ABSTRACT

SHIPMAN, BENJAMIN HOWARD. Calculation of the Generalized Stress Intensity Factors for a V-notched Anisotropic Body. (Under the direction of Dr. F. G. Yuan)

A robust method for calculating a generalized stress intensity factor for a V-notched anisotropic body under symmetric and/or anti-symmetric deformation is derived for plane stress or plane strain. The compact formulation for the generalized stress intensity factors is derived based on Stroh formalism. A path-independent line integral together with an auxiliary field solution, called the interaction *M*-integral, is utilized to solve for these generalized stress intensity factors. Through numeric evaluation of the interaction *M*-integral using a finite element solution, the generalized stress intensity factors can be found. These generalized stress intensity factors can be used to predict the failure conditions without the need for a detailed notch-tip field solution. Since the interaction *M*-integral is path-independent, the calculation can be carried out in the region away from the notch tip where a conventional finite element solution is sufficient to perform this analysis.

Numeric results for the generalized stress intensity factors are given for a thin rectangular plate with double edge notches. The specimen geometry used follows that in the ASTM standard D 5379/D 5379M–93 for shear property testing of fiber-reinforced composite materials. The method is first verified for three example problems. Then, the generalized stress intensity factors are given for a wide range of notch depths and angles

for isotropic and anisotropic material property cases. Two in-plane fiber orientations of a unidirectional fiber-reinforced graphite/epoxy composite are considered. Two loading cases are given to produce symmetric and anti-symmetric deformation. The generalized stress intensity factor results given here for anti-symmetric deformation are unprecedented.

CALCULATION OF THE GENERALIZED STRESS INTENSITY FACTORS FOR A V-NOTCHED ANISOTROPIC BODY

by

Benjamin Howard Shipman

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

AEROSPACE ENGINEERING

Raleigh
2002

Approved By:

Dr. Fuh-Gwo Yuan
Chairman of Advisory Committee

Dr. Jeffrey W. Eischen

Dr. Eric C. Klang

BIOGRAPHY

Benjamin Howard Shipman was born in Warner Robins, Georgia, U.S.A, in October 1975. He graduated from Eastern Wayne High School located in Goldsboro, North Carolina, U.S.A, in June 1994. He received his Bachelor of Science degree in Aerospace Engineering from North Carolina State University (NCSU) in Raleigh, North Carolina, U.S.A in May 1998 and continued his graduate studies there in August 1998 under the direction of Dr. F. G. Yuan at the Mars Mission Research Center (MMRC). His research for his Master of Science degree in Aerospace Engineering focused on computational fracture mechanics of composite materials. He participated in the Langley Aerospace Research Summer Scholars (LARSS) program at the NASA Langley Research Center (LaRC) from June through August of 1999. In this program he analyzed a mixed-mode bending delamination test (4MMB) for fiber-reinforced composite materials proposed by Dr. James R. Reeder to determine if it was better than the previous test designed by Dr. Reeder (MMB).

ACKNOWLEDGEMENTS

First, I would like to thank my advisor, Dr. F. G. Yuan for the direction and support he has given me for my entire graduate career. I have grown much professionally from his knowledge, experience, enthusiasm, and desire to help whether he is in his office day or night. I appreciate the contribution Dr. S. Yang gave with developing the mathematical formulation. I also thank my other committee members, Dr. J. W. Eischen and Dr. E. C. Klang, for the education they have given me on my committee and in the classroom. I am glad for the support for this research from the North Carolina Space Grant Consortium. I would also like to thank the North Carolina Supercomputer Center for their computing support.

I have had many great teachers while I have spent most of my life in school. I would like to thank them for their contributions to my academic success.

I also thank my family for their support. Without the help of my parents, Howard and Ellen, I could not have come this far. I thank my sisters, Carolyn and Heather, for their friendship too.

I cannot forget the enjoyable time I spent with my friends Brian and Gary and their support that I am very grateful for.

TABLE OF CONTENTS

List of Tables

List of Figures

Nomenclature

Latin symbols:

$\boldsymbol{A}$	Complex matrix containing Stroh eigenvectors
a	Notch depth
$\boldsymbol{B}$	Complex matrix containing Stroh eigenvectors
C_{ijkm}	Elastic stiffness tensors
d_k	Constant in Lekhnitskii Formalism
E_1, E_2, E_3	Young's moduli of an orthotropic material
$\boldsymbol{g}_n$	Complex constant vectors in Stroh formalism
$\widetilde{\boldsymbol{g}}^{(s)}$	Normalized first complex vector for symmetric deformation
$\widetilde{\boldsymbol{g}}^{(a)}$	Normalized first complex vector for anti-symmetric deformation
G_{12}, G_{23}, G_{13}	Shear moduli of an orthotropic material
$\boldsymbol{h}_m$	Auxiliary constant vectors in Stroh formalism
$\boldsymbol{h}^{(s)}$	Auxiliary first complex vector for symmetric deformation
$\boldsymbol{h}^{(a)}$	Auxiliary first complex vector for anti-symmetric deformation
$\mathbf{I}$	Identity matrix
i	$\sqrt{-1}$
k_1, k_2	Normalization factors in Stroh eigenvectors
$k^{(a)}$	Generalized stress intensity factor for anti-symmetric deformation
$\widetilde{k}^{(a)}$	Non-dimensional generalized stress intensity factor for anti-symmetric deformation
$k^{(s)}$	Generalized stress intensity factor for symmetric deformation
$\widetilde{k}^{(s)}$	Non-dimensional generalized stress intensity factor for symmetric deformation
M_M	Interaction M-integral

n_j	The *j*-th directional component of the unit normal vector
p_1, p_2	Expressions in Stroh eigenvectors
q_1, q_2	Expressions in Stroh eigenvectors
$\boldsymbol{s}$	Compliance matrix
s_{ij}	Components of compliance matrix
t_i	Traction vector
t_i^a	Traction vector for auxiliary field
$\boldsymbol{u}$	Displacement vector
$\boldsymbol{u}^a$	Auxiliary displacement vector
u_i	Components of displacement vector
u_i^a	Components of auxiliary displacement vector
x_1, x_2	Cartesian coordinates of a local (notch tip) coordinate system
W	Strain energy density
w	Specimen width

Greek symbols:

α	Notch angle
δ_n	Stress exponents for the actual field
$\delta^{(s)}$	First stress exponent for symmetric deformation
$\delta^{(a)}$	First stress exponent for anti-symmetric deformation
Δ_m	Stress exponents for the auxiliary field
ε_{ij}	Components of strain tensor
ε_{ij}^a	Components of auxiliary strain tensor
Φ	Complex stress potential function

Φ^a	Auxiliary stress potential function
Γ	An arbitrary path around the notch tip
μ_α	Eigenvalues of the elastic constants
σ_{ij}	Components of stress tensor
σ_{ij}^a	Components of auxiliary stress tensor

1 Introduction

According to small deformation elasticity theory, a V-notch in an elastic body may cause a stress singularity at the tip. The high stress values at the notch tip caused by this singularity can cause fracture leading to catastrophic failure of structures made of brittle materials such as fiber-reinforced graphite/epoxy composites. Therefore, accurate prediction of the near-tip field is important. The singular nature of the stress field makes it difficult to accurately calculate stresses near notch tips using regular finite element or boundary element methods. Special techniques have been used to adapt finite elements for use in regions of singular fields such as the singular hybrid finite element method (Pian *et al.*, 1973), the enriched finite element method (Im *et al.*, 1996), and hybrid finite element approach for composite materials (Wang and Yuan, 1983). An advantage of the present method is that it is used with a regular displacement-based finite element method without singular elements being needed at the notch tip. Instead of depending on the finite element solution to be accurate near the notch tip, the present method uses the finite element solution away from the notch tip where it is more accurate. The present method uses a finite element solution to obtain generalized stress intensity factors through the use of a path-independent line integral. The path-independent line integral used in this method is the interaction M-integral, M_M.

The interaction M-integral is a line-integral that is path independent for the V-notch. This path-independence of an interactive M-integral is achieved by using an auxiliary field solution along with the actual field solution. This auxiliary field solution satisfies the zero traction boundary conditions on the notch surfaces and the elastic field governing equations. The concept of the interaction M-integral has been applied to a generic isotropic wedge (Im and Kim, 2000). The use of the interaction M-integral allows the far-field solution to be used to find the near-tip field. Therefore, because this

method uses the interaction *M*-integral, elements that can accurately predict the singular field near the notch tip, such as singular finite elements, are not needed.

The generalized stress intensity factor is derived from the first-term or singular-term solution to the Stroh field equations (Yuan, 1998). Deriving the generalized stress intensity factor from the Stroh field equations is advantageous in that the method can be applied to any material property case from anisotropic to isotropic. Taking advantage of this material property generality, the present method is applied to a fiber-reinforced composite material.

The present method also has the advantage that it can be applied to any loading case. For the results here, the loading cases are separated into those causing both symmetric and anti-symmetric deformations. The symmetric deformation case is the result of uniform tensile loading on the opposite boundaries of the body. Wu and Chen produced some generalized stress intensity factor results for a V-notched body under symmetric deformation (Wu and Chen, 1996). For the results here, the present method is applied to a much wider range of notch angle and notch depth cases. The results here also cover the case of anti-symmetric deformation. The loading to produce anti-symmetric deformation is derived from the Iosipescu shear test (Iosipescu, 1967). The Iosipescu shear test uses a double V-notched specimen to find shear properties of a material through experimental testing. The loading used for an Iosipescu shear test produces pure shear loading (no bending moment) on the line between the double V-notches, which produces anti-symmetric deformation at the notch tips. The Iosipescu shear test is applied in the ASTM standard D 5379/D 5379M-93 for shear property testing of fiber-reinforced composite materials. The same specimen geometry was used for the results here as in D 5379/D 5379M-93. After determining the critical generalized stress intensity factors experimentally, this method could be used to predict the failure of the specimens used in D 5379/D 5379M-93. The results here were produced for two orthogonal in-plane fiber orientations. These two fiber orientations were also used by Kumosa and Hull in their stress analysis of an Iosipescu shear test specimen (Hull and

Kumosa, 1987). As in the results by Kumosa and Hull, the results here show that under anti-symmetric deformation the stress components decrease to zero at the notch tip instead of going to infinity after some critical notch angle is reached.

2 Mathematical Formulation

In this section, the mathematical formulation used to obtain the "generalized stress intensity factors", $k^{(s)}$ and $k^{(a)}$, for symmetric and anti-symmetric deformation, respectively, is presented. The parameters $k^{(s)}$ and $k^{(a)}$ characterize the singular stress field at the V-notch tip and so can therefore be used to predict when fracture will occur at the tip. The solutions for $k^{(s)}$ and $k^{(a)}$ are derived through substituting the first-term solution to the Stroh field equations into the interaction *M*-integral equation. In this section, the Stroh formalism for anisotropic elasticity is briefly discussed. Then, because the Lekhnitskii formalism was used in the FORTRAN program SINGULARITY for finding the stress exponents, the in-plane Lekhnitskii field solution is presented. Then, the formulation of the interaction *M*-integral is presented. Finally, $k^{(s)}$ and $k^{(a)}$ are extracted from the interaction *M*-integral. Once $k^{(s)}$ and $k^{(a)}$ are isolated in terms of the interaction *M*-integral, values for $k^{(s)}$ and $k^{(a)}$ can be calculated through evaluating the interaction *M*-integral numerically using an actual finite element field solution with auxiliary field solution.

2.1 Stroh Formalism

The Stroh formalism provides elegant and compact stress and displacement field solutions for homogeneous anisotropic elastic solids under two-dimensional deformations. First, the actual field solution is presented. Then, a puesdo field, the auxiliary field solution, is presented. The auxiliary field solution is used to make the interaction *M*-integral path-independent.

2.1.1 Actual Field Solution

The stress and displacement components can be placed in the form of a complex potential function (Yuan, 1998)

$$\boldsymbol{\Phi} = \boldsymbol{B}\langle f(z)\rangle \boldsymbol{B}^{-1}\boldsymbol{g} \tag{2.1}$$

Then, the stress components are given by

$$\boldsymbol{\sigma}_{i1} = -\mathrm{Re}[\boldsymbol{\Phi}_{i,2}] \quad \boldsymbol{\sigma}_{i2} = \mathrm{Re}[\boldsymbol{\Phi}_{i,1}] \tag{2.2}$$

where $_{,k}$ denotes differentiation with respect to the k th Cartesian coordinate.

And, the displacement vector $\boldsymbol{u}$ is expressed by

$$\boldsymbol{u} = \mathrm{Re}[\boldsymbol{A}\langle f(z)\rangle \boldsymbol{B}^{-1}\boldsymbol{g}] \tag{2.3}$$

or

$$\boldsymbol{u} = \mathrm{Re}[\boldsymbol{A}\boldsymbol{B}^{-1}\boldsymbol{\Phi}] \tag{2.4}$$

Here, $f(z)$ is an arbitrary function, $\boldsymbol{g}$ is an unknown complex constant vector depending on material properties, loading, and geometry, and $\boldsymbol{A}$ and $\boldsymbol{B}$ are the corresponding Stroh matrices of eigenvectors. For plane stress or plane strain in the x_1-x_2 plane,

$\langle f(z)\rangle$ is a diagonal matrix defined as $\langle f(z)\rangle = diag[f(z_1), f(z_2)]$

$$z_\alpha = x_1 + \mu_\alpha x_2, \ \mathrm{Im}[\mu_\alpha] > 0, \ \alpha = 1, 2$$

where μ_α are the eigenvalues of the elastic constants and x_1 and x_2 are Cartesian coordinates with the origin at the notch tip.

To provide a notch-tip field solution, an eigenfunction expansion series is suggested as

$$\boldsymbol{\Phi} = \sum_n \boldsymbol{B}\langle z_\alpha^{\delta_n+1}\rangle \boldsymbol{B}^{-1}\boldsymbol{g}_n \tag{2.5}$$

$$\boldsymbol{u} = \sum_n \mathrm{Re}[\boldsymbol{A}\langle z_\alpha^{\delta_n+1}\rangle \boldsymbol{B}^{-1}\boldsymbol{g}_n] \tag{2.6}$$

Since here we are only interested in predicting fracture at the notch tip, only a first-term or singular-term solution is needed. The singular-term solution represents the field

solution as the notch tip is approached ($r \to 0$) where the field terms become pronouncedly affected by the notch.

If the material axis is aligned with the global axis, the compliance matrix is

$$\boldsymbol{s} = \begin{bmatrix} s_{11} & s_{12} & s_{13} & 0 & 0 & 0 \\ s_{12} & s_{22} & s_{23} & 0 & 0 & 0 \\ s_{13} & s_{23} & s_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & s_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & s_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & s_{66} \end{bmatrix} = \begin{bmatrix} \frac{1}{E_1} & \frac{-\nu_{12}}{E_1} & \frac{-\nu_{13}}{E_1} & 0 & 0 & 0 \\ \frac{-\nu_{12}}{E_1} & \frac{1}{E_2} & \frac{-\nu_{23}}{E_2} & 0 & 0 & 0 \\ \frac{-\nu_{13}}{E_1} & \frac{-\nu_{23}}{E_2} & \frac{1}{E_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{23}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{13}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{12}} \end{bmatrix} \tag{2.7}$$

Under the assumption of plane strain, μ_α are the roots of the characteristics equation

$$s'_{11}\mu^4 - 2s'_{16}\mu^3 + (2s'_{12} + s'_{66})\mu^2 - 2s'_{26}\mu + s'_{22} = 0 \tag{2.8}$$

with positive imaginary parts where $s'_{ij} = s'_{ji}$ are the reduced compliance coefficients defined by $s'_{ij} = s_{ij} - s_{i3}s_{j3}/s_{33}$. In the above and following equations, s'_{ij} is replaced by s_{ij} for the plane stress case.

A and ***B*** are the Stroh matrices of eigenvectors, which are

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{a}_1 & \boldsymbol{a}_2 \end{bmatrix} = \begin{bmatrix} p_1 & p_2 \\ q_1 & q_2 \end{bmatrix} \tag{2.9}$$

where

$$p_\alpha = s'_{11}\mu_\alpha^2 - s'_{16}\mu_\alpha + s'_{12}, \quad q_\alpha = s'_{12}\mu_\alpha - s'_{26} + s'_{22}/\mu_\alpha \tag{2.10}$$

and

$$\boldsymbol{B} = [\boldsymbol{b}_1 \quad \boldsymbol{b}_2] = \begin{bmatrix} -\mu_1 & -\mu_2 \\ 1 & 1 \end{bmatrix}, \; \boldsymbol{B}^{-1} = \frac{1}{\mu_1 - \mu_2} \begin{bmatrix} -1 & -\mu_2 \\ 1 & \mu_1 \end{bmatrix} \tag{2.11}$$

The normalization factors k_α can be introduced so that

$$\boldsymbol{A} = \begin{bmatrix} k_1 p_1 & k_2 p_2 \\ k_1 q_1 & k_2 q_2 \end{bmatrix} \qquad \boldsymbol{B} = \begin{bmatrix} -k_1 \mu_1 & -k_2 \mu_2 \\ k_1 & k_2 \end{bmatrix} \tag{2.12}$$

where k_α can be solved from

$$2k_1^2(q_1 - p_1\mu_1) = 1 \quad 2k_2^2(q_2 - p_2\mu_2) = 1 \tag{2.13}$$

The matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ defined by Eq. (2.12) satisfy the orthogonality relations (Ting, 1996)

$$\boldsymbol{A}^T\boldsymbol{B} + \boldsymbol{B}^T\boldsymbol{A} = \mathbf{I} \quad \text{and} \quad \boldsymbol{A}^T\overline{\boldsymbol{B}} + \boldsymbol{B}^T\overline{\boldsymbol{A}} = \mathbf{0} \tag{2.14}$$

where $\mathbf{I}$ is the 2×2 identity matrix for two dimensions.

2.1.2 Auxiliary Field Solution

For each eigenfunction term of the actual field, there is a corresponding higher-order auxiliary field term that also satisfies both the boundary condition of zero traction on the notch surfaces and the field governing equations for anisotropic solids. However, the auxiliary field is unrealistic because each eigenfunction with higher-order singularites has unbounded strain energy near the notch tip and thus corresponds to some concentrated source at the tip. Superimposing the actual field term with a unique order auxiliary field term, makes an interaction integral path-independent. Chosing the

auxiliary field stress exponents, Δ_m, to satisfy $\delta_n + \Delta_m + 2 = 0$ makes the interaction *M*-integral path-independent. The auxiliary stress potential equation is

$$\boldsymbol{\Phi}^a = \boldsymbol{B}\left\langle z^{\Delta_m+1}\right\rangle \boldsymbol{B}^{-1}\boldsymbol{h}_m \qquad m = 1, 2, 3, \cdots. \tag{2.15}$$

where Δ_m is the higher-order stress exponent and $\boldsymbol{h}_m$ is the corresponding eigenvector. And, the auxiliary field stress components are

$$\sigma_{i1}^a = -\mathrm{Re}[\boldsymbol{\Phi}_{i,2}^a] \quad \sigma_{i2}^a = \mathrm{Re}[\boldsymbol{\Phi}_{i,1}^a] \tag{2.16}$$

And, the auxiliary displacement vector is

$$\boldsymbol{u}^a = \mathrm{Re}[\boldsymbol{A}\left\langle z^{\Delta_m+1}\right\rangle \boldsymbol{B}^{-1}\boldsymbol{h}_m] \tag{2.17}$$

or

$$\boldsymbol{u}^a = \mathrm{Re}[\boldsymbol{A}\boldsymbol{B}^{-1}\boldsymbol{\Phi}^a] \tag{2.18}$$

2.2 Lekhnitskii Formalism

Here, the field solution from Lekhnitskii formalism for homogeneous anisotropic elastic solids is presented. The in-plane stress and displacement component equations are given. These stress and displacement equations are used in the FORTRAN program SINGULARITY in Appendix A.1 to solve for the stress exponent, δ. Solving for δ using the Lekhnitskii formalism was easier to implement in SINGULARITY than through using the Stroh formalism. The stress exponents from Lekhnitskii and Stroh formalisms are equivalent.

The in-plane stress components are (Lekhnitskii, 1981)

$$\sigma_1 = 2\sum_{k=1}^{2}\mathrm{Re}[\mu_k^2 F_k''(z_k)],\ \sigma_2 = 2\sum_{k=1}^{2}\mathrm{Re}[F_k''(z_k)],\ \sigma_{12} = -2\sum_{k=1}^{2}\mathrm{Re}[\mu_k F_k''(z_k)] \tag{2.19}$$

where μ_α are the roots of the characteristics equation given by Eq. (2.8).

The in-plane displacement components are

$$u_1 = 2\sum_{k=1}^{2} \mathrm{Re}[\, p_k F_k'(z_k)], \;\; u_2 = 2\sum_{k=1}^{2} \mathrm{Re}[q_k F_k'(z_k)] \tag{2.20}$$

where p_k and q_k are the complex constants from Eq. (2.10). The form chosen for the unknown function F_k is

$$F_k = d_k \frac{z_k^{\delta+2}}{(\delta+1)(\delta+2)}, \; F_k' = d_k \frac{z_k^{\delta+1}}{(\delta+1)}, \; F_k'' = z_k^{\delta} d_k \tag{2.21}$$

where d_k are constants.

Expanding the in-plane stress components gives

$$\sigma_1 = 2\sum_{k=1}^{2} \mathrm{Re}[\mu_k^2 F_k''(z_k)] = 2\,\mathrm{Re}[\mu_1^2 z_1^{\delta} d_1 + \mu_2^2 z_2^{\delta} d_2] \tag{2.22}$$

$$\sigma_2 = 2\sum_{k=1}^{2} \mathrm{Re}[F_k''(z_k)] = 2\,\mathrm{Re}[z_1^{\delta} d_1 + z_2^{\delta} d_2] \tag{2.23}$$

$$\sigma_{12} = 2\sum_{k=1}^{2} \mathrm{Re}[(-\mu_k) F_k''(z_k)] = -2\,\mathrm{Re}[\mu_1 z_1^{\delta} d_1 + \mu_2 z_2^{\delta} d_2] \tag{2.24}$$

Expanding the in-plane displacement components gives

$$u_1 = 2\sum_{k=1}^{2} \mathrm{Re}[\, p_k F_k'(z_k)] = 2\,\mathrm{Re}[\, p_1 \frac{z_1^{\delta+1}}{(\delta+1)} d_1 + p_2 \frac{z_2^{\delta+1}}{(\delta+1)} d_2] \tag{2.25}$$

$$u_2 = 2\sum_{k=1}^{2} \mathrm{Re}[q_k F_k'(z_k)] = 2\,\mathrm{Re}[q_1 \frac{z_1^{\delta+1}}{(\delta+1)} d_1 + q_2 \frac{z_2^{\delta+1}}{(\delta+1)} d_2] \tag{2.26}$$

Using the identity $\mathrm{Re}[z] = \frac{1}{2}[z + \bar{z}]$, where the over-bar denotes the complex conjugate, the stress components can be put in the form

$$\sigma_1 = [\mu_1^2 z_1^{\delta} d_1 + \bar{\mu}_1^2 \bar{z}_1^{\delta} \bar{d}_1 + \mu_2^2 z_2^{\delta} d_2 + \bar{\mu}_2^2 \bar{z}_2^{\delta} \bar{d}_2] \tag{2.27}$$

$$\sigma_2 = [z_1^{\delta} d_1 + \bar{z}_1^{\delta} \bar{d}_1 + z_2^{\delta} d_2 + \bar{z}_2^{\delta} \bar{d}_2] \tag{2.28}$$

$$\sigma_{12} = [-\mu_1 z_1^{\delta} d_1 - \bar{\mu}_1 \bar{z}_1^{\delta} \bar{d}_1 - \mu_2 z_2^{\delta} d_2 - \bar{\mu}_2 \bar{z}_2^{\delta} \bar{d}_2] \tag{2.29}$$

and the displacement components can be put in the form

$$u_1 = [\, p_1 \frac{z_1^{\delta+1}}{(\delta+1)} d_1 + \overline{p}_1 \frac{\overline{z}_1^{\delta+1}}{(\delta+1)} \overline{d}_1 + p_2 \frac{z_2^{\delta+1}}{(\delta+1)} d_2 + \overline{p}_2 \frac{\overline{z}_2^{\delta+1}}{(\delta+1)} \overline{d}_2 \,] \tag{2.30}$$

$$u_2 = [\, q_1 \frac{z_1^{\delta+1}}{(\delta+1)} d_1 + \overline{q}_1 \frac{\overline{z}_1^{\delta+1}}{(\delta+1)} \overline{d}_1 + q_2 \frac{z_2^{\delta+1}}{(\delta+1)} d_2 + \overline{q}_2 \frac{\overline{z}_2^{\delta+1}}{(\delta+1)} \overline{d}_2 \,] \tag{2.31}$$

2.3 Generalized Stress Intensity Factor Derivation

The field solution for a V-notched body can be determined from the asymptotic solution with multiplicative constants depending on the material properties, notch angle, loading, and geometry. These constants are needed to complete the field solution. A path-independent line integral called the interaction *M*-integral can be utilized to obtain these constants numerically. Here, the constants for the first-term or singular-term solution called the generalized stress intensity factors, $k^{(s)}$ and $k^{(a)}$, are solved in terms of the interaction *M*-integral.

The interaction *M*-integral is a conservation integral for two elastic equilibrium states. If two elastic states are represented by *A* and *B*, and the interaction *M*-integral is M_M, then a superimposed elastic state is expressed as

$$M = M^A + M^B + M_M \tag{2.32}$$

The *M*-integral for an elastic state is

$$M = \int_\Gamma (W n_k - t_i \frac{\partial u_i}{\partial x_k}) x_k ds \tag{2.33}$$

where n_k are components of the unit outward normal vector on the path Γ, σ_{ij} are the stress components, ε_{ij} are the strain components, C_{ijkm} are the elastic stiffness tensors, W is the strain energy density ($W = C_{ijkm}\varepsilon_{ij}\varepsilon_{km}/2$), t_i are the traction components

($t_i = \sigma_{ij} n_j$), u_i are the displacement components, x_k are Cartesian coordinates, and ds is an increment of the path around the notch tip, Γ. Figure 2.1 shows the arbitrary integration path, Γ, which starts on the lower straight face of the notch and ends on the upper straight face of the notch enclosing the notch tip. The origin of the Cartesian coordinate system is at the notch tip and the angles to the notch surfaces are ϕ_1 and ϕ_2.

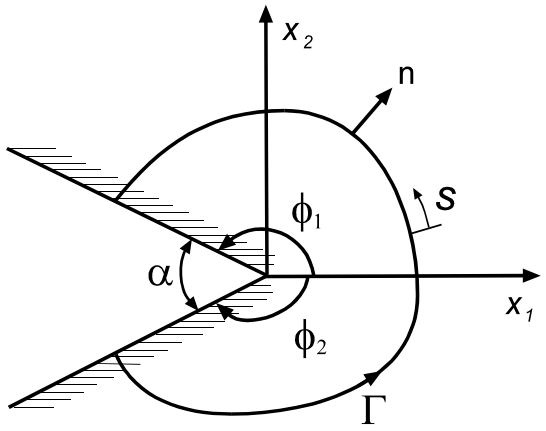

Figure 2.1 An arbitrary integration path, Γ, around a V-notch in a body.

From Eqs. (2.32) and (2.33), the interaction *M*-integral, M_M, is

$$M_M = \int_\Gamma [C_{ijkm} \varepsilon_{ij}^A \varepsilon_{km}^B n_n - (t_i^A \frac{\partial u_i^B}{\partial x_n} + t_i^B \frac{\partial u_i^A}{\partial x_n})] x_n ds \tag{2.34}$$

If the two elastic states are the actual and auxiliary fields (where the superscript *a* is used to denote the auxiliary field) and a comma is used to indicate partial differentiation with respect to the Cartesian coordinate x_k, the interaction *M*-integral is rewritten as

$$M_M = \int_\Gamma [\sigma_{ij} \varepsilon_{ij}^a n_k - (t_i u_{i,k}^a + t_i^a u_{i,k})] x_k ds \tag{2.35}$$

where $\sigma_{ij} = C_{ijkm} \varepsilon_{km}$. The interaction *M*-integral will be denoted by M_M from here on.

Now, the generalized stress intensity factors, $k^{(s)}$ and $k^{(a)}$, are defined and derived from the interaction *M*-integral using Stroh formalism. Substituting $\varepsilon_{ij}^a = u_{i,j}^a$, and $t_i = \sigma_{ij} n_j$ into the first two terms of the integrand and using $n_1 = \frac{dx_2}{ds}$ and $n_2 = -\frac{dx_1}{ds}$ gives

$$\sigma_{ij} x_k (u_{i,j}^a n_k - u_{i,k}^a n_k) = (\sigma_{i2} x_1 - \sigma_{i1} x_2) \frac{du_i^a}{ds} \tag{2.36}$$

Therefore, the new form for M_M is

$$M_M = \int_\Gamma [(\sigma_{i2} x_1 - \sigma_{i1} x_2) \frac{du_i^a}{ds} - t_i^a u_{i,k} x_k] ds \tag{2.37}$$

Now, the first bracketed term of the integrand of M_M, $(\sigma_{i2} x_1 - \sigma_{i1} x_2) du_i^a$, is derived in terms of the Stroh formalism. Substituting the stress component equations, $\sigma_{i1} = -\mathrm{Re}[\boldsymbol{\Phi}_{i,2}]$ and $\sigma_{i2} = \mathrm{Re}[\boldsymbol{\Phi}_{i,1}]$, gives

$$(\sigma_{i2} x_1 - \sigma_{i1} x_2) du_i^a = du_i^a \, \mathrm{Re}[x_1 \Phi_{i,1} + x_2 \Phi_{i,2}] = d(\boldsymbol{u}^a)^T \, \mathrm{Re}[x_1 \boldsymbol{\Phi}_{,1} + x_2 \boldsymbol{\Phi}_{,2}] \tag{2.38}$$

Substituting the auxiliary displacement equation, $\boldsymbol{u}^a = \mathrm{Re}[\boldsymbol{A}\boldsymbol{B}^{-1}\boldsymbol{\Phi}^a]$, gives

$$d(\boldsymbol{u}^a)^T \, \mathrm{Re}[x_1 \boldsymbol{\Phi}_{,1} + x_2 \boldsymbol{\Phi}_{,2}] = d[\mathrm{Re}(\boldsymbol{A}\boldsymbol{B}^{-1}\boldsymbol{\Phi}^a)^T \, \mathrm{Re}(z\boldsymbol{\Phi}')] \tag{2.39}$$

From the identity $\mathrm{Re}(\boldsymbol{C}) \, \mathrm{Re}(\boldsymbol{D}) = \frac{1}{2} \mathrm{Re}[\boldsymbol{C}(\boldsymbol{D} + \overline{\boldsymbol{D}})]$,

$$d[\mathrm{Re}(\boldsymbol{A}\boldsymbol{B}^{-1}\boldsymbol{\Phi}^a)^T \, \mathrm{Re}(z\boldsymbol{\Phi}')] = \frac{1}{2} \mathrm{Re}[(\boldsymbol{A}\boldsymbol{B}^{-1} d\boldsymbol{\Phi}^a)^T (z\boldsymbol{\Phi}' + c.c.)] \tag{2.40}$$

where c.c. denotes the complex conjugate of the preceding term, i.e. $F + c.c. = F + \overline{F}$.

