MIXED FINITE ELEMENT ANALYSIS WITH APPLICATION TO SPOT WELDING

A thesis submitted for the degree of
Doctor of Philosophy of
The University of Newcastle upon Tyne

by

Jian Wen

The Materials Division
Department of Mechanical, Materials and Manufacturing Engineering
The University of Newcastle upon Tyne

May 1994

PREFACE

This thesis describes original work which has not been submitted for a degree at any other university.

The investigations were carried out in the Materials Division, Department of Mechanical, Materials and Manufacturing Engineering, The University of Newcastle upon Tyne, United Kingdom, During the period of December 1990 to April 1994 under supervision of Dr. H. W. Chandler.

This thesis describes the establishment of a mixed FE formulation by first-order stress function. Its application to the elasticity, rigid-perfect-plasticity and elasto-plasticity problems as well as the simulation of part of the spot-welding process was also carried out.

This thesis is dedicated to

my father, mother, Wen Jin and Wen Yuan.

ACKNOWLEDGEMENT

Firstly, I would like to express my most sincere gratitude to my supervisor, Dr. H. W. Chandler, whose advise and encouragement throughout my PhD study were essential in helping to accomplish this thesis.

I would like to thank Dr. J. T. Evans who initially proposed a study into the FE simulation of spot-welding, and also my sponsor, Alcan Ltd, who financially made this project possible.

I would like to thank all the staff in Computing Laboratory, Newcastle University, who gave me valuable advice on using the computing facilities and also the staff in Wolfson Laboratory whose kindness and friendship gave me a very wonderful and memorable time during my PhD study.

Finally, I would thank my wife, Ningyun, for her patience and helpful advice throughout the past three and a half years.

ABSTRACT

A mixed finite element method is introduced in this thesis by two or three first-order C^0 stress functions for plane or axisymmetric problems respectively, which satisfy the force equilibrium equations, along with a constraint to impose the moment equilibrium equations. The stresses so expressed are equivalent to those in terms of the higher order Airy or Love stress function. With compatibility condition satisfied in the same way as in a displacement finite element (FE) method, the remaining constitutive relation in elasticity, i.e. Hooke's law, is satisfied by minimizing a mixed functional, with variables of the displacement vector and two/three first-order stress functions.

Some elementary problems in plane and axisymmetric elasticity are solved by this method. It is found that for an incompressible solid and a solid with a crack, the mixed model yields better results than the conventional FE method. The effects of Gaussian integration and Poisson's ratio on the solution are discussed in detail. Special attention is paid in bending a beam and a disc, where the importance of the constraint to enforce moment equilibrium is studied.

For rigid-perfect-plasticity, the Levy-Mises flow rule and the corresponding yield condition are satisfied by another extremum principle. By substituting the plastic part of the elasto-plastic strain into the extremum for rigid plasticity, and the elastic part of the elasto-plastic strain into the extremum for elasticity, an extremum principle for elasto-plasticity is established straightaway. Applications of this method to some well-known examples are discussed. In comparison with the conventional displacement method and/or analytical solution, this method offers very satisfactory results and good convergence of the solution.

An interesting feature of this method is that the value of each functional indicates in some degree the solution error at a giving point or region. This may provide useful information for accuracy control or a remeshing procedure.

A more sophisticated problem is solved by a so-called mixed fluid-FE model, which is the simulation of the flow of an adhesive between two aluminium sheets squeezed by a pair of electrodes in spot-process. The effects of various factors on the formation of the entrapment of the adhesive in the central area of faying surface are studied in detail. Very close results between displacement method and the mixed method are obtained in this study.

CONTENTS

CHAPTER 1
INTRODUCTION

The finite element method offers a way of interpolating functions in space. In this thesis the functions used are stress functions (to be defined later) and the components of a displacement vector. Their formulation allows mixed extremum principles to be used to find numerical solutions to a number of problems in solid mechanics. The validity of the algorithms produced will be checked by a few test cases in plane elasticity. By carefully choosing three stress functions, the same idea will be applied to axisymmetric elasticity.

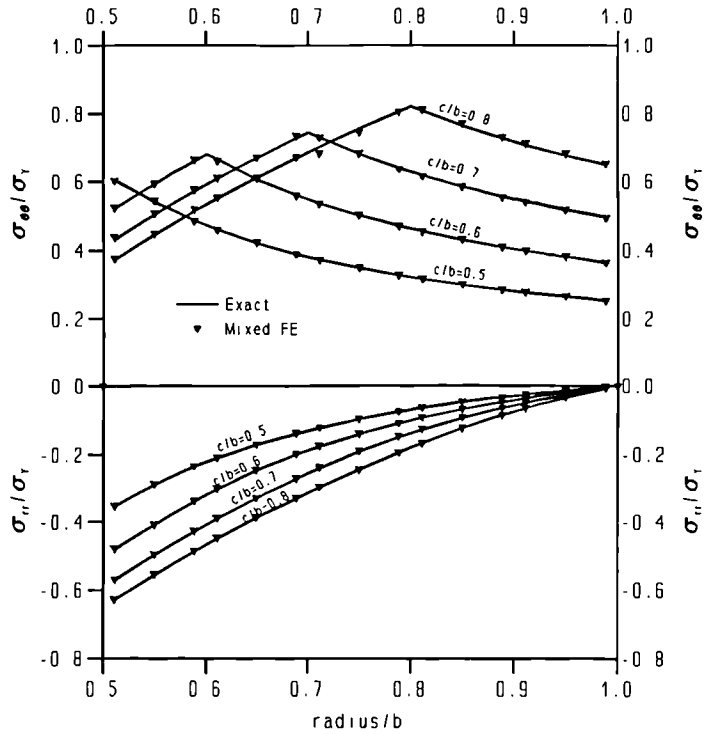


Fig. 1.1 Stress distributions in an elasto-perfectly plastic cylinder under internal pressure.

The principal objective of this thesis is to establish a mixed-extremum principle for plane and axisymmetric elasto-plasticity, based on the discretization of stresses tested in elasticity problems and an extremum principle for rigid-perfect-plasticity. This formulation is expected to give better results than a conventional displacement method for the problems with a large scale of plasticity, e.g. an elasto-perfectly plastic cylinder under internal pressure. The power of this method is demonstrated by the excellent

agreement between the finite element results and the theoretical results shown in fig.1.1 (discussed in detail in **Chapter six**). A more complex problem is demonstrated in the simulation of part of the spot-welding process by this mixed model.

1.1 GENERAL

The finite element (FE) method was originated by engineers in the 1950s to analyse aircraft structural systems using the emerging digital computer, where naturally discrete systems such as structural frameworks are presented. Turner et al.[116] (1956) published the first paper, followed by Argyris [4] (1960) and Clough [24] (1960), who first time in history used the name ***finite elements***. As the finite element method applied to structural problem matured, the concept of a "force balance" at a node was replaced by theory founded in the variational calculus and the classical Rayleigh-Ritz method (Rayleigh[90], 1877; Ritz[96], 1909). These are known as ***variational approaches***. Application of the finite element method to non-structural problems was reported by Zienkiewicz and Cheung[128](1965).

Later the establishment of the finite element method was extended on a broader base in terms of the ***weighted residual approaches***[31], a term first used by Crandall[25](1956). Since then, the finite element method has become one of the most significant tools in the history of computational methods. Modern finite element packages allow large areas of theoretical physics to be put to practical use. Typical problems are heat transfer, fluid mechanics, electromagnetism, geo-mechanics and acoustics.

1.2 SINGLE-FIELD AND MULTI-FIELD FINITE ELEMENT MODELS

If a problem in the 2D theory of elasticity (plane strain) is investigated there are four components of stress, three components of strain and two components of displacement. In order to solve such a problem one needs to take account of the two equations of equilibrium and the three strain-displacement relations. It is usual however to reduce the number of variables at the outset to just the displacements by eliminating stress and

strain. This is known as reducing the problem to a ***single-field*** formulation.

In fact, this formulation in terms of displacement, or ***displacement model***, is the most successful model and is used extensively in practical applications and in the development of commercial structural analysis packages. This provides the simplest approach to element formulation and to the assembly of the element equations into the global system. However the displacement model has some shortcomings[77]:

- Difficulty in constructing compatible shape functions **N** for C^1 elements, (e.g., plate and shell elements based on Kirchhoff theory)
- Poor performance in constrained media problems, (e.g., incompressible and nearly incompressible materials)
- Loss of accuracy in calculating secondary fields (e.g., stress calculations using displacement models)
- convergence of a solution can not be guaranteed.

As the understanding of finite element method deepened, alternative models, ***multi-field*** models, in which there are more than one unknown field, have been proposed in the hope of overcoming such shortcomings. The hybrid models and mixed models are two kinds of such multi-field models.

1.2.1 Hybrid finite element formulation

Hybrid models are multi-field models in which the formulation is based on a modified variational principle with relaxed interelement requirements. In the modified principles some or all of the interelement requirements are treated as constraints and are enforced in an average sense by means of Lagrange multipliers, which will be continuous functions of the boundary coordinates. Therefore, the formulation involves independent approximation of the fields variables within the element and the Lagrange multipliers along the element boundaries.

In solid mechanical problems, hybrid finite element models are obtained by relaxing the condition of displacement continuity or traction reciprocity along the interelement boundaries. It was first introduced by Pian[83] in 1964 for plane elasticity problems.

Since then, hybrid models have been successfully applied to many problems in solid and fluid mechanics, such as in plate bending[111], laminated materials[67] and fracture mechanics[6,112]. The most favourable character of these models is that the elements so built can be incorporated into any existing package with the displacement model. However this model usually involves many arithmetic operations including the inverse of a matrix at the element level.

1.2.2 Mixed finite element formulation

On the other hand, in ***mixed models***, the element parameters contain more than one field, and unlike hybrid models all the field variables are described within the element as well as on its boundary. It is usually founded on mixed variational principles, such as Hu-Washizu principle[44,120] or Hellinger-Reissner principle[39,93], or their modifications in which two or three fields are included. Two types of mixed models may be distinguished:

a. ***Discontinuous models***

   Here the continuity of only one field is enforced across element boundaries, the parameters used in approximating the other fields can be discontinuous and eliminated at the element level, and the size of the system analysis is the same as that in a single-field solution. Again, similar to a hybrid model, many arithmetic operations including inverse of a matrix at the element level must be done.

b. ***Continuous models***

   If the continuity of all field variables is enforced across element boundaries, the size of the system analysis is expanded well beyond that in a conventional single-field solution. Therefore a very efficient approximation of all fields in this model is required to reduce the variables to as few as possible.

1.2.3 Comments on multi-field models

The multi-field models possess several major advantages over single-field models.

Because the governing equations of the mixed models are the more basic equations of mechanics (or physics) with lower-order derivatives, the interelement continuity requirements on the assumed fields are of lower order than for the conventional single-field models. Furthermore the development of the element matrices in FE formulations for multi-field models involves fewer arithmetic operations than that of their equivalent single-field models. It is also true that for some nonlinear problems multi-field models lead to a simpler problem to solve than those obtained by a single-field model.

Multi-field models can directly yield quantities of practical interest thus saving time for further calculations, such as the stress computations in solid mechanics and evaluation of stress intensity factors in fracture mechanics. Multi-field models of the combination of mixed and hybrid (mixed-hybrid) types provide a convenient way to match approximate solutions for different regions which make them most suited for analysing nonhomogeneous media with discontinuous properties.

Insensitivity regarding the variations in the structure dimensions and the material properties is an important factor for a finite element method. It is one of the difficulties inherent in the single-field model, where special precautions have to be taken to assure the accuracy of the answer. The multi-field models are considerably less sensitive to variations in such characteristics of the structure than are the single field (displacement) models. For example, in solid mechanics, the performance of the displacement model for nearly incompressible materials give problems. The application of shear-flexible plate (or shell) elements to analyze thin plates (or shells) with negligible shear deformation can cause "locking" problems.

Some techniques are used to overcome these problems in a single-field method. For example a reduced integration scheme is used to solve incompressible materials, while an extra constraint is introduced in the virtual work principle for plate (or shell) to prevent "locking". As Zienkiewicz pointed out in ref [126], these are all equivalent to corresponding multi-field models.

While there are a number of advantages in multi-field finite element models, difficulties of the mixed variational principles arise from the fact that these principle are not extremum principles, but stationary principles. Unlike in the single-field model where the system matrix is positive definite, the final equations derived from mixed variational principles include a non-positive-definite matrix. Therefore some efficient

algorithms used with the single-field models cannot be applied.

For the mixed continuous models, the increase in the number of degrees of freedom results in a substantial increase in the size of the element matrices, and in the number of simultaneous equations, as compared with the single-field models. It is also found that some of the hybrid models and the mixed discontinuous models contain spurious mechanisms (associated with zero eigenvalues).

1.3 SCOPE OF THIS THESIS

The main effort of the three years work by the author was trying to develop finite element formulations from very elementary principles, e.g. Airy and Love stress function, Hooke's law, von Mises flow rule and the **Least squares method**(to be defined later). It in turn provides a better understanding of all kinds of relationships in elasticity, rigid perfectly-plasticity and elasto-plasticity.

This thesis treats the establishment and application of some mixed finite element models in solid mechanics. Three new mixed models based on corresponding ***mixed extremum principles***, suitable for elasticity, rigid plasticity or elasto-plasticity respectively, will be presented. In each case, some elementary problems are studied along with conventional FE method and/or analytical method as bench-mark tests. A more complicated application of the mixed model for elasto-plasticity is dealt with in the mechanical simulation of part of a resistance spot-welding process.

The components of the stress will be interpreted in terms of two little known C^0 constrained stress functions in plane problems, mentioned by Brezzi's work[12]. For axi-symmetric problems, a novel interpretation of stresses with three C^0 stress functions will be conceived by author. The well-known isoparametric shape function is used for discretizing both the displacement components and stress functions. 4 variables per node are needed in 2D cases, while 5 are needed for axisymmetric cases.

In **Chapter two** of this thesis a historic review of the mixed finite element formulation will be given. All the basic formulations in the mixed FE method, such as the discretization of the displacement field and stress field as well as an extremum principle, are introduced in **Chapter three** with plane elasticity. The axisymmetry elasticity will be studied in **Chapter four** with the new description of the stress field.

Applications to simulation of spot-welding along with some simple solutions are carried out. Nonlinear problems are first introduced in **Chapter five** with rigid-perfect plasticity. There an extremum principle will be proposed to impose Levy-Mises flow rule and yield condition. An optimum method, the Golden Section Search, will be utilized for solving the nonlinear equations. The extremum principle is further developed for elasto-plasticity in **Chapter six**. Some basic elasto-plastic problems are solved to demonstrate the efficiency of this model. A practical application of this method will be shown in the mechanical simulation of part of a spot welding process with an adhesive interlayer in **Chapter seven**. Finally, **Chapter eight** will give the final remarks about these models.

For the convenience of future reference, some basic concepts in FE method and primary knowledge for the construction of its formulations, such as about discretization, variational principles and their modifications, will be introduced in the next section of this chapter.

1.4 APPROACHES TO CONSTRUCT FE FORMULATIONS

In this section, a brief description of approaches to establish a finite element formulation in solid mechanics will be given, where emphases are on the discretization of the basic field(s) and some approaches to create the fundamental principles, on which the finite element formulations are based, will also be reviewed.

1.4.1 Finite Element Discretization

Basically the finite element method solves a problem starting with ***discretization*** of the original domain. We will seek an unknown function $\mathbf{u}$ such that it satisfies a differential equation

$$\mathbf{P}(\mathbf{u}) = \mathbf{0} \tag{1.1}$$

in a domain Ω (fig.1.2), together with certain boundary conditions

$$\mathbf{p}(\mathbf{u}) = \mathbf{0} \tag{1.2}$$

on the boundaries Γ of the domain (fig.1.2). Here $\mathbf{P}$ and $\mathbf{p}$ are operators defining

governing differential equations and boundary conditions.

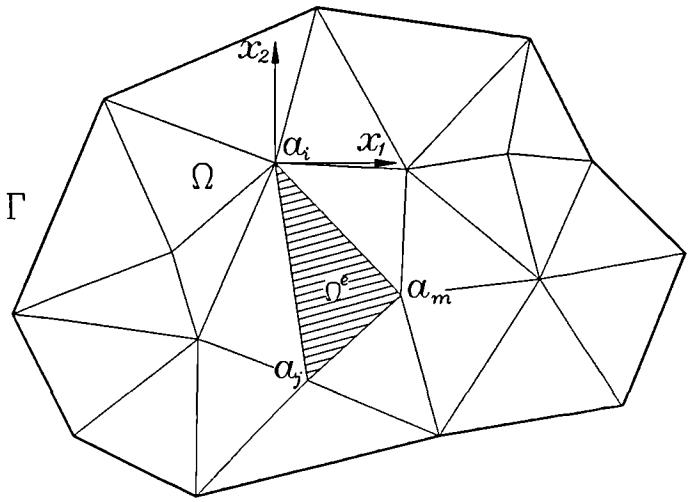


Fig. 1.2 The discretization of domain Ω by finite element mesh.

If we divide the whole domain Ω by a number of subdivisions Ω^e (fig.1.2), known as "elements", then within each element we can approximate the unknown function $\mathbf{u}$ by a much simpler function $\mathbf{u}^*$ defined by a few unknown parameters a_i within each element

$$\mathbf{u} \approx \mathbf{u}^* = \sum_{i=1}^{n} N_i a_i = \mathbf{N}\mathbf{a} \tag{1.3}$$

where N_i is an interpolation function——the ***shape function*** prescribed locally for the element in terms of independent variables (such as the coordinates x, y, etc.). Usually we define a_i as the values of $\mathbf{u}$ at the nodes defining the geometry of the element. Therefore a_i are often called the ***nodal variables***. So far we have simplified the original problem of finding a continuous unknown function $\mathbf{u}$ into a problem of finding a number of nodal variables a_i element by element, this process is termed as ***discretization***.

However if the differential equation (1.1) plus boundary equation (1.2) can be equivalent to an integral form, the approach of which will be discussed in section 1.4.2,

$$\int_\Omega \mathbf{G}(\mathbf{u}^*)d\Omega + \int_\Gamma \mathbf{g}(\mathbf{u}^*)d\Gamma = 0 \tag{1.4}$$

in which **G** and **g** are known operators, by the merits of integration we have,

$$\sum_{e=1}^{m}\left[\int_{\Omega^e}\mathbf{G}(\mathbf{u}^*)d\Omega + \int_{\Gamma^e}\mathbf{g}(\mathbf{u}^*)d\Gamma\right] = 0 \tag{1.5}$$

A useful example arises from linear elasticity in solid mechanics where the differential equations are linear, i.e.

$$\mathbf{P}(\mathbf{u}) \equiv \mathbf{L}\,\mathbf{u} + \mathbf{l}$$

$$\mathbf{p}(\mathbf{u}) \equiv \mathbf{M}\,\mathbf{u} + \mathbf{m} \tag{1.6}$$

where **L**, **M**, **l** and **m** are constant operators. Eq.(1.5) will then yield a set of linear equations of the form

$$\mathbf{K}\,\mathbf{a} + \mathbf{f} = 0 \tag{1.7}$$

with

$$K_{ij} = \sum_{e=1}^{m} K_{ij}^{e} \quad ; \quad f_i = \sum_{e=1}^{m} f_i^{e} \tag{1.8}$$

where K_{ij} and f_i are two constant coefficient matrices.

1.4.2 Weighted residual approaches and variational approaches

Creation of detailed integral formulations of the type shown in eq.(1.4) is the most critical step in establishing the finite element formulation. Two distinct procedures are available for obtaining such integral forms. The first is the ***weighted residual approach***[31], the second is the ***variational approach***[71,70].

As the differential equations (1.1) and (1.2) have to be satisfied at each point of the domain Ω, it follows that

$$\int_{\Omega}\mathbf{v}^{\mathbf{T}}\mathbf{P}(\mathbf{u})d\Omega \equiv 0 \quad ; \quad \int_{\Gamma}\bar{\mathbf{v}}^{\mathbf{T}}\mathbf{p}(\mathbf{u})d\Gamma \equiv 0 \tag{1.9}$$

where $\mathbf{v}$ and $\bar{\mathbf{v}}$ are any functions. If the above statement is valid for the reverse case, i.e., if eq.(1.9) is always true for arbitrary functions $\mathbf{v}$ and $\bar{\mathbf{v}}$, the differential equation (1.1) and (1.2) must be satisfied at all points of the domain. Indeed, the integral

statement that

$$\int_{\Omega} \mathbf{v}^{\mathbf{T}} \mathbf{P}(\mathbf{u}) d\Omega + \int_{\Gamma} \bar{\mathbf{v}}^{\mathbf{T}} \mathbf{p}(\mathbf{u}) d\Gamma = 0 \tag{1.10}$$

is satisfied for all $\mathbf{v}$ and $\bar{\mathbf{v}}$ is equivalent to the satisfaction of the differential equation (1.1) and its boundary equation (1.2)[31]. Therefore we have achieved the construction of the integral form equivalent to the differential equation, which is just what the finite element formulation needs. Because such a statement can be intuitively interpreted as the requirement of inbalance of differential equations (1.1) and (1.2) to be zero in a weighted average sense, the procedure based on eq.(1.10) is named as the ***weighted residual approach***, and both $\mathbf{v}$ and $\bar{\mathbf{v}}$ are called the weight functions. Obviously this method is valid for any continuum problem for which differential equations are well-defined. The well-known ***Galerkin method*** is only a special case in which $\mathbf{v}$ is chosen to be identical to the shape functions $\mathbf{N}$.

On the other hand, the variational approach is based on the existence of a ***variational principle***: the solution to a continuum problem is a function $\mathbf{u}$ which makes a functional Π stationary with respect to a small variation $\delta\mathbf{u}$, i.e.

$$\delta\Pi = 0 \tag{1.11}$$

where Π is defined by an integral form

$$\Pi = \int_{\Omega} H(\mathbf{u}, \frac{\partial}{\partial x}\mathbf{u}, \ldots) d\Omega + \int_{\Gamma} h(\mathbf{u}, \frac{\partial}{\partial x}\mathbf{u}, \ldots) d\Gamma \tag{1.12}$$

where $\mathbf{H}$ and $\mathbf{h}$ are known operators. If a variational principle can be found, an approximate method is immediately established for obtaining solutions in the standard, integral form suitable for finite element formulation.

From eq.(1.3), eq.(1.11) can be further expressed in terms of nodal variables a_i as.

$$\delta\Pi = \frac{\partial \Pi}{\partial a_1}\delta a_1 + \frac{\partial \Pi}{\partial a_2}\delta a_2 + \cdots + \frac{\partial \Pi}{\partial a_n}\delta a_n = 0 \tag{1.13}$$

This being true for any variations $\delta\mathbf{a}$ yields a set of equations

$$\frac{\partial \Pi}{\partial \mathbf{a}} = \left[\frac{\partial \Pi}{\partial a_1} \quad \cdots \quad \frac{\partial \Pi}{\partial a_n} \right]^T = 0 \tag{1.14}$$

from which parameters a_i are found. The famous Rayleigh-Ritz[90,96] method falls into this category.

Some "natural" variational principles exist in the physical world such as minimization of total potential energy to achieve equilibrium in mechanical systems, the least energy dissipation principle in slow viscous flow, etc. Unfortunately they do not exist for all continuum problems for which well-defined differential equation may be formulated. A thorough study for deriving natural variational principles from linear differential equations is available in the works of Mikhlin[71,70]

1.4.3 Constrained variational principles: Lagrange multiplier methods

Consider the problem of making a functional Π stationary, subject to the unknown $\mathbf{u}$ obeying some set of additional differential equations as constraints

$$\mathbf{Q}(\mathbf{u}) = \mathbf{0} \quad \text{in } \Omega \qquad ; \qquad \mathbf{q}(\mathbf{u}) = \mathbf{0} \quad \text{in } \Gamma \tag{1.15}$$

we can introduce this constraint by forming another functional

$$\Pi_L(\mathbf{u},\lambda,\bar{\lambda}) = \Pi(\mathbf{u}) + \int_\Omega \lambda^T \mathbf{Q}(\mathbf{u}) d\Omega + \int_\Gamma \bar{\lambda}^T \mathbf{q}(\mathbf{u}) d\Gamma \tag{1.16}$$

in which λ and $\bar{\lambda}$ are some set of functions of the independent coordinates in a domain Ω and boundary Γ respectively, known as Lagrange multipliers. The variation of the new functional is now

$$\begin{aligned}\delta\Pi_L(\mathbf{u},\lambda,\bar{\lambda}) = \delta\Pi(\mathbf{u}) &+ \int_\Omega \delta\lambda^T \mathbf{Q}(\mathbf{u}) d\Omega + \int_\Gamma \delta\bar{\lambda}^T \mathbf{q}(\mathbf{u}) d\Gamma \\ &+ \int_\Omega \lambda^T \delta\mathbf{Q}(\mathbf{u}) d\Omega + \int_\Gamma \bar{\lambda}^T \delta\mathbf{q}(\mathbf{u}) d\Gamma\end{aligned} \tag{1.17}$$

And this is zero providing $\mathbf{Q}(\mathbf{u}) = \mathbf{0}$ and $\mathbf{q}(\mathbf{u}) = \mathbf{0}$, and simultaneously $\delta\Pi = 0$. Eq.(1.17) is known as the ***modified variational principle***. The most general variational principle in solid mechanics, the Hu-Washizu principle[44,120], was created using such an approach.

Naturally, we may use the governing differential equation (1.1) and boundary condition eq.(1.2) themselves as "constraints", i.e. $\mathbf{P}(\mathbf{u}) = \mathbf{Q}(\mathbf{u})$ in domain Ω and

$\mathbf{p}(\mathbf{u}) = \mathbf{q}(\mathbf{u})$ on boundary Γ, and leave $\delta\Pi = 0$. It gives

$$\begin{aligned}\delta\Pi_L(\mathbf{u},\lambda,\bar{\lambda}) = &\int_\Omega \delta\lambda^T \mathbf{P}(\mathbf{u})d\Omega + \int_\Gamma \delta\bar{\lambda}^T \mathbf{p}(\mathbf{u})d\Gamma \\ &+ \int_\Omega \lambda^T \delta\mathbf{P}(\mathbf{u})\,d\Omega + \int_\Gamma \bar{\lambda}^T \delta\mathbf{p}(\mathbf{u})d\Gamma\end{aligned} \tag{1.18}$$

It is worthwhile to mention that the new principle so obtained has more variables ($\mathbf{u}$, $\delta\lambda$ and $\delta\bar{\lambda}$) than before, and the zero diagonal will exist in the final matrix, which means it is no longer positive-definite.

One interesting case is that when $\mathbf{P}$ and $\mathbf{p}$ are linear operators as defined in eq.(1.6), where $\delta\mathbf{P}$ and $\delta\mathbf{p}$ are no longer the functions of $\mathbf{u}$, eq.(1.18) can be totally decoupled as two equations

$$\begin{aligned}\delta\Pi_L(\mathbf{u}) &= \int_\Omega \delta\lambda^T \mathbf{P}(\mathbf{u})d\Omega + \int_\Gamma \delta\bar{\lambda}^T \mathbf{p}(\mathbf{u})d\Gamma = 0 \\ \delta\Pi_L(\lambda,\bar{\lambda}) &= \int_\Omega \lambda^T \delta\mathbf{P}\,d\Omega + \int_\Gamma \bar{\lambda}^T \delta\mathbf{p}\,d\Gamma = 0\end{aligned} \tag{1.19}$$

The second equation in eq.(1.19) only includes Lagrange multipliers λ and $\bar{\lambda}$. Therefore the λ and $\bar{\lambda}$ can be obtained independent of variables $\mathbf{u}$. The first equation in it has variables $\mathbf{u}$ with some known function λ and $\bar{\lambda}$. With comparison to eq.(1.10) on which weighted residual approaches are based, it is easy to see they are identical if arbitrary functions $\mathbf{v}$ and $\bar{\mathbf{v}}$ being replaced by $\delta\lambda$ and $\delta\bar{\lambda}$. However, unlike $\mathbf{v}$ and $\bar{\mathbf{v}}$ can be any function in weighted residual approaches, $\delta\lambda$ and $\delta\bar{\lambda}$ are governed by the second equation in eq.(1.19) in Lagrange multiplier methods.

1.4.4 Constrained variational principles: Penalty function methods

In section 1.4.3, we found the constrained variational principle can be obtained by Lagrange multiplier method at the expense of increasing the total number of unknowns. In addition, even in linear problems, the algebraic equations are now complicated by having zero diagonal terms. In this section we shall consider another technique to introduce constraints which does not have such drawbacks.

Consider once again the problem of obtaining stationary of Π with a set of constraint equations $\mathbf{Q}(\mathbf{u}) = 0$ in domain Ω and $\mathbf{q}(\mathbf{u}) = 0$ on boundary Γ. We note

that the products

$$\begin{aligned} \mathbf{Q}^{\mathbf{T}}\mathbf{Q} &= Q_1^2 + Q_2^2 + Q_3^2 + \cdots \\ \mathbf{q}^{\mathbf{T}}\mathbf{q} &= q_1^2 + q_2^2 + q_3^2 + \cdots \end{aligned} \tag{1.20}$$

where

$$\begin{aligned} \mathbf{Q}^{\mathbf{T}} &= [Q_1 \;\; Q_2 \;\; Q_3 \;\; \cdots] \\ \mathbf{q}^{\mathbf{T}} &= [q_1 \;\; q_2 \;\; q_3 \;\; \cdots] \end{aligned}$$

have a minimum of zero when all the constraints are satisfied. We can immediately write a new functional

$$\Pi_P(\mathbf{u}) = \Pi(\mathbf{u}) + \alpha \int_\Omega \mathbf{Q}^{\mathbf{T}}(\mathbf{u})\mathbf{Q}(\mathbf{u}) d\Omega + \bar{\alpha} \int_\Gamma \mathbf{q}^{\mathbf{T}}(\mathbf{u})\mathbf{q}(\mathbf{u}) d\Gamma \tag{1.21}$$

in which α and $\bar{\alpha}$ are *penalty numbers*. If the minimum of Π is the solution then the penalty numbers should be positive number. The solution obtained by the extremum of the functional Π_P will satisfy the constraints only approximately. The larger the value of α and $\bar{\alpha}$ are, the better will the constraints be satisfied. Obviously, it does not increase the unknown functions $\mathbf{u}$, nor will it yield the zero diagonal terms.

Similar to the previous section, if the constraints become simply the governing equation (1.1) and boundary condition (1.2) of the problem, i.e. $\mathbf{P(u)=Q(u)}$ in Ω and $\mathbf{p(u)=q(u)}$ at Γ, we can write a variational principle

$$\Pi_P = \alpha \int_\Omega \mathbf{P}^{\mathbf{T}}(\mathbf{u})\,\mathbf{P}(\mathbf{u}) d\Omega + \bar{\alpha} \int_\Gamma \mathbf{p}^{\mathbf{T}}(\mathbf{u})\,\mathbf{p}(\mathbf{u}) d\Gamma \tag{1.22}$$

for any set of differential equations.

Clearly equation (1.22) is simply a requirement that the sum of squares of the errors of the differential equations (1.1) and boundary conditions (1.2) should be a minimum at the true solution. This is the well-known ***Least squares method*** of approximation. The least squares method is a very powerful alternative procedure for obtaining integral forms and have been used with considerable success[63,127].

CHAPTER 2
LITERATURE REVIEW

Mixed finite element models were initially developed in the 1960s to avoid the difficulty in constructing a compatible displacement field for plate-bending problems which are governed by fourth order differential equations. In 1966 Herrmann[40] developed the first mixed finite element model for the linear static bending analysis of thin plates. In this chapter, we will give a historic review of the development of mixed FE models.

2.1 FUNDAMENTAL PRINCIPLES

In general, there are three approaches for formulating the mixed finite element models (or mixed models for simplicity) which are the ***direct method***, the ***variational approach*** and the ***weighted residual approach***.

2.1.1 Direct method

The ***direct method*** for formulating a mixed model combines the fundamental equations in their natural or primitive form for each element directly. In structural and solid mechanics problems, the fundamental equations include equations of motion (or equilibrium), constitutive relations, and strain-displacement relations. This approach is useful for simple elements such as truss and beam elements in structural mechanics. Some of the early applications of mixed models in structural mechanics were made by Klein[54] by such methods. However, it becomes difficult to apply to more complex elements and will not be discussed further.

2.1.2 Variational Approach

The ***variational approach*** to formulating mixed models is by far the most commonly used to date. Almost all the reported applications of the variational approach in the solid mechanics field are based on the two field Hellinger-Reissner[39,89,93] or three-field Hu-Washizu[44,120] variational principle, or their modifications.

Hellinger presented in 1914[39] a two-field mixed variational principle for finite elasticity problems, in which stresses and displacements are varied independently. For linear elasticity problems, Prange[89] modified Hellinger's principle by including the boundary conditions. In 1950 Reissner presented[93] an equivalent variational theorem for linear elasticity with independent displacement and stress fields, which includes the boundary conditions. This variational principle is the famous Hellinger-Reissner (H-R) principle. There are no subsidiary conditions needed for the stress components and displacement vector. It later became one of the most popular principles for mixed models.

Hu, on the other hand, presented in 1955[44] a three-field mixed variational theorem, which includes independent displacement, stress and strain fields as well as all the boundary conditions. The complete fifteen equations in elasticity[110] are the Euler equations of this variational principle. A similar theorem was also reported independently by Washizu[120] (1955). This variational principle is the well-known Hu-Washizu (H-W) principle, which is another basic principle for the mixed models.

The earlier applications of mixed variational principle date back to the mid-1960s when the use of the mixed models for plate-bending was proposed, independently, by Herrmann[40](1966) and by Hellan[38](1967). Herrmann used an alternative form of H-R principle for bending of the thin plates by relaxing the constraint conditions of C^1 continuity for normal displacement. Dunham and Pister[30](1968) used the classic H-R variational principle to produce mixed finite elements for plane elasticity and plate bending problem.

During the following decades, the mixed models based on the H-R principle or the H-W principle were successfully applied in almost every area where the conventional displacement methods had been applied. The mixed models do show superior results over those obtained by a connventional displacement model in constrained media problems, e.g. incompressible and nearly incompressible materials[8,117], for problems with high gradients (or singularities) in strains or stresses, such as near sharp crack[78] or for laminated composites[76,84,123,23], and highly inelasticity such as rubber-like materials[34,117].

Chouchaoui and Shirazi-Adl (1992) in their work[23] reproted a mixed formulation based on the H-R variational principle to analyse composite structures. It demonstrated

to have a satisfactory degree of accuracy, especially in view of the observed inability of the conventional compatible method to yield reliable stress results.

The potential advantages of the mixed models in contact problems are realised by many scientists working in this area[102,82,21]. First, because both displacements and stress are retained as variables, both the kinematic and equilibrium conditions on a region of contact can be incorporated explicitly. Secondly, a study made by Brezzi(1991) et al in ref. [12] indicated a superior accuracy over the displacement method in analysing certain stress concentration problems.

For handling the contact boundary conditions, Tseng and Olsen(1981)[115] modified Hellinger-Reissner principle by adding an integration term, which acts as a Lagrange multiplier term. The displacements and stresses boundaries are explicitly included in the functional. In a progressive contact problem which is frictionless and between two elastic components, it shows better results in predicting the contact area than those from displacement method.

More recently, Cescotto and Charlier[21](1993) applied a modified Hu-Washizu principle to friction contact problems. It was first presented in the framework of infinitesimal deformations and subsequently was extended to large inelastic strains.

For soil-structure interaction and construction sequence problems, Desai and Lightner(1985)[28] proposed a mixed finite element procedure with triangular and quadrilateral elements to nonlinear elastic and elastoplastic analysis. A number of special techniques such as symbolic programming, utilization of dihedral symmetry and the frontal solution method were employed to achieve computational economy. Again, the classic H-R principle was used for the formation of the global equations, which causes matrices to be non-positive definite. A number of examples in plane problems were presented which gave the stresses to high accuracy. In a multi-media problem where discontinuous stress exists this mixed finite element model gives reasonable results.

For large displacement motion of solid bodies, a mixed updated Lagrangian formulation was derived by Liao and Tsai(1992)[59], based on the incremental nonlinear theory of solid and H-R principle. Two quadrilateral isoparametric mixed elements were used in the study with continuous stresses across the elements. In the linear analysis of plane problems, the mixed model can obtain accurate displacements and

stresses, but in nonlinear analysis, the predicted stresses do not show monotonic convergence.

The applications of mixed models to axisymmetric problem and 3D problems are quite uncommon. Mahapatra(1988)[64] derived the Hellinger-Reissner expression for axisymmetric statement for both elastic and elasto-plastic material behaviour. A thick cylinder under internal pressure and the Boussinesq point load problem were used to demonstrate the efficiency of this model. This model was further extended to some axisymmetric geomechanic problems in 1991[65].

A three-field mixed formulation in terms of displacements, stresses and an "enhanced" strain field, based on a modified Hu-Washizu principle, was presented by Simo and Rifai(1990)[105]. Because there is no subsidiary condition of strains needed, they chose a so-called "enhanced" strain which is merely a superposition of the conventional compatible strain and a incompatible "enhanced" strain. They had demonstrated how the classic method of incompatible modes arise merely as a result of particular interpolations within a certain class of mixed methods. Analysis of Cook's membrane and the problem of a clamped arch in elasto-plasticity shows that this model possesses very good convergence and accuracy in the calculation of displacements and strains. The method was later extended to include geometrically nonlinear by Simo et al (1992) in ref.[106].

In 1991, the analysis of elastoplastic large strains was reported by Jetteur and Cescotto[48]. It is also based on the Hu-Washizu principle and uses a co-rotational formulation for describing large strain. A 4-node element with 1 integration point was used in the formation of the matrices. Few results were given.

2.1.3 Weighted Residual Approach

The ***weighted residual approaches***, on the other hand, has a broader scope than either of the direct approaches or variational approaches, though for some problems, the Galerkin process of the weighted residual approaches yield results which are identical to those obtained by variational approaches. As mentioned in section 1.4.2, they do not require the existence of a variational principle and therefore can be easily applied to non-structural mechanics.

Most of the applications of the mixed finite element models based on the weighted residual approach have been in the fluid mechanics or solid-fluid interaction problems (coupling problems) where variational principles are difficult to construct. Meissner[68] (1973) used a mixed model in potential flow problems. Hutton et al[45](1980) and Jackson[46] (1981) applied a mixed method on the solution of viscous incompressible flow. Silvester[103](1990) and Jenson[47](1992) discussed the Navier-stokes equation by a mixed model. Applications of the weighted residual approaches to steady fluid and metal flow problems[125](Yamada et al, 1975) and soil-structure interaction problems were reported by Lightner[60](1981) and Sandhu[97](1990). A discontinuous mixed FE model in solving two-phase incompressible flow was presented by Chavent[20] et al (1990).

In 1990, Spilker and Maxian[109] reported a mixed-penalty FE formulation of the linear biphasic theory for soft tissues, which was represented by a continuum theory of mixtures involving intrinsically incompressible solid and incompressible inviscid fluid phases. In this formulation, the continuity equation of the mixture was replaced by a penalty form of this equation which was introduced along with the momentum equation and mechanical boundary condition for each phase into a weighted residual form.

Heyliger and Kriz[41](1989) used a "variational formulation" in solving crack problems, where the analytical solutions of the displacement and stress field near crack tip were added to the conventional displacement and stress terms. The stress intensity factors are included in the general variables. However, the procedure for obtaining the formulation is exactly the same as the weighted residual approaches, where the weight functions were taken as the first variations of the displacement components and the stress components.

Loula et al[61](1990) solved steady-state creep problems with monotone constitutive laws. Finite element approximations are constructed based on mixed Petrov-Galerkin formulations for constrained problems. Numerical results are presented confirming the convergence of predicted by the Babuska-Brezzi criterion[12].

In solid mechanics, most of the reported applications of the weighted residual approaches have used either Galerkin or a least-square approach.

2.1.4 Other Approaches

So far, however, in this review all the principles adopted in this particular area of the finite element method were fallen into two main categories as already described in section 1.4.2. With the mixed variational principles, based on Hu-Washizu or Hellinger-Reissner principles, the applications to more sophisticated situations use extra terms added to the already complex system. With the weighted residual approach the governing equations are normally chosen residual form, which are usually high-order differential equations.

Sohn and Heinrich[107](1990) proposed a penalty finite element models to solve viscous incompressible flows, where the second derivatives appearing in the weak formulation of the Poisson equation are calculated from C^0 velocity approximation using a least squares method. A similar models was also reported by Reddy[92](1993) in study of viscous incompressible flows.

On the other hand, Shyu et al[102](1989) proposed a mixed finite element model for friction-contact from a perturbed Lagrangian variational principle. The displacements and contact pressure in an element were independently approximated. Some examples were included in the study.

A very interesting application of weighted residual approaches was reported by Moitinho de Almeda[73](1992) in which compatibility or equilibrium is satisfied locally. In the approach, linearly independent functions were defined within each element irrespectively of the location of the nodes. The lack of interelement continuity, due to abandoning the concept of the nodal variables, is eliminated *a posteriori* by utilizing the weighted residual approach. As the approach used is independent of the choice of basic functions and of the shape of the elements, it is very general. The matrices obtained for the governing system are symmetric, sparse and semi-definite. However, the results in the examples show that the interelement continuity is not secured very well when lower order basic functions are chosen for approximating the stress or displacement within the element.

A mixed variational principle based upon a combination of modified potential and complementary energy principles was shown by Day and Yang[26](1982). Compatibility and equilibrium are satisfied throughout the domain *a priori*, leaving only the boundary conditions to be satisfied by a variational principle. The nodal concept is also abandoned due to the difficulties to establish stress distribution identically satisfying the

equilibrium equations and still expressible in terms of a convenient set of nodal quantities, but the continuity between elements is not explicitly satisfied. This is enforced in a weighted average sense by an extra term in the variational principle. The resulting system equation is unsymmetric. Numerical examples show that high order of trial functions are generally needed to maintain the good results and convergency.

In the area of 2D linear elasticity, progress was reported by Chandler[15](1992) where a mixed extremum principle was developed. The continuity of displacements and the force equilibrium equations are both satisfied within an element and at the element boundary by means of shape functions and vector stress functions *a priori*. The Hooke's Law can be approximately imposed at the maximum of the functional, which approximately brings the satisfaction of the moment equilibrium. The advantage over some existing mixed formulations is that the global matrix is symmetric and positive-definite and only C^0 continuity is needed between the elements for both types of variables. An application of this model was reported in the analysis of an incompressible plane strain disc.

2.2 DISCRETIZATION IN MIXED MODELS

As we have just shown in section 2.1 that mixed models utilize different fundamental principles, such as mixed variational principles, from those in the conventional displacement method to construct the model. In addition, the mixed models also involve the discretization of the secondary variable(s) within each element, i.e. stresses and/or strains, while only displacements are needed to be discretized in displacement methods.

2.2.1 Discretization with Continuous Variable(s)

A natural way to discretize stresses is, analogous to formulations in the displacement methods, to approximate each component of stresses by means of shape functions

$$\sigma(x,y) = N^{\sigma}_i \, \sigma_i, \quad \text{together with} \quad u(x,y) = N^{u}_i \, u_i \tag{2.1}$$

where $\sigma = \{ \sigma_{xx} , \sigma_{yy} , \sigma_{xy} \}^T$ and $u = \{ u_x , u_y \}^T$ are the stress and displacement fields respectively within an element for 2D problems. σ_i and u_i are the nodal variables of the element. N^{σ}_i and N^{u}_i are shape functions for stress components and displacement vectors respectively.

Usually the shape functions of the displacement variables in mixed model are C^0 continuity functions, identical to those in a displacement model in plane problems. The shape functions for stresses are independent of those of displacement variables but they can be the same. The character of this manner of discretization is that both the displacement and the stress are continuous across the interelement boundaries and the nodal variables also include both of them. It is the kind of discretization which is used by the continuous models defined in section 1.2.2

Desai[28] in his work used such approaches to construct a 6-noded triangle elements and an 8-noded isoparametric elements in plane elasticity. The degree-of-freedom (DOF) is 5 per node, i.e. u_x, u_y, σ_{xx}, σ_{yy} and σ_{xy}. Similar approximations are adopted by Liao[59], where 4-noded and 8-noded quadrilateral elements are used.

A 3-noded triangle element with 5 degree-of-freedom per node is used by Tseng[115] for the analysis of contact problems in a similar way. Mahapatra[64], in axisymmetric problems, uses the continuous stress and displacement approximation, where 2 displacements u_x and u_y, and 4 stress components σ_{rr}, σ_{zz}, σ_{rz} and $\sigma_{\theta\theta}$ are the system variables, e.g. $6 \times 8 = 48$ DOF for an 8-noded element.

There are also some models where superpositions are used in expressing the stress variables. Normally, the higher order polynomial series, (e.g. the analytical solution of the stress field near crack), are added to the conventional continuous stress approximation. Heyliger[41] used such an approach in the solution of the crack problems. The polynomial series are taken as the analytical solution of the crack tip displacement and stress fields. The approximation functions are

$$\begin{aligned} \mathbf{u} &= \sum_i^n N_i u_i + \left\{ \sum_i^m \Theta_i \beta_i \right\} \cdot R \\ \sigma &= \sum_i^n N_i \sigma_i + \left\{ \sum_i^m \Xi_i \beta_i \right\} \cdot R \end{aligned} \tag{2.2}$$

where n and m are the number of nodes per element and number of terms of polynomial

series respectively. u_i and σ_i are the nodal variables of displacements and stresses,

$$\mathbf{u}_i = \begin{Bmatrix} u_{x_i} \\ u_{y_i} \end{Bmatrix} \quad , \quad \boldsymbol{\sigma}_i = \begin{Bmatrix} \sigma_{xx_i} \\ \sigma_{yy_i} \\ \sigma_{xy_i} \end{Bmatrix} \tag{2.3}$$

and β_i are unknown parameters to define the analytical solutions. Θ_i and Ξ_i are known coefficient matrices. The transition function R is employed to ensure the continuity of both displacement and stress between the "enriched" elements and the conventional elements. R is taken as 1.0 next to the enriched elements and as 0.0 next to the conventional elements. The number of total variables per elements is $5 \times n + m$.

The difficulty faced by continuous mixed models is to construct shape functions which automatically ensure that the resulting stresses satisfy *a priori* the equilibrium equations. Therefore the equilibrium condition has to be imposed *a posteriori*.

2.2.2 Discontinuous Mixed Models

Although the C^0 continuous stress approximation of the mixed method produces more accurate results than the standard displacement analysis does[126], the disadvantages are obvious: it brings additional variables into the system, and when there is a rapid change of the material properties, the discontinuity of the true solution can not be reproduced. Zienkiewicz[126] described this disability as ***excess continuity***.

In fact, in applying the principles, such as the mixed variational or weighted residual approaches, it is not necessary to maintain the continuity of stresses or even the traction reciprocity along the interelement boundary, it is permissible to express the stresses in terms of stress parameters which are only defined locally in each element. The stress parameters can then be eliminated in the element level and the system remains the same as if displacements are the only variables. The mixed models based on such a discretization are called the discontinuous models defined in section 1.2.2

Herrmann[40] in his pioneering work in a mixed bending model, used a linear C^0 function for the transverse displacement w, which is identical to it in plane problems, but constant C^{-1} distributions for the three stress components. The final equations

leaded to only 6 variables per element, which is the same as in a displacement model.

A three field mixed model by Simo[105] also discretizes both strains and stresses into discontinuous functions. The stresses are then eliminated from the system.

In solid-fluid problems or incompressible problems, the discontinuous pressure p is usually chosen. This was adopted by Spilker in ref.[109]. In the mixed penalty formulation, solid displacement, fluid velocity and pressure are independently interpolated within the element; C^0 continuity is required for the displacement/velocity, and discontinuity(C^{-1}) is required for the pressure. For the axisymmetric 6 node triangle, the displacement/velocity interpolation is quadratic and continuous between elements, while the pressure interpolation is linear but discontinuous. After interpolation, the unknown pressure will be eliminated at the element level.

The discontinuous mixed model gives more flexibility when constructing the stress approximate function. The nodal variables for the stress and/or strain are replaced by the parameters in a polynomial series. However nodal variables for displacement remain in the system since continuous displacement is generally required. The most clever option is to choose a polynomial series which satisfies the equilibrium equations or simply uses the analytical solutions as the approximate functions, which can greatly reduce the work in constructing the variational principles. The corresponding parameters are included in the system variables.

Zienkiewicz[126] gave a survey on the varieties of the mixed models. In discontinuous mixed model, the ***principle of limitation*** has to be taken care of: If the approximation for the secondary variables is capable of reproducing precisely the same type of variation as that determinable from the displacement method, no additional accuracy will result, and indeed the two approximations will yield identical answers. This discovery was made by Fraeijs de Veubeke[33].

Clearly, this limitation is not applicable to the mixed models with C^0 continuous stresses and displacements since C^0 displacement can not reproduced a C^0 continuous stress.

Considering that both the displacements and stresses and/or strains are discontinuous across boundaries, the nodal variables can be eventually abandoned in the whole system. In fact, this was done by Day[26] in his work, who introduced a mixed model with polynomial displacements and stress approximations which are both

discontinuous across the interelement boundaries. The compatibility and equilibrium are satisfied throughout the domain *a priori* by carefully choosing the polynomial functions. No nodal variable appears in the model. The system variables are parameters of the polynomials, called general displacement parameters and general stress parameters. Although the formulations representing each field are independent in the theoretical sense, the practical examples show that two assumed functions should be comparable, i.e. the order for stresses should be equal to or less than it for displacements. In order to get a reasonable results in this model, a fifth order displacement trial and a fifth order stress trial function are desired, where there are 22 displacement and 23 stress parameters per element. Obviously, the system variables are formidable compared with the conventional displacement method, e.g. 4-node element with 8 DOF and 8-node with 16 DOF in 2D problems. Although the idea of this model is new, it is doubtful that any practical interests will result from it. A similar idea was also conceived by Moitinho[73] in his work, in the hope of establishing a general approach to the construction of finite element formulations for solids.

2.2.3 Discretization of Stress Function

All the discretization approaches mentioned so far try to discretize directly the components of the stress and/or strain. The approximate stress field so chosen may or may not satisfy the local equilibrium equations. If they satisfy the equilibrium requirement in the domain and obey the traction reciprocity along the interelement boundaries, a simpler principle is needed to construct a model and also fewer operations are needed to form the element matrices. But these functions are more difficult to find. Alternatively, the easy-to-find functions, which do not satisfy the equilibrium equations but are continuous, leads to more complicated principles in order to approximately impose the equilibrium equations. Thanks to the versatility of the variational principle approaches and weighted residual approaches, this is not a problem but at the expense of more operations. Therefore, the ideal discretization of stresses is the one that satisfy both equilibrium equations and traction reciprocity, and are easy to express, i.e. not too many variables or high order function involved.

The well-known stress functions in solid mechanics are clearly the best candidate.

Stress derived from them automatically obeys the equilibrium conditions. If the stress functions are continuous in the discretized solid, the traction reciprocity along any interelement boundary within the domain is also obeyed. In 2D problems, the Airy function[5] is the stress function,

$$\sigma_{xx} = \frac{\partial^2 \phi}{\partial y \partial y} \quad ; \quad \sigma_{yy} = \frac{\partial^2 \phi}{\partial x \partial x} \quad ; \quad \sigma_{xy} = \sigma_{yx} = -\frac{\partial^2 \phi}{\partial x \partial y} \tag{2.4}$$

If we discretize ϕ within each element in the same way as discretizing displacements which are continuous of course, the following expression can be obtained,

$$\phi = \sum_{i}^{n} N_i \phi_i \tag{2.5}$$

Sarigul[98] uses such approximate stress in his assumed stress function finite element method in 2D elasticity, in which complementary energy is used to formulate the system equations.

One critical shortcoming of this approach is that the approximate stress function ϕ has to have C^1 continuity since the second order differential of ϕ is present. This is why the stress function discretization is rarely used in the mixed model.

In fact, two first order stress functions $\phi/_x$ and ϕ_y with C^0 continuity, were reported by Wood[124] and Chandler[15] independently, from which the stresses can be derived

$$\begin{aligned} \sigma_{xx} &= \frac{\partial \phi_y}{\partial y} & \sigma_{xy} &= -\frac{\partial \phi_x}{\partial y} \\ \sigma_{yx} &= -\frac{\partial \phi_y}{\partial x} & \sigma_{yy} &= \frac{\partial \phi_x}{\partial x} \end{aligned} \tag{2.6}$$

and so as to satisfy force equilibrium equation $\sigma_{ij,i} = 0$. But it does not satisfy moment equilibrium equation $\sigma_{ij} = \sigma_{ji}$. This is only satisfied if

$$\phi_x = \frac{\partial \phi}{\partial x} \quad ; \quad \phi_y = \frac{\partial \phi}{\partial y} \tag{2.7}$$

where ϕ is the Airy stress function.

In Wood's study[124], moment equilibriums are not required in the couple-stress problems. However this requirement has to be satisfied in conventional elasticity. In ref [15], this was accomplished by the approximate satisfaction of the Hooke's law *a posteriori* by an extremum principle. It is also found that vector stress functions allow permissible stress jumps across a surface, as would be required if the material properties change as the surface was crossed.

A plane strain disc under internal pressure loading was analysed by such method in ref [15] for both compressible and incompressible materials. The results show that this mixed model does represent the moment equilibrium properly and has much better performance in incompressible cases. However, the more difficult cases, where shear strain plays a crucial role, have not yet been encountered in the example, e.g. beam bending under tip traverse loading, where an error in calculation of the shear stresses may arise due to lack of symmetry of stress tensor in the stress approximation.

2.3 MATERIALS PROPERTIES

Before getting further the application of the mixed models to nonlinear problems, it is useful to have historical remarks about the development of various theories of material behaviours.

2.3.1 Mathematical Theories of Elasticity and Plasticity

The description of elastic behaviour is firmly established, for example the Hooke's law[110]: $\sigma_{ij} = C_{ijkl}\, \varepsilon_{kl}$. The hypothesis on which it bases is that the stress of the material is proportional to the strain sustained and the removal of the stress will leads to no *residual strain* or *permanent strain* in the material. The material is said to be elastic.

Most materials have an initial region on the stress-strain diagram in which the material behaves both elastically and linearly. However when the stress is beyond a limit, i.e the *elastic limit*, Hooke's law is no longer correct because plasticity takes place.

The study of plasticity was initiated by Tresca[113] in 1864. In that year he published a preliminary account of experiments on punching and extrusion, which led him to state that a metal yielded plastically when the maximum shear stress attained a critical value. The actual formulation of the theory was done in 1870 by St. Venant[27], who introduced the basic constitutive relations for what today we would call rigid, perfectly plastic materials in plane stress. It remained for Levy[58] later in 1871 to obtain the general equations in three dimensions. A generalization similar to the results of Levy was arrived at independently by von Mises in a landmark paper in 1913[118], accompanied by his well-known yield criterion. The established relationship of stress-strain for rigid-plastic is called the ***Levy-Mises equations***.

In 1924 Prandtl[119] extended Levy-Mises equation for plane continuum problems to include the elastic component of strain. Reuss[94] in 1930 carried out the extension to three dimensions. The relation is called the ***Prandtl-Reuss equation*** in elastic-perfectly plasticity. In 1928, Von Mises[119] generalized his previous work for rigid, perfectly plastic solid to include a general yield function and discussed the relation between the direction of plastic strain rate (increment) and the regular or smooth yield surface, thus introducing formally the concept of using the yield function as a plastic potential in the incremental stress-train relations of flow theory, known as the ***associated flow theory***. It is well-known now that the Von Mises yield function may be regarded as a plastic potential for Levy-Mises stress-strain relations.

For hardening materials, independently of the work of Melan[69] in 1938, Prager[86], in a significant paper published in 1949, gave a general framework for the plastic constitutive relations. A very important concept of work hardening, termed the material stability postulate, was proposed by Drucker[29] in 1951. With this concept, the plastic stress-strain relations together with many related fundamental aspects of the subject may be treated in a unified manner.

2.3.2 Mixed FE Models for Elasto-Plasticity

Now we are in the position to discuss the mixed finite element model with regard to the material properties. In the framework of displacement finite element formulations, plastic loading is tested at each quadrature point of the element, and a return mapping

algorithm is performed at this point for given incremental displacements. Only total and plastic strains are the independent variables, while the stress is regarded as dependent variable which is computed from the elastic strains by means of stress-strain relations.

In contrast to this view, many literature on plasticity has been concerned with the formulation if the elasto-plastic problem with the stress field as the independent variable. In a mixed model this is natural true because both displacement and stress are almost certainly independent variables. Normally stresses so obtained only satisfy the equilibrium conditions but do not obey the yield criterion. Similar algorithms of stress integration for calculating stress components are normally borrowed from the conventional displacement method, such as the well-known elastic predictor-radial return method[55,79] in order to make the stress obey the yield criterion. A detailed survey of stress integration can be found in Ortiz's work[80].

Mahapatra and Dasgupta[64](1988) used an mapping algorithm in his mixed model, which is identical to the one in displacement method, in elasto-plastic axisymmetric problems. A similar algorithm is also used by Jetteur[48](1991) in a large inelastic strain study, and by Liao[59](1992) for large displacement and elasto-plastic and post-buckling response of plane problems.

Simo et al[104](1989) presented a interesting formulation for elasto-plasticity based on the principle of maximum plastic dissipation. The fundamental difference is that the plastic return mapping algorithm can no longer be formulated independently at each Gauss point, in contrast with displacement-like methods. The closest-point-projection iteration that restores consistency is performed at the global element level and involves all the Gauss points within the element. It was found that for a given mesh the mixed model converges slower than the displacement model.

Chandler[18] reported an interesting variational principle in rigid plasticity for granular materials, where Drucker's postulate is inappropriate. Since the requirements in advance are a displacement field that obeys the kinematic boundary conditions, and a stress field that obeys equilibrium and the static boundary conditions, the variational principle itself was to impose the constitutive relation of the granular materials. Therefore, the stresses obtained by the stationary of the functional will not only satisfy equilibrium equations, but also the yield criterion.

2.4 CONVERGENCE OF MIXED MODELS

The most favourite character of mixed models is that the sufficient and necessary condition of the stability and convergence of a solution exists: if Babuska-Brezzi(B-B) condition is satisfied, a solution must converge to the true solution[12]. Such a criterion is simply not available in conventional displacement methods. B-B condition is very general and complicated. Therefore for some specific mixed models, simplified criterions of solution may be desired.

A lot of researchers have devoted themselves to this topic. Silvester[103](1990) gave a sufficient condition for the stability of low-order mixed finite element methods. Pinsky[85](1991) studied the dependency of the numerical stability and accuracy of the results from the mixed shell elements on the balance between the stress, displacement and Lagrange multiplier fields.

In 1990, Brezzi and Bathe[11] discussed the general mathematical conditions for solvability, stability and optimal error bounds of mixed finite element discretizations. A numerical test that is useful to identify numerically wether, a given finite element discretization satisfies the stability and optimal error bound conditions, is also given.

CHAPTER 3
MIXED MODEL IN PLANE ELASTICITY

A good example of problem in continuum mechanics is 2D linear elasticity. To solve such problems successfully is the first step in solving more difficult problems.

3.1 MIXED EXTREMUM PRINCIPLE

Consider a body B with surface Γ and with Cartesian coordinates x_i. An exact linear elastic solution requires:

- A continuous displacement vector field, with components u_i, which obeys the kinematic boundary conditions and from which the components of the strain tensor e_{ij} can be derived by using

$$e_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}) \tag{3.1}$$

which is symmetric. The subscript after the comma represents differentiation with respect to the corresponding spacial coordinate. Such a displacement field is termed kinematically admissible.

