STRESS COMPATIBLE FINITE ELEMENTS FOR BIMATERIAL INTERFACE PROBLEMS

by Michael Angelides, B. Eng.

Department of Civil Engineering and Applied Mechanics

McGill University, Montreal

July 1987

A Thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Engineering.

ISBN 0-315-44363-4

STRESS COMPATIBLE FINITE ELEMENTS

To my wife *Marika*

Abstract

A new finite element has been developed satisfying the required continuity of the stress vector at bimaterial interface points in order to alleviate the problem of high stress discontinuity predictions by the conventional displacement finite element method. The formulation of the element has been carried out within the framework of the displacement method assuming perfect bond conditions at the interface and isotropic, linear elastic materials. Two general finite element programs have been developed incorporating the interface element, one for two-dimensional plane-stress/plane-strain and axisymmetric analyses and one for three-dimensional analyses. A series of validation tests have been carried out to assess the correctness of the stress distribution obtained by the new element at interfaces of highly dissimilar materials. The results of the tests are compared to analytical solutions and to results from analyses performed by the conventional displacement method. Overall, the proposed element has been demonstrated to have a very satisfactory degree of reliability, especially in view of the observed inability of the conventional method to yield interpretable interface stress values for most cases analysed. Finally, the new element has been applied to the analysis of an axisymmetric model of the knee tibial implant. The superiority of the interface element over the conventional method has been demonstrated in this case by a convergence study.

Résumé

Un nouvel élément fini a été développé satisfaisant la continuité requise du vecteur contrainte pour des points donnés sur une interface bimatérielle afin d'éliminer le problème de la discontinuité des contraintes calculées par la méthode conventionnelle des déplacements. La formulation de l'élément a été basée sur la méthode des déplacements et suppose des matériaux isotropes, linéaires et élastiques et des conditions d'adhésion parfaite à l'interface. Deux programmes généraux d'éléments finis ont été développés incorporant l'élément d'interface: l'un pour des analyses de problèmes bidimensionnels (contrainte plane, déformation plane et symétrie axiale); l'autre pour des analyses de problèmes tridimensionnels. Plusieurs problèmes de validation ont été analysés pour évaluer la performance de l'élément nouveau à des interfaces de matériaux très différents. Les résultats des essais sont comparés avec des solutions théoriques et avec des résultats provenant de la méthode conventionnelle des déplacements. L'élément proposé a été démontré très satisfaisant, surtout en comparaison avec l'impuissance observée de la méthode conventionnelle de produire des contraintes d'interface interprétables pour la plupart des cas analysés. Finalement, le nouvel élément a été utilisé pour l'analyse d'un modèle axisymétrique de la prothèse tibiale du genou. La superiorité de l'élément d'interface vis-à-vis de la méthode conventionnelle a été demontrée dans ce cas par une étude de convergence.

Acknowledgements

I would like to take this opportunity to express my gratitude to a number of people who contributed either directly or indirectly to the completion of this work. Professor S.C. Shrivastava (Department of Civil Engineering and Applied Mechanics, McGill University) has provided me with invaluable guidance in the course of the present study and his assistance is deeply appreciated. I am also indebted to Dr. A. Shirazi-Adl (Department of Mechanical Engineering, Ecole Polytechnique) for constructive discussions, advice and encouragement. Professor A.M. Ahmed (Department of Mechanical Engineering, McGill University) has introduced me to the very exciting field of Orthopaedic Biomechanics and I am thankful for his advice and his comments concerning the biomechanical aspect of this work. I would also like to thank Professor J.W. Provan (Department of Mechanical Engineering, McGill University) for his friendly help and interest in my work.

I am deeply indebted to Mrs. N. Procyshyn for her painstaking efforts in the preparation of the printed form of this thesis. The majority of the figures included have been prepared by Mr. S. Shi and Miss H. Eli and I am thankful to them.

The financial assistance provided by the Medical Research Council of Canada and the National Sciences and Engineering Research Council is gratefully acknowledged. The computer work carried out has been made possible through generous funding by the Computing Centre of McGill University.

Finally, I would like to thank my wife for her patience and support throughout the duration of this work.

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION

1.1 The problem of the interface between two highly dissimilar materials

Materials with different properties are often combined in the design of engineering structures (e.g. reinforced concrete). The analysis of such structures is usually carried out by transforming them into 'equivalent' unimaterial models or by using numerical techniques like the finite element method. In general, the approach towards structures composed of different materials has been similar to that towards homogeneous structures, because high variations in material properties are not expected; for example, approximate values of elastic moduli for concrete and reinforcing steel are 30000 MPa and 200000 MPa respectively. Furthermore, the interface stress distribution is not usually of primary interest to the analyst.

However, recent advances in areas such as composite materials and biomechanics have created a need for a different approach in the analysis of bimaterial structures. In these cases, reliable stress calculations at interfaces of typically highly dissimilar materials are among the main objectives of the analysis. In fibre-reinforced composite materials, for example, the ratio of elastic moduli is approximately 25 (for a matrix of Araldite CT200 and fibres of duralumin) (Soh [32]), while in biomechanical applications the ratio of elastic moduli is usually as high as 100 (for steel prosthesis and Polymethylmethylacrylate bone cement) (Shrivastava et al. [31]). In both cases, failure of the bimaterial interface constitutes one of the main failure modes of the structure (fibre pull-out or delamination failure in composite materials and prosthesis loosening in biomechanics).

A stress analysis of such structures should be able to account for the interface boundary conditions which are critical to the behaviour of the whole structure.

1.2 Finite element method for solving the problem

The finite element method constitutes one of the most powerful numerical procedures currently available to stress analysts. Although there are many different forms of the method, the displacement-based method has emerged as the most popular in the field of structural mechanics (Cook [7]). A significant amount of research and numerous computer programs have been devoted to this method. As a matter of fact, most commercially available finite element computer packages are implementations of the displacement method.

In the displacement method, however, interelement nodal stress compatibility is not enforced. This means that elements sharing a node usually predict different stress values at that node. In the cases of homogeneous structures, these stress discontinuities, which are usually small, can be 'corrected' by several proposed averaging and 'smoothening' techniques (for example Herrmann [12], Hinton and Campbell [13], Loubignac et al. [18]).

As it has been pointed out (Cook [7], Loubignac et al. [18]), however, these procedures cannot be applied to such stress fields as the ones encountered at bimaterial interfaces or across sudden thickness changes. Generally, when a structure consisting of different materials is analysed by the conventional finite element method, severe violation of force equilibrium may take place at the intermaterial boundary points when the stresses are computed on the basis of displacement compatibility and constitutive relations alone (Salama and Utku [29]). Any attempt, therefore, to 'smoothen' the discontinuity by averaging methods would be inappropriate and erroneous. The usual procedure in these cases has been to repeat the analysis of the problem using a finer mesh (Cook [7]). However, this procedure can be very time consuming, since the convergence rate will likely be very slow, especially for cases that involve large differences in the properties of the adjacent materials.

A different approach would be to use a formulation having stresses as primary unknowns such as the mixed element or the hybrid element methods (e.g. Gallagher [9], Zienkiewicz [38]). In this way stresses can be continuous across element boundaries.

1.3 Previous displacement-based studies of the interface problem

With reference to the stress discontinuity obtained by the displacement method at bimaterial interfaces, few alternatives to the costly mesh refinement have been proposed so far. Salama and Utku [29] have suggested the application of the method of best fit strain tensors to the stress computation at intermaterial boundary points. In this method, the displacements are first computed in the usual way, and then, the strains and stresses are obtained through a procedure that acccounts for the interface stress boundary conditions. Compatibility, constitutive, and equilibrium conditions are satisfied and the proposed method seems quite successful. No conditions are imposed, however, on the nodal displacements at the interface so that they may not necessarily be consistent with the stress field satisfying the interface boundary conditions. Furthermore, significant computer memory is required due to the fact that a nodal set (list of nodes coinciding with the vertices of all elements meeting at that node) and a set of nodal lines (list of lines joining the node with the other nodes in the nodal set) for each node and element, respectively, must be retained in memory. That is, in addition to the usual information necessary for a conventional displacement method analysis. Finally, the examples chosen are not representative of the potential severity of the problem of bimaterial interfaces; the stress discontinuity obtained by standard finite element techniques was found to be small in these cases, and therefore, the value of the proposed method could not be fully appreciated. In the example of the vertical wedge, in particular, the analytical solution developed for a unimaterial case was assumed to apply to the bimaterial case as well, and, as such, was compared to the results of the proposed method.

More recently, Soh [32] proposed a modification to the conventional displacement method in order to achieve interface stress continuity. His technique is based on the nine-node rectangular element which is used to obtain the nodal displacements in the traditional way. The calculation of the stresses at an interface is then achieved after imposing the necessary and sufficient equilibrium and compatibility equations at that boundary. This process involves the fitting of the original displacement functions for both adjacent elements onto a nine node region, centered at the interface, and finally, reduces to the solution of a system of 36 equations. In addition to the computational disadvantages of this method (use of nine-node elements, complex procedure for stress calculation), its superiority over the conventional method was not apparent in the application presented. This application consists of the analysis of the fibre-matrix interface of a fibre-reinforced medium. Results from the proposed method were compared to conventional finite element analysis results and to results obtained from a photoelastic test performed on the same structure. The only case where there was clear evidence of superiority of the proposed technique over the conventional method was in the prediction of the location and magnitude of the stress peak occurring at the fibre tip.

1.4 Present study; objectives and organization of report

1.4.1 Objectives

The growing demand for more reliable stress analyses in problems involving interfaces of highly dissimilar materials renders the inefficiency of the displacement based finite element method unacceptable. Bimaterial interfaces are becoming increasingly common, especially in composite materials and in biomechanical applications. In the field of Orthopaedic Biomechanics, in particular, the lack of reliability of the displacement method concerning the calculation of stresses at prosthesis/cement and prosthesis/bone interfaces has been repeatedly pointed out (e.g. Huiskes and Chao [15], Rohlmann et al. [28]). The present study was initiated by the realization of the need for a modified technique to determine interface stress distributions.

The main objective of this study is to formulate a finite element capable of representing the correct stress and displacement boundary conditions at a bimaterial interface. It was decided to carry out the formulation of the element within the framework of the displacement method. The main reason for this was that it is intended, at a later stage, to incorporate the developed element into a large commercial finite element package (these programs are almost exclusively based on the displacement method). Hence, a widespread use of the element can be facilitated, since such programs are readily available. It was also decided to address the interface problem directly at the formulation stage, that is, before the displacements are computed. In this way, the calculated displacement, strain, and stress fields will be consistent and will satisfy the required interface boundary conditions, unlike the proposed methods of Salama and Utku [29] and Soh [32], where the displacements were computed before imposing the interface conditions. For the purposes of this study, a condition of perfect bond (no separation and no slip) will be assumed at the interface between the two materials. These materials will be considered isotropic, linear elastic.

After the development and validation of the element, a first application will be carried out. It consists of the calculation of the prosthesis/cement interface stress distribution at a prosthetic knee joint.

1.4.2 Organization of report

A general introduction to the problem of the bimaterial interface, previous displacement-based studies, and the main objectives of the present study have been included in this chapter.

The formulation of the interface element for the two-dimensional case is presented in Chapter 2, as well as a discussion on the boundary conditions at a bimaterial interface. Finally, some notes on the finite element program developed for the purpose of testing and applying the two-dimensional interface element are also included in Chapter 2.

The validation tests carried out to assess the performance of the proposed element are presented in Chapter 3, while Chapter 4 deals with an application of the element to a real-case interface problem from the field of Biomechanics.

A three-dimensional interface element has also been developed together with a three-dimensional finite element program. These are presented in Chapter 5 together with a validation test.

Chapter 6 provides a summary of the conclusions and observations made in the course of this study, as well as recommendations for future research.

Chapter 2

TWO-DIMENSIONAL FORMULATION

2.1 Perfect-bond continuity and discontinuity conditions

2.1.1 Theoretical considerations

The modelling of perfect bonding between two elastic media in contact is achieved through certain conditions on stress, strain, and displacements which must be satisfied at the bimaterial interface. These conditions will be presented in this section first for the general three-dimensional case, and then specialized for the two-dimensional case.

In order to simplify the form of the equations, tensor notation will be used throughout this section; the basic rules are briefly presented here, while a more detailed treatment is available in the literature (e.g. Sokolnikoff [33]).

Let the reference axes x, y, z be right handed rectangular Cartesian, and let them be named also as x_1, x_2, x_3. Lower case subscripts (i, j, k) are indices ranging from 1 to 3 in three dimensions and from 1 to 2 in two dimensions. A single subscript variable, for example u_i, denotes the components of a first order tensor (i.e. a vector) while a double subscript variable, for example ε_{ij}, denotes the components of a second order tensor. A subscript preceded by a comma denotes partial differentiation with respect to the coordinates; for example, $u_{i,j}$ denotes partial differentiation of u_i with respect to x_j. Finally, repeated indices indicate summation over the coordinate range of these indices, so that $u_{i,i}$ stands for $u_{1,1} + u_{2,2} + u_{3,3}$ (The summation rule will however be suspended at some places in the following presentation).

The term perfect bond is employed to express the condition that no separation or slip is allowed at the interface. In other words, perfect bond means that the displacement vector u_i is continuous at the interface points.

Figure 2.1 shows a typical bimaterial interface with its local coordinate system (axes x_2 and x_3 lie in the plane of the interface, while axis x_1 is in the direction of the normal to the interface). With reference to this figure and in view of the continuity of u_i, the following terms are also seen to be continuous at the interface : $u_{1,2}$ and $u_{1,3}$. On the other hand, the $u_{i,1}$ terms are not required to be continuous at the interface.

From the strain-displacement relations:

$$\varepsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}), \tag{2.1}$$

and taking into account the above mentioned continuity and discontinuity of displacement gradients, it follows that in the strain tensor representation at an interface point:

$$\begin{bmatrix} \varepsilon_{11}^* & \varepsilon_{12}^* & \varepsilon_{13}^* \\ \varepsilon_{12}^* & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{13}^* & \varepsilon_{23} & \varepsilon_{33} \end{bmatrix}, \tag{2.2}$$

the components without an asterisk are those which must be con-

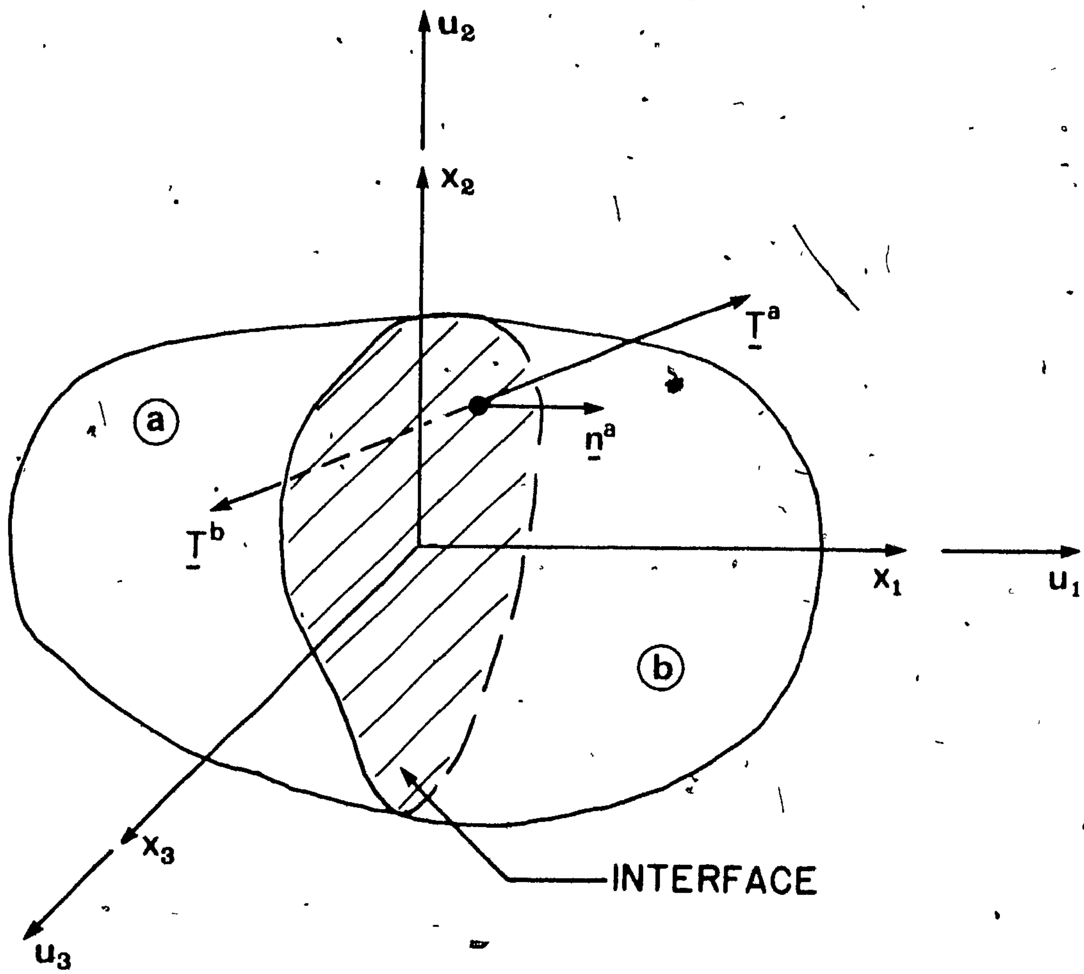

Figure 2.1: Schematic representation of a bimaterial interface.

tinuous, while those with an asterisk are permitted to be discontinuous. Thus, the condition of perfect bond implies not only the interface displacement continuity but also the continuity of the in-plane axial and shear strains. On the other hand, the out-of-plane strains can be discontinuous, depending on the material properties of the adjacent media.

The stress boundary conditions at the interface are determined from the required continuity of the stress (traction) vector T_i. Referring again to Fig. 2.1, the following equation relates the stress vectors of media a and b at the interface (principle of action-reaction):

$$T_i^a = -T_i^b, \tag{2.3}$$

where the superscripts a and b identify the two materials on the two sides of the interface.

Now it may be recalled (Sokolnikoff [33]) that the stress vector on a surface with outward normal n_j is expressible in terms of the stress components by the following formula :

$$T_i = \sigma_{ij} n_j. \tag{2.4}$$

Hence, since $n_j^a = -n_j^b$ at the interface, Eq.(2.3) becomes

$$\sigma_{ij}^a n_j^a = \sigma_{ij}^b n_j^a. \tag{2.5}$$

The unit normal to the interface is taken here parallel to the x axis so that

$$n_i^a = < 1, 0, 0 > . \tag{2.6}$$

Equations (2.5) and (2.6) yield the condition that at the interface one must have

$$\sigma_{11}^a = \sigma_{11}^b, \quad \sigma_{12}^a = \sigma_{12}^b, \quad \sigma_{13}^a = \sigma_{13}^b. \tag{2.7}$$

Thus, in the matrix representation of the stress tensor at an interface point:

$$\begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_{22}^* & \sigma_{23}^* \\ \sigma_{13} & \sigma_{23}^* & \sigma_{33}^* \end{bmatrix}, \tag{2.8}$$

the out-of-plane components identified by absence of asterisks are the ones which are required to be continuous; the in-plane components marked by asterisks are allowed to be discontinuous. It is interesting to compare the forms of Eqs. (2.2) and (2.8) and conclude that, regardless of the materials involved, if a stress component is required to be continuous, the corresponding strain component is allowed to be discontinuous; similarly, if a strain component is required to be continuous, the corresponding stress component is permitted to be discontinuous.

2.1.2 Finite element considerations

The conventional finite element displacement method by definition accounts for the nodal displacement compatibility. Moreover, the displacement functions are usually chosen to be such as to also satisfy the inter-element displacement compatibility. This inter-element compatibility therefore means that in so far as the strain compatibility is concerned it is satisfied fully in the sense that the strain components which must be continuous are indeed required by the formulation to be continuous , whereas those which can be discontinuous are allowed to assume different values at the inter-element boundary.

On the other hand, as is well known, as far as the stresses are concerned generally no account is taken of the local stress equilibrium conditions either within the bulk of the material or at the interfaces. The equilibrium is satisfied in the gross sense by minimizing the total potential energy of the structure with respect to its nodal displacements. This minimization process results in a set of algebraic 'equilibrium' equations with the nodal displacements as unknowns. Solution of this system of equations furnishes the nodal displacements (and unknown reaction forces) corresponding to the loading applied to the structure. Knowing the nodal displacements, the displacement and the strain field within each individual element can be determined. Then knowing the strain field, the stress field within an element can be determined by invoking the constitutive law of the material of the element. For linear, isotropic, elastic material behaviour, stresses are given in terms of strains by the following expressions:

$$\begin{aligned}
\sigma_{11} &= \lambda(\varepsilon^*_{11} + \varepsilon_{22} + \varepsilon_{33}) + 2\mu\varepsilon^*_{11}, \\
\sigma_{22} &= \lambda(\varepsilon^*_{11} + \varepsilon_{22} + \varepsilon_{33}) + 2\mu\varepsilon_{22}, \\
\sigma_{33} &= \lambda(\varepsilon^*_{11} + \varepsilon_{22} + \varepsilon_{33}) + 2\mu\varepsilon_{33}, \\
\sigma_{12} &= 2\mu\varepsilon^*_{12}, \\
\sigma_{13} &= 2\mu\varepsilon^*_{13}, \\
\sigma_{23} &= 2\mu\varepsilon_{23},
\end{aligned} \tag{2.9}$$

in which λ and μ are Lamé's constants. Since at an inter-element boundary, the strains $\varepsilon^*_{11}, \varepsilon^*_{12}$, and ε^*_{13} are allowed to be discontinuous for the two elements on the two sides of the interface, all stress components, with the exception of σ_{23}, are in general discontinuous even if the material of the two elements is the same. So that when the material properties for the two elements are in fact different, which is the case for a bimaterial interface, then without exception all stress components are made discontinuous across the interface.

If the interface stress continuity is to be enforced, then Eqs. (2.7) must be satisfied. In view of the strain compatibility this implies that the strain components must be related to satisfy the following conditions:

$$\begin{aligned}(\lambda^a\varepsilon^a_{11} - \lambda^b\varepsilon^b_{11}) + (\lambda^a - \lambda^b)(\varepsilon_{22} + \varepsilon_{33}) + 2(\mu^a\varepsilon^a_{11} - \mu^b\varepsilon^b_{11}) &= 0,\\ 2(\mu^a\varepsilon^a_{12} - \mu^b\varepsilon^b_{12}) &= 0,\\ 2(\mu^a\varepsilon^a_{13} - \mu^b\varepsilon^b_{13}) &= 0,\end{aligned} \tag{2.10}$$

where the superscripts a and b identify the quantities related to the a and b sides of the interface.

Now, ideally, the displacement functions should be chosen so as to satisfy the above conditions at every point of the inter-element interface boundary. However, this appears to be a rather difficult requirement to meet, and in the present work the satisfaction of the above conditions is restricted to just one point of the interface boundary common to two elements.

In two-dimensional plane stress or plane strain cases in the $x_1 - x_2$ plane, the interface surface is assumed to be a cylindrical surface with normal perpendicular to the x_3 axis (i.e. lying in the $x_1 - x_2$ plane). The stress and strain tensors are expressible respectively as :

$$\begin{bmatrix} \sigma_{nn} & \sigma_{nt} \\ \sigma_{nt} & \sigma^*_{tt} \end{bmatrix}, \tag{2.11}$$

and

$$\begin{bmatrix} \varepsilon^*_{nn} & \varepsilon^*_{nt} \\ \varepsilon^*_{nt} & \varepsilon_{tt} \end{bmatrix}, \tag{2.12}$$

where n is the normal direction and t is the tangential direction determined from the right-hand rule $\underline{n} \times \underline{t} = \underline{k}$, $\underline{k}$ being the unit vector in the $+x_3$ direction and $\underline{n}$ and $\underline{t}$ being the unit vectors respectively perpendicular and parallel to the interface (Fig. 2.2).

The two-dimensional stress-strain relations can be expressed as:

$$
\begin{aligned}
\sigma_{nn} &= \lambda_1(\varepsilon^*_{nn} + \varepsilon_{tt}) + 2\mu\varepsilon^*_{nn}, \\
\sigma_{tt} &= \lambda_1(\varepsilon^*_{nn} + \varepsilon_{tt}) + 2\mu\varepsilon^*_{nn}, \\
\sigma_{nt} &= 2\mu\varepsilon_{nt},
\end{aligned}
\tag{2.13}
$$

where λ_1 is defined to be:

$$\lambda_1 = \frac{2\mu\lambda}{2\mu + \lambda} \quad ; \quad \text{for the plane stress case } (\sigma_{zz} = 0), \tag{2.14}$$

and

$$\lambda_1 = \lambda \quad ; \quad \text{for the plane strain case } (\varepsilon_{zz} = 0). \tag{2.15}$$

It also follows that:

$$\varepsilon_{zz} = \frac{-\lambda}{2\mu + \lambda}(\varepsilon_{nn} + \varepsilon_{tt}) \quad ; \quad \text{in plane stress ,} \tag{2.16}$$

and

$$\sigma_{zz} = \lambda(\varepsilon^*_{nn} + \varepsilon_{tt}) \quad ; \quad \text{in plane strain.} \tag{2.17}$$

The interface continuity conditions on the stress vector, Eq. (2.7), can now be expressed as:

$$
\begin{gathered}
(\lambda_1^a\varepsilon_{nn}^a - \lambda_1^b\varepsilon_{nn}^b) + (\lambda_1^a - \lambda_1^b)\varepsilon_{tt} + 2(\mu^a\varepsilon_{nn}^a - \mu^b\varepsilon_{nn}^b) = 0, \\
2(\mu^a\varepsilon_{nt}^a - \mu^b\varepsilon_{nt}^b) = 0.
\end{gathered}
\tag{2.18}
$$

Note that in the finite element formulation of the interface element to be presented, imposition of the interface stress compatibility conditions, Eqs. (2.10) or Eqs. (2.18), is effected in an alternative way using the matrix notation and material constants E and ν rather than λ and μ.

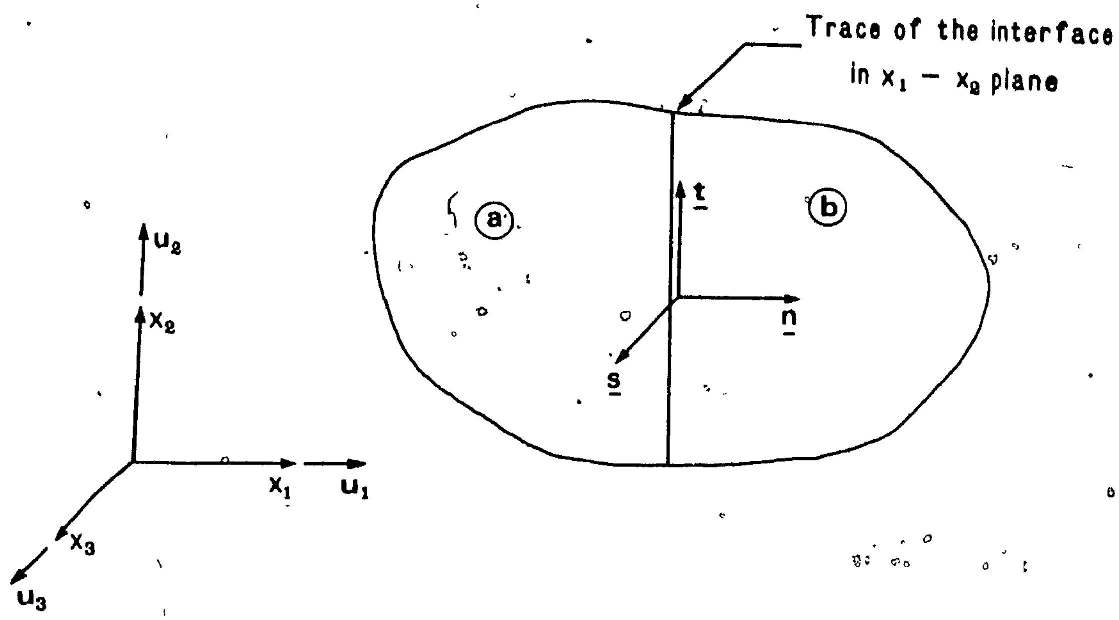

Figure 2.2: Two-dimensional interface problem.

2.2 Present plane-stress/plane-strain interface element

The proposed interface element is an element consisting of two adjoining quadrilateral elements, one on each side of the interface with the interface as their common boundary. The stress continuity conditions are imposed at the midpoint of the interface side of the elements. A midside node on the interface is a reasonable choice since stress computations at boundaries are considered to be most accurate at midside points (Cook [7]).

Considered individually, this interface node is the fifth node for each of the two elements of the combined interface element. Thus, each of the two elements is a five-node element, and its stiffness matrix can be determined in the usual fashion as shown below. The difference is, however, that instead of keeping the two degrees of freedom associated with the fifth node as free, they are selected so as to satisfy the two stress compatibility conditions, Eq. (2.18) or equivalently Eq. (2.7). This last step is the essence of the proposed formulation.

One of the five-node elements is shown in Fig. 2.3a together with its "natural" coordinate system. The derivation of a potential energy expression for the element follows the standard procedure presented in any finite element text (e.g. Cook [7]). Only a brief account is given here.

In this section, as well as in all subsequent ones, the formulation will be carried out in matrix notation, as it is very convenient for computer implementation. Square brackets, [], will represent matrices, while curly brackets, {}, will represent column vectors. No summation over repeated indices will be assumed unless explicitly stated.

An isoparametric quadrilateral element in the x-y plane is a square element in the $\xi - \eta$ plane. The term isoparametric means that the transformation linking the $\xi - \eta$ coordinates with x-y coordinates is the same as the one expressing the connection between the nodal displacements and the displacement field within the element. Thus, one has for the five-node element:

$$\begin{aligned} x &= N_1 x_1 + N_2 x_2 + N_3 x_3 + N_4 x_4 + N_5 x_5, \\ y &= N_1 y_1 + N_2 y_2 + N_3 y_3 + N_4 y_4 + N_5 y_5, \end{aligned} \tag{2.19}$$

and

$$\begin{aligned} u &= N_1 u_1 + N_2 u_2 + N_3 u_3 + N_4 u_4 + N_5 u_5, \\ v &= N_1 v_1 + N_2 v_2 + N_3 v_3 + N_4 v_4 + N_5 v_5, \end{aligned} \tag{2.20}$$

where $(x_1, y_1) \cdots (x_5, y_5)$ are the nodal coordinates and $(u_1, v_1) \cdots (u_5, v_5)$ are the nodal displacements of the element. The shape functions N_i for the five-node element are:

$$\begin{aligned} N_1 &= -\xi(\xi - 1)(\eta - 1)/4, \\ N_2 &= -\xi(\xi + 1)(\eta - 1)/4, \\ N_3 &= (\xi + 1)(\eta + 1)/4, \\ N_4 &= -(\xi - 1)(\eta + 1)/4, \\ N_5 &= (\xi^2 - 1)(\eta - 1)/2. \end{aligned} \tag{2.21}$$

It can be verified that these shape functions have the property:

$$\sum_i N_i = 1, \tag{2.22}$$

which together with Eqs. (2.19) and (2.20) restated:

$$\sum_i N_i x_i = x \quad \text{and} \quad \sum_i N_i y_i = y \quad , \tag{2.23}$$

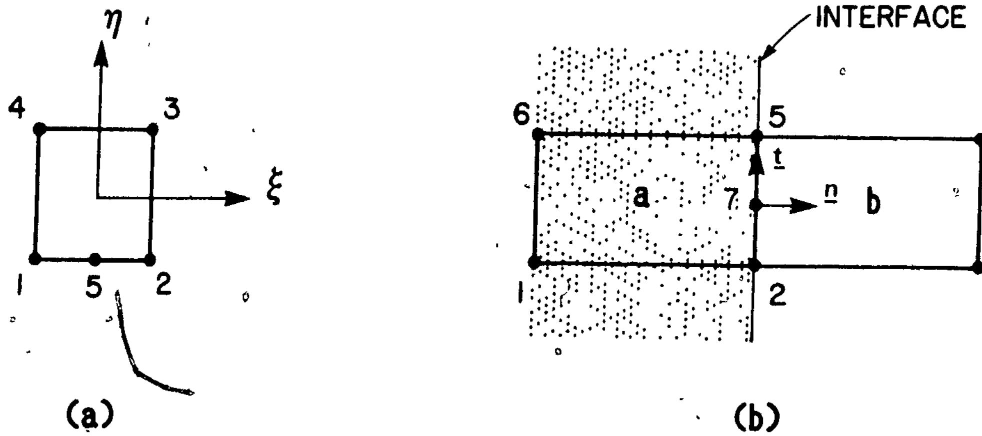

Figure 2.3: a) Five node quadrilateral isoparametric element;
b) Typical pair of five-node elements sharing an interface.

ensure that the displacement functions chosen are capable of modelling the rigid body displacement and rotation of the element. Moreover, since the displacements along an element side depend only on the displacements of the nodes belonging to that side, the inter-element displacement (and strain) compatibility is satisfied regardless of the particular values of the nodal displacements and regardless of the material properties of the elements.

Considering now the element a, belonging to the material a, the strains are expressible as:

$$
\begin{aligned}
\varepsilon^a_{xx} &= \frac{\partial u^a}{\partial x} = \frac{\partial N_1}{\partial x}u_1 + \frac{\partial N_2}{\partial x}u_2 + \cdots + \frac{\partial N_5}{\partial x}u_5, \\
\varepsilon^a_{yy} &= \frac{\partial v^a}{\partial y} = \frac{\partial N_1}{\partial y}v_1 + \frac{\partial N_2}{\partial y}v_2 + \cdots + \frac{\partial N_5}{\partial y}v_5, \\
\gamma^a_{xy} &= 2\varepsilon^a_{xy} = (\frac{\partial u^a}{\partial y} + \frac{\partial v^a}{\partial x}) = \\
&= [(\frac{\partial N_1}{\partial y}u_1 + \frac{\partial N_1}{\partial x}v_1) + \cdots + (\frac{\partial N_5}{\partial y}u_5 + \frac{\partial N_5}{\partial x}v_5)],
\end{aligned}
\tag{2.24}
$$

or, symbolically in the matrix notation as:

$$\{\varepsilon^a\} = [B^a]\{\Delta^a\}, \tag{2.25}$$

where $\{\Delta^a\}$ is the vector of nodal displacements:

$$\{\Delta^a\}^T =< u_1, v_1, \cdots, u_5, v_5 >, \tag{2.26}$$

and

$$\{\varepsilon^a\}^T =< \varepsilon^a_{xx}, \varepsilon^a_{yy}, \gamma^a_{xy} > . \tag{2.27}$$

$[B^a]$ is the matrix (of size 3x10) whose elements are the derivatives of the shape functions $N_i (i = 1 \quad \text{to} \quad 5)$ with respect to x and y:

$$[B^a] = \begin{bmatrix} \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & 0 & \cdots \\ 0 & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & \cdots \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial x} & \cdots \end{bmatrix}. \tag{2.28}$$

However, since N_i are functions of ξ and η, the above differentiation with respect to x and y cannot be carried out explicitly, but by using the relations of the type:

$$\frac{\partial N_i}{\partial x} = \frac{\partial N_i}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial N_i}{\partial \eta}\frac{\partial \eta}{\partial x}. \tag{2.29}$$

Thus, the problem of determining $[B^a]$ is reduced to first computing $\frac{\partial \xi}{\partial x}, \frac{\partial \eta}{\partial x}$ and $\frac{\partial \xi}{\partial y}, \frac{\partial \eta}{\partial y}$. However, again since x and y are assumed to be functions of ξ and η, we may easily compute $\frac{\partial x}{\partial \xi}$ etc. but computing $\frac{\partial \xi}{\partial x}$ etc. requires (numerical) inversion of the so called Jacobian matrix [J] given by:

$$[J] = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}. \tag{2.30}$$

Now since:

$$\begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{2.31}$$

we have:

$$[\Gamma] = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \end{bmatrix} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} = [J]^{-1}. \tag{2.32}$$

Combining Eqs. (2.28), (2.29), and (2.32) the strain-displacement matrix $[B^a]$ can be written as:

$$[B^a] = [B_1^a][B_2^a], \tag{2.33}$$

where

$$[B_1^a] = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} & 0 & 0 \\ 0 & 0 & \Gamma_{21} & \Gamma_{22} \\ \Gamma_{21} & \Gamma_{22} & \Gamma_{11} & \Gamma_{12} \end{bmatrix}, \tag{2.34}$$

and

$$[B_2^a] = \begin{bmatrix} \frac{\partial N_1}{\partial \xi} & 0 & \frac{\partial N_2}{\partial \xi} & 0 & \cdots \\ \frac{\partial N_1}{\partial \eta} & 0 & \frac{\partial N_2}{\partial \eta} & 0 & \cdots \\ 0 & \frac{\partial N_1}{\partial \xi} & 0 & \frac{\partial N_2}{\partial \xi} & \cdots \\ 0 & \frac{\partial N_1}{\partial \eta} & 0 & \frac{\partial N_2}{\partial \eta} & \cdots \end{bmatrix}. \tag{2.35}$$

The potential energy functional for the element is given by the following expression:

$$\Pi_p^a = \frac{1}{2}\int_V \{\epsilon^a\}^T\{\sigma^a\}dV \text{ - P.E. of loads}, \tag{2.36}$$

where $\{\sigma^a\}$ is the vector of the stress components:

$$\{\sigma^a\}^T = < \sigma_x^a, \sigma_y^a, \tau_{xy}^a > . \tag{2.37}$$

For a linear elastic material, the stress vector $\{\sigma\}$ is related to the strain vector $\{\epsilon\}$ through Hooke's law:

$$\{\sigma^a\} = [E^a]\{\epsilon^a\}, \tag{2.38}$$

where $[E^a]$ is the matrix of elastic moduli, which in the case of plane stress is:

$$[E^a] = \frac{E^a}{(1-(\nu^a)^2)} \begin{bmatrix} 1 & \nu^a & 0 \\ \nu^a & 1 & 0 \\ 0 & 0 & \frac{1-\nu^a}{2} \end{bmatrix}, \qquad (2.39)$$

and in the plane strain case is:

$$[E^a] = \frac{E^a}{(1+\nu^a)(1-2\nu^a)} \begin{bmatrix} 1-\nu^a & \nu^a & 0 \\ \nu^a & 1-\nu^a & 0 \\ 0 & 0 & \frac{1-2\nu^a}{2} \end{bmatrix}. \qquad (2.40)$$

Using Eq. (2.38), the potential energy expression in Eq. (2.36) is rewritten as:

$$\Pi_p^a = \frac{1}{2}\int_V \{\varepsilon^a\}^T[E^a]\{\varepsilon^a\}\mathrm{dV} \text{ - P.E. of loads,} \qquad (2.41)$$

and by substituting in the above equation the strain-displacement relation of Eq. (2.25) as:

$$\Pi_p^a = \frac{1}{2}\{\Delta^a\}^T(\int_V [B^a]^T[E^a][B^a]\mathrm{dV})\{\Delta^a\} \quad \text{ - P.E. of loads.} \qquad (2.42)$$

This functional is now defined in terms of $\{\Delta^a\}$, since $[B^a]$ is given by Eq.(2.33) and $[E^a]$ is given by Eq. (2.39) or (2.40). The equilibrium equations are found by making the potential energy functional stationary with respect to the nodal displacements $\{\Delta^a\}$. This results in the following formula for the stiffness matrix of the element:

$$[K^a] = \int_V [B^a]^T[E^a][B^a]\mathrm{dV} = t\int_{-1}^{1}\int_{-1}^{1}[B^a]^T[E^a][B^a]\det\,[\mathrm{J}]\; d\xi d\eta, \qquad (2.43)$$

where, since the coefficients of $[B^a]$ are functions of ξ and η, the integration must be carried out with respect to these variables by substituting dV = tdet[J]d ξ d η and where the thickness t of the element is taken to be constant. The integration is usually carried out numerically by evaluating the integrand at a selected number of Gauss points within the elements, weighting the quantities so obtained appropriately and then summing them.

A typical pair of elements sharing a common interface is shown in Fig. 2.3b. Node 7 is the interface fifth node of each element, where the stress boundary conditions will be imposed. From Hooke's law, Eq. (2.38), the stresses at the two sides of node 7 are:

$$\{\sigma^a\} = [E^a]\{\varepsilon^a\}, \tag{2.44}$$

and

$$\{\sigma^b\} = [E^b]\{\varepsilon^b\}. \tag{2.45}$$

Substituting the strain-displacement equations (2.25) into Eqs. (2.44) and (2.45) one obtains:

$$\{\sigma^a\} = [E^a][B^a]\{\Delta^a\}, \tag{2.46}$$

and

$$\{\sigma^b\} = [E^b][B^b]\{\Delta^b\}. \tag{2.47}$$

The boundary conditions derived in Section 2.1 require the continuity of the normal and shear stress components at node 7. The local n-t coordinate system shown in Fig. 2.3b has its origin on node 7 and the n-axis is the normal to the interface side, while the t-axis is tangential to the interface in the nodal direction 1-2. With reference to this n-t system, the stress boundary conditions at node 7 are:

$$\left\{ \begin{matrix} \sigma_n^a \\ \tau_{nt}^a \end{matrix} \right\} = \left\{ \begin{matrix} \sigma_n^b \\ \tau_{nt}^b \end{matrix} \right\}. \tag{2.48}$$

In order to obtain these local stress components in the n-t-z system, from the global components in the x-y-z system, the following transformation must be carried out:

$$\left\{ \begin{array}{c} \sigma_n \\ \tau_{nt} \end{array} \right\} = [T]\{\sigma\}, \tag{2.49}$$

where

$$[T] = \left[\begin{array}{ccc} c^2 & s^2 & 2cs \\ -cs & cs & c^2 - s^2 \end{array} \right], \tag{2.50}$$

with c and s as the direction cosines of the n axis with respect to the global x-y axes.

By combining Eqs. (2.46), (2.47), (2.48), and (2.49), the equality of the interface stress vector is then expressible as:

$$[T][E^a][B^a]\{\Delta^a\} = [T][E^b][B^b]\{\Delta^b\}, \tag{2.51}$$

which by introducing the notation:

$$[Q^a] = [T][E^a][B^a] \quad \text{and} \quad [Q^b] = [T][E^b][B^b], \tag{2.52}$$

can be rewritten as:

$$[Q^a]\{\Delta^a\} = [Q^b]\{\Delta^b\}. \tag{2.53}$$

A pair of five-node elements, such as the ones shown in Fig. 2.3b, form a new 7-node interface element. The local numbering of this composite element is also shown in this Fig. 2.3b. The objective now is to select the degrees of freedom of the seventh node of this element so as to satisfy the interface stress equality conditions, Eqs. (2.53).

Equations (2.53) are now rewritten in such a way as to isolate the degrees of freedom of the interface node 7:

$$[Q^a]\{\Delta^a\} = [Q1^a]\{d^c\} + [Q2^a]\left\{ \begin{array}{c} u_7 \\ v_7 \end{array} \right\}; \tag{2.54}$$

and

$$[Q^b]\{\Delta^b\} = [Q1^b]\{d^c\} + [Q2^b]\left\{ \begin{array}{c} u_7 \\ v_7 \end{array} \right\}, \tag{2.55}$$

where

$$\{d^c\}^T =< u_1, v_1, u_2, v_2, u_3, v_3, u_4, v_4, u_5, v_5, u_6, v_6 > \tag{2.56}$$

is the vector of unrestrained nodal displacements of the composite element, and $[Q1^a]$ etc. are the matrices resulting from the decomposition of $[Q^a]$ and $[Q^b]$ as follows:

$$[Q1^a] = \left[\begin{array}{cccccccccccc} Q^a_{1,7} & Q^a_{1,8} & Q^a_{1,1} & Q^a_{1,2} & 0 & 0 & 0 & 0 & Q^a_{1,3} & Q^a_{1,4} & Q^a_{1,5} & Q^a_{1,6} \\ Q^a_{2,7} & Q^a_{2,8} & Q^a_{2,1} & Q^a_{2,2} & 0 & 0 & 0 & 0 & Q^a_{2,3} & Q^a_{2,4} & Q^a_{2,5} & Q^a_{2,6} \end{array} \right],$$

$$[Q2^a] = \left[\begin{array}{cc} Q^a_{1,9} & Q^a_{1,10} \\ Q^a_{2,9} & Q^a_{2,10} \end{array} \right], \tag{2.57}$$

$$[Q1^b] = \left[\begin{array}{cccccccccccc} 0 & 0 & Q^b_{1,3} & Q^b_{1,4} & Q^b_{1,5} & Q^b_{1,6} & Q^b_{1,7} & Q^b_{1,8} & Q^b_{1,1} & Q^b_{1,2} & 0 & 0 \\ 0 & 0 & Q^b_{2,3} & Q^b_{2,4} & Q^b_{2,5} & Q^b_{2,6} & Q^b_{2,7} & Q^b_{2,8} & Q^b_{2,1} & Q^b_{2,2} & 0 & 0 \end{array} \right]$$

$$[Q2^b] = \left[\begin{array}{cc} Q^b_{1,9} & Q^b_{1,10} \\ Q^b_{2,9} & Q^b_{2,10} \end{array} \right]. \tag{2.58}$$

Equations (2.54) and (2.55) can now be used to express the connection between $< u_7, v_7 >$ and $\{d^c\}$ which must exist by virtue of the equality of the stress vector at the interface node. This connection is expressible as:

$$\left\{ \begin{array}{c} u_7 \\ u_7 \end{array} \right\} = [L]\{d^c\} \quad , \tag{2.59}$$

where

$$[L] = [[Q2^b] - [Q2^a]]^{-1}[[Q1^a] - [Q1^b]] \quad . \tag{2.60}$$

Combining Eqs. (2.54), (2.55) and (2.59), the degrees of freedom of each of the two elements a and b can be expressed in terms of the global degrees of freedom of the interface element, $\{d^c\}$:

$$\begin{aligned} \{\Delta^a\} &= [R^a]\{d^c\} \quad , \\ \{\Delta^b\} &= [R^b]\{d^c\} \quad ; \end{aligned} \tag{2.61}$$

where

$$\begin{aligned} [R^a] &= [Q^a]^{-1}[[Q1^a] + [Q2^a][L]] \quad , \\ [R^b] &= [Q^b]^{-1}[[Q1^b] + [Q2^b][L]] \quad . \end{aligned} \tag{2.62}$$

The potential energy of the interface element is formed by adding the potential energy expressions for elements a and b. Thus, using Eq. (2.42)

$$\Pi_p^{\mathrm{i}} = \frac{1}{2}\{\Delta^a\}^T[K^a]\{\Delta^a\} + \frac{1}{2}\{\Delta^b\}^T[K^b]\{\Delta^b\} \quad \text{- P.E. of nodal loads.} \tag{2.63}$$

Substitution for $\{\Delta^a\}$ and $\{\Delta^b\}$ in terms of $\{d^c\}$, Eqs. (2.61), then yields:

$$\Pi_p^{\mathrm{i}} = \frac{1}{2}\{d^c\}^T[[R^a]^T[K^a][R^a]+[R^b]^T[K^b][R^b]]\{d^c\} \quad \text{- P.E. of nodal loads.} \tag{2.64}$$

From Eq. (2.64) it then follows that the stiffness matrix of the composite interface element is:

$$[K^{\mathrm{i}}] = [R^a]^T[K^a][R^a] + [R^b]^T[K^b][R^b] \quad . \tag{2.65}$$

The assembly of the global stiffness matrix and the calculation of the nodal displacements are done in the same way as in the conventional method. The interface nodal displacements are not part of the global displacement vector, since they were replaced by the stress compatibility conditions, Eqs. (2.59). However, once the global displacements have been obtained, Eqs. (2.59), (2.25), (2.46) and (2.47) can be used to obtain respectively the displacements, strains $\{\varepsilon^a\}$ and $\{\varepsilon^b\}$, and stresses $\{\sigma^a\}$ and $\{\sigma^b\}$ at the interface node. By virtue of the formulation these values are in accordance with the required continuity and permissible discontinuity conditions of perfect bond.

From a computational point of view, the proposed procedure is not very different from the standard technique. The global degrees of freedom of the structure, which constitute the most important factor in cost considerations, remain exactly the same as if regular quadrilateral elements had been used instead of interface elements. Additional computational effort is required only for the calculation of the [L] matrices, which is, however, a simple procedure involving multiplication of low order matrices. Thus, it is to be appreciated that the interface element enforces the stress and displacement compatibility at a bimaterial boundary point at a minimal additional computational cost without significantly affecting the usual displacement method procedure. For this reason, it is especially suited for insertion into a general-purpose commercial finite element program that allows for user-defined elements.

The fundamentals of the new element formulation have been presented in this section for the two-dimensional plane stress/plane strain case. The following section deals with the modifications and special considerations for the case of axisymmetric analysis.

2.3 Axisymmetric case

For the purposes of this study, an axisymmetric model is assumed to consist of a structure of axisymmetric geometry subjected to axisymmetric loading; the materials continue to be assumed as linear, isotropic elastic. The case of axisymmetric structures subjected to nonaxisymmetric loading conditions will not be considered in this work.

Although an axisymmetric analysis deals physically with a three-dimensional problem, the fact that all variables are independent of the circumferential coordinate allows for a mathematically correct two-dimensional treatment. With reference to the cylindrical coordinate system shown in Fig. 2.4, the axisymmetric condition implies that:

$$\begin{aligned} u_\theta &= 0, \\ \varepsilon_{r\theta} &= \varepsilon_{\theta z} = 0, \\ \sigma_{r\theta} &= \sigma_{\theta z} = 0. \end{aligned} \tag{2.66}$$

The displacement components u and v now correspond to the radial and axial, i.e. r and z, directions. The non-zero stress and strain components may be written in the vector form as:

$$\begin{aligned} \{\sigma\}^T &= < \sigma_r, \sigma_\theta, \sigma_z, \tau_{zr} > \quad , \\ \{\varepsilon\}^T &= < \varepsilon_r, \varepsilon_\theta, \varepsilon_z, \gamma_{zr} > \quad . \end{aligned} \tag{2.67}$$

where the strain components are related to the displacement field as follows:

$$\begin{aligned} \varepsilon_{rr} &= \frac{\partial u}{\partial r} \quad , \\ \varepsilon_{\theta\theta} &= \frac{u}{r} \quad , \\ \varepsilon_{zz} &= \frac{\partial v}{\partial z} \quad , \\ \gamma_{rz} &= \frac{\partial u}{\partial z} + \frac{\partial v}{\partial r} \quad . \end{aligned} \tag{2.68}$$

The stress-strain relationship (Hooke's law) is:

$$\{\sigma\} = [E]\{\varepsilon\} \quad , \tag{2.69}$$

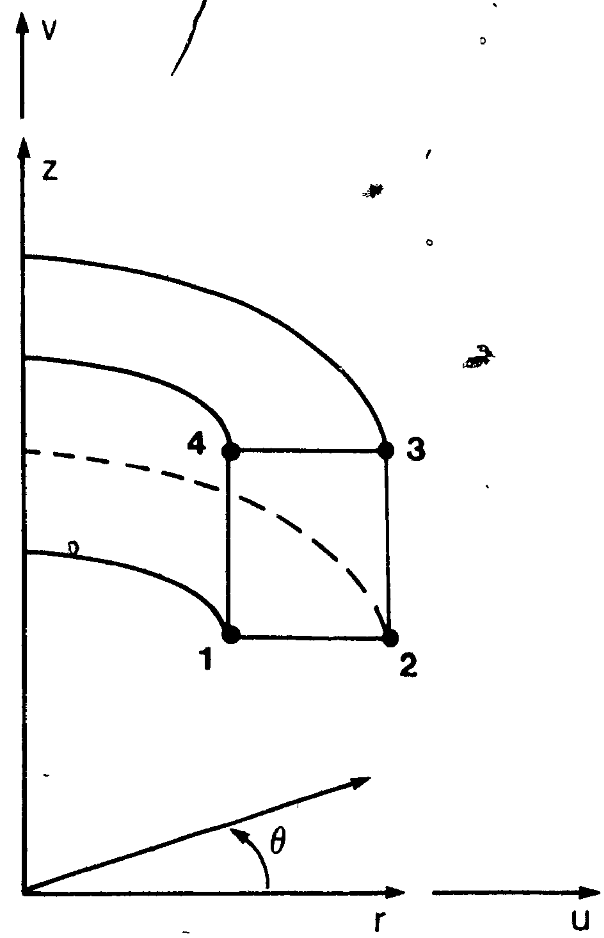

Figure 2.4: Cylindrical coordinate system and typical axisymmetric element.

where now:

$$[E] = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 \\ \nu & 1-\nu & \nu & 0 \\ \nu & \nu & 1-\nu & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} . \tag{2.70}$$

In the finite element implementation, the independence with respect to the θ-coordinate means that it is sufficient to consider the quantities within any one radial, r-z, plane. We again consider isoparametric quadrilateral elements (which are in fact ring elements of quadrilateral cross-sections) and assume displacement functions and isoparametric coordinate transformations as:

$$u = \sum_i N_i u_i, \quad v = \sum_i N_i v_i \quad , \tag{2.71}$$

and

$$r = \sum_i N_i r_i, \quad z = \sum_i N_i z_i \quad , \tag{2.72}$$

where the summation extends to four terms for regular quadrilateral elements and to five in cases of five-node quadrilateral elements used for constructing the composite interface element. The shape functions N_i are exactly the same functions of ξ and η as chosen before for the two- dimensional cases. The development below assumes a five node quadrilateral element.