Now, the other term of the integrand of M_M, $t_i^a u_{i,k} x_k ds$, is derived in terms of the Stroh formalism. Using $t_i^a = -\mathrm{Re}\left[\frac{d\boldsymbol{\Phi}_i^a}{ds}\right]$, the term gives

$$t_i^a u_{i,k} x_k ds = -\mathrm{Re}(\frac{d\boldsymbol{\Phi}_i^a}{ds}) u_{i,k} x_k ds \tag{2.41}$$

Expanding for two dimensions so that $u_{i,k} x_k = u_{i,1} x_1 + u_{i,2} x_2$ and substituting $\boldsymbol{u} = \mathrm{Re}[\boldsymbol{A}\boldsymbol{B}^{-1}\boldsymbol{\Phi}]$ into Eq. (2.41) results in

$$-\mathrm{Re}(\frac{d\boldsymbol{\Phi}_i^a}{ds}) u_{i,k} x_k ds = -\mathrm{Re}(d\boldsymbol{\Phi}^a)^T\ \mathrm{Re}[\boldsymbol{A}\boldsymbol{B}^{-1}(x_1\boldsymbol{\Phi}_{,1} + x_2\boldsymbol{\Phi}_{,2})] \tag{2.42}$$

Substituting $\mathrm{Re}[x_1\boldsymbol{\Phi}_{,1} + x_2\boldsymbol{\Phi}_{,2}] = \mathrm{Re}[z\boldsymbol{\Phi}']$ and using the identity $\mathrm{Re}(\boldsymbol{C})\,\mathrm{Re}(\boldsymbol{D}) = \frac{1}{2}\mathrm{Re}[\boldsymbol{C}(\boldsymbol{D} + \overline{\boldsymbol{D}})]$ gives

$$-\mathrm{Re}(d\boldsymbol{\Phi}^a)^T\ \mathrm{Re}[\boldsymbol{A}\boldsymbol{B}^{-1}(x_1\boldsymbol{\Phi}_{,1} + x_2\boldsymbol{\Phi}_{,2})] = -\frac{1}{2}\mathrm{Re}\{(d\boldsymbol{\Phi}^a)^T[\boldsymbol{A}\boldsymbol{B}^{-1}(z\boldsymbol{\Phi}') + c.c.]\} \tag{2.43}$$

Now, M_M becomes

$$M_M = \frac{1}{2}\mathrm{Re}\left\{\int_\Gamma [(\boldsymbol{A}\boldsymbol{B}^{-1}d\boldsymbol{\Phi}^a)^T(z\boldsymbol{\Phi}' + c.c.)] + (d\boldsymbol{\Phi}^a)^T[\boldsymbol{A}\boldsymbol{B}^{-1}(z\boldsymbol{\Phi}') + c.c.]\right\} \tag{2.44}$$

Substituting $\boldsymbol{\Phi}$ and $\boldsymbol{\Phi}^a$ from Eqs. (2.5) and (2.15) into Eq. (2.44) gives

$$\begin{aligned} M_M = \frac{1}{2}\mathrm{Re}[\sum_n \int_\Gamma (&\boldsymbol{h}_m^T \boldsymbol{B}^{-T}\left\langle df^a\right\rangle \boldsymbol{A}^T \boldsymbol{B}\left\langle z f_n'\right\rangle \boldsymbol{B}^{-1}\boldsymbol{g}_n + \boldsymbol{h}_m^T \boldsymbol{B}^{-T}\left\langle df^a\right\rangle \boldsymbol{A}^T \overline{\boldsymbol{B}}\left\langle \bar{z}\bar{f}_n'\right\rangle \overline{\boldsymbol{B}}^{-1}\overline{\boldsymbol{g}}_n \\ &+ \boldsymbol{h}_m^T \boldsymbol{B}^{-T}\left\langle df^a\right\rangle \boldsymbol{B}^T \boldsymbol{A}\left\langle z f_n'\right\rangle \boldsymbol{B}^{-1}\boldsymbol{g}_n + \boldsymbol{h}_m^T \boldsymbol{B}^{-T}\left\langle df^a\right\rangle \boldsymbol{B}^T \overline{\boldsymbol{A}}\left\langle \bar{z}\bar{f}_n'\right\rangle \overline{\boldsymbol{B}}^{-1}\overline{\boldsymbol{g}}_n)] \\ = \frac{1}{2}\mathrm{Re}[\sum_n \int_\Gamma (&\boldsymbol{h}_m^T \boldsymbol{B}^{-T}\left\langle df^a\right\rangle (\boldsymbol{A}^T\boldsymbol{B} + \boldsymbol{B}^T\boldsymbol{A})\left\langle z f_n'\right\rangle \boldsymbol{B}^{-1}\boldsymbol{g}_n \\ &+ \boldsymbol{h}_m^T \boldsymbol{B}^{-T}\left\langle df^a\right\rangle (\boldsymbol{A}^T\overline{\boldsymbol{B}} + \boldsymbol{B}^T\overline{\boldsymbol{A}})\left\langle \bar{z}\bar{f}_n'\right\rangle \overline{\boldsymbol{B}}^{-1}\overline{\boldsymbol{g}}_n)] \end{aligned} \tag{2.45}$$

Eq. (2.45) is simplified using the orthogonality relations in Eq. (2.14) to give

$$M_M = \frac{1}{2}\mathrm{Re}\left[\sum_n \int_\Gamma \boldsymbol{h}_m^T \boldsymbol{B}^{-T} \left\langle df^a \right\rangle \left\langle z f_n' \right\rangle \boldsymbol{B}^{-1} \boldsymbol{g}_n \right]$$
$$= \frac{1}{2}\mathrm{Re}\left[\sum_n \boldsymbol{h}_m^T \boldsymbol{B}^{-T} \left\langle \int_\Gamma z f_n'(z) df^a(z) \right\rangle \boldsymbol{B}^{-1} \boldsymbol{g}_n \right] \tag{2.46}$$

By defining the integral term,

$$R_{mn} \equiv \int_\Gamma z f_n'(z_\alpha) df^a(z_\alpha) = (\delta_n + 1)(\Delta_m + 1)\int_\Gamma z_\alpha^{\delta_n + \Delta_m + 1} dz_\alpha \tag{2.47}$$

Eq. (2.46) can be reduced to

$$M_M = \frac{1}{2}\mathrm{Re}\left[\sum_n \boldsymbol{h}_m^T \boldsymbol{B}^{-T} \left\langle R_{mn} \right\rangle \boldsymbol{B}^{-1} \boldsymbol{g}_n \right] \tag{2.48}$$

By examining the exponents proportional to the path radius, it can be seen that for path independence the actual and auxiliary stress singularities must follow the relation $\delta_n + \Delta_m + 2 = 0$. Using this relation and representing the points on the notch surface defining the limits of the arbitrary path, Γ, by z_1 and z_2 results in

$$R_{mn} = (\delta_n + 1)(\Delta_m + 1)\int_\Gamma z_\alpha^{\delta_n + \Delta_m + 1} dz_\alpha = -(\delta_n + 1)^2 \int_{z_2}^{z_1} \frac{dz_\alpha}{z_\alpha} \equiv R_n \tag{2.49}$$

The eigenvalues of the characteristics equation, μ_α, can be separated into real and imaginary parts giving $\mu_\alpha = \mu_\alpha' + i\mu_\alpha''$ (Ting, 1996). Substituting into the equation for z_α gives

$$z_\alpha = x_1 + \mu_\alpha x_2 = (x_1 + \mu_\alpha' x_2) + i\mu_\alpha'' x_2$$
$$= r[(\cos\theta + \mu_\alpha' \sin\theta) + i\mu_\alpha'' \sin\theta] = r R_\alpha e^{i\psi_\alpha} \tag{2.50}$$

where ψ and R are determined from

$$R_\alpha \cos\psi_\alpha = \cos\theta + \mu'_\alpha \sin\theta \tag{2.51}$$

$$R_\alpha \sin\psi_\alpha = \mu''_\alpha \sin\theta \tag{2.52}$$

where θ is measured from the x_1 axis and positive in the counter-clockwise direction.

Eq. (2.49) is integrated and the substitution $z_\alpha = rR_\alpha e^{i\psi_\alpha}$ is made to give

$$R_n = -(\delta_n + 1)^2 \int_{z_2}^{z_1} \frac{dz_\alpha}{z_\alpha} = -(\delta_n + 1)^2 \left[\ln r + \ln R_\alpha + i\psi_\alpha\right]_{\phi_2}^{\phi_1} \tag{2.53}$$

The generalized stress intensity factors, $k^{(s)}$ and $k^{(a)}$, are derived from the first-term interaction *M*-integral solution. The first term becomes the dominate term as the notch tip is approached ($r \to 0$). The first-term interaction *M*-integral solution is

$$M_M = \frac{1}{2}\mathrm{Re}\left[\boldsymbol{h}_1^T \boldsymbol{B}^{-T} \langle R_1 \rangle \boldsymbol{B}^{-1} \boldsymbol{g}_1\right] \tag{2.54}$$

where

$$\langle R_1 \rangle = -(\delta_1 + 1)^2 \begin{bmatrix} \left[\ln((R_1)_1/(R_1)_2) + i\left[(\psi_1)_1 - (\psi_1)_2\right]\right] & 0 \\ 0 & \left[\ln((R_2)_1/(R_2)_2) + i\left[(\psi_2)_1 - (\psi_2)_2\right]\right] \end{bmatrix} \tag{2.55}$$

where

$$(\psi_\alpha)_j = \tan^{-1}\left(\frac{\mu''_\alpha \sin\phi_j}{\cos\phi_j + \mu'_\alpha \sin\phi_j}\right) \qquad (R_\alpha)_j = \frac{\mu''_\alpha \sin\phi_j}{\sin(\psi_\alpha)_j} \tag{2.56}$$

A phase shift is necessary for the calculation of $(\psi_\alpha)_j$ so that

$$(\psi_\alpha)_1 = \tan^{-1}\left(\frac{\mu''_\alpha \sin\phi_1}{\cos\phi_1 + \mu'_\alpha \sin\phi_1}\right) + \pi \qquad (\psi_\alpha)_2 = \tan^{-1}\left(\frac{\mu''_\alpha \sin\phi_2}{\cos\phi_2 + \mu'_\alpha \sin\phi_2}\right) - \pi \tag{2.57}$$

Now, the generalized stress intensity factors are defined using a relation for the first-term complex eigenvector, $\boldsymbol{g}_1$. The generalized stress intensity factors for symmetric and anti-symmetric deformation, respectively, are defined as

$$k^{(s)} = \lim_{r\to 0}[(2\pi r)^{-\delta^{(s)}}\ \sigma_{22}(r,\theta=0)] \tag{2.58}$$

$$k^{(a)} = \lim_{r\to 0}[(2\pi r)^{-\delta^{(a)}}\ \sigma_{12}(r,\theta=0)] \tag{2.59}$$

where $\delta^{(s)}$ and $\delta^{(a)}$ denote the first-term stress exponents for symmetric and anti-symmetric deformation, respectively. Using Eqs. (2.5) and (2.2) leads to $\boldsymbol{g}_1$ being expressed as

$$\boldsymbol{g}^{(i)} = \frac{(2\pi)^{\delta^{(i)}}}{(\delta^{(i)}+1)} k^{(i)} \tilde{\boldsymbol{g}}^{(i)} \qquad i = a,\ s \tag{2.60}$$

where *s* represents the symmetric deformation mode, *a* represents the anti-symmetric deformation mode, and $\tilde{\boldsymbol{g}}^{(i)}$ represents the normalized form of $\boldsymbol{g}^{(i)}$. The normalized first complex eigenvector for symmetric deformation, $\tilde{\boldsymbol{g}}^{(s)} = \left[\tilde{g}_1^{(s)},\ \tilde{g}_2^{(s)}\right]^{\mathrm{T}}$, and the normalized first complex eigenvector for anti-symmetric deformation, $\tilde{\boldsymbol{g}}^{(a)} = \left[\tilde{g}_1^{(a)},\ \tilde{g}_2^{(a)}\right]^{\mathrm{T}}$, are normalized so that $\tilde{g}_2^{(s)} = 1$ and $\tilde{g}_1^{(a)} = 1$. In the case of a crack, the definitions for the generalized stress intensity factors in Eqs. (2.58) and (2.59) are equivalent to the conventional definitions for the mode-I and mode-II stress intensity factors, which are, respectively,

$$K_I = \lim_{r\to 0}[\sqrt{2\pi r}\ \sigma_{22}(r,\theta=0)] \tag{2.61}$$

$$K_{II} = \lim_{r\to 0}[\sqrt{2\pi r}\ \sigma_{12}(r,\theta=0)] \tag{2.62}$$

Substituting Eq. (2.60) into Eq. (2.54) $k^{(s)}$ and $k^{(a)}$ are expressed by the M_M solution,

$$k^{(i)} = \frac{2(\delta^{(i)}+1)M_M}{(2\pi)^{\delta^{(i)}}\ \mathrm{Re}[\boldsymbol{h}^{(i)T}\boldsymbol{B}^{-T}\left\langle R^{(i)}\right\rangle \boldsymbol{B}^{-1}\tilde{\boldsymbol{g}}^{(i)}]} \qquad i = a,\ s \tag{2.63}$$

where $\left\langle R^{(i)}\right\rangle = \left\langle R_1\right\rangle$ with $\delta_1 = \delta^{(i)}$. Therefore, values for $k^{(s)}$ and $k^{(a)}$ can now be found after M_M is evaluated numerically from finite element solution and the unknown auxiliary

field data. A form for M_M that can be evaluated numerically is derived in the Appendices.

The solutions for $k^{(i)}$ can be simplified for the case of a crack. In this case,

$$\delta^{(i)} = -\frac{1}{2}, \quad \left\langle R^{(i)} \right\rangle = -\frac{i\pi}{2}\mathbf{I}, \quad \boldsymbol{h}^{(i)T}\tilde{\boldsymbol{g}}^{(i)} = 1 \qquad i = a, s \tag{2.64}$$

where $\mathbf{I}$ is the 2×2 identity matrix. Using Eq. (2.64),

$$\boldsymbol{B}^{-T}\boldsymbol{B}^{-1} = -2i\boldsymbol{L}^{-1} \tag{2.65}$$

where for plane strain

$$\boldsymbol{L}^{-1} = s'_{11}\begin{bmatrix} b & d \\ d & e \end{bmatrix} \tag{2.66}$$

and

$$\mu_1 + \mu_2 = a + ib, \quad \mu_1\mu_2 = c + id, \quad e = ad - bc = \mathrm{Im}[\mu_1\mu_2(\overline{\mu}_1 + \overline{\mu}_2)] \tag{2.67}$$

allows Eq. (2.63) to be simplified in the case of a crack to

$$k^{(i)} = -\sqrt{\frac{2}{\pi}}\frac{M_M}{\mathrm{Re}[\boldsymbol{h}^{(i)T}\boldsymbol{L}^{-1}\tilde{\boldsymbol{g}}^{(i)}]} \qquad i = a, s \tag{2.68}$$

where for a crack $\tilde{\boldsymbol{g}}^{(s)T} = \boldsymbol{h}^{(s)T} = [0 \quad 1]$ and $\tilde{\boldsymbol{g}}^{(a)T} = \boldsymbol{h}^{(a)T} = [1 \quad 0]$

Eq. (2.68) can be further simplified for isotropic materials. For isotropic materials with plane strain,

$$\boldsymbol{L}^{-1} = \frac{2(1-\nu^2)}{E}\mathbf{I} \tag{2.69}$$

Therefore, for the case of a crack in isotropic materials with plane strain,

$$k^{(i)} = -\frac{E}{\sqrt{2\pi}(1-\nu^2)}M_M \qquad i = a, s \tag{2.70}$$

In the case of plane stress, s'_{ij} is replaced by s_{ij}. Thus, Eq. (2.70) reduces to

$$k^{(i)} = -\frac{E}{\sqrt{2\pi}} M_M \qquad i = a, s \tag{2.71}$$

3 Numerical Results

The two loading cases that produce symmetric and anti-symmetric deformation, respectively, are studied. To produce symmetric deformation, a uniform tensile stress is applied at the opposite ends of the specimen as shown in Figure 3.1. To produce anti-symmetric deformation, a force-couple loading is applied according to the ASTM standard D 5379/D 5379M –93 as shown in Figure 3.2. This force-couple loading produces a pure shear force loading (no bending moment) on the line between the two notch tips. This specimen geometry shown in Figure 3.2 has been used for shear property testing of fiber-reinforced composite materials according to the ASTM standard D 5379/D 5379M –93. The specimen dimensions show that the plate has a height to width ratio, *h/w*, of four. According to ASTM D5379/D 5379M –93, the specimen used for in-plane testing has a very small thickness to width ratio, so the assumption of plane stress is made. Results are given for a widely varying notch geometry with the notch depth ratio, *a/w*, varying from 0.1 to 0.4 and the notch angle, α, varying from 0^o (a crack) to 160^o as was permissible for the given notch depth and loading. Now, the generalized stress intensity factors, $k^{(s)}$ and $k^{(a)}$, are to be made non-dimensional. The non-dimensional generalized stress intensity factor for symmetric deformation is

$$\tilde{k}^{(s)} = \frac{k^{(s)}}{\sigma_o (\pi a)^{-\delta^{(s)}}} \tag{3.1}$$

where a is the notch depth and σ_o is the uniform tensile stress applied to produce symmetric deformation. For the crack case, $\tilde{k}^{(s)}$ is equivalent to the conventional normalized stress intensity factor. Now, the non-dimensional generalized stress intensity factor for anti-symmetric deformation is

$$\tilde{k}^{(a)} = \frac{k^{(a)}}{\frac{P}{w}(\pi a)^{-\delta^{(a)}}} \tag{3.2}$$

where *P* is a force, *a* is the notch depth, and *w* is the specimen width.

The results are given for three material property cases. These material cases are an isotropic material and 0^o and 90^o in-plane fiber orientations for a unidirectional fiber-reinforced graphite/epoxy composite. The isotropic material properties chosen are for aluminum with $E = 70\,\text{GPa}$ and $\nu = 0.33$. The material properties for the graphite/epoxy composite are shown below

$$E_1 = 138.0\,GPa,\ E_2 = E_3 = 8.9\,GPa$$
$$G_{12} = G_{13} = 5.17\,GPa,\ G_{23} = 2.89\,GPa$$
$$\nu_{12} = \nu_{13} = 0.30,\ \nu_{23} = 0.54$$

The 0^o and 90^o fiber directions are shown in Figure 3.1. The 0^o fiber direction corresponds to the fibers aligned in the longest direction of the plate. For the 0^o case, the material properties are obtained simply by coordinate transformations.

Computer programs and finite element solutions were used to calculate $k^{(s)}$ and $k^{(a)}$. Examples of these FORTRAN and ANSYS programs are shown and described in the Appendices. The numeric value of the interaction *M*-integral, M_M, used to calculate the $k^{(i)}$ results was generally taken as the average from three independent paths. While M_M is theoretically path independent, there are some small variations between paths when M_M is evaluated numerically. Excellent path independence was accieved for symmetric deformation. For all three material property cases, the variation between paths was 0.1% or less. The variation was up to 2% in only a couple of cases. But, the variation between paths was greater for anti-symmetric deformation with the amount of variation changing with the material case. The most approximate numeric path independence was accieved with the 0^o fiber orientation material properties. The variation between paths was generally 1% or less with 4% variation for only a few cases.

The most variation between paths was encountered with the 90^o fiber orientation material properties. The variation was generally 1 to 2% with around 5% variation for some cases. While, for the results with isotropic material properties, the variation was usually around 1% with up to 5% variation for a few cases.

The First Stress Exponent

The first stress exponent, δ_1 controls the singularity of the stress components. For the case here of a V-notch in a homogeneous material, the values of δ_1 are real. For $\delta_1 < 0$, the stress components become singular as the notch tip is approached ($r \to 0$), but for $\delta_1 > 0$, the stress components decrease as $r \to 0$. The magnitude of δ_1 controls the rate of increase or decrease of the stress components as $r \to 0$. The first stress exponent varies with the deformation mode (symmetric or anti-symmetric), the material properties, and the notch angle, α. The values of $\delta^{(s)}$ and $\delta^{(a)}$ for the isotropic and anisotropic material property cases are shown in Table 3.1 and plotted in Figure 3.3. The values of the first stress exponents shown in Table 3.1 for the composite material cases are based on the assumption of plane stress. The assumption of plane stress or plane strain does not affect the first stress exponents for an isotropic material. The values of $\delta^{(s)}$ and $\delta^{(a)}$ based on the assumption of plane strain shown in Table 3.2. The first stress exponents for the 0^o fiber orientation are slightly more positive with the plane strain assumption. While, for the 90^o fiber orientation, the first stress exponents are slightly more negative. Figure 3.3 shows that for the crack case ($\alpha = 0^o$) $\delta^{(s)}$ and $\delta^{(a)}$ are -0.5 independent of the loading conditions and material properties. But, as the notch opens (α increases) the first stress exponents, $\delta^{(s)}$ and $\delta^{(a)}$, vary with the deformation mode and material properties.

The first stress exponents under the symmetric deformation mode are negative for all notch angles for each material case and become less negative as the notch angle opens (α increases). The $\delta^{(s)}$ values in terms of magnitude are greater for the 0^o fiber orientation than for the 90^o fiber orientation. The values of $\delta^{(s)}$ for an isotropic material are in between those for the 0^o and 90^o cases for a given notch angle.

Under the anti-symmetric deformation mode, with $\alpha = 0^o$, $\delta^{(a)} = -0.5$. But $\delta^{(a)}$ alters it's sign as α increases for all three material cases. The $\delta^{(a)}$ values become positive for a smaller α for the 90^o fiber case than for the 0^o fiber case. For the 90^o fiber case, $\delta^{(a)}$ becomes positive when α is about 67^o. Whereas, for the 0^o fiber case, $\delta^{(a)}$ becomes positive when α is about 137^o. For an isotropic material, $\delta^{(a)}$ becomes positive when α is about 103^o, which is between the 90^o and 0^o fiber cases. When $\delta^{(a)}$ becomes positive, the first-term stress components decrease to zero as the notch tip is approached ($r \to 0$).

The Eigenvectors

The normalized first complex eigenvectors, $\tilde{\boldsymbol{g}}^{(i)}$, are eigenvectors corresponding to the first stress exponents, $\delta^{(i)}$. The normalized first complex eigenvector for symmetric deformation, $\tilde{\boldsymbol{g}}^{(s)} = \left[\tilde{g}_1^{(s)},\ \tilde{g}_2^{(s)}\right]^{\mathrm{T}}$, is normalized such that $\tilde{g}_2^{(s)} = 1$. Whereas, the normalized first complex eigenvector for anti-symmetric deformation, $\tilde{\boldsymbol{g}}^{(a)} = \left[\tilde{g}_1^{(a)},\ \tilde{g}_2^{(a)}\right]^{\mathrm{T}}$, is normalized so that $\tilde{g}_1^{(a)} = 1$. The values of $\tilde{\boldsymbol{g}}^{(s)}$ and $\tilde{\boldsymbol{g}}^{(a)}$ for an isotropic material with $\alpha = 0^o$ to 180^o are shown in Table 3.3. The values of $\tilde{\boldsymbol{g}}^{(s)}$ and $\tilde{\boldsymbol{g}}^{(a)}$ for the composite material with 0^o and 90^o fiber orientations are shown in Table 3.4.

The auxiliary first complex eigenvectors, $\boldsymbol{h}^{(i)}$, are eigenvectors corresponding to the auxiliary first stress exponents, $\Delta^{(i)} = -\delta^{(i)} - 2$. The auxiliary first complex

eigenvector for symmetric deformation, $\boldsymbol{h}^{(s)} = \left[h_1^{(s)},\ h_2^{(s)}\right]^{\mathrm{T}}$, is normalized so that $h_2^{(s)} = 1$. Whereas, the auxiliary first complex eigenvector for anti-symmetric deformation, $\boldsymbol{h}^{(a)} = \left[h_1^{(a)},\ h_2^{(a)}\right]^{\mathrm{T}}$, is normalized so that $h_1^{(a)} = 1$. The values of $\boldsymbol{h}^{(s)}$ and $\boldsymbol{h}^{(a)}$ for an isotropic material with $\alpha = 0^{\mathrm{o}}$ to 180^{o} are shown in Table 3.5. The values of $\boldsymbol{h}^{(s)}$ and $\boldsymbol{h}^{(a)}$ for the composite material with 0^{o} and 90^{o} fiber orientations are shown in Table 3.6.

3.1 Numerical Verification

The numeric values of $\tilde{k}^{(s)}$ and $\tilde{k}^{(a)}$ were calculated for an isotropic plate to allow for verification with existing results. The numeric values of $\tilde{k}^{(s)}$ were verified for the two important notch angle, α, cases of 0^{o} and 90^{o} for an isotropic rectangular plate. For the crack case where $\alpha = 0^{\mathrm{o}}$, $\tilde{k}^{(s)}$ is equivalent to the conventional normalized mode-I stress intensity factor, $\tilde{K}_I$. So, for $\alpha = 0^{\mathrm{o}}$, the values for $\tilde{k}^{(s)}$ were verified using the stress intensity factor solution for the double edge cracked rectangular plate under uniform tension in a manual edited by Murakami (Murakami, 1987). The normalized mode-I stress intensity factor, $\tilde{K}_I$, is given by

$$\tilde{K}_I(\beta) = 1.122 - 0.154\beta + 0.807\beta^2 - 1.894\beta^3 + 2.494\beta^4 \qquad (3.3)$$

where $\beta = \dfrac{2a}{w}$. The accuracy of this equation is stated as $\pm 0.5\%$ for $\beta \le 0.8$. Table 3.7 shows that for a double edge cracked isotropic plate, the difference between the values for $\tilde{k}^{(s)}$ and $\tilde{K}_I$ is less than 0.5%.

For $\alpha = 90^{\mathrm{o}}$, the $\tilde{k}^{(s)}$ values were compared to the generalized stress intensity factor results by Wu and Chen for an isotropic plate with double edge notches (Wu and

Chen, 1996). The same height to width ratio, *h/w*, of 1.4 and plane strain conditions used by Wu and Chen were used for the calculation of the $\tilde{k}^{(s)}$ values for these cases. The definition of Wu and Chen's generalized stress intensity factor, $\tilde{K}_I$, differs from the definition of $\tilde{k}^{(s)}$. Thus, a relation allowing comparison between the values for $\tilde{K}_I$ and $\tilde{k}^{(s)}$ had to be derived. The equivalent $\tilde{k}^{(s)}$ value calculated from Wu and Chen's $\tilde{K}_I$ value is

$$\tilde{k}^{(s)} = \frac{\tilde{K}_I}{\sqrt{2\pi}} \left(\frac{a}{w} \right)^{\delta^{(s)}} \tag{3.4}$$

The difference between the $\tilde{k}^{(s)}$ values calculated from Wu and Chen's $\tilde{K}_I$ values and from the present method are shown in Table 3.8. Table 3.8 shows that the difference between the $\tilde{k}^{(s)}$ values calculated from Wu and Chen's results and the present method is only about 2%. Therefore, the generalized stress intensity factor results from the present method agree well with the previous results by Wu and Chen.

The numeric value of $\tilde{k}^{(a)}$ for an isotropic plate was verified for $\alpha = 0^{\circ}$ with $a/w = 0.4165$. For $\alpha = 0^{\circ}$, $\tilde{k}^{(a)}$ is equivalent to the conventional normalized mode-II stress intensity factor, $\tilde{K}_{II}$. In this case, the value of $\tilde{k}^{(a)}$ was verified using a stress intensity factor solution for the double edge cracked rectangular plate subjected to four-point-shear loading from a manual edited by Murakami (Murakami, 1987). From the table in Murakami's book, $\tilde{K}_{II}$ is about 1.61 for $a/w = 0.4165$ with a stated accuracy of 1% - 2%. The calculated value for $\tilde{k}^{(a)}$ is 1.52, which is 5.4% less than the value in Murakami's book, which was calculated using a finite element method. Considering the stated accuracy of 1% - 2%, the value of $\tilde{k}^{(a)}$ is in good agreement with results in Murakami's book.

3.2 Isotropic Plate

The numeric values of $\tilde{k}^{(s)}$ and $\tilde{k}^{(a)}$ were calculated for an isotropic plate with a varying notch geometry.

Symmetric Deformation

The $\tilde{k}^{(s)}$ values for $a/w = 0.1$, 0.2, 0.3, and 0.4 with $\alpha = 0^{o}$ to 160^{o} for the plate in Figure 3.1 are shown in Table 3.9, and these values are plotted in Figure 3.4. The numerical results show that $\tilde{k}^{(s)}$ increases with *a/w*. Therefore, the increase in $\tilde{k}^{(s)}$ with *a/w* follows the expected increase in the strength of the stress field at the notch tip as *a/w* increases for a given notch angle. Also, as *a/w* increases, $\tilde{k}^{(s)}$ increases faster with the increase in α. Therefore, as *a/w* increases, the strength of the stress field decreases faster as α increases.

The values for $k^{(s)}$ and $\delta^{(s)}$ were used to calculate the first-term solution for σ_{22}/σ_o on the line between the notch tips, $\theta = 0^{o}$. The first-term and finite element solutions for σ_{22}/σ_o were studied to illustrate their trend as the notch tip is approached ($r \to 0$). The first-term solution for σ_{22}/σ_o is directly proportional to $k^{(s)}$. The first term is the dominant term as the notch tip is approached. More terms of the eigenfunction expansion series are needed to give an accurate solution for the stress field away from the notch tip. Therefore, the finite element solution is the accurate solution away from the notch tip. But, the finite element solution using non-singular elements looses accuracy as the notch tip is approached. The first-term solution is the accurate solution as the notch tip is approached. The first-term and finite element solutions for σ_{22}/σ_o are shown in Figure 3.5 for a notch with $\alpha = 110^{o}$ and *a/w* = 0.2. In Figure 3.6, the convergence of the first-term and finite element solutions is shown for a notch with

$\alpha = 30^o$ and *a/w* = 0.2. Figures 3.5 and 3.6 illustrate the change of the solutions with the change in the notch angle and thus $\delta^{(s)}$. The first-term and finite element solutions converge closer to the notch tip as the notch angle decreases and magnitude of $\delta^{(s)}$ increases (the stress singularity becomes stronger). Figures 3.5 and 3.6 also show that σ_{22}/σ_o increases faster as the notch tip is approached as the notch angle decreases and magnitude of $\delta^{(s)}$ increases.

Anti-symmetric Deformation

The $\tilde{k}^{(a)}$ values for *a/w*=0.1, 0.2, 0.3, and 0.4 with $\alpha = 0^o$ to 110^o for the isotropic plate in Figure 3.2 are shown in Table 3.10, and these values are plotted in Figure 3.7. These results show that $\tilde{k}^{(a)}$ increases with *a/w*. But, the increase in $\tilde{k}^{(a)}$ with *a/w* is not as significant as it was for symmetric deformation. The change in $\tilde{k}^{(a)}$ from *a/w* = 0.1 to 0.2 is about the same as from *a/w* = 0.2 to 0.3, but the change in $\tilde{k}^{(a)}$ from *a/w* = 0.3 to 0.4 is much greater. The increase in $\tilde{k}^{(a)}$ with *a/w* follows the expected increase in the strength of the stress field at the notch tip as *a/w* increases. This means that increasing the notch depth will reduce the failure load, as expected. Also, as *a/w* increases, $\tilde{k}^{(a)}$ increases faster with increasing α. Therefore, as *a/w* increases, the strength of the stress field decreases faster as α increases. Therefore, as *a/w* increases, the failure load increases more as α increases.

The values for $k^{(a)}$ and $\delta^{(a)}$ were used to calculate the first-term solution for σ_{12}/σ_o on the line between the notch tips, $\theta = 0^o$. The first-term and finite element solutions for σ_{12}/σ_o were studied to compare their distributions as the notch tip is approached ($r \to 0$). Figure 3.8 shows the first-term and finite element distributions of σ_{12}/σ_o for a notch with $\alpha = 110^o$ and *a/w* = 0.2. For this case with $\alpha = 110^o$,

$\delta^{(a)} = 0.06022$. Since $\delta^{(a)}$ is positive, the first-term solution for $\sigma_{12}/\sigma_o \to 0$ as $r \to 0$. But, since the finite element solution includes more terms, the finite element solution for σ_{12}/σ_o does not decrease to zero. But, for the case shown in Figure 3.9 with $\alpha = 110^o$ and *a/w* = 0.2, $\delta^{(a)} = -0.40181$ and the first-term and finite element solutions converge as $r \to 0$.