- A stress tensor field with components σ_{ij}, which satisfies the force equilibrium

$$\sigma_{ij,i} = 0 \tag{3.2}$$

and the moment equilibrium

$$\sigma_{ij} = \sigma_{ji} \tag{3.3}$$

in the interior B and is consistent with the prescribed traction

$$t_j = \sigma_{ij} n_i \tag{3.4}$$

on the boundary surface. Such a stress field is termed statically admissible. The moment equilibrium implies that stress tensor σ_{ij} is symmetric. Note that the Einstein summation convention is used here and elsewhere in this thesis.

- These components of stress and strain obey Hooke's law,

$$\sigma_{ij} = C_{ijkl} e_{kl} \tag{3.5}$$

where C_{ijkl} is the stiffness tensor.

This can also be written in a matrix form as

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon} \tag{3.6}$$

where $\boldsymbol{\sigma}$ and $\boldsymbol{\varepsilon}$ are vector forms of stress and strain tensors respectively defined by

$$\begin{aligned} \boldsymbol{\varepsilon} &= [\varepsilon_{11} \;\; \varepsilon_{22} \;\; \varepsilon_{33} \;\; \varepsilon_{12} \;\; \varepsilon_{23} \;\; \varepsilon_{13}]^T \\ \boldsymbol{\sigma} &= [\sigma_{11} \;\; \sigma_{22} \;\; \sigma_{33} \;\; \sigma_{12} \;\; \sigma_{23} \;\; \sigma_{13}]^T \end{aligned} \tag{3.7}$$

and $\mathbf{D}$ is the elastic stiffness matrix for isotropic elasticity defined by

$$\mathbf{D} = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \frac{\nu}{(1-\nu)} & \frac{\nu}{(1-\nu)} & 0 & 0 & 0 \\ \frac{\nu}{(1-\nu)} & 1 & \frac{\nu}{(1-\nu)} & 0 & 0 & 0 \\ \frac{\nu}{(1-\nu)} & \frac{\nu}{(1-\nu)} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} \end{bmatrix} \tag{3.8}$$

for three dimensional (3D) cases. The constants ν and E are the Poisson's ratio and the Young's modulus respectively. The overall number of the independent components of the stress and strain are six due to the fact of $\sigma_{ij}=\sigma_{ji}$ and $\varepsilon_{ij}=\varepsilon_{ji}$.

Alternatively, it is sometimes, especially in elasto-plasticity, more convenient to describe the Hooke's law in terms of deviatoric stress s_{ij} and deviatoric strain d_{ij} together with mean stress σ_m and volume strain e_m

$$s_{ij} = 2Gd_{ij} \qquad ; \qquad \sigma_m = Ke_m \tag{3.9}$$

where

$$\begin{aligned} \sigma_m &= \frac{1}{3}\sigma_{kk} & e_m &= e_{kk} \\ s_{ij} &= \sigma_{ij} - \delta_{ij}\sigma_m & d_{ij} &= e_{ij} - \frac{1}{3}\delta_{ij}e_m \end{aligned} \tag{3.10}$$

The independent constants K and G in eq.(3.9) are shear modulus and bulk modulus respectively and are both positive.

In a conventional displacement method, Hooke's law is explicitly incorporated in deducing the fundamental principles, such as in minimum potential energy principle. As to the kinematic admissibility and static admissibility requirements, only one is satisfied in advance in such single-field models. On the other hand, most of the mixed models choose an approximate displacement which is fully kinematically admissible, and an approximate stress which is only locally statically admissible. Therefore for the mixed models, satisfaction of the overall equilibrium is left to the task of the mixed principles. In both conventional models and mixed models the boundary conditions are imposed by fundamental principles, which becomes one of the main advantages of the FE method. They largely simplify the solution of a boundary value problem.

We may also assume that both kinematic condition and static conditions are fully satisfied in our approximate displacement and stress field, which are called as ***admissible displacement*** and ***admissible stress*** respectively. Now only the stress-strain relation is imposed by the mixed principle. This is the case in this chapter where a mixed extremum principle is proposed in a least square sense to approximately enforce Hooke's law. As reviewed in chapter 2, very little work appears in the literature where the constitutive relations are to be imposed by extremum principles. The most favourable point of this scheme is that principles suitable for any materials can be readily obtained by imposing the explicit constitutive relationship, without change of formulation in the admissible displacement and stress. We will see later in chapter 5 and chapter 6 that it is of special interest if the constitutive relationship is taken as the one for rigid perfectly-plasticity or elasto-plasticity.

In the case of linear isotropic elasticity we can construct an extremum principle very easily. Give an assumed strain e_{ij}^* and an assumed stress σ_{ij}^* which are either kinematically admissible or statically admissible. The corresponding deviatoric strain d_{ij}^*, stress s_{ij}^*, mean stress σ_m^* and volume strain e_m^* can be derived by substituting e_{ij}^* and σ_{ij}^* for e_{ij} and σ_{ij} in eq.(3.10). Since they do not satisfy eq.(3.9), it follows

$$R_1^2(s_{ij}^*, d_{ij}^*) = (s_{ij}^*/2G - d_{ij}^*)^2 \geq 0 \quad ; \quad R_2^2(\sigma_m^*, e_m^*) = (\sigma_m^*/K - e_m^*)^2 \geq 0 \qquad (3.11)$$

where R_1 and R_2 can be considered as some kind of residual functions of admissible

stress and strain. It is straightforward to obtain the following inequality for elasticity

$$\Pi_{elas} = \int_{\Omega}[\alpha_1 R_1^2(s_{ij}^*, d_{ij}^*) + \alpha_2 R_2^2(\sigma_m^*, e_m^*)]dV \geq 0 \qquad (3.12)$$

where α_1 and α_2 are two arbitrary positive constants.

It is evident that Π_{elas} has a minimum of zero, and the equality is only true when d_{ij}^*, s_{ij}^*, σ_m^* and e_m^* fully satisfy the Hooke's law, i.e. equation (3.9).

Recall what we have described in section 1.4.4. It immediately turns out that eq.(3.12) is identical to the least squares method in the penalty function approach. Thus the two coefficients α_1 and α_2 can be simply regarded as penalty numbers. It comes to a question: how to choose the penalty numbers and what are the best choices? One answer is $\alpha_1 = G$ and $\alpha_2 = K/2$ which will then allow the formulation to be reduce to the principles of minimum potential energy and minimum complementary energy. It can be verified as follows.

Expanding eq.(3.12) by substituting eq.(3.11), it follows that

$$\Pi = \int_{\Omega}\alpha_1\left(\frac{s_{ij}s_{ij}}{4G^2} - \frac{2}{2G}s_{ij}d_{ij} + d_{ij}d_{ij}\right) + \alpha_2\left(\frac{\sigma_m^2}{K^2} - \frac{2}{K}\sigma_m e_m + e_m^2\right)dV \qquad (3.13)$$

Replacing α_1 and α_2 by G and $K/2$ respectively and reordering this equation, it is very interesting to find that the right hand side is composed of three terms

$$\Pi_{elas} = \int_{\Omega} -2(\frac{1}{2}\sigma_m e_m + \frac{1}{2}s_{ij}d_{ij}) + [\frac{1}{2}(2G)d_{ij}d_{ij} + \frac{1}{2}Ke_m^2] + [\frac{1}{2(2G)}s_{ij}s_{ij} + \frac{1}{2K}\sigma_m^2]dV \quad (3.14)$$

The first term in bracket is twice the internal energy, the second the strain energy U_s and the third the complementary strain energy U_c. Therefore eq.(3.12) can be written

$$\Pi_{elas} = -2\sigma_{ij}e_{ij} + U_s + U_C \geq 0 \qquad (3.15)$$

If stresses are statically admissible and the strains are kinematically admissible, then the theorem of virtual work[108]

$$\int_{\Omega}\sigma_{ij}e_{ij}d\Omega = \int_{\Gamma_k} f_j\overline{u}_j d\Gamma + \int_{\Gamma_s}\overline{f}_j u_j d\Gamma$$

can be used to replace the internal energy with the external work done in eq.(3.15)

$$\int_\Omega (U_s + U_c)d\Omega - \int_{\Gamma_k} f_j \overline{u}_j d\Gamma - \int_{\Gamma_s} \overline{f}_j u_j d\Gamma \geq 0$$

where Γ_s is the boundary with the traction f_i fixed, while Γ_k is the boundary with the displacement u_i fixed.

Let displacements and strains be functions of α_i and stresses be functions of β_i. The minimum of Π can be found by

$$\begin{aligned}\delta\Pi = &\left[\int_\Omega \frac{\partial U_s}{\partial \alpha_i} d\Omega - \int_{\Gamma_s} \overline{f}_j \frac{\partial u_j}{\partial \alpha_i} d\Gamma\right] \delta\alpha_i \\ &+ \left[\int_\Omega \frac{\partial U_c}{\partial \beta_i} d\Omega - \int_{\Gamma_k} \frac{\partial f_j}{\partial \beta_i} \overline{u}_j d\Gamma\right] \delta\beta_i = 0\end{aligned} \tag{3.16}$$

Because α_i and β_i in eq.(3.16) are independent variables, the above equation yields the following equations:

$$\delta\Pi = \int_\Omega \delta U_s d\Omega - \int_{\Gamma_s} \overline{f}_j \delta u_j d\Gamma = 0$$

which is the *principle of minimum potential energy*, and

$$\delta\Pi = \int_\Omega \delta U_c d\Omega - \int_{\Gamma_k} \delta f_j \overline{u}_j d\Gamma = 0$$

which is *principle of minimum complementary energy*.

From now on, G and $K/2$ replace α_1 and α_2 in all expressions of Π_{elas}.

3.2 DISCRETIZATION OF DISPLACEMENTS FOR PLANE PROBLEMS

Following the procedure in a conventional displacement model, the discretization of displacements in this mixed model is accomplished by means of the isoparametric formulation[126]

$$\begin{Bmatrix} u(\xi,\eta) \\ v(\xi,\eta) \end{Bmatrix} = \sum_{i=1}^{m} N_i(\xi,\eta) \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} u_i \\ v_i \end{Bmatrix} \tag{3.17}$$

where ξ and η are local coordinates in an element, $u(\xi,\eta)$ and $v(\xi,\eta)$ are approximate displacements in x-direction and y-direction respectively. u_i and v_i are the nodal

variables of the element corresponding to $u(\xi,\eta)$ and $v(\xi,\eta)$, N_i is the shape function of node i in an element, m is the number of nodes per element, as shown in fig.3.1. Shape functions in isoparametric are defined as[126]

For a 4-noded element

$$N_i(\xi,\eta)=\frac{1}{4}(1+\xi\xi_i)(1+\eta\eta_i)$$

Local node number	ξ_i	η_i
1	-1	-1
2	1	-1
3	1	1
4	-1	1

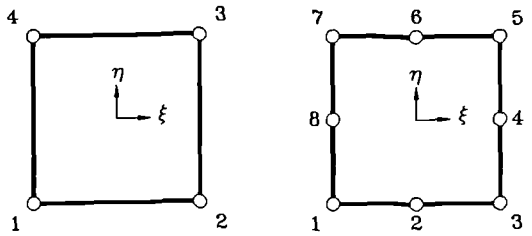


Fig.3.1 The geometric layouts of a 4-noded and a 8-noded isoparametric element.

For a 8-noded element

$$N_i(\xi,\eta)=\frac{1}{4}(1+\xi\xi_i)(1+\eta\eta_i)(\xi\xi_i+\eta\eta_i-1) \qquad i=1,3,5,7$$

$$N_i(\xi,\eta)=\frac{1}{2}\xi_i^2(1+\xi\xi_i)(1-\eta^2)+\frac{1}{2}\eta_i^2(1+\eta\eta_i)(1-\xi^2); \quad i=2,4,6,8$$

Local node number	ξ_i	η_i
1	-1	-1
2	0	-1
3	1	-1
4	1	0
5	1	1
6	-1	1
7	-1	0
8	0	0

The continuity condition of displacement within the element and on the inter-element boundary is ensured by such formulations[126]. If the strain is expressed by eq.(3.1), the compatibility condition is also ensured, which implies that the displacement field so obtained is kinematically admissible.

In 2D plane stress problems, the vector forms of strain and deviatoric strain tensors are introduced for convenience of FE formulation as

$$\mathbf{e} = \{e_{xx}\ e_{xy}\ e_{yx}\ e_{yy}\ e_{zz}\}^T \quad ; \quad \mathbf{d} = \{d_{xx}\ d_{xy}\ d_{yx}\ d_{yy}\ d_{zz}\}^T$$

where 5 components of strain are used in order to take into account the fact that $\sigma_{xy}=\sigma_{yx}$ does not apply in our model. Then strain, deviatoric strain tensors and volume strain can be approximated in matrix forms as

$$\mathbf{e} = \sum_{i=1}^{m} \mathbf{B_i}\ \beta_i^d \quad ; \quad \mathbf{d} = \sum_{i=1}^{m} \mathbf{B_{Di}}\ \beta_i^d \quad ; \quad e_m = \sum_{i=1}^{m} \mathbf{B_{mi}}\ \beta_i^d \tag{3.18}$$

where

$$\mathbf{B_{Di}} = \mathbf{B_i} - \frac{1}{3}\mathbf{T}\mathbf{B_{mi}^T}$$

$$\mathbf{B_i} = \begin{bmatrix} N_{i,x} & \frac{1}{2}N_{i,y} & \frac{1}{2}N_{i,y} & 0 & -\frac{\nu}{1-\nu}N_{i,x} \\ 0 & \frac{1}{2}N_{i,x} & \frac{1}{2}N_{i,x} & N_{i,y} & -\frac{\nu}{1-\nu}N_{i,y} \end{bmatrix}^T$$

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}^T \tag{3.19}$$

$$\mathbf{B_{mi}} = [N_{i,x}\ \ N_{i,y}] \qquad \text{(for plane stress)}$$

$$\mathbf{B_{mi}} = \frac{1}{1-\nu}[N_{i,x}\ \ N_{i,y}] \qquad \text{(for plane strain)}$$

$$\beta_i^d = \{u_i\ \ v_i\}^T$$

For plane strain, the last row of $\mathbf{B_i}$ should be replaced by [0 , 0] .

3.3 DISCRETIZATION OF STRESSES FOR PLANE PROBLEMS

The discretization of stresses is always an important but difficult task in the mixed

model. A good formulation for discretizing the stress field will enable the model to be highly accurate and efficient.

To satisfy of the force equilibrium equation (3.2), it is easy to verify that the following representation of the components of stress is appropriate

$$\begin{aligned} \sigma_{xx} &= \frac{\partial F_y}{\partial y} \quad ; \quad \sigma_{xy} = -\frac{\partial F_x}{\partial y} \\ \sigma_{yx} &= -\frac{\partial F_y}{\partial x} \quad ; \quad \sigma_{yy} = \frac{\partial F_x}{\partial x} \end{aligned} \tag{3.20}$$

where F_x and F_y are two independent functions. Because no explicit second order derivative of F_x and F_y is included, the continuity of these functions is C^0. Therefore functions F_x and F_y are called the first-order stress functions. However it does not imply the satisfaction of the moment equilibrium equation (3.3) since

$$\frac{\partial F_x}{\partial y} = \frac{\partial F_y}{\partial x} \tag{3.21}$$

is not true for two arbitrary functions F_x and F_y. If we upgrade the order of eq.(3.20) by specifying that

$$F_x = \frac{\partial \phi}{\partial y} \qquad \text{and} \qquad F_y = \frac{\partial \phi}{\partial x}$$

where ϕ is a second-order continuous function, the equilibrium conditions (3.2) and (3.3) are fully satisfied. The components of the stress can then be expressed

$$\begin{aligned} \sigma_{xx} &= \frac{\partial^2 \phi}{\partial y^2} \quad ; \quad \sigma_{xy} = -\frac{\partial^2 \phi}{\partial x \partial y} \\ \sigma_{yx} &= -\frac{\partial^2 \phi}{\partial x \partial y} \quad ; \quad \sigma_{yy} = \frac{\partial^2 \phi}{\partial x^2} \end{aligned} \tag{3.22}$$

in which ϕ is the well-known Airy stress function[5]. Since the Airy stress function is the lowest order and the simplest function to express equilibrated stress components in 2D problems, it is inevitable that to impose both force and moment equilibrium conditions C^1 second-order functions must be used.

However if we give the functions F_x and F_y a constraint such that eq.(3.21) is true, the description of the stress in eq.(3.20) will fully satisfy equilibrium conditions

eq.(3.2) and eq.(3.3). In other words, the second-order Airy stress function is decomposed into two first-order functions, which obey force equilibrium condition, and a constraint which imposes moment equilibrium condition. Now we have achieved the construction of a first order stress function together with a constraint, or a ***constrained first order stress function***. The constraint will be treated later as in the constrained variational principles by a penalty function described in section 1.4.4.

With C^0 stress function in hand, the discretization of the stress becomes the discretization of the C^0 stress functions F_x and F_y, which can be thought as two components of a vector. Therefore isoparametric formulation can be used again to discretize stress function,

$$\begin{Bmatrix} F_x(\xi,\eta) \\ F_y(\xi,\eta) \end{Bmatrix} = \sum_{i=1}^{m} N_i(\xi,\eta) \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} F_{xi} \\ F_{yi} \end{Bmatrix} \tag{3.23}$$

where $F_x(\xi,\eta)$ and $F_y(\xi,\eta)$ are two approximate stress functions. F_{xi} and F_{yi} are nodal variables of the element corresponding to $F_x(\xi,\eta)$ and $F_y(\xi,\eta)$, N_i is the shape function of node i in the element and m is the number of nodes per element. Again for the convenience of FE formulation, vector forms of stress and deviatoric stress tensors are introduced

$$\boldsymbol{\sigma} = \{\sigma_{xx} \;\; \sigma_{xy} \;\; \sigma_{yx} \;\; \sigma_{yy} \;\; \sigma_{zz}\}^T \quad ; \quad \mathbf{s} = \{s_{xx} \;\; s_{xy} \;\; s_{yx} \;\; s_{yy} \;\; s_{zz}\}^T$$

Then from eqs.(3.20) and (3.23) the stress, deviatoric stress and mean stress for plane stress cases can be expressed in matrix forms as

$$\boldsymbol{\sigma} = \sum_{i=1}^{m} \mathbf{A_i}\, \beta_i^s \quad ; \quad \mathbf{s} = \sum_{i=1}^{m} \mathbf{A_{Di}}\, \beta_i^s \quad ; \quad \sigma_m = \sum_{i=1}^{m} \mathbf{A_{mi}}\, \beta_i^s \tag{3.24}$$

where

$$\mathbf{A_i} = \begin{bmatrix} 0 & N_{i,y} \\ -N_{i,y} & 0 \\ 0 & -N_{i,x} \\ N_{i,x} & 0 \\ 0 & 0 \end{bmatrix} \quad ; \quad \begin{aligned} \mathbf{A_{Di}} &= \mathbf{A_i} - \mathbf{T}\mathbf{A_{mi}^T} \\ \mathbf{A_{mi}} &= \left[\tfrac{1}{3}N_{i,x} \;\; \tfrac{1}{3}N_{i,y}\right] \quad \text{(for plane stress)} \\ \mathbf{A_{mi}} &= \left[\tfrac{1+\nu}{3}N_{i,x} \;\; \tfrac{1+\nu}{3}N_{i,y}\right] \quad \text{(for plane strain)} \\ \beta_i^s &= \{F_{xi} \;\; F_{yi}\}^T \end{aligned} \tag{3.25}$$

T is the same as in eq.(3.19). For plane strain problems, the last row of matrix $\mathbf{A_i}$ should be replaced by $[\nu N_{ix}, \nu N_{iy}]$.

As we have reviewed in chapter 2, there are difficulties and inaccuracies brought in by the discretization of the stress field with either continuous or discontinuous stress across inter-element boundaries. For example, the models with discontinuous stress do not satisfy the traction reciprocity along the inter-element boundaries. Although this violation of the equilibrium condition can be partly cured by the mixed principles, it brings an inaccuracy into the solutions. On the other hand, the models with continuous stress do not permit the discontinuity of stress when there is a rapid change of the material properties, though traction reciprocity along the inter-element is ensured. This is termed as *excess continuity*[126]. However, the model in this thesis can solve the above problems entirely: it allows "stress jump" where material properties change rapidly, and at the same time satisfies the traction reciprocity at inter-element boundaries.

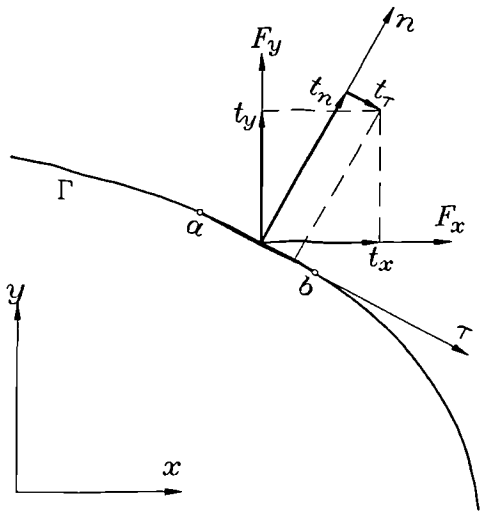


Fig.3.2 Local coordinate system at the element boundary.

Consider a surface Γ with a normal vector **n**, and a tangent vector τ. These two vectors form a local Cartesian coordinate system, as shown in fig.(3.2). The traction at any point on this surface is obtained by eq.(3.4) as follows,

$$\begin{cases} t_x = \sigma_{xx} n_x + \sigma_{yx} n_y \\ t_y = \sigma_{xy} n_x + \sigma_{yy} n_y \end{cases} \tag{3.26}$$

where n_x and n_y are direction cosines of the normal vector **n**. By projecting t_x and t_y in the normal and tangent directions, the two components of the traction are

$$\begin{cases} t_n = n_x t_x + n_y t_y \\ t_\tau = n_y t_x - n_x t_y \end{cases} \tag{3.27}$$

From eq.(3.26), it follows that

$$\begin{cases} t_n = n_x^2 \sigma_{xx} + n_y^2 \sigma_{yy} + n_x n_y (\sigma_{xy} + \sigma_{yx}) \\ t_\tau = n_x n_y (\sigma_{xx} - \sigma_{yy}) + n_y^2 \sigma_{yx} - n_x^2 \sigma_{xy}) \end{cases} \tag{3.28}$$

Remembering the following relations,

$$\begin{cases} \dfrac{\partial}{\partial x} = n_y \dfrac{\partial}{\partial \tau} + n_x \dfrac{\partial}{\partial n} \\ \dfrac{\partial}{\partial y} = -n_x \dfrac{\partial}{\partial \tau} + n_y \dfrac{\partial}{\partial n} \end{cases}$$

and substituting eq.(3.20) into eq.(3.28), the traction can be expressed in terms of stress functions as

$$\begin{cases} t_n = -n_x \dfrac{\partial F_y}{\partial \tau} + n_y \dfrac{\partial F_x}{\partial \tau} \\ t_\tau = -n_x \dfrac{\partial F_x}{\partial \tau} - n_y \dfrac{\partial F_y}{\partial \tau} \end{cases} \tag{3.29}$$

It shows that there is only the derivatives of F_x or F_y with respect to τ. Now assume two elements A and B, which are adjacent to one another. Since C^0 continuity of F_x and F_y is ensured by the isoparametric formulations, then $(F_x)^A = (F_x)^B$ and $(F_y)^A = (F_y)^B$ are true. Furthermore since the two elements are sharing common boundary, then $(\partial/\partial\tau)^A = (\partial/\partial\tau)^B$ is also true. It leads to the conclusion that traction reciprocity is exactly satisfied by this model, i.e. $(t_n)^A \equiv (t_n)^B$ and $(t_\tau)^A \equiv (t_\tau)^B$. It is of special interest that while traction reciprocity is satisfied exactly, the discontinuity of stress across inter-element boundaries is also permitted. This is automatically true in our model since the stress functions F_x and F_y are of C^0 continuity, the components of the stress expressed by their first derivatives, such as in

eq.(3.20), must be of C^{-1} continuity, i.e. discontinuous across inter-element boundaries. Therefore a "stress jump" is reproducible.

3.4 BOUNDARY CONDITIONS

In conventional single field models, boundary conditions are easy to be imposed by means of virtual work principle. However in using this, the symmetry of a stress tensor is essential. This is no longer true for our model, and therefore a different approach must be employed. The boundary conditions can be treated as some sort of constraints in the mathematical sense. For the variational principle approach, there are two constrained principles available, Langrange multiplier method or penalty function method. The penalty function method, of course, is the best choice since it does not bring extra variables to the system, neither does it cast the non-positive-definite of the matrix, as discussed in section 1.4.4.

Say ($\overline{u}$, $\overline{v}$) are given displacements at any point i on the boundary. By means of a quadratic penalty function, the true u and v are those which make the following functional

$$\Pi_d = (\overline{\beta_i^d} - \beta_i^d)^T \boldsymbol{\rho}_\mathbf{i}^\mathbf{d} (\overline{\beta_i^d} - \beta_i^d) \tag{3.30}$$

minimum, where

$$\boldsymbol{\rho}_\mathbf{i}^\mathbf{d} = \begin{bmatrix} \rho_{ui}^d & 0 \\ 0 & \rho_{vi}^d \end{bmatrix} \quad ; \quad \beta_i^d = \begin{Bmatrix} u_i \\ v_i \end{Bmatrix} \quad ; \quad \overline{\beta_i^d} = \begin{Bmatrix} \overline{u_i} \\ \overline{v_i} \end{Bmatrix}$$

In eq.(3.30), $\boldsymbol{\rho}_\mathbf{i}^\mathbf{d}$ is a 2×2 penalty matrix, the diagonal components of which are either a large positive number (penalty number) or zero depending on which degree of freedom at node i is to be fixed. It follows that the first derivative of Π_d with respect to β_i^d can be expressed in a matrix form

$$\frac{\partial \Pi_d}{\partial \beta_i^d} = \mathbf{k}_\mathbf{ii}^\mathbf{d} \beta_i^d + \mathbf{F}_\mathbf{i}^\mathbf{d} \tag{3.31}$$

where

$$\mathbf{k}_{\mathbf{ii}}^{\mathbf{d}} = 2\begin{bmatrix} \rho_{ui}^{d} & 0 \\ 0 & \rho_{vi}^{d} \end{bmatrix} \quad ; \quad \mathbf{F}_{\mathbf{i}}^{\mathbf{d}} = 2\begin{Bmatrix} \rho_{ui}^{d}\overline{u_i} \\ \rho_{vi}^{d}\overline{v_i} \end{Bmatrix}$$

Let us turn our attention to the stress boundary conditions. Say there is a boundary line Γ_e of the length dL pointing from node a to b. The positive direction of this line is defined as when walking from a to b, the right hand is always pointing to the outside of the body. If the length dL is small, the derivatives of F_x and F_y with respect to τ can be approximated by

$$\frac{\partial F_x}{\partial \tau} \approx \frac{1}{dL}(F_{xj} - F_{xi}) \quad ; \quad \frac{\partial F_y}{\partial \tau} \approx \frac{1}{dL}(F_{yj} - F_{yi}) \tag{3.32}$$

Substitute eq.(3.32) into eq.(3.29) and rewrite it in a matrix form

$$t_n = \frac{1}{dL}(-n_y \;\; n_x \;\; n_y \;\; -n_x)\begin{Bmatrix} \beta_a^s \\ \beta_b^s \end{Bmatrix} \quad ; \quad t_\tau = \frac{1}{dL}(n_x \;\; n_y \;\; -n_x \;\; n_y)\begin{Bmatrix} \beta_a^s \\ \beta_b^s \end{Bmatrix} \tag{3.33}$$

where the nodal variables β_a^s and β_b^s are defined as in eq.(3.25). The value of the prescribed traction $\overline{t_n}$ and $\overline{t_\tau}$ along a small segment of the boundary line $d\Gamma$ can be imposed again using a quadratic penalty function

$$\Pi_s = \int_{d\Gamma}\left[\begin{Bmatrix} \overline{t_n} \\ \overline{t_\tau} \end{Bmatrix} - \begin{Bmatrix} t_n \\ t_\tau \end{Bmatrix}\right]^T \boldsymbol{\rho}^{\mathbf{s}} \left[\begin{Bmatrix} \overline{t_n} \\ \overline{t_\tau} \end{Bmatrix} - \begin{Bmatrix} t_n \\ t_\tau \end{Bmatrix}\right] d\Gamma \tag{3.34}$$

where $\boldsymbol{\rho}^{\mathbf{s}}$ is a 2×2 matrix similar to $\boldsymbol{\rho}^{\mathbf{d}}$

$$\boldsymbol{\rho}^{\mathbf{s}} = \begin{bmatrix} \rho_n^s & 0 \\ 0 & \rho_\tau^s \end{bmatrix}$$

where ρ_n^s and ρ_τ^s are two penalty numbers. It follows that the first derivative of Π_s with respect to β_a^s and β_b^s is

$$\begin{Bmatrix} \dfrac{\partial \Pi_s}{\partial \beta_a^s} \\ \dfrac{\partial \Pi_s}{\partial \beta_b^s} \end{Bmatrix} = \mathbf{k}_{\mathbf{ab}}^{\mathbf{s}}\begin{Bmatrix} \beta_a^s \\ \beta_b^s \end{Bmatrix} + \mathbf{F}^{\mathbf{s}} \tag{3.35}$$

where

$$\mathbf{k}_{\mathbf{ab}}^{\mathbf{s}} = \int_{\Gamma^e} (\mathbf{T}_{\mathbf{n}} + \mathbf{T}_{\tau}) d\Gamma \quad ; \quad \mathbf{F}^{\mathbf{s}} = \int_{\Gamma^e} \overline{t_n} \mathbf{f}_{\mathbf{n}} + \overline{t_\tau} \mathbf{f}_{\tau}) d\Gamma$$

$$\mathbf{T}_{\mathbf{n}} = \frac{2\rho_n^s}{dL^2} \begin{bmatrix} n_y n_y & -n_x n_y & -n_y n_y & n_x n_y \\ -n_x n_y & n_x n_x & n_x n_y & -n_x n_x \\ -n_y n_y & n_x n_y & n_y n_y & -n_x n_y \\ n_x n_y & -n_x n_x & -n_x n_y & n_x n_x \end{bmatrix} \quad ; \quad \mathbf{f}_{\mathbf{n}} = \frac{2\rho_n^s}{dL} \begin{Bmatrix} -n_x \\ n_x \\ n_y \\ -n_x \end{Bmatrix}$$

$$\mathbf{T}_{\tau} = \frac{2\rho_\tau^s}{2L^2} \begin{bmatrix} n_x n_x & n_x n_y & -n_x n_x & -n_x n_y \\ n_x n_y & n_y n_y & -n_x n_y & -n_y n_y \\ -n_x n_x & -n_x n_y & n_x n_x & n_x n_y \\ -n_x n_y & -n_y n_y & n_x n_y & n_y n_y \end{bmatrix} \quad ; \quad \mathbf{f}_{\tau} = \frac{2\rho_\tau^s}{dL} \begin{Bmatrix} n_x \\ n_y \\ -n_x \\ -n_y \end{Bmatrix}$$

In the next section, we will discuss how to add these constraints to the functional Π_{elas} in eq.(3.14) to form a complete system for a 2D elasticity problem.

3.5 GLOBAL SYSTEM OF EQUATIONS

We have given a functional Π_{elas} in section 3.1, whose minimum corresponds to the true status of elasticity on condition that the displacement field and stress field are kinematically and statically admissible respectively. This in turn is ensured by the discretization of the displacement, strain and stress fields as described in sections 3.2 and 3.3. In section 3.4. two additional functionals Π_d and Π_s with regard to the displacement and traction boundary conditions are also introduced. The boundary conditions are satisfied approximately when these functionals are minimum. In other words, the true status of a solid under specific boundary conditions is the one which makes the functional

$$\Pi_I = \Pi_{elas} + \Pi_d + \Pi_s + \Pi_c \tag{3.36}$$

an extremum. Functionals Π_{elas}, Π_d and Π_s were defined in eqs.(3.14), (3.30) and (3.34) respectively. Π_c in eq.(3.36) is a penalty function term to impose the symmetry

$\sigma_{ij}=\sigma_{ji}$ by means of the constraint of stress functions eq.(3.21), which is

$$\Pi_c = \int_\Omega \rho^c \left[\frac{\partial F_x}{\partial y} - \frac{\partial F_y}{\partial x} \right]^2 d\Omega \tag{3.37}$$

where ρ^c is a penalty number. The first derivative of Π_c in an element Ω^e with respect to β_i^s can be expressed by

$$\frac{\partial \Pi_c}{\partial \beta_j^s} = \sum_{i=1}^{m} \mathbf{k}_{\mathbf{ij}}^{\mathbf{c}} \beta_i^s \tag{3.38}$$

where

$$\mathbf{k}_{\mathbf{ij}}^{\mathbf{c}} = \int_{\Omega^e} 2\rho_c \begin{bmatrix} N_{i,y}N_{j,y} & -N_{i,x}N_{j,y} \\ -N_{i,x}N_{j,y} & N_{i,x}N_{j,x} \end{bmatrix} d\Omega$$

With eqs.(3.18) and (3.24), it is straightforward to express the first derivative of Π_{elas} in any element Ω^e with respect to general nodal variables β_i in a matrix form

$$\frac{\partial \Pi_{elas}}{\partial \beta_j} = \sum_{i=1}^{m} \mathbf{k}_{\mathbf{ij}}^{\mathbf{e}} \beta_i \tag{3.39}$$

where

$$\mathbf{k}_{\mathbf{ij}}^{\mathbf{e}} = \int_{\Omega^e} \begin{bmatrix} 2G\mathbf{B}_{\mathbf{Di}}^{\mathbf{T}}\mathbf{B}_{\mathbf{Dj}} + K\mathbf{B}_{\mathbf{mi}}^{\mathbf{T}}\mathbf{B}_{\mathbf{mj}} & -\mathbf{B}_{\mathbf{i}}^{\mathbf{T}}\mathbf{A}_{\mathbf{j}} \\ -\mathbf{A}_{\mathbf{i}}^{\mathbf{T}}\mathbf{B}_{\mathbf{j}} & \mathbf{A}_{\mathbf{Di}}^{\mathbf{T}}\mathbf{A}_{\mathbf{Dj}}/2G + \mathbf{A}_{\mathbf{mi}}^{\mathbf{T}}\mathbf{A}_{\mathbf{mj}}/K \end{bmatrix} d\Omega \quad ; \quad \beta_i = \begin{Bmatrix} \beta_i^d \\ \beta_i^s \end{Bmatrix}$$

Because each item in eq.(3.36) is only the function of general nodal β_i, the minimum of Π_I requires

$$\frac{\partial \Pi_I}{\partial \beta_i} = \frac{\partial \Pi_{elas}}{\partial \beta_i} + \frac{\partial \Pi_d}{\partial \beta_i} + \frac{\partial \Pi_d}{\partial \beta_i} + \frac{\partial \Pi_c}{\partial \beta_i} = 0 \tag{3.40}$$

Substitute eqs.(3.31), (3.35), (3.37) and (3.39) into eq.(3.40), the global system equations can be expressed by a linear equation in terms of general nodal variables β

$$\mathbf{k}_\mathbf{I}\beta + \mathbf{F}_\mathbf{I} = 0 \tag{3.41}$$

where $\mathbf{k_I}$ takes the place of the well-known element stiffness matrix and $\mathbf{F_I}$ becomes the load vector of equivalent nodal forces of the element

$$\begin{aligned} \mathbf{k_I} &= \mathbf{k^e} + \mathbf{k^d} + \mathbf{k^s} + \mathbf{k^c} \\ \mathbf{F_I} &= \mathbf{F^d} + \mathbf{F^s} \end{aligned} \tag{3.42}$$

The elements of the matrices are described in eqs.(3.31), (3.35), (3.37) and (3.39) respectively. Expansion of the original matrices $\mathbf{k^d}$, $\mathbf{k^s}$ $\mathbf{k^c}$ are needed to construct $\mathbf{k_I}$ since the variables are neither β_i^d nor β_i^s but general variable β_i. Also it is worthwhile to mention that $\mathbf{k^d}$ and $\mathbf{k^s}$ exist only when the element has at least one node or one side at the boundary Γ. Otherwise they are replaced by zero.

The standard assembly procedure is used to form the global stiffness matrix and load vector element by element. Because the stiffness matrix is symmetric, positive-definite, eq. (3.41) can be easily solved by any efficient approaches used in conventional displacement method, e.g. the LDLT procedure. A variable bandwidth algorithm is used in the work described in this thesis.

An interesting feature of this model is that the value of the functional Π_{elas} is a direct measure of the accuracy of the solution in that region and therefore could be used to show where further refinement of the mesh might be appropriate. The reason for this is very simple. Because the equilibrium condition eq.(3.2) and compatible condition eq.(3.1) are exactly satisfied in advance, errors can only result from the inaccuracy of the stress-strain relation, eq.(3.5). There is no need to account for the error from eq.(3.3) since it can be reflected in the error of the stress-strain relation.

3.6 NUMERICAL TEST CASES

The examples presented below illustrate some features of this method when used to solve linear elastic problems in plane strain and plane stress. These include: very good performance when modelling essentially incompressible materials without recourse to reduced integration; the ability to cope with changes in material properties from element to element, which can not be handled by a normal continuous mixed model; and the existence of a natural error measure. The examples discussed are internal pressure in a hollow cylinder; an elastic but stiff punch pushed into a softer elastic material;

bending of a cantilever with transverse tip loading; and a ribbon crack under remote tension. For each case, the convergence of the model against the element size is also discussed.

3.6.1 Hollow Cylinder under Internal Pressure

Consider a hollow cylinder of internal radius a and external radius $2a$ in which there is an internal pressure P. For simplicity, the plane strain condition is assumed. Due to the symmetry, only 1/4 cylinder will be used in the finite element model. In the first instance a fine mesh with 50 8-noded quadrilaterals elements is used in discretizing the model, as shown in fig.3.3.

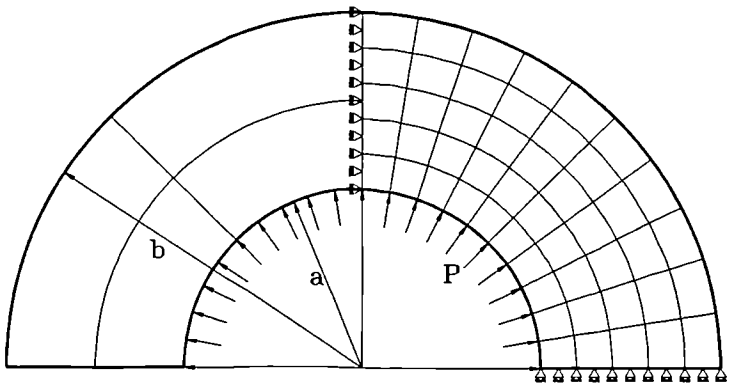


Fig. 3.3 Finite element models of a hollow cylinder under internal pressure.

In order to study the sensitivity of the mixed model with regard to the changes of Poisson's ratio, four values of Poisson's ratio ν=0.3, 0.4, 0.45 and 0.49 are used in the solutions.

Let us define the error of a FE solution as the difference between the analytical solution and the FE solution. Then the errors of the axial and hoop stresses at the Gauss points by the mixed method and the displacement method with 2×2 integration, are shown in fig.3.4(a)-(d). Both models show quite good agreement with the analytical solution for almost all values of ν. The errors of σ_{rr} and $\sigma_{\theta\theta}$ appearing in the figures are almost identical.

As Zienkiewicz mentioned in his review[126], the situation is quite different when 3×3 integration is used in solving the problem, as shown in fig.3.5(a)-(d). When

Poisson's ratio is getting bigger (>0.40), the mixed model has much better performance than the displacement model. In fig.3.5(d) where ν is as 0.49, the results from the displacement model are obviously meaningless, but those from the mixed model can remain quite similar even at ν=0.499999(not shown).

Before comparing the convergence of both models, a criterion needs to be given. In this case, the ***average error*** is introduced as the criterion

$$\epsilon_{av} = \frac{1}{2m}\sum_{i=1}^{m}(\epsilon_{rr}^{i} + \epsilon_{\theta\theta}^{i}) \tag{3.43}$$

where m is the number of integral points along the r-axis. $\epsilon_{rr}{}^{i}$ and $\epsilon_{\theta\theta}{}^{i}$ are the relative errors of the radial and hoop stresses at a particular Gauss point.

$$\epsilon_{rr}^{i} = |(\sigma_{rr}^{FEM} - \sigma_{rr}^{exact})|/P \quad ; \quad \epsilon_{\theta\theta}^{i} = |(\sigma_{\theta\theta}^{FEM} - \sigma_{\theta\theta}^{exact})|/P \tag{3.44}$$

Four meshes with 2, 3, 4 and 5 elements along r-axis are used to obtain the average errors. In fig.3.6 for 2×2 integration, the convergence of the two models is similar for different Poisson's ratios. However the mixed model has much lower errors than those of displacement model for every mesh when 3×3 integration is used, as shown in fig.3.7. In this case, the convergence of the mixed model remains insensitive to ν, while that of the displacement model is affected severely by Poisson's ratio. The larger is ν, the more severe the accuracy of the solution by the displacement model depends on element size.

In fig.3.8 and fig.3.9, the average values of functional Π_{elas} in the mixed model defined by

$$\Pi_{av} = \frac{1}{m}\sum_{i=1}^{m}\Pi_{elas}^{i}/\sigma^{T}e \tag{3.45}$$

are plotted together with the corresponding average errors. It is interesting to find that the figures of Π_{elas} show similar patterns to those of average errors calculated directly from the stress for each ν and integration. Therefore it is fair to say that the value of the Π_{elas} do indicate local errors at some degree.

The superiority of the 2×2 integration seems not to be a disadvantage as it needs less computing time. However, in the next example we will show an example in which 2×2 integration can cause problems when a very coarse mesh is used.

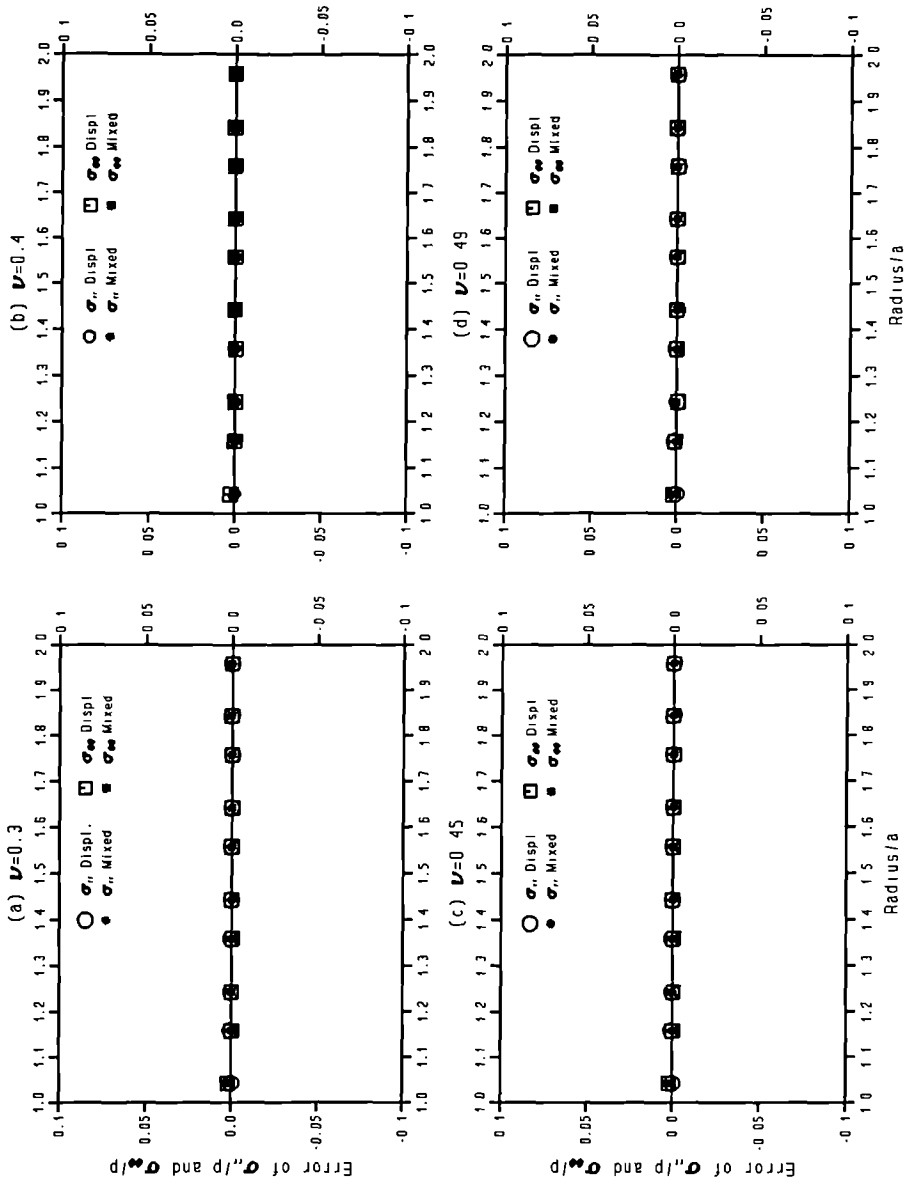


Fig.3.4 The average errors of σ_{rr} and $\sigma_{\theta\theta}$ obtained by the mixed model and the displacement model under different Poisson's ratios with 2x2 integration.

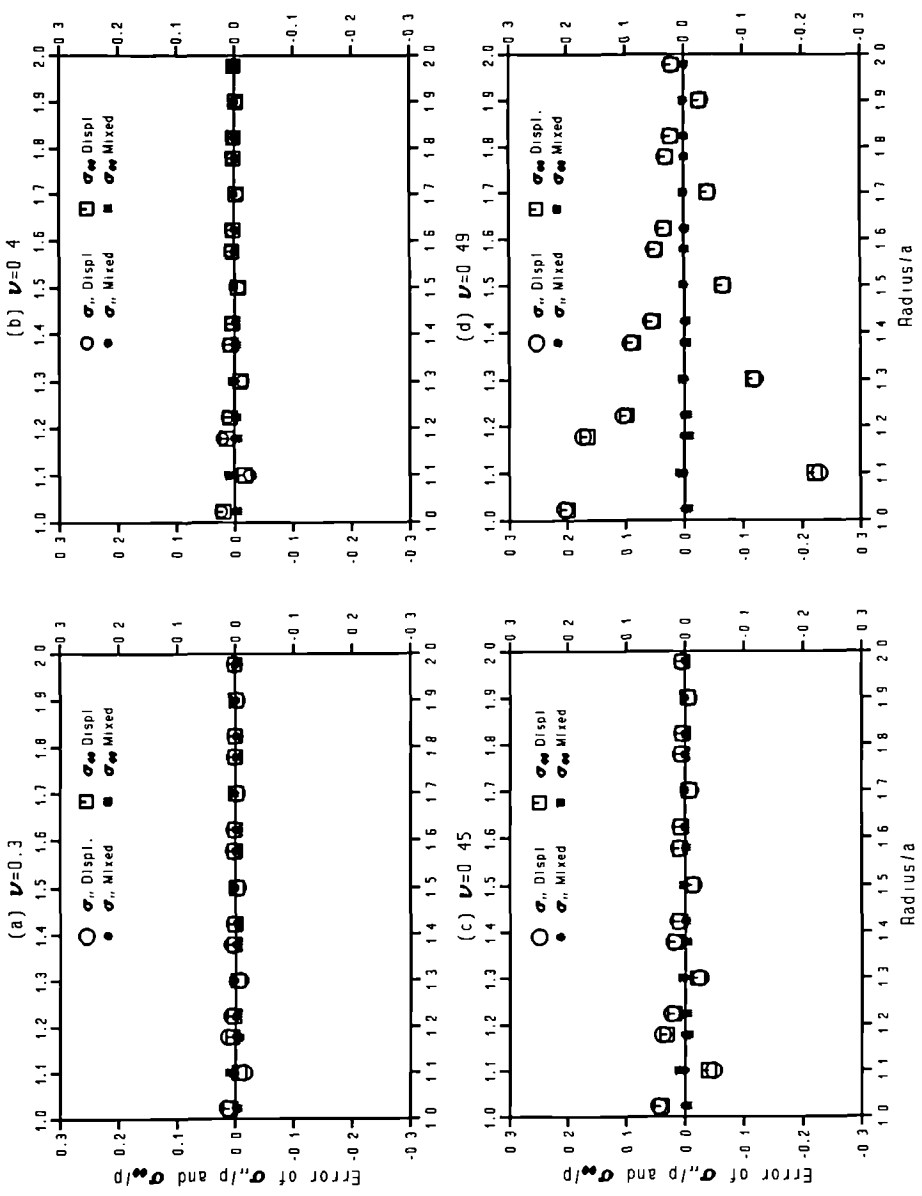


Fig.3.5 The average errors of σ_{rr} and $\sigma_{\theta\theta}$ obtained by the mixed model and the displacement model under different Poisson's ratios with 3x3 integration.

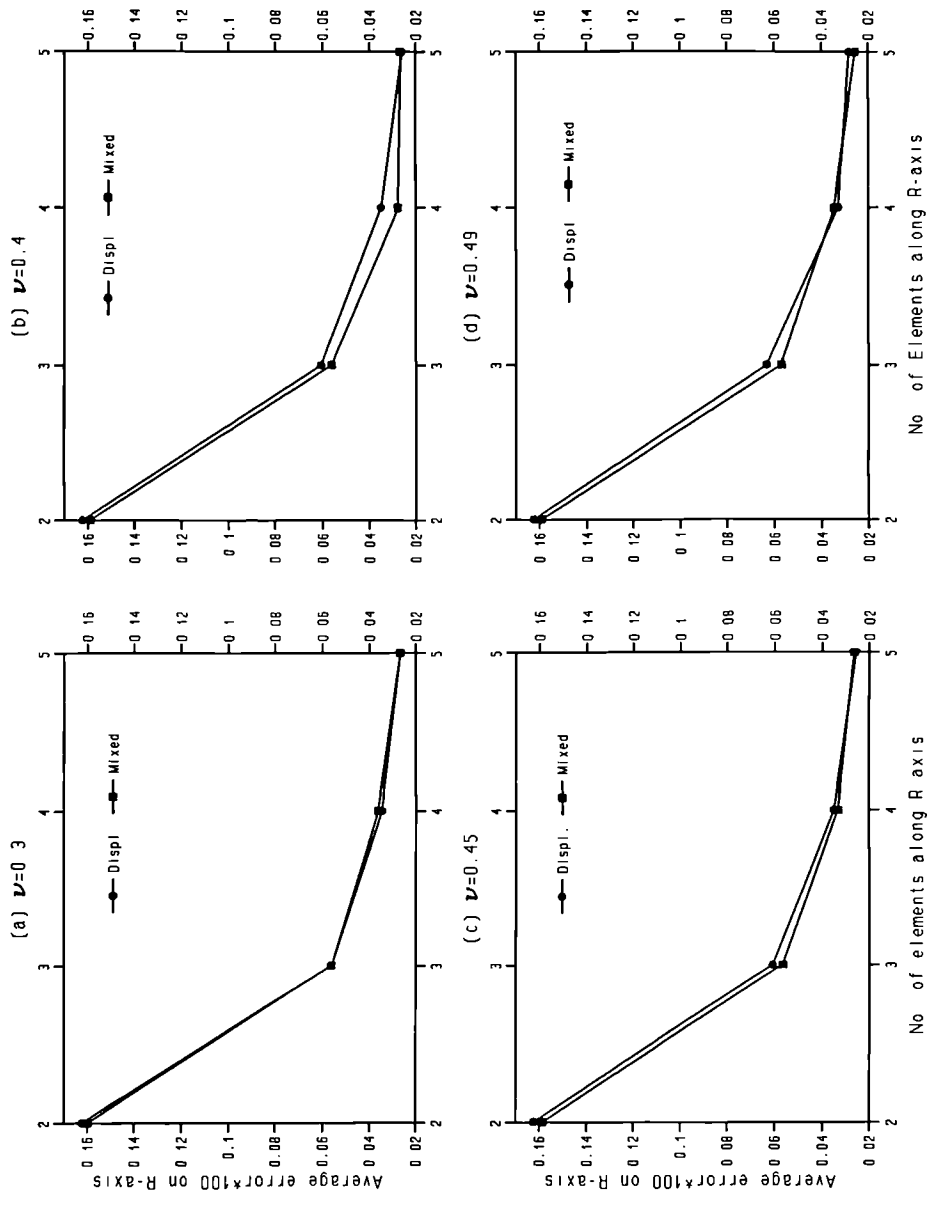


Fig.3.6 Convergencies of the mixed model and the displacement model in terms of the average errors against number of elements on *r*-axis when 2x2 integration is used.

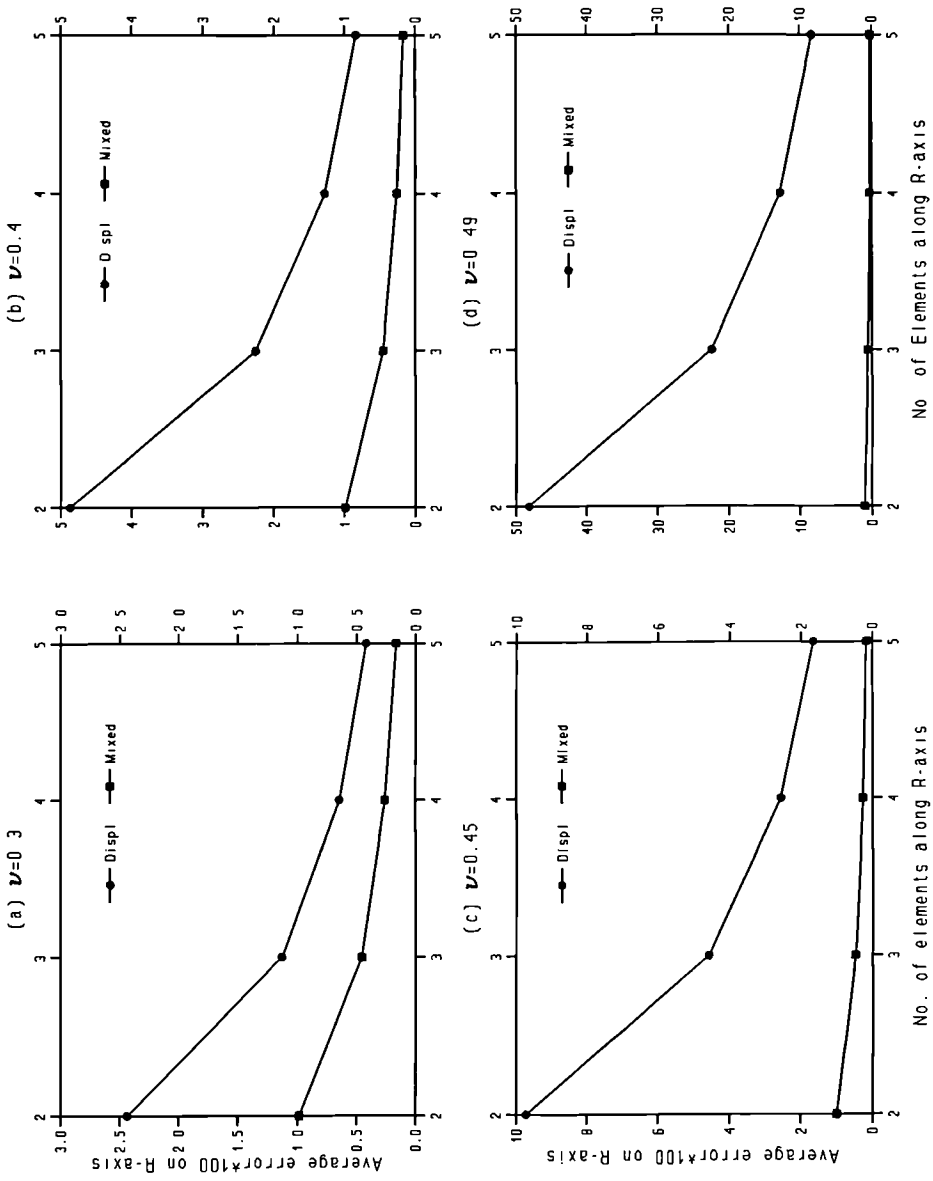


Fig.3.7 Convergencies of the mixed model and the displacement model in terms of the average errors against number of elements on r-axis when 3x3 integration is used.

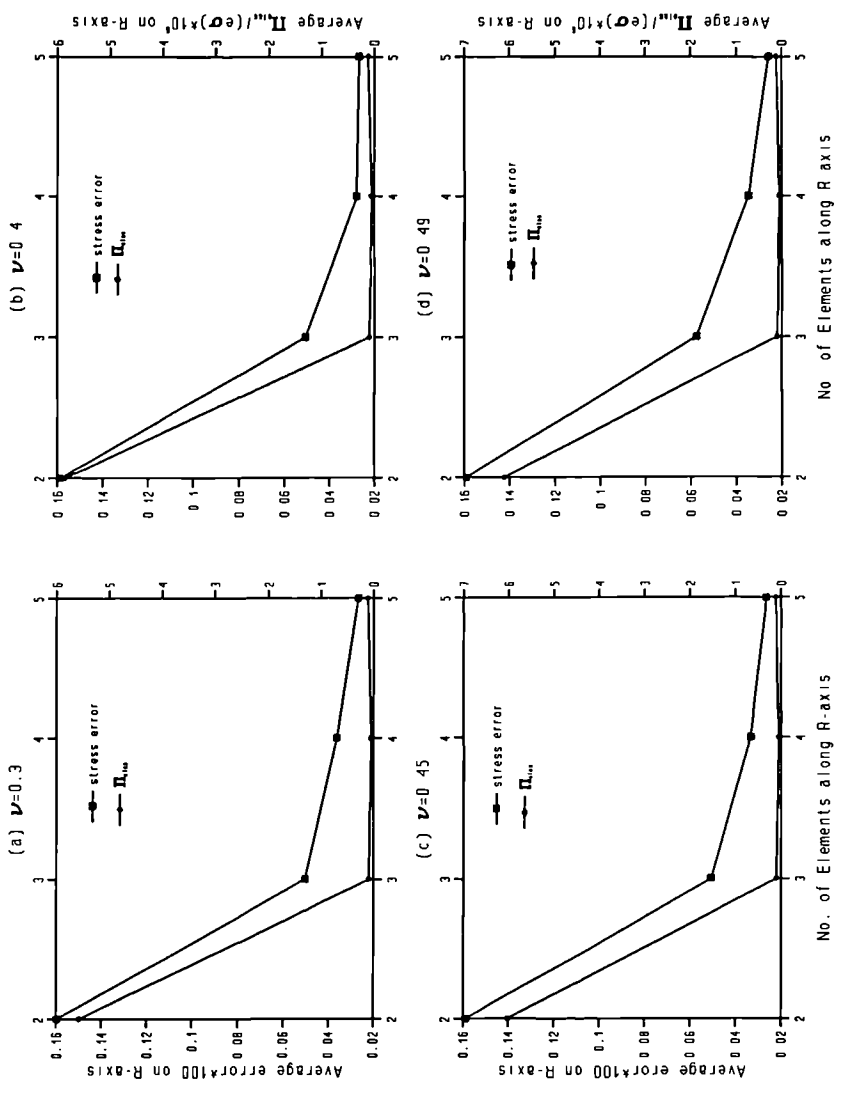


Fig.3.8 Comparisons of the average errors and the average value of Π_{elas} varying with the number of elements on r-axis in the mixed model when 2x2 integration is used.

