_{xz}(x,y,o) = 0 \tag{39}$$

$$\sigma_{yz}(x,y,o) = 0 \tag{40}$$

$$u_x(\pm c,y,z) = \pm k \tag{41}$$

$$u_y(x,\pm\frac{b}{2},z) = 0 \tag{42}$$

$$\frac{\partial u_x}{\partial z}(x,y,\ell) = 0 \tag{43}$$

$$\frac{\partial u_y}{\partial z}(x,y,\ell) \rightarrow 0 \tag{44}$$

$$u_z(x,y,\ell) \rightarrow 0 \tag{45}$$

The last three equations indicate that the condition of plane strain will be approached with increasing distance from the free surface $z = 0$. The distance ℓ can be chosen freely such that it will be more than four times the dimension of the diagonal of the base triangle. This assumption has been based mainly on St. Venant's principle. Our numerical results verify the correct selection of ℓ in this respect.

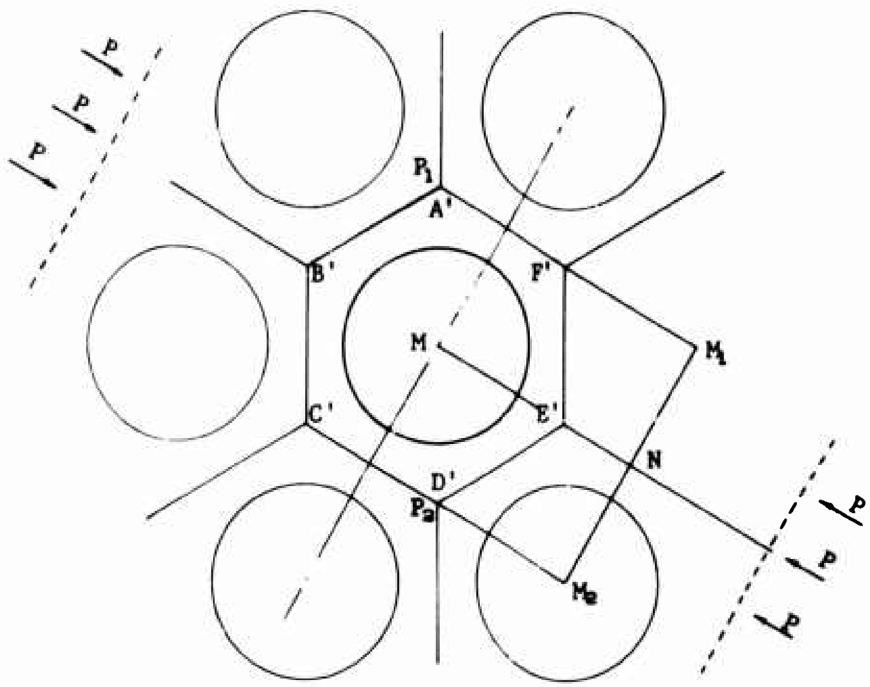


Figure 3. Hexagonal Array of the Fibers.

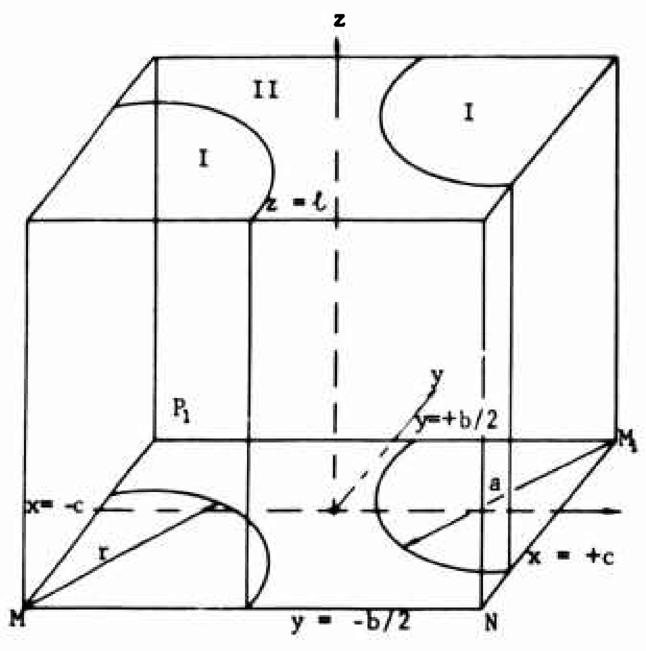


Figure 4. Prismatic Representative Element.

Within the region, two types of elastic materials are considered. The cylindrical bodies have the modulus of elasticity E_f and Poisson's ratio ν_f . Outside of the cylindrical domains, the associated elastic constants are E_m and ν_m .

With these premises, it is now possible to compute the stresses by applying the method of finite elements. In order to do this, it is first necessary to subdivide the interior of the above-specified three-dimensional region into a system of elementary tetrahedrons.

The System of Elementary Tetrahedrons

In Figure 4, we refer to the free-boundary plane $z = 0$ as "ground floor." Furthermore, we define eight equidistant planes parallel to the ground floor, and we call the n^{th} plane the "n^{th} floor." The top plane, $z = \ell$, is referred to as the "tenth floor." These 10 planes subdivide the parallelepiped into nine layers, each having the height $d_z = \ell/9$.

Next, we have to subdivide each layer into prisms with triangular bases, since each prism can be again subdivided into three tetrahedrons. The simplest way to do this is to subdivide each layer in the same way, which means that we erect a set of prisms on the ground floor, each prism having the height ℓ . Each layer will thus have the same layout of triangular bases as the one on the ground floor. Figure 5 shows the way the ground floor has been subdivided into 96 triangle bases (8 rows with 12 triangles in each row). Each triangle represents a prism with the height of the layer thickness d_z . The top triangle of the prism is a replica of the base triangle and by itself is the base triangle for the next prism above the second layer. In this fashion, all prisms in each layer above a base triangle are identical geometrically. The base triangles are numbered on the ground floor in a consecutive manner. Because of central symmetry properties, only 55 triangles need to be considered. With the same scheme, the lattice points at the ground floor are numbered consecutively in each row and column. Because of central symmetry properties, the last point on the ground floor is the origin, which is lattice point number 32. Continuation of the numbering starts with the lattice point above number 1 in the same manner as in the ground plane below. In this way, we can calculate the corresponding lattice point number J in the p^{th} floor above the I^{th} lattice point by

$$J = I + 32(p - 1) \tag{46}$$

As pointed out earlier, the indices scheme of the force and displacement components can be linked with the indexing scheme of the lattice points. Consequently, the q^{th} component ($q = 1$ is x-component, $q = 2$ is y-component, and $q = 3$ is z-component) of the vector quantity has considered the following index Q :

$$Q = 3(J - 1) + 9 \quad \text{or} \quad Q = 3[I + 32(p - 1) - 1] + q \tag{47}$$

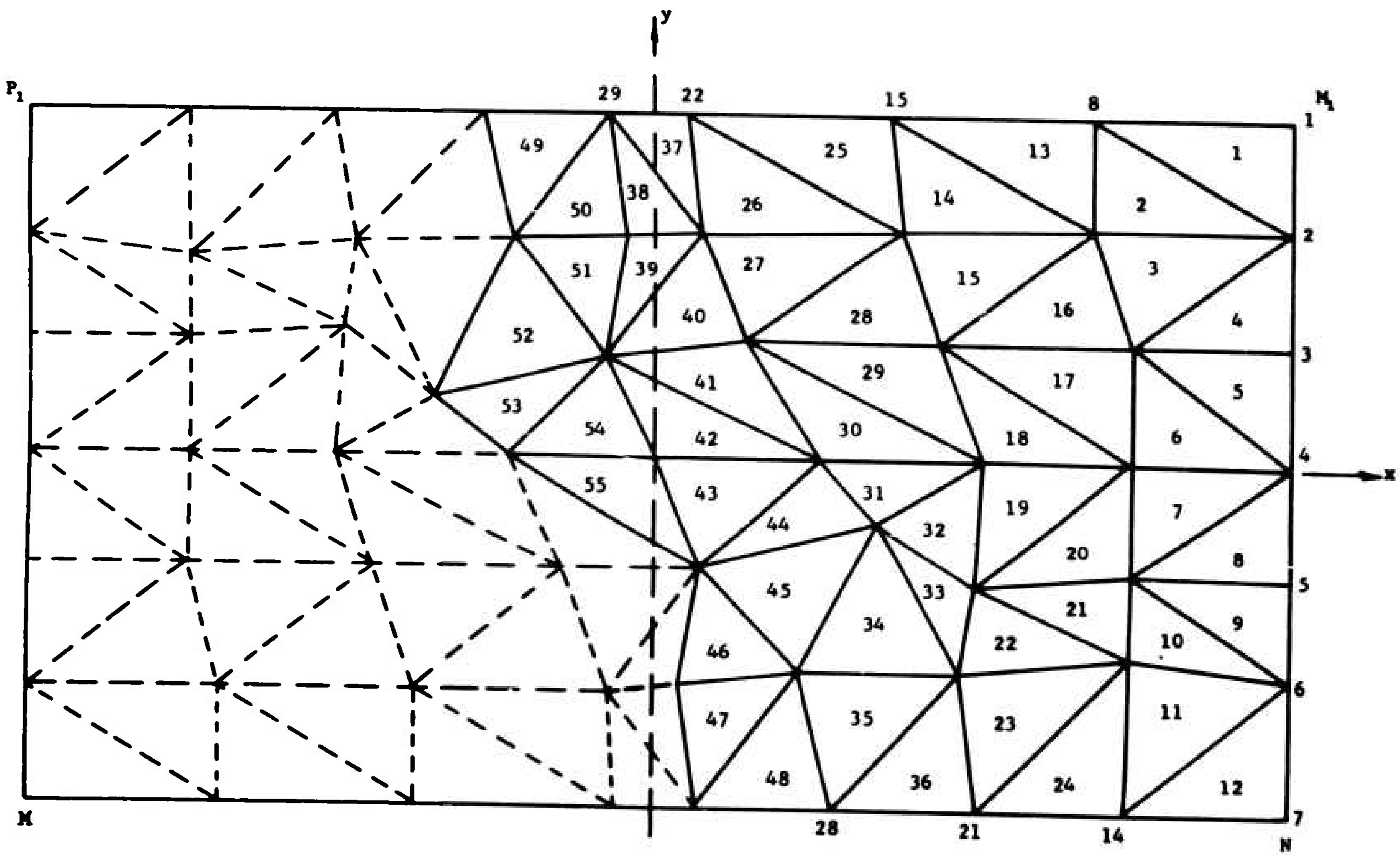


Figure 5. Finite Elements of a Basic Representative Cross-Sectional Area of Floor Number 1 Used in the Numerical Analysis.

We call the prism above the base triangle between two consecutive floors the elementary prism. Since a triangular prism can be subdivided into three tetrahedrons in several ways, we must establish some basic rules. Three tetrahedrons in any triangular prism are generated through intersecting the prism by two planes. These two planes are generated by the three lines of diagonals in each of the three rectangles forming the mantle of the prism. See Figure 6. Furthermore, it is necessary that one diagonal line intersect the other two diagonals. In this way, all three diagonals form a linked train of lines (train of diagonals) with disconnected, or open, ends. The first basic rule we apply in order to generate the tetrahedrons is the following: The diagonals of coinciding rectangles belonging to adjacent prisms coincide.

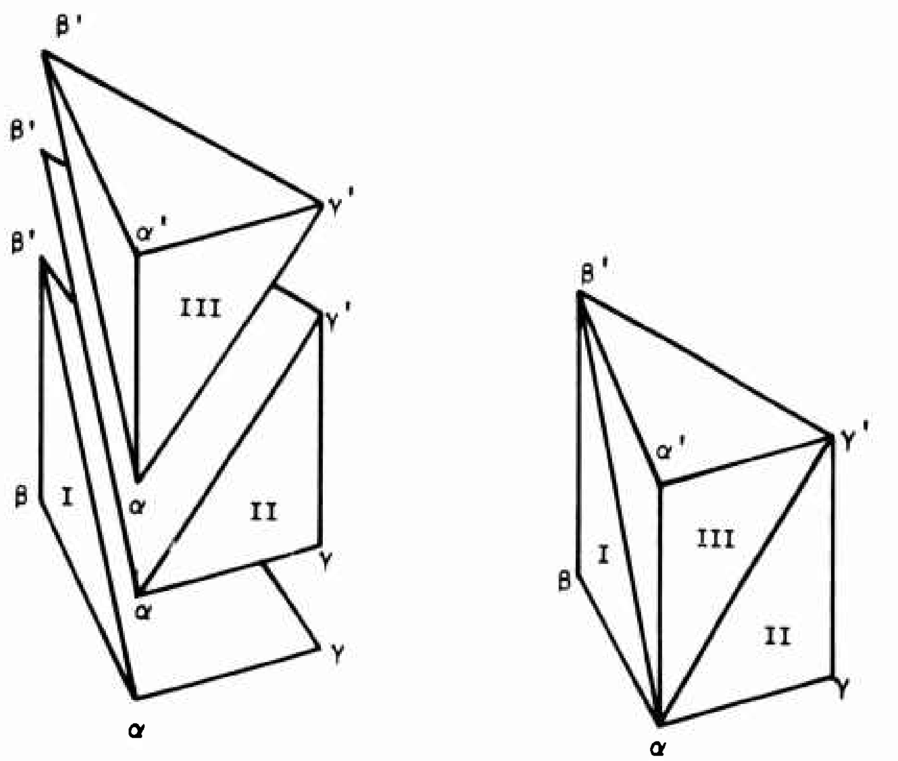


Figure 6. Tetrahedron Forming a Triangular Prism.

At this point, we distinguish each point of the triangle base as to the topological property of the train of diagonals with respect to the relative position to the triangle base proper. A lattice point of a base triangle at which two diagonal lines intersect each other is said to be of a topological α-property. When two diagonal lines intersect at the lattice point, then the lattice point considered has a topological β-property. Finally, there is a lattice point topological γ-property when one diagonal goes through the lattice point under consideration and the other diagonal goes through the lattice point above or below the one under consideration.

The first conclusion from this is that a base triangle must contain points of all three topological properties. The second conclusion that follows from this topological consideration is that all apexes of base triangles having one common lattice point are of the same topological type.

According to the last theorem, it is therefore appropriate to think in terms of topological properties of lattice points only. If we know the topological property of each lattice point in a base triangle, we know the lattice points of the four apexes for each of the three tetrahedrons above the base triangle. With reference to Figure 6, we can state the following:

Tetrahedron I. The three base apexes are the three lattice points of the base itself. The fourth apex is the β-point in the next floor (β' in Figure 6).

Tetrahedron II. The three base apexes are the α-point at the base, the β-point above the base (β' in Figure 6), and the γ-point above the base (γ' in Figure 6). The fourth apex is the γ-point in the base triangle.

Tetrahedron III. The three base apexes are the three lattice points of the triangle above the base. The fourth apex is the γ-point in the base triangle.

The last three properties contain the algorithm to generate the four lattice numbers corresponding to the four apexes of each tetrahedron. It is therefore necessary to know the topological properties of each lattice point. In the numerical program, the coordinates of each ground floor lattice point are stored as input quantities. With the information of the lattice point numbers of the tetrahedron apexes, the associated point coordinates can be selected and the stiffness matrix [K] of the tetrahedron can be computed according to equation (26). The sequence of four lattice point indices corresponding to the four apexes is computed and stored for each tetrahedron. In the numerical program, it is called the IV matrix. This matrix is later needed to help identify the line and column of the [K] matrix with respect to the displacement and force indices (with the help of equation 47). After identification, a look-up routine of the index number in the MSK matrix allows generation of the matrix $[K^*_{ij}]$ appearing at the left- and right-hand sides of equation (37). This look-up routine is a screening process that is also used to accumulate the proper terms of the force components.

The objective of this study was to autogenerate the IV matrix. This was accomplished in two steps. The first was to generate the three lattice points for each base triangle (the nonordered sequence is the IM(J,1) matrix in the program, with J = 1,2,3); the second was to obtain, from the nonordered sequence, the α-β-γ ordered sequence of lattice numbers (the IM(J,2) matrix in the program, with J = 1,2,3). The algorithm for both steps follows from some simple topological considerations.

The configuration of triangles in Figure 5 allows us to derive the sequence of node point indices as a function of triangle index. Let $I \le 48$ be the triangle index. Then we have node 1, the lattice index,

$$IM(1,1) = 1 + \left[\frac{I}{2}\right] + \left[\frac{I-I_e(12)}{12}\right] \tag{48}$$

with $I_e(12)$ equalling zero if mod(I,12) = 1 , or one if mod(I,12) = 0 ,

$$IM(2,1) = IM(1,1) + \left(\frac{1}{2} \quad 1 - (-1)^{[I/2]}\right) + 7 \tag{49}$$

$$IM(3,1) = IM(1,1) + 7 \tag{50}$$

The ratios in brackets indicate that the largest integer contained in the quotient is taken.

For all triangles with numbers greater than 48, a mirror-image configuration exists, and the lattice indices are computed according to the following:

$$IM(1,1) = 1 + \left[\frac{I}{2}\right] + \left[\frac{I-I_e(12)}{12}\right] \tag{51}$$

with $I_e(12)$ equalling zero if mod(I,12) = 1 , or one if mod(I,12) = 0 ,

$$IM(2,1) = IM(1,1) + 7 \tag{52}$$

$$IM(3,1) = IM(1,1) + \left(\frac{1}{2} \quad 1 - (-1)^{[I/2]}\right) + 7 \tag{53}$$

So far, we have obtained the nonordered sequence of lattice point indices for each triangle. Before we continue to obtain the set of α-β-γ ordered lattice point indices, it is appropriate to incorporate the central symmetry condition existing with the problem. This symmetry affects all displacements at lattice points with indices larger than 32.

Because of central symmetry of the displacement distribution, we have

$$u_x(-x,-y,z) = -u_x(x,y,z) \tag{54}$$

$$u_y(-x,-y,z) = -u_y(x,y,z) \tag{55}$$

$$u_z(-x,-y,z) = +u_z(x,y,z) \tag{56}$$

This means that all displacement components at lattice points with the indices beyond 32 remain dependent upon the independent displacement components already introduced at the first 32 lattice points.

Because of equations (54) through (56), the axial-symmetric triangle configuration has been chosen in Figure 5 and the last considered lattice point 32 terminates at the origin. The force equilibrium equations of all components at the first 32 lattice points (and of the corresponding points in the floors above) therefore represent one complete set of independent equations of the problem when relations (54) through (56) are taken into account. In order to include all [K] matrix elements of triangles associated with all symmetrically independent (96 per floor) force components, the [K] matrices up to triangle 55 must be computed. Displacement components for all lattice points above 32 are dependent according to relations (54) through (56). Therefore, it is not necessary to introduce more displacement indices or lattice point indices, but rather to use the central symmetry property in assigning the lattice index numbers for points beyond 32. If, in equations (48) through (53), IM is larger than 32, it will be replaced by 64 - IM, and a weight number IKO = +1 will be attached to this point. In this way, the displacement indices of the symmetric points are made equal, but the weight number assigned to each lattice allows us to distinguish between independent and symmetrically dependent indices (and therefore displacements). For lattice points with index numbers smaller than 32 (belonging to independent displacements), the IKO value is zero.

In this way, a sequence of numbers to each base triangle, which is called the IKO matrix, is generated along with the IM matrices. The purpose of the IKO matrix is twofold:

1. To generate the coordinates for lattice points beyond index 32 (only coordinates of lattice points 1 through 32 are given).
2. To multiply the K matrix elements with +1 or -1 before collecting by the summation process to generate [K*] . All elements associated with displacement components at lattice points smaller than 32 are multiplied by +1.

[K] matrix elements associated with displacements in the z-direction, for lattice points beyond 32, are also multiplied by +1 before collecting. However, the [K] matrix elements associated with displacements in the x- and y-directions at lattice points beyond 32 are to be multiplied by (-1) before additive storing in the matrix [K*] . This scheme follows from relations (54) through (56).

For instance, in triangle 44 there is

$$IM(3,1) = 64 - 34 = 30$$

with IKO(3,1) = 1 ; in triangle 50 there is

$$IM(3,1) = 30$$

with IKO(3,1) = 0 .

With the IM(J,1) matrix and the IKO(J,1) matrix generated, where J = 1,2,3 , the lattice points for a specified triangle are known. Since the index number of the lattice point in the IM matrix does not reveal whether it is an independent or axial-symmetric dependent point, the IKO matrix carries this information for this purpose.

From the IM(J,1) matrix, we get the α-β-γ ordered matrix by using the topological pattern existing for the lattice points in each row. The first row points alternate between γ-α points, the second row points between β-γ points, and the third row points between α-β points; the fourth row shows the same cycling as the first row, and the fifth shows the same cycling as the second row. We assign to each point with the lattice number I a number $\ominus(I)$, such that

$$\ominus(I) = +1 \quad \left\{ \begin{array}{l} \text{if I is a} \\ \text{topological} \\ \alpha\text{-point} \end{array} \right.$$

$$\ominus(I) = -1 \quad \left\{ \begin{array}{l} \text{if I is a} \\ \text{topological} \\ \beta\text{-point} \end{array} \right.$$

$$\ominus(I) = 0 \quad \left\{ \begin{array}{l} \text{if I is a} \\ \text{topological} \\ \gamma\text{-point} \end{array} \right.$$

Corresponding to the γ-α alternation of the first seven lattice points, the associated $\ominus$ values form a sequence 0,+1,0,+1,0,+1,0 . The next seven form a sequence -1,0,-1,0,-1,0,-1 , and the seven lattice points in the third row form a sequence +1,-1,+1,-1,+1,-1,+1 .

The following 21 numbers are periodic to the first 21. This sequence of 21 numbers is identical with the elements of the sequence $\{ \ominus (I) \}$, where $I = 1,2,.,21$, with

$$\ominus (I) = \frac{1}{3}\left\{ \frac{1+(-1)^I}{2} \left[2 \cos \frac{\pi}{3} (1 + 2k) - 2 \cos \frac{\pi}{3} (3 + 2k)\right] - (-1)^k \left[1 - 2 \cos \frac{\pi}{3} (-1 + 2k)\right]\right\} \tag{57}$$

with $k = [I-I_e(7)/7]$ and with $I_e(7)$ equalling one if $\mathrm{mod}(I,7) = 0$, or equalling zero if $\mathrm{mod}(I,7) \neq 0$. Formula (57) serves as a guide in making a decision about the topological property of the lattice point having the index I . Figure 7 is a flow diagram of a version following the logical content of formula (57) that was used in this investigation.

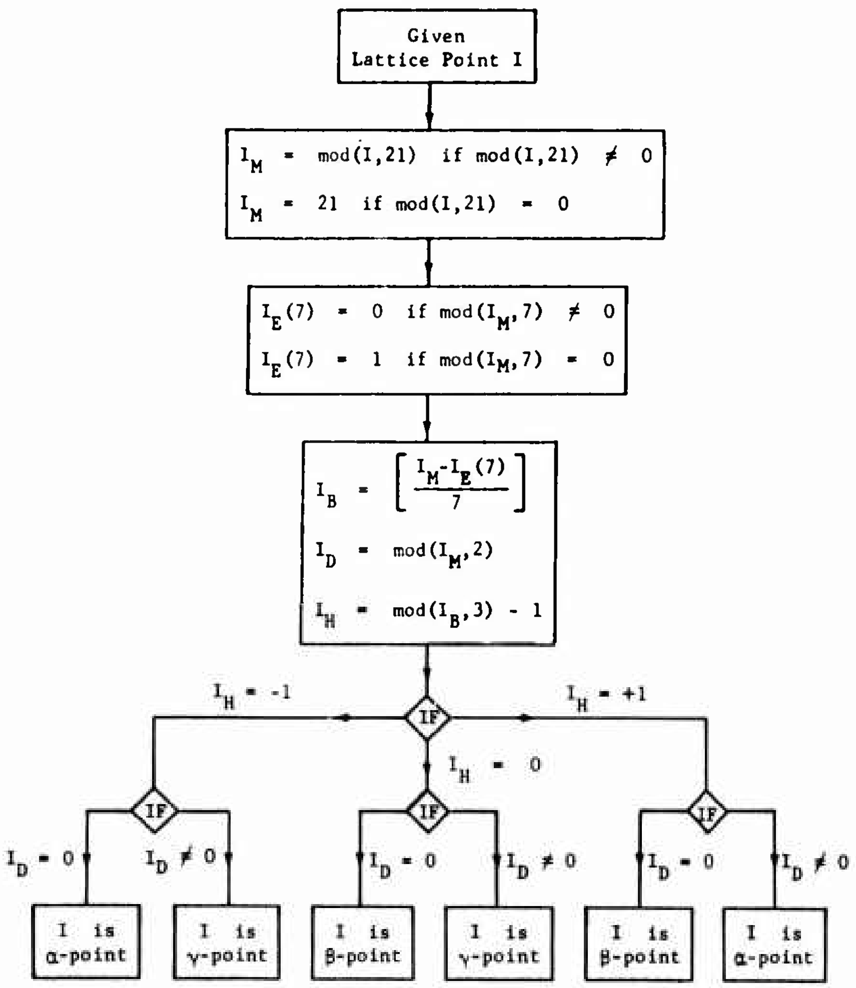


Figure 7. Computer Flow Diagram to Determine Topological Property of Lattice Point.

Going with each lattice point index through this routine, we generate the α-β-γ ordered vector $\{IM(J,2)\}$, where $J = 1,2,3$, with the meaning

$$\left.\begin{aligned} IM(1,2) &= \text{index of } \alpha\text{-point} \\ IM(2,2) &= \text{index of } \beta\text{-point} \\ IM(3,2) &= \text{index of } \gamma\text{-point} \end{aligned}\right\} \text{of a particular base triangle} \qquad (58)$$

With the $\{IM\}$ matrix, the four lattice point indices of the apexes for all three tetrahedrons above the base triangle follow immediately from the above definition of generating the apexes of the tetrahedrons.

First tetrahedron above the J^{th} triangle:

$$\begin{aligned} IV(J,1,1) &= IM(1,2) \\ IV(J,2,1) &= IM(2,2) \\ IV(J,3,1) &= IM(3,2) \\ IV(J,4,1) &= IM(2,2) + 32 \end{aligned} \qquad (59)$$

Second triangle above the J^{th} triangle:

$$\begin{aligned} IV(J,1,2) &= IM(1,2) \\ IV(J,2,2) &= IM(2,2) + 32 \\ IV(J,3,2) &= IM(3,2) + 32 \\ IV(J,4,2) &= IM(3,2) \end{aligned} \qquad (60)$$

Third tetrahedron above the J^{th} triangle:

$$\begin{aligned} IV(J,1,3) &= IM(1,2) + 32 \\ IV(J,2,3) &= IM(2,2) + 32 \\ IV(J,3,3) &= IM(3,2) + 32 \\ IV(J,4,3) &= IM(1,2) \end{aligned} \qquad (61)$$

With the IV matrix, the set of four apex indices is known; therefore, the coordinates of the apexes can be obtained and the elements of the [K] matrix by means of equation (30) can now be computed. The IV matrix must be saved, since it is used to identify row and column of the [K] matrix as well as to generate the desired elements of the [K] matrix.

In the next section, a description is given of how the desired elements of the matrix [K*] are generated from the matrices [K] .

Generation of the Matrices Required To Solve for the Unknown Displacements

The main objective was to find the coefficient matrices of equation (37) and to find solutions for the unkown displacement components $u_j^{(u)}$, where $j = M_1 + 1, M_1 + 2, \ldots M$.

Earlier in this report, a detailed description was given of the generation of the elements of the matrix [K*] . Equation (31) reveals the algorithm for obtaining the coefficients of [K*] , and equation (37) indicates which of the coefficients are actually used. These two propositions are observed in generating the matrix elements of equation (37).

The actual procedure calls for re-indexing of the unknown displacements in a consecutive, natural number sequence, since the indexing derived from the lattice point indices (see equation 47) represents a nonsequentially ordered subset of natural numbers. Re-indexing is conveniently done by means of the sequence containing the indices of unknown displacement components (or known force components) as elements. This sequence is called the MSK matrix and has been generated simply by picking up only displacement indices which are unknown in accordance with boundary conditions (38) through (45). The first three numbers in the first column at each lattice point are the indices of the x,y,z displacements at that lattice point. Behind each index is the argument number of its appearance in the MSK matrix.

From the three-dimensional lattice configuration, wherein Figure 5 is the ground-floor portion, we can see that the first 80 equilibrium equations contain terms involving the first 160 unknowns only. This is because the force components of the first floor are influenced by displacement coefficients of the first and second floors only. The first 80 lines of the matrix [K*] thus have nonzero elements in the first 160 columns only; all further columns contain zero elements.

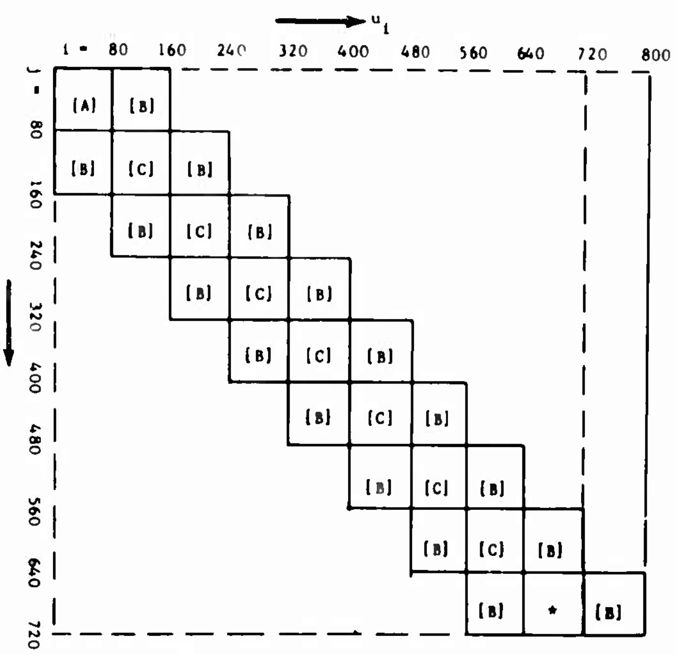
The 80 equilibrium equations of all known force components at the second floor (equations 81 through 160) are in general connected to unknown displacements in the first, second, and third floors. This means that the [K*] matrix elements for lines 81 through 160 contain, in general, nonzero elements in the first 240 columns.

The equilibrium equations of the third-floor force components are of the same nature as the corresponding equations of the second floor. The only difference is that all the unknown displacement coefficient indices increase by 80. This means that the submatrix contained in lines 81 through 160 is repeated in lines 161 through 240, but that all columns are shifted to the right by 80 columns. The same considerations apply for all floors following. Matrix [K*] therefore contains three diagonal non-zero submatrices which are the same in the entire matrix, except for the first and last diagonal matrices. Figure 8 depicts the [K*] matrix.

Because the [K*] matrix is symmetric, only three different types of 80 x 80 submatrices are involved in defining [K*] . These are the submatrices [A],[B],[C] . This means that it is sufficient to generate the [K*] elements for the first 160 lines only. In other words, only the proper elements of the [K] matrices of the set of tetrahedrons between the first and third floors need to be taken into account in order to generate [A],[B],[C] .

The scheme applied to generate the matrix elements of the 80 x 80 matrix [AS22] = [[A],[B]] and the 80 x 160 matrix [KS22] = [[B],[C],[B]] is basically a screening routine applied to each [K] matrix element extended over all [K] matrices of tetrahedrons in the first two layers. The [K] matrix of a particular tetrahedron in the first layer is identical to the [K] matrix of the tetrahedron that is its second-layer counterpart. Therefore, the [K] matrices for the three first-floor tetrahedrons above a triangle are sufficient, for our purposes, to generate [AS22] and [KS22] .

In the process of giving a program description, it was shown in the preceding section that the [K] matrix for each of the three tetrahedrons above a base triangle is obtained. With the additional information gathered in this section, it is now possible to identify each element of the matrix with respect to that element of the matrix [AS22] or [KS22] (including the right-hand-side independent terms) to which it will be added. A diagram of this scheme, Figure 9, shows how this is done.



* [C] when 720 x 800 + boundary condition

[C'] when 720 x 720 (broken line matrix for incorporated boundary conditions)

Figure 8. The [K*] Matrix Configuration.