3.3 Anisotropic Plate

Taking advantage of the generality of the present method, numerical results were produced for a unidirectional fiber-reinforced graphite/epoxy composite rectangular plate. Results are shown for in-plane fiber orientations of 0^o and 90^o.

3.3.1 The 0^o Fiber Orientation

The 0^o fiber direction is shown on the rectangular plate in Figure 3.1. With the fibers perpendicular to the line connecting the notch tips in the 0^o case, the value of $\delta^{(i)}$ is less for a given notch angle with the 0^o orientation than with the 90^o orientation. Therefore, with the other variable in the first-term stress solution, $k^{(i)}$, included, the first-term stress components are greater at the notch tip with the 0^o orientation than with the 90^o orientation.

Symmetric Deformation

Now, the $\tilde{k}^{(s)}$ results for a varying notch geometry for the rectangular plate with the 0^o fiber orientation under the loading shown in Figure 3.1 are presented. The $\tilde{k}^{(s)}$ values for *a/w* = 0.1, 0.2, 0.3, and 0.4 with $\alpha = 0^o$ to 160^o are shown in Table 3.11, and

these values are plotted in Figure 3.10. The numerical results show that $\tilde{k}^{(s)}$ increases with *a/w* as it did for the isotropic material case. The increase in $\tilde{k}^{(s)}$ with *a/w* follows the expected increase in the strength of the stress field at the notch tip as *a/w* increases. Also, as *a/w* increases, $\tilde{k}^{(s)}$ increases faster with the increase in α. But, $\tilde{k}^{(s)}$ does not increase as much with increasing α as it did for the isotropic case. Therefore, as *a/w* increases, the strength of the stress field decreases faster as α increases, but the strength of the stress field does not decrease as fast with increasing α as it does for an isotropic material.

The values of $k^{(s)}$ and $\delta^{(s)}$ for the 0^o fiber orientation were used to calculate the first-term solution for σ_{22}/σ_o on the line between the notch tips, $\theta = 0^o$. Figure 3.11 shows the convergence of the first-term and finite element solutions for σ_{22}/σ_o as a notch in the rectangular plate is approached with $\alpha = 110^o$ and *a/w* = 0.2 with the 0^o fiber orientation. Figure 3.12 shows the convergence of the solutions when the notch angle is changed to $\alpha = 30^o$. Since the strength of the singularity increases little when the notch angle is changed from 110^o to 30^o, σ_{22}/σ_o does not increase much faster as the notch tip is approached.

Anti-symmetric Deformation

The $\tilde{k}^{(a)}$ results for a varying notch geometry for the rectangular plate with the 0^o fiber orientation with the loading shown in Figure 3.2 are presented. The $\tilde{k}^{(a)}$ values for *a/w* = 0.1, 0.2, 0.3, and 0.4 with $\alpha = 0^o$ to 110^o are shown in Table 3.12, and these values are plotted in Figure 3.13. The numeric values of $\tilde{k}^{(a)}$ increase with *a/w*. But, as for the isotropic case, the increase in $\tilde{k}^{(a)}$ with *a/w* is not as significant as it was for symmetric deformation. The change in $\tilde{k}^{(a)}$ with the change in *a/w* from 0.2 to 0.3 is a

little less than the change from *a/w* = 0.1 to 0.2, but the change in $\tilde{k}^{(a)}$ from *a/w* = 0.3 to 0.4 is greater than it was for the other equal increments in *a/w*. The increase in $\tilde{k}^{(a)}$ with *a/w* follows the expected increase in the strength of the stress field at the notch tip as *a/w* increases. Increasing the notch depth will reduce the failure load. Also, as *a/w* increases, $\tilde{k}^{(a)}$ increases faster with increasing α. But, the increase in $\tilde{k}^{(a)}$ with α as *a/w* increases is not as much as in the isotropic material case. Therefore, as *a/w* increases, the strength of the stress field decreases faster as α increases, but it does not decrease as quickly as it did for the isotropic case. Therefore, as *a/w* increases, the failure load increases more as α increases.

The values for $k^{(a)}$ and $\delta^{(a)}$ were used to calculate the first-term solution for σ_{12}/σ_o on the line between the notch tips, $\theta = 0^o$. Figure 3.14 shows the convergence of the first-term and finite element solutions for σ_{12}/σ_o as a notch in the rectangular plate is approached with $\alpha = 110^o$ and *a/w* = 0.2 with the 0^o fiber orientation. Figure 3.15 shows the convergence of the solutions when $\alpha = 30^o$. These figures show that as the notch angle is changed from 110^o to 30^o, σ_{12}/σ_o increases faster as the notch tip is approached ($r \to 0$).

3.3.2 The 90° Fiber Orientation

The 90^o fiber direction is shown on the rectangular plate in Figure 3.1. With the fibers parallel to the line connecting the notch tips in the 90^o case, the value of $\delta^{(i)}$ is greater for a given notch angle with the 90^o orientation than with the 0^o orientation. After the other variable in the first-term stress solution, $k^{(i)}$, is calculated, it can be seen that the first-term stress components are less at the notch tip with the 90^o orientation than with the 0^o orientation.

Symmetric Deformation

Now, the $\tilde{k}^{(s)}$ results for a varying notch geometry for the rectangular plate with the 90^o fiber orientation under the loading shown in Figure 3.1 are presented. The $\tilde{k}^{(s)}$ values for $a/w = 0.1, 0.2, 0.3,$ and 0.4 with $\alpha = 0^o$ to 160^o are shown in Table 3.13, and these values are plotted in Figure 3.16. The numerical results show that $\tilde{k}^{(s)}$ increases with *a/w* as it did for the isotropic and 0^o fiber orientation cases. The increase in $\tilde{k}^{(s)}$ with *a/w* follows the expected increase in the strength of the stress field at the notch tip as *a/w* increases. Also, as *a/w* increases, $\tilde{k}^{(s)}$ increases faster with the increase in α. With the 90^o fiber orientation, the increase in $\tilde{k}^{(s)}$ with increasing α is greater than with the isotropic and 0^o fiber orientation cases. Therefore, as *a/w* increases, the strength of the stress field decreases faster as α increases, and the decrease in the strength of the stress field with increasing α is greater than for the isotropic and 0^o fiber orientation cases.

The values of $k^{(s)}$ and $\delta^{(s)}$ for the 90^o fiber orientation were used to calculate the first-term solution for σ_{22}/σ_o on the line between the notch tips, $\theta = 0^o$. Figure 3.17 shows the convergence of the first-term and finite element solutions for σ_{22}/σ_o as a notch in the rectangular plate is approached with $\alpha = 110^o$ and *a/w* = 0.2 with the 90^o fiber orientation. Figure 3.18 shows the convergence of the solutions when the notch angle is changed to $\alpha = 30^o$. As for the other material cases with symmetric deformation when the notch angle is reduced, the strength of the singularity increases, the first-term and finite element solutions converge closer to the notch tip, and σ_{22}/σ_o increases faster as the notch tip is approached ($r \to 0$).

Anti-symmetric Deformation

The $\tilde{k}^{(a)}$ results for a varying notch geometry for the plate with the 90^o fiber orientation are presented. The $\tilde{k}^{(a)}$ values for *a/w* = 0.1, 0.2, 0.3, and 0.4 with $\alpha = 0^o$ to 110^o are shown in Table 3.14, and these values are plotted in Figure 3.19. The numerical values of $\tilde{k}^{(a)}$ increase with *a/w*. But, as for the isotropic and 0^o fiber orientation cases, the increase in $\tilde{k}^{(a)}$ with *a/w* is not as significant as it was for symmetric deformation. The change in $\tilde{k}^{(a)}$ with the change in *a/w* from 0.2 to 0.3 is less than the change from *a/w* = 0.1 to 0.2 for notch angles close to 0. But, as the notch angle increases, the change in $\tilde{k}^{(a)}$ with the change in *a/w* from 0.2 to 0.3 is a little greater than the change from *a/w* = 0.1 to 0.2. For notch angles close to 0, the change in $\tilde{k}^{(a)}$ with the change in *a/w* from 0.3 to 0.4 is about the same as the change from *a/w* = 0.1 to 0.2. But, as the notch angle increases, the change in $\tilde{k}^{(a)}$ with the change in *a/w* = 0.3 to 0.4 becomes much greater than it was for the other equal increments in *a/w*. The increase in $\tilde{k}^{(a)}$ with *a/w* follows the expected increase in the strength of the stress field at the notch tip as *a/w* increases. Increasing the notch depth will reduce the failure load. Also, as *a/w* increases, $\tilde{k}^{(a)}$ increases faster with increasing α. The increase in $\tilde{k}^{(a)}$ as α increases is more than with the 0^o fiber orientation. Therefore, as *a/w* increases, the strength of the stress field decreases faster as α increases, and the strength of the stress field decreases faster than with the 0^o fiber orientation. So, as *a/w* increases, the failure load increases more as α increases, and the failure load increases faster as α increases with the 90^o fiber orientation than with the 0^o fiber orientation.

The values for $k^{(a)}$ and $\delta^{(a)}$ were used to calculate the first-term solution for σ_{12}/σ_o on the line between the notch tips, $\theta = 0^o$. Figure 3.20 shows the convergence of the first-term and finite element solutions for σ_{12}/σ_o as a notch in the plate is approached with $\alpha = 110^o$ and *a/w* = 0.2 with the 90^o fiber orientation. Since $\delta^{(a)}$ is

positive for this case, the first-term solution for $\sigma_{12}/\sigma_o \to 0$ as $r \to 0$. When α is reduced to 30^o, $\delta^{(a)}$ becomes negative, and Figure 3.21 shows that the first-term and finite element solutions converge as σ_{12}/σ_o increases as $r \to 0$.

Table 3.1 Symmetric and anti-symmetric first stress exponents, $\delta^{(s)}$ and $\delta^{(a)}$, respectively, with various notch angles in a composite with the 0^o and 90^o fiber orientations (plane stress) and an isotropic material.

α^o	0^o fiber orientation		90^o fiber orientation		Isotropic	
	$\delta^{(s)}$	$\delta^{(a)}$	$\delta^{(s)}$	$\delta^{(a)}$	$\delta^{(s)}$	$\delta^{(a)}$
0	-0.50000	-0.50000	-0.50000	-0.50000	-0.50000	-0.50000
10	-0.49999	-0.47860	-0.49944	-0.40962	-0.49995	-0.47064
30	-0.49973	-0.43071	-0.48996	-0.23149	-0.49855	-0.40181
50	-0.49869	-0.37605	-0.46945	-0.09307	-0.49307	-0.31770
70	-0.49620	-0.31418	-0.44010	0.02480	-0.48015	-0.21556
90	-0.49110	-0.24338	-0.40146	0.14036	-0.45552	-0.09147
110	-0.48102	-0.15914	-0.35109	0.26676	-0.41372	0.06022
130	-0.45995	-0.04971	-0.28474	0.41639	-0.34773	0.24804
145	-0.42642	0.06878	-0.22077	0.55242	-0.27678	0.42060
160	-0.34978	0.26985	-0.14063	0.71724	-0.18131	0.63052
170	-0.23481	0.52301	-0.07590	0.84797	-0.09996	0.79893
180	0.00000	1.00000	0.00000	1.00000	0.00000	1.00000

* $\delta^{(s)} = 0$ is always a root.

Table 3.2 Symmetric and anti-symmetric first stress exponents, $\delta^{(s)}$ and $\delta^{(a)}$, respectively, with various notch angles in a composite with the 0^o and 90^o fiber orientations (plane strain).

α^o	0^o fiber orientation		90^o fiber orientation	
	$\delta^{(s)}$	$\delta^{(a)}$	$\delta^{(s)}$	$\delta^{(a)}$
0	-0.50000	-0.50000	-0.50000	-0.50000
10	-0.49999	-0.47504	-0.49954	-0.41184
30	-0.49963	-0.41911	-0.49171	-0.24055
50	-0.49830	-0.35673	-0.47473	-0.11057
70	-0.49525	-0.28907	-0.45009	-0.00204
90	-0.48934	-0.21527	-0.41677	0.10385
110	-0.47826	-0.13110	-0.37151	0.22156
130	-0.45611	-0.02484	-0.30842	0.36643
145	-0.42185	0.08924	-0.24394	0.50472
160	-0.34501	0.28413	-0.15870	0.68070
170	-0.23098	0.53185	-0.08686	0.82597
180	0.00000	1.00000	0.00000	1.00000

* $\delta^{(s)} = 0$ is always a root.

Table 3.3 Symmetric and anti-symmetric eigenvectors, $\tilde{\boldsymbol{g}}^{(s)}$ and $\tilde{\boldsymbol{g}}^{(a)}$, respectively, for various notch angles in an isotropic material.

	Isotropic	
α^o	$\tilde{\boldsymbol{g}}^{(s)}$	$\tilde{\boldsymbol{g}}^{(a)}$
0	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
10	$\begin{bmatrix} -0.0019063i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.006175i \end{bmatrix}$
30	$\begin{bmatrix} -0.017315i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.067190i \end{bmatrix}$
50	$\begin{bmatrix} -0.048799i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.243156i \end{bmatrix}$
70	$\begin{bmatrix} -0.096911i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.724741i \end{bmatrix}$
90	$\begin{bmatrix} -0.160814i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -2.927082i \end{bmatrix}$
110	$\begin{bmatrix} -0.237036i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 6.941545i \end{bmatrix}$
130	$\begin{bmatrix} -0.319207i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 2.488704i \end{bmatrix}$
145	$\begin{bmatrix} -0.379949i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.924224i \end{bmatrix}$
160	$\begin{bmatrix} -0.436303i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.668287i \end{bmatrix}$
170	$\begin{bmatrix} -0.470074i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.568116i \end{bmatrix}$
180	$\begin{bmatrix} -0.5i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.5i \end{bmatrix}$

Table 3.4 Symmetric and anti-symmetric eigenvectors, $\tilde{\boldsymbol{g}}^{(s)}$ and $\tilde{\boldsymbol{g}}^{(a)}$, respectively, for various notch angles in a composite with the 0^o and 90^o fiber orientations.

	0^o fiber orientation		90^o fiber orientation	
α^o	$\tilde{\boldsymbol{g}}^{(s)}$	$\tilde{\boldsymbol{g}}^{(a)}$	$\tilde{\boldsymbol{g}}^{(s)}$	$\tilde{\boldsymbol{g}}^{(a)}$
0	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
10	$\begin{bmatrix} -0.000357888 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.00447572 \end{bmatrix}$	$\begin{bmatrix} -0.0204453 - 7i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.0199434 \end{bmatrix}$
30	$\begin{bmatrix} -0.00322736 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.0459711 \end{bmatrix}$	$\begin{bmatrix} -0.128809 - 5i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.245625 - 5i \end{bmatrix}$
50	$\begin{bmatrix} -0.00901895 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.150887 \end{bmatrix}$	$\begin{bmatrix} -0.243635 - 5i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -1.325826 - 6i \end{bmatrix}$
70	$\begin{bmatrix} -0.017921 - 5i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.370100 \end{bmatrix}$	$\begin{bmatrix} -0.342517 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 8.226477 \end{bmatrix}$
90	$\begin{bmatrix} -0.030418 - 3i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -0.847912 \end{bmatrix}$	$\begin{bmatrix} -0.426845 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 2.164174 \end{bmatrix}$
110	$\begin{bmatrix} -0.047543 - 3i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -2.179385 \end{bmatrix}$	$\begin{bmatrix} -0.499483 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.615063 \end{bmatrix}$
130	$\begin{bmatrix} -0.071320 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ -11.91333 \end{bmatrix}$	$\begin{bmatrix} -0.562034 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.433419 \end{bmatrix}$
145	$\begin{bmatrix} -0.095620 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 13.726819 \end{bmatrix}$	$\begin{bmatrix} -0.602615 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.369528 \end{bmatrix}$
160	$\begin{bmatrix} -0.127233 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 6.334804 \end{bmatrix}$	$\begin{bmatrix} -0.637686 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.333601 \end{bmatrix}$
170	$\begin{bmatrix} -0.151074 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 5.432322 \end{bmatrix}$	$\begin{bmatrix} -0.657932 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.318710 \end{bmatrix}$
180	$\begin{bmatrix} -0.171580 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 5.153346 \end{bmatrix}$	$\begin{bmatrix} -0.674973 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.308650 \end{bmatrix}$

Table 3.5 Symmetric and anti-symmetric eigenvectors for the auxiliary field, $\boldsymbol{h}^{(s)}$ and $\boldsymbol{h}^{(a)}$, respectively, for various notch angles in an isotropic material.

	Isotropic	
α^o	$\boldsymbol{h}^{(s)}$	$\boldsymbol{h}^{(a)}$
0	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
10	$\begin{bmatrix} 0.0057199\mathbf{3} \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.0019005\mathbf{8} \end{bmatrix}$
30	$\begin{bmatrix} 0.0521496 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.0168927 \end{bmatrix}$
50	$\begin{bmatrix} 0.149145\mathbf{5}i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.0459203 \end{bmatrix}$
70	$\begin{bmatrix} 0.306767i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.0875470 \end{bmatrix}$
90	$\begin{bmatrix} 0.545266i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.140284i \end{bmatrix}$
110	$\begin{bmatrix} 0.908851i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.202885 \end{bmatrix}$
130	$\begin{bmatrix} 1.516750i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.274590i \end{bmatrix}$
145	$\begin{bmatrix} 2.365541i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.334348i \end{bmatrix}$
160	$\begin{bmatrix} 4.376659i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.399874i \end{bmatrix}$
170	$\begin{bmatrix} 8.935575i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.447597i \end{bmatrix}$
180	$\begin{bmatrix} \infty i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.5i \end{bmatrix}$

Table 3.6 Symmetric and anti-symmetric eigenvectors for the auxiliary field, $\boldsymbol{h}^{(s)}$ and $\boldsymbol{h}^{(a)}$, respectively, for various notch angles in a composite with the 0^o and 90^o fiber orientations.

	0^o fiber orientation		90^o fiber orientation	
α^o	$\boldsymbol{h}^{(s)}$	$\boldsymbol{h}^{(a)}$	$\boldsymbol{h}^{(s)}$	$\boldsymbol{h}^{(a)}$
0	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
10	$\begin{bmatrix} 0.0010744\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.0014068\,i \end{bmatrix}$	$\begin{bmatrix} 0.0620948\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.005071\,i \end{bmatrix}$
30	$\begin{bmatrix} 0.0097544\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.0125213\,i \end{bmatrix}$	$\begin{bmatrix} 0.424977\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.028989\,i \end{bmatrix}$
50	$\begin{bmatrix} 0.027664\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.034133\,i \end{bmatrix}$	$\begin{bmatrix} 0.905503\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.051560\,i \end{bmatrix}$
70	$\begin{bmatrix} 0.056373\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.065552\,i \end{bmatrix}$	$\begin{bmatrix} 1.466832\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.070991\,i \end{bmatrix}$
90	$\begin{bmatrix} 0.099569\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.106848\,i \end{bmatrix}$	$\begin{bmatrix} 2.170006\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.089119\,i \end{bmatrix}$
110	$\begin{bmatrix} 0.165869\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.159995\,i \end{bmatrix}$	$\begin{bmatrix} 3.164167\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.107157\,i \end{bmatrix}$
130	$\begin{bmatrix} 0.278082\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.231384\,i \end{bmatrix}$	$\begin{bmatrix} 4.835736\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.125652\,i \end{bmatrix}$
145	$\begin{bmatrix} 0.435572\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.307307\,i \end{bmatrix}$	$\begin{bmatrix} 7.262607\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.139794\,i \end{bmatrix}$
160	$\begin{bmatrix} 0.810688\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.423581\,i \end{bmatrix}$	$\begin{bmatrix} 13.24859\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.153859\,i \end{bmatrix}$
170	$\begin{bmatrix} 1.682296\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.539239\,i \end{bmatrix}$	$\begin{bmatrix} 27.15890\,i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.162945\,i \end{bmatrix}$
180	$\begin{bmatrix} \infty i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.675503\,i \end{bmatrix}$	$\begin{bmatrix} \infty i \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.171446\,i \end{bmatrix}$

Table 3.7 The comparison between $\tilde{k}^{(s)}$ and $\tilde{K}_I$ for a double edge cracked, $\alpha = 0^o$, isotropic plate for $a/w = 0.1$, 0.2, 0.3, and 0.4.

a/w	$\tilde{k}^{(s)}$	$\tilde{K}_I$ (Murakami, 1987)	Difference
0.1	1.107	1.112	0.450%
0.2	1.128	1.132	0.353%
0.3	1.232	1.234	0.162%
0.4	1.567	1.567	0.000%

Table 3.8 A comparison between the $\tilde{k}^{(s)}$ results from (Wu and Chen, 1996) and the present method for an isotropic plate with double edge notches with $\alpha = 90^o$ and $a/w = 0.15$, 0.20, and 0.25.

a/w	$\tilde{k}^{(s)}$ (Wu & Chen)	$\tilde{k}^{(s)}$ (Present)	Difference
0.15	1.28	1.25	2.34%
0.20	1.30	1.28	1.54%
0.25	1.34	1.32	1.49%

Table 3.9 Non-dimensional generalized stress intensity factors and first stress exponents for the symmetric deformation mode, $\tilde{k}^{(s)}$ and $\delta^{(s)}$, respectively, for various notch angles and depths in the isotropic plate.

α^o	$\delta^{(s)}$	$\tilde{k}^{(s)}$			
		$a/w = 0.1$	$a/w = 0.2$	$a/w = 0.3$	$a/w = 0.4$
0	-0.50000	1.10727	1.12814	1.23242	1.56737
10	-0.49995	1.10769	1.12872	1.23237	1.56445
30	-0.49855	1.11117	1.13741	1.24185	1.57402
50	-0.49307	1.13912	1.16317	1.26687	1.60192
70	-0.48015	1.17786	1.20129	1.30725	1.65979
90	-0.45552	1.22779	1.25180	1.36577	1.75302
110	-0.41372	1.28623	1.31255	1.44376	1.91509
130	-0.34773	1.34475	1.38006	1.56181	2.20973
145	-0.27678	1.36477	1.42822	1.70000	2.59334
160	-0.18131	1.33985	1.49366	1.92956	*NA*

Table 3.10 Non-dimensional generalized stress intensity factors and first stress exponents for the anti-symmetric deformation mode, $\tilde{k}^{(a)}$ and $\delta^{(a)}$, respectively, for various notch angles and depths in the isotropic plate.

α^o	$\delta^{(a)}$	$\tilde{k}^{(a)}$			
		$a/w = 0.1$	$a/w = 0.2$	$a/w = 0.3$	$a/w = 0.4$
0	-0.50000	0.33579	0.69419	0.99817	1.44628
10	-0.47064	0.37336	0.72977	1.07507	1.56578
30	-0.40181	0.44167	0.84753	1.26967	1.94436
50	-0.31770	0.51813	0.99285	1.49989	2.42030
70	-0.21556	0.61035	1.16750	1.79244	3.06753
90	-0.09147	0.71492	1.38086	2.16659	4.03363
110	0.06022	0.82167	1.59516	2.64747	5.52536

Table 3.11 Non-dimensional generalized stress intensity factors and first stress exponents for the symmetric deformation mode, $\tilde{k}^{(s)}$ and $\delta^{(s)}$, respectively, for various notch angles and depths in a composite plate with the 0^{o} fiber orientation.

α^{o}	$\delta^{(s)}$	$\tilde{k}^{(s)}$			
		$a/w = 0.1$	$a/w = 0.2$	$a/w = 0.3$	$a/w = 0.4$
0	-0.50000	1.07106	1.11535	1.23602	1.57131
10	-0.49999	1.07112	1.11544	1.23490	1.57249
30	-0.49973	1.07028	1.12099	1.23422	1.57501
50	-0.49869	1.07661	1.12362	1.23980	1.57630
70	-0.49620	1.08515	1.12995	1.24899	1.58646
90	-0.49110	1.09721	1.14566	1.26444	1.60643
110	-0.48102	1.12334	1.16772	1.29017	1.64320
130	-0.45995	1.15970	1.20674	1.33665	1.72677
145	-0.42642	1.20485	1.25737	1.40403	1.86303
160	-0.34978	1.28538	1.35807	1.56599	*NA*

Table 3.12 Non-dimensional generalized stress intensity factors and first stress exponents for the anti-symmetric deformation mode, $\tilde{k}^{(a)}$ and $\delta^{(a)}$, respectively, for various notch angles and depths in a composite plate with the 0^{o} fiber orientation.

α^{o}	$\delta^{(a)}$	$\tilde{k}^{(a)}$			
		$a/w = 0.1$	$a/w = 0.2$	$a/w = 0.3$	$a/w = 0.4$
0	-0.50000	0.50717	0.86814	1.12681	1.51139
10	-0.47860	0.52321	0.90406	1.18036	1.61502
30	-0.43071	0.58193	0.99929	1.31689	1.85442
50	-0.37605	0.65423	1.10611	1.45705	2.13248
70	-0.31418	0.73018	1.21934	1.61563	2.45127
90	-0.24338	0.80581	1.34402	1.79496	2.84104
110	-0.15914	0.88895	1.46443	2.00386	3.35821

Table 3.13 Non-dimensional generalized stress intensity factors and first stress exponents for the symmetric deformation mode, $\tilde{k}^{(s)}$ and $\delta^{(s)}$, respectively, for various notch angles and depths in a composite plate with the 90^{o} fiber orientation.

α^{o}	$\delta^{(s)}$	$\tilde{k}^{(s)}$			
		$a/w = 0.1$	$a/w = 0.2$	$a/w = 0.3$	$a/w = 0.4$
0	-0.50000	1.05774	1.09683	1.21594	1.55444
10	-0.49944	1.06010	1.10079	1.21851	1.56149
30	-0.48996	1.09443	1.13141	1.25420	1.60704
50	-0.46945	1.13946	1.17655	1.30454	1.68874
70	-0.44010	1.18595	1.22552	1.36715	1.80713
90	-0.40146	1.23218	1.27946	1.44598	1.97095
110	-0.35109	1.27617	1.33710	1.54638	2.19305
130	-0.28474	1.31414	1.40427	1.68994	2.53177
145	-0.22077	1.32618	1.46385	1.83971	2.92922
160	-0.14063	1.31241	1.52441	2.04756	*NA*

Table 3.14 Non-dimensional generalized stress intensity factors and first stress exponents for the anti-symmetric deformation mode, $\tilde{k}^{(a)}$ and $\delta^{(a)}$, respectively, for various notch angles and depths in a composite plate with the 90^{o} fiber orientation.

α^{o}	$\delta^{(a)}$	$\tilde{k}^{(a)}$			
		$a/w = 0.1$	$a/w = 0.2$	$a/w = 0.3$	$a/w = 0.4$
0	-0.50000	0.49025	0.88316	1.09616	1.45143
10	-0.40962	0.53275	0.96322	1.34408	1.87593
30	-0.23149	0.60478	1.13888	1.70931	2.88546
50	-0.09307	0.69862	1.31366	2.05732	3.95705
70	0.02480	0.74869	1.45329	2.42589	5.06479
90	0.14036	0.81055	1.59709	2.75393	6.40582
110	0.26676	0.87457	1.74662	3.08188	8.06767

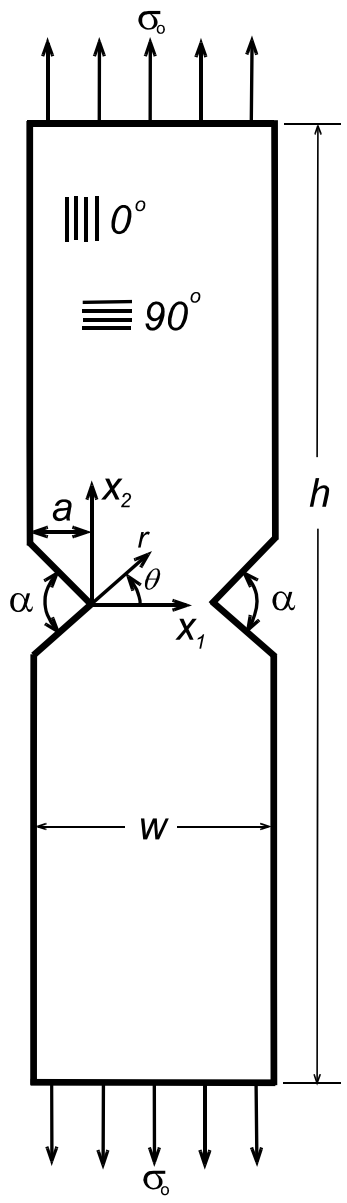

Figure 3.1 Double V-notched rectangular plate with uniform tensile stress applied to produce symmetric deformation. (h = 76 *mm* , w = 19 *mm*)

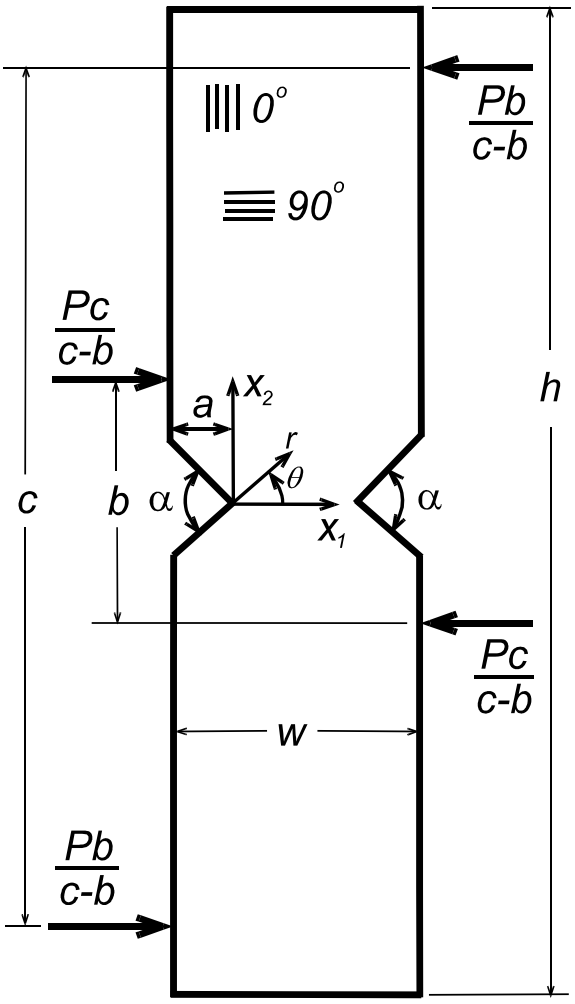

Figure 3.2 Double V-notched rectangular plate with force-couple loading applied to produce anti-symmetric deformation. (h = 76 *mm*, w = 19 *mm*, b = 22.8 *mm*, c = 69 *mm*)

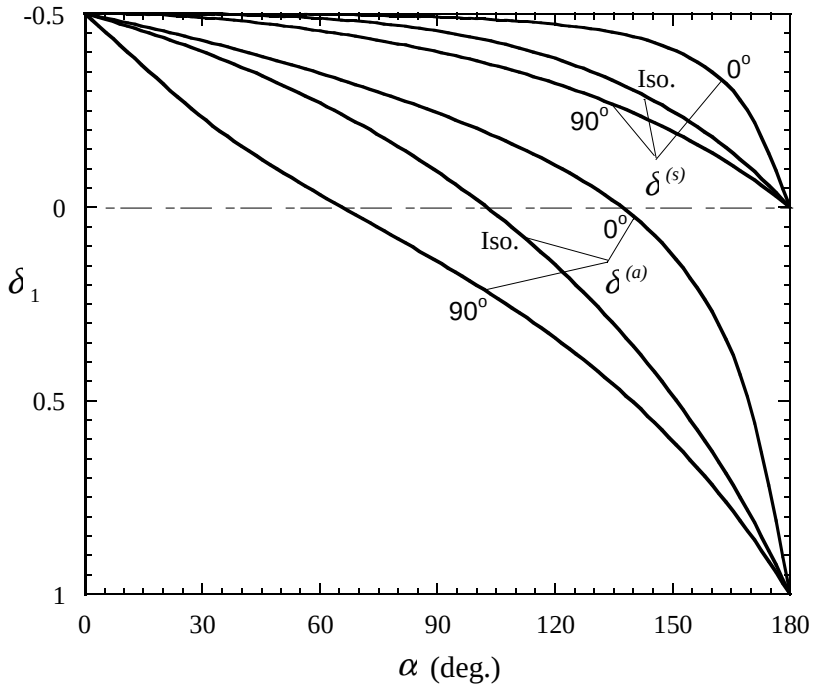

Figure 3.3 The variation of $\delta^{(s)}$ and $\delta^{(a)}$ with α for a composite with 0^o and 90^o fiber orientations (plane stress) and an isotropic material. ($\delta_1 = 0$ is always a root for symmetric deformation)

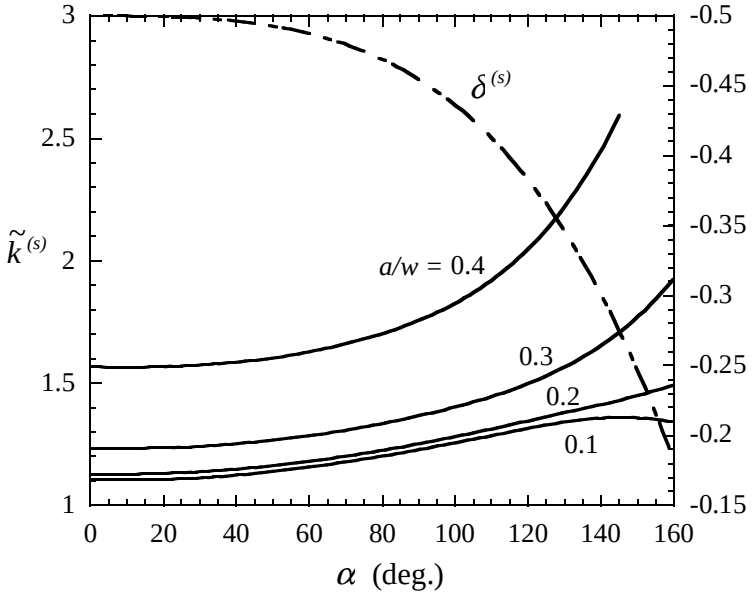

Figure 3.4 The variation of $\tilde{k}^{(s)}$ and $\delta^{(s)}$ with notch angles, α, for various *a/w* in an isotropic plate.