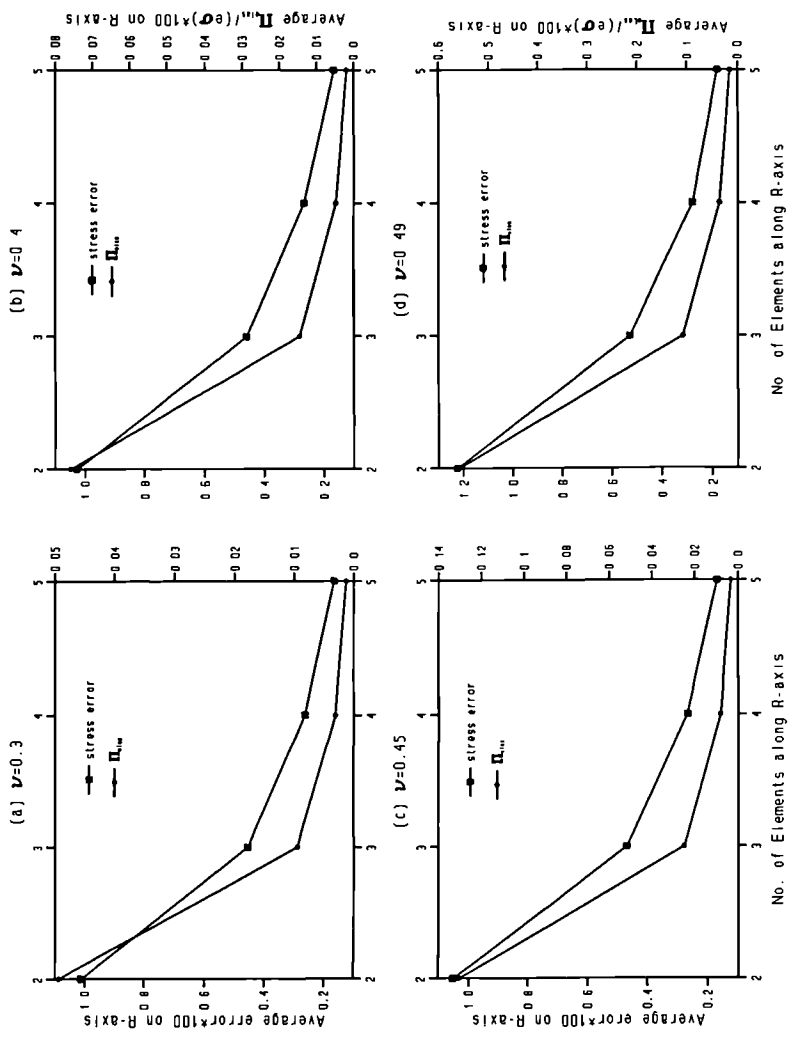


Fig.3.9 Comparisons of the average errors and the average value of Π_{elas} varying with the number of elements on r-axis in the mixed model when 3x3 integration is used.

3.6.2 Elastic Punch Pressing into a Compliant Layer

Consider a punch made from an elastic material with Young's modulus E_p and Poisson's ratio ν. It presses into, allowing no slip at the interface, a linear elastic material with Young's modulus E_b and the same Poisson's ratio. The ratio E_p/E_b is 10000. Two different meshes of FE models are shown as fig.3.10, where 7 and 34 elements of 8-noded quadrilaterals are used in each model. Solutions are carried out under the Plane strain assumption with both 3×3 and 2×2 Gaussian integration.

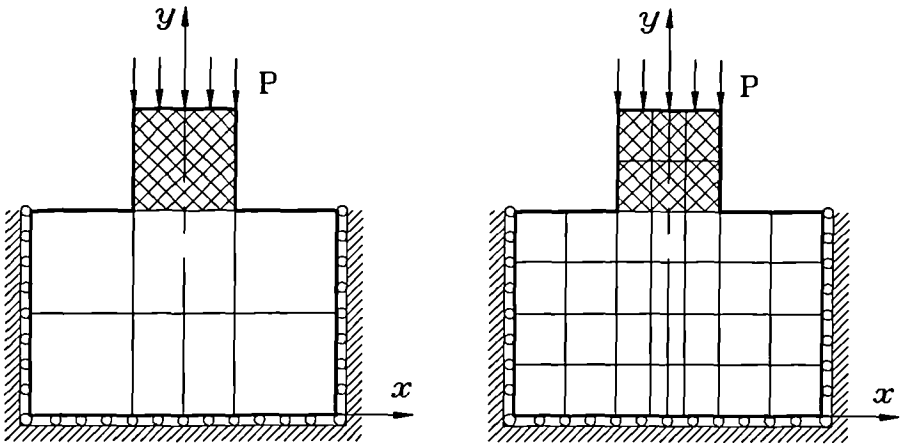


Fig.3.10 FE models for analysis of a punch pressing into a layer.

First let us look at the deformation modelled by either model. In fig.3.11 the displacement model with a coarse mesh gives poor displacement results (deformation of the mesh) for $\nu = 0.3$ and $\nu = 0.49$ when 2×2 integration is used: the deformation of the punch should be rather smaller. A similar problem is found in the solution of the mixed model under 2×2 integration, as shown in fig.3.12. When 3×3 integration is used in the solutions, both models yield sensible predictions for deformation in the case of either Poisson' ratio, as shown in fig.3.13 and fig.3.14.

Now let us turn our attention to the stress results. There is a little problem in judging the stress results since an exact solution of such a problem is normally not available. However by comparing the consistency of the results under different integration schemes and different mesh size, the performance of each model for this kind of problems with a tremendous change of the material properties can be studied at some degree.

In the case of $\nu = 0.49$, seen in fig.3.15(c) and fig.3.15(d), σ_{xx} and σ_{yy} obtained from mixed model show consistency between 2×2 and 3×3 integrations. But this is not found in the case of the displacement model. As a matter of fact, the results from the displacement model under 3×3 integration become meaningless. Although at $\nu = 0.3$, results from the displacement model are much better, there still exists a inconsistency between results obtained under different integrations, as seen in fig.3.15(a) and fig.3.15(b).

One explanation of this is that in the displacement model, stresses are calculated from the displacements, which are different under 2×2 and 3×3 integrations, as shown in fig.3.11(a) and fig.3.13(a). Although a similar problem exists in the deformations from the mixed model (fig.3.11(a) and fig.3.13(a)), it does not affect the stresses since they are calculated directly from the nodal stress functions. Thus stresses by the mixed model under different integrations are almost identical, as seen in fig.3.15(a) and fig.3.15(b).

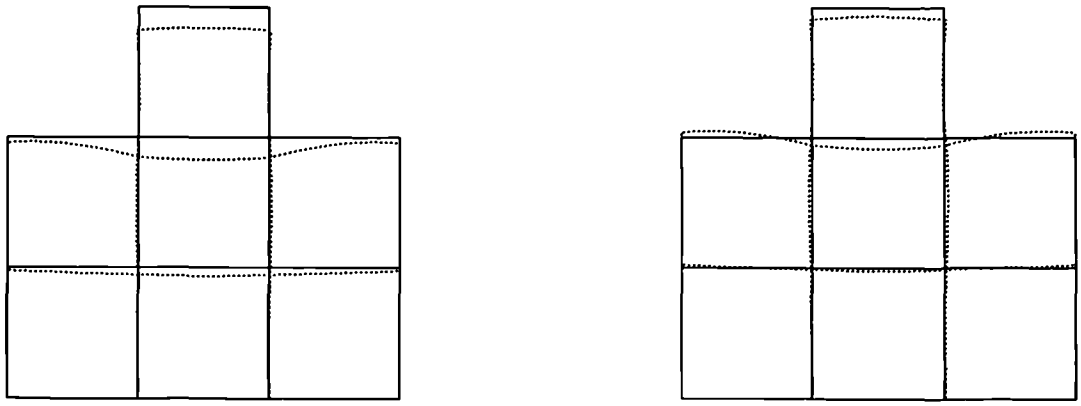
Problems of this type with a coarse mesh can create problems for the displacement method because they give poor stresses when 3×3 integration is used and spurious modes of deformation when 2×2 integration is used[126]. The mixed method also gives poor displacements but good stresses when reduced integration is used, while both good stresses and good displacements are obtained when 3×3 integration is used.

There is one solution for this problem, which is to use finer mesh. In fig.3.16, fig.3.17, fig.3.18 and fig.3.19, deformations predicted by both the displacement model and the mixed model with 34 elements mesh are shown under two Poisson's ratio $\nu = 0.3$ and $\nu = 0.49$ as well as 2×2 and 3×3 integrations. No spurious modes occur under 2×2 integration from either model. This offers the displacement model a chance to give both good displacements and stresses by 2×2 integration at $\nu = 0.49$, as seen in fig.3.16(b) and fig.3.20(c).

At $\nu = 0.49$, the stresses from either model are reasonably close, shown in fig.3.20(a) and fig.3.20(b). A "kink" exists in σ_{yy} obtained from the displacement model, while the mixed model gives a smoother curve. In the case of $\nu = 0.49$ with 3×3 integration (fig.3.20(d)), the displacement model gives bad stress results again.

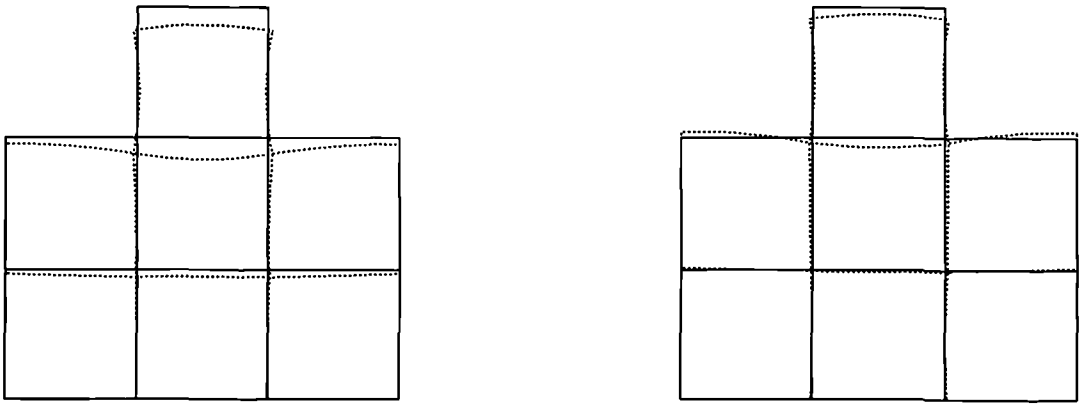
Once again, the mixed model shows its insensitivity against Poisson's ratio and the integration schemes. All four graphs in fig.3.20 are in a very similar pattern.

The discontinuity of σ_{xx} in this problem can also be clearly identified in fig.3.15 and fig.3.20. This means the mixed model can cope with "stress jumps" across a boundary where material properties change rapidly.



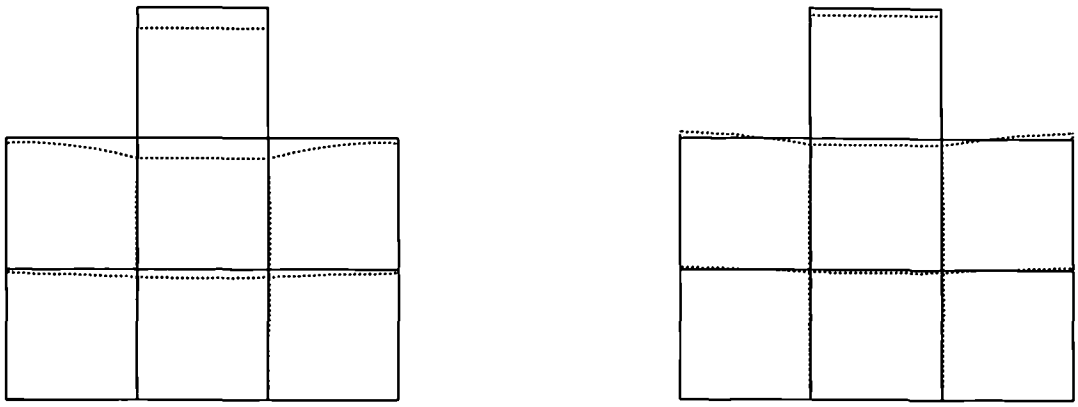
(a) (b)

Fig. 3.11 Deformations obtained from the displacement model by a FE mesh with 7 elements when 2×2 integration is used. (a): $\nu=0.3$, (b):$\nu=0.49$.



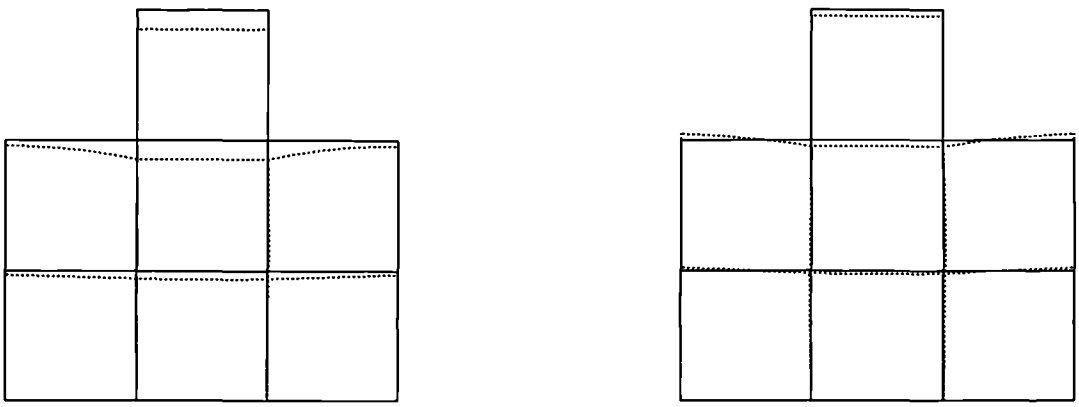
(a) (b)

Fig. 3.12 Deformations obtained from the mixed model by a FE mesh with 7 elements when 2×2 integration is used. (a): $\nu=0.3$, (b):$\nu=0.49$.



(a) (b)

Fig. 3.13 Deformations obtained from the displacement model by a FE mesh with 7 elements when 3×3 integration is used. (a): $\nu=0.3$, (b):$\nu=0.49$.



(a) (b)

Fig. 3.14 Deformations obtained from the mixed model by a FE mesh with 7 elements when 3×3 integration is used. (a): $\nu=0.3$, (b):$\nu=0.49$.

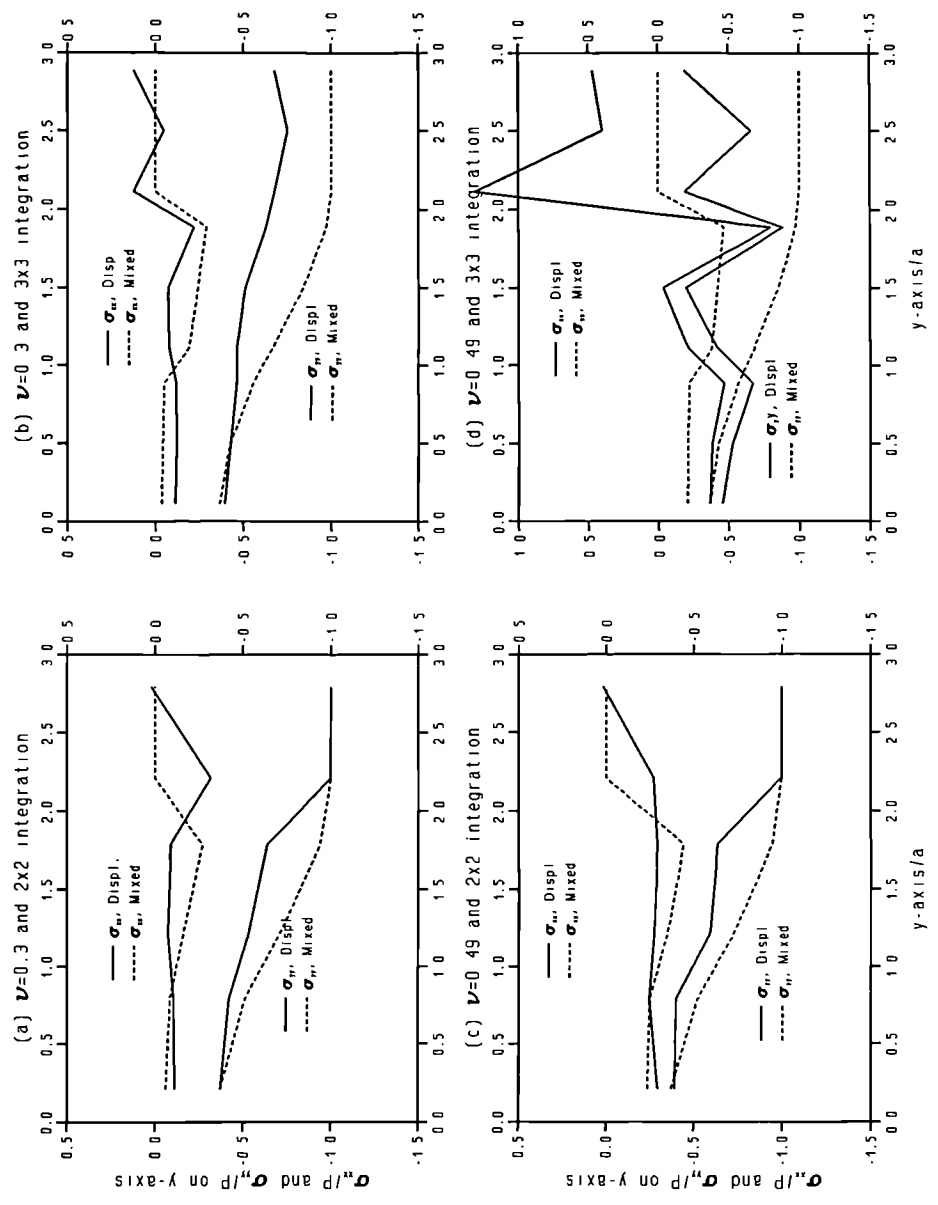
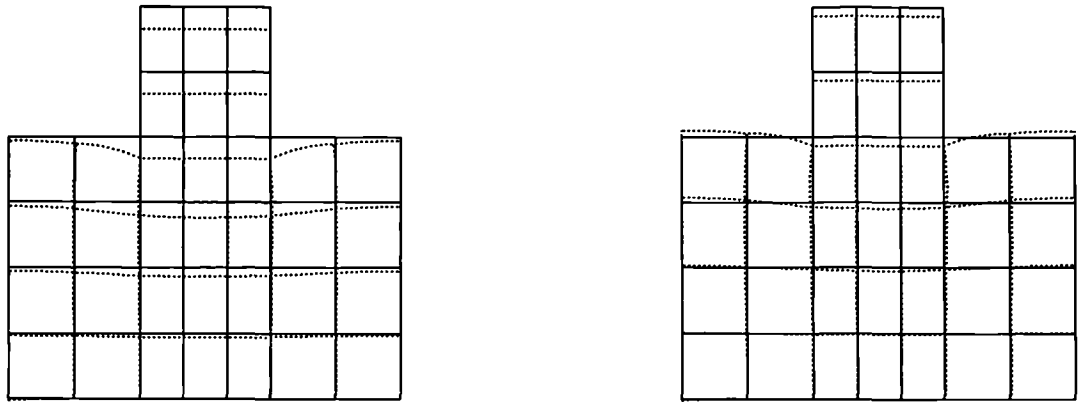
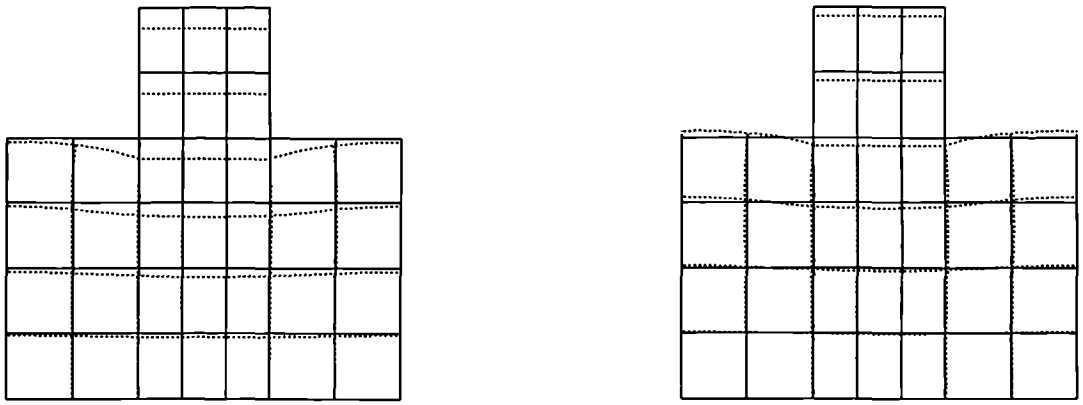


Fig.3.15 The distributions of σ_{xx} and σ_{yy} along y-axis by the mixed model and the displacement model with 7 elements.



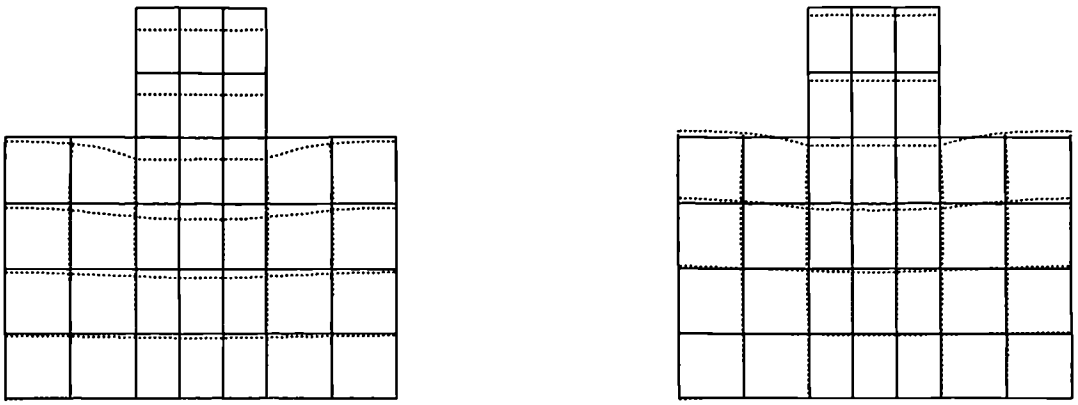
(a) (b)

Fig. 3.16 Deformations obtained from the displacement model by a FE mesh with 34 elements when 2×2 integration is used. (a): $\nu=0.3$, (b):$\nu=0.49$.



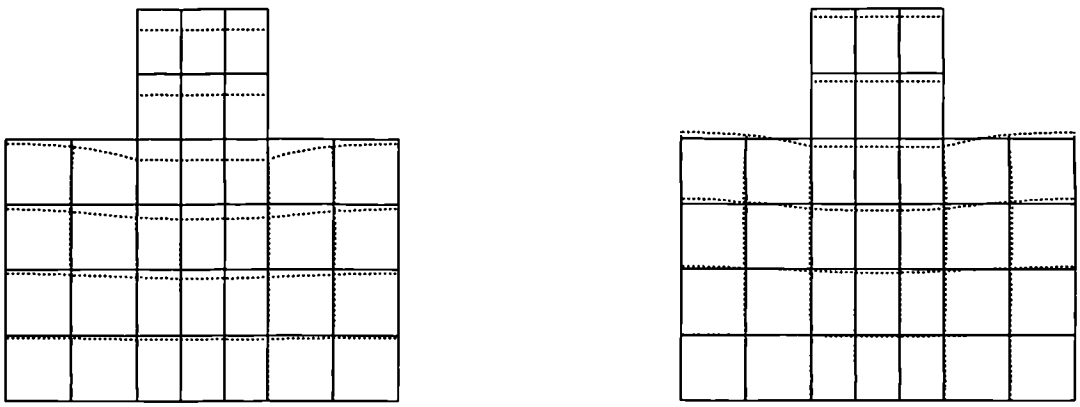
(a) (b)

Fig. 3.17 Deformations obtained from the mixed model by a FE mesh with 34 elements when 2×2 integration is used. (a): $\nu=0.3$, (b):$\nu=0.49$.



(a) (b)

Fig. 3.18 Deformations obtained from the displacement model by a FE mesh with 34 elements when 3×3 integration is used. (a): $\nu=0.3$, (b):$\nu=0.49$.



(a) (b)

Fig. 3.19 Deformations obtained from the mixed model by a FE mesh with 34 elements when 3×3 integration is used. (a): $\nu=0.3$, (b):$\nu=0.49$.

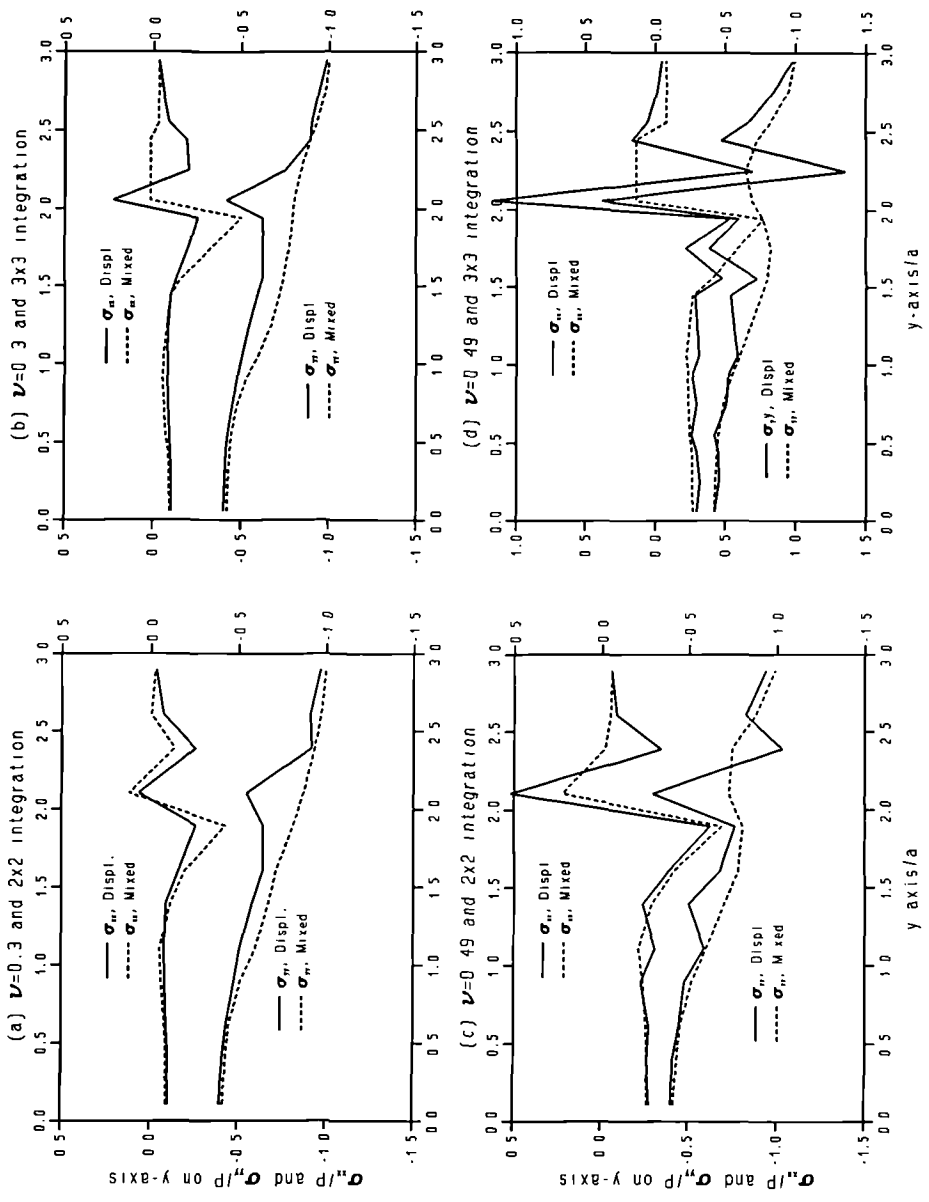


Fig.3.20 The distributions of σ_{xx} and σ_{yy} along y-axis by the mixed model and the displacement model with 34 elements.

3.6.3 Plane with a Central Crack under Remote Tension

It is easy to anticipate that the mixed model will produce more accurate stress results than a displacement model. This is particularly true when stress changes tremendously, such as near a crack or in laminated composite materials, as reviewed in chapter 2. In this section, an analysis of a plane with a central crack with length $2a$ under remote tension will be made to demonstrate the performance of our mixed model when applied to a case where stress singularity exists.

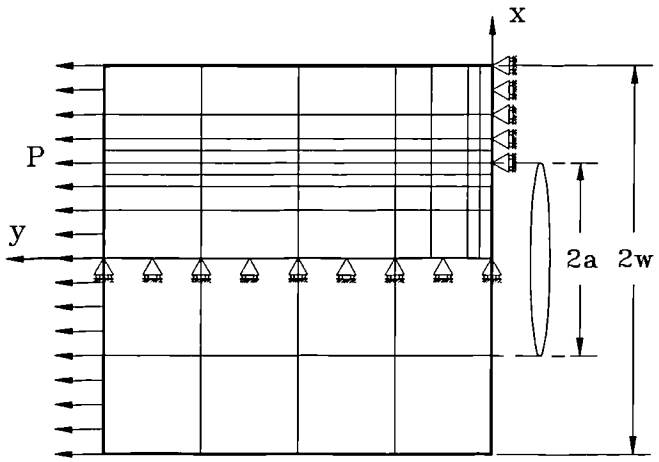


Fig.3.21 FE model for a plane with a central crack.

The finite element model used in this problem is shown in fig.3.21. Only one quarter of the plate is presented because of the symmetry. Four different meshes with 8, 16, 37 and 56 8-noded quadrilateral elements were used in each model and the solutions were obtained with 3×3 Gaussian integration in the plane stress condition. Poisson's ratio was $\nu = 0.3$ and a unique distributed tension at remote end of the plate was P. In fig.3.22(a)-(d) results of σ_{yy} obtained by both the displacement model and the mixed model for four different meshes are presented, as well as an analytical solution.

In order to investigate the basic properties of each model in coping with the singularity of stress, no dislocation of the mid-node of the elements around the crack tip was made in either model in the first instance. It is seen that in each mesh, the mixed model always gives the better prediction of σ_{yy} at the nearest point from the crack tip comparing with the exact solution[5], while the displacement model behaves clearly poorer at this location. The gradient of the stress given by the displacement

model is less than that by the mixed method in the area where singularity of the stress exists. This implies that the mixed model can represent the singularity of stress more accurately than the conventional displacement method in the case of normal element (without mid-node moved).

However if the mid-node of the element surrounding the crack tip is moved by one quarter toward crack tip, we shall call them *distorted element* here, the results from displacement model are improved significantly and are superior to those from the mixed method. The improvement in the mixed model is not that outstanding, as shown in fig.3.23(a)-(d). This may be explained as that the singularity of the stress functions caused by moving mid-node of the element is not as obvious as it for the displacements.

Near the edges of the plate, both models converge together, but deviate from the analytical solution. This is because the analytical solution in ref [5] is only correct near the crack tip.

Now let us see if there is any relation between the value of $\Pi_{elas}/\sigma e^{T}$ and the error calculated by $\epsilon = (\sigma_{yy}^{FE} - \sigma_{yy}^{a})/P$ at a given point. The point is selected at $(2a/8, 2a/8)$ which is on 45° from the crack line and $2a\sqrt{2}/8$ from the crack tip. In fig.3.24, the ϵ and $\Pi_{elas}/\sigma e^{T}$ are plotted against crack length $2a$ / minimum element's size. It appears that both curves almost coincide: the higher value of $\Pi_{elas}/\sigma e^{T}$, the larger value of ϵ. When element's size becomes very small, both values are approaching zero.

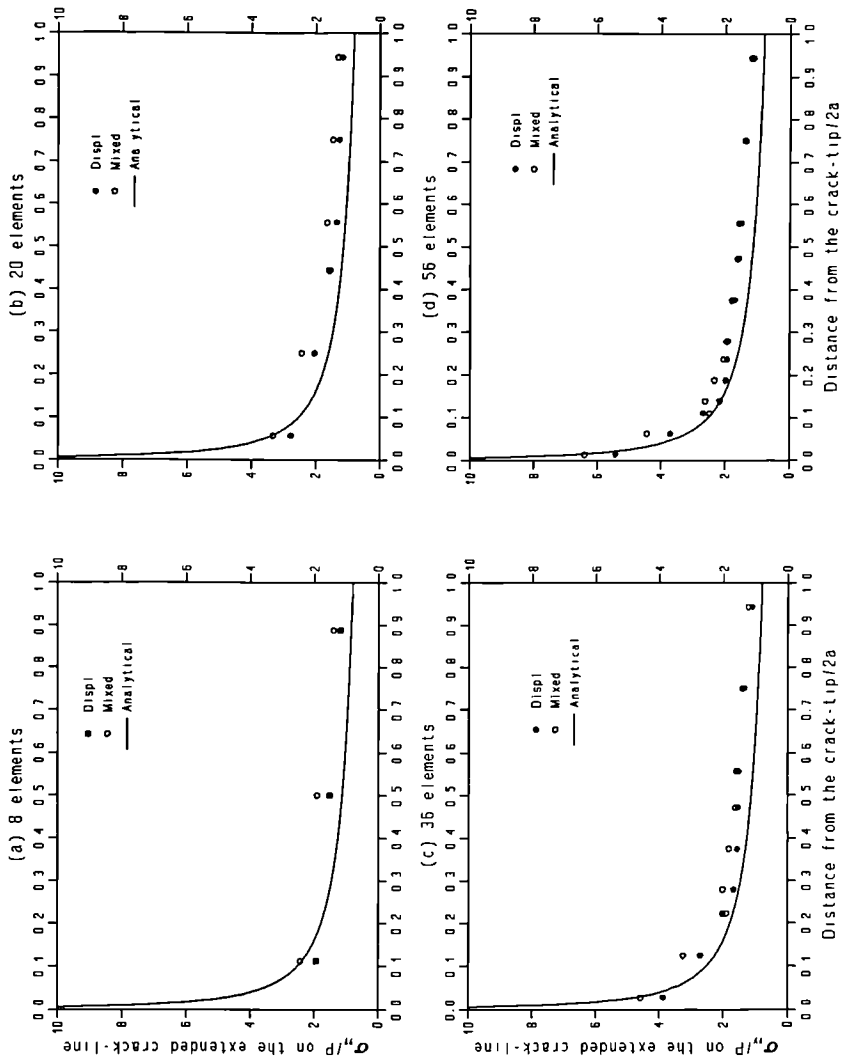


Fig.3.22 σ_{yy} along the extended crack-line by the displacement model and the mixed model, both with 3x3 integration, as well as analytical solution. In both FE models, no distorted element is used.

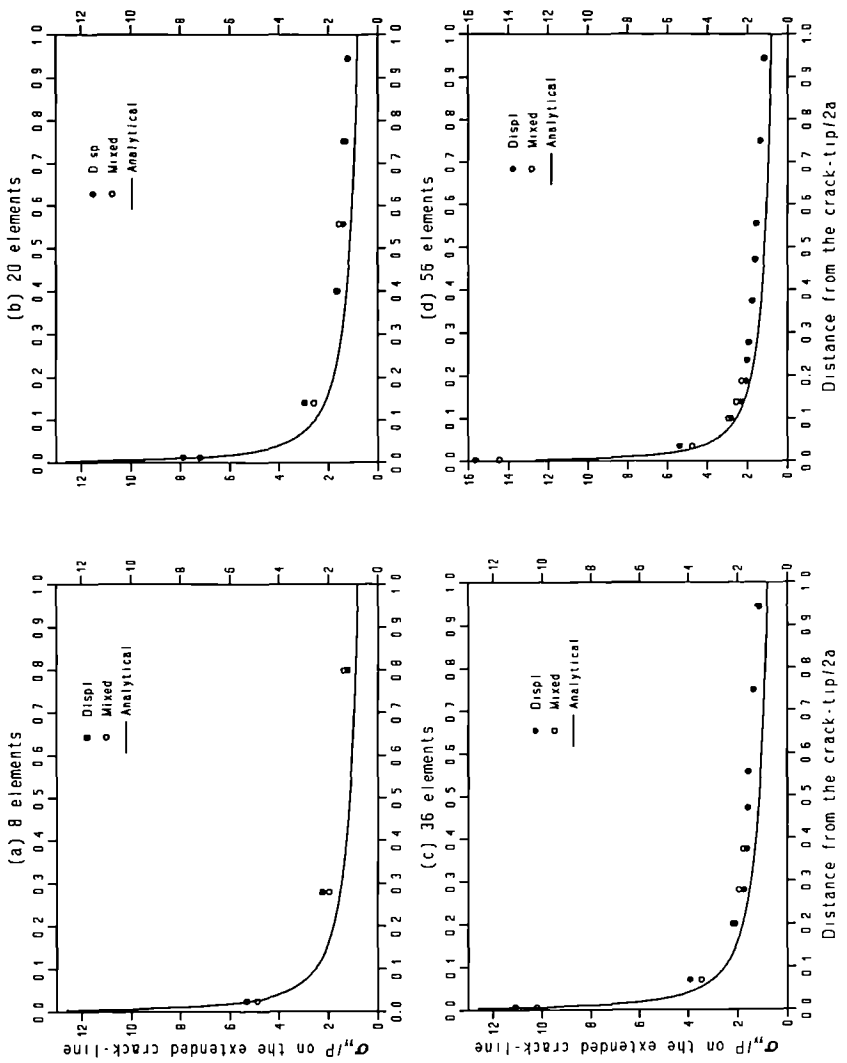


Fig.3.23 σ_{yy} along the extended crack-line by the displacement model and the mixed model, both with 3x3 integration, as well as analytical solution. In both FE models, distorted elements are used.

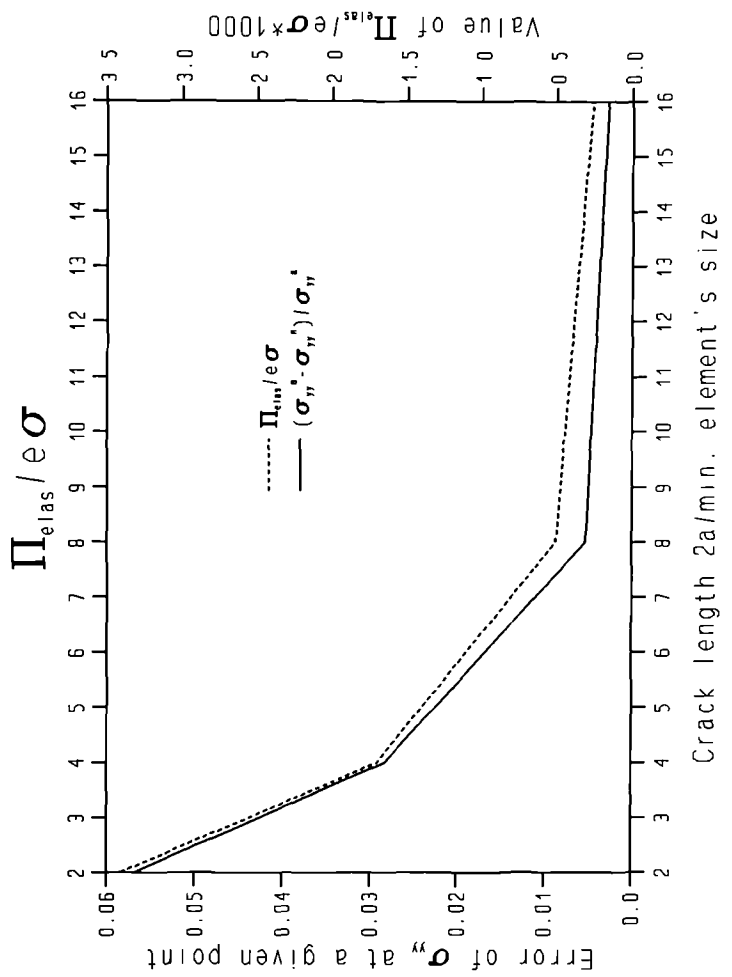


Fig.3.24 The error of σ_{yy} and the value of $\Pi_{elas}/\sigma e^{T}$ at given point varying with the element size when 3x3 integration is used.

3.6.4 Beam Bending under Transverse Loading at End

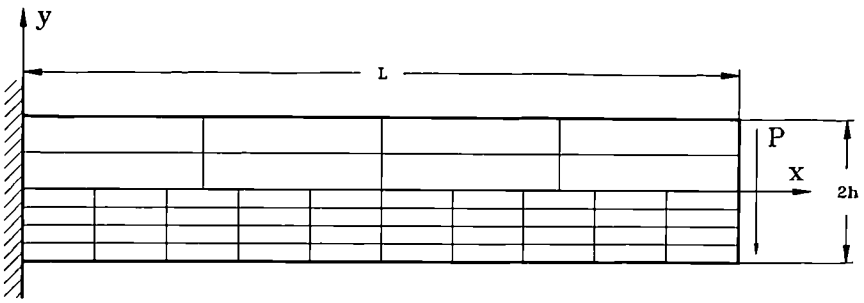


Fig. 3.25 The FE models of the beam bending by transverse load.

The difficulty faced by this mixed model is that the stress obtained will not be precisely symmetric, i.e. shear stress σ_{ij} will not be equal to σ_{ji} since the assumed stress only satisfies the force equilibrium eq.(3.2), but does not satisfy the moment equilibrium eq.(3.3). The problem will become more severe if shear is dominating the deformation, such as in beam bending under transverse loading. We have proposed, in chapter three, a penalty function eq.(3.37) to overcome this problem. In order to study the efficiency of the penalty function and the mixed model, a beam bending is studied below.

Two FE meshes with 4×4 and 8×10 of 8-noded plane stress elements are presented as in fig.3.25. Two set of different values of ρ_c are used, i.e. ρ_c = 0, 1.0, 5.0, 10.0 for 2×2 integration and ρ_c = 0, 1.0, 10.0, 50.0 for 3×3 integration where ρ_c = 0 corresponds to Chandler's work[15].

In the case of 4×4 mesh, the results of σ_{xx} cross the section at x/L=0.5 are presented in fig.3.26 with 2×2 integration and fig.3.27 with 3×3 one. It is seen that with 2×2 one, mixed model gives reasonable when $\rho_c \geq 5.0$. However $\rho_c \geq 10.0$ is required to reach the similar accuracy when 3×3 integration is used. A similar situation is also found in the results of the shear stress σ_{xy} and σ_{yx} in fig.3.28 and fig.3.29. When penalty number ρ_c increase, the inequality between σ_{xy} and σ_{yx} disappears. It is interesting to find that if we take the simple average $0.5\times(\sigma_{xy}+\sigma_{yx})$ as the approximate shear stress, then very good results can be obtained at ρ_c=1.0 for 2×2 case and ρ_c=10.0 at 3×3 case, and these are both better than those obtained from the displacement model.

In the case of the 8×10 mesh, both models give better results and the smaller penalty number is required to obtain results of σ_{xx}, σ_{xy} and σ_{yx} at the similar accuracies as at 4×4 mesh. In fig.3.30 and fig.3.32, results of σ_{xx} and $0.5\times(\sigma_{xy}+\sigma_{yx})$ under 2×2 integration are quite good compared with the exact solution at $\rho_c \geq 1.0$, while those under 3×3 one are very good at $\rho_c \geq 10.0$, as shown in fig.3.31 and fig.3.33.

In fig.3.34 to fig.3.37 the deflections of the beam are given under both meshes and different ρ_c. The accuracy of the results also greatly depends on the penalty number. $\rho_c \geq 5.0$ is enough to get good results for 4×4 mesh (fig.3.34) and $\rho_c \geq 1.0$ for 8×10 mesh (fig.3.36), both with 2×2 integration, which are similar to the stress calculations. However the case of 4×4 with 3×3 integration requires larger penalty number e.g. $\rho_c \geq 10.0$ to get reasonable deflection, as seen in fig.3.35, where only $\rho_c \geq 5.0$ is needed for the stress calculation. The deflection obtained by 8×10 mesh with 3×3 integration requires similar ρ_c as in the stress ones, as seen fig.3.37.

In all cases, 8×10 mesh gives better results than those from 4×4 and $\rho_c=5.0$ is generally required.

3.6.5 Comments on the test cases

In previous sections of this chapter, a few problems are solved by the mixed model in order to justify the mixed extremum principle along with the C^0 stress functions. In comparison with an exact solution, the mixed model often shows superior properties over displacement one. It gives better stress results in the nearly incompressible materials (section 3.6.1). It represents stress singularity more accurately (section 3.6.2), although most mixed models do. It handles the "stress jump" at the interface of two materials correctly (section 3.6.3), which is normally not the case for continuous mixed models. Although the mixed model may have potential difficulties when applied to the cases where shear is important, these can be entirely overcome by introducing the penalty term eq.(3.37) (section 3.6.4) to reinforce the symmetry of the stress tensor. The most interesting feature of this mixed model is that it has a natural "error estimator" which seems very meaningful and is simple to calculate. Therefore it is fair to say that the newly established mixed extremum principle is successful and the stress discretization used in this chapter is efficient to many plane problems.

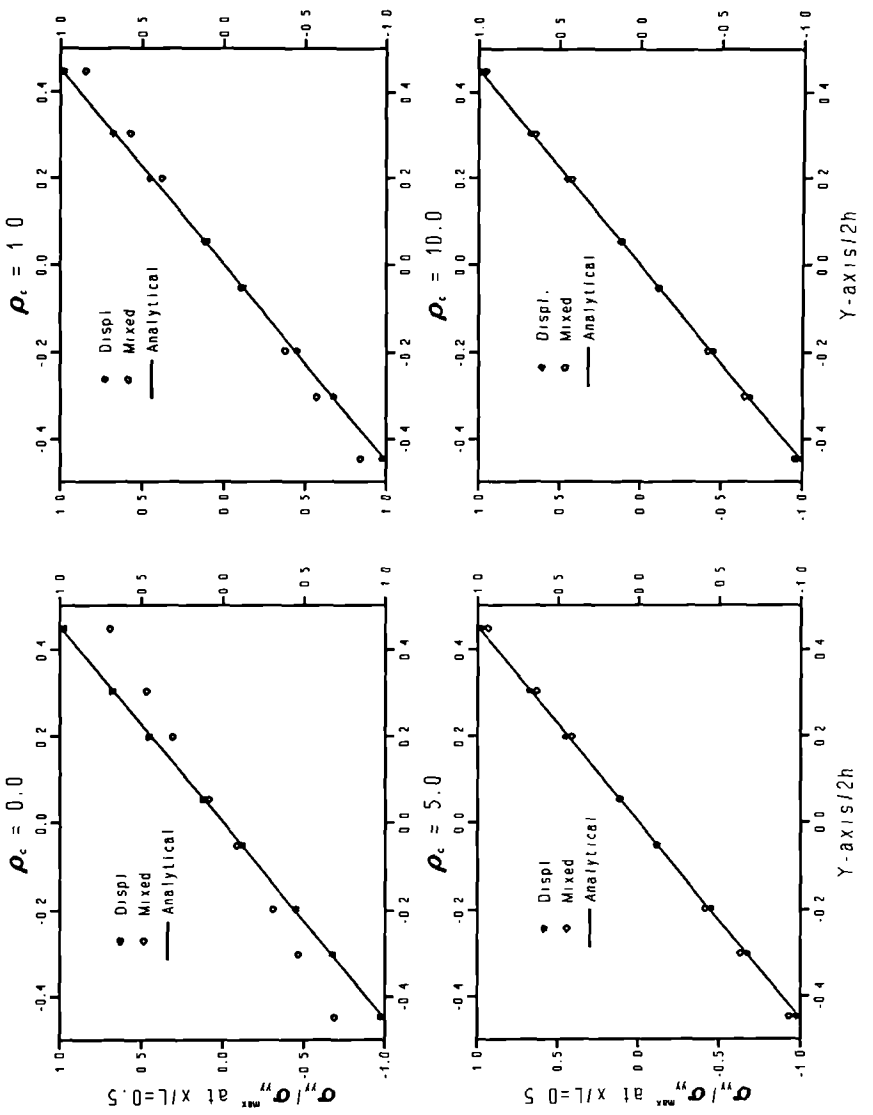


Fig.3.26 Distributions of σ_{xx} along *y*-axis at $x/L=0.5$ in a beam under an end loading. The results are obtained by a FE mesh with 16 elements and 2x2 integration.

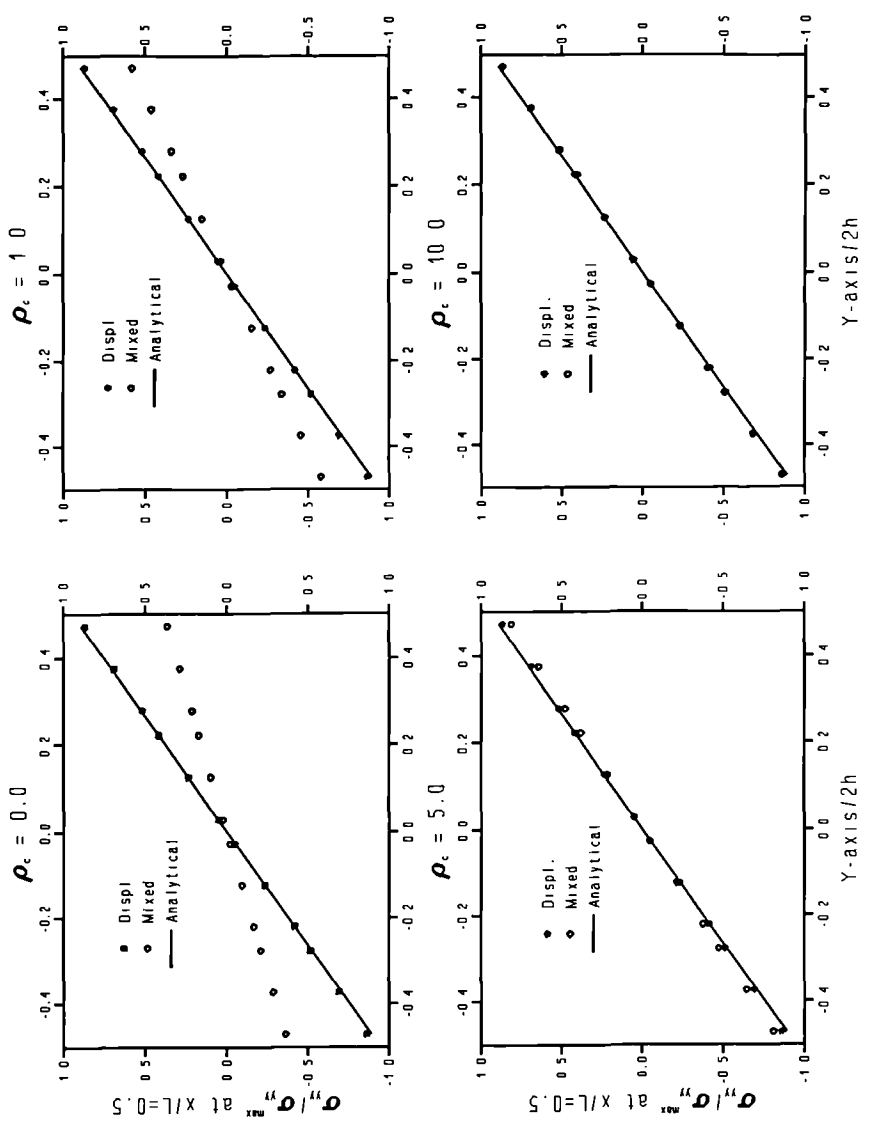


Fig.3.27 Distributions of σ_{xx} along y-axis at x/L=0.5 in a beam under an end loading. The results are obtained by FE mesh with 16 elements and 3x3 integration.

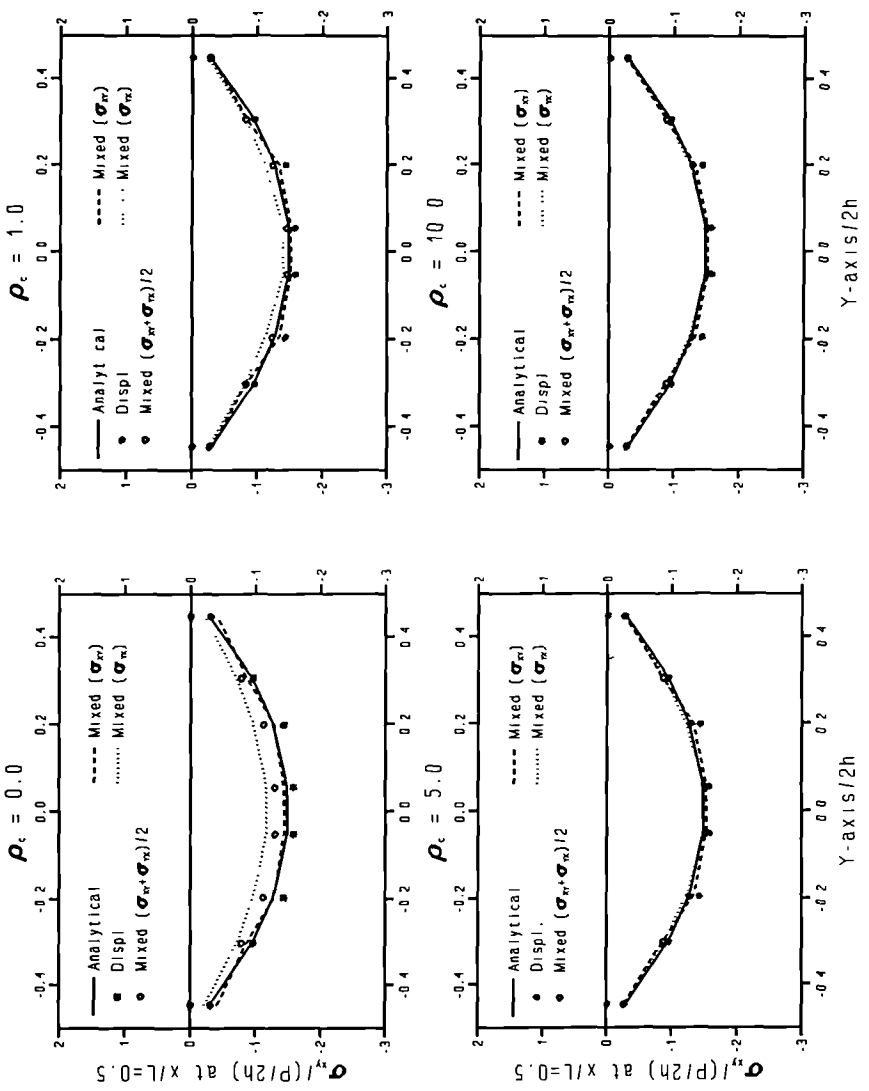


Fig.3.28 Distributions of σ_{xy} and σ_{yx} along y-axis at x/L=0.5 in a beam under an end loading. The results are obtained by a FE mesh with 16 elements and 2x2 integration.

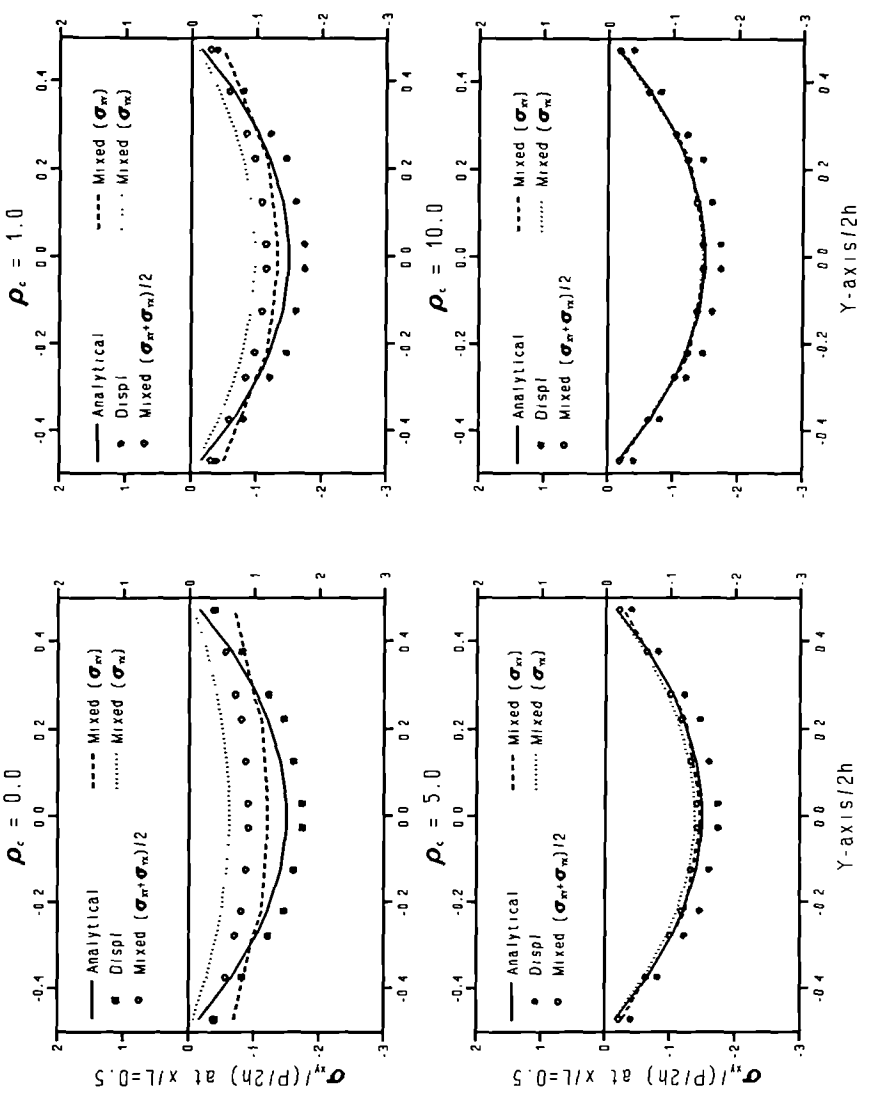


Fig.3.29 Distributions of σ_{xy} and σ_{yx} along y-axis at x/L=0.5 in a beam under an end loading. The results are obtained by a FE mesh with 16 elements and 3x3 integration.