The strain field corresponding to the assumed displacement field, Eqs. (2.71), can be obtained by using Eqs. (2.68) and can be expressed as:

$$\{\varepsilon\} = [B]\{\Delta\} \quad , \tag{2.73}$$

where $\{\Delta\}^T = < u_1, v_1, u_2, v_2, \cdots, u_5, v_5 >$ is the vector of nodal displacements of the element, and the [B] matrix is given by:

$$[B] = \begin{bmatrix} \frac{\partial N_1}{\partial r} & 0 & \frac{\partial N_2}{\partial r} & 0 & \cdots \\ \frac{N_1}{r} & 0 & \frac{N_2}{r} & 0 & \cdots \\ 0 & \frac{\partial N_1}{\partial z} & 0 & \frac{\partial N_2}{\partial z} & \cdots \\ \frac{\partial N_1}{\partial z} & \frac{\partial N_1}{\partial r} & \frac{\partial N_2}{\partial z} & \frac{\partial N_2}{\partial r} & \cdots \end{bmatrix} \quad (2.74)$$

Again, since N_i are functions of ξ and η, the above differentiation cannot be carried out explicitly, and one needs analoguous to the two-dimensional cases the inverse of the Jacobian matrix:

$$[J] = \begin{bmatrix} \frac{\partial r}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial r}{\partial \eta} & \frac{\partial z}{\partial \eta} \end{bmatrix} \quad (2.75)$$

Denoting the inverse as:

$$[\Gamma] = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \end{bmatrix} = \begin{bmatrix} \frac{\partial \xi}{\partial r} & \frac{\partial \eta}{\partial r} \\ \frac{\partial \xi}{\partial z} & \frac{\partial \eta}{\partial z} \end{bmatrix} = [J]^{-1} \quad , \quad (2.76)$$

the [B] matrix of Eq. (2.74) can be expressed as:

$$[B] = [B_1][B_2] \quad , \quad (2.77)$$

where

$$[B_1] = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{r} \\ 0 & 0 & \Gamma_{21} & \Gamma_{22} & 0 \\ \Gamma_{21} & \Gamma_{22} & \Gamma_{11} & \Gamma_{12} & 0 \end{bmatrix} \quad , \quad (2.78)$$

and

$$[B_2] = \begin{bmatrix} \frac{\partial N_1}{\partial \xi} & 0 & \frac{\partial N_2}{\partial \xi} & 0 & \cdots \\ \frac{\partial N_1}{\partial \eta} & 0 & \frac{\partial N_2}{\partial \eta} & 0 & \cdots \\ 0 & \frac{\partial N_1}{\partial \xi} & 0 & \frac{\partial N_2}{\partial \xi} & \cdots \\ 0 & \frac{\partial N_1}{\partial \eta} & 0 & \frac{\partial N_2}{\partial \eta} & \cdots \\ N_1 & 0 & N_2 & 0 & \cdots \end{bmatrix} \quad . \tag{2.79}$$

The potential energy of the element can then be expressed as:

$$\Pi_p = \frac{2\pi}{2}\{\Delta\}^T[\int_{-1}^{1}\int_{-1}^{1}[B]^T[E][B]\mathrm{rdet[J]d}\xi\mathrm{d}\eta\{\Delta\} \quad \text{- P.E. of loads.} \tag{2.80}$$

from which it follows that [K], the stiffness matrix per radian of the circumferential angle is:

$$[K] = \int_{-1}^{1}\int_{-1}^{1}[B]^T[E][B]\mathrm{rdet[J]d}\xi\mathrm{d}\eta \quad . \tag{2.81}$$

As before, Gaussian integration is necessary to evaluate the terms of this stiffness matrix.

The formulation of the interface element proceeds along the same lines as for the two-dimensional elements. This element consists of a composite of two adjacent 5-node quadrilateral elements, one on each side of the bimaterial interface with the fifth node at the middle of the interface side common to the two quadrilateral elements. The stress continuity conditions still remain the same, requiring continuity of the normal stress and the shear stress at an interface point. The two degrees of freedom pertaining to the fifth node (node no. 7) are again so chosen as to satisfy the required continuity of the above two stress components at this node.

The [Q] matrices in the stress continuity condition:

$$[Q^a]\{\Delta^a\} = [Q^b]\{\Delta^b\} \quad , \tag{2.82}$$

are given by:

$$[Q^a] = [T][E^a][B^a] \quad \text{and} \quad [Q^b] = [T][E^b][B^b] \quad , \tag{2.83}$$

where the [E] and [B] matrices for materials a and b are those given respectively by Eqs. (2.70) and (2.77), and [T] is the transformation matrix given by:

$$[T] = \begin{bmatrix} c^2 & 0 & s^2 & 2cs \\ -cs & 0 & cs & c^2 - s^2 \end{bmatrix} \quad , \tag{2.84}$$

with c and s as the direction cosines of the normal with respect to r and z axes.

Isolating the degrees of freedom of the common node no. 7, the above equation may be written as:

$$[Q^a]\{\Delta^a\} = [Q1^a]\{d^c\} + [Q2^a]\left\{ \begin{matrix} u_7 \\ v_7 \end{matrix} \right\} \quad , \tag{2.85}$$

and

$$[Q^b]\{\Delta^b\} = [Q1^b]\{d^c\} + [Q2^b]\left\{ \begin{matrix} u_7 \\ v_7 \end{matrix} \right\} \quad , \tag{2.86}$$

$$\text{where} \quad \{d^c\}^T = < u_1, v_1, u_2, v_2, \cdots, u_6, v_6 > \tag{2.87}$$

is the vector of nodal displacements of the composite element. The matrices $[Q1^a]$ etc. are expressible as:

$$[Q1^a] = \begin{bmatrix} Q^a_{1,7} & Q^a_{1,8} & Q^a_{1,1} & Q^a_{1,2} & 0 & 0 & 0 & 0 & Q^a_{1,3} & Q^a_{1,4} & Q^a_{1,5} & Q^a_{1,6} \\ Q^a_{2,7} & Q^a_{2,8} & Q^a_{2,1} & Q^a_{2,2} & 0 & 0 & 0 & 0 & Q^a_{2,3} & Q^a_{2,4} & Q^a_{2,5} & Q^a_{2,6} \end{bmatrix} \quad ,$$

$$[Q2^a] = \begin{bmatrix} Q^a_{1,9} & Q^a_{1,10} \\ Q^a_{2,9} & Q^a_{2,10} \end{bmatrix} \quad , \tag{2.88}$$

$$[Q1^b] = \begin{bmatrix} 0 & 0 & Q^b_{1,3} & Q^b_{1,4} & Q^b_{1,5} & Q^b_{1,6} & Q^b_{1,7} & Q^b_{1,8} & Q^b_{1,1} & Q^b_{1,2} & 0 & 0 \\ 0 & 0 & Q^b_{2,3} & Q^b_{2,4} & Q^b_{2,5} & Q^b_{2,6} & Q^b_{2,7} & Q^b_{2,8} & Q^b_{2,1} & Q^b_{2,2} & 0 & 0 \end{bmatrix} \quad ,$$

$$[Q2^b] = \begin{bmatrix} Q^b_{1,9} & Q^b_{1,10} \\ Q^b_{2,9} & Q^b_{2,10} \end{bmatrix} \quad . \tag{2.89}$$

The stress continuity conditions, Eq. (2.82), then lead to:

$$\begin{Bmatrix} u_7 \\ v_7 \end{Bmatrix} = [L]\{d^c\} \quad , \tag{2.90}$$

where

$$[L] = [[Q2^b] - [Q2^a]]^{-1}[[Q1^a] - [Q1^b]] \quad . \tag{2.91}$$

Substituting Eqs. (2.90) into Eqs. (2.85), the nodal displacement vectors of the two individual elements may be written as:

$$\{\Delta^a\} = [R^a]\{d^c\} \quad , \tag{2.92}$$

and

$$\{\Delta^b\} = [R^b]\{d^c\} \quad , \tag{2.93}$$

where:

$$[R^a] = [Q^a]^{-1}[[Q1^a] + [Q2^a][L]] \quad , \tag{2.94}$$

and

$$[R^b] = [Q^b]^{-1}[[Q1^b] + [Q2^b][L]] \quad . \tag{2.95}$$

Hence, by virtue of Eq. (2.81) and Eqs. (2.94) and (2.95) the stiffness matrix of the composite element can now be written as:

$$[K^c] = [R^a]^T[K^a][R^a] + [R^b]^T[K^b][R^b] \quad . \tag{2.96}$$

We thus see that the formulation of the axisymmetric interface composite element is very similar to that for the corresponding two- dimensional element except that the matrix sizes in the intermediate steps are different because of the circumferential strain and stress terms. It is also of interest to note that while the circumferential strain, $\varepsilon_{\theta\theta}$, must be continuous, the circumferential stress, $\sigma_{\theta\theta}$, is permitted to vary discontinuously.

2.4 Notes on the development of the program

A two-dimensional finite element program with plane-stress/plane-strain and axisymmetric analysis capabilities has been developed for the purpose of testing and validating the new element. The complete listing is presented in Appendix B.1. This section consists of a brief description of the elements and the subroutines incorporated into the program.

2.4.1 Library of elements

1. Spring element (BAR) : This is a one-dimensional linear element resisting only axial forces. The formulation may be found in any text, e.g. Cook [7].

2. Constant strain triangular element (CST) : The formulation for this well-known element is based on Cook [7] for the plane-stress/plane-strain case, and on Wilson [36] for the axisymmetric case.

3. Linear quadrilateral isoparametric element (QUAD4) : This element is based on the standard isoparametric formulation, e.g. Cook [7]. The user has a choice of a 2x2 or a 3x3 Gauss integration scheme. There is also an option to include incompatible bending modes in order to improve the bending performance of

the element. The implementation of the incompatible modes is based on the work of Wilson et al. [37] and Taylor et al. [34], although caution must be exercised for distorted, non-rectangular elements (Cook [7]). The QUAD4 element may be used for either plane-stress/plane-strain or axisymmetric analyses. In the axisymmetric case, the problem of the indeterminacy (1/r factor at r=0) encountered at 'core' elements (i.e. elements with nodes on the axis of symmetry) is resolved by setting the radius of the nodal circle to a very small finite value (1.0×10^{-5}) (Wilson [36]).

4. Quadrilateral isoparametric interface element (QUAD5) :

   This element is based on the formulation presented in this chapter. It is a five node quadrilateral, its fifth node being at the midpoint of the interface side. Normal and shear stress continuity are enforced across the interface at this node. The user has again the option to choose between two Gauss quadrature schemes (2x2 or 3x3). The QUAD5 element may be used in either plane-stress/plane-strain or axisymmetric analyses. Pairs of QUAD5 elements sharing an interface are automatically combined to form six-node composite elements satisfying continuity of the stress vector at the interface node.

2.4.2 Library of subroutines

A schematic flowchart is shown in Fig. 2.5 outlining the Main Program functions. The functions of the subroutines will be briefly presented in this subsection. All subroutines were developed by the author except where an explicit reference is given.

GENER : Generation of nodes and elements. The user must specify the first and last card in the sequence and the node or element

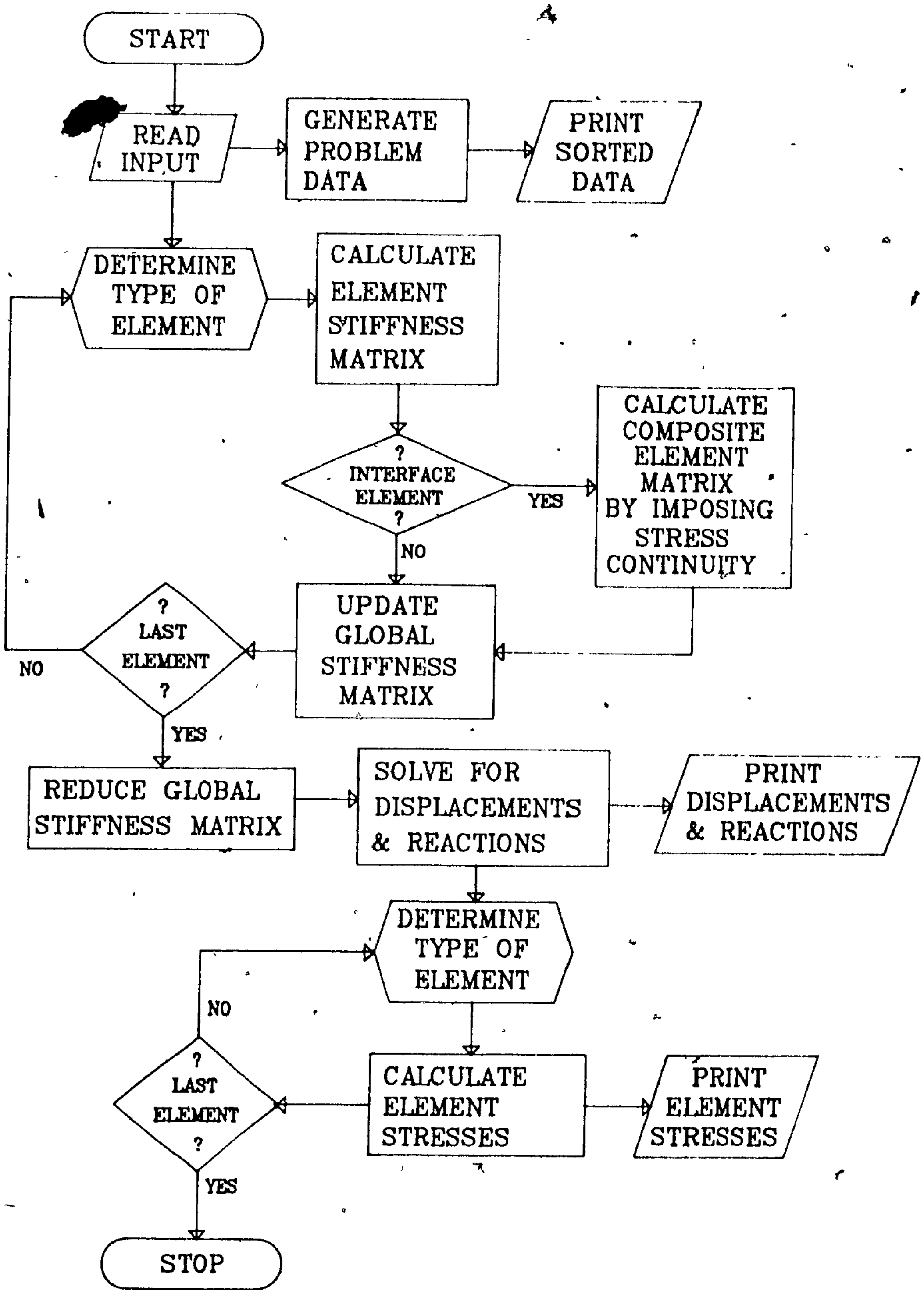

Figure 2.5: Schematic flowchart of the main functions of the developed finite element program.

number increment. The incremented nodal coordinates or element nodes are automatically computed. The elements need not be in element number sequence, as is the case with other programs (e.g. SAP IV).

DATA : Printing of the input data in sorted order.

STIFF : Assembly of the global stiffness matrix. This subroutine requests the stiffness matrix of each element and assembles it into the global matrix in symmetric banded format.

BAR : Calculation of the stiffness matrix for the BAR element.

CST : Calculation of the stiffness matrix for the CST element.

QUAD4 : Calculation of the stiffness matrix for the QUAD4 and QUAD5 elements. The Gaussian integration scheme was adapted from Cook [7].

SHAPEF : Calculation of the strain-displacement matrix for the QUAD4 and QUAD5 elements; adapted from Cook [7].

REL : Calculation of the interface stress compatibility conditions. It computes the matrix [L] according to the formulation of section 2.2.

YOUNG : Calculation of the elasticity matrices for the cases of plane-stress/plane-strain and axisymmetry.

GREDUC : Reduction of the stiffness equation to upper triangular form using the Gauss procedure. The symmetric banded format is used; adapted from Hinton and Owen [14].

BAKSUB : Calculation of the reaction and displacement vectors through back substitution. The symmetric banded format is used; adapted from Hinton and Owen [14].

DISPL : Calculation of the interface nodal displacements and output of the results for displacements and reactions.

FORCE : Calculation of the axial force and the elongation for the BAR element.

STR : Calculation of the stresses for the CST element.

STRES : Calculation of the stresses for the QUAD4 and QUAD5 elements.

PRINC : Calculation of the principal stresses and their directions.

TRANSF : Transformation of the stresses into the local n-t interface coordinate system (Fig. 2.2).

MATMAT : Calculation of the product of two matrices.

MATVEC : Calculation of the product between a matrix and a vector.

DOT : Calculation of the dot product between two vectors.

Subroutines TRIAX, QUADAX, SHAPAX, RELAX, STRAX, STRIAX are the axisymmetric counterparts of the plane subroutines CST, QUAD4, SHAPEF, REL, STR, STRES, respectively.

Further modifications to the program are possible in order to improve its efficiency, since the primary concern during the present development was the correct implementation of the proposed method rather than the cost efficiency. The listing presented in Appendix B.1 corresponds to the version adapted to the FUJITSU FORTRAN compiler at the Computing Center of McGill University.

Chapter 3

VALIDATION TESTS FOR TWO-DIMENSIONAL ELEMENT

3.1 Introduction

The evaluation of a new element usually follows a long procedure of tests to assess its performance in reference to such criteria as accuracy of results, convergence characteristics, numerical stability, etc. In the present study, priority was given to the assessment of the correctness of the computed results, especially at bimaterial interfaces , since this was essentially the motivation for the development of the interface element. Moreover, a series of patch tests that were successfully carried out using the new element guarantee convergence to exact solutions. A typical arrangement used in a patch test is shown in Fig. 3.1. In the remainder of this chapter, the evaluation of the interface element will be made in reference to the quality of the computed results only .

The formulation of the interface element presented in the previous chapter ensures continuity of the stress vector at the intermaterial boundary. The purpose of the validation tests described in this chapter

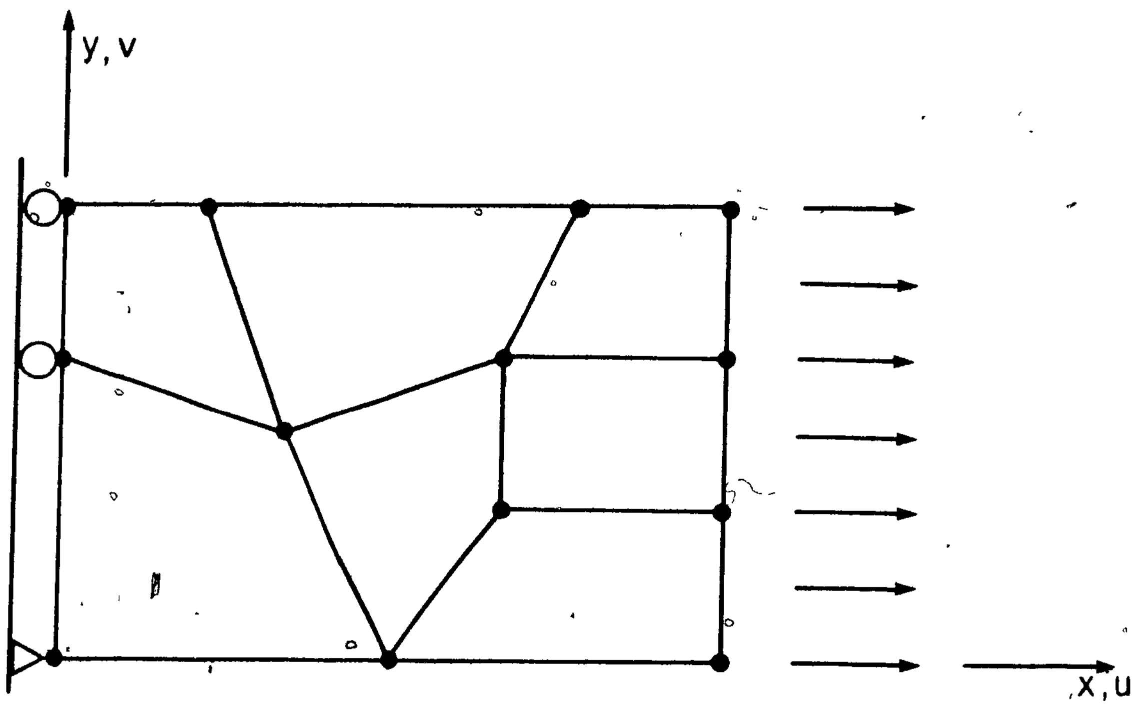

Figure 3.1: Typical arrangement for a patch test.

is to establish the degree of accuracy of the interfacial stress values as computed by the new element. A comparison is also made between the performance of the proposed element and that of the conventional linear method.

The determination of the correct stress distribution at a bimaterial interface by theoretical procedures is not always possible. In some simple problems, however, existing or easily derivable closed-form solutions allowed for a direct comparison with the results of the interface element.

Four validation tests are presented in this chapter. The first one (section 3.2) involves a bimaterial vertical wedge under tip loading. Although a closed-form solution is not directly available for this case, in a similar analysis, Salama and Utku [29] have used the theoretical solution developed for a homogeneous structure to estimate the correct stress distribution at the bimaterial interface. This assumption is further investigated in section 3.2. The remaining validation tests involve a composite cantilever beam under vertical end load, an infinite plate with a circular inclusion under uniaxial tension, and an annular composite disk under internal pressure, presented in sections 3.3, 3.4, and 3.5, respectively. In the three latter cases, theoretical closed-form solutions are used to validate the results obtained from the new element.

Furthermore, in order to compare between the performance of the proposed element and that of the conventional linear quadrilateral, the validation test problems were also analysed by the SAP-IV (Bathe et al. [4]) finite element program using linear quadrilateral elements (element type 4 in SAP-IV). Although the developed program also includes that element type, as was mentioned in the previous section, preference was given to the SAP-IV program due to its superior cost-efficiency, in view of the planned successive refinements. Furthermore, SAP-

IV was chosen over the other major finite element program available at McGill, MSC/NASTRAN [20], because NASTRAN's quadrilateral element (the QUAD4) is not a direct implementation of the standard isoparametric procedure; reduced order integration is used for the shear terms in order to improve performance in bending problems (MacNeal [19]).

Since the validation tests do not reflect real cases, the units of displacement and loading are arbitrary. The units of stress, however, are consistent and are defined as: units of load/(square of units of displacement).

In all the problems, the elastic moduli between adjacent materials are assigned in a ratio of 100. This value is comparable to the one encountered in biomechanical applications (modulus of steel prosthesis = 200,000 Mpa, modulus of bone cement = 2000 Mpa). Furthermore, wherever mesh refinements have been carried out, the following convention has been adopted in order to achieve consistency : the sides of all elements along the interface are reduced by half, while the aspect ratios are maintained constant.

3.2 Composite wedge under vertical load

This example is, in the unimaterial case, an application of the classic problem of a force acting on the end of a wedge (Timoshenko and Goodier [35]). Salama and Utku [29] have used the wedge problem (for horizontal loading) to demonstrate the capabilities of their proposed generalized best fit strain tensors method and they assumed the theoretical solution (developed for a one-material case) to apply in the bimaterial case as well, in order to validate their results.

In this example, mesh refinement is used in an attempt to establish a more reliable stress distribution and the theoretical one-material solution is included in the comparison in order to investigate the validity of assumptions such as the one made by Salama and Utku.

The meshes used for the analysis are shown in Fig. 3.2. They were analysed by both SAP-IV and the program using the interface element. The bimaterial interface is horizontal,and the material at the bottom is 100 times stiffer (hard) than the material at the top (soft). The Poisson's ratios of both materials are 0.3 while the wedge thickness is of unity. Only the symmetric half of the wedge is analysed. The load is a vertical concentrated downwards unit force acting at the tip of the wedge.

The results at the interface are shown in Fig.3.3 and 3.4, for the normal and shear stresses respectively. In these plots, the unique interface stressses obtained by the new element are seen to be identical to the stresses obtained by SAP-IV on the soft side of the interface. Furthermore, the stress discontinuity obtained by SAP-IV was very small in this case, so that a single mesh refinement yielded very reasonable values.

For both normal and shear stresses, the performance of the new element was very satisfactory as its results followed the convergence trend. The analytical one-material solution could be considered as an acceptable first estimate of the interface stress distribution in this case, although its use for validation of interface elements is questionable.

Generally, in this case, the new element yielded a continuous normal and shear interface stress distribution that can be considered reasonable, in view of the convergence pattern. Nevertheless, no distinct superiority to SAP-IV can be claimed here, mainly because of the lack of pronouced discontinuity in the SAP-IV values.

Interestingly, if the relative stiffnesses of the materials are reversed, that is, if the bottom material becomes the soft medium and the top material becomes the hard medium, then the unimaterial analytical so-

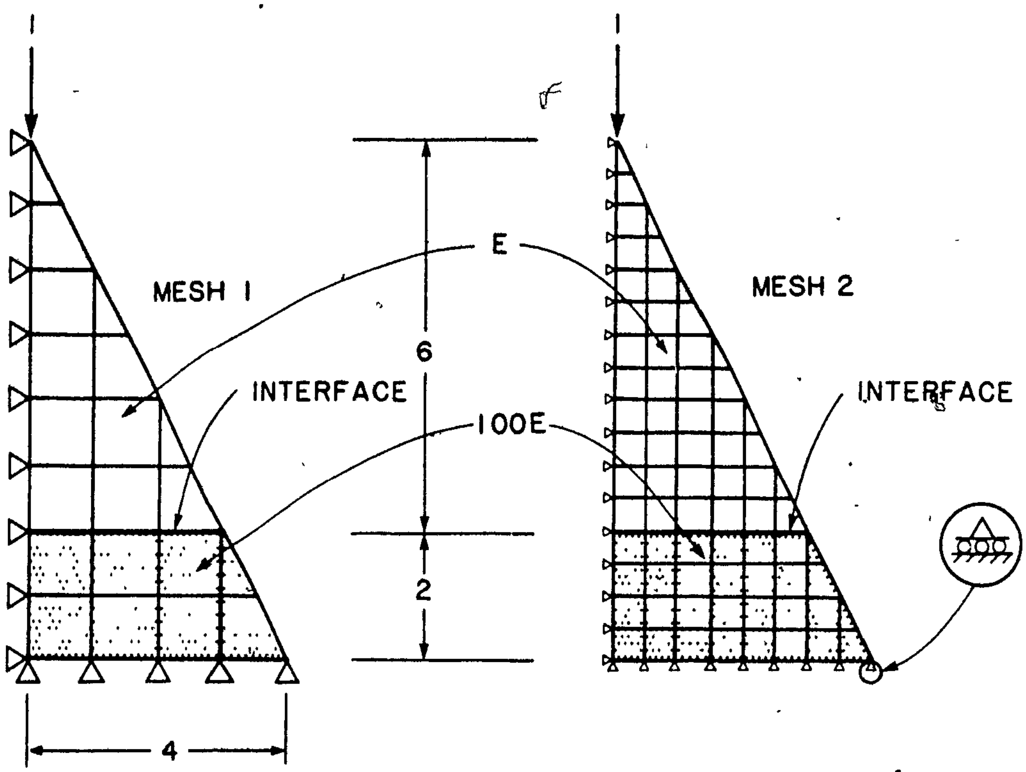

Figure 3.2: Composite wedge problem; loading and meshes used.

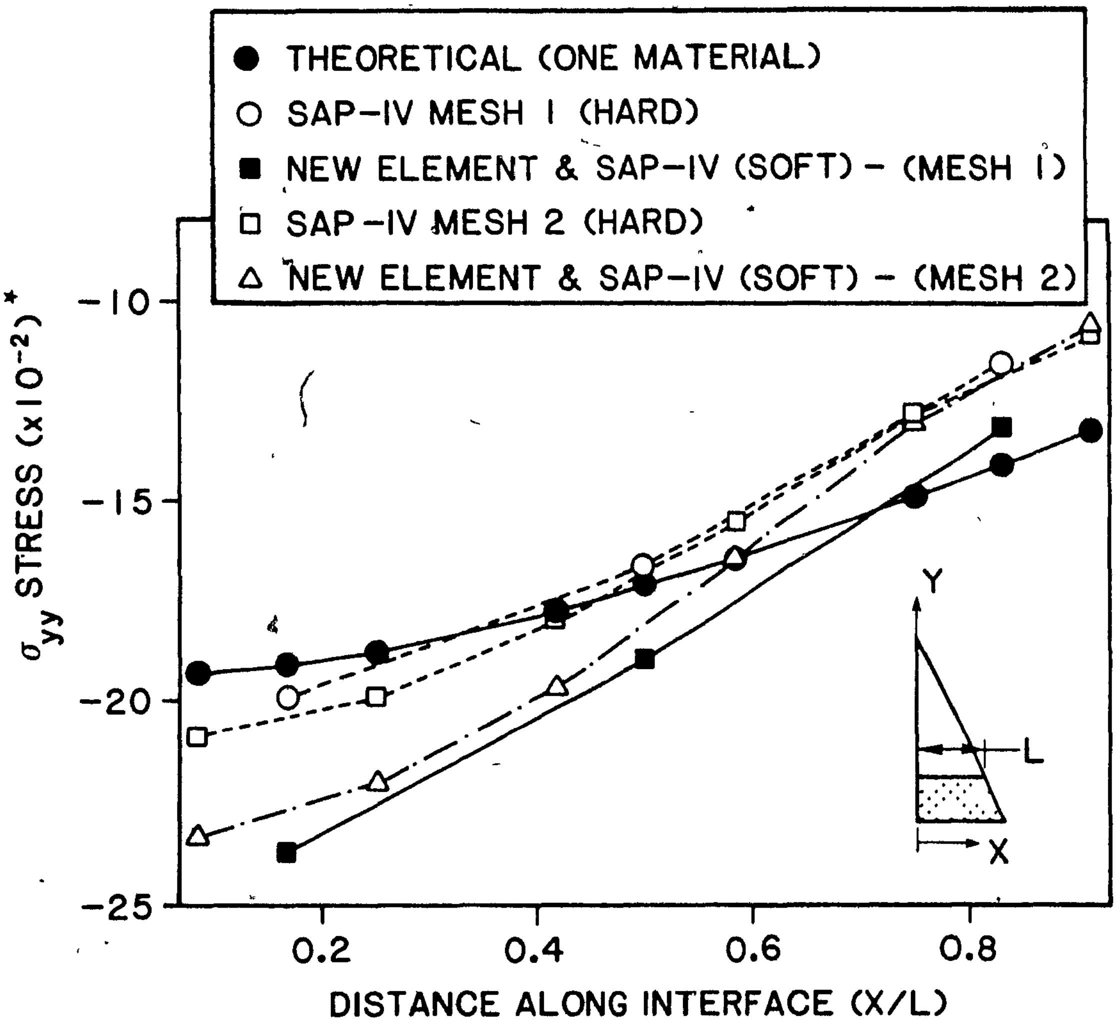

Figure 3.3: Composite wedge problem; normal stresses at the bimaterial interface. * the stress units are consistent: force units/square of length units.

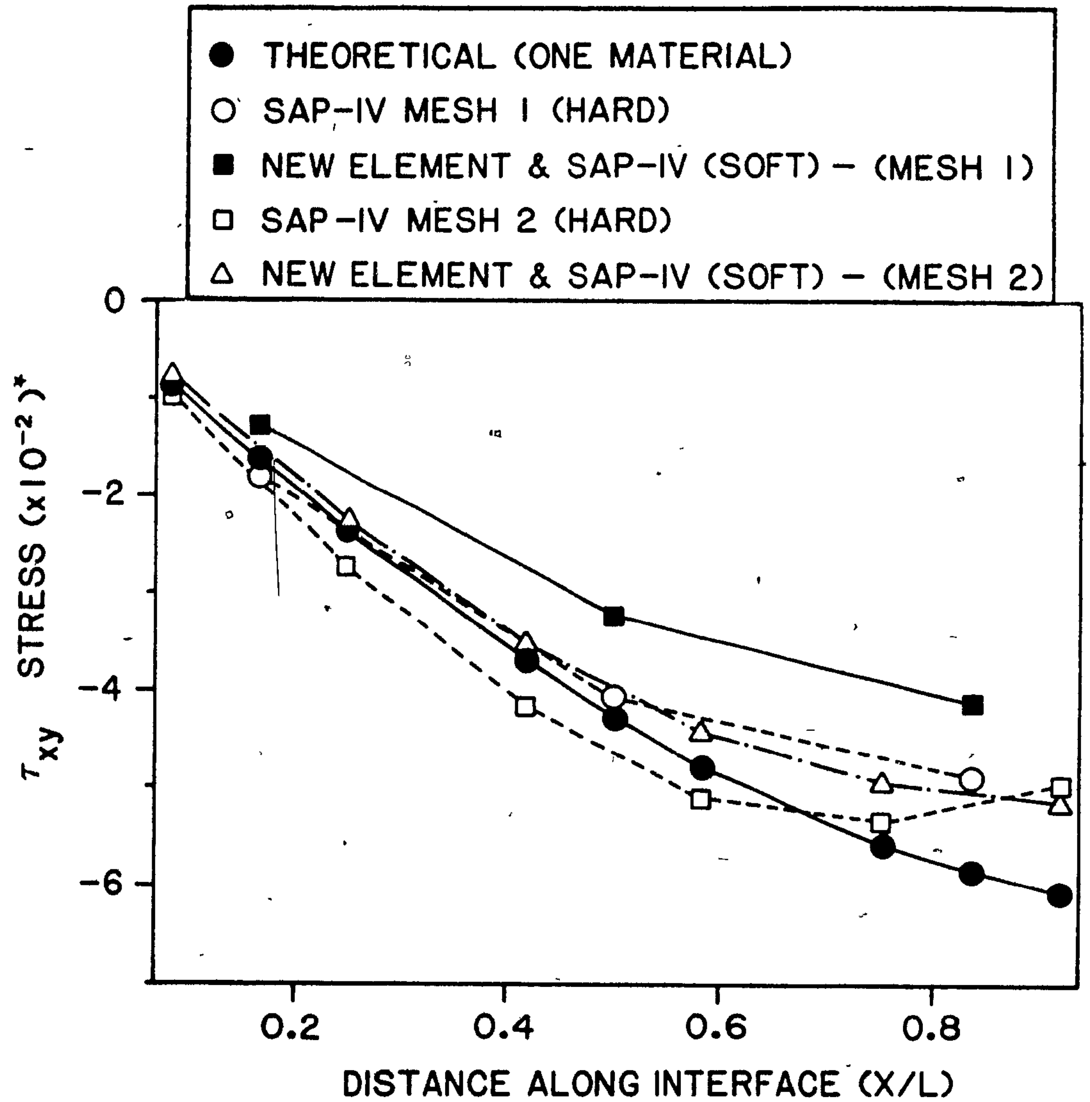

Figure 3.4: Composite wedge problem; shear stresses at the bimaterial interface. * the stress units are consistent: force units/square of length units.

lution can no longer be considered an acceptable approximation of the true interface stress distribution, as shown in Fig. 3.5. This demonstrates the fact that, in the absence of exact solutions to problems involving bimaterial interfaces, extreme care must be taken before deciding to consider the unimaterial solution, especially when the region of interest is bimaterial interface itself.

3.3 Bending of a composite cantilever beam

The second test case is shown in Fig. 3.6 along with the three different meshes used. It is a cantilever beam of two materials loaded at its tip by a concentrated vertical unit force. The length of the span is L distance units, the depth is L/4 units, and the thickness is one unit. The bimaterial interface is along the length of the beam, the stiffer material (E = 30000 stress units) being at the bottom. Both materials have a Poisson's ratio of 0.3. Mesh 2 was analysed by both SAP-IV and the developed program, while meshes 1 and 3, coarser and finer respectively than mesh 2, were analysed by SAP-IV only in order to observe convergence.

A general solution to the problem of bending of composite prismatic bars has been developed by Muskhelishvili [23]. Based on that solution, a plane stress solution for the case of cantilever bimaterial beams of rectangular cross-sections has been obtained and is included in Appendix A.2. According to this solution, in the problem at hand, the transverse stress should be zero throughout the beam, while the shear stress should have a constant value of -5.5×10^{-2} stress units.

In the finite element analysis of the problem the load was applied at the tip rather than being parabolically distributed along the edge and all the nodes on the fixed end were constrained,rather than fixing only

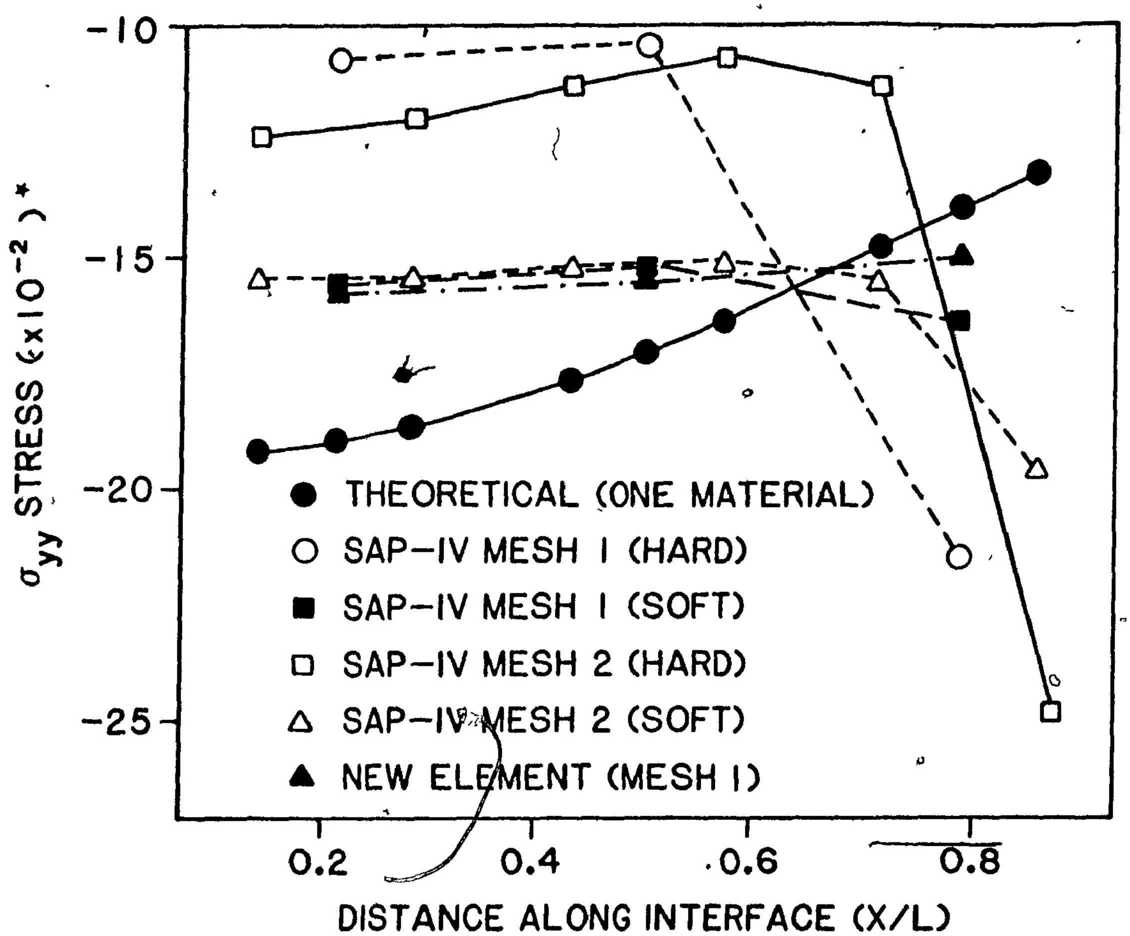

Figure 3.5: Composite wedge problem - relative stiffnesses reversed; normal stresses at the bimaterial interface. * the stress units are consistent: force units/square of length units.

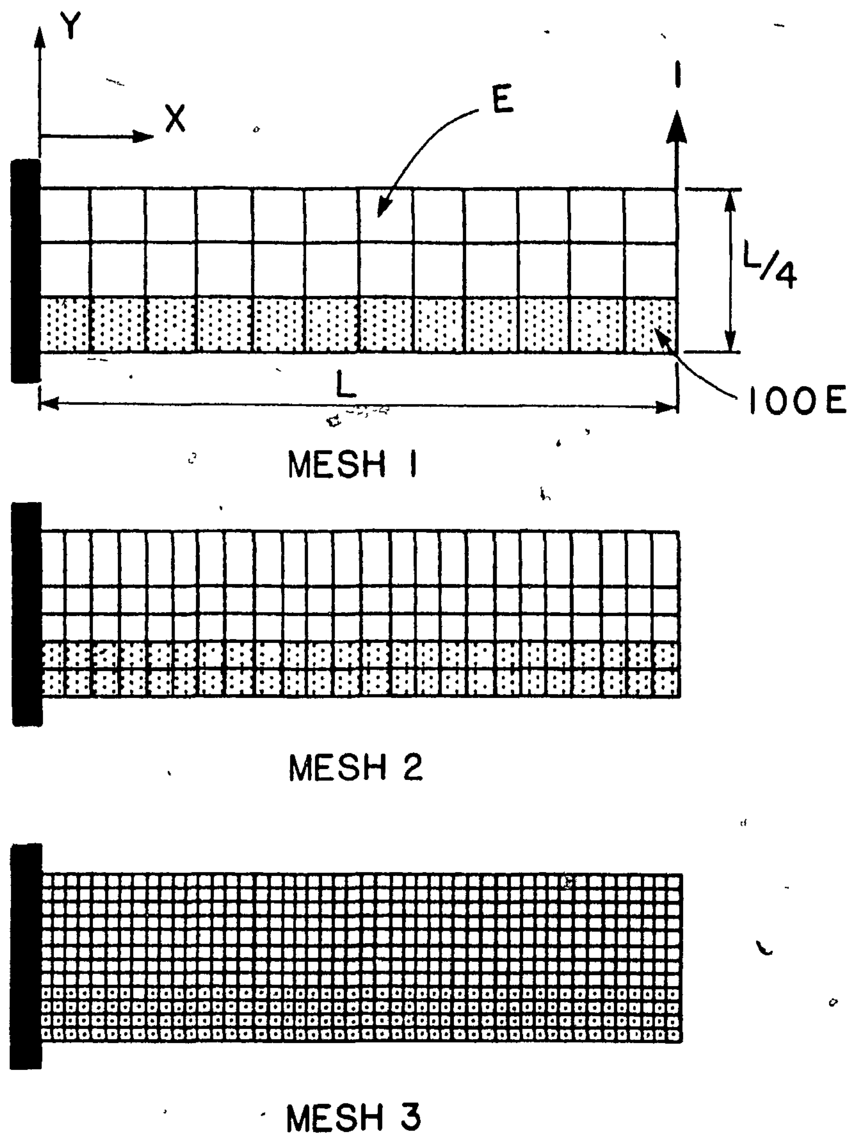

Figure 3.6: Composite cantilever beam problem; loading and meshes used.

the end of the neutral axis, as was assumed in the theoretical solution. It is expected that these simplifications will result in local deviations from the anticipated results at the two ends of the beam, so that the results in these regions will not be considered in the comparison between theoretical and finite element solutions.

Figs. 3.7 and 3.8 show the transverse stress distributions obtained at the bimaterial interface. Because of the very high discontinuity in the SAP-IV values across the interface (over two orders of magnitude in difference), separate plots using different scales were necessary for the hard side, Fig. 3.7, and the soft side, Fig. 3.8. In view of the fact that the theoretical normal stress has a constant value of zero units, it appears that mesh refinement does little to improve the high values obtained by SAP-IV on the hard side of the interface. The soft side stresses, however, are more reasonable. The new interface element predicted unique values on both sides of the interface, although shown on different scales in Fig. 3.7 and 3.8. These unique values are almost exact,and the superiority of the new element is therefore evident.

The interface shear stress distribution is shown in Figs. 3.9 and 3.10. In this case too, different scales had to be used for the SAP-IV results on the hard side, Fig. 3.9 and on the soft side of the interface, Fig. 3.10. The discontinuity obtained by SAP-IV is significant, although not as pronouced as in the case of the normal stresses. Still, the stresses predicted by SAP-IV on the hard side fail to show an indication of convergence: they remain approximately three times higher than the correct distribution even after the last refinement. On the other hand, the soft side results of SAP-IV are again very reasonable. The new element predicted unique shear stress values which are almost identical to the ones computed by SAP-IV on the soft side of the interface, for the same mesh size. These values are very close to the correct stress distribution.

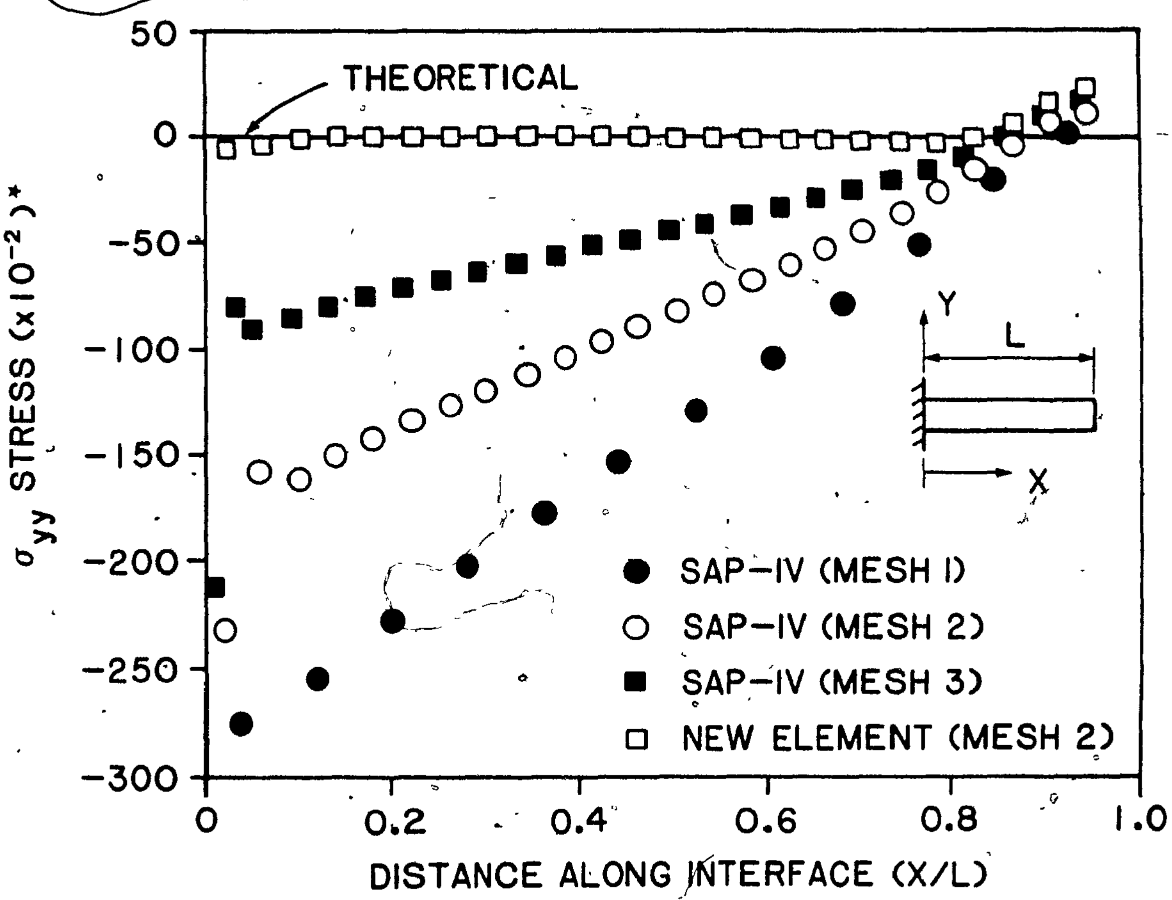

Figure 3.7: Composite cantilever beam problem; normal stresses on the hard side of the bimaterial interface. * the stress units are consistent: force units/square of length units.

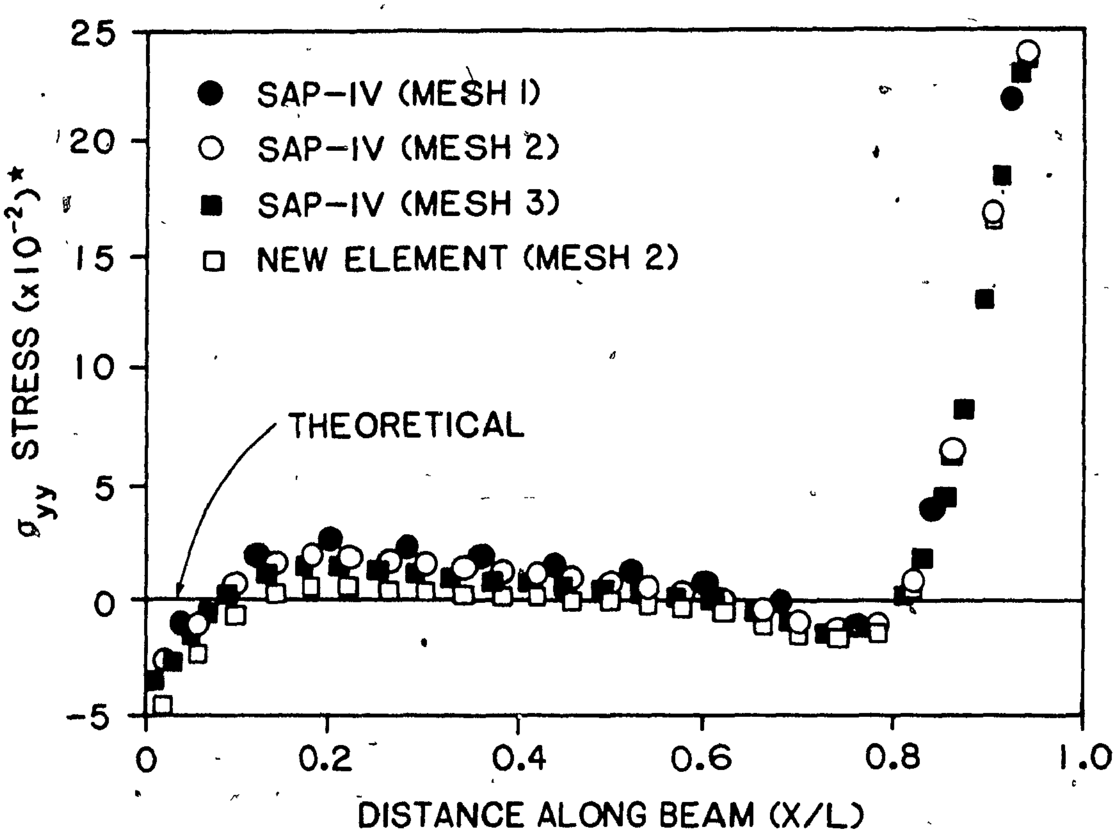

Figure 3.8: Composite cantilever beam problem; normal stresses on the soft side of the bimaterial interface. * the stress units are consistent: force units/square of length units.