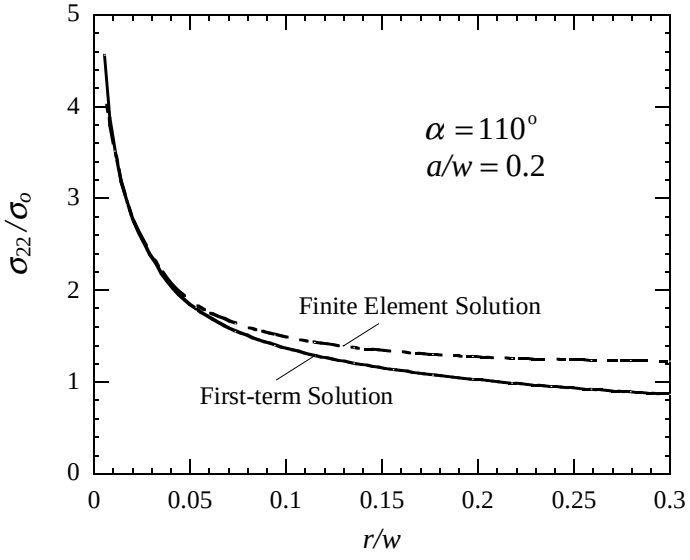

Figure 3.5 The distribution of σ_{22}/σ_o ahead of the notch tip ($\theta = 0^o$) for symmetric deformation of an isotropic plate. ($\delta^{(s)} = -0.41372$)

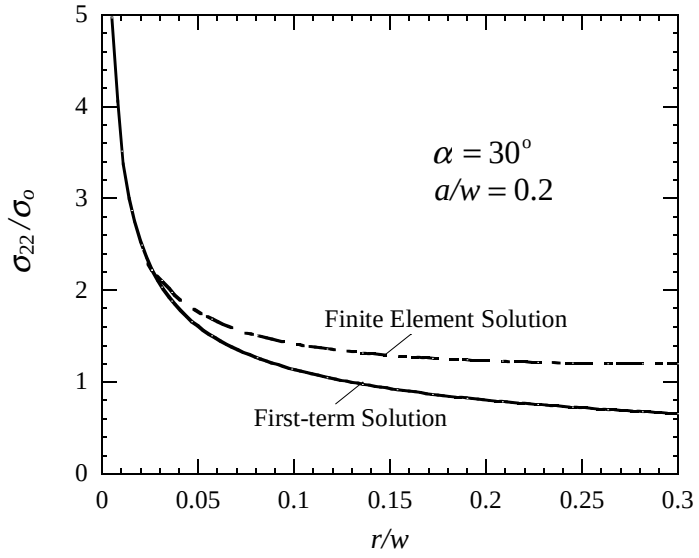

Figure 3.6 The distribution of σ_{22}/σ_o ahead of the notch tip ($\theta = 0^o$) for symmetric deformation of an isotropic plate. ($\delta^{(s)} = -0.49855$)

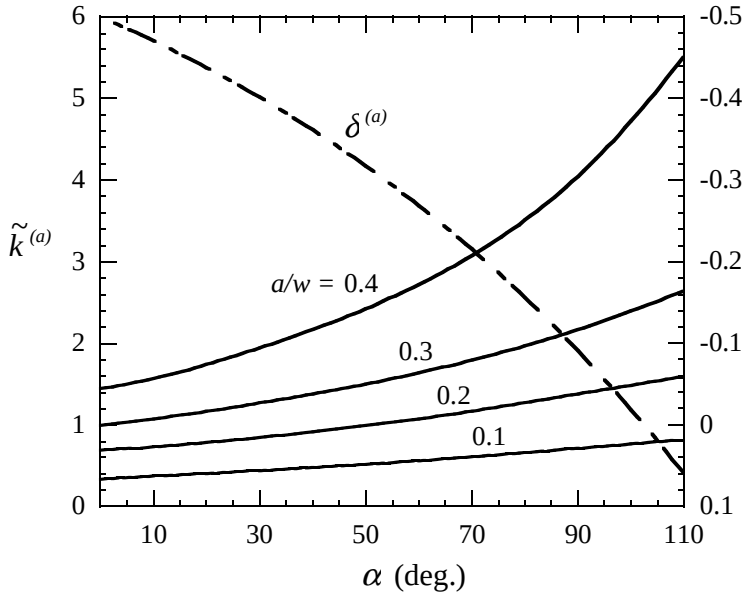

Figure 3.7 The variation of $\tilde{k}^{(a)}$ and $\delta^{(a)}$ with notch angles, α, for various *a/w* in an isotropic plate.

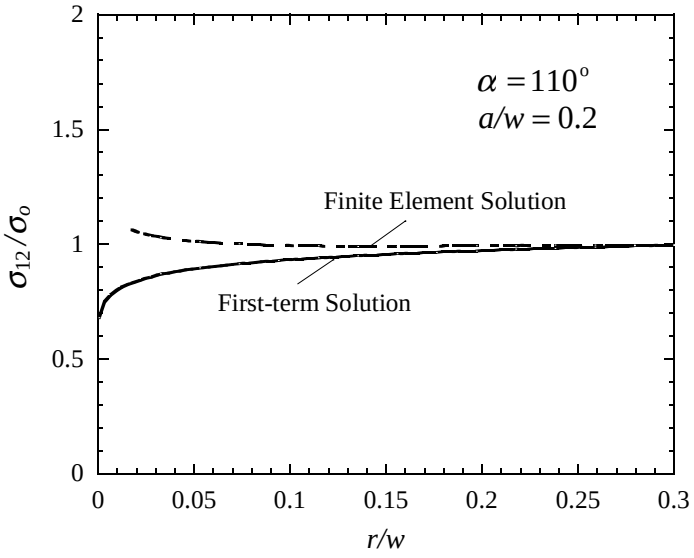

Figure 3.8 The distribution of σ_{12}/σ_o ahead of the notch tip ($\theta = 0^o$) for anti-symmetric deformation of an isotropic plate. ($\delta^{(a)} = 0.06022$)

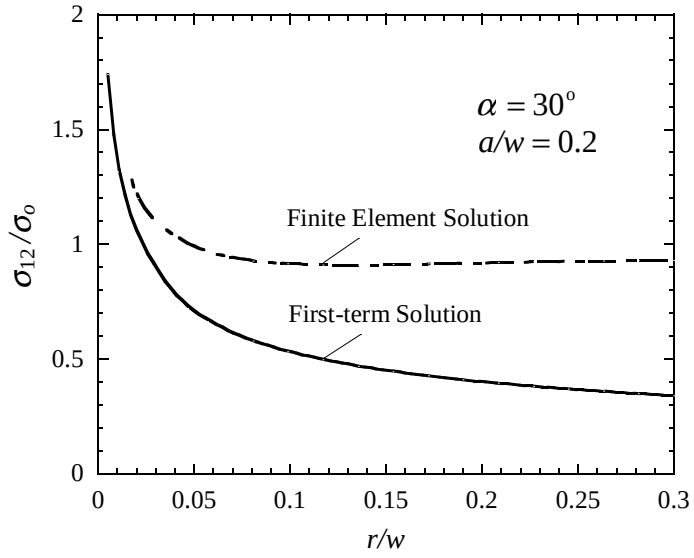

Figure 3.9 The distribution of σ_{12}/σ_o ahead of the notch tip ($\theta = 0^o$) for anti-symmetric deformation of an isotropic plate. ($\delta^{(a)} = -0.40181$)

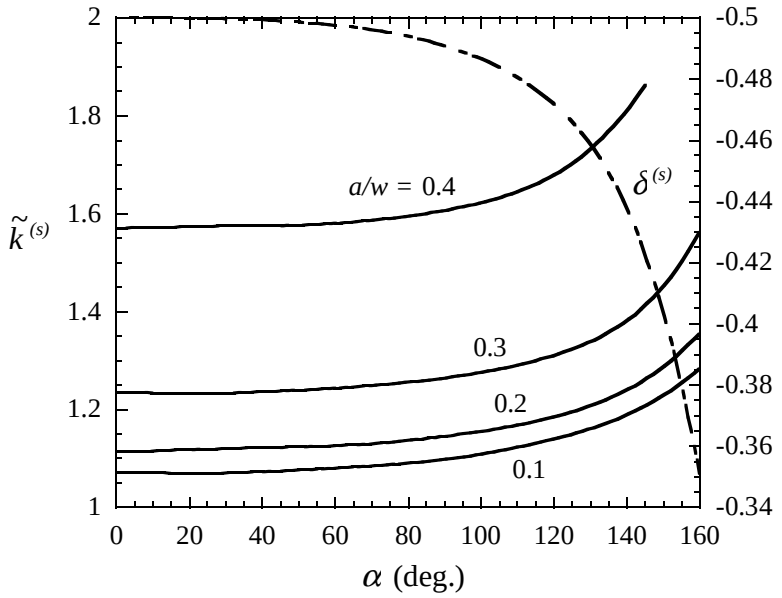

Figure 3.10 The variation of $\tilde{k}^{(s)}$ and $\delta^{(s)}$ with notch angles, α, for various *a/w* in a composite plate with the 0^o fiber orientation.

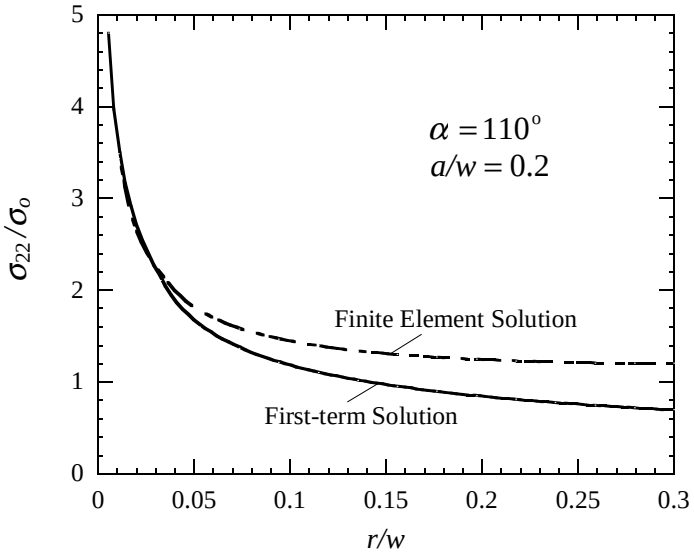

Figure 3.11 The distribution of σ_{22}/σ_o ahead of the notch tip ($\theta = 0^o$) for symmetric deformation with the 0^o fiber orientation. ($\delta^{(s)} = -0.48102$)

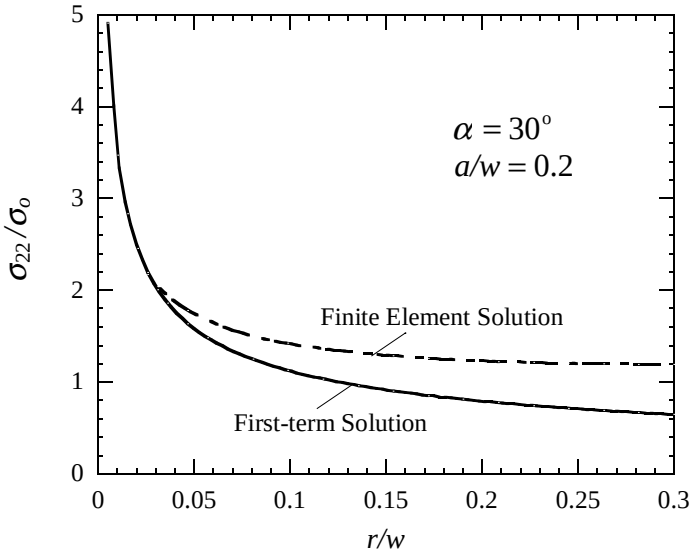

Figure 3.12 The distribution of σ_{22}/σ_o ahead of the notch tip $(\theta = 0^o)$ for symmetric deformation with the 0^o fiber orientation. $(\delta^{(s)} = -0.49973)$

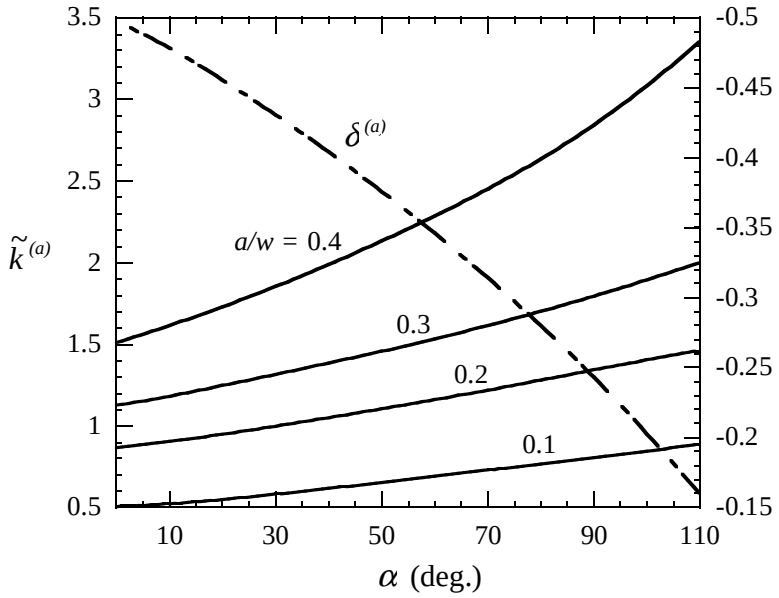

Figure 3.13 The variation of $\tilde{k}^{(a)}$ and $\delta^{(a)}$ with notch angles, α, for various *a/w* in a composite plate with the 0^o fiber orientation.

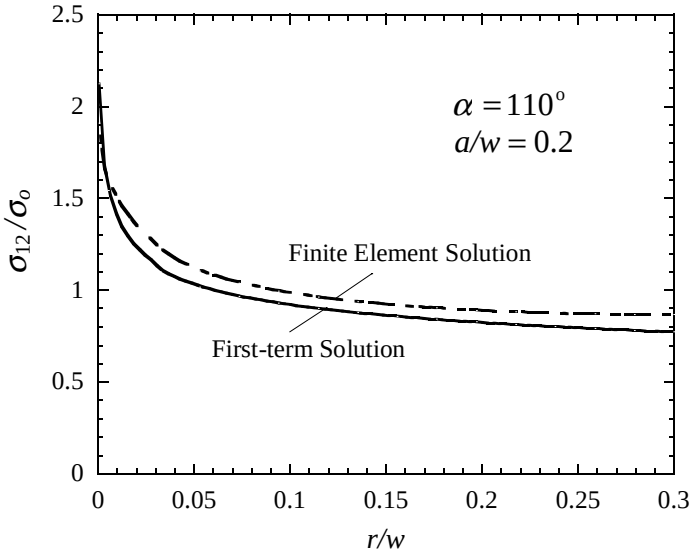

Figure 3.14 The distribution of σ_{12}/σ_o ahead of the notch tip ($\theta = 0^o$) for anti-symmetric deformation with the 0^o fiber orientation. ($\delta^{(a)} = -0.15914$)

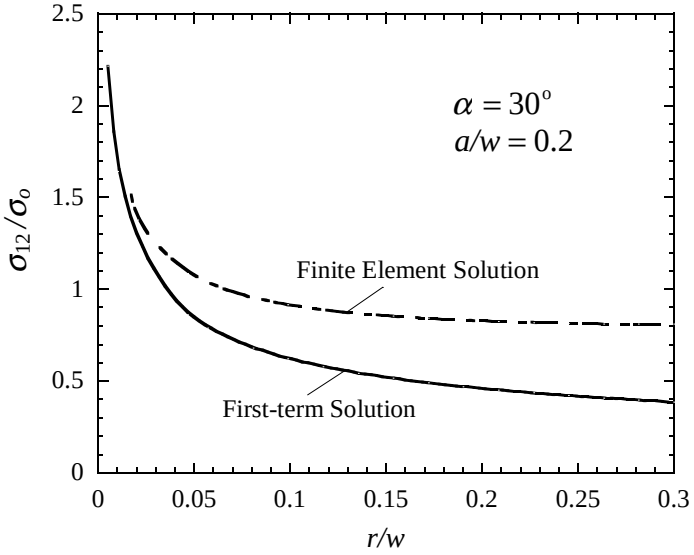

Figure 3.15 The distribution of σ_{12}/σ_o ahead of the notch tip ($\theta = 0^o$) for anti-symmetric deformation with the 0^o fiber orientation. ($\delta^{(a)} = -0.43071$)

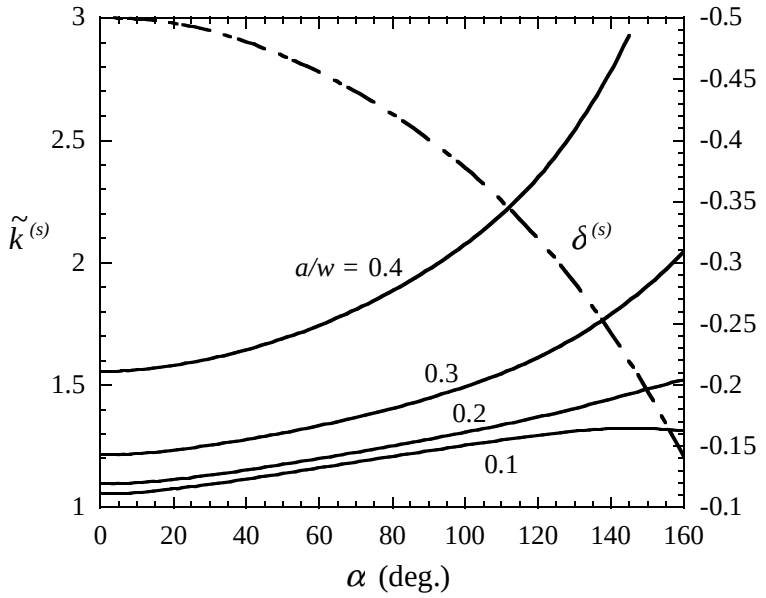

Figure 3.16 The variation of $\tilde{k}^{(s)}$ and $\delta^{(s)}$ with notch angles, α, for various *a/w* in a composite plate with the 90^{o} fiber orientation.

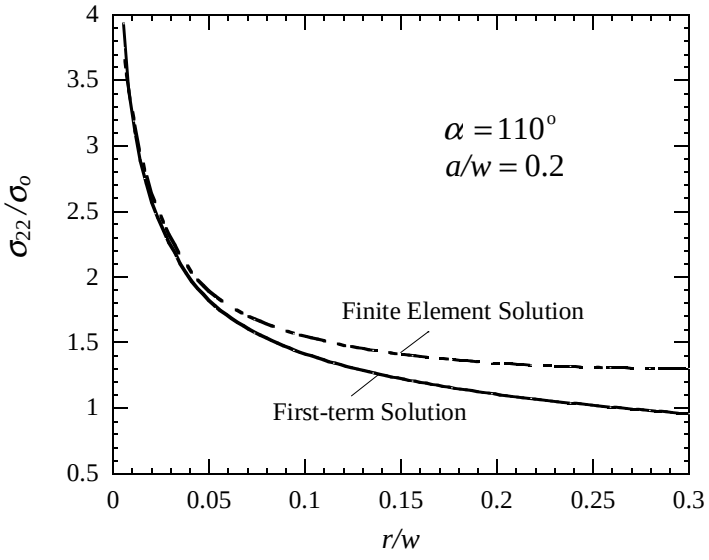

Figure 3.17 The distribution of σ_{22}/σ_{o} ahead of the notch tip ($\theta = 0^{o}$) for symmetric deformation with the 90^{o} fiber orientation. ($\delta^{(s)} = -0.35109$)

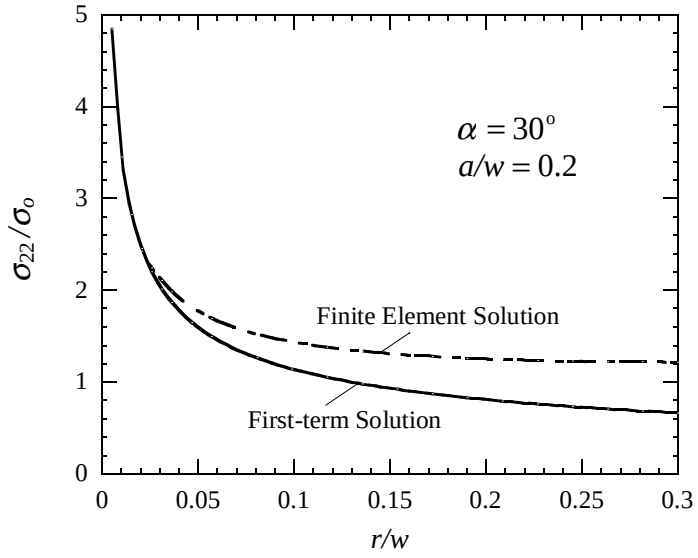

Figure 3.18 The distribution of σ_{22}/σ_o ahead of the notch tip ($\theta = 0^o$) for symmetric deformation with the 90^o fiber orientation. ($\delta^{(s)} = -0.48996$)

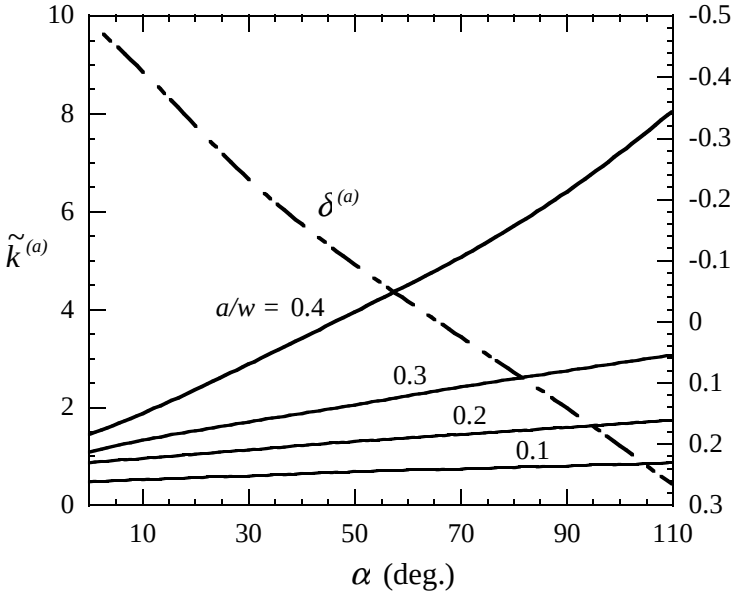

Figure 3.19 The variation of $\tilde{k}^{(a)}$ and $\delta^{(a)}$ with notch angles, α, for various *a/w* in a composite plate with the 90^o fiber orientation.

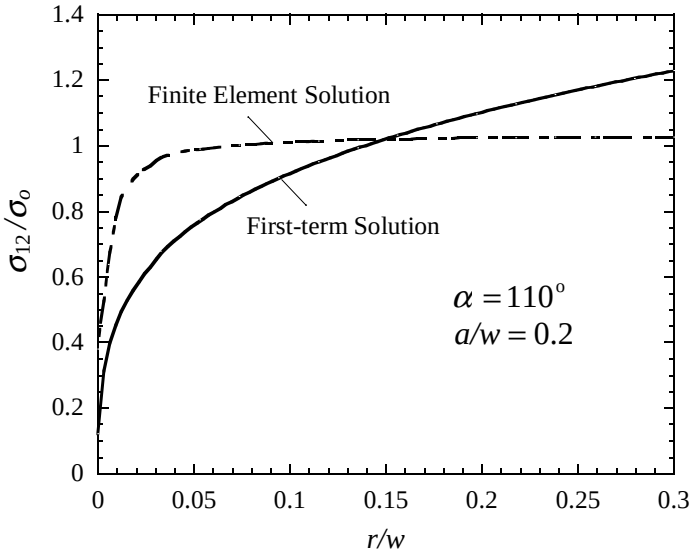

Figure 3.20 The distribution of σ_{12}/σ_o ahead of the notch tip ($\theta = 0^o$) for anti-symmetric deformation with the 90^o fiber orientation. ($\delta^{(a)} = 0.26676$)

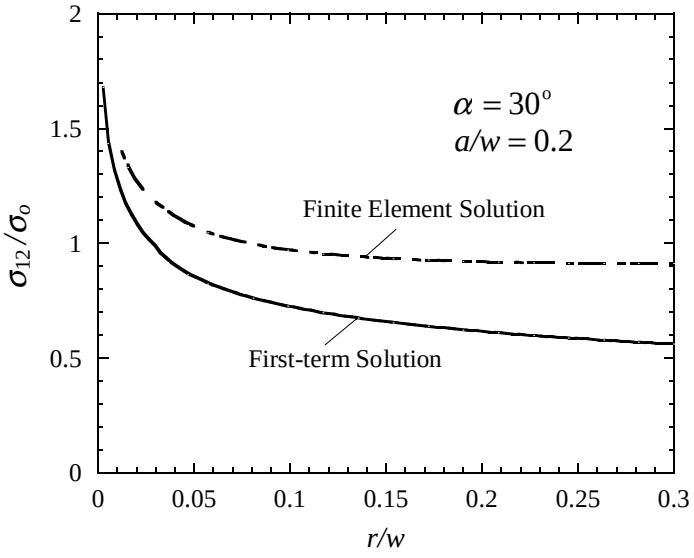

Figure 3.21 The distribution of σ_{12}/σ_o ahead of the notch tip ($\theta = 0^o$) for anti-symmetric deformation with the 90^o fiber orientation. ($\delta^{(a)} = -0.23149$)

4 Summary and Conclusions

A compact form based on Stroh formalism for extracting generalized stress intensity factors (symmetric and anti-symmetric) for a V-notched anisotropic body have been derived using a path-independent interaction *M*-integral. The formula can be further applied to the degenerate cases such as a cracked body, an isotropic solid, etc. Two types of loadings, which introduce symmetric and anti-symmetric deformation modes, respectively, have been studied in detail. The generalized stress intensity factor results have been first verified for two general notch angle cases for an isotopic rectangular plate under symmetric deformation and one case under anti-symmetric deformation. This verification demonstrates the accuracy of this generalized stress intensity factor method.

The generalized stress intensity factors were then calculated for the geometry of the Iosipescu shear test specimen used in ASTM standard D 5379/D 5379M –93 for shear property testing of fiber-reinforced composite materials. For this geometry, the generalized stress intensity factors were found for various notch depths and notch angles under both symmetric and anti-symmetric deformation modes for two orthogonal in-plane fiber orientations of a fiber-reinforced composite material and an isotropic material. The generalized stress intensity factor and its angular distribution could be used to correlate the fracture load and failure paths. The generalized stress intensity factor could also be used to predict how a chosen notch depth and angle would affect the fracture load. This would allow the notch angle and depth to be tailored to provide for more accurate prediction of the material strength. Finally, the coefficients for the high-order real stress exponents can be obtained in a similar manner.

References

[1] ASTM D5379/D5379M – 93 Standard Test Method for Shear Properties of Composite Materials by the V-Notched Beam Method.

[2] Atkinson, C., Bastero, J. M., and Martinez-Esnaola, J. M., "Stress Analysis in Sharp Angular Notches using Auxiliary Fields," *Engineering Fracture Mechanics*, Vol. 31, pp. 637-646, 1988.

[3] Hull, D. and Kumosa, M., "Mixed-mode fracture of composites using Iosipescu Shear Test," *International Journal of Fracture*, Vol. 35, pp. 83-102, 1987.

[4] Hyer, M. W., *Stress Analysis of Fiber-Reinforced Composite Materials*, McGraw-Hill Companies, Inc., New York, NY, U.S.A, 1998.

[5] Im, S., Jeon, I., and Kim, Y., "Enriched Finite Element Analysis for a Delamination Crack in a Laminated Composite Strip," *Composite Mechanics*, Vol. 16, pp. 1-8, 1996.

[6] Im, S. and Kim, K. S., "An application of two-state *M*-integral for computing the intensity of the singular near-tip field for a generic wedge," *Journal of Mechanics and Physics of Solids*, Vol. 48, pp. 129-151, 2000.

[7] Iosipescu, N., "New Accurate Procedure for Single Shear Testing of Metals," *Journal of Materials*, Vol. 2, pp. 537-566, 1967.

[8] Lekhnitskii, S. G., *Theory of Elasticity of an Anisotropic Body*, Mir Publishers, Moscow, Russia, 1981.

[9] Murakami, Y., *Stress Intensity Factors Handbook Vol. 1*, Pergamon Press, New York, 1987.

[10] Pian, T. H. H., Larsry, S. J., and Tong. P., "A Hybrid-Element Approach to Crack Problems in Plane Elasticity," *International Journal of Numerical Methods in Engineering*, Vol. 7, pp. 297-308, 1973.

[11] Ting, T. C. T., *Anisotropic Elasticity: Theory and Applications*, Oxford University Press, Inc., New York, NY, U.S.A, 1996.