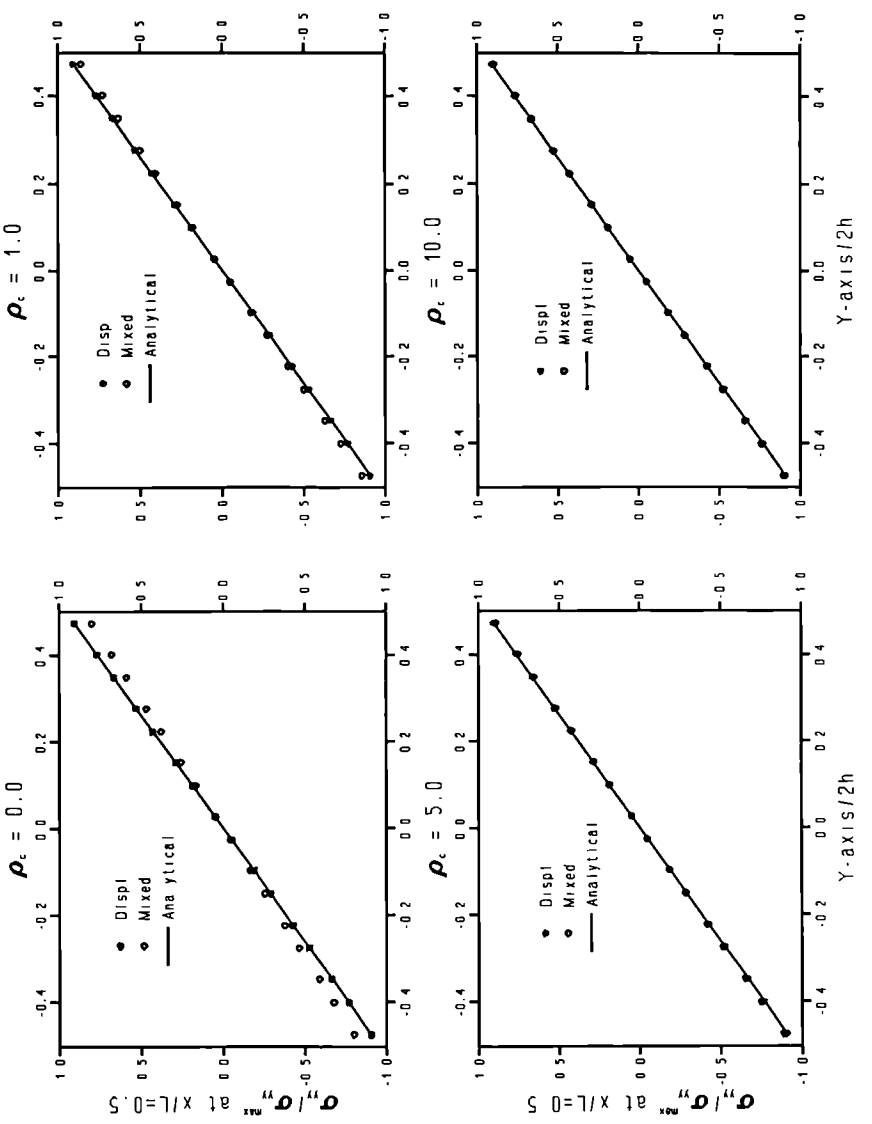


Fig.3.30 Distributions of σ_{xx} along y-axis at x/L=0.5 in a beam under an end loading. The results are obtained by a FE mesh with 80 elements and 2x2 integration.

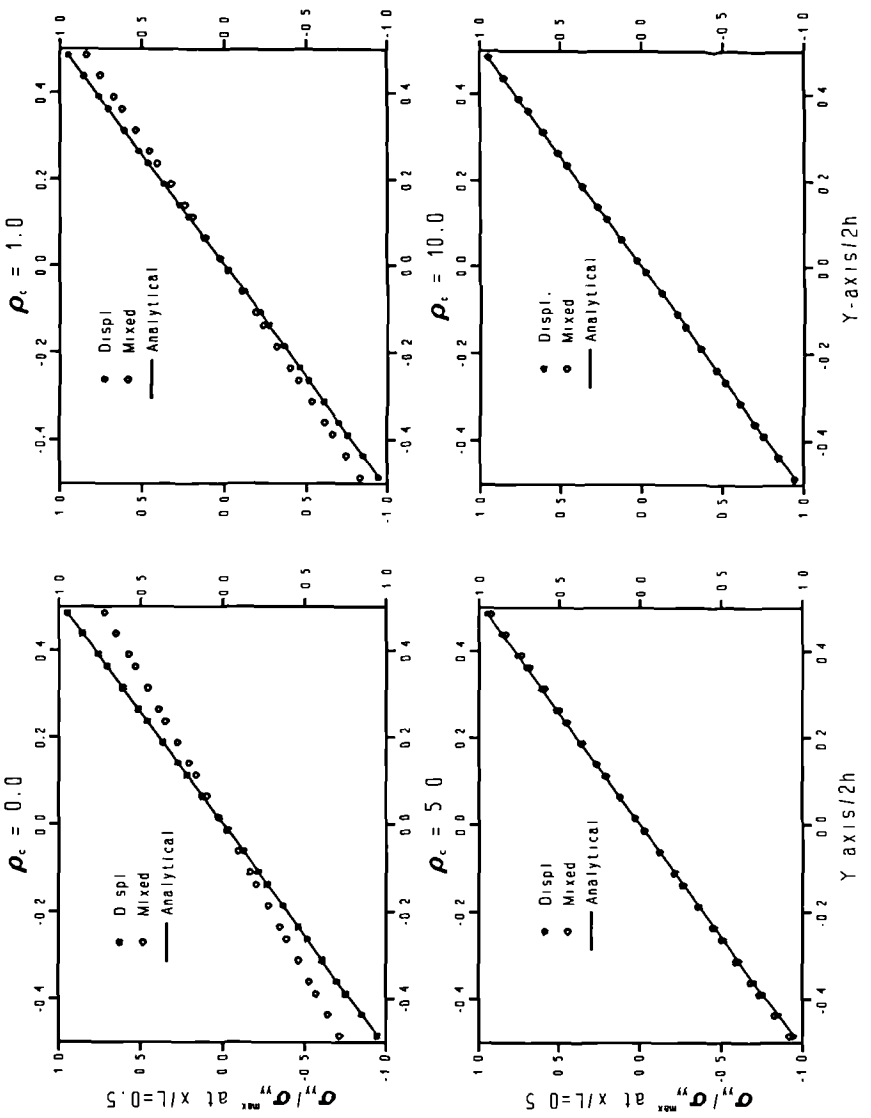


Fig.3.31 Distributions of σ_{xx} along y-axis at x/L=0.5 in a beam under an end loading. The results are obtained by a FE mesh with 80 elements and 3x3 integration.

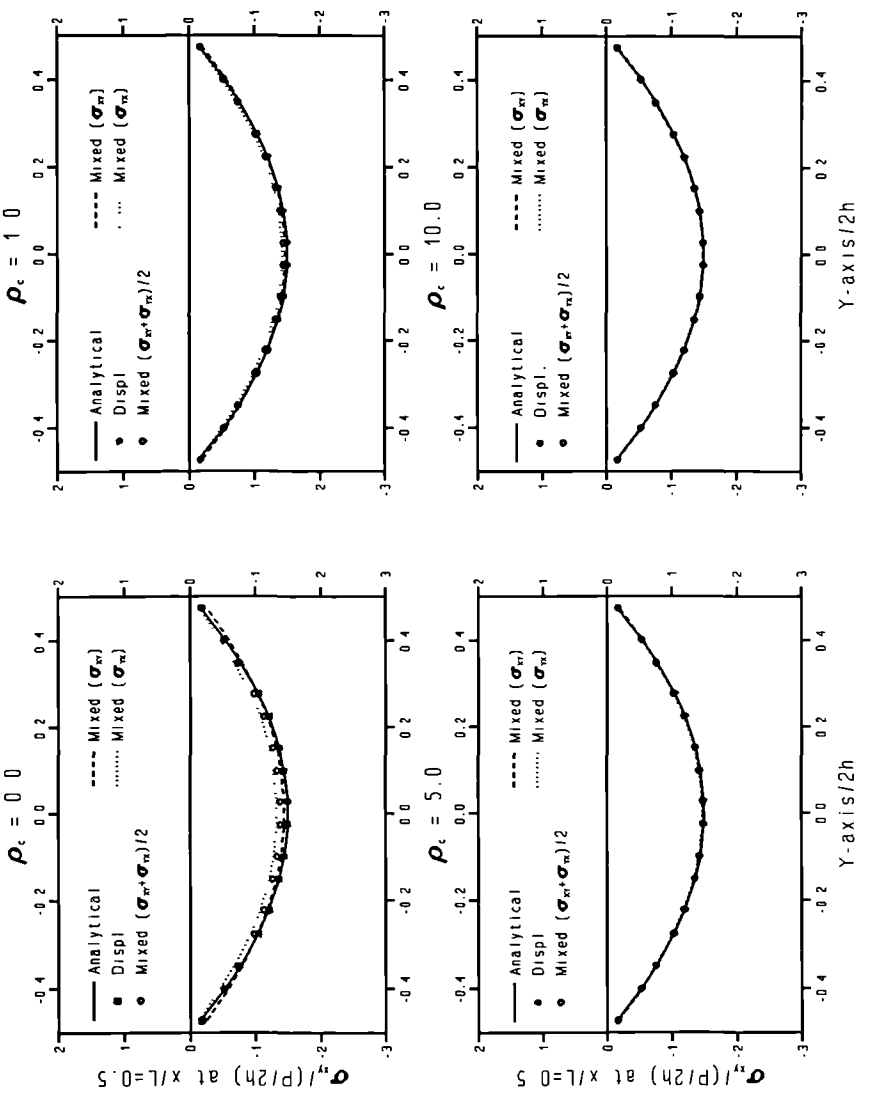


Fig.3.32 Distributions of σ_{xy} and σ_{yx} along y-axis at x/L=0.5 in a beam under an end loading. The results are obtained by a FE mesh with 80 elements and 2x2 integration.

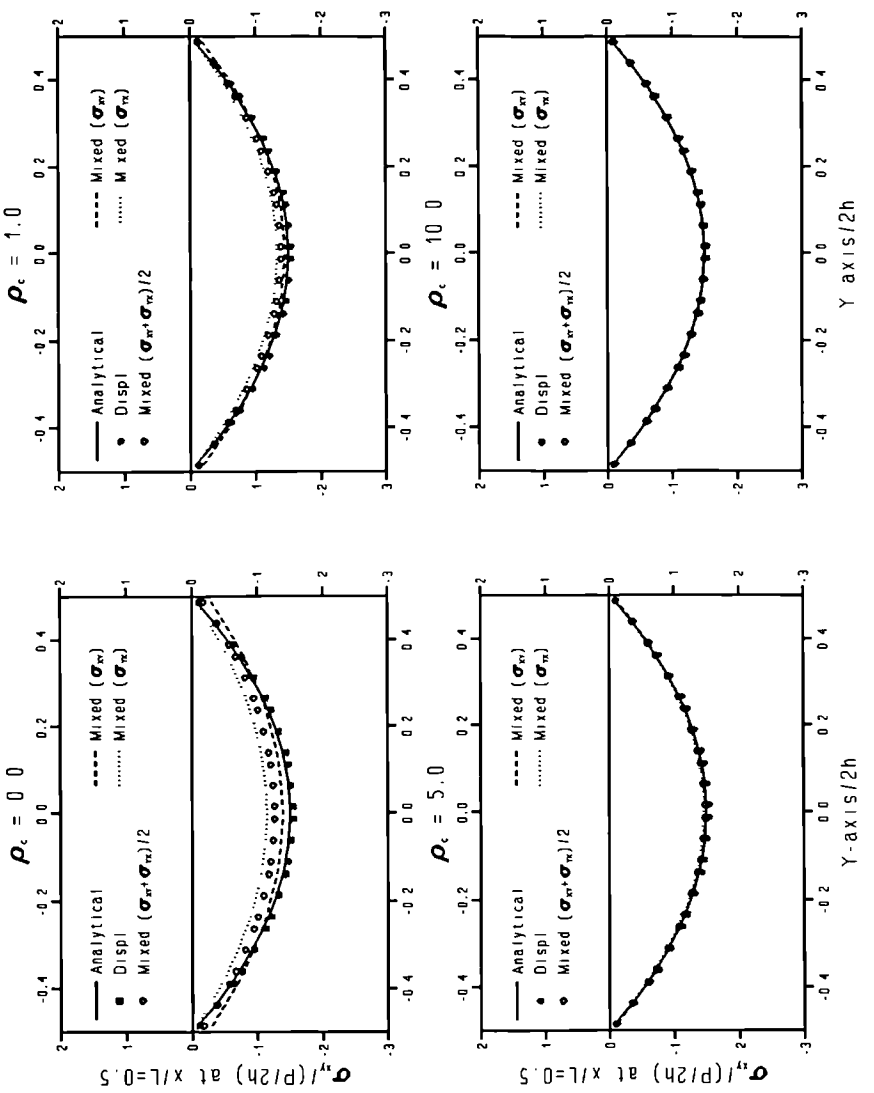


Fig.3.33 Distributions of σ_{xy} and σ_{yx} along y-axis at $x/L=0.5$ in a beam under an end loading. The results are obtained by a FE mesh with 80 elements and 3x3 integration.

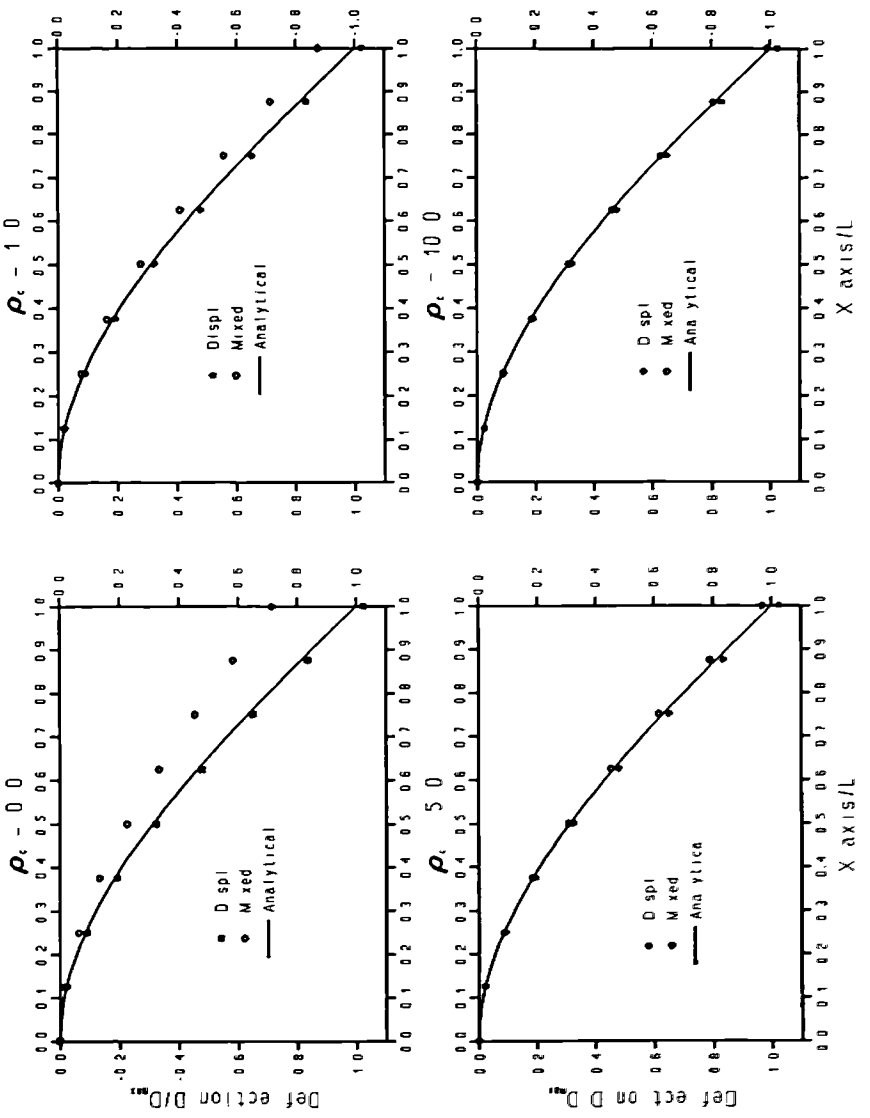


Fig.3.34 Deflection of the beam along *x*-axis under an end loading. The results are obtained by a FE mesh with 16 elements and 2X2 integration.

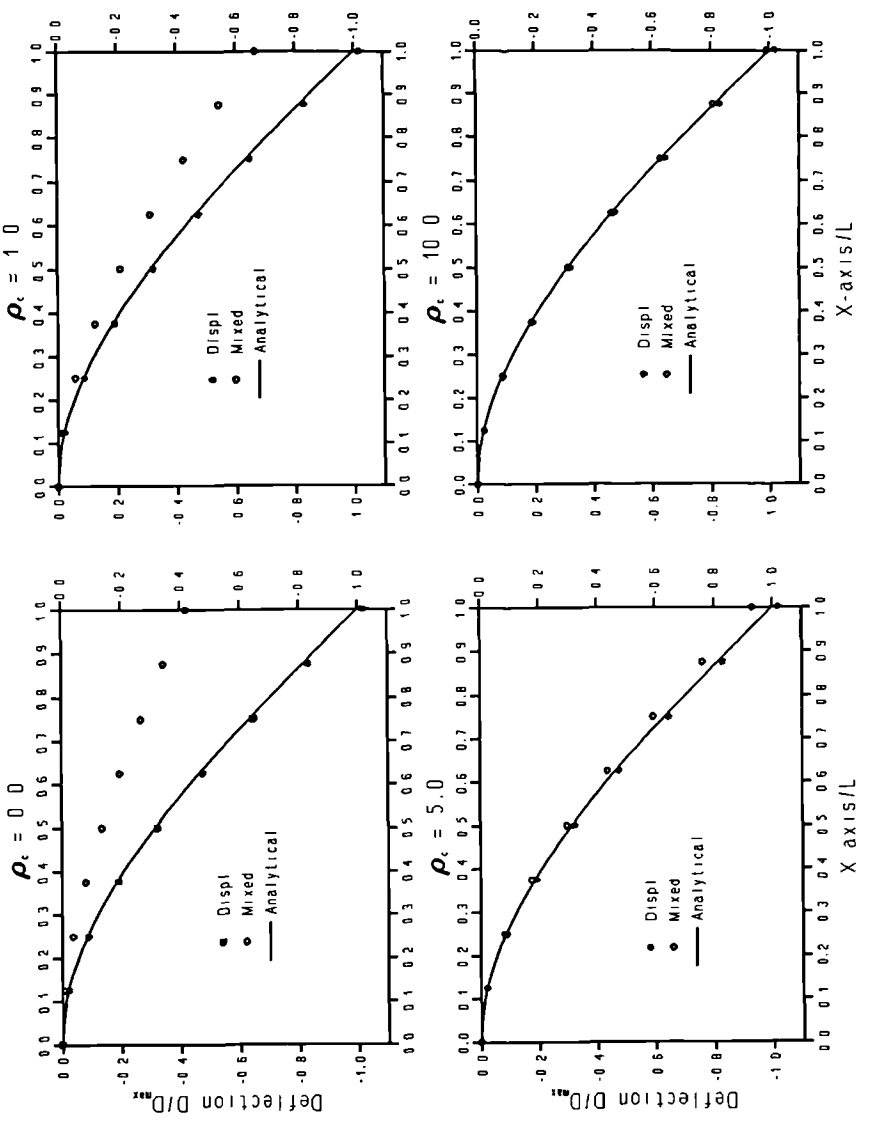


Fig.3.35 Deflection of the beam along x-axis under an end loading. The results are obtained by a FE mesh with 16 elements and 3×3 integration.

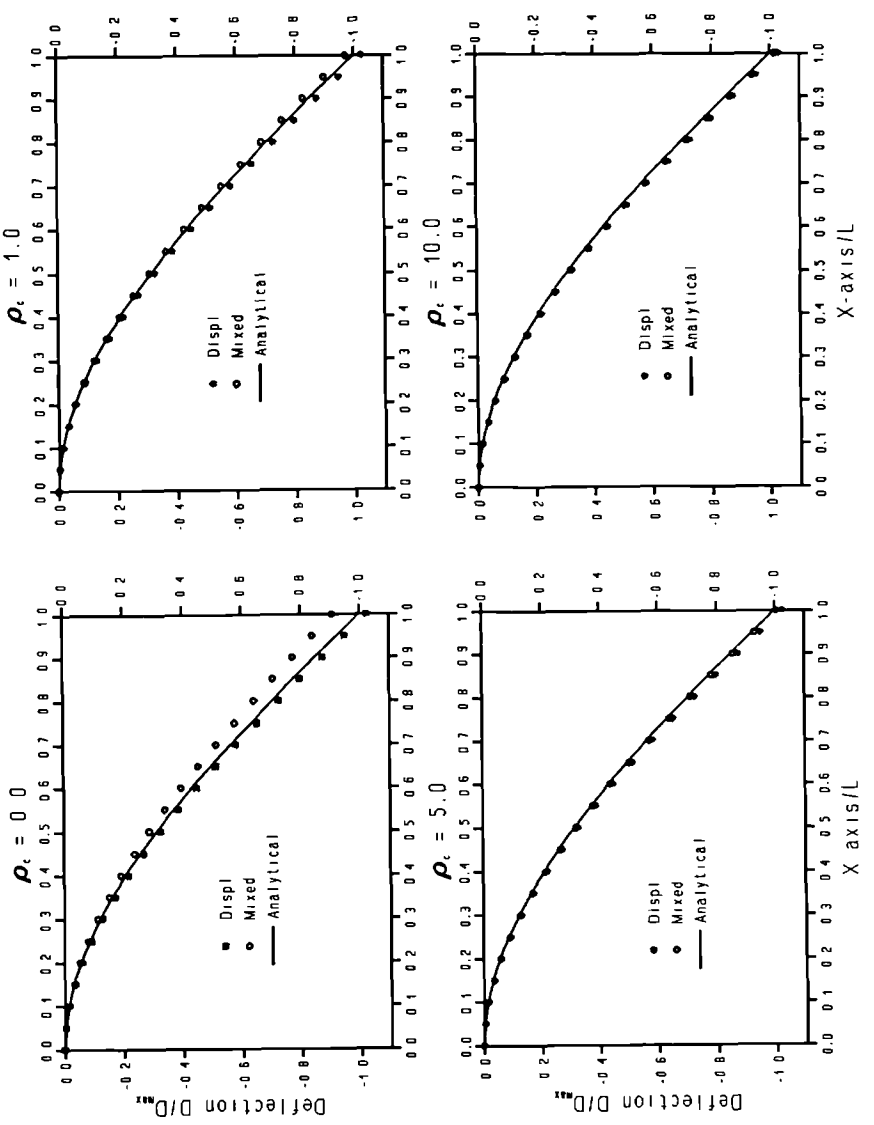


Fig.3.36 Deflection of the beam along x-axis under an end loading. The results are obtained by a FE mesh with 80 elements and 2×2 integration.

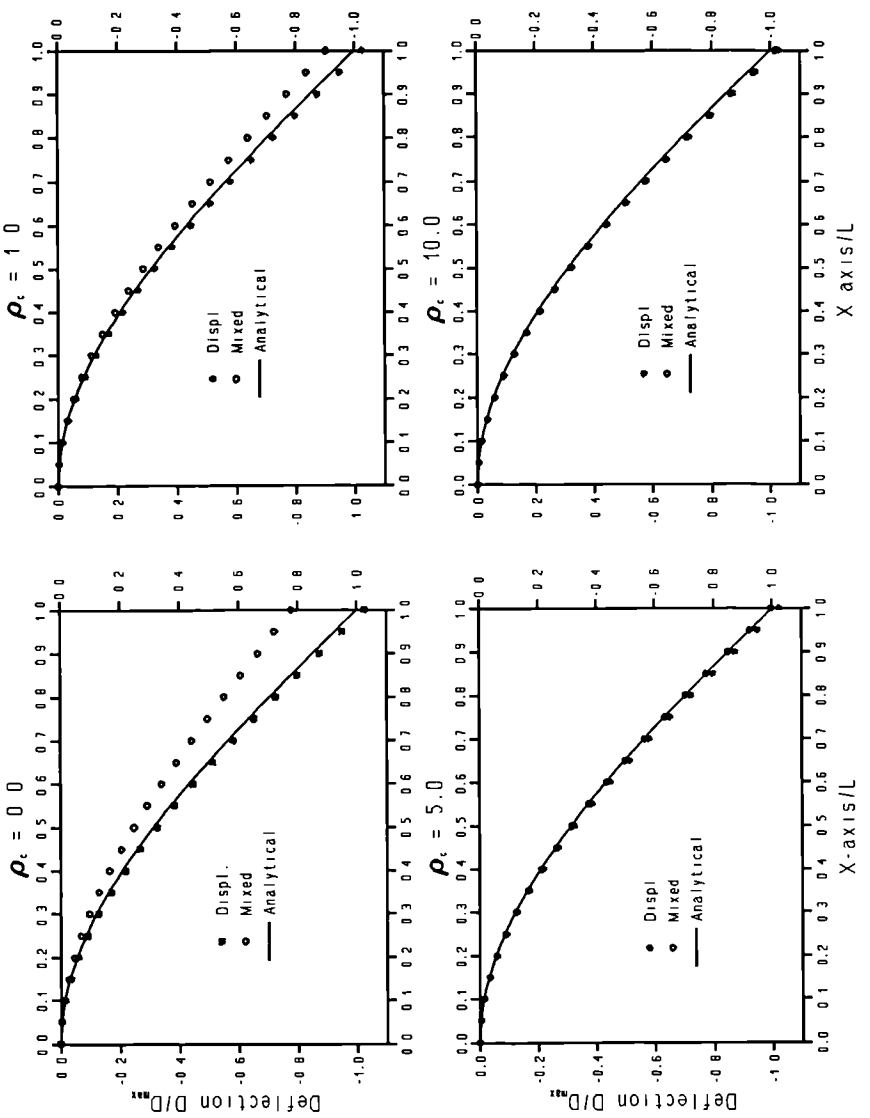


Fig.3.37 Deflection of the beam along x-axis under an end loading. The results are obtained by a FE mesh with 80 elements and 3×3 integration.

CHAPTER 4
MIXED MODEL IN AXISYMMETRIC ELASTICITY

In practical engineering, there are many structures or components of structure with a geometry that are formed by simply rotating a curve around an axis. This is called axisymmetry. Often the force applied on the structure is also symmetrical about the axis and produces no torsion, such as the pressure within a gas cylinder. This particular symmetry enables a simple 2D formulation to describe the 3D structure since all parameters do not vary with the angle of rotation θ and are only functions of the radius r and height z. The basic formulation suitable for axisymmetric problems is firmly established in the classical theory of elasticity[110]. Many applications have also been successfully made by the displacement model of FE method. Although axi-symmetric problems has become one of the major areas in the application of FE method, only few such examples by mixed models are found in literature, as reviewed in chapter 2. Unlike in the displacement model, where no substantial changes need to be made to include an axisymmetric formulation in an existing 2D plane formulation, the mixed models need to approximate the new stress field which is much more complicated than in plane problems. By introducing a stress function ϕ the equilibrium equations are satisfied if we take

$$\begin{aligned} \sigma_{rr} &= \frac{\partial}{\partial z}\left[\nu\nabla^2\phi - \frac{\partial^2\phi}{\partial r^2}\right] & \sigma_{zz} &= \frac{\partial}{\partial z}\left[(2-\nu)\nabla^2\phi - \frac{\partial^2\phi}{\partial z^2}\right] \\ \sigma_{\theta\theta} &= \frac{\partial}{\partial z}\left[\nu\nabla^2\phi - \frac{1}{r}\frac{\partial\phi}{\partial r}\right] & \sigma_{rz} &= \frac{\partial}{\partial r}\left[(1-\nu)\nabla^2\phi - \frac{\partial^2\phi}{\partial z^2}\right] \end{aligned} \tag{4.1}$$

provided that the stress function ϕ satisfies the equation

$$\left[\frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{\partial^2}{\partial z^2}\right]\left[\frac{\partial^2\phi}{\partial r^2} + \frac{1}{r}\frac{\partial\phi}{\partial r} + \frac{\partial^2\phi}{\partial z^2}\right] = \nabla^2\nabla^2\phi = 0 \tag{4.2}$$

ϕ is the well-known Love stress function[62] for axisymmetric problems. Compared with the Airy stress function in eq.(3.22), it is much more complex so it is easy to see how much extra work needs to be done in discretizing the stress field. This is the

reason why mixed models have not been extensively used in practice.

In order to exploit the potential merits of the mixed model in axisymmetric problems we will introduce new constrained first-order stress functions in this chapter.

4.1 CONSTRAINED FIRST-ORDER STRESS FUNCTIONS FOR AXISYMMETRIC PROBLEMS

The complexity of the Love stress function makes it difficult to directly decrease the order of the stress function from the third to the first, which was done to Airy stress function. Look at the equilibrium equations in axisymmetric problems first, which is

$$\begin{cases} \dfrac{\partial \sigma_{rr}}{\partial r} + \dfrac{\partial \sigma_{zr}}{\partial z} + \dfrac{\sigma_{rr} - \sigma_{\theta\theta}}{r} = 0 \\ \dfrac{\partial \sigma_{rz}}{\partial r} + \dfrac{\partial \sigma_{zz}}{\partial z} + \dfrac{\sigma_{rz}}{r} = 0 \end{cases} \tag{4.3}$$

and

$$\sigma_{rz} = \sigma_{zr} \tag{4.4}$$

It is not difficult to see that the second equation in eq.(4.3) is quite similar to the one in plane problems (eq.3.2) except for an extra term σ_{rz}/r. Now recall the expression of the components of the stress in eq.(3.20) for plane problems, and replace x, y, z of Cartesian system for plane problems by the relevant r, z, θ of the cylindrical system for axisymmetric problems. With the new stress functions F_r and F_z, it is straightforward to satisfy the second equation in eq.(4.3) by simply adding a term F_r/r to σ_{zz}, i.e. $\sigma_{zz} = F_{r,r} + F_r/r$, to balance σ_{rz}/r. If σ_{rr} is conceived in the same way by adding F_r/r, i.e. $\sigma_{rr} = F_{z,z} + F_r/r$, one left component $\sigma_{\theta\theta}$ is immediately obtained by substituting σ_{rr} and σ_{zr} into the first equation in eq.(4.3), i.e. $\sigma_{\theta\theta} = F_{z,z} + F_{r,r}$.

However in order to satisfy eq.(4.3), it is not necessary to use F_r in the expression of σ_{rr} and $\sigma_{\theta\theta}$ since any variable, say F_θ, is eligible. Therefore the components of the stress obeying force equilibrium equation (4.3) in axisymmetric problems finally take the form of[17]

$$\begin{aligned} \sigma_{rr} &= \frac{\partial F_z}{\partial z} + \frac{F_\theta}{r} \quad ; \quad \sigma_{rz} = -\frac{\partial F_r}{\partial z} \quad ; \\ \sigma_{zr} &= -\frac{\partial F_z}{\partial r} \quad ; \quad \sigma_{zz} = \frac{\partial F_r}{\partial r} + \frac{F_r}{r} \quad ; \quad \sigma_{\theta\theta} = \frac{\partial F_z}{\partial z} + \frac{\partial F_\theta}{\partial r} \end{aligned} \tag{4.5}$$

where F_r, F_z and F_θ are three first-order stress functions.

As we did in plane elasticity in chapter 3, the moment equilibrium eq.(4.4) is again satisfied approximately *a posteriori* as a constraint

$$\frac{\partial F_r}{\partial z} = \frac{\partial F_z}{\partial r} \qquad \text{in domain } \Omega \tag{4.6}$$

by means of a penalty function. Therefore F_r, F_z and F_θ are also termed as the *constrained first-order stress functions.*

We shall now show that the components of the stress defined by eq.(4.5) possess the same merits as by eq.(3.20) for plane elasticity: it allows the discontinuity of stress, and at the same time satisfies the traction reciprocity at inter-element boundaries.

Consider a surface with normal vector **n**, and a tangent vector τ. After performing similar operations as in eqs.(3.26) and (3.27), and using the new expression of the components of stress in eq.(4.5), we have obtained the components t_n and t_τ of traction at any point on this surface

$$\begin{cases} t_n = -n_r \dfrac{\partial F_z}{\partial \tau} + n_z \dfrac{\partial F_r}{\partial \tau} + n_r^2 \dfrac{F_\theta}{r} + n_z^2 \dfrac{F_r}{r} \\ t_\tau = -n_r \dfrac{\partial F_r}{\partial \tau} - n_z \dfrac{\partial F_z}{\partial \tau} - n_r n_z \dfrac{F_r - F_\theta}{r} \end{cases} \tag{4.7}$$

where n_r and n_z are direction cosines defined by $n_r = \cos(\mathbf{n},\mathbf{r})$; $n_z = \cos(\mathbf{n},\mathbf{z})$.

Eq.(4.7) shows that there are only the derivatives of F_r, F_z or F_θ with respect to τ involved. If the C^0 continuity of F_r, F_z and F_θ is ensured, at the common boundary between two adjacent elements a and b the following relations

$$(F_r)^a = (F_r)^b \quad ; \quad (F_z)^a = (f_z)^b \quad ; \quad (F_\theta)^a = (F_\theta)^b$$

and

$$\left(\frac{\partial F_r}{\partial \tau}\right)^a = \left(\frac{\partial F_r}{\partial \tau}\right)^b \quad ; \quad \left(\frac{\partial F_z}{\partial \tau}\right)^a = \left(\frac{\partial F_z}{\partial \tau}\right)^b \quad ; \quad \left(\frac{\partial F_\theta}{\partial \tau}\right)^a = \left(\frac{\partial F_\theta}{\partial \tau}\right)^b$$

are exactly true. It means that the traction reciprocity is exactly obeyed, i.e.

$$(t_n)^a \equiv (t_n)^b \quad ; \quad (t_\tau)^a \equiv (t_\tau)^b \tag{4.8}$$

Certainly the same argument as for plane problems is also suitable for explaining the ability for this model to cope with the discontinuity at the inter-element boundary in axisymmetric problems.

4.2 DISCRETIZATION OF DISPLACEMENT AND STRESS FOR AXISYMMETRIC PROBLEMS

In the classical theory of elasticity, the components of a compatible strain can be expressed by the displacements as

$$\begin{aligned} e_{rr} &= \frac{\partial u_r}{\partial r} \quad ; \quad e_{rz} = \frac{\partial w}{\partial r} + \frac{\partial u}{\partial z} \quad ; \\ e_{zr} &= \frac{\partial w}{\partial r} + \frac{\partial u}{\partial z} \quad ; \quad e_{zz} = \frac{\partial w}{\partial z} \quad ; \quad e_{\theta\theta} = \frac{u}{r} \end{aligned} \tag{4.9}$$

in which, u are w are the radial and axial displacements respectively. Comparing with eq.(4.5) of the components of stress, it is not difficult to find that both stress and strain fields are expressed by the first order-derivatives of r or z, which means that only C^0 continuous functions are required to approximate the components of strain or stress by u, w or F_r, F_z and F_θ. Therefore the same discretization of displacement and stress function as in plane problems, i.e. isoparametric formulation can then be directly applied in axisymmetric problems

$$\begin{Bmatrix} \mathbf{u}(\xi,\eta) \\ \mathbf{w}(\xi,\eta) \end{Bmatrix} = \sum_{i=1}^{m} N_i(\xi,\eta) \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} u_i \\ w_i \end{Bmatrix} \tag{4.10}$$

and

$$\begin{Bmatrix} F_r(\xi,\eta) \\ F_z(\xi,\eta) \\ F_\theta(\xi,\eta) \end{Bmatrix} = \sum_{i=1}^{m} N_i(\xi,\eta) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} F_{ri} \\ F_{zi} \\ F_{\theta i} \end{Bmatrix} \tag{4.11}$$

where u, w, F_r, F_z and F_θ are approximate displacements and stress functions at point

(ξ,η), while u_i, w_i, F_{ri}, F_{zi} and $F_{\theta i}$ are the corresponding nodal variables of the element. For the convenience of FE formulation, the following matrix forms of stress and strain tensors are introduced

$$\begin{aligned} \mathbf{e} &= \{e_{rr}\ e_{rz}\ e_{zr}\ e_{zz}\ e_{\theta\theta}\}^T \ ; \ \mathbf{d} = \{d_{rr}\ d_{rz}\ d_{zr}\ d_{zz}\ d_{\theta\theta}\}^T \\ \boldsymbol{\sigma} &= \{\sigma_{rr}\ \sigma_{rz}\ \sigma_{zr}\ \sigma_{zz}\ \sigma_{\theta\theta}\}^T \ ; \ \mathbf{s} = \{s_{rr}\ s_{rz}\ s_{zr}\ s_{zz}\ s_{\theta\theta}\}^T \end{aligned} \tag{4.12}$$

where **e** and **d** are strain and deviatoric strain, while $\boldsymbol{\sigma}$ and **s** are stress and deviatoric stress. The nodal variables for axisymmetric problem are defined as

$$\beta_i^d = \{u_i\ w_i\}^T \quad ; \quad \beta_i^s = \{F_{ri}\ F_{zi}\ f_{\theta i}\}^T \tag{4.13}$$

Substituting eq.(4.10) and eq.(4.11) into eq.(4.9) and making use of eq.(3.9) and eq.(3.10), $\boldsymbol{\sigma}$, **e**, **s**, **d**, mean stress $\sigma_{\mathbf{m}}$ and volume strain $\mathbf{e_m}$ can be expressed in matrix forms as

$$\mathbf{e} = \sum_{i=1}^{m} \mathbf{B_i}\, \beta_i^d \quad ; \quad \mathbf{d} = \sum_{i=1}^{m} \mathbf{B_{Di}}\, \beta_i^d \quad ; \quad e_m = \sum_{i=1}^{m} \mathbf{B_{mi}}\, \beta_i^d \tag{4.14}$$

where

$$\begin{aligned} \mathbf{B_{Di}} &= \mathbf{B_i} - \frac{1}{3}\mathbf{T_d}\mathbf{B_{mi}^T} \\ \mathbf{B_{mi}} &= \left[N_{i,x} + N_i/r \quad N_{i,y}\right] \end{aligned} \tag{4.15}$$

$$\mathbf{B_i} = \begin{bmatrix} N_{i,r} & N_{i,z} & N_{i,z} & 0 & N_i/r \\ 0 & N_{i,r} & N_{i,r} & N_{i,z} & 0 \end{bmatrix}^T$$

and

$$\boldsymbol{\sigma} = \sum_{i=1}^{m} \mathbf{A_i}\, \beta_i^s \quad ; \quad \mathbf{s} = \sum_{i=1}^{m} \mathbf{A_{Di}}\, \beta_i^s \quad ; \quad \sigma_m = \sum_{i=1}^{m} \mathbf{A_{mi}}\, \beta_i^s \tag{4.16}$$

where

$$\begin{aligned} \mathbf{A_{Di}} &= \mathbf{A_i} - \mathbf{T_s}\mathbf{A_{mi}^T} \\ \mathbf{A_{mi}} &= \left[\frac{1}{3}(N_{i,r} + N_i/r) \quad \frac{2}{3}N_{i,z} \quad \frac{1}{3}(N_{i,r} + N_i/r)\right] \end{aligned} \tag{4.17}$$

$$\mathbf{A_i} = \begin{bmatrix} 0 & -N_{i,z} & 0 & N_{i,r}+N_i/r & 0 \\ N_{i,z} & 0 & -N_{i,r} & 0 & N_{i,z} \\ N_i/r & 0 & 0 & 0 & N_{i,r} \end{bmatrix}^T$$

$\mathbf{T_d}$ and $\mathbf{T_s}$ are constant coefficient matrices defined by

$$\mathbf{T_d} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}^T \quad ; \quad \mathbf{T_s} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}^T$$

4.3 BOUNDARY CONDITIONS

The displacement boundary conditions can be imposed by exactly the same method as in eqs.(3.30) and (3.31) for plane elasticity except that v and v_i should be replaced by w and w_i respectively. However to impose the traction boundary conditions more work has to be done due to the obviously different expressions of traction t_n and t_τ. Consider a segment ab at the boundary, as defined in section 3.3. If the length dL is small, the stress functions F_r, F_z and F_θ can be treated as linear functions of τ. By means of linear Lagrange interpolation the following approximate relations exist

$$\frac{\partial F_r}{\partial \tau} \approx \frac{1}{dL}(F_{rb}-F_{ra}) \quad ; \quad \frac{\partial F_z}{\partial \tau} \approx \frac{1}{dL}(F_{zb}-F_{za}) \tag{4.18}$$

and

$$\begin{aligned} F_r(\tau) &= (\tfrac{1}{2}-\tfrac{\tau}{dL})F_{ra} + (\tfrac{1}{2}+\tfrac{\tau}{dL})F_{rb} \\ F_z(\tau) &= (\tfrac{1}{2}-\tfrac{\tau}{dL})F_{za} + (\tfrac{1}{2}+\tfrac{\tau}{dL})F_{zb} \quad ; \quad \tau \subseteq \left[-\frac{dL}{2}, \frac{dL}{2}\right] \\ F_\theta(\tau) &= (\tfrac{1}{2}-\tfrac{\tau}{dL})F_{\theta a} + (\tfrac{1}{2}+\tfrac{\tau}{dL})F_{\theta b} \end{aligned} \tag{4.19}$$

The derivatives of stress functions are obviously constants. Substituting eqs.(4.18) and (4.19) into eq.(4.7) and omitting the mathematical details for clarity, we finally have

$$t_n = \mathbf{V_n}\beta^s_{\mathbf{ab}} \quad ; \quad t_\tau = \mathbf{V_\tau}\beta^s_{\mathbf{ab}} \tag{4.20}$$

where

$$\mathbf{V_n} = \frac{1}{dL}\left[(-1+\alpha_1 n_z)n_z \quad n_r \quad \alpha_1 n_r^2 \quad (1+\alpha_2 n_z)n_z \quad -n_z \quad \alpha_2 n_r^2\right]$$

$$\mathbf{V_\tau} = \frac{1}{dL}\left[(1-\alpha_1 n_z)n_r \quad n_z \quad \alpha_1 n_r n_z \quad -(1+\alpha_2 n_z)n_r \quad -n_z \quad \alpha_2 n_r n_z\right]$$

$$\boldsymbol{\beta}^{\mathbf{s}}_{\mathbf{ab}} = \{\beta_a^s \;\; \beta_b^s\}^T \;\; ; \;\; \alpha_1 = \frac{dL}{2r} - \frac{\tau}{r} \;\; ; \;\; \alpha_2 = \frac{dL}{2r} + \frac{\tau}{r}$$

β_i^s is the nodal stress function variables defined in eq.(4.13). The value of a prescribed traction $\overline{t_n}$ and $\overline{t_\tau}$ along the small segment *ab* of the boundary line Γ can be imposed again by using a quadratic penalty function

$$\Pi_s = \int_{d\Gamma}\left[\begin{Bmatrix}\overline{t_n}\\ \overline{t_\tau}\end{Bmatrix} - \begin{Bmatrix}t_n\\ t_\tau\end{Bmatrix}\right]^T \boldsymbol{\rho}^{\mathbf{s}}\left[\begin{Bmatrix}\overline{t_n}\\ \overline{t_\tau}\end{Bmatrix} - \begin{Bmatrix}t_n\\ t_\tau\end{Bmatrix}\right] d\Gamma \tag{4.21}$$

where $\boldsymbol{\rho}^{\mathbf{s}}$ is defined exactly the same as that for plane elasticity. Writing eq.(4.21) in a matrix form by using t_n and t_τ in eq.(4.20), it follows the first derivative of Π_s with respect to β_a^s and β_b^s

$$\frac{\partial \Pi_s}{\partial \boldsymbol{\beta}^{\mathbf{s}}_{\mathbf{ab}}} = \mathbf{k}^{\mathbf{s}}_{\mathbf{ab}}\boldsymbol{\beta}^{\mathbf{s}}_{\mathbf{ab}} + \mathbf{F}^{\mathbf{s}} \tag{4.22}$$

where

$$\mathbf{k}^{\mathbf{s}}_{\mathbf{ab}} = \int_\Gamma (\mathbf{T}^{\mathbf{n}}_{\mathbf{s}} + \mathbf{T}^{\tau}_{\mathbf{s}})d\Gamma \;\; ; \;\; \mathbf{F}^{\mathbf{s}} = \int_\Gamma (\overline{t_n}\mathbf{f}^{\mathbf{n}}_{\mathbf{s}} + \overline{t_\tau}\mathbf{f}^{\tau}_{\mathbf{s}})d\Gamma$$

and

$$\begin{cases} \mathbf{T}^{\mathbf{n}}_{\mathbf{s}} = 2\rho_s^n \mathbf{V}^{\mathbf{T}}_{\mathbf{n}}\mathbf{V}_{\mathbf{n}} \\ \mathbf{T}^{\tau}_{\mathbf{s}} = 2\rho_s^\tau \mathbf{V}^{\mathbf{T}}_{\tau}\mathbf{V}_{\tau} \end{cases} ; \quad \begin{cases} \mathbf{f}^{\mathbf{n}}_{\mathbf{s}} = 2\rho_s^n \mathbf{V}^{\mathbf{T}}_{\mathbf{n}} \\ \mathbf{f}^{\tau}_{\mathbf{s}} = 2\rho_s^\tau V_\tau^T \end{cases}$$

4.4 GLOBAL SYSTEM EQUATIONS FOR AXISYMMETRIC PROBLEMS

The global functional Π_I in eq.(3.36) is true for any elasticity problems on condition that the displacement field is kinematically admissible and the stress field is statically admissible regardless of geometry dimensions. When applied to particular problems,

it is necessary to substitute the suitable expressions of admissible displacement, strain and stress, and their correspondent boundary conditions. Therefore eq.(3.39) for the first order-derivative of Π_{elas}, eq.(3.38) for the first derivative of Π_c are applicable to axisymmetric cases. However all the matrices in the expression of $\mathbf{k_{ij}}^{\mathbf{e}}$ in eq.(3.39) should use the corresponding ones defined in eqs.(4.14) and (4.16). The $\mathbf{k_{ii}}^{\mathbf{d}}$ in eq.(3.31) with regard to displacement boundary is also true with the change of v_i to w_i. $\mathbf{k_{ij}}^{\mathbf{c}}$ in eq.(3.39) should be written as

$$\mathbf{k}_{\mathbf{ij}}^{\mathbf{c}} = \int_{\Omega^e} 2\rho_c \begin{bmatrix} N_{i,z}N_{j,z} & -N_{i,r}N_{j,z} \\ -N_{i,r}N_{j,z} & N_{i,r}N_{j,r} \end{bmatrix} d\Omega \tag{4.23}$$

It is important to note that the general nodal variable β_i is changed from 4 components to 5 components, i.e.

$$\beta_i = \begin{Bmatrix} \beta_i^d \\ \beta_i^s \end{Bmatrix} = \{u_i \;\; w_i \;\; F_{ri} \;\; F_{zi} \;\; F_{\theta i}\}^T \tag{4.24}$$

which means that the size of the sub-matrix $\mathbf{k_{ij}}^{\mathbf{e}}$ becomes 5×5 instead of 4×4. The element system equation is identical to the eqs.(3.41) and (3.42) for plane problems.

The expansion of $\mathbf{k}^{\mathbf{d}}$ and $\mathbf{k}^{\mathbf{s}}$ is needed to perform the operations in matrix form. The standard assembling procedure is used to form the global system equations, i.e. the global stiffness matrix and the global load vector element by element. Because the global stiffness matrix is again symmetric positive-definite, the solution is readily obtained by a classical variable bandwidth algorithm. In the next section, some numerical examples will be presented in order to demonstrate the quality of this model.

4.5 ELASTIC SOLUTIONS IN AXISYMMETRIC PROBLEMS

Some axisymmetric problems in linear elasticity are solved below by the mixed method in this section to illustrate how this model can cope with these kinds of problems. These include: bending of a circular plate and spherical container under internal uniform pressure. Then a more complex problem is to be solved by this model which is the mechanical modelling of the early stages of the spot-welding process.

4.5.1 Bending of a Circular Plate

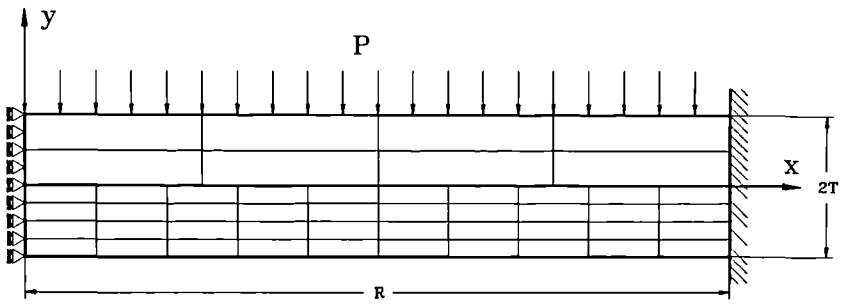


Fig. 4.1 FE models for bending of a circular plate under uniform pressure.

This example is to investigate the capability of this model to cope the problems with severe shear deformation, as we did in section 3.6.4 in plane stress condition. A similar FE model to fig.3.20 is used to analyse a circular plate with radius R, thickness $2T$ under a uniform transverse pressure p at the upper boundary, as shown in fig.4.1. The analytical results can be found in ref. [72].

Only one FE mesh with 8×10 of 8-noded axisymmetric elements is used in the analysis. Two sets of different values of ρ_c are used again, i.e. ρ_c = 0, 1.0, 5.0, 10.0 for 2×2 integration and ρ_c = 0, 1.0, 10.0, 50.0 for 3×3 integration.

The results of σ_{rr} at the surface along r-axis are presented in fig.4.2 for 2×2 integration and fig.4.3 for 3×3. It is seen that with 2×2, the mixed model gives very good values for σ_{rr} when $\rho_c \geq 5.0$. However only when ρ_c = 5.0 does it yield a good result when 3×3 integration is used. In the other words the results from mixed model with 3×3 are conditional: you cannot use small, nor use large ρ_c to get a good results.

In the case of $\sigma_{\theta\theta}$, the results of mixed model with 3×3 integration in fig.4.5 are similar to fig.4.3 for σ_{rr}, i.e. ρ_c = 5.0 gives best results. However those with 2×2 one are not quite so good compared with the analytical results, as seen in fig.4.4. Obviously results obtained with ρ_c = 5.0 and ρ_c = 10.0 do give better than those with ρ_c = 0.0 or ρ_c = 1.0, but they yield fairly large deviation along r-axis, especially near the clamped end.

For the deflection of the central line of the disc, a larger penalty number ρ_c is generally required. In fig.4.6 and fig.4.7, they show that a good result can only be

obtained when $\rho_c = 50.0$ is employed for both 2×2 and 3×3 integrations.

For all above cases, the displacement model gives constantly better results than the mixed model. Does this mean that this mixed model is not suitable for the axi-symmetric problem? Not necessarily. Since this example is an extreme case which deliberately tests this approaches, it behaves reasonably well. In the next example, a spherical container under internal uniform pressure will be analysed and better results from the mixed model are obtained.

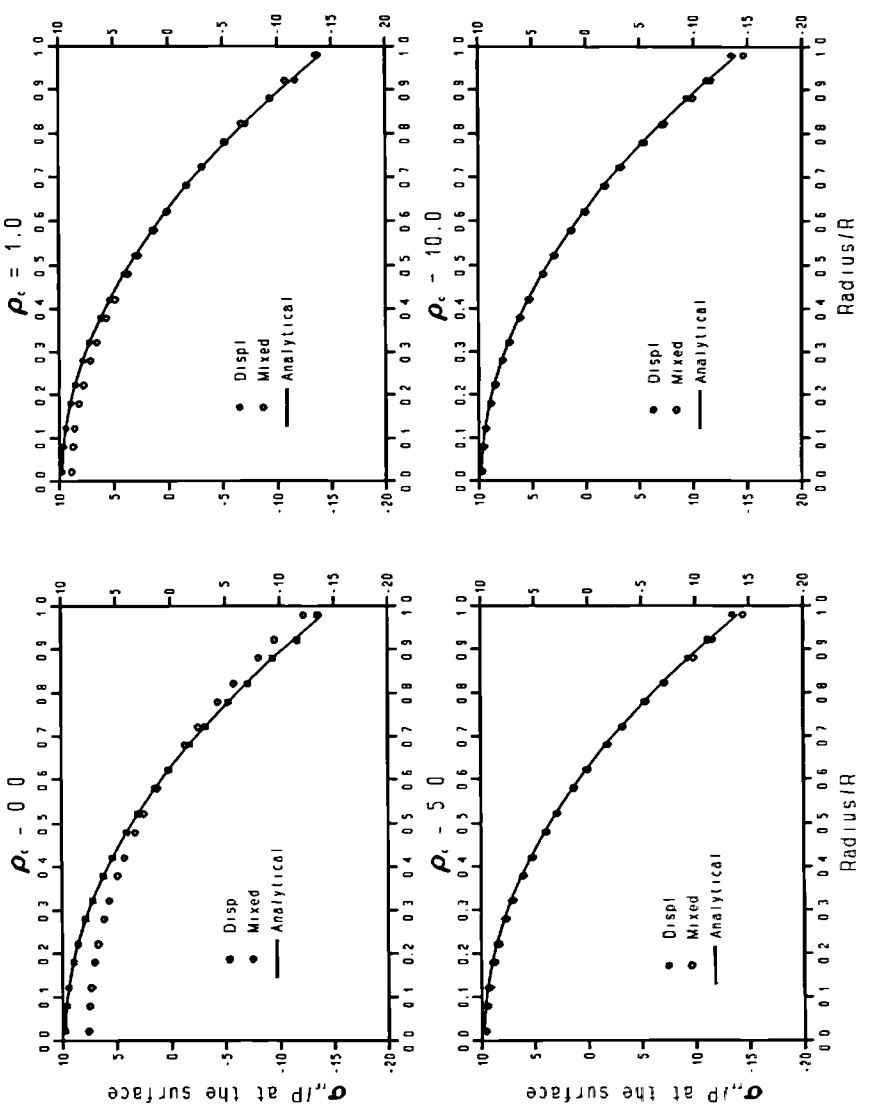


Fig.4.2 Distributions of σ_{rr} at the surface of a disc along r-axis under uniformly pressure. The results are obtained by a FE mesh with 80 8-noded elements and 2×2 integration.

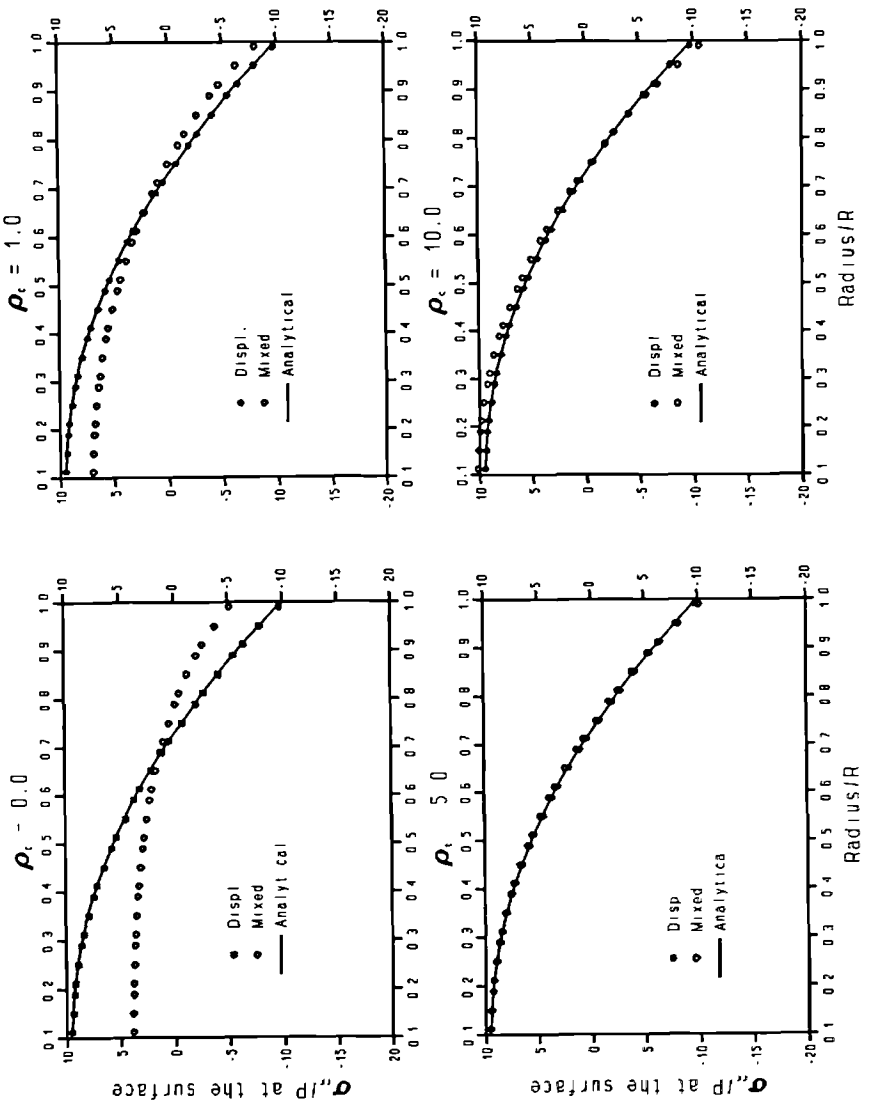


Fig.4.3 Distributions of σ_{rr} at the surface of a disc along r-axis under uniformly pressure. The results are obtained by a FE mesh with 80 8-noded elements and 3×3 integration.

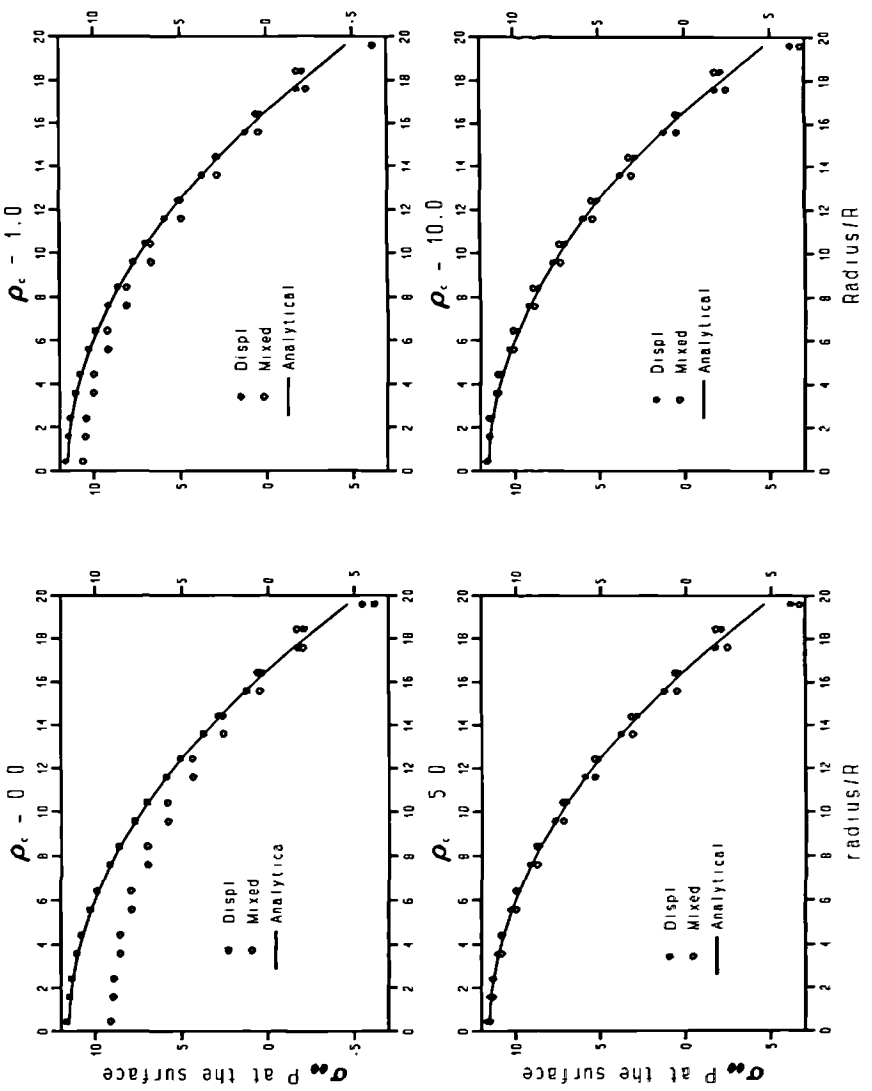


Fig.4.4 Distributions of $\sigma_{\theta\theta}$ at the surface of a disc along r-axis under uniformly pressure. The results are obtained by a FE mesh with 80 8-noded elements and 2×2 integration.