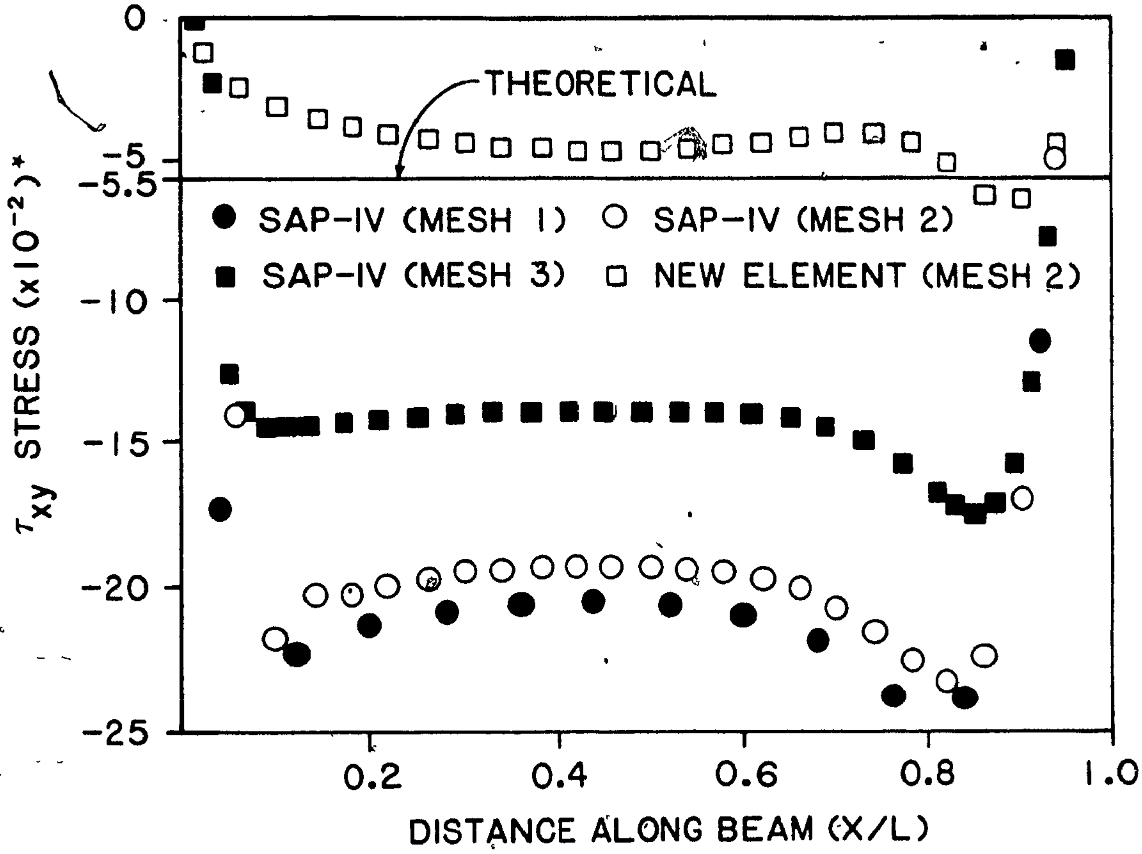

Figure 3.9: Composite cantilever beam problem; shear stresses on the hard side of the bimaterial interface. * the stress units are consistent: force units/square of length units.

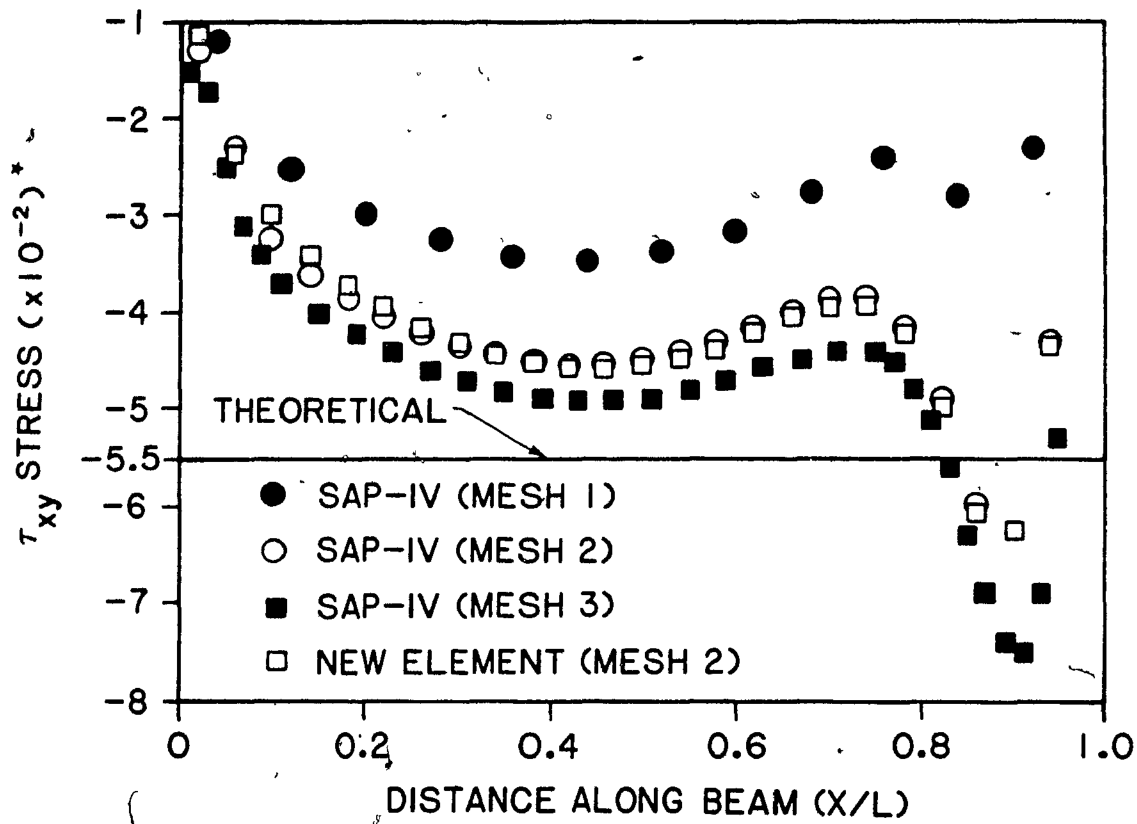

Figure 3.10: Composite cantilever beam problem; shear stresses on the soft side of the bimaterial interface. * the stress units are consistent: force units/square of length units.

The new element enforces continuity of the stress vector at a bimaterial interface, but it does not affect the stress component which is not required to be continuous. In this problem, the bending stress, σ_{xx}, must be discontinuous at the interface. Fig. 3.11 shows the variation of the bending stress along the depth of the beam at three locations along the cantilever span. Both SAP-IV and the interface element predicted identical values for the same mesh size. Furthermore, the use of the new element along the bimaterial interface did not affect the bending stresses above and below the interface, which are accurately determined and are identical to the ones obtained by SAP-IV.

The maximum vertical displacement of the structure (14.3×10^{-2} units) was computed satisfactorily by both SAP-IV (14.9×10^{-2} units) and by the program using the new element (14.6×10^{-2} units) for mesh 2. Small differences between the analytical and the finite element methods are probably attributable to the different assumptions made in the two methods concerning load application and boundary conditions. It must be noted here that if interface stresses are the important variables in an analysis by the conventional finite element method, convergence should not be judged on the basis of displacements alone, since it has been seen in this example that although displacements are accurately determined by SAP-IV, the interface stresses are highly discontinuous.

Finally, if the order of the materials is reversed, that is if the hard material is at the top and the soft material is at the bottom, then the interface stress discontinuity obtained by SAP-IV was reduced, but still remained relatively unaffected by mesh refinement. The new element, again, yielded values very close to the theoretical ones and comparable to those obtained by SAP-IV on the soft side of the interface.

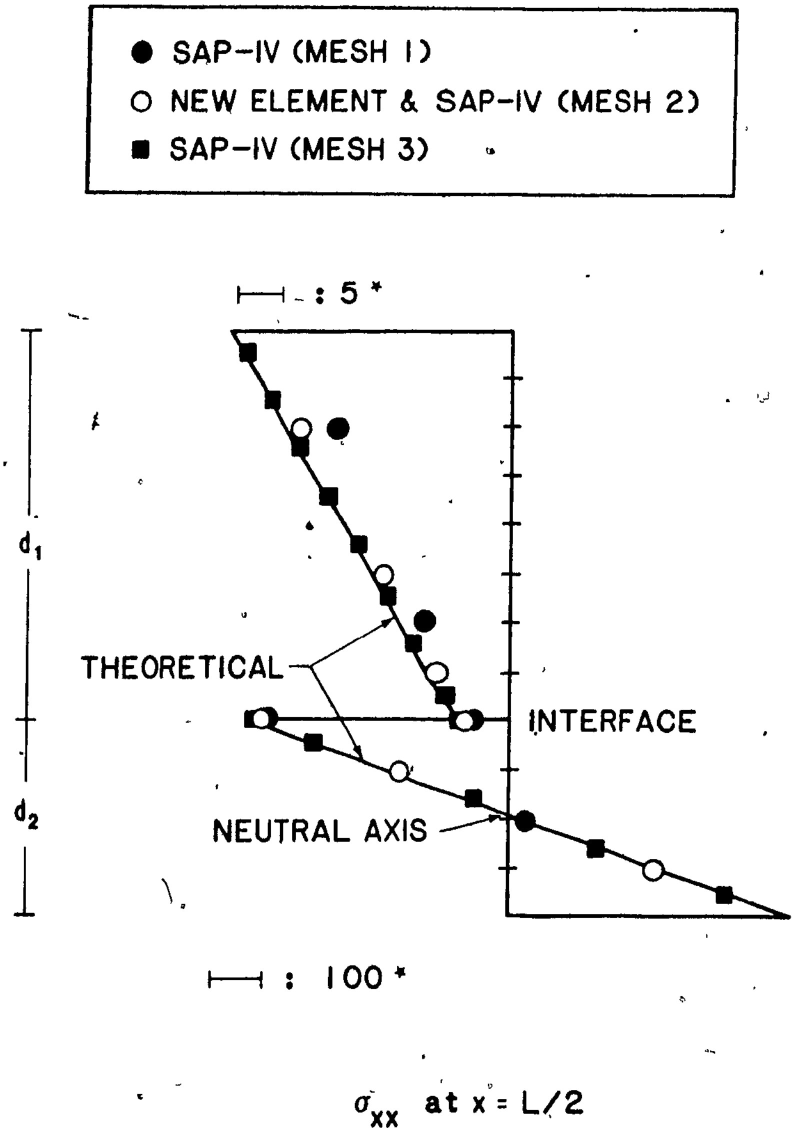

Figure 3.11: Composite cantilever beam problem; variation of bending stresses along the span and the depth of the beam. * the stress units are consistent: force units/square of length units.

3.4 Plate with circular inclusion

The problem of an infinite plate with a circular inclusion under uniaxial tension, Fig. 3.12a, has been solved in Muskhelishvili's classic work [23]. The example used for this third validation test is based on that problem and is shown in Fig. 3.12b. Only one quarter of the plate was modelled, since the structure and the loading have two perpendicular axes of symmetry. The inclusion was assigned a modulus of elasticity of 300 stress units and a Poisson's ratio of 0.25. The surrounding plate was taken to be 100 stiffer than the inclusion and was assigned a Poisson's ratio of 0.333. The structure was subjected to uniform unit tension.

The model was analysed by both SAP-IV and the program using the new element. The resulting interface normal and shear stress distributions are plotted in Fig. 3.13 and 3.14 respectively, together with the theoretical distributions. The discontinuity obtained by SAP-IV for both normal and shear stress components is evident. It is not as pronoüced, however, as in the previous example of the cantilever beam.

The normal and shear stresses calculated by SAP-IV on the soft side of the interface are very close to the theoretical distribution. One the other hand, the stresses obtained on the hard side are significantly higher in magnitude. The unique interface values obtained by the new element are very similar to the theoretical values and to the values computed by SAP-IV on the soft side.

An important observation that can be made in this example is that in the case of the shear stresses, the theoretical distribution lies outside the range defined by the distributions obtained by SAP-IV on the soft and hard sides of the interface. The same observation was made in a subsequent analysis of the same model with reversed relative stiffnesses, that is, with hard inclusion and soft plate. The distribution of the

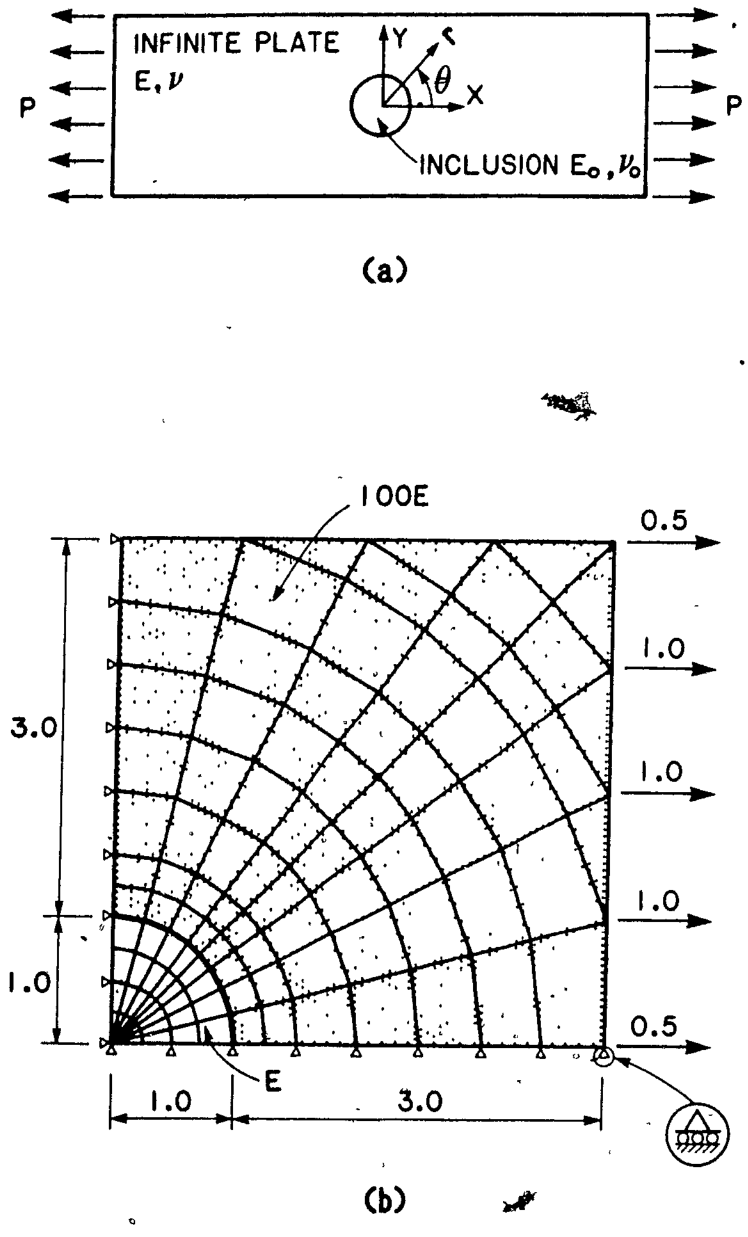

Figure 3.12: a) Problem of plate with circular inclusion; b) finite element mesh and loading used.

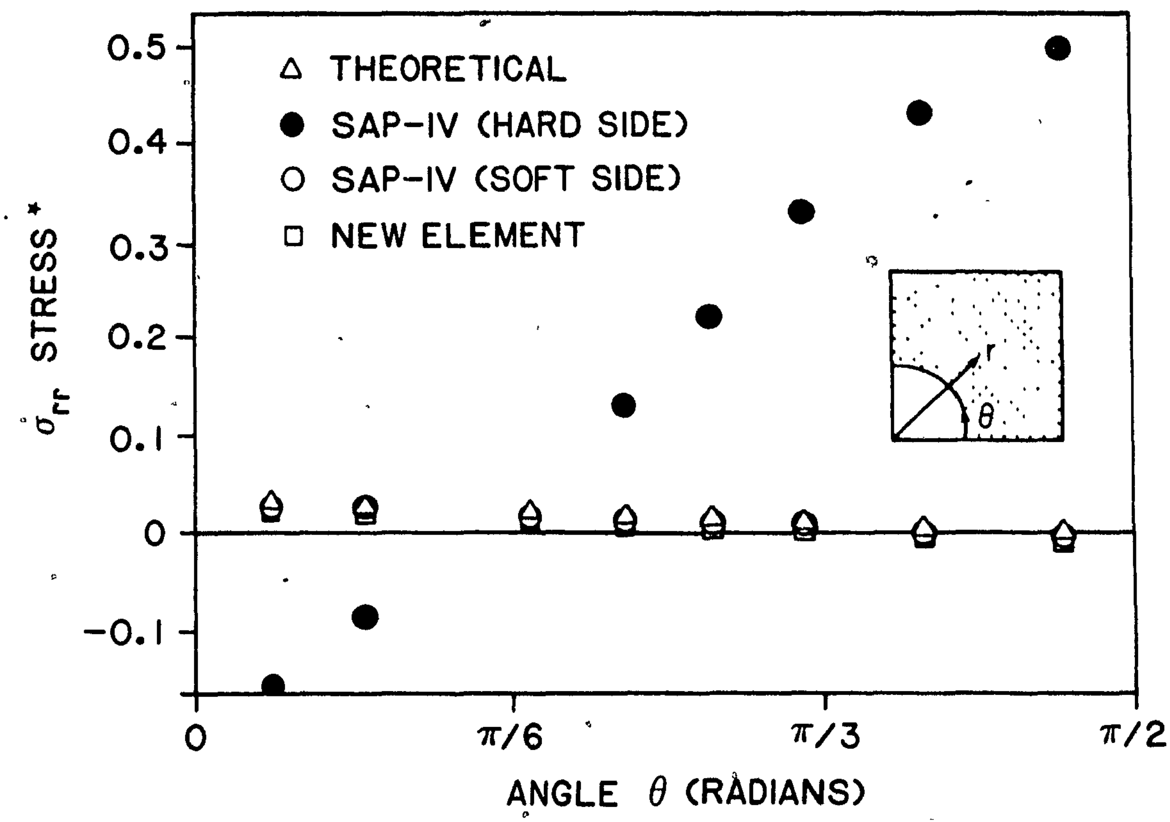

Figure 3.13: Plate with circular inclusion; normal (radial) stresses at the bimaterial interface. * the stress units are consistent: force units/square of length units.

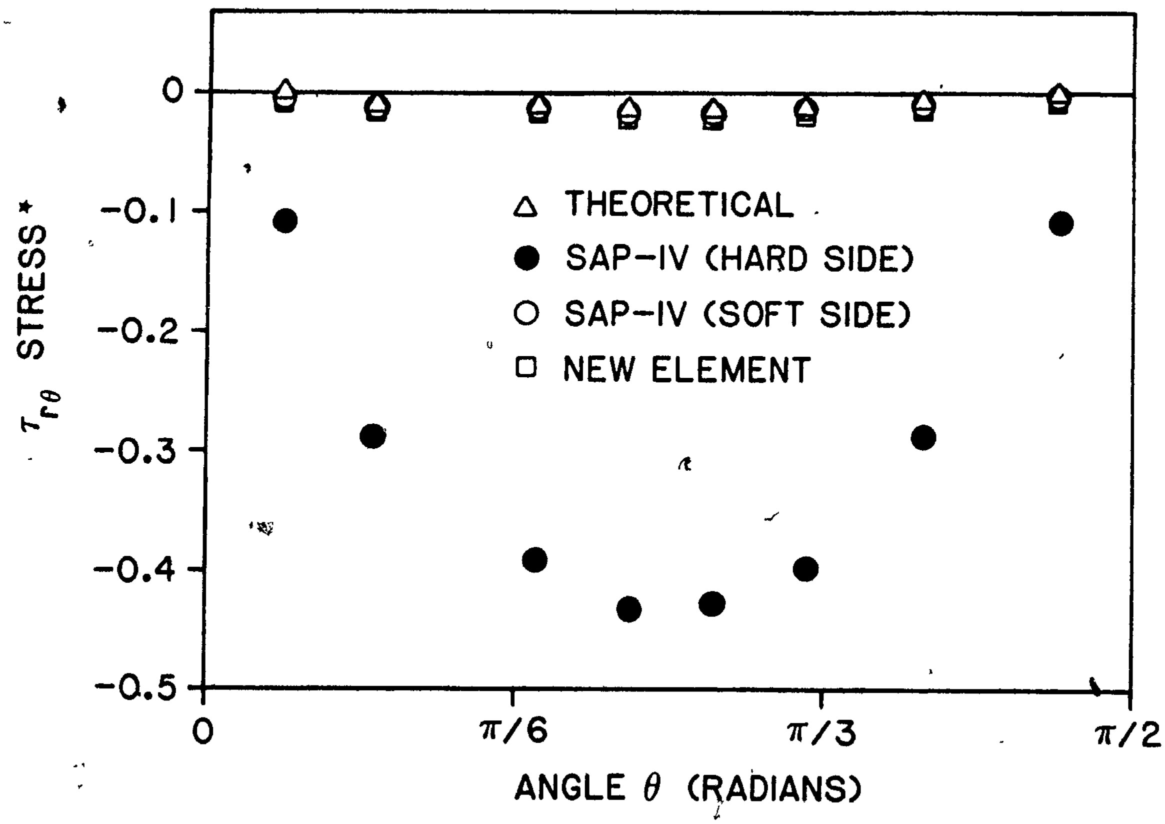

Figure 3.14: Plate with circular inclusion; shear stresses at the bimaterial interface.
* the stresses are consistent: force units/square of length units.

interface normal stresses in that analysis is shown in Fig. 3.15. It can be seen in Fig. 3.15 that the stess discontinuity obtained by SAP-IV was significantly reduced (approximately 20 per cent difference between the hard side and the soft side values). Interestingly, the stresses on the hard side of the interface were closer to the theoretical solution than those on the soft side.

3.5 Composite annular disk with internal pressure

This fourth validation test was used to assess the performance of the axisymmetric interface element, formulated in Section 2.3. The problem consists of an annular disk of unit thickness composed of two materials (Fig. 3.16a). The inner annulus was assigned an elastic modulus of 30000 stress units and a Poisson's ratio of 0.2. The outer annulus was 100 times softer than the inner annulus and had a Poisson's ratio of 0.0. The load applied was a uniform internal unit pressure.

An exact plane stress solution may be obtained for this problem, based on Lamé's work on the unimaterial case (e.g. Timoshenko and Goodier [35]). The developed solution is outlined in Appendix A.1. According to that solution, the exact interface radial stress in the problem at hand has a value of 0.004 pressure units.

The problem was analysed by both SAP-IV and the developed program using the new element. The finite element mesh is shown in Fig. 3.16b. Table 3.1 summarizes the result from these analyses as well as the theoretically expected results. It is evident from Table 3.1 that SAP-IV fails to approximate the correct interface radial stress (0.004 units) by computing 0.14 units on the hard side and 0.0028 units on the soft side. On the other hand, the new element yields a unique radial stress value (0.0038 units) that is very close to the exact value. Although the mesh used in this analysis is very coarse, the point that

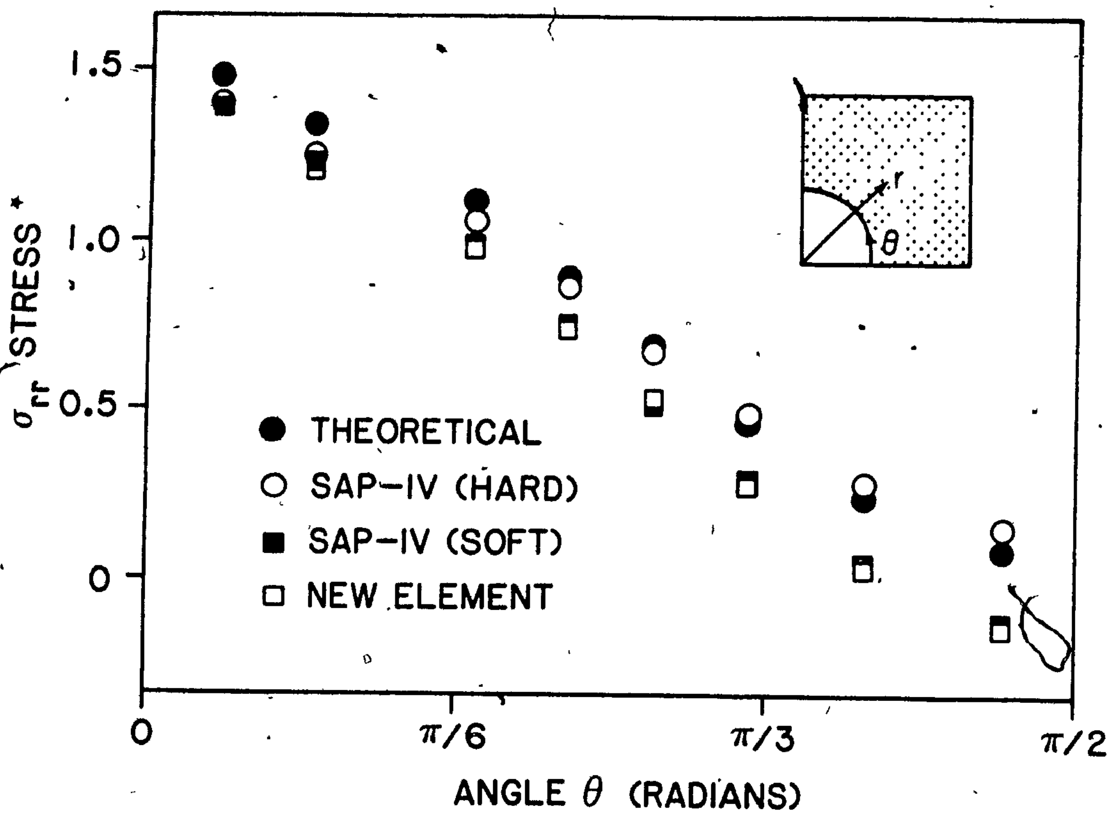

Figure 3.15: Plate with circular inclusion - relative stiffnesses reversed; normal (radial) stresses at the bimaterial interface. * the stress units are consistent:force units/square of length units.

is made is that, nevertheless, the new element computed an almost exact radial stress value.

The other interface stress and strain components that are included in Table 3.1 indicate that the use of the new element does not affect those stess and strain components that can be discontinuous $(\sigma_{\theta\theta}, \varepsilon_{rr})$ while it permits the $\varepsilon_{\theta\theta}$ component to be continuous at the interface.

If the order of the materials is reversed, that is if the inner annulus is softer, then the discontinuity obtained by SAP-IV is reduced; however, the new element still predicts an interface radial stress value that is closer to the theoretical one.

3.6 Concluding remarks

In view of the foregoing, several conclusions may be drawn concerning the perfomance of the new element and that of the conventional linear one. Enforcing interface normal and shear stress continuity in the formulation of the element yields very reliable stress distributions at interfaces where the adjacent materials differ by as much as 100 times in elastic moduli. These distributions satisfy interelement equilibrium (at an interface point) and have been validated by comparison to existing theoretical solutions.

Despite the fact that it cannot be claimed that the interface element is always superior to the linear quadrilateral element, it was found that its performance is very consistant, thus always assuring the user of a reliable stress analysis. On the other hand, the stress discontinuity obtained by the conventional finite element method is highly dependent on material properties of adjacent media and on structure geometry, loading, and size of mesh used.

The importance of the interface element is further enhanced in view of the observed fact that the exact solution is not necessarily bounded

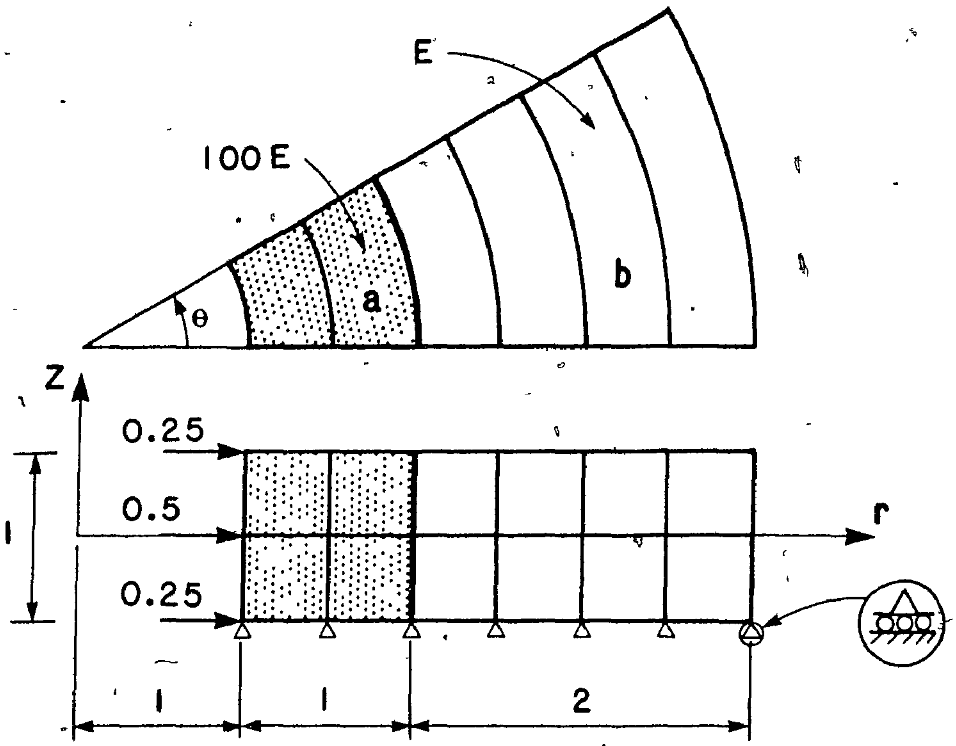

Figure 3.16: a) Problem of composite annular disk with internal pressure; b) finite element mesh and loading (per radian) used.

Quantity	Theoretical	Numerical Value	
		SAP-IV	New Element
σ_{rr}^{a}	0.0040	0.1400	0.0038
σ_{rr}^{b}	0.0040	0.0028	0.0038
$\sigma_{\theta\theta}^{a}$	0.6601	0.6224	0.6870
$\sigma_{\theta\theta}^{b}$	0.0066	0.0065	0.0068
ϵ_{rr}^{a}	-4.27×10^{-6}	5.17×10^{-7}	-4.45×10^{-6}
ϵ_{rr}^{b}	8.87×10^{-6}	4.92×10^{-6}	8.28×10^{-6}
$\epsilon_{\theta\theta}^{a}$	2.20×10^{-5}	1.98×10^{-5}	2.29×10^{-5}
$\epsilon_{\theta\theta}^{b}$	2.20×10^{-5}	2.17×10^{-5}	2.26×10^{-5}

Table 3.1: Composite annular disk problem; results at the bimaterial interface.

by the distributions obtained by the conventional method on either side of the interface. An improvement over the performance of the conventional linear element can, therefore, be claimed in cases where the interface stress discontinuity, obtained by programs such as SAP-IV, is high, thus rendering impossible the prediction of the range of the correct stress distribution. In such cases, the traditional practices of averaging or even using weighting techniques across bimaterial interefaces are likely to have unpredictable results.

More tests are certainly needed before this proposed element can be unconditionally guaranteed. Early indications are, however, that it can be very useful in analyses requiring the calculation of the stress distribution at a bimaterial interface. A suggested procedure at this stage would be an initial analysis by the conventional method on a relatively coarse mesh and, if significant discontinuities are detected, a subsequent analysis by the proposed element on a locally refined mesh.

Chapter 4

BIOMECHANICAL APPLICATION: STRESS ANALYSIS OF A PROSTHETIC TIBIA MODEL

4.1 Introduction

Fixation of a deformed or fractured human joint by the insertion of an artificial prosthesis is an important and frequent orthopaedic surgical procedure. It consists of surgically removing the damaged joint and then fixing in its place an artificial prosthesis. The procedure is called total joint replacement. Ideally, the prosthesis should be designed so as to transmit the load in a manner simulating the physiological stress distribution in the intact joint. Fulfillment of this objective is desirable in order to avoid the possible adverse reaction of the host bone receiving the prosthesis.

Another important design concern is that of the mechanical fixation of the prosthesis into the bone, as it directly effects the functional longevity of the prosthetic implant. These design considerations become very important in cases of joints transmitting significant loads, for example the knee and hip joints. The objective of the present chapter is to demonstrate the applicability of the newly developed stress-compatible finite elements in determining the prosthesis interface stresses in an axisymmetric model of the tibial part of the knee joint.

A resurfaced knee joint is shown schematically in Fig. 4.1. Several types of fixation systems and prosthesis designs have been proposed and are being presently used in knee resurfacing operations. The tibial component of the prosthesis, which is relevant to the present analysis, typically consists of a metal plate with one or more stems protruding from it. Although the single stem design has been favoured by several researchers (Lewis et al. [17], Reilly et al. [27], Bartel et al. [3]), it has the disadvantage of 'tilting' under unsymmetric loadings (Lewis et al. [17]). More recent studies seem to favour the multiple stem design (Eftekhar [8], Cheal et al. [6]).

Now, as regards fixing the prosthesis to the bone, there are two techniques of achieving it. The first consists in interposing a thin layer of PMMA (Polymethylmethylacrylate) bone cement between the stem(s) and the bone and thus providing the necessary bond. Precoating is applied to the metal surface to improve the bond strength (Ahmed et al. [1]). In the second and more recent technique, the bone is allowed to grow in a 'weaving' fashion on and around the surface of the stem (or stems) possessing special porous coating. The disadvantage of the cemented fixation lies mainly with the fact that it is susceptible to early loosening of the prosthesis. There is an often reported link between clinical failure of the system and loosening of the prosthesis because of the cement fixation failure (e.g. Lewis et al. [17]). The bone ingrowth method offers a stronger fixation but requires a longer immobilisation period for the patient and complicates revision surgery (subsequent

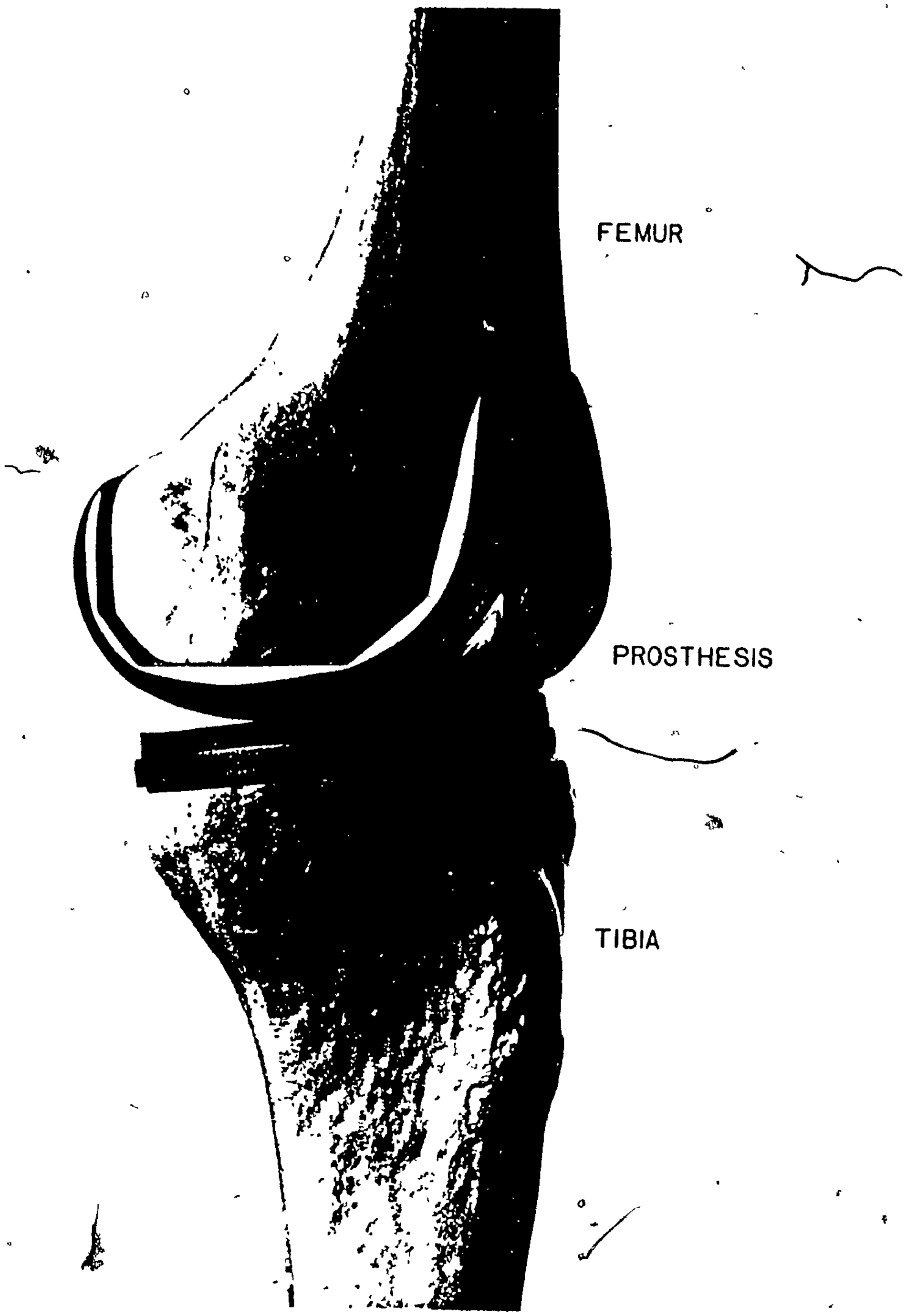

Figure 4.1: Schematic representation of a knee joint after total knee arthroplasty.

operation for replacement of the prosthesis) by making the removal of the implant from the bone more difficult.

Thus, in order to improve the performance of the cemented fixation, it appears imperative to achieve a strong bond between the cement and the prosthesis, and also between the cement and the bone, and to ensure that the stresses at these interfaces are below the bond strengths of the joined materials. Therefore, the determination of the correct prosthesis/cement interface stress distribution is an important initial step in the design evaluation of an artificial joint.

The objective of the present work is to establish the general validity of previous analyses in view of the uncertainty of stress computation by the displacement finite element method at interfaces between dissimilar materials (as is the case at the bone/cement and metal/cement interfaces). Furthermore, the effect of mesh refinement on the interfacial stress discontinuity obtained by conventional finite element programs (such as the SAP-IV program) is also investigated. With reference to the above-mentioned objectives, it will be possible to assess the applicability of the developed interface element to real-case problems and its potential contribution to the quality of the stress analyses in biomechanical applications.

4.2 Present finite element analysis model

The specific objective of the present analysis is to evaluate the performance of the displacement based finite element method and that of the new element with respect to the stress distribution at the implant/cement interfaces of a prosthetic tibia model. In view of this objective two types of finite element analyses were performed on an axisymmetric model of a prosthetic tibia: one, using the SAP-IV program and refined meshes, and two, by using the developed program incorporating the new element.

The model chosen for the analysis is an axisymmetric representation of a prosthetically resurfaced proximal tibia. This choice was dictated by the geometric and loading characteristics of the upper tibia , which can be qualitatively approximated by an axisymmetric formulation. Also at the time when this biomechanical application was carried out, the program incorporating the three-dimensional version of the interface element had not yet been completed. Therefore the analysis had to be limited to an axisymmetric case.

4.2.1 Geometry

The geometry used in the finite element model is the same as that used by Shirazi-Adl and Ahmed [30], obtained from in-vitro measurements, Fig. 4.2.

Only the proximal 40 mm of the tibia is included in the model, since the effects of the insertion of the prosthesis on the stress distribution are localised proximally. The choice of a prosthesis model was obvious, in view of the axisymmetric requirements : a circular horizontal plate with a central cylindrical stem. The thickness of the plate is taken to be 2 mm and its radius is equal to 32 mm. The plate lies entirely on the cancellous bone and no load is directly transferred to the cortical shell. This is a 'safe' design assumption in view of the fact that in surgery it is very difficult to achieve and guarantee the contact of the prosthesis with the cortical shell. The length of the stem has been found to be critical in the load transfer mechanism and in the minimization of the proximal bone and cement stresses; longer stems are reported to be more desirable (Murase et al. [21]). For this reason, a 25 mm long stem is used. The thickness of the cement layer is taken to be 3 mm all around the prosthesis for the purpose of this analysis (this thickness

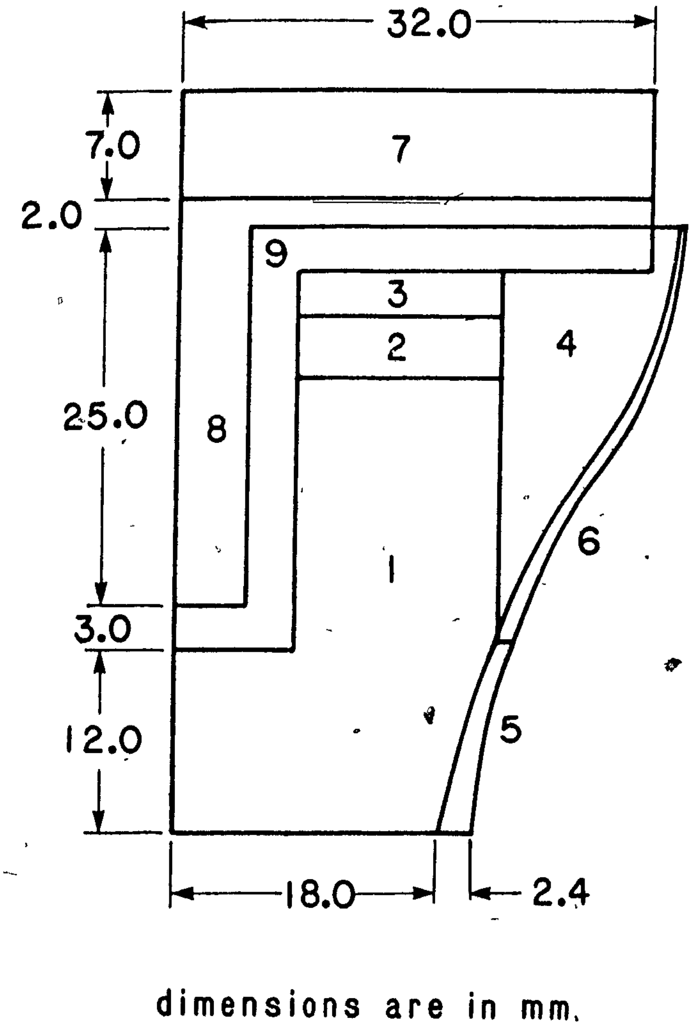

Figure 4.2: Geometrical dimensions of the axisymmetric prosthetic tibia model in a radial plane, and the distribution of different material regions. The numbers in the material regions refer to properties listed in Table 4.1.

Material		E-modulus	Poisson's	Reference
number*	type	(MPa)	ratio	
1	cancellous bone	50	0.2	Goldstein et al. [10]
2	cancellous bone	100	0.2	Goldstein et al. [10]
3	cancellous bone	150	0.2	Goldstein et al. [10]
4	cancellous bone	300	0.2	Goldstein et al. [10]
5	cortical bone	14000	0.3	Murray et al. [22]
6	cortical bone	7000	0.3	Murray et al. [22]
7	UHMWP	1000	0.35	Parker [24]
8	stainless steel	200000	0.3	Popov [25]
9	PMMA bone cement	2000	0.3	Haas et al. [11]

* numbers refer to Fig. 4.2

Table 4.1: Material properties used in the tibia model. The numbers refer to material regions shown in Fig. 4.2.

conforms to the value used by Lewis et al. [17]).

4.2.2 Material regions and properties

With reference to Fig. 4.2, the material regions indicated by numbers correspond to the properties listed in Table 4.1. The distribution of the material properties within the cancellous bone is in accordance with the results of Goldstein et al. [10]. All the materials are assumed to be linear elastic and isotropic.

4.2.3 Finite element meshes

The model of the resurfaced tibia is analysed by SAP-IV and by the developed program. For a consistent comparison, the same finite element mesh is used in both cases and is shown in Fig. 4.3. The elements used are isoparametric axisymmetric 'ring' elements with quadrilateral and triangular radial cross-sections. The 'quadrilateral' elements have a linear displacement variation along their edges, while the 'triangular' elements are based on a constant strain formulation. In the analysis by the developed program, interface ring elements are used. The interface elements were used to model the cement/implant interface. These elements have a quadrilateral cross-section and their formulation has been described in Section 2.3. As was mentioned in Chapter 2, the use of interface elements does not require additional degrees of freedom, so that the mesh involved in the analysis by the developed program was identical to the one used for the SAP-IV analysis.

In all, the model consists of 854 nodes and 887 axisymmetric elements, of which 162 have triangular cross-sections and 725 have quadrilateral cross-sections. In the analysis by the developed program, 55 pairs of quadrilateral elements combining to form 55 interface elements were used. The use of triangular elements was limited to areas of mesh

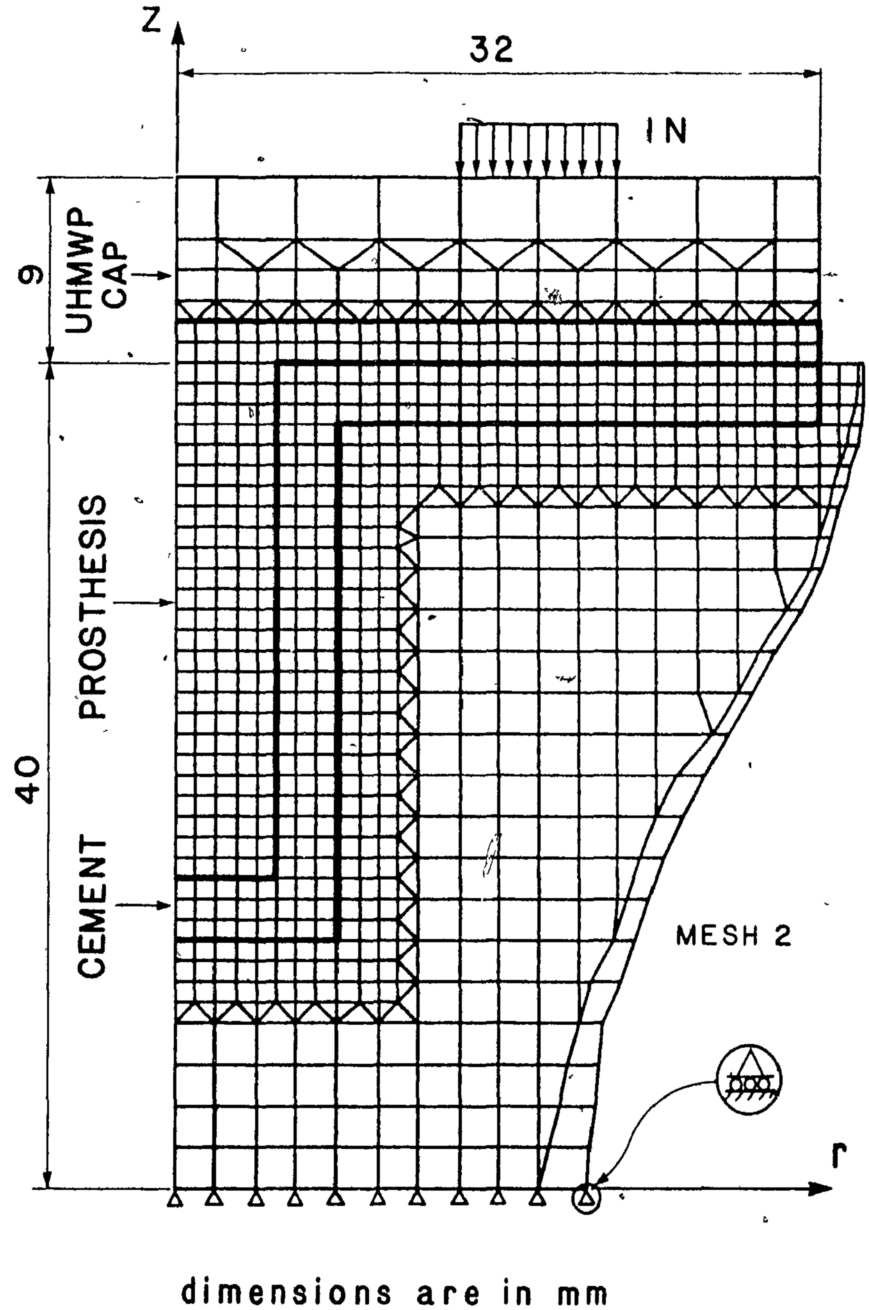

Figure 4.3: Axisymmetric finite element model of a resurfaced tibia model; mesh 2.

gradation and sharp geometrical changes, since their performance is considered to be inferior to that of the quadrilateral elements. Care was taken to maintain an aspect ratio (defined as the ratio of the longest element dimension to the shortest one) of approximately one and a rectangular shape for most quadrilateral elements; exceptions were unavoidable in the regions in and near the cortical shell, where, for the present purposes accuracy in stress computations is not considered to be of prime importance. The mesh is more refined around the prosthesis/cement interface; this is the region of present interest and the elements used here are small squares of 1 mm side.

4.2.4 Boundary conditions

The boundary conditions imposed consisted in fixing the distal end of the tibia against longitudinal (vertical) movement. Since only the proximal 40 mm of the bone (corresponding to the interest area of the joint) have been modelled, the above-mentioned boundary condition implies that the load is expected to be transferred along the longitudinal (axial) direction of the shaft in the diaphyseal (midshaft) region of the tibia. This is, indeed, the observed load bearing mechanism in the tibia. In accordance with the axisymmetric condition, nodes on the axis of symmetry were fixed against radial movement.

4.2.5 Loading

The model was analysed for a reference vertical compressive load of 1.0 N uniformly distributed over an annular ring of inner and outer radia encompassing an area approximately equal to 900 mm^2. Although a more realistic loading condition would involve a symmetrical loading (simulating the contact of the femoral condyles) or even a non- symmetrical one (in the case of single condyle contact), the formulation of the SAP-IV program, as well as that of the developed program, could not accommodate a non-axisymmetric loading. The choice of a total load of 1 N allows for easy interpretation of the results in real cases by a single multiplication of the obtained stress values by the real load value in Newtons. The load is applied through a 7 mm thick Ultra-High Molecular Weight Polyethylene (UHMWP) articular surface that lies on top of the horizontal plate of the prosthesis (Eftekhar [8], Shirazi-Adl and Ahmed [30]).

4.3 SAP-IV analyses with mesh refinement

Despite the fact that mesh refinement is a powerful and usually the only tool for validating finite element results, it has rarely been used in biomechanical applications (Huiskes and Chao [15]). In the present study, the results of mesh refinement by the conventional displacement method (SAP-IV) are used to establish reliable interface stress distributions in order to compare the results obtained on one hand by using the SAP-IV program and on the other by using the developed program.

Two new meshes were created, one coarser and the other finer than the one shown in Fig. 4.3 and considered as the 'optimum' one. The convergence study was limited to the computation of stresses at or around the horizontal interface region, as it was considered the most critical region because of the presence of stress discontinuities. Therefore, only this area was chosen for the mesh refinement.

The three different meshes used are shown in Fig. 4.4. The intermediate mesh (mesh 2) has been described in section 4.2. The coarse mesh (mesh 1) consists of 596 nodes and 606 axisymmetric elements. Both the metal plate and the cement layer are modelled by only one element across their respective thicknesses, keeping the aspect ratio 1.0 for the prosthesis elements and 1.5 for the cement elements.

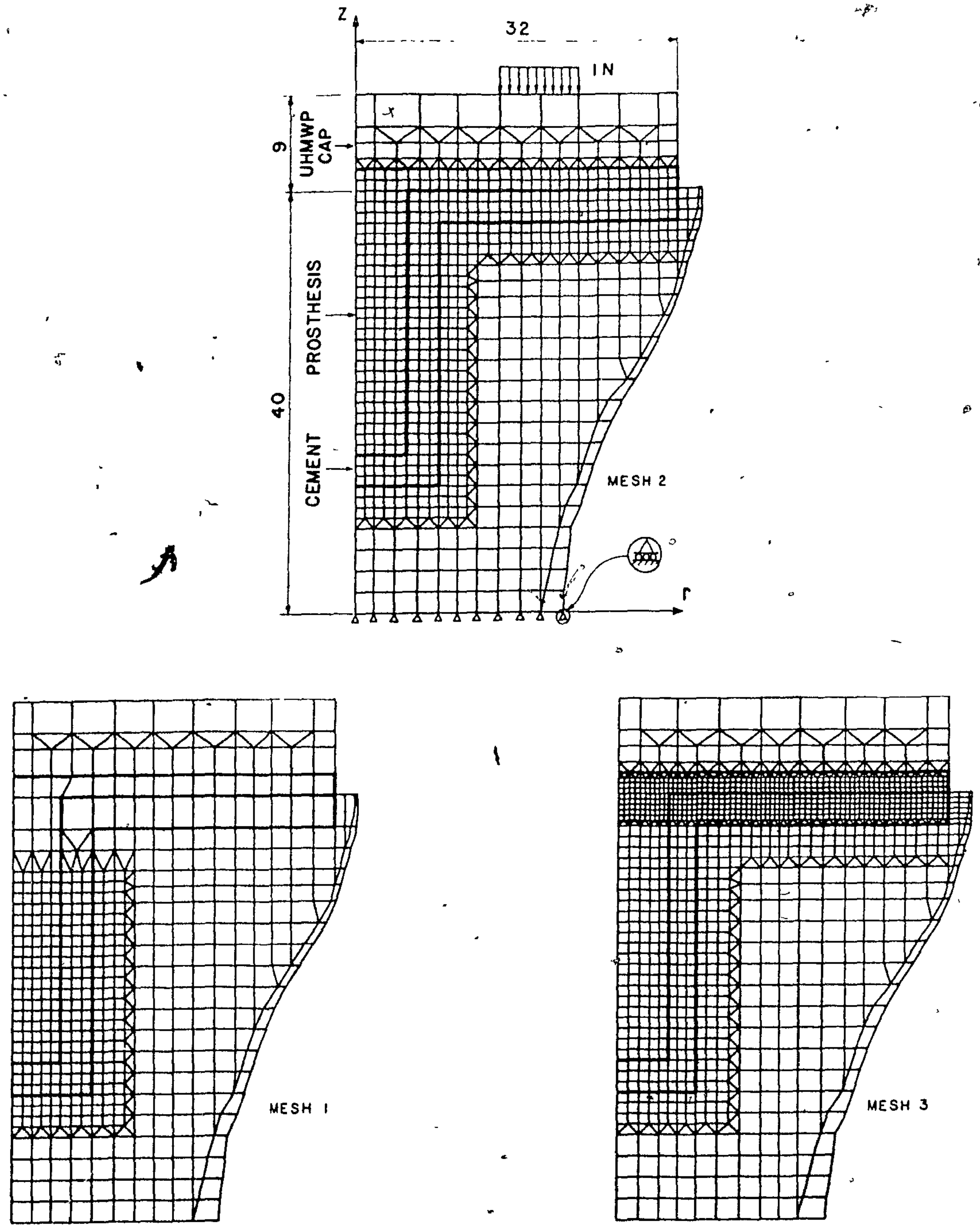

Figure 4.4: Axisymmetric finite element meshes used in the convergence study.