[12] Wang, S. S. and Yuan, F. G., "A Hybrid Finite Element Approach to Composite

Laminate Elasticity Problems with Singularities," ASME, *Journal of Applied Mechanics*, Vol. 105, pp. 835-844, 1983.

[13] Wu, K. C. and Chen, C. T., "Stress Analysis of Anisotropic Elastic V-notched Bodies," *International Journal of Solids Structures*, Vol. 33, pp. 2403-2416, 1996.

[14] Yuan, F. G., "Determination of Stress Coefficient Terms in Cracked Solids for Monoclinic Materials with Plane Symmetry at $x_3 = 0$", NASA CR-1998-208729, October 1998.

Appendices

Calculating the generalized stress intensity factors in the following equation

$$k^{(i)} = \frac{2(\delta^{(i)}+1)M_M}{(2\pi)^{\delta^{(i)}}\ \mathrm{Re}[\boldsymbol{h}^{(i)T}\boldsymbol{B}^{-T}\langle R^{(i)}\rangle \boldsymbol{B}^{-1}\tilde{\boldsymbol{g}}^{(i)}]} \qquad i = a,\ s \tag{2.63}$$

requires computer programs and a finite element solution. The FORTRAN program SINGULARITY in Appendix A.1 calculates the variables that are a function of the material properties and notch angle alone. These variables include the eigenvalues of the elastic constant, μ_α, the complex matrix containing Stroh eigenvectors, $\boldsymbol{B}$, the first stress exponents, $\delta^{(i)}$ (where $i = s$ for symmetric deformation and $i = a$ for anti-symmetric deformation), the normalized first complex eigenvectors, $\tilde{\boldsymbol{g}}^{(i)}$, and the auxiliary first complex eigenvectors, $\boldsymbol{h}^{(i)}$. The variables $\delta^{(i)}$, $\tilde{\boldsymbol{g}}^{(i)}$, and $\boldsymbol{h}^{(i)}$ satisfy the material constitutive equations and traction-free boundary conditions on the notch surfaces.

The actual field solution needed to evaluate M_M can be found using a finite element model. Appendix B.1 contains the ANSYS program to produce the finite element model for the case with $\alpha = 90^o$, $a/w = 0.2$, 0^o fiber orientation, and loading for the anti-symmetric deformation mode. A numeric value for M_M was evaluated for three independent paths for each case. These paths through the finite element model were produced using the FORTRAN program GAUSSPOINT in Appendix A.2. The version of GAUSSPOINT in Appendix A.2 is edited to produce the third path for the case with $\alpha = 90^o$ and $a/w = 0.2$. The input file for GAUSSPOINT, which contains the element numbers, the nodes that make up each element, and the coordinates of the nodes, is created using the ANSYS program NODEE in Appendix B.2. The output file from GAUSSPOINT, 135AW2path3.txt, is in Appendix A.2. This output file is an ANSYS program called PATH. Using the solution to the finite element model in ANSYS, PATH calls the ANSYS program MIN in Appendix B.3 to interpolate the solution results to the points on the path. The output file produced for this example is 135AAW22mint3.txt.

Finally, to calculate a numeric value of M_M, 135AAW22mint3.txt and Eigen.txt are read by MINTCALC as it is executed.

The numeric values for M_M were produced using the FORTRAN program MINTCALC in Appendix A.3. The form of M_M that is numerically integrated using MINTCALC is now derived.

Starting with

$$M_M = \int_\Gamma [(\sigma_{i2}x_1 - \sigma_{i1}x_2)\frac{du_i^a}{ds} - t_i^a u_{i,k} x_k]ds \tag{1}$$

and expanding M_M for two dimenions, gives

$$M_M = \int_\Gamma [[(\sigma_{12}x_1 - \sigma_{11}x_2)\frac{du_1^a}{ds} + (\sigma_{22}x_1 - \sigma_{21}x_2)\frac{du_2^a}{ds}] \tag{2}$$

$$-[t_1^a(\frac{du_1}{dx_1}x_1 + \frac{du_1}{dx_2}x_2) + t_2^a(\frac{du_2}{dx_1}x_1 + \frac{du_2}{dx_2}x_2)]]ds$$

Now, substituting $t_i^a = \sigma_{ij}^a n_j$ into Eq. (2) gives

$$M_M = \int_\Gamma [[(\sigma_{12}x_1 - \sigma_{11}x_2)\frac{du_1^a}{ds} + (\sigma_{22}x_1 - \sigma_{21}x_2)\frac{du_2^a}{ds}] \tag{3}$$

$$-[(\sigma_{11}^a n_1 + \sigma_{12}^a n_2)(\frac{du_1}{dx_1}x_1 + \frac{du_1}{dx_2}x_2) + (\sigma_{21}^a n_1 + \sigma_{22}^a n_2)(\frac{du_2}{dx_1}x_1 + \frac{du_2}{dx_2}x_2)]]ds$$

Finally, substituting $\frac{du_i^a}{ds} = \frac{du_i^a}{dx_2}n_1 - \frac{du_i^a}{dx_1}n_2$ into Eq. (3), gives a form of the interaction *M*-integral that can be evaluated numerically from the finite element solution and auxiliary field data,

$$M_M = \int_\Gamma [[(\sigma_{12}x_1 - \sigma_{11}x_2)(\frac{du_1^a}{dx_2}n_1 - \frac{du_1^a}{dx_1}n_2) + (\sigma_{22}x_1 - \sigma_{21}x_2)(\frac{du_2^a}{dx_2}n_1 - \frac{du_2^a}{dx_1}n_2)] \tag{4}$$

$$-[(\sigma_{11}^{a}n_1+\sigma_{12}^{a}n_2)(\frac{du_1}{dx_1}x_1+\frac{du_1}{dx_2}x_2)+(\sigma_{21}^{a}n_1+\sigma_{22}^{a}n_2)(\frac{du_2}{dx_1}x_1+\frac{du_2}{dx_2}x_2)]]ds$$

Evaluating M_M requires an actual field solution and a corresponding auxiliary field solution. The auxiliary field solution is calculated from Eigen.txt (Appendix A.1), which contains the auxiliary first stress exponent and auxiliary first complex vector.

A. FORTRAN Programs

The following FORTRAN programs are used in calculating the generalized stress intensity factors, $k^{(s)}$ and $k^{(a)}$. These programs are written in the FORTRAN 77 syntax.

A.1 SINGULARITY

The FORTRAN program SINGULARITY calculates the eigenvalues of elastic constants, μ_α, the matrices of Stroh eigenvectors, $\boldsymbol{A}$ and $\boldsymbol{B}$, the eigenvalues that are the stress exponents in Stroh and Lekhnitskii formalisms, δ_n, the normalized Stroh eigenvectors corresponding to δ_n, $\tilde{\boldsymbol{g}}_n$, the auxiliary eigenvalues, Δ_m, and the Stroh auxiliary eigenvectors corresponding to Δ_m, h_m.

```
PROGRAM SINGULARITY

*       This program calculates the stress exponent eigenvalues and their corresponding eigenvectors for a
* V-notched anisotropic homogeneous body.  In the most general case, the eigenvalues can be calculated
* for a V-notch at the interface of two anisotropic materials.
*
*   Definition of Selected Variables
*
*   A(i,j,k)        The Stroh eigenvector A for material i = 1..2.  The reduced 2 x 2 form of A is
*                   calculated for in-plane deformation, so j = 1..2 and k = 1..2.
*
*   B(i,j,k)        The Stroh eigenvector B for material i = 1..2.  The reduced 2 x 2 form of B is
*                   calculated for in-plane deformation, so j = 1..2 and k = 1..2.
*
*   E1(i)           Young's modulus in the x1 direction. (the origin of the Cartesian
*                   coordinate system is at the notch tip with the x1 axis bisecting the
*                   notch.  The x1 axis is along the interface for the case of two different
*                   materials.) (i = 1 represents the material in the positive x2 direction
*                   and i = 2 represents the material in the negative x2 direction.)
*
*   E2(i)           Young's modulus in the x2 direction for material i = 1..2.
*
*   E3(i)           Young's modulus in the x3 direction for material i = 1..2.
*
*   G23(i)          Shear modulus for the x2-x3 plane for material i = 1..2.
*
*   G13(i)          Shear modulus for the x1-x3 plane for material i = 1..2.
*
*   G12(i)          Shear modulus for the x1-x2 plane for material i = 1..2.
*
*   MU1(i)          The first Stroh eigenvalue for material i = 1..2.
*
*   MU2(i)          The second Stroh eigenvalue for material i = 1..2.
*
*   Q(i)            The eigenvector for the chosen eigenvalue E.
*
*   RS(i,j,k)       Array for the reduced compliance matrix for plane strain for material
*                   i= 1..2., j = 1..6, k = 1..6.
*
*   ROOT(i)         Initial guesses for the stress exponent solution in the input file
*                   guesses.txt.
*
*   S(i,j,k)        Array for the compliance matrix for material i = 1..2. j = 1..6, k = 1..6.
*
*   THETA(i)        The angle in degrees from the x1 axis to the notch surface for material i with the
*                   counter-clockwise direction positive.
*
*   V23(i)          Poisson's ratio for strain in the x3 direction resulting from stress
*                   applied in the x2 direction for material i = 1..2.
*
*   V13(i)          Poisson's ratio for strain in the x3 direction resulting from stress
*                   applied in the x1 direction for material i = 1..2.
```

```
*
*     V12(i)          Poisson's ratio for strain in the x2 direction resulting from stress
*                     applied in the x1 direction for material i = 1..2.
*
*     Z(i)            The stress exponent eigenvalues.
*
************************************************************************

      INTEGER NUM,E
      DOUBLE PRECISION E1(2),E2(2),E3(2),G23(2),G13(2),G12(2),V23(2)
      DOUBLE PRECISION V13(2),V12(2),S(2,6,6),RS(2,6,6),THETA(2)
      COMPLEX*16 MU1(2),MU2(2),ROOT(20),Z(20)
      COMPLEX*16 A(2,2,2),B(2,2,2),BI(2,2,2),Q(4)

      DATA E1(1),E2(1),E3(1)/8.9,138.0,8.9/
      DATA G23(1),G13(1),G12(1)/5.17,2.89,5.17/
      DATA V23(1),V13(1),V12(1)/.30,.54,.0193478/

      DATA E1(2),E2(2),E3(2)/8.9,138.0,8.9/
      DATA G23(2),G13(2),G12(2)/5.17,2.89,5.17/
      DATA V23(2),V13(2),V12(2)/.30,.54,.0193478/

* Angle from material interface to notch face of material 1
            THETA(1)=135.D0

* Angle from material interface to notch face of material 2
            THETA(2)=-THETA(1)

* Calculates the elastic compliances.
            CALL COMPLIANCES(E1,E2,E3,G23,G13,G12,V23,V13,V12,S,RS)

* Calculates the roots of the characteristics equation from Stroh formalism.  The variable S is input for the
* case of plane stress.  The variable S is replaced by RS for the case of in-plane deformation (plane strain).
            CALL CHARACT(S,MU1,MU2)

* Calculates the eigenvalues which are the stress exponents in Stroh and Lekhnitskii
*  formalisms.  The variable S is input for the case of plane stress.  The variable S is replaced by RS for
*  the case of in-plane deformation (plane strain).
            CALL SINGSOL(S,MU1,MU2,THETA,NUM,ROOT,Z)

* Calculates the Stroh eigenvectors A and B and the inverse of B, BI.  The variable S is input for the case
*  of plane stress.  The variable S is replaced by RS for the case of in-plane deformation (plane strain).
            CALL STROHEGN(S,MU1,MU2,A,B,BI)

* Calculates the eigenvector, Q, cooresponding to the eigenvalue, Z.
            CALL EIGENVECTOR(B,BI,MU1,MU2,THETA,Z,Q,E)

* Outputs the data to the files Eigen.txt and sing.txt.
            CALL OUTPUT(E1,E2,E3,G23,G13,G12,V23,V13,V12,S,RS,MU1,MU2,NUM,
           &                  ROOT,Z,A,B,BI,Q,E)

            STOP
```

```
      END


      SUBROUTINE COMPLIANCES(E1,E2,E3,G23,G13,G12,V23,V13,V12,S,RS)

**********************************************************************
*
*     This subroutine calculates the elastic compliances from the material properties.
*
*     Input: E1,E2,E3,G23,G13,G12,V23,V13,V12
*
*     Output: S,RS
*
**********************************************************************

      INTEGER I,J,M
      DOUBLE PRECISION E1(2),E2(2),E3(2),G23(2),G13(2),G12(2),V23(2)
      DOUBLE PRECISION V13(2),V12(2),S(2,6,6),RS(2,6,6)

      DO 3 M=1,2
            DO 4 I=1,6
                  DO 5 J=1,6
                        S(M,1,1)=1.D0/E1(M)
                        S(M,1,2)=-V12(M)/E1(M)
                        S(M,1,3)=-V13(M)/E1(M)
                        S(M,2,2)=1.D0/E2(M)
                        S(M,2,3)=-V23(M)/E2(M)
                        S(M,3,3)=1.D0/E3(M)
                        S(M,4,4)=1.D0/G23(M)
                        S(M,5,5)=1.D0/G13(M)
                        S(M,6,6)=1.D0/G12(M)
                        S(M,J,I)=S(M,I,J)
5                 CONTINUE
4           CONTINUE
3     CONTINUE


*  RS(M,I,J) are the reduced compliance constants for in-plane deformation for material M.
      DO 80 M=1,2
            DO 81 I=1,6
                  DO 82 J=1,6
                        RS(M,I,J)=S(M,I,J)-S(M,3,I)*S(M,3,J)/S(M,3,3)
                        RS(M,J,I)=RS(M,I,J)
82                CONTINUE
81          CONTINUE
80    CONTINUE

      RETURN
      END
```

```
      SUBROUTINE CHARACT(RS,MU1,MU2)

************************************************************************************
*
*       This subroutine finds the roots of the characteristic equation, which are the eigenvalues of elastic
*  constants, μ_α .
*
*       Input: RS(i,j,k)
*
*       Output: MU1(i),MU2(i)
*
************************************************************************************

      INTEGER M
      DOUBLE PRECISION RS(2,6,6)
      COMPLEX*16 MU1(2),MU2(2),CD1,CD2,D1

      DO 75 M=1,2
            CD1=2*RS(M,1,2)+RS(M,6,6)
            CD2=(2*RS(M,1,2)+RS(M,6,6))**2-4*RS(M,1,1)*RS(M,2,2)
            D1=CDSQRT(CD2)
            MU1(M)=CDSQRT(0.5D0*(-CD1+D1)/RS(M,1,1))
            MU2(M)=CDSQRT(0.5D0*(-CD1-D1)/RS(M,1,1))
75    CONTINUE
      RETURN
      END


      SUBROUTINE SINGSOL(RS,MU1,MU2,THETA,NUM,IZ,Z)

************************************************************************
*
*       This subroutine reads the initial guesses for the eigenvalues, ROOT(i), and calls the
*   subroutine MULLER to find the eigenvalues, Z(i).
*
*   Input: RS(i,j,k),MU1(i),MU2(i),THETA(i)
*
*   Output: NUM,IZ(i),Z(i)
*
*   Definition of local variables:
*
*   NUM             Number of initial guesses for the eigenvalues.
*
*   ROOT(i), IZ(i)  Initial eigenvalue guesses.
*
*   Z(i)            Eigenvalue solutions (the stress exponents in the Stroh and
*                   Lekhnitskii formalisms).
*
************************************************************************

      INTEGER I,NUM
```

```
      DOUBLE PRECISION RS(2,6,6),THETA(2)
      COMPLEX*16 MU1(2),MU2(2),ROOT(20),Z(20),IZ(20),FZ

      OPEN(5,FILE='guesses.txt',STATUS='OLD')

      READ(5,*)NUM

      DO 27 I=1,NUM
            READ(5,140)ROOT(I)
140         FORMAT(2D9.6)
            IZ(I)=ROOT(I)
            CALL MULLER(ROOT(I),MU1,MU2,RS,THETA,FZ)
            Z(I)=FZ
27    CONTINUE

      CLOSE(5)
      RETURN
      END



      SUBROUTINE MULLER(Z1,MU1,MU2,RS,THETA,Z)

**********************************************************************
*
*              This subroutine uses Muller's Method, which uses quadratic
*     interprelation among three points, to find the complex eigenvalues.
*
*  Input:  Z1,MU1(i),MU2(i),RS(i,j,k),THETA(i)
*
*  Output:  Z
*
**********************************************************************

      INTEGER ITER,RFOUND,LIMIT
      DOUBLE PRECISION RS(2,6,6),ERROR,THETA(2)
      COMPLEX*16 Z,Z1,Z2,Z3,Q,A,B,C,D,P,MU1(2),MU2(2),DETD

      DATA LIMIT,ERROR/60,1.D-20/

      P=(0.001D0,0.001)
      RFOUND=0

      ITER=1

      DO WHILE ((ITER.LT.LIMIT).AND.(RFOUND.EQ.0))
            Z3=Z1+P
            Z2=Z1-P

            Q=(Z3-Z1)/(Z1-Z2)
            A=Q*DETD(Z3,MU1,MU2,RS,THETA)-
     &      Q*(1+Q)*DETD(Z1,MU1,MU2,RS, THETA)+Q**2*DETD(Z2,MU1,
```

```
     &          MU2,RS,THETA)
                B=(2*Q+1)*DETD(Z3,MU1,MU2,RS,THETA)-(1+Q)**2*DETD(Z1,
     &          MU1,MU2,RS,THETA)+Q**2*DETD(Z2,MU1,MU2,RS,THETA)
     &          C=(1+Q)*DETD(Z3,MU1,MU2,RS,THETA)

                IF((CABS(B+(B**2-4*A*C)**(.5))).GT.(CABS(B-(B**2-4*A*C)**
     &          (.5))))THEN
                    D=B+(B**2-4*A*C)**(.5)
                ELSE
                    D=B-(B**2-4*A*C)**(.5)
                END IF

                Z1=Z3-(Z3-Z1)*(2*C)/D

85        FORMAT(' Z1=',2F13.9,5X,'F(Z1)=',2F20.17)
          ITER=ITER+1

                IF (CABS(DETD(Z1,MU1,MU2,RS,THETA)).LT.ERROR) THEN
                    RFOUND=1
                END IF
          END DO

          Z=Z1


          RETURN
          END


          FUNCTION DETD(Z,MU1,MU2,RS,THETA)

**********************************************************************
*
*         This function is used by the subroutine MULLER to find the eigenvalue
*    solutions.  This function forms matrices from the stress and displacement
*    coefficients from Lekhnitskii formalism and uses them to calculate the determinate
*    of the matrix CIABD, DETD.
*
*    Input: Z,MU1(i),MU2(i),RS(i,j,k),THETA(i)
*
*    Output: DETD
*
**********************************************************************

          INTEGER I
          DOUBLE PRECISION THETA(2),ALPHA,RS(2,6,6)
          COMPLEX*16 MU1(2),MU2(2),SY(2,4),SXY(2,4),DX(2,4),DY(2,4)
          COMPLEX*16 A(4,4),B(4,4),C(2,4),D(2,4),Z
          COMPLEX*16 IA(4,4),CIA(4,4),CIAB(2,4),CIABD(4,4),DETD,DET4
          COMPLEX*16 IAB(4,4),ST(2,4),SRT(2,4)

          ALPHA=0.D0    !Interface angle
```

```
      CALL STRESST(1,Z,THETA(1),MU1,MU2,ST)
      CALL STRESSRT(1,Z,THETA(1),MU1,MU2,SRT)
      DO 11 I=1,4
            C(1,I)=ST(1,I)
            C(2,I)=SRT(1,I)
11    CONTINUE

      CALL STRESST(2,Z,THETA(2),MU1,MU2,ST)
      CALL STRESSRT(2,Z,THETA(2),MU1,MU2,SRT)
      DO 21 I=1,4
            D(1,I)=ST(2,I)
            D(2,I)=SRT(2,I)
21    CONTINUE

      CALL STRESSY(1,Z,ALPHA,MU1,MU2,SY)
      CALL STRESSXY(1,Z,ALPHA,MU1,MU2,SXY)
      CALL DISPX(1,RS,Z,ALPHA,MU1,MU2,DX)
      CALL DISPY(1,RS,Z,ALPHA,MU1,MU2,DY)
      DO 31 I=1,4
            A(1,I)=SY(1,I)
            A(2,I)=SXY(1,I)
            A(3,I)=DX(1,I)
            A(4,I)=DY(1,I)
31    CONTINUE

      CALL STRESSY(2,Z,ALPHA,MU1,MU2,SY)
      CALL STRESSXY(2,Z,ALPHA,MU1,MU2,SXY)
      CALL DISPX(2,RS,Z,ALPHA,MU1,MU2,DX)
      CALL DISPY(2,RS,Z,ALPHA,MU1,MU2,DY)
      DO 41 I=1,4
            B(1,I)=SY(2,I)
            B(2,I)=SXY(2,I)
            B(3,I)=DX(2,I)
            B(4,I)=DY(2,I)
41    CONTINUE

      CALL INVERSEA(A,IA)

      CALL MATMUL(C,IA,CIA,2,4,4,4)
      CALL MATMUL(IA,B,IAB,4,4,4,4)
      CALL MATMUL(CIA,B,CIAB,2,4,4,4)
      DO 42 I=1,4
            CIABD(1,I)=CIAB(1,I)
            CIABD(2,I)=CIAB(2,I)
            CIABD(3,I)=D(1,I)
            CIABD(4,I)=D(2,I)
42    CONTINUE
      CALL DETCALC4(CIABD,DET4)
      DETD=DET4

      RETURN
```

```
END


SUBROUTINE STRESSY(MAT,D,THETA,MU1,MU2,SY)

INTEGER MAT
DOUBLE PRECISION THETA,PI
COMPLEX*16 D,MU1(2),MU2(2),ZETA1,ZETA2,SY(2,4)

PI=4.D0*DATAN(1.D0)

ZETA1=DCOS(THETA*PI/180.D0)+MU1(MAT)*DSIN(THETA*PI/180.D0)
ZETA2=DCOS(THETA*PI/180.D0)+MU2(MAT)*DSIN(THETA*PI/180.D0)

SY(MAT,1)=ZETA1**D
SY(MAT,2)=ZETA2**D
SY(MAT,3)=CONJG(ZETA1)**D
SY(MAT,4)=CONJG(ZETA2)**D

RETURN
END


SUBROUTINE STRESSXY(MAT,D,THETA,MU1,MU2,SXY)

INTEGER MAT
DOUBLE PRECISION THETA,PI
COMPLEX*16 D,MU1(2),MU2(2),ZETA1,ZETA2,SXY(2,4)

PI=4.D0*DATAN(1.D0)

ZETA1=DCOS(THETA*PI/180.D0)+MU1(MAT)*DSIN(THETA*PI/180.D0)
ZETA2=DCOS(THETA*PI/180.D0)+MU2(MAT)*DSIN(THETA*PI/180.D0)

SXY(MAT,1)=-MU1(MAT)*ZETA1**D
SXY(MAT,2)=-MU2(MAT)*ZETA2**D
SXY(MAT,3)=-CONJG(MU1(MAT))*CONJG(ZETA1)**D
SXY(MAT,4)=-CONJG(MU2(MAT))*CONJG(ZETA2)**D

RETURN
END


SUBROUTINE STRESSX(MAT,D,THETA,MU1,MU2,SX)

INTEGER MAT
DOUBLE PRECISION THETA,PI
COMPLEX*16 D,MU1(2),MU2(2),ZETA1,ZETA2,SX(2,4)

PI=4.D0*DATAN(1.D0)

ZETA1=DCOS(THETA*PI/180.D0)+MU1(MAT)*DSIN(THETA*PI/180.D0)
```

```
      ZETA2=DCOS(THETA*PI/180.D0)+MU2(MAT)*DSIN(THETA*PI/180.D0)

      SX(MAT,1)=MU1(MAT)**2*ZETA1**D
      SX(MAT,2)=MU2(MAT)**2*ZETA2**D
      SX(MAT,3)=(CONJG(MU1(MAT)))**2*CONJG(ZETA1)**D
      SX(MAT,4)=(CONJG(MU2(MAT)))**2*CONJG(ZETA2)**D

      RETURN
      END


      SUBROUTINE STRESST(MAT,D,THETA,MU1,MU2,ST)
      INTEGER MAT,I
      DOUBLE PRECISION THETA,PI
      COMPLEX*16 D,MU1(2),MU2(2),SX(2,4),SY(2,4),SXY(2,4),ST(2,4)

      PI=4.D0*DATAN(1.D0)

      CALL STRESSX(MAT,D,THETA,MU1,MU2,SX)
      CALL STRESSY(MAT,D,THETA,MU1,MU2,SY)
      CALL STRESSXY(MAT,D,THETA,MU1,MU2,SXY)

      DO 5 I=1,4
            ST(MAT,I)=SY(MAT,I)*DCOS(THETA*PI/180.D0)**2+SX(MAT,I)*
     &      DSIN(THETA*PI/180.D0)**2-2*SXY(MAT,I)*
     &      DSIN(THETA*PI/180.D0)*DCOS(THETA*PI/180.D0)
5     CONTINUE

      RETURN
      END


      SUBROUTINE STRESSRT(MAT,D,THETA,MU1,MU2,SRT)
      INTEGER MAT,I
      DOUBLE PRECISION THETA,PI
      COMPLEX*16 D,MU1(2),MU2(2),SX(2,4),SY(2,4),SXY(2,4),SRT(2,4)

      PI=4.D0*DATAN(1.D0)

      CALL STRESSX(MAT,D,THETA,MU1,MU2,SX)
      CALL STRESSY(MAT,D,THETA,MU1,MU2,SY)
      CALL STRESSXY(MAT,D,THETA,MU1,MU2,SXY)

      DO 6 I=1,4
            SRT(MAT,I)=-(SY(MAT,I)-SX(MAT,I))*DSIN(THETA*PI/180.D0)
     &      *DCOS(THETA*PI/180.D0)-SXY(MAT,I)*(DCOS(THETA
     &      *PI/180.D0)**2-SIN(THETA*PI/180.D0)**2)
6     CONTINUE

      RETURN
      END
```

```
      SUBROUTINE DISPX(MAT,RS,D,THETA,MU1,MU2,DX)

      INTEGER MAT
      DOUBLE PRECISION RS(2,6,6),THETA,PI
      COMPLEX*16 D,MU1(2),MU2(2),ZETA1,ZETA2,P1,P2,DX(2,4)

      PI=4.D0*DATAN(1.D0)

      ZETA1=DCOS(THETA*PI/180.D0)+MU1(MAT)*DSIN(THETA*PI/180.D0)
      ZETA2=DCOS(THETA*PI/180.D0)+MU2(MAT)*DSIN(THETA*PI/180.D0)

      P1=RS(MAT,1,1)*MU1(MAT)**2-RS(MAT,1,6)*MU1(MAT)+RS(MAT,1,2)
      P2=RS(MAT,1,1)*MU2(MAT)**2-RS(MAT,1,6)*MU2(MAT)+RS(MAT,1,2)

      DX(MAT,1)=P1*ZETA1**(D+1)/(D+1)
      DX(MAT,2)=P2*ZETA2**(D+1)/(D+1)
      DX(MAT,3)=CONJG(P1)*CONJG(ZETA1)**(D+1)/(D+1)
      DX(MAT,4)=CONJG(P2)*CONJG(ZETA2)**(D+1)/(D+1)

      RETURN
      END


      SUBROUTINE DISPY(MAT,RS,D,THETA,MU1,MU2,DY)

      INTEGER MAT
      DOUBLE PRECISION RS(2,6,6),THETA,PI
      COMPLEX*16 D,MU1(2),MU2(2),ZETA1,ZETA2,Q1,Q2,DY(2,4)

      PI=4.D0*DATAN(1.D0)

      ZETA1=DCOS(THETA*PI/180.D0)+MU1(MAT)*DSIN(THETA*PI/180.D0)
      ZETA2=DCOS(THETA*PI/180.D0)+MU2(MAT)*DSIN(THETA*PI/180.D0)

      Q1=RS(MAT,1,2)*MU1(MAT)-RS(MAT,2,6)+RS(MAT,2,2)/MU1(MAT)
      Q2=RS(MAT,1,2)*MU2(MAT)-RS(MAT,2,6)+RS(MAT,2,2)/MU2(MAT)

      DY(MAT,1)=Q1*ZETA1**(D+1)/(D+1)
      DY(MAT,2)=Q2*ZETA2**(D+1)/(D+1)
      DY(MAT,3)=CONJG(Q1)*CONJG(ZETA1)**(D+1)/(D+1)
      DY(MAT,4)=CONJG(Q2)*CONJG(ZETA2)**(D+1)/(D+1)

      RETURN
      END


      SUBROUTINE STROHEGN(RS,MU1,MU2,A,B,BI)

**********************************************************
*
```

```
*                   This subroutine calculates the matrices of Stroh eigenvectors, A and B,
*         and the inverse of B, BI.
*
*          Input: RS(i,j,k),MU1(i),MU2(i)
*
*          Output: A(i,j,k),B(i,j,k),BI(i,j,k)
*
*****************************************************************

          DOUBLE PRECISION RS(2,6,6)

*     P(MAT,MU1 or MU2)
*     K1(MAT)

          COMPLEX*16 MU1(2),MU2(2),A(2,2,2),B(2,2,2),P(2,2),Q(2,2)
          COMPLEX*16 K1(2),K2(2),BI(2,2,2)

*     Material 1
          P(1,1)=RS(1,1,1)*MU1(1)**2-RS(1,1,6)*MU1(1)+RS(1,1,2)
          P(1,2)=RS(1,1,1)*MU2(1)**2-RS(1,1,6)*MU2(1)+RS(1,1,2)

          Q(1,1)=RS(1,1,2)*MU1(1)-RS(1,2,6)+RS(1,2,2)/MU1(1)
          Q(1,2)=RS(1,1,2)*MU2(1)-RS(1,2,6)+RS(1,2,2)/MU2(1)

          K1(1)=CDSQRT(1/(2*(Q(1,1)-P(1,1)*MU1(1))))
          K2(1)=CDSQRT(1/(2*(Q(1,2)-P(1,2)*MU2(1))))

          A(1,1,1)=K1(1)*P(1,1)
          A(1,1,2)=K2(1)*P(1,2)
          A(1,2,1)=K1(1)*Q(1,1)
          A(1,2,2)=K2(1)*Q(1,2)

          B(1,1,1)=-K1(1)*MU1(1)
          B(1,1,2)=-K2(1)*MU2(1)
          B(1,2,1)=K1(1)
          B(1,2,2)=K2(1)

          BI(1,1,1)=-1/(K1(1)*(MU1(1)-MU2(1)))
          BI(1,1,2)=-MU2(1)/(K1(1)*(MU1(1)-MU2(1)))
          BI(1,2,1)=1/(K2(1)*(MU1(1)-MU2(1)))
          BI(1,2,2)=MU1(1)/(K2(1)*(MU1(1)-MU2(1)))


*     Material 2
          P(2,1)=RS(2,1,1)*MU1(2)**2-RS(2,1,6)*MU1(2)+RS(2,1,2)
          P(2,2)=RS(2,1,1)*MU2(2)**2-RS(2,1,6)*MU2(2)+RS(2,1,2)

          Q(2,1)=RS(2,1,2)*MU1(2)-RS(2,2,6)+RS(2,2,2)/MU1(2)
          Q(2,2)=RS(2,1,2)*MU2(2)-RS(2,2,6)+RS(2,2,2)/MU2(2)

          K1(2)=CDSQRT(1/(2*(Q(2,1)-P(2,1)*MU1(2))))
          K2(2)=CDSQRT(1/(2*(Q(2,2)-P(2,2)*MU2(2))))
```

```
      A(2,1,1)=K1(2)*P(1,1)
      A(2,1,2)=K2(2)*P(1,2)
      A(2,2,1)=K1(2)*Q(1,1)
      A(2,2,2)=K2(2)*Q(1,2)

      B(2,1,1)=-K1(2)*MU1(2)
      B(2,1,2)=-K2(2)*MU2(2)
      B(2,2,1)=K1(2)
      B(2,2,2)=K2(2)

      BI(2,1,1)=-1/(K1(2)*(MU1(2)-MU2(2)))
      BI(2,1,2)=-MU2(2)/(K1(2)*(MU1(2)-MU2(2)))
      BI(2,2,1)=1/(K2(2)*(MU1(2)-MU2(2)))
      BI(2,2,2)=MU1(2)/(K2(2)*(MU1(2)-MU2(2)))

      RETURN
      END


      SUBROUTINE EIGENVECTOR(B,BI,MU1,MU2,THETA,Z,Q,E)

**********************************************************************
*
*       This subroutine forms the complex potential function matrix, A, for the notch
*    surfaces.  This matrix is used to solve for the unknown complex constant eigenvector
*    in the Stroh field equations.  The potential function matrix is set equal to 0 following
*    the assumption that the notch surfaces are traction free.
*
*       Input: B(i,j,k),BI(i,j,k),MU1(i),MU2(i),THETA(i),Z(i)
*
*       Output: Q(i),E
*
**********************************************************************
      INTEGER I,J,K,S,E
      DOUBLE PRECISION THETA(2)
      COMPLEX*16 B(2,2,2),BI(2,2,2),A(4,4),Z(20),MU1(2)
      COMPLEX*16 A1(2,2),A2(2,2),A3(2,2),A4(2,2),MU2(2)
      COMPLEX*16 A11(3,3),A22(3,3),A33(3,3),A44(3,3),AR(4,3,3),X(3)
      COMPLEX*16 DET1,DET2,DET3,DET4,DET(4),MAXD,VS(4),AUG(3,4),Q(4)
      LOGICAL SINGUL

      CALL AMATRIX(1,B,BI,MU1,MU2,THETA(1),Z,A1,A2,A3,A4,E)
      CALL AMATRIX(2,B,BI,MU1,MU2,THETA(2),Z,A1,A2,A3,A4,E)

      DO 80 I=1,2
         DO 81 J=1,2
            A(I,J)=A1(I,J)
            A(I,J+2)=A2(I,J)
            A(I+2,J)=A3(I,J)
            A(I+2,J+2)=A4(I,J)
81       CONTINUE
```

```
80      CONTINUE

        CALL REDUCE(A,4,4,1,1,A11)
        CALL REDUCE(A,4,4,2,2,A22)
        CALL REDUCE(A,4,4,3,3,A33)
        CALL REDUCE(A,4,4,4,4,A44)

        DO 46 J=1,3
            DO 47 K=1,3
                AR(1,J,K)=A11(J,K)
                AR(2,J,K)=A22(J,K)
                AR(3,J,K)=A33(J,K)
                AR(4,J,K)=A44(J,K)
47          CONTINUE
46      CONTINUE

        CALL DETCALC3(A11,DET1)
        CALL DETCALC3(A22,DET2)
        CALL DETCALC3(A33,DET3)
        CALL DETCALC3(A44,DET4)

        DET(1)=DET1
        DET(2)=DET2
        DET(3)=DET3
        DET(4)=DET4

        MAXD=DET(1)
        DO 67 I=1,4
            IF (CABS(DET(I)).GT.CABS(MAXD))THEN
                MAXD=DET(I)
            END IF
67      CONTINUE

        DO 73 I=1,4
            IF (CABS(DET(I)).EQ.CABS(MAXD))THEN
                DO 68 J=1,3
                    DO 69 K=1,3
                        AUG(J,K)=AR(I,J,K)
                        S=I
69                  CONTINUE
68              CONTINUE
            END IF
73      CONTINUE

        DO 74 I=1,4
            VS(I)=A(I,S)
74      CONTINUE

        DO 75 I=S,3
            VS(I)=VS(I+1)
75      CONTINUE
```

```
      DO 76 I=1,3
            AUG(I,4)=-VS(I)
76    CONTINUE

      CALL GAUSS(AUG,3,4,3, X, SINGUL)

      DO 82 I=1,4
            Q(I)=X(I)
82    CONTINUE

*     Shifts vector values so that value of (1.0,0.0) can be assigned
*     to the value not solved for so that Q(4) is the full constant
*     vector in the field equations.

      IF(S.LT.4)THEN
            DO 77 I=S,3
                  Q(I+1)=X(I)
77          CONTINUE
            Q(S)=(1.0,0.0)      !Value of (1.0,0.0) is assigned to the value
      ELSE                      !not solved for
            DO 78 I=1,3         !Shifts values for the case of S=4
                  Q(I)=X(I)
78          CONTINUE
            Q(4)=(1.0,0.0)
      END IF

      RETURN
      END


      SUBROUTINE AMATRIX(MAT,B,BI,MU1,MU2,THETA,Z,A1,A2,A3,A4,E)

**********************************************************************
*
*           This subroutine calculates the Stroh stress potential functions:
*
*     A1: for material 1 and upper notch surface
*     A2: for complex conjugate of A1
*     A3: for material 2 and lower notch surface
*     A4: for complex conjugate of A3
*
*     Input: MAT,B(i,j,k),BI(i,j,k),MU1(i),MU2(j),THETA(i),Z
*
*     Output: A1(i,j),A2(i,j),A3(i,j),A4(i,j),E
*
**********************************************************************
      INTEGER MAT,I,J,E
      DOUBLE PRECISION THETA,PI
      COMPLEX*16 B(2,2,2),BI(2,2,2),ZD(2,2),BZD(2,2)
      COMPLEX*16 Z(20),BM(2,2),A1(2,2),A2(2,2),A3(2,2),A4(2,2)
      COMPLEX*16 BMC(2,2),BIM(2,2),BIMC(2,2),ZDC(2,2),BZDC(2,2)
      COMPLEX*16 ZETA1,ZETA2,MU1(2),MU2(2)
```

```
      PI=4.D0*DATAN(1.D0)

      ZETA1=DCOS(THETA*PI/180.D0)+MU1(MAT)*DSIN(THETA*PI/180.D0)
      ZETA2=DCOS(THETA*PI/180.D0)+MU2(MAT)*DSIN(THETA*PI/180.D0)

      E=4 !chooses eigenvalue 4 ********************************

      ZD(1,1)=ZETA1**(Z(E)+(1.0,0.0))
      ZD(1,2)=(0,0)
      ZD(2,1)=(0,0)
      ZD(2,2)=ZETA2**(Z(E)+(1.0,0.0))

      DO 70 I=1,2
            DO 71 J=1,2
                  BM(I,J)=B(MAT,I,J)
                  BMC(I,J)=CONJG(BM(I,J))
                  BIM(I,J)=BI(MAT,I,J)
                  BIMC(I,J)=CONJG(BIM(I,J))
                  ZDC(I,J)=CONJG(ZD(I,J))
71          CONTINUE
70    CONTINUE

      CALL MATMUL(BM,ZD,BZD,2,2,2,2)
      CALL MATMUL(BMC,ZDC,BZDC,2,2,2,2)

      IF(MAT.EQ.1)THEN
            CALL MATMUL(BZD,BIM,A1,2,2,2,2)
            CALL MATMUL(BZDC,BIMC,A2,2,2,2,2)
      ELSE
            CALL MATMUL(BZD,BIM,A3,2,2,2,2)
            CALL MATMUL(BZDC,BIMC,A4,2,2,2,2)
      END IF

      RETURN
      END

      SUBROUTINE REDUCE(A,NROW,NCOL,N,M,DR)

*****************************************************************
*
*     This subroutine reduces the given matrix, A(NROW,NCOL), by
*  eliminating the row N, and column M.  The reduced matix is DR(NROW-1,
*  NCOL-1).
*
*     Input: A(i,j),NROW,NCOL,N,M
*
*     Output: DR(i,j)
*
*****************************************************************

      INTEGER NROW,NCOL,N,M,I,J
```

```
      COMPLEX*16 A(NROW,NCOL),D(NROW,NCOL),DR(NROW-1,NCOL-1)

      DO 27 I=1,NROW
            DO 28 J=1,NCOL
                  D(I,J)=A(I,J)
28          CONTINUE
27    CONTINUE

      DO 35 J=1,NCOL
            DO 36 I=N,NROW-1
                  D(I,J)=D(I+1,J)
36          CONTINUE
35    CONTINUE

      DO 37 I=1,NROW-1
            DO 38 J=M,NCOL-1
                  D(I,J)=D(I,J+1)
38          CONTINUE
37    CONTINUE

      DO 42 I=1,NROW-1
            DO 43 J=1,NCOL-1
                  DR(I,J)=D(I,J)
43          CONTINUE
42    CONTINUE

      RETURN
      END



      SUBROUTINE GAUSS(AUG, LIM, LIMAUG, N, X, SINGUL)

**********************************************************
*        This subroutine finds the solution of the linear  system AX = B using Gaussian
*    elimination, providing a unique solution exists.  If the matrix is (nearly) singular,
*    SINGUL is returned as true, and the solution X is undefined.
*
* Ref.:  D. Johnstone, FORTRAN 77 for Engineers and Scientists: 3rd Ed., Macmillan
*          Publishing Company, 1992.
*
*    Definition of local variables:
*
*    I,J,K        Indices
*    MULT      Multiplier used to eliminate an unknown
*    ASBPIV    Absolute value of pivot element
*    PIVROW    Row containing pivot element
*    EPSIL     A small positive real value ("almost zero")
*    TEMP      Used to interchange rows of matrix
**********************************************************
```

```
      INTEGER I,J,K,N,PIVROW,LIM,LIMAUG
      COMPLEX*16 AUG(LIM,LIMAUG),X(LIM),TEMP,MULT
      DOUBLE PRECISION EPSIL,ABSPIV
      PARAMETER (EPSIL = 1E-6)
      LOGICAL SINGUL

      SINGUL = .FALSE.
      DO 50 I=1,N

*     locate pivot element
      ABSPIV=CABS(AUG(I,I))
      PIVROW=I
      DO 10 K=I+1,N
            IF(CABS(AUG(K,I)).GT.ABSPIV)THEN
                  ABSPIV=CABS(AUG(K,I))
                  PIVROW=K
            END IF
10    CONTINUE

*     Check if matrix is (nearly) singular
      IF(ABSPIV .LT. EPSIL)THEN
            SINGUL = .TRUE.
            RETURN
      END IF

*     It isn't, so interchange rows PIVROW and I if necessary
      IF(PIVROW.NE.I)THEN
              DO 20 J=1,N+1
                    TEMP=AUG(I,J)
                    AUG(I,J)=AUG(PIVROW,J)
                    AUG(PIVROW,J) = TEMP
20            CONTINUE
      END IF

*     Eliminate Ith unknown from equations I + 1, ..., N
      DO 40 J=I+1,N
            MULT=-AUG(J,I)/AUG(I,I)
            DO 30 K=I,N+1
                  AUG(J,K)=AUG(J,K)+MULT*AUG(I,K)
30          CONTINUE
40    CONTINUE

50    CONTINUE

*     Find the solutions by back substitution
      X(N)=AUG(N,N+1)/AUG(N,N)
      DO 70 J=N-1,1,-1
            X(J)=AUG(J,N+1)
            DO 60 K=J+1,N
                  X(J)=X(J)-AUG(J,K)*X(K)
60          CONTINUE
            X(J)=X(J)/AUG(J,J)
```

```
70      CONTINUE

        END


        SUBROUTINE INVERSEA(A,IA)

*****************************************************************
*
*       This subroutine calculates the inverse of the 4 x 4 matrix A, IA.
*
*****************************************************************

        COMPLEX*16 A(4,4),IA(4,4),C(4,4),CT(4,4),DET4

        CALL DETCALC4(A,DET4)
        CALL CMATRIX(A,C)
        CALL CTRANSP(C,CT)
        CALL INVERSE(DET4,CT,IA)

        RETURN
        END


        SUBROUTINE DETCALC4(A4,DET4)

*********************************************************************
*
*       This subroutine calculates the determinate of the 4x4 matrix A4, DET4.
*
*********************************************************************
        COMPLEX*16 A4(4,4),DET4

        DET4=A4(1,1)*A4(2,2)*A4(3,3)*A4(4,4)-A4(1,1)*A4(2,2)*A4(3,4)
     &     *A4(4,3)-A4(1,1)*A4(3,2)*A4(2,3)*A4(4,4)+A4(1,1)*A4(3,2)
     &     *A4(2,4)*A4(4,3)+A4(1,1)*A4(4,2)*A4(2,3)*A4(3,4)-A4(1,1)
     &     *A4(4,2)*A4(2,4)*A4(3,3)-A4(2,1)*A4(1,2)*A4(3,3)*A4(4,4)
     &     +A4(2,1)*A4(1,2)*A4(3,4)*A4(4,3)+A4(2,1)*A4(3,2)*A4(1,3)
     &     *A4(4,4)-A4(2,1)*A4(3,2)*A4(1,4)*A4(4,3)-A4(2,1)*A4(4,2)
     &     *A4(1,3)*A4(3,4)+A4(2,1)*A4(4,2)*A4(1,4)*A4(3,3)+A4(3,1)
     &     *A4(1,2)*A4(2,3)*A4(4,4)-A4(3,1)*A4(1,2)*A4(2,4)*A4(4,3)
     &     -A4(3,1)*A4(2,2)*A4(1,3)*A4(4,4)+A4(3,1)*A4(2,2)*A4(1,4)
     &     *A4(4,3)+A4(3,1)*A4(4,2)*A4(1,3)*A4(2,4)-A4(3,1)*A4(4,2)
     &     *A4(1,4)*A4(2,3)-A4(4,1)*A4(1,2)*A4(2,3)*A4(3,4)+A4(4,1)
     &     *A4(1,2)*A4(2,4)*A4(3,3)+A4(4,1)*A4(2,2)*A4(1,3)*A4(3,4)
     &     -A4(4,1)*A4(2,2)*A4(1,4)*A4(3,3)-A4(4,1)*A4(3,2)*A4(1,3)
     &     *A4(2,4)+A4(4,1)*A4(3,2)*A4(1,4)*A4(2,3)

        RETURN
        END
```

```
      SUBROUTINE DETCALC3(A3,DET3)

**********************************************************************
*
*     This subroutine calculates the determinant of the 3x3 matrix A3, DET3.
*
**********************************************************************
      COMPLEX*16 A3(3,3),DET3

      DET3=A3(1,1)*A3(2,2)*A3(3,3)-A3(1,1)*A3(2,3)*A3(3,2)-A3(2,1)
     &     *A3(1,2)*A3(3,3)+A3(2,1)*A3(1,3)*A3(3,2)+A3(3,1)*A3(1,2)
     &     *A3(2,3)-A3(3,1)*A3(1,3)*A3(2,2)

      RETURN
      END


      SUBROUTINE CMATRIX(A4,C)

**********************************************************************
*
*     This subroutine calculates the coefficent matrix C from A4 to be used in
*    finding the inverse of A4.
*
**********************************************************************
      INTEGER I,J,K,L,M,N,Q,R,S,T
      COMPLEX*16 A4(4,4),A34(3,4),A33(3,3),C(4,4),DET3

      DO 24 I=1,4
          DO 23 J=1,4
          IF(I.EQ.1)THEN
              DO 12 K=1,4
                  A34(1,K)=A4(2,K)
                  A34(2,K)=A4(3,K)
                  A34(3,K)=A4(4,K)
12            CONTINUE
          ELSE IF(I.EQ.2)THEN
              DO 13 L=1,4
                  A34(1,L)=A4(1,L)
                  A34(2,L)=A4(3,L)
                  A34(3,L)=A4(4,L)
13            CONTINUE
          ELSE IF(I.EQ.3)THEN
              DO 14 M=1,4
                  A34(1,M)=A4(1,M)
                  A34(2,M)=A4(2,M)
                  A34(3,M)=A4(4,M)
14            CONTINUE
          ELSE
              DO 15 N=1,4
                  A34(1,N)=A4(1,N)
                  A34(2,N)=A4(2,N)
```

```
                    A34(3,N)=A4(3,N)
15              CONTINUE
            END IF

            IF(J.EQ.1)THEN
                DO 18 Q=1,3
                    A33(Q,1)=A34(Q,2)
                    A33(Q,2)=A34(Q,3)
                    A33(Q,3)=A34(Q,4)
18              CONTINUE
            ELSE IF(J.EQ.2)THEN
                DO 19 R=1,3
                    A33(R,1)=A34(R,1)
                    A33(R,2)=A34(R,3)
                    A33(R,3)=A34(R,4)
19          CONTINUE
            ELSE IF(J.EQ.3)THEN
                DO 21 S=1,3
                    A33(S,1)=A34(S,1)
                    A33(S,2)=A34(S,2)
                    A33(S,3)=A34(S,4)
21              CONTINUE
            ELSE
                DO 22 T=1,3
                    A33(T,1)=A34(T,1)
                    A33(T,2)=A34(T,2)
                    A33(T,3)=A34(T,3)
22              CONTINUE
            END IF
                CALL DETCALC3(A33,DET3)
                C(I,J)=(-1)**(I+J)*DET3
23          CONTINUE
24      CONTINUE

        RETURN
        END


    SUBROUTINE CTRANSP(C,CT)

************************************************************
*
*       This subroutine calculates the transpose of the 4x4 matrix C, CT.
*
*       Input:  C
*       Output: CT
*
************************************************************
        INTEGER I,J
        COMPLEX*16 C(4,4),CT(4,4)

        DO 25 I=1,4
```

```
            DO 26 J=1,4
                CT(I,J)=C(J,I)
26          CONTINUE
25      CONTINUE

        RETURN
        END


        SUBROUTINE INVERSE(DET4,CT,IA)

*****************************************************************
*
*       This subroutine calculates the inverse of the 4x4 matrix A, IA, by dividing
*   the terms of the coefficent matrix CT by the determinant of A, DET4.
*
*   Inputs: CT,DET4
*
*   Output: IA
*
*****************************************************************
        INTEGER I,J
        COMPLEX*16 DET4,CT(4,4),IA(4,4)

        DO 44 I=1,4
            DO 45 J=1,4
                IA(I,J)=CT(I,J)/DET4
45          CONTINUE
44      CONTINUE

        RETURN
        END


        SUBROUTINE MATMUL(MAT1,MAT2,PROD,M,N,P,Q)

*****************************************************************
*       This subroutine finds the product of the matrices MAT1 and MAT2.
*
*  Ref.:  D. Johnstone, FORTRAN 77 for Engineers and Scientists: 3rd Ed., Macmillan
*           Publishing Company, 1992.
*
*       Inputs: MAT1,MAT2,M,N,P,Q
*
*       Output: PROD
*
*       Definition of local variables:
*
*       M       Number of rows in first matrix.
*       N       Number of columns in first matrix.
*       P       Number of rows in second matrix.
*       Q       Number of columns in second matrix.
```

```
*       MAT1   First matrix.
*       MAT2   Second matrix.
*       PROD   Product of MAT1 and MAT2.
*
****************************************************************

        INTEGER M,N,P,Q,I,J,K
        COMPLEX*16 MAT1(M,N),MAT2(P,Q),PROD(M,Q)
        COMPLEX*16 SUM

        IF (N.EQ.P) THEN
              DO 32 I=1,M
                    DO 33 J=1,Q
                          SUM=(0.D0,0.D0)
                          DO 34 K=1,N
                                SUM=SUM+MAT1(I,K)*MAT2(K,J)
34                              CONTINUE
                          PROD(I,J)=SUM
33                  CONTINUE
32            CONTINUE
        END IF

        RETURN
        END



        SUBROUTINE OUTPUT(E1,E2,E3,G23,G13,G12,V23,V13,V12,S,RS,MU1,
      &                           MU2,NUM,ROOT,Z,A,B,BI,Q,E)

*************************************************************************************
*
* This subroutine creates the output files sing.txt and Eigen.txt, which is read by the program MINTCALC.
*
*************************************************************************************

        INTEGER I,J,M,NUM,E
        DOUBLE PRECISION E1(2),E2(2),E3(2),G23(2),G13(2),G12(2),V23(2)
        DOUBLE PRECISION V13(2),V12(2),S(2,6,6),RS(2,6,6)
        COMPLEX*16 MU1(2),MU2(2),ROOT(20),Z(20),A(2,2,2),B(2,2,2),Q(4)
        COMPLEX*16 BI(2,2,2)

        OPEN(6,FILE='sing.txt',STATUS='UNKNOWN')

        WRITE(6,*)'MATERIAL PROPERTIES'
        DO 1 I=1,2
              WRITE(6,*)
              WRITE(6,*)'MATERIAL',I
              WRITE(6,2)E1(I),E2(I),E3(I),G23(I),G13(I),G12(I),V23(I),
      &                 V13(I),V12(I)
2       FORMAT(/,
```

```
     &   5X,'YOUNG MODULOUS IN X-DIRECTION [E1].....',F15.5/
     &   5X,'YOUNG MODULOUS IN Y-DIRECTION [E2].....',F15.5/
     &   5X,'YOUNG MODULOUS IN Z-DIRECTION [E3].....',F15.5/
     &   5X,'SHEAR MODULOUS [G23].....',F15.5/
     &   5X,'SHEAR MODULOUS [G13].....',F15.5/
     &   5X,'SHEAR MODULOUS [G12].....',F15.5/
     &   5X,'POISSON RATIO [V23].....',F15.5/
     &   5X,'POISSON RATIO [V13].....',F15.5/
     &   5X,'POISSON RATIO [V12].....',F15.5//)

1      CONTINUE

       DO 6 M=1,2
            IF (M.EQ.1)THEN
                WRITE(6,*)'THE MATRICIES OF COMPLIANCE CONSTANTS'
            END IF
                WRITE(6,*)
                WRITE(6,*)'MATERIAL',M
                WRITE(6,40)((S(M,J,K),K=1,6),J=1,6)
40          FORMAT(/,6F11.7)
6      CONTINUE

       DO 7 M=1,2
            IF(M.EQ.1)THEN
                WRITE(6,51)
51              FORMAT(//,'THE MATRICIES OF REDUCED COMPLIANCE
     &                    CONSTANTS')
            END IF
                WRITE(6,'(/,"MATERIAL",5X,I2)')M
                WRITE(6,40)((RS(M,I,J),J=1,6),I=1,6)
7      CONTINUE

       DO 8 M=1,2
            IF(M.EQ.1)THEN
                WRITE(6,59)
59              FORMAT(/,10X,'ROOTS OF CHARACTERISTIC EQUATIONS',//,
     &              'MATERIAL 1',/)
            ELSE
                WRITE(6,66)
66              FORMAT(/,'MATERIAL 2',/)
            END IF
                WRITE(6,61)MU1(M)
61          FORMAT('ROOT 1=',2X,'(',F9.6,',',F9.6,1X,')',/)
            WRITE(6,62)MU2(M)
62          FORMAT('ROOT 2=',2X,'(',F9.6,',',F9.6,1X,')',/)
8      CONTINUE

       DO 10 M=1,2
            IF (M.EQ.1)THEN
                WRITE(6,*)'STROH EIGENVECTORS'
            END IF
            WRITE(6,*)
```

```
            WRITE(6,*)'MATERIAL',M
            WRITE(6,'(/,3X,"A")')
            WRITE(6,41)((A(M,J,K),K=1,2),J=1,2)
41          FORMAT(/,2('(',F11.7,',',F11.7,')',2X),/)
            WRITE(6,'(/,3X,"B")')
            WRITE(6,41)((B(M,J,K),K=1,2),J=1,2)
10      CONTINUE


        WRITE(6,63)
63      FORMAT(//,'Initial Guesses and Calculated Eigenvalues',//)
        WRITE(6,64)
64      FORMAT('Initial Guesses',10X,'Calculated',/)
        DO 9 I=1,NUM
            WRITE(6,65)ROOT(I),Z(I)
65          FORMAT(2F9.6,10X,2F14.10,/)
9       CONTINUE

        WRITE(6,*)E

        WRITE(6,68)
68      FORMAT(//,'Eigenvector, Q = q1, q2, conj(q1), conj(q2)',//)
        DO 5 I=1,4
            WRITE(6,67)Q(I)
67      FORMAT(2F16.11,/)
5       CONTINUE

        CLOSE(6)

        OPEN(7,FILE='Eigen.txt',STATUS='UNKNOWN')

        DO 77 M=1,2
            WRITE(7,42)MU1(M)
            WRITE(7,42)MU2(M)
77      CONTINUE

        DO 78 M=1,2
            WRITE(7,42)((A(M,J,K),K=1,2),J=1,2)
42          FORMAT(2F15.10)
78      CONTINUE

        DO 79 M=1,2
            WRITE(7,42)((B(M,J,K),K=1,2),J=1,2)
79      CONTINUE

        DO 80 M=1,2
            WRITE(7,42)((BI(M,J,K),K=1,2),J=1,2)
80      CONTINUE

        WRITE(7,42)Z(E)

        WRITE(7,42)Q(1)
```

```
      WRITE(7,42)Q(2)

      CLOSE(7)

      RETURN
      END
```