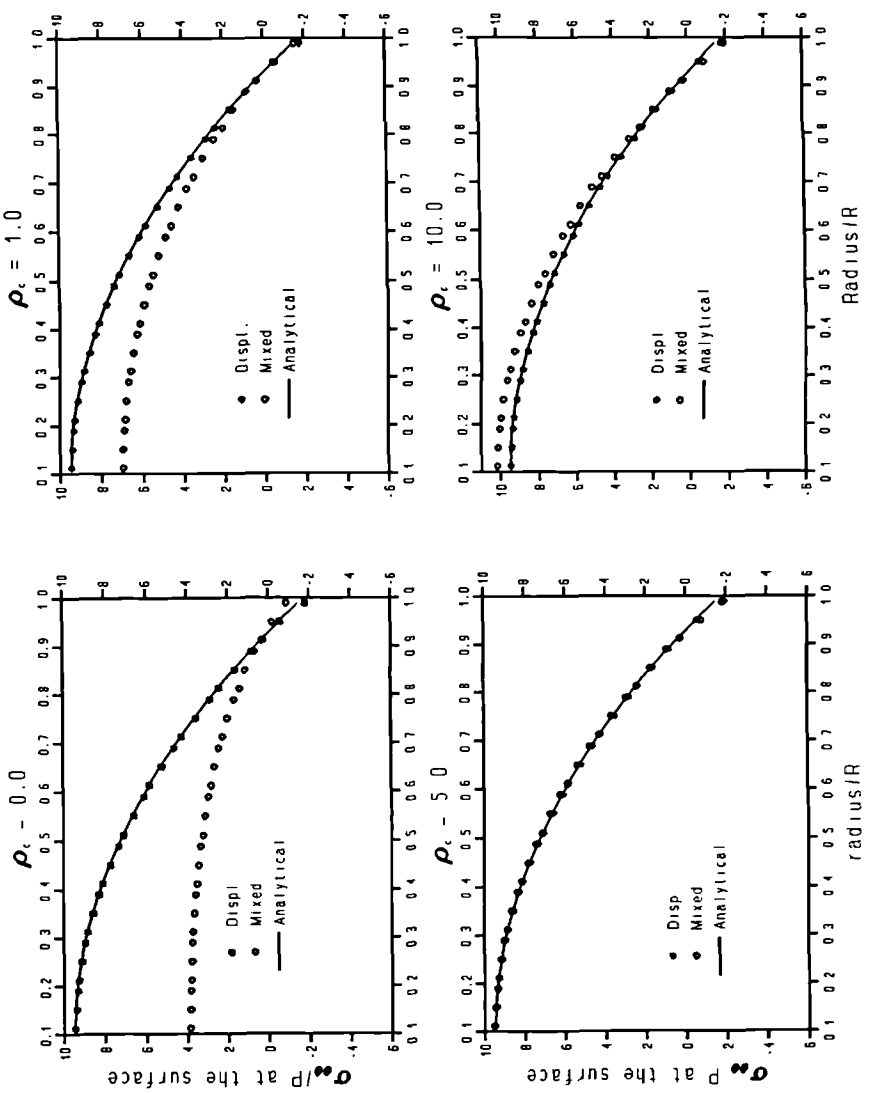


Fig.4.5 Distributions of $\sigma_{\theta\theta}$ at the surface of a disc along *r*-axis under uniformly pressure. The results are obtained by a FE mesh with 80 8-noded elements and 3×3 integration.

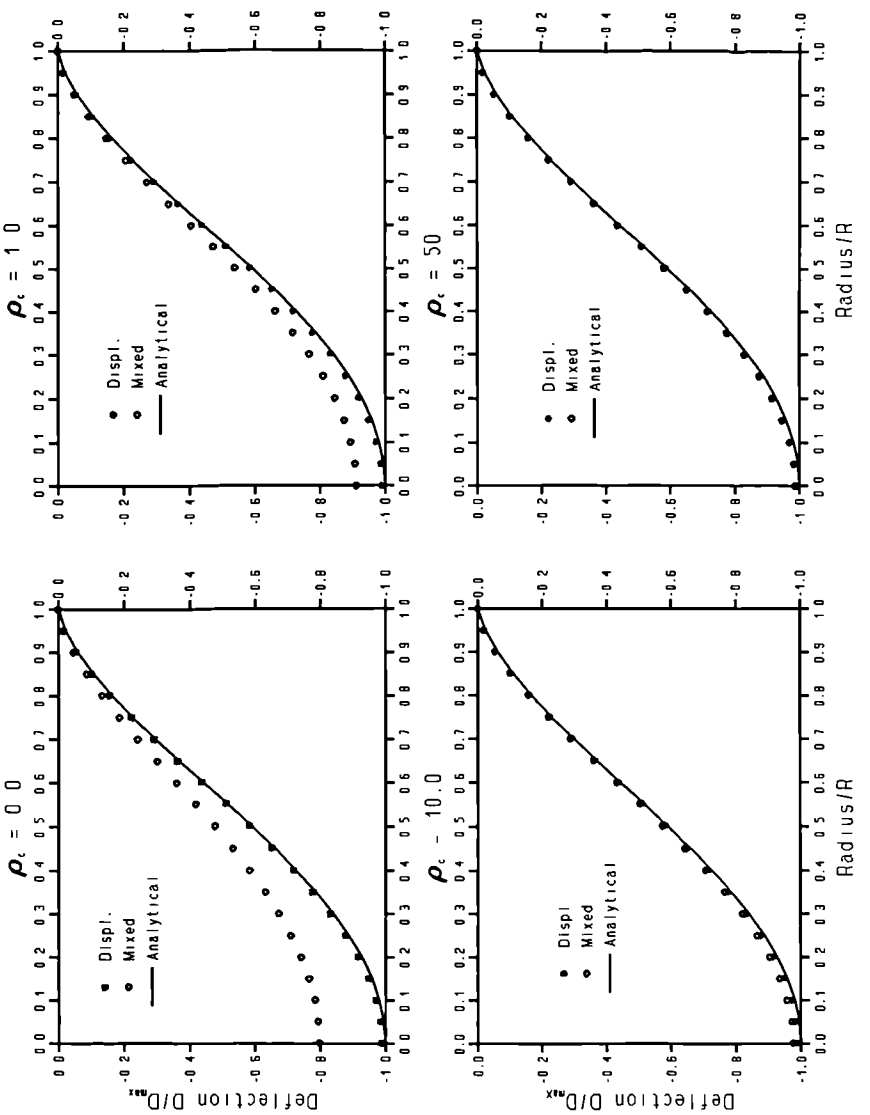


Fig.4.6 *z*-deflections of the disc along *r*-axis under uniformly pressure. The results are obtained by a FE mesh with 80 8-noded elements and 2×2 integration.

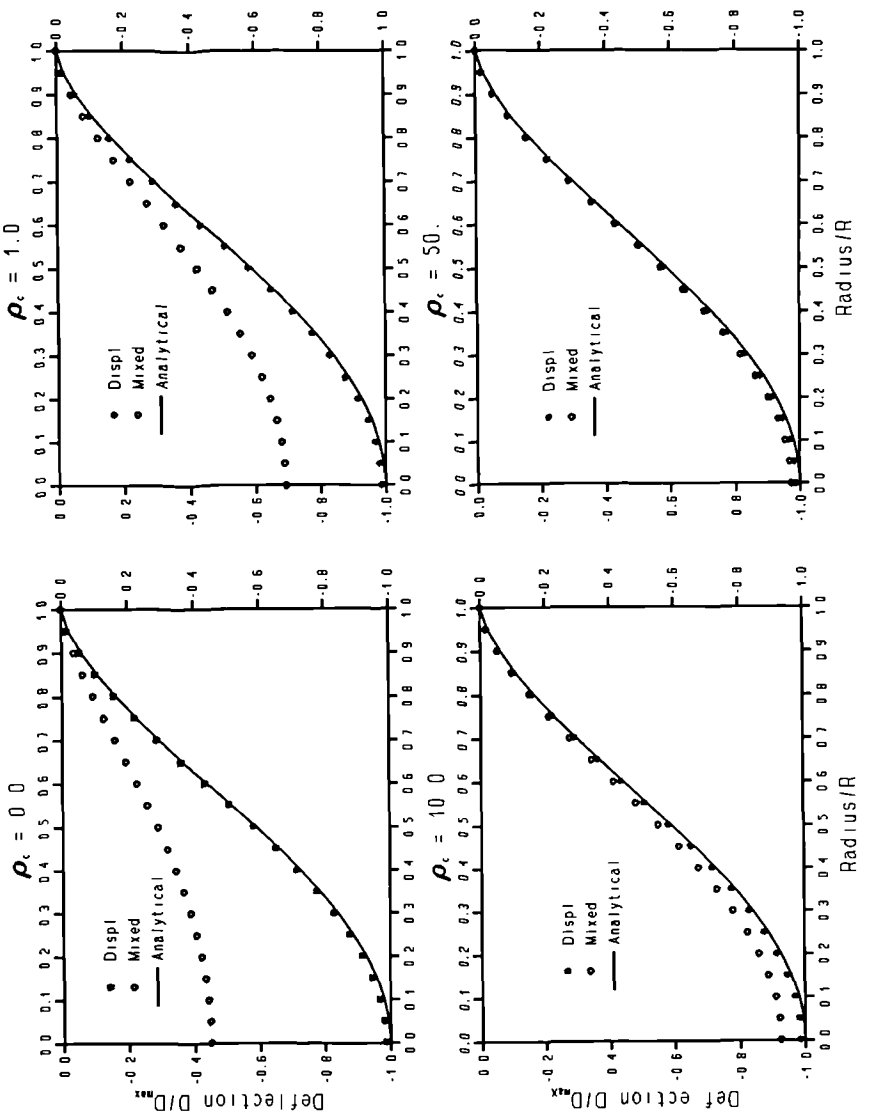


Fig.4.7 z-deflections of the disc along *r*-axis under uniformly pressure. The results are obtained by a FE mesh with 80 8-noded elements and 3×3 integration.

4.5.2 Spherical Container under Internal Uniform Pressure

A sphere with internal radius a and external radius $b=2a$ in which there is an internal pressure p are analysed in this section. It is straightforward to solve this problem by applying the same finite element mesh in fig.3.3 in section 3.6.1 for a hollow cylinder case. Various tests are carried out to note the effects on stress of different integration schemes, element sizes and Poisson's ratio. The results are shown in fig.4.8 to fig.4.13.

For the 50 element mesh, fig.4.8 and fig.4.9 show the errors of σ_{rr} and $\sigma_{\theta\theta}$ along r-axis under 2×2 and 3×3 integrations respectively. Unlike the expansion of a cylinder where similar figures between displacement and mixed models are found with 2×2 integration for various Poisson's ratios ν, in this example mixed model shows much better results than those from the displacement model when $\nu \geq 0.4$, even with 2×2 integration. For 3×3 cases, the mixed model naturally shows its consistent merits over the displacement one. In fig.4.9 it is noted that $\sigma_{\theta\theta}$ from the mixed model is poorer that σ_{rr}.

Fig.4.10 and fig.4.11 give the convergence of both models against the number of elements in the r-axis with either integration scheme. Four FE meshes are used to obtain these curves and the average error defined in eq.(3.43) is used to measure the error in each solution. Again the mixed one shows consistency under different Poisson ν and when a fine mesh is used the error approaches zero. This does not happen in the results from the displacement model where the convergence varies with ν and for some cases, i.e. in fig.4.10, it converges very slowly. For a specific mesh, the mixed model always yields less error than the displacement one.

The consistencies of the average error and the value of the functional Π_{elas} of the mixed model are presented in fig.4.12 for 2×2 integration and fig.4.13 for 3×3 one. The former shows some relations between them but it is not very good since the error in 2×2 case is so little that the calculations of both the average error and Π_{elas} can be affected by the computer precision. The good results are found in fig.4.13 for 3×3 integration. Therefore an error estimator also exists in the mixed model for axisymmetric problem which is the value of the functional Π_{elas}.

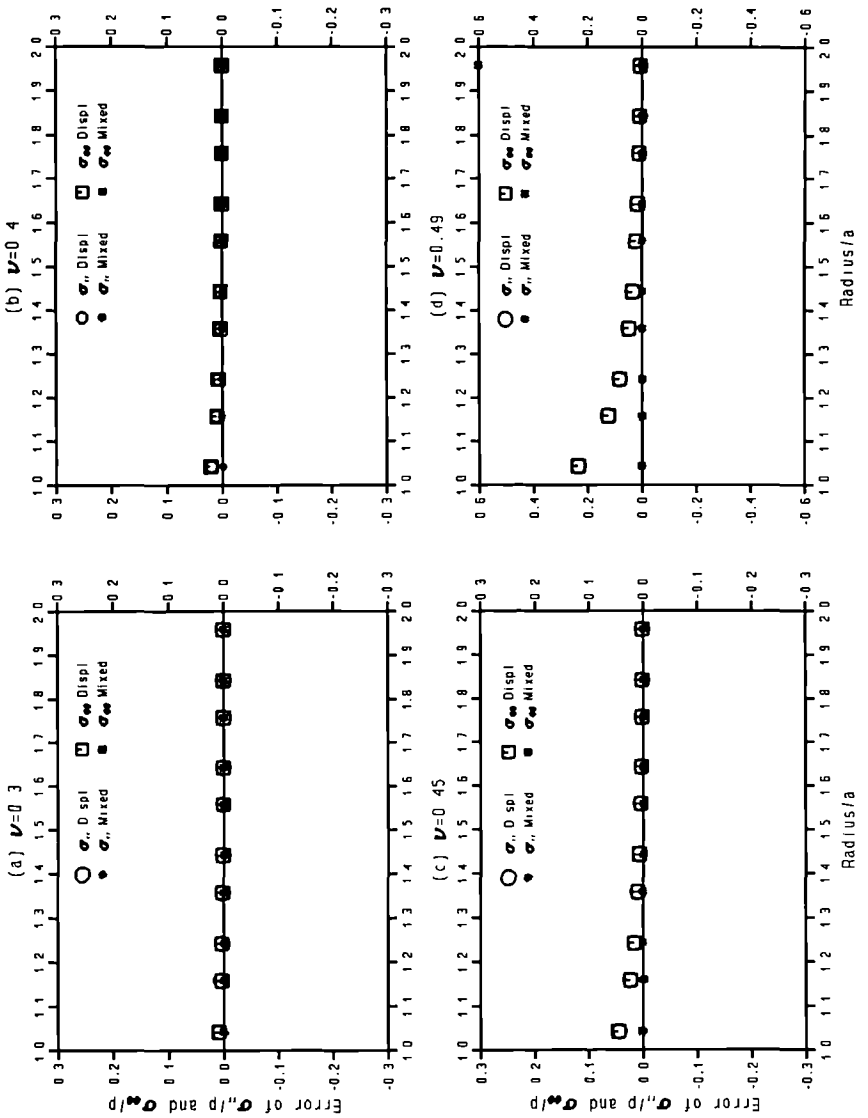


Fig.4.8 Errors of σ_{rr} and $\sigma_{\theta\theta}$ in the mixed model and the displacement model when 80 elements and 2x2 integration is used.

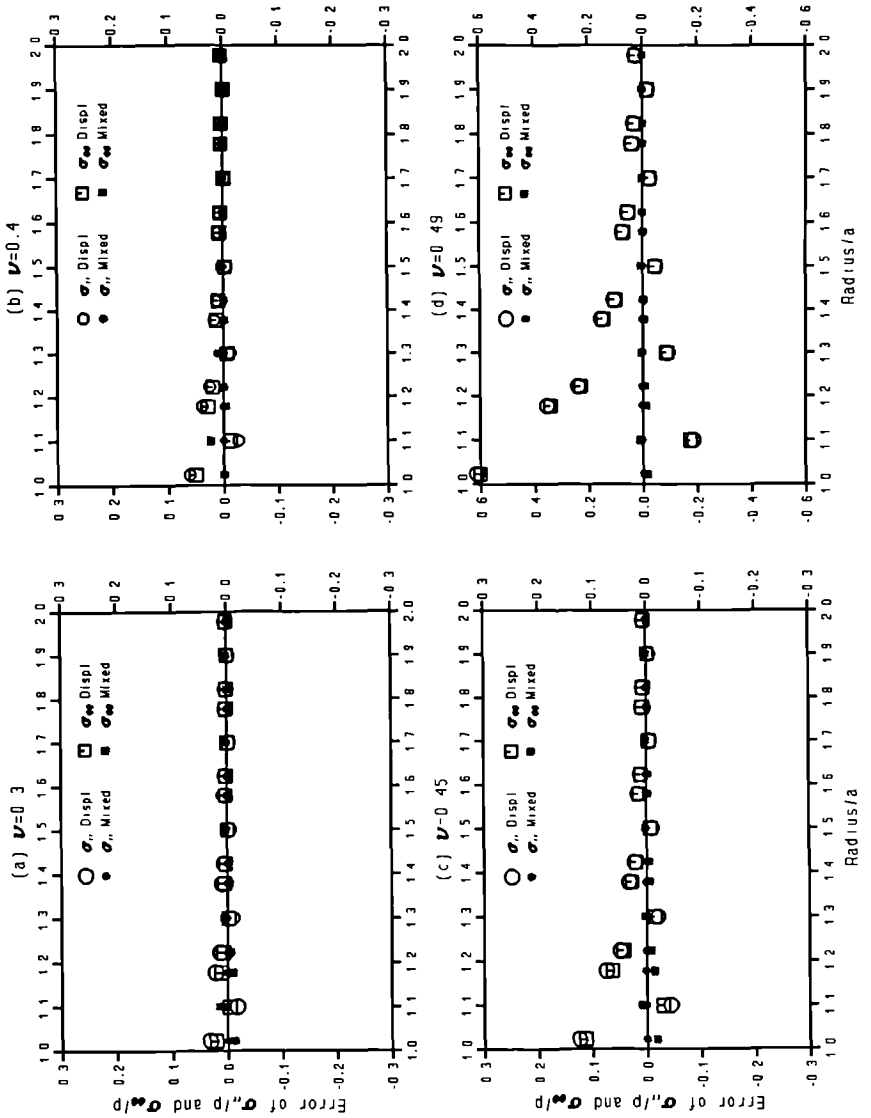


Fig.4.9 Errors of σ_{rr} and $\sigma_{\theta\theta}$ in the mixed model and the displacement model when 80 elements and 3x3 integration is used.

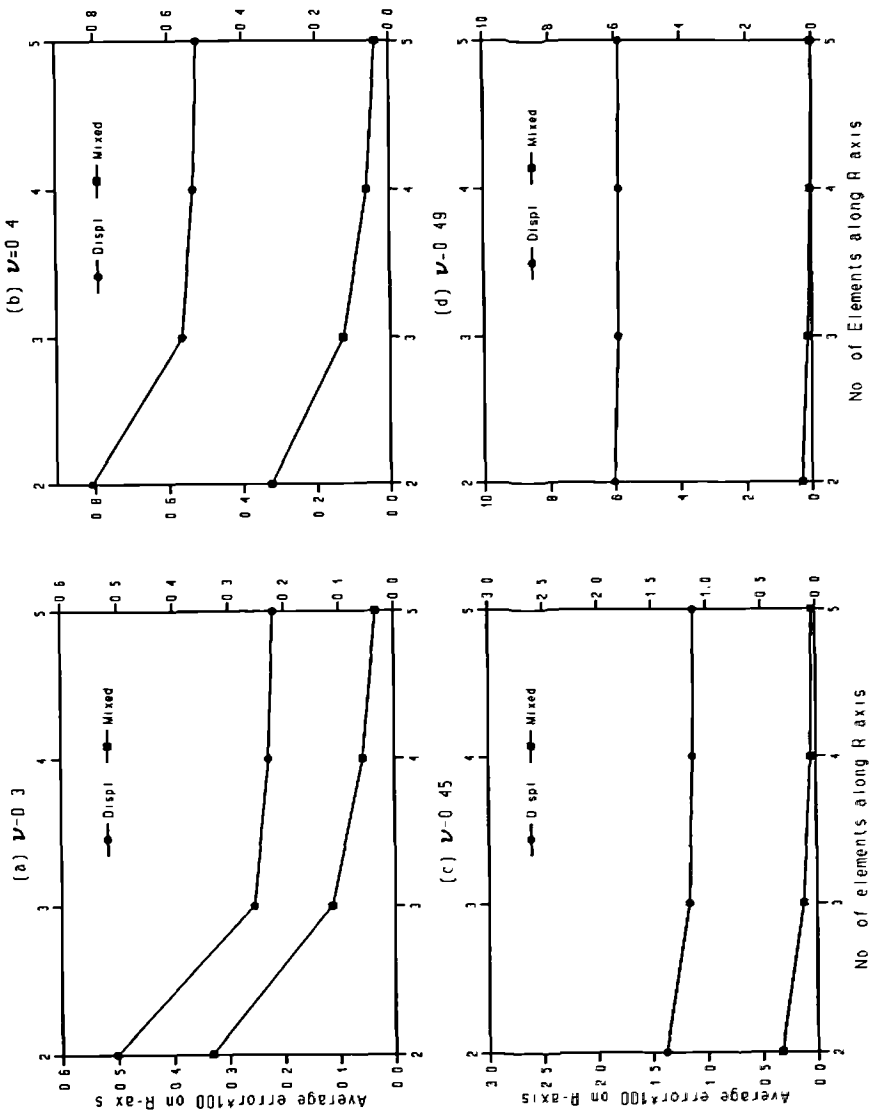


Fig.4.10 Convergencies of the displacement model and the mixed model in terms of the average error varying with the number of elements in r-axis when 2×2 integration is used.

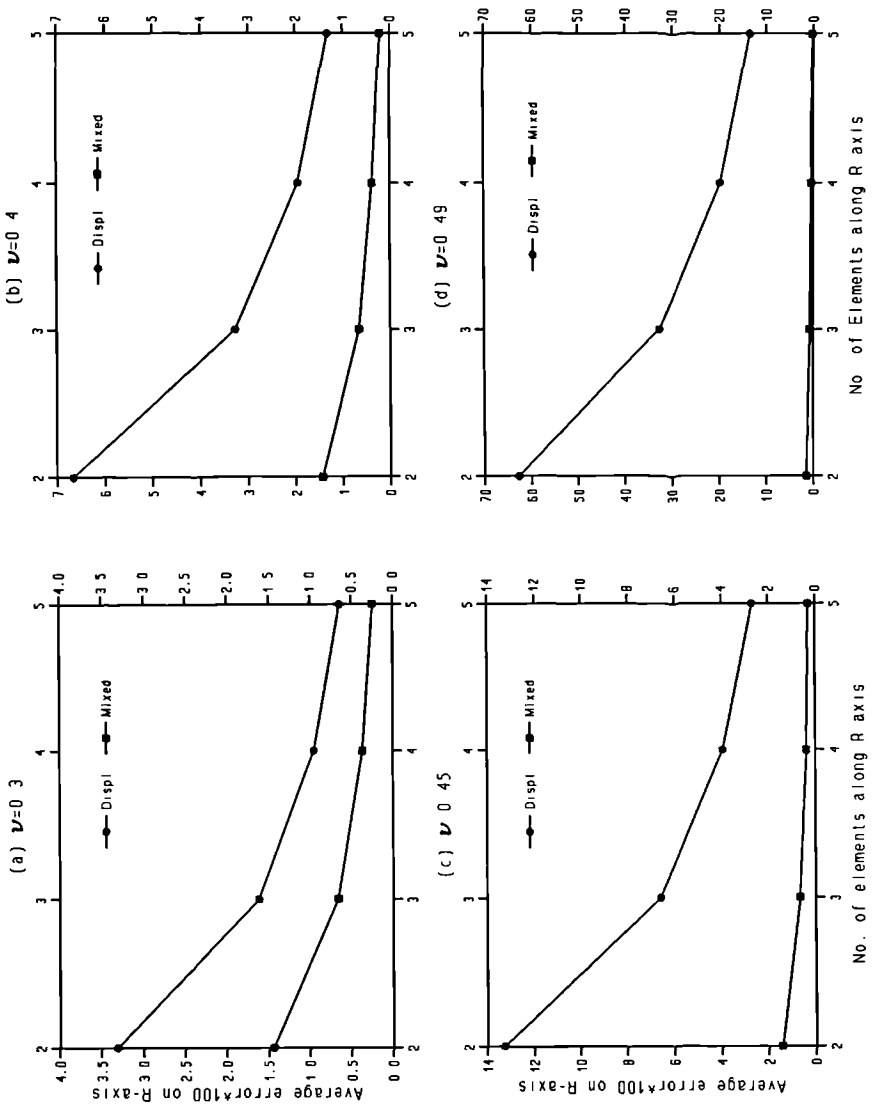


Fig.4.11 Convergencies of the displacement model and the mixed model in terms of the average error varying with the number of elements in r-axis when 3×3 integration is used.

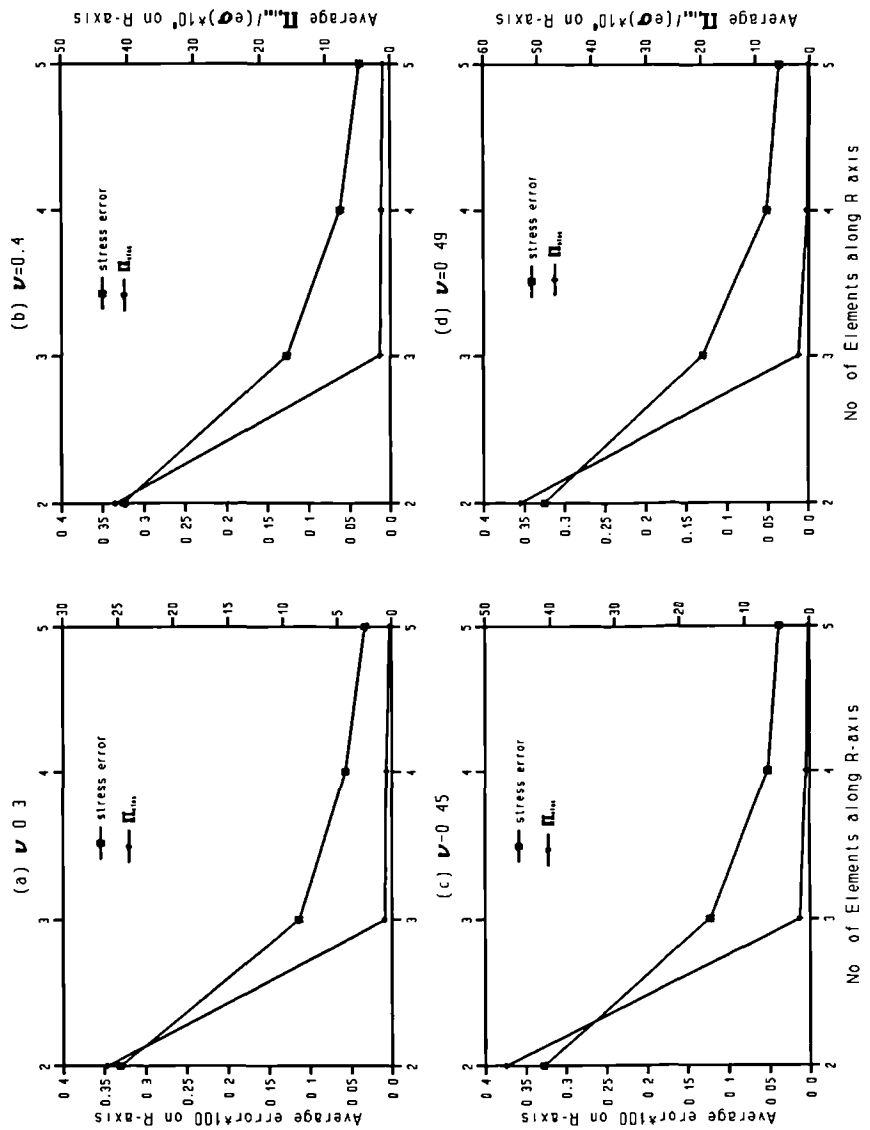


Fig.4.12 Comparison of the average error and the value of Π_{elas} varying with the number of elements in r-axis in the mixed model when 2x2 integration is used.

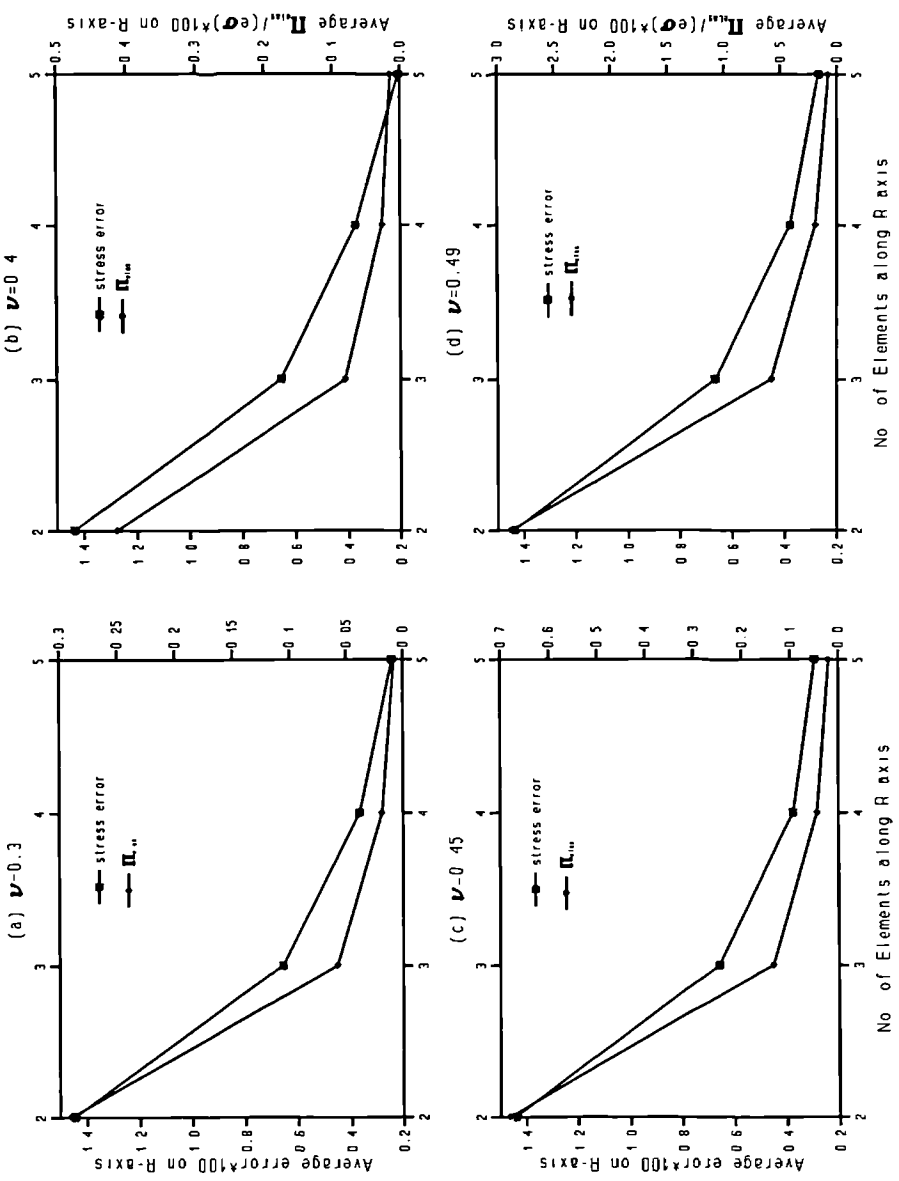


Fig.4.13 Comparison of the average error and the value of Π_{elas} varying with the number of elements in *r*-axis in the mixed model when 3x3 integration is used.

4.5.3 Electrode Squeezing Aluminium Sheets

In the last two examples, we dedicated ourselves to some classical elementary problems, in order to illustrate the validity of this model to some basic axisymmetric problems. A more complicated problem will be dealt with in this section, which is a pair of aluminium sheets squeezed by a pair of electrodes (fig.4.14a).

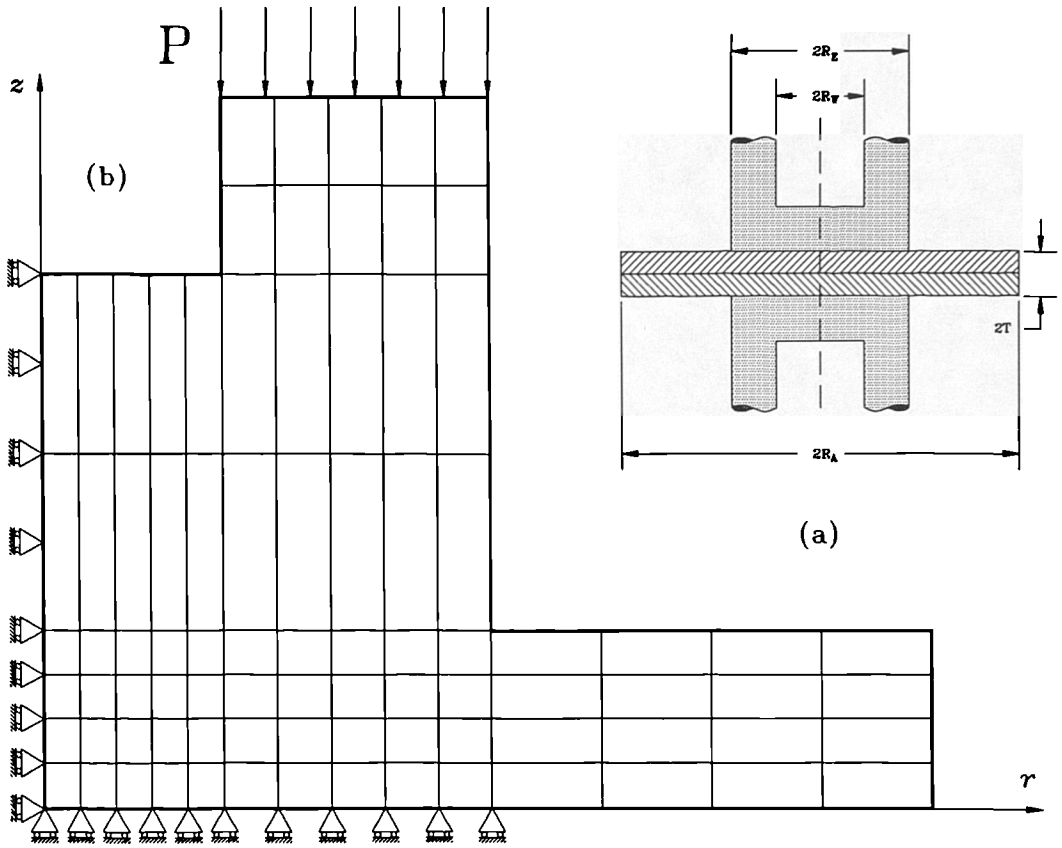


Fig.4.14 The layout of a pair of electrodes and aluminium sheets (a) and the FE model for the analysis (b).

Because of the symmetry only one quarter of the whole structure is modelled, and there is no movement of the structure at r=0 at r direction, nor at z=0 at z direction. The load is represented by a uniform pressure at a section of electrode. The FE mesh and constraints are shown in fig.4.14b. There are 30 elements used to model the electrode and another 56 ones used to model the aluminium sheet. In the first instance the 4-noded isoparametric element is used in the FE model.

The contact condition between the electrode and aluminium sheet is assumed as no-

slip, i.e. the friction at the surface is large enough to prevent any relative movement at the contact surface. On the other hand no friction exists at the faying surface since there is no relative movement at a symmetric surface.

The material properties for the copper electrode are Young's modulus E_c=120GPa and Poisson's ratio ν_c=0.25. For the aluminium sheet they are E_{al}=75GPa and ν_{al}=0.33. The thickness of the aluminium sheet is T=2mm, the radius concerned is R_S=10mm. The radius of the water cooling chamber is R_W=2.0mm and the radius of the electrode itself is R_E=5.0.

When the load is applied, the separation of the two aluminium sheets is also found, as we can anticipate. This causes the size of contact area at both the electrode surface and the faying surface vary with loading, which results in the change of electric resistance.

Deformations of both the aluminium sheet and the electrode predicted by the mixed model and the displacement one with 4-noded elements are shown as in fig.4.15 and fig.4.16 respectively. Almost identical results can be found though the mixed one gives slightly smaller separation in the outer range of the aluminium sheet than the displacement method. This can be cured by using a larger penalty number ρ_c.

The results of the pressure and shear distributions at the electrode surface and the faying surface are shown in fig.4.17. Both mixed model and displacement model give similar results in most areas. However the gap is wider near the edge of the electrode. Generally speaking, the former gives smooth σ_{rz} but bad σ_{zz}, while the latter gives smooth σ_{zz} but jerky σ_{rz}. Therefore it is natural to find that the mixed model obviously gives worse results of normal stress (σ_{zz}) than the displacement model at the faying surface, as in fig.4.18.

Now we use 8-node isoparametric element to model this problem with the same mesh. At the electrode surface (fig.4.19), the shear stress σ_{rz} virtually coincide each other in the most area except in the last element at the edge. The displacement model tends to give a rapidly increasing shear stress in that region, while the mixed model gives a near zero shear stress. This situation needs to be further studied.

The curves for normal stress σ_{zz} are also close to each other. The most significant results come from normal stress σ_{zz} at the faying surface in fig.4.20. The curves for two methods are consistently close through the whole range.

A more complicated model of the spot welding will be dealt with in chapter seven where elasto-plasticity is included.

4.5.4 Comments on the Mixed Model for Axisymmetric Elasticity

Three axisymmetric problems, axial expansion, transverse bending and later a complex load modes i.e. squeeze of aluminium sheets by electrodes, have just been discussed to demonstrate the performance of three C^0 stress functions. Except for the bending of a disc (section 4.5.2) where results depend on the choice of penalty number ρ_c, in the rest two examples, the mixed model gives very good stress and displacement results on various mesh sizes and integration schemes. As in plane problems, the penalty number is a constant depending merely on the integration scheme, i.e. ρ_c=5.0 for 2×2 and ρ_c=10.0 for 3×3.

For near incompressible materials, the mixed model gives much better results over the displacement model under both 2×2 and 3×3 integrations, rather than only under 3×3 one in plane problems (section 4.5.1).

When 4-noded elements are used in the solution, mixed model tends to approximate the σ_{zz} as constant function within an element. Thus the results turn to be discontinuous. A similar problem exists in the results of σ_{rz} from the displacement model by 4-noded elements. Therefore if an accurate σ_{zz} is needed 8-noded elements must be employed in the mixed model.

A similar error estimator exists in the mixed model which is value of functional Π_{elas}. The larger it is the bigger error a solution has; the smaller it the less error a solution has. The zero corresponds to the exact solution.

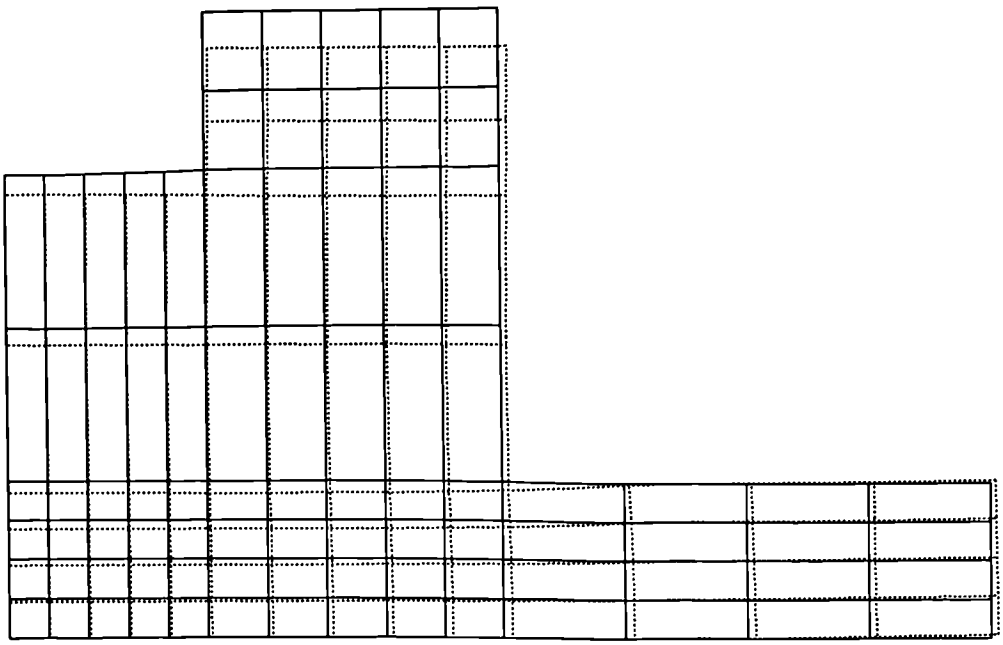


Fig.4.15 Deformation of the FE mesh of the electrode and the aluminium sheet in spot welding, obtained by mixed models.

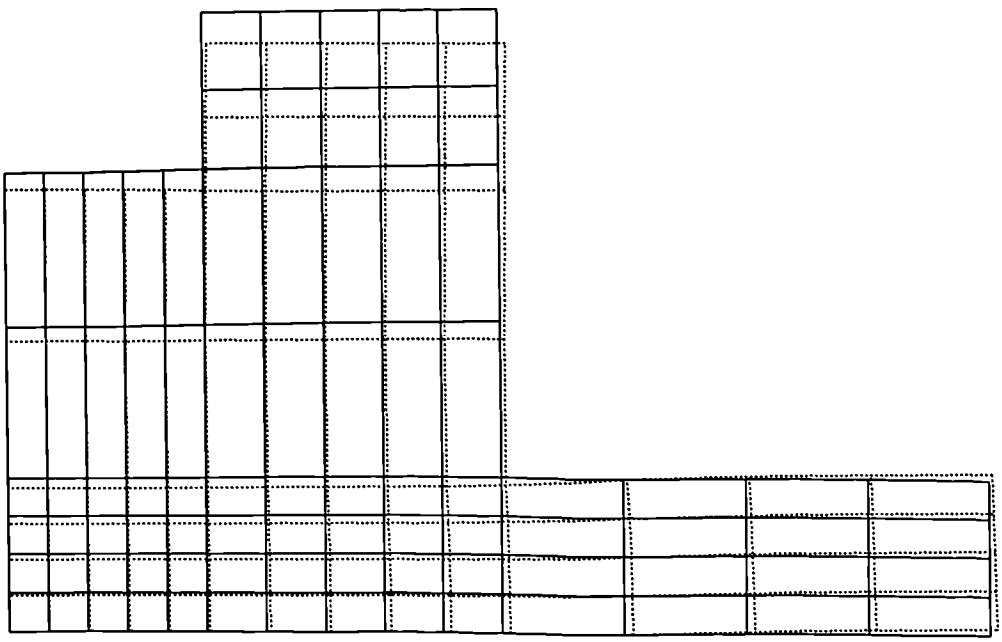


Fig.4.16 Deformation of the FE mesh of the electrode and the aluminium sheet in spot welding, obtained by displacement models.

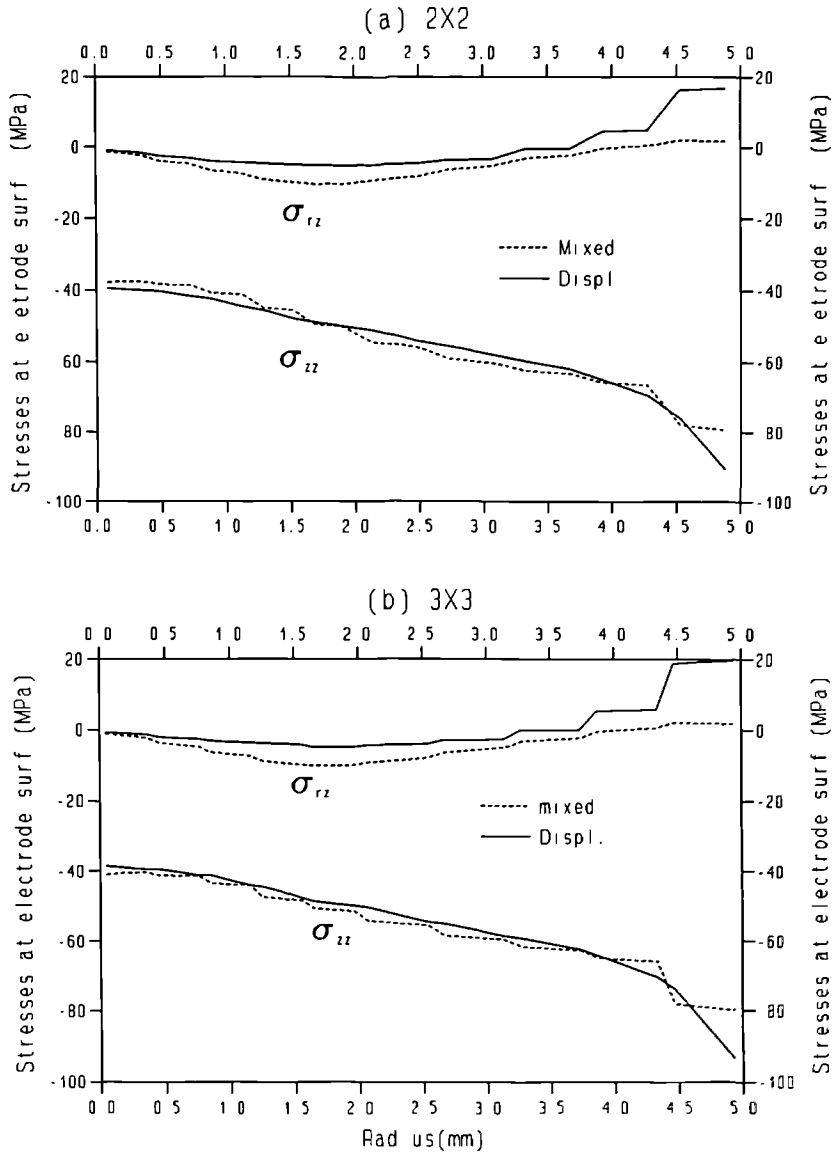


Fig.4.17 σ_{zz} and σ_{rz} on the electrode surface. The FE mesh is made of 86 4-noded isoparametric elements. (a) 2×2 integration; (b) 3×3 integration.

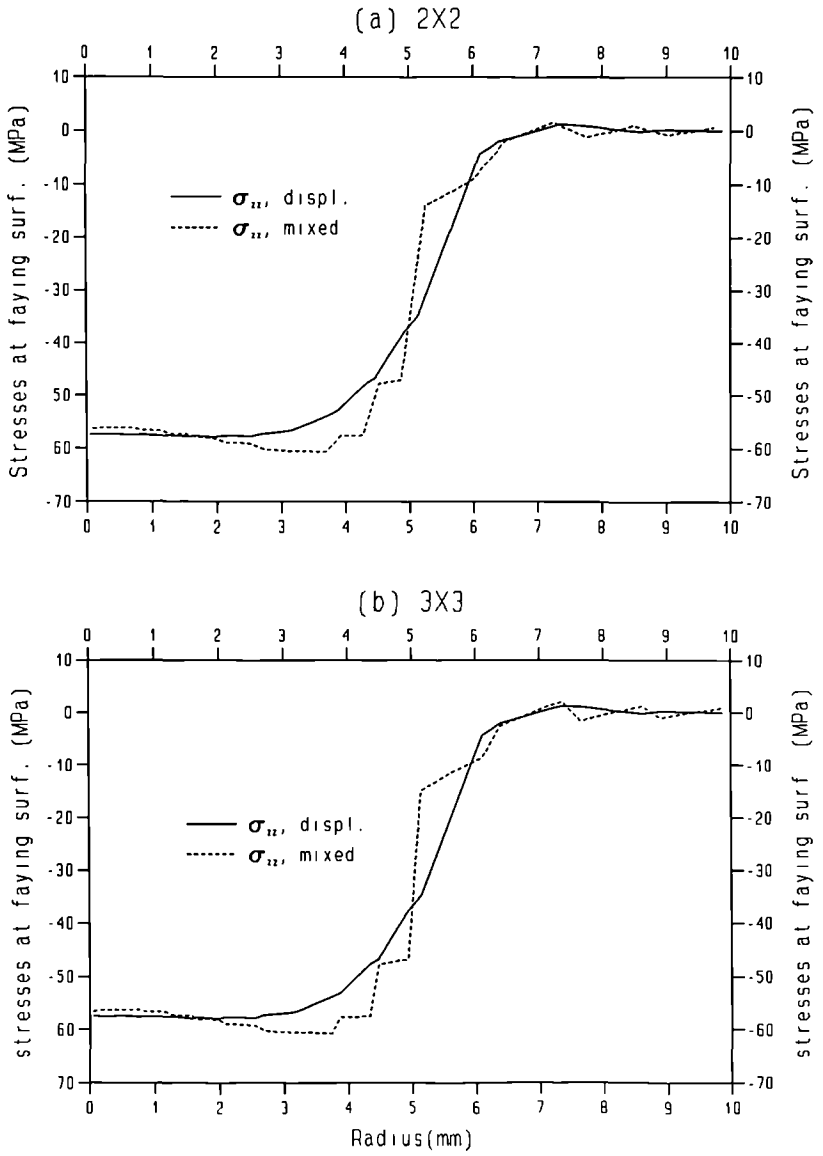


Fig.4.18 σ_{zz} on the faying surface. The FE mesh is made of 86 4-noded isoparametric elements. (a) 2×2 integration; (b) 3×3 integration.

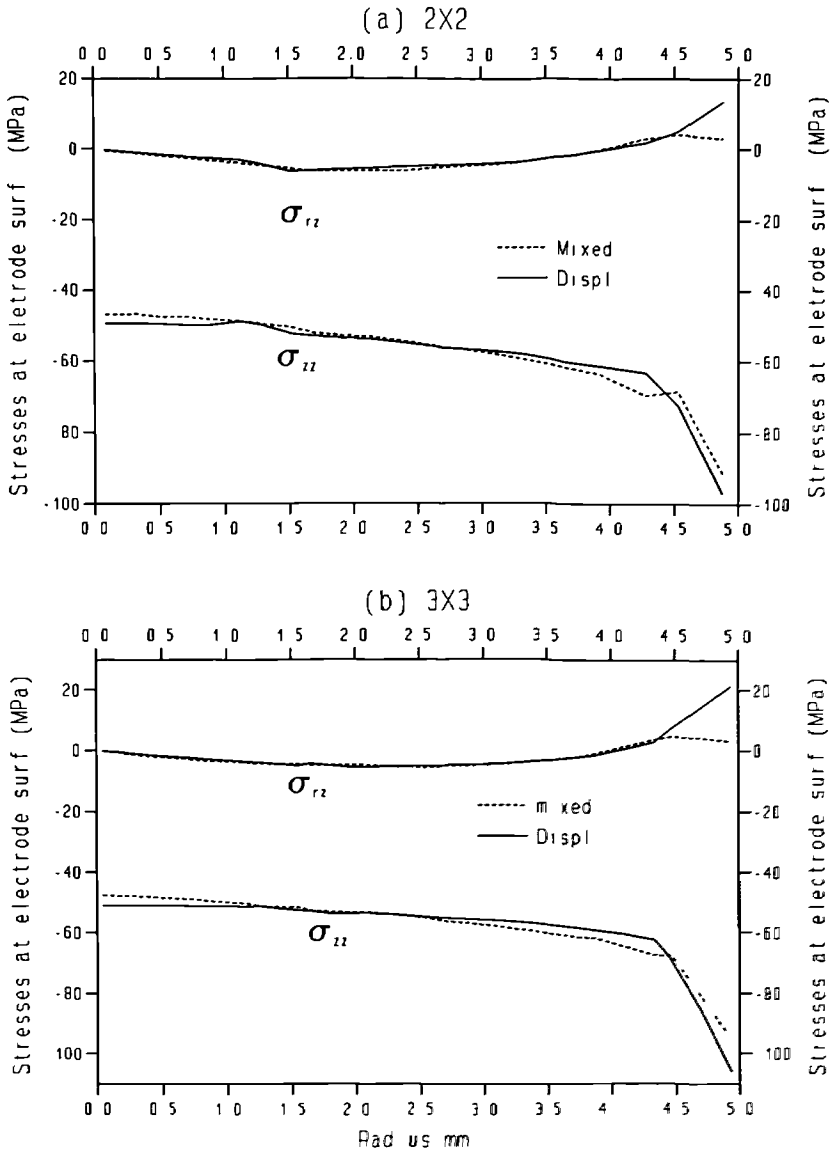


Fig.4.19 σ_{zz} and σ_{rz} on the electrode surface. The FE mesh is made of 86 8-noded isoparametric elements. (a) 2×2 integration; (b) 3×3 integration.

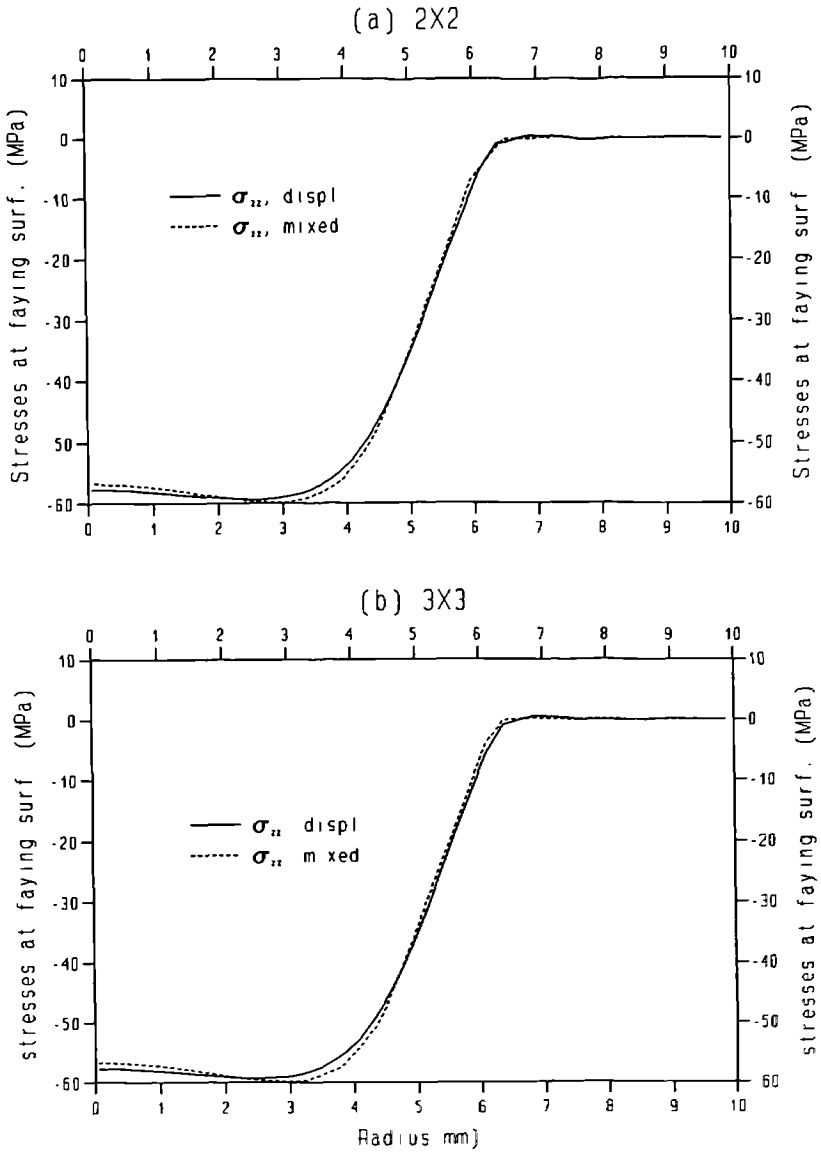


Fig.4.20 σ_{zz} on the faying surface. The FE mesh is made of 86 8-noded isoparametric elements. (a) 2×2 integration; (b) 3×3 integration.

CHAPTER 5
MIXED MODEL FOR RIGID-PERFECT-PLASTICITY

During structural collapse or in metal forming plastic strains are very much larger than elastic strains. Under these circumstances a useful approximation to the deformation field can be found by assuming that the material is either rigid or flowing plastically at a constant yield stress. This material model is commonly known as rigid-perfectly-plastic.

In this chapter, a mixed extremum principle is proposed and used to construct a mixed FE formulation for rigid-perfect-plasticity. This formulation forms a base on which the FE formulation for elasto-plasticity will be constructed in chapter 6.

5.1 MIXED EXTREMUM PRINCIPLE FOR RIGID-PERFECT-PLASTICITY

A problem in the theory of rigid-perfect-plasticity should be defined as follows. At a given time, a body Ω, composed of rigid-perfect-plastic material, is assumed to be in a state of quasi-static equilibrium. Now, the application of an external force F_i, $i=1,2,3$ is prescribed on a boundary Γ_s, while a surface velocity $\dot{u}_i$, $i=1,2,3$ is prescribed on a boundary Γ_u. Here, $\dot{u}_i$ denotes a component of the velocity with respect to the rectangular Cartesian coordinates. The stress σ_{ij} and velocities $\dot{u}_i$ induced in the body are unknown fields which need to be found. Thus the governing equations for the problem are expressed as

- *Equation of equilibrium* $$\sigma_{ij,i} = 0 \tag{5.1}$$

- *Strain rate - velocity relations* $$\dot{e}_{ij} = \dot{u}_{i,j} + \dot{u}_{j,i} \tag{5.2}$$

- *Condition of incompressibility* $$\dot{e}_{ii} = 0 \tag{5.3}$$

- *Boundary conditions*

$$\begin{aligned} \sigma_{ij} n_i &= \overline{F_j} \quad \text{on} \quad \Gamma_s \\ \dot{u}_i &= \overline{\dot{u}_i} \quad \text{on} \quad \Gamma_u \end{aligned} \tag{5.4}$$

The basis of rigid-plasticity is that the stresses must lie on the yield surface for plastic flow to take place, otherwise the material is rigid. We must also consider strain increments instead of strains. As plasticity considered hence is rate dependent, we can use the dot notation to indicate an increment. The simplest constitutive relation of this type has two parts:

- *von Mises yield condition*

$$s_{ij} s_{ij} = 2k^2 \tag{5.5}$$

where s_{ij} is the component of the deviatoric stress defined in eq.(3.10) and k is a material constant with regard to the yield stress for rigid-perfect-plasticity;

- *Levy-Mises flow rule*

$$\dot{d}_{ij} = \frac{\dot{D} s_{ij}}{2k^2} \quad \text{where} \quad \dot{D} = \sqrt{2k^2 \dot{d}_{ij} \dot{d}_{ij}} \tag{5.6}$$

where $\dot{D}$ is the energy dissipation rate. This represents incompressible plastic flow and is appropriate for metal plasticity[19].

It is seen that the above problem is defined in a way similar to a linear elasticity theory, except for the stress-strain relations and introduction of velocity and strain rate. It is of note that no stress rates appear this theory. Once the problem in incremental theory has been thus formulated, problems of finite plastic deformation can be obtained by integrating the relations along the prescribed loading path.