The fine mesh (mesh 3) consists of 1316 nodes and 1348 axisymmetric elements. There are 4 layers of elements across the plate thickness and 6 layers of elements across the cement thickness. The aspect ratios in the refined area are of unity (the element cross-sections are squares of 0.5 mm sides).

The results from the convergence tests are shown in Figs. 4.5 through 4.8. These figures also include the results of the analysis of mesh 2 by the new element, which will be discussed in the next section.

Figures 4.5 and 4.6 show the distribution of normal stresses on the cement and on on the prosthesis side of the interface respectively, while Figures 4.7 and 4.8 show the distribution of shear stress on the cement and prosthesis sides of the interface. In actual fact, both the normal and shear stresses must be continuous. However, the SAP-IV analysis yielded such highly discontinuous results, that the stresses computed on the two sides of the interface could not be included in the same graph and separate plots with a different scale for each interface side were necessary.

In the normal stress case , Figs. 4.5 and 4.6, in particular, this discontinuity is so severe that the metal side of the interface is computed to be mostly in tension, while the cement one is predicted to be mostly in compression. This illogical discrepancy is not alleviated, even after employing the finest mesh (mesh 3). The metal side tensile stresses along the horizontal interface are clearly improbable in view of the applied vertical compressive load, and this fact accounts for their dramatic reduction from a maximum value of approximately 1.6×10^{-2} MPa/N for mesh 1 to a maximum value of approximately 0.3×10^{-2} MPa/N for mesh 3. The cement side stresses, however, are not significantly affected by changes in mesh size; in fact, the second refinement hardly produces any changes, thus indicating a more stable and probably more accurate stress distribution Excluding the high stresses in

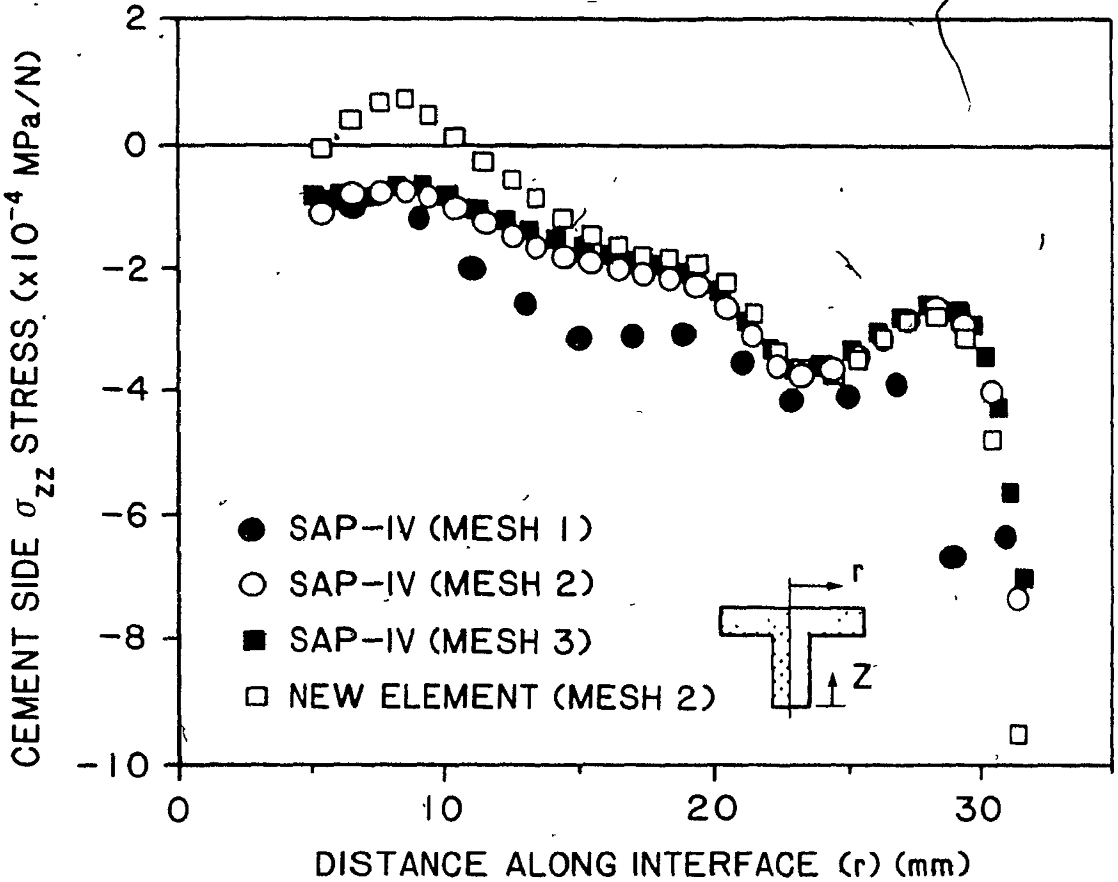

Figure 4.5: Normal stresses at the cement side of the horizontal interface. Results from the analysis of meshes 1, 2, 3 by the SAP-IV program and mesh 2 by the program using the new element.

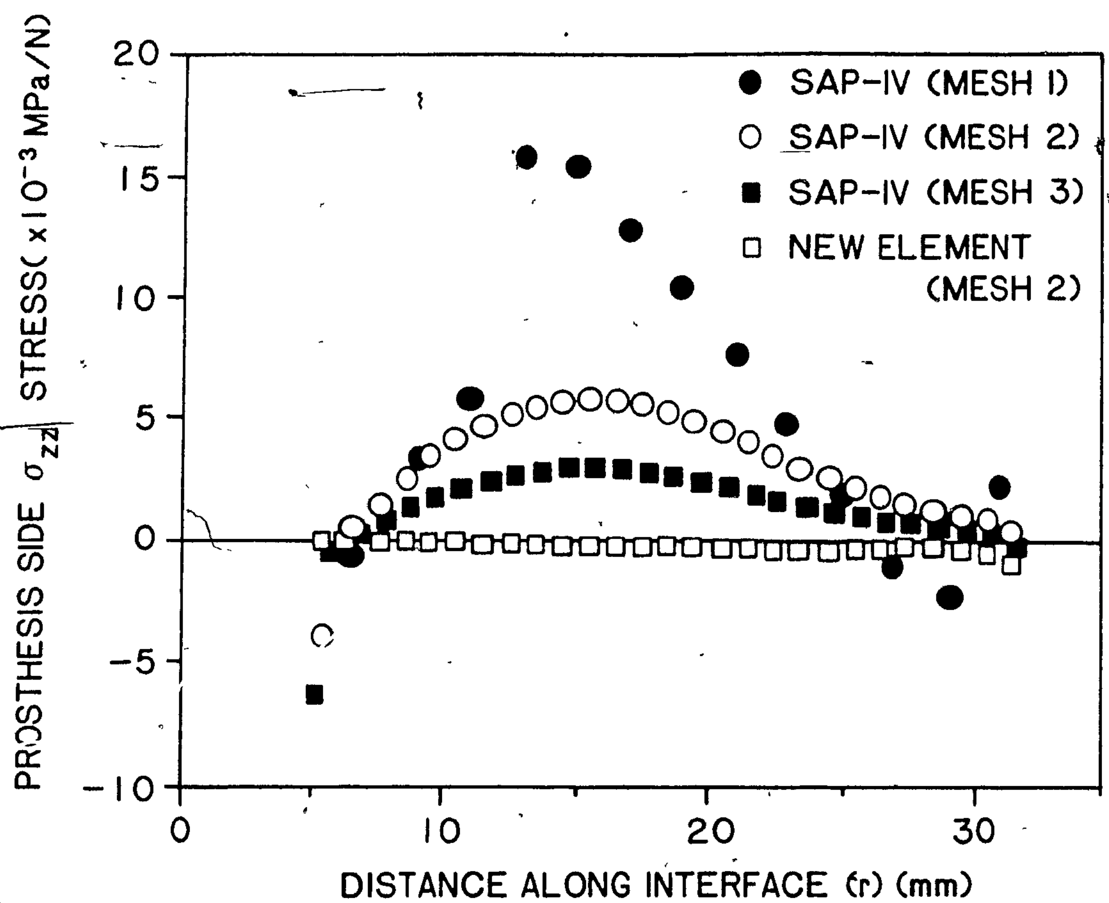

Figure 4.6: Normal stresses at the prosthesis side of the horizontal interface. Results from the analysis of meshes 1, 2, 3 by the SAP-IV program and mesh 2 by the program using the new element.

the edge region of the prosthesis , mesh 1 yielded a maximum compressive stress of approximately 0.41×10^{-3} MPa/N, mesh 2 approximately 0.37×10^{-3} MPa/N, and mesh 3 approximately 0.36×10^{-3} MPa/N.

The discontinuity is not as pronounced in the case of the shear stresses, Figs. 4.7 and 4.8. Qualitatively similar stress distributions are obtained on both sides of the interface. The numerical values are, however, very different, so that different plots had again to be used for the cement, Fig. 4.7, and the prosthesis, Fig. 4.8, sides of the interface. The shear stresses at the horizontal interface alternate between positive values in the first third of the 'cantilever span' and negative values in the remaining part. The maximum negative stress values on the metal side range from approximately -1.7×10^{-3} MPa/N in mesh 1, to -0.9×10^{-3} Mpa/N in mesh 2, to finally -0.6×10^{-3} MPa/N in mesh 3. The corresponding values on the cement side are almost identical for all three meshes (approximately -0.15×10^{-3} MPa/N).

A general conclusion that can be drawn from the mesh refinement studies carried out by the SAP-IV program is that despite the three different mesh sizes used, the stress discontinuity obtained at the cement/metal interface remains significantly high (approximately 900%). An interesting observation , however, was that the convergence on the metal side was very slow compared to that on the cement side. As a matter of fact, the cement side stresses obtained by the analysis of mesh 2 can be considered relatively stable, since they only changed by approximately 3% after the analysis of mesh 3. The consideration of mesh 2 as the 'optimum' one is, therefore, justified and the cement side stresses obtained by SAP-IV can be used as a relative standard for the assessment of the results obtained by the new element.

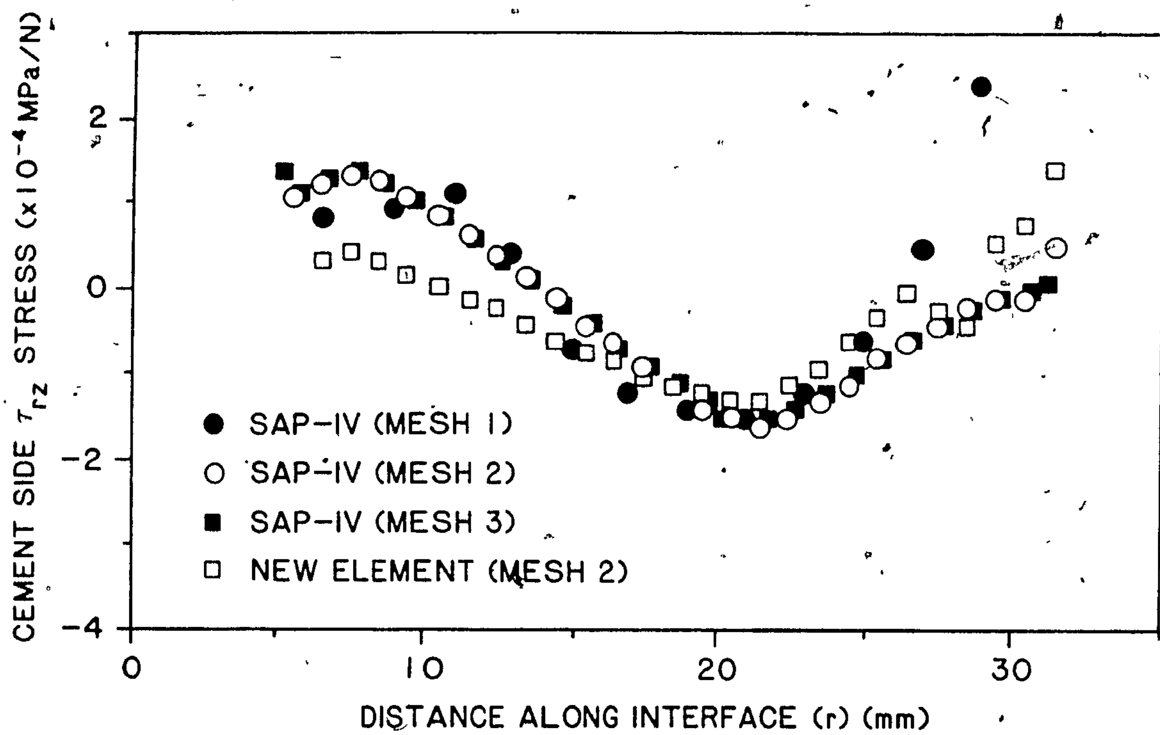

Figure 4.7: Shear stresses at the cement side of the horizontal interface. Results from the analysis of meshes 1, 2, 3 by the SAP-IV program and mesh 2 by the program using the new element.

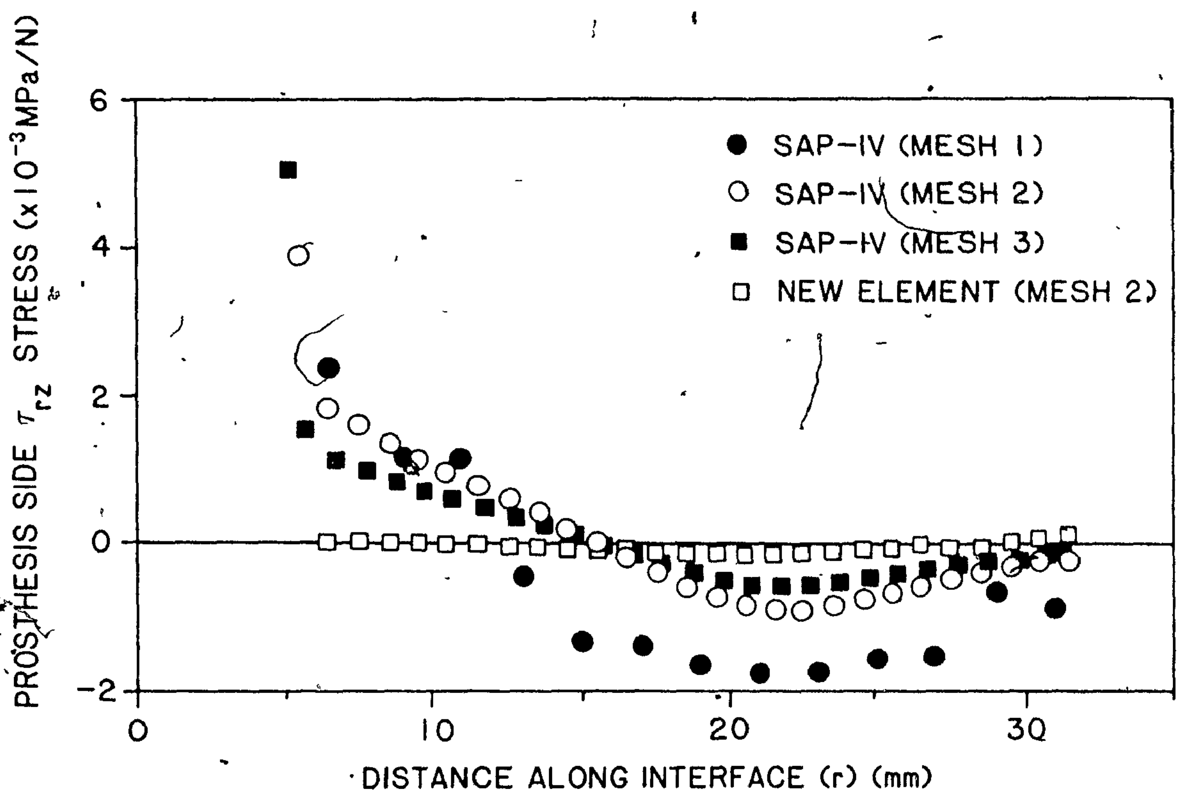

Figure 4.8: Shear stresses at the prosthesis side of the horizontal interface. Results from the analysis of meshes 1, 2, 3 by the SAP-IV program and mesh 2 by the program using the new element.

4.4 Analysis with the new element and comparison with SAP-IV analyses

4.4.1 Comparison with SAP-IV results at the horizontal interface

The unique interface normal and shear stress values obtained from the analysis of mesh 2 using the new interface element are also shown in Figs. 4.5 through 4.8.

From Fig. 4.6 it is evident that the normal stresses predicted by the interface element are very close to those obtained by SAP-IV on the cement side of the interface for the finest mesh, mesh 3. The normal stress distribution is almost identical to the one predicted by SAP-IV (cement side, mesh 3 : 2% difference) for the last two thirds of the horizontal interface. In the first third, however, the interface was found to be in tension, an observation in agreement with that made by Lewis et al. [17] (where experimental evidence from in-vitro studies is also presented).

The shear stress distribution obtained by the new element at the horizontal interface is, again, quite close to the one obtained by SAP-IV on the cement side, using the refined mesh. There is a 3% difference between the maximum negative shear values obtained by the new element and by SAP-IV. The positive shear values predicted by the new element are, however, significantly lower than the respective SAP-IV ones (approximately 65% difference).

Thus, in the present case of the tibial analysis, the new element results (mesh 2) compare favourably with those obtained by SAP-IV on the cement side of the interface provided a more refined mesh (mesh 3) is used.

The unique interface normal and shear stresses obtained by using the interface element are isolated from the previous figures and are shown in Figs. 4.9 and 4.10.

4.4.2 Results at the vertical interface

In the vertical interface (along the stem), both the normal and the shear stress distributions are of great importance because of their direct impact on fixation efficiency. In this problem it is generally expected that normal stresses at the stem/cement region would be small. The results obtained in this analysis, Figs. 4.11 and 4.12, show the vertical interface to be in radial compression over the distal half of the stem, while in the proximal half the radial stresses vary from tensile to compressive and, finally, reach high tensile values near the junction with the horizontal plate, Fig. 4.11. The stress peak at the junction is probably caused by the sharp corner assumed in the model and could possibly be alleviated by a smoother design. This high tensile stress at the tip and the fact that the stress changes from tensile to compressive over the proximal third of the stem suggests that the tip region might be a site for separation (loosening) of the prosthesis depending on the tensile strength of the metal/cement bond. Generally, the quality of interfacial bond developed between metal and cement depends on different factors including metal type, prosthesis surface (precoating, etc.), and insertion procedure (cement curing, etc.). Although a quantitative interpretation of the results is not within the objectives of this study, a comparison of the peak tensile stress obtained under a load of 1 kN (0.36 MPa) with values reported for the tensile bond strength (between 4 and 13 Mpa) (Keller et al. [16]) indicates that the stresses are low.

The shear stress distribution at the vertical interface , Fig. 4.12, tends to increase distally starting from minimal values at the top of

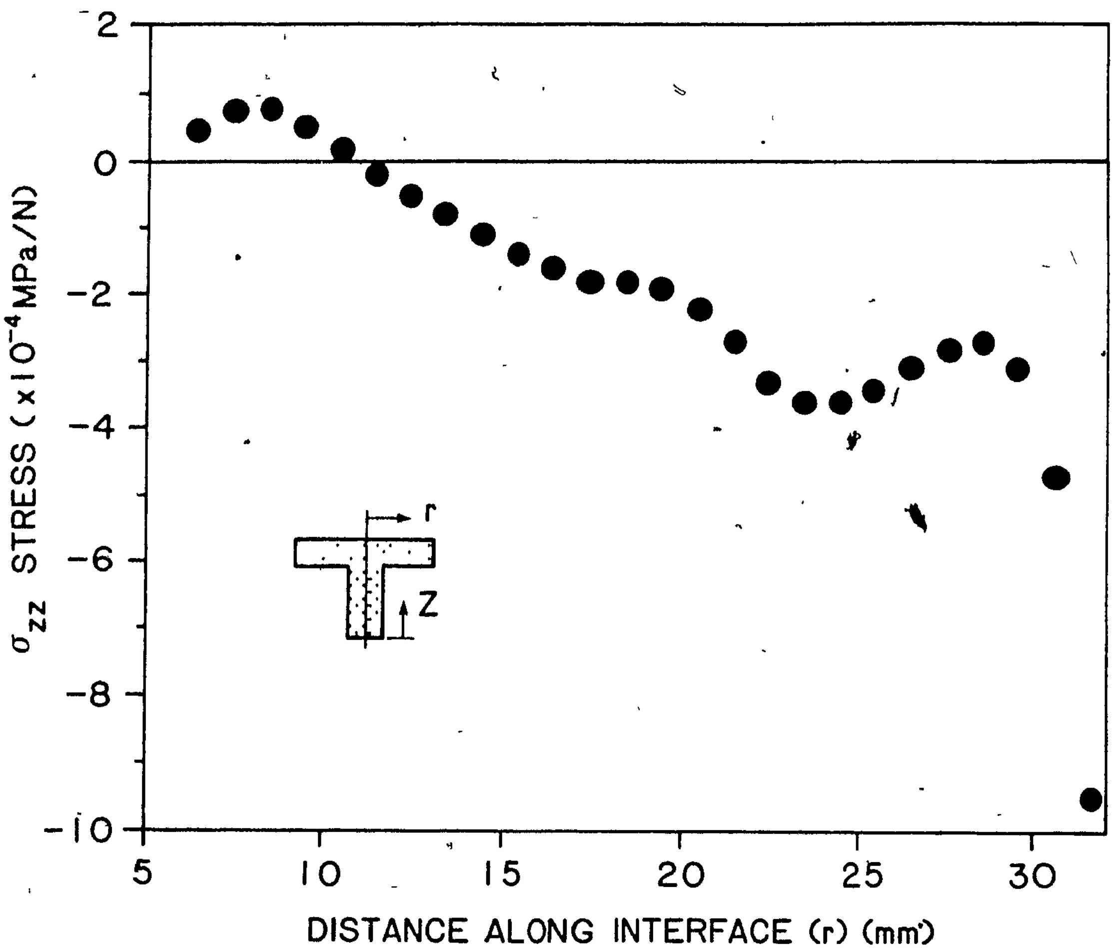

Figure 4.9: Normal stresses at the horizontal interface. Results from the analysis of mesh 2 by the program using the interface element.

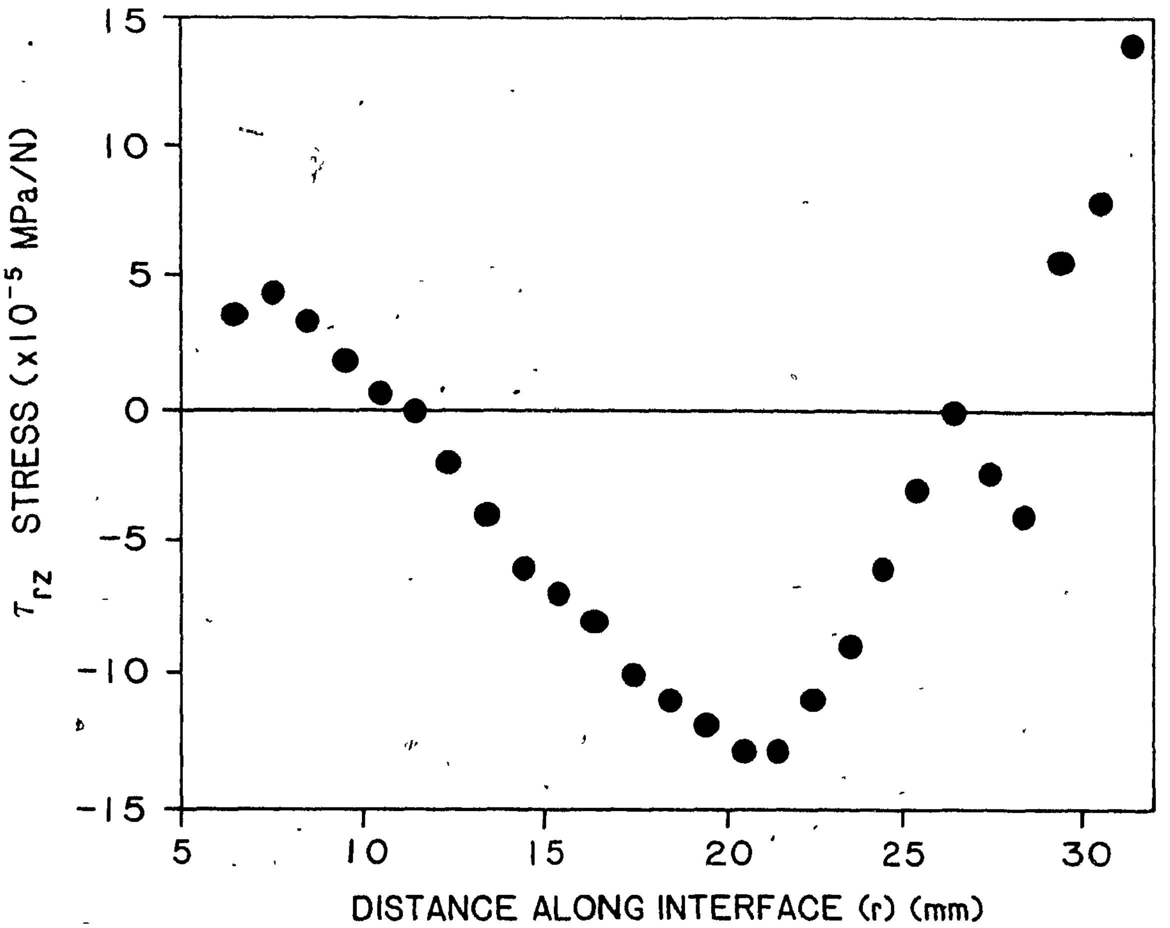

Figure 4.10: Shear stresses at the horizontal interface. Results from the analysis of mesh 2 by the program using the interface element.

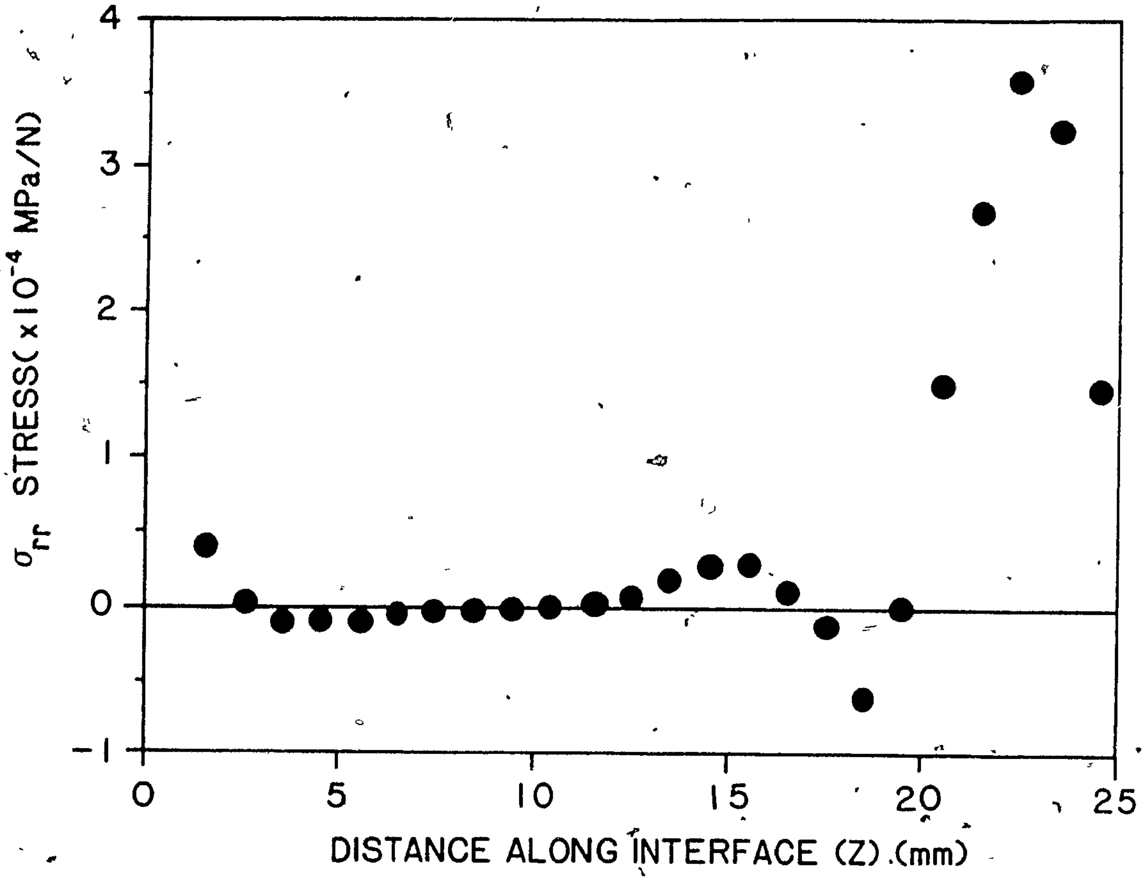

Figure 4.11: Normal stresses at the vertical interface. Results from the analysis of mesh 2 by the program using the interface element.

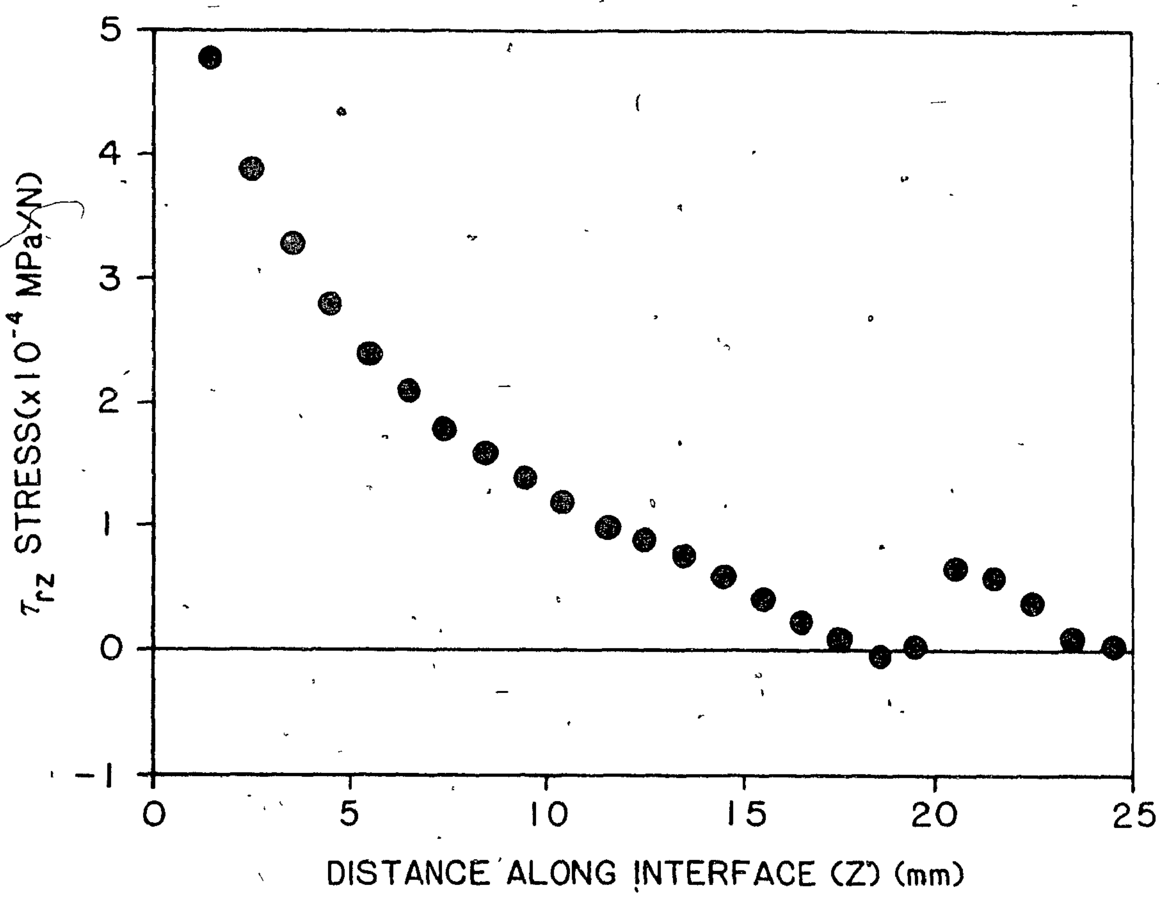

Figure 4.12: Shear stresses at the vertical interface. Results from the analysis of mesh 2 by the program using the new element.

the stem. Again, for a body weight of 1 kN, the maximum shear stress, approximately 0.48 MPa, occurring at the stem tip is much lower than reported values for shear bond strength (between 2 and 12 MPa) (Raab et al. [26]).

4.5 Conclusions

This analysis is not specifically aimed towards determining the actual interface stress distribution of a prosthetic tibia. Such an objective would require an anatomically correct three-dimensional model and would involve a series of parametric tests to establish the contribution of each variable (cement thickness, prosthesis design, material properties) to the stress field. The purpose of this study was mainly to demonstrate the capabilities of the new element in a sample application, and as such the discussion of the results will be limited to a qualitative interpretation and comparison of those obtained by using the new element, SAP-IV, and others available in the literature.

The application of the SAP-IV program to the analysis of the axisymmetric model of a resurfaced tibia resulted in a severe discontinuity of the stress vector at the implant/cement interface (in violation of the required continuity of this vector). This discontinuity remained significant even after two successive mesh refinements. However, the stresses obtained at the 'softer' cement side of the interface did appear to approach convergence after the second refinement and also compared very well with those obtained by using the developed interface element. From this point of view, the performance of the new element may be considered satisfactory, especially since it alleviates the need for mesh refinement; the latter becomes indispensable in analyses by the conventional finite element method in attempting to overcome the ambiguity of the stress discontinuity.

A similar analysis of a resurfaced tibia (for axisymmetric geometry but non-axisymmetric loading) has been carried out by Shrivastava et al. [31] using an in-house developed program. Their results (for a diametrically symmetric loading) compare well with those obtained by the new element. However, because of the differences in the manner of loading and the model used, the comparison can only be qualitative.

Despite the fact that there are numerous analyses of resurfaced knee joints in the literature, very few directly address the problem of the implant/cement stress distribution. The high stress discontinuity obtained at the interface by the conventional finite element method has led researchers to estimate the interface stress values by either extrapolating from the cement element centroidal stresses (Askew et al. [2]) or by accepting the cement element boundary stresses (Shrivastava et al. [31]). The latter assumption is now substantiated in view of the observation made in the course of this study that the 'soft' side stresses are more accurate than the 'hard' side ones. However, there can be no general guarantee as to the accuracy of the results obtained by a conventional displacement method if a convergence study or other validation procedure is not carried out. Hence, the use of the new interface element is strongly recommended for such biomechanical applications or other analyses involving bimaterial interfaces.

Chapter 5

THREE-DIMENSIONAL INTERFACE ELEMENT

5.1 Formulation

The formulation of the two dimensional interface element has been presented in Chapter 2. This formulation is extended here to three dimensions and a solid interface element is developed. The present section deals with the three dimensional formulation, while in Section 5.2 some notes on the development of the 3-D finite element program are discussed. Finally, Section 5.3 involves a validation analysis carried out to test the solid interface element.

Let u, v, w be the displacement components corresponding to the x, y, z directions of a Cartesian coordinate system. The stress and strain components can be written as:

$$\begin{aligned} \{\sigma\}^T &= < \sigma_x, \sigma_y, \sigma_z, \tau_{xy}, \tau_{yz}, \tau_{zx} > \quad , \\ \{\varepsilon\}^T &= < \varepsilon_x, \varepsilon_y, \varepsilon_z, \gamma_{xy}, \gamma_{yz}, \gamma_{zx} > \quad . \end{aligned} \tag{5.1}$$

The strain components are related to the displacement field as follows :

$$
\begin{aligned}
\varepsilon_x &= \frac{\partial u}{\partial x} \quad , \quad \gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \quad , \\
\varepsilon_y &= \frac{\partial v}{\partial y} \quad , \quad \gamma_{yz} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \quad , \\
\varepsilon_z &= \frac{\partial w}{\partial z} \quad , \quad \gamma_{zx} = \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \quad ,
\end{aligned}
\tag{5.2}
$$

and the stresses are related to the strains through Hooke's law:

$$
\{\sigma\} = [E]\{\varepsilon\} \quad , \tag{5.3}
$$

where

$$
[E] = \frac{E}{(1-\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \tag{5.4}
$$

It has again been assumed that the materials involved are isotropic linear elastic.

The proposed solid interface element is composed of two adjoining hexahedral elements, one on each side of the interface with the interface as their common boundary (similarly to the 2-D case). In order for the interelement interface to be unique for each couple of adjoining elements, that interface must be an element face in the three dimensional case. The centroid of that face is chosen as the location where stress compatibility is to be enforced (similar to the midside of the interface edge in 2-D). Therefore, the interface element in 3-D consists of a pair of nine-node hexahedral elements. A typical nine-node hexahedron is shown in Fig. 5.1.

Assuming coordinate transformations and isoparametric displacement functions as:

$$x = \Sigma N_i x_i \quad , \quad y = \Sigma N_i y_i \quad , \quad z = \Sigma N_i z_i \quad , \tag{5.5}$$

and

$$u = \Sigma N_i u_i \quad , \quad v = \Sigma N_i v_i \quad , \quad w = \Sigma N_i w_i \quad , \tag{5.6}$$

where the summation extends to eight terms for regular hexahedral elements and to nine in cases of nine-node hexahedrons used to construct the interface element. The shape functions N_i for the nine-node hexahedrons are selected such that the displacement fields along all edges are linear, in order not to violate displacement compatibility with other neighbouring linear elements. This is achieved through the following procedure:

First, the shape functions of a standard 13-node hexahedron (extra nodes at midsides and at centroid of face 1-2-3-4) are obtained. These are (e.g. Cook [6]):

$$\begin{aligned}
N_1^{13} &= \frac{1}{8}\eta\varsigma(1-\xi)(1-\eta)(1-\varsigma), \\
N_2^{13} &= \frac{1}{8}\eta\varsigma(1-\xi)(1-\eta)(1+\varsigma), \\
N_3^{13} &= \frac{1}{8}\eta\varsigma(1-\xi)(1-\eta)(1+\varsigma), \\
N_4^{13} &= -\frac{1}{8}\eta\varsigma(1-\xi)(1+\eta)(1-\varsigma),
\end{aligned}$$

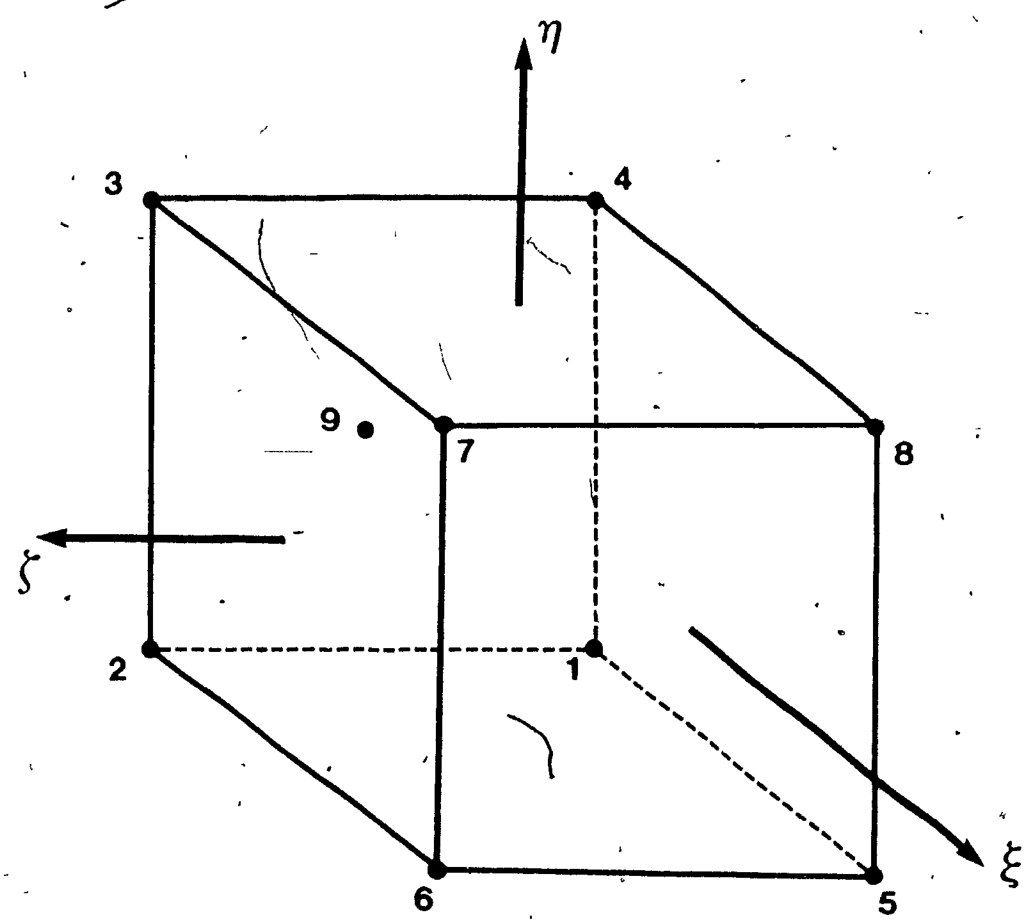

Figure 5.1: Typical 9-node hexahedral element and its natural coordinate system.

$$
\begin{aligned}
N_5^{13} &= \frac{1}{8}(1+\xi)(1-\eta)(1-\varsigma),\\
N_6^{13} &= \frac{1}{8}(1+\xi)(1-\eta)(1+\varsigma),\\
N_7^{13} &= \frac{1}{8}(1-\xi)(1+\eta)(1+\varsigma),\\
N_8^{13} &= \frac{1}{8}(1-\xi)(1+\eta)(1+\varsigma),\\
N_9^{13} &= \frac{1}{2}(1-\xi)(1-\eta^2)(1+\varsigma^2),\\
N_{10}^{13} &= -\frac{1}{4}\eta(1-\xi)(1-\eta)(1-\varsigma^2),\\
N_{11}^{13} &= \frac{1}{4}\varsigma(1-\xi)(1-\eta^2)(1+\varsigma),\\
N_{12}^{13} &= \frac{1}{4}\eta(1-\xi)(1+\eta)(1-\varsigma^2),\\
N_{13}^{13} &= -\frac{1}{4}\varsigma(1-\xi)(1-\eta^2)(1-\varsigma).
\end{aligned} \tag{5.7}
$$

The midside nodes (10, 11, 12, 13) are deleted through the use of the following assumed linear relationships between the displacements of the nodes involved:

$$
\begin{aligned}
u_{10} &= \frac{u_1+u_2}{2} \quad ,\\
u_{11} &= \frac{u_2+u_3}{2} \quad ,\\
u_{12} &= \frac{u_3+u_4}{2} \quad ,\\
u_{13} &= \frac{u_4+u_1}{2} \quad .
\end{aligned} \tag{5.8}
$$

Substituting the above values into the formula $u = \Sigma N_i u_i$:

$$
\begin{aligned}
u = {} & (N_1^{13} + \frac{N_{10}^{13} + N_{13}^{13}}{2})u_1 + (N_2^{13} + \frac{N_{10}^{13} + N_{11}^{13}}{2})u_2 + \\
& (N_3^{13} + \frac{N_{11}^{13} + N_{12}^{13}}{2})u_3 + (N_4^{13} + \frac{N_{12}^{13} + N_{13}^{13}}{2})u_4 + \\
& N_5^{13}u_5 + N_6^{13}u_6 + \cdots + N_9^{13}u_9.
\end{aligned}
\tag{5.9}
$$

The final shape functions of the nine-node hexahedral elements are, therefore:

$$
\begin{aligned}
N_1^9 &= N_1^{13} + \tfrac{N_{10}^{13}+N_{13}^{13}}{2} = \tfrac{1}{8}(1-\xi)(1-\eta)(1-\varsigma)(-\eta\varsigma-\eta-\varsigma), \\
N_2^9 &= N_2^{13} + \tfrac{N_{10}^{13}+N_{11}^{13}}{2} = \tfrac{1}{8}(1-\xi)(1-\eta)(1+\varsigma)(\eta\varsigma-\eta+\varsigma), \\
N_3^9 &= N_3^{13} + \tfrac{N_{11}^{13}+N_{12}^{13}}{2} = \tfrac{1}{8}(1-\xi)(1-\eta)(1+\varsigma)(-\eta\varsigma+\eta+\varsigma), \\
N_4^9 &= N_4^{13} + \tfrac{N_{12}^{13}+N_{13}^{13}}{2} = \tfrac{1}{8}(1-\xi)(1+\eta)(1-\varsigma)(\eta\varsigma+\eta-\varsigma), \\
N_5^9 &= N_5^{13} = \tfrac{1}{8}(1+\xi)(1-\eta)(1-\varsigma), \\
N_6^9 &= N_6^{13} = \tfrac{1}{8}(1+\xi)(1-\eta)(1+\varsigma), \\
N_7^9 &= N_7^{13} = \tfrac{1}{8}(1+\xi)(1+\eta)(1+\varsigma), \\
N_8^9 &= N_8^{13} = \tfrac{1}{8}(1+\xi)(1+\eta)(1-\varsigma), \\
N_9^9 &= N_9^{13} = \tfrac{1}{8}(1-\xi)(1-\eta^2)(1-\varsigma^2)
\end{aligned}
\tag{5.10}
$$

It can be easily verified that these shape functions satisfy rigid body motion and constant strain states, since:

$$\Sigma N_i = 1 \quad , \quad \Sigma N_{i,\xi} = 0 \quad , \quad \Sigma N_{i,\eta} = 0 \quad , \quad \Sigma N_{i,\varsigma} = 0. \tag{5.11}$$

The strain field corresponding to the assumed displacement field can be obtained through Eqs. (5.2) and can be expressed as:

$$\{\varepsilon\} = [B]\{\Delta\}, \tag{5.12}$$

where $\{\Delta\}^T =< u_1, u_2, \cdots u_9, v_1, v_2, \cdots v_9, w_1, w_2, \cdots w_9 >$ is the vector of nodal displacements of the element, and the strain-displacement matrix [B] is given by:

$$[B] = \begin{bmatrix} \frac{\partial N_1}{\partial x} & 0 & 0 & \frac{\partial N_2}{\partial x} & 0 & 0 & \cdots & \frac{\partial N_9}{\partial x} & 0 & 0 \\ 0 & \frac{\partial N_1}{\partial y} & 0 & 0 & \frac{\partial N_2}{\partial y} & 0 & \cdots & 0 & \frac{\partial N_9}{\partial y} & 0 \\ 0 & 0 & \frac{\partial N_1}{\partial z} & 0 & 0 & \frac{\partial N_2}{\partial z} & \cdots & 0 & 0 & \frac{\partial N_9}{\partial z} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial x} & 0 & \cdots & \frac{\partial N_9}{\partial y} & \frac{\partial N_9}{\partial x} & 0 \\ 0 & \frac{\partial N_1}{\partial z} & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial z} & \frac{\partial N_2}{\partial y} & \cdots & 0 & \frac{\partial N_9}{\partial z} & \frac{\partial N_9}{\partial y} \\ \frac{\partial N_1}{\partial z} & 0 & \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial z} & 0 & \frac{\partial N_2}{\partial x} & \cdots & \frac{\partial N_9}{\partial z} & 0 & \frac{\partial N_9}{\partial x} \end{bmatrix} . \tag{5.13}$$

As was mentioned in the two-dimensional case, the above differentiations cannot be carried out explicitly, N_i being functions of and ξ, η and ς. Therefore, use is made again of the inverse of the Jacobian matrix:

$$[J] = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \varsigma} & \frac{\partial y}{\partial \varsigma} & \frac{\partial z}{\partial \varsigma} \end{bmatrix} . \tag{5.14}$$

Denoting the inverse with the same symbols as in Chapter 2:

$$[\Gamma] = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} & \Gamma_{13} \\ \Gamma_{21} & \Gamma_{22} & \Gamma_{23} \\ \Gamma_{31} & \Gamma_{32} & \Gamma_{33} \end{bmatrix} = [J]^{-1} \quad , \tag{5.15}$$

the [B] matrix of Eq. (5.1.13) can be expressed as:

$$[B] = [B_1][B_2] \quad , \tag{5.16}$$

where

$$[B_1] = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} & \Gamma_{13} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Gamma_{21} & \Gamma_{22} & \Gamma_{23} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_{31} & \Gamma_{32} & \Gamma_{33} \\ \Gamma_{21} & \Gamma_{22} & \Gamma_{23} & \Gamma_{11} & \Gamma_{12} & \Gamma_{13} & 0 & 0 & 0 \\ 0 & 0 & 0 & \Gamma_{31} & \Gamma_{32} & \Gamma_{33} & \Gamma_{21} & \Gamma_{22} & \Gamma_{23} \\ \Gamma_{31} & \Gamma_{32} & \Gamma_{33} & 0 & 0 & 0 & \Gamma_{11} & \Gamma_{12} & \Gamma_{13} \end{bmatrix} \quad , \tag{5.17}$$

and

$$[B_2] = \begin{bmatrix} \frac{\partial N_1}{\partial \xi} & 0 & 0 & \frac{\partial N_2}{\partial \xi} & 0 & 0 & \cdots & \frac{\partial N_9}{\partial \xi} & 0 & 0 \\ \frac{\partial N_1}{\partial \eta} & 0 & 0 & \frac{\partial N_2}{\partial \eta} & 0 & 0 & \cdots & \frac{\partial N_9}{\partial \eta} & 0 & 0 \\ \frac{\partial N_1}{\partial \varsigma} & 0 & 0 & \frac{\partial N_2}{\partial \varsigma} & 0 & 0 & \cdots & \frac{\partial N_9}{\partial \varsigma} & 0 & 0 \\ 0 & \frac{\partial N_1}{\partial \xi} & 0 & 0 & \frac{\partial N_2}{\partial \xi} & 0 & \cdots & 0 & \frac{\partial N_9}{\partial \xi} & 0 \\ 0 & \frac{\partial N_1}{\partial \eta} & 0 & 0 & \frac{\partial N_2}{\partial \eta} & 0 & \cdots & 0 & \frac{\partial N_9}{\partial \eta} & 0 \\ 0 & \frac{\partial N_1}{\partial \varsigma} & 0 & 0 & \frac{\partial N_2}{\partial \varsigma} & 0 & \cdots & 0 & \frac{\partial N_9}{\partial \varsigma} & 0 \\ 0 & 0 & \frac{\partial N_1}{\partial \xi} & 0 & 0 & \frac{\partial N_2}{\partial \xi} & \cdots & 0 & 0 & \frac{\partial N_9}{\partial \xi} \\ 0 & 0 & \frac{\partial N_1}{\partial \eta} & 0 & 0 & \frac{\partial N_2}{\partial \eta} & \cdots & 0 & 0 & \frac{\partial N_9}{\partial \eta} \\ 0 & 0 & \frac{\partial N_1}{\partial \varsigma} & 0 & 0 & \frac{\partial N_2}{\partial \varsigma} & \cdots & 0 & 0 & \frac{\partial N_9}{\partial \varsigma} \end{bmatrix}. \quad (5.18)$$

The strain energy of the element can then be expressed as:

$$\Pi_s = \frac{1}{2}\{\Delta\}^T[\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1}[B]^T[E][B] \det\ [\mathrm{J}] \quad d\xi d\eta d\varsigma]\{\Delta\}, \quad (5.19)$$

from which it follows that the stiffness matrix is:

$$[K] = \int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1}[B]^T[E][B] \det\ [\mathrm{J}]\ d\xi\, d\eta\, d\,\varsigma. \quad (5.20)$$

As was the case in the two-dimensional formulation, Gaussian integration is necessary to evaluate the terms of this stiffness matrix.