Input file for SINGULARITY: guesses.txt

```
11
-1.4000  -.0100000
-1.5000  -.0100000
-1.6000  -.0100000
-1.7000  -.0100000
-1.8000  -.0100000
-.10000  -.0100000
-.20000  -.0100000
-.30000  -.0100000
-.40000  -.0100000
-.50000  -.0100000
-.60000  -.0100000
```

Output file for SINGULARITY: sing.txt

MATERIAL PROPERTIES

MATERIAL 1

YOUNG MODULOUS IN X-DIRECTION [E1]…….	8.90000
YOUNG MODULOUS IN Y-DIRECTION [E2]…….	138.00000
YOUNG MODULOUS IN Z-DIRECTION [E3]…….	8.90000
SHEAR MODULOUS [G23]……	5.17000
SHEAR MODULOUS [G13]……	2.89000
SHEAR MODULOUS [G12]……	5.17000
POISSON RATIO [V23]……...	.30000
POISSON RATIO [V13]……...	.54000
POISSON RATIO [V12]……...	.01935

MATERIAL 2

YOUNG MODULOUS IN X-DIRECTION [E1]…….	8.90000

YOUNG MODULOUS IN Y-DIRECTION [E2]…….	138.00000
YOUNG MODULOUS IN Z-DIRECTION [E3]…….	8.90000
SHEAR MODULOUS [G23]……...	5.17000
SHEAR MODULOUS [G13]……...	2.89000
SHEAR MODULOUS [G12]……...	5.17000
POISSON RATIO [V23]……..	.30000
POISSON RATIO [V13]……..	.54000
POISSON RATIO [V12]……..	.01935

THE MATRICIES OF COMPLIANCES CONSTANTS

MATERIAL 1

.1123596	-.0021739	-.0606742	.0000000	.0000000	.0000000
-.0021739	.0072464	-.0021739	.0000000	.0000000	.0000000
-.0606742	-.0021739	.1123596	.0000000	.0000000	.0000000
.0000000	.0000000	.0000000	.1934236	.0000000	.0000000
.0000000	.0000000	.0000000	.0000000	.3460207	.0000000
.0000000	.0000000	.0000000	.0000000	.0000000	.1934236

MATERIAL 2

.1123596	-.0021739	-.0606742	.0000000	.0000000	.0000000
-.0021739	.0072464	-.0021739	.0000000	.0000000	.0000000
-.0606742	-.0021739	.1123596	.0000000	.0000000	.0000000
.0000000	.0000000	.0000000	.1934236	.0000000	.0000000
.0000000	.0000000	.0000000	.0000000	.3460207	.0000000
.0000000	.0000000	.0000000	.0000000	.0000000	.1934236

THE MATRICIES OF REDUCED COMPLIANCE CONSTANTS

MATERIAL 1

.0795955	-.0033478	.0000000	.0000000	.0000000	.0000000
-.0033478	.0072043	.0000000	.0000000	.0000000	.0000000
.0000000	.0000000	.0000000	.0000000	.0000000	.0000000
.0000000	.0000000	.0000000	.1934236	.0000000	.0000000

.0000000 .0000000 .0000000 .0000000 .3460207 .0000000

.0000000 .0000000 .0000000 .0000000 .0000000 .1934236

MATERIAL 2

.0795955 -.0033478 .0000000 .0000000 .0000000 .0000000

-.0033478 .0072043 .0000000 .0000000 .0000000 .0000000

.0000000 .0000000 .0000000 .0000000 .0000000 .0000000

.0000000 .0000000 .0000000 .1934236 .0000000 .0000000

.0000000 .0000000 .0000000 .0000000 .3460207 .0000000

.0000000 .0000000 .0000000 .0000000 .0000000 .1934236

ROOTS OF CHARACTERISTIC EQUATIONS

MATERIAL 1

ROOT 1= (.000000, .198092)

ROOT 2= (.000000, 1.282004)

MATERIAL 2

ROOT 1= (.000000, .198092)

ROOT 2= (.000000, 1.282004)

STROH EIGENVECTORS

MATERIAL 1

A

(-.0174184, -.0174184) (-.1943344, .1943344)

(.0979326, -.0979326) (-.0087778, -.0087778)

B

(.5241506, -.5241506) (-1.3334223, -1.3334223)

(2.6460010, 2.6460010) (1.0401079, -1.0401079)

MATERIAL 2

A

(-.0174184, -.0174184) (-.1943344, .1943344)

(.0979326, -.0979326) (-.0087778, -.0087778)

B

(.5241506, -.5241506) (-1.3334223, -1.3334223)

(2.6460010, 2.6460010) (1.0401079, -1.0401079)

Initial Guesses and Calculated Eigenvalues

Initial Guesses		Calculated	
-1.400000	-.010000	-1.5089047329	.0000000000
-1.500000	-.010000	-1.5089047329	.0000000000
-1.600000	-.010000	-1.5089047329	.0000000000
-1.700000	-.010000	-1.7566214722	.0000000000
-1.800000	-.010000	-1.7566214722	.0000000000
-.100000	-.010000	.0000000000	.0000000000
-.200000	-.010000	-.2433785278	.0000000000
-.300000	-.010000	-.2433785278	.0000000000
-.400000	-.010000	-.4910952671	.0000000000
-.500000	-.010000	-.4910952671	.0000000000
-.600000	-.010000	-.4910952671	.0000000000

4

Eigenvector, Q = q1, q2, conj(q1), conj(q2)

1.00000000000 .00000000000

.00000000000 .1068478598

1.00000000000 .00000000000

0.00000000000 -.1068478598

Output file for SINGULARITY: Eigen.txt

.0000000000 .1980916185
.0000000000 1.2820039122
.0000000000 .1980916185
.0000000000 1.2820039122
-.0174184437 -.0174184437
-.1943344424 .1943344424
.0979326494 -.0979326494
-.0087778282 -.0087778282
-.0174184437 -.0174184437
-.1943344424 .1943344424
.0979326494 -.0979326494
-.0087778282 -.0087778282
.5241506200 -.5241506200
-1.3334223460 -1.3334223460
2.6460009972 2.6460009972
1.0401078603 -1.0401078603
.5241506200 -.5241506200
-1.3334223460 -1.3334223460
2.6460009972 2.6460009972
1.0401078603 -1.0401078603
-.1743355109 -.1743355109
.2234988070 -.2234988070
-.4435039415 .4435039415
-.0878544136 -.0878544136
-.1743355109 -.1743355109
.2234988070 -.2234988070
-.4435039415 .4435039415
-.0878544136 -.0878544136
-1.7566214722 .0000000000
1.0000000000 .0000000000
.0000000000 .1068478598