As we did in plane elasticity, the kinematic condition eq.(5.2) and static condition eq.(5.1) are fully satisfied by the admissible displacement eq.(3.18) and stress eq.(3.24). The boundary conditions in eq.(5.4) will be handled in the same manner as before by two penalty function Π_d in eq.(3.30) and Π_s eq.(3.34). In order to impose the incom-pressibility condition eq.(5.3), the yield condition eq.(5.5) and the Levy-Mises equation eq.(5.6), the following functional is constructed

$$\Pi_{rgd} = \sqrt{\dot{d}_{ij}\dot{d}_{ij}}\sqrt{s_{ij}s_{ij}} - s_{ij}\dot{d}_{ij} + \rho\psi^2 + \rho(\dot{e}_m)^2 \tag{5.7}$$

where $\dot{e}_m$ is defined in the same way as e_m in eq.(3.10), but in a form of strain rate instead of strain, and ρ is a positive penalty number. ψ is a function defined as

$$\psi = \frac{\sqrt{s_{ij}s_{ij}}}{\sqrt{2}k} - 1 \tag{5.8}$$

It is straightforward to see that ψ is the criterion of plastic yield. If it is equal to zero then plastic flow can occur; otherwise the material is rigid. There results a mixed extremum principle which may be stated as follows:

Among all the admissible strains and stresses which satisfy the compatibility condition and equilibrium conditions, as well as the kinematic and static boundary conditions, the true solution renders the functional Π_{rgd} *an absolute minimum.*

This can be justified as follows. Let the exact deviatoric stress, strain rate and velocity of the exact solution be denoted by s_{ij}, $\dot{d}_{ij}$ and $\dot{u}_i$, and the deviatoric stress, strain rate and velocity of an admissible solution by s_{ij}^*, $\dot{d}_{ij}^*$ and $\dot{u}_i^*$. Then

$$s_{ij}^*\,\dot{d}_{ij}^* \le \sqrt{s_{ij}^*\,s_{ij}^*}\,\sqrt{\dot{d}_{ij}^*\,\dot{d}_{ij}^*} \tag{5.9}$$

by Schwarz's inequality, and

$$s_{ij}\dot{d}_{ij} = \sqrt{s_{ij}s_{ij}}\,\sqrt{\dot{d}_{ij}\dot{d}_{ij}} \tag{5.10}$$

since s_{ij} and $\dot{d}_{ij}$ are proportional to each other. The admissible solution by s_{ij}^*, $\dot{d}_{ij}^*$ also yields

$$\psi^{*2} = \left[\frac{\sqrt{s_{ij}^*\,s_{ij}^*}}{\sqrt{2}k} - 1\right]^2 \ge \psi^2 = \left[\frac{\sqrt{s_{ij}s_{ij}}}{\sqrt{2}k} - 1\right]^2 \equiv 0 \tag{5.11}$$

and

$$\dot{e}_m^{*2} \ge \dot{e}_m^2 \equiv 0$$

Combining eq.(5.9), eq.(5.10) and eq.(5.11), we obtain

$$\begin{aligned} 0 &\equiv \sqrt{s_{ij}s_{ij}}\sqrt{\dot{d}_{ij}\dot{d}_{ij}} - s_{ij}\dot{d}_{ij} + \rho\psi^2 + \rho\dot{e}_m^2 \\ &\le \sqrt{s_{ij}^* s_{ij}^*}\sqrt{\dot{d}_{ij}^* \dot{d}_{ij}^*} - s_{ij}^* \dot{d}_{ij}^* + \rho\psi^{*2} + \rho\dot{e}_m^{*2} \end{aligned} \tag{5.12}$$

Therefore the functional Π_{rgd} has an extremum of zero.

At the extremity, there is no doubt that eq.(5.10) is satisfied. If we pre-multiply both sides of eq.(5.10) by s_{ij},

$$(s_{kl}s_{kl})\dot{d}_{ij} = (\sqrt{s_{ij}s_{ij}}\sqrt{\dot{d}_{ij}\dot{d}_{ij}})s_{ij}$$

or

$$\dot{d}_{ij} = \frac{\sqrt{\dot{d}_{ij}\dot{d}_{ij}}\sqrt{s_{ij}s_{ij}}}{s_{ij}s_{ij}}s_{ij}$$

equation (5.12) is immediately transformed to the Levy-Mises flow rule eq.(5.6) by the fact that at the extremity s_{ij} must obey the von Mises yield condition eq.(5.5).

Origin of this functional is known paper by Markov[66].

5.2 FE FORMULATION FOR RIGID-PERFECT-PLASTICITY

Approximations of velocity $\dot{u}_i$ and stress σ_{ij} are obviously independent of constitutive equations. Therefore eqs.(3.18), (3.19), (3.24) and (3.25) for plane elasticity and eqs.(4.14), (4.15), (4.16) and (4.17) for axisymmetric elasticity are directly applicable to the formulation for rigid plasticity. However the stiffness matrix, being closely related to the material properties, is completely different.

5.2.1 Calculation of Stiffness Matrix

The fact that the functional Π_{rgd} is not a quadratic function of the stress and strain requires that the extremum of it be found by solving a non-linear set of equations. For various nonlinear solution procedure, the Hessian matrix, which is the matrix of second-order differential of Π_{rgd}, is essential to the iteration procedure. In linear elasticity, the Hessian matrix is identical to the elastic stiffness matrix, while in elasto-plasticity, it is elasto-plastic stiffness matrix.

There are two ways to get the second-order differential of a function. The

analytical approach, which is accurate but algebraically complicated, and the numerical approach, which is usually simpler to obtain but is only an approximation. Normally an analytical method is more efficient than the corresponding numerical one. However, if the analytical differential turns out to be very complicated, and a lot of arithmetic operations are needed, the efficiency may be reduced. It is not impossible that the efficiency of an analytical approach reduces so much that it becomes even lower that the numerical one. In that case, the better choice might be a numerical one. Therefore, before we code the program, the efficiencies of both approaches must be compared for the particular functional Π_{rgd}.

Let us start with the numerical one. For the convenience of FE formulation, we will employ matrix notations rather than tensor notations from now on. Two vectors $\dot{\mathbf{d}}$ and $\mathbf{s}$ with the components $\dot{d}_i$ and s_i respectively, representing deviatoric strain rate and deviatoric stress, are defined in the same way as in sections 3.2 and 3.3 for plane elasticity. It is easy to see that functional Π_{rgd} in eq.(5.7) is a function of $\dot{\mathbf{d}}$, $\mathbf{s}$ and $\dot{e}_m$. Let the following terms be defined,

$$L_{\dot{d}}^{i} = \frac{\partial \Pi_{rgd}}{\partial \dot{d}_i} \quad ; \quad L_{s}^{i} = \frac{\partial \Pi_{rgd}}{\partial s_i} \quad ; \quad L_{\dot{e}_m} = \frac{\partial \Pi_{rgd}}{\partial \dot{e}_m}$$

$$L_{\dot{d}\dot{d}}^{ij} = \frac{\partial^2 \Pi_{rgd}}{\partial \dot{d}_i \partial \dot{d}_j} \quad ; \quad L_{\dot{d}s}^{ij} = \frac{\partial^2 \Pi_{rgd}}{\partial \dot{d}_i \partial s_j} \quad ; \quad L_{ss}^{ij} = \frac{\partial^2 \Pi_{rgd}}{\partial s_i \partial s_j}$$

Three first-order differentials with respect to $\dot{\mathbf{d}}$, $\mathbf{s}$ and $\dot{e}_m$ can be obtained by the central-difference formula[122]

$$\begin{aligned} L_{s}^{i}(\mathbf{s},\dot{\mathbf{d}},\dot{e}_m) &= \frac{\Pi_{rgd}(\mathbf{s}+\Delta_{\mathbf{i}},\dot{\mathbf{d}},\dot{e}_m) - \Pi_{rgd}(\mathbf{s}-\Delta_{\mathbf{i}},\dot{\mathbf{d}},\dot{e}_m)}{2c} \\ L_{\dot{d}}^{i}(\mathbf{s},\dot{\mathbf{d}},\dot{e}_m) &= \frac{\Pi_{rgd}(\mathbf{s},\dot{\mathbf{d}}+\Delta_{\mathbf{i}},\dot{e}_m) - \Pi_{rgd}(\mathbf{s},\dot{\mathbf{d}}-\Delta_{\mathbf{i}},\dot{e}_m)}{2c} \\ L_{\dot{e}_m}(\mathbf{s},\dot{\mathbf{d}},\dot{e}_m) &= \frac{\Pi_{rgd}(\mathbf{s},\dot{\mathbf{d}},\dot{e}_m+c) - \Pi_{rgd}(\mathbf{s},\dot{\mathbf{d}},\dot{e}_m-c)}{2c} \end{aligned} \tag{5.13}$$

where $\Delta_{\mathbf{i}}$ is a vector with components $\Delta_k = \delta_{ik}\, c$. δ_{ik} is the Kronecker Delta and c is a constant of small value. Following the same routine, the second-order differentials of functional Π_{rgd} can be obtained by making use of the corresponding first-order

differentials

$$
\begin{aligned}
L^{ij}_{ss}(\mathbf{s},\dot{\mathbf{d}},\dot{e}_m) &= \frac{L^i_s(\mathbf{s}+\Delta_\mathbf{j},\dot{\mathbf{d}},\dot{e}_m) - L^i_s(\mathbf{s}-\Delta_\mathbf{j},\dot{\mathbf{d}},\dot{e}_m)}{2c} \\
L^{ij}_{\dot{d}\dot{d}}(\mathbf{s},\dot{\mathbf{d}},\dot{e}_m) &= \frac{L^i_{\dot{d}}(\mathbf{s},\dot{\mathbf{d}}+\Delta_\mathbf{j},\dot{e}_m) - L^i_{\dot{d}}(\mathbf{s},\dot{\mathbf{d}}-\Delta_\mathbf{j},\dot{e}_m)}{2c} \\
L^{ij}_{\dot{d}s}(\mathbf{s},\dot{\mathbf{d}},\dot{e}_m) &= \frac{L^i_{\dot{d}}(\mathbf{s}+\Delta_\mathbf{j},\dot{\mathbf{d}},\dot{e}_m) - L^i_{\dot{d}}(\mathbf{s}-\Delta_\mathbf{j},\dot{\mathbf{d}},\dot{e}_m)}{2c} \\
L^{ij}_{s\dot{d}}(\mathbf{s},\dot{\mathbf{d}},\dot{e}_m) &= \frac{L^i_s(\mathbf{s},\dot{\mathbf{d}}+\Delta_\mathbf{j},\dot{e}_m) - L^i_s(\mathbf{s},\dot{\mathbf{d}}-\Delta_\mathbf{j},\dot{e}_m)}{2c}
\end{aligned}
\tag{5.14}
$$

It is straightforward to get the following differential with respect to $\dot{e}_m$ by simple observation

$$
\frac{\partial \Pi_{rgd}}{\partial \dot{e}_m \partial \dot{e}_m} = -2\rho \quad ; \quad \frac{\partial \Pi_{rgd}}{\partial s_i \partial \dot{e}_m} = 0 \quad ; \quad \frac{\partial \Pi_{rgd}}{\partial \dot{d}_i \partial \dot{e}_m} = 0 \tag{5.15}
$$

If Π_{rgd} is a continuous function, the third and fourth equations in eq.(5.14) must yield identical results. In appendix A, results of the differential of Π_{rgd} with the numerical approach are presented.

Now let us turn our attention to the analytical method. If we look up the functional Π_{rgd} in eq.(5.7) carefully it is easy to see that the difficulty of the differentials comes from the terms of the form

$$
\mathrm{f}(\mathbf{x}) = \sqrt{x_i x_i} \quad ; \quad i=1,5 \tag{5.16}
$$

From eq.(5.16), the functional Π_{rgd} becomes

$$
\begin{aligned}
\Pi_{rgd} &= \mathrm{f}(\dot{\mathbf{d}})\,\mathrm{f}(\mathbf{s}) - \mathbf{s}^{\mathrm{T}}\dot{\mathbf{d}} + \rho\psi^2 + \rho\dot{e}_m^2 \\
\text{where} \qquad \psi &= \frac{\mathrm{f}(\mathbf{s})}{\sqrt{2}\,k} - 1
\end{aligned}
\tag{5.17}
$$

Let,

$$
f'_i(\mathbf{x}) = \frac{d\mathrm{f}(\mathbf{x})}{dx_i} \quad \text{and} \quad f''_{ij}(\mathbf{x}) = \frac{d^2\mathrm{f}(\mathbf{x})}{dx_i dx_j}
$$

then differentiating eq.(5.17) with respect to $\dot{\mathbf{d}}$, $\mathbf{s}$ and $\dot{e}_m$ twice gives

$$L_s^i = \mathrm{f}(\dot{\mathbf{d}}) f_i'(\mathbf{s}) - \dot{d}_i + 2\rho\psi \frac{\partial \psi}{\partial s_i}$$
$$L_{\dot{d}}^i = \mathrm{f}(\mathbf{s}) f_i'(\dot{\mathbf{d}}) - s_i + 2\rho\psi \frac{\partial \psi}{\partial \dot{d}_i}$$
$$L_{\dot{e}_m} = 2\rho \dot{e}_m$$

and

$$\begin{aligned} L_{ss}^{ij} &= \mathrm{f}(\dot{\mathbf{d}}) f_{ij}''(\mathbf{s}) + 2\rho \left(\psi \frac{\partial^2 \psi}{\partial s_i \partial s_j} + \frac{\partial \psi}{\partial s_i} \frac{\partial \psi}{\partial s_j}\right) \\ L_{\dot{d}\dot{d}}^{ij} &= \mathrm{f}(\mathbf{s}) f_{ij}''(\dot{\mathbf{d}}) + 2\rho \left(\psi \frac{\partial^2 \psi}{\partial \dot{d}_i \partial \dot{d}_j} + \frac{\partial \psi}{\partial \dot{d}_i} \frac{\partial \psi}{\partial \dot{d}_j}\right) \\ L_{\dot{d}s}^{ij} &= -\delta_{ij} + f_i'(\dot{\mathbf{d}}) f_j'(\mathbf{s}) + 2\rho \left(\psi \frac{\partial^2 \psi}{\partial \dot{d}_i \partial s_j} + \frac{\partial \psi}{\partial \dot{d}_i} \frac{\partial \psi}{\partial s_j}\right) \end{aligned} \tag{5.18}$$

together with the differentials in eq.(5.15). In eq.(5.18), the derivatives of f(x) and ψ with respect to **x** and differentials of ψ with respect to **s** have to be deduced

$$f_i'(\mathbf{x}) = \frac{x_i}{\mathrm{f}(\mathbf{x})} \quad ; \quad f_{ij}''(\mathbf{x}) = \frac{\delta_{ij}}{\mathrm{f}(\mathbf{x})} - \frac{x_i x_j}{\mathrm{f}^3(\mathbf{x})} \tag{5.19}$$

and

$$\frac{\partial \psi}{\partial \dot{d}_i} = \frac{\partial^2 \psi}{\partial \dot{d}_i \partial \dot{d}_j} = \frac{\partial^2 \psi}{\partial \dot{d}_i \partial s_j} \equiv 0$$
$$\frac{\partial \psi}{\partial s_i} = \frac{1}{\sqrt{2}\,k} f_i'(\mathbf{s}) \quad ; \quad \frac{\partial^2 \psi}{\partial s_i \partial s_j} = \frac{1}{\sqrt{2}\,k} f_{ij}''(\mathbf{s})$$

Substituting them into eq.(5.18), it follows the differentials in terms of $f(\mathbf{x})$:

$$\begin{aligned} L_{ss}^{ij} &= \mathrm{f}(\dot{\mathbf{d}}) f_{ij}''(\mathbf{s}) + 2\rho \left\{\frac{1}{2k^2}\left[f(\mathbf{s}) f_{ij}''(\mathbf{s}) + f_i'(\mathbf{s}) f_j'(\mathbf{s})\right] - \frac{1}{\sqrt{2}\,k} f_{ij}''(\mathbf{s})\right\} \\ L_{\dot{d}\dot{d}}^{ij} &= \mathrm{f}(\mathbf{s}) f_{ij}''(\dot{\mathbf{d}}) \\ L_{\dot{d}s}^{ij} &= -\delta_{ij} + f_i'(\dot{\mathbf{d}}) f_j'(\mathbf{s}) \end{aligned} \tag{5.20}$$

A comparison of the accuracies and the computing time of these two approaches is presented in **Appendix A** when applied to a practical example. The numerical method shows numerical instability and is more time-consuming. It takes about 30 times more computing time than the analytical one to differentiate the Π_{rgd}. Together with the higher accuracy and efficiency, the analytical method obviously proves to be the better one for the functional Π_{rgd}.

5.2.2 Global System Equation

We have given a functional Π_{rgd} in section 5.1 (in order to impose the Levy-Mises flow rule, the von Mises yield condition and the incompressibility condition), and its differentials with respect to the deviatoric strain rate $\dot{\mathbf{d}}$, deviatoric stress $\mathbf{s}$ and volume strain rate $\dot{e}_m$. A complete problem in rigid plasticity also includes the satisfaction of some boundary conditions. Furthermore, since the same admissible stress is used as in elasticity, the constraints for moment equilibrium equation should again be imposed. Therefore a functional Π_{II} is constructed similar to that use to solve elastic problems

$$\Pi_{II} = \Pi_{rgd} + \Pi_d + \Pi_s + \Pi_c \tag{5.21}$$

in which, Π_d, Π_s and Π_c are penalty terms described in eqs.(3.30), (3.34) and (3.37) for imposing displacement boundary, traction boundary and moment equilibrium equation respectively. The admissible stress field and strain rate field are introduced in the same manner as those in elasticity in eqs.(3.24) and (3.18) by nodal variables β_i^d and β_i^s. The extremum of Π_{II} can be found by solving the following equation

$$\frac{\partial \Pi_{II}}{\partial \beta} = \frac{\partial \Pi_{rgd}}{\partial \beta} + \frac{\partial \Pi_d}{\partial \beta} + \frac{\partial \Pi_s}{\partial \beta} + \frac{\partial \Pi_c}{\partial \beta} = 0 \tag{5.22}$$

where β is the general variable defined in eq.(3.39). The resulting equations are obviously nonlinear.

For various iterative procedures for solving nonlinear problems, linearizing the original nonlinear equation by an incremental form is usually required. This linearized equation can then be solved step by step. Say $\beta^{(i)}$ is a known solution at time i and $\beta^{(i+1)} = \beta^{(i)} + \Delta\beta^{(i)}$ is the unknown variable at time $i+1$. With Taylor's expansion,

$$\Pi_{II}^{(i+1)} \approx \Pi_{II}^{(i)} + \Delta\beta^{(i)} \frac{\partial \Pi_{II}^{(i)}}{\partial \beta} + \frac{1}{2}(\Delta\beta^{(i)})^2 \frac{\partial^2 \Pi_{II}^{(i)}}{\partial \beta \partial \beta} \tag{5.23}$$

where $\Pi_{II}^{(i)} = \Pi_{II}(\beta^{(i)})$. Note that the first and the second differential of $\Pi_{II}^{(i)}$ at time i are constants and substitute the approximate expression of Π_{II} in eq.(5.23) into eq.(5.22). It follows the linearized equation at the *ith* step

$$\left[\frac{\partial^2\Pi_{rgd}^{(i)}}{\partial\beta\partial\beta}+\frac{\partial^2\Pi_{d}^{(i)}}{\partial\beta\partial\beta}+\frac{\partial^2\Pi_{s}^{(i)}}{\partial\beta\partial\beta}+\frac{\partial^2\Pi_{c}^{(i)}}{\partial\beta\partial\beta}\right]\Delta\beta+\left[\frac{\partial\Pi_{rgd}^{(i)}}{\partial\beta}+\frac{\partial\Pi_{d}^{i}}{\partial\beta}+\frac{\partial\Pi_{s}^{(i)}}{\partial\beta}+\frac{\partial\Pi_{c}^{(i)}}{\partial\beta}\right]\approx 0 \tag{5.24}$$

The second to fourth terms in the first bracket of eq.(5.24) are obviously the constant matrices $\mathbf{k^d}$, $\mathbf{k^s}$ and $\mathbf{k^c}$ defined in eqs.(3.31), (3.35) and (3.38) respectively. The corresponding terms in the second bracket are residual terms $\Delta\mathbf{F^d}$, $\Delta\mathbf{F^s}$ and $\Delta\mathbf{F^c}$

$$\Delta\mathbf{F^d} = \mathbf{F^{d(i+1)}} - \mathbf{F^{d(i)}}\ ;\quad \Delta\mathbf{F^s} = \mathbf{F^{s(i+1)}} - \mathbf{F^{s(i)}}\ ;\quad \Delta\mathbf{F^c} = 0$$

where $\mathbf{F^d}$ and $\mathbf{F^s}$ were defined in eq.(3.31) and eq.(3.35) respectively. If the first term in the first bracket and the second bracket are designated as $\mathbf{k^{rgd}}$ and $\Delta\mathbf{F^{rgd}}$ respectively, they can be obtained by making use of the differentials of Π_{rgd} in eq.(5.15) and eq.(5.20)

$$k_{pq}^{rgd}=\left[L_{ss}^{ij}\frac{\partial s_j}{\partial\beta_q}+L_{\dot{d}s}^{ij}\frac{\partial\dot{d}_j}{\partial\beta_q}\right]\frac{\partial s_i}{\partial\beta_p}+\left[L_{\dot{d}s}^{ij}\frac{\partial s_j}{\partial\beta_q}+L_{\dot{d}\dot{d}}^{ij}\frac{\partial\dot{d}_j}{\partial\beta_q}\right]\frac{\partial\dot{d}_i}{\partial\beta_p}+2\rho\frac{\partial\dot{e}_m}{\partial\beta_p}\frac{\partial\dot{e}_m}{\partial\beta_q} \tag{5.25}$$

$$\Delta F_p^{rgd}=L_s^i\frac{\partial s_i}{\partial\beta_p}+L_{\dot{d}}^i\frac{\partial\dot{d}_i}{\partial\beta_p}+2\rho\dot{e}_m\frac{\partial\dot{e}_m}{\partial\beta_p}$$

From eq.(3.17) and eq.(3.21), the following relations exist.

$$\left\{\frac{\partial s_i}{\partial\beta_p}\right\}=\begin{Bmatrix}0\\ \mathbf{A_{Dp}^T}\end{Bmatrix}\ ;\quad \left\{\frac{\partial\dot{e}_i}{\partial\beta_p}\right\}=\begin{Bmatrix}\mathbf{B_p^T}\\ 0\end{Bmatrix}\ ;\quad \left\{\frac{\partial\dot{e}_m}{\partial\beta_p}\right\}=\begin{Bmatrix}\mathbf{B_{mp}^T}\\ 0\end{Bmatrix}$$

$\mathbf{k^{rgd}}$ and $\Delta\mathbf{F^{rgd}}$ in eq.(5.25) can then be written in matrix forms as

$$k_{pq}^{rgd}=\begin{bmatrix}\mathbf{B_q^T}L_{\dot{d}\dot{d}}^{ij}\mathbf{B_q}+2\rho\mathbf{B_{mp}^T}\mathbf{B_{mq}} & \mathbf{B_p^T}L_{\dot{d}s}^{ij}\mathbf{A_{Dq}}\\ \mathbf{A_{Dp}^T}L_{s\dot{d}}^{ij}\mathbf{B_q} & \mathbf{A_{Dp}^T}L_{ss}^{ij}\mathbf{A_{Dp}}\end{bmatrix}\ ;\ \Delta F_p^{rgd}=\begin{Bmatrix}\mathbf{B_p^T}L_{\dot{d}}^i+2\rho\dot{e}_m\mathbf{B_{mp}^T}\\ \mathbf{A_{Dp}^T}L_S^i\end{Bmatrix} \tag{5.26}$$

where k_{pq}^{rgd} is a 4×4 sub-matrix of $\mathbf{k^{rgd}}$ and ΔF_p^{rgd} is a 4×1 sub-matrix of $\Delta\mathbf{F^{rgd}}$ in the case of plane problems. Therefore the global system equation for rigid plasticity at arbitrary time i can be written in a more familiar form as

$$\mathbf{k_\Pi^{(i)}}\Delta\beta^{(i)}+\Delta\mathbf{F_\Pi^{(i)}}=0 \tag{5.27}$$

where $\mathbf{k_\Pi}$ is the rigid-perfect-plasticity stiffness matrix, while $\Delta\mathbf{F_\Pi}$ is the residual force vector

$$\mathbf{k}_{\Pi} = \mathbf{k}^{rgd} + \mathbf{k}^{d} + \mathbf{k}^{s} + \mathbf{k}^{c}$$
$$\Delta\mathbf{F}_{\Pi} = \Delta\mathbf{F}^{rgd} + \Delta\mathbf{F}^{d} + \Delta\mathbf{F}^{s} \tag{5.28}$$

5.3 MINIMIZATION — LINE - SEARCH METHOD

In the last section, it is shown how a minimization of the functional Π_{II} can be equivalent to solving a nonlinear equation (5.27) where only the differentials of the functional is presented. Then an incremental load or displacement control along with full or modified Newton-Raphson iterations[122] are normally required to solve it. Although these techniques still provide the basis for most nonlinear finite element programs, additional sophistication can be achieved to produce a more effective, more robust solution algorithm by making use of the functional itself. The line-search method is one of these methods.

The line-search method is an important numerical technique for most unconstrained optimisation and can be used with a wide range of iterative solution procedures[10]. Using such a technique, one would obtain a direction from an iteration procedure such as the full Newton-Raphson iteration in eq.(5.27), i.e.

$$\Delta\beta^{(i)} = -(\mathbf{k}_{\Pi}^{(i)})^{-1}\Delta\mathbf{F}^{(i)} \tag{5.29}$$

where $\mathbf{k}_{\Pi}^{(i)}$ is the stiffness matrix at the end of the previous iteration. Then general variables $\beta^{(i+1)}$ at time $i+1$ would be updated according to

$$\beta^{(i+1)} = \beta^{(i)} + \eta\Delta\beta^{(i)} \tag{5.30}$$

where $\beta^{(i)}$ is the fixed general variable at the end of the previous iteration and $\Delta\beta^{(i)}$ the fixed direction obtained from eq.(5.29). For the Newton-Raphson procedures, the scalar η in eq.(5.30) is set to unity. With the introduction of line-searches, the scalar η becomes the iterative "step length" which is the only variable, and is chosen to give the lowest value of Π_{rgd}. The multi-dimensional minimization has therefore been converted to a one-dimensional optimisation problem with regard to scalar η.

In this thesis, **Brent's method**, one of many line-search methods, is used to find the line-minimum of the functional Π_{II} along the direction obtained by solving the

eq.(5.27) by Newton-Raphson method. A brief description of such a method will be given in **Appendix B**. A flow-chart of the solution of rigid-plasticity by the mixed model is presented in fig.5.1. Here only one load step is discussed.

5.4 EXPANSION OF A HOLLOW CYLINDER IN RIGID-PERFECT-PLASTICITY

In this chapter, the expansion of a hollow cylinder under a given displacement loading will be analysed. The load is assumed large enough such that the cylinder is in rigid-perfect-plasticity everywhere, as a rigid region cannot be treated in this method. The same FE model with 24 8-noded elements as shown in fig.3.3 in section 3.6.1 are used in this study.

The analytical solution is obtained from ref.[42] under Tresca's yield criterion. However the results by this mixed model is based on the von Mises. In order to compare these results with each other, a translation of the analytical results under the Tresca's criterion to those under the von Mises's criterion is required.

In plane strain condition, a simple relation between those two criterion exists: if results under the Tresca's criterion are obtained at yield stress σ_y then the corresponding results under the von Mises's criterion can be obtained by changing the yield stress as $2\sigma_y/\sqrt{3}$. More details about this can be found in **Appendix C**.

The radial stress σ_{rr} and the hoop stress $\sigma_{\theta\theta}$ are shown in fig.5.2. When compared with the analytical results, it can be seen that the mixed model gives excellent results for both stress components.

Fig.5.3 presents the convergence of the iteration process. The average error and average Π_{rgd} are calculated as eq.(3.43) and eq.(3.45). It is seen that the convergency of solution is very good. After one iteration, the error falls within 0.1 %. On the other hand the value of Π_{rgd} shows a similar convergent pattern as the average error. It again illustrates a natural error estimator in the mixed model which is the value of the functional Π_{rgd}.

Only one example is discussed here as more examples will be presented in the next chapter when elasto-plasticity is dealt with by a mixed model based on this model. This allows the "rigid" region to be introduced.

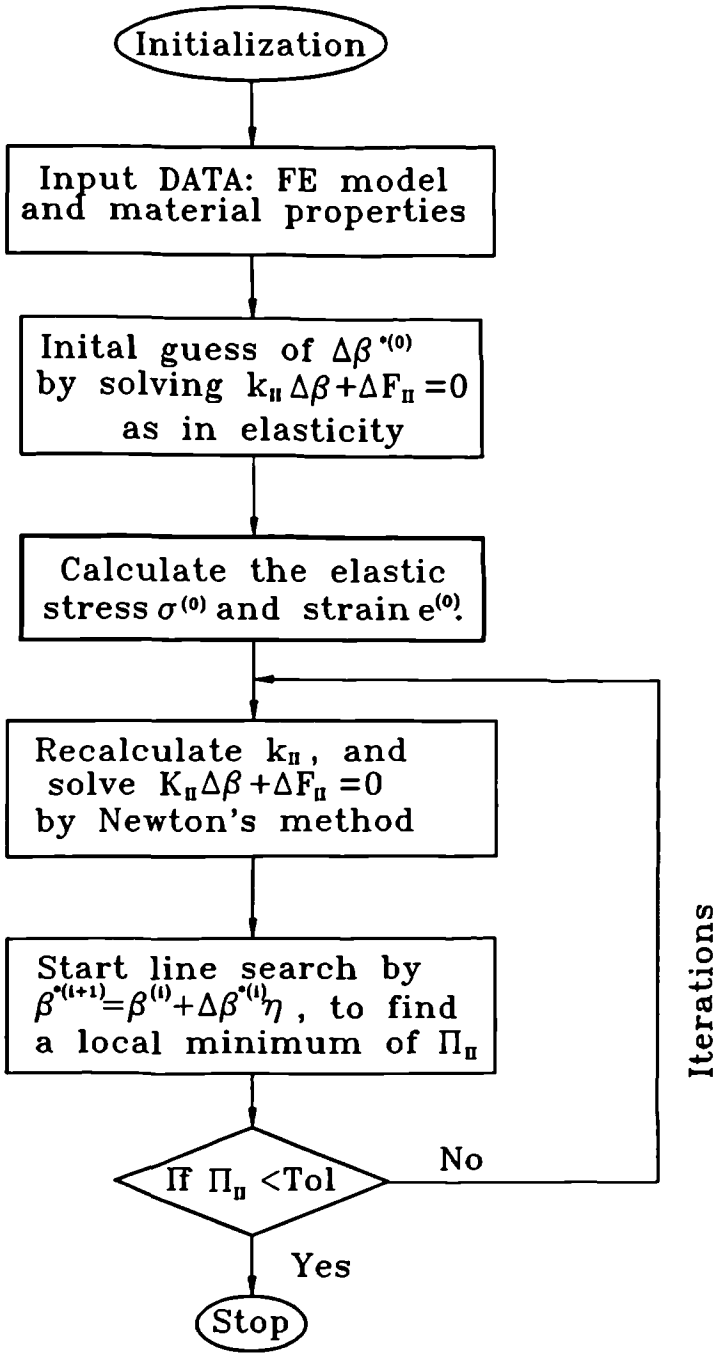


Fig.5.1 Flow-chart of the solution procedure for rigid-plasticity by the mixed model.

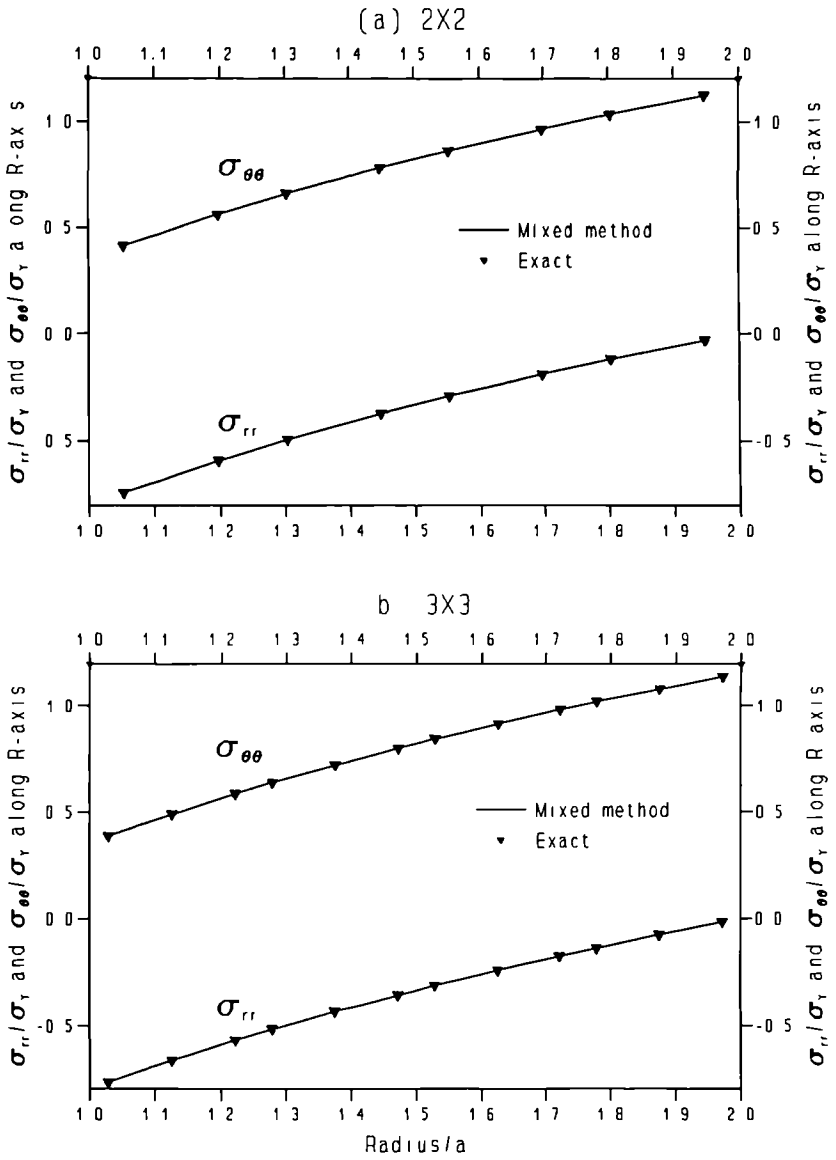


Fig.5.2 Stress distributions in a rigid-perfectly plastic cylinder under displacement loading at 10th iteration. A FE mesh with 24 8-noded elements are used in the computation.

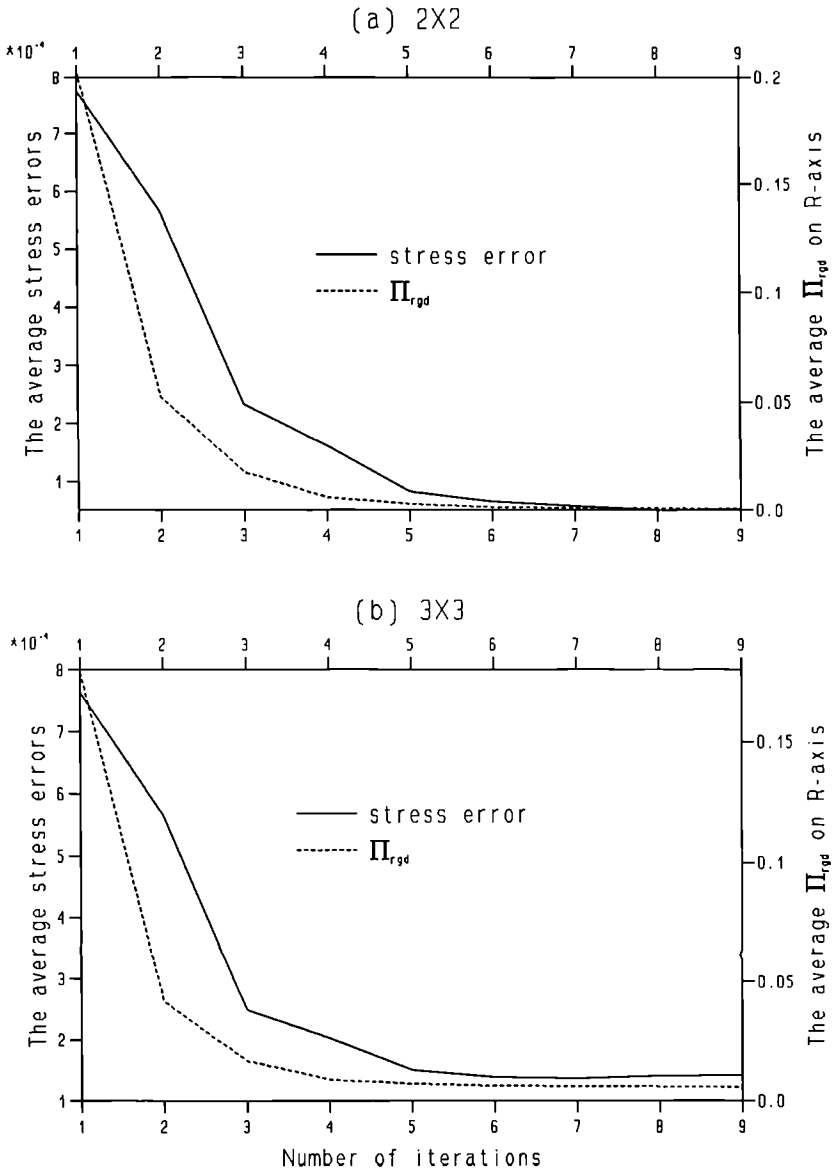


Fig.5.3 Comparisons of convergencies of the solution of a rigid-perfectly-plastic cylinder. Both 2×2 and 3×3 integrations are presented.

CHAPTER 6
MIXED MODEL FOR ELASTO-PLASTICITY

The three previous chapters contain discussions of the extreme cases of elasticity and rigid-perfectly-plasticity. In this chapter, the problem of elasto-plasticity will be addressed. For the last three decades or so, elasto-plastic FE method has successfully been applied to solve many practical problems, by making use of elasto-plastic model of materials. There are plenty of excellent techniques concerning with an elasto-plastic FE method in literature. However, the basic routine of FE method to solve elasto-plasticity problems remains unchanged. Most of the methods normally try to establish elasto-plastic FE model by modifying the existing elastic FE model.

In these formulations elasto-plasticity is simulated by perturbing the elastic solution by plastic strains. Instead, in this chapter, a rigid-plastic solution will be perturbed by elastic strains. This is particular straightforward in a mixed method as the elastic strains can immediately be calculated from the current stresses. In the plastic region, the yield criterion is approximated as well as the flow rule for the plastic strains, by minimizing the functional given in the last chapter. In the region of no plastic flow the plastic strain is made zero by minimizing the sum of the invariants $\dot{d}^p_{ij}\dot{d}^p_{ij}$ and $(\dot{e}^p_m)^2$

The details of this model will be given first followed by examples which can be accompanied with analytical solutions. It must be noted that a finite deformation solution would require the Jouman stress rate. For the sake of simplicity this sophistication has been ignored. So the present formulation is only applicable to small deformation.

6.1 MATERIALS MODEL FOR ELASTO-PLASTICITY

It is well established that metals obey the von-Mises yield criterion,

$$\frac{1}{2}s_{ij}s_{ij} - k^2(\kappa) = 0 \tag{6.3}$$

where k is a material property reflecting yield stress σ_y and κ is hardening parameter which depends on the total strain. s_{ij} is the deviatoric stress defined in eq.(3.10).

Materials obeying such a yield condition are called von-Mises materials in this thesis, and if the associated flow rule is used to establish the constitutive relation of such materials, then

$$\dot{d}_{ij}^{ep} = \dot{d}_{ij}^{e} + \dot{d}_{ij}^{p} = \frac{\dot{s}_{ij}}{2G} + \frac{\dot{D}s_{ij}}{2k^2} \quad \text{where} \quad \dot{D} = \sqrt{2}k\sqrt{\dot{d}_{ij}^{p}\dot{d}_{ij}^{p}}$$
$$\dot{e}_{m}^{ep} = \dot{e}_{m}^{e} = \frac{\dot{\sigma}_m}{k} \tag{6.4}$$

where $\dot{d}_{ij}^{e}$ is the elastic part of the full deviatoric strain rate and $\dot{d}_{ij}^{p}$ is the plastic part of the full deviatoric strain rate. The plastic part of the full elasto-plastic deviatoric strain rate $\dot{d}_{ij}^{p}$ obeys the same equation as the Levy-Mises equation (5.6) governing the rigid-plasticity. The elastic part of the $\dot{e}_{ij}^{ep}$ obviously obeys the Hooke's law eq.(3.9) in elasticity but with the stresses replaced by stress rates.

The assumption of linear isotropic hardening leads to the parameter k in eq.(6.1)

$$k = k_0 + k_t\sqrt{d_{ij}^{p}d_{ij}^{p}} \tag{6.3}$$

where k_0 and k_t are constants. It remains to decide how to impose this constitutive relation in the model. In the conventional method, eq.(6.2) is used to express $\dot{e}_{ij}^{ep}$ as a piece-wise linear relation, such as by introducing an elasto-plastic modulus matrix

$$\{\dot{e}_{ij}^{ep}\} = [\mathbf{D}^{\mathbf{ep}}]^{-1}\{\dot{\sigma}_{ij}\} \tag{6.4}$$

and is then solved incrementally as in elasticity, with modified stiffness and load terms to account for the elasto-plasticity. Because there is no explicit involvement of the yield condition in the solution process, the stress results have to be corrected to satisfy the yield condition eq.(6.1).

In this chapter, since the plastic part of the full elasto-plastic deviatoric strain rate obeys the same equation as in rigid-plasticity, the mixed extremum principle for rigid plasticity introduced in chapter 5 can be directly applied to solve the elasto-plastic problem by introducing the full elasto-plastic strain rate $\dot{e}_{ij}^{ep}$ into the functional implicitly by,

$$\dot{e}_{ij}^{p} = \dot{e}_{ij}^{ep} - \dot{e}_{ij}^{e} \tag{6.5}$$

which will be discussed in the next section.

6.2 MIXED-EXTREMUM PRINCIPLE FOR ELASTO-PLASTICITY

If we replace $\dot{d}_{ij}$ and $\dot{e}_m$ in Π_{rgd} in eq.(5.7) by $\dot{d}_{ij}^p$ and $\dot{e}_m^p$, then the functional is immediately suitable for expressing the plastic part of elasto-plasticity

$$\Pi_{ep-p} = \int_\Omega \sqrt{s_{ij}s_{ij}}\sqrt{\dot{d}_{ij}^p\dot{d}_{ij}^p} - s_{ij}\dot{d}_{ij}^p + \rho\psi^2 + \rho(\dot{e}_m^p)^2 \, dV \tag{6.6}$$

where $\dot{d}_{ij}^p$ and $\dot{e}_m^p$ are the plastic deviatoric strain rate and the plastic volumetric strain rate respectively, defined in eq.(6.5). By means of the Hooke's law eq.(3.9), the elastic strain rate in eq.(6.5) can be replaced by the corresponding stress rate, and by rewriting in a deviatoric form, eq.(6.5) turns into

$$\dot{d}_{ij}^p = \dot{d}_{ij}^{ep} - \frac{\dot{s}_{ij}}{2G} \quad ; \quad \dot{e}_m^p = \dot{e}_m^{ep} - \frac{\dot{\sigma}_m}{K} \tag{6.7}$$

where $\dot{\sigma}_m$ and $\dot{s}_{ij}$ are mean stress rate and deviatoric stress rate respectively. There is only one problem arising from eq.(6.6). This is when,

$$\dot{e}_{ij}^p \equiv 0 \quad \text{or} \quad \begin{cases} \dot{d}_{ij}^p = \dot{d}_{ij}^{ep} - \dfrac{\dot{s}_{ij}}{2G} \equiv 0 \\ \dot{e}_m^p = \dot{e}_m^{ep} - \dfrac{\dot{\sigma}_m}{K} \equiv 0 \end{cases} \tag{6.8}$$

which corresponds to the elastic area. In such an area the functional $\Pi_{ep\text{-}p}$ is no longer suitable, since it comes from the functional for rigid plasticity which is only applicable in plastic area.

It is quite natural to think that we may turn to the functional Π_{elas} in eq.(3.12) for elasticity to solve this problem. Putting $\dot{d}_{ij}^{ep}$, $\dot{e}_m^{ep}$, $\dot{s}_{ij}$ and $\dot{\sigma}_m$ into two residual functions R_1 and R_2 in eq.(3.11), the functional for the Hooke's law in eq.(3.12) yields

$$\int_\Omega G(\dot{d}_{ij}^{ep} - \frac{1}{2G}\dot{s}_{ij})^2 + \frac{K}{2}(\dot{e}_m^{ep} - \frac{1}{K}\dot{\sigma}_m)^2 \, dV \geq 0$$

However, we have already used equation (6.7) so this reduces to

$$\Pi_{ep-e} = \int_{\Omega} G\dot{d}_{ij}^{p}\dot{d}_{ij}^{p} + \frac{K}{2}(\dot{e}_{m}^{p})^{2}\, dV \tag{6.9}$$

Before we build a complete elasto-plastic model which can switch between an elastic area and an elasto-plastic area automatically, a criterion to indicate the elastic area and the plastic area needs to be proposed. Here we choose ψ in eq.(6.6) as this criterion, which is equivalent to the yield criterion in eq.(6.1). This is because that ψ is an intrinsic term of Π_{ep-p} and thus no extra work is needed to calculate it at any location in Ω. The new mixed-extremum principle is then presented as

$$\Pi_{ep} = \begin{cases} \int_{\Omega} \sqrt{s_{ij}s_{ij}}\sqrt{\dot{d}_{ij}^{p}\dot{d}_{ij}^{p}} - s_{ij}\dot{d}_{ij}^{p} + \rho\psi^{2} + \rho(\dot{e}_{m}^{p})^{2}\, dV \;, & \text{when} \quad \psi \geq 0 \\ \int_{\Omega} G\dot{d}_{ij}^{p}\dot{d}_{ij}^{p} + \frac{K}{2}(\dot{e}_{m}^{p})^{2}\, dV \;, & \text{when} \quad \psi < 0 \end{cases} \tag{6.10}$$

where

$$\psi = \frac{\sqrt{s_{ij}s_{ij}}}{\sqrt{2}K} - 1 \quad ; \quad \dot{d}_{ij}^{p} = \dot{d}_{ij}^{ep} - \frac{1}{2G}\dot{s}_{ij} \quad \text{and} \quad \dot{e}_{m}^{p} = \dot{e}_{m}^{ep} - \frac{1}{K}\dot{\sigma}_{m}$$

There are four variables within Π_{ep}: the stress rate $\dot{\sigma}_{ij}$, the stress σ_{ij}, the elasto-plastic strain rate $\dot{e}_{ij}^{ep}$ and the elasto-plastic strain e_{ij}. By integrating $\dot{\sigma}_{ij}$ and $\dot{e}_{ij}$ along the loading path, σ_{ij} and e_{ij} can be uniquely determined. Furthermore, elasto-plastic strain rate $\dot{e}_{ij}^{ep}$ can be expressed as a linear function of velocity $\dot{u}_i$, as shown in eq.(5.4). In conclusion, Π_{ep} is only a functional of stress rate $\dot{\sigma}_{ij}$ and velocity $\dot{u}_i$. Therefore extremum principle falls into a two-field mixed extremum principle, which can be stated as follows:

Among all the admissible strains and stresses which satisfy the compatibility condition and equilibrium conditions, as well as the kinematic and static boundary conditions, the true solution renders the functional Π_{ep} *an absolute minimum.*

By making use of the displacement-strain rate relation eq.(5.4) and the constrained first-order stress rate functions $\dot{F}_x$ and $\dot{F}_y$,

$$\dot{\sigma}_{xx} = \frac{\partial \dot{F}_y}{\partial y} \quad ; \quad \dot{\sigma}_{xy} = -\frac{\partial \dot{F}_x}{\partial y}$$
$$\dot{\sigma}_{yx} = -\frac{\partial \dot{F}_y}{\partial x} \quad ; \quad \dot{\sigma}_{yy} = \frac{\partial \dot{F}_x}{\partial x} \tag{6.11}$$

along with a constraint $\dot{\sigma}_{xy} = \dot{\sigma}_{yx}$, which is a rate form of eq.(3.3), the kinematic conditions and static conditions are implied, as described in previous chapters. Two kinds of boundary conditions, displacement and traction boundary conditions, are again imposed in the same way as we did before by introducing the penalty functions Π_d in eq.(3.30) and Π_s in eq.(3.34). Finally, the general functional for elasto-plasticity is

$$\Pi_{III} = \Pi_{ep} + \Pi_d + \Pi_s + \Pi_c \tag{6.12}$$

where Π_c is the penalty term for imposing the constraint of stress rate functions. The rate form of eq.(3.37) should be used at this case.

6.3 GLOBAL SYSTEM EQUATIONS

In chapter 3 and chapter 4 for linear elastic problems, system variables were only total stress and displacement. While in chapter 5 for rigid-plasticity, we introduced a rate type variables, i.e. velocity $\dot{u}_i$ in place of displacement u_i. No stress rate is employed since no unloading is allowed, thus the solution is independent of the loading history.

However in elasto-plasticity, all variables should be in a rate form since the solution strongly depends on the load history. Therefore the system variables in this chapter should be taken as velocity, strain rate and stress rate.

In the meantime, the total displacement, strain and stress also remain in the system equations. For instance, total plastic strain will be used to calculate the hardening of yield, while total stress is necessary to evaluate the effective stress. But they are not essential variables and can be obtained by integrating rate variables along load path.

For the convenience of reference, in the next section we will reintroduce the admissible velocity, strain rate and stress rate in terms of nodal rate variable $\dot{\beta}_i^d$ and $\dot{\beta}_i^s$.

6.3.1 Stress Function Rate and Velocity in FE Implementation

In section 6.2, tensor notation is used to establish the mixed-extremum principle. For the convenience of FE formulation, we will employ matrix notation in the following sections. Several basic vectors are defined as follows:

plastic strain rate $\dot{\mathbf{e}}^{\mathbf{p}}$ with component $\dot{e}_i^p$,

$$\dot{\mathbf{e}}^{\mathbf{p}} = \left\{\dot{e}_{xx}^p \; \dot{e}_{xy}^p \; \dot{e}_{yx}^p \; \dot{e}_{yy}^p \; \dot{e}_{zz}^p\right\}^T$$

total deviatoric stress $\mathbf{s}$ with component s_i

$$\mathbf{s} = \left\{s_{xx} \; s_{xy} \; s_{yx} \; s_{yy} \; s_{zz}\right\}^T$$

and deviatoric stress rate $\dot{\mathbf{s}}$ with component $\dot{s}_i$,

$$\dot{\mathbf{s}} = \left\{\dot{s}_{xx} \; \dot{s}_{xy} \; \dot{s}_{yx} \; \dot{s}_{yy} \; \dot{s}_{zz}\right\}^T$$

total elasto-plastic deviatoric strain $\mathbf{d}^{\mathbf{ep}}$ with component d_i^{ep}

$$\mathbf{d}^{\mathbf{ep}} = \left\{d_{xx}^{ep} \; d_{xy}^{ep} \; d_{yx}^{ep} \; d_{yy}^{ep} \; d_{zz}^{ep}\right\}^T$$

and elasto-plastic deviatoric strain rate $\dot{\mathbf{d}}^{\mathbf{ep}}$ with component $\dot{d}_i^{ep}$.

$$\dot{\mathbf{d}}^{\mathbf{ep}} = \left\{\dot{d}_{xx}^{ep} \; \dot{d}_{xy}^{ep} \; \dot{d}_{yx}^{ep} \; \dot{d}_{yy}^{ep} \; \dot{d}_{zz}^{ep}\right\}^T$$

The discretization of the velocity for plane problem is similar to the one of displacement in eq.(3.17)

$$\begin{Bmatrix}\dot{u}(\xi,\eta)\\ \dot{v}(\xi,\eta)\end{Bmatrix} = \sum_{i=1}^{m} N_i(\xi,\eta)\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{Bmatrix}\dot{u}_i\\ \dot{v}_i\end{Bmatrix} \tag{6.13}$$

where $\dot{u}(\xi,\eta)$ and $\dot{v}(\xi,\eta)$ are approximate velocities in x and y directions respectively. $\dot{u}_i$ and $\dot{v}_i$ are the corresponding nodal variables of an element. N_i is the shape function of node i in the element, m is the number of nodes per element, as defined in section 3.2. Thus the elasto-plastic deviatoric and the volumetric strain rate for plane problems are expressed by

$$\dot{\mathbf{d}}^{\mathbf{ep}} = \sum_{i=1}^{m} \mathbf{B}_{\mathbf{Di}}\, \dot{\beta}_i^d \quad ; \quad \dot{e}_m = \sum_{i=1}^{m} \mathbf{B}_{\mathbf{mi}}\, \dot{\beta}_i^d \tag{6.14}$$

in which matrices $\mathbf{B}_{\mathbf{Di}}$ and $\mathbf{B}_{\mathbf{mi}}$ are defined as same as in eq.(3.19) for plane elasticity. In case of axisymmetric problem, they should be those defined eq.(4.15). $\dot{\beta}_i^d$ is nodal velocity variables. The total elasto-plastic strain can only be calculated by integrating the strain rate.

Because only C^0 continuity of stress function rate is required, the approximate stress function rates $\dot{F}_x$ and $\dot{F}_y$ are also presented in terms of isoparametric formulation as

$$\begin{Bmatrix} \dot{F}_x(\xi,\eta) \\ \dot{F}_y(\xi,\eta) \end{Bmatrix} = \sum_{i=1}^{m} N_i(\xi,\eta) \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} \dot{F}_{xi} \\ \dot{F}_{yi} \end{Bmatrix} \tag{6.15}$$

where $\dot{F}_x(\xi,\eta)$ and $\dot{F}_y(\xi,\eta)$ are two approximate stress rate functions. $\dot{F}_{xi}$ and $\dot{F}_{yi}$ are nodal variables of the element. The components of the deviatoric stress, deviatoric stress rate and mean stress rate can then be expressed for plane problems as

$$\mathbf{s} = \sum_{i=1}^{m} \mathbf{A}_{\mathbf{Di}}\, \beta_i^s \quad ; \quad \dot{\mathbf{s}} = \sum_{i=1}^{m} \mathbf{A}_{\mathbf{Di}}\, \dot{\beta}_i^s \quad ; \quad \dot{\sigma}_m = \sum_{i=1}^{m} \mathbf{A}_{\mathbf{mi}}\, \dot{\beta}_i^s \tag{6.16}$$

in which constant matrices $\mathbf{A}_{\mathbf{Di}}$ and $\mathbf{A}_{\mathbf{mi}}$ are defined in eq.(3.25) for plane elasticity. In case of axisymmetric problem, they should be those defined in eq.(4.17). $\dot{\beta}_i^s$ is nodal stress function rate variables. The total stress can be calculated by integrating the stress rate. Finally, the plastic deviatoric stain rate $\dot{\mathbf{d}}^{\mathbf{p}}$ and plastic volumetric strain rate $\dot{e}_m^p$ defined as in eq.(6.10) can be expressed in a matrix form

$$\dot{\mathbf{d}}^{\mathbf{p}} = \sum_{i=1}^{m} \left[\mathbf{B}_{\mathbf{Di}} \; -\frac{1}{2G}\mathbf{A}_{\mathbf{Di}} \right] \{\dot{\beta}_i\} \quad ; \quad \dot{e}_m^p = \sum_{i=1}^{m} \left[\mathbf{B}_{\mathbf{mi}} \; -\frac{1}{K}\mathbf{A}_{\mathbf{mi}} \right] \{\dot{\beta}_i\} \tag{6.17}$$

by introducing the general nodal variable, $\dot{\beta}_i = \{\dot{\beta}_i^d \;\; \dot{\beta}_i^s\}^T$, and making use of eq.(6.14) and eq.(6.16).

6.3.2 Calculation of Stiffness Matrix and Residual Force Rate

The nature of this kind of problem decides that the system equation will be nonlinear. It is believed that the Hessian matrix, which is defined by the second-order differential of Π_{III}, is essential to solve this kind of equations. In section 5.3.1, we have demonstrated that the analytical differentiation of the functional Π_{rgd} is superior to the numerical one. We can anticipate that for the differentiation of Π_{ep} the conclusion is still the same since there is little difference between these two functionals as far as computing time is concerned.

Again the function $f(x) = \sqrt{x_i x_i}$ has been introduced in order to simplify the differential operation and the functional $\Pi_{ep\text{-}p}$ turns out to be

$$\Pi_{ep-p} = f(\dot{\mathbf{d}}^{\mathbf{p}})\, f(\mathbf{s}) - \mathbf{s}^{\mathbf{T}}\dot{\mathbf{d}}^{\mathbf{p}} + \rho\psi^2 + \rho\dot{e}_m^{p2} \tag{6.18}$$

where ψ and k are defined as eq.(5.17) and eq.(6.3) respectively. Only minor changes from Π_{rgd} in eq.(5.17) is found in the new functional $\Pi_{ep\text{-}p}$ in eq.(6.18):$\dot{\mathbf{d}}^{\mathbf{p}}$ and $\dot{e}_m^p$ in eq.(6.18) are in place of the corresponding $\dot{\mathbf{d}}$ and $\dot{e}_m$ in eq.(5.17). Consequently, all the differentials of $\Pi_{ep\text{-}p}$ remain the same as those of Π_{rgd} in eq.(5.18),except that$\dot{\mathbf{d}}$ and $\dot{e}_m$ should be replaced by$\dot{\mathbf{d}}^{\mathbf{p}}$ and $\dot{e}_m^p$.

On the other hand, differentials of $\Pi_{ep\text{-}e}$ with respect to $\dot{d}_{ij}^p$ are very easy to obtain by the observation of the second equation in eq.(6.10) as follows

$$\begin{aligned} \frac{\partial \Pi_{ep-e}}{\partial \dot{d}_i^p} &= 2G\dot{d}_i^p \quad ; \quad \frac{\partial^2 \Pi_{ep-e}}{\partial \dot{d}_i^p \partial \dot{d}_i^p} = 2G \\ \frac{\partial \Pi_{ep-e}}{\partial \dot{e}_m^p} &= K\dot{e}_m^p \quad ; \quad \frac{\partial^2 \Pi_{ep-e}}{\partial \dot{e}_m^p \partial \dot{e}_m^p} = K \end{aligned} \tag{6.19}$$

It remains to determine the differentials of the yield function ψ with respect to both $\dot{\mathbf{s}}$ and $\dot{\mathbf{d}}^{\mathbf{p}}$. In chapter 5, the situation was simplified by the fact that the constant k does not vary with plastic deviatoric strain $\mathbf{d}^{\mathbf{p}}$ or deviatoric strain rate $\dot{\mathbf{d}}^{\mathbf{p}}$. In this chapter, hardening effect has been taken into account, which means that k is function of $\dot{\mathbf{d}}^{\mathbf{p}}$. Therefore corresponding to eq.(5.20), differentials of ψ with respect to $\dot{\mathbf{d}}^{\mathbf{p}}$ should be

given as

$$\frac{\partial \psi}{\partial \dot{d}_i^p} = -\frac{k_t \mathrm{f}(\dot{\mathbf{s}})}{\sqrt{2}k^2}\frac{\dot{d}_i^p}{\mathrm{f}(\dot{\mathbf{d}}^{\mathbf{p}})} \quad ; \quad \frac{\partial^2 \psi}{\partial \dot{d}_i^p \partial \dot{s}_j} = -\frac{k_t \dot{d}_i^p \dot{s}_j}{\sqrt{2}k^2 \mathrm{f}(\dot{\mathbf{d}}^{\mathbf{p}})\mathrm{f}(\dot{\mathbf{s}})}$$
$$\frac{\partial^2 \psi}{\partial \dot{d}_i^p \partial \dot{d}_j^p} = \frac{2k_t^2 \dot{d}_i^p \dot{d}_j^p \mathrm{f}(\dot{\mathbf{s}})}{\sqrt{2}k^3 \mathrm{f}^2(\dot{\mathbf{d}}^{\mathbf{p}})} - \frac{k_t \mathrm{f}(\dot{\mathbf{s}})(\delta_{ij}\mathrm{f}^2(\dot{\mathbf{d}}^{\mathbf{p}}) - \dot{d}_i^p \dot{d}_j^p)}{\sqrt{2}k^2 \mathrm{f}^3(\dot{\mathbf{d}}^{\mathbf{p}})} \tag{6.20}$$

The differential of f with respect to **x** is identical to the one in last chapter, as seen in eq.(5.19).