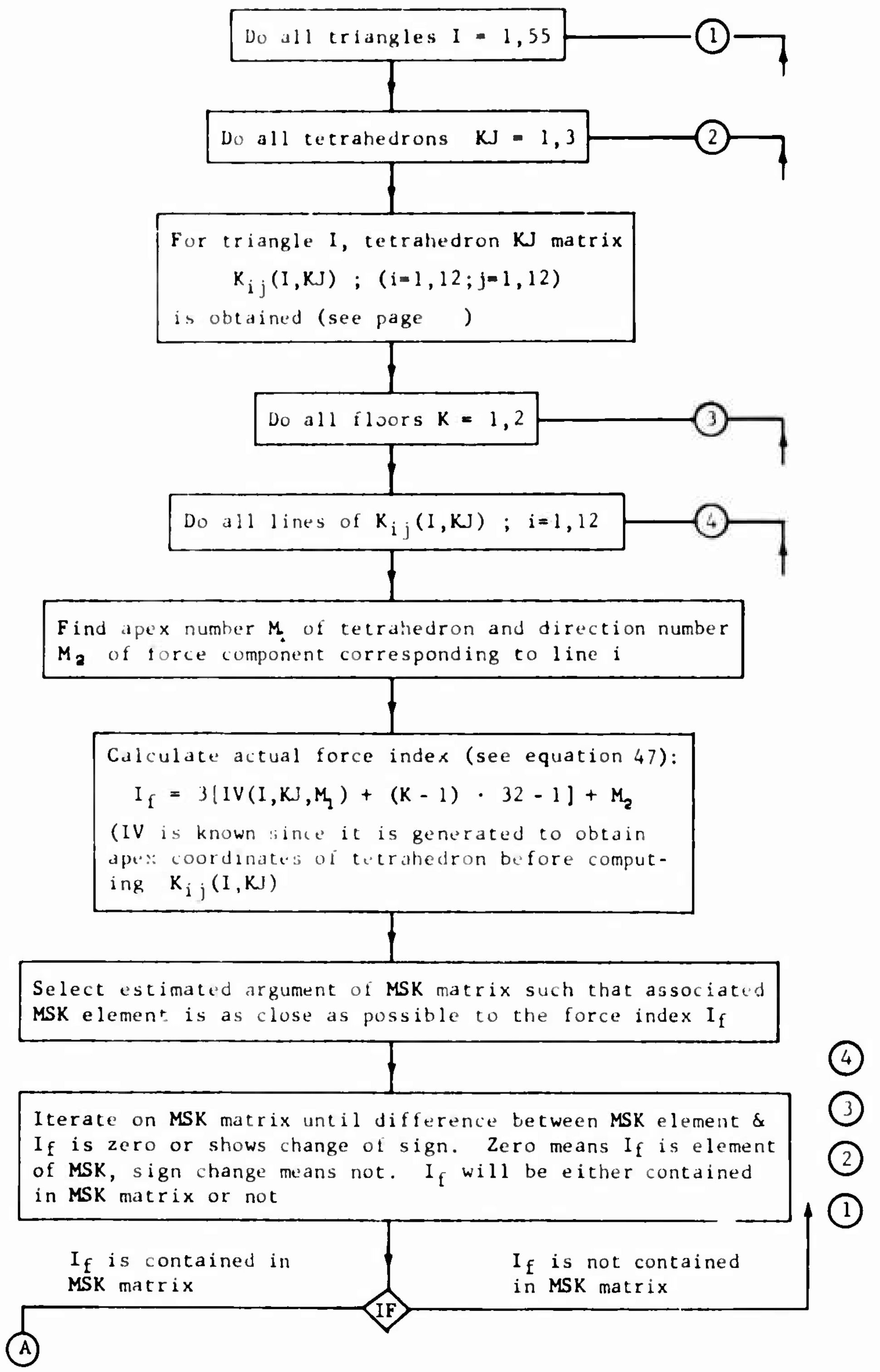
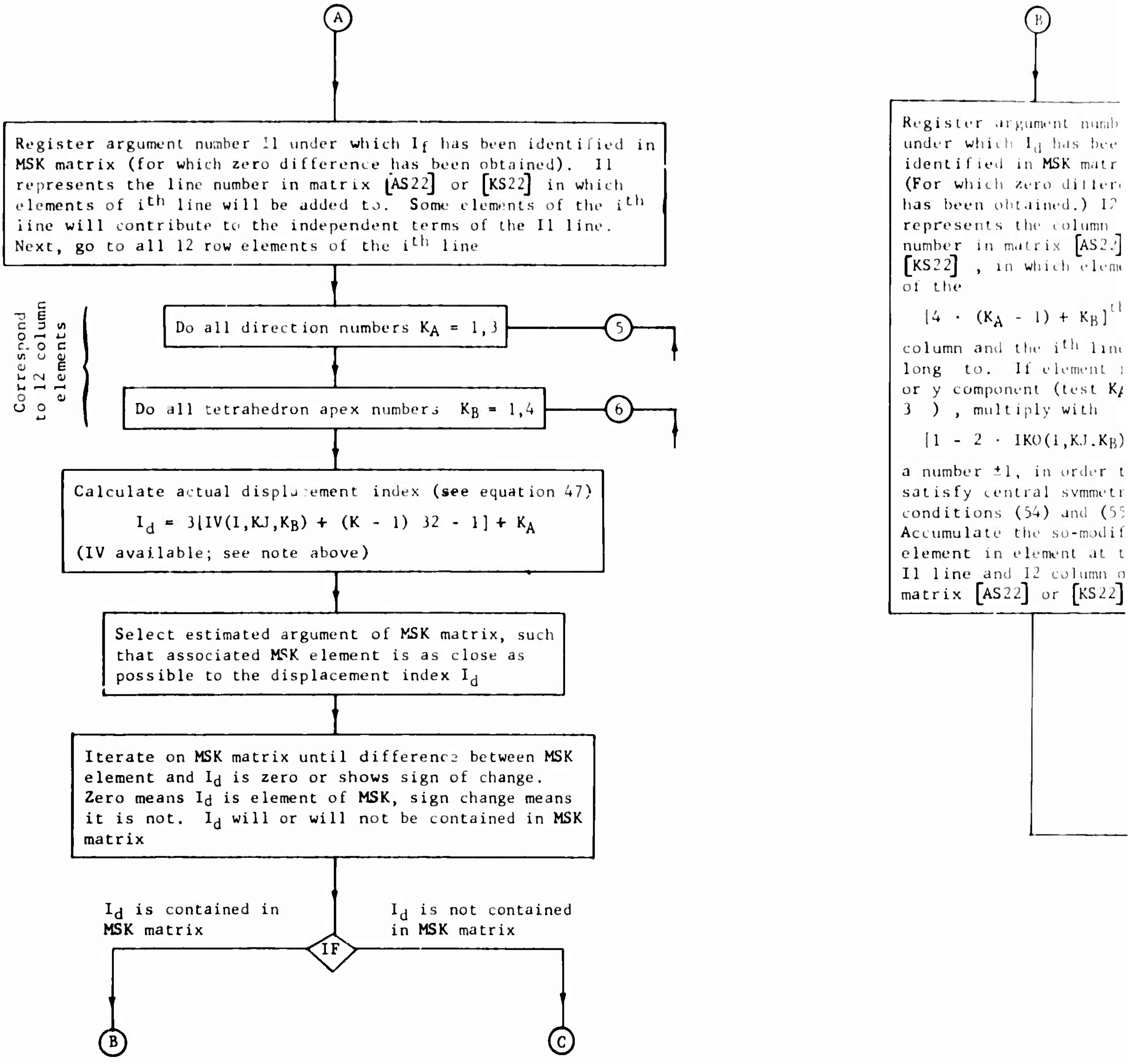
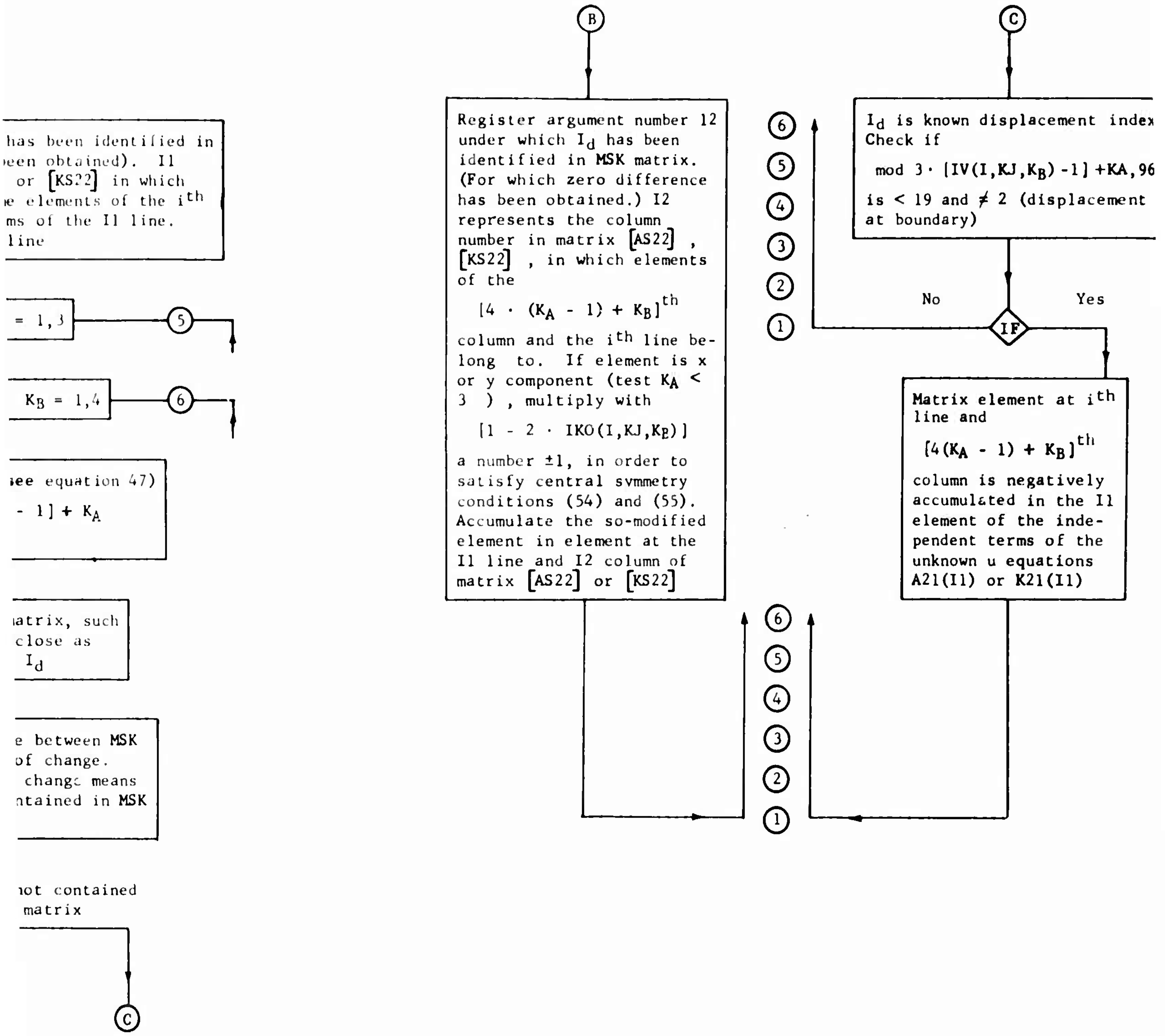


Figure 9. Computer Flow Diagram for Calculating The Stiffness Matrix [K*].


A
Register argument number I1 under which I_f has been identified in MSK matrix (for which zero difference has been obtained). I1 represents the line number in matrix [AS22] or [KS22] in which elements of i^{th} line will be added to. Some elements of the i^{th} line will contribute to the independent terms of the I1 line. Next, go to all 12 row elements of the i^{th} line
Correspond to 12 column elements
Do all direction numbers K_A = 1,3
5
Do all tetrahedron apex numbers K_B = 1,4
6
Calculate actual displacement index (see equation 47)
$I_d = 3[IV(I,KJ,K_B) + (K - 1)\ 32 - 1] + K_A$
(IV available; see note above)
Select estimated argument of MSK matrix, such that associated MSK element is as close as possible to the displacement index I_d
Iterate on MSK matrix until difference between MSK element and I_d is zero or shows sign of change. Zero means I_d is element of MSK, sign change means it is not. I_d will or will not be contained in MSK matrix
I_d is contained in MSK matrix
IF
I_d is not contained in MSK matrix
B
C
B
Register argument
under which I_d has
identified in MSK
(For which zero
has been obtained.)
represents the column
number in matrix
[KS22] , in which
of the
$[4 \cdot (K_A - 1) + K_B]$
column and the i^{th}
long to. If element
or y component (test
3) , multiply with
$[1 - 2 \cdot IKO(I,KJ,K_B)]$
a number ±1, in order
satisfy central symmetr
conditions (54) and
Accumulate the so-modif
element in element at
I1 line and I2 column
matrix [AS22] or [KS22]

A



has been identified in
een obtained). I1
or [KS22] in which
e elements of the ith
ms of the I1 line.
line
= 1,3
5
KB = 1,4
6
see equation 47)
- 1] + KA
atrix, such
close as
Id
e between MSK
of change.
change means
ntained in MSK
not contained
matrix
C
B
Register argument number I2 under which Id has been identified in MSK matrix. (For which zero difference has been obtained.) I2 represents the column number in matrix [AS22], [KS22], in which elements of the
[4 · (KA - 1) + KB]th
column and the ith line belong to. If element is x or y component (test KA < 3), multiply with
[1 - 2 · IKO(I,KJ,KB)]
a number ±1, in order to satisfy central symmetry conditions (54) and (55). Accumulate the so-modified element in element at the I1 line and I2 column of matrix [AS22] or [KS22]
6
5
4
3
2
1
C
Id is known displacement index. Check if
mod 3 · [IV(I,KJ,KB) -1] +KA,96
is < 19 and ≠ 2 (displacement at boundary)
No
IF
Yes
Matrix element at ith line and
[4(KA - 1) + KB]th
column is negatively accumulated in the I1 element of the independent terms of the unknown u equations A21(I1) or K21(I1)

B

Since the IV matrix is generated for each triangle index I according to Figure 9, the only input required for this routine is the x- and y-coordinates of the 32 ground-floor lattice points.

After compl ion of all the indicated looping processes, all of the 12 x 12 elements of each [K] matrix belonging to every tetrahedron in the two layers have been screened and properly accumulated in the matrices [AS22] and [KS22] as well as in the independent vectors [A21] and [K21] . Therefore, the elements of the matrices [AS22] and [KS22] along with the independent vectors [A21] and [K21] are complete and considered to be generated. The matrix represented in Figure 8 is therefore established.

The systems of equations belonging to the Figure 8 matrix is undetermined, since the matrix has 720 lines and 800 columns. The 80 missing equations are obtained by taking the internal boundary conditions (43), (44), and (45) — in other words, those "inside" the body — into account. In line with the first-order approximation used throughout this study, we can enforce an approximate version of (43) and (44) by making all u_x and u_y components at the tenth floor equal to the corresponding u_x and u_y components at the ninth floor. By corresponding u_x and u_y components, we mean that the u_x and u_y values for lattice points with the same x- and y-coordinates in both floors are considered. Condition (45) is satisfied by putting all u_z values equal to zero at floor number 10. These three additional conditions represent 80 linear equations which now supplement the set of 720 linear equations with 800 unknowns (reference the Figure 8 matrix). The 80 equations are as follows:

$$u_{n(i)+720} = u_{n(i)+640} \tag{62}$$

$$u_{m(j)+720} = 0 \tag{63}$$

where $i = 1,2...48$ and $j = 1,2...32$. Here, $n(i)$ is the sub-sequence of the first 80 elements of the [MSK] matrix that contains only x- and y-directed unknown displacement indices; $m(j)$ is the sub-sequence that contains elements of the [MSK] matrix associated with unknown z-directed displacement indices.

The unknown with indices greater than 720 appear in the last 80 equations of the system of 720 equations. Matrix [B] represents the coefficients for them. Substitution of equations (62) and (63) into these last 80 equations modifies the set of equations to the extent that matrix [B] is absorbed into matrix [C] .

We may write the last 80 equations in the following manner.

$$P_i^{(K)} - \sum_{j=1}^{M_1} K^*_{ij} u_j^{(K)} = \sum_{j=561}^{640} K^*_{ij} u_j^{(u)} +$$

$$+ \sum_{j=1}^{48} K^*_{i,640+n(j)} u^{(u)}_{640+n(j)}$$

$$+ \sum_{j=1}^{32} K^*_{i,640+m(j)} u^{(u)}_{640+m(j)}$$

$$+ \sum_{j=1}^{48} K^*_{i,720+n(j)} u^{(u)}_{720+n(j)}$$

$$+ \sum_{j=1}^{32} K^*_{i,720+m(j)} u^{(u)}_{720+m(j)} \tag{64}$$

where $i = 641, 642, \ldots 720$. In equation (63), we have the first sum term, the [B] matrix term of Figure 8. The next two terms (the [C] matrix terms), the last two terms (the [B] matrix terms), and the two separate terms for the [B] and [C] matrices correspond to the separation into x- and y-directed displacements and z-directed displacements. Now we substitute equations (61) and (62) into (63) and obtain an equation containing only displacement indices 561...720 :

$$P_i^{(K)} - \sum_{j=1}^{M_1} K^*_{ij} u_j^{(K)} = \sum_{j=561}^{640} K^*_{ij} u_j^{(u)} +$$

$$+ \sum_{j=1}^{48} \left(K^*_{i,640+n(j)} + K^*_{i,720+n(j)} \right) u^{(u)}_{640+n(j)}$$

$$+ \sum_{j=1}^{32} K^*_{i,640+m(j)} u^{(u)}_{640+m(j)} \tag{65}$$

where $i = 641, 642, \ldots 720$. The first term (corresponding to the [B] matrix and the independent term) is not altered. The second two sum terms correspond to the new matrix [C'] .

According to equation (65), matrix [C'] can be generated in the following manner. First, all column elements of the [B] matrix with the indices $m(j)$, where $j = 1,2,\ldots 32$, are zeroed out. The matrix obtained is called [B'] and is an (80 x 80) matrix. Matrix [C'] is the sum of matrices [C] and [B'] .

With the "internal" boundary conditions (61) and (62), along with the system of equations based on the matrix depicted earlier in Figure 8, we can obtain an equivalent system of 720 equations which correspond to the matrix inside the broken line of Figure 8. In this case, the sub-matrix [C] of the last line is replaced by the matrix [C'] .

Solutions for this system of equations have been obtained by applying the method of Choleski.* The accuracy of this method has been verified by computing a test case for homogeneous material, so that the stress and displacement distribution could be obtained by analytical means.

For our boundary problem, the displacements in the homogeneous test case are exactly linear. Therefore, the approximation used in this program becomes an exact solution. The displacements obtained as solutions of the system of 720 linear equations (using the Choleski method) agreed in the first 7 digits with the displacements obtained by analytical means.

This test result carries with it the implication of the correctness of the [K*] matrix coefficients. Correctness of the [K*] matrix is maintained for the fiber-matrix case, since the scheme to generate the elements of [K*] is independent of input values.

Computation of the Stress Components

The stresses for each tetrahedron can now be calculated with the use of equation (21), since all u's are now known; in other words, they are either given or are solutions of the $u_j^{(u)}$ equations.

The remaining task is to find the 12 proper displacement components in the correct order for each tetrahedron in every layer. This is accomplished in a manner similar to the way the equation indices of a [K] matrix element were obtained.

* Also known as Crout's scheme.

In Figure 10, the scheme to obtain the six stress elements (the σ matrix) for each tetrahedron is demonstrated. Given are all known displacements and the solution vector u(i) , where i = 1,2,...720 .

The meaning of the six stress components in the order appearing in equation (16) is obtainable from equation (10):

$$\sigma_1 = \sigma_x \qquad \sigma_4 = \sigma_{xy}$$

$$\sigma_2 = \sigma_y \qquad \sigma_5 = \sigma_{xz}$$

$$\sigma_3 = \sigma_z \qquad \sigma_6 = \sigma_{yz}$$

With all of the σ components obtained, the stress distribution within the three-dimensional domain can be obtained, since the coordinates of the tetrahedron centroids are the field point coordinates of this distribution approximation.

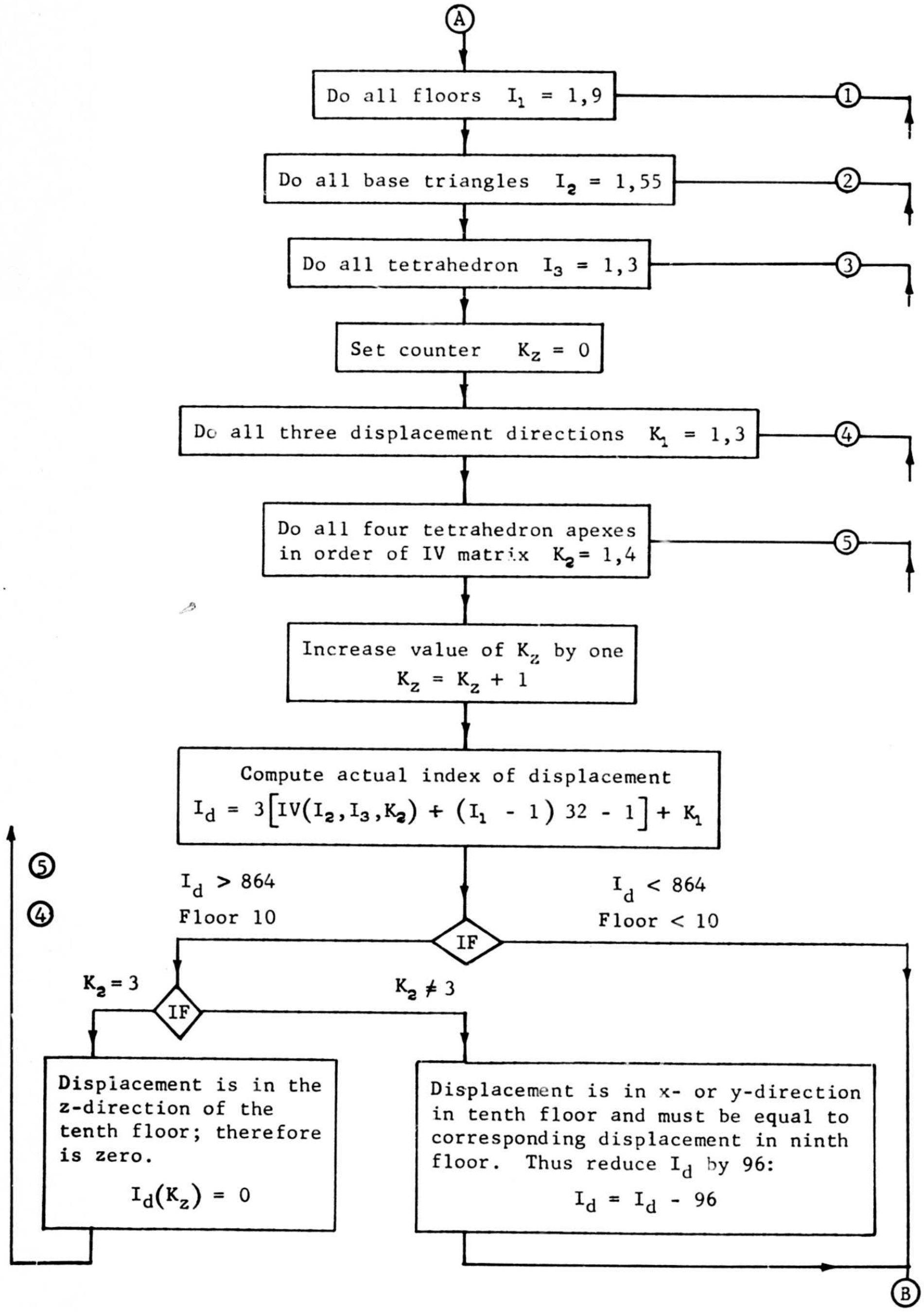
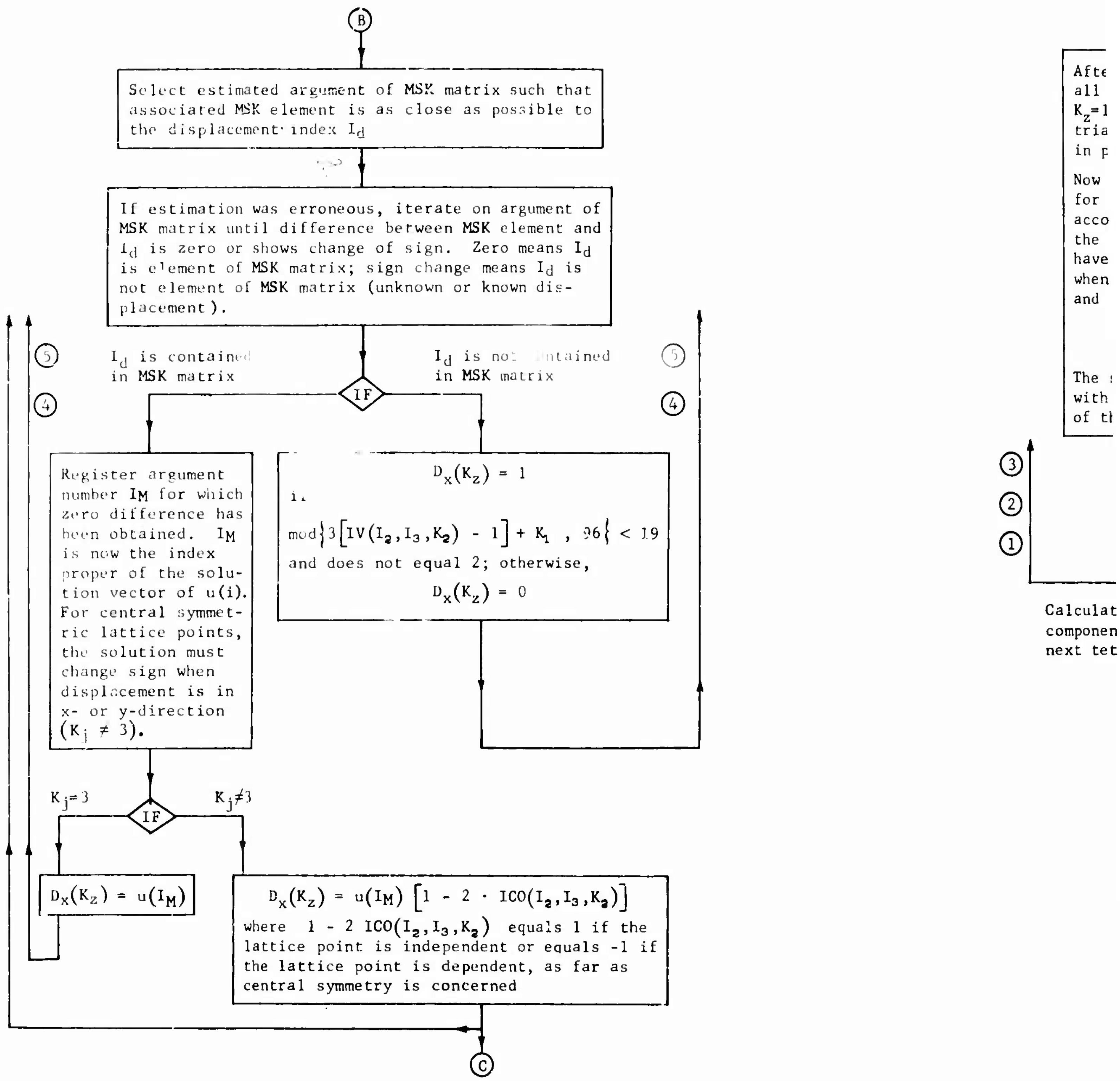


Figure 10. Computer Flow Diagram for Calculating Stress Components.



B
Select estimated argument of MSK matrix such that associated MSK element is as close as possible to the displacement index I_d
If estimation was erroneous, iterate on argument of MSK matrix until difference between MSK element and I_d is zero or shows change of sign. Zero means I_d is element of MSK matrix; sign change means I_d is not element of MSK matrix (unknown or known displacement).
I_d is contained in MSK matrix
I_d is not contained in MSK matrix
IF
5
4
Register argument number I_M for which zero difference has been obtained. I_M is now the index proper of the solution vector of u(i). For central symmetric lattice points, the solution must change sign when displacement is in x- or y-direction ($K_j \neq 3$).
$D_x(K_z) = 1$
$\mathrm{mod}\{3[IV(I_2,I_3,K_2) - 1] + K_1\ ,\ 96\} < 19$
and does not equal 2; otherwise,
$D_x(K_z) = 0$
$K_j=3$
$K_j \neq 3$
IF
$D_x(K_z) = u(I_M)$
$D_x(K_z) = u(I_M)\,[1 - 2 \cdot ICO(I_2,I_3,K_2)]$
where $1 - 2\ ICO(I_2,I_3,K_2)$ equals 1 if the lattice point is independent or equals -1 if the lattice point is dependent, as far as central symmetry is concerned
C
3
2
1

A

B

·gument of MSK matrix such that
nent is as close as possible to
ıdex I_d

·roneous, iterate on argument of
·ference between MSK element and
change of sign. Zero means I_d
ıtrix; sign change means I_d is
ıatrix (unknown or known dis-

IF

I_d is not contained in MSK matrix

5

4

$D_x(K_z) = 1$

d$\}3[IV(I_2, I_3, K_2) - 1] + K_1 \;, 96\{ < 19$
.d does not equal 2; otherwise,

$D_x(K_z) = 0$

z) $= u(I_M)\,[1 - 2 \cdot ICO(I_2, I_3, K_2)]$
1 - 2 $ICO(I_2, I_3, K_2)$ equals 1 if the
point is independent or equals -1 if
tice point is dependent, as far as
symmetry is concerned

C

C

After exhaustion of all K_1 and K_2, all 12 displacements $D_x(K_z)$, where $K_z = 1, 2, \ldots 12$, of tetrahedron I_3 of triangle I_2 at Floor I are obtained in proper order.

Now we can calculate the stresses for this particular tetrahedron according to equation (16), since the matrices $[C][D][A]^{-1} = \{CDA\}$ have been obtained as a by-product when matrices $\{K\}$ were calculated and then stored.

$$\sigma = \}CDA\{\}D_x\{$$

The six σ values are printed along with the coordinates of the centroid of the tetrahedron

3

2

1

Calculate stress components $\{\sigma\}$ for next tetrahedron

NUMERICAL RESULTS FOR THE THREE-DIMENSIONAL SOLUTION

Figure 11 depicts a cross section of a composite under transverse loading. Figures 12 through 20 illustrate the variation of the stresses in the fiber direction at the interface points, indicated in Figure 11, in regions near the free end of the composite. These stresses, which appear because of the difference of elastic constants in the fiber and the resin, are similar to the stresses obtained with the plane stress assumption for planes far from the free end.

The three-dimensional analysis shows the existence of shear stresses σ_{xz} and σ_{yz} of considerable magnitude at the interface points depicted in Figure 11. In fact, these shear stresses are about one-half the peak shear stress σ_{xy} from the plane analysis. The numbers have been derived from numerical calculations of the stresses within the area of triangle 41 (see Figure 5).

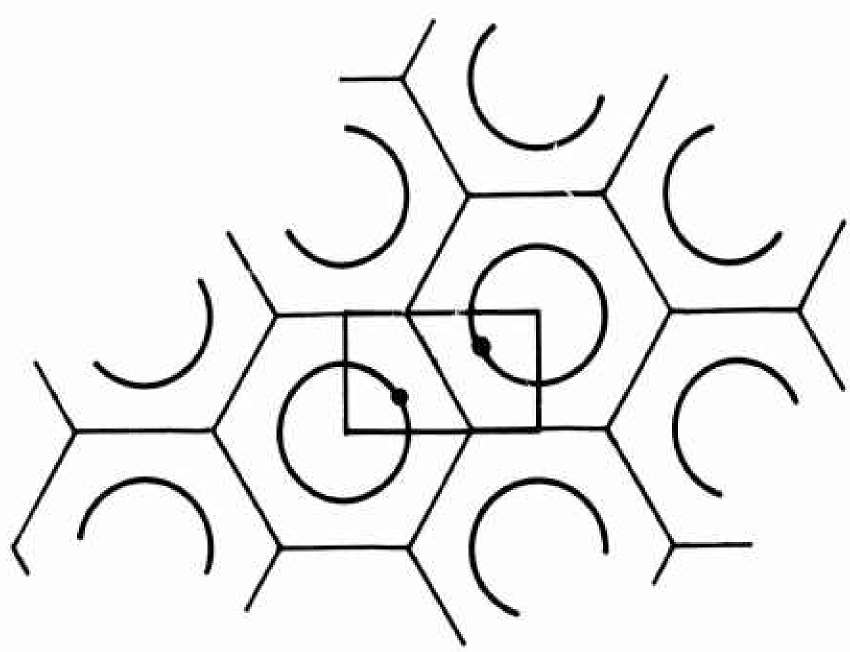


Figure 11. Cross Section of a Composite Under Transverse Loading.