A.2 GAUSSPOINT

The FORTRAN program GAUSSPOINT generates ANSYS input files that are integration paths through the gaussian integration points with three points per element. These integration path files are read by ANSYS to interpolate the finite element solution data to the integration points.

```
*234567
      PROGRAM GAUSSPOINT
*
*          This program creates an ANSYS input file of an integration path through
*    the gaussian integration points from the given finite elements defining the corners of
*    the path.
*
*       Input:    File created by NODEE.txt containing the nodes defining the elements and
*                 the coordinates of the elements (for this case 135nodesAW2.txt).
*
*       Output:   ANSYS input file defining the integration path around the notch tip (for
*                 this case 135AW2path3.txt).  The ANSYS finite element solution data is
*                 interprelated to the path points in this file by the ANSYS program
*                 MIN.txt.  The interaction M-integral is numerically integrated through
*                 these path points.
*
*
*    Definition of Selected Variables:
*
*    NN              Number of nodes.
*    NE              Number of elements.
*    NODEY(N)        Node numbers with the nodes ordered first by Y then by X.
*    ELEM(N)         Element numbers.
*    N1(N)           The first node of element N.
*    N2(N)           The second node of element N.
*    N3(N)           The third node of element N.
*    N4(N)           The fourth node of element N.
*    N5(N)           The fifth node of element N.
*    N6(N)           The sixth node of element N.
*    N7(N)           The seventh node of element N.
*    N8(N)           The eighth node of element N.
*    E1              The first element used to define the path.
*    E2              The second element used to define the path.
*    E3              The third element used to define the path.
*    E4              The fourth element used to define the path.
*    XCYX(N)         The X coordinate of node N with the nodes ordered by Y then X.
*    YCYX(N)         The Y coordinate of node N with the nodes ordered by Y then X.
```

```
*    XCXY(N)        The X coordinate of node N with the nodes ordered by X then Y.
*    YCXY(N)        The Y coordinate of node N with the nodes ordered by X then Y.
*    GPX(N)         The X coordinate of gauss point N.
*    GPY(N)         The Y coordinate of gauss point N.
*    G1Y(N)         The Y coordinate of the first gauss point of path defining element N.
*    G1X(N)         The X coordinate of the first gauss point of path defining element N.
*    G2Y(N)         The Y coordinate of the second gauss pt. of path defining element N.
*    G2X(N)         The X coordinate of the second gauss pt. of path defining element N.
*    G3Y(N)         The Y coordinate of the third gauss pt. of path defining element N.
*    G3X(N)         The X coordinate of the third gauss pt. of path defining element N.
*
**********************************************************************

      INTEGER NN,NE,J,K,L

      PARAMETER(NN=9821,NE=3192)
      INTEGER NODEY(NN),ELEM(NE)
      INTEGER N5(NE),N6(NE),N7(NE),N8(NE),E1,E2,E3,E4
      DOUBLE PRECISION XC(NN),YC(NN)
      DOUBLE PRECISION GPX(1000),GPY(1000)
      DOUBLE PRECISION G1Y(4),G2Y(4),G3Y(4)
      DOUBLE PRECISION G1X(4),G2X(4),G3X(4)


* Reads the file containing the elemental and nodal data
      CALL READDATA(NN,NE,ELEM,N5,N6,N7,N8,NODEY,XC,YC)

* Finds the guassian integration points of the elements defining the path
      CALL CORNELEM(NN,NE,N5,N6,N7,N8,XC,YC,G1Y,G2Y,G3Y,G1X,G2X,
     &                    G3X,E1,E2,E3,E4)

* Finds the guassian integration points of the elements between the elements defining the path
      CALL GAUSSP(NN,NE,N5,N6,N7,N8,XC,YC,GPX,GPY,E1,E2,E3,E4,
     &                  J,K,L)

* Outputs the guassian integration points to an ANSYS input file
      CALL OUTPUT(J,K,L,G1Y,G2Y,G3Y,G1X,G2X,G3X,GPX,GPY)

      STOP
      END


      SUBROUTINE READDATA(NN,NE,ELEM,N5,N6,N7,N8,NODEY,XC,YC)

**********************************************************************
*
*           This subroutine reads the file with the element numbers, nodes of each element,
*     and coordinates of the nodes (135nodesAW2.txt).
*
*
*     Input: NN,NE
*
```

```
*       Output: ELEM,N5,N6,N7,N8,NODEY,XC,YC
*
**********************************************************************

      INTEGER NN,NE,N,I
      INTEGER NODEY(NN),ELEM(NE),N1(NE),N2(NE),N3(NE),N4(NE)
      INTEGER N5(NE),N6(NE),N7(NE),N8(NE)
      DOUBLE PRECISION XC(NN),YC(NN)

      OPEN(10,FILE='135nodesAW2.txt',STATUS='OLD')

      READ(10,1)ELEM(1),N1(1),N2(1),N3(1),N4(1),N5(1),N6(1),N7(1),N8(1)
1     FORMAT(12/,I10,20X,8I6)

      I=1
      DO 4 N=2,NE
          IF(I.EQ.20)THEN
              READ(10,2)ELEM(N),N1(N),N2(N),N3(N),N4(N),N5(N),
     &        N6(N),N7(N),N8(N)
2             FORMAT(///I10,20X,8I6)
              I=0
          ELSE
              READ(10,3)ELEM(N),N1(N),N2(N),N3(N),N4(N),N5(N),
     &                  N6(N),N7(N),N8(N)
3             FORMAT(I10,20X,8I6)
          END IF
          I=I+1
4     CONTINUE

      READ(10,5)NODEY(1),XC(NODEY(1)),YC(NODEY(1))
5     FORMAT(5/,I9,2D21.10)

      I=1
      DO 8 N=2,NN
          IF(I.EQ.20)THEN
              READ(10,6)NODEY(N),XC(NODEY(N)),YC(NODEY(N))
6             FORMAT(//I9,2D21.10)
              I=0
          ELSE
              READ(10,7)NODEY(N),XC(NODEY(N)),YC(NODEY(N))
7             FORMAT(I9,2D21.10)
          END IF
          I=I+1
8     CONTINUE

      RETURN
      END


      SUBROUTINE  CORNELEM(NN,NE,N5,N6,N7,N8,XC,YC,G1Y,G2Y,G3Y,
     &                     G1X,G2X,G3X,E1,E2,E3,E4)
```

```
**********************************************************************
*
*           This subroutine finds the guassian integration points of the elements
*         defining the path with 3 points per element.
*
*         Input: NN,NE,N5,N6,N7,N8,XC,YC
*
*         Output: G1Y,G2Y,G3Y,G1X,G2X,G3X,E1,E2,E3,E4
*
**********************************************************************
        INTEGER NN,NE,E1,E2,E3,E4
        INTEGER N5(NE),N6(NE),N7(NE),N8(NE)
        DOUBLE PRECISION XC(NN),YC(NN)
        DOUBLE PRECISION CY,G1Y(6),G2Y(6),G3Y(6)
        DOUBLE PRECISION G1X(6),G2X(6),G3X(6)

        E1=751  !elements defining the path
        E2=817
        E3=230
        E4=160 !*****************************


        CX=SQRT(3.0/5.0)*(XC(N6(E1))-XC(N8(E1)))/2.0

        G1Y(1)=YC(N6(E1))
        G1X(1)=XC(N8(E1))+(XC(N6(E1))-XC(N8(E1)))/2.0-CX

        G2Y(1)=G1Y(1)
        G2X(1)=XC(N8(E1))+(XC(N6(E1))-XC(N8(E1)))/2.0

        G3Y(1)=G1Y(1)
        G3X(1)=G2X(1)+CX


        CY=SQRT(3.0/5.0)*(YC(N7(E2))-YC(N5(E2)))/2.0
        CX=SQRT(3.0/5.0)*(XC(N6(E2))-XC(N8(E2)))/2.0

        G1Y(2)=YC(N6(E2))
        G1X(2)=XC(N7(E2))-CX

        G2Y(2)=YC(N6(E2))
        G2X(2)=XC(N5(E2))

        G3Y(2)=YC(N6(E2))+CY
        G3X(2)=XC(N5(E2))


        CY=SQRT(3.0/5.0)*(YC(N7(E3))-YC(N5(E3)))/2.0
        CX=SQRT(3.0/5.0)*(XC(N6(E3))-XC(N8(E3)))/2.0

        G1Y(3)=YC(N6(E3))-CY
        G1X(3)=XC(N5(E3))
```

```
      G2Y(3)=YC(N6(E3))
      G2X(3)=XC(N5(E3))

      G3Y(3)=YC(N6(E3))
      G3X(3)=XC(N5(E3))-CX


      CX=SQRT(3.0/5.0)*(XC(N6(E4))-XC(N8(E4)))/2.0

      G1Y(4)=YC(N6(E4))
      G1X(4)=XC(N8(E4))+(XC(N6(E4))-XC(N8(E4)))/2.0+CX

      G2Y(4)=G1Y(4)
      G2X(4)=XC(N8(E4))+(XC(N6(E4))-XC(N8(E4)))/2.0

      G3Y(4)=G1Y(4)
      G3X(4)=G2X(4)-CX

      RETURN
      END


      SUBROUTINE GAUSSP(NN,NE,N5,N6,N7,N8,XC,YC,GPX,GPY,E1,E2,E3,
     &                  E4,J,K,L)

**********************************************************************
*
*           This subroutine finds the guassian integration points with 3 points per element
*     for the elements between the path defining elements.
*
*     Input: NN,NE,N5,N6,N7,N8,XC,YC,E1,E2,E3,E4
*
*     Output: GPX,GPY,J,K,L
*
**********************************************************************
      INTEGER NN,NE,M,N,J,K,L,GPN(1000),E1,E2,E3,E4
      INTEGER N5(NE),N6(NE),N7(NE),N8(NE),NS1,NS2,BETA
      DOUBLE PRECISION XC(NN),YC(NN),CX,CY
      DOUBLE PRECISION GPX(1000),GPY(1000)


*     This loop finds the nodes on path segment 1

      J=0
      DO 21 N=1,NN
         IF((YC(N6(E1)).EQ.YC(N)).AND.(XC(N).GE.XC(N6(E1)))
     &      .AND.(XC(N).LE.XC(N8(E2)))) THEN
            J=J+1
            GPN(J)=N
         END IF
21    CONTINUE
```

```
      BETA=0
      DO WHILE(BETA.EQ.0)
            BETA=1
            DO 24 N=1,J-1
                  IF(XC(GPN(N+1)).LT.XC(GPN(N)))THEN
                        NS1=GPN(N)
                        NS2=GPN(N+1)
                        GPN(N)=NS2
                        GPN(N+1)=NS1
                        BETA=0
                  END IF
24          CONTINUE
      END DO

*     This loop calculates the X and Y coordinates of the gauss points
*     from the coordinates of the nodes on path segment 1.
      M=1
      DO 25 N=1,J-1
            CX=SQRT(3.0/5.0)*(XC(GPN(N))-XC(GPN(N+1)))/2
            GPY(M+1)=YC(GPN(N))
            GPX(M+1)=XC(GPN(N))-(XC(GPN(N))-XC(GPN(N+1)))/2

            GPY(M)=YC(GPN(N))
            GPX(M)=GPX(M+1)+CX

            GPY(M+2)=YC(GPN(N))
            GPX(M+2)=GPX(M+1)-CX

            M=M+3
25    CONTINUE


*     This loop finds the nodes on path segment 2
      K=0
      DO 30 N=1,NN
            IF((XC(N5(E2)).EQ.XC(N)).AND.(YC(N).GE.YC(N7(E2)))
     &          .AND.(YC(N).LE.YC(N5(E3)))) THEN
                  K=K+1
                  GPN(K)=N
            END IF
30    CONTINUE


      BETA=0
      DO WHILE(BETA.EQ.0)
            BETA=1
            DO 31 N=1,K-1
                  IF(YC(GPN(N+1)).LT.YC(GPN(N)))THEN
                        NS1=GPN(N)
                        NS2=GPN(N+1)
                        GPN(N)=NS2
```

```
                        GPN(N+1)=NS1
                        BETA=0
                    END IF
31              CONTINUE
            END DO


*       This loop calculates the X and Y coordinates of the gauss points
*       from the coordinates of the nodes on path segment 2.

            DO 32 N=1,K-1
                CY=SQRT(3.0/5.0)*(YC(GPN(N))-YC(GPN(N+1)))/2
                GPY(M+1)=YC(GPN(N))-(YC(GPN(N))-YC(GPN(N+1)))/2
                GPX(M+1)=XC(GPN(N))

                GPY(M)=GPY(M+1)+CY
                GPX(M)=XC(GPN(N))

                GPY(M+2)=GPY(M+1)-CY
                GPX(M+2)=XC(GPN(N))
                M=M+3
32          CONTINUE


*           This loop finds the nodes on path segment 3
            L=0
            DO 41 N=1,NN
                IF((YC(N6(E3)).EQ.YC(N)).AND.(XC(N).LE.XC(N8(E3)))
        &           .AND.(XC(N).GE.XC(N6(E4)))) THEN
                    L=L+1
                    GPN(L)=N
                END IF
41          CONTINUE

            BETA=0
            DO WHILE(BETA.EQ.0)
                BETA=1
                DO 42 N=1,L-1
                    IF(XC(GPN(N+1)).GT.XC(GPN(N)))THEN
                        NS1=GPN(N)
                        NS2=GPN(N+1)
                        GPN(N)=NS2
                        GPN(N+1)=NS1
                        BETA=0
                    END IF
42              CONTINUE
            END DO


*       This loop calculates the X and Y coordinates of the gauss points
*       from the coordinates of the nodes on path segment 3.
```

```
      DO 43 N=1,L-1
            CX=SQRT(3.0/5.0)*(XC(GPN(N))-XC(GPN(N+1)))/2
            GPY(M+1)=YC(GPN(N))
            GPX(M+1)=XC(GPN(N))-(XC(GPN(N))-XC(GPN(N+1)))/2

            GPY(M)=YC(GPN(N))
            GPX(M)=GPX(M+1)+CX

            GPY(M+2)=YC(GPN(N))
            GPX(M+2)=GPX(M+1)-CX
            M=M+3
43    CONTINUE

      RETURN
      END


      SUBROUTINE OUTPUT(J,K,L,G1Y,G2Y,G3Y,G1X,G2X,G3X,GPX,GPY)

**********************************************************************
*
*           This subroutine creates an ANSYS input file defining the integration path through the guassian
*  integration points with 3 points per element (For this case the path file is 135AW2path3.txt.).
*
*     Input: J,K,L,G1Y,G2Y,G3Y,G1X,G2X,G3X,GPX,GPY
*
*     Output: none
*
**********************************************************************
      INTEGER N,J,K,L,M,R
      DOUBLE PRECISION GPX(1000),GPY(1000)
      DOUBLE PRECISION G1Y(4),G2Y(4),G3Y(4)
      DOUBLE PRECISION G1X(4),G2X(4),G3X(4)

* file with coordinates of gauss points

      OPEN(20,FILE='135gauss3.txt',STATUS='UNKNOWN')

      WRITE(20,23)G1X(1),G1Y(1)
 23   FORMAT(2F10.5)
      WRITE(20,23)G2X(1),G2Y(1)
      WRITE(20,23)G3X(1),G3Y(1)

      M=1
      DO 22 N=1,J-1
            WRITE(20,23)GPX(M),GPY(M)
            WRITE(20,23)GPX(M+1),GPY(M+1)
            WRITE(20,23)GPX(M+2),GPY(M+2)
            M=M+3
 22   CONTINUE

      WRITE(20,23)G1X(2),G1Y(2)
```

```
      WRITE(20,23)G2X(2),G2Y(2)
      WRITE(20,23)G3X(2),G3Y(2)


      DO 26 N=1,K-1
            WRITE(20,23)GPX(M),GPY(M)
            WRITE(20,23)GPX(M+1),GPY(M+1)
            WRITE(20,23)GPX(M+2),GPY(M+2)
            M=M+3
26    CONTINUE


      WRITE(20,23)G1X(3),G1Y(3)
      WRITE(20,23)G2X(3),G2Y(3)
      WRITE(20,23)G3X(3),G3Y(3)

      DO 28 N=1, L-1
            WRITE(20,23)GPX(M),GPY(M)
            WRITE(20,23)GPX(M+1),GPY(M+1)
            WRITE(20,23)GPX(M+2),GPY(M+2)
            M=M+3
28    CONTINUE


      WRITE(20,23)G1X(4),G1Y(4)
      WRITE(20,23)G2X(4),G2Y(4)
      WRITE(20,23)G3X(4),G3Y(4)

      CLOSE(20)

* ANSYS input file for path through gaussian integration points.
      OPEN(30,FILE='135AW2path3.txt',STATUS='UNKNOWN')

      WRITE(30,*)'!MACRO PATH'
      WRITE(30,*)
      WRITE(30,*)'! This ANSYS macro creates integration path 3 around the notch tip.'
      WRITE(30,*)'! The ANSYS macro MIN is called to interpolate the ANSYS finite '
      WRITE(30,*)'! element solution data to each path point.'
      WRITE(30,*)
      WRITE(30,*)'! NOTCH ANGLE 135 degrees (alpha = 90 deg.)'
      WRITE(30,*)
      WRITE(30,*)'! NOTCH DEPTH, a/w = 0.2'
      WRITE(30,*)
      WRITE(30,*)'/POST1'
      WRITE(30,*)'SET,1'
      WRITE(30,*)
      WRITE(30,*)'PATH,Path3,*?*,,1'

      WRITE(30,31)G1X(1),G1Y(1)
31    FORMAT(1X'PPATH,  1 ,,',F11.6,',',F11.6,', 0.0')
      WRITE(30,32)G2X(1),G2Y(1)
32    FORMAT(1X'PPATH,  2 ,,',F11.6,',',F11.6,', 0.0')
```

```
      WRITE(30,34)G3X(1),G3Y(1)
34    FORMAT(1X'PPATH,  3 ,,',F11.6,',',F11.6,', 0.0')

      M=1
      R=1
      DO 35 N=1, J-1
            WRITE(30,36)R+3,GPX(M),GPY(M)
36          FORMAT(1X'PPATH,',I3,' ,,',F11.6,',',F11.6,', 0.0')
            WRITE(30,36)R+4,GPX(M+1),GPY(M+1)
            WRITE(30,36)R+5,GPX(M+2),GPY(M+2)
            M=M+3
            R=R+3
35    CONTINUE

      R=R+3
      WRITE(30,36)R,G1X(2),G1Y(2)
      WRITE(30,36)R+1,G2X(2),G2Y(2)
      WRITE(30,36)R+2,G3X(2),G3Y(2)


      R=R+3
      DO 37 N=1,K-1
            WRITE(30,36)R,GPX(M),GPY(M)
            WRITE(30,36)R+1,GPX(M+1),GPY(M+1)
            WRITE(30,36)R+2,GPX(M+2),GPY(M+2)
            M=M+3
            R=R+3
37    CONTINUE


      WRITE(30,36)R,G1X(3),G1Y(3)
      WRITE(30,36)R+1,G2X(3),G2Y(3)
      WRITE(30,36)R+2,G3X(3),G3Y(3)

      R=R+3
      DO 38 N=1,L-1
            WRITE(30,36)R,GPX(M),GPY(M)
            WRITE(30,36)R+1,GPX(M+1),GPY(M+1)
            WRITE(30,36)R+2,GPX(M+2),GPY(M+2)
            M=M+3
            R=R+3
38    CONTINUE

      WRITE(30,36)R,G1X(4),G1Y(4)
      WRITE(30,36)R+1,G2X(4),G2Y(4)
      WRITE(30,36)R+2,G3X(4),G3Y(4)

      WRITE(30,*)'*USE,MIN.txt !Calls the macro MIN.txt'
      CLOSE(30)

      RETURN
      END
```

Output file for GAUSSPOINT: 135AW2path3.txt

```
! MACRO PATH

! This ANSYS macro creates integration path 3 around the notch tip.  The ANSYS macro MIN is called to
! interpolate the ANSYS finite element solution data to each path point.

! NOTCH ANGLE 135 degrees (alpha = 90 deg.)

! NOTCH DEPTH, a/w = 0.2

/POST1
SET,1

PATH,Path3,138,,1
PPATH,  1 ,,  -3.191871,  -3.240179, 0.0
PPATH,  2 ,,  -3.025863,  -3.240179, 0.0
PPATH,  3 ,,  -2.859855,  -3.240179, 0.0
PPATH,  4 ,,  -2.763240,  -3.240179, 0.0
PPATH,  5 ,,  -2.597232,  -3.240179, 0.0
PPATH,  6 ,,  -2.431224,  -3.240179, 0.0
PPATH,  7 ,,  -2.334609,  -3.240179, 0.0
PPATH,  8 ,,  -2.168601,  -3.240179, 0.0
PPATH,  9 ,,  -2.002593,  -3.240179, 0.0
PPATH, 10 ,,  -1.905978,  -3.240179, 0.0
PPATH, 11 ,,  -1.739970,  -3.240179, 0.0
PPATH, 12 ,,  -1.573962,  -3.240179, 0.0
PPATH, 13 ,,  -1.477347,  -3.240179, 0.0
PPATH, 14 ,,  -1.311339,  -3.240179, 0.0
PPATH, 15 ,,  -1.145331,  -3.240179, 0.0
PPATH, 16 ,,  -1.048716,  -3.240179, 0.0
PPATH, 17 ,,   -.882708,  -3.240179, 0.0
PPATH, 18 ,,   -.716700,  -3.240179, 0.0
PPATH, 19 ,,   -.620085,  -3.240179, 0.0
PPATH, 20 ,,   -.454077,  -3.240179, 0.0
PPATH, 21 ,,   -.288069,  -3.240179, 0.0
PPATH, 22 ,,   -.191454,  -3.240179, 0.0
PPATH, 23 ,,   -.025446,  -3.240179, 0.0
PPATH, 24 ,,    .140562,  -3.240179, 0.0
PPATH, 25 ,,    .237176,  -3.240179, 0.0
PPATH, 26 ,,    .403185,  -3.240179, 0.0
PPATH, 27 ,,    .569193,  -3.240179, 0.0
PPATH, 28 ,,    .674781,  -3.240179, 0.0
PPATH, 29 ,,    .871625,  -3.240179, 0.0
PPATH, 30 ,,   1.068469,  -3.240179, 0.0
PPATH, 31 ,,   1.183031,  -3.240179, 0.0
PPATH, 32 ,,   1.379875,  -3.240179, 0.0
PPATH, 33 ,,   1.576719,  -3.240179, 0.0
PPATH, 34 ,,   1.691281,  -3.240179, 0.0
PPATH, 35 ,,   1.888125,  -3.240179, 0.0
```

PPATH, 36 ,, 1.888125, -3.095633, 0.0
PPATH, 37 ,, 1.888125, -3.011510, 0.0
PPATH, 38 ,, 1.888125, -2.866964, 0.0
PPATH, 39 ,, 1.888125, -2.722419, 0.0
PPATH, 40 ,, 1.888125, -2.638295, 0.0
PPATH, 41 ,, 1.888125, -2.493750, 0.0
PPATH, 42 ,, 1.888125, -2.349205, 0.0
PPATH, 43 ,, 1.888125, -2.265081, 0.0
PPATH, 44 ,, 1.888125, -2.120536, 0.0
PPATH, 45 ,, 1.888125, -1.975990, 0.0
PPATH, 46 ,, 1.888125, -1.891867, 0.0
PPATH, 47 ,, 1.888125, -1.747321, 0.0
PPATH, 48 ,, 1.888125, -1.602776, 0.0
PPATH, 49 ,, 1.888125, -1.518652, 0.0
PPATH, 50 ,, 1.888125, -1.374107, 0.0
PPATH, 51 ,, 1.888125, -1.229562, 0.0
PPATH, 52 ,, 1.888125, -1.165194, 0.0
PPATH, 53 ,, 1.888125, -1.088542, 0.0
PPATH, 54 ,, 1.888125, -1.011889, 0.0
PPATH, 55 ,, 1.888125, -.967278, 0.0
PPATH, 56 ,, 1.888125, -.890625, 0.0
PPATH, 57 ,, 1.888125, -.813972, 0.0
PPATH, 58 ,, 1.888125, -.769361, 0.0
PPATH, 59 ,, 1.888125, -.692708, 0.0
PPATH, 60 ,, 1.888125, -.616056, 0.0
PPATH, 61 ,, 1.888125, -.571444, 0.0
PPATH, 62 ,, 1.888125, -.494792, 0.0
PPATH, 63 ,, 1.888125, -.418139, 0.0
PPATH, 64 ,, 1.888125, -.373528, 0.0
PPATH, 65 ,, 1.888125, -.296875, 0.0
PPATH, 66 ,, 1.888125, -.220222, 0.0
PPATH, 67 ,, 1.888125, -.175611, 0.0
PPATH, 68 ,, 1.888125, -.098958, 0.0
PPATH, 69 ,, 1.888125, -.022306, 0.0
PPATH, 70 ,, 1.888125, .022306, 0.0
PPATH, 71 ,, 1.888125, .098958, 0.0
PPATH, 72 ,, 1.888125, .175611, 0.0
PPATH, 73 ,, 1.888125, .220222, 0.0
PPATH, 74 ,, 1.888125, .296875, 0.0
PPATH, 75 ,, 1.888125, .373528, 0.0
PPATH, 76 ,, 1.888125, .418139, 0.0
PPATH, 77 ,, 1.888125, .494792, 0.0
PPATH, 78 ,, 1.888125, .571444, 0.0
PPATH, 79 ,, 1.888125, .616056, 0.0
PPATH, 80 ,, 1.888125, .692708, 0.0
PPATH, 81 ,, 1.888125, .769361, 0.0
PPATH, 82 ,, 1.888125, .813972, 0.0
PPATH, 83 ,, 1.888125, .890625, 0.0
PPATH, 84 ,, 1.888125, .967278, 0.0
PPATH, 85 ,, 1.888125, 1.011889, 0.0
PPATH, 86 ,, 1.888125, 1.088542, 0.0
PPATH, 87 ,, 1.888125, 1.165194, 0.0

```
PPATH, 88 ,,   1.888125,   1.229562, 0.0
PPATH, 89 ,,   1.888125,   1.374107, 0.0
PPATH, 90 ,,   1.888125,   1.518652, 0.0
PPATH, 91 ,,   1.888125,   1.602776, 0.0
PPATH, 92 ,,   1.888125,   1.747321, 0.0
PPATH, 93 ,,   1.888125,   1.891867, 0.0
PPATH, 94 ,,   1.888125,   1.975990, 0.0
PPATH, 95 ,,   1.888125,   2.120536, 0.0
PPATH, 96 ,,   1.888125,   2.265081, 0.0
PPATH, 97 ,,   1.888125,   2.349205, 0.0
PPATH, 98 ,,   1.888125,   2.493750, 0.0
PPATH, 99 ,,   1.888125,   2.638295, 0.0
PPATH,100 ,,   1.888125,   2.722419, 0.0
PPATH,101 ,,   1.888125,   2.866964, 0.0
PPATH,102 ,,   1.888125,   3.011510, 0.0
PPATH,103 ,,   1.888125,   3.095633, 0.0
PPATH,104 ,,   1.888125,   3.240179, 0.0
PPATH,105 ,,   1.691281,   3.240179, 0.0
PPATH,106 ,,   1.576719,   3.240179, 0.0
PPATH,107 ,,   1.379875,   3.240179, 0.0
PPATH,108 ,,   1.183031,   3.240179, 0.0
PPATH,109 ,,   1.068469,   3.240179, 0.0
PPATH,110 ,,    .871625,   3.240179, 0.0
PPATH,111 ,,    .674781,   3.240179, 0.0
PPATH,112 ,,    .569193,   3.240179, 0.0
PPATH,113 ,,    .403185,   3.240179, 0.0
PPATH,114 ,,    .237176,   3.240179, 0.0
PPATH,115 ,,    .140562,   3.240179, 0.0
PPATH,116 ,,   -.025446,   3.240179, 0.0
PPATH,117 ,,   -.191454,   3.240179, 0.0
PPATH,118 ,,   -.288069,   3.240179, 0.0
PPATH,119 ,,   -.454077,   3.240179, 0.0
PPATH,120 ,,   -.620085,   3.240179, 0.0
PPATH,121 ,,   -.716700,   3.240179, 0.0
PPATH,122 ,,   -.882708,   3.240179, 0.0
PPATH,123 ,,  -1.048716,   3.240179, 0.0
PPATH,124 ,,  -1.145331,   3.240179, 0.0
PPATH,125 ,,  -1.311339,   3.240179, 0.0
PPATH,126 ,,  -1.477347,   3.240179, 0.0
PPATH,127 ,,  -1.573962,   3.240179, 0.0
PPATH,128 ,,  -1.739970,   3.240179, 0.0
PPATH,129 ,,  -1.905978,   3.240179, 0.0
PPATH,130 ,,  -2.002593,   3.240179, 0.0
PPATH,131 ,,  -2.168601,   3.240179, 0.0
PPATH,132 ,,  -2.334609,   3.240179, 0.0
PPATH,133 ,,  -2.431224,   3.240179, 0.0
PPATH,134 ,,  -2.597232,   3.240179, 0.0
PPATH,135 ,,  -2.763240,   3.240179, 0.0
PPATH,136 ,,  -2.859855,   3.240179, 0.0
PPATH,137 ,,  -3.025863,   3.240179, 0.0
PPATH,138 ,,  -3.191871,   3.240179, 0.0
*USE,MIN.txt !Calls the macro MIN.txt
```