So far all the necessary differentials have been calculated. Then we will start deducing the stiffness matrix $\mathbf{k}^{\mathbf{ep}}$. It is worthwhile to mention that all the processes in the deduction of $\mathbf{k}^{\mathbf{ep}}$ should be carried out in two different cases: elastic case and elasto-plastic case. Say

$$L_{\dot{d}\dot{d}}^{ij} = \frac{\partial^2 \Pi_{ep-p}}{\partial \dot{d}_i^p \partial \dot{d}_j^p} \quad ; \quad L_{\dot{d}\dot{s}}^{ij} = \frac{\partial^2 \Pi_{ep-p}}{\partial \dot{d}_i^p \partial \dot{s}_j} \quad ; \quad L_{\dot{s}\dot{s}}^{ij} = \frac{\partial^2 \Pi_{ep-p}}{\partial \dot{s}_i \partial \dot{s}_j} \tag{6.21}$$

we have

$$\begin{cases} k_{pq}^{ep-p} = \left[L_{ss}^{ij}\dfrac{\partial \dot{s}_j}{\partial \dot{\beta}_q} + L_{\dot{d}s}^{ij}\dfrac{\partial \dot{d}_j^p}{\partial \dot{\beta}_q} \right]\dfrac{\partial \dot{s}_i}{\partial \dot{\beta}_p} + \left[L_{\dot{d}s}^{ij}\dfrac{\partial \dot{s}_j}{\partial \dot{\beta}_q} + L_{\dot{d}\dot{d}}^{ij}\dfrac{\partial \dot{d}_j^p}{\partial \dot{\beta}_q} \right]\dfrac{\partial \dot{d}_i^p}{\partial \dot{\beta}_p} + 2\rho \dfrac{\partial \dot{e}_m^p}{\partial \dot{\beta}_p}\dfrac{\partial \dot{e}_m^p}{\partial \dot{\beta}_q} \\ k_{pq}^{ep-e} = 2G\dfrac{\partial \dot{d}_i^p}{\partial \dot{\beta}_p}\dfrac{\partial \dot{d}_i^p}{\partial \dot{\beta}_q} + K\dfrac{\partial \dot{e}_m^p}{\partial \dot{\beta}_p}\dfrac{\partial \dot{e}_m^p}{\partial \dot{\beta}_q} \end{cases} \tag{6.22}$$

Following eqs.(6.16) and (6.17), the differentials of $\dot{\mathbf{s}}$, $\dot{\mathbf{d}}^{\mathbf{p}}$ and $\dot{e}_m^p$ with respect to the general nodal variable $\dot{\beta}$ are

$$\left\{\frac{\partial \dot{s}_i}{\partial \dot{\beta}_p}\right\} = \begin{Bmatrix} \mathbf{0} \\ \mathbf{A}_{\mathbf{Dp}}^{\mathbf{T}} \end{Bmatrix} \quad ; \quad \left\{\frac{\partial \dot{d}_i^p}{\partial \dot{\beta}_p}\right\} = \begin{Bmatrix} \mathbf{B}_{\mathbf{Dp}}^{\mathbf{T}} \\ -\dfrac{1}{2G}\mathbf{A}_{\mathbf{Dp}}^{\mathbf{T}} \end{Bmatrix} \quad ; \quad \left\{\frac{\partial \dot{e}_m^p}{\partial \dot{\beta}_p}\right\} = \begin{Bmatrix} \mathbf{B}_{\mathbf{mp}}^{\mathbf{T}} \\ -\dfrac{1}{K}\mathbf{A}_{\mathbf{mp}}^{\mathbf{T}} \end{Bmatrix} \tag{6.23}$$

Substituting eq.(6.23) into eq.(6.22), the stiffness matrix $\mathbf{k}^{\mathbf{ep}}$ is immediately obtained

$$\mathbf{k}_{pq}^{ep} = \int_{\Omega^e} \begin{bmatrix} 2G\mathbf{B}_{Dp}^{T}\mathbf{B}_{Dq} + K\mathbf{B}_{mp}^{T}\mathbf{B}_{mq} & -(\mathbf{B}_{Dp}^{T}\mathbf{A}_{Dq} + \mathbf{B}_{mp}^{T}\mathbf{A}_{Dq}) \\ -(\mathbf{A}_{Dp}^{T}\mathbf{B}_{Dq} + \mathbf{A}_{mp}^{T}\mathbf{B}_{mq}) & \mathbf{A}_{Dp}^{T}\mathbf{A}_{Dq}/2G + \mathbf{A}_{mp}^{T}\mathbf{A}_{mq}/K \end{bmatrix} d\Omega \qquad (6.24)$$

when $\psi < 0$, and

$$\mathbf{k}_{pq}^{ep} = \int_{\Omega^e} \begin{bmatrix} \mathbf{B}_{Dp}^{T} L_{\dot{e}\dot{e}}^{ij} \mathbf{B}_{Dq} + 2\rho \mathbf{B}_{mp}^{T}\mathbf{B}_{mq} & \mathbf{B}_{Dp}^{T}\Lambda^{ij}\mathbf{A}_{Dq} - \dfrac{2\rho}{K}\mathbf{B}_{mp}^{T}\mathbf{A}_{mq} \\ \mathbf{A}_{Dp}^{T}\Lambda^{ij}\mathbf{B}_{Dq} - \dfrac{2\rho}{K}\mathbf{A}_{mp}^{T}\mathbf{B}_{mq} & \mathbf{A}_{Dp}^{T}\mathrm{X}^{ij}\mathbf{A}_{Dp} + \dfrac{2\rho}{K^2}\mathbf{A}_{mp}^{p}\mathbf{A}_{mq} \end{bmatrix} d\Omega \qquad (6.25)$$

when $\psi \geq 0$, in which the intermediate variables Λ and X are

$$\Lambda^{ij} = L_{\dot{s}\dot{s}}^{ij} - \frac{1}{2G} L_{\dot{d}\dot{d}}^{ij} \quad \text{and} \quad \mathrm{X}^{ij} = L_{\dot{s}\dot{s}}^{ij} - \frac{1}{2G} L_{\dot{d}\dot{s}}^{ij} - L_{\dot{d}\dot{s}}^{ij} + \frac{1}{2G} L_{\dot{d}\dot{d}}^{ij}$$

Note that if the following equality

$$\mathbf{B}_{p}^{T}\mathbf{A}_{q} \equiv \mathbf{B}_{Dp}^{T}\mathbf{A}_{Dq} + \mathbf{B}_{mp}^{T}\mathbf{A}_{mq}$$

is substituted into eq.(6.24), it is easy to find that $\mathbf{k}^{ep}$ in eq.(6.24) is identical to $\mathbf{k}^{e}$ in eq.(3.39) for elasticity.

Let

$$L_{\dot{s}}^{i} = \frac{\partial \Pi_{ep-p}}{\partial \dot{s}_i} \quad ; \quad L_{\dot{d}}^{i} = \frac{\partial \Pi_{ep-p}}{\partial \dot{d}_i^{p}}$$

following the similar routine of the one in chapter 5, the "residual force rate" vector, which is defined as

$$\Delta\dot{\mathbf{F}}^{ep} = \frac{\partial \Pi_{ep}}{\partial \dot{\beta}}$$

corresponding to the stiffness $\mathbf{k}^{ep}$ in Newton's method, can also be expressed in terms of differentials of $\dot{\mathbf{e}}^{p}$ and $\dot{e}_m$ with respect to the general nodal variable $\dot{\beta}$ as

$$\begin{cases} \Delta\dot{F}_p^{ep} = \int_\Omega L_s^i \frac{\partial \dot{s}_i}{\partial \dot{\beta}_p} + L_d^i \frac{\partial \dot{d}_i^p}{\partial \dot{\beta}_p} + 2\rho \dot{e}_m \frac{\partial \dot{e}_m}{\partial \dot{\beta}_p} dV & ; \quad \text{when} \psi \geq 0 \\ \Delta\dot{F}_p^{ep} = \int_\Omega 2G\dot{d}_i^p \frac{\partial \dot{d}_i^p}{\partial \dot{\beta}_p} + K\dot{e}_m^i \frac{\partial \dot{e}_m^i}{\partial \dot{\beta}_p} dV & ; \quad \text{when} \psi < 0 \end{cases} \tag{6.26}$$

then the matrix form of $\Delta\dot{\mathbf{F}}^{\mathbf{ep}}$ is readily obtained by making use of eq.(6.23)

$$\Delta\dot{F}_p^{ep} = \begin{Bmatrix} 2G\dot{d}_i^p B_{Dp}^T + K\dot{e}_m^p \mathbf{B}_{\mathbf{mp}}^{\mathbf{T}} \\ -\dot{d}_i^p \mathbf{A}_{\mathbf{Dp}}^{\mathbf{T}} - \dot{e}_m^p \mathbf{A}_{\mathbf{mp}}^{\mathbf{T}} \end{Bmatrix} \quad ; \quad \text{when } \psi < 0 \tag{6.27}$$

and

$$\Delta\dot{F}_p^{ep} = \begin{Bmatrix} \mathbf{B}_{\mathbf{Dp}}^{\mathbf{T}} L_d^i + 2\rho \dot{e}_m^p \mathbf{B}_{\mathbf{mp}}^{\mathbf{T}} \\ \mathbf{A}_{\mathbf{Dp}}^{\mathbf{T}} (L_s^i - L_d^i/2G) - 2\rho \dot{e}_m^p \mathbf{A}_{\mathbf{mp}}^{\mathbf{T}}/K \end{Bmatrix} \quad ; \quad \text{when } \psi \geq 0 \tag{6.28}$$

6.3.3 Solving the Linearized System Equation

The solution of a complete elasto-plastic problem can be obtained by finding the extremum of Π_{III}, as stated in section 6.2. No subsidiary condition is needed if admissible strain rate and stress are given by eq.(6.14) and eq.(6.16). By Newton's method, the nonlinear equation of finding the minimum of Π_{III} is linearized at any given time t as

$$\mathbf{k}_{\mathbf{III}} \Delta\dot{\beta} + \Delta\dot{F}_{III} = 0 \tag{6.29}$$

From eq.(6.12), $\mathbf{k}_{\mathbf{III}}$ and $\Delta\dot{F}_{III}$ are obtained by

$$\begin{aligned} \mathbf{k}_{\mathbf{III}} &= \frac{\partial^2 \Pi_{ep}}{\partial\dot{\beta}\partial\dot{\beta}} + \frac{\partial^2 \Pi_d}{\partial\dot{\beta}\partial\dot{\beta}} + \frac{\partial^2 \Pi_s}{\partial\dot{\beta}\partial\dot{\beta}} + \frac{\partial^2 \Pi_c}{\partial\dot{\beta}\partial\dot{\beta}} \\ \Delta\dot{F}_{III} &= \frac{\partial \Pi_{ep}}{\partial\dot{\beta}} + \frac{\partial \Pi_d}{\partial\dot{\beta}} + \frac{\partial \Pi_s}{\partial\dot{\beta}} + \frac{\partial \Pi_c}{\partial\dot{\beta}} \end{aligned} \tag{6.30}$$

The last three terms in both equations are $\mathbf{k}^{\mathbf{d}}$, $\mathbf{k}^{\mathbf{s}}$ and $\mathbf{k}^{\mathbf{c}}$ as well as the rates of $\Delta\mathbf{F}^{\mathbf{d}}$,

$\Delta \mathbf{F}^{s}$ and $\Delta \mathbf{F}^{c}$

$$\Delta \dot{\mathbf{F}}^{d} = \dot{\mathbf{F}}^{d(i+1)} - \dot{\mathbf{F}}^{d(i)} \quad ; \quad \Delta \dot{\mathbf{F}}^{s} = \dot{\mathbf{F}}^{s(i+1)} - \dot{\mathbf{F}}^{s(i)} \quad ; \quad \Delta \dot{\mathbf{F}}^{c} = 0$$

defined in section 5.2.2. The first term in these equations are $\mathbf{k}^{ep}$ defined in eq.(6.24) and eq.(6.25) and $\Delta \dot{\mathbf{F}}^{ep}$ in eq.(6.27) and eq.(6.28). It immediately comes out

$$\begin{aligned} \mathbf{k}_{III} &= \mathbf{k}^{ep} + \mathbf{k}^{d} + \mathbf{k}^{s} + \mathbf{k}^{c} \\ \Delta \dot{\mathbf{F}}_{III} &= \Delta \dot{\mathbf{F}}^{ep} + \Delta \dot{\mathbf{F}}^{d} + \Delta \dot{\mathbf{F}}^{s} \end{aligned} \tag{6.31}$$

The linear search method described in section 5.3 is again used to find the minimum of Π_{III}, the search direction $\Delta \dot{\beta}$ of which is obtained by solving eq.(6.29). The whole solution procedure includes the following steps:

Say at a particular time i the true velocity $\dot{\mathbf{u}}^{(i)}$ and true stress $\sigma^{(i)}$ have been obtained. To calculate the velocity, strain and stress at the next step $i+1$,

a) solve the linear equations of elasticity to get first approximate velocity $\dot{\mathbf{u}}^{(i+1)*}$, strain rate $\dot{\mathbf{e}}^{(i+1)*}$ and stress rate $\dot{\sigma}^{(i+1)*}$. The approximate stress $\sigma^{(i+1)*}$, obtained by integrating the stress rate, satisfies the equilibrium equation but does not obey the yield condition.

b) The approximate stress is then used to give the first evaluation of elastic zone and plastic zone from which stiffness matrix $\mathbf{k}^{ep}$ and force $\Delta \dot{\mathbf{F}}^{ep}$ in eq.(6.24) / eq.(6.25) and eq.(6.27) / eq.(6.28) are able to be obtained.

c) Solve the linearized equation eq.(6.29), from which the line search direction $\Delta \dot{\beta}^{(i)}$ is obtained. By means of line search, $\dot{\beta}^{(i+1)*} = \dot{\beta}^{(i)} + \eta \Delta \dot{\beta}^{(i)}$, the line minimum of the functional Π_{III} is found giving a specific value of η.

d)The rate variable $\dot{\beta}^{(i+1)*}$ is used to calculate the velocity $\dot{\mathbf{u}}^{(i+1)*}$, strain rate $\dot{\mathbf{e}}^{(i+1)*}$ and stress rate $\dot{\sigma}^{(i+1)*}$ by eqs.(6.13), (6.14) and (6.16), from which the total displacement and stress at step $i+1$ are then obtained as follows

$$\mathbf{u}^{(i+1)*} = \mathbf{u}^{(i)} + \dot{\mathbf{u}}^{(i+1)*} \cdot dt \quad ; \quad \sigma^{(i+1)*} = \sigma^{(i)} + \dot{\sigma}^{(i+1)*} \cdot dt \tag{6.32}$$

where for a static plastic flow, the time increment dt is taken as 1.0.

e) If the value of Π_{III} with $\dot{\beta} = \dot{\beta}^{(i+1)*}$ is close to zero at a given tolerance, then the solution of the stress at this step $\sigma^{(i+1)}$ is taken as $\sigma^{(i+1)*}$ and that of the displacement $\mathbf{u}^{(i+1)}$ is taken as $\mathbf{u}^{(i+1)*}$. If it is not, then go back to step b).

The flow chart of the solution procedure is shown in fig.6.1.

6.3.4 Solution Procedure for Various Load Conditions

Elasto-plasticity, is different from either rigid-perfect-plasticity or elasticity in that the solution is history dependent. Special remarks need to be made on the capability of the current model to deal with this. Briefly, a history-dependent constitutive relation implies that when *loading*, *unloading* or *reloading*, the constitutive relations appear in different forms

a) Loading——occur when extra loads are added to the current level. From the incremental theory of plasticity the constitutive relation remains elasto-plastic, as shown by line **OA** in fig.6.2. In the solution procedure described in section 6.3.3, the first estimate under the assumption of elastic relation gives higher stress level than reality. As a result, the minimum of Π_{III} is not found at this step, and other iterations are needed. At this moment, the switch function ψ, being the same as the yield function, is greater than zero. Then eq.(6.25) and eq.(6.28) are used to construct the global system equation eq.(6.29), which implies the constitutive relation of elasto-plasticity.

b) Unloading——is to remove a load increment from the current level. Here only a small increments of loads is concerned since the isotropic hardening is only valid in such a case, as shown by line **AO'** in fig.6.2. In the solution procedure described section 6.3.3, the first guess for the unloading step is as in elasticity. The yield function ψ will not be greater than zero at this moment since the updated stress is well

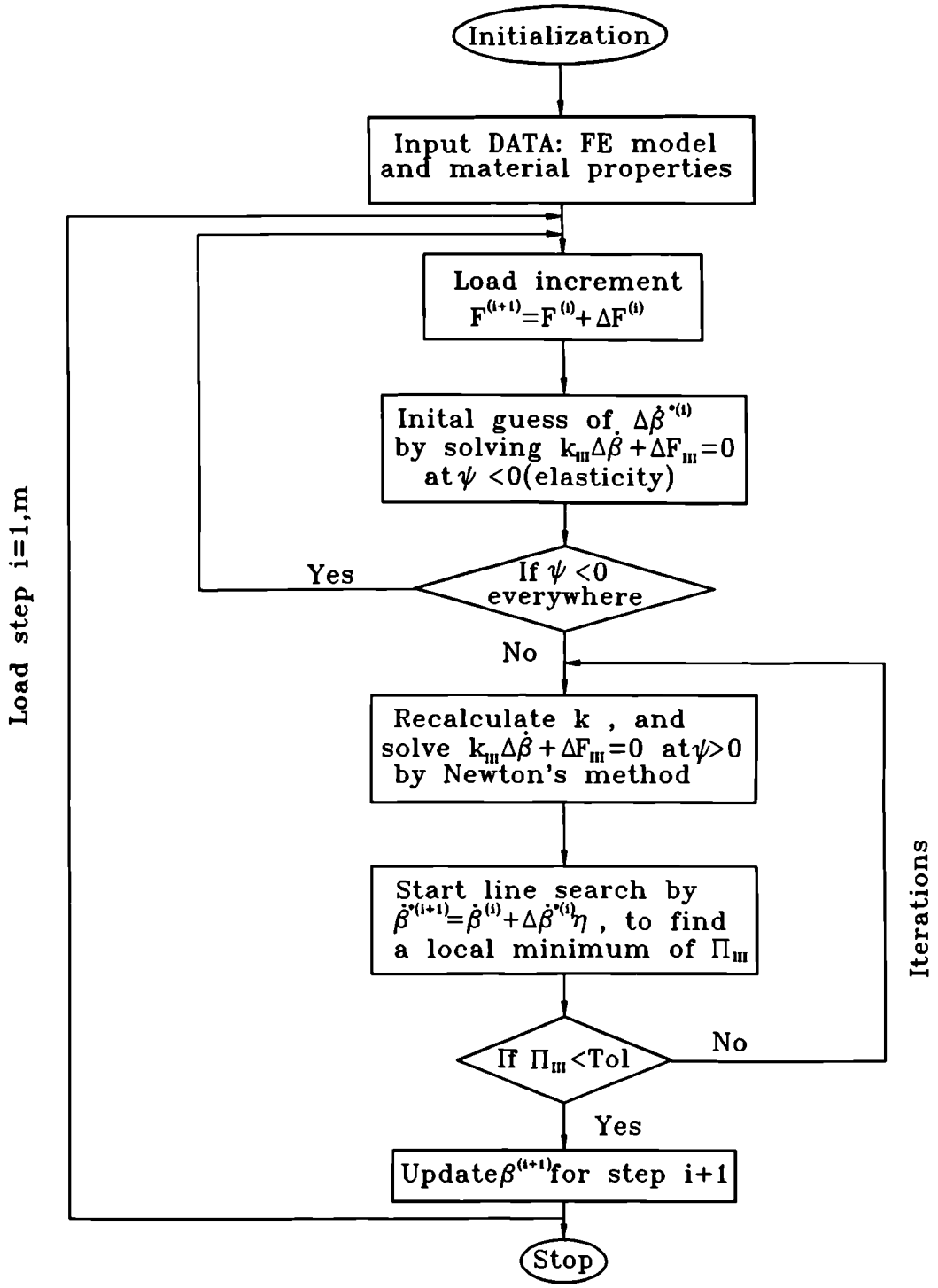


Fig.6.1 Flow-chart of the solution procedure for elasto-plasticity by the mixed model.

below the yield surface. No further correction is needed since eq.(6.24) and eq.(6.27) have already been used to obtain the first guess. This represents an elastic solution.

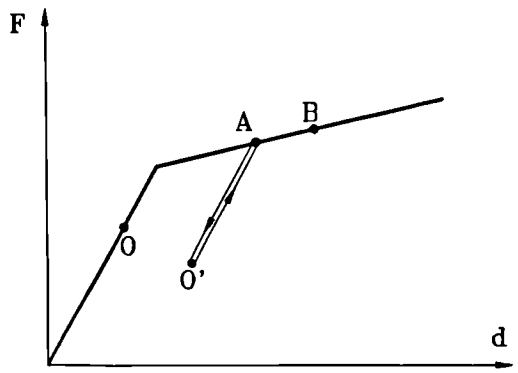


Fig.6.2 Loading conditions in elasto-plasticity

c) Reloading——here is termed as a loading process after an unloading from yield surface, as shown by line **O'A** or **O'B** in fig.6.2. For the case of reloading from point **O'** to point **A**, the first elastic estimate will immediately yield the correct answer since $\psi<0$ thus the minimum of Π_{III} will be reached by such an estimate. The iteration terminates. It means that the procedure switches the constitutive relation to elastic one as the situation arises.

For the case of reloading from point **O'** to point **B** the first guess will certainly not bring the Π_{III} to the minimum since it is beyond the yield surface and the first equation of eq.(6.15) must replace the second equation of it from which the first guess is obtained. This brings the solution procedure to the normal elasto-plastic one.

In conclusion the model proposed in this chapter can handle all the three loading cases described above automatically and correctly. If kinametic hardening is used, it should be able to predict residual stresses.

6.4 SOLUTIONS OF ELASTO-PLASTIC PROBLEMS

There are only a few analytical solutions for some simple plasticity problems which are available in literature. In this section similar problems to the previous chapter in elasticity will be solved in plasticity to demonstrate the efficiency and validity of this mixed model concerned with elasto-plasticity. Thus no new mesh is required.

6.4.1 Expansion of a Thick Spherical Shell

A thick-walled spherical shell, whose internal and external radii are a and $b=2a$ respectively, is subjected to uniform internal pressure p of gradually increasing magnitude, as shown in fig 6.3. If the internal pressure is increased to a critical value p_e, plastic yielding begin at the radius where yield criterion is first satisfied, which is obvious at the inner radius $r = a$.

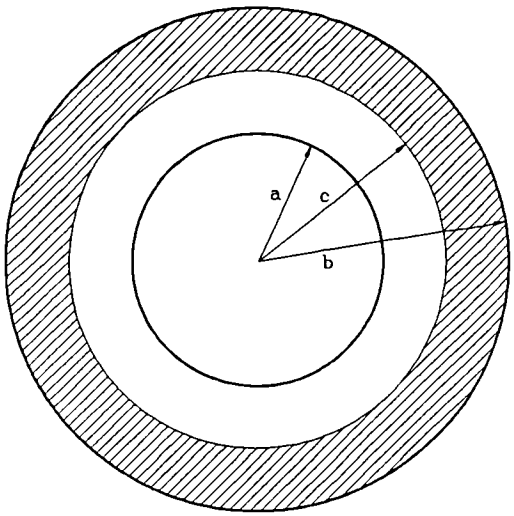


Fig. 6.3 Expansion of a thick spherical shell and its FE model.

With a further increase in the internal pressure, the plastic zone spreads outward and the elasto/plastic boundary, say $r = c$, is a spherical surface at each stage, as in fig.6.3.

In our finite element model, only one mesh with 50 8-noded axisymmetric elements is used in the analysis, as seen in fig.3.1. 3×3 integration scheme is adopted in the calculation of the stiffness matrix $\mathbf{k}^{ep}$ and the stress and strain. In fig.6.4 the hoop stresses $\sigma_{\theta\theta}/\sigma_Y$ and radial stresses σ_{rr}/σ_Y under four different c/b, 0.5, 0.6, 0.7 and 0.8, corresponding different pressure p are plotted along radius/b. For various loads, only one load-step is used in the solution with the full pressure p instead of applying it incrementally.

It is seen that the mixed model gives very good radial stress σ_{rr} for each c/b. However $\sigma_{\theta\theta}$ slightly deviates the analytical solution when $c/b=0.8$, especially near the elasto-plastic boundary. The rest of the cases are in excellent agreement. Nevertheless,

the mixed has superb capability to solve the elasto-plasticity, considering only one load step is used.

The penalty number ρ_c was chosen to be the same as it was in elasticity. However another penalty number ρ is needed for imposing the yield condition, as in eq.(6.15). It is suggested that it should be chosen to be between 50.0 to 500.0 for a normal elasto-plastic problem.

In fig.6.5, the logarithms of the value of Π_{ep} (a) and the average errors (b) are plotted against the iteration numbers. At c/b=0.6, it yields a results after two iterations at an average error less than 0.2 % fig.6.5(b). But at c/b=0.7, three iterations are needed for an average error about 0.7 % while at c/b=0.8, four iterations give an average error around 1.0%.

The curves in fig.6.5(a) for Π_{ep} are similar to fig.6.5(b). Therefore the error of a elasto-plastic solution can also be estimated by the value of Π_{ep} in this model.

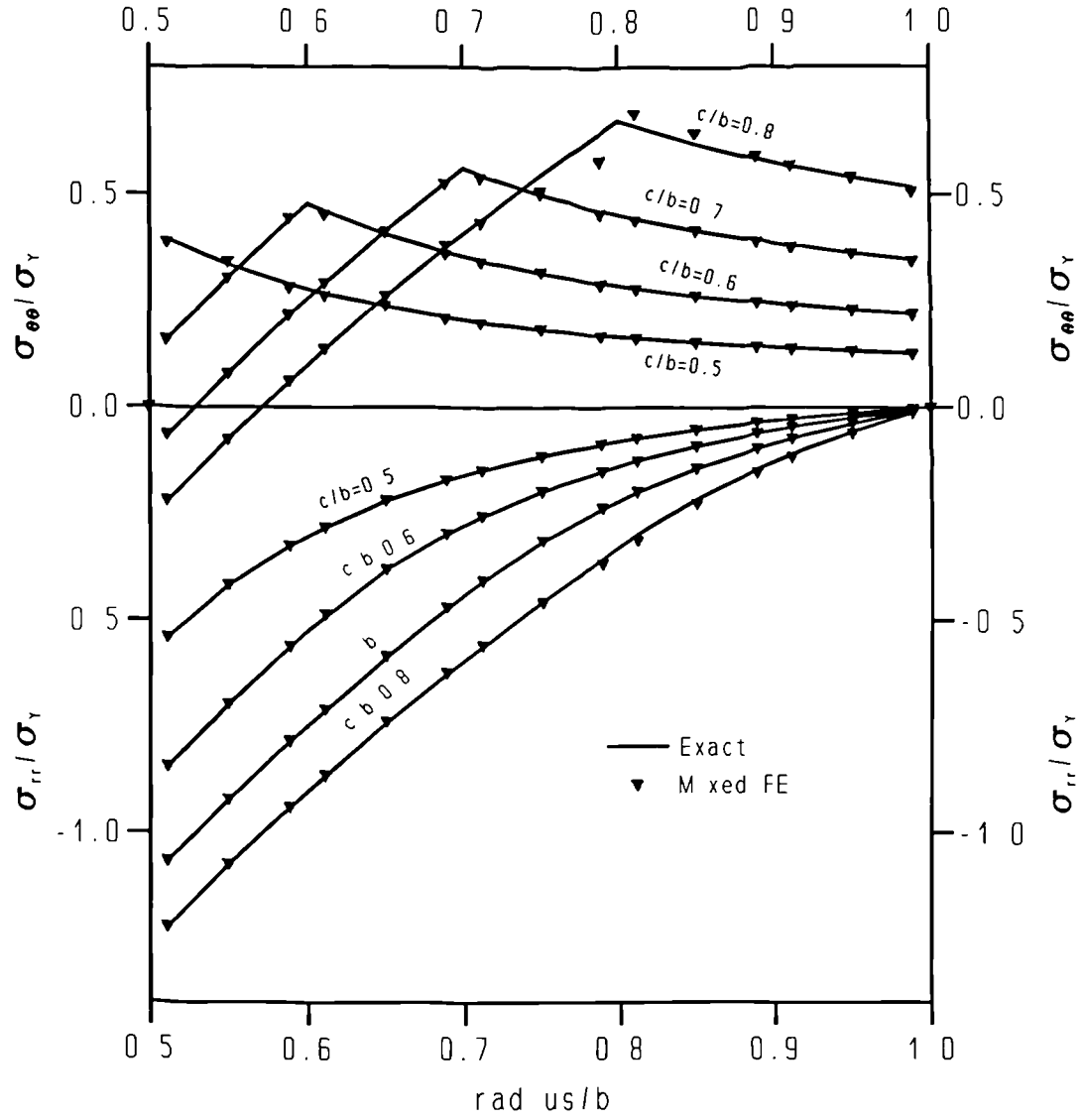


Fig.6.4 Stress distributions in an elasto-perfectly plastic hollow sphere under internal pressure. A FE mesh with 50 8-noded elements are used in the computation.

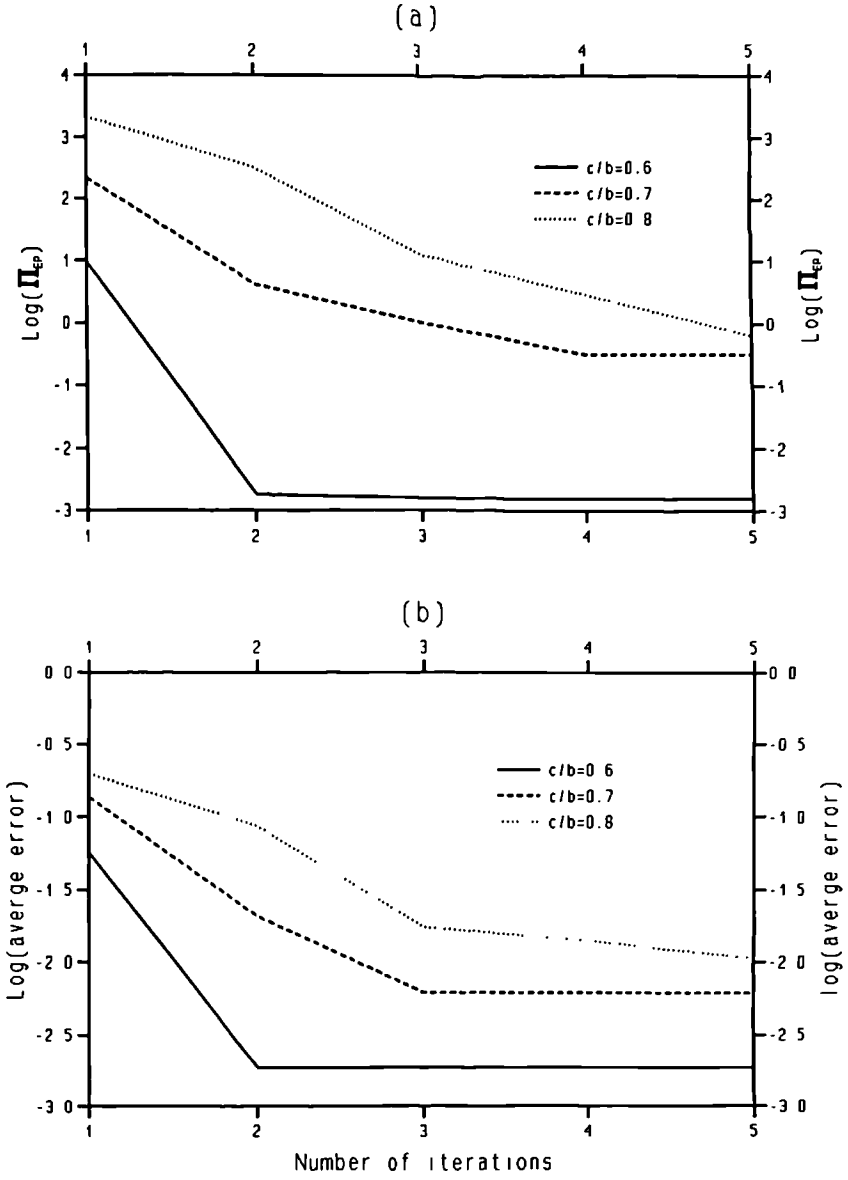


Fig.6.5 Convergence of the elasto-plastic solution of a hollow sphere under internal pressure. A FE mesh of 50 8-noded elements with 3×3 integration are used in the computation.

6.4.2 Expansion of a Thick-walled Cylinder

Although for analytical solution, a 2D cylinder is more complicated than a full 3D sphere, it is usually simpler when it is solved by the finite element method. Except that the element type should be replaced by a plane one, the entire models of the last section for spherical shell are directly applicable to the current problems.

The radial stress σ_{rr}/σ_Y and the hoop stress $\sigma_{\theta\theta}/\sigma_Y$ are presented in fig.6.6 obtained from the mixed model and the analytical solution[42]. Almost identical results are found between two methods under four different plastic range: $c/b=$ 0.5, 0.6, 0.7 and 0.8. Only a point near elasto-plastic boundary at $c/b=$ 0.8 is slightly apart from the exact solution. The reason is that for such a large plastic deformation, one load step is too few to achieve a desirable solution. This is not unusual in the solution of the conventional displacement model too. In fact the displacement model is diverged when solving this problem at $c/b=$ 0.8 with a single load-step.

The convergence of the mixed is also good for this problem. Fig.6.7(b) shows that when $c/b=$ 0.6, two iteration is enough to yield a result with an average error less than 0.2%, while figure for $c/b=$ 0.8 is three iterations to yield a result with an average error about 1.7%. Fig.6.7(a) again shows consistency between the error and the value of Π_{ep}.

In comparison with the analytical solution, excellent results for both spherical shell and cylinder cases have been obtained by the mixed model. However these cases are governed by the problems where stress distribution is either axially or spherically symmetrical. It remains to demonstrate how this model behaves for problems with emphasis on the shear stress in the next example.

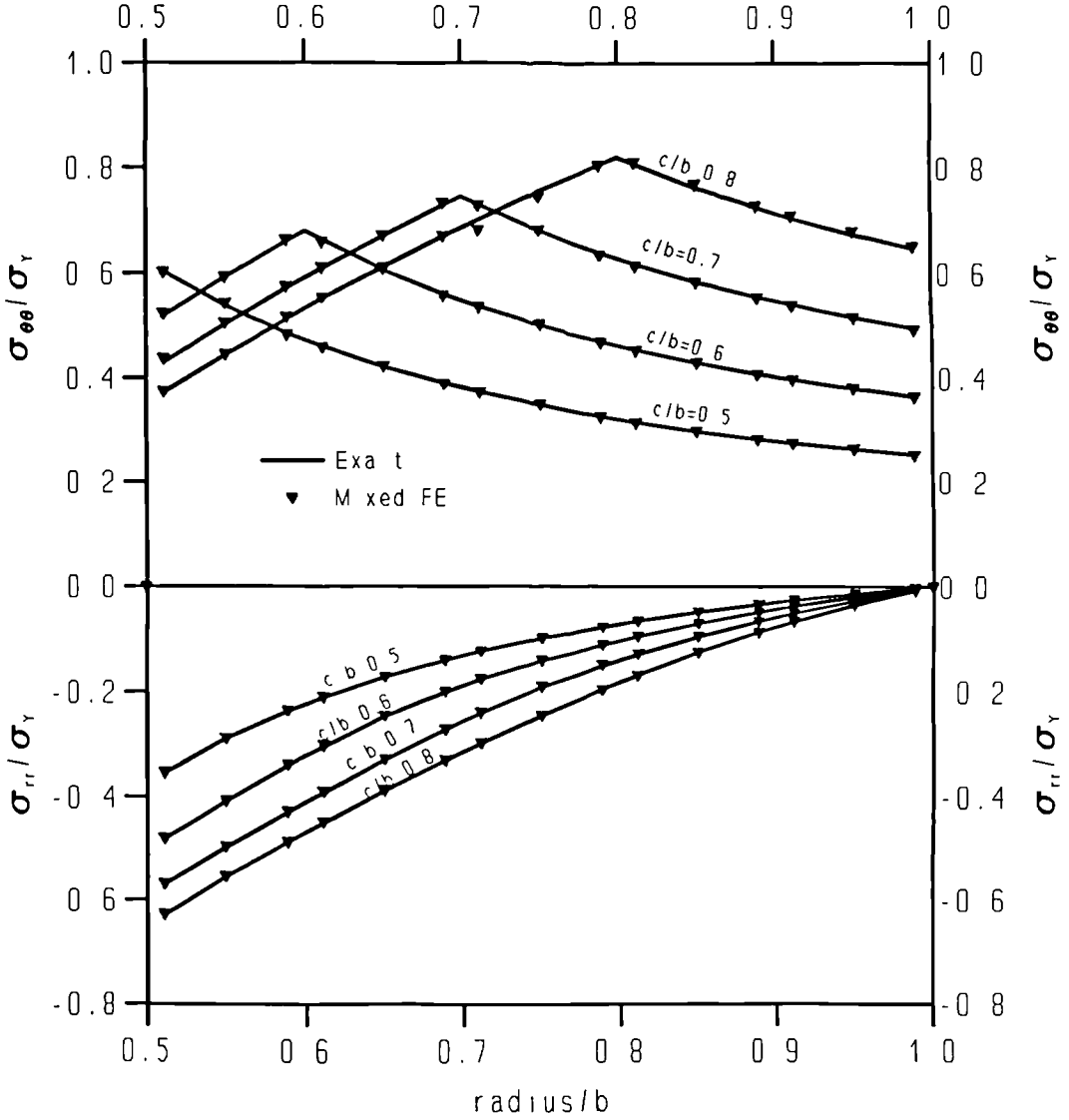


Fig.6.6 Stress distributions in an elasto-perfectly plastic cylinder under internal pressure. A FE mesh with 50 8-noded elements are used in the computation.

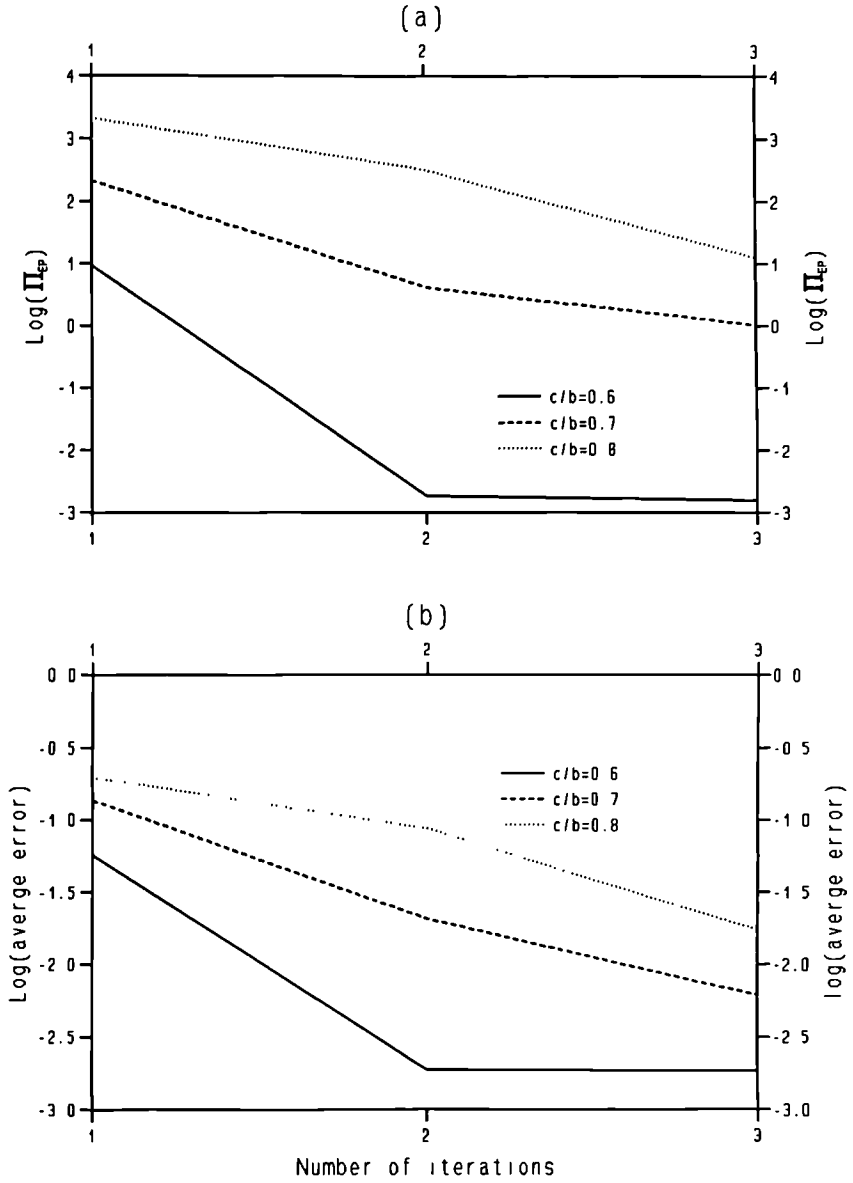


Fig.6.7 Convergence of the elasto-plastic solution of a cylinder under internal pressure. A FE mesh of 50 8-noded elements with 3×3 integration are used in the computation.

6.4.3 Cantilever by a Load at an End

A uniform cantilever of length L, having a rectangular cross section of width b and height $2T$, is loaded by a concentrated load P at the free end. It can be anticipated that yielding first occurs at the top and bottom corner of the built-in end, where the bending moment has its greatest value, the corresponding load M_E being such that $M_E = P_E L$. When the load P exceeds P_E, the plastic zones spread symmetrically inward from the corners.

The FE model is as same as in fig.3.25. Only one mesh with 80 8-noded elements is used in the study. The integration used is 3×3. Two penalty numbers are taken as $\rho_c = 10.0$ and $\rho = 50.0$.

The load-deflection curve of the elasto-plastic beam is plotted dimensionlessly in fig.6.9. D is the deflection of the beam at the free end under load P. D_E is the maximum elastic deflection of the beam at the free end, which corresponds to load limit for elasticity P_E.

It is seen that at the early stage of yielding ($P/P_E < 1.3$), the displacement model and the mixed model give very similar curves, both deviate slightly from the analytical one. This is probably due to the inaccuracy of the analytical solution resulting from the simple beam assumption.

When the load is getting larger ($P/P_E > 1.3$), the mixed model gives much more accurate deflection than displacement model in comparison with the analytical one. The conventional displacement model tends to give small deflections when the plastic range is built up. Even surprisingly, the results obtained by the displacement model are solved in ten load-steps, while those by the mixed model is only one single load-step.

Theoretically the maximum load is $P/P_E > 1.5$ at which the D/D_e is infinite. Both FE models fail to get a converged result when $P/P_E > 1.44$.

The normal stress and shear stress are shown in fig.6.9. Since the mixed model has a difficulty near the built in end of the beam, we choose a section $x/L = 0.15$ to analyse. In the case of σ_{xx}, results from the mixed model are very close to the analytical one especially in the area of plasticity. Like other examples, difficulties arise near the elastic-plastic boundary. The mixed model again gives less good results in this area.

As to the shear stress σ_{xy}, the mixed model gives very good results near the upper and lower edges, where plastic deformation occurs. However the higher shear stress is predicted near the mid area of the section. The shear stress shown is calculated in terms of the arithmetic average of σ_{xy} and σ_{yx}.

6.4.4 Comments on the Mixed model for Elasto-plasticity

Three elasto-plastic solutions have just been produced in the previous sections. In these applications, it was demonstrated that the mixed FE formulation described in this chapter can simulate problems with large portion of plastic deformations correctly. The larger the plastic strain is, the greater value of ρ to impose the yield criterion eq.(6.1) should be chosen. For a normal elasto-plastic problem it should be chosen to be 50.0 to 500.0.

The stress and deformation results obtained from the mixed model are also highly accurate, both in the case with simple axial stress (section 6.4.1 and section 6.4.2) as well as with severe shear stress (section 6.4.3). Compared with analytical results, the mixed model gave closer deformation results in the beam bending in section (6.4.3) than the displacement model.

In addition, the mixed model also has better convergence to the solution. All solutions in these three examples were obtained with one single load step by the mixed model, and only a few iterations (3-5) are enough to give converged results. While by the displacement model, multi-load steps, each with a few iterations, are required.

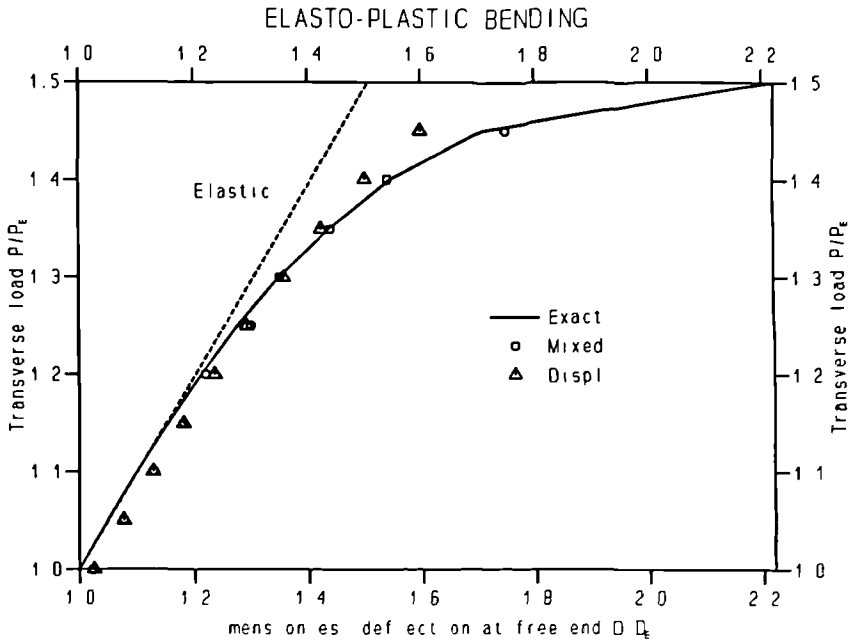


Fig.6.8 The tip deflection of the elasto-plastic bending of a beam. A FE mesh of 80 8-noded elements with 3×3 integration is used. $\rho_c=10.0$ is taken.

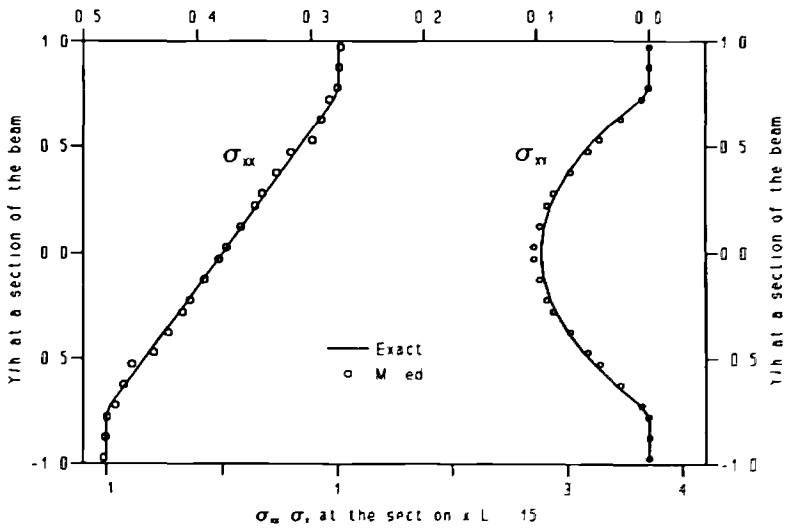


Fig.6.9 Stress distributions of the elasto-plastic bending of a beam at section $x/L=0.15$. A FE mesh of 80 8-noded elements with 3×3 integration is used. $\rho_c=10.0$ is taken.

CHAPTER 7
MECHANICAL SIMULATION OF SPOT-WELDING BY THE MIXED METHOD

7.1 INTRODUCTION OF RESISTANCE SPOT-WELDING

Resistance spot-welding was invented in 1877 by Elihu Thomson and has been widely used since then as a manufacturing process for joining sheet metal. Even though resistance spot-welding is over 100 years old, the physics of the process has not been well understood, however this has not hindered its industrial application for joining a large variety of metals. A schematic representation of a single point resistance spot-welding is shown in fig.7.1.

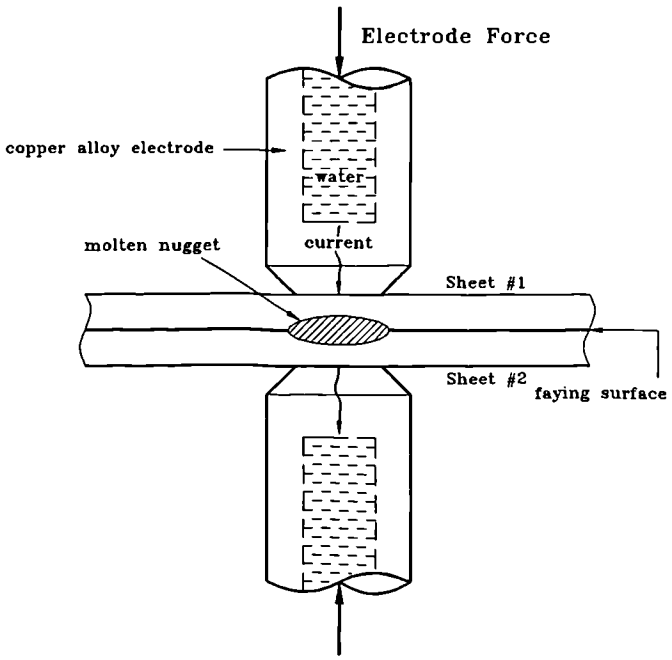


Fig.7.1 The schematic picture of a single point spot-welding

7.1.1 Physics Process of Resistance Spot-Welding

Basically, welding is accomplished by passing a large electrical current through the materials to be joined. The welding current is introduced through a pair of electrodes

which also clamp the workpieces tightly together. The Joulean thermal energy generated in the region of current concentration between the electrodes produces a molten nugget. The molten nugget cools upon termination of the welding current and solidifies into a fused, welded joint.

Physically, resistance spot-welding is a coupled electrical-thermal-mechanical problem. The discrete stages in the process are: a) electrode squeeze, b) weld, c) hold and d) release.

A squeeze load is applied to the sheets to be attached through electrodes, which create intimate contact between the sheets of metal and a pathway for current flow during the welding stage. It also might damage surface oxide on the faying surface. When a current passes through, the contact resistance at the electrode-sheet interface and at the faying surface results in the greatest heat generate at these surfaces. This heat generation produces a temperature increase at all of the surfaces, with the increase being greatest and most rapid at the faying surface due to poorer cooling (the copper electrodes are water cooled). Melting begins at the faying surface and the resulting nugget grows towards the electrodes until current flow is terminated. The electrode squeeze load continues to applied during the hold stage until cooling has occurred.

7.1.2 Resistance Spot-Welding of Aluminium

Resistance spot-welding is the most common joining method used in the automotive industry. The method has been well proven for joining *low carbon steels*. All the parameters for controlling the spot-welding process have been thoroughly studied and methods to control the quality of a weld in low carbon steels are well established.

However for the resistance spot-welding of aluminium, such a position has not been reached. The high electrical and high thermal conductivities of aluminium cause big problems for spot-welding since heat generation and concentration are two important factors in producing a weld. The lack of specific standards in spot-welding of aluminium becomes a big obstacle if aluminium is to be used in the automotive industry.

In recent years there has been a move towards improving the degree of corrosion protection so as to increase vehicle life or decrease weight of the vehicle. Both

aluminium and *coated steel* offer better corrosion protection than the low carbon steels currently used. However aluminium is roughly one third the weight of steel and has considerable strength, which means that a smaller engine with less emission is generally sufficient for the car to achieve a similar performance to a steel car.

Therefore an understanding of the resistance spot-welding of aluminium becomes an important current issue.

7.1.3 Controlling Parameters in Spot-Welding

The fact that resistance spot-welding is a coupled electrical-thermal-mechanical process means that controlling parameters in spot-welding may be divided into three categories:

a) Electrical aspects. These include: the type of current (AC, DC or MF); density and period of the current applied to the work-piece and the contact resistance at the electrode surfaces and the faying surfaces (mating surfaces).
b) Thermal aspects. The cooling system within electrodes can be controlled: the temperature and the flow speed of the water in cooling hole and the location of cooling hole within the electrodes. The thermal conductivities of electrodes and the workpiece can affect welding, but they are not controlling parameters of spot-welding equipment as they are usually keep constant.
c) Mechanical aspects. There are a lot of parameters which affect mechanical behaviours of the electrodes and the work-pieces. But only some are related to the formation of spot-welding[53]: the squeeze loading pattern; the geometry of the head of the electrodes and location of the cooling hole. These can largely affect the contact resistance at the electrode surface and the faying surface.

The above parameters acts in a coupled manner. For example, changes of electrical parameters obviously affect temperatures of the work-pieces and the electrode which will lead to changes of mechanical aspects. The varied mechanical behaviour will normally alter the contact resistance, which in turn affect electrical and thermal aspects. The more details concerning the controlling parameters can be found in Kim's work in ref.[53]. Also a review of current experimental studies and numerical simulation of

spot-welding will be given in **Appendix D.**

7.1.4 Work in this chapter

The **Spot-Welding in Aluminium group** in Newcastle University is devoted to the numerical simulation of both electro-thermal and thermo-mechanical behaviours during spot-welding. This is sponsored by *Alcan International Ltd.*

In this chapter, the mixed FE model introduced in previous chapters will be used to solve a very simple, but very useful case in the thermo-mechanical simulation of spot-welding. That is the squeezing of the aluminium sheets by a pair of truncated electrodes, where the sheets are initially attached together by adhesive materials.

The fact that adhesive is introduced in the structure causes difficulties to produce a weld since adhesives are not conductors. It also increase difficulties for the numerical simulation since both solid and fluid have to be treated together to model the thin layer of adhesives, aluminium sheet itself and the electrode.

In the following section, an axisymmetric elasto-plastic model will be developed with the adhesives taken into account in terms of thin-film theory. Our purpose is to identify some factors which are related to the deformation of the adhesives during the initial squeeze of aluminium sheets in the cold condition, and eventually control the minimum thickness of the deformed adhesive and the size of the **entrapment** between two aluminium sheets. In the meantime, it is our intention to verify the validity of the mixed model when applied to such a complicated problem.

A more complete study of the thermo-mechanical simulation of spot-welding with an pair of curved electrodes is presented in **Appendix D** where only conventional displacement is used.

7.2 FE-FLUID MODEL FOR SPOT-WELDING

In this section, a new FE model is proposed in order to simulate the initial squeeze of attached aluminium sheets by electrodes. It is constructed by introducing a traction boundary condition which can be obtained for the **thin-film theory**[1] in fluid dynamics, which replaces the displacement constraint at the faying surface of the FE mesh.

In section 7.2.1, fluid pressure of adhesive caused by a deformation of aluminium sheets will be deduced by means of the thin-film theory. Then in section 7.2.2, an interactive model, **FE-fluid model**, will be presented by incorporating this fluid pressure onto a normal mixed FE model in terms of a deformation dependent applied traction.

7.2.1 Fluid Pressure of a Deformed Adhesive

Suppose that we have an incompressible Newtonian fluid of constant density ρ and constant viscosity μ. Its motion is governed by the **Navier-Stokes equations**

$$\frac{\partial \dot{\mathbf{u}}}{\partial t} + (\dot{\mathbf{u}} \cdot \nabla)\dot{\mathbf{u}} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \dot{\mathbf{u}} + g \tag{7.1}$$

and the continuity equation

$$\nabla \cdot \dot{\mathbf{u}} = 0 \tag{7.2}$$

The character of a steady viscous flow depends strongly on the relative magnitude of the terms $(\dot{\mathbf{u}} \cdot \nabla)\dot{\mathbf{u}}$ and $\nu \nabla^2 \dot{\mathbf{u}}$ in the Navier-Stokes equation (7.1). Let h be the thickness of flow and L the typical horizontal length scale of the flow. When $h \ll L$, the equation of motion can be simplified by ignoring inertia terms and considering only flow in the long direction in eq.(7.1). In axisymmetric cases, it becomes[1]

$$\frac{\partial p}{\partial r} = \mu \frac{\partial^2 \dot{u}_r}{\partial^2 z} \tag{7.3}$$

where $\dot{u}_r$ is the velocity of fluid at r direction and p is the pressure of the fluid being a function of r and t only. The incompressibility condition $\nabla \cdot \dot{\mathbf{u}} = 0$ takes form of

$$\frac{1}{r}\frac{\partial (r\dot{u}_r)}{\partial r} + \frac{\partial \dot{u}_z}{\partial z} = 0 \tag{7.4}$$

in axisymmetric cases. Eq.(7.3) and eq.(7.4) are called the thin-film equations[1]. Now let us derive the pressure p caused by a movement at z-direction ($\dot{u}_z$) by making use

of thin-film equations (7.3) and (7.4).

Say a fluid is enclosed by two symmetric solid bodies with boundaries at $z=H(r,t)$. The moving speed of the boundaries is

$$\dot{u}_z = \frac{dH}{dt} \tag{7.5}$$

as seen in fig.7.1.

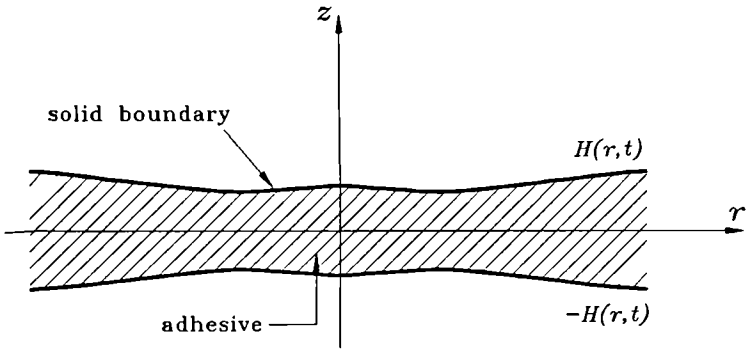


Fig.7.2 The adhesive enclosed by symmetric solid boundaries.