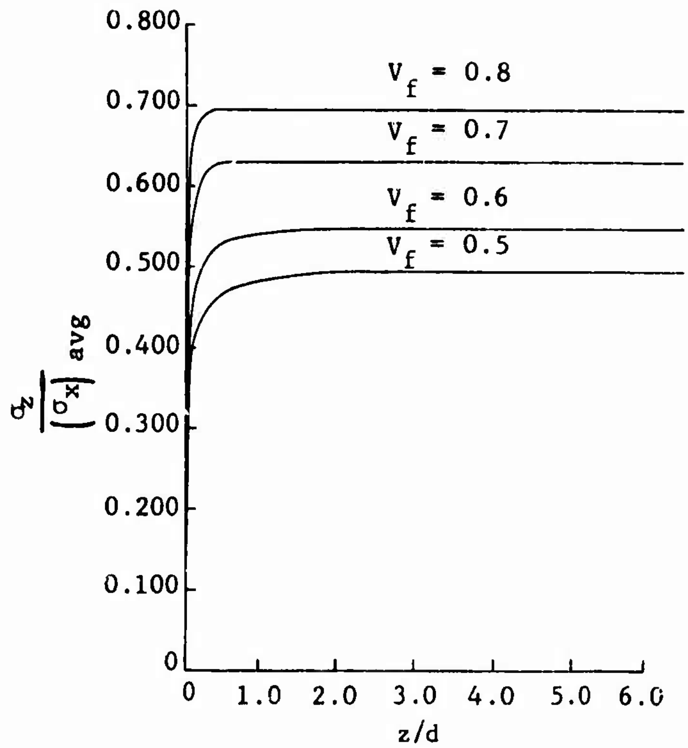


Figure 12. Longitudinal Stresses σ_z at Location Indicated in Figure 11 Due to Transverse Normal Loading σ_x Average.

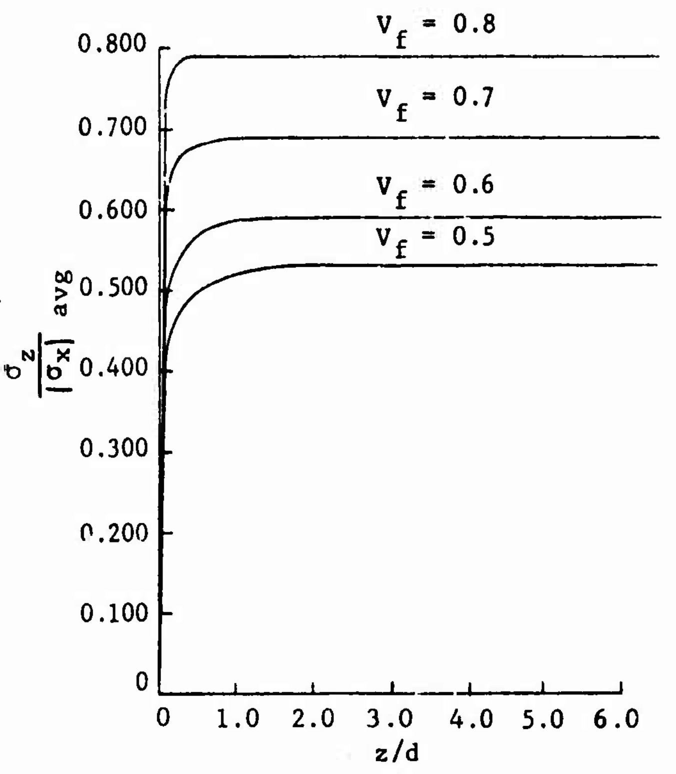


Figure 13. Longitudinal Stresses σ_x at Location Indicated in Figure 11 Due to Transverse Normal Loading σ_x Average.

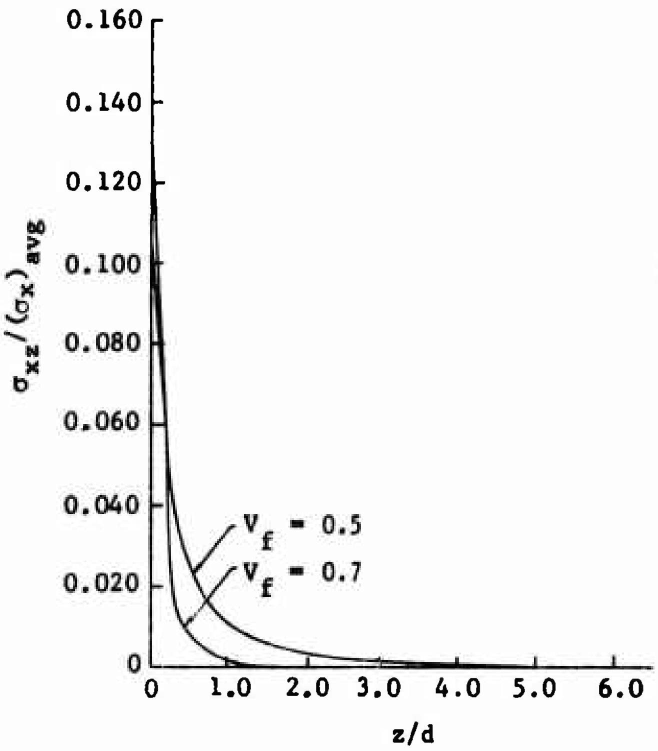
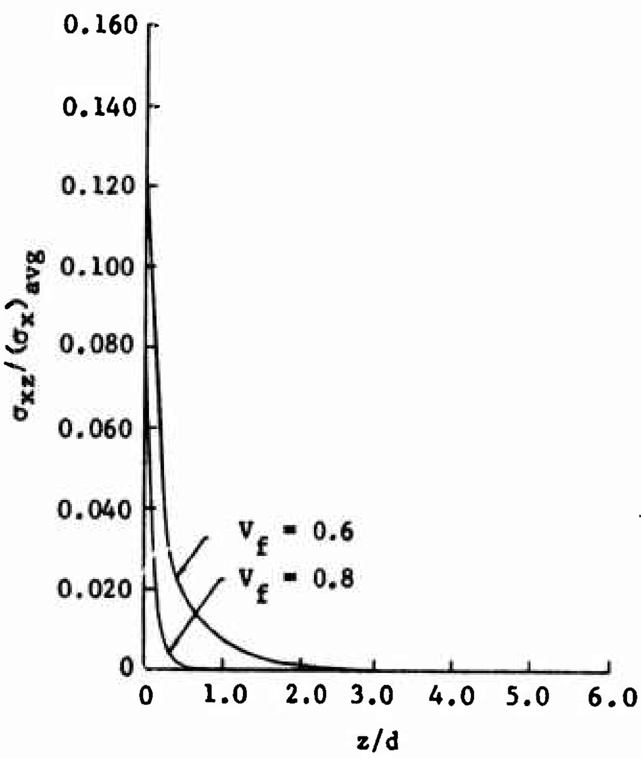


Figure 14. Longitudinal Shear Stresses σ_{xz} at Location Indicated in Figure 11 Due to Transverse Normal Loading σ_x Average.

Figure 15. Longitudinal Shear Stresses σ_{xz} at Lpaction Indicated in Figure 11 Due to Transverse Normal Loading σ_x Average.



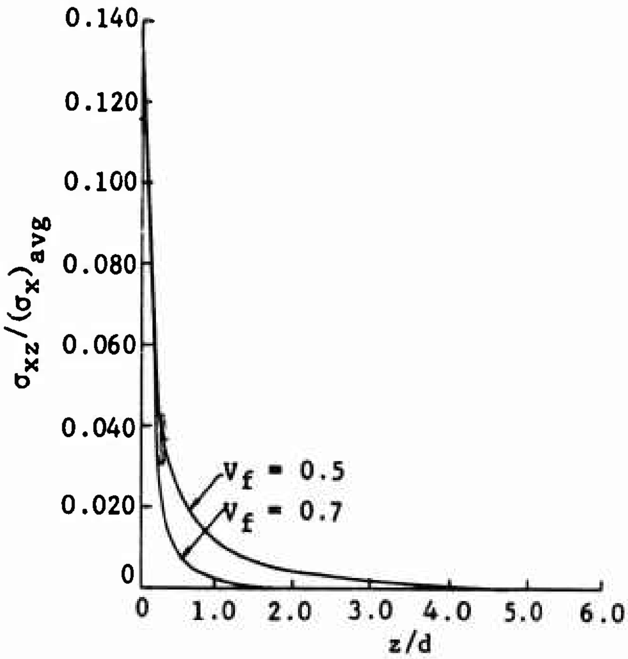
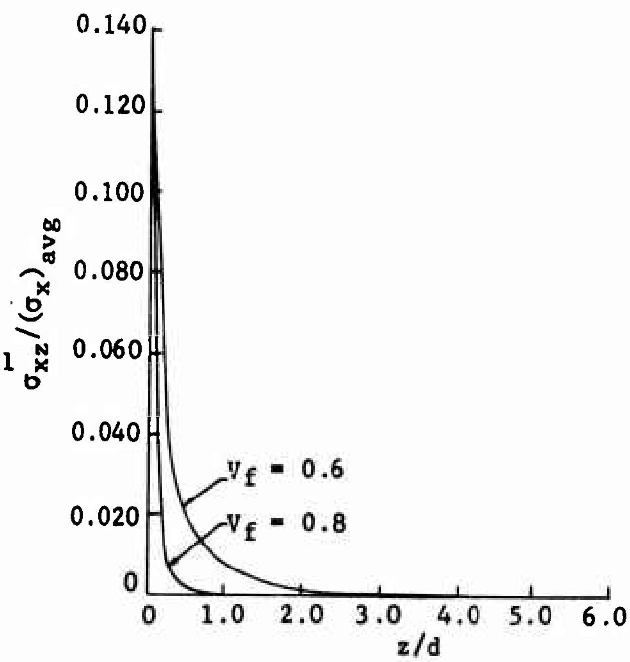


Figure 16. Longitudinal Shear Stresses σ_{xz} at Location Indicated in Figure 11 Due to Transverse Normal Loading Average σ_x.

Figure 17. Longitudinal Shear Stresses σ_{xz} at Location Indicated in Figure 11 Due to Transverse Normal Loading σ_x Average.



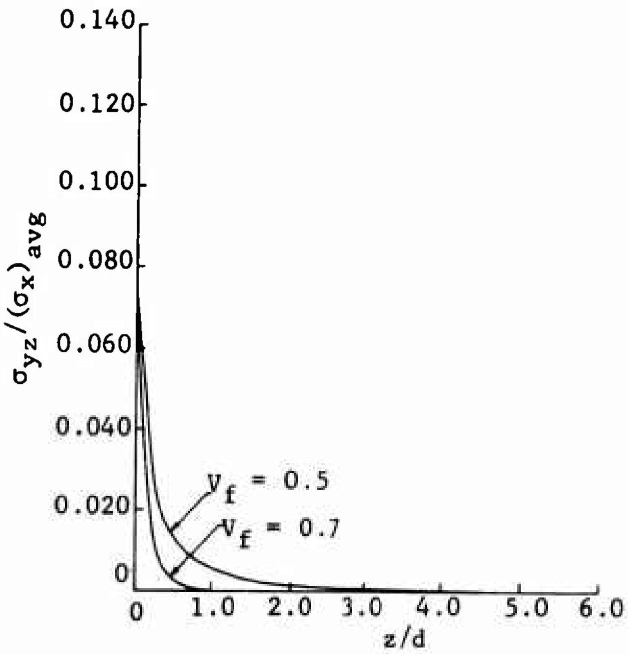
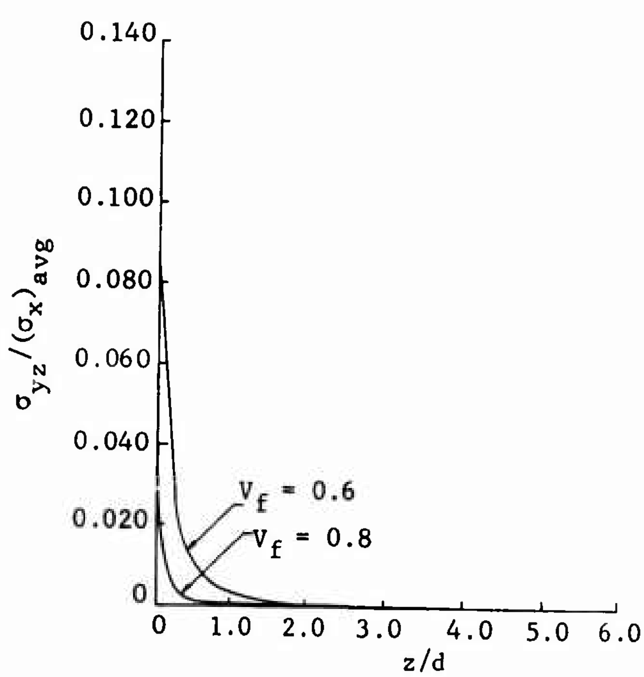


Figure 18. Longitudinal Shear Stresses σ_{xz} at Location Indicated in Figure 11 Due to Transverse Normal Loading σ_x Average.

Figure 19. Longitudinal Shear Stresses σ_{xz} at Location Inidcated in Figure 11 Due to Transverse Normal Loading σ_x Average.



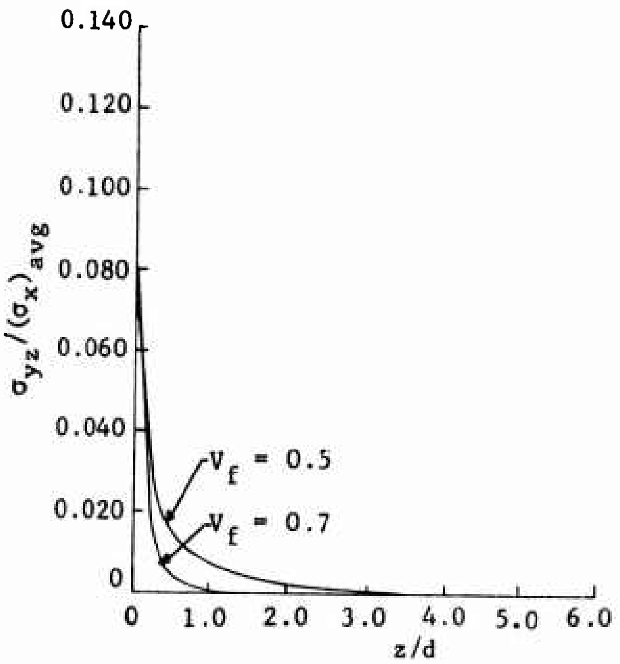
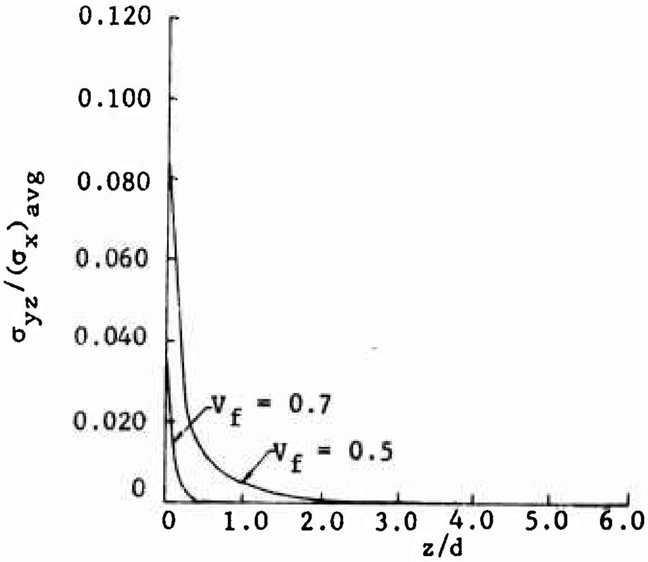


Figure 20. Longitudinal Shear Stresses σ_{xz} at Location Indicated in Figure 11 Due to Transverse Normal Loading σ_x Average.

Figure 21. Longitudinal Shear Stresses σ_{xz} at Location Indicated in Figure 11 Due to Transverse Normal Loading σ_x Average.



SOLUTIONS OF THE PLANE PROBLEMS

In regions of the composite under transverse loading that are far from the ends (see Figure 22), it is possible to realistically assume plane strain or plane stress. It will be plane strain if the displacement u_z is constrained at the ends $z = 0$ and $z = \ell$; it will be plane stress if the ends are free of loads.

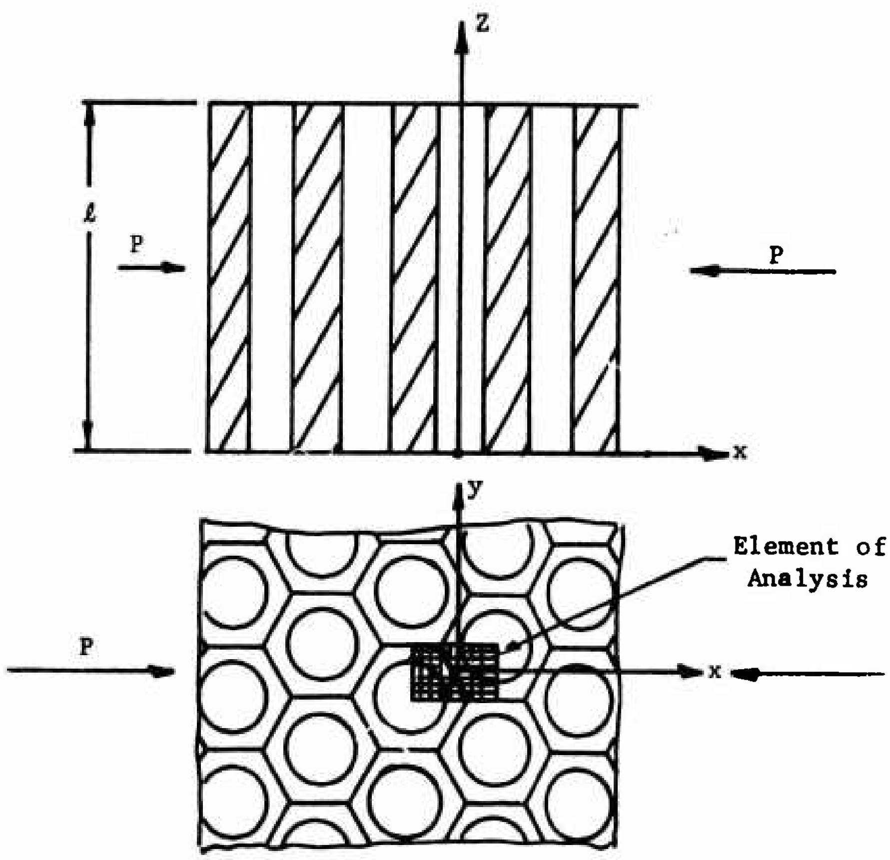


Figure 22. Composite Under Transverse Load.

It is assumed that the transverse loads P are applied to the composite far from the chosen representative element of analysis. Also, it is assumed that the composite cannot have expansions in the y-direction; in other words, compared with the dimensions of the typical element, the length of the composite is infinite in the y-direction.

With these assumptions, the boundary conditions for the representative element (Figure 23) are:

$$\left(u_x\right)_{x=\pm c} = \text{constant} = \pm k$$

$$\left(\sigma_{xy}\right)_{x=\pm c} = 0$$

$$\left(u_y\right)_{y=\pm\frac{b}{2}} = 0 \qquad (66)$$

$$\left(\sigma_{xy}\right)_{y=\pm\frac{b}{2}} = 0$$

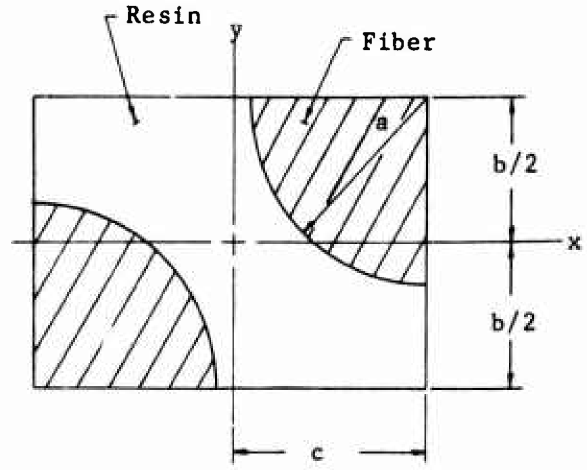


Figure 23. Representative Element.

At the interface, between the fiber and matrix, the following continuity conditions must be satisfied:

$$u_x^f = u_x^m \quad , \quad u_y^f = u_y^m$$

$$\sigma_n^f = \sigma_n^m \quad , \quad \sigma_{nt}^f = \sigma_{nt}^m \qquad (67)$$

The superscript f indicates fiber and the superscript m indicates matrix. The n and t are the normal and tangential directions at the interface points, respectively.

Moreover, a symmetry condition exists for the displacements and, consequently, for the stresses of the representative element. That is,

$$u_x(x,y) = -u_x(-x,-y)$$

$$u_y(x,y) = -u_y(-x,-y) \qquad (68)$$

The stress distribution was solved using the finite element method with the triangular net shown in Figure 24. This was explained in detail for the three-dimensional solutions in the last section.

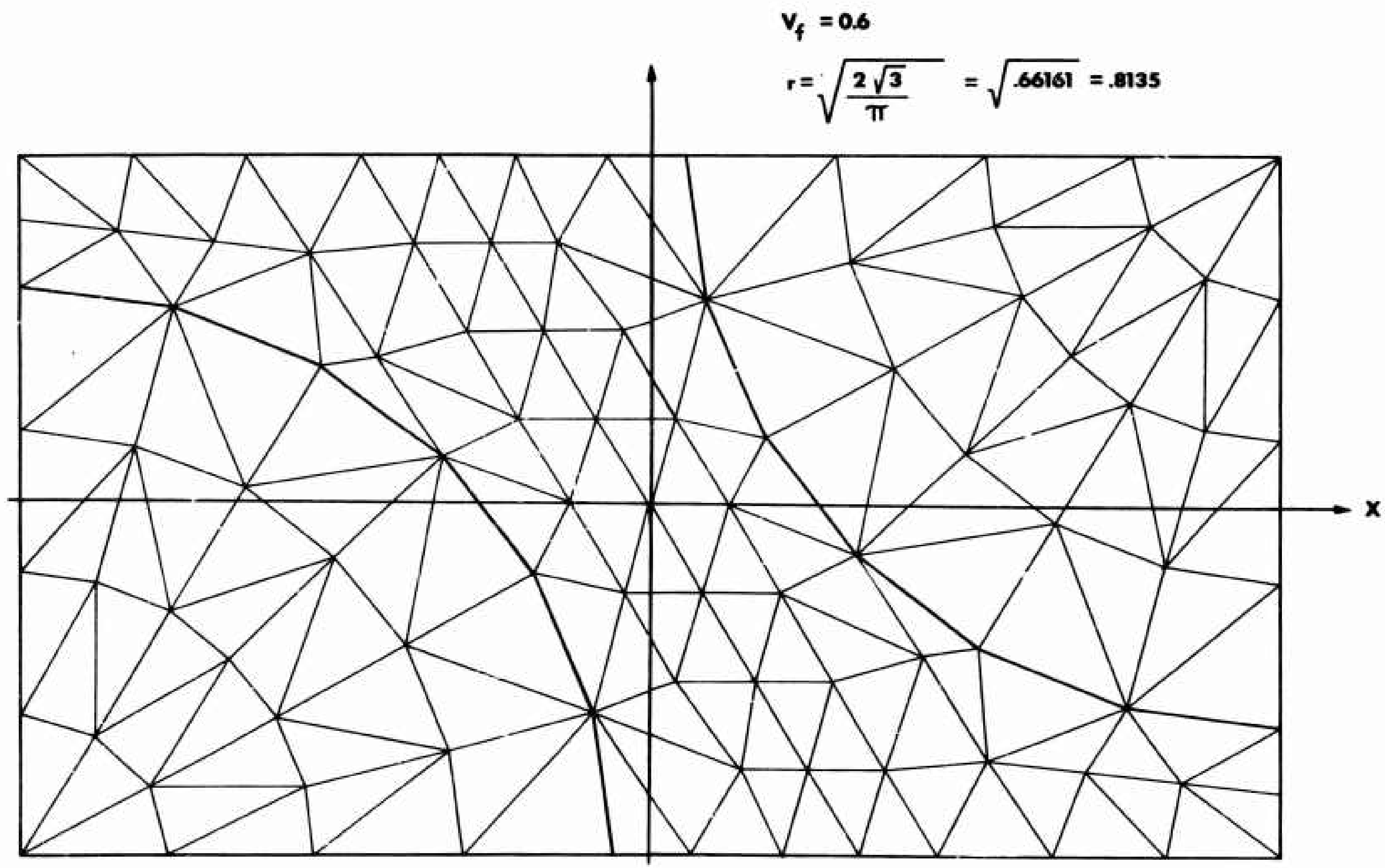


Figure 24. Finite Elements of a Basic Representative Cross-Sectional Area of Floor Number 1 Used in Two-Dimensional Numerical Analysis.

Once the stress distribution is known, the transverse modulus of the composite E_T can be found. In fact,

$$E_T = \frac{\sigma_c}{\epsilon} = \frac{P/b}{k/c}$$

where

$$c = \sqrt{3}\ \frac{b}{2}$$

Then,

$$E_T = \frac{\sqrt{3}}{2} \cdot \frac{P}{k} \tag{69}$$

The total applied force P is obtained by adding the forces at the x direction (σ_x · dimension at the y direction) for the triangles bordering the boundary $x = c$.

By divising equation (69) setting $k = 1$ by the modulus of the fiber E_f , the nondimensional expression is obtained.

$$\chi = \frac{E_T}{E_f} = \frac{3}{2} \cdot \frac{P}{E_f} \tag{70}$$

In the following section, numerical results of the two-dimensional analysis are given.

NUMERICAL RESULTS OF THE PLANE PROBLEMS AND TEST RESULTS

The trajectories of the principal stresses in the typical element of a composite material under transverse loading are represented in Figure 25. The nonhomogeneity introduced by the presence of the fiber causes a deviation in the trajectories with respect to the homogeneous case where the trajectories are obviously parallel to the x and y axes. This fact suggests the presence of stress concentrations, which will be demonstrated effectively in the following discussion.

A parametric study was performed to evaluate the influence of the matrix and fiber properties on the stress distribution. Figure 26 shows the radial and tangential stresses along the interface for composites with 60 percent of the fibers containing the same matrix material but with different fiber material. One composite has a modulus relationship of $E_f/E_m = 20$, and the other has a relationship of $E_f/E_m = 120$. These correspond approximately to glass fiber and boron fiber composites with epoxy resin, respectively. From the curves of the figure, it is possible to deduct the slight influence the material of the fiber has in the stress distribution. This conclusion is correct in all cases in which the fiber is considerably harder in comparison with the matrix, as usually occurs in most of the composites.

The nondimensional ordinates σ_r/σ_{avg} and $\tau_{r\theta}/\sigma_{avg}$ of the curves were obtained by dividing the actual stresses σ_r and $\tau_{r\theta}$ by the average stress σ_{avg} found by averaging the stresses σ_x at the triangles 3, 12, 24, 36, 44 and 49.

Another parametric study was performed by taking the volumetric content as variable, keeping the modulus relationship E_f/E_m constant. The results are indicated in Figures 27 and 28, where the stresses along the interface are plotted. As is easy to imagine, the peak stresses are greater when V_F is increased.

Figure 29 indicates the displacement component at the x-direction for composites with $E_f/E_m = 20$ and 120. In these curves it is possible to observe the small influence of the fiber deformation in comparison with the total deformation, which is carried almost completely by the matrix.

TRANSVERSE MODULUS

A parametric study was performed to compute the transverse modulus of several composite types. Two different conditions were considered: (1) plane stress on the x,y plane, and (2) plane strain on the same plane.

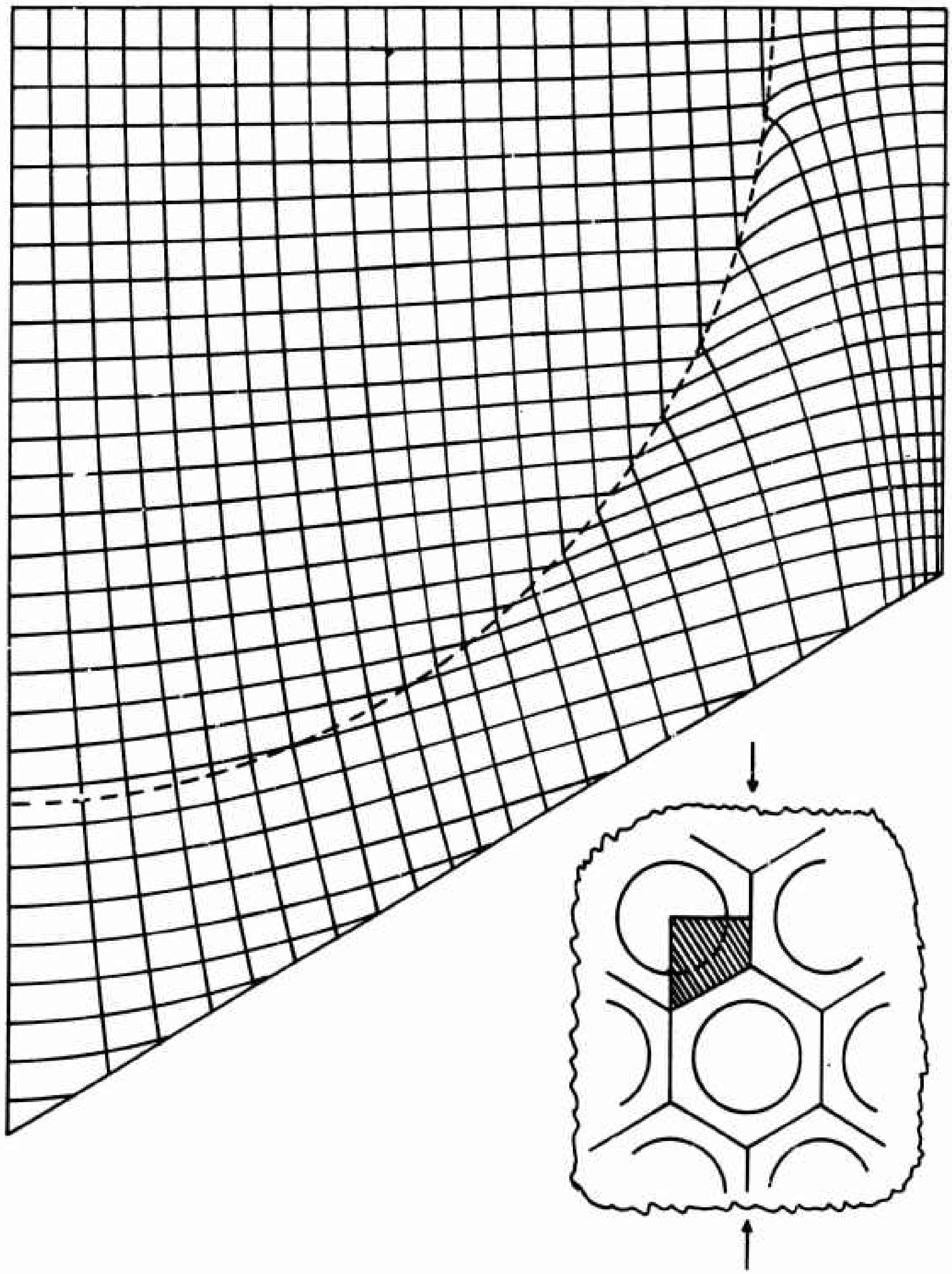


Figure 25. Stress Trajectories of Transverse Loading.