A typical solid interface element is shown in Fig. 5.2. It consists of two adjacent nine-node hexahedral elements sharing the interface

(face 1-2-3-4). Let n, t_1, t_2 be a local coordinate system such that n is the outward normal to the interface (face 1-2-3-4), t_1 is tangent to the interface in the direction 1-2, and t_2 is tangent to the interface in the direction 1-4. Let the origin of the system be at the ninth node of the element . With reference to this system and in view of the stress continuity conditions developed in Section 2.1, the required stress boundary conditions at the interface are continuity of the σ_{nn} normal stress component and of the two shear stress components, τ_{nt_1} and τ_{nt_2}. The three degrees of freedom pertaining to the ninth node of each hexahedron (node 13) are again so chosen as to satisfy the required continuity of the above three stress components at this node.

The [Q] matrices in the stress continuity condition:

$$[Q^a]\{\Delta^a\} = [Q^b]\{\Delta^b\} \tag{5.21}$$

are given by:

$$[Q^a] = [T][E^a][B^a] \quad \text{and} \quad [Q^b] = [T][E^b][B^b], \tag{5.22}$$

where the [E] and [B] matrices for materials a and b are those given respectively by Eqs. (5.4) and (5.13), and [T] is the transformation matrix given by:

$$[T] = \begin{bmatrix} \ell_1^2 & m_1^2 & n_1^2 & 2\ell_1 m_1 & 2m_1 m_2 & 2n_1\ell_1 \\ \ell_1\ell_2 & m_1 m_2 & n_1 n_2 & (\ell_1 m_2 + \ell_2 m_1) & (m_1 n_2 + m_2 n_1) & (n_1\ell_2 + n_2\ell_1) \\ \ell_3\ell_1 & m_3 m_1 & n_3 n_1 & (\ell_3 m_1 + \ell_1 m_3) & (m_3 n_1 + m_1 n_3) & (n_3\ell_1 + n_1\ell_3) \end{bmatrix}, \tag{5.23}$$

with l_i, m_i, n_i the direction cosines between the axes of the local n, t_1, t_2 system and those of the global x, y, z system.

Isolating the degrees of freedom of the common node (node 13), the

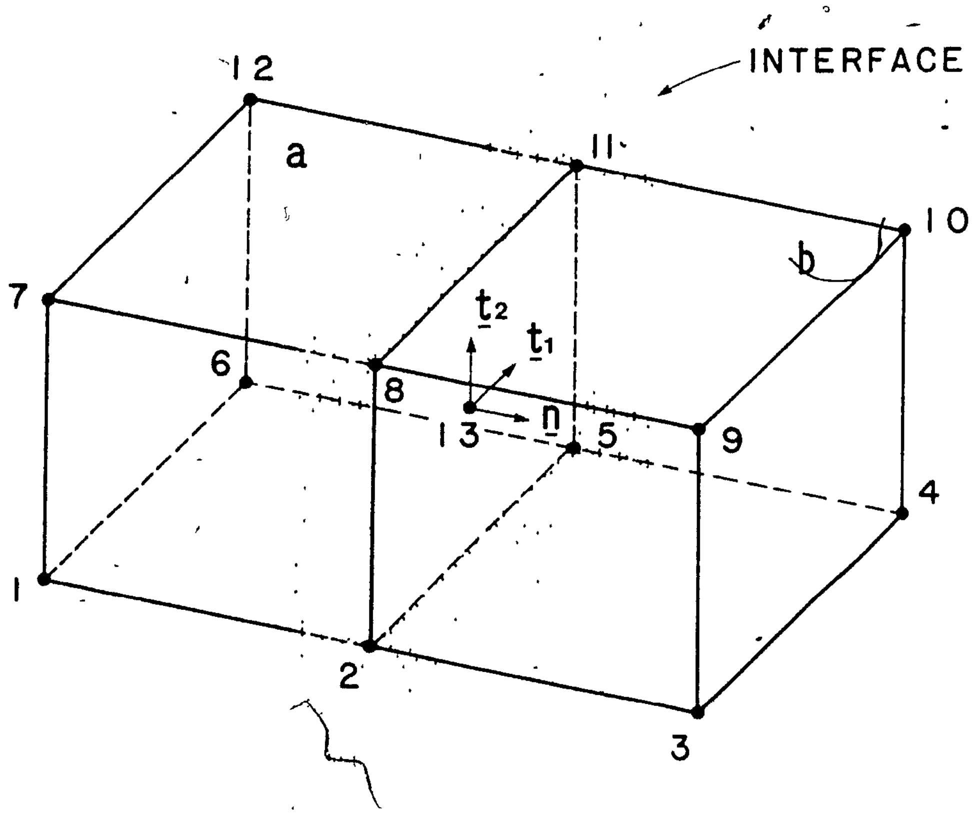

Figure 5.2: Typical pair of 9-node hexahedral elements sharing an interface.

above equations may be written as:

$$[Q^a]\{\Delta^a\} = [Q1^a]\{d^c\} + [Q2^a]\left\{\begin{array}{c} u_{13} \\ v_{13} \\ w_{13} \end{array}\right\} \tag{5.24}$$

and

$$[Q^b]\{\Delta^b\} = [Q1^b]\{d^c\} + [Q2^b]\left\{\begin{array}{c} u_{13} \\ v_{13} \\ w_{13} \end{array}\right\} \tag{5.25}$$

where $\{d^c\} =< u_1, v_1, w_1, u_2, v_2, w_2, \cdots, u_{12}, v_{12}, w_{12} >$, is the vector of nodal displacements of the composite interface element.

The matrices $[Q1^a]$ etc. are expressible as:

$[Q1^a] =$

$$\left[\begin{array}{cccccccccccc} Q^a_{1,13} & Q^a_{1,14} & Q^a_{1,15} & Q^a_{1,1} & Q^a_{1,2} & Q^a_{1,3} & 0 & 0 & 0 & 0 & 0 & 0 & Q^a_{1,10} & Q^a_{1,11} & Q^a_{1,12} \\ Q^a_{2,13} & Q^a_{2,14} & Q^a_{2,15} & Q^a_{2,1} & Q^a_{2,2} & Q^a_{2,3} & 0 & 0 & 0 & 0 & 0 & 0 & Q^a_{2,10} & Q^a_{2,11} & Q^a_{2,12} \\ Q^a_{3,13} & Q^a_{3,14} & Q^a_{3,15} & Q^a_{3,1} & Q^a_{3,2} & Q^a_{3,3} & 0 & 0 & 0 & 0 & 0 & 0 & Q^a_{3,10} & Q^a_{3,11} & Q^a_{3,12} \end{array}\right.$$

$$\begin{array}{ccccccccccccccc} Q^a_{1,22} & Q^a_{1,23} & Q^a_{1,24} & Q^a_{1,16} & Q^a_{1,17} & Q^a_{1,18} & Q^a_{1,4} & Q^a_{1,5} & Q^a_{1,6} & 0 & 0 & 0 & 0 & 0 & 0 \\ Q^a_{2,22} & Q^a_{2,23} & Q^a_{2,24} & Q^a_{2,16} & Q^a_{2,17} & Q^a_{2,18} & Q^a_{2,4} & Q^a_{2,5} & Q^a_{2,6} & 0 & 0 & 0 & 0 & 0 & 0 \\ Q^a_{3,22} & Q^a_{3,23} & Q^a_{3,24} & Q^a_{3,16} & Q^a_{3,17} & Q^a_{3,18} & Q^a_{3,4} & Q^a_{3,5} & Q^a_{3,6} & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

$$\left.\begin{array}{cccccc} Q^a_{1,7} & Q^a_{1,8} & Q^a_{1,9} & Q^a_{1,19} & Q^a_{1,20} & Q^a_{1,21} \\ Q^a_{2,7} & Q^a_{2,8} & Q^a_{2,9} & Q^a_{2,19} & Q^a_{2,20} & Q^a_{2,21} \\ Q^a_{3,7} & Q^a_{3,8} & Q^a_{3,9} & Q^a_{3,19} & Q^a_{3,20} & Q^a_{3,21} \end{array}\right]$$

$$[Q2^a] = \left[\begin{array}{ccc} Q^a_{1,25} & Q^a_{1,26} & Q^a_{1,27} \\ Q^a_{2,25} & Q^a_{2,26} & Q^a_{2,27} \\ Q^a_{3,25} & Q^a_{3,26} & Q^a_{3,27} \end{array}\right] \tag{5.26}$$

$$[Q1^b] =$$

$$\left[\begin{array}{lllllllllllll} 000 & Q^b_{1,4} & Q^b_{1,5} & Q^b_{1,6} & Q^b_{1,16} & Q^b_{1,17} & Q^b_{1,18} & Q^b_{1,19} & Q^b_{1,20} & Q^b_{1,21} & Q^b_{1,7} & Q^b_{1,8} & Q^b_{1,9} \\ 000 & Q^b_{2,4} & Q^b_{2,5} & Q^b_{2,6} & Q^b_{2,16} & Q^b_{2,17} & Q^b_{2,18} & Q^b_{2,19} & Q^b_{2,20} & Q^b_{2,21} & Q^b_{2,7} & Q^b_{2,8} & Q^b_{2,9} \\ 000 & Q^b_{3,4} & Q^b_{3,5} & Q^b_{3,6} & Q^b_{3,16} & Q^b_{3,17} & Q^b_{3,18} & Q^b_{3,19} & Q^b_{3,20} & Q^b_{3,21} & Q^b_{3,7} & Q^b_{3,8} & Q^b_{3,9} \end{array}\right.$$

$$\begin{array}{lllllllllllllll} 0 & 0 & 0 & 0 & 0 & 0 & Q^b_{1,1} & Q^b_{1,2} & Q^b_{1,3} & Q^b_{1,13} & Q^b_{1,14} & Q^b_{1,15} & Q^b_{1,22} & Q^b_{1,23} & Q^b_{1,24} \\ 0 & 0 & 0 & 0 & 0 & 0 & Q^b_{2,1} & Q^b_{2,2} & Q^b_{2,3} & Q^b_{2,13} & Q^b_{2,14} & Q^b_{2,15} & Q^b_{2,22} & Q^b_{2,23} & Q^b_{2,24} \\ 0 & 0 & 0 & 0 & 0 & 0 & Q^b_{3,1} & Q^b_{3,2} & Q^b_{3,3} & Q^b_{3,13} & Q^b_{3,14} & Q^b_{3,15} & Q^b_{3,22} & Q^b_{3,23} & Q^b_{3,24} \end{array}$$

$$\left.\begin{array}{llllll} Q^b_{1,10} & Q^b_{1,11} & Q^b_{1,12} & 0 & 0 & 0 \\ Q^b_{2,10} & Q^b_{2,11} & Q^b_{2,12} & 0 & 0 & 0 \\ Q^b_{3,10} & Q^b_{3,11} & Q^b_{3,12} & 0 & 0 & 0 \end{array}\right]$$

$$[Q2^b] = \begin{bmatrix} Q^b_{1,25} & Q^b_{1,26} & Q^b_{1,27} \\ Q^b_{2,25} & Q^b_{2,26} & Q^b_{2,27} \\ Q^b_{3,25} & Q^b_{3,26} & Q^b_{3,27} \end{bmatrix} \tag{5.27}$$

The stress continuity condition, Eq. (5.21), then lead to

$$\begin{Bmatrix} u_{13} \\ v_{13} \\ w_{13} \end{Bmatrix} = [L]\{d^c\}, \tag{5.28}$$

where

$$[L] = [[Q2^b] - [Q2^a]]^{-1}[[Q1^a] - [Q1^b]]. \tag{5.29}$$

Substituting Eq. (5.28) into Eqs. (5.24) and (5.25), the nodal displacement vectors of the two individual elements may be written as:

$$\{\Delta^a\} = [R^a]\{d^c\}, \tag{5.30}$$

and

$$\{\Delta^b\} = [R^b]\{d^c\}, \tag{5.31}$$

where

$$[R^a] = [Q^a]^{-1}[Q1^a] + [Q^a]^{-1}[Q2^a][L], \tag{5.32}$$

and

$$[R^b] = [Q^b]^{-1}[Q1^b] + [Q^b]^{-1}[Q2^b][L]. \tag{5.33}$$

Hence, by virtue of Eq. (5.20) and Eqs. (5.31) and (5.32), the stiffness matrix of the composite interface element can be written as:

$$[K^i] = [R^a]^T[K^a][R^a] + [R^b]^T[K^b][R^b]. \tag{5.34}$$

It is clear from the above that the formulation of the three dimensional interface element is essentially an extension of that of the two dimensional one. The only difference lies in the order of the matrices involved and in the fact that three stress components are required to be continuous across the interface (calling for the appropriate selection of three degrees of freedom to satisfy that condition).

5.2 Notes on the 3-D program

A three-dimensional finite element program has been developed in order to test the new solid interface element. This program is similar to the two-dimensional program described in Section 2.4 and listed in Appendix B.1. The subroutines dealing with the stiffness matrix assembly, decomposition and back-substitution are essentially identical to the two dimensional ones, except for the size of the matrices involved. The only significant modifications that need to be addressed in this section pertain to the library of elements. The full listing of the three dimensional program is included in Appendix B.2.

The three-dimensional elements included in the program are the isoparametric linear hexahedron, the isoparametric linear pentahedron,

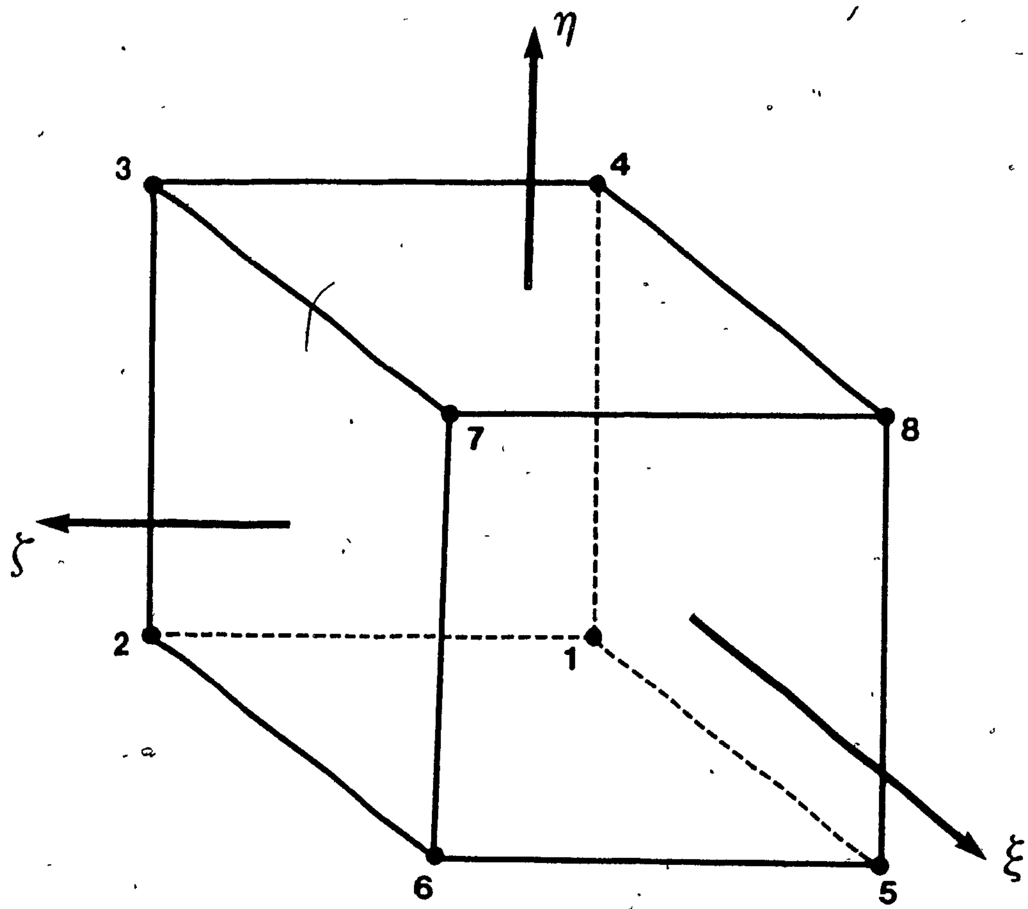

Figure 5.3: Linear hexahedral element.

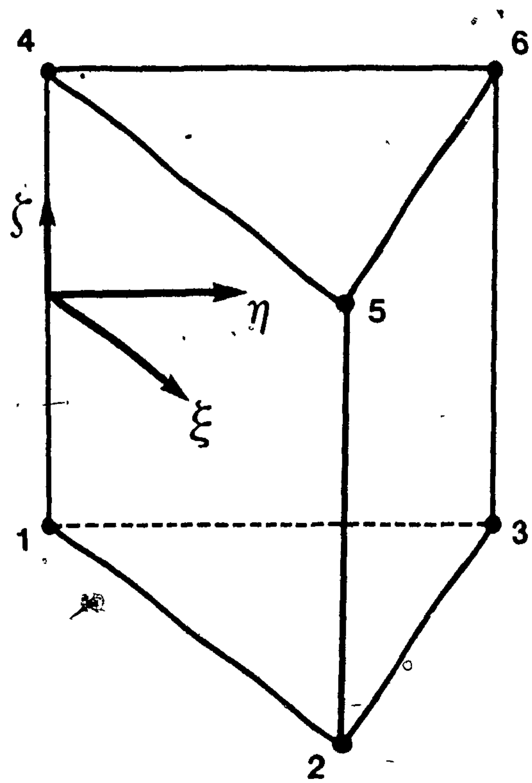

Figure 5.4: Linear pentahedral element.

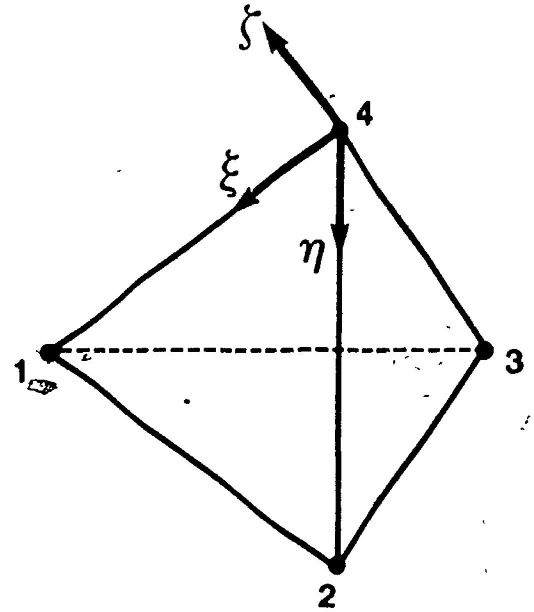

Figure 5.5: Constant strain tetrahedral element.

the constant strain tetrahedron, and the solid interface element. The formulation of the last one been described in the previous section.

The linear hexahedral element (Fig. 5.3) is based on the standard isoparametric formulation (e.g. Cook [7]).

In the development of the linear pentahedral element (Fig. 5.4) the isoparametric assumption has also been made. The shape functions are those given by Brebbia and Connors [5].

The development of the pentahedral element stiffness matrix follows the same procedure as that described in Section 5.1 for the nine-node hexahedral element. However, in the numerical integration of the stiffness matrix, a cubic integration order has been used (Brebbia and Connors [5]).

The development of the tetrahedral element (Fig. 5.5) is based on an isoparametric constant strain formulation (Brebbia and Connors [5]). Therefore, the integration of the stiffness matrix could be carried out explicitly so that no numerical integration is necessary.

The pentahedral and terahedral elements have been tested individually by comparing them to equivalent ones in the MSC/NASTRAN [20] finite element program (PENTA and TETRA respectively).

5.3 Validation test for the solid interface element

The new three-dimensional interface element has been tested in a sample analysis the results of which are presented in this section. As was the case in the two-dimensional validation tests (Chapter 3) and in the sample biomechanical analysis (Chapter 4), the three-dimensional test involves analyses of the same mesh by both the developed program and the SAP-IV program and comparison of the results with the theoretical solutions.

The problem chosen for this test has already been presented in Chapter 3. It consists of an annular composite disk under internal pressure (Fig. 3.16a). The relative scarcity of three dimensional exact solutions involving bimaterial interfaces has led to the choice of this essentially two dimensional problem.

The finite element mesh used in both analyses (by the devloped program and by SAP-IV) is shown in Fig. 5.6. Because of symmetry in geometry and in loading, only one quarter of the disk has been modelled. The inner third of the annulus has again been assumed to be 100 times stiffer than the outer part. In all, 24 solid elements have been used; they are all hexahedrals except in the analysis by the new element where four pairs of hexahedrals (nine-node element) have been combined to form four interface elements. The boundary conditions involved fixation of the nodes on the two axes of symmetry (x,y) against motion normal to the respective axes, and fixation of all nodes against motion in the vertical (z) direction to simulate axisymmetric plane stress conditions. The applied load was again an internal unit pressure.

The theoretical solution to this problem is presented in Appendix A.1 According to that solution, the exact value of the interface radial stress is 0.004 pressure units. The analysis by the developed solid interface element yielded a unique interface radial stress of approximately 0.0039 pressure units. The analysis by SAP-IV yielded an interface radial stress of 0.080 pressure units on the 'hard' side of the interface and 0.0032 pressure units on the 'soft' side of the interface (Table 5.1). The superiority of the new element is obvious from these results. It can also be noted that, again, the 'soft' side stresses obtained by SAP-IV are much more accurate than the 'hard' side values.

Unfortunately, because of time limitations, further three dimensional validation tests could not be carried out. It appears, however, that the proposed interface element is as successful in three dimensions

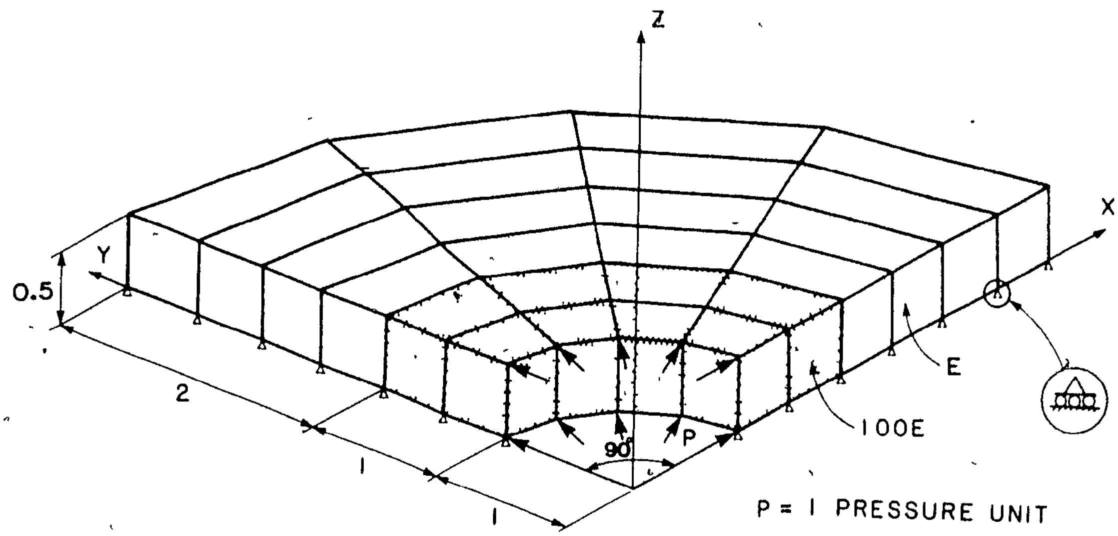

Figure 5.6: Composite annular disk under internal pressure; 3-d finite element mesh.

Theoretical	3-D SAP-IV		3-D New Element
	hard side	soft side	
0.0040	0.0080	0.0032	0.0039

Table 5.1: Composite annular disk problem; Radial stresses at the bimaterial interface (results from the 3-d analysis).

as it was found to be in two dimensional cases.

Chapter 6

SUMMARY AND CONCLUSIONS

6.1 Summary

An interface finite element has been formulated satisfying the required continuity of the stress vector across a bimaterial interface. The stress vector continuity is enforced at a point on the interface. This is achieved by selecting the degrees of freedom corresponding to that point so as to satisfy the required stress boundary conditions. Since the element is based on the displacement method, displacement compatibility is satisfied at all points along the interface. Two general finite element programs have been developed incorporating the new interface element: one for plane-stress/plane-strain and axisymmetric problems and one for three dimensional problems. A series of validation tests have been performed that compare the results obtained by the new element to those obtained by a conventional displacement based program (SAP-IV) and to the exact theoretical results. The interface element was found to be very reliable in predicting stress distributions at two-material interfaces. Furthermore, in view of the fact that the conventional method was seen to yield unpredictable results when used at bimaterial interfaces, the use of the new element can be proposed in cases involving interfaces of highly dissimilar materials. Finally, a stress analysis of an axisymmetric resurfaced tibia model was carried out using the developed element along the cement/metal interface.

6.2 Conclusions

The conclusions of this study are presented here in two parts. First, some observations are made concerning the use of the conventional displacement method in bimaterial interface problems. The conclusions pertaining to the performance of the new element then follow.

6.2.1 Analysis of bimaterial interface problems by the conventional displacement method

In a conventional displacement based finite element analysis, two distinct stress distributions are obtained on the two sides of a bimaterial interface. The differences between the components of these distributions that are required to be continuous are gradually decreased, as the mesh is refined. At the limit, these stress components are expected to become identical, as coincidence with the exact stress distribution is achieved. This exact solution, however, is not necessarily bounded by any pair of discontinuous interface distributions obtained in the course of mesh refinement. This fact is demonstrated in Fig. 3.5 (rectangular plate with circular inclusion). It is therefore evident that in cases of high stress discontinuity, it is virtually impossible to predict a reliable range for the exact solution without extensive mesh refinement. Small discontinuities, however, indicate reasonable results, since the degree of discontinuity is generally associated with the degree of convergence.

The ratio of the elastic moduli of the adjacent materials is not the only factor affecting stress discontinuity. It has been seen that the type of structure and loading involved and the type of mesh used, as well as the relative location of the material regions have a significant bearing on the reliability of the stresses obtained. The cantilever beam case, Section 3.3, is a typical example of a structure yielding highly discontinuous interface stresses despite successive refinements. On the other hand, the vertical wedge problem, Section 3.2, is an example of a structure where neither the mesh size nor the material properties significantly affect the very small interface discontinuity.

The intuitive assumption of many researchers to accept the stresses on the softer side as representative of the interface distribution is supported by most of the validation tests and by the convergence tests for the axisymmetric analysis of the resurfaced tibia. This assumption cannot be guaranteed to always apply, however, as was seen in Section 3.4.

The conventional displacement based finite element method, therefore, cannot be generally relied upon to accurately determine the stress distribution at bimaterial interfaces.

6.2.2 Conclusions concerning the new stress compatible finite element

The new element enforces the continuity of the stress vector at a point on the bimaterial interface. Consequently, interelement force equilibrium is satisfied at that point. Unique values are therefore obtained for those interface stress components that are required to be continuous.

The interface stresses calculated by the new element were found to be very reasonable and usually very accurate. The stress components which are not required to be continuous at the interface are not affected by the new element. Similarly, the stress distribution away from the interface is not affected either by the interface elements along the intermaterial boundary. Therefore, the new element can be relied upon to accurately determine the stress distribution at the interfaces of highly dissimilar materials without affecting the stresses in the rest of the structure.

No additional modelling effort is involved in the case of the interface elements, since the same mesh can be used as that for an analysis using conventional linear quadrilateral elements. Moreover, the use of the new element does not incur significant additional computational costs, since the global degrees of freedom of the problem remain the same as those for linear quadrilateral elements.

In view of the foregoing, a proposed procedure for the analysis of problems involving bimaterial interfaces would be to first carry out an initial analysis by the conventional displacment method on a relatively coarse mesh, and in the the event of detected high interface stress discontinuities, to perform a subsequent analysis using the interface element on a locally more refined mesh.

6.3 Suggestions for further research

Further validation tests ~~are~~ necessary, especially in the three dimensional case, for a better assessment of the performance of the interface element.

As a second step towards developing a general interface element, boundary conditions other than those of perfect bond should be considered. That is, the formulation should be extended to allow for separation and/or slip at the interface.

Finally, only static cases have been considered in this study. Dynamic effects are very important, however, at bimaterial interfaces, so that is envisaged to include such effects at a later stage.

Bibliography

[1] Ahmed, A.M., Raab, S., and Miller, J.E., "Metal/Cement Interface Strength in Cemented Stem Fixation," *Journal of Orthopaedic Research*, Vol. 2, pp. 105-108, 1984.

[2] Askew, M.J. and Lewis, J.L., "Analysis of Model Variables and Fixation Post Length Effects on Stresses around a Prosthesis in the Proximal Tibia," *Journal of Biomechanical Engineering*, Vol. 103, pp. 239-245, 1981.

[3] Bartel, D.L., Burstein, A.H., Santavicca, E.A., and Insall, J.N., "Performance of the Tibial Component in Total Knee Replacement," *Journal of Bone and Joint Surgery*, Vol. 64-A, pp. 1026-1033, 1982.

[4] Bathe, K.J., Wilson, E.L., and Peterson, F.E., "SAP-IV – A Structural Analysis Program for Static and Dynamic Response of Linear Systems. Earthquake Engineering Research Center, Report No. EERC 73-11," University of California Berkeley, 1973.

[5] Brebbia, C.A. and Connor, J.J., *Fundamentals of Finite Element Techniques*, John Wiley & Sons, 1974.

[6] Cheal, E.J., Hayes, W.C., Lee, C.H., Snyder, B.D., and Miller, J., "Stress Analysis of a Condylar Knee Tibial Component: Influence

of Metaphyseal Shell Properties and Cement Injection Depth," *Journal of Orthopaedic Research*, Vol. 3, pp. 424-434, 1985.

[7] Cook, R.D., *Concepts and Applications of Finite Element Analysis*, 2nd Edition, John Wiley & Sons, New York, 1981.

[8] Eftekhar, N.S., "Total Knee-Replacement Arthroplasty. Results with the Intramedullary Adjustable Total Prosthesis," *Journal of Bone and Joint Surgery*, Vol. 65-A, pp. 293-309, 1983.

[9] Gallagher, R.H., *Finite Element Analysis. Fundamentals*, Prentice-Hall Inc., New Jersey, 1975.

[10] Goldstein, S.A., Wilson, D.L., Sonstegard, D.A., and Mathews, L.S., "The Mechanical Properties of Human Tibial Trabecular Bone as a Function of Metaphyseal Location," *Journal of Biomechanics*, Vol. 16, pp. 965-969, 1983.

[11] Haas, S.S., Brauer, G.M., and Dickson, M.A., "A Characterization of Poly-(methyl Methacrylate) Bone Cement," *Journal of Bone and Joint Surgery*, Vol. 57-A, pp. 380-391, 1975.

[12] Herrmann, L.R., "Improved Stress Calculation for Simple Quadrilateral Elements," *Computers and Structures*, Vol. 6, No. 2, pp. 141-148, 1976.

[13] Hinton, E. and Campbell, J.S., "Local and Global Smoothing of Discontinuous Finite Element Functions Using a Least Squares Method," *International Journal of Numerical Methods in Engineering*, Vol. 8, No. 3, pp. 461-480, 1974.

[14] Hinton, E. and Owen, D.R.J., *An Introduction to Finite Element Computations*, Pineridge Press Ltd., Swansea, 1979.

[15] Huiskes, R. and Chao, E.Y.S., "A Survey of Finite Element Analysis in Orthopaedic Biomechanics: The First Decade," *Journal of Biomechanics*, Vol. 16, pp. 385-409, 1983.

[16] Keller, J.C., Lautenschlager, E.P., Marshall, Jr., G.W., and Meyer, Jr., P.R., "Factors Affecting Surgical Alloy/Bone Cement Interface Adhesion," *Journal of Biomedical Materials Research*, Vol. 14, pp. 639-652, 1980.

[17] Lewis, J.L., Askew, M.J., and Jaycox, D.P., "A Comparative Evaluation of Tibial Component Designs of Total Knee Prostheses," *Journal of Bone and Joint Surgery*, Vol. 64-A, pp.129-135, 1982.

[18] Loubignac, G., Cantin, G., and Touzot, G., "Continuous Stress Fields in Finite Element Analysis," Technical Notes, *AIAA Journal*, Vol. 15, No. 11, pp. 1645-1647, 1977.

[19] MacNeal, R.H., "A Simple Quadrilateral Shell Element," *Computers and Structures*, Vol. 8, pp. 175-183, 1978.

[20] The MacNeal-Schwendler Corporation, MSC/NASTRAN User's Manual, Version 62, Los Angeles, 1983.

[21] Murase, K., Crowninshield, R.D., Pederson, D.R., and Chang, T-S., "An Analysis of Tibial Component Design in Total Knee Arthroplasty," *Journal of Biomechanics*, Vol. 16, pp. 13-22, 1982.

[22] Murray, R.P., Hayes, W.C., Edwards, W.T., and Harry, J.D., "Mechanical Properties of the Subchondral Plate and the Metaphyseal Shell," *Transactions of the 30th Annual Meeting of the Orthopaedic Research Society*, ORS, Chicago, Vol. 9, p. 197, 1984.

[23] Muskhelishvili, N.I., *Some Basic Problems of the Mathematical Theory of Elasticity*, (translated from the Russian by J.R.M. Radok), 3rd Edition, P. Noordhoff Ltd., Groningen, Holland, 1953.

[24] Parker, E.B., (Editor), *Materials Data Book*, McGraw-Hill, New York, 1967.

[25] Popov, E.P., *Mechanics of Materials*, 2nd Edition, Prentice/Hall Inc., Englewood Cliffs, NJ, 1976.

[26] Raab, S., Ahmed, A.M., and Provan, J.W., "The Quasi-Static and Fatigue Performance of the Implant/Bone-Cement Interface," *Journal of Biomedical Materials Research*, Vol. 15, pp. 159-182, 1981.

[27] Reilly, D., Walker, P.S., Ben-Dov, M., and Ewald, F.C., "Effects of Tibial Components on Load Transfer in the Upper Tibia," *Clinical Orthopaedics*, Vol. 165, pp. 273-282, 1982.

[28] Rohlmann, A., Bergmann, G., and Kölbel, R., "Aussagewert und Grenzen der Spannungsberechnung mit der Finiten-Elementen-Methoden (FEM) bei Orthopädischen Problemen," *Zeitschrift für Orthopädie*, Vol. 118, pp. 122-131, 1980.

[29] Salama, M. and Utku, S., "Stress Computation in Displacement Methods for Two-Material Elastic Media," *Computer Methods in Applied Mechanics and Engineering*, Vol. 10, pp. 325-338, 1977.

[30] Shirazi-Adl, A. and Ahmed, A.M., "Micro-Motions at the Bone/Prosthesis Interface in Porous-Surface Metal Tibial Implants - An Axisymmetric Finite Element Study," *Transactions of the 32nd Annual Meeting of the Orthopaedic Research Society*, ORS, Chicago, Vol. 11, p. 262, 1986.

[31] Shrivastava, S.C., Ahmed, A.M., Shirazi-Adl, A., and Burke, D.L., "Effect of a Cement-Bone Composite Layer and Prosthesis Geometry on Stresses in a Prosthetically Resurfaced Tibia," *Journal of Biomedical Materials Research*, Vol. 16, pp. 929-949, 1982.

[32] Soh, A.K., "A 'Modified' Finite Element Technique for the Determination of Interfacial Stresses in a Composite Material," *Proceedings of International Conference on Finite Elements in Computational Mechanics*, Pergamon Press Ltd., Bombay, India, pp. 249-257, 1985.

[33] Sokolnikoff, I.S., *Mathematical Theory of Elasticity*, 2nd Edition, McGraw-Hill, New York, 1956.

[34] Taylor, R.L., Beresford, P.J., and Wilson, E.L., "A Non- Conforming Element for Stress Analysis," *International Journal of Numerical Methods in Engineering*, Vol. 10, No. 6, pp. 1211-1219, 1976.

[35] Timoshenko, S. and Goodier, J.N., *Theory of Elasticity*, 2nd Edition, McGraw-Hill, New York, 1964.

[36] Wilson, E.L., "Structural Analysis of Axisymmetric Solids," *AIAA Journal*, Vol. 3, No. 12, pp. 2269-2274, 1965.

[37] Wilson, E.L., Taylor, R.L., Doherty, W.P., and Ghaboussi, J., "Incompatible Displacement Models," *Numerical and Computer Methods in Structural Mechanics*, S.J. Fenves et al. (Editors), Academic Press, New York, pp. 43-57, 1973.

[38] Zienkiewicz, O.C., *The Finite Element Method*, 3rd Edition, McGraw-Hill, London, U.K., 1977.

Appendix A

DERIVATION OF CLOSED-FORM SOLUTIONS USED IN THE VALIDATION TESTS

A.1 Interface radial stress distribution of a composite disk

The problem of a circular cylinder under internal and external pressure is a well known one (Lamé problem) and analytical solutions are available for the unimaterial case in elasticity textbooks (Timoshenko and Goodier [34]). The example presented in the section on the validation tests (Chapter 2) involves a two-material cylinder of unit height (Fig. A.1). The solution for the interface radial stress distribution for this case is briefly outlined here.

Stress-strain relations for plane stress state:

$$\begin{aligned} \sigma_{rr} &= \kappa(\varepsilon_{rr} + \nu\varepsilon_{\theta\theta}), \\ \sigma_{\theta\theta} &= \kappa(\nu\varepsilon_{rr} + \varepsilon_{\theta\theta}), \end{aligned} \tag{A.1}$$

$$
\begin{aligned}
\tau_{r\theta} &= 0 \quad \text{(because of axisymmetry)}, \\
\text{where,} \quad \kappa &= E/(1-\nu^2).
\end{aligned}
$$

Strain-displacement relations:

$$
\begin{aligned}
\varepsilon_{rr} &= \frac{du}{dr}, u = u(r), \\
\varepsilon_{\theta\theta} &= \frac{(r+u)d\theta - rd\theta}{rd\theta} = \frac{u}{r}.
\end{aligned} \tag{A.2}
$$

Equilibrium equation:

$$
\frac{\partial \sigma_{rr}}{\partial r} + \frac{1}{r}\frac{\partial \tau_{r\theta}}{\partial \theta} + \frac{\sigma_{rr} - \sigma_{\theta\theta}}{r} + R = 0. \tag{A.3}
$$

Since, body force R $= 0$ and $\tau_{r\theta} = 0$,

$$
\frac{\partial \sigma_{rr}}{\partial r} + \frac{\sigma_{rr} - \sigma_{\theta\theta}}{r} = 0. \tag{A.4}
$$

Substitution of equations (A.1) and (A.2) and (A.4) yields, after rearrangement, the displacement differential equation:

$$
\frac{d^2u}{dr^2} + \frac{1}{r}\frac{du}{dr} - \frac{u}{r^2} = 0. \tag{A.5}
$$

The solution to equation (A.5) is achieved through the transformation $r = e^t$:

$$
u = Ar + \frac{B}{r} \tag{A.6}
$$

where A,B are constants depending on the boundary conditions. Finally, use of equations (A.1), (A.2) and (A.6) yields the stress distribution:

$$
\begin{aligned}
\sigma_{rr} &= \kappa[(1+\nu)A - \frac{B}{r^2}(1-\nu)], \\
\sigma_{\theta\theta} &= \kappa[(1+\nu)A + \frac{B}{r^2}(1-\nu)].
\end{aligned} \tag{A.7}
$$

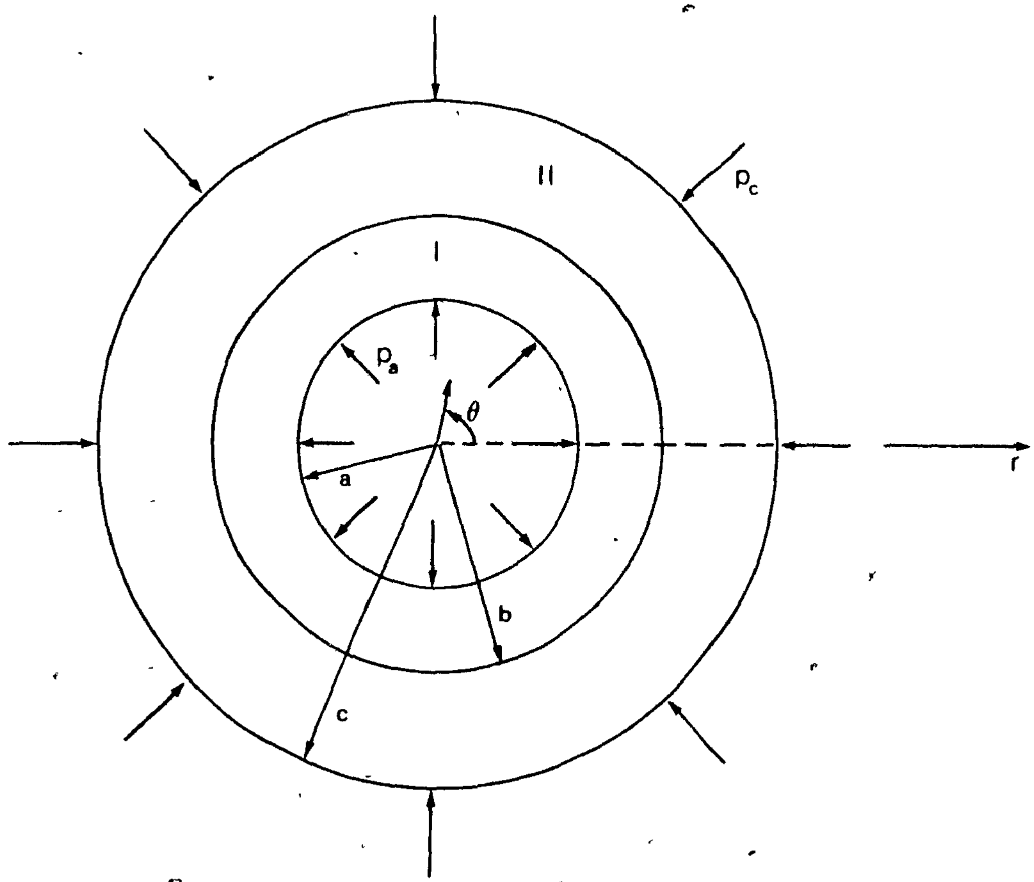

Figure A.1: Composite annular disk under external and internal pressure.

The boundary conditions imposed in this case are:

$$
\begin{aligned}
\sigma_{rr}^{I} &= -p_a \text{ at } r = a, \\
\sigma_{rr}^{II} &= -p_c \text{ at } r = c, \\
u^{I} &= u^{II} \text{ at } r = b \text{ (interface condition)}, \\
\sigma_{rr}^{II} &= \sigma_{rr}^{I} = -p_b \text{ at } r = b \text{ (interface condition)},
\end{aligned}
\tag{A.8}
$$

where the superscripts refer to material regions.

The implementation of these boundary conditions into equations (A.7) leads to the determination of constants A,B for both material regions:

$$
\begin{aligned}
A^{I} &= \frac{a^2 p_a - b^2 p_b}{(b^2 - a^2)\kappa^{I}(1 + \nu^{I})}, \\
B^{I} &= \frac{-a^2 b^2 (p_a - p_b)}{(b^2 - a^2)\kappa^{I}(\nu^{I} - 1)}, \\
A^{II} &= \frac{c^2 p_c - b^2 p_b}{(b^2 - c^2)\kappa^{II}(1 + \nu^{II})}, \\
B^{II} &= \frac{b^2 c^2 (p_b - p_c)}{(c^2 - b^2)\kappa^{II}(\nu^{II} - 1)}.
\end{aligned}
\tag{A.9}
$$

Finally, the condition of displacement compatibility, $u^{I} = u^{II}$ at the interface ($r = b$) leads through the use of equations (A.6) and (A.9) to the derivation of a closed-form solution for the interface radial stress, p_b:

$$
\begin{aligned}
& p_b \left[\frac{b(a^2 + a^2\nu^{I} - b^2\nu^{I} + b^2)}{(b^2 - a^2)\kappa^{I}(\nu^{I2} - 1)} - \frac{b(c^2 + c^2\nu^{II} - b^2\nu^{II} + b^2)}{(b^2 - c^2)\kappa^{II}(\nu^{II2} - 1)} \right] \\
&= p_a \left[\frac{2a^2 b}{(b^2 - a^2)\kappa^{I}(\nu^{I2} - 1)} \right] + p_c \left[\frac{2bc^2}{(b^2 - c^2)\kappa^{II}(\nu^{II2} - 1)} \right].
\end{aligned}
\tag{A.10}
$$

A.2 Plane stress solution for a bimaterial cantilever beam

The solution to the problem of the bending of a composite cantilever beam has been presented in Muskhelishvili's work [22] for the case of a circular cross-section. For the case of a rectangular cross-section (Fig. A.2), as was the case of the example presented in Chapter 2, a similar procedure can be followed, and it is briefly outlined here.

A.2.1 Exact solution for zero Poisson's ratio

At first, an exact solution is sought to the case of zero Poisson's ratio ($\nu = 0$). The exact stress distribution will then be used to obtain a plane stress solution for the cases of non-zero values of Poisson's ratio. Assuming a displacement field of the form:

$$
\begin{aligned}
u &= A(\frac{\ell z^2}{2} - \frac{z^3}{6}), \\
v &= 0, \\
w &= -A[x(\ell z - \frac{z^2}{2}) + \chi + xy^2],
\end{aligned}
\tag{A.11}
$$

where A is a constant to be determined, a corresponding stress field can be found:

$$
\begin{aligned}
\sigma_{xx} &= \sigma_{yy} = \sigma_{xy} = 0, \\
\sigma_{xz} &= -B_j(\frac{\partial \chi}{\partial x} + y^2), \\
\sigma_{yz} &= -B_j(\frac{\partial \chi}{\partial y} + 2xy), \\
\sigma_{zz} &= -K_j(\ell - z)x.
\end{aligned}
\tag{A.12}
$$

χ is a function which must be harmonic for equilibrium and B_j, K_j

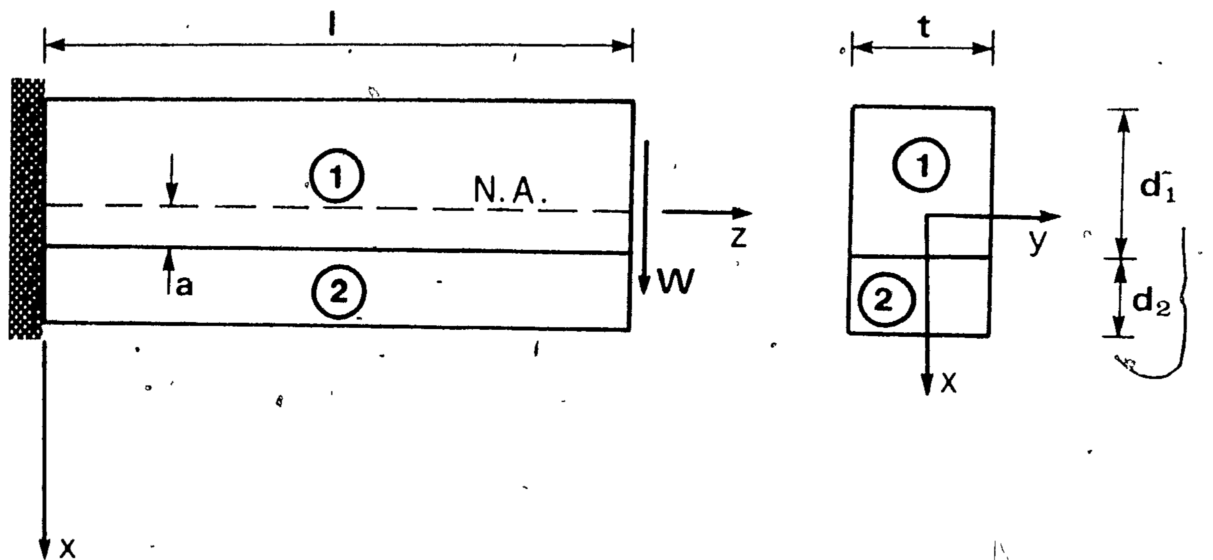

Figure A.2: Composite cantilever beam.

are constants given by:

$$\begin{aligned} B_1 &= \frac{AE_1}{2}, \\ B_2 &= \frac{AE_2}{2}, \\ K_1 &= AE_1, \\ K_2 &= AE_2. \end{aligned} \tag{A.13}$$

Letting $\sigma_{yz} = 0$, then, $\frac{\partial \chi}{\partial y} = -2xy$, and

$$\chi = -xy^2 + f(x). \tag{A.14}$$

Since χ is harmonic:

$$\frac{\partial^2 \chi}{\partial x^2} + \frac{\partial^2 \chi}{\partial y^2} = 0. \tag{A.15}$$

Substituting Eq. (A.14) into (A.15) yields:

$$f''(x) - 2x = 0. \tag{A.16}$$

The function $f(x)$ is therefore given by:

$$f_j(x) = \frac{x^3}{3} + C_{ij}x + C_{2j}, \tag{A.17}$$

where C_{ij} and C_{2j} are constants to be determined. Substituting Eq. (A.17) into (A.14), the function χ is described by:

$$\chi_j = -xy^2 + \frac{x^3}{3} + C_{ij}x + C_{2j}. \tag{A.18}$$

The stress field (Eqs. (A.12) can now be rewritten using Eqs. (A.18) and (A.13):

$$\begin{aligned} \sigma_{xz} &= -\frac{AE_j}{2}(x^2 + C_{ij}), \\ \sigma_{yz} &= 0, \\ \sigma_{zz} &= -AE_j(\ell - z)x. \end{aligned} \tag{A.19}$$

Now; σ_{xz} must be zero at the top and bottom surfaces, and also σ_{xz} must be continuous at the interface. These stress boundary conditions can be written as:

$$\begin{aligned} \sigma_{xz}^1 &= 0 \quad ; \quad \text{at} \quad x = -(d_1 - a), \\ \sigma_{xz}^2 &= 0 \quad ; \quad \text{at} \quad x = (d_2 + a), \\ \sigma_{xz}^1 &= \sigma_{xz}^2 \quad ; \quad \text{at} \quad x = a. \end{aligned} \tag{A.20}$$

Substituting Eqs. (A.19) into Eqs. (A.20), the constants C_{ij}, and the location a, of the neutral axis can be determined:

$$\begin{aligned} C_{11} &= -(d_1 - a)^2, \\ C_{12} &= -(d_2 + a)^2, \\ a &= \frac{E_1 d_1^2 - E_2 d_2^2}{2(E_1 d_1 + E_2 d_2)} \end{aligned} \tag{A.21}$$

The applied load, W, can be expressed as:

$$W = \int_{-(d_1-a)}^{a} \sigma_{xz}^1 \, t \, dx + \int_{a}^{(d_2+a)} \sigma_{xz}^2 \, t \, dx, \tag{A.22}$$

and by substituting Eqs. (A.19) into Eq. (A.22), and making use of Eqs. (A.21), the constant A can be determined:

$$A = \frac{W}{t(\frac{E_1 d_1^3}{3} + \frac{E_2 d_2^3}{3} - \frac{E_1 d_1^2 a}{2} + \frac{E_2 d_2^2 a}{2})} \tag{A.23}$$

The stress field is, therefore, completely defined:

$$\begin{aligned} \sigma_{xz}^1 &= -\frac{AE_1}{2}[x^2 - (d_1 - a^2], \\ \sigma_{xz}^2 &= -\frac{AE_2}{2}[x^2 - (d_2 + a)^2], \\ \sigma_{zz}^1 &= -AE_1(\ell - z)x, \\ \sigma_{zz}^2 &= -AE_2(\ell - z)x, \end{aligned} \tag{A.24}$$

where a is defined in Eqs (A.21) and A is given by Eq. (A.23).

A.2.2 Plane stress solution for non-zero values of Poisson's ratio

In this plane stress solution, the stress field is assumed to have the same form as in the case for zero Poisson's ratio, and a displacement field is sought satisfying the boundary conditions of the problem.

Hence, let the stress distribution be:

$$\begin{aligned}
\sigma^{j}_{xz} &= \frac{-AE_j}{2}(x^2 + C_{ij}), \\
\sigma^{j}_{zz} &= -AE_j(\ell - z), \\
\sigma_{xx} &= 0,
\end{aligned} \tag{A.25}$$

where C_{ij} and A are given by Eqs. (A.21) and (A.23) respectively.