A.3 MINTCALC

The FORTRAN program MINTCALC calculates the value of the interaction *M*-integral by numerically integrating the finite element solution data interpolated to the integration path.

```
      PROGRAM MINTCALC

*     This program calculates the value of the interaction M-integral by numerical integration of path
* data from a finite element solution.  (For this case the finite element data file is 135AAW22mint3.txt.)
*
*   NP          Number of data points on the path.
*   S           Distance along path to that point from the first path point.
*   DS          Change in the path length between consecutive points.
*   XC          X coordinate of data point.
*   YC          Y coordinate of data point.
*   NX          Component of unit outward normal vector in the X direction
*   NY          Component of unit outward normal vector in the Y direction
*   SX          Stress component in the X direction at data point
*   SY          Stress component in the Y direction at data point
*   SXY         Shear stress in the XY plane at data point
*   DUDX        Derivative of displacement in the X direction with respect to the X
*               direction.
*   DVDX        Derivative of displacement in the Y direction with respect
*               to the Y direction.
*   DUDY        Derivative of displacement in the X direction with respect
*               to the Y direction.
*   DVDY        Derivative of displacement in the Y direction with respect
*               to the Y direction.
*   TRAP        Value of equation being integrated from the Trapezoidal rule


      INTEGER NP,PD,MAT,I
      PARAMETER(NP=138)  !*********************************************
      DOUBLE PRECISION S(NP),XC(NP),YC(NP),NX(NP),NY(NP),SX(NP),
     &                 SXY(NP)
      DOUBLE PRECISION SY(NP),DUDX(NP),DVDX(NP),DUDY(NP),
     &                 DVDY(NP)
      DOUBLE PRECISION SXA1(NP),SXYA1(NP),SYA1(NP),DUDXA1(NP),
     &                 DUDYA1(NP)
      DOUBLE PRECISION DVDXA1(NP),DVDYA1(NP)
      DOUBLE PRECISION SXA2(NP),SXYA2(NP),SYA2(NP),DUDXA2(NP),
     &                 DUDYA2(NP)
      DOUBLE PRECISION DVDXA2(NP),DVDYA2(NP)
      COMPLEX*16 MU1(2),MU2(2),A(2,2,2),B(2,2,2),BI(2,2,2),D,Q(2)

      CALL READDATA(NP,S,XC,YC,NX,NY,SX,SXY,SY,DUDX,DVDX,DUDY,
```

```
     &                    DVDY,MU1,MU2,A,B,BI,D,Q)

       CALL AUXILIARY(NP,XC,YC,MU1,MU2,A,B,BI,D,Q,SXA1,SXYA1,
     &                SYA1,SXA2,SXYA2,SYA2,DUDXA1,DUDYA1,
     &                DVDXA1, DVDYA1,DUDXA2,DUDYA2,DVDXA2,
     &                DVDYA2)

       DO 10 I=1,4
          IF(I.EQ.1)THEN              !PATH SEGMENT 1
               PD=2        !PD=2 for horizontal path segment
               MAT=2       !MAT=2 for the material through this path segment
               NB=1        !NB is the first point on this path segment
               NE=34       !NE is the next to last point on this path segment
          ELSE IF(I.EQ.2)THEN         !PATH SEGMENT 4
               PD=2
               MAT=1       !MAT=1 for the material through this path segment
               NB=104
               NE=137
          ELSE IF(I.EQ.3)THEN         !PATH SEGMENT 2
               PD=1        !PD=1 for vertical path segment
               MAT=2
               NB=35
               NE=68
           ELSE IF(I.EQ.4)THEN         !PATH SEGMENT 3
               PD=1
               MAT=1
               NB=70
               NE=103
           END IF

       CALL INTEGRATE(NP,S,XC,YC,NX,NY,SX,SXY,SY,DUDX,DVDX,DUDY,
     &                DVDY,SXA1,SXYA1,SYA1,DUDXA1,DUDYA1,DVDXA1,
     &                DVDYA1,SXA2,SXYA2,SYA2,DUDXA2,DUDYA2,DVDXA2,
     &                DVDYA2,PD,NB,NE,MAT)

10     CONTINUE

       STOP
       END


       SUBROUTINE READDATA(NP,S,XC,YC,NX,NY,SX,SXY,SY,DUDX,
     &                           DVDX,DUDY,DVDY,MU1,MU2,A,B,BI,D,Q)
       INTEGER M,N,NP
       DOUBLE PRECISION S(NP),XC(NP),YC(NP),NX(NP),NY(NP),SX(NP),
     &                SXY(NP)
       DOUBLE PRECISION SY(NP),DUDX(NP),DVDX(NP),DUDY(NP),
     &                DVDY(NP)
       COMPLEX*16 MU1(2),MU2(2),A(2,2,2),B(2,2,2),BI(2,2,2),D,Q(2)

*      opens data file
       OPEN(10,FILE='135AAW22mint3.txt',STATUS='OLD')     !DATA FILE
```

```
      READ(10,1)S(1),XC(1),YC(1),NX(1),NY(1)
1     FORMAT(/////////////////,D16.9,4D15.9)

      DO 4 N=2,NP
            READ(10,2)S(N),XC(N),YC(N),NX(N),NY(N)
2           FORMAT(D16.9,4D15.9)
4     CONTINUE

      READ(10,3)S(1),SX(1),SY(1),SXY(1)
3     FORMAT(//,3X,4D15.9)

      DO 6 N=2,NP
            READ(10,7)S(N),SX(N),SY(N),SXY(N)
7           FORMAT(3X,4D15.9)
6     CONTINUE


      READ(10,8)S(1),DUDX(1),DVDX(1)
8     FORMAT(//,3X,3D15.9)

      DO 9 N=2,NP
            READ(10,11)S(N),DUDX(N),DVDX(N)
11          FORMAT(3X,3D15.9)
9     CONTINUE

      READ(10,12)S(1),DUDY(1),DVDY(1)
12    FORMAT(//,3X,3D15.9)

      DO 13 N=2,NP
            READ(10,14)S(N),DUDY(N),DVDY(N)
14          FORMAT(3X,3D15.9)
13    CONTINUE
      CLOSE(10)

      OPEN(7,FILE='Eigen.txt',STATUS='OLD')

      DO 77 M=1,2
            READ(7,42)MU1(M)
            READ(7,42)MU2(M)
77    CONTINUE

      DO 78 M=1,2
            READ(7,42)((A(M,J,K),K=1,2),J=1,2)
42          FORMAT(2F15.10)
78   CONTINUE

      DO 79 M=1,2
            READ(7,42)((B(M,J,K),K=1,2),J=1,2)
79    CONTINUE

      DO 80 M=1,2
```

```
            READ(7,42)((BI(M,J,K),K=1,2),J=1,2)
80      CONTINUE

        READ(7,42)D

        READ(7,42)Q(1)
        READ(7,42)Q(2)

        CLOSE(7)


        RETURN
        END


        SUBROUTINE AUXILIARY(NP,XC,YC,MU1,MU2,A,B,BI,D,Q,
       &                     SXA1,SXYA1,SYA1,SXA2,SXYA2,SYA2,
       &                     DUDXA1, DUDYA1,DVDXA1, DVDYA1,
       &                     DUDXA2 ,DUDYA2,DVDXA2,DVDYA2)

        INTEGER NP,I
        DOUBLE PRECISION XC(NP),YC(NP)

        DOUBLE PRECISION SXA1(NP)       ! X component of normal stress for
                                        ! auxiliary field for material 1

        DOUBLE PRECISION SYA1(NP)       ! Y component of normal stress for
                                        ! auxiliary field for material 1

        DOUBLE PRECISION SXYA1(NP)      ! Shear stress in X-Y plane for
                                        ! auxiliary field for material 1

        DOUBLE PRECISION SXA2(NP)       ! X component of normal stress for
                                        ! auxiliary field for material 2

        DOUBLE PRECISION SYA2(NP)       !Y component of normal stress for
                                        ! auxiliary field for material 2

        DOUBLE PRECISION SXYA2(NP)      ! Shear stress in X-Y plane for
                                        ! auxiliary field for material 2

        DOUBLE PRECISION DUDXA1(NP)     ! du/dx for auxiliary field for mat. 1
        DOUBLE PRECISION DUDYA1(NP)     ! du/dy for auxiliary field for mat. 1
        DOUBLE PRECISION DVDXA1(NP)     ! dv/dx for auxiliary field for mat. 1
        DOUBLE PRECISION DVDYA1(NP)     ! dv/dy for auxiliary field for mat. 1

        DOUBLE PRECISION DUDXA2(NP)     ! du/dx for auxiliary field for mat. 2
        DOUBLE PRECISION DUDYA2(NP)     ! du/dy for auxiliary field for mat. 2
        DOUBLE PRECISION DVDXA2(NP)     ! dv/dx for auxiliary field for mat. 2
        DOUBLE PRECISION DVDYA2(NP)     ! dv/dy for auxiliary field for mat. 2
```

```
      COMPLEX*16 MU1(2),MU2(2),A(2,2,2),B(2,2,2),BI(2,2,2),D,Q(2)
      COMPLEX*16 ZDX(2,2,2),ZDY(2,2,2)
      COMPLEX*16 AM1(2,2),AM2(2,2),BM1(2,2),BM2(2,2),BIM1(2,2),BIM2(2,2)
      COMPLEX*16 ZDXM1(2,2),ZDXM2(2,2),ZDYM1(2,2),ZDYM2(2,2)
      COMPLEX*16 AZ(2,2),AZBI(2,2)
      COMPLEX*16 UAXM1(2),UAXM2(2),UAYM1(2),UAYM2(2)
      COMPLEX*16 SAXM1(2),SAXM2(2),SAYM1(2),SAYM2(2)

      DO 20 I=1,2
         DO 30 J=1,2
            AM1(I,J)=A(1,I,J)
            BM1(I,J)=B(1,I,J)
            BIM1(I,J)=BI(1,I,J)
            AM2(I,J)=A(2,I,J)
            BM2(I,J)=B(2,I,J)
            BIM2(I,J)=BI(2,I,J)
30       CONTINUE
20    CONTINUE

* MATERIAL 2
      DO 10 I=1,NP/2
         CALL ZMATRIX(2,I,NP,XC,YC,MU1,MU2,D,ZDX,ZDY)
         DO 21 J=1,2
            DO 31 K=1,2
               ZDXM2(J,K)=ZDX(2,J,K)
               ZDYM2(J,K)=ZDY(2,J,K)
31          CONTINUE
21       CONTINUE

         CALL MATMUL(AM2,ZDXM2,AZ,2,2,2,2)
         CALL MATMUL(AZ,BIM2,AZBI,2,2,2,2)
         CALL MATMUL(AZBI,Q,UAXM2,2,2,2,1)

         DUDXA2(I)=REAL(UAXM2(1))
         DVDXA2(I)=REAL(UAXM2(2))

         CALL MATMUL(AM2,ZDYM2,AZ,2,2,2,2)
         CALL MATMUL(AZ,BIM2,AZBI,2,2,2,2)
         CALL MATMUL(AZBI,Q,UAYM2,2,2,2,1)

         DUDYA2(I)=REAL(UAYM2(1))
         DVDYA2(I)=REAL(UAYM2(2))


         CALL MATMUL(BM2,ZDXM2,AZ,2,2,2,2)
         CALL MATMUL(AZ,BIM2,AZBI,2,2,2,2)
         CALL MATMUL(AZBI,Q,SAXM2,2,2,2,1)

         SXYA2(I)=REAL(SAXM2(1))
         SYA2(I)=REAL(SAXM2(2))

         CALL MATMUL(BM2,ZDYM2,AZ,2,2,2,2)
```

```
                CALL MATMUL(AZ,BIM2,AZBI,2,2,2,2)
                CALL MATMUL(AZBI,Q,SAYM2,2,2,2,1)

                SXA2(I)=-REAL(SAYM2(1))
10      CONTINUE

* MATERIAL 1
        DO 11 I=NP/2+1, NP
                CALL ZMATRIX(1,I,NP,XC,YC,MU1,MU2,D,ZDX,ZDY)

                DO 22 J=1, 2
                        DO 32 K=1,2
                                ZDXM1(J,K)=ZDX(1,J,K)
                                ZDYM1(J,K)=ZDY(1,J,K)
32                      CONTINUE
22              CONTINUE

                CALL MATMUL(AM1,ZDXM1,AZ,2,2,2,2)
                CALL MATMUL(AZ,BIM1,AZBI,2,2,2,2)
                CALL MATMUL(AZBI,Q,UAXM1,2,2,2,1)

                DUDXA1(I)=REAL(UAXM1(1))
                DVDXA1(I)=REAL(UAXM1(2))

                CALL MATMUL(AM1,ZDYM1,AZ,2,2,2,2)
                CALL MATMUL(AZ,BIM1,AZBI,2,2,2,2)
                CALL MATMUL(AZBI,Q,UAYM1,2,2,2,1)

                DUDYA1(I)=REAL(UAYM1(1))
                DVDYA1(I)=REAL(UAYM1(2))

                CALL MATMUL(BM1,ZDXM1,AZ,2,2,2,2)
                CALL MATMUL(AZ,BIM1,AZBI,2,2,2,2)
                CALL MATMUL(AZBI,Q,SAXM1,2,2,2,1)

                SXYA1(I)=REAL(SAXM1(1))
                SYA1(I)=REAL(SAXM1(2))

                CALL MATMUL(BM1,ZDYM1,AZ,2,2,2,2)
                CALL MATMUL(AZ,BIM1,AZBI,2,2,2,2)
                CALL MATMUL(AZBI,Q,SAYM1,2,2,2,1)

                SXA1(I)=-REAL(SAYM1(1))
11      CONTINUE

        RETURN
        END


        SUBROUTINE ZMATRIX(M,I,NP,XC,YC,MU1,MU2,D,ZDX,ZDY)
        INTEGER M,I,NP
        DOUBLE PRECISION XC(NP),YC(NP)
```

```
      COMPLEX*16 MU1(2),MU2(2),D,ZDX(2,2,2),ZDY(2,2,2)
      COMPLEX*16 ZD1,ZD2

      ZD1=(D+(1.0,0.0))*(XC(I)+MU1(M)*YC(I))**D
      ZD2=(D+(1.0,0.0))*(XC(I)+MU2(M)*YC(I))**D

      ZDX(M,1,1)=ZD1
      ZDX(M,1,2)=(0,0)
      ZDX(M,2,1)=(0,0)
      ZDX(M,2,2)=ZD2

      ZDY(M,1,1)=MU1(M)*ZD1
      ZDY(M,1,2)=(0,0)
      ZDY(M,2,1)=(0,0)
      ZDY(M,2,2)=MU2(M)*ZD2

      RETURN
      END



      SUBROUTINE INTEGRATE(NP,S,XC,YC,NX,NY,SX,SXY,SY,DUDX,
     &                  DVDX, DUDY,DVDY,SXA1,SXYA1,SYA1,DUDXA1,
     &                  DUDYA1,DVDXA1, DVDYA1,SXA2,SXYA2,SYA2,
     &                  DUDXA2, DUDYA2,DVDXA2,DVDYA2,PD,NB,NE,MAT)

      INTEGER N,NP,NB,NE,MAT,FC,SC,PD
      DOUBLE PRECISION S(NP),DS(NP-1),XC(NP),YC(NP),NX(NP),NY(NP)
      DOUBLE PRECISION SX(NP),SXY(NP),SY(NP),MCOM
      DOUBLE PRECISION SXA1(NP),SXYA1(NP),SYA1(NP)
      DOUBLE PRECISION SXA2(NP),SXYA2(NP),SYA2(NP)
      DOUBLE PRECISION DUDX(NP),DVDX(NP),DUDY(NP),DVDY(NP)
      DOUBLE PRECISION DUDXA1(NP),DUDYA1(NP),DVDXA1(NP),
      DOUBLE PRECISION DVDYA1(NP),DUDXA2(NP)
      DOUBLE PRECISION,DUDYA2(NP),DVDXA2(NP),DVDYA2(NP)
      DOUBLE PRECISION MINTC1,MINTC2,MINTC3,MINTC4,MINT
      DOUBLE PRECISION TRAP

      DO 5 N=1,NP-1
            DS(N)=S(N+1)-S(N)
5     CONTINUE


      FC=29  !FIRST CORNER POINT   **********************************
      SC=62  !SECOND CORNER POINT          ****************************

      IF (PD.EQ.1)THEN
                              ! Normal components at corner points used
          NX(FC)=1.0          ! when calculating with data along a vertical
          NY(FC)=0.0          ! path segment
          NX(SC)=1.0
          NY(SC)=0.0
```

```
      ELSE

          NX(FC)=0.0          ! Normal components at corner points used
          NY(FC)=-1.0         ! when calculating with data along horizontal
          NX(SC)=0.0          ! path segment
          NY(SC)=1.0

      END IF


* Integrates first integral

      MINTC1=0.0
      IF (MAT.EQ.1) THEN
            DO 10 N=NB,NE,1
                  TRAP=(MCOM(N,NP,SXY,-SX,XC,YC,NX,NY,
     &                        DUDYA1,-DUDXA1)+MCOM(N+1,NP,SXY,
     &                        -SX,XC,YC,NX, NY,DUDYA1,-DUDXA1))/2.0
                  MINTC1=MINTC1+TRAP*DS(N)
10          CONTINUE
      END IF

      IF (MAT.EQ.2) THEN
            DO 11 N=NB,NE,1
                  TRAP=(MCOM(N,NP,SXY,-SX,XC,YC,NX,NY,DUDYA2,
     &                        -DUDXA2)+MCOM(N+1,NP,SXY,-SX,XC,YC,NX,NY,
     &                        DUDYA2,-DUDXA2))/2.0
                  MINTC1=MINTC1+TRAP*DS(N)
11          CONTINUE
      END IF


* Integrates second integral

      MINTC2=0.0
      IF (MAT.EQ.1) THEN
            DO 20 N=NB,NE,1
                  TRAP=(MCOM(N,NP,SY,-SXY,XC,YC,NX,NY,
     &                        DVDYA1,-DVDXA1)+MCOM(N+1,NP,SY,
     &                        -SXY,XC,YC,NX,NY, DVDYA1,-DVDXA1))/2.0
                  MINTC2=MINTC2+TRAP*DS(N)
20          CONTINUE
      END IF

      IF (MAT.EQ.2) THEN
            DO 21 N=NB,NE,1
                  TRAP=(MCOM(N,NP,SY,-SXY,XC,YC,NX,NY,DVDYA2,
     &                        -DVDXA2)+MCOM(N+1,NP,SY,-SXY,XC,YC,NX,NY,
     &                        DVDYA2,-DVDXA2))/2.0
                  MINTC2=MINTC2+TRAP*DS(N)
21          CONTINUE
```

```
      END IF

* Integrates third integral

      MINTC3=0.0
      IF (MAT.EQ.1) THEN
            DO 30 N=NB,NE,1
                  TRAP=(MCOM(N,NP,SXA1,SXYA1,NX,NY,XC,YC,
     &                        DUDX,DUDY)+MCOM(N+1,NP,SXA1,SXYA1,NX,NY,
     &                        XC,YC,DUDX,DUDY))/2.0
                  MINTC3=MINTC3-TRAP*DS(N)
30          CONTINUE
      END IF

      IF (MAT.EQ.2) THEN
          DO 31 N=NB,NE,1
            TRAP=(MCOM(N,NP,SXA2,SXYA2,NX,NY,XC,YC,
     &              DUDX,DUDY)+MCOM(N+1,NP,SXA2,SXYA2,NX,NY,XC,
     &              YC, DUDX,DUDY))/2.0
            MINTC3=MINTC3-TRAP*DS(N)
31        CONTINUE
      END IF

* Integrates fourth integral

      MINTC4=0.0
      IF (MAT.EQ.1) THEN
            DO 40 N=NB,NE,1
                  TRAP=(MCOM(N,NP,SXYA1,SYA1,NX,NY,XC,YC,DVDX,
     &                  DVDY)+MCOM(N+1,NP,SXYA1,SYA1,NX,NY,XC,YC,
     &                  DVDX,DVDY))/2.0
                  MINTC4=MINTC4-TRAP*DS(N)
40          CONTINUE
      END IF

      IF (MAT.EQ.2) THEN
            DO 41 N=NB,NE,1
                  TRAP=(MCOM(N,NP,SXYA2,SYA2,NX,NY,XC,YC,DVDX,
     &                        DVDY)+MCOM(N+1,NP,SXYA2,SYA2,NX,NY,XC,YC,
     &                        DVDX,DVDY))/2.0
                  MINTC4=MINTC4-TRAP*DS(N)
41          CONTINUE
      END IF

* The interaction M-integral value is the sum of all four integrals MINT
      MINT=MINTC1+MINTC2+MINTC3+MINTC4

      OPEN(20,FILE='mintOUT.txt',STATUS='UNKNOWN')
      WRITE(20,*)MINT

      RETURN
      END
```

```
* Calculates the value of the equation being integrated at the given point
      FUNCTION MCOM(N,NP,SC1,SC2,XN1,XN2,XN3,XN4,DU1,DU2)
      INTEGER N,NP
      DOUBLE PRECISION SC1(NP),SC2(NP),XN1(NP),XN2(NP),XN3(NP),XN4(NP)
      DOUBLE PRECISION DU1(NP),DU2(NP),MCOM

      MCOM=(SC1(N)*XN1(N)+SC2(N)*XN2(N))*(DU1(N)*XN3(N)+DU2(N)*
     &          XN4(N))

      RETURN
      END

* Multiplies maticies MAT1 and MAT2
**********************************************************************
      SUBROUTINE MATMUL(MAT1,MAT2,PROD,M,N,P,Q)
*
*  Ref.:  D. Johnstone, FORTRAN 77 for Engineers and Scientists: 3rd Ed., Macmillan
*           Publishing Company, 1992.
*
*    Definition of Local Variables:
*
*    M          Number of rows in first matrix
*    N          Number of columns in first matrix
*    P          Number of rows in second matrix
*    Q          Number of columns in second matrix
*    MAT1       First matrix
*    MAT2       Second matrix
*    PROD       Product of MAT1 and MAT2
*
**********************************************************************

      INTEGER M,N,P,Q,I,J,K
      COMPLEX*16 MAT1(M,N),MAT2(P,Q),PROD(M,Q)
      COMPLEX*16 SUM

      IF (N.EQ.P) THEN
            DO 32 I=1,M
                  DO 33 J=1,Q
                        SUM=(0.D0,0.D0)
                        DO 34 K=1,N
                              SUM=SUM+MAT1(I,K)*MAT2(K,J)
34                      CONTINUE
                        PROD(I,J)=SUM
33                CONTINUE
32          CONTINUE
      END IF

      RETURN
      END
```

B. ANSYS Programs

The following ANSYS programs are used to create finite element models of the double V-notched plates, generate element and node lists for these models, and process the finite element solution data.

B.1 FINITE ELEMENT MODEL

The ANSYS program FINITE ELEMENT MODEL generates the finite element model for a double V-notched plate with a notch angle of $\theta = 135^{o}$ ($\alpha = 90^{o}$)and notch depth, *a/w*, of 0.2 with the 0^{o} fiber orientation and loading for the anti-symmetric deformation mode.

```
! FINITE ELEMENT MODEL
!
! Generates finite element model for double V-notched specimen with loading applied
! for anti-symmetric deformation mode.

/PREP7
/TITLE, NOTCH ANGLE 135 deg.  A/W=0.2

W=4.75          ! quarter of width (mm)
HM=38.0
H=.3*HM         ! Vertical distance from notch tip to the intersection of the notch face
                ! and the edge.
TW=.13*W        ! Horizontal distance from notch tip to where element sides
                ! become vertical.

PI=4.0*ATAN(1.0)
THETA=135*PI/180 ! NOTCH ANGLE

! a/w=.2

! Material properties

E1=8.9 ! (GPa)
E2=138.0
E3=8.9
```

```
G23=5.17
G13=2.89
G12=5.17  ! (GPa)
NU23=.30
NU13=.54
NU12=.0193478   ! NU21=.30

! Keypoints

K,1,0,0
K,2,TW,0
K,3,1.2*W,0
K,4,-W/4,(W/4)/TAN(THETA-PI/2)
K,5,TW,(W/4)/TAN(THETA-PI/2)
K,6,1.2*W,(W/4)/TAN(THETA-PI/2)
K,7,-0.8*W,0.8*W/TAN(THETA-PI/2)
K,8,TW,0.8*W/TAN(THETA-PI/2)
K,9,1.2*W,0.8*W/TAN(THETA-PI/2)
K,10,-0.8*W,H
K,11,TW,H
K,12,1.2*W,H

K,13,1.2*W,-H
K,14,TW,-H
K,15,-0.8*W,-H
K,16,1.2*W,-0.8*W/TAN(THETA-PI/2)
K,17,TW,-0.8*W/TAN(THETA-PI/2)
K,18,-0.8*W,-0.8*W/TAN(THETA-PI/2)
K,19,1.2*W,-(W/4)/TAN(THETA-PI/2)
K,20,TW,-(W/4)/TAN(THETA-PI/2)
K,21,-W/4,-(W/4)/TAN(THETA-PI/2)

SN=(2*(1.2*W))

K,22,SN,0
K,23,SN-TW,0
K,24,SN+W/4,(W/4)/TAN(THETA-PI/2)
K,25,SN-TW,(W/4)/TAN(THETA-PI/2)
K,26,SN+0.8*W,0.8*W/TAN(THETA-PI/2)
K,27,SN-TW,0.8*W/TAN(THETA-PI/2)
K,28,SN+0.8*W,H
K,29,SN-TW,H
K,30,SN-TW,-H
K,31,SN+0.8*W,-H
K,32,SN-TW,-0.8*W/TAN(THETA-PI/2)
K,33,SN+0.8*W,-0.8*W/TAN(THETA-PI/2)
K,34,SN-TW,-(W/4)/TAN(THETA-PI/2)
K,35,SN+W/4,-(W/4)/TAN(THETA-PI/2)

K,36,-0.8*W,HM
K,37,TW,HM
K,38,1.2*W,HM
```

```
K,39,SN-TW,HM
K,40,SN+.8*W,HM
K,41,-0.8*W,-HM
K,42,TW,-HM
K,43,1.2*W,-HM
K,44,SN-TW,-HM
K,45,SN+0.8*W,-HM


! Lines connecting keypoints

L,1,2,9  !1
L,2,3,10  !2
L,4,5,9  !3
L,5,6,10  !4
L,7,8,9  !5
L,8,9,10  !6
L,10,11,9 !7
L,11,12,10 !8
L,1,4,6   !9
L,2,5,6   !10
L,3,6,6   !11
L,4,7,7   !12
L,5,8,7   !13
L,6,9,7   !14
L,7,10,20  !15
L,8,11,20  !16
L,9,12,20  !17

L,21,20,9  !18
L,20,19,10 !19
L,18,17,9  !20
L,17,16,10 !21
L,15,14,9  !22
L,14,13,10 !23
L,1,21,6   !24
L,2,20,6   !25
L,3,19,6   !26
L,21,18,7  !27
L,20,17,7  !28
L,19,16,7  !29
L,18,15,20  !30
L,17,14,20  !31
L,16,13,20  !32

L,12,29,10  !33
L,29,28,9   !34
L,9,27,10   !35
L,27,26,9   !36
L,6,25,10   !37
L,25,24,9   !38
L,3,23,10   !39
```

```
L,23,22,9   !40
L,19,34,10  !41
L,34,35,9   !42
L,16,32,10  !43
L,32,33,9   !44
L,13,30,10  !45
L,30,31,9   !46
L,27,29,20  !47
L,26,28,20  !48
L,25,27,7   !49
L,24,26,7   !50
L,23,25,6   !51
L,22,24,6   !52
L,34,23,6   !53
L,35,22,6   !54
L,32,34,7   !55
L,33,35,7   !56
L,30,32,20  !57
L,31,33,20  !58
L,36,37,9   !59
L,37,38,10  !60
L,38,39,10  !61
L,39,40,9   !62
L,10,36,9   !63
L,11,37,9   !64
L,12,38,9   !65
L,29,39,9   !66
L,28,40,9   !67
L,41,42,9   !68
L,42,43,10  !69
L,43,44,10  !70
L,44,45,9   !71
L,41,15,9   !72
L,42,14,9   !73
L,43,13,9   !74
L,44,30,9   !75
L,45,31,9   !76


!Areas enclosed by lines

AL,1,10,3,9  !1
AL,2,11,4,10 !2
AL,3,13,5,12 !3
AL,4,14,6,13 !4
AL,5,16,7,15 !5
AL,6,17,8,16 !6

AL,18,25,1,24 !7
AL,19,26,2,25 !8
AL,20,28,18,27 !9
AL,21,29,19,28 !10
```

```
AL,22,31,20,30 !11
AL,23,32,21,31 !12

AL,35,47,33,17  !13
AL,36,48,34,47  !14
AL,37,49,35,14  !15
AL,38,50,36,49  !16
AL,39,51,37,11  !17
AL,40,52,38,51  !18
AL,41,53,39,26  !19
AL,42,54,40,53  !20
AL,43,55,41,29  !21
AL,44,56,42,55  !22
AL,45,57,43,32  !23
AL,46,58,44,57  !24

AL,7,64,59,63   !25
AL,8,65,60,64   !26
AL,33,66,61,65  !27
AL,34,67,62,66  !28
AL,68,73,22,72  !29
AL,69,74,23,73  !30
AL,70,75,45,74  !31
AL,71,76,46,75  !32


ET,1,PLANE82,,,3  ! plane stress
R,1,1.0  !thickness of 1.0 mm  for plane stress

MP,EX,1,E1
MP,EY,1,E2
MP,EZ,1,E3
MP,GXY,1,G12
MP,GYZ,1,G23
MP,GXZ,1,G13
MP,PRXY,1,NU12
MP,PRYZ,1,NU23
MP,PRXZ,1,NU13

AMESH,1,32   ! Meshes areas
WSORT,Y

! Displacement boundary conditions.

D,1,ALL,0

NSEL,S,NODE,,1,19
NSEL,A,NODE,,194,213
NSEL,A,NODE,,5511,5530
NSEL,A,NODE,,5691,5708
D,ALL,UY,0
NSEL,ALL
```

```
FINISH

/SOLU

! Point force loading according to the Iosipescu shear test for the anti-symmetric
! deformation mode.

F,8551,FX,-.482
F,847,FX,1.482
F,7194,FX,-1.482
F,8815,FX,.482
```

B.2 NODEE

The ANSYS program NODEE creates a file called nodec.i, which lists the nodes that form each element and the coordinates of each node. The file nodec.i is read by the FORTRAN program GAUSSPOINT to generate the integration paths.

```
! MACRO NODEE
! Outputs the element and node lists to nodec.i
!
!       This ANSYS program creates a file called nodec.i, which lists the nodes that form
! each element and the coordinates of each node.  The first node list is ordered first by the
! y coordinate and then by the x coordinate.  The second node list is ordered first by the x
! coordinate and then by the y coordinate.
!
!       The file nodec.i is read by the FORTRAN program GAUSSPOINT to create the
! paths through the gaussian integration points.

ESEL,ALL
NSEL,ALL

/OUTPUT,nodec,i
/FORMAT,7,F,15,9
ELIST,ALL
NLIST,ALL,,,COORD,Y,X
NLIST,ALL,,,COORD,X,Y
/OUTPUT
```

B.3 MIN

The ANSYS program MIN interpolates the stress components, normal vectors, and displacement derivatives along the path and outputs them to mint.i, which is read by the FORTRAN program MINTCALC to numerically integrate the interaction *M*-integral. PATH, which is the output file from GAUSSPOINT, calls the ANSYS program MIN for each of the integration path points.

```
! MACRO MIN
!
!          This ANSYS program interprelates the stresses, normal vectors, and displacement
!  derivatives along the path and outputs them to mint.i, which is read by the FORTRAN
!  program MINTCALC to numerically integrate the interaction M-integral.
!
!  Ref.:  J. S. Solecki, Fracture Mechanics, Swanson Analysis Systems, Inc., 1989.
!
! Definition of Selected Variables:
!
! C1        The derivative of the displacement in the x1 direction
!           with respect to the x1 direction (du1/dx1).
!
! C2        The derivative of the displacement in the x2 direction
!           with respect to the x1 direction (du2/dx1).
!
! C3        The derivative of the displacement in the x1 direction
!           with respect to the x2 direction (du1/dx2).
!
! C4        The derivative of the displacement in the x2 direction
!           with respect to the x2 direction (du2/dx2).
!
! NX        The component of the unit outward normal vector in the x or x1 direction.
! NY        The component of the unit outward normal vector in the y or x2 direction.
! SX        The normal stress in the x or x1 direction.
! SY        The normal stress in the y or x2 direction.
! SXY       The shear stress in the x-y or x1-x2 plane.
! XG        The x or x1 coordinate of the path point (origin at notch tip).
! YG        The y or x2 coordinate of the path point (origin at notch tip).


PDEF,SX,S,X
PDEF,SY,S,Y
PDEF,SXY,S,XY
PVECT,NORM,NX,NY,NZ
```

```
*GET,DX,PLAST,S  ! calculation of du1/dx1 and du2/dx1
DX=DX/1000
PCALC,ADD,XG,XG,,,,-DX/2
PDEF,UX1,U,X
PDEF,UY1,U,Y
PCALC,ADD,XG,XG,,,,DX
PDEF,UX2,U,X
PDEF,UY2,U,Y
PCALC,ADD,XG,XG,,,,-DX/2
C=(1/DX)
PCALC,ADD,C1,UX2,UX1,C,-C   ! C1=du1/dx1
PCALC,ADD,C2,UY2,UY1,C,-C   ! C2=du2/dx1

*GET,DY,PLAST,S  ! calculation of du1/dx2 and du2/dx2
DY=DY/1000
PCALC,ADD,YG,YG,,,,-DY/2
PDEF,UX1,U,X
PDEF,UY1,U,Y
PCALC,ADD,YG,YG,,,,DY
PDEF,UX2,U,X
PDEF,UY2,U,Y
PCALC,ADD,YG,YG,,,,-DY/2
C=(1/DY)
PCALC,ADD,C3,UX2,UX1,C,-C  ! C3=du1/dx2
PCALC,ADD,C4,UY2,UY1,C,-C  ! C4=du2/dx2

/OUTPUT,mint,i
/FORMAT,7,F,15,9
/HEADER,OFF,OFF,OFF,OFF,OFF,ON
PRPATH,XG,YG,NX,NY
PRPATH,SX,SY,SXY
PRPATH,C1,C2
PRPATH,C3,C4
/OUTPUT
```