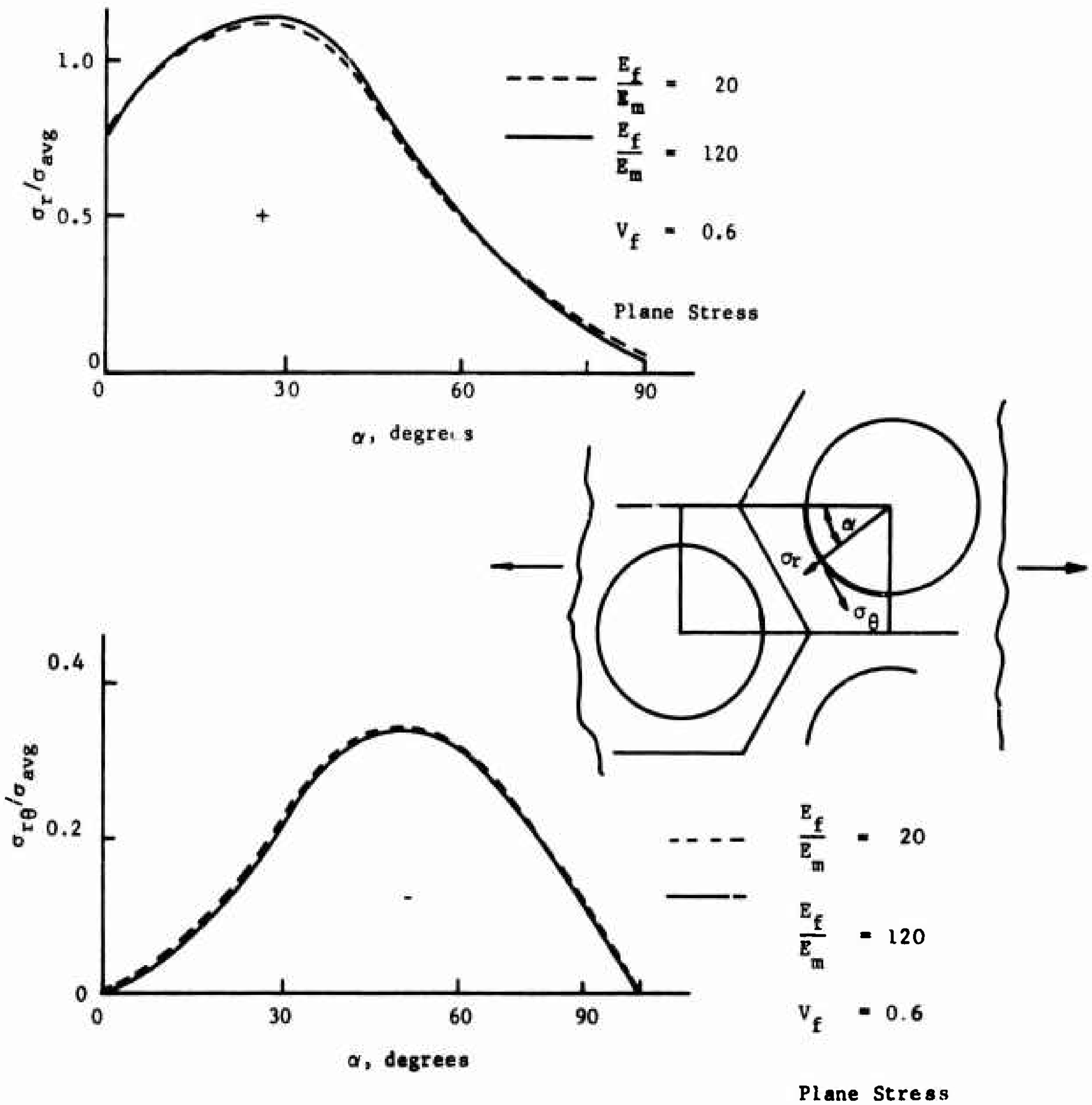


Figure 26. Radial and Tangential Stresses
Along the Interface.

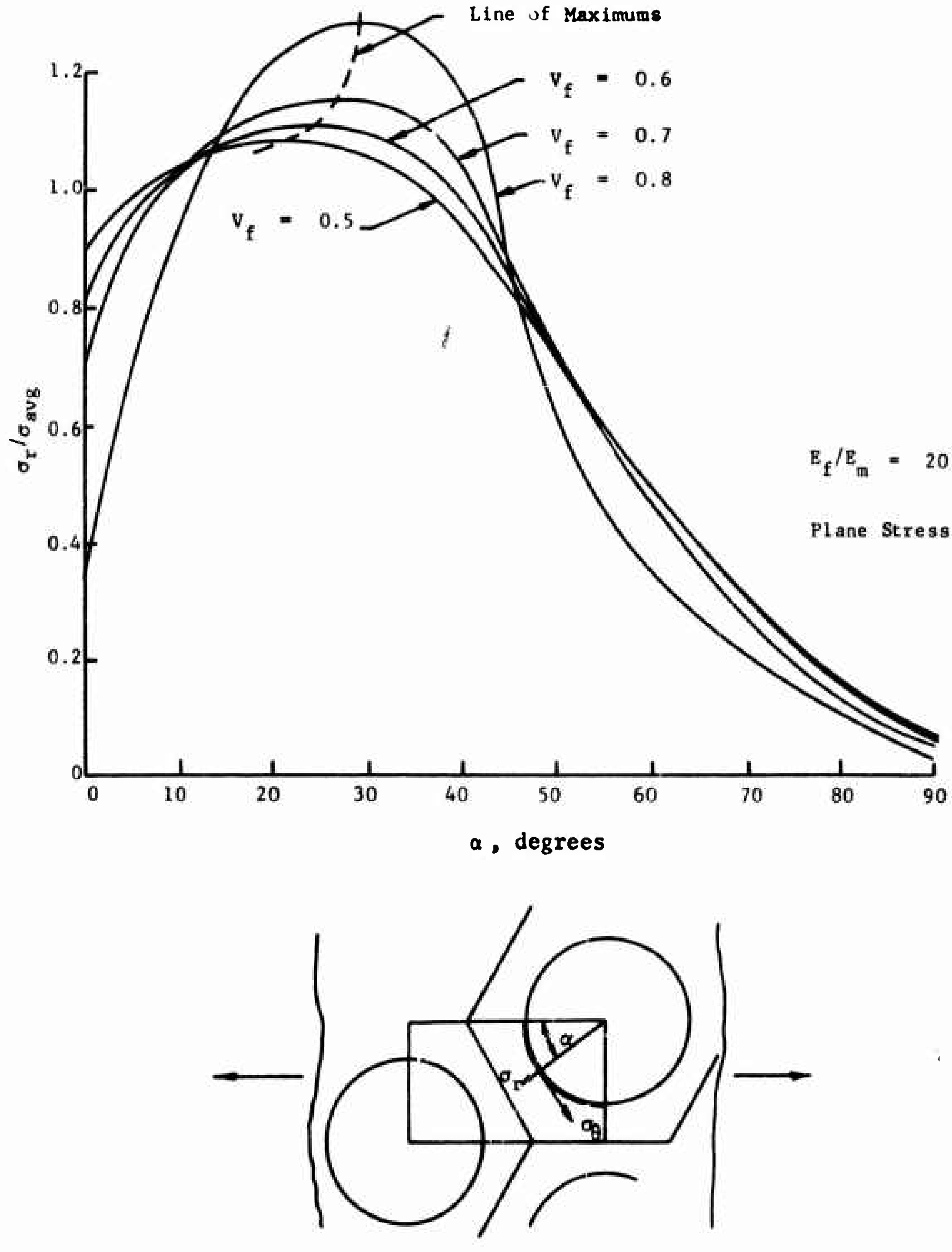


Figure 27. Stresses Along the Interface With Volumetric Content Variable and Modulus Relationship Constant.

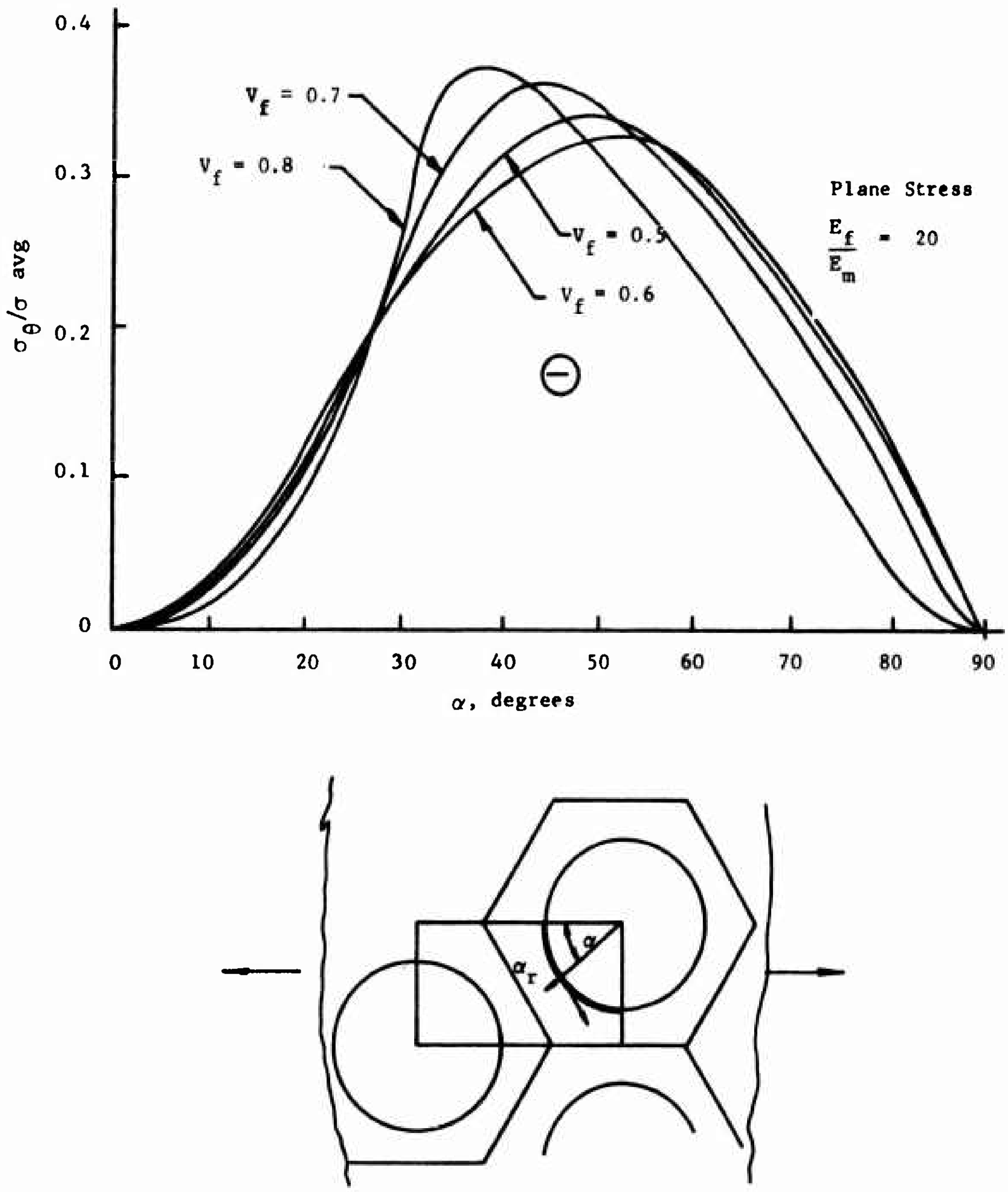
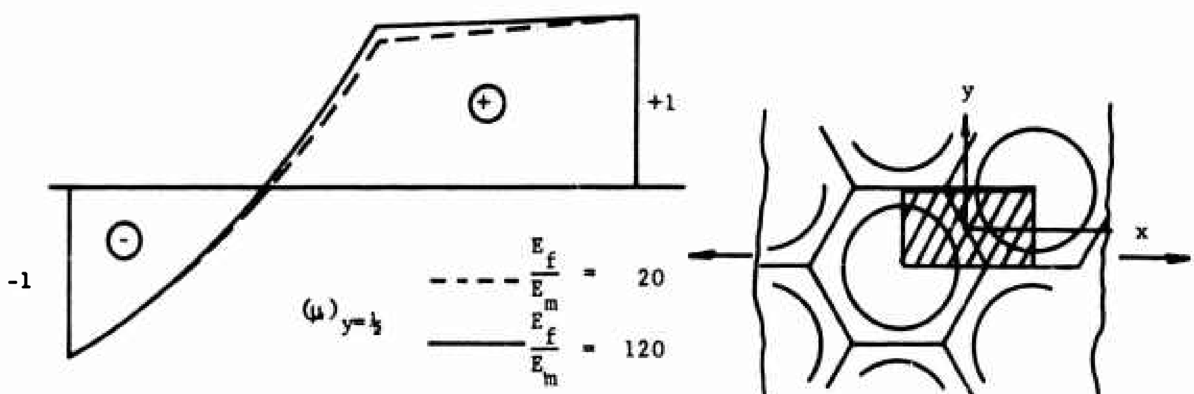
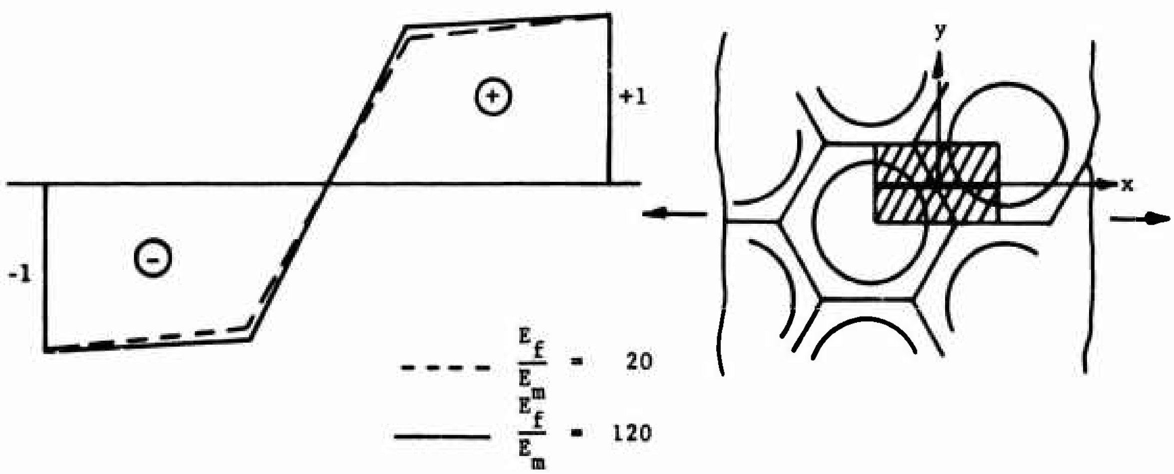


Figure 28. Stresses Along the Interface With Volumetric Content Variable and Modulus Relationship Constant.



Plane Stress



Plane Stress

Figure 29. Displacement Component at the x-Direction.

Figure 30 gives the transverse modulus of the composite E_T for a plane stress condition. Figure 31 gives the modulus of the composite for plane strain as a function of the modulus ratio E_f/E_m, and by taking the volumetric content as parameter. It can be appreciated that the influence of the fiber modulus on the composite modulus is small, in accordance with the displacement distribution discussed previously.

Figure 32 gives a comparison of the transverse modulus obtained with Ekvall's formula,

$$E_T = \frac{E_f E_m}{V_f E_m + V_m E_f} , \tag{71}$$

and the results from the computer analysis.

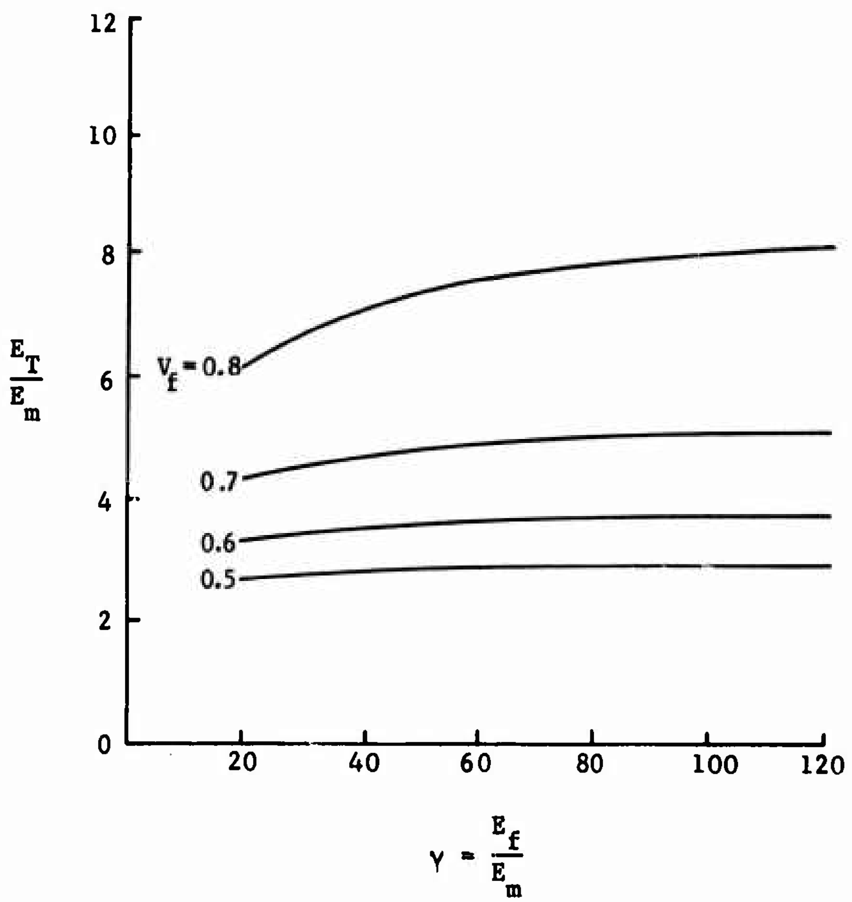


Figure 30. Transverse Modulus of a Composite as a Function of Fiber and Matrix Modulus and Volume Percentage.

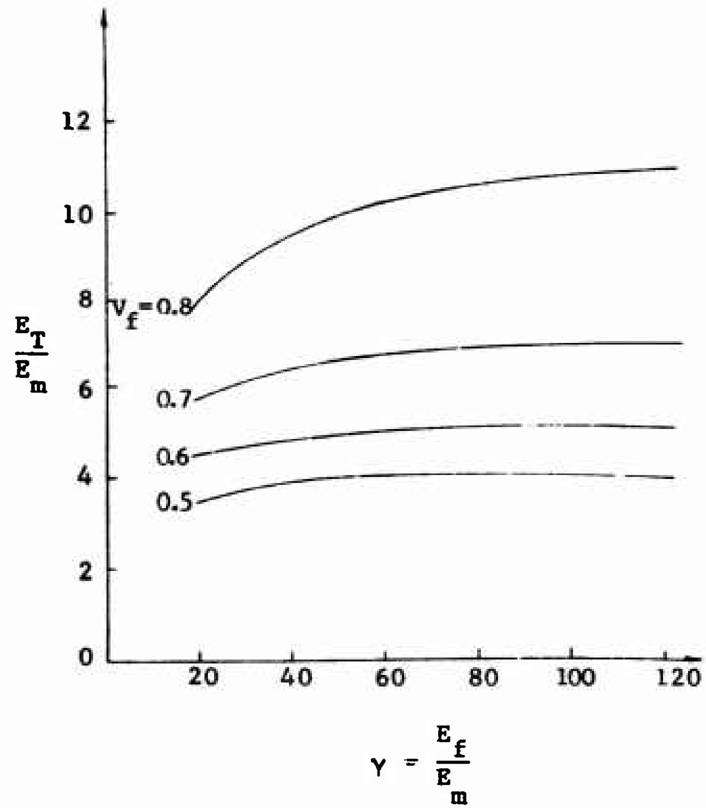


Figure 31. Transverse Modulus of a Composite as a Function of Fiber and Matrix Modulus and Volume Percentage.

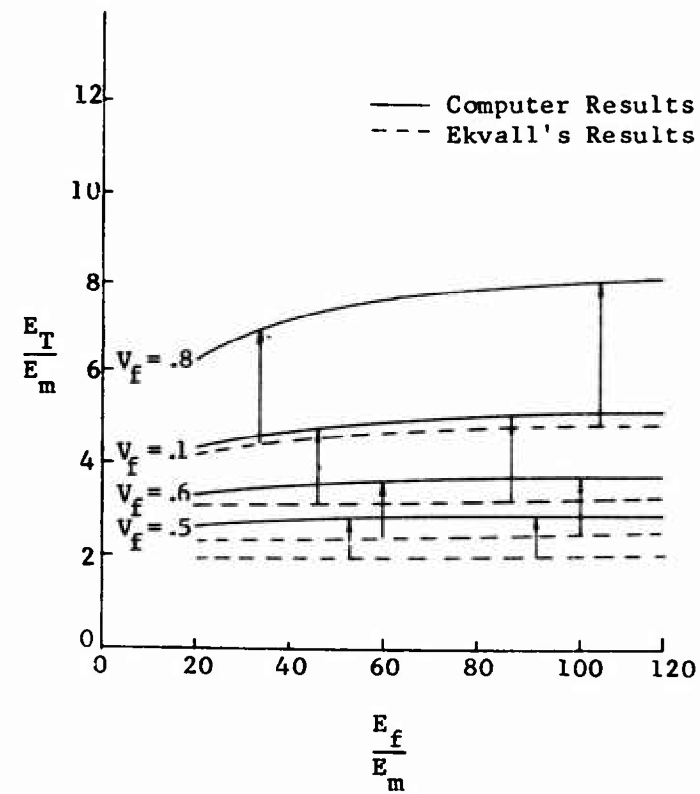


Figure 32. Comparison of Transverse Modulus Obtained With Ekvall's Formula and Computer Results.

Figure 33 presents the comparison with respect to the Shaffer's formula,

$$E_T = E_m \frac{1 - \left(1 - \frac{E_m}{E_f}\right)\left(0.8247\sqrt{V_f} - V_f\right)}{1 - 0.8247\, V_f\left(1 - \frac{E_m}{E_f}\right)} \quad \text{for } V_f<0.68 \qquad (72)$$

and

$$E_T = \frac{E_m}{1 - V_f\left(1 - \frac{E_m}{E_f}\right)} \quad \text{for } V_f \geq 0.68 \qquad (73)$$

As observed in the figures, the transverse modulus from the micromechanical analysis is greater than the corresponding modulus in the Ekvall and Shaffer formulas, especially for the higher volumetric contents.

The transverse modulus of a composite obtained by plane stress analysis was presented in Figure 30 for different volumes and material contents

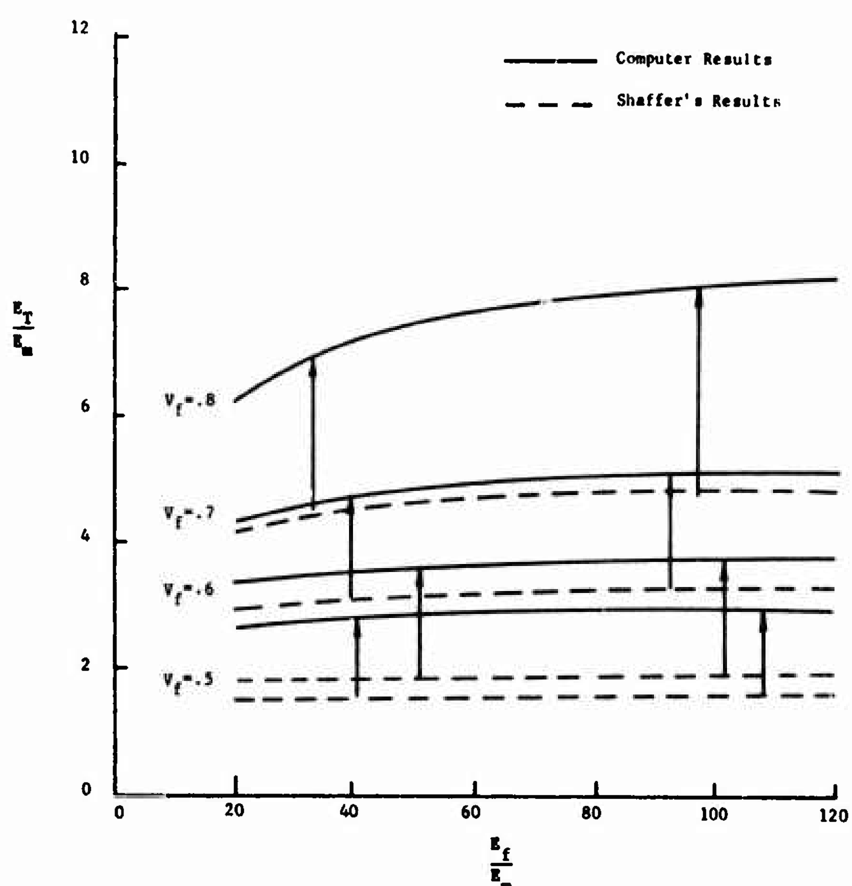


Figure 33.

Comparison of Transverse Modulus Obtained With Shaffer's Formula and Computer Results.

of the reinforcement and the matrix. This modulus can be approximated by the following formula:

$$\frac{E_T}{E_m} = \frac{\frac{E_f}{E_m}\left(1+V_f\right) + \left(1-V_f\right)}{\frac{E_f}{E_m}\left(1-V_f\right) + \left(1+V_f\right)} \tag{74}$$

It is interesting to note that this formula is identical to Rosen's formula* for longitudinal shear. The difference is that instead of the G moduli, the E moduli of the different materials must be used as indicated in equation (74). Allowing for little error for low fiber percentages and $E_f/E_m < 100$, this formula has an error showing a 4 percent higher transverse modulus at higher values. The following is a formula which permits errors of less than 1 percent for all possible combinations.

$$\frac{E_T}{E_m} = \left(11\cdot 78V_f^2 - 13V_f + 3\cdot 78\right)\ln\left(\frac{E_f}{E_m}\right) + 6\cdot 666V_f - 1\cdot 27 \tag{75}$$

* B. W. Rosen, N. F. Don, and Z. Hashin, Mechanical Properties of Fibrous Composites, NASA CR-31, Contract NAS-470, General Electric Corporation, April 1964.

APPENDIX I

PROBLEM OF FIBER-REINFORCED COMPOSITE SUBJECTED TO TRANSVERSE LOADING SOLVED BY POINT-MATCHING METHOD

When the weight is the only body force, the plane strain problem can be solved by finding the polynomial solution of the differential equation

$$\frac{\partial^4 \theta}{\partial x^4} + 2\,\frac{\partial^4 \theta}{\partial x^2 \partial y^2} + \frac{\partial^4 \theta}{\partial y^4} = 0 \tag{76}$$

which satisfies the boundary conditions. In equation (76), θ is the Airy stress function. Stresses and displacements are then defined as follows:

$$\sigma_x = \frac{\partial^2 \theta}{\partial y^2} \tag{77}$$

$$\sigma_y = \frac{\partial^2 \theta}{\partial x^2} \tag{78}$$

$$\sigma_{xy} = -\frac{\partial^2 \theta}{\partial x\, \partial y} \tag{79}$$

$$u_x = \frac{(1+\nu)}{E}\left[\int (1-\nu)\,\frac{\partial^2 \theta}{\partial y^2}\,dx - \nu\,\frac{\partial \theta}{\partial x}\right] + f_1(y) \tag{80}$$

$$u_y = \frac{(1+\nu)}{E}\left[\int (1-\nu)\,\frac{\partial^2 \theta}{\partial x^2}\,dy - \nu\,\frac{\partial \theta}{\partial y}\right] + f_2(x) \tag{81}$$

The strains are

$$\epsilon_x = \frac{\partial u_x}{\partial x} = \frac{1+\nu}{E}\left[(1-\nu)\,\frac{\partial^2 \theta}{\partial y^2} - \nu\,\frac{\partial^2 \theta}{\partial x^2}\right] \tag{82}$$

$$\epsilon_y = \frac{\partial u_y}{\partial y} = \frac{1+\nu}{E}\left[-\nu\,\frac{\partial^2 \theta}{\partial y^2} + (1-\nu)\,\frac{\partial^2 \theta}{\partial x^2}\right] \tag{83}$$

$$\epsilon_{xy} = \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} - \frac{2(1+\nu)}{E}\,\frac{\partial^2 \theta}{\partial x \partial y} \tag{84}$$

From equations (80), (81), and (84), we have

$$\int \frac{\partial^3\theta}{\partial y^3}\,dx + \int \frac{\partial^3\theta}{\partial x^3}\,dy + 2\,\frac{\partial^2\theta}{\partial x\partial y} + \frac{E}{1-\nu^2}\left(\frac{df_1}{dy} + \frac{df_2}{dx}\right) = 0 \qquad (85)$$

The polynomial solution to the Laplace equation (harmonic equation) is

$$(x+iy)^N = \sum_{n'=0}^{N} \binom{N}{n'} (iy)^{n'} x^{N-n'}$$

$$= \sum_{n=0}^{N} \binom{N}{2n} (iy)^{2n} x^{N-2n} + \sum_{n=0}^{N} \binom{N}{2n+1} (iy)^{2n+1} x^{N-2n-1}$$

$$= \sum_{n=0}^{N} \binom{N}{2n} (-1)^n y^{2n} x^{N-2n} + i \sum_{n=0}^{N} \binom{N}{2n+1} (-1)^n y^{2n+1} x^{N-2n-1}$$

$$= \Phi_N(x,y) + i\,\chi_N(x,y) \qquad (86)$$

Therefore, the solution to the biharmonic equation, in region f(fiber), is as follows:

$$\theta^f = K_1xy^3 + K_2xy^2 + K_3xy + K_4x + K_5y^3 +$$

$$K_6y^2 + K_7y + K_8x^3y + K_9x^2y + K_{10}x^3 + K_{11}x^2$$

$$\sum_{m=2}^{M} A_m \sum_{n=0}^{m} \binom{2m}{2n} (-1)^n y^{2n} x^{2m-2n+1} +$$

$$\sum_{m=2}^{M} B_m \sum_{n=0}^{m} \binom{2m}{2n} (-1)^n y^{2n+1} x^{2m-2n} +$$

$$\sum_{m=2}^{M} C_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n y^{2n+1} x^{2m-2n+1} +$$

$$+ \sum_{m=1}^{M} D_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n \, y^{2n+2} \, x^{2m-2n} \qquad (87)$$

By substituting θ from equation (87) into equations (77) through (81), the following stresses and displacements are obtained:

$$\sigma_x^f = 6K_1 xy + 2K_2 x + 6K_5 y + 2K_6 +$$

$$\sum_{m=2}^{M} A_m \sum_{n=0}^{m} \binom{2m}{2n} (2n)(2n-1)(-1)^n \, y^{2(n-1)} \, x^{2m-2n+1} +$$

$$\sum_{m=2}^{M} B_m \sum_{n=0}^{m} \binom{2m}{2n} (2n+1)(2n)(-1)^n \, y^{2n-1} \, x^{2m-2n} +$$

$$\sum_{m=2}^{M} C_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (2n+1)(2n)(-1)^n \, y^{2n-1} \, x^{2m-2n+1} +$$

$$\sum_{m=1}^{M} D_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (2n+2)(2n+1)(-1)^n \, y^{2n} \, x^{2m-2n} \qquad (88)$$

$$\sigma_y^f = 6K_8 xy + 2K_9 y + 6K_{10} x + 2K_{11} +$$

$$\sum_{m=2}^{M} A_m \sum_{n=0}^{m} \binom{2m}{2n} (2m-2n+1)(2m-2n)(-1)^n \, y^{2n} \, x^{2m-2n-1} +$$

$$\sum_{m=2}^{M} B_m \sum_{n=0}^{m} \binom{2m}{2n} (2m-2n)(2m-2n-1)(-1)^n \, y^{2n+1} \, x^{2m-m-2} +$$

$$\sum_{m=2}^{M} C_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (2m-2n+1)(2m-2n)(-1)^n \, y^{2n+1} \, x^{2m-2n-1} +$$