The strains are therefore given by:

$$\begin{aligned}
\varepsilon^{j}_{xx} &= \frac{\sigma_{xx}}{E_j} - \frac{\nu_j}{E_j}\sigma_{zz} = \nu_j A(\ell - z)x, \\
\varepsilon^{j}_{zz} &= \frac{\sigma_{zz}}{E_j} - \frac{\nu_j}{E_j}\sigma_{xx} = -A(\ell - z)x, \\
\varepsilon^{j}_{xz} &= \frac{\sigma^{j}_{xz}}{E_j}(1 + \nu_j) = \frac{-(1 + \nu_j)}{2}A(x^2 + C_{ij}).
\end{aligned} \tag{A.26}$$

Invoking the strain-displacement conditions:

$$\begin{aligned}
\varepsilon_{xx} &= \frac{\partial u}{\partial x}, \\
\varepsilon_{zz} &= \frac{\partial w}{\partial z}, \\
\varepsilon_{xz} &= \frac{1}{2}(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}).
\end{aligned} \tag{A.27}$$

The first two of Eqs. (A.26) then yield the following expressions for the displacements:

$$u = A\nu_j(\ell z)\frac{x^2}{2} + f(z),$$

$$w = -A(\ell z - \frac{z^2}{2})x + g(x). \tag{A.28}$$

Substituting Eqs. (A.28) into the third of Eqs. (A.26), the functions $f(z), g(x)$ can be expressed as:

$$\begin{aligned} f(z) &= A(\frac{\ell z^2}{2} - \frac{z^3}{6}) + M_j z + N_j, \\ g(x) &= -A(2+\nu_j)\frac{x^3}{6} + S_j x + Q_j, \end{aligned} \tag{A.29}$$

where M_j, N_j, S_j, and Q_j are constants to be determined. The displacement field is now written as:

$$\begin{aligned} u^j &= A\nu_j(\ell - z)\frac{x^2}{2} + A(\frac{\ell z^2}{2} - \frac{z^3}{6}) + M_j z + N_j, \\ w^j &= -A(\ell z - \frac{z^2}{2})x - A(2+\nu_j)\frac{x^3}{6} + S_j x + Q_j. \end{aligned} \tag{A.30}$$

The boundary conditions that must be satisfied by the displacements are prevention against rigid body translation and rotation at the fixed end and continuity at the interface. These can be expressed as:

$$\begin{aligned} u^j &= w^j = 0 \quad ; \quad \text{at} \quad x = a \quad \text{and} \quad z = 0, \\ u^1 &= u^2 \quad ; \quad \text{at} \quad x = a, \\ w^1 &= w^2 \quad ; \quad \text{at} \quad x = a, \\ \frac{\partial u_j}{\partial z} &= 0 \quad ; \quad \text{at} \quad x = a \quad \text{and} \quad z = 0. \end{aligned} \tag{A.31}$$

Substituting Eqs. (A.30) into Eq. (A.31) the constants N_j, M_j, S_j, and Q_j can be determined:

$$
\begin{aligned}
N_1 &= \frac{-A\nu_1 \ell a^2}{2}, \\
N_2 &= \frac{-A\nu_2 \ell a^2}{2}, \\
M_1 &= A\nu_1 \frac{a^2}{2}, \\
M_2 &= A\nu_2 \frac{a^2}{2}, \\
S_1 &= -A[(1+\nu_1)C_{11} + \nu_1 \frac{a^2}{2}], \\
S_2 &= -A[(1+\nu_2)C_{12} + \nu_2 \frac{a^2}{2}], \\
Q_1 &= A(2+\nu_1)\frac{a^3}{6} + Aa[(1+\nu_1)C_{11} + \nu_1 \frac{a^2}{2}], \\
Q_2 &= A(2+\nu_2)\frac{a^3}{6} + Aa[(1+\nu_2)C_{12} + \nu_2 \frac{a^2}{2}].
\end{aligned}
\tag{A.32}
$$

The displacement field is now fully defined and satisfies the boundary conditions for the problem. The stress field represented in Eqs. (A.25) therefore constitutes a plane stress solution to the problem.

Appendix B

LISTING OF SOURCE CODE OF DEVELOPED FINITE ELEMENT PROGRAMS

B.1 Two-dimensional program

```
C  TWO-DIMENSIONAL FINITE ELEMENT PROGRAM
C  FOR THE ANALYSIS OF PLANE STRESS,
C  PLANE STRAIN, AND AXISYMMETRIC PROBLEMS
C
C  BY  MICHAEL ANGELIDES
C
C  MCGILL UNIVERSITY,
C  DEPARTMENT OF CIVIL ENGINEERING
C  AND APPLIED MECHANICS
C
C  MARCH  1986
C
C  LIBRARY OF ELEMENTS:
C  - BAR ELEMENT
C  - CONSTANT STRAIN TRIANGLE
C  - LINEAR QUADRILATERAL (ISOPARAMETRIC)
C  - INTERFACE ELEMENT
C
```

```
C
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER N,NEL,NDOF,NLOAD,DOF,NGEN,INTER,MEQNS,KEQNS,REDOF,NCOUNT
      INTEGER NID(1000),BCX(1000),BCY(1000),ELID(1000),NTYPE(1000)
      INTEGER N1(1000),N2(1000),N3(1000),N4(1000),N5(1000),NODRED(1000)
      INTEGER ELIDB,N11,N22,N33,N44,N55,NELA(1000),NELB(1000)
      INTEGER NODFOR(10),KK1,KK2,IFPRE(2000),MM,NGAUS1,NGAUS2,INCOMP
      DOUBLE PRECISION X(1000),Y(1000),XDEF(1000),YDEF(1000)
      DOUBLE PRECISION A(1000),E(1000),NU(1000),T(1000),FX(10),FY(10)
      DOUBLE PRECISION KGLOB(300000),ASLOD(2000)
      DOUBLE PRECISION LOAD(2000),FIXED(2000),REACT(2000),XDISP(2000)
      DOUBLE PRECISION KEL(20,20)
      CHARACTER*80 TITLE
      COMMON/GLOB/X,Y,A,E,NU,T
      DATA MSTIF/300000/
      READ *,TITLE
      READ *,NCASE,N,NEL,NDOF,NLOAD,NGEN,INTER,NGAUS1,NGAUS2,INCOMP
      MEQNS=1000
      KEQNS=1000
      DOF=N*NDOF
      IF(NGEN .EQ. 1)THEN
        CALL GENER(N,NDOF,BCX,BCY,NEL,NID,X,Y,XDEF,YDEF,ELID,
     +             NTYPE,N1,N2,N3,N4,N5,A,E,NU,T,KEQNS,MEQNS)
      ELSE
        READ *,(NID(I),X(I),Y(I),BCX(I),BCY(I),XDEF(I),YDEF(I),
     +         I=1,N)
        READ *,(ELID(I),NTYPE(I),N1(I),N2(I),N3(I),N4(I),N5(I),A(I),
     +         E(I),NU(I),T(I),I=1,NEL)
      END IF
      IF(INTER .NE. 0) THEN
         READ *,(NELA(I),NELB(I),I=1,INTER)
      END IF
      IF(NLOAD .NE. 0) THEN
        READ *,(NODFOR(I),FX(I),FY(I),I=1,NLOAD)
      END IF
      DOF=N*NDOF
      NCOUNT=0
      DO 40 I=1,N
         NODRED(NID(I))=NID(I)-NCOUNT
```

```
      IF(INTER .NE. 0) THEN
        DO 30 J=1,INTER
          IF(NID(I) .EQ. NB(NELA(J))) THEN
            NODRED(NID(I))=0
            NCOUNT=NCOUNT+1
          END IF
30      CONTINUE
      END IF
40 CONTINUE
   REDOF=DOF-NDOF*INTER
   CALL DATA(N,NDOF,DOF,BCX,BCY,NEL,NID,X,Y,XDEF,YDEF,ELID,NTYPE,N1,
  +          N2,N3,N4,N5,A,E,NU,T,NODFOR,FX,FY,KEQNS,MEQNS,NLOAD,
  +          INTER,NGAUS1,NGAUS2,TITLE,INCOMP,NCASE)
   NBANDW=0
   DO 900 I=1,MSTIF
       KGLOB(I)=0.0D0
900 CONTINUE
   DO 1005 I=1,NEL
     MM=2*NTYPE(I)
     NCOND=1
     IF(NTYPE(I) .EQ. 5) THEN
       NCOND=0
       DO 1000 J=1,INTER
         IF(ELID(I) .EQ. NELA(J)) THEN
           ELIDB=NELB(J)
           N11=N1(ELIDB)
           N22=N2(ELIDB)
           N33=N3(ELIDB)
           N44=N4(ELIDB)
           N55=N5(ELIDB)
           MM=12
           CALL STIFF(ELID(I),ELIDB,NTYPE(I),N1(I),N2(I),N3(I),
  +                   N4(I),N5(I),N11,N22,N33,N44,N55,KGLOB,KEL,
  +                   MEQNS,NEL,MM,NGAUS1,NGAUS2,NODRED,INCOMP,
  +                   NCASE,NBAND,MSTIF,REDOF)
           IF(NBAND .GT. NBANDW) NBANDW=NBAND
         END IF
1000   CONTINUE
     END IF
```

```
          IF(NCOND .EQ. 1) THEN
          CALL STIFF(ELID(I),ELIDB,NTYPE(I),N1(I),N2(I),N3(I),N4(I),N5(I)
     +              ,N11,N22,N33,N44,N55,KGLOB,KEL,MEQNS,NEL,MM,NGAUS1,
     +               NGAUS2,NODRED,INCOMP,NCASE,NBAND,MSTIF,REDOF)
          IF(NBAND .GT. NBANDW) NBANDW=NBAND
          END IF
 1005 CONTINUE
      NHALF=NBANDW+1
      DO 1020 I=1,REDOF
          ASLOD(I)=0.0D0
 1020 CONTINUE
      IF(NLOAD .NE. 0) THEN
          DO 1030 I=1,NLOAD
             KK1=2*NODRED(NODFOR(I))-1
             KK2=KK1+1
             ASLOD(KK1)=FX(I)
             ASLOD(KK2)=FY(I)
 1030     CONTINUE
      END IF
      DO 1035 I=1,REDOF
          LOAD(I)=ASLOD(I)
 1035 CONTINUE
      DO 1040 I=1,N
          K=NODRED(NID(I))
          IF(K .NE. 0) THEN
            IFPRE(2*K-1)=BCX(NID(I))
            IFPRE(2*K)  =BCY(NID(I))
            FIXED(2*K-1)=XDEF(NID(I))
            FIXED(2*K)  =YDEF(NID(I))
          END IF
 1040 CONTINUE
      PRINT 1041
 1041 FORMAT('-')
      PRINT 1042,NHALF
 1042 FORMAT('- SEMI BANDWIDTH FOR STRUCTURE MODEL : ',I5)
      CALL GREDUC(MEQNS,ASLOD,KGLOB,IFPRE,FIXED,NEQNS,MSTIF,REDOF,
     +            NBANDW)
      CALL BAKSUB(MEQNS,ASLOD,KGLOB,IFPRE,FIXED,XDISP,REACT,NEQNS,MSTIF,
     +            REDOF,NBANDW)
```

```
      CALL DISPL(N,DOF,NID,NODRED,N1,N2,N3,N4,N5,NELA,NELB,INTER,
     +           XDISP,REACT,MEQNS,NCASE)
      PRINT 1100
 1100 FORMAT('1 ELEMENT STRESSES')
      DO 1200 I=1,NEL
         IF(NTYPE(I) .EQ  2)THEN
           CALL FORCE(ELID(I),N1(I),N2(I),X(N1(I)),Y(N1(I)),X(N2(I)),
     +                Y(N2(I)),A(I),E(I),MEQNS,XDISP)
         ELSE
           IF(NTYPE(I) .EQ. 3) THEN
             IF(NCASE  EQ. 4) THEN
               CALL STRIAX(ELID(I),N1(I),N2(I),N3(I),X(N1(I)),Y(N1(I)),
     +                     X(N2(I)),Y(N2(I)),X(N3(I)),Y(N3(I)),E(I),
     +                     NU(I),XDISP,MEQNS)
             ELSE
               CALL STR(ELID(I),N1(I),N2(I),N3(I),X(N1(I)),Y(N1(I)),
     +                  X(N2(I)),Y(N2(I)),X(N3(I)),Y(N3(I)),E(I),NU(I),
     +                  T(I),XDISP,MEQNS)
             END IF
           ELSE
             IF(NTYPE(I)  EQ. 4) THEN
               X5=0.D0
               Y5=0.D0
             ELSE
               X5=X(N5(I))
               Y5=Y(N5(I))
             END IF
             IF(NCASE .EQ. 4) THEN
               CALL STRAX(ELID(I),N1(I),N2(I),N3(I),N4(I),N5(I),
     +                    X(N1(I)),Y(N1(I)),X(N2(I)),Y(N2(I)),X(N3(I)),
     +                    Y(N3(I)),X(N4(I)),Y(N4(I)),X5,Y5,E(I),NU(I),
     +                    XDISP,MEQNS,NTYPE(I))
             ELSE
               CALL STRES(ELID(I),N1(I),N2(I),N3(I),N4(I),N5(I),
     +                    X(N1(I)),Y(N1(I)),X(N2(I)),Y(N2(I)),X(N3(I)),
     +                    Y(N3(I)),X(N4(I)),Y(N4(I)),X5,Y5,E(I),NU(I),
     +                    T(I),XDISP,MEQNS,NTYPE(I),INCOMP)
             END IF
           END IF
```

```
        END IF
 1200 CONTINUE
      PRINT 9000
 9000 FORMAT('1')
      STOP
      END
C
C
      SUBROUTINE DATA(N,NDOF,DOF,BCX,BCY,NEL,NID,X,Y,XDEF,YDEF,
     +                ELID,NTYPE,N1,N2,N3,N4,N5,A,E,NU,T,NODFOR,FX,FY,
     +                KEQNS,MEQNS,NLOAD,INTER,NGAUS1,NGAUS2,TITLE,
     +                INCOMP,NCASE)
      IMPLICIT REAL*8 (A-H,O-Z)
      DOUBLE PRECISION NU
      DIMENSION BCX(KEQNS),BCY(KEQNS),NID(KEQNS),X(KEQNS),Y(KEQNS)
      DIMENSION XDEF(KEQNS),YDEF(KEQNS),ELID(MEQNS),N1(MEQNS),FY(KEQNS)
      DIMENSION N2(MEQNS),A(MEQNS),E(MEQNS),NODFOR(KEQNS),FX(KEQNS)
      DIMENSION N3(MEQNS),N4(MEQNS),N5(MEQNS),NU(MEQNS),T(MEQNS)
      DIMENSION NTYPE(MEQNS)
      CHARACTER*5 TYPE(1000)
      CHARACTER*80 TITLE
      CHARACTER*3 MODES
      INTEGER DOF,BCX,BCY,ELID,INCOMP
      PRINT 110,TITLE
  100 FORMAT('- PLANE STRESS ANALYSIS  -   DATA  ECHO')
  102 FORMAT('- PLANE STRAIN ANALYSIS  -   DATA  ECHO')
  105 FORMAT('- AXISYMMETRIC ANALYSIS  -   DATA  ECHO')
      IF(NCASE .EQ. 2) THEN
        PRINT 100
      ELSE
        IF(NCASE .EQ. 3) THEN
          PRINT 102
        ELSE
          IF(NCASE .EQ. 4) THEN
            PRINT 105
          END IF
        END IF
      END IF
  110 FORMAT('1',A80)
```

```
      PRINT 120,N
  120 FORMAT('- NUMBER OF JOINTS',T32,':',I5)
      PRINT 130,NDOF
  130 FORMAT('- NUMBER OF D.O.F. PER JOINT',T32,':',I3)
      PRINT 140,DOF
  140 FORMAT('- TOTAL NUMBER OF D.O.F.',T32,':',I5)
      NRES=0
      DO 160 I=1,N
         NRES=NRES+BCX(I)+BCY(I)
  160 CONTINUE
      PRINT 170,NRES
  170 FORMAT('- NUMBER OF RESTRAINED D.O.F.',T32,':',I5)
      PRINT 180,DOF-NRES
  180 FORMAT('- NUMBER OF UNRESTRAINED D.O.F.',T32,':',I5)
      NEL1=0
      NEL2=0
      NEL3=0
      NEL4=0
      DO 185 I=1,NEL
         IF(NTYPE(I) .EQ. 2) THEN
           NEL1=NEL1+1
           TYPE(I)=' BAR'
         ELSE
           IF(NTYPE(I) .EQ. 3) THEN
             NEL2=NEL2+1
             IF(NCASE .EQ. 4) THEN
               TYPE(I)='TRIAX'
             ELSE
               TYPE(I)='C S T'
             END IF
           ELSE
             IF(NTYPE(I) .EQ. 4) THEN
               NEL3=NEL3+1
               TYPE(I)='QUAD4'
             ELSE
               NEL4=NEL4+1
               TYPE(I)='QUAD5'
             END IF
           END IF
```

```
      END IF
185 CONTINUE
    PRINT 190,NEL1
190 FORMAT('- NUMBER OF BAR ELEMENTS',T32,':',I5)
    PRINT 195,NEL2
195 FORMAT('- NUMBER OF TRIANG ELEMENTS',T32,':',I5)
    PRINT 196,NEL3
196 FORMAT('- NUMBER OF QUAD4 ELEMENTS',T32,':',I5)
    PRINT 197,NEL4
197 FORMAT('- NUMBER OF INTERFACE ELEMENTS',T32,':',I5)
    PRINT 198,NGAUS1,NGAUS1
198 FORMAT('- QUAD4 INTEGRATION ORDER',T32,':',I2,' BY',I2)
    PRINT 199,NGAUS2,NGAUS2
199 FORMAT('- QUAD5 INTEGRATION ORDER',T32,':',I2,' BY',I2)
    IF(INCOMP .EQ. 1) THEN
      MODES = 'YES'
    ELSE
      MODES = 'NO'
    END IF
    PRINT 200,MODES
200 FORMAT('- INCOMPATIBLE BENDING MODES',T32,': ',A3)
    PRINT 209
209 FORMAT('1 NODE COORDINATES')
    PRINT 210
210 FORMAT('- NODE',T15,'X',T25,'Y',T32,'X-BC',T42,'Y-BC',T55,'X-DEF',
   +        T65,'Y-DEF')
    PRINT 220,(NID(I),X(I),Y(I),BCX(I),BCY(I),XDEF(I),YDEF(I),I=1,N)
220 FORMAT('-',I3,T10,F7.3,T20,F7.3,T32,I3,T42,I3,T52,F7 1,T62,F7 1)
    PRINT 230
230 FORMAT('1 ELEMENT INCIDENCES')
    PRINT 240
240 FORMAT('- ELEMENT',T11,'TYPE',T17,'NODE-1',T24,'NODE-2',T31,
   +       'NODE-3',T38,'NODE-4',T45,'NODE-5',T56,'AREA',T68,'E',
   +        T77,'NU',T87,'T')
    PRINT 250,(ELID(I),TYPE(I),N1(I),N2(I),N3(I),N4(I),N5(I),A(I),
   +           E(I),NU(I),T(I),I=1,NEL)
250 FORMAT('-',T4,I3,T11,A5,T19,I3,T26,I3,T33,I3,T40,I3,T47,I3,T53,
   +           E9.2,T63,E10.3,T76,F5.2,T83,E10.3)
    PRINT 260
```

```
260 FORMAT('1 APPLIED LOADS')
    IF(NLOAD .NE. 0) THEN
      PRINT 270
270   FORMAT('- NODE',T13,'X-FORCE',T28,'Y-FORCE')
      PRINT 280,(NODFOR(I),FX(I),FY(I),I=1,NLOAD)
280   FORMAT('-',I4,T11,E10.3,T26,E10.3)
    ELSE
      PRINT 290
290   FORMAT('- NO CONCENTRATED LOADS APPLIED')
    END IF
    RETURN
    END
C
C
    SUBROUTINE STIFF(ELID,ELIDB,NTYPE,N1,N2,N3,N4,N5,N11,N22,N33,N44,
   +                 N55,KGLOB,KEL,MEQNS,NEL,MM,NGAUS1,NGAUS2,NODRED,
   +                 INCOMP,NCASE,NBAND,MSTIF,REDOF)
    IMPLICIT REAL*8 (A-H,O-Z)
    DOUBLE PRECISION KPRIM(4,4),KEL(MM,MM),NU
    DOUBLE PRECISION KELA(10,10),KELB(10,10),KELARA(10,12)
    DOUBLE PRECISION KELBRB(10,12),KEL1(12,12),KEL2(12,12)
    DOUBLE PRECISION KGLOB(MSTIF)
    DIMENSION TR(4,4),B(3,10)
    DIMENSION Q(2,12),RA(10,12),RB(10,12),RAT(12,10)
    DIMENSION RBT(12,10)
    INTEGER N1,N2,N3,N4,N5,N11,N22,N33,N44,N55,ELID,ELIDB,NGLOB(12)
    INTEGER NOD(6),NODRED(MEQNS),INCOMP,REDOF
    COMMON/GLOB/X(1000),Y(1000),A(1000),E(1000),NU(1000),T(1000)
    NFUNC(I,J)=(J-I)*(2*REDOF + 1 - J + I)/2 + I
    NNTYPE=NTYPE
    NDIM=NTYPE*2
    NOD(1)=N1
    NOD(2)=N2
    NOD(3)=N3
    NOD(4)=N4
    IF(NTYPE .EQ. 2) THEN
      CALL BAR(ELID,N1,N2,X(N1),Y(N1),X(N2),Y(N2),A(ELID),E(ELID),
   +           KEL,TR,KPRIM)
    ELSE
```

```
      IF(NTYPE .EQ. 3) THEN
        IF(NCASE .EQ. 4) THEN
          CALL TRIAX(ELID,N1,N2,N3,X(N1),Y(N1),X(N2),Y(N2),X(N3),
     +               Y(N3),E(ELID),NU(ELID),KEL)
        ELSE
          CALL CST(ELID,N1,N2,N3,X(N1),Y(N1),X(N2),Y(N2),X(N3),Y(N3),
     +             E(ELID),NU(ELID),T(ELID),B,KEL)
        END IF
      ELSE
        IF(NTYPE .EQ. 4) THEN
          IF(NCASE .EQ. 4) THEN
            CALL QUADAX(NGAUS1,ELID,NTYPE,N1,N2,N3,N4,N5,KEL,NDIM)
          ELSE
            CALL QUAD4(NGAUS1,ELID,NTYPE,N1,N2,N3,N4,N5,KEL,NDIM,
     +                 INCOMP)
          END IF
        ELSE
          IF(NCASE .EQ. 4) THEN
            CALL RELAX(ELID,ELIDB,N1,N2,N3,N4,N5,N11,N22,N33,N44,N55,
     +                 Q)
            CALL QUADAX(NGAUS2,ELID,NTYPE,N1,N2,N3,N4,N5,KELA,NDIM)
            CALL QUADAX(NGAUS2,ELIDB,NTYPE,N11,N22,N33,N44,N55,KELB,
     +                  NDIM)
          ELSE
            CALL REL(ELID,ELIDB,N1,N2,N3,N4,N5,N11,N22,N33,N44,N55,Q)
            CALL QUAD4(NGAUS2,ELID,NTYPE,N1,N2,N3,N4,N5,KELA,NDIM,
     +                 INCOMP)
            CALL QUAD4(NGAUS2,ELIDB,NTYPE,N11,N22,N33,N44,N55,KELB,
     +                 NDIM,INCOMP)
          END IF
          DO 50 LI=1,8
             DO 50 LJ=1,12
                RA(LI,LJ)=0.D0
                RB(LI,LJ)=0.D0
50        CONTINUE
          RA(7,1)=1.D0
          RA(8,2)=1.D0
          RA(1,3)=1.D0
          RA(2,4)=1.D0
```

```
            DO 55 LK=3,6
               LM=LK+6
               RA(LK,LM)=1.DO
55          CONTINUE
            DO 58 KK=3,8
               RB(KK,KK)=1.DO
58          CONTINUE
            RB(1,9)=1.DO
            RB(2,10)=1.DO
            DO 60 K=9,10
               DO 60 L=1,12
                  MMM=K-8
                  RA(K,L)=Q(MMM,L)
                  RB(K,L)=Q(MMM,L)
60          CONTINUE
            DO 80 M=1,10
               DO 80 N=1,12
                  RAT(N,M)=RA(M,N)
                  RBT(N,M)=RB(M,N)
80          CONTINUE
            CALL MATMAT(10,10,12,KELA,RA,KELARA)
            CALL MATMAT(12,10,12,RAT,KELARA,KEL1)
            CALL MATMAT(10,10,12,KELB,RB,KELBRB)
            CALL MATMAT(12,10,12,RBT,KELBRB,KEL2)
            DO 90 K1=1,12
               DO 90 K2=1,12
                  KEL(K1,K2)=KEL1(K1,K2)+KEL2(K1,K2)
90          CONTINUE
          END IF
        END IF
      END IF
      IF(NTYPE .EQ. 5) THEN
        NOD(1)=N4
        NOD(2)=N22
        NOD(3)=N33
        NOD(4)=N44
        NOD(5)=N2
        NOD(6)=N3
        NNTYPE=6
```

```
      END IF
      DO 150 I=1,NNTYPE
         J=2*I - 1
         K=2*I
         NGLOB(J)=2*NODRED(NOD(I))-1
         NGLOB(K)=2*NODRED(NOD(I))
150   CONTINUE
      NBAND=0
      NNODE=MM/2
      DO 200 INODE=1,NNODE
         DO 200 IDOFN=1,2
            NROWS=(NODRED(NOD(INODE))-1)*2 + IDOFN
            NROWE=(INODE-1)*2 + IDOFN
            DO 200 JNODE=1,NNODE
               DO 200 JDOFN=1,2
                  NCOLS=(NODRED(NOD(JNODE))-1)*2 + JDOFN
                  NCOLE=(JNODE-1)*2 + JDOFN
                  IF(NCOLS .LT. NROWS) GO TO 200
                  NDIFF=NCOLS-NROWS
                  IF(NDIFF .GT. NBAND) NBAND=NDIFF
                  NPOS=NFUNC(NROWS,NCOLS)
                  KGLOB(NPOS)=KGLOB(NPOS) + KEL(NROWE,NCOLE)
200   CONTINUE
      RETURN
      END
C
C
      SUBROUTINE BAR(ELID,N1,N2,X1,Y1,X2,Y2,A,E,KEL,TR,KPRIM)
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER N1,N2,ELID,NGLOB(4)
      REAL*8 L,L2,COS,SIN,C2,S2,XDIF,YDIF,CS,KEL(4,4),K,LSMALL,TR(4,4)
      REAL*8 KPRIM(4,4)
      XDIF=X2-X1
      YDIF=Y2-Y1
      L2=XDIF*XDIF + YDIF*YDIF
      L=DSQRT(L2)
      COS=XDIF/L
      SIN=YDIF/L
      C2=COS*COS
```

```
      S2=SIN*SIN
      CS=COS*SIN
      K = A*E/L
      KEL(1,1)=C2*K
      KEL(1,2)=CS*K
      KEL(1,3)=-KEL(1,1)
      KEL(1,4)=-KEL(1,2)
      KEL(2,2)=S2*K
      KEL(2,3)=-KEL(1,2)
      KEL(2,4)=-KEL(2,2)
      KEL(3,3)=KEL(1,1)
      KEL(3,4)=KEL(1,2)
      KEL(4,4)=KEL(2,2)
      KEL(2,1)=KEL(1,2)
      KEL(3,1)=KEL(1,3)
      DO 1100 J=1,4
         DO 1100 I=1,4
            KEL(I,J)=KEL(J,I)
 1100 CONTINUE
      DO 1200 I=1,4
         DO 1200 J=1,4
            TR(I,J)=0.0
            KPRIM(I,J)=0.0
 1200 CONTINUE
      TR(1,1)=COS
      TR(1,2)=SIN
      TR(2,1)=-SIN
      TR(2,2)=COS
      TR(3,3)=COS
      TR(4,3)=-SIN
      TR(4,4)=COS
      KPRIM(1,1)=K
      KPRIM(1,3)=-K
      KPRIM(3,1)=-K
      KPRIM(3,3)=K
      RETURN
      END
C
C
```

```
      SUBROUTINE CST(ELID,N1,N2,N3,X1,Y1,X2,Y2,X3,Y3,E,NU,T,B,KEL)
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER ELID
      REAL*8 X1,Y1,X2,Y2,X3,Y3,NU,T,EM(3,3),B(3,6),KEL(6,6)
      REAL*8 BT(6,3),BTE(6,3),K1(6,6),A
      B(1,1)=Y2-Y3
      B(1,2)=0.0
      B(1,3)=Y3-Y1
      B(1,4)=0.0
      B(1,5)=Y1-Y2
      B(1,6)=0.0
      B(2,1)=0.0
      B(2,2)=X3-X2
      B(2,3)=0.0
      B(2,4)=X1-X3
      B(2,5)=0.0
      B(2,6)=X2-X1
      B(3,1)=B(2,2)
      B(3,2)=B(1,1)
      B(3,3)=B(2,4)
      B(3,4)=B(1,3)
      B(3,5)=B(2,6)
      B(3,6)=B(1,5)
      A=(X2*Y3-Y2*X3-X1*Y3+Y1*X3+X1*Y2-Y1*X2)/2.
      DO 20 I=1,3
         DO 20 J=1,6
            B(I,J)=B(I,J)/(2.*A)
   20 CONTINUE
      DO 50 I=1,6
         DO 50 J=1,3
            BT(I,J)=B(J,I)
   50 CONTINUE
      CALL YOUNG(EM,E,NU,3,0)
      CALL MATMAT(6,3,3,BT,EM,BTE)
      CALL MATMAT(6,3,6,BTE,B,K1)
      DO 100 I=1,6
         DO 100 J=1,6
            KEL(I,J)=K1(I,J)*T*A
  100 CONTINUE
```

```
      RETURN
      END
C
C
      SUBROUTINE QUAD4(NGAUSS,NELEM,NORDER,N1,N2,N3,N4,N5,STIFEL,NDIM,
     +                 INCOMP)
      IMPLICIT REAL*8 (A-H,O-Z)
      DOUBLE PRECISION JACOB,NU,KEL(14,14)
      DIMENSION XX(5),YY(5),PLACE(3,3),WGT(3,3),B(3,14),BTE(14,3)
      DIMENSION EM(3,3),STIFEL(NDIM,NDIM)
      COMMON/Q4/EN(5),JACOB(2,2)
      COMMON/GLOB/X(1000),Y(1000),A(1000),E(1000),NU(1000),T(1000)
      DATA PLACE(1,1),PLACE(2,1),PLACE(2,3),PLACE(3,1),PLACE(3,2)/
     +     5*0.000000000000000D0/
      DATA PLACE(1,2)/-0.577350269189626D0/
      DATA PLACE(2,2)/ 0.577350269189626D0/
      DATA PLACE(1,3)/-0.774596669241483D0/
      DATA PLACE(3,3)/ 0.774596669241483D0/
      DATA WGT(1,1)/2.000000000000000D0/,WGT(2,3)/0.888888888888889D0/
      DATA WGT(1,2),WGT(2,2)/2*1.000000000000000D0/
      DATA WGT(2,1),WGT(3,1),WGT(3,2)/3*0.000000000000000D0/
      DATA WGT(1,3),WGT(3,3)/2*0.555555555555556D0/
      CALL YOUNG(EM,E(NELEM),NU(NELEM),3,0)
      MORDER=2*NORDER
      NSIZE = NDIM + 2*INCOMP
      XX(1)=X(N1)
      XX(2)=X(N2)
      XX(3)=X(N3)
      XX(4)=X(N4)
      YY(1)=Y(N1)
      YY(2)=Y(N2)
      YY(3)=Y(N3)
      YY(4)=Y(N4)
      IF(NORDER .NE. 4) THEN
        XX(5)=X(N5)
        YY(5)=Y(N5)
      END IF
      DO 40 K=1,NSIZE
        DO 40 L=K,NSIZE
```

```
      KEL(K,L)=0.D0
40 CONTINUE
   DO 180 NA=1,NGAUSS
      XI=PLACE(NA,NGAUSS)
      DO 160 NB=1,NGAUSS
         ET=PLACE(NB,NGAUSS)
         CALL SHAPEF(XI,ET,XX,YY,DETJAC,B,NORDER,INCOMP)
         DV=WGT(NA,NGAUSS)*WGT(NB,NGAUSS)*T(NELEM)*DETJAC
         KORDER = NSIZE/2
         DO 80 J=1,KORDER
            L=2*J
            K=L-1
            DO 60 N=1,3
               BTE(K,N)=B(1,K)*EM(1,N) + B(3,K)*EM(3,N)
               BTE(L,N)=B(2,L)*EM(2,N) + B(3,L)*EM(3,N)
60          CONTINUE
80       CONTINUE
         DO 140 NROW=1,NSIZE
            DO 120 NCOL=NROW,NSIZE
               DUM=0.D0
               DO 100 J=1,3
                  DUM=DUM + BTE(NROW,J)*B(J,NCOL)
100            CONTINUE
               KEL(NROW,NCOL)=KEL(NROW,NCOL) + DUM*DV
120         CONTINUE
140      CONTINUE
160   CONTINUE
180 CONTINUE
   DO 200 K=1,NSIZE
      DO 200 L=K,NSIZE
         KEL(L,K)=KEL(K,L)
200 CONTINUE
   IF(INCOMP .EQ. 1) THEN
     DO 340 K=1,4
        LL = NSIZE - K
        KK = LL + 1
        DO 320 L=1,LL
           IF(KEL(KK,L) .EQ. 0.) GO TO 320
             DUM = KEL(KK,L)/KEL(KK,KK)
```

```
                DO 300 M=1,L
                   KEL(L,M) = KEL(L,M) - KEL(KK,M)*DUM
 300            CONTINUE
 320         CONTINUE
 340      CONTINUE
          LL=NSIZE-4
          DO 350 K=1,LL
             DO 350 L=1,K
                KEL(L,K) = KEL(K,L)
 350      CONTINUE
       END IF
       DO 400 I = 1,MORDER
          DO 400 J = 1,MORDER
             STIFEL(I,J)=KEL(I,J)
 400   CONTINUE
       RETURN
       END
C
C
       SUBROUTINE SHAPEF(XI,ET,XX,YY,DETJAC,B,NORDER,INCOMP)
       IMPLICIT REAL*8 (A-H,O-Z)
       DOUBLE PRECISION JACOB,NU
       DIMENSION RXI(5),RET(5),RK(5),RL(5)
       DIMENSION XX(5),YY(5),B(3,14),EM(3,3),ENXI(7),ENET(7)
       COMMON/Q4/EN(5),JACOB(2,2)
       COMMON/GLOB/X(1000),Y(1000),A(1000),E(1000),NU(1000),T(1000)
       DATA RXI/-1.,1.,1.,-1.,1./,RET/-1.,-1.,1.,1.,1./
       DATA RK/1.,1.,1.,1.,0./
       RL(1)=1.
       RL(2)=1.
       RL(3)=0.
       RL(4)=0.
       RL(5)=-2.
       IF(NORDER .EQ. 4) THEN
         RL(1)=0.
         RL(2)=0.
       END IF
       DO 20 L=1,NORDER
          F1=(1. + RXI(L)*XI)
```

```
      F2=(1. + RET(L)*ET)
      F3=(1. - XI*XI)
      F4=(1  - ET)
      EN(L)=RK(L)*F1*F2/4. - RL(L)*F3*F4/4.
      ENXI(L)=RK(L)*RXI(L)*F2/4. + RL(L)*XI*F4/2.
      ENET(L)=RK(L)*RET(L)*F1/4. + RL(L)*F3/4.
   20 CONTINUE
      MORDER = NORDER
      IF(INCOMP .EQ. 1) THEN
        LMIN = NORDER + 1
        LMAX = NORDER + 2
        ENXI(LMIN) = -2.DO*XI
        ENET(LMIN) = 0.DO
        ENXI(LMAX) = 0.DO
        ENET(LMAX) = -2.DO*ET
        MORDER = LMAX
      END IF
      NSIZE = 2*MORDER
      DO 40 I=1,3
         DO 40 J=1,NSIZE
            B(I,J)=0.DO
   40 CONTINUE
      JACOB(1,1)=0.DO
      JACOB(1,2)=0.DO
      JACOB(2,1)=0.DO
      JACOB(2,2)=0.DO
      DO 60 L=1,NORDER
         JACOB(1,1)=JACOB(1,1)+ENXI(L)*XX(L)
         JACOB(1,2)=JACOB(1,2)+ENXI(L)*YY(L)
         JACOB(2,1)=JACOB(2,1)+ENET(L)*XX(L)
         JACOB(2,2)=JACOB(2,2)+ENET(L)*YY(L)
   60 CONTINUE
      DETJAC=JACOB(1,1)*JACOB(2,2)-JACOB(1,2)*JACOB(2,1)
      F5=JACOB(1,1)/DETJAC
      JACOB(1,1)=JACOB(2,2)/DETJAC
      JACOB(1,2)=-JACOB(1,2)/DETJAC
      JACOB(2,1)=-JACOB(2,1)/DETJAC
      JACOB(2,2)=F5
      DO 80 J=1,MORDER
```

```
      L=2*J
      K=L-1
      B(1,K)=JACOB(1,1)*ENXI(J) + JACOB(1,2)*ENET(J)
      B(2,L)=JACOB(2,1)*ENXI(J) + JACOB(2,2)*ENET(J)
      B(3,K)=B(2,L)
      B(3,L)=B(1,K)
   80 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE REL(NELEM1,NELEM2,N1,N2,N3,N4,N5,N11,N22,N33,N44,N55,Q)
      IMPLICIT REAL*8 (A-H,O-Z)
      DOUBLE PRECISION NUA,NUB,NU
      DIMENSION XA(5),YA(5),XB(5),YB(5),TRAN1(2,3),BA(3,14),BB(3,14)
      DIMENSION EA(3,3),EB(3,3),PROD1(3,10),TRAN2(2,3),QA(2,10),QB(2,10)
      DIMENSION Q1(2,2),Q2(2,12),Q(2,12),XX1(5),XX2(5),YY1(5),YY2(5)
      COMMON/GLOB/X(1000),Y(1000),A(1000),E(1000),NU(1000),T(1000)
      PI=DACOS(-1.DO)
      DETJAC=O.DO
      XX1(1)=X(N1)
      XX1(2)=X(N2)
      XX1(3)=X(N3)
      XX1(4)=X(N4)
      XX1(5)=X(N5)
      YY1(1)=Y(N1)
      YY1(2)=Y(N2)
      YY1(3)=Y(N3)
      YY1(4)=Y(N4)
      YY1(5)=Y(N5)
      XX2(1)=X(N11)
      XX2(2)=X(N22)
      XX2(3)=X(N33)
      XX2(4)=X(N44)
      XX2(5)=X(N55)
      YY2(1)=Y(N11)
      YY2(2)=Y(N22)
      YY2(3)=Y(N33)
      YY2(4)=Y(N44)
```

```
      YY2(5)=Y(N55)
      CALL TRANSF(NELEM1,XX1(1),XX1(2),YY1(1),YY1(2),TRAN1,2,3)
      CALL TRANSF(NELEM2,XX2(1),XX2(2),YY2(1),YY2(2),TRAN2,2,3)
      XI=0.DO
      ET=-1 DO
      CALL SHAPEF(XI,ET,XX1,YY1,DETJAC,BA,5,0)
      CALL YOUNG(EA,E(NELEM1),NU(NELEM1),3,0)
      CALL SHAPEF(XI,ET,XX2,YY2,DETJAC,BB,5,0)
      CALL YOUNG(EB,E(NELEM2),NU(NELEM2),3,0)
      CALL MATMAT(3,3,10,EA,BA,PROD1)
      CALL MATMAT(2,3,10,TRAN1,PROD1,QA)
      CALL MATMAT(3,3,10,EB,BB,PROD1)
      CALL MATMAT(2,3,10,TRAN2,PROD1,QB)
      F1=(QB(1,9)-QA(1,9))*(QB(2,10)-QA(2,10))
      F2=(QB(1,10)-QA(1,10))*(QB(2,9)-QA(2,9))
      DETQ=(F1-F2)
      Q1(1,1)=(QB(2,10)-QA(2,10))/DETQ
      Q1(1,2)=-(QB(1,10)-QA(1,10))/DETQ
      Q1(2,1)=-(QB(2,9)-QA(2,9))/DETQ
      Q1(2,2)=(QB(1,9)-QA(1,9))/DETQ
      DO 80 L=1,2
         Q2(L,1)=QA(L,7)
         Q2(L,2)=QA(L,8)
         Q2(L,3)=QA(L,1)-QB(L,3)
         Q2(L,4)=QA(L,2)-QB(L,4)
         Q2(L,5)=-QB(L,5)
         Q2(L,6)=-QB(L,6)
         Q2(L,7)=-QB(L,7)
         Q2(L,8)=-QB(L,8)
         Q2(L,9)=QA(L,3)-QB(L,1)
         Q2(L,10)=QA(L,4)-QB(L,2)
         Q2(L,11)=QA(L,5)
         Q2(L,12)=QA(L,6)
   80 CONTINUE
      CALL MATMAT(2,2,12,Q1,Q2,Q)
      RETURN
      END
C
C
```

```
      SUBROUTINE YOUNG(EM,E,NU,NDIM,NPLANE)
      REAL*8 E,NU,EM(NDIM,NDIM),COEF1,COEF2
      DO 20 I=1,NDIM
         DO 20 J=1,NDIM
            EM(I,J)=0.DO
   20 CONTINUE
      IF(NDIM .EQ. 3) THEN
        IF(NPLANE .EQ. 0) THEN
          COEF1=E/(1.DO-NU*NU)
          EM(1,1)=COEF1
          EM(1,2)=COEF1*NU
          EM(2,1)=EM(1,2)
          EM(2,2)=COEF1
          EM(3,3)=COEF1*(1.DO-NU)/2.DO
        ELSE
          COEF2=E/((1.DO+NU)*(1.DO-2.DO*NU))
          EM(1,1)=COEF2*(1.DO-NU)
          EM(1,2)=COEF2*NU
          EM(2,1)=EM(1,2)
          EM(2,2)=EM(1,1)
          EM(3,3)=COEF2*(1.DO-2.DO*NU)/2.DO
        END IF
      ELSE
        COEF2=E/((1.DO+NU)*(1.DO-2.DO*NU))
        EM(1,1)=COEF2*(1.DO-NU)
        EM(1,2)=COEF2*NU
        EM(1,3)=EM(1,2)
        EM(2,1)=EM(1,2)
        EM(2,2)=EM(1,1)
        EM(2,3)=EM(1,2)
        EM(3,1)=EM(1,2)
        EM(3,2)=EM(1,2)
        EM(3,3)=EM(1,1)
        EM(4,4)=COEF2*(1.DO-2.DO*NU)/2.DO
        IF(NDIM .EQ. 6) THEN
          EM(5,5)=EM(1,1)
          EM(6,6)=EM(1,1)
        END IF
      END IF
```

```
      RETURN
      END
C
C
      SUBROUTINE MATVEC(N,M,A,Z,V,NEQNS,MEQNS)
      DIMENSION A(NEQNS,MEQNS),Z(MEQNS)
      REAL*8 SUM,A,V(N),Z
      DO 40 I=1,N
         SUM=0.0
         DO 20 J=1,M
            SUM=SUM+A(I,J)*Z(J)
20       CONTINUE
         V(I)=SUM
40    CONTINUE
      RETURN
      END
C
C
      SUBROUTINE DOT(N,A,B,PRODUC,MEQNS)
      DIMENSION A(MEQNS),B(MEQNS)