(Continued)

$$+ \sum_{m=1}^{M} D_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (2m-2n)(2m-2n-1)(-1)^n \; y^{2n+2} \; x^{2m-2n-2} \tag{89}$$

$$\sigma_{xy}^{f} = -3K_1 y^2 - 2K_2 y - K_3 - 3K_8 x^2 - 2K_9 x -$$

$$\sum_{m=2}^{M} A_m \sum_{n=0}^{m} \binom{2m}{2n} (2n)(2m-2n+1)(-1)^n \; y^{2n-1} \; x^{2m-2n} -$$

$$\sum_{m=2}^{M} B_m \sum_{n=0}^{m} \binom{2m}{2n} (2n+1)(2m-2n)(-1)^n \; y^{2n} \; x^{2m-2n-1} -$$

$$\sum_{m=2}^{M} C_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (2n+1)(2m-2n+1)(-1)^n \; y^{2n} \; x^{2m-2n} -$$

$$\sum_{m=1}^{M} D_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (2n+2)(2m-2n)(-1)^n \; y^{2n+1} \; x^{2m-2n-1} \tag{90}$$

$$u_x^f = \frac{1+\nu_f}{E_f} \cdot \Big\{ K_1 \Big[3\left(1-\nu_f\right) x^2 y - \left(2-\nu_f\right) y^3 \Big] + K_2 \Big[\left(1-\nu_f\right) x^2 -$$

$$\left(2-\nu_f\right) y^2 \Big] - K_3 \left(2-\nu_f\right) y - K_4 \nu_f + 6K_8 \left(1-\nu_f\right) xy +$$

$$2K_8 \left(1-\nu_f\right) x - K_8 \Big[\left(1-\nu_f\right) y^3 + 3\nu_f x^2 y \Big] - K_9 \left(2\nu_f xy\right) -$$

$$K_{10} \Big[3\left(1-\nu_f\right) y^2 + 3\nu_f x^2 \Big] - 2K_{11} x \nu_f + W_0 y \left(1-\nu_f\right) \Big\} +$$

$$\frac{1+\nu_f}{E_f} \left\{ \sum_{m=2}^{M} A_m \sum_{n=0}^{m} \binom{2m}{2n} (-1)^n \Big[\left(1-\nu_f\right) \frac{(2n)(2n-1)}{2m-2n+2} \cdot \right.$$

$$\left. y^{2(n-1)} \; x^{2m-2n+2} - \nu_f (2m-2n+1) \; y^{2n} \; x^{2m-2n} \Big] + \right.$$

$$+ \sum_{m=2}^{M} B_m \sum_{n=0}^{m} \binom{2m}{2n} (-1)^n \left[\left(1-\nu_f\right) \frac{(2n+1)(2n)}{2m-2n+1} \cdot \right.$$

$$\left. y^{2n-1} x^{2m-2n+1} - \nu_f (2m-2n) y^{2n+1} x^{2m-2n-1} \right] +$$

$$\sum_{m=2}^{M} C_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n \left[\left(1-\nu_f\right) \frac{(2n+1)(2n)}{2m-2n+2} \cdot \right.$$

$$\left. y^{2n-1} x^{2m-2n+2} - \nu_f (2m-2n+1) y^{2n+1} x^{2m-2n} \right] +$$

$$\sum_{m=1}^{M} D_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n \left[\left(1-\nu_f\right) \frac{(2n+2)(2n+1)}{2m-2n+1} \cdot \right.$$

$$\left.\left. y^{2n} x^{2m-2n+1} - \nu_f (2m-2n) y^{2n+2} x^{2m-2n-1} \right]\right\} \qquad (91)$$

$$u_y^f = \frac{1+\nu_f}{E_f} \left\{ -K_1 \left[\left(1-\nu_f\right) x^3 + 3\nu_f x y^2\right] - K_2 \left(2\nu_f xy\right) - K_3 \left(\nu_f x\right) - \right.$$

$$\left[K_5 \, 3\left(1-\nu_f\right) x^2 + 3\nu_f y^2\right] - K_6 \left(2\nu_f y\right) - K_7 \nu_f +$$

$$K_8 \left[3\left(1-\nu_f\right) x y^2 - \left(2-\nu_f\right) x^3\right] + K_9 \left[\left(1-\nu_f\right) y^2 - \left(2-\nu_f\right) x^2\right] +$$

$$\left. K_{10} \left[\left(1-\nu_f\right) 6xy\right] + K_{11} \left[2\left(1-\nu_f\right) y\right] + W_0 \left[-\left(1-\nu_f\right) x\right] \right\} +$$

$$\frac{1+\nu_f}{E_f} \left\{ \sum_{m=2}^{M} A_m \sum_{n=0}^{m} \binom{2m}{2n} (-1)^n \left[\left(1-\nu_f\right) \frac{(2m-2n+1)(2m-2n)}{2n+1} \cdot \right.\right.$$

$$\left. y^{2n+1} x^{2m-2n-1} - \nu_f (2n) y^{2n-1} x^{2m-2n+1} \right] +$$

$$\sum_{m=2}^{M} B_m \sum_{n=0}^{m} \binom{2m}{2n} (-1)^n \left[\left(1-\nu_f\right) \frac{(2m-2n)(2m-2n-1)}{2n+2} \cdot \right.$$

(Continued)

$$y^{2n+2}\ x^{2m-2n-2} - \nu_f(2n+1)\ y^{2n}\ x^{2m-2n}\Big] +$$

$$\sum_{m=2}^{M} C_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n \left[\left(1-\nu_f\right) \frac{(2m-2n+1)(2m-2n)}{2n+21} \cdot\right.$$

$$y^{2n+2}\ x^{2m-2n-1} - \nu_f(2n+1)\ y^{2n}\ x^{2m-2n+1}\Big] +$$

$$\sum_{m=1}^{M} D_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n \left[\left(1-\nu_f\right) \frac{(2m-2n)(2m-2n-1)}{2n+3} \cdot\right.$$

$$\left. y^{2n+3}\ x^{2m-2n-2} - \nu_f(2n+2)\ y^{2n+1}\ x^{2m-2n}\Big]\right\} \qquad (92)$$

In order to satisfy the equation $\theta(x,y) = \theta(-x,-y)$, the polynomial solution to the biharmonic equation in region m(matrix) is simplified as follows:

$$\theta^m = \bar{K}_1 xy^3 + \bar{K}_3 xy + \bar{K}_6 y^2 + \bar{K}_8 x^3 y + \bar{K}_{11} x^2 +$$

$$\sum_{m=2}^{M} \bar{C}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n\ y^{2n+1}\ x^{2m-2n+1} +$$

$$\sum_{m=1}^{M} \bar{D}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n\ y^{2n+2}\ x^{2m-2n} \qquad (93)$$

The stresses and displacements can thus be found by combining equations (77) through (81) and (93):

$$\sigma_x^m = 6\bar{K}_1 xy + 2\bar{K}_6 +$$

$$\sum_{m=2}^{M} \bar{C}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n (2n+1)\ 2ny^{2n-1}\ x^{2m-2n+1} +$$

$$+ \sum_{m=1}^{M} \bar{D}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n (2n+2)(2n+1)\, y^{2n}\, x^{2m-2n} \tag{94}$$

$$\sigma_y^m = 6\bar{K}_8 xy + 2\bar{K}_{11} +$$

$$\sum_{m=2}^{M} \bar{C}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n (2m-2n+1)(2m-2n)\, y^{2n+1}\, x^{2m-2n-1} +$$

$$\sum_{m=1}^{M} \bar{D}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n (2m-2n)(2m-2n-1)\, y^{2n+2}\, x^{2m-2n-2} \tag{95}$$

$$\tau_{xy}^m = -3\bar{K}_1 y^2 - \bar{K}_3 - 3K_8 x^2 -$$

$$\sum_{m=2}^{M} \bar{C}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n (2n+1)(2m-2n+1)\, y^{2n}\, x^{2m-2n} -$$

$$\sum_{m=1}^{M} \bar{D}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n (2n+2)(2m-2n)\, y^{2n+1}\, x^{2m-2n-1} \tag{96}$$

$$\frac{E_m u_x}{1+\nu_m} = \bar{K}_1 \left[\left(1-\nu_m\right)(3x^2y-2y^3)\right] - \nu_m y^3 + \bar{K}_3\left[-\nu_m y - 2\left(1-\nu_m\right)y\right] +$$

$$\bar{K}_6\left[2\left(1-\nu_m\right)x\right] + \bar{K}_8\left[-3\nu_m x^2 y - \left(1-\nu_m\right)y^3\right] + \bar{K}_{11}\left(-2\nu_m x\right) +$$

$$\left(1-\nu_m\right) W_0 y + \sum_{m=2}^{M} \bar{C}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n \cdot$$

$$\left[\left(1-\nu_m\right) \frac{2n(2n+1)}{2m-2n+2}\, y^{2n-1}\, x^{2m-2n+2} - \nu_m(2m-2n+1)\, y^{2n+1}\, x^{2m-2n}\right] +$$

(Continued)

$$+ \sum_{m=1}^{M} \bar{D}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n \Bigg[\left(1-\nu_m\right) \frac{(2n+2)(2n+1)}{2m-2n+1} \cdot$$

$$y^{2n} x^{2m-2n+1} - \nu_m (2m-2n) y^{2n+2} x^{2m-2n-1} \Bigg] \tag{97}$$

$$\frac{E_m u_y^m}{1+\nu_m} = \bar{K}_1 \left[-3\nu_m xy^2 - \left(1-\nu_m\right) x^3 \right] + \bar{K}_3 \left(-\nu_m x\right) + \bar{K}_6 \left(-2\nu_m y\right) +$$

$$\bar{K}_8 \left[\left(1-\nu_m\right) (3xy^2 - 2x^3) - \nu_m x^3 \right] + \bar{K}_{11} \left[2 \left(1-\nu_m\right) y \right] +$$

$$W_0 \left[- \left(1-\nu_m\right) x \right] + \sum_{m=2}^{M} \bar{C}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n \cdot$$

$$\Bigg[\left(1-\nu_m\right) \frac{(2m-2n+1)(2m-2n)}{2n+2} y^{2n+2} x^{2m-2n-1} -$$

$$\nu_m (2n+1) y^{2n} x^{2m-2n+1} \Bigg] + \sum_{m=1}^{M} \bar{D}_m \sum_{n=0}^{m} \binom{2m+1}{2n+1} (-1)^n \cdot$$

$$\Bigg[\left(1-\nu_m\right) \frac{(2m-2n)(2m-2n-1)}{2n+3} y^{2n+3} x^{2m-2n-2} -$$

$$\nu_m (2n+2) y^{2n+1} x^{2m-2n} \Bigg] \tag{98}$$

A basic representative element of fiber-reinforced composites under lateral loading is depicted in Figure 34. For the convenience of numerical calculation, the boundary conditions for each assigned point of the element are listed in Table I. The Cartesian coordinates of all points are listed in Table II. The normal and tangential stresses at the interface are shown in Figure 35. These stresses can be expressed as

$$\sigma_n = \sigma_x \cos^2\varphi + \sigma_y \sin^2\varphi \; 2\tau_{xy} \sin\varphi \cos\varphi \tag{99}$$

$$\tau_t = \sigma_y - \sigma_x \sin\varphi \cos\varphi + \tau_{xy} \cos^2\varphi - \sin^2\varphi \tag{100}$$

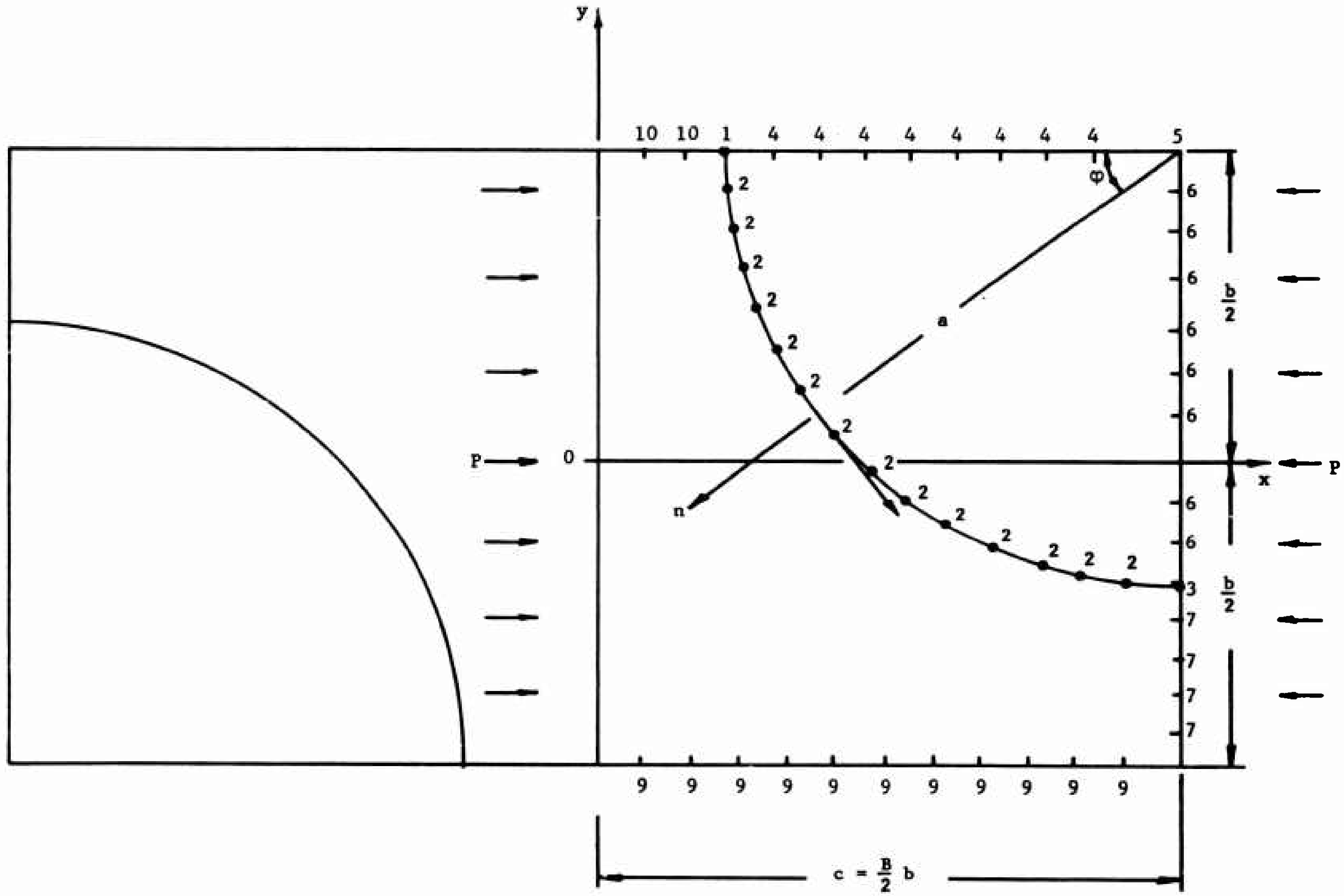


Figure 34. Element of Fiber-Reinforced Composites Under Lateral Loading.

TABLE I. LIST OF BOUNDARY CONDITIONS OF VARIOUS POINTS							
Point No.	Boundary Conditions				No. of Boundary Conditions*	No. of Points	Total Boundary Conditions
1	$\sigma_x^f = \sigma_x^m$	$\tau_{xy}^f = \tau_{xy}^m = 0$	$u^f = u^m$	$v^f = v^m = 0$	6	1	6
2	$\sigma_n^f = \sigma_n^m$	$\tau_{nt}^f = \tau_{nt}^m$	$u_n^f = u_n^m$, $u_t^f = u_t^m$		4	14	56
3	$\sigma_y^f = \sigma_y^m$	$\tau_{xy}^f = \tau_{xy}^m = 0$	$u^f = u^m = k$	$v^f = v^m$	6	1	6
4	$\sigma_{xy}^f = 0$			$v^m = 0$	2	9	18
5	$\sigma_{xy}^f = 0$		$u^f = k$		3	1	3
6	$\sigma_{xy}^f = 0$		$u^f = k$		2	9	18
7	$\sigma_{xy}^m = 0$		$u^m = k$		2	4	8
8	$\sigma_{xy}^m = 0$		$u^m = k$	$v^m = 0$	3	1	3
9	$\sigma_{xy}^m = 0$			$v^m = 0$	2	12	24
10	$\sigma_{xy}^m = 0$			$v^m = 0$	2	3	6
					Total	55	148

* At each point.

TABLE II. CARTESIAN COORDINATES OF VARIOUS POINTS					
Point No.	x	y	ϕ	j	No. of Points
1	$c-a$	$\frac{b}{2}$	0	1	1
2	$c-a$ coz ϕy	$\frac{b}{2} - a \sin\phi j$	$\frac{\Pi}{30} j$	1,2,...,14	14
3	c	$\frac{b}{2} - a$	$\frac{\Pi}{2}$	1	1
4	$c-a + j\frac{a}{10}$	$\frac{b}{2}$		1,2,...,9	9
5	c	$\frac{b}{2}$		1	1
6	c	$\frac{b}{2} - j\frac{a}{10}$	$\frac{\Pi}{2}$	1,2,...,9	9
7	c	$\frac{b}{2} - a - j\frac{b-a}{5}$		1,2,3,4	4
8	c	$-\frac{b}{2}$		1	1
9	$j\frac{c}{12}$	$-\frac{b}{2}$		0,1,2,...,11	12
10	$j\frac{c-a}{3}$	$-\frac{b}{2}$		0,1,2	3
				Total No. of Points	55

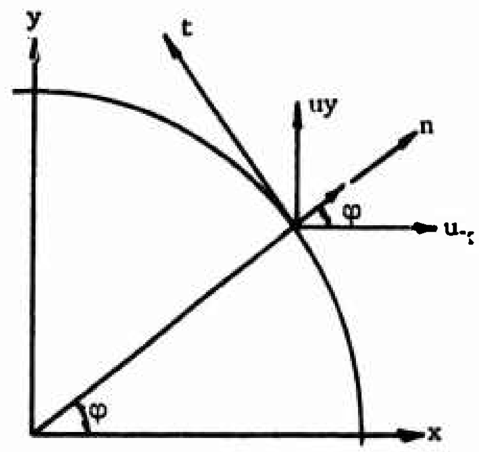


Figure 35. Interface Axis.

$$\sigma_n = \sigma_x \cos^2\varphi + \sigma_y \sin^2\varphi + 2\sigma_{xy} \sin\varphi \cos\varphi$$

$$\sigma_{mt} = \left(\sigma_y - \sigma_x\right) \sin\varphi \cos\varphi + \sigma_{xy}\left(\cos^2\varphi - \sin^2\varphi\right)$$

$$\left.\begin{aligned} u_n &= u \cos\varphi + v \sin\varphi \\ u_t &= -u \sin\varphi + v \cos\varphi \end{aligned}\right\} \qquad (101)$$

At point b_i,

$$\left.\begin{aligned} x_i &= c - a \cos\varphi_i \\ y_i &= \frac{b}{2} - a \sin\varphi_i \end{aligned}\right\} \qquad (102)$$

Equations which satisfy the ten types of boundary conditions are listed in this section. In these equations, $\alpha_i \left(\alpha_1^f, \alpha_2^f, \ldots, \alpha_{37}, \alpha_1^m, \alpha_2^m, \ldots, \alpha_{19}\right)$ identifies the 19 unknowns $\left(A_m, B_m, C_m, D_m, \overline{C}_m, \overline{D}_m, K, \overline{K}, W_o, \overline{W}_o\right)$ of equations (88) through (92) and (94) through (98).

TYPE 1 EQUATIONS

$$\sigma_x^f\left(c-a,\frac{b}{2},\alpha_i^f\right) - \sigma_x^m\left(c-a,\frac{b}{2},\alpha_i^m\right) = 0 \tag{103}$$

$$\sigma_{xy}^f\left(c-a,\frac{b}{2},\alpha_i^f\right) = 0 \tag{104}$$

$$\sigma_{xy}^m\left(c-a,\frac{b}{2},\alpha_i^m\right) = 0 \tag{105}$$

$$u^f\left(c-a,\frac{b}{2},\alpha_i^f\right) - u^m\left(c-a,\frac{b}{2},\alpha_i^m\right) = 0 \tag{106}$$

$$v\left(c-a,\frac{b}{2},\alpha_i^f\right) = 0 \tag{107}$$

$$v\left(c-a,\frac{b}{2},\alpha_i^m\right) = 0 \tag{108}$$

TYPE 2 EQUATIONS

$$\begin{aligned}
&\left\{\sigma_x^f\left[c-a\cos\left(\frac{\pi j}{30}\right), \frac{b}{2} - a\sin\left(\frac{\pi j}{30}\right), \alpha_i^f\right] - \right.\\
&\qquad \left.\sigma_x^m\left[c-a\cos\left(\frac{\pi j}{30}\right), \frac{b}{2} - a\sin\left(\frac{\pi j}{30}\right), \alpha_i^m\right]\right\}\cos^2\left(\frac{\pi j}{30}\right) + \\
&\qquad \left\{\sigma_y^f\left[c-a\cos\left(\frac{\pi j}{30}\right), \frac{b}{2} - a\sin\left(\frac{\pi j}{30}\right), \alpha_i^f\right] - \right.\\
&\qquad \left.\sigma_y^m\left[c-a\cos\left(\frac{\pi j}{30}\right), \frac{b}{2} - a\sin\left(\frac{\pi j}{30}\right), \alpha_i^m\right]\right\}\sin^2\left(\frac{\pi j}{30}\right) + \\
&\qquad \left\{\sigma_{xy}^f\left[c-a\cos\left(\frac{\pi j}{30}\right), \frac{b}{2} - a\sin\left(\frac{\pi j}{30}\right), \alpha_i^f\right] - \right.\\
&\qquad \left.\sigma_{xy}^m\left[c-a\cos\left(\frac{\pi j}{30}\right), \frac{b}{2} - a\sin\left(\frac{\pi j}{30}\right), \alpha_i^m\right]\right\}\sin\left(\frac{\pi j}{15}\right) = 0
\end{aligned} \tag{109}$$

where

$j = 1,\ldots,14$

$$
\begin{aligned}
\frac{1}{2}\Bigg\{\sigma_y^f\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right),\ \alpha_i^f\right] - \\
\sigma_y^m\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right),\ \alpha_i^m\right] - \\
\sigma_x^f\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right),\ \alpha_i^f\right] - \\
\sigma_x^m\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right),\ \alpha_i^m\right]\Bigg\}\sin\left(\frac{\pi j}{15}\right) + \\
\Bigg\{\sigma_{xy}^f\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right)\ \alpha_i^f\right] - \\
\sigma_{xy}^m\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\,\mathrm{sfn}\left(\frac{\pi j}{30}\right),\ \alpha_i^m\right]\Bigg\}\cos\left(\frac{\pi j}{15}\right) = 0 \qquad (110)
\end{aligned}
$$

where

$j = 1,\ldots,14$

$$
\begin{aligned}
\Bigg\{u^f\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right),\ \alpha_i^f\right] - \\
u^m\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right),\ \alpha_i^m\right]\Bigg\}\cos\left(\frac{\pi j}{30}\right) + \\
\Bigg\{v^f\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right),\ \alpha_i^f\right] - \\
v^m\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right),\ \alpha_i^m\right]\Bigg\}\sin\left(\frac{\pi j}{30}\right) = 0 \qquad (111)
\end{aligned}
$$

where

$j = 1,\ldots,14$

$$
\begin{aligned}
\Bigg\{u^f\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right),\ \alpha_i^f\right] - \\
u^m\left[c\text{-}a\cos\left(\frac{\pi j}{30}\right),\ \frac{b}{2}-a\sin\left(\frac{\pi j}{30}\right),\ \alpha_i^m\right]\Bigg\}\sin\left(\frac{\pi j}{30}\right) -
\end{aligned}
$$

$$- \left\{ u_y^f\left[c-a\cos\left(\frac{\pi j}{30}\right), \frac{b}{2} - a\sin\left(\frac{\pi j}{30}\right), \alpha_i^f\right] - u_y^m\left[c-a\cos\left(\frac{\pi j}{30}\right), \frac{b}{2} - a\sin\left(\frac{\pi j}{30}\right), \alpha_i^m\right]\right\} \cos\left(\frac{\pi j}{30}\right) = 0 \qquad (112)$$

where

$j = 1,\ldots,14$

TYPE 3 EQUATIONS

$$\sigma_y^f\left(c,\frac{b}{2} - a,\alpha_i^f\right) - \sigma_y^m\left(c,\frac{b}{2} - a,\alpha_i^m\right) = 0 \qquad (113)$$

$$\sigma_{xy}^f\left(c,\frac{b}{2} - a,\alpha_i^f\right) = 0 \qquad (114)$$

$$\sigma_{xy}^m\left(c,\frac{b}{2} - a,\alpha_i^m\right) = 0 \qquad (115)$$

$$u_x^f\left(c,\frac{b}{2} - a,\alpha_i^f\right) = k \qquad (116)$$

$$u_x^m\left(c,\frac{b}{2} - a,\alpha_i^m\right) = k \qquad (117)$$

$$u_y^f\left(c,\frac{b}{2} - a,\alpha_i^f\right) - u^m\left(c,\frac{b}{2} - a,\alpha_i^m\right) = 0 \qquad (118)$$

TYPE 4 EQUATIONS

$$\sigma_{xy}^f\left(c-a+j\frac{a}{10},\frac{b}{2},\alpha_i^f\right) = 0 \qquad (119)$$

where

$j = i,\ldots,9$

$$u_y^f\left(c-a+j\frac{a}{10},\frac{b}{2},\alpha_i^f\right) = 0 \tag{120}$$

where

$j = 1,...,9$

TYPE 5 EQUATIONS

$$\tau_{xy}^f\left(c,\frac{b}{2},\alpha_i^f\right) = 0 \tag{121}$$

$$u^f\left(c,\frac{b}{2},\alpha_i^f\right) = k \tag{122}$$

$$u_y^f\left(c,\frac{b}{2},\alpha_i^f\right) = 0 \tag{123}$$

TYPE 6 EQUATIONS

$$\sigma_{xy}^f\left(c,\frac{b}{2} - j\frac{a}{10},\alpha_i^f\right) = 0 \tag{124}$$

where

$j = 1,...,9$

$$u^f\left(c,\frac{b}{2} - j\frac{a}{10},\alpha_i^f\right) = k \tag{125}$$

where

$j = 1,...,9$

TYPE 7 EQUATIONS

$$\sigma_{xy}^m\left[c,\frac{b}{2} - a - j\left(\frac{b-a}{5}\right),\alpha_i^m\right] = 0 \tag{126}$$

where

$j = 1,2,3,4$

$$u_x^m\left[c,\frac{b}{2}-a-j\left(\frac{b-a}{5}\right),\alpha_i^m\right] = k \tag{127}$$

where

$j = 1,2,3,4$

<u>**TYPE 8 EQUATIONS**</u>

$$\sigma_{xy}^m\left(c,-\frac{b}{2},\alpha_i^m\right) = 0 \tag{128}$$

$$u^m\left(c,-\frac{b}{2},\alpha_i^m\right) = k \tag{129}$$

$$u_y^m\left(c,-\frac{b}{2},\alpha_i^m\right) = 0 \tag{130}$$

<u>**TYPE 9 EQUATIONS**</u>

$$\sigma_{xy}^m\left(j\frac{c}{12},-\frac{b}{2},\alpha_i^m\right) = 0 \tag{131}$$

where

$j = 0,1,\ldots,11$

$$u_y^m\left(j\frac{c}{12},-\frac{b}{2},\alpha_i^m\right) = 0 \tag{132}$$

where

$j = 0,1,\ldots,11$

<u>**TYPE 10 EQUATIONS**</u>

$$\sigma_{xy}^m\left(j\frac{c-a}{3},\frac{b}{2},\alpha_i^m\right) = 0 \tag{133}$$

where

$j = 0,1,2$

$$u_y^m\left(j\frac{c-a}{3},\frac{b}{2},\alpha_i^m\right) = 0 \tag{134}$$

where

$j = 0,1,2$

The total number of equations is 148, but the total number of unknowns to be determined (α_1^f, α_2^f,..., α_{37}^f , α_1^m, α_2^m,...,α_{19}^m) is 19. Therefore, we can use the method of Least Squares to obtain the results. When solving the 148 simultaneous equations (103) through (134), we assumed that $k = -1$.

In theory, the Point Matching method was used to solve the problem, and the computer program is presented in Appendix II. The finite element method was eventually used, however, since it was found that the Point-Matching method does not give good convergence in the solution.

APPENDIX II

COMPUTER PROGRAM FOR THE SOLUTION OF THE THREE-DIMENSIONAL PROBLEM

```
$JOB,91503-003. TED NEFF.
$TAPE,SCR=02,OLD=00,NEW=00.
RUN(S,,,,,,240000)
SET(0)
LGO.
-
      PROGRAM THREED(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,TAPE2 )
      COMMON/1/ MSK(1000), KS22(80,240,3),IV(55,4,3),X1(4,3),X2(4,3),
     1  X3(4,3),            K21(82)    ,KMX(12,12),CDA(72,3,55)   ,
     2  IN, MM, IP, IPL,             A21(82) , X(32,2) , ICO(4,3,55)
     3 , CENT(3,3,55),    DZ
      DIMENSION  AS22(80,240)
      EQUIVALENCE  (S(19201 ),AS22)
      COMMON /INPUT/   XNU(2), E(2) ,XLAM(2) , G(2) , CMX(6,6,2) , DO

      DIMENSION   C(6,6,2) , TEMP(10)
      DIMENSION   S(80,80,5)
      EQUIVALENCE  (KS22(19201),S)   ,   (CMX,C)
C
C  THIS PROGRAM, WITH ITS ASSOCIATED SUBROUTINES, SOLVES THE THREE-
C  DIMENSIONAL STRESS PROBLEM BY THE METHOD OF FINITE ELEMENTS.
C
C  MM --   HIGHEST NODAL POINT INDEX OF TRIANGLES IN GROUND PLANE
C  IN - HIGHEST INDEX NUMBER OF BASE TRIANGLE
C  IP - HIGHEST INDEX OF BASE PLANE NORMAL TO THE Z DIRECTION HAVING
C       THE DISTANCE  IP*DZ FROM PLANE  Z = 0.
      MM = 32
      IN = 55
      IP =  9
      READ 1000,(TEMP(I),I=1,10)
 1000 FORMAT(10A8)
      PRINT 1010, (TEMP(I),I=1,10)
 1010 FORMAT(1H1,10A8)
      READ 1001, DZ,E(1), E(2), XNU(1), XNU(2), DO
 1001 FORMAT(6(5X, E8.4))
      DO 1 I=1,MM
    1 READ 1005, K, X(K,1) , X(K,2)
 1005 FORMAT(I3,E12.4,E12.4)
      DO 10 I=1,2
      XLAM(I) = XNU(I)*E(I) / ((1.+XNU(I))*(1.-2.*XNU(I)) )
      G(I) = E(I) / (2.*(1.+XNU(I)))
   10 CONTINUE
C
C  DERIVE C MATRIX CMX
C
      DO 20 I=1,6
      DO 20 J=1,6
      DO 20 K=1,2
      CMX(I,J,K) = 0.
   20 CONTINUE
      DO 25 I=1,2
      C(1,1,I) = XLAM(I)+2.*G(I)
      C(2,2,I) =    C(1,1,I)
      C(3,3,I) =    C(1,1,I)
      C(4,4,I) = G(I)
      C(5,5,I) = C(4,4,I)
      C(6,6,I) = C(4,4,I)
      C(2,1,I) = XLAM(I)
      C(3,1,I) = C(2,1,I)
      C(3,2,I) = C(2,1,I)
      C(1,2,I) = C(2,1,I)
      C(1,3,I) = C(2,1,I)
```

```
      C(2,3,I) = C(2,1,I)
   25 CONTINUE
C     PRINT OUT THE INPUT QUANTITIES
C
      PRINT 1090, (XLAM(J),J=1,2),(G(J),J=1,2),(XNU(J),J=1,2),
     1            (E(J),J=1,2)
 1090 FORMAT(1H1, 5X,9HLAMBDA(1),5X,9HLAMBDA(2),10X,4HG(1),10X,4HG(2),
     1   9X,5HNU(1),9X,5HNU(2),10X,4HE(1),10X,4HE(2) /1X,8(3XE11.4)///)
      PRINT 1091, DZ, DO
 1091 FORMAT(1H ,12X,2HDZ,12X,2HDO/X,2(3X,E11.4)///)
      DO 30  J = 1,8
      I1 = 4*(J-1)+1
      I2 =  I1+3
 1092 FORMAT(1H ,4( 7X, 2HX(,I2,3H,1), 7X,2HX(,I2,3H,2) )/X8(3XE11.4)/)
   30 PRINT 1092,(((II,II),II=I1,I2),((X(II,1),X(II,2)),II=I1,I2))
C
C  PR'NT OUT C MATRICES, 1 PAGE
C
      PRINT 2001
 2000 FORMAT(2H C,I1/)
      DO 5001 I=1,2
      PRINT 2000,I
      DO 5000 J=1,6
 5000 PRINT 2002,(C(J,K,I),K=1,6)
 5001 PRINT 2003
 2001 FORMAT(1H1)
 2003 FORMAT(///)
 2002 FORMAT(1H ,6(E13.6,4X))
      CALL NOWAK
C
C  NOW HAVE COEFFICIENTS OF UNKNOWN DISPLACEMENT MATRIX
C  AS22(80,160)  IS FIRST PARTITION ROW
C  KS22(80X240,1) IS USED FOR ROWS TWO THROUGH 8
C  KS22(80X240,2)  IS GENERATED IN CHL3D
C
      CALL CHL3D
C
C  NOW HAVE UNKNOWN DISPLACEMENTS IN SP(82,9) = S(1,1,1)
C
      CALL SIGMAS
      STOP
      END
```

```
      SUBROUTINE GETKMX(KJ,I)
      COMMON/1/ MSK(1000), KS22(80,240,3),IV(55,4,3),X1(4,3),X2(4,3),
     1  X3(4,3),              K21(82)   ,KMX(12,12),CDA(72,3,55)   ,
     2  IN, MM, IP, IPL,                A21(82) , X(32,2) , ICO(4,3,55)
     3 , CENT(3,3,55),    DZ
      DIMENSION  S(80,80,5)
      DIMENSION  AS22(80,240)
      EQUIVALENCE  (S(19201 ),AS22)
      EQUIVALENCE  (KS22(19201),S)
C
C  THIS SUBROUTINE DERIVES THE (12X12) K-MATRICES FOR USE
C  IN DERIVING ROWS OF THE BIG MATRIX K22
C
      DIMENSION DXT(72) , DX(72)
      DIMENSION   JMX(12,12) , AMX(12,12)
      DIMENSION XI(4), ETA(4), ZETA(4), ASTR(4,4)
      COMMON /INPUT/   XNU(2), E(2) ,XLAM(2) , G(2) , CMX(6,6,2) ,
      REAL KMX
      DATA (DXT(J),J=1,72)/0.,1.,16*0.,1.,16*0.,1.,0.,0.,1.,0.,0.,1.,
     1      9*0.,1.,5*0.,1.,9*0.,1.,0.,0.,1.,0./
C
      DATA (DX (J),J=1,72)/6*0.,1.,8*0.,1.,6*0.,1.,10*0.,1.,3*0.,1.,
     1      9*0.,1.,10*0.,1.,6*0.,1.,2*0.,1.,3*0./
C
      IF(I.LE.10) GO TO 15
      IF(I.LE.12) GO TO 10
      IF(I.LE.21) GO TO 15
      IF(I.LE.24) GO TO 10
      IF(I.LE.32) GO TO 15
   10 IREG = 2
      GO TO 20
   15 IREG = 1
   20 CONTINUE
C
C  IREG = 1,  FIBER
C  IREG = 2, RESIN
C
      DO 30 IBC= 2,4
      XI(IBC) = X1(IBC,KJ) - X1(1,KJ)
      ETA(IBC) = X2(IBC,KJ) - X2(1,KJ)
      ZETA(IBC) = X3(IBC,KJ) - X3(1,KJ)
   30 CONTINUE
C
C  GET INVERSE OF AMX
C
      ASTR(1,1) = XI(2)*(ETA(3)*ZETA(4) - ETA(4)*ZETA(3))
     1            -XI(3)*(ETA(2)*ZETA(4)- ETA(4)*ZETA(2))
     2            +XI(4)*(ETA(2)*ZETA(3)- ETA(3)*ZETA(2))
      ASTR(2,1) = -(ETA(3)*ZETA(4)  - ETA(4)*ZETA(3) )
     1            +(ETA(2)*ZETA(4)  - ETA(4)*ZETA(2) )
     2            -(ETA(2)*ZETA(3)  - ETA(3)*ZETA(2) )
      ASTR(3,1) =  (XI(3)*ZETA(4)  - XI(4)*ZETA(3))
     1            -(XI(2)*ZETA(4)  - XI(4)*ZETA(2))
     2            +(XI(2)*ZETA(3)  - XI(3)*ZETA(2))
      ASTR(4,1) = -(XI(3)*ETA(4)  - XI(4)*ETA(3) )
     1            +(XI(2)*ETA(4)  - XI(4)*ETA(2) )
     2            -(XI(2)*ETA(3)  - XI(3)*ETA(2) )
      ASTR(1,2) = 0.
      ASTR(2,2) =  (ETA(3)*ZETA(4)  - ETA(4)*ZETA(3) )
      ASTR(3,2) = -(XI(3)*ZETA(4)  - XI(4)*ZETA(3))
      ASTR(4,2) =  (XI(3)*ETA(4)  - XI(4)*ETA(3))
      ASTR(1,3) = 0.
```

```
      ASTR(2,3) = -(ETA(2)*ZETA(4) - ETA(4)*ZETA(2))
      ASTR(3,3) =  (XI(2)*ZETA(4) - XI(4)*ZETA(2))
      ASTR(4,3) = -(XI(2)*ETA(4)-XI(4)*ETA(2))
      ASTR(1,4) = 0.
      ASTR(2,4) =  (ETA(2)*ZETA(3) - ETA(3)*ZETA(2))
      ASTR(3,4) = -(XI(2)*ZETA(3) - XI(3)*ZETA(2))
      ASTR(4,4) =  (XI(2)*ETA(3) - XI(3)*ETA(2) )
C
      DELTA = XI(2) * (ETA(3)*ZETA(4) - ETA(4)*ZETA(3))
     1      + XI(3) * (-ETA(2)*ZETA(4) + ETA(4)*ZETA(2))
     2      + XI(4) * (ETA(2)*ZETA(3) - ETA(3)*ZETA(2))
      DO 35 IQ = 1,4
      DO 35 IR = 1,4
   35 ASTR(IQ,IR) = ASTR(IQ,IR) / DELTA
      DO 50 IQ =1,4
      DO 50 IR =1,4
      AMX(IQ,IR) = ASTR(IQ,IR)
      AMX(IQ+4,IR+4) = ASTR(IQ,IR)
      AMX(IQ+8,IR+8) = ASTR(IQ,IR)
      AMX(IQ+4,IR   ) = 0.
      AMX(IQ+8,IR   ) = 0.
      AMX(IQ+8,IR+4) = 0.
      AMX(IQ   ,IR+4) = 0.
      AMX(IQ   ,IR+8) = 0.
      AMX(IQ+4,IR+8) = 0.
   50 CONTINUE
C
C  NOW  AMX CONTAINS INVERSE OF A
C
      V = ABS(DELTA) / 6.
      CALL MXMULT(DX,AMX,KMX,6,12,12)
      CALL MXMULT(CMX(1,1,IREG),KMX,CDA(1,KJ,I),6,6,12)
C
C  NOW HAVE CDA(I)
C  CDA IS (6 X 12). SECOND SUBSCRIPT IS TETRAHEDRON NUMBER KJ.
C  THIRD SUBSCRIPT IS BASE TRIANGLE INDEX.
C
      DO 60 IQ =1,4
      DO 60 IR =1,4
      AMX(IQ,IR) =ASTR(IR,IQ)
      AMX(IQ+4,IR+4) =ASTR(IR,IQ)
   60 AMX(IQ+8,IR+8) =ASTR(IR,IQ)
C
C  AMX NOW CONTAINS TRANSPOSE OF INVERSE OF A
C
      CALL MXMULT(AMX,DXT,JMX,12,12,6)
      CALL MXMULT(JMX,CDA(1,KJ,I),KMX,12,6,12)
      CALL MXCON(KMX,KMX,V,12,12)
C
C  K MATRIX NOW IN KMX(J,K) , J=1,12  , K=1,12
C
   40 RETURN
      END
```

```
      SUBROUTINE NOWAK
      COMMON/1/ MSK(1000), KS22(80,240,3),IV(55,4,3),X1(4,3),X2(4,3),
     1  X3(4,3),                K21(82)   ,KMX(12,12),CDA(72,3,55)   ,
      COMMON/1/ MSK(1000), KS22(80,240,3),IV(55,4,3),X1(4,3),X2(4,3),
     1  X3(4,3),                K21(82)   ,KMX(12,12),CDA(72,3,55)   ,
     2  IN, MM, IP, IPL,                 A21(82) , X(32,2) , ICO(4,3,55)
     3 , CENT(3,3,55),    DZ
      DIMENSION AS22(80,240)   ,KX(90),PAZ(3)
      EQUIVALENCE   (S(192J1 ),AS22)
      COMMON /INPUT/   XNU(2), E(2) ,XLAM(2) , G(2) , CMX(6,6,2) , DO
      EQUIVALENCE (KS22(19201),QS22)
      DIMENSION  S(80,80,5)
      DIMENSION QS22(12800)   ,   PP(1000)
      EQUIVALENCE   (KS22(19201),S)
      DIMENSION IM(35,3) , LM(35,3)
      DIMENSION  IKO(3,2)
      REAL KS22, K21 , KMX
C
C   THIS SUBROUTINE DERIVES ALL NON-ZERO COEFFICIENTS FOR THE UNKNOWN
C   DISPLACEMENT MATRIX. THE DISPLACEMENT MATRIX IS (720X720), AND IS
C   DIVIDED INTO 9 ROWS OF SUBMATRICES WHICH  HAVE (80X80) ELEMENTS EACH.
C   AT MOST, THREE OF THESE SUBMATRICES CONTAIN NON-ZERO ELEMENTS.  THUS,
C   DIVIDED INTO 9 ROWS OF SUBMATRICES WHICH  HAVE (80X80) ELEMENTS EACH.
C   AT MOST, THREE OF THESE SUBMATRICES CONTAIN NON-ZERO ELEMENTS.  THUS,
C   ONLY THESE THREE NON-ZERO SUBMATRICES ARE DEVELOPED.
C   SUBROUTINE NOWAK PUTS THE TWO SUBMATRICES OF INTEREST FOR THE FIRST
C   PARTITION ROW INTO AS22(80X160).  FOR ROWS TWO THROUGH EIGHT, THE
C   NON-ZERO SUBMATRICES REPEAT.  THESE THREE MATRICES ARE STORED IN
C   KS22(80X240,1).
C   THE TWO PARTITION ELEMENTS IN ROW NINE ARE GENERATED IN
C   SUBROUTINE CHL3D WHEN NEEDED ON THE NINTH PASS.
C
C
C   THE INDEPENDENT TERMS CORRESPONDING TO ROW ONE WILL BE IN A21.
C   THE INDEPENDENT TERMS FOR ROWS TWO THRU NINE WILL BE IN K21(80,1)
C
C
C
C   X(I,1) - X COORDINATE OF ITH NODAL POINT
C   X(I,2) - Y COORDINATE OF ITH NODAL POINT
C   DZ - DISTANCE BETWEEN TWO CONSECUTIVE BASE PLANES
C   E(1) - MODULUS OF ELASTICITY OF MEDIUM 1(FIBER)
C   E(2) - MODULUS OF ELASTICITY OF MEDIUM 2(RESIN)
C   XNU(1) - POISSONS RATIO OF FIBER
C   XNU(2) - POISSONS RATIO OF RESIN
C   ICO(4,3) -  ZERO OR ONE
C   IKO(3,2) -  ZERO OR ONE
C   DO - DISPLACEMENT IN X DIRECTION ATPLANE X= (SQRT(3.)/2.)*B
C
      PYF = 0.
      PZF = 0.
      PXS = 0.
      PZS = 0.
C   P IS FORCE.
C   F IS FRONT , S IS SIDE, X,Y,Z ARE DIRECTIONS
C
      PAZ(1) = 0.
      PAZ(2) = 0.
      PAZ(3) = 0.
C
C    PAZ(1) IS X COMPONENT OF FORCE AT PLANE Z=0(BOUNDARY LOAD)
C    PAZ(2) IS Y COMPONENT OF FORCE AT PLANE Z=0(BOUNDARY LOAD)
```

```
 PAZ(3) IS Z COMPONENT OF FORCE AT PLANE Z=0(BOUNDARY LOAD)
FOR PAZ(1) = PAZ(2) = PAZ(3)= 0   WE HAVE FREE SURFACE CONDITIONS.

      IX = 0
      JX = 0
      DO 103 M = 1,MM
      IF(M.LT.8) GO TO 100
      IF(MOD(M,7).EQ.0.OR.MOD(M-1,7).EQ.0) GO TO 101
      DO 105 MN1=1,3
      IX = IX + 1
      JX = JX + 1
      MSK(IX) = 3*(M-1) + MN1
      PP(IX) = PAZ(MN1)
  105 KX(JX) = IX
      GO TO 103
  100 IF(M.EQ.1.OR.M.EQ.7)  GO TO 102
      IX = IX + 1
      MSK(IX)= 3*(M-1) + 2
      PP(IX) = PYF
      IX = IX + 1
      MSK(IX) = 3*(M-1)+3
      PP(IX) = PZF
      GO TO 103
  102 IX = IX+1
      MSK(IX) = 3*M
      PP(IX) = PZS
      GO TO 103
  101 IX = IX + 1
      MSK(IX)= 3 * (M-1) + 1
      PP(IX) = PXS
      IX = IX+1
      MSK(IX) = 3*(M-1)+3
      PP(IX) = PZS
  103 CONTINUE
      IXMAX = IX
      JXMAX = JX
      DO 104 N=1,11
      DO 109 IX = 1, IXMAX
      MSK(N*IXMAX+IX)=MSK(IX) + N * 3 * MM
  109 PP(N*IXMAX+IX) = PP(IX)
      DO 108 JX = 1,JXMAX
      KXJX=KX(JX)
  108 PP(KXJX+N*IXMAX) = 0.
  104 CONTINUE
C
      DO 169 I = 1, IN
      IFF = 1
      IF(MOD(I,2).EQ.0)  IFF = -1
      IF(MOD(I,12).EQ.0)GO TO 2
      IE12 = 0
      GO TO 4
    2 IE12 = 1
    4 IM(1,1) = 1 +I/2  +(I-IE12)/12
      IKO(1,1) = 0
      IZ = I/2
      IF(MOD(IZ,2).EQ.0) GO TO 6
      IF(I.GT.48)  GO TO 12
      IM(2,1) = IM(1,1) +IFF + 7
      IKO(2,1)= 0
      GO TO 7
   12 IM(3,1) = IM(1,1)+IFF
      IKO(3,1) = 0
```

```
      GO TO 7
    6 IF(I.GT.48)   GO TO 11
      IM(2,1) = IM(1,1) + IFF
      IKO(2,1) = 0
      GO TO 7
   11 IM(3,1) = IM(1,1) + IFF + 7
      IKO(3,1) = 0
    7 IF(I.GT.48)  GO TO 13
      IM(3,1) = IM(1,1) + 7
      IKO(3,1) = 0
      GO TO 30
   13 IM(2,1) = IM(1,1) + 7
      IKO(2,1) = 0
   30 IF(IM(1,1).GT.32)  GO TO 31
      GO TO 32
   31 IM(1,1) = 64 - IM(1,1)
      IKO(1,1) = 1
   32 IF(IM(2,1).GT.32) GO TO 33
      GO TO 34
   33 IM(2,1) = 64 - IM(2,1)
      IKO(2,1) = 1
   34 IF(IM(3,1).GT.32) GO TO 35
      GO TO 36
   35 IM(3,1) = 64 - IM(3,1)
      IKO(3,1) = 1
   36 DO 20 K = 1,3
C
C     DETERMINE ALPHA, BETA, GAMMA, FOR EACH NODE IN EACH BASE TRIANGLE I
C
C     DETERMINE ALPHA, BETA, GAMMA, FOR EACH NODE IN EACH BASE TRIANGLE I
C
C
C     K=1 CORRESPONDS TO POINT ALPHA
C     K=2 CORRESPONDS TO POINT BETA
C     K=3 CORRESPONDS TO POINT GAMMA
C
      IMM = MOD(IM(K,1),21)
      IF(IMM.EQ.0)   IMM=21
      IF(MOD(IMM,7).EQ.0)  GO TO 8
      IE7= 0
      GO TO 10
    8 IE7 = 1
   10 IBAR = (IMM-IE7) / 7
      ID2 = MOD(IMM,2)
      IH1 = MOD(IBAR,3) -1
      IF(IH1) 14,15,16
   14 IF(ID2.EQ.0) GO TO 17
      IM(3,2) = IM(K,1)
      IKO(3,2) = IKO(K,1)
      GO TO 20
   17 IM(1,2) = IM(K,1)
      IKO(1,2) = IKO(K,1)
      GO TO 20
   15 IF(ID2.EQ.0) GO TO 18
      IM(3,2) = IM(K,1)
      IKO(3,2) = IKO(K,1)
      GO TO 20
   18 IM(2,2) = IM(K,1)
      IKO(2,2) = IKO(K,1)
      GO TO 20
   16 IF (ID2.EQ.0) GO TO 19
      IM(1,2) = IM(K,1)
```

```
      IKO(1,2) = IKO(K,1)
      GO TO 20
   19 IM(2,2) = IM(K,1)
      IKO(2,2) = IKO(K,1)
   20 CONTINUE
C
      IF(MOD(I,2).EQ.1) GO TO 27
      LX = 2
      MX = 3
      MM1 = MM
      MM2 = 0
      XM31 = DZ
      XM32= 0.
      GO TO 26
   27 LX = 3
      MX = 2
      MM1 = 0
      MM2 = MM
      XM31 = 0.
      XM32= DZ
   26 IV(I,1,1) = IM(1,1)
      IV(I,2,1) = IM(LX,1)
      IV(I,3,1) = IM(MX,1)
      IV(I,4,1) = IM(2,2) + MM
      IV(I,1,2) = IM(1,2)
      IV(I,2,2) = IM(LX,2) + MM1
      IV(I,3,2) = IM(MX,2)  + MM2
      IV(I,4,2) = IM(3,2) + MM
      IV(I,1,3) = IM(1,1) + MM
      IV(I,2,3) = IM(MX,1) + MM
      IV(I,3,3) = IM(LX,1) + MM
      IV(I,4,3) = IM(1,2)
C  MATRIX IV RELATES THE NODES TO THE TETRAHEDRONS
C  FIRST INDEX OF IV IS THE BASE TRIANGLE INDEX I
C  SECOND INDEX IS NODE POSITION WITHIN TETRAHEDRON I
C  THIRD INDEX IS THE NUMBER OF THE TETRAHEDRON ABOVE BASE TRIANGLE I
C
C  THIRD INDEX IS THE NUMBER OF THE TETRAHEDRON ABOVE BASE TRIANGLE I
C
      X3(1,1) = 0.
      X3(2,1) = X3(1,1)
      X3(3,1) = X3(1,1)
      X3(4,1) = X3(1,1) + DZ
      X3(1,2) = X3(1,1)
      X3(2,2) = XM31
      X3(3,2) = XM32
      X3(4,2) = X3(4,1)
      X3(1,3) = X3(4,1)
      X3(2,3) = X3(4,1)
      X3(3,3) = X3(4,1)
      X3(4,3) = X3(1,1)
C
C  X1(I,J) IS THE X COORDINATE OF THE ITH NODE OF THE JTH TETRAHEDRON
C  X2(I,J) IS THE Y COORDINATE OF THE ITH NODE OF THE JTH TETRAHEDRON
C  X3(I,J) IS THE Z COORDINATE OF THE ITH NODE OF THE JTH TETRAHEDRON
C        ABOVE THE CURRENT BASE TRIANGLE
C
      ICO(1,1,I) = IKO(1,1)
      ICO(2,1,I) = IKO(LX,1)
      ICO(3,1,I) = IKO(MX,1)
      ICO(4,1,I) = IKO(2,2)
      ICO(1,2,I) = IKO(1,2)
```

```
      ICO(2,2,I) = IKO(LX,2)
      ICO(3,2,I) = IKO(MX,2)
      ICO(4,2,I) = IKO(3,2)
      ICO(1,3,I) = IKO(1,1)
      ICO(2,3,I) = IKO(MX,1)
      ICO(3,3,I) = IKO(LX,1)
      ICO(4,3,I) = IKO(1,2)
      DO 50 IC1 = 1,3
      DO 50 IC2 = 1,3
   50 CENT(IC1,IC2,I) = 0.
      DO 98 KJ = 1,3
      X1(1,KJ) = X(MOD(IV(I,1,KJ),MM),1)*(-1)**ICO(1,KJ,I)
      X2(1,KJ) = X(MOD(IV(I,1,KJ),MM),2)*(-1)**ICO(1,KJ,I)
      X1(2,KJ) = X(MOD(IV(I,2,KJ),MM),1)*(-1)**ICO(2,KJ,I)
      X2(2,KJ) = X(MOD(IV(I,2,KJ),MM),2)*(-1)**ICO(2,KJ,I)
      X1(3,KJ) = X(MOD(IV(I,3,KJ),MM),1)*(-1)**ICO(3,KJ,I)
      X2(3,KJ) = X(MOD(IV(I,3,KJ),MM),2)*(-1)**ICO(3,KJ,I)
      X1(4,KJ) = X(MOD(IV(I,4,KJ),MM),1)*(-1)**ICO(4,KJ,I)
      X2(4,KJ) = X(MOD(IV(I,4,KJ),MM),2)*(-1)**ICO(4,KJ,I)
C
C     CENT(I,J,K) IS THE ITH CENTROID COMPONENT OF THE JTH TETRAHEDRON,
C      ABOVE THE KTH TRIANGLE
C
C     CALCULATE THE CENTROIDS
C
      IC1 = KJ
      DO 55 IC3=1,4
      CENT(1,IC1,I) = CENT(1,IC1,I) + X1(IC3,IC1)
      CENT(2,IC1,I) = CENT(2,IC1,I) + X2(IC3,IC1)
   55 CENT(3,IC1,I) = CENT(3,IC1,I) + X3(IC3,IC1)
      DO 56 IC4 = 1,3
      CENT(IC4,IC1,I) = CENT(IC4,IC1,I) / 4.
   56 CONTINUE
C
      CALL GETKMX(KJ,I)
  421 DO 99 K = 1,2
      DO 799 NZ = 1,12
      ISA1 = 1
      M1 = MOD(NZ,4)
      IF(M1.EQ.0) GO TO 700
      IE4 = 0
      GO TO 800
  700 M1 = 4
      IE4 = 1
  800 M2 = 1 + (NZ - IE4) / 4
      IF(ICO(M1,KJ,I).EQ.1)  GO TO 799
      IPL = 3*(IV(I,M1,KJ)+(K-1)*MM-1)+M2
      IF(IPL.LE.11)  GO TO 300
      IPL = IPL-10*(IPL/96+1)
  300 CONTINUE
      IF(3*(IV(I,M1,KJ)+(K-1)*MM-1)+M2-MSK(IPL))301,302,303
  302 IF(IPL.GT.160)  GO TO 799
      II1 = IPL
      GO TO 327
  301 IF(IPL.LT.3)  GO TO 799
      IMR = IPL - 1
  306 IF(3*(IV(I,M1,KJ)+(K-1)*MM-1)+M2-MSK(IMR)) 304,305,799
  305 IF(IMR.GT.160) GO TO 799
      II1 = IMR
      GO TO 327
  304 IMR = IMR - 1
      GO TO 306
```

```
303 IMR = IPL + 1
307 IF(3*(IV(I,M1,KJ)+(K-1)*MM-1)+M2-MSK(IMR)) 799,305,310
310 IMR = IMR+1
    GO TO 307
327 IF(III.LT.81)  GO TO 391
    GO TO 392
391 A21(III) = A21(III)+PP (III)
    GO TO 395
392 K21(III- 80) = K21(III- 80) + PP(III)
395 DO 81 KA = 1,3
    DO 82 KB = 1,4
    IPL = 3*(IV(I,KB,KJ) + (K-1)*MM-1)+KA
    IF(IPL.LE.11)  GO TO 396
    IPL = IPL-10*(IPL/96+1)
396 CONTINUE
    IF(3*(IV(I,KB,KJ)+(K-1)*MM-1)+KA-MSK(IPL)) 349,350,351
350 II2 = IPL
    ISA2 = ISA1
    IF(KA.LT.3)  ISA2 = ISA1*(1-2*ICO(KB,KJ,I))
    QX 4X 487
    IF(KA.LT.3)  ISA2 = ISA1*(1-2*ICO(KB,KJ,I))
    GO TO 376
349 IF(IPL.EQ.1)  GO TO 374
    IF(IPL.EQ.2)  GO TO 82
    IMR = IPL - 1
353 IF(3*(IV(I,KB,KJ)+(K-1)*MM-1)+KA-MSK(IMR)) 354,352,375
354 IMR = IMR - 1
    GO TO 353
352 II2 = IMR
    ISA2 = ISA1
    IF(KA.LT.3)  ISA2 = ISA1*(1-2*ICO(KB,KJ,I))
    GO TO 376
351 IMR = IPL + 1
358 IF(3*(IV(I,KB,KJ)+(K-1)*MM-1)+KA-MSK(IMR))  375,352,364
364 IMR = IMR + 1
    GO TO 358
376 IF(III.LT.81)  GO TO 517
    GO TO 519
517 AS22(III,II2) = AS22(III,II2) + KMX(NZ,4*(KA-1)+KB)*ISA2
    GO TO 82
519 KS22(III- 80,II2    ,1) = KS22(III- 80,II2    ,1) +
   1    KMX(NZ,4*(KA-1)+KB)*ISA2
    GO TO 82
375 MLD = MOD(3*(IV(I,KB,KJ)-1)+KA,3*MM)
    IF(MLD.GT.19.OR.MLD.EQ.2)  GO TO 82
374 IF(III.LT.81)  GO TO 518
    GO TO 520
518 A21(III) = A21(III)-KMX(NZ,4*(KA-1)+KB)*DO
    GO TO 82
520 K21(III-80) = K21(III-80) - KMX(NZ,4*(KA-1)+KB)*DO
 82 CONTINUE
 81 CONTINUE
799 CONTINUE
 99 CONTINUE
 98 CONTINUE
169 CONTINUE
    RETURN
    END
```

```
      SUBROUTINE CHL3D
C
      COMMON/1/ MSK(1000), KS22(80,240,3),IV(55,4,3),X1(4,3),X2(4,3),
     1  X3(4,3),              K21(82)   ,KMX(12,12),CDA(72,3,55)   ,
     2  IN, MM, IP, IPL,                A21(82)  , X(32,2)  , ICO(4,3,55)
     1  X3(4,3),              K21(82)   ,KMX(12,12),CDA(72,3,55)   ,
     2  IN, MM, IP, IPL,                A21(82)  , X(32,2)  , ICO(4,3,55)
     3  , CENT(3,3,55),     DZ
      DIMENSION JRC(32)
      DIMENSION K9(80,160)
      EQUIVALENCE (S,K9)
      DIMENSION  AS22(80,240)
      EQUIVALENCE  (S(19201 ),AS22)
      DIMENSION DELX(80,9) , GSAV(80,9)
      EQUIVALENCE (AS22 ,Z)
      DIMENSION Z(12800)
      DIMENSION  S(80,80,5)
      DIMENSION AS(80,80)
      EQUIVALENCE  (KS22(19201),S)
C  THIS ROUTINE SOLVES  SU=G , WHERE S IS A TRI-DIAGONAL MATRIX IN
C  SUBMATRICES, WITH ELEMENTS OF ORDER N.
C  S IS KNOWN
C  SP IS A VECTOR OF DIMENSION (NXM) WHERE M IS THE NUMBER OF DIVISIONS OF S
C  C IS WRITTEN ONTO TAPE AFTER DERIVATION ON THE FORWARD PASS,
C  SP IS A VECTOR OF DIMENSION (NXM) WHERE M IS THE NUMBER OF DIVISIONS OF S
C  C IS WRITTEN ONTO TAPE AFTER DERIVATION ON THE FORWARD PASS,
C  AND READ BACK IN ON THE BACKSWEEP
C  S(1,1,1)  INITIALLY CONTAINS  S(I,I-1)
C  S(1,1,2)  INITIALLY CONTAINS  S(I,I  )
C  S(1,1,4)  INITIALLY CONTAINS  S(I,I+1)
C  SP CORRESPONDS TO P IN THE WRITEUP BY GATEWOOD ON THE FORWARD PASS.
C  ON THE BACKSWEEP, IT CORRESPONDS TO U.
      DIMENSION   SP(80, 9)
      DIMENSION   C(6400)
      EQUIVALENCE  (S(25601),C)
      EQUIVALENCE  (  AS22(12801),SP)
      DIMENSION Q(12800)
      EQUIVALENCE (S,Q)
      REAL KS22,K21
      REAL K9
      DIMENSION G(82)
      DIMENSION ND(80)
      DIMENSION SQ(80,9)
      DATA (ND(I),I=1,80) / 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 3, 1, 3,
     1  1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 3, 1, 3, 1,
     2  2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 3, 1, 3, 1, 2, 3,
     3  1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 3, 1, 3, 1, 2, 3, 1,
     4  2, 3, 1, 2, 3 /
      DATA (JRC(I),I=1,32) / 1, 3, 5, 7, 9, 11, 12, 14, 17, 20, 23,
     1   26, 29, 31, 33, 36, 39, 42, 45, 48, 50, 52, 55, 58, 61, 64,
     2   67, 69, 71, 74, 77, 80 /
      XLIMIT = 1.E-8
      REWIND 1
      REWIND 2
      N = 80
      N2 = 160
      KREM = 80
      KREM2 = 160
      M = 9
      DO 30 ICYCLE =1,M
      IF(ICYCLE.GT.1) GO TO 12
      WRITE(2) (Z(I),I=1,12800)
```

```
      DO 11 I=1,80
      G(I) = A21(I)
      GSAV(I,ICYCLE) = G(I)
      DO 11 J=1,80
      S(I,J,2) = AS22(I,J)
   11 S(I,J,4) = AS22(I,J+80)
      GO TO 17
   12 IF(ICYCLE.EQ.M) GO TO 14
      DO 13 I=1,80
      G(I) = K21(I)
      GSAV(I,ICYCLE) = G(I)
      DO 13 J=1,80
      S(I,J,1) = KS22(I,J,1)
      S(I,J,2) = KS22(I,J+80,1)
   13 S(I,J,4) = KS22(I,J+160,1)
      GO TO 17
   14 INEW = 0
      JJRC = 1
      DO 480 J=1,80
      G(J) = K21(J)
      GSAV(J,ICYCLE) = G(J)
      IF (JRC(JJRC).LT.J)  JJRC = JJRC + 1
      IF(J-JRC(JJRC)) 460,465,460
  460 INEW = INEW + 1
      MSK(INEW+720) = MSK(J+720)
      MSK(INEW+800) = MSK(J+800)
  465 DO 480 I=1,80
      S(I,J,1) = KS22(I,J,1)
      IF(J-JRC(JJRC)) 470,475,470
  470 S(I,J,2) = KS22(I,J+80,1) + KS22(I,J+160,1)
      GO TO 480
  475 S(I,J,2) = KS22(I,J+80,1)
  480 CONTINUE
      WRITE (2) (Q(I),I=1,12800)
      DO 490 J=1,48
      MSK(J+768) = MSK(J+800)
  490 CONTINUE
   17 CONTINUE
    1 K1 = N
      K2 = N
      K3 = N2
      K4 = N
    4 CONTINUE
      IF(ICYCLE.EQ.1)  GO TO 10
      CALL MXMULT(S(1,1,1) , S(1,1,5) , S(1,1,3) ,K1,K2,K1)
C
C  S(1,1,5) CONTAINS C FROM LAST CYCLE
C
      CALL MXSUB(S(1,1,2) , S(1,1,3) , S(1,1,2) ,K1,K1)
   10 CONTINUE
C
C  B(I,I)  NOW IN S(1,1,2)
C
      CALL INVERT(S(1,1,2) , K1 , K3, XLIMIT , FLAG)
      IF (FLAG.NE.0.) GO TO 500
C
C  INVERSE OF B(I,I) NOW IN  S(1,1,2)
C
      IF (ICYCLE.EQ.1) GO TO 20
      CALL MXMULT(S(1,1,1) ,SP(1,ICYCLE-1) , S(1,1,3) ,K1, K2 , 1)
      CALL MXSUB (G  ,  S(1,1,3), G,K1, 1)
   20 CONTINUE
```

```
      CALL MXMULT(S(1,1,2) , G ,SP(1, ICYCLE  ) , K1 , K1 ,  1)
      IF (ICYCLE.GE.M) GO TO 35
      CALL MXMULT (S(1,1,2) , S(1,1,4) , S(1,1,5), K1 , K1 ,K4)
      NSQ = N*K4
      WRITE (1) (C(J),J=1,NSQ)
   30 CONTINUE
   35 CONTINUE
C
C  NOW IN BACKSWEEP, SOLVING FOR U
C
      DO 60 I = 2,M
      JCYCLE = M-I+1
      IF(JCYCLE.LT.M-1) GO TO 36
      K1 = KREM
      GO TO 37
   36 K1 = N
   37 CONTINUE
      NSQ = K1*N
      IF(JCYCLE.EQ.M-1)  GO TO 41
      BACKSPACE 1
   41 CONTINUE
      BACKSPACE 1
      READ (1)  (C(J),J=1,NSQ)
C
C  U(M)=SP(M) , CONSIDER FIRST (M-1)TH CYCLE
C
      CALL MXMULT(S(1,1,5) ,SP(1,JCYCLE+1),S(1,1,1),N,K1, 1)
      CALL MXSUB(SP(1,JCYCLE), S(1,1,1) ,SP(1,JCYCLE),  N ,  1)
   60 CONTINUE
C
C  U(NS,1) NOW STORED IN SP(N,I) , I=1,M
C
      REWIND 2
      DO 400 IC = 1,9
      IF(IC-2) 300,320,320
  300 READ (2) (Q(I),I=1,12800)
      CALL MXMULT(S,SP(1,1),DELX(1,1),80,160,1)
      GO TO 400
  320 IF(IC.EQ.M) GO TO 350
  310 DO 315 I=1,80
      DO 315 J=1,240
  315 KS22(I,J,2) = KS22(I,J,1)
      CALL MXMULT(S,SP(1,IC-1),DELX(1,IC),80,240,1)
      GO TO 400
  350 READ (2) (Q(I),I=1,12800)
      CALL MXMULT(Q,SP(1,8),DELX(1,IC),80,160,1)
  400 CONTINUE
      PRINT 4001
 4001 FORMAT(1H1,5X,3HROW,11X,4HDELX,12X,3HK21)
      DO 450 I=1,9
      DO 450 J=1,80
      K = 80*(I-1)+J
      IF(MOD(K,50).EQ.0)  PRINT 4001
  450 PRINT 4000,K,DELX(J,I),GSAV(J,I)
 4000 FORMAT(6X,I3,2(2X,E13.6))
      RETURN
  500 CONTINUE
      PRINT 1000 , ICYCLE
 1000 FORMAT (31H1COULD NOT INVERT MATRIX IN ROW,I2)
      STOP
      END
```

```
      SUBROUTINE SIGMAS
      COMMON/1/ MSK(1000), KS22(80,240,3),IV(55,4,3),X1(4,3),X2(4,3),
     1 X3(4,3),              K21(82)   ,KMX(12,12),CDA(72,3,55)   ,
      COMMON/1/ MSK(1000), KS22(80,240,3),IV(55,4,3),X1(4,3),X2(4,3),
     1 X3(4,3),              K21(82)   ,KMX(12,12),CDA(72,3,55)   ,
     2 IN, MM, IP, IPL,                A21(82) , X(32,2) , ICO(4,3,55)
     3 , CENT(3,3,55),   DZ
      EQUIVALENCE  (KS22,STRS)
      DIMENSION  AS22(80,240)
      EQUIVALENCE  (AS22(12801),DEL)
      EQUIVALENCE  (S(19201 ),AS22)
      COMMON /INPUT/   XNU(2), E(2) ,XLAM(2) , G(2) , CMX(6,6,2) , DO
      DIMENSION  S(80,80,5)
      EQUIVALENCE  (KS22(19201),S)
      DIMENSION   STRS(6,1500) , DX(12) , DEL(1000)
C
C
C
C  PRINT OUT DISPLACEMENTS, 6 PAGES
C

      PRINT 1000
 1000 FORMAT(1H1,50X,13HDISPLACEMENTS,//)
      JCNT = 0
      DO 15 J=1,102
      JCNT = JCNT + 1
      IF(JCNT.LE.18)  GO TO 14
      PRINT 1000
      JCNT = 0
   14 JFIR = 7*(J-1)+1
      JLAST = JFIR + 6
      PRINT 1001, (K,K=JFIR,JLAST)
 1001 FORMAT(1H ,7(8X,4HDEL(,I3,1H)))
      PRINT 1002, (DEL(K),K=JFIR,JLAST)
 1002 FORMAT(1H ,7(2X,E14.7)/)
   15 CONTINUE
      PRINT 1001, (K,K=715,720)
      PRINT 1002, (DEL(K),K=715,720)
C
      I123 = 0
      DO 406 I1=1,9
      DO 406 I2=1,IN
      DO 406 I3=1,3
      I123 = I123+1
C
C  I1 COUNTS FLOORS
C  I2 COUNTS BASE TRIANGLES
C  I3 COUNTS TETRAHEDRONS ABOVE BASE
C  I123 COUNTS ALL OF THEM
C
      KZ = 0
      DO 405  KJ = 1,3
      DO 405  KK = 1,4
      KZ = KZ+1
      ISA = 1
      IPL = 3*(IV(I2,KK,I3)+(I1-1)*MM-1)+KJ
      IF(IPL.GT.864) GO TO 305
      GO TO 306
  305 IF(KJ.EQ.3) GO TO 307
      IPL=IPL-96
      GO TO 306
  307 DX(KZ)=0.
```

```
      GO TO 405
  306 IPM=IPL
      IF(IPL.LE.11)  GO TO 300
      IPL = IPL-10*(IPL/ 96+1)
  300 CONTINUE
      IF(IPM-MSK(IPL)) 409,418,411
  418 IF(KJ.LT.3) ISA = 1-2*ICO(KK,I3,I2)
  410 DX(KZ) = DEL(IPL)   * ISA
      GO TO 405
  409 IF(IPL.EQ.1)  GO TO 416
      IF(IPL.EQ.2)  GO TO 425
      IMR = IPL-1
  412 IF(IPM-MSK(IMR)) 414,419,416
  419 IF(KJ.LT.3) ISA = 1-2*ICO(KK,I3,I2)
  415 DX(KZ) = DEL(IMR) * ISA
      GO TO 405
  414 IMR = IMR-1
      GO TO 412
  416 DX(KZ) = DO
      MLD = MOD(3*(IV(I2,KK,I3)-1)+KJ,3*MM)
      IF(MLD.GT.19.OR.MLD.EQ.2)  DX(KZ) = 0.
      GO TO 405
  411 IMR = IPL - 1
  417 IF(IPM-MSK(IMR)) 416,415,421
  421 IMR = IMR+1
      GO TO 417
  425 DX(KZ) = 0.
  405 CONTINUE
      CALL MXMULT(CDA(1,I3,I2),DX,STRS(1,I123),6,12,1)
C
  406 CONTINUE
C
C  STRS NOW CONTAINS THE SIX STRESS COMPONENTS
C  PRINT OUT CENTROIDS, 710 PAGES
C
      JCNT = 1
      PRINT 1027
      I12 = 0
      DO 50 I1 = 1,9
      DZD = DZ*(I1-1)
      DO 50 I2 = 1, IN
      DO 500 I3= 1,3
      CENT(3,I3,I2) = CENT(3,I3,I2) + DZD
  500 CONTINUE
      I12 = I12 + 1
      I123 = 3*(I12-1) + 1
      I123P2 =  I123+2
      I123M1 = I123-1
      PRINT 1020
      PRINT 1021,((I1,STRS(1,J)),J=I123,I123P2)
      PRINT 1022,((I2,STRS(2,J)),J=I123,I123P2)
      PRINT 1023, ((J,STRS(3,J+I123M1)),J=1,3)
      PRINT 1024,(( CENT(1,J,I2),STRS(4,I123M1+J)),J=1,3)
      PRINT 1025,(( CENT(2,J,I2),STRS(5,I123M1+J)),J=1,3)
      PRINT 1025,(( CENT(3,J,I2),STRS(6,I123M1+J)),J=1,3)
      PRINT 1026
      JCNT = JCNT+1
      IF(JCNT.LT.7)  GO TO 50
      PRINT 1027
      JCNT = 0
   50 CONTINUE
 1020 FORMAT(1H ,3(20X,13HSTRESS VECTOR,6X))
```

```
1021 FORMAT(1H ,3(5HLAYEP,9X,I3,3X,E13.6,6X))
1022 FORMAT(1H ,3(8HTRIANGLE,6X,I3,3X,E13.6,6X))
1023 FORMAT(1H ,3(11HTETRAHEDRON,3X,I3,3X,E13.6,6X))
1024 FORMAT(1H ,3(8HCENTROID,2X,F7.4,3X,E13.6,6X))
1025 FORMAT(1H ,3(10X,F7.4,3X,E13.6,6X))
1026 FORMAT(1H )
1027 FORMAT(1H1)
     RETURN
     END
```

```
      SUBROUTINE INVERT(B,K,K2,XMIN,FLAG)
C  THIS SUBROUTINE SETS UP A UNIT MATRIX ADJACENT TO B(K,K)
C  ELEMENTARY ROW OPERATIONS ARE THEN PERFORMED ON THE NEW K X 2K MATRIX
C  TO REDUCE B(K,K) TO A UNIT MATRIX.  THIS WILL PLACE THE INVERSE OF
C  ELEMENTARY ROW OPERATIONS ARE THEN PERFORMED ON THE NEW K X 2K MATRIX
C  TO REDUCE B(K,K) TO A UNIT MATRIX.  THIS WILL PLACE THE INVERSE OF
C  THE MATRIX B(K,K) IN THE RIGHT HALF OF B(K,2K)
C  ON EXIT, THE INVERSE OF B REPLACES B
C  B IS AN ARRAY OF 2*K**2 LOCATIONS CONTAINING THE MATRIX
C  K IS THE DIMENSION OF THE SQUARE MATRIX B
C  K2 IS 2*K
C  XMIN IS THE SMALLEST ALLOWABLE MAGNITUDE OF THE PIVOT
C  FLAG WILL BE RETURNED AS 0. IF THE INVERSION WENT OFF OK
C  FLAG WILL BE RETURNED AS 10. IF A PIVOT ELEMENT WAS TOO SMALL
C  FLAG SHOULD BE TESTED AFTER EACH CALL TO THIS ROUTINE
C
      DIMENSION B(K,K2)
C
      FLAG = 0.
C
C  SET UP UNIT MATRIX
C
      DO 1  I=1,K
      DO 1  J=1,K
      B(I,K+J) = 0.
      IF(I.EQ.J)  B(I,K+J) = 1.
    1 CONTINUE
C
C  FIND LEADING ELEMENT WITH GREATEST MAGNITUDE
C
      DO 6  J=1,K
      M = J
      N = J+1
      IF(N.GT.K)  GO TO 21
      DO 2  L=N,K
      IF (ABS(B(M,J)).LT.ABS(B(L,J)))  M=L
    2 CONTINUE
   21 CONTINUE
      IF (ABS(B(M,J)).LT.XMIN)  GO TO 10
C
C  INTERCHANGE JTH AND MTH ROWS
C
      DO 3  L=J,K2
      D = B(J,L)
      B(J,L) = B(M,L)
      B(M,L) = D
    3 CONTINUE
C
C  ZERO OUT PIVOTAL JTH COLUMN,  SKIPPING PIVOTAL JTH ELEMENT
C
C  DIVIDE JTH ROW BY PIVOT
C
      DO  4  M=N,K2
      B(J,M) = B(J,M) / B(J,J)
    4 CONTINUE
      DO 6  M=1,K
C
C  M DETERMINES ROW BEING MODIFIED, ONE WHOLE ROW AT A TIME
C
      IF ( M.EQ.J )   GO TO 6
      DO 5  L=N,K2
```

```
C
C   L DETERMINES ELEMENT IN THE MTH ROW
C
       B(M,L) = B(M,L) - B(M,J) * B(J,L)
     5 CONTINUE
     6 CONTINUE
C
C   INVERSE OF B IS NOW IN RIGHT HALF OF B(K,K2)
C   NOW MOVE B INVERSE TO WHERE B WAS
       DO 7  I=1,K
       DO 7  J=1,K
       B(I,J) = B(I,J+K)
     7 CONTINUE
       RETURN
    10 FLAG = 10.
       RETURN
       END
```

```
      SUBROUTINE MXMUL1(A,B,C,M,N,K,M1,N1,L1)
C
C  THIS SUBROUTINE MULTIPLIES MATRIX A BY MATRIX B  AND STORES THE
C  PRODUCT IN C.   (C CANNOT BE THE SAME AS A OR B.)
C  A IS (M X N)
C  B IS (N X K)
C  C IS (M X K)
C
C M1 IS NUMBER OF ROWS IN ARRAY A IN CALLING PROGRAM, M1.GE.M
C  N1  IS NUMBER OF ROWS IN ARRAY B IN CALLING PROGRAM, N1.GE.N
C  L1 IS NUMBER OF ROWS IN ARRAY C IN CALLING PROGRAM, L1.GE.M
C
      DIMENSION  A(M1,N) , B(N1,K) , C(L1 ,K)
C
      DO 1  I=1,M
      DO 1  L=1,K
      C(I,L) = 0.
      DO 1  J=1,N
      C(I,L) = C(I,L) + A(I,J) * B(J,L)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MXCON(A,B,X,M,N)
C
C  THIS SUBROUTINE MULTIPLIES MATRIX A (MXN) BY CONSTANT X, RESULT IN B
C  A MAY BE SAME AS B.
C  THIS SUBROUTINE MULTIPLIES MATRIX A (MXN) BY CONSTANT X, RESULT IN B
C  A MAY BE SAME AS B.
      DIMENSION  A(M,N) , B(M,N)
      DO 1 I=1,M
      DO 1 J=1,N
      B(I,J)= X*A(I,J)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MXMULT(A,B,C,M,N,K)
C
C   THIS SUBROUTINE MULTIPLIES MATRIX A BY MATRIX B  AND STORES THE
C   PRODUCT IN C.    (C CANNOT BE THE SAME AS A OR B.)
C
C   A IS (M X N)
C   B IS (N X K)
C   C IS (M X K)
C
      DIMENSION  A(M,N) , B(N,K) , C(M,K)
C
      DO 1  I=1,M
      DO 1  L=1,K
      C(I,L) = 0.
      DO 1  J=1,N
      C(I,L) = C(I,L) + A(I,J) * B(J,L)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MXSUB(A,B,C,M,N)
C
C  THIS SUBROUTINE SUBTRACTS MATRIX B FROM MATRIX A,STORES RESULT  IN C
C
C  THIS SUBROUTINE SUBTRACTS MATRIX B FROM MATRIX A,STORES RESULT  IN C
C
C  THIS SUBROUTINE SUBTRACTS MATRIX B FROM MATRIX A,STORES RESULT  IN C
C
C  A, B, AND C ARE (M X N)       (C CAN BE THE SAME AS A OR B)
C
      DIMENSION   A(M,N) , B(M,N) ,C(M,N)
C
      DO 1  I=1,M
      DO 1  J=1,N
      C(I,J) = A(I,J) - B(I,J)
    1 CONTINUE
      RETURN
      END
```