      REAL*8 A,B,PRODUC,SUM
      SUM=0.0
      DO 20 I=1,N
         SUM=SUM + A(I)*B(I)
20    CONTINUE
      PRODUC=SUM
      RETURN
      END
C
C
      SUBROUTINE MATMAT(M,N,K,A,B,C)
      INTEGER M,N,K,R,S,I
      DOUBLE PRECISION A(M,N),B(N,K),C(M,K),SUM
      R=1
      DO 80 WHILE (R .LE. M)
            S=1
            DO 60 WHILE(S .LE. K)
                 SUM=0.0
                 I=1
```

```
            DO 40 WHILE(I .LE. N)
                  SUM=SUM+A(R,I)*B(I,S)
                  I=I+1
 40         CONTINUE
            C(R,S)=SUM
            S=S+1
 60      CONTINUE
         R=R+1
 80   CONTINUE
      RETURN
      END
C
C
      SUBROUTINE GREDUC(MEQNS,ASLOD,ASTIF,IFPRE,FIXED,NEQNS,MSTIF,
     +                  REDOF,NBAND)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION ASLOD(2000),ASTIF(MSTIF),FIXED(2000)
      DIMENSION IFPRE(2000)
      INTEGER IEQNS,IEQN1,ICOLS,REDOF
      REAL*8 PIVOT,FACTOR
      NFUNC(I,J)=(J-I)*(2*REDOF + 1 - J + I)/2 + I
      NEQNS=REDOF
      DO 50 IEQNS=1,NEQNS
         NLOCA=IEQNS+NBAND
         IF(NLOCA .GT. NEQNS) NLOCA=NEQNS
         IF(IFPRE(IEQNS) .EQ. 1) THEN
           DO 40 IROWS=IEQNS,NLOCA
              NPOS=NFUNC(IEQNS,IROWS)
              ASLOD(IROWS)=ASLOD(IROWS)-ASTIF(NPOS)*FIXED(IEQNS)
 40        CONTINUE
           NPOS3=NFUNC(IEQNS,IEQNS)
           ASTIF(NPOS3)=0.DO
           GO TO 50
         END IF
         NPOS=NFUNC(IEQNS,IEQNS)
         PIVOT=ASTIF(NPOS)
         IF(DABS(PIVOT) .LE. 0.1E-8)THEN
           PRINT 100,IEQNS
100        FORMAT('1',5X,'INCORRECT PIVOT',5X,'EQUATION NUMBER : ',I6)
```

```
        STOP
      END IF
      IF(IEQNS .EQ. NEQNS)THEN
        GO TO 50
      END IF
      IEQN1=IEQNS+1
      DO 20 IROWS = IEQN1,NLOCA
         NPOS=NFUNC(IEQNS,IROWS)
         FACTR=ASTIF(NPOS)/PIVOT
         IF(FACTR .EQ. 0.0) GO TO 20
         DO 10 ICOLS=IEQN1,NLOCA
            IF(IROWS .GT. ICOLS) GO TO 10
            NPOS2=NFUNC(IROWS,ICOLS)
            NPOS3=NFUNC(IEQNS,ICOLS)
            ASTIF(NPOS2)=ASTIF(NPOS2)-FACTR*ASTIF(NPOS3)
10       CONTINUE
         ASLOD(IROWS)=ASLOD(IROWS)-FACTR*ASLOD(IEQNS)
20 CONTINUE
50 CONTINUE
   RETURN
   END
C
C
   SUBROUTINE BAKSUB(MEQNS,ASLOD,ASTIF,IFPRE,FIXED,XDISP,REACT,
  +                  NEQNS,MSTIF,REDOF,NBAND)
   IMPLICIT REAL*8 (A-H,O-Z)
   DIMENSION ASLOD(2000),ASTIF(MSTIF),IFPRE(2000)
   DIMENSION FIXED(2000),XDISP(2000),REACT(2000)
   INTEGER NEQN1,NBACK,NBAC1,REDOF
   REAL*8 PIVOT,RESID
   NFUNC(I,J)=(J-I)*(2*REDOF + 1 - J + I)/2 + I
   NEQNS=REDOF
   DO 5 IEQNS=1,NEQNS
      REACT(IEQNS)=0.0
 5 CONTINUE
   NEQN1=NEQNS+1
   DO 30 IEQNS=1,NEQNS
      NBACK=NEQN1-IEQNS
      NPOS=NFUNC(NBACK,NBACK)
```

```
      PIVOT=ASTIF(NPOS)
      RESID=ASLOD(NBACK)
      IF(NBACK .EQ. NEQNS)GO TO 20
      NBAC1=NBACK+1
      NLOCA=NBACK+NBAND
      IF(NLOCA .GT. NEQNS) NLOCA=NEQNS
      DO 10 ICOLS=NBAC1,NLOCA
         NPOS2=NFUNC(NBACK,ICOLS)
         RESID=RESID-ASTIF(NPOS2)*XDISP(ICOLS)
10    CONTINUE
20    IF(IFPRE(NBACK)  EQ. 0) THEN
         XDISP(NBACK)=RESID/PIVOT
      ELSE
         XDISP(NBACK)=FIXED(NBACK)
         REACT(NBACK)=-RESID
      END IF
30 CONTINUE
   RETURN
   END
C
C
   SUBROUTINE  DISPL(N,DOF,NID,NODRED,N1,N2,N3,N4,N5,NELA,NELB,INTER,
  +            XDISP,REACT,MEQNS,NCASE)
   IMPLICIT REAL*8 (A-H,O-Z)
   INTEGER N,DOF,NID(MEQNS),NODRED(MEQNS),N1(MEQNS),N2(MEQNS)
   INTEGER N3(MEQNS),N4(MEQNS),N5(MEQNS),NELA(MEQNS),NELB(MEQNS)
   INTEGER INTER,ELID,ELIDB,NOD1,NOD2,NOD3,NOD4,NOD5
   INTEGER NOD11,NOD22,NOD33,NOD44,NOD55,NN(6)
   DOUBLE PRECISION NU
   DIMENSION XDISP(2000),REACT(2000),Q(2,12),XR(12),XX(2)
   DIMENSION XRDISP(2000),RREACT(2000)
   COMMON/GLOB/X(1000),Y(1000),A(1000),E(1000),NU(1000),T(1000)
   IF(INTER .NE. 0) THEN
     DO 60 I=1,INTER
       ELID=NELA(I)
       ELIDB=NELB(I)
       NOD1=N1(ELID)
       NOD2=N2(ELID)
       NOD3=N3(ELID)
```

```
      NOD4=N4(ELID)
      NOD5=N5(ELID)
      NOD11=N1(ELIDB)
      NOD22=N2(ELIDB)
      NOD33=N3(ELIDB)
      NOD44=N4(ELIDB)
      NOD55=N5(ELIDB)
      IF(NCASE .EQ. 4) THEN
        CALL RELAX(ELID,ELIDB,NOD1,NOD2,NOD3,NOD4,NOD5,NOD11,NOD22
     +             ,NOD33,NOD44,NOD55,Q)
      ELSE
        CALL REL(ELID,ELIDB,NOD1,NOD2,NOD3,NOD4,NOD5,NOD11,NOD22
     +           ,NOD33,NOD44,NOD55,Q)
      END IF
      NN(1)=NOD4
      NN(2)=NOD22
      NN(3)=NOD33
      NN(4)=NOD44
      NN(5)=NOD2
      NN(6)=NOD3
      DO 40 L=1,6
         LK=2*L-1
         LL=2*L
         XR(LK)=XDISP(2*NODRED(NN(L))-1)
         XR(LL)=XDISP(2*NODRED(NN(L)))
40    CONTINUE
      CALL MATVEC(2,12,Q,XR,XX,2,12)
      XRDISP(2*N5(ELID)-1)=XX(1)
      XRDISP(2*N5(ELID))=XX(2)
60   CONTINUE
     DO 80 I=1,N
      IF(NODRED(NID(I)) .NE. 0) THEN
        XRDISP(2*NID(I)-1)=XDISP(2*NODRED(NID(I))-1)
        XRDISP(2*NID(I))=XDISP(2*NODRED(NID(I)))
        RREACT(2*NID(I)-1)=REACT(2*NODRED(NID(I))-1)
        RREACT(2*NID(I))=REACT(2*NODRED(NID(I)))
      ELSE
        RREACT(2*NID(I)-1)=0.
        RREACT(2*NID(I))=0.
```

```
          END IF
 80     CONTINUE
        DO 100 I=1,DOF
          XDISP(I)=XRDISP(I)
100     CONTINUE
      ELSE
        DO 200 I=1,DOF
          RREACT(I)=REACT(I)
200     CONTINUE
      END IF
      PRINT 150
150 FORMAT('1 REACTIONS AND DISPLACEMENTS AT NODES')
      PRINT 160
160 FORMAT('- NODE',T8,'X-REACTION',T22,'Y-REACTION',T50,'X-DISPL',
   +        T62,'Y-DISPL')
      PRINT 170,(K,RREACT(2*K-1),RREACT(2*K),XDISP(2*K-1),XDISP(2*K),
   +          K=1,N)
170 FORMAT('-',I3,T8,E10.3,T22,E10.3,T49,E10.3,T61,E10.3)
      RETURN
      END
C
C
      SUBROUTINE FORCE(ELID,N1,N2,X1,Y1,X2,Y2,A,E,MEQNS,XDISP)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XDISP(2000)
      INTEGER ELID
      REAL*8 DELTA(4),XDIF,YDIF,L,C,S,K,TR(4,4),KPRIM(4,4),UPRIM(4)
      REAL*8 FPRIM(4),KEL(4,4)
      CALL BAR(ELID,N1,N2,X1,Y1,X2,Y2,A,E,KEL,TR,KPRIM)
      DELTA(1)=XDISP(2*N1-1)
      DELTA(2)=XDISP(2*N1)
      DELTA(3)=XDISP(2*N2-1)
      DELTA(4)=XDISP(2*N2)
      PRINT 50,ELID,N1,N2
 50 FORMAT('-  ELEMENT NO. : ',I3,2X,'BAR',' -  ',I3,' TO ',I3)
      DX=X2-X1
      DY=Y2-Y1
      RL=DSQRT(DX*DX + DY*DY)
      COSB=DX/RL
```

```
      SINB=DY/RL
      EL=(DELTA(3)-DELTA(1))*COSB + (DELTA(4)-DELTA(2))*SINB
      FL=(EL/RL)*E*A
      PRINT 200,RL
  200 FORMAT('- MEMBER LENGTH ',T22,': ',E10.3)
      PRINT 220,EL
  220 FORMAT('- MEMBER ELONGATION ',T22,': ',E10.3)
      PRINT 240,FL
  240 FORMAT('- MEMBER FORCE ',T22,': ',E10 3)
      RETURN
      END
C
C
      SUBROUTINE STR(ELID,N1,N2,N3,X1,Y1,X2,Y2,X3,Y3,E,NU,T,XDISP,MEQNS)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XDISP(2000)
      INTEGER ELID
      REAL*8 KEL(6,6),U(6),D1(6),ENERGY,STRESS(3),STRAIN(3),B(3,6)
      REAL*8 EM(3,3),SMAX,SMIN,S1,S2,S3,S4,ANGLE,NU,XDISP
      REAL*8 E1,E2,E3,EMAX,EMIN,PI
      PI=DACOS(-1.0D0)
      CALL YOUNG(EM,E,NU,3,0)
      CALL CST(ELID,N1,N2,N3,X1,Y1,X2,Y2,X3,Y3,E,NU,T,B,KEL)
      U(1)=XDISP(2*N1-1)
      U(2)=XDISP(2*N1)
      U(3)=XDISP(2*N2-1)
      U(4)=XDISP(2*N2)
      U(5)=XDISP(2*N3-1)
      U(6)=XDISP(2*N3)
      CALL MATVEC(3,6,B,U,STRAIN,3,6)
      CALL MATVEC(3,3,EM,STRAIN,STRESS,3,3)
      S1=(STRESS(1)+STRESS(2))/2.
      E1=(STRAIN(1)+STRAIN(2))/2.
      S2=(STRESS(1)-STRESS(2))/2.
      E2=(STRAIN(1)-STRAIN(2))/2.
      S3=DSQRT(S2*S2 + STRESS(3)*STRESS(3))
      E3=DSQRT(E2*E2 + STRAIN(3)*STRAIN(3))
      SMAX=S1+S3
      EMAX=E1+E3
```

```
      SMIN=S1-S3
      EMIN=E1-E3
      IF(S2 .EQ. 0.0) THEN
        ANGLE=PI/2.
      ELSE
        ANGLE=(DATAN(STRESS(3)/S2))/2.
      END IF
      CALL MATVEC(6,6,KEL,U,D1,6,6)
      ENERGY=0.0
      DO 20 I=1,6
         ENERGY = ENERGY+D1(I)*U(I)
   20 CONTINUE
      ENERGY=ENERGY/2.
      PRINT 50,ELID
   50 FORMAT('- ELEMENT NO. :',I3,' C.S.T.')
      PRINT 60,STRESS(1),STRAIN(1)
   60 FORMAT('- S 1',T10,':',E10.3,T30,'E 1',T37,':',E10.3)
      PRINT 70,STRESS(2),STRAIN(2)
   70 FORMAT('- S 2',T10,':',E10.3,T30,'E 2',T37,':',E10.3)
      PRINT 80,STRESS(3),STRAIN(3)
   80 FORMAT('- T XY',T10,':',E10.3,T30,'E XY',T37,':',E10.3)
      PRINT 90,SMAX,EMAX
   90 FORMAT('- S MAX',T10,':',E10.3,T30,'E MAX',T37,':',E10.3)
      PRINT 100,SMIN,EMIN
  100 FORMAT('- S MIN',T10,':',E10.3,T30,'E MIN',T37,':',E10.3)
      PRINT 110,ANGLE,ENERGY
  110 FORMAT('- ANGLE',T10,':',E10.3,T30,'STRAIN ENERGY ',' ',E10.3)
      RETURN
      END
C
C
      SUBROUTINE STRES(ELID,N1,N2,N3,N4,N5,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,
     +                 Y5,E,NU,T,XDISP,MEQNS,NTYPE,INCOMP)
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER ELID
      DOUBLE PRECISION NU
      DIMENSION XDISP(2000),XI(6),ET(6),EM(3,3),NOD(6),XX(5),YY(5)
      DIMENSION XLOC(10),B(3,14),PROD(3,10),S(3),TRAN(3,3),SLOC(3)
      CHARACTER*5 TYPE
```

```
      DATA XI/-1 ,1.,1.,-1.,0.,0./
      ET(1)=-1.
      ET(2)=-1.
      ET(3)=1.
      ET(4)=1.
      ET(5)=-1.
      ET(6)=0.
      IF(NTYPE .EQ. 4) THEN
        ET(5)=0.
        TYPE='QUAD4'
      ELSE
        TYPE='QUAD5'
      END IF
      PRINT 20,ELID,TYPE
   20 FORMAT('- ELEMENT NO. :',I3,2X,A5)
      MSIZE=2*NTYPE
      NSIZE=NTYPE+1
      CALL YOUNG(EM,E,NU,3,0)
      NOD(1)=N1
      NOD(2)=N2
      NOD(3)=N3
      NOD(4)=N4
      NOD(5)=N5
      NOD(6)=0
      XX(1)=X1
      XX(2)=X2
      XX(3)=X3
      XX(4)=X4
      XX(5)=X5
      YY(1)=Y1
      YY(2)=Y2
      YY(3)=Y3
      YY(4)=Y4
      YY(5)=Y5
      DO 40 I=1,NTYPE
         XLOC(2*I-1)=XDISP(2*NOD(I)-1)
         XLOC(2*I)=XDISP(2*NOD(I))
   40 CONTINUE
      PRINT 50
```

```
 50 FORMAT('-',T2,'NODE',T12,'S 11',T24,'S 22',T36,'T XY',T48,'S MAX',
     +        T60,'S MIN',T72,'ANGLE')
      DO 80 I=1,NSIZE
        CALL SHAPEF(XI(I),ET(I),XX,YY,DETJAC,B,NTYPE,INCOMP)
        CALL MATMAT(3,3,MSIZE,EM,B,PROD)
        CALL MATVEC(3,MSIZE,PROD,XLOC,S,3,MSIZE)
        IF(NTYPE .EQ. 5) THEN
          IF(I .EQ. 5) THEN
            CALL TRANSF(ELID,X1,X2,Y1,Y2,TRAN,3,3)
            CALL MATVEC(3,3,TRAN,S,SLOC,3,3)
            S(1)=SLOC(1)
            S(2)=SLOC(2)
            S(3)=SLOC(3)
          END IF
        END IF
        CALL PRINC(NOD(I),S)
 80   CONTINUE
      PRINT 100
100   FORMAT('-')
      RETURN
      END
C
C
      SUBROUTINE PRINC(NODE,S)
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER NODE
      DIMENSION S(3)
      PI=DACOS(-1.D0)
      S1=(S(1)+S(2))/2.
      S2=(S(1)-S(2))/2.
      S3=DSQRT(S2*S2 + S(3)*S(3))
      SMAX=S1+S3
      SMIN=S1-S3
      IF(S2 .EQ. 0.D0) THEN
        ANGLE=PI/2.
      ELSE
        ANGLE=(DATAN(S(3)/S2))/2.
      END IF
      ANGLE=(ANGLE*180.D0)/PI
```

```
      PRINT 40,NODE,S(1),S(2),S(3),SMAX,SMIN,ANGLE
   40 FORMAT('-',I3,T9,E10.3,T21,E10.3,T33,E10.3,T45,E10.3,T57,E10.3,
     +        T69,E10.3)
      RETURN
      END
C
C
      SUBROUTINE TRANSF(ELID,X1,X2,Y1,Y2,TRAN,NROW,NCOL)
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER ELID
      DIMENSION TRAN(NROW,NCOL)
      XA=X1-X2
      YA=Y2-Y1
      RL = DSQRT(XA*XA + YA*YA)
      C = YA/RL
      S = XA/RL
      IF(NROW .EQ. 2) THEN
        IF(NCOL .EQ. 3) THEN
          TRAN(1,1) = C*C
          TRAN(1,2) = S*S
          TRAN(1,3) = 2.DO*C*S
          TRAN(2,1) = -C*S
          TRAN(2,2) = C*S
          TRAN(2,3) = C*C - S*S
        ELSE
          TRAN(1,1) = C*C
          TRAN(1,2) = 0.DO
          TRAN(1,3) = S*S
          TRAN(1,4) = 2.DO*C*S
          TRAN(2,1) = -C*S
          TRAN(2,2) = 0.DO
          TRAN(2,3) = C*S
          TRAN(2,4) = C*C - S*S
        END IF
      END IF
      IF(NROW .EQ. 3) THEN
        TRAN(1,1) = C*C
        TRAN(1,2) = S*S
        TRAN(1,3) = 2.DO*C*S
```

```
        TRAN(2,1) = TRAN(1,2)
        TRAN(2,2) = TRAN(1,1)
        TRAN(2,3) = -TRAN(1,3)
        TRAN(3,1) = -C*S
        TRAN(3,2) = -TRAN(3,1)
        TRAN(3,3) = C*C - S*S
      END IF
      IF(NROW .EQ. 4) THEN
        TRAN(1,1) = C*C
        TRAN(1,2) = 0.D0
        TRAN(1,3) = S*S
        TRAN(1,4) = 2.D0*C*S
        TRAN(2,1) = 0.D0
        TRAN(2,2) = 1.D0
        TRAN(2,3) = 0.D0
        TRAN(2,4) = 0.D0
        TRAN(3,1) = TRAN(1,3)
        TRAN(3,2) = 0.D0
        TRAN(3,3) = TRAN(1,1)
        TRAN(3,4) = -TRAN(1,4)
        TRAN(4,1) = -C*S
        TRAN(4,2) = 0.D0
        TRAN(4,3) = -TRAN(4,1)
        TRAN(4,4) = C*C -S*S
      END IF
      RETURN
      END
C
C
      SUBROUTINE TRIAX(ELID,N1,N2,N3,R1,Z1,R2,Z2,R3,Z3,E,NU,KEL)
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER ELID
      REAL*8 NU,KEL(6,6),EM(4,4),H(6,6),F(6,6),HT(6,6),PROD(6,6)
      CALL YOUNG(EM,E,NU,4,0)
      RL = R2*(Z3-Z1) + R1*(Z2-Z3) + R3*(Z1-Z2)
      DO 20 I=1,6
         DO 20 J=1,6
            H(I,J)=0.D0
            KEL(I,J)=0.D0
```

```
      F(I,J)=0.DO
20 CONTINUE
   H(1,1) = (R2*Z3-R3*Z2)/RL
   H(1,3) = (R3*Z1-R1*Z3)/RL
   H(1,5) = (R1*Z2-R2*Z1)/RL
   H(2,2) = H(1,1)
   H(2,4) = H(1,3)
   H(2,6) = H(1,5)
   H(3,1) = (Z2-Z3)/RL
   H(3,3) = (Z3-Z1)/RL
   H(3,5) = (Z1-Z2)/RL
   H(4,2) = H(3,1)
   H(4,4) = H(3,3)
   H(4,6) = H(3,5)
   H(5,1) = (R3-R2)/RL
   H(5,3) = (R1-R3)/RL
   H(5,5) = (R2-R1)/RL
   H(6,2) = H(5,1)
   H(6,4) = H(5,3)
   H(6,6) = H(5,5)
   DO 40 I=1,6
      DO 40 J=1,6
         HT(I,J)=H(J,I)
40 CONTINUE
   A = ((R2-R1)*(Z3-Z1) - (R3-R1)*(Z2-Z1))/2.DO
   RC=(R1+R2+R3)/3.DO
   ZC=(Z1+Z2+Z3)/3.DO
   F(1,1) = EM(2,2)*(1.DO/RC)*A
   F(1,3) =(EM(2,1) + EM(2,2))*A
   F(1,5) = EM(2,2)*(ZC/RC)*A
   F(1,6) = EM(2,3)*A
   F(3,3) =(EM(1,1)+EM(1,2)+EM(2,1)+EM(2,2))*(RC)*A
   F(3,5) =(EM(1,2)+EM(2,2))*ZC*A
   F(3,6) =(EM(1,3)+EM(2,3))*RC*A
   F(4,4) = EM(4,4)*RC*A
   F(4,5) = F(4,4)
   F(5,5) = EM(2,2)*ZC*ZC/RC +
  +         EM(4,4)*RC*A
   F(5,6)= EM(2,3)*ZC*A
```

```
      F(6,6) = EM(3,3)*RC*A
      DO 60 I=1,5
        DO 60 J=I,6
          F(J,I)=F(I,J)
   60 CONTINUE
      CALL MATMAT(6,6,6,F,H,PROD)
      CALL MATMAT(6,6,6,HT,PROD,KEL)
      RETURN
      END
C
C
      SUBROUTINE QUADAX(NGAUSS,NELEM,NORDER,N1,N2,N3,N4,N5,STIFEL,NDIM)
      IMPLICIT REAL*8 (A-H,O-Z)
      DOUBLE PRECISION JACOB,NU,KEL(10,10)
      DIMENSION XX(5),YY(5),PLACE(3,3),WGT(3,3),B(4,10),BTE(10,4)
      DIMENSION EM(4,4),STIFEL(NDIM,NDIM)
      COMMON/Q4/EN(5),JACOB(2,2)
      COMMON/GLOB/X(1000),Y(1000),A(1000),E(1000),NU(1000),T(1000)
      DATA PLACE(1,1),PLACE(2,1),PLACE(2,3),PLACE(3,1),PLACE(3,2)/
     +     5*0.000000000000000D0/
      DATA PLACE(1,2)/-0.577350269189626D0/
      DATA PLACE(2,2)/ 0.577350269189626D0/
      DATA PLACE(1,3)/-0 774596669241483D0/
      DATA PLACE(3,3)/ 0.774596669241483D0/
      DATA WGT(1,1)/2.000000000000000D0/,WGT(2,3)/0.888888888888889D0/
      DATA WGT(1,2),WGT(2,2)/2*1.000000000000000D0/
      DATA WGT(2,1),WGT(3,1),WGT(3,2)/3*0.000000000000000D0/
      DATA WGT(1,3),WGT(3,3)/2*0.555555555555556D0/
      PI=DACOS(-1.D0)
      CALL YOUNG(EM,E(NELEM),NU(NELEM),4,0)
      MORDER=2*NORDER
      NSIZE = NDIM
      XX(1)=X(N1)
      XX(2)=X(N2)
      XX(3)=X(N3)
      XX(4)=X(N4)
      YY(1)=Y(N1)
      YY(2)=Y(N2)
      YY(3)=Y(N3)
```

```
      YY(4)=Y(N4)
      IF(NORDER .NE. 4) THEN
        XX(5)=X(N5)
        YY(5)=Y(N5)
      END IF
      DO 40 K=1,NSIZE
         DO 40 L=K,NSIZE
             KEL(K,L)=0.D0
 40   CONTINUE
      DO 180 NA=1,NGAUSS
         XI=PLACE(NA,NGAUSS)
         DO 160 NB=1,NGAUSS
             ET=PLACE(NB,NGAUSS)
             CALL SHAPAX(XI,ET,XX,YY,DETJAC,B,NORDER)
             R=0.D0
             DO 50 NR=1,NORDER
                R=R+EN(NR)*XX(NR)
 50          CONTINUE
             DV=WGT(NA,NGAUSS)*WGT(NB,NGAUSS)*DETJAC
             KORDER = NSIZE/2
             DO 80 J=1,KORDER
                L=2*J
                K=L-1
                DO 60 N=1,3
                   BTE(K,N)=B(1,K)*EM(1,N) + B(2,K)*EM(2,N)
                   BTE(L,N)=B(3,L)*EM(3,N)
 60             CONTINUE
                BTE(L,4)=B(4,L)*EM(4,4)
                BTE(K,4)=B(4,K)*EM(4,4)
 80          CONTINUE
             DO 140 NROW=1,NSIZE
                DO 120 NCOL=NROW,NSIZE
                   DUM=0.D0
                   DO 100 J=1,4
                      DUM=DUM + BTE(NROW,J)*B(J,NCOL)
100                CONTINUE
                   KEL(NROW,NCOL)=KEL(NROW,NCOL) + DUM*DV*R
120             CONTINUE
140          CONTINUE
```

```
160    CONTINUE
180 CONTINUE
    DO 200 K=1,NSIZE
       DO 200 L=K,NSIZE
          KEL(L,K)=KEL(K,L)
200 CONTINUE
    DO 400 I = 1,MORDER
       DO 400 J = 1,MORDER
          STIFEL(I,J)=KEL(I,J)
400 CONTINUE
    RETURN
    END
C
C
    SUBROUTINE SHAPAX(XI,ET,XX,YY,DETJAC,B,NORDER)
    IMPLICIT REAL*8 (A-H,O-Z)
    DOUBLE PRECISION JACOB,NU
    DIMENSION RXI(5),RET(5),RK(5),RL(5)
    DIMENSION XX(5),YY(5),B(4,10),EM(4,4),ENXI(5),ENET(5)
    COMMON/Q4/EN(5),JACOB(2,2)
    COMMON/GLOB/X(1000),Y(1000),A(1000),E(1000),NU(1000),T(1000)
    DATA RXI/-1.,1.,1.,-1.,1./,RET/-1.,-1.,1.,1.,1./
    DATA RK/1.,1.,1.,1.,0./
    RL(1)=1.
    RL(2)=1.
    RL(3)=0.
    RL(4)=0.
    RL(5)=-2.
    IF(NORDER .EQ. 4) THEN
     RL(1)=0.
     RL(2)=0.
    END IF
    DO 20 L=1,NORDER
       F1=(1. + RXI(L)*XI)
       F2=(1. + RET(L)*ET)
       F3=(1. - XI*XI)
       F4=(1. - ET)
       EN(L)=RK(L)*F1*F2/4. - RL(L)*F3*F4/4.
       ENXI(L)=RK(L)*RXI(L)*F2/4. + RL(L)*XI*F4/2.
```

```
      ENET(L)=RK(L)*RET(L)*F1/4. + RL(L)*F3/4.
   20 CONTINUE
      MORDER=NORDER
      NSIZE = 2*MORDER
      DO 40 I=1,4
         DO 40 J=1,NSIZE
            B(I,J)=0.DO
   40 CONTINUE
      JACOB(1,1)=0.DO
      JACOB(1,2)=0.DO
      JACOB(2,1)=0.DO
      JACOB(2,2)=0.DO
      DO 60 L=1,NORDER
         JACOB(1,1)=JACOB(1,1)+ENXI(L)*XX(L)
         JACOB(1,2)=JACOB(1,2)+ENXI(L)*YY(L)
         JACOB(2,1)=JACOB(2,1)+ENET(L)*XX(L)
         JACOB(2,2)=JACOB(2,2)+ENET(L)*YY(L)
   60 CONTINUE
      DETJAC=JACOB(1,1)*JACOB(2,2)-JACOB(1,2)*JACOB(2,1)
      F5=JACOB(1,1)/DETJAC
      JACOB(1,1)=JACOB(2,2)/DETJAC
      JACOB(1,2)=-JACOB(1,2)/DETJAC
      JACOB(2,1)=-JACOB(2,1)/DETJAC
      JACOB(2,2)=F5
      DO 80 J=1,MORDER
         L=2*J
         K=L-1
         B(1,K)=JACOB(1,1)*ENXI(J) + JACOB(1,2)*ENET(J)
         B(3,L)=JACOB(2,1)*ENXI(J) + JACOB(2,2)*ENET(J)
         B(4,K)=B(3,L)
         B(4,L)=B(1,K)
   80 CONTINUE
      R=0.DO
      DO 100 I=1,MORDER
         IF(XX(I) .EQ. 0.DO) THEN
           XX(I) = 1.D-6
         END IF
         R=R+EN(I)*XX(I)
  100 CONTINUE
```

```
      DO 120 I=1,MORDER
         L=2*I - 1
         B(2,L)=EN(I)/R
  120 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE RELAX(NELEM1,NELEM2,N1,N2,N3,N4,N5,N11,N22,N33,N44,N55,
     +                 Q)
      IMPLICIT REAL*8 (A-H,O-Z)
      DOUBLE PRECISION NUA,NUB,NU
      DIMENSION XA(5),YA(5),XB(5),YB(5),TRAN1(2,4),BA(4,10),BB(4,10)
      DIMENSION EA(4,4),EB(4,4),PROD1(4,10),TRAN2(2,4),QA(2,10),QB(2,10)
      DIMENSION Q1(2,2),Q2(2,12),Q(2,12),XX1(5),XX2(5),YY1(5),YY2(5)
      COMMON/GLOB/X(1000),Y(1000),A(1000),E(1000),NU(1000),T(1000)
      PI=DACOS(-1.D0)
      DETJAC=0.D0
      XX1(1)=X(N1)
      XX1(2)=X(N2)
      XX1(3)=X(N3)
      XX1(4)=X(N4)
      XX1(5)=X(N5)
      YY1(1)=Y(N1)
      YY1(2)=Y(N2)
      YY1(3)=Y(N3)
      YY1(4)=Y(N4)
      YY1(5)=Y(N5)
      XX2(1)=X(N11)
      XX2(2)=X(N22)
      XX2(3)=X(N33)
      XX2(4)=X(N44)
      XX2(5)=X(N55)
      YY2(1)=Y(N11)
      YY2(2)=Y(N22)
      YY2(3)=Y(N33)
      YY2(4)=Y(N44)
      YY2(5)=Y(N55)
      CALL TRANSF(NELEM1,XX1(1),XX1(2),YY1(1),YY1(2),TRAN1,2,4)
```

```
      CALL TRANSF(NELEM2,XX2(1),XX2(2),YY2(1),YY2(2),TRAN2,2,4)
      XI=0.D0
      ET=-1.D0
      CALL SHAPAX(XI,ET,XX1,YY1,DETJAC,BA,5)
      CALL YOUNG(EA,E(NELEM1),NU(NELEM1),4,0)
      CALL SHAPAX(XI,ET,XX2,YY2,DETJAC,BB,5)
      CALL YOUNG(EB,E(NELEM2),NU(NELEM2),4,0)
      CALL MATMAT(4,4,10,EA,BA,PROD1)
      CALL MATMAT(2,4,10,TRAN1,PROD1,QA)
      CALL MATMAT(4,4,10,EB,BB,PROD1)
      CALL MATMAT(2,4,10,TRAN2,PROD1,QB)
      F1=(QB(1,9)-QA(1,9))*(QB(2,10)-QA(2,10))
      F2=(QB(1,10)-QA(1,10))*(QB(2,9)-QA(2,9))
      DETQ=(F1-F2)
      Q1(1,1)=(QB(2,10)-QA(2,10))/DETQ
      Q1(1,2)=-(QB(1,10)-QA(1,10))/DETQ
      Q1(2,1)=-(QB(2,9)-QA(2,9))/DETQ
      Q1(2,2)=(QB(1,9)-QA(1,9))/DETQ
      DO 80 L=1,2
         Q2(L,1)=QA(L,7)
         Q2(L,2)=QA(L,8)
         Q2(L,3)=QA(L,1)-QB(L,3)
         Q2(L,4)=QA(L,2)-QB(L,4)
         Q2(L,5)=-QB(L,5)
         Q2(L,6)=-QB(L,6)
         Q2(L,7)=-QB(L,7)
         Q2(L,8)=-QB(L,8)
         Q2(L,9)=QA(L,3)-QB(L,1)
         Q2(L,10)=QA(L,4)-QB(L,2)
         Q2(L,11)=QA(L,5)
         Q2(L,12)=QA(L,6)
   80 CONTINUE
      CALL MATMAT(2,2,12,Q1,Q2,Q)
      RETURN
      END
C
C
      SUBROUTINE STRIAX(ELID,N1,N2,N3,R1,Z1,R2,Z2,R3,Z3,E,NU,XDISP,
     +                  MEQNS)
```

```
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER ELID
      REAL*8 NU
      DIMENSION EM(4,4),H(6,6),G(4,6),S(4)
      DIMENSION XDISP(2000),XLOC(6),PROD1(4,6),PROD2(4,6)
      DO 20 I=1,6
         DO 20 J=1,6
            H(I,J)=0.DO
20    CONTINUE
      DO 40 I=1,4
         DO 40 J=1,6
            G(I,J)=0.DO
40    CONTINUE
      RL=R2*(Z3-Z1) + R1*(Z2-Z3) + R3*(Z1-Z2)
      CALL YOUNG(EM,E,NU,4,0)
      H(1,1)=(R2*Z3-R3*Z2)/RL
      H(1,3)=(R3*Z1-R1*Z3)/RL
      H(1,5)=(R1*Z2-R2*Z1)/RL
      H(2,2)=H(1,1)
      H(2,4)=H(1,3)
      H(2,6)=H(1,5)
      H(3,1)=(Z2-Z3)/RL
      H(3,3)=(Z3-Z1)/RL
      H(3,5)=(Z1-Z2)/RL
      H(4,2)=H(3,1)
      H(4,4)=H(3,3)
      H(4,6)=H(3,5)
      H(5,1)=(R3-R2)/RL
      H(5,3)=(R1-R3)/RL
      H(5,5)=(R2-R1)/RL
      H(6,2)=H(5,1)
      H(6,4)=H(5,3)
      H(6,6)=H(5,5)
      R=(R1+R2+R3)/3.DO
      Z=(Z1+Z2+Z3)/3.DO
      G(1,3)=1.DO
      G(2,1)=1.DO/R
      G(2,3)=1.DO
      G(2,5)=Z/R
```

```
      G(3,6)=1.DO
      G(4,4)=1.DO
      G(4,5)=1.DO
      XLOC(1)=XDISP(2*N1-1)
      XLOC(2)=XDISP(2*N1)
      XLOC(3)=XDISP(2*N2-1)
      XLOC(4)=XDISP(2*N2)
      XLOC(5)=XDISP(2*N3-1)
      XLOC(6)=XDISP(2*N3)
      CALL MATMAT(4,4,6,EM,G,PROD1)
      CALL MATMAT(4,6,6,PROD1,H,PROD2)
      CALL MATVEC(4,6,PROD2,XLOC,S,4,6)
      PI=DACOS(-1.DO)
      S1=(S(1) + S(3))/2.DO
      S2=(S(1) - S(3))/2.DO
      S3=DSQRT(S2*S2 + S(4)*S(4))
      SMAX=S1 + S3
      SMIN=S1 - S3
      IF(S2 .EQ. 0.0) THEN
        ANGLE=PI/2.DO
      ELSE
        ANGLE=(DATAN(S(4)/S2))/2 DO
      END IF
      PRINT 100,ELID
  100 FORMAT('- ELEMENT NO. : ',I3,' TRIAX')
      PRINT 120,S(1),SMAX
  120 FORMAT('- S R',T10,': ',E10.3,T30,'S MAX',T37,': ',E10 3)
      PRINT 140,S(2),SMIN
  140 FORMAT('- S TH',T10,': ',E10.3,T30,'S MIN',T37,': ',E10 3)
      PRINT 160,S(3),ANGLE
  160 FORMAT('- S Z',T10,': ',E10.3,T30,'ANGLE',T37,': ',E10.3)
      PRINT 180,S(4)
  180 FORMAT('- T RZ',T10,': ',E10 3)
      RETURN
      END
C
C
      SUBROUTINE STRAX(ELID,N1,N2,N3,N4,N5,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,
     +                 Y5,E,NU,XDISP,MEQNS,NTYPE)
```

```
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER ELID
      DOUBLE PRECISION NU
      DIMENSION XDISP(2000),XI(6),ET(6),EM(4,4),NOD(6),XX(5),YY(5)
      DIMENSION XLOC(10),B(4,10),PROD(4,10),S(4),TRAN(4,4),SLOC(4)
      CHARACTER*5 TYPE
      DATA XI/-1.,1.,1.,-1.,0.,0./
      ET(1)=-1.
      ET(2)=-1.
      ET(3)=1.
      ET(4)=1.
      ET(5)=-1.
      ET(6)=0.
      IF(NTYPE .EQ. 4) THEN
        ET(5)=0.
        TYPE='QUAD4'
      ELSE
        TYPE='QUAD5'
      END IF
      PRINT 20,ELID,TYPE
   20 FORMAT('- ELEMENT NO. :',I3,2X,A5)
      MSIZE=2*NTYPE
      NSIZE=NTYPE+1
      CALL YOUNG(EM,E,NU,4,0)
      NOD(1)=N1
      NOD(2)=N2
      NOD(3)=N3
      NOD(4)=N4
      NOD(5)=N5
      NOD(6)=0.
      XX(1)=X1
      XX(2)=X2
      XX(3)=X3
      XX(4)=X4
      XX(5)=X5
      YY(1)=Y1
      YY(2)=Y2
      YY(3)=Y3
      YY(4)=Y4
```

```
      YY(5)=Y5
      DO 40 I=1,NTYPE
         XLOC(2*I-1)=XDISP(2*NOD(I)-1)
         XLOC(2*I)=XDISP(2*NOD(I))
   40 CONTINUE
      PRINT 50
   50 FORMAT('-',T2,'NODE',T12,'S  R',T24,'S TH',T36,'S  Z',T48,'T ZR',
     +        T60,'S MAX',T72,'S MIN',T84,'ANGLE')
      DO 80 I=1,NSIZE
         CALL SHAPAX(XI(I),ET(I),XX,YY,DETJAC,B,NTYPE)
         CALL MATMAT(4,4,MSIZE,EM,B,PROD)
         CALL MATVEC(4,MSIZE,PROD,XLOC,S,4,MSIZE)
         IF(NTYPE .EQ.5) THEN
           IF(I .EQ. 5) THEN
             CALL TRANSF(ELID,X1,X2,Y1,Y2,TRAN,4,4)
             CALL MATVEC(4,4,TRAN,S,SLOC,4,4)
             S(1)=SLOC(1)
             S(2)=SLOC(2)
             S(3)=SLOC(3)
             S(4)=SLOC(4)
           END IF
         END IF
         CALL PRINAX(NOD(I),S)
   80 CONTINUE
      PRINT 100
  100 FORMAT('-')
      RETURN
      END
C
C
      SUBROUTINE PRINAX(NODE,S)
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER NODE
      DIMENSION S(4)
      PI=DACOS(-1.DO)
      S1=(S(1)+S(3))/2.
      S2=(S(1)-S(3))/2.
      S3=DSQRT(S2*S2 + S(4)*S(4))
      SMAX=S1+S3
```

```
      SMIN=S1-S3
      IF(S2 .EQ. 0.D0) THEN
        ANGLE=PI/2.
      ELSE
        ANGLE=(DATAN(S(4)/S2))/2.
      END IF
      ANGLE=(ANGLE*180.D0)/PI
      PRINT 40,NODE,S(1),S(2),S(3),S(4),SMAX,SMIN,ANGLE
   40 FORMAT('-',I3,T9,E10.3,T21,E10.3,T33,E10.3,T45,E10.3,T57,E10.3,
     +       T69,E10.3,T81,E10.3)
      RETURN
      END
C
C
      SUBROUTINE GENER(N,NDOF,BCX,BCY,NEL,NID,X,Y,XDEF,YDEF,ELID,
     +                  NTYPE,N1,N2,N3,N4,N5,A,E,NU,T,KEQNS,MEQNS)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION BCX(KEQNS),BCY(KEQNS),NID(KEQNS),X(KEQNS),Y(KEQNS)
      DIMENSION XDEF(KEQNS),YDEF(KEQNS),ELID(MEQNS),N1(MEQNS)
      DIMENSION N2(MEQNS),N3(MEQNS),E(MEQNS),NU(MEQNS),T(MEQNS)
      DIMENSION N4(MEQNS),N5(MEQNS),NTYPE(MEQNS),A(MEQNS)
      INTEGER BCX,BCY,ELID,BCX1,BCY1
      REAL*8 NU,NU1
      LL=1
      DO 40 WHILE(LL .LE. N)
           READ *,NID1,X1,Y1,BCX1,BCY1,XDEF1,YDEF1,KGEN
           IF(KGEN .NE. 0) THEN
             READ *,NID2,X2,Y2,KK
             KFACT=NID2-NID1
             KFACT1=KFACT/KK + 1
             DO 20 MM=1,KFACT1
                NIDGEN = NID1 + (MM-1)*KK
                NID(NIDGEN) = NIDGEN
                X(NIDGEN) = X1 + (X2-X1)*(MM-1)/(KFACT/KK)
                Y(NIDGEN) = Y1 + (Y2-Y1)*(MM-1)/(KFACT/KK)
                BCX(NIDGEN) = BCX1
                BCY(NIDGEN) = BCY1
                XDEF(NIDGEN) = XDEF1
                YDEF(NIDGEN) = YDEF1
```

```
20          CONTINUE
            LM = KFACT1
          ELSE
            NID(NID1) = NID1
            X(NID1) = X1
            Y(NID1) = Y1
            BCX(NID1) = BCX1
            BCY(NID1) = BCY1
            XDEF(NID1) = XDEF1
            YDEF(NID1) = YDEF1
            LM = 1
          END IF
          LL = LL + LM
40    CONTINUE
      LL = 1
      DO 80 WHILE(LL .LE. NEL)
          READ *,NELID1,NNTYPE,NN1,NN2,NN3,NN4,NN5,A1,E1,NU1,T1,
     +            KGEN
          IF(KGEN .NE. 0) THEN
            READ *,NELID2,NNTYPE,NNN1,NNN2,NNN3,NNN4,NNN5,KK
            KELFAC = NELID2-NELID1
            KEL1 = KELFAC/KK
            KELF1 = KEL1 + 1
            DO 60 MM = 1,KELF1
               NELID = NELID1 + (MM-1)*KK
               ELID(NELID) = NELID
               NTYPE(NELID) = NNTYPE
               N1(NELID) = NN1 + (MM-1)*(NNN1-NN1)/KEL1
               N2(NELID) = NN2 + (MM-1)*(NNN2-NN2)/KEL1
               N3(NELID) = NN3 + (MM-1)*(NNN3-NN3)/KEL1
               N4(NELID) = NN4 + (MM-1)*(NNN4-NN4)/KEL1
               N5(NELID) = NN5 + (MM-1)*(NNN5-NN5)/KEL1
               A(NELID) = A1
               E(NELID) = E1
               NU(NELID) = NU1
               T(NELID) = T1
60          CONTINUE
            LM = KELF1
          ELSE
```

```
            ELID(NELID1) = NELID1
            NTYPE(NELID1) = NNTYPE
            N1(NELID1) = NN1
            N2(NELID1) = NN2
            N3(NELID1) = NN3
            N4(NELID1) = NN4
            N5(NELID1) = NN5
            A(NELID1) = A1
            E(NELID1) = E1
            NU(NELID1) = NU1
            T(NELID1) = T1
            LM = 1
         END IF
         LL = LL + LM
80    CONTINUE
      RETURN
      END
```