1056 CARDS

COMPUTER PROGRAM FOR POINT-MATCHING METHOD

```
$RF4 $NEFF
$T/ 5T$P/ 59T$C/200E$
$FTN,L,P.
      PROGRAM PMATCH
C
C   THIS PROGRAM READS THE INPUTS AND SERVES AS A DRIVER FOR THE SUBROUTINES
C   SOLVING THE TWO-DIMENSIONAL STRESS PROBLEM OF A FIBER-REINFORCED SAMPLE
C   USING THE POINT MATCHING METHOD. THE EQUATION SOLVED IS GX = Y , WHERE
C   G IS (NEQ X NUNK), X IS (NUNK X 1), AND Y IS (NUNK X 1)
C   SHOULD M AND/OR MPR BE CHANGED, DIMENSIONS OF COEF SHOULD BE RE-ESTABLISHED,
C   AS WELL AS DIMENSIONS OF G AND GN, AND NEQ,NX,AND NUNK
      COMMON /1/  G(148,57) , GN(56,57) , E(2) , NU(2) ,
     1      AIN, BIN, CIN,  VF
      DIMENSION ERR(148) , Y(56) ,  GNSAV(56,57)
      TYPE REAL NU
      COMMON /SIZES/ NEQ, M, MPR, NX, NUNK , NUNKP1
      DIMENSION  COMENT(10)
      DATA(PI=3.1415927) , (NEQ=148) , (M=7) , (MPR=7) , (NX=37)
      DATA(ELMIN=1.E-8) , (NUNK=56), (NUNKP1=57)
      JDUM = INIT(1,1)
C
C   INIT IS CALLED HERE TO INITIALIZE BINOMIAL COEFFICIENTS TO ZERO
C
      KASENO =1
      READ 1000,(COMENT(I),I=1,10)
 1000 FORMAT(10A8)
    1 READ 1001,E(1), E(2), NU(1), NU(2),  VF, BIN
 1001 FORMAT(6E10.4)
      IF(BIN.EQ.0.)  GO TO 2
      PRINT 2002 , KASENO
 2002 FORMAT(13H1 CASE NUMBER,I2)
      KASENO = KASENO + 1
      PRINT 2000 ,(COMENT(I),I=1,10)
 2000 FORMAT(1X10A8)
      CIN=SQRT(3.)/2.*BIN
      AIN= SQRT(4.*  CIN*BIN*VF/PI)
      PRINT 2001, E(1), E(2), NU(1), NU(2),  VF, AIN, BIN, CIN
 2001 FORMAT(7X    2HEI,12X,3HEII,11X,4HNUI  ,10X,4HNUII,11X,2HVF,13X,1HA,
     1    13X,1HB,13X,1HC /1X,8(2XE12.5) )
C
C   NEQ - NUMBER OF EQUATIONS IN OVER-DEFINED SYSTEM
C   M - SUMMATION LIMIT, REGION II
C   MPR - SUMMATION LIMIT, REGION I
C   NX - TOTAL NUMBER OF UNKNOWNS , REGION I
C   NUNK - NUMBER OF UNKNOWNS REGION I PLUS REGION II.
C   NUNKP1  -  NUMBER OF UNKNOWNS PLUS 1
C   ELMIN  - MINIMUM ALLOWABLE MAGNITUDE FOR A PIVOTAL ELEMENT
C
C
C   SET UP THE MATRIX G(NEQXNUNK), THE COEFFICIENT MATRIX
C   IN THE OVER-DEFINED SYSTEM  GX = Y , AND ALSO SET UP Y, THE INDEPENDENT
C   VECTOR
      CALL GETG
C
C   NOW HAVE KNOWN MATRIX G AND INDEPENDENT VECTOR Y. GET PRODUCT  GN =
C   (G-TRANSPOSE) * G. GN IS(NUNK X NUNK) .
C
      CALL MXTMUL(G,G,GN,NEQ,NUNK,NUNKP1)
C
C   NOW HAVE LEAST-SQUARES COEFFICIENTS IN GN (NUNK X NUNKP1)
C
C   SAVE GN FOR BACK SUBSTITUTION
```

```
C
      DO 9100 I=1,NUNK
      DO 9100 J=1,NUNKP1
 9100 GNSAV(I,J) = GN(I,J)
C
      CALL EQNSLV(GN,NUNK,NUNKP1,FLAG,ELMIN)
      IF (FLAG.NE.0.)  GO TO 100
C
C  NOW NUNKP1 TH COLUMN OF GN CONTAINS UNKNOWN VECTOR
C
C  CALCULATE ERRORS IN PRIMITIVE SYSTEM AND PRINT THEM
C
      PRINT 6070
 6070 FORMAT( 1H1,20X,26HERRORS IN PRIMITIVE SYSTEM///)
      DO 150  I=1,148
      CALL VECMUL(G,GN(1,NUNKP1),ERR(I),NEQ,NUNK,I)
  150 ERR(I) = G(I,NUNKP1) - ERR(I)
      DO 200  I = 1,21
      IFIR =7*(I-1)+1
      ILAST=6 + IFIR
      IF(I.EQ.15) PRINT 6070
      PRINT 6072,(J,J=IFIR,ILAST)
  200 PRINT 6071,(ERR(J),J=IFIR,ILAST)
      PRINT 6072 , 148
      PRINT 6071 , ERR(148)
 6071 FORMAT(1H ,7(3XE13.6)/ )
 6072 FORMAT(1H ,7(10X,2HX(,I3,1H),)  )
C
C  CALCULATE AND PRINT ERRORS IN SQUARED SYSTEM
C
      PRINT 6080
 6080 FORMAT(1H1,20X,24HERRORS IN SQUARED SYSTEM///)
      DO 180 I=1,56
      CALL VECMUL(GNSAV,   GN(1,NUNKP1),ERR(I),NUNK,NUNK,I)
  180 ERR(I) = GNSAV(I,NUNKP1) -ERR(I)
      DO 190 I=1,8
      IFIR = 7*(I-1)+1
      ILAST = IFIR+6
      PRINT 6072,(J,J=IFIR,ILAST)
  190 PRINT 6071,(ERR(J),J= IFIR,ILAST)
C  SUBSTITUTE UNKNOWNS INTO EQUATIONS FOR STRESSES AND DISPLACEMENTS.
C
      CALL BACKSB
C
C  ALL STRESSES AND DISPLACEMENTS ARE KNOWN.  GET MODULUS EC / EII
C
      GO TO 2
  100 PRINT 2005, ELMIN
 2005 FORMAT(56H1A LARGEST PIVOTAL ELEMENT WAS SMALLER IN MAGNITUDE THAN
     1       ,E11.4)
    2 CONTINUE
      STOP
      END
```

```
      SUBROUTINE GETG
      COMMON /1/  G(148,57) , GN(56,57) , E(2) , NU(2) ,
     1      AIN, BIN, CIN,  VF
      TYPE REAL NU
      COMMON /SIZES/ NEQ, M, MPR, NX, NUNK , NUNKP1
C
C THIS SUBROUTINE GENERATES THE COEFFICIENT MATRIX (G MATRIX)
C OF THE KNOWN OVERDEFINED SYSTEM
C
      TYPE REAL  KAY
      DATA (PI=3.1415927)
      KAY =-E(1)/(1.+NU(1))
C ZERO OUT AUGMENTED MATRIX
      DO 1 I=1,NEQ
      DO 1 J=1,NUNKP1
      G(I,J)=0.
    1 CONTINUE
C EQUATIONS 30 TO 35
      XX=CIN-AIN
      YY=BIN/2.
      CALL SIGX(1,1,XX,YY,1.)
      CALL SIGX(1,2,XX,YY,-1.)
      CALL TAUXY(2,1,XX,YY,1.)
      CALL TAUXY(3,2,XX,YY,1.)
      CALL U(4,1,XX,YY,1.)
      CALL U(4,2,XX,YY,-1.)
      CALL V(5,1,XX,YY,1.)
      CALL V(6,2,XX,YY,1.)
C
C EQUATION 36
      DO 10 J=1,14
      CPJ=COS(PI*J/30.)
      SPJ=SIN(PI*J/30.)
      SPJ2=SIN(PI*J/15.)
      XX=CIN-AIN*CPJ
      YY=BIN/2.-AIN*SPJ
      CALL SIGX(J+6,1,XX,YY,CPJ**2)
      CALL SIGX(J+6,2,XX,YY,-CPJ**2)
      CALL SIGY(J+6,1,XX,YY,SPJ**2)
      CALL SIGY(J+6,2,XX,YY,-SPJ**2)
      CALL TAUXY(J+6,1,XX,YY,SPJ2)
      CALL TAUXY(J+6,2,XX,YY,-SPJ2)
C EQUATION 37
      CPJ2= COS(PI*J/15.)
      SPJ2=SPJ2/2.
      CALL SIGY(J+20,1,XX,YY,SPJ2)
      CALL SIGY(J+20,2,XX,YY,-SPJ2)
      CALL SIGX(J+20,1,XX,YY,-SPJ2)
      CALL SIGX(J+20,2,XX,YY,SPJ2)
      CALL TAUXY(J+20,1,XX,YY,CPJ2)
      CALL TAUXY(J+20,2,XX,YY,-CPJ2)
C
C EQUATION 38
C
      CALL U(J+34,1,XX,YY,CPJ)
      CALL U(J+34,2,XX,YY,-CPJ)
      CALL V(J+34,1,XX,YY,SPJ)
      CALL V(J+34,2,XX,YY,-SPJ)
C
C EQUATION 39
C
      CALL U(J+48,1,XX,YY,SPJ)
```

```
      CALL U(J+48,2,XX,YY,-SPJ)
      CALL V(J+48,1,XX,YY,-CPJ)
      CALL V(J+48,2,XX,YY,CPJ)
   10 CONTINUE
C
C  EQUATIONS 63 TO 68
C
      XX=CIN
      YY=BIN/2.-AIN
      CALL SIGY(63,1,XX,YY,1.)
      CALL SIGY(63,2,XX,YY,-1.)
      CALL TAUXY(64,1,XX,YY,1.)
      CALL TAUXY(65,2,XX,YY,1.)
      CALL U(66,1,XX,YY,1.)
      CALL U(67,2,XX,YY,1.)
      CALL V(68,1,XX,YY,1.)
      CALL V(68,2,XX,YY,-1.)
      G(66,NUNKP1) = KAY
      G(67,NUNKP1) = KAY
C
C  EQUATIONS 46 AND 47
C
      DO 20 J=1,9
      XX=CIN-AIN+J*AIN/10.
      YY=BIN/2.
      CALL TAUXY(J+68,1,XX,YY,1.)
      CALL V(J+77,1,XX,YY,1.)
C
   20 CONTINUE
C  EQUATIONS 48 TO 50
C
      XX=CIN
      YY=BIN/2.
      CALL TAUXY(87,1,XX,YY,1.)
      CALL U(88,1,XX,YY,1.)
      CALL V(89,1,XX,YY,1.)
      G(88,NUNKP1) = KAY
C  EQUATIONS 51 AND 52
C
      DO 25 J=1,9
      YY=BIN/2.-J*(AIN/10.)
      CALL TAUXY(J+89,1,XX,YY,1.)
      CALL U(J+98,1,XX,YY,1.)
C
      G(J+98,NUNKP1) = KAY
   25 CONTINUE
C  EQUATIONS 53 AND 54
C
      DO 30 J=1,4
      YY=BIN/2.-AIN-J*(BIN-AIN)/5.
      CALL TAUXY(J+107,2,XX,YY,1.)
      CALL U(J+111,2,XX,YY,1.)
      G(J+111,NUNKP1) = KAY
   30 CONTINUE
C
C  EQUATIONS 55 TO 57
C
      YY=-BIN/2.
      CALL TAUXY(116,2,XX,YY,1.)
      CALL U(117,2,XX,YY,1.)
      G(117,NUNKP1) = KAY
      CALL V(118,2,XX,YY,1.)
```

```
C
C   EQUATIONS 58 AND 59
C
        DO 40 I=1,12
        XX =(I-1) *(CIN/12.)
        YY=-BIN/2.
        CALL TAUXY(I+118,2,XX,YY,1.)
        CALL V(I+130,2,XX,YY,1.)
     40 CONTINUE
C
C   EQUATIONS 60 AND 61
C
        DO 45 I=1,3
        XX =(I-1) *(CIN-AIN)/3.
        YY=BIN/2.
        CALL TAUXY(I+142,2,XX,YY,1.)
        CALL V(I+145,2,XX,YY,1.)
     45 CONTINUE
        RETURN
        END
```

```
      SUBROUTINE BACKSB
      COMMON /1/  G(148,57) , GN(56,57) , E(2) , NU(2) ,
     1     AIN, BIN, CIN,  VF
      TYPE REAL NU
      COMMON /SIZES/ NEQ, M, MPR, NX, NUNK , NUNKP1
      DIMENSION DELP(2) , AYE(2) , SIGN(20,2) , TAUN(20,2)
      DIMENSION  JI(11,11)
      DIMENSION  XF(20,11) , YF(20,11) , STRESS(20,11,5),PHI(20)
      DIMENSION PHIDEG(20)
      TYPE REAL  JI
C
      TYPE REAL  KAY
      DATA  ((AYE(I),I=1,2)=2H I,2HII)
      KAY =-E(1)/(1.+NU(1))
C   DEVELOP VALUES OF SIGMAX, SIGMAY ON 10X10 GRID, STARTING AT X=0.
C   Y = B/2. , WALKING ALONG ROW TO RIGHT.
C
      DELX = CIN/10.
      DELY = -BIN/10.
      X1 = -DELX
      IPRNT = 0
      DO 10 J=1,11
      IPRNT = IPRNT+1
      GO TO (21,22) , IPRNT
   21 PRINT 1000
      GO TO 23
   22 PRINT 1070
      IPRNT = 0
   23 CONTINUE
      X1 = X1+DELX
      Y0 = BIN/2.-DELY
      DO 10 I=1,11
      Y0 = Y0 + DELY
      XF(I,J) = X1
      YF(I,J) = Y0
      IF((CIN-XF(I,J))**2+(BIN/2.-YF(I,J))**2.GT.AIN**2)
     1GO TO 5
      L = 1
      JI(I,J) = AYE(1)
      GO TO 6
    5 L = 2
      JI(I,J) = AYE(2)
    6 CONTINUE
C
C   KI TELLS WHICH HALF OF UNKNOWN VECTOR TO USE IN MULTIPLICATION
C
C    ZERO OUT MATRIX G AS REQUIRED
C
      DO 7 I21=1,5
      DO 7 J21 = 1,NUNK
    7 G(I21,J21) = 0.
C
      CALL SIGX(1,L,XF(I,J),YF(I,J),1.)
      CALL SIGY(2,L,XF(I,J),YF(I,J),1.)
      CALL TAUXY(3,L,XF(I,J),YF(I,J),1.)
      CALL U(4,L,XF(I,J),YF(I,J),-1./KAY)
      CALL V(5,L,XF(I,J),YF(I,J),-1./KAY)
C
      DO  9 IM=1,5
    9 CALL VECMUL(G,GN(1,NUNKP1), STRESS(I,J,IM), NEQ,NUNK,IM)
C   PRINT OUT GRID VARIABLES
C
```

```
      PRINT 1001, XF(I,J) , YF(I,J) , (STRESS(I,J,K),K=1,5), JI(I,J)
   10 CONTINUE
C
C  STRESS(I,J,1) CONTAINS SIGMA X , I=1,11 ,  J=1,11
C  STRESS(I,J,2) CONTAINS SIGMA Y , I=1,11 ,  J=1,11
C  ET CETERA
C
 1000 FORMAT(1H1,13X,1HX,13X,1HY,8X,6HSIGMAX,8X,6HSIGMAY,9X,5HTAUXY,13X,
     1      1HU,13X,1HV,10X,6HREGION/)
 1001 FORMAT(1H 2(3X,F11.4),5(3X,E11.4),12X,A2 /)
 1070 FORMAT(////)
C
C  NOW MOVE ALONG EDGE OF FIBER,FROM THE TOP, COUNTER-CLOCKWISE.
C
      DELPHI = 6.
      PHI1 = 0.
      DO 33 I=1,14
      PHI1=PHI1+DELPHI
      PHIDEG(I) = PHI1
      PHI(I) = PHI1 /57.29578
      XF(I,1)=CIN-AIN*COS(PHI(I))
      YF(I,1)=BIN/2.-AIN*SIN(PHI(I))
      DO 29 J=1,2
C
C   ZERO OUT MATRIX G AS REQUIRED
C
      DO 28 I21=1,5
      DO 28 J1 = 1,NUNK
   28 G(I21,J1)= 0.
      CALL SIGX(1, J, XF(I,1),YF(I,1),1.  )
      CALL SIGY(2, J, XF(I,1),YF(I,1),1.  )
      CALL TAUXY(3,J, XF(I,1),YF(I,1),1.  )
      CALL      U(4,J, XF(I,1),YF(I,1),-1./KAY)
      CALL      V(5,J, XF(I,1),YF(I,1),-1./KAY)
      DO 29 K=1,5
   29 CALL VECMUL(G,GN(1,NUNKP1), STRESS(I,K,J),NEQ,NUNK,K )

C
C
C  SIGMAX J  IN STRESS(I,1,J) , J=1,2
C  SIGMAY J  IN STRESS(I,2,J) , J=1,2
C  TAUXY  J  IN STRESS(I,3,J) , J=1,2
C  U      J  IN STRESS(I,4,J) , J=1,2
C  V      J  IN STRESS(I54,J) , J=1,2
C
C
      DO 30 J=1,2
      SIGN(I,J) = STRESS(I,1,J)*COS(PHI(I))**2+STRESS(I,2,J)
     1             * SIN(PHI(I))**2+STRESS(I,3,J)*2.
     2            * SIN(PHI(I))* COS(PHI(I))
   30 TAUN(I,J) = (STRESS(I,2,J)-STRESS(I,1,J))*SIN(PHI(I))
     1             * COS(PHI(I)) + STRESS(I,3,J)
     2            *(COS(PHI(I))**2 - SIN(PHI(I))**2)
   33 CONTINUE
C
C  NOW HAVE VALUES ON INTERFACE,PRINT THEM OUT
C
 1005 FORMAT(1H1)
      PRINT 1037
 1037 FORMAT(1H1,15X,19HVALUES ON INTERFACE,////)
      DO 44 J=1,2
      DO 44 I=1,14
```

```
      IF(I.EQ.1)  PRINT 1010
      PRINT 1011, AYE(J),PHIDEG(I),XF(I,1),YF(I,1),SIGN(I,J),TAUN(I,J)
     1      ,STRESS(I,4,J),  STRESS(I,5,J)
 1010 FORMAT(8H REGION ,  8X,3HPHI,13X,1HX,14X1HY,11X,7HSIGMA N, 9X,
     1      5HTAU N,11X,1HU,14X,1HV/)
      IF(I.EQ.14) PRINT 1012
 1012 FORMAT(////)
   44 CONTINUE
 1011 FORMAT(1H 4X,A2,4X,F10.2,5X,6(3X,E12.5))
C
C  NOW FIND VALUE OF P
C
C  DELP(J)--INTEGRATION STEP SIZE IN REGION J
C
      P = 0.
      P1 = 0.
      DELP(1) = AIN / 20.
      DELP(2) = (BIN-AIN) / 20.
      Y1 = BIN/2.+DELP(1)
      DO 60 J=1,2
C
C  J=1, REGION I ,   J=2, REGION II
C
      DO 60 I = 1,20
C
C  I STEPS ALONG LINE X=C
C
      IF(I.EQ.1.AND.J.EQ.2) GO TO 60
      Y1 = Y1 - DELP(J)
C
C   ZERO OUT MATRIX G AS REQUIRED
C
      DO 50   J21 =1,NUNK
   50 G(1,J21) = 0.
      CALL SIGX(I,J,CIN,Y1,1.)
      CALL VECMUL(G,GN(1,NUNKP1),SX,NEQ,NUNK,1)
C
C  INTEGRATE SX FOR P, USING SIMPSONS RULE.
C  THE INTEGRAL IS SET EQUAL TO  ZERO ON THE FIRST PASS.
C  TRAPEZOIDAL INTEGRATION IS USED ON THE SECOND PASS
C  WHEN J GOES FROM ONE TO TWO, THE INTEGRATION STEP CHANGES SO THAT
C  AGAIN SIMPSON,S RULE COMMENCES ON I=3.
C
      IF(I-2) 56,52,53
   52 P = P1 + DELP(J)*(SX+SX1)/2.
      GO TO 55
   53 P = P2 + DELP(J)*(SX2+ 4.*SX1+ SX) / 3.
   55 CONTINUE
C
C  SAVE PAST VALUES OF P,SX
C
      P2 = P1
      P1 = P
   56 SX2 = SX1
      SX1= SX
      PRINT 9091,I,P,P1,P2
 9091 FORMAT(3H I=,I3,3HP =,E13.6,3HP1=,E13.6,3HP2=,E13.6)
      PRINT 9092, SX,SX1,SX2
 9092 FORMAT(4H SX=,E13.6,4HSX1=,E13.6,4HSX2=,E13.6)
   60 CONTINUE
C  P  - PRESENT VALUE OF INTEGRAL
C  P1 - 1ST PAST VALUE OF P
```

```
C   P2 - 2ND PAST VALUE OF P
C   SX - PRESENT VALUE OF SIGMAX
C   SX1 - 1ST PAST VALUE OF SIGMAX
C   SX2 - 2ND PAST VALUE OF SIGMAX
C
C
C   NOW HAVE P, PRINT IT
C
      PRINT 1020,P
 1020 FORMAT(////4H P= ,E15.8)
      RETURN
      END
```

```
      SUBROUTINE VECMUL(A,B,C,I,J,K)
C
C  THIS SUBROUTINE MULTIPLIES THE KTH ROW OF MATRIX A TIMES
C     THE COLUMN VECTOR STORED IN B AND STORES THE RESULT IN C.
C
C  MATRIX A HAS I ROWS
C  MATRIX A HAS J COLUMNS
      DIMENSION  A(I,J) , B(J)
      C = 0.
      DO 1 L=1,J
    1 C = C+A(K,L)*B(L)
      RETURN
      END
      SUBROUTINE EQNSLV( B ,K ,KP1, FLAG,ELMIN)
C
C  THIS SUBROUTINE DOES A GAUSSIAN ELIMINATION PROCEDURE ON THE
C  AUGMENTED MATRIX  B  = AY ,  WHERE  AX=Y, X UNKNOWN.
C  THE INDEPENDENT VECTOR Y  STORED AS THE (K+1)TH COLUMN OF  B .
C   B   IS (KXK+1). THE SOLUTION ALGORITHM PROGRESSES ACROSS
C  COLUMNS TO THE RIGHT, CHOOSING THE COLUMN ELEMENT WITH LARGEST
C  MODULUS AS THE PIVOT. IF THIS ELEMENT IS LESS THAN ELMIN, FLAG
C  IS SET TO 10. AND A RETURN TO THE CALLING PROGRAM IS EXECUTED.
C  FLAG IS RETURNED AS 0. IF NORMAL COMPLETION OCCURS. FLAG SHOULD
C  ALWAYS BE TESTED ON RETURN TO CALLING ROUTINE.
C  KP1 IS K+1, REQUIRED FOR VARIABLE DIMENSIONING
C  ON RETURN, SOLUTION VECTOR IS IN LAST COLUMN OF  B , REPLACING Y.
C
      DIMENSION  B(K,KP1)
C
      FLAG = 0.
C
C
C  FIND LEADING ELEMENT WITH GREATEST MAGNITUDE
C
      DO 6  J=1,K
      M = J
      N = J+1
      DO 2  L=N,K
      IF (ABS(B(M,J)).LT.ABS(B(L,J)))   M=L
    2 CONTINUE
      IF (ABS(B(M,J)).LT.ELMIN) GO TO 10
C
C  INTERCHANGE JTH AND MTH ROWS
C
      DO 3 L=J,KP1
      D = B(J,L)
      B(J,L) = B(M,L)
      B(M,L) = D
    3 CONTINUE
C
C  ZERO OUT PIVOTAL JTH COLUMN,  SKIPPING PIVOTAL JTH ELEMENT
C
C  DIVIDE JTH ROW BY PIVOT
C
      DO 4 M=N,KP1
      B(J,M) = B(J,M) / B(J,J)
    4 CONTINUE
      DO 6  M=1,K
C
C  M DETERMINES ROW BEING MODIFIED, ONE WHOLE ROW AT A TIME
C
```

```
      IF ( M.EQ.J )    GO TO 6
      DO 5 L=N,KP1
C
C  L DETERMINES ELEMENT IN THE MTH ROW
      B(M,L) = B(M,L) - B(M,J) * B(J,L)
    5 CONTINUE
    6 CONTINUE
C
      RETURN
   10 FLAG = 10.
      RETURN
      END
```

```
      FUNCTION COEF(I,J)
C
C  THIS FUNCTION CALCULATES THE BINOMIAL COEFFICIENT ( I )
C                                                    ( J )
C
      DIMENSION BINOM(15,15 )
      IF(J.NE.0) GO TO 1
      COEF = 1.
      GO TO 31
    1 COEF = BINOM(I,J)
      IF(COEF.NE.0.) RETURN
      IF(J-1) 2,2,3
    2 COEF = I
      GO TO 31
    3 CONTINUE
      K= I-J+1
      COEF = 1.
      DO 10 L =K,I
      COEF = COEF*L
   10 CONTINUE
      DO 20 N =1,J
      COEF = COEF/ N
   20 CONTINUE
      BINOM(I,J) = COEF
      RETURN
      ENTRY INIT
      DO 30 K=1,15
      DO 30 L=1,15
      BINOM(K,L) = 0.
   30 CONTINUE
   31 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INVERT(B,K,K2,XMIN,FLAG)
C   THIS SUBROUTINE SETS UP A UNIT MATRIX ADJACENT TO B(K,K)
C   ELEMENTARY ROW OPERATIONS ARE THEN PERFORMED ON THE NEW K X 2K MATRIX
C   TO REDUCE B(K,K) TO A UNIT MATRIX.  THIS WILL PLACE THE INVERSE OF
C   THE MATRIX B(K,K) IN THE RIGHT HALF OF B(K,2K)
C   ON EXIT, THE INVERSE OF B REPLACES B
C   B IS AN ARRAY OF 2*K**2 LOCATIONS CONTAINING THE MATRIX
C   K IS THE DIMENSION OF THE SQUARE MATRIX B
C   K2 IS 2*K
C   XMIN IS THE SMALLEST ALLOWABLE MAGNITUDE OF THE PIVOT
C   FLAG WILL BE RETURNED AS 0. IF THE INVERSION WENT OFF OK
C   FLAG WILL BE RETURNED AS 10. IF A PIVOT ELEMENT WAS TOO SMALL
C   FLAG SHOULD BE TESTED AFTER EACH CALL TO THIS ROUTINE
C
      DIMENSION B(K,K2)
C
      FLAG = 0.
C
C   SET UP UNIT MATRIX
C
      DO 1  I=1,K
      DO 1  J=1,K
      B(I,K+J) = 0.
      IF(I.EQ.J)  B(I,K+J) = 1.
    1 CONTINUE
C
C   FIND LEADING ELEMENT WITH GREATEST MAGNITUDE
C
      DO 6  J=1,K
      M = J
      N = J+1
      DO 2  L=N,K
      IF (ABS(B(M,J)).LT.ABS(B(L,J)))  M=L
    2 CONTINUE
      IF (ABS(B(M,J)).LT.XMIN)  GO TO 10
C
C   INTERCHANGE JTH AND MTH ROWS
C
      DO 3  L=J,K2
      D = B(J,L)
      B(J,L) = B(M,L)
      B(M,L) = D
    3 CONTINUE
C
C   ZERO OUT PIVOTAL JTH COLUMN,  SKIPPING PIVOTAL JTH ELEMENT
C
C   DIVIDE JTH ROW BY PIVOT
C
      DO  4  M=N,K2
      B(J,M) = B(J,M) / B(J,J)
    4 CONTINUE
      DO 6  M=1,K
C
C   M DETERMINES ROW BEING MODIFIED, ONE WHOLE ROW AT A TIME
C
      IF ( M.EQ.J )   GO TO 6
      DO 5  L=N,K2
C
C   L DETERMINES ELEMENT IN THE MTH ROW
C
      B(M,L) = B(M,L) - B(M,J) * B(J,L)
```

```
    5 CONTINUE
    6 CONTINUF
C
C   INVERSE OF B IS NOW IN RIGHT HALF OF B(K,K2)
C   NOW MOVE B INVERSE TO WHERE B WAS
      DO 7  I=1,K
      DO 7  J=1,K
      B(I,J) = B(I,J+K)
    7 CONTINUE
      RETURN
   10 FLAG = 10.
      RETURN
      END
```

```
      FUNCTION MINUS1(N)
C
C  THIS FUNCTION RETURNS (-1)**N , 0.LE.N.LE.19
C
      DIMENSION M(20)
      DATA((M(I),I=1,20)= 1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,
     1                    1,-1 )
      MINUS1 = M(N+1)
      RETURN
      END
```

```
      SUBROUTINE NORMAL(A,PROD,M,N)
C
C  THIS SUBROUTINE MULTIPLIES A*(A-TRANSPOSE), PUTS PRODUCT INTO PROD
C  A IS (MXN), PROD IS THEREFORE (M X M)
C
      DIMENSION A(M,N) , PROD(M,M)
C
      DO 1 I=1,M
      DO 1 L=1,N
      PROD(I,L)=0.
      DO 1 J=1,N
    1 PROD(I,L) = PROD(I,L) + A(I,J)*A(L,J)
      RETURN
      END
```

```
      SUBROUTINE MXMULT(A,B,C,M,N,K)
C
C  THIS SUBROUTINE MULTIPLIES MATRIX A BY MATRIX B  AND STORES THE
C  PRODUCT IN C.    (C CANNOT BE THE SAME AS A OR B.)
C
C  A IS (M X N)
C  B IS (N X K)
C  C IS (M X K)
C
      DIMENSION  A(M,N) , B(N,K) , C(M,K)
C
      DO 1  I=1,M
      DO 1  L=1,K
      C(I,L) = 0.
      DO 1  J=1,N
      C(I,L) = C(I,L) + A(I,J) * B(J,L)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MXTMUL(A,B,C,M,N,K)
C
      DIMENSION  A(M,N), B(M,K) , C(N,K)
C  THIS SUBROUTINE MULTIPLIES (A-TRANSPOSE) * B, PUTS PRODUCT IN C
C  A IS (M X N)
C  A-TRANSPOSE IS (N X M)
C  B IS (M X K)
C  C IS (N X K)
      DO 1 I=1,N
      DO 1 L=1,K
      C(I,L) = 0.
      DO 1 J=1,M
    1 C(I,L) = C(I,L) + A(J,I) * B(J,L)
      RETURN
      END
```

```
      SUBROUTINE SIGX(IROW ,II    ,X,Y ,CONST)
      COMMON /SIZES/ NEQ, M, MPR, NX, NUNK , NUNKP1
      COMMON /INIT/  INITA,INITB,INITC(2),INITD(2), INITK1(2), INITK2,
     1              INITK3(2) , INITK4 , INITK5, INITK6(2) , INITK7,
     2              INITK8(2) , INITK9 , INITK10, INITK11(2), INITOM(2)
      COMMON /1/  G(148,57) , GN(56,57) , E(2) , NU(2) ,
     1       AIN, BIN, CIN,  VF
      TYPE REAL NU
      DATA  (INITA=10) , (INITB=16) , (INITC=22,41) , (INITD=29,48) ,
     1      (INITK1=1,38) , (INITK2=2),(INITK3=3,39), (INITK4=4)      ,
     2      (INITK5=5) , (INITK6=6,40) , (INITK7=7) , (INITK8=8,41) ,
     3      (INITK9=9) , (INITK10=10)   , (INITK11=11,42) ,(INITOM=37,56)
C
C  THE VARIOUS INITS DEFINE THE POSITION OF THE UNKNOWNS IN THE G MATRIX
C
C  INITA+1 WILL BE THE COLUMN OF UNKNOWN A1, REGION I
C  INITB+1 WILL BE THE COLUMN OF UNKNOWN B1, REGION I
C  INITC(J)+1 WILL BE THE COLUMN OF C1, REGION J, J=I,II
C  INITD(J)+1 WILL BE THE COLUMN OF D1, REGION J, J=I,II
C
C  (NOTE THAT A1,B1,C1 DO NOT EXIST, SO THAT A1 OVERLIES INITK11,
C  B1 OVERLIES A7, AND C1 OVERLIES B7)
C
C  INITKJ WILL BE COLUMN OF KJ , J=2,4,5,7,9,10
C  INITKJ(L) WILL BE COLUMN OF KJ, REGION L , J = 1,3,6,8,11 , L=I,II
C  INITOM(J) WILL BE COLUMN OF OMEGA ZERO IN REGION J , J=I,II
C
C  A , B , K2, K4, K5, K7, K9, K10, DO NOT EXIST IN REGION II
C
      G(IROW,INITK1(II) ) = G(IROW,INITK1(II))+ CONST* 6.*X*Y
      G(IROW,INITK6(II) ) = G(IROW,INITK6(II))+ CONST* 2.
      GO TO (5,10), II
    5 G(IROW,INITK2)       = G(IROW,INITK2) + CONST* 2.*X
      G(IROW,INITK5)       = G(IROW,INITK5) + CONST* 6.*Y
      MII = MPR
      GO TO 15
   10 MII = M
   15 DO 45 M1 = 1, MII
      M3 = M1 + 1
      DO 45 N2 = 1,M3
      N1 = N2 - 1
      IF(M1.NE.N1) GO TO 20
      XD = 1.
      XC = X
      XB = XD
      XA = XC
      GO TO 25
   20 XD = X**(2*(M1-N1))
      XC = X * XD
      XB = XD
      XA = XC
   25 IF(N1.NE.0) GO TO 30
      YD =1.
      YC =0.
      YB =YC
      YA =YC
      GO TO 35
   30 IF(N1.NE.1)  GO TO 33
      YA = 1.
      GO TO 34
   33 YA = Y**(2*N1-2)
   34 YB = Y * YA
```

```
      YC = YB
      YD = Y*YC
   35 G(IROW,INITD(II)+M1)  = G(IROW,INITD(II)+M1)+ CONST
     1       * MINUS1(N1)*2.*(N1+1)*(2.*N1+1)* YD * XD
     2      * COEF(2*M1+1 ,2*N1+1)
      IF(M1.EQ.1) GO TO 44
C
      G(IROW,INITC(II)+M1)  = G(IROW,INITC(II)+M1)+ CONST
     1       * MINUS1(N1)*2.*N1*(2.*N1+1.) * YC * XC
     2      * COEF(2*M1+1 ,2*N1+1)
      GO TO (40,44),II
   40 G(IROW,INITB+M1) = G(IROW,INITB+M1) + CONST
     1       * MINUS1(N1)*2.*N1*(2.*N1+1.) * YB * XB
     2      * COEF(2*M1,2*N1)
      G(IROW,INITA+M1) = G(IROW,INITA+M1) + CONST
     1       * MINUS1(N1)*2.*N1*(2.*N1-1.) * YA * XA
     2      * COEF(2*M1,2*N1)
   44 CONTINUE
   45 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SIGY(IROW,II ,X,Y,CONST)
      TYPE REAL NU
      COMMON /SIZES/ NEQ, M, MPR, NX, NUNK , NUNKP1
      COMMON /1/  G(148,57) , GN(56,57) , E(2) , NU(2) ,
     1     AIN, BIN, CIN,  VF
      COMMON /INIT/  INITA,INITB,INITC(2),INITD(2), INITK1(2), INITK2,
     1          INITK3(2) , INITK4 , INITK5, INITK6(2) , INITK7,
     2          INITK8(2) , INITK9 , INITK10, INITK11(2), INITOM(2)
      G(IROW,INITK8(II) ) = G(IROW,INITK8(II) ) + CONST * 6.* X * Y
      G(IROW,INITK11(II)) = G(IROW,INITK11(II)) + CONST * 2.
      GO TO (5,10), II
    5 G(IROW,INITK9)  = G(IROW,INITK9) + CONST * 2.* Y
      G(IROW,INITK10) = G(IROW,INITK10) + CONST * 6. * X
      MII = MPR
      GO TO 15
   10 MII = M
   15 DO 45 M1 = 1, MII
      M3 = M1 + 1
      DO 45 N2 = 1,M3
      N1 = N2 - 1
      IF(M1.NE.N1+1) GO TO 20
      XD = 1.
      XC = X
      XB = XD
      XA = XC
      GO TO 30
   20 IF(M1.NE.N1) GO TO 25
      XD = 0.
      XC = 0.
      XB = 0.
      XA = 0.
      GO TO 30
   25 XD = X**(2*(M1-N1-1))
      XC = X* XD
      XB = XD
      XA = XC
   30 IF(N1.NE.0) GO TO 35
      YA = 1.
      GO TO 40
   35 YA = Y **(2 *N1)
   40 YB = Y * YA
      YC = YB
      YD = Y * YB
      G(IROW,INITD(II)+M1)  = G(IROW,INITD(II)+M1)+ CONST
     1      * MINUS1(N1)*2.*(M1-N1)*(2.*(M1-N1)-1.) * YD *XD
     2      * COEF(2*M1+1 ,2*N1+1)
      IF(M1.EQ.1) GO TO 44
C
      G(IROW,INITC(II)+M1)  = G(IROW,INITC(II)+M1)+ CONST
     1      *(2.*(M1-N1)+1.)*2.*(M1-N1)*MINUS1(N1) * XC * YC
     2      * COEF(2*M1+1 ,2*N1+1)
      GO TO (42,44),II
   42 G(IROW,INITB+M1) = G(IROW,INITB+M1) + CONST
     1      *  MINUS1(N1)*2*(M1-N1)*(2.*(M1-N1)-1.) *YB * XB
     2      * COEF(2*M1,2*N1)
      G(IROW,INITA+M1) = G(IROW,INITA+M1) + CONST
     1      *  MINUS1(N1)*2.*(M1-N1)*(2.*(M1-N1)+1.) * XA * YA
     2      * COEF(2*M1,2*N1)
   44 CONTINUE
   45 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  TAUXY(IROW,   II,X,Y,CONST)
      COMMON /1/  G(148,57) , GN(56,57) , E(2) , NU(2) ,
     1      AIN, BIN, CIN,  VF
      COMMON /SIZES/ NEQ, M, MPR, NX, NUNK , NUNKP1
      COMMON /INIT/  INITA,INITB,INITC(2),INITD(2), INITK1(2), INITK2,
     1          INITK3(2) , INITK4 , INITK5, INITK6(2) , INITK7,
     2          INITK8(2) , INITK9 , INITK10, INITK11(2), INITOM(2)
      TYPE REAL NU
C
      G(IROW,INITK1(II)) = G(IROW,INITK1(II)) -3.*Y**2 * CONST
      G(IROW,INITK3(II)) = G(IROW,INITK3(II)) -CONST
      G(IROW,INITK8(II)) = G(IROW,INITK8(II)) - CONST * 3.*X**2
      GO TO (5,10), II
    5 G(IROW,INITK2)     = G(IROW,INITK2)     - 2.* CONST * Y
      G(IROW,INITK9)     = G(IROW,INITK9)     - 2.* CONST * X
      MII = MPR
      GO TO 15
   10 MII = M
   15 DO 45 M1 = 1, MII
      M3 = M1 + 1
      DO 45 N2 = 1, M3
      N1 = N2 - 1
      IF(M1.NE.N1) GO TO 20
      XD = 0.
      XC = 1.
      XB = 0.
      XA = 1.
      GO TO 25
   20 XD = X**(2*(M1-N1)-1)
      XC = X * XD
      XB = XD
      XA = XC
   25 IF(N1.NE.0 ) GO TO 30
      YA = 0.
      YB = 1.
      GO TO 35
   30 YA = Y ** (2*N1-1)
      YB = Y * YA
   35 YC = YB
      YD = Y*YC
      G(IROW,INITD(II)+M1) = G(IROW,INITD(II)+M1) - CONST
     1      * MINUS1(N1) *(2.*N1+2.) * 2.*(M1-N1)* YD *XD
     2     * COEF(2*M1+1 ,2*N1+1)
      IF(M1.EQ.1) GO TO 44
C
      G(IROW,INITC(II)+M1) = G(IROW,INITC(II)+M1) - CONST
     1      * MINUS1(N1)* (2.*N1+1.) *(2.*(M1-N1)+1.) * YC * XC
     2     * COEF(2*M1+1 ,2*N1+1)
      GO TO (40,44),II
   40 G(IROW,INITB+M1)      = G(IROW,INITB+M1) - CONST
     1      * MINUS1(N1)* (2.*N1+1)* 2.*(M1-N1)   * YB * XB
     2     * COEF(2*M1,2*N1)
      G(IROW,INITA+M1)      = G(IROW,INITA+M1) - CONST
     1      * MINUS1(N1)*2.*N1*(2.*(M1-N1)+1.)*XA*YA
     2     * COEF(2*M1,2*N1)
   44 CONTINUE
   45 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE U(IROW,II,X,Y,CONST)
      COMMON /1/  G(148,57) , GN(56,57) , E(2) , NU(2) ,
     1     AIN, BIN, CIN,  VF
      COMMON /SIZES/ NEQ, M, MPR, NX, NUNK , NUNKP1
      COMMON /INIT/  INITA,INITB,INITC(2),INITD(2), INITK1(2), INITK2,
     1          INITK3(2) , INITK4 , INITK5, INITK6(2) , INITK7,
     2          INITK8(2) , INITK9 , INITK10, INITK11(2), INITOM(2)
      TYPE REAL NU
      ONENU = 1. - NU(II)
      TWONU = 2. - NU(II)
      REVNU = 0. - NU(II)
C
      ECON=((II-1)*((1.+NU(2))*E(1)/(E(2)*(1.+NU(1)))-1.)+1.)* CONST
C
      G(IROW,INITK1(II)) = G(IROW,INITK1(II))+ECON*(3.*ONENU*Y*X**2
     1                   - TWONU*Y**3 )
      G(IROW,INITK3(II)) = G(IROW,INITK3(II))+ECON*(-TWONU*Y)
      G(IROW,INITK6(II)) = G(IROW,INITK6(II))+ECON*(2.*ONENU*X)
      G(IROW,INITK8(II)) = G(IROW,INITK8(II))+ECON*(3.*REVNU*Y*X**2
     1                   - ONENU * Y**3)
      G(IROW,INITK11(II))= G(IROW ,INITK11(II))+ECON*(2.*REVNU*X)
      G(IROW,INITOM(II)) = G(IROW ,INITOM(II))+ECON*(ONENU*Y)
      GO TO (5,10), II
    5 G(IROW,INITK2) = G(IROW,INITK2)+ECON*(ONENU*X**2-TWONU*Y**2)
      G(IROW,INITK4) = G(IROW,INITK4)+ECON*REVNU
      G(IROW,INITK5) = G(IROW,INITK5)+ECON*(6.*ONENU*X*Y)
      G(IROW,INITK9) = G(IROW,INITK9)+ECON*(2.*REVNU*X*Y)
      G(IROW,INITK10)= G(IROW,INITK10)+ECON*(-3.*ONENU* Y**2
     1            +3.*REVNU*X**2)
      MII = MPR
      GO TO 15
   10 MII = M
   15 DO 45   MI = 1, MII
      M3 = M1 + 1
      DO 45   N2 = 1, M3
      N1 =  N2 - 1
      IF(M1.NE.N1) GO TO 20
      XD2 = 0.
      XC2 = 1.
      XB2 = 0.
      XA2 = 1.
      GO TO 25
   20 XD2 = X **(2*(M1-N1)-1)
      XC2 = X * XD2
      XB2 = XD2
      XA2 = XC2
   25 XD1 = X **(2*(M1-N1)+1)
      XC1 = X * XD1
      XB1 = XD1
      XA1 = XC1
      IF(N1.NE.0) GO TO 30
      YD1 = 1.
      YC1 = 0.
      YB1 = 0.
      YA1 = 0.
      YA2 = 1.
      GO TO 40
   30 IF(N1.NE.1) GO TO 35
      YD1 = Y**2
      YC1 = Y
      YB1 = Y
      YA1 = 1.
```

```
      YA2 = Y**2
      GO TO 40
   35 YA1 = Y**(2*N1-2)
      YB1 = Y* YA1
      YC1 = YB1
      YD1 = Y* YC1
      YA2 = YD1
   40 YB2 = Y*YA2
      YC2 = YB2
      YD2 = Y*YC2
      G(IROW,INITD(II)+M1)  = G(IROW,INITD(II)+M1)+ ECON
     1      * MINUS1(N1) * (ONENU*(2.*N1+1.)*2.*(N1+1.)/(2.*(M1-N1)+1.)
     2      * YD1 * XD1 +2.*REVNU *(M1-N1) * XD2 * YD2  )
     3      * COEF(2*M1+1 ,2*N1+1)
C
      IF(M1.EQ.1) GO TO 44
C
      G(IROW,INITC(II)+M1)  = G(IROW,INITC(II)+M1)+ ECON
     1      * MINUS1(N1) * (ONENU*(2.*N1+1.)* N1/(M1-N1+1.) * YC1 * XC1
     2      + REVNU * (2.*(M1-N1)+1.)   * YC2 *XC2 )
     3      * COEF(2*M1+1 ,2*N1+1)
      GO TO (42,44),II
   42 G(IROW,INITB+M1) = G(IROW,INITB+M1) + ECON
     1      * MINUS1(N1)*(ONENU*(2.*N1+1)*2.*N1/(2.*(M1-N1)+1.) * YB1*XB1
     2      + 2.*REVNU*(M1-N1) *YB2 * XB2  )
     3      * COEF(2*M1,2*N1)
      G(IROW,INITA+M1) = G(IROW,INITA+M1) + ECON
     1      * MINUS1(N1) * (ONENU*   N1*(2.*N1-1)/    (M1-N1 +1.)
     2      * YA1*XA1 + REVNU*(2.*(M1-N1)+1.) * YA2 * XA2 )
     3      * COEF(2*M1,2*N1)
   44 CONTINUE
   45 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE V(IROW,II,X,Y,CONST)
      COMMON /1/  G(148,57) , GN(56,57) , E(2) , NU(2) ,
     1     AIN, BIN, CIN,  VF
      COMMON /SIZES/ NEQ, M, MPR, NX, NUNK , NUNKP1
      COMMON /INIT/  INITA,INITB,INITC(2),INITD(2), INITK1(2), INITK2,
     1          INITK3(2) , INITK4 , INITK5, INITK6(2) , INITK7,
     2          INITK8(2) , INITK9 , INITK10, INITK11(2), INITOM(2)
      TYPE REAL NU
      REVNU = 0. - NU(II)
      TWONU = 2. - NU(II)
      ONENU = 1. - NU(II)
C
      ECON=((II-1)*((1.+NU(2))*E(1)/(E(2)*(1.+NU(1)))-1.)+1.)* CONST
C
      G(IROW,INITK3(II)) = G(IROW,INITK3(II))+ECON*REVNU*X
      G(IROW,INITK1(II)) = G(IROW,INITK1(II))+ECON*(3*REVNU*X*Y**2
     1                     - ONENU* X**3)
      G(IROW,INITK6(II)) = G(IROW,INITK6(II))+ECON*REVNU*2.*Y
      G(IROW,INITK8(II)) = G(IROW,INITK8(II))+ECON*(ONENU*X*Y**2 *3.
     1                     - TWONU * X**3)
      G(IROW,INITK11(II)) = G(IROW,INITK11(II))+ ECON*2.*ONENU*Y
      G(IROW,INITOM(II)) =  G(IROW,INITOM(II)) + ECON*(-ONENU*X)
      GO TO (5,10), II
    5 G(IROW,INITK2)= G(IROW,INITK2) + ECON* 2.*REVNU * X * Y
      G(IROW,INITK5)= G(IROW,INITK5) + ECON*3*(-ONENU*X**2+REVNU* Y**2)
      G(IROW,INITK7)= G(IROW,INITK7) + ECON*REVNU
      G(IROW,INITK9)= G(IROW,INITK9) + ECON*(ONENU*Y**2-TWONU* X**2)
      G(IROW,INITK10) = G(IROW,INITK10) +ECON* ONENU*6.*X*Y
      MII = MPR
      GO TO 15
   10 MII = M
   15 DO 45 M1 = 1, MII
      M3 = M1 + 1
      DO 45  N2 = 1, M3
      N1 = N2 - 1
      IF(M1.NE.N1) GO TO 20
      XA1 = 0.
      XB1 = 0.
      XC1 = 0.
      XD1 = 0.
      XA2 = X
      XB2 = 1.
      XC2 = X
      XD2 = 1.
      GO TO 30
   20 IF(M1.NE.N1+1) GO TO 25
      XB1 = 1.
      XD1 = 1.
      GO TO 28
   25 XB1 = X**(2*(M1-N1-1))
      XD1 = XB1
   28 XC1 = X * XB1
      XA1 = XC1
      XB2 = X* XA1
      XA2 = X* XB2
      XC2 = XA2
      XD2 = XB2
   30 IF(Y.NE.0) GO TO 35
      YA2 = 0.
      YB2 = 1.
      YC2 = 1.
      GO TO 40
```

```
35 YA2 = Y**(2*N1-1)
   YB2 = Y*YA2
   YC2 = YB2
40 YD2 = Y*YB2
   YA1 = YD2
   YB1 = Y*YA1
   YC1 = YB1
   YD1 = Y*YC1
   G(IROW,INITD(II)+M1)  = G(IROW,INITD(II)+M1)+ ECON
  1        * MINUS1(N1) *(ONENU*2.*(M1-N1)*(2.*(M1-N1)-1.)/(2.*N1+3.)
  2        * XD1 * YD1 + REVNU*2. *(N1+1.) * YD2 * XD2)
  3       * COEF(2*M1+1 ,2*N1+1)
   IF(M1.EQ.1) GO TO 44
   G(IROW,INITC(II)+M1)  = G(IROW,INITC(II)+M1)+ ECON
  1        * MINUS1(N1) *(ONENU*(2.*(M1-N1)+1.)*   (M1-N1)/(N1+1.)
  2        * YC1 * XC1 + REVNU*(2.*N1+1.) * YC2 *XC2 )
  3       * COEF(2*M1+1 ,2*N1+1)
   GO TO (42,44),II
42 G(IROW,INITB+M1) = G(IROW,INITB+M1) + ECON
  1        * MINUS1(N1)*(ONENU*(M1-N1)*(2.*(M1-N1)-1.) /(N1+1.)
  2        * XB1 * YB1 + REVNU * (2.*N1+1.) * YB2 * XB2)
  3       * COEF(2*M1,2*N1)
   G(IROW,INITA+M1) = G(IROW,INITA+M1) + ECON
  1        * MINUS1(N1)*(ONENU*(2.*(M1-N1)+1.) *2.*(M1-N1)/(2.*N1+1.)
  2        * YA1 * XA1 + REVNU*2.*N1 * YA2 * XA2 )
  3       * COEF(2*M1,2*N1)
44 CONTINUE
45 CONTINUE
   RETURN
   END
                                                        1089 CARDS
```

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author): WHITTAKER CORPORATION, Research & Development/San Diego, San Diego, California
2a. REPORT SECURITY CLASSIFICATION: Unclassified
2b. GROUP:
3. REPORT TITLE: TRANSVERSE LOADING OF UNIDIRECTIONAL FIBER COMPOSITES
4. DESCRIPTIVE NOTES (Type of report and inclusive dates): Final Report (26 December 1967 to 26 December 1968
5. AUTHOR(S) (First name, middle initial, last name): Dr. Juan A. Haener, Alberto H. Puppo, Ming-yuan Feng
6. REPORT DATE: July 1969
7a. TOTAL NO. OF PAGES: 131
7b. NO. OF REFS:
8a. CONTRACT OR GRANT NO.: DAAJ02-68-C-0037
b. PROJECT NO.: Task 1F162204A17002
c.
d.
9a. ORIGINATOR'S REPORT NUMBER(S): USAAVLABS Technical Report 69-45
9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report): Whittaker MJO 341 Final Report
10. DISTRIBUTION STATEMENT: This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of US Army Aviation Materiel Laboratories, Fort Eustis, Virginia 23604
11. SUPPLEMENTARY NOTES:
12. SPONSORING MILITARY ACTIVITY: U.S. Army Aviation Materiel Laboratories, Fort Eustis, Virginia

13. ABSTRACT

The study is a part of an effort being directed toward solving the problem of a composite material subjected to general oblique loading.

This analysis was conducted in order to find the internal micromechanics of a fiber-reinforced composite due to transverse normal loading. Special emphasis ~~has been~~ given to studying the stress distribution near free surfaces, which led to solving a three-dimensional elasticity problem. The numerical method of finite elements ~~has been~~ employed in this analysis. On the other hand, it was necessary to study the behavior far from free surfaces. For this purpose, a two-dimensional program was used. Findings from these two approaches were anticipated, showing that stress conditions become two dimensional a relatively short distance from the end of the composite. Extensive parametric studies ~~have been~~ performed from the combined outputs of these two schemes. Significant diagrams exhibiting elastic properties of different composite materials have been obtained.

DD FORM 1 NOV 65 1473 REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE.

Unclassified

Security Classification

Unclassified

Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
COMPOSITE MATERIALS						
MICROMECHANICS						
TRANSVERSE LOADING						
ELASTIC CONSTANTS						

Unclassified

Security Classification

7622-69