B.2 Three-dimensional program

```
C      THREE-DIMENSIONAL FINITE ELEMENT PROGRAM
C      FOR THE ANALYSIS OF ELASTIC SOLIDS
C
C      BY  MICHAEL ANGELIDES
C
C      MCGILL UNIVERSITY
C      DEPARTMENT OF  CIVIL ENGINEERING
C      AND APPLIED MECHANICS
C
C      APRIL 1986
C
C
C  ELEMENT LIBRARY :
C
C  - BAR ELEMENT (BAR)
C  - CONSTANT STRAIN TETRAHEDRAL (TETRA)
C  - ISOPARAMETRIC LINEAR PENTAHEDRAL (PENTA)
C  - ISOPARAMETRIC LINEAR HEXAHEDRAL (HEXA)
C  - ISOPARAMETRIC LINEAR HEXAHEDRAL WITH ENFORCED INTERELEMENT
C    NORMAL AND SHEAR STRESS CONTINUITY AT ONE FACE (INTER)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NU,KGLOB(300,300),LOAD(300),KEL(36,36)
      INTEGER ELID(200),BCX(200),BCY(200),BCZ(200),DOF,ELIDB
      CHARACTER*80 TITLE
      DIMENSION NID(200),NTYPE(200)
      DIMENSION NODFOR(100),FX(100),FY(100)
      DIMENSION FZ(100),ASTIF(300,300),ASLOD(300),IFPRE(300)
      DIMENSION XDISP(300),REACT(300),PLOAD(300)
C
      COMMON/GLOB1/X(200),Y(200),Z(200),E(200),NU(200)
      COMMON/GLOB2/N1(200),N2(200),N3(200),N4(200),N5(200),N6(200),
     +             N7(200),N8(200)
      COMMON/ELCON/NELA(100),NELB(100),INTER
C
      READ *,TITLE
```

```
      READ *,N,NEL,NLOAD,INTER,NMAT,NGAUS1,NGAUS2
      LEQNS=300
      MEQNS=200
      KEQNS=200
      DOF=3*N
      CALL GENER(N,BCX,BCY,BCZ,NEL,NID,X,Y,Z,ELID,NTYPE,NMAT,
     +           N1,N2,N3,N4,N5,N6,N7,N8,E,NU,KEQNS,MEQNS)
      IF(INTER .NE. 0) THEN
        READ *,(NELA(I),NELB(I),I=1,INTER)
      END IF
      IF(NLOAD .NE. 0) THEN
        READ *,(NODFOR(I),FX(I),FY(I),FZ(I),I=1,NLOAD)
      END IF
      CALL DATA(N,DOF,BCX,BCY,BCZ,NEL,NID,X,Y,Z,ELID,NTYPE,
     +          N1,N2,N3,N4,N5,N6,N7,N8,E,NU,NODFOR,FX,FY,FZ,
     +          KEQNS,MEQNS,NLOAD,INTER,NGAUS1,NGAUS2,TITLE)
      DO 1000 I=1,DOF
         DO 1000 J=1,DOF
            KGLOB(I,J)=0.DO
 1000 CONTINUE
      DO 1100 I=1,NEL
         IF(NTYPE(I) .EQ. 4) THEN
           MM=12
         ELSE
           IF(NTYPE(I) .EQ. 5) THEN
             MM=18
           ELSE
             IF(NTYPE(I) .EQ. 6) THEN
               MM=24
             ELSE
               MM=36
             END IF
           END IF
         END IF
         NCOND=1
         IF(NTYPE(I) .EQ. 9) THEN
           NCOND=0
           DO 1050 J=1,INTER
              IF(ELID(I) .EQ. NELA(J)) THEN
```

```
                  ELIDB=NELB(J)
                  CALL STIFF(ELID(I),ELIDB,NTYPE(I),KGLOB,LEQNS,MM,
     +                      NGAUS2,KEL)
                END IF
 1050         CONTINUE
            END IF
            IF(NCOND .EQ. 1) THEN
              CALL STIFF(ELID(I),ELIDB,NTYPE(I),KGLOB,LEQNS,MM,
     +                  NGAUS1,KEL)
            END IF
 1100 CONTINUE
      DO 1200 I=1,DOF
         DO 1200 J=1,DOF
            ASTIF(I,J)=KGLOB(I,J)
 1200 CONTINUE
      DO 1300 I=1,DOF
         ASLOD(I)=0.DO
 1300 CONTINUE
      IF(NLOAD .NE. 0) THEN
        DO 1400 I=1,NLOAD
           KK1=3*NID(NODFOR(I))-2
           KK2=KK1+1
           KK3=KK1+2
           ASLOD(KK1)=FX(I)
           ASLOD(KK2)=FY(I)
           ASLOD(KK3)=FZ(I)
 1400   CONTINUE
      END IF
      DO 1500 I=1,DOF
         PLOAD(I)=ASLOD(I)
 1500 CONTINUE
      DO 1600 I=1,N
         K=NID(I)
         IFPRE(3*K-2)=BCX(K)
         IFPRE(3*K-1)=BCY(K)
         IFPRE(3*K)=BCZ(K)
 1600 CONTINUE
      CALL GREDUC(LEQNS,ASLOD,ASTIF,IFPRE,DOF)
      CALL BAKSUB(N,LEQNS,ASLOD,ASTIF,IFPRE,XDISP,REACT,DOF)
```

```
      PRINT 2000
 2000 FORMAT('1 ELEMENT STRESSES')
      DO 2500 I=1,NEL
         IF(NTYPE(I)  EQ. 4) THEN
           CALL STETRA(ELID(I),XDISP,LEQNS)
         ELSE
           IF(NTYPE(I) .GE. 5) THEN
             CALL STRES(ELID(I),XDISP,LEQNS,NTYPE(I))
           END IF
         END IF
 2500 CONTINUE
      PRINT 9000
 9000 FORMAT('1')
      STOP
      END
C
C
      SUBROUTINE GENER(N,BCX,BCY,BCZ,NEL,NID,X,Y,Z,ELID,NTYPE,NMAT,
     +                 N1,N2,N3,N4,N5,N6,N7,N8,E,NU,KEQNS,MEQNS)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NU,NUM
      INTEGER BCX,BCY,BCZ,ELID,BCX1,BCY1,BCZ1
      DIMENSION BCX(KEQNS),BCY(KEQNS),BCZ(KEQNS),NID(KEQNS),X(KEQNS)
      DIMENSION Y(KEQNS),Z(KEQNS),ELID(MEQNS),N1(MEQNS),N2(MEQNS)
      DIMENSION N3(MEQNS),N4(MEQNS),N5(MEQNS),N6(MEQNS),N7(MEQNS)
      DIMENSION N8(MEQNS),E(MEQNS),NU(MEQNS),NTYPE(MEQNS),MAT(200)
      DIMENSION NIDMAT(10),EM(10),NUM(10)
      LL=1
      DO 30 WHILE(LL .LE. N)
           READ *,NID1,X1,Y1,Z1,BCX1,BCY1,BCZ1,KGEN
           IF(KGEN .NE. 0) THEN
             READ *,NID2,X2,Y2,Z2,KK
             KFACT=NID2-NID1
             KFACT1=KFACT/KK + 1
             DO 20 MM=1,KFACT1
                NIDGEN=NID1 + (MM-1)*KK
                NID(NIDGEN)=NIDGEN
                X(NIDGEN)=X1 + (X2-X1)*(MM-1)/(KFACT/KK)
                Y(NIDGEN)=Y1 + (Y2-Y1)*(MM-1)/(KFACT/KK)
```

```
               Z(NIDGEN)=Z1 + (Z2-Z1)*(MM-1)/(KFACT/KK)
               BCX(NIDGEN)=BCX1
               BCY(NIDGEN)=BCY1
               BCZ(NIDGEN)=BCZ1
20          CONTINUE
            LM=KFACT1
          ELSE
            NID(LL)=NID1
            X(LL)=X1
            Y(LL)=Y1
            Z(LL)=Z1
            BCX(LL)=BCX1
            BCY(LL)=BCY1
            BCZ(LL)=BCZ1
            LM=1
          END IF
          LL=LL+LM
30 CONTINUE
   LL=1
   DO 60 WHILE(LL .LE. NEL)
        READ *,NELID1,NNTYPE,NN1,NN2,NN3,NN4,NN5,NN6,NN7,NN8,MAT1,
  +            KGEN
        IF(KGEN .NE. 0) THEN
          READ *,NELID2,NNN1,NNN2,NNN3,NNN4,NNN5,NNN6,NNN7,NNN8,KK
          KELFAC=NELID2-NELID1
          KEL1=KELFAC/KK
          KELF1=KEL1+1
          DO 40 MM=1,KELF1
             NELID=NELID1 + (MM-1)*KK
             ELID(NELID)=NELID
             NTYPE(NELID)=NNTYPE
             N1(NELID)=NN1 + (MM-1)*(NNN1-NN1)/KEL1
             N2(NELID)=NN2 + (MM-1)*(NNN2-NN2)/KEL1
             N3(NELID)=NN3 + (MM-1)*(NNN3-NN3)/KEL1
             N4(NELID)=NN4 + (MM-1)*(NNN4-NN4)/KEL1
             N5(NELID)=NN5 + (MM-1)*(NNN5-NN5)/KEL1
             N6(NELID)=NN6 + (MM-1)*(NNN6-NN6)/KEL1
             N7(NELID)=NN7 + (MM-1)*(NNN7-NN7)/KEL1
             N8(NELID)=NN8 + (MM-1)*(NNN8-NN8)/KEL1
```

```
          MAT(NELID)=MAT1
40      CONTINUE
        LM=KELF1
      ELSE
        ELID(LL)=NELID1
        NTYPE(LL)=NNTYPE
        N1(LL)=NN1
        N2(LL)=NN2
        N3(LL)=NN3
        N4(LL)=NN4
        N5(LL)=NN5
        N6(LL)=NN6
        N7(LL)=NN7
        N8(LL)=NN8
        MAT(LL)=MAT1
        LM=1
      END IF
      LL=LL+LM
50 CONTINUE
   READ *,(NIDMAT(I),EM(I),NUM(I),I=1,NMAT)
   DO 80 I=1,NEL
      E(I)=EM(MAT(I))
      NU(I)=NUM(MAT(I))
80 CONTINUE
C
   RETURN
   END
C
   SUBROUTINE DATA(N,DOF,BCX,BCY,BCZ,NEL,NID,X,Y,Z,ELID,NTYPE,
  +                N1,N2,N3,N4,N5,N6,N7,N8,E,NU,NODFOR,FX,FY,FZ,
  +                KEQNS,MEQNS,NLOAD,INTER,NGAUS1,NGAUS2,TITLE)
   IMPLICIT REAL*8 (A-H,O-Z)
   REAL*8 NU
   INTEGER DOF,BCX,BCY,BCZ,ELID,GAUS1,GAUS2
   CHARACTER*80 TITLE
   CHARACTER*5 TYPE(200)
   DIMENSION BCX(KEQNS),BCY(KEQNS),BCZ(KEQNS),NID(KEQNS),X(KEQNS)
   DIMENSION Y(KEQNS),Z(KEQNS),ELID(MEQNS),N1(MEQNS),N2(MEQNS)
   DIMENSION N3(MEQNS),N4(MEQNS),N5(MEQNS),N6(MEQNS),N7(MEQNS)
```

```
      DIMENSION N8(MEQNS),FX(100),FY(100),FZ(100),NODFOR(100)
      DIMENSION E(MEQNS),NU(MEQNS),NTYPE(MEQNS)
      PRINT 100,TITLE
100   FORMAT('1',A80)
      PRINT 110
110   FORMAT('- 3-D  STRESS  ANALYSIS  -  DATA  ECHO')
      PRINT 120,N
120   FORMAT('- NUMBER OF NODES',T34,':',I4)
      PRINT 130,DOF
130   FORMAT('- TOTAL NUMBER OF D.O.F.',T34,':',I4)
      NRES=0
      DO 150 I=1,N
        NRES=NRES+BCX(I)+BCY(I)+BCZ(I)
150   CONTINUE
      PRINT 180,NRES
180   FORMAT('- NUMBER OF RESTRAINED D.O.F.',T34,':',I4)
      PRINT 200,DOF-NRES
200   FORMAT('- NUMBER OF UNRESTRAINED D.O.F.',T34,':',I4)
      NEL4=0
      NEL5=0
      NEL6=0
      NEL12=0
      DO 220 I=1,NEL
        IF(NTYPE(I) .EQ. 4) THEN
          NEL4=NEL4+1
          TYPE(I)='TETRA'
        ELSE
          IF(NTYPE(I) .EQ. 5) THEN
            NEL5=NEL5+1
            TYPE(I)='PENTA'
          ELSE
            IF(NTYPE(I) .EQ. 6) THEN
              NEL6=NEL6+1
              TYPE(I)=' HEXA'
            ELSE
              NEL12=NEL12+1
              TYPE(I)='INTER'
            END IF
          END IF
```

```
      END IF
220 CONTINUE
    PRINT 240,NEL4
240 FORMAT('- NUMBER OF TETRA ELEMENTS',T34,':',I4)
    PRINT 260,NEL5
260 FORMAT('- NUMBER OF PENTA ELEMENTS',T34,':',I4)
    PRINT 280,NEL6
280 FORMAT('- NUMBER OF HEXA ELEMENTS',T34,':',I4)
    PRINT 300,NEL12
300 FORMAT('- NUMBER OF INTERFACE ELEMENTS',T34,':',I4)
    PRINT 320,NGAUS1,NGAUS1
320 FORMAT('- HEXA INTEGRATION ORDER',T34,':',I2,' BY',I2)
    PRINT 340,NGAUS2,NGAUS2
340 FORMAT('- INTER INTEGRATION ORDER',T34,':',I2,' BY',I2)
    PRINT 400
400 FORMAT('1 NODE COORDINATES')
    PRINT 420
420 FORMAT('- NODE',T15,'X',T25,'Y',T35,'Z',T42,'X-BC',T52,'Y-BC',
   +       T62,'Z-BC')
    PRINT 440,(NID(I),X(I),Y(I),Z(I),BCX(I),BCY(I),BCZ(I),I=1,N)
440 FORMAT('-',I3,T10,F7.1,T20,F7.1,T30,F7.1,T42,I3,T52,I3,T62,I3)
    PRINT 460
460 FORMAT('1 ELEMENT INCIDENCES')
    PRINT 480
480 FORMAT('- ELEMENT',T12,'TYPE',T17,'NODE-1',T24,'NODE-2',T31,
   +       'NODE-3',T38,'NODE-4',T45,'NODE-5',T52,'NODE-6',T59,
   +       'NODE-7',T66,'NODE-8',T78,'E',T88,'NU')
    PRINT 500,(ELID(I),TYPE(I),N1(I),N2(I),N3(I),N4(I),N5(I),N6(I),
   +          N7(I),N8(I),E(I),NU(I),I=1,NEL)
500 FORMAT('-',T4,I3,T11,A5,T19,I3,T26,I3,T33,I3,T40,I3,T47,I3,
   +       T54,I3,T61,I3,T68,I3,T74,E10.3,T87,F4.2)
    PRINT 520
520 FORMAT('1 APPLIED LOADS')
    IF(NLOAD .NE. 0) THEN
      PRINT 540
540   FORMAT('- NODE',T13,'X-FORCE',T28,'Y-FORCE',T43,'Z-FORCE')
      PRINT 560,(NODFOR(I),FX(I),FY(I),FZ(I),I=1,NLOAD)
560   FORMAT('-',I4,T11,E10.3,T26,E10.3,T41,E10.3)
    ELSE
```

```
      PRINT 580
 580  FORMAT('- NO CONCENTRATED LOADS APPLIED')
      END IF
C
      RETURN
      END

C
C
      SUBROUTINE STIFF(ELID,ELIDB,NTYPE,KGLOB,LEQNS,MM,NGAUSS,KEL)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 KEL(MM,MM),KELA(27,27),KELB(27,27)
      REAL*8 KELARA(27,36),KELBRB(27,36),KEL1(36,36),KEL2(36,36)
      REAL*8 KGLOB(LEQNS,LEQNS),NU
      INTEGER ELID,ELIDB
      DIMENSION Q(3,36),RA(27,36),RB(27,36),RAT(36,27),RBT(36,27)
      DIMENSION NOD(12),NGLOB(36),B(6,12)
      COMMON/GLOB1/X(200),Y(200),Z(200),E(200),NU(200)
      COMMON/GLOB2/N1(200),N2(200),N3(200),N4(200),N5(200),N6(200),
     +             N7(200),N8(200)
      IF(NTYPE .EQ. 4) THEN
        CALL TETRA(ELID,B,KEL)
        NNTYPE=4
      ELSE
        IF(NTYPE .EQ. 5) THEN
          CALL PENTA(ELID,KEL)
          NNTYPE=6
        ELSE
          IF(NTYPE .EQ. 6) THEN
            CALL HEXA(ELID,NTYPE,KEL,NGAUSS,24)
            NNTYPE=8
          ELSE
            CALL REL(ELID,ELIDB,Q)
            CALL HEXA(ELID,NTYPE,KELA,NGAUSS,27)
            CALL HEXA(ELIDB,NTYPE,KELB,NGAUSS,27)
            DO 100 LI=1,24
              DO 100 LJ=1,36
                RA(LI,LJ)=0.DO
                RB(LI,LJ)=0.DO
 100        CONTINUE
```

```
RA(1,4)=1.DO
RA(2,5)=1.DO
RA(3,6)=1.DO
RA(4,22)=1.DO
RA(5,23)=1.DO
RA(6,24)=1.DO
RA(7,31)=1.DO
RA(8,32)=1.DO
RA(9,33)=1.DO
RA(10,13)=1.DO
RA(11,14)=1.DO
RA(12,15)=1.DO
RA(13,1)=1.DO
RA(14,2)=1.DO
RA(15,3)=1.DO
RA(16,19)=1.DO
RA(17,20)=1.DO
RA(18,21)=1.DO
RA(19,34)=1.DO
RA(20,35)=1.DO
RA(21,36)=1.DO
RA(22,16)=1.DO
RA(23,17)=1.DO
RA(24,18)=1.DO
RB(1,22)=1.DO
RB(2,23)=1.DO
RB(3,24)=1.DO
RB(4,4)=1.DO
RB(5,5)=1.DO
RB(6,6)=1.DO
RB(7,13)=1.DO
RB(8,14)=1.DO
RB(9,15)=1.DO
RB(10,31)=1.DO
RB(11,32)=1.DO
RB(12,33)=1.DO
RB(13,25)=1.DO
RB(14,26)=1.DO
RB(15,27)=1.DO
```

```
                  RB(16,7)=1.DO
                  RB(17,8)=1.DO
                  RB(18,9)=1.DO
                  RB(19,10)=1.DO
                  RB(20,11)=1.DO
                  RB(21,12)=1.DO
                  RB(22,28)=1.DO
                  RB(23,29)=1.DO
                  RB(24,30)=1.DO
                  DO 120 K=25,27
                     DO 120 L=1,36
                        MMM=K-24
                        RA(K,L)=Q(MMM,L)
                        RB(K,L)=Q(MMM,L)
120               CONTINUE
                  DO 140 M1=1,27
                     DO 140 M2=1,36
                        RAT(M2,M1)=RA(M1,M2)
                        RBT(M2,M1)=RB(M1,M2)
140               CONTINUE
                  CALL MATMAT(27,27,36,KELA,RA,KELARA)
                  CALL MATMAT(36,27,36,RAT,KELARA,KEL1)
                  CALL MATMAT(27,27,36,KELB,RB,KELBRB)
                  CALL MATMAT(36,27,36,RBT,KELBRB,KEL2)
                  DO 160 K1=1,36
                     DO 160 K2=1,36
                        KEL(K1,K2)=KEL1(K1,K2) + KEL2(K1,K2)
160               CONTINUE
                END IF
              END IF
            END IF
            IF(NTYPE .EQ. 9) THEN
              NOD(1)=N5(ELID)
              NOD(2)=N1(ELID)
              NOD(3)=N6(ELIDB)
              NOD(4)=N7(ELIDB)
              NOD(5)=N3(ELIDB)
              NOD(6)=N8(ELID)
              NOD(7)=N6(ELID)
```

```
      NOD(8)=N2(ELID)
      NOD(9)=N5(ELIDB)
      NOD(10)=N8(ELIDB)
      NOD(11)=N4(ELIDB)
      NOD(12)=N7(ELID)
      NNTYPE=12
    ELSE
      NOD(1)=N1(ELID)
      NOD(2)=N2(ELID)
      NOD(3)=N3(ELID)
      NOD(4)=N4(ELID)
      IF(NTYPE .GT. 4)THEN
        NOD(5)=N5(ELID)
        NOD(6)=N6(ELID)
        IF(NTYPE .EQ. 6)THEN
          NOD(7)=N7(ELID)
          NOD(8)=N8(ELID)
        END IF
      END IF
    END IF
    DO 200 I=1,NNTYPE
       J=3*I-2
       K=3*I-1
       L=3*I
       NGLOB(J)=3*NOD(I)-2
       NGLOB(K)=3*NOD(I)-1
       NGLOB(L)=3*NOD(I)
200 CONTINUE
    DO 500 I=1,MM
      DO 500 J=1,MM
        KGLOB(NGLOB(I),NGLOB(J))=KGLOB(NGLOB(I),NGLOB(J))+KEL(I,J)
500 CONTINUE
C
    RETURN
    END
C
C
    SUBROUTINE TETRA(ELID,B,KEL)
    IMPLICIT REAL*8(A-H,O-Z)
```

```
      INTEGER ELID
      REAL*8 NU,KEL,JACOB
      DIMENSION KEL(12,12),B(6,12),EM(6,6),JACOB(3,3)
      DIMENSION GAMMA(3,3),BT(12,6),BTE(12,6)
      COMMON/GLOB1/X(200),Y(200),Z(200),E(200),NU(200)
      COMMON/GLOB2/N1(200),N2(200),N3(200),N4(200),N5(200),
     +             N6(200),N7(200),N8(200)
      JACOB(1,1)=X(N1(ELID))-X(N4(ELID))
      JACOB(1,2)=Y(N1(ELID))-Y(N4(ELID))
      JACOB(1,3)=Z(N1(ELID))-Z(N4(ELID))
      JACOB(2,1)=X(N2(ELID))-X(N4(ELID))
      JACOB(2,2)=Y(N2(ELID))-Y(N4(ELID))
      JACOB(2,3)=Z(N2(ELID))-Z(N4(ELID))
      JACOB(3,1)=-X(N3(ELID))+X(N4(ELID))
      JACOB(3,2)=-Y(N3(ELID))+Y(N4(ELID))
      JACOB(3,3)=-Z(N3(ELID))+Z(N4(ELID))
      DETJAC=JACOB(1,1)*(JACOB(2,2)*JACOB(3,3)-JACOB(2,3)*JACOB(3,2))-
     +       JACOB(2,1)*(JACOB(1,2)*JACOB(3,3)-JACOB(1,3)*JACOB(3,2))+
     +       JACOB(3,1)*(JACOB(1,2)*JACOB(2,3)-JACOB(1,3)*JACOB(2,2))
      IF(DETJAC .EQ. 0)THEN
        PRINT 20
20      FORMAT('1',2X,'*** FATAL ERROR ***')
        PRINT 25,ELID
25      FORMAT('-',2X,'JACOBIAN MATRIX FOR ELEMENT NO. : ',I5,
     +         ' IS ZERO')
        PRINT 30
30      FORMAT('-',2X,'POSSIBLE CAUSE : BAD NODAL NUMBERS',
     +         ' OR BAD NODAL COORDINATES')
        PRINT 31
31      FORMAT('1')
        STOP
      END IF
      IF(DETJAC .LT. 1.D-5)THEN
        PRINT 35
35      FORMAT('1',2X,'*** FATAL ERROR ***')
        PRINT 40,ELID
40      FORMAT('-',2X,'JACOBIAN MATRIX FOR ELEMENT NO.:',I5,
     +         ' IS NEGATIVE')
        PRINT 45
```

```
 45   FORMAT('-',2X,'POSSIBLE CAUSE : CLOCKWISE NUMBERING',
     +        ' OF NODES')
      PRINT 50
 50   FORMAT('-',2X,'NODES 1-2-3 SHOULD BE COUNTERCLOCKWISE',
     +        ' WHEN VIEWED FROM NODE 4')
      PRINT 51
 51   FORMAT('1')
      STOP
    END IF
    GAMMA(1,1)=(JACOB(2,2)*JACOB(3,3)-JACOB(2,3)*JACOB(3,2))/DETJAC
    GAMMA(1,2)=(JACOB(1,3)*JACOB(3,2)-JACOB(1,2)*JACOB(3,3))/DETJAC
    GAMMA(1,3)=(JACOB(1,2)*JACOB(2,3)-JACOB(1,3)*JACOB(2,2))/DETJAC
    GAMMA(2,1)=(JACOB(2,3)*JACOB(3,1)-JACOB(2,1)*JACOB(3,2))/DETJAC
    GAMMA(2,2)=(JACOB(1,1)*JACOB(3,3)-JACOB(1,3)*JACOB(3,1))/DETJAC
    GAMMA(2,3)=(JACOB(1,3)*JACOB(2,1)-JACOB(1,1)*JACOB(2,3))/DETJAC
    GAMMA(3,1)=(JACOB(2,1)*JACOB(3,2)-JACOB(2,2)*JACOB(3,1))/DETJAC
    GAMMA(3,2)=(JACOB(1,2)*JACOB(3,1)-JACOB(1,1)*JACOB(3,2))/DETJAC
    GAMMA(3,3)=(JACOB(1,1)*JACOB(2,2)-JACOB(1,2)*JACOB(2,1))/DETJAC
    DO 100 I=1,6
       DO 100 J=1,12
          B(I,J)=0.D0
100 CONTINUE
    DO 120 I=1,2
       IJ=3*I-2
       IK=3*I-1
       IL=3*I
       B(1,IJ)=GAMMA(1,I)
       B(2,IK)=GAMMA(2,I)
       B(3,IL)=GAMMA(3,I)
       B(4,IJ)=B(2,IK)
       B(4,IK)=B(1,IJ)
       B(5,IK)=B(3,IL)
       B(5,IL)=B(2,IK)
       B(6,IJ)=B(3,IL)
       B(6,IL)=B(1,IJ)
120 CONTINUE
    B(1,7)=-GAMMA(1,3)
    B(2,8)=-GAMMA(2,3)
    B(3,9)=-GAMMA(3,3)
```

```
      B(4,7)=-GAMMA(2,3)
      B(4,8)=-GAMMA(1,3)
      B(5,8)=-GAMMA(3,3)
      B(5,9)=-GAMMA(2,3)
      B(6,7)=-GAMMA(3,3)
      B(6,9)=-GAMMA(1,3)
      B(1,10)=-GAMMA(1,1)-GAMMA(1,2)+GAMMA(1,3)
      B(2,11)=-GAMMA(2,1)-GAMMA(2,2)+GAMMA(2,3)
      B(3,12)=-GAMMA(3,1)-GAMMA(3,2)+GAMMA(3,3)
      B(4,10)=B(2,11)
      B(4,11)=B(1,10)
      B(5,11)=B(3,12)
      B(5,12)=B(2,11)
      B(6,10)=B(3,12)
      B(6,12)=B(1,10)
      DO 140 I=1,12
         DO 140 J=1,6
            BT(I,J)=B(J,I)
  140 CONTINUE
      CALL YOUNG(EM,E(ELID),NU(ELID))
      CALL MATMAT(12,6,6,BT,EM,BTE)
      CALL MATMAT(12,6,12,BTE,B,KEL)
      DO 160 I=1,12
         DO 160 J=1,12
            KEL(I,J)=KEL(I,J)*DETJAC/6.DO
  160 CONTINUE
C
      RETURN
      END
C
C
      SUBROUTINE SHAPEF(XI,ET,ZET,B,ELID,NTYPE,DETJAC)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 JACOB(3,3),NU
      DIMENSION RXI(9),RET(9),RZET(9),XX(9),YY(9),ZZ(9)
      DIMENSION RC(9),RD(9),RE(9)
      DIMENSION B(6,27),EM(6,6),ENXI(9),ENET(9),ENZET(9)
      INTEGER ELID
      COMMON/Q8/EN(9),GAMMA(3,3)
```

```
      COMMON/GLOB1/X(200),Y(200),Z(200),E(200),NU(200)
      COMMON/GLOB2/N1(200),N2(200),N3(200),N4(200),N5(200),N6(200),
     +            N7(200),N8(200)
      DATA RXI/-1.D0,-1.D0,-1.D0,-1.D0,1.D0,1.D0,1.D0,1.D0,-1.D0/
      DATA RET/-1.D0,-1.D0,1.D0,1.D0,-1.D0,-1.D0,1.D0,1.D0,-1.D0/
      DATA RZET/-1.D0,1.D0,1.D0,-1.D0,-1.D0,1.D0,1.D0,-1.D0,-1.D0/
      DATA RC/1.D0,1.D0,1.D0,1.D0,0.D0,0.D0,0.D0,0.D0,0.D0/
      DATA RD/0.D0,0.D0,0.D0,0.D0,1.D0,1.D0,1.D0,1.D0,0.D0/
      DATA RE/0.D0,0.D0,0.D0,0.D0,0.D0,0.D0,0.D0,0.D0,4.D0/
      IF(NTYPE .EQ. 6) THEN
        NORDER=8
      ELSE
        NORDER=9
      END IF
      IF(NTYPE .EQ. 5)THEN
        NORDER=6
        FF1=1.-XI-ET
        FF2=1.-ZET
        FF3=1.+ZET
        ENXI(1)=-FF2/2.
        ENXI(2)=FF2/2.
        ENXI(3)=0.D0
        ENXI(4)=-FF3/2.
        ENXI(5)=FF3/2.
        ENXI(6)=0.D0
        ENET(1)=-FF2/2.
        ENET(2)=0.D0
        ENET(3)=FF2/2.
        ENET(4)=-FF3/2.
        ENET(5)=0.D0
        ENET(6)=FF3/2.
        ENZET(1)=-FF1/2.
        ENZET(2)=-XI/2.
        ENZET(3)=-ET/2.
        ENZET(4)=FF1/2.
        ENZET(5)=XI/2.
        ENZET(6)=ET/2.
      ELSE
        DO 40 I=1,NORDER
```

```
            F1=1. + RXI(I)*XI
            F2=1. + RET(I)*ET
            F3=1. + RZET(I)*ZET
            F4=RET(I)*ET+RZET(I)*ZET-RET(I)*RZET(I)*ET*ZET
            F5=(1.+ET)*(1.+ZET)
            F6=1. + 2.*RET(I)*ET-2.*RET(I)*RZET(I)*ET*ZET
            F7=ET*(1.+ZET)
            F8=1. + 2.*RZET(I)*ZET-2.*RET(I)*RZET(I)*ET*ZET
            F9=ZET*(1.+ET)
            IF(NTYPE .EQ. 6) THEN
              EN(I)=F1*F2*F3/8.
              ENXI(I)=RXI(I)*F2*F3/8.
              ENET(I)=RET(I)*F1*F3/8.
              ENZET(I)=RZET(I)*F1*F2/8.
            ELSE
              EN(I)=F1*F2*F3*(RC(I)*F4+RD(I)+RE(I)*F5)/8.
              ENXI(I)=RXI(I)*F2*F3*(RC(I)*F4+RD(I)+RE(I)*F5)/8.
              ENET(I)=RET(I)*F1*F3*(RC(I)*F6+RD(I)+2.*RE(I)*F7)/8.
              ENZET(I)=RZET(I)*F1*F2*(RC(I)*F8+RD(I)+2.*RE(I)*F9)/8.
            END IF
 40      CONTINUE
      END IF
      NSIZE=3*NORDER
      DO 60 I=1,6
         DO 60 J=1,NSIZE
            B(I,J)=0.DO
 60   CONTINUE
      DO 80 I=1,3
         DO 80 J=1,3
            JACOB(I,J)=0.DO
 80   CONTINUE
      XX(1)=X(N1(ELID))
      XX(2)=X(N2(ELID))
      XX(3)=X(N3(ELID))
      XX(4)=X(N4(ELID))
      XX(5)=X(N5(ELID))
      XX(6)=X(N6(ELID))
      YY(1)=Y(N1(ELID))
      YY(2)=Y(N2(ELID))
```

```
      YY(3)=Y(N3(ELID))
      YY(4)=Y(N4(ELID))
      YY(5)=Y(N5(ELID))
      YY(6)=Y(N6(ELID))
      ZZ(1)=Z(N1(ELID))
      ZZ(2)=Z(N2(ELID))
      ZZ(3)=Z(N3(ELID))
      ZZ(4)=Z(N4(ELID))
      ZZ(5)=Z(N5(ELID))
      ZZ(6)=Z(N6(ELID))
      IF(NTYPE .GT. 5)THEN
        XX(7)=X(N7(ELID))
        XX(8)=X(N8(ELID))
        YY(7)=Y(N7(ELID))
        YY(8)=Y(N8(ELID))
        ZZ(7)=Z(N7(ELID))
        ZZ(8)=Z(N8(ELID))
        XX(9)=(XX(1)+XX(2)+XX(3)+XX(4))/4.
        YY(9)=(YY(1)+YY(2)+YY(3)+YY(4))/4.
        ZZ(9)=(ZZ(1)+ZZ(2)+ZZ(3)+ZZ(4))/4.
      END IF
      DO 100 I=1,NORDER
         JACOB(1,1)=JACOB(1,1) + ENXI(I)*XX(I)
         JACOB(1,2)=JACOB(1,2) + ENXI(I)*YY(I)
         JACOB(1,3)=JACOB(1,3) + ENXI(I)*ZZ(I)
         JACOB(2,1)=JACOB(2,1) + ENET(I)*XX(I)
         JACOB(2,2)=JACOB(2,2) + ENET(I)*YY(I)
         JACOB(2,3)=JACOB(2,3) + ENET(I)*ZZ(I)
         JACOB(3,1)=JACOB(3,1) + ENZET(I)*XX(I)
         JACOB(3,2)=JACOB(3,2) + ENZET(I)*YY(I)
         JACOB(3,3)=JACOB(3,3) + ENZET(I)*ZZ(I)
  100 CONTINUE
      DETJAC=JACOB(1,1)*(JACOB(2,2)*JACOB(3,3)-JACOB(2,3)*JACOB(3,2))-
     +        JACOB(2,1)*(JACOB(1,2)*JACOB(3,3)-JACOB(1,3)*JACOB(3,2))+
     +        JACOB(3,1)*(JACOB(1,2)*JACOB(2,3)-JACOB(1,3)*JACOB(2,2))
      GAMMA(1,1)=(JACOB(2,2)*JACOB(3,3)-JACOB(2,3)*JACOB(3,2))/DETJAC
      GAMMA(1,2)=(JACOB(1,3)*JACOB(3,2)-JACOB(1,2)*JACOB(3,3))/DETJAC
      GAMMA(1,3)=(JACOB(1,2)*JACOB(2,3)-JACOB(1,3)*JACOB(2,2))/DETJAC
      GAMMA(2,1)=(JACOB(2,3)*JACOB(3,1)-JACOB(2,1)*JACOB(3,3))/DETJAC
```

```
      GAMMA(2,2)=(JACOB(1,1)*JACOB(3,3)-JACOB(1,3)*JACOB(3,1))/DETJAC
      GAMMA(2,3)=(JACOB(1,3)*JACOB(2,1)-JACOB(1,1)*JACOB(2,3))/DETJAC
      GAMMA(3,1)=(JACOB(2,1)*JACOB(3,2)-JACOB(2,2)*JACOB(3,1))/DETJAC
      GAMMA(3,2)=(JACOB(1,2)*JACOB(3,1)-JACOB(1,1)*JACOB(3,2))/DETJAC
      GAMMA(3,3)=(JACOB(1,1)*JACOB(2,2)-JACOB(1,2)*JACOB(2,1))/DETJAC
      DO 200 I=1,NORDER
         J=3*I
         K=3*I-1
         L=3*I-2
         B(1,L)=GAMMA(1,1)*ENXI(I)+GAMMA(1,2)*ENET(I)+
     +          GAMMA(1,3)*ENZET(I)
         B(1,K)=0.DO
         B(1,J)=0.DO
         B(2,L)=0.DO
         B(2,K)=GAMMA(2,1)*ENXI(I)+GAMMA(2,2)*ENET(I)+
     +          GAMMA(2,3)*ENZET(I)
         B(2,J)=0.DO
         B(3,L)=0.DO
         B(3,K)=0.DO
         B(3,J)=GAMMA(3,1)*ENXI(I)+GAMMA(3,2)*ENET(I)+
     +          GAMMA(3,3)*ENZET(I)
         B(4,L)=B(2,K)
         B(4,K)=B(1,L)
         B(4,J)=0.DO
         B(5,L)=0.DO
         B(5,K)=B(3,J)
         B(5,J)=B(2,K)
         B(6,L)=B(3,J)
         B(6,K)=0.DO
         B(6,J)=B(1,L)
  200 CONTINUE
C
      RETURN
      END
C
C
      SUBROUTINE PENTA(NELEM,KEL)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NU,KEL(18,18)
```

```
      DIMENSION PLACE1(4,3),B(6,18),BTE(18,6)
      DIMENSION EM(6,6),PLACE2(2,2)
      COMMON/GLOB1/X(200),Y(200),Z(200),E(200),NU(200)
      PLACE1(1,1)=1.D0/3.D0
      PLACE1(1,2)=PLACE1(1,1)
      PLACE1(1,3)=-9.D0/32.D0
      PLACE1(2,1)=3.D0/5.D0
      PLACE1(2,2)=1.D0/5.D0
      PLACE1(2,3)=25.D0/96.D0
      PLACE1(3,1)=PLACE1(2,2)
      PLACE1(3,2)=PLACE1(2,1)
      PLACE1(3,3)=PLACE1(2,3)
      PLACE1(4,1)=PLACE1(2,2)
      PLACE1(4,2)=PLACE1(2,2)
      PLACE1(4,3)=PLACE1(2,3)
      PLACE2(1,1)=1.D0/DSQRT(3.D0)
      PLACE2(1,2)=1.D0
      PLACE2(2,1)=-PLACE2(1,1)
      PLACE2(2,2)=1.D0
      CALL YOUNG(EM,E(NELEM),NU(NELEM))
      NSIZE=18
      NORDER=6
      DO 40 I=1,NSIZE
         DO 40 J=1,NSIZE
            KEL(I,J)=0.D0
   40 CONTINUE
      DO 200 NA=1,4
         XI=PLACE1(NA,1)
         ET=PLACE1(NA,2)
         DO 180 NB=1,2
            ZET=PLACE2(NB,1)
            CALL SHAPEF(XI,ET,ZET,B,NELEM,5,DETJAC)
            DV=PLACE1(NA,3)*PLACE2(NB,2)*DETJAC
            DO 100 J=1,NORDER
               K=3*J-2
               L=3*J-1
               M=3*J
               DO 80 I=1,3
                  BTE(K,I)=B(1,K)*EM(1,I)
```

```
               BTE(L,I)=B(2,L)*EM(2,I)
               BTE(M,I)=B(3,M)*EM(3,I)
 80         CONTINUE
            BTE(K,5)=0.D0
            BTE(L,6)=0.D0
            BTE(M,4)=0.D0
            BTE(K,4)=B(4,K)*EM(4,4)
            BTE(K,6)=B(6,K)*EM(6,6)
            BTE(L,4)=B(4,L)*EM(4,4)
            BTE(L,5)=B(5,L)*EM(5,5)
            BTE(M,5)=B(5,M)*EM(5,5)
            BTE(M,6)=B(6,M)*EM(6,6)
100      CONTINUE
         DO 140 NROW=1,NSIZE
            DO 120 NCOL=NROW,NSIZE
               DUM=0.D0
               DO 110 JR=1,6
                  DUM=DUM+BTE(NROW,JR)*B(JR,NCOL)
110            CONTINUE
               KEL(NROW,NCOL)=KEL(NROW,NCOL)+DUM*DV
120         CONTINUE
140      CONTINUE
180   CONTINUE
200 CONTINUE
    DO 220 K=1,NSIZE
       DO 220 L=K,NSIZE
          KEL(L,K)=KEL(K,L)
220 CONTINUE
C
    RETURN
    END
C
C
    SUBROUTINE HEXA(NELEM,NTYPE,KEL,NGAUSS,NM)
    IMPLICIT REAL*8(A-H,O-Z)
    REAL*8 NU,KEL(NM,NM)
    DIMENSION PLACE(4,4),WGT(4,4),B(6,27),BTE(27,6)
    DIMENSION EM(6,6),BT(27,6)
    COMMON/Q8/EN(9),GAMMA(3,3)
```

```
      COMMON/GLOB1/X(200),Y(200),Z(200),E(200),NU(200)
      DATA PLACE(1,1),PLACE(2,1),PLACE(3,1),PLACE(4,1)/4*0.D0/
      DATA PLACE(3,2),PLACE(4,2),PLACE(2,3),PLACE(4,3)/4*0.D0/
      DATA PLACE(1,2)/-0.577350269189626D0/
      DATA PLACE(2,2)/0.577350269189626D0/
      DATA PLACE(1,3)/-0.774596669241483D0/
      DATA PLACE(3,3)/0.774596669241483D0/
      DATA PLACE(1,4)/-0.861136311594053D0/
      DATA PLACE(2,4)/-0.339981043584856D0/
      DATA PLACE(3,4)/0.339981043584856D0/
      DATA PLACE(4,4)/0.861136311594053D0/
      DATA WGT(1,1)/2.D0/,WGT(1,2),WGT(2,2)/2*1.D0/
      DATA WGT(2,1),WGT(3,1),WGT(4,1),WGT(3,2)/4*0.D0/
      DATA WGT(4,2),WGT(4,3)/2*0.D0/,WGT(2,3)/0.888888888888889D0/
      DATA WGT(1,3),WGT(3,3)/2*0.555555555555556D0/
      DATA WGT(1,4),WGT(4,4)/2*0.347854845137454D0/
      DATA WGT(2,4),WGT(3,4)/2*0.652145154862546D0/
      CALL YOUNG(EM,E(NELEM),NU(NELEM))
      IF(NTYPE .EQ. 6) THEN
        NSIZE=24
        NORDER=8
      ELSE
        NSIZE=27
        NORDER=9
      END IF
      DO 40 I=1,NSIZE
         DO 40 J=1,NSIZE
            KEL(I,J)=0.D0
   40 CONTINUE
      DO 200 NA=1,NGAUSS
         XI=PLACE(NA,NGAUSS)
         DO 180 NB=1,NGAUSS
            ET=PLACE(NB,NGAUSS)
            DO 160 NC=1,NGAUSS
               ZET=PLACE(NC,NGAUSS)
               CALL SHAPEF(XI,ET,ZET,B,NELEM,NTYPE,DETJAC)
               DV=WGT(NA,NGAUSS)*WGT(NB,NGAUSS)*WGT(NC,NGAUSS)*DETJAC
               DO 60 I=1,6
                  DO 60 J=1,NSIZE
```

```
            BT(J,I)=B(I,J)
 60       CONTINUE
          DO 100 J=1,NORDER
            K=3*J - 2
            L=3*J - 1
            M=3*J
            DO 80 I=1,3
              BTE(K,I)=B(1,K)*EM(1,I)
              BTE(L,I)=B(2,L)*EM(2,I)
              BTE(M,I)=B(3,M)*EM(3,I)
 80         CONTINUE
            BTE(K,5)=0.DO
            BTE(L,6)=0.DO
            BTE(M,4)=0.DO
            BTE(K,4)=B(4,K)*EM(4,4)
            BTE(K,6)=B(6,K)*EM(6,6)
            BTE(L,4)=B(4,L)*EM(4,4)
            BTE(L,5)=B(5,L)*EM(5,5)
            BTE(M,5)=B(5,M)*EM(5,5)
            BTE(M,6)=B(6,M)*EM(6,6)
100       CONTINUE
          DO 140 NROW=1,NSIZE
            DO 120 NCOL=NROW,NSIZE
              DUM=O.DO
              DO 110 J=1,6
                DUM=DUM+BTE(NROW,J)*B(J,NCOL)
110           CONTINUE
              KEL(NROW,NCOL)=KEL(NROW,NCOL) + DUM*DV
120         CONTINUE
140       CONTINUE
160     CONTINUE
180   CONTINUE
200 CONTINUE
    DO 220 K=1,NSIZE
      DO 220 L=K,NSIZE
        KEL(L,K)=KEL(K,L)
220 CONTINUE
C
    RETURN
```

```
      END
C
C
      SUBROUTINE REL(NELEM1,NELEM2,Q)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 NU
      DIMENSION TRAN1(3,6),TRAN2(3,6),BA(6,27),BB(6,27)
      DIMENSION EA(6,6),EB(6,6),PROD1(6,27)
      DIMENSION QA(3,27),QB(3,27),Q1(3,3),Q2(3,36),Q(3,36),
      COMMON/GLOB1/X(200),Y(200),Z(200),E(200),NU(200)
      COMMON/GLOB2/N1(200),N2(200),N3(200),N4(200),N5(200),N6(200),
     +             N7(200),N8(200)
      CALL TRANSF(NELEM1,X(N1(NELEM1)),X(N2(NELEM1)),X(N3(NELEM1)),
     +           X(N4(NELEM1)),Y(N1(NELEM1)),Y(N2(NELEM1)),
     +           Y(N3(NELEM1)),Y(N4(NELEM1)),Z(N1(NELEM1)),
     +           Z(N2(NELEM1)),Z(N3(NELEM1)),Z(N4(NELEM1)),TRAN1,3,6)
      XI=-1.DO
      ET=O.DO
      ZET=O.DO
      CALL SHAPEF(XI,ET,ZET,BA,NELEM1,12,DETJAC)
      PRINT 26,BA(2,8),BA(3,9),BA(4,10),BA(4,17),BA(5,11)
      CALL YOUNG(EA,E(NELEM1),NU(NELEM1))
      CALL SHAPEF(XI,ET,ZET,BB,NELEM2,12,DETJAC)
      PRINT 26,BB(2,8),BB(3,9),BB(4,10),BB(4,17),BB(5,11)
   26 FORMAT('-',2X,5(E10.3))
      CALL YOUNG(EB,E(NELEM2),NU(NELEM2))
      CALL MATMAT(6,6,27,EA,BA,PROD1)
      CALL MATMAT(3,6,27,TRAN1,PROD1,QA)
      CALL MATMAT(6,6,27,EB,BB,PROD1)
      CALL MATMAT(3,6,27,TRAN1,PROD1,QB)
      A11=QB(1,25)-QA(1,25)
      A12=QB(1,26)-QA(1,26)
      A13=QB(1,27)-QA(1,27)
      A21=QB(2,25)-QA(2,25)
      A22=QB(2,26)-QA(2,26)
      A23=QB(2,27)-QA(2,27)
      A31=QB(3,25)-QA(3,25)
      A32=QB(3,26)-QA(3,26)
      A33=QB(3,27)-QA(3,27)
```

```
      PRINT 19
      DETA=A11*(A22*A33-A23*A32)-A21*(A12*A33-A13*A32)+A31*(A12*A23-
     +          A13*A22)
      Q1(1,1)=(A22*A33-A32*A23)/DETA
      Q1(1,2)=(A32*A13-A12*A33)/DETA
      Q1(1,3)=(A12*A23-A22*A13)/DETA
      Q1(2,1)=(A23*A31-A33*A21)/DETA
      Q1(2,2)=(A33*A11-A13*A31)/DETA
      Q1(2,3)=(A13*A21-A23*A11)/DETA
      Q1(3,1)=(A21*A32-A31*A22)/DETA
      Q1(3,2)=(A31*A12-A11*A32)/DETA
      Q1(3,3)=(A11*A22-A21*A12)/DETA
      DO 100 I=1,3
         Q2(I,1)=QA(I,13)
         Q2(I,2)=QA(I,14)
         Q2(I,3)=QA(I,15)
         Q2(I,4)=QA(I,1)-QB(I,4)
         Q2(I,5)=QA(I,2)-QB(I,5)
         Q2(I,6)=QA(I,3)-QB(I,6)
         Q2(I,7)=-QB(I,16)
         Q2(I,8)=-QB(I,17)
         Q2(I,9)=-QB(I,18)
         Q2(I,10)=-QB(I,19)
         Q2(I,11)=-QB(I,20)
         Q2(I,12)=-QB(I,21)
         Q2(I,13)=QA(I,10)-QB(I,7)
         Q2(I,14)=QA(I,11)-QB(I,8)
         Q2(I,15)=QA(I,12)-QB(I,9)
         Q2(I,16)=QA(I,22)
         Q2(I,17)=QA(I,23)
         Q2(I,18)=QA(I,24)
         Q2(I,19)=QA(I,16)
         Q2(I,20)=QA(I,17)
         Q2(I,21)=QA(I,18)
         Q2(I,22)=QA(I,4)-QB(I,1)
         Q2(I,23)=QA(I,5)-QB(I,2)
         Q2(I,24)=QA(I,6)-QB(I,3)
         Q2(I,25)=-QB(I,13)
         Q2(I,26)=-QB(I,14)
```

```
      Q2(I,27)=-QB(I,15)
      Q2(I,28)=-QB(I,22)
      Q2(I,29)=-QB(I,23)
      Q2(I,30)=-QB(I,24)
      Q2(I,31)=QA(I,7)-QB(I,10)
      Q2(I,32)=QA(I,8)-QB(I,11)
      Q2(I,33)=QA(I,9)-QB(I,12)
      Q2(I,34)=QA(I,19)
      Q2(I,35)=QA(I,20)
      Q2(I,36)=QA(I,21)
  100 CONTINUE
      CALL MATMAT(3,3,36,Q1,Q2,Q)
C
      RETURN
      END
C
C
      SUBROUTINE TRANSF(NELEM,X1,X2,X3,X4,Y1,Y2,Y3,Y4,Z1,Z2,Z3,Z4,
     +                  TRAN,NROW,NCOL)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION TRAN(NROW,NCOL)
      REAL*8 L1,L2,L3,M1,M2,M3,N1,N2,N3
      XA=(X1+X2)/2.
      YA=(Y1+Y2)/2.
      ZA=(Z1+Z2)/2.
      XB=(X2+X3)/2.
      YB=(Y2+Y3)/2.
      ZB=(Z2+Z3)/2.
      XC=(X3+X4)/2.
      YC=(Y3+Y4)/2.
      ZC=(Z3+Z4)/2.
      XD=(X4+X1)/2.
      YD=(Y4+Y1)/2.
      ZD=(Z4+Z1)/2.
      RL2=DSQRT((XB-XD)*(XB-XD) + (YB-YD)*(YB-YD) +(ZB-ZD)*(ZB-ZD))
      L2=(XB-XD)/RL2
      M2=(YB-YD)/RL2
      N2=(ZB-ZD)/RL2
      RL3=DSQRT((XC-XA)*(XC-XA) + (YC-YA)*(YC-YA) +(ZC-ZA)*(ZC-ZA))
```

```
L3=(XC-XA)/RL3
M3=(YC-YA)/RL3
N3=(ZC-ZA)/RL3
X=(YB-YD)*(ZC-ZA)-(YC-YA)*(ZB-ZD)
Y=(ZB-ZD)*(XC-XA)-(ZC-ZA)*(XB-XD)
Z=(XB-XD)*(YC-YA)-(XC-XA)*(YB-YD)
RL1=DSQRT(X*X + Y*Y + Z*Z)
L1=X/RL1
M1=Y/RL1
N1=Z/RL1
TRAN(1,1)=L1*L1
TRAN(1,2)=M1*M1
TRAN(1,3)=N1*N1
TRAN(1,4)=2.DO*L1*M1
TRAN(1,5)=2.DO*M1*N1
TRAN(1,6)=2.DO*N1*L1
IF(NROW .EQ. 3) THEN
  TRAN(2,1)=L1*L2
  TRAN(2,2)=M1*M2
  TRAN(2,3)=N1*N2
  TRAN(2,4)=L1*M2 + L2*M1
  TRAN(2,5)=M1*N2 + M2*N1
  TRAN(2,6)=N1*L2 + N2*L1
  TRAN(3,1)=L3*L1
  TRAN(3,2)=M3*M1
  TRAN(3,3)=N3*N1
  TRAN(3,4)=L3*M1 + L1*M3
  TRAN(3,5)=M3*N1 + M1*N3
  TRAN(3,6)=N3*L1 + N1*L3
ELSE
  TRAN(2,1)=L2*L2
  TRAN(2,2)=M2*M2
  TRAN(2,3)=N2*N2
  TRAN(2,4)=2.DO*L2*M2
  TRAN(2,5)=2.DO*M2*N2
  TRAN(2,6)=2.DO*N2*L2
  TRAN(3,1)=L3*L3
  TRAN(3,2)=M3*M3
  TRAN(3,3)=N3*N3
```

```
          TRAN(3,4)=2.D0*L3*M3
          TRAN(3,5)=2.D0*M3*N3
          TRAN(3,6)=2.D0*N3*L3
          TRAN(4,1)=L1*L2
          TRAN(4,2)=M1*M2
          TRAN(4,3)=N1*N2
          TRAN(4,4)=L1*M2 + L2*M1
          TRAN(4,5)=M1*N2 + M2*N1
          TRAN(4,6)=N1*L2 + N2*L1
          TRAN(5,1)=L2*L3
          TRAN(5,2)=M2*M3
          TRAN(5,3)=N2*N3
          TRAN(5,4)=L2*M3 + M2*L3
          TRAN(5,5)=M2*N3 + M3*N2
          TRAN(5,6)=N2*L3 + N3*L2
          TRAN(6,1)=L3*L1
          TRAN(6,2)=M3*M1
          TRAN(6,3)=N3*N1
          TRAN(6,4)=L3*M1 + L1*M3
          TRAN(6,5)=M3*N1 + M1*N3
          TRAN(6,6)=N3*L1 + N1*L3
        END IF
C
        RETURN
        END
C
C
        SUBROUTINE STETRA(ELID,XDISP,LEQNS)
        IMPLICIT REAL*8(A-H,O-Z)
        INTEGER ELID
        REAL*8 NU,KEL
        DIMENSION B(6,12),S(6),XDISP(LEQNS),XLOC(12)
        DIMENSION EM(6,6),PROD(6,12),KEL(12,12),NOD(4)
        COMMON/GLOB1/X(200),Y(200),Z(200),E(200),NU(200)
        COMMON/GLOB2/N1(200),N2(200),N3(200),N4(200),N5(200),
     +               N6(200),N7(200),N8(200)
        NOD(1)=N1(ELID)
        NOD(2)=N2(ELID)
        NOD(3)=N3(ELID)
```

```
      NOD(4)=N4(ELID)
      DO 20 I=1,4
         XLOC(3*I-2)=XDISP(3*NOD(I)-2)
         XLOC(3*I-1)=XDISP(3*NOD(I)-1)
         XLOC(3*I)=XDISP(3*NOD(I))
   20 CONTINUE
      CALL YOUNG(EM,E(ELID),NU(ELID))
      CALL TETRA(ELID,B,KEL)
      CALL MATMAT(6,6,12,EM,B,PROD)
      CALL MATVEC(6,12,PROD,XLOC,S,6,12)
      PRINT 40,ELID
   40 FORMAT('- ELEMENT NO. : ',I4,2X,'TETRA')
      PRINT 50
   50 FORMAT('-',T2,'NODE',T9,'S 11',T20,'S 22',T31,'S 33',T42,'T XY',
     +          T53,'T YZ',T64,'T ZX',T75,'S I',T86,'S II',T97,'S III')
      CALL PRINC(0,ELID,S)
C
      RETURN
      END
C
C
      SUBROUTINE STRES(NELEM,XDISP,LEQNS,NTYPE)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NU
      DIMENSION XDISP(LEQNS),XI(8),ET(8),ZET(8),S(6)
      DIMENSION NOD(8),XLOC(27),B(6,27),EM(6,6),XX(3),PROD(6,27)
      CHARACTER*5 TYPE
      COMMON/GLOB1/X(200),Y(200),Z(200),E(200),NU(200)
      COMMON/GLOB2/N1(200),N2(200),N3(200),N4(200),N5(200),N6(200),
     +            N7(200),N8(200)
      DATA XI/-1.D0,-1.D0,-1.D0,-1.D0,1.D0,1.D0,1.D0,1.D0/
      DATA ET/-1.D0,-1.D0,1.D0,1.D0,-1.D0,-1.D0,1.D0,1.D0/
      DATA ZET/-1.D0,1.D0,1.D0,-1.D0,-1.D0,1.D0,1.D0,-1.D0/
      IF(NTYPE .EQ. 5)THEN
        NT=6
        NTIME=6
        TYPE='PENTA'
      ELSE
        NT=8
```

```
        IF(NTYPE .EQ. 6) THEN
          NTIME=8
          TYPE=' HEXA'
        ELSE
          NTIME=9
          TYPE='INTER'
        END IF
      END IF
      PRINT 20,NELEM,TYPE
   20 FORMAT('- ELEMENT NO. : ',I3,2X,A5)
      MSIZE=3*NTIME
      NSIZE=NTIME+1
      CALL YOUNG(EM,E(NELEM),NU(NELEM))
      NOD(1)=N1(NELEM)
      NOD(2)=N2(NELEM)
      NOD(3)=N3(NELEM)
      NOD(4)=N4(NELEM)
      NOD(5)=N5(NELEM)
      NOD(6)=N6(NELEM)
      IF(NTYPE .GT. 5)THEN
        NOD(7)=N7(NELEM)
        NOD(8)=N8(NELEM)
      END IF
      DO 100 I=1,NT
         XLOC(3*I-2)=XDISP(3*NOD(I)-2)
         XLOC(3*I-1)=XDISP(3*NOD(I)-1)
         XLOC(3*I)=XDISP(3*NOD(I))
  100 CONTINUE
      IF(NTIME .EQ. 9) THEN
        CALL DISPL(NELEM,XDISP,XX,LEQNS)
        XLOC(25)=XX(1)
        XLOC(26)=XX(2)
        XLOC(27)=XX(3)
      END IF
      PRINT 150
  150 FORMAT('-',T2,'NODE',T9,'S 11',T20,'S 22',T31,'S 33',T42,'T XY',
     +       T53,'T YZ',T64,'T ZX',T75,'S I',T86,'S II',T97,'S III')
      IF(NTIME .EQ. 9) THEN
        CALL SHAPEF(-1.DO,0.DO,0.DO,B,NELEM,NTYPE,DETJAC)
```

```
          CALL MATMAT(6,6,MSIZE,EM,B,PROD)
          CALL MATVEC(6,MSIZE,PROD,XLOC,S,6,MSIZE)
          CALL PRINC(-1,NELEM,S)
        END IF
        IF(NTYPE .EQ. 5)THEN
          XI1=1.D0/3.D0
          ET1=1.D0/3.D0
          CALL SHAPEF(XI1,ET1,0.D0,B,NELEM,5,DETJAC)
        ELSE
          CALL SHAPEF(0.D0,0.D0,0.D0,B,NELEM,NTYPE,DETJAC)
        END IF
        CALL MATMAT(6,6,MSIZE,EM,B,PROD)
        CALL MATVEC(6,MSIZE,PROD,XLOC,S,6,MSIZE)
        CALL PRINC(0,NELEM,S)
C
        RETURN
        END
C
C
        SUBROUTINE PRINC(NOD,NELEM,S)
        IMPLICIT REAL*8 (A-H,O-Z)
        DIMENSION S(6),SP(6),SL(6)
        DIMENSION TRAN(6,6),A(4),ZR(3)
        REAL*8 NU
        CHARACTER*1 A1,A2
        COMMON/GLOB1/X(200),Y(200),Z(200),E(200),NU(200)
        COMMON/GLOB2/N1(200),N2(200),N3(200),N4(200),N5(200),N6(200),
       +             N7(200),N8(200)
        COMMON/ELCON/NELA(100),NELB(100),INTER
        A1='I'
        A2='C'
        IF(NOD .LT. 0) THEN
          NELID=NELEM
          I=1
          DO 20 WHILE(NELEM .NE. NELB(I) .AND. I .LE. INTER)
               I=I+1
  20      CONTINUE
          IF(NELEM .EQ. NELB(I))THEN
            NELEM=NELA(I)
```

```
      END IF
      CALL TRANSF(NELEM,X(N1(NELEM)),X(N2(NELEM)),X(N3(NELEM)),
     +             X(N4(NELEM)),Y(N1(NELEM)),Y(N2(NELEM)),Y(N3(NELEM)),
     +             Y(N4(NELEM)),Z(N1(NELEM)),Z(N2(NELEM)),Z(N3(NELEM)),
     +             Z(N4(NELEM)),TRAN,6,6)
      CALL MATVEC(6,6,TRAN,S,SL,6,6)
      A(1)=1.D0
      A(2)=-(SL(1)+SL(2)+SL(3))
      A(3)=SL(1)*SL(2)+SL(2)*SL(3)+SL(1)*SL(3)-SL(4)*SL(4)-
     +     SL(5)*SL(5)-SL(6)*SL(6)
      A(4)=-(SL(1)*SL(2)*SL(3)+2.*SL(4)*SL(5)*SL(6)-SL(1)*SL(5)*SL(5)-
     +       SL(2)*SL(6)*SL(6)-SL(3)*SL(4)*SL(4))
      CALL CUBIC(A,ZR)
      PRINT 50,A1,(SL(I),I=1,6),(ZR(K),K=1,3)
 50   FORMAT('-',A3,T6,E10.3,T17,E10.3,T28,E10.3,T39,E10.3,T50,E10.3,
     +      T61,E10.3,T72,E10.3,T83,E10.3,T94,E10.3)
    ELSE
      A(1)=1.
      A(2)=-(S(1)+S(2)+S(3))
      A(3)=S(1)*S(2)+S(2)*S(3)+S(1)*S(3)-S(4)*S(4)-S(5)*S(5)-S(6)*S(6)
      A(4)=-(S(1)*S(2)*S(3)+2.*S(4)*S(5)*S(6)-S(1)*S(5)*S(5)-
     +       S(2)*S(6)*S(6)-S(3)*S(4)*S(4))
      CALL CUBIC(A,ZR)
      PRINT 100,A2,(S(I),I=1,6),(ZR(K),K=1,3)
100   FORMAT('-',A3,T6,E10.3,T17,E10.3,T28,E10.3,T39,E10.3,T50,E10.3,
     +       T61,E10.3,T72,E10.3,T83,E10.3,T94,E10.3)
    END IF
C
    RETURN
    END
C
C
    SUBROUTINE DISPL(NELEM,XDISP,XX,LEQNS)
    IMPLICIT REAL*8 (A-H,O-Z)
    DIMENSION XDISP(LEQNS),XX(3),XR(36)
    DIMENSION Q(3,36),NN(12)
    COMMON/ELCON/NELA(100),NELB(100),INTER
    COMMON/GLOB2/N1(200),N2(200),N3(200),N4(200),N5(200),N6(200),
   +              N7(200),N8(200)
```

```
      DO 50 I=1,INTER
         IF(NELEM .EQ. NELA(I)) THEN
           NELEM1=NELEM
           NELEM2=NELB(I)
         ELSE
           IF(NELEM .EQ. NELB(I)) THEN
             NELEM1=NELA(I)
             NELEM2=NELEM
           END IF
         END IF
   50 CONTINUE
      CALL REL(NELEM1,NELEM2,Q)
      NN(1)=N5(NELEM1)
      NN(2)=N1(NELEM1)
      NN(3)=N6(NELEM2)
      NN(4)=N7(NELEM2)
      NN(5)=N3(NELEM2)
      NN(6)=N8(NELEM1)
      NN(7)=N6(NELEM1)
      NN(8)=N2(NELEM1)
      NN(9)=N5(NELEM2)
      NN(10)=N8(NELEM2)
      NN(11)=N4(NELEM2)
      NN(12)=N7(NELEM1)
      DO 100 I=1,12
         LK=3*I-2
         LL=3*I-1
         LM=3*I
         XR(LK)=XDISP(3*NN(I)-2)
         XR(LL)=XDISP(3*NN(I)-1)
         XR(LM)=XDISP(3*NN(I))
  100 CONTINUE
      CALL MATVEC(3,36,Q,XR,XX,3,36)
C
      RETURN
      END
C
C
      SUBROUTINE YOUNG(EM,E,NU)
```

```
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 NU
      DIMENSION EM(6,6)
      DO 20 I=1,6
         DO 20 J=1,6
            EM(I,J)=0.D0
   20 CONTINUE
      COEF=E/((1.D0+NU)*(1.D0-2.D0*NU))
      EM(1,1)=COEF*(1.D0-NU)
      EM(1,2)=COEF*NU
      EM(1,3)=EM(1,2)
      EM(2,1)=EM(1,2)
      EM(2,2)=EM(1,1)
      EM(2,3)=EM(1,2)
      EM(3,1)=EM(1,3)
      EM(3,2)=EM(2,3)
      EM(3,3)=EM(1,1)
      EM(4,4)=E/(2.D0*(1.D0+NU))
      EM(5,5)=EM(4,4)
      EM(6,6)=EM(4,4)
C
      RETURN
      END
C
C
      SUBROUTINE MATVEC(N,M,A,Z,V,NEQNS,MEQNS)
      DIMENSION A(NEQNS,MEQNS),Z(MEQNS)
      REAL*8 SUM,A,V(N),Z
      DO 40 I=1,N
        SUM=0.D0
        DO 20 J=1,M
           SUM=SUM+A(I,J)*Z(J)
   20   CONTINUE
        V(I)=SUM
   40 CONTINUE
C
      RETURN
      END
C
```

```
C
      SUBROUTINE MATMAT(M,N,K,A,B,C)
      INTEGER M,N,K,R,S,I
      REAL*8 A(M,N),B(N,K),C(M,K),SUM
      R=1
      DO 30 WHILE(R. LE. M)
           S=1
           DO 20 WHILE(S .LE. K)
                SUM=0.DO
                I=1
                DO 10 WHILE(I .LE. N)
                     SUM=SUM+A(R,I)*B(I,S)
                     I=I+1
   10           CONTINUE
                C(R,S)=SUM
                S=S+1
   20      CONTINUE
           R=R+1
   30 CONTINUE
C
      RETURN
      END
C

C
      SUBROUTINE GREDUC(LEQNS,ASLOD,ASTIF,IFPRE,NEQNS)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION ASLOD(LEQNS),ASTIF(LEQNS,LEQNS),IFPRE(LEQNS)
      DO 50 IEQNS=1,NEQNS
         IF(IFPRE(IEQNS) .EQ. 1) THEN
           DO 40 IROWS=IEQNS,NEQNS
              ASTIF(IROWS,IEQNS)=O.DO
   40      CONTINUE
           GO TO 50
         END IF
         PIVOT=ASTIF(IEQNS,IEQNS)
         IF(DABS(PIVOT) .LE. 0.1E-8)THEN
           PRINT 100
  100      FORMAT('1',5X,'INCORRECT PIVOT')
           STOP
```

```
      END IF
      IF(IEQNS .EQ. NEQNS) THEN
        GO TO 50
      END IF
      IEQN1=IEQNS+1
      DO 20 IROWS=IEQN1,NEQNS
        FACTR=ASTIF(IROWS,IEQNS)/PIVOT
        IF(FACTR .EQ.0.0) THEN
          GO TO 20
        END IF
        DO 10 ICOLS=IEQNS,NEQNS
          ASTIF(IROWS,ICOLS)=ASTIF(IROWS,ICOLS)-FACTR*ASTIF(IEQNS,
     +                                                    ICOLS)
 10     CONTINUE
        ASLOD(IROWS)=ASLOD(IROWS)-FACTR*ASLOD(IEQNS)
 20   CONTINUE
 50   CONTINUE
C
      RETURN
      END
C
C
      SUBROUTINE BAKSUB(N,LEQNS,ASLOD,ASTIF,IFPRE,XDISP,REACT,NEQNS)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION ASLOD(LEQNS),ASTIF(LEQNS,LEQNS),IFPRE(LEQNS)
      DIMENSION XDISP(LEQNS),REACT(LEQNS)
      DO 5 IEQNS=1,NEQNS
        REACT(IEQNS)=0.DO
  5   CONTINUE
      NEQN1=NEQNS+1
      DO 30 IEQNS=1,NEQNS
        NBACK=NEQN1-IEQNS
        PIVOT=ASTIF(NBACK,NBACK)
        RESID=ASLOD(NBACK)
        IF(NBACK .EQ. NEQNS) THEN
          GO TO 20
        END IF
        NBAC1=NBACK+1
        DO 10 ICOLS=NBAC1,NEQNS
```

```
          RESID=RESID-ASTIF(NBACK,ICOLS)*XDISP(ICOLS)
   10.    CONTINUE
   20     IF(IFPRE(NBACK) .EQ. 0) THEN
            XDISP(NBACK)=RESID/PIVOT
          ELSE
            XDISP(NBACK)=0.DO
            REACT(NBACK)=-RESID
          END IF
   30 CONTINUE
      PRINT 200
  200 FORMAT('1 REACTIONS AND DISPLACEMENTS AT NODES')
      PRINT 220
  220 FORMAT('- NODE',T10,'X-REACTION',T24,'Y-REACTION',T38,'Z-REACTION'
     +        ,T63,'X-DISPL',T77,'Y-DISPL',T91,'Z-DISPL')
      PRINT 240,(K,REACT(3*K-2),REACT(3*K-1),REACT(3*K),XDISP(3*K-2),'
     +             XDISP(3*K-1),XDISP(3*K),K=1,N)
  240 FORMAT('-',I4,T10,E10.3,T24,E10.3,T38,E10.3,T61,E10.3,T75,E10.3,
     +        T89,E10.3)
C
      RETURN
      END
C
C
      SUBROUTINE CUBIC(C,ZR)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION C(4),ZR(3)
      COMPLEX*16 OR,R,Z1,Z2,Z3,Y1,Y2,Y3,X1,X2,X3,RCON
      A=C(2)/(3.DO*C(1))
      B=C(3)/(3.DO*C(1))
      G=C(4)/C(1)
      P=B-A*A
      Q=2.DO*A*A*A - 3.DO*A*B + G
      OR=Q*Q + 4.DO*P*P*P
      R=(-Q + CDSQRT(OR))/2.DO
      RCON=DCONJG(R)
      RPROD=R*RCON
      IF(RPROD .EQ. 0.0) THEN
        ZR(1)=-A
        ZR(2)=-A
```

```
        ZR(3)=-A
      ELSE
        Z1=CDEXP(CDLOG(R)/3.DO)
        Z2=(-Z1 + CDSQRT(-3.DO*Z1*Z1))/2.DO
        Z3=(-Z1 - CDSQRT(-3.DO*Z1*Z1))/2.DO
        Y1=Z1 - P/Z1
        Y2=Z2 - P/Z2
        Y3=Z3 - P/Z3
        X1=Y1 - A
        X2=Y2 - A
        X3=Y3 - A
        ZR(1)=X1
        ZR(2)=X2
        ZR(3)=X3
      END IF
C
      RETURN
C
C
      END
```