By integrating eq.(7.3) with respect to z twice and applying the no slip condition at boundaries:

$$\dot{u}_r = 0 \quad \text{at} \quad \begin{cases} z = -H(r,t) \\ z = H(r,t) \end{cases}$$

it follows the velocity at radial direction $\dot{u}_r$

$$\dot{u}_r = \frac{1}{2\mu}\frac{\partial p}{\partial r}(z^2 - H^2) \tag{7.6}$$

It is straightforward to derive $\dot{u}_z$ by substituting eq.(7.6) into eq.(7.4).

$$\frac{\partial u_z}{\partial z} = -\frac{1}{2\mu r}\frac{\partial}{\partial r}\left[r\frac{\partial p}{\partial r}(z^2 - H^2)\right] \tag{7.7}$$

Integrating both side of eq.(7.7) with respect to z and imposing boundary condition eq.(7.5), it follows that

$$\frac{dH}{dt} = \frac{1}{3\mu r}\frac{\partial}{\partial r}\left[r\frac{\partial p}{\partial r}H^3\right] \tag{7.8}$$

in which the symmetric condition $\dot{u}_z = 0$ at $z=0$ has been used. The pressure p is then easily obtained by integrating eq.(7.8) twice with respect to r to give

$$p = -\int_r^a \left[\frac{3\mu}{rH^3}\int_0^r (\frac{dH}{dt}r)dr\right]dr \tag{7.9}$$

The boundary condition pressure $p=0$ at $r=a$ has been used, where a is the outer radius of fluid. A special case is that the solid boundary is simply a flat disc which means H is not a function of r. This leads a much simpler expression for p

$$p = \frac{3\mu}{4H^3}\frac{dH}{dt}(r^2-a^2) \tag{7.10}$$

Say a load F is acted on the disc, then following relation exists

$$F = \int_0^{2\pi}\left(\int_0^a pr\,dr\right)d\theta = -\frac{3\pi}{8}\frac{\mu a^4}{H^3}\frac{dH}{dt} \tag{7.11}$$

Therefore a load F acted on a flat fluid will cause a pressure p_0 with a distribution like

$$p_0 = \frac{2F}{\pi a^4}(r^2-a^2) \tag{7.12}$$

In next section, we will show how to incorporate the pressure just obtained into a conventional FE method.

7.2.2 Solution Procedure of the FE-Fluid Model

For a solid-fluid coupled problem, there are two types of approaches available. The first is to solve it uniquely by means of FE method where both solid and fluid are treated as an integrated continuum which has big changes of materials properties among each area. As a results, only some modifications of a FE method are required. This also implies that the normal restrictions to the conventional FE method may also apply to this case.

The second is to solve the solid and fluid separately, each of which is modelled

with a different approach. For instance, the solid part is modelled by a FE model while the fluid is modelled by a finite difference model. These two part are combined to form a integrated system such that they satisfy some common boundary conditions at the solid-fluid boundary.

The problem we are facing now is to model such a thin-film fluid that the first approach (FEM) can not be used due to the geometry restraints of FE method in the fluid part. Therefore the second approach is chosen, which is to model solid part by FE method and fluid part by the thin-film theory. In the remainder of this section, we will show how our FE-fluid model works.

Say there is a thin-film adhesive with a initial thickness $H^0=H(r,t)$ and a load P acting it through electrodes and aluminium sheets. Since no deformation of aluminium sheets is made at time $t=0$, the fluid pressure p_0 caused by load P at this time is calculated by eq.(7.12).

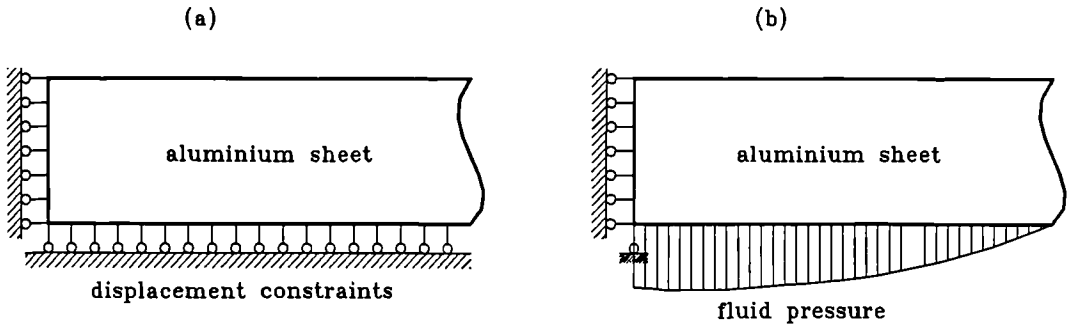


Fig.7.3 The modifications of the force and displacement boundary at the aluminium/adhesive surface. (a) in a conventional model without adhesive; (b) in the FE-fluid model.

The external force P and the pressure p_0 form a balanced system applied upon the solid part, as seen in fig.7.3. Therefore the deformation of aluminium sheets at the faying surface $u_z^{(0)*}(r)$ can be easily obtained by elasto-plastic FE method.

However $u_z^{(0)*}(r)$ is not the whole movement of the aluminium surface near adhesive. This is because when FE method is used to model the solid part of spot-welding, it is required that the z-displacement at the lower-central point of the aluminium sheet has to be fixed in order to prevent the rigid movement of aluminium sheets at z direction, i.e.

$$u_{z0}^{(i)} = u_z^{(i)}|_{r=0,z=0} = 0$$

In reality this point at aluminium surface does have a movement at z-direction along the change of the thickness of the adhesive. Therefore the whole movement of the aluminium surface will be the displacement $u_z^{(0)*}(r)$ plus an unknown movement at z-direction at the central point of aluminium $u_z^{(i)}(0)$Writing for any time $t=i$, it is

$$u_z^{(i)}(r) = u_z^{(i)}(0) + u_z^{(i)*}(r) \tag{7.13}$$

The solid-fluid boundary $H^i(r)$ for the adhesive at $t=i$ is then assumed as

$$H^{i+1}(r) = H^i(r) + u_z^{(i)}(r) \tag{7.14}$$

The velocity of the boundary $H^{(i+1)}$ is approximately calculated by

$$\frac{dH^{i+1}}{dt} \equiv \dot{u}_z^{(i+1)} = u_z^{(i+1)}/\Delta t^i \tag{7.15}$$

where Δt^i is the ith time increment of the load P. The pressure p^{i+1*} is then calculated by eq.(7.9) with eq.(7.15) and the modified boundary in eq.(7.14).

This pressure p^{i+1*} is then applied again to the FE model, which gives a updated solid-fluid boundary $H^{(i+1)*}$. Theoretically $H^{(i+1)}$ and $H^{(i+1)*}$ should be the same when the solution is found. In reality, a few iterations of the above circle are to perform.

It remains to decide how to get the unknown movement of adhesive/aluminium at the central point $u_z^{(i)}(0)$, on which p and H^{i+1} both depend on. This can be easily obtained by the fact that H^{i+1} with a velocity of $\dot{u}_z^{(i+1)}$ should yield a pressure p which should balance the external load P at z-direction of the aluminium sheets. Golden-line search introduced in **Appendix B** is used to find it.

7.3 RESULTS AND DISCUSSION

In this section, we are going to give a practical application of FE-fluid model, which is the simulation of the initial squeeze of two attached aluminium sheets by electrode. The geometry and mesh of the FE-Fluid model used in this section is the same as the one defined in section 4.5.3. The constraints of the FE-Fluid are modified to include

the pressure boundary conditions at the surface between the aluminium sheets and the adhesive, as shown in fig.7.3.

The material properties for both aluminium sheets and the copper electrode are shown in Tab.I. All the data are for the room temperature (25°C).

Tab.I Material constants of the aluminium sheet and the electrode

Material constants	Young's modulus	Poisson's ratio	Yield stress(cold)	Plastic modulus
aluminium	72000MPa	0.25	158MPa	294MPa
copper	126000MPa	0.25	560MPa	294MPa

The constitutive relation of both the materials is taken as a linear hardening elasto-plasticity. Since loading is a dynamic process, the $P(t)$ pattern needs to be given too. For simplicity, the linear relation $P(t) = P_{\max} t$ is chosen. Also it is assumed that whole P_{max} is reached by one single time (or loading) step in the computation.

The parameters concerning the layer of adhesive is that the initial thickness of the layer of adhesive is $H_0 = 100\mu$m. Two viscosities, 500PaS and 1000PaS, of the adhesive, two forces P_{max}, -5000N and -10,000N, and two duration times, $t = 0.5$sec and $t = 1.0$sec of loading are used in the simulations.

In order to judge the validity of this FE-fluid model in which the mixed FE model is used to model the solid part of the construction, another FE-fluid model was constructed in the same way as we described in the last section except that the solid part is modelled by means of the conventional displacement model. (All the results presented in **Appendix D** are obtained by such a model)

7.3.1 Deformation of the Adhesive

During spot welding process, it is required that the insulation layer of adhesive between two aluminium sheets be broken down successfully by squeezing electrodes at the initial stage, so that the current can then pass through. In the following part of the section, the deformation of the adhesive squeezed by electrodes through two aluminium sheets

will be analysed by the FE-fluid models just discussed.

In fig.7.4 the relative thickness of adhesive and the corresponding fluid pressure caused by the deformation are plotted along the radius. The viscosity of the adhesive is assumed as 1000 PaS, while the force and its duration are -10000N and half a second.

It is interesting to find that the minimum thickness of the layer is not situated at the centre of the faying surface but a distance away from it. For example, the minimum relative thickness is about 0.322 at r=3.5mm, but at the centre r=0, it is about 0.33.

Imagine that the location with the minimum thickness is where two aluminium sheets first contact each other at faying surface, and therefore the electric current first passes through. It is possible that the centre parts of aluminium sheets will never contact each other. In practical spot-welded specimen, it is seen that the faying surface of some spot-welds shows a ring-alike weld rather than a solid circle[16]. In other words, there must be some adhesive "trapped" inside this small space. This is verified by the fact that in some specimens there is the remains of the adhesive in the centre portion of the weld. To find out what is happening here is one of our aims of studying the deformation of the adhesive.

The results in fig.7.4 show that FE-fluid models with displacement model and the mixed model give very consistent fluid pressures and thicknesses. When time duration is taken as one second, the thickness from the mixed model is slightly different from that from the displacement model while pressure is still close, as shown in fig.7.5. The minimum relative thickness becomes about 0.26 at r=3.5, while at the centre where r=0.0, it is about 0.27.

When the viscosity of the adhesive is reduced as 500 PaS, as in fig.7.6, the minimum relative thickness further reduces to about 0.21 at r=0.38 and at the centre it is about 0.22. Near the minimum one, the deviation between the displacement model and the mixed model is bigger than at the centre portion. But the biggest one is near the edge of the sheet.

Among all results in fig.7.4, fig7.5 and fig.7.6, the patterns of the deformation of the adhesive are similar. So are fluid pressure patterns in them. However if the squeeze force P_{max} is changed, say P_{max}=-5000N, the situation is no longer the same, as seen in fig.7.7.

Obviously, the fluid pressure should be different since it has to balance the squeeze force P_{max}. The deformation of adhesive is also slightly different than before. The minimum relative thickness is located at r=3.5 with 0.266, while at the centre it is about 0.27. The difference between them is 0.04, less than those in last three cases where it is around 0.1. Therefore it might say that the entrapment of the adhesive is less severe than in the last three cases.

Further discussions about the effects on the entrapment by various factors can be found in **Appendix D**.

7.3.2 Stress Distribution within the Electrode and the Aluminium Sheet

The stress distributions of the electrode and the aluminium sheet are shown in fig.7.8, fig.7.9, fig.7.10 and fig.7.11 in terms of colour contour maps. Only one quarter of the whole structure is shown in those graphs.

In fig.7.8 the normal stresses (σ_{zz}) obtained the mixed model are presented. It is seen that the stress levels in the lower-central region is the highest in the aluminium sheet, which is consistent with the pressures seen in fig.7.4 to fig.7.7. While at the electrode/aluminium surface, the highest stress level is found near the edge of the electrode. In fact, there should be a singularity at the edge if only elasticity is modelled, as seen in fig.4.17 and fig.4.19.

The corresponding shear stress distributions τ_{rz} obtained from the mixed model are presented in fig.7.9 in the same way. The shear stress is calculated by the average of τ_{rz} and τ_{zr}.

Generally speaking, the shear stress level in both aluminium sheet and the electrode are low. At the faying surface, it is zero since it is a symmetry plane. The biggest shear stress exists at the electrode/aluminium sheet interface. It provides a tearing force to break down the oxides at aluminium surface which enable an electric current to pass from the electrode to the aluminium sheet.

The corresponding results from the FE-fluid model by displacement model are shown in fig.7.10 and fig.7.11. It is seen that the normal stresses (σ_{zz}) obtained from both models (fig.7.8 and fig.7.10) are quite similar. However the contour maps of shear stresses (fig.7.9 and fig.7.11) appear slightly different.

It is understood that the mixed model faces a big difficulty at a corner with no shear on one side and a severe shear on the other side because the moment equilibrium (3.3) is not satisfied automatically in the mixed model. It is imposed approximately by a penalty term, which becomes incompetent in this extreme condition. The corner formed by the electrode wall and the top surface of the aluminium sheet is just in this kind of situation. As a result, the shear stress predicted by the mixed model is lower than it should be.

7.3.3 Comments on the FE-fluid Model with Mixed Model

The results we have just shown demonstrate that the mixed model is capable of handling a more complicated situation in addition to those in chapter 3 to chapter 6 where only some basic problems were discussed. In comparison with the conventional displacement model, it gives very good normal stresses, fluid pressure and the deformation of the adhesive results. But the shear stresses in some region is not quite good, which can be improved by using smaller elements.

Together, these results provide useful information about the mechanical behaviours during initial squeeze stage of the spot welding process. They also give some ideas about how the adhesive deforms under the squeeze and how to minimize the entrapment of it.

Although only a single loading step with a linear function of time is used in the simulation, it is readily to extend such a FE-fluid model to a more realistic one where *P(t)* can put into the model in a arbitrary function of time by using multi-loading step with varying time increment at each step. As the matter of fact such a method is used in **Appendix D**.

Results from the mixed model in this chapter are obtained under penalty number ρ_c = 100.0. Deformation of the adhesive obtained under smaller one tends to be "flatter" than those we have just shown. The reason of this may arise from the fact that there is a fairly large shear stress existing near the electrode/sheet interface and therefore higher ρ_c is needed to maintain the satisfaction of $\tau_{zr} = \tau_{rz}$.

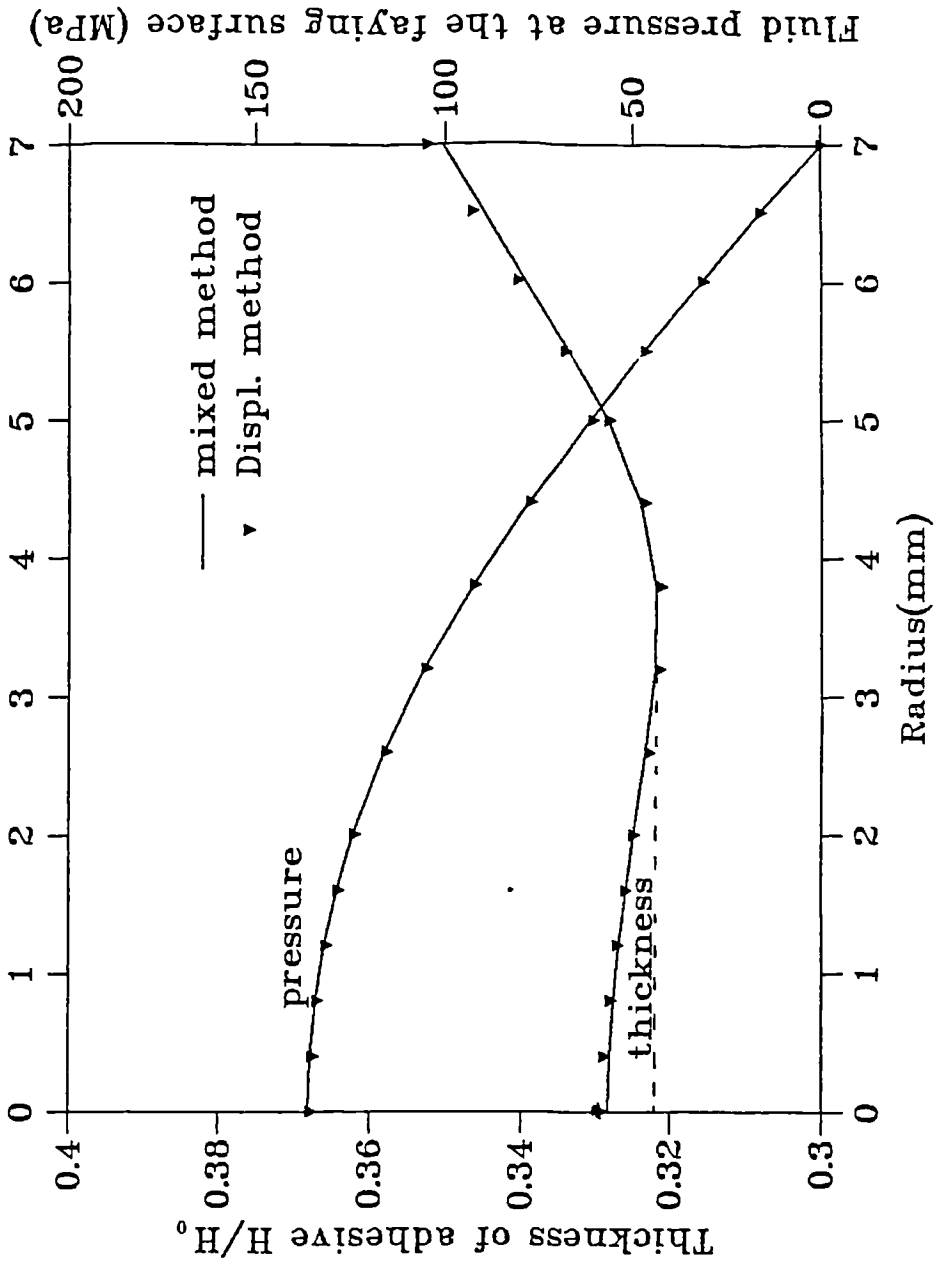


Fig.7.4 The deformation of adhesive and the fluid pressure caused by it at the faying surface. Load duration T=0.5sec., applied force F=-10,000N and viscosity of adhesive μ=1000PaS.

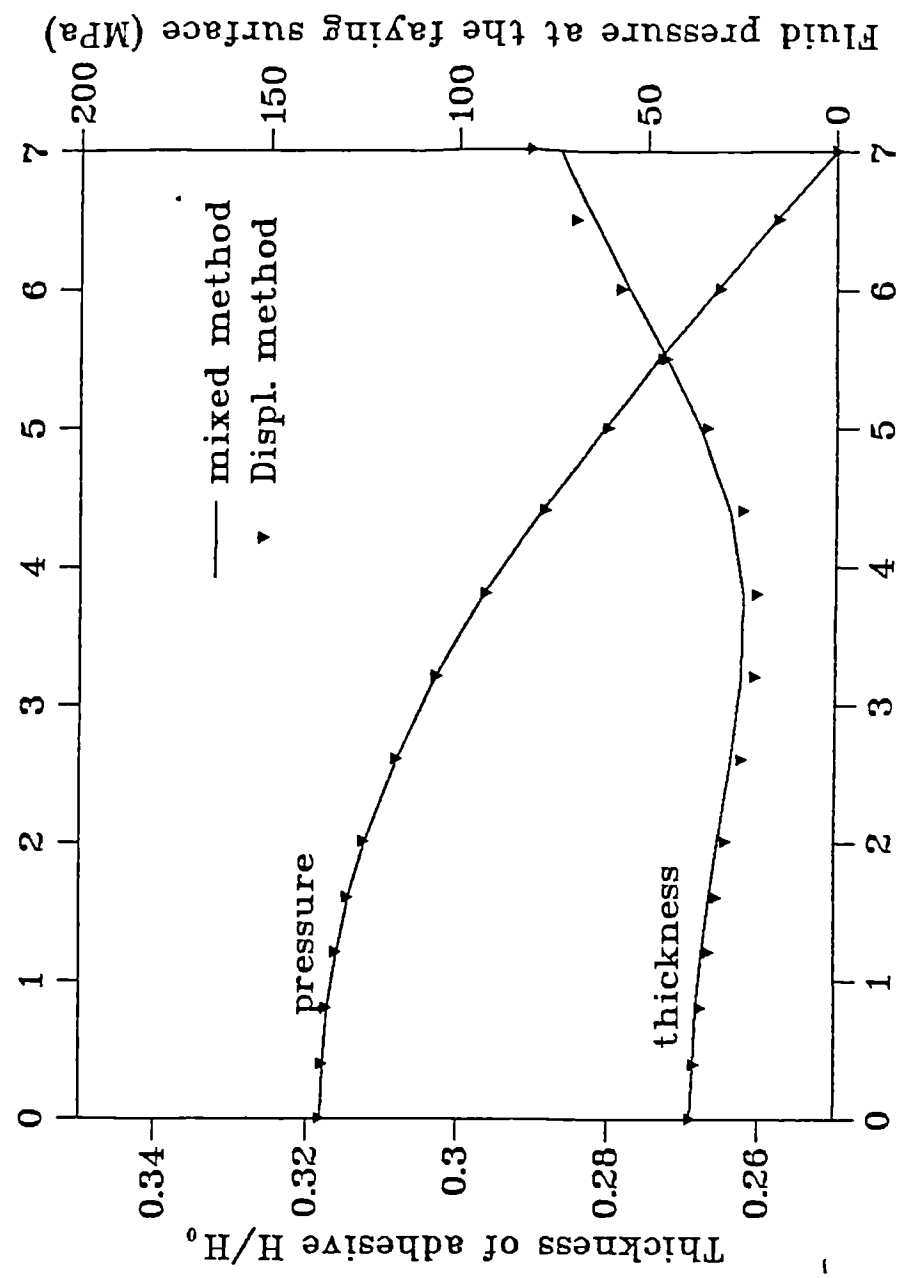


Fig.7.5 The deformation of adhesive and the fluid pressure caused by it at the faying surface. Load duration T=1.0sec., applied force F=-10,000N and viscosity of adhesive μ=1000PaS.

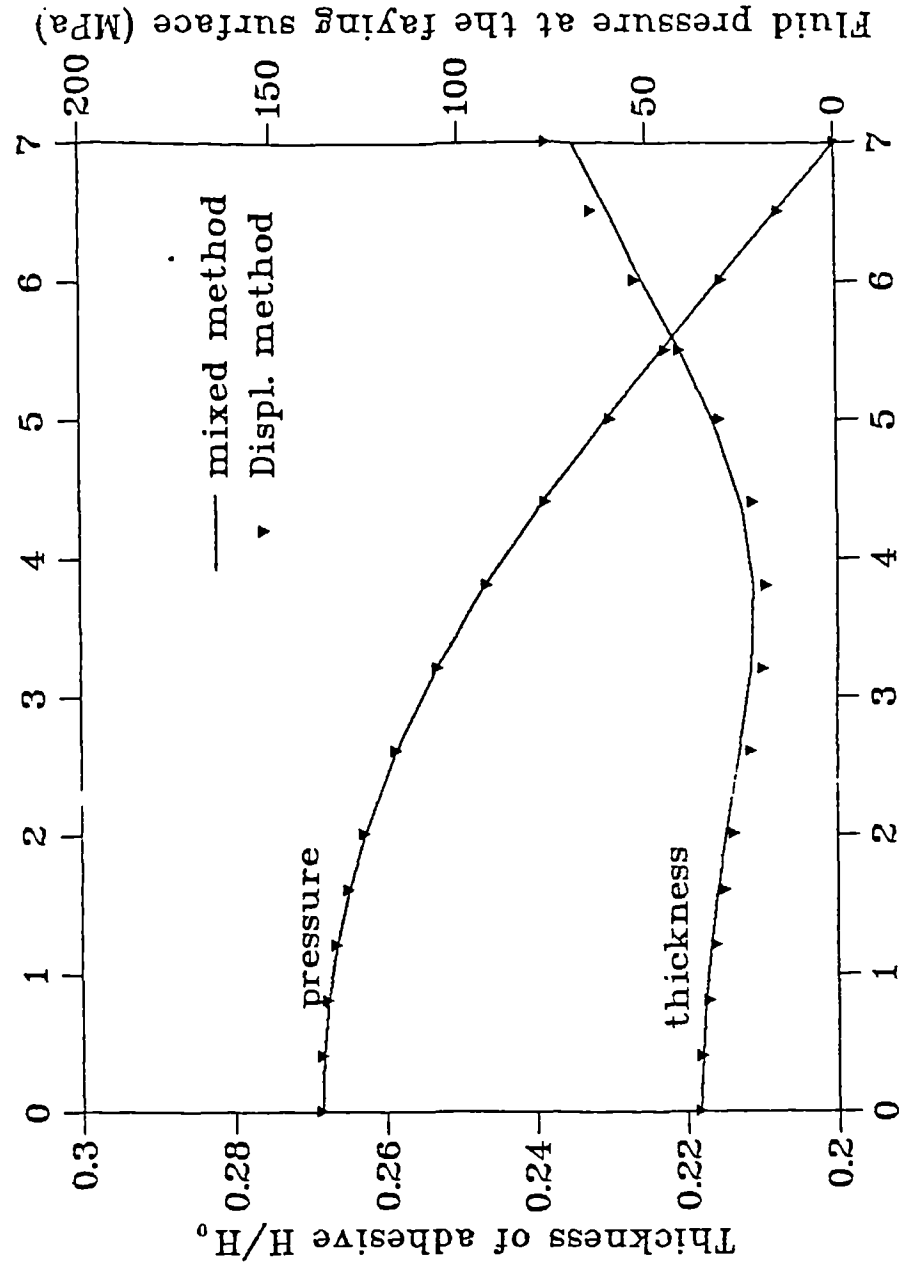


Fig.7.6 The deformation of adhesive and the fluid pressure caused by it at the faying surface. Load duration T=1.0sec., applied force F=-10,000N and viscosity of adhesive μ=500PaS.

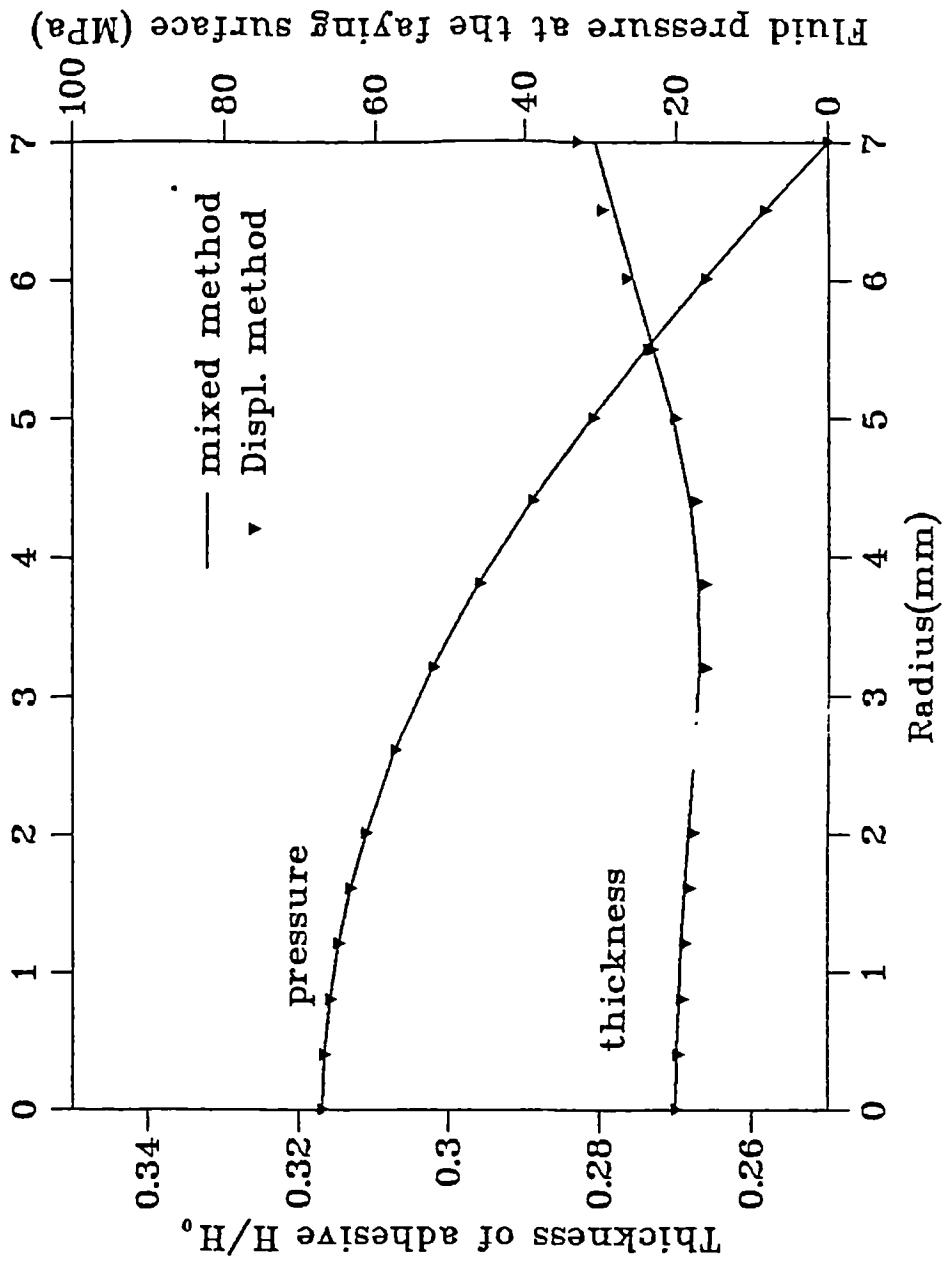


Fig.7.7 The deformation of adhesive and the fluid pressure caused by it at the faying surface. Load duration T=1.0sec., applied force F=-5,000N and viscosity of adhesive μ=500PaS.

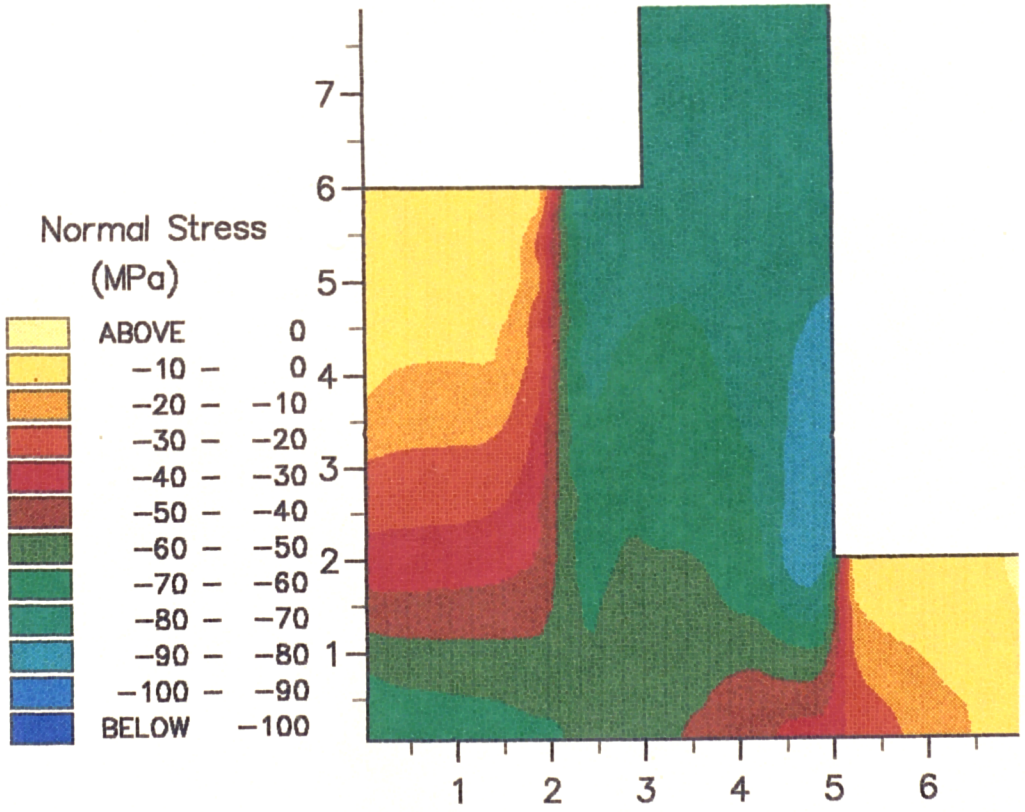


Fig.7.8 The normal stress (σ_{zz}) obtained from the FE-fluid model by mixed model. Load duration T=1.0sec., applied force F=-5,000N and viscosity of adhesive μ=1,000PaS.

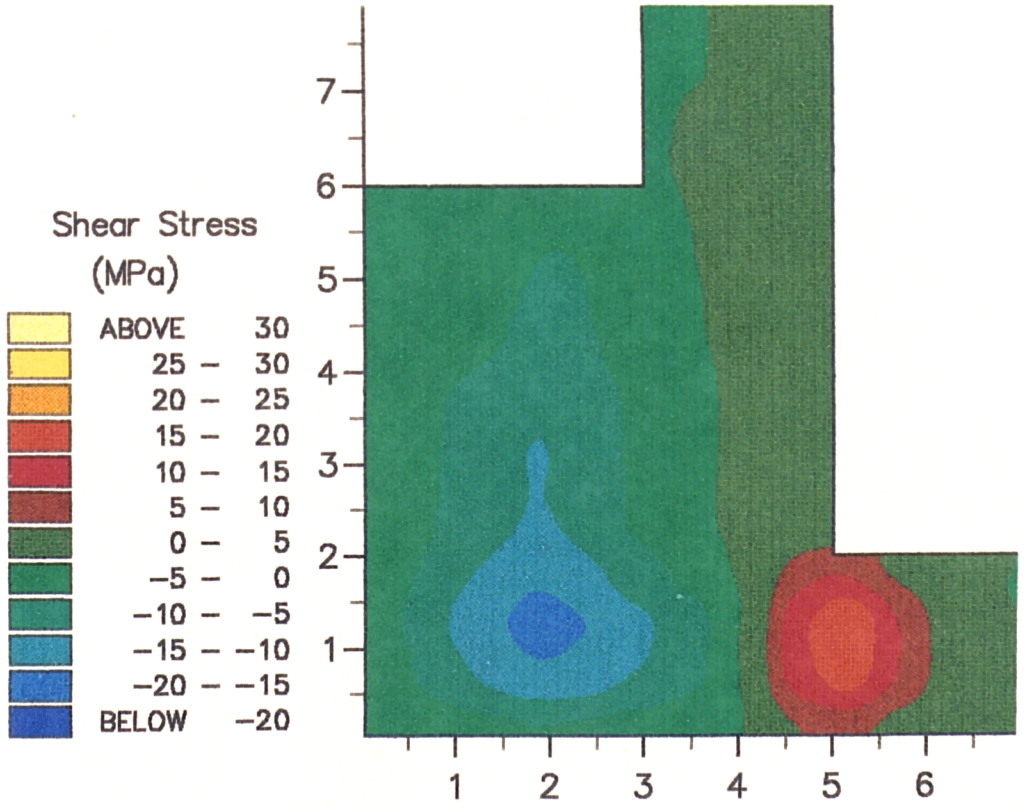


Fig.7.9 The shear stress (τ_{rz}) obtained from the FE-fluid model by mixed model. Load duration T=1.0sec., applied force F=-5,000N and viscosity of adhesive μ=1,000PaS.

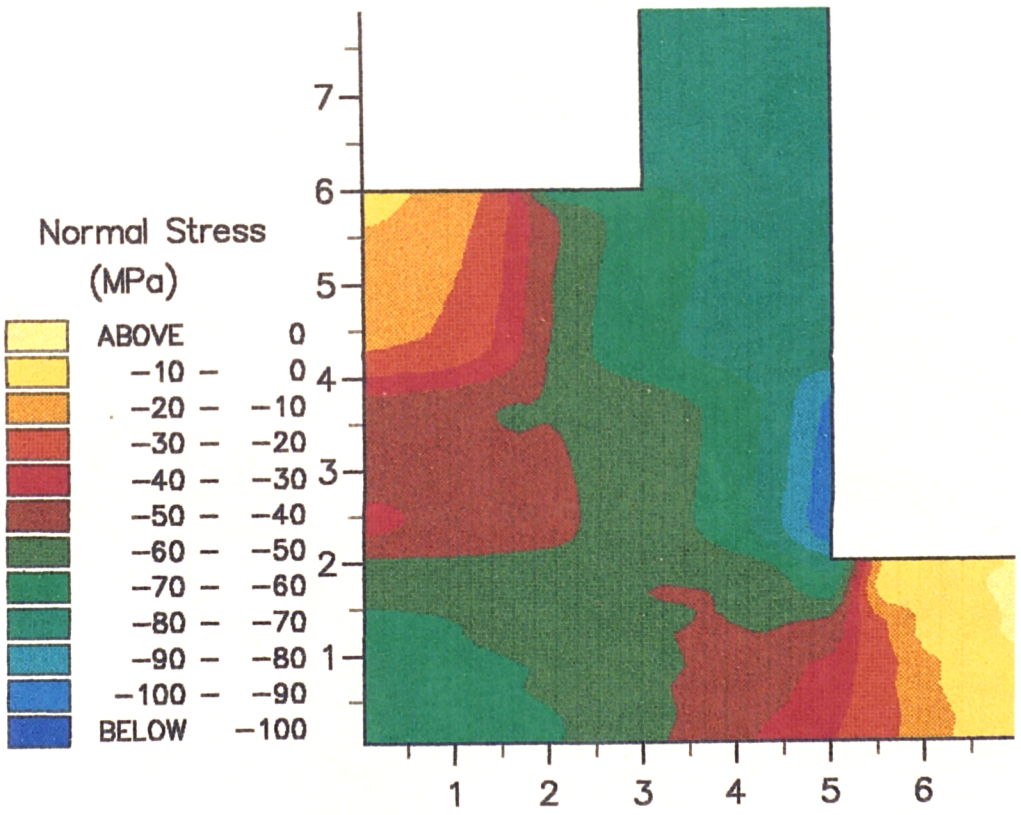


Fig.7.10 The normal stress (σ_{zz}) obtained from the FE-fluid model by displacement model. Load duration T=1.0sec., applied force F=-5,000N and viscosity of adhesive μ=1,000PaS.

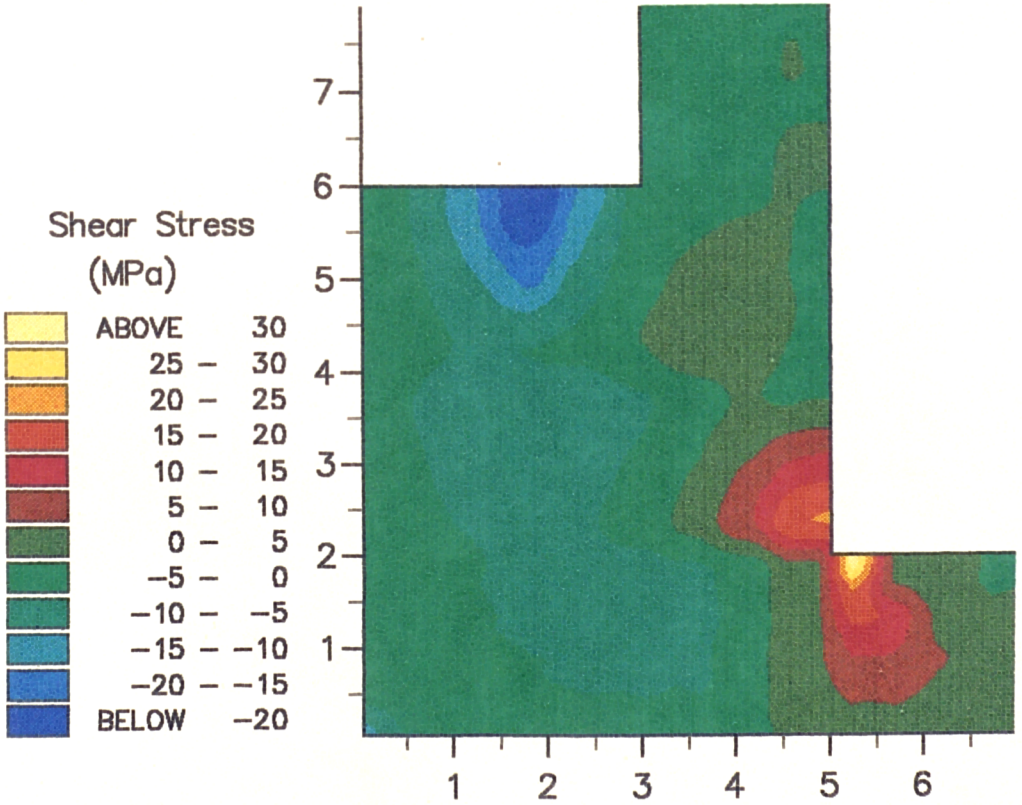


Fig.7.11 The shear stress (τ_{rz}) obtained from the FE-fluid model by displacement model. Load duration T=1.0sec., applied force F=-5,000N and viscosity of adhesive μ=1,000PaS.

CHAPTER 8
DISCUSSION AND CONCLUSION

A new mixed model was introduced in the previous chapters. It was applied to some elementary examples as well as in the simulation of part of the spot-welding process. As addressed in chapter one, the main objective of this thesis was to establish a model which arises from very basic concepts of continuum mechanics, and to provide a better understanding of mutual relationship between all the basic equations encountered in solid mechanics. In the remains of this chapter, we will give some remarks on this model.

8.1 PERFORMANCE OF THE MIXED MODEL

Basically the work in this thesis was about constructing and testing three new mixed extremum principles which approximately imposed the constitutive relationships for elasticity, rigid plasticity and elasto-plasticity respectively. The kinematic and static constraints to be satisfied in advance by the strain-displacement relation and by means of little used first order stress functions for plane problems and a newly established stress functions for axisymmetric problems.

Chapter three and chapter four dealt with elastic problems. For some problems where shear stress is small, such as in expansion of a thick cylinder and a hollow sphere, the model gave very good stress and displacement results.

Particularly for nearly incompressible materials, it had a superior performance over the displacement method. Unlike the displacement model where 2×2 integration must be used in order to obtain a meaningful stress results, the mixed model gave good stress results under both 2×2 and 3×3 integrations. Therefore for the problem with small shear the mixed model is not sensitive to the integration scheme.

For those that the shear stress is obviously as important as the normal stress, as in bending of a short beam (section 3.6.4) and bending of a disc (section 4.5.1), an extra penalty function is needed to maintain the symmetry of a stress tensor. The reason for this is that the stresses represented by the first order stress do not automatically satisfy the moment equilibrium eq.(3.3).

Different penalty numbers ρ_c were used in the analysis of beam bending and disc bending. It is seen that for plane problems, $\rho_c = 5.0$ is generally sufficient for obtained good displacement and stress results for 4×4 and 8×10 meshes with 2×2 integration. The finer mesh is, the better results can be obtained.

However, for such problems integration schemes did have some effect on the results of the mixed model. For example, in order to obtain good displacement and stress results under 3×3 integration, $\rho_c = 10.0$ was generally required.

For axisymmetric problems, the requirement of symmetry of the stress tensor is emphasized by the fact that a larger penalty number is needed for obtaining displacements and stresses with the same accuracy as for plane problems. For example in bending a disc in section 4.5.1, $\rho_c = 10.0$ was needed for 2×2 integration, while $\rho_c = 50.0$ for 3×3 integration.

The above examples demonstrated that the idea to decompose the Love's stress function into three first-order stress functions, which satisfy the force equilibrium eq.(4.3), along with a constraint which imposes the moment equilibrium eq.(4.4) is successful. This enables a continuous mixed model for axisymmetric problem to be constructed with C^0 continuity.

As is mentioned in chapter two, "excess continuity" can occur in a mixed model with continuous variables where stress discontinuity exists, e.g. at interface of two materials. However, the mixed model in this thesis can reproduce a stress discontinuity, though it is a continuous mixed model. This was illustrated in section 3.6.2 where the elastic punch pressing into a compliant layer was analysed. The discontinuity of stress σ_{xx} can be clearly identified in fig.3.15 and fig.3.20.

A interesting feature of this mixed model is that it has an intrinsic "error estimator" which seems very meaningful and is simple to use. This was illustrated in section 3.6.3 where a plate with a central crack under remote tension was analysed. The error of the solution decreased with the element size in a similar manner as the value of the function Π_{elas} varies at a given point. A similar phenomenon can be found in other examples. In conclusion, the value of the functional Π can be used to judge the error of a solution locally or globally.

Near the area with high stress singularity, such as a crack tip, the mixed model tends to give stresses with a higher singularity than the displacement model when a

normal element is used. However if the corresponding "distorted" element is used, the displacement model overtakes the mixed model, though both models have improved results. This can also be found in section 3.6.3. This means that the singularity caused by moving the mid-point of the element near the crack tip is less effective on the mixed model than it is on the displacement model.

The favourable feature of the mixed model on this is that if we have no idea where to put the "distorted" element, the mixed model can simulate the singularity of the stress more accurately than the displacement model.

In chapter five of this thesis, rigid plasticity was discussed. By means of the Schwarz inequality, the constitutive relation for a rigid plasticity, i.e. Levy-Mises equation, was equivalent to the minimum of a mixed extremum principle Π_{rgd}. The discretization of the displacement and the stress were exactly the same as chapter three and four for elasticity.

Since the system equation from the minimizing the Π_{rgd} was a nonlinear equation, linear search method was used to solve it iteratively. In a simple example presented in section (5.4), which was the expansion of a rigid plastic cylinder, the mixed method gave wonderful results compared with the analytical solution. In addition to the high accuracy of the solution, it also converged very quickly. If the initial guess of the solution is chosen by the corresponding elastic solution, then only one or two iterations were normally enough to get a satisfactory rigid-plasticity solution.

Not surprisingly, the value of the functional Π_{rgd} again showed consistence with the error of the solution.

In plasticity theory, it is well established that the elasto-plastic strain (full strain) is composed of an elastic strain and a plastic strain. In this thesis, this concept was extended to express the relationship among three basic solution methods in solid mechanics, i.e. elastic solution, rigid-plastic solution and elasto-plasticity solution. It can be stated as follows:

An elasto-plastic solution is equivalent to a rigid plastic solution if the elastic strain in full strain is removed in the plastic area. The corresponding elastic area becomes the rigid area in the rigid plastic solution.

This enables us to construct a functional for elasto-plasticity: in the elasto-plastic area, the functional for rigid-plasticity Π_{rgd} is used, while in the elastic area the

functional for elasticity Π_{elas} should be switched to.

In chapter six, such a assumption was tested by various examples in section 6.4. The mixed model gave very satisfactory results in comparison with the analytical results and those from the displacement model. The convergence of the solution was well behaved, especially when a large part of body concerned is in plasticity.

For elasto-plastic bending of a beam, ρ_c was again used. The deflection-load curve predicted by the mixed model was closer to the analytical solution than that by the displacement model.

Therefore we conclude that the assumption made in this thesis is correct. In addition to give better displacement and stress results in some cases, it possesses another advantage which is a quick convergence.

All the solutions made by the mixed model were solved in a single load step. The more plastic strain, the more iteration within this load step is required. However in the displacement model, multi-load step is generally essential to solve the corresponding problems. For a highly plastic problem, more load steps and more iterations are needed.

Part of the reason for this is that in the mixed model, no stress correction is performed. The solution obtained by minimizing Π_{ep} satisfies all the equations governing elasto-plasticity. This is simply not true in the displacement model in the way that stresses are calculated from displacements, thus no yield condition is involved. To perform a normal stress correction, such as predictor-radial return method, a sufficiently small step is necessary.

For a more complex problem, chapter seven solved a fluid-solid coupling problem, which was the squeeze of two aluminium sheets by a pair of electrodes. The deformation and the fluid pressure of the adhesive were simulated by both the mixed model and the displacement model. Very similar results were obtained with both methods. This demonstrated that the mixed model is capable of solving complex cases as well as simple cases. For the fluid-solid problem encountered in chapter seven, the mixed model showed its convenience in handling the fluid pressure in the way that the pressure calculated from the fluid theory can be directly applied to the mixed model as a pressure boundary condition, instead of transforming the pressure into nodal forces in the displacement model.

8.2 MATHEMATICAL ASPECTS OF THE MIXED MODEL

The mixed model in elasticity yields a global matrix which is positive definite. Therefore a standard solver for a displacement model can be used to solve the system equation in this model. This is a advantage of this model over the other mixed models, which may have a non-positive-definite matrix or semi-positive-definite matrix.

However, the mixed model for rigid plasticity and elasto-plasticity mentioned in this thesis can have a non-positive-definite matrix when the iteration is far from the solution. A simple trick is used to handle this: allow a negative diagonal terms in *LTLD* solver during the decomposition of the matrix. All the results shown in this thesis were solved by such a treatment.

Another worry may arise from the discontinuity of the functional Π_{ep} at the elastic/plastic boundary, which can lead to an indefinite derivative of the functional Π_{ep}. However, numerically this will never happen. Since calculations of the derivative of Π_{ep} are always carried out at a specific Gauss point, which is either in elastic area or in plastic area, the value of the derivative are definite. This was verified in practical solutions.

With regard to the conditioning of the system equation, i.e. eqs.(3.41), (5.27) and (6.29), it is clear these equations were very badly conditioned, or ill-conditioned. This is because that the magnitude of the diagonals of the coefficient matrices $\mathbf{k}_{\mathbf{I}}$, $\mathbf{k}_{\mathbf{II}}$ and $\mathbf{k}_{\mathbf{III}}$ varied tremendously. For instance, in $\mathbf{k}_{\mathbf{I}}$ for elasticity, the magnitude of the first diagonal was about G^2 times larger than that of the second diagonal.

Such a problem was found when material properties were taken for a metal. However, this can be cured by redefining the general variable β_i as

$$\beta_i^T = \left[\beta_i^d \quad \frac{1}{2G}\beta_i^s \right]$$

The diagonals in $\mathbf{k}_{\mathbf{I}}$ are then in a similar magnitude. A same substitution was also carried out for the system equations for rigid plasticity and elasto-plasticity, which largely improved the results.

The nonlinear equations in rigid-plasticity and elasto-plasticity were also to be treated specially. Newton's method or modified Newton's method is not particularly

good at solving those equations. Therefore. the linear-search method was used in conjunction with the Newton's method in the way that Newton's method gives the search direction, and then linear-search method finds the line minium along this direction.

Such a method was used in obtaining all the results in rigid-plasticity and elasto-plasticity. The performance of it was satisfactory, although if the initial guess is not properly chosen the linear search might take a relatively long time to find a minimum.

8.3 SOME DRAWBACKS OF THE MIXED MODEL

With the above merits, the mixed model also has some shortcomings. Like most FE models by mixed formulations, the enormous computing time for a solution is the most serious problems with the mixed model. In particular, the mixed model in this thesis belongs to the continuous mixed model, where more degrees of freedom per node must be employed. As a result, the system equation is much larger than the corresponding displacement model.

In addition, the mixed model for elasto-plasticity in this thesis also suffers another problem with the computing time, which comes from the line search process. Firstly at each iteration, a direction is found by solving eq.(6.29) by Newton's method. Then a line search along this direction is performed to find a line minium of the functional. Surprisingly, this stage takes even more computing time than the solution of the linearized system equation (5.27) or eq.(6.29), which is supposed to take most computing time in the solution. Therefore a better algorithm is desired to find the minimum of the functional after Newton's method.

Another weakness of the mixed model is that for some kinds of problems, this model is not robust enough. This is because that the results may depend on a couple of penalty numbers at some degree. For example, in a bending problem, the penalty ρ_c can affect the stress and displacement results largely. Although for most problems the penalty number is not difficult to choose, the value of it is not determined by the system itself, but by a user, which largely depends on his or her experience.

For elasto-plastic problems, it is also required that penalty number ρ, for imposing the yield condition, is to be specified. Generally speaking, the more plastic strain, the

bigger value of it must be used. The big ρ tends to give more accurate local stresses and displacement results, and it only has very small effects on the results at the global level. Therefore this will not cause problems in reality.

The mixed model is also found to give inaccurate shear stress near a corner, especially when one face of the corner is free and another one is subjected to a considerable shear strain. For example at the corner of the built-in end of a beam and at the corner formed by the electrode wall and the top surface of the aluminium sheet, the mixed model fails to predict the shear stress correctly. The difficulties arises from imposement of the moment equilibrium equation (3.3): at the free surface, the shear stress must be zero, while at another face it has certain values. Only right at the corner node, shear stresses at each face is equal to each another. However in this mixed model, the force boundary condition is imposed in an average tense within an element, instead of a node. Therefore eq.(3.3) will never be exactly satisfied in reality at the boundary. As a result, this leads to a lower value of shear stress at the fixed surface and in the near area.

In order to solve this problem, sufficiently small elements are required at those corners, or only the stress some distance away from the corner can be used.

8.4 CONCLUSIONS AND FURTHER WORK

After three years work on this mixed model, we would like to draw the following conclusions:

(a) The Love's stress function for axisymmetric problems can decomposed into three first-order stress functions with C^0 continuity respectively, with which force equilibrium is satisfied, in conjunction with a constraint to impose the moment equilibrium.

(b) The resulting mixed FE model with above stress discretization is a continuous mixed model, which can reproduce the traction reciprocity as well as the stress discontinuity correctly.

(c) The mixed model for elasto-plasticity was established by a functional for rigid-plasticity and a functional for elasticity, where elasto-plastic strain was introduced implicitly within a plastic strain to the system. The examples prove that this mixed model is correct. The convergence of the solution by this model is rapid.

(d) The value of the functional at any point in a body can be used to indicated the error of the solution by the mixed model at that point. The bigger it is, the more error the solution is.

It is also desired that some further work be done in order to improve the efficiency and the robustness. Firstly, the way to impose the stress boundary needs to be updated, probably by using more accurate integration methods. This might improve the capability of the model to give the better stress results near a corner.

Secondly, a better solution method for the elasto-plastic problems to replace the current line search method by Brent' method is to be used, which should take much less time to find a line minium of the function after solving the linearized equation by the Newton's method. Alternatively, we may choose another solution method to replace the Newton's method, and eventually discard the line-search process.

Thirdly, the hardening treatment in the elasto-plastic model needs to be extended to include the kinematic hardening, which enables the mixed model to handle cycling loads and predict residual stresses.

APPENDIX A

Comparison of Differential Methods in the Formation of Stiffness Matrices

In order to compare the efficiency and the accuracy of the numerical differential method and the analytical methods described in chapter five, a practical differentiation of the functional Π_{rgd} are presented below. Say the deviatoric stress and strain are taken arbitrarily as

s = [1.0, 0.0, 0.0, 0.0, 0.3]; e = [1.0, 0.0, 0.0, 0.0, 0.3]; k=100

Substituting above data into eq.(5.18), the differentials from analytical method are

$$L_{ss} = \begin{bmatrix} 0.3005E+03 & 0.0000E+00 & 0.0000E+00 & 0.3590E+03 & 0.1105E+03 \\ 0.0000E+00 & 0.7700E+03 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & 0.7700E+03 & 0.0000E+00 & 0.0000E+00 \\ 0.3590E+03 & 0.0000E+00 & 0.0000E+00 & 0.4955E+03 & -0.8447E+02 \\ 0.1105E+03 & 0.0000E+00 & 0.0000E+00 & -0.8447E+02 & 0.7440E+03 \end{bmatrix}$$

$$L_{se} = \begin{bmatrix} -0.3897E+00 & 0.0000E+00 & 0.0000E+00 & -0.4667E+00 & -0.1405E+00 \\ 0.0000E+00 & -0.1000E+01 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & -0.1000E+01 & 0.0000E+00 & 0.0000E+00 \\ -0.4667E+00 & 0.0000E+00 & 0.0000E+00 & -0.6431E+00 & 0.1098E+00 \\ -0.1436E+00 & 0.0000E+00 & 0.0000E+00 & 0.1098E+00 & -0.9662E+00 \end{bmatrix}$$

$$L_{ee} = \begin{bmatrix} -0.5227E-01 & 0.0000E+00 & 0.0000E+00 & -0.6364E-01 & -0.1958E-01 \\ 0.0000E+00 & -0.1355E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & -0.1355E+00 & 0.0000E+00 & 0.0000E+00 \\ -0.6364E-01 & 0.0000E+00 & 0.0000E+00 & -0.8683E-01 & 0.1497E-01 \\ -0.1958E-01 & 0.0000E+00 & 0.0000E+00 & 0.1497E-01 & -0.1309E+00 \end{bmatrix}$$

(A.1)

For the numerical differential method, an internal variable c, which is a small constant defined in eq.(5.13), has to be chosen. From the theoretical point of view, the smaller the c is, the better the approximation of the differentials can be obtained. However, in practical calculation, the round-off error and truncated error of computer have to be taken into account. It means that too small c is likely to bring computing error into the results. Therefore for a particular problem there is an optimum value of c.

In following examples, three values $c = 10^{-2}$, $c = 10^{-4}$, $c = 10^{-8}$ are used in the calculations and the results are listed below.

$$c = 10^{-2}$$

$$L_{ii} = \begin{bmatrix} 0.6925E+02 & 0.0000E+00 & 0.0000E+00 & 0.1064E+00 & 0.3269E-01 \\ 0.0000E+00 & 0.6930E+02 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & 0.6930E+02 & 0.0000E+00 & 0.0000E+00 \\ 0.1064E+00 & 0.0000E+00 & 0.0000E+00 & 0.6927E+02 & -0.2497E-01 \\ 0.3269E-01 & 0.0000E+00 & 0.0000E+00 & -0.2497E-01 & 0.6930E+02 \end{bmatrix}$$

$$L_{is} = \begin{bmatrix} -0.9425E+00 & 0.0000E+00 & 0.0000E+00 & -0.4395E-01 & -0.1352E-01 \\ 0.0000E+00 & -0.1000E+01 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & -0.1000E+01 & 0.0000E+00 & 0.0000E+00 \\ -0.4390E-01 & 0.0000E+00 & 0.0000E+00 & -0.9664E+00 & 0.1033E-01 \\ -0.1394E-01 & 0.0000E+00 & 0.0000E+00 & 0.1032E-01 & -0.9968E+00 \end{bmatrix}$$

$$L_{si} = \begin{bmatrix} -0.5229E-01 & 0.0000E+00 & 0.0000E+00 & -0.6363E-01 & -0.1958E-01 \\ 0.0000E+00 & -0.1355E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & -0.1355E+00 & 0.0000E+00 & 0.0000E+00 \\ -0.6363E-01 & 0.0000E+00 & 0.0000E+00 & -0.8684E-01 & 0.1497E-01 \\ -0.1958E-01 & 0.0000E+00 & 0.0000E+00 & 0.1497E-01 & -0.1309E+00 \end{bmatrix}$$

(A.2)

$$c = 10^{-4}$$

$$L_{ii} = \begin{bmatrix} 0.3075E+03 & 0.0000E+00 & 0.0000E+00 & 0.3567E+03 & 0.1087E+03 \\ 0.0000E+00 & 0.7615E+03 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & 0.7615E+03 & 0.0000E+00 & 0.0000E+00 \\ 0.3567E+03 & 0.0000E+00 & 0.0000E+00 & 0.4997E+03 & -0.8257E+02 \\ 0.1087E+03 & 0.0000E+00 & 0.0000E+00 & -0.8257E+02 & 0.7372E+03 \end{bmatrix}$$

$$L_{is} = \begin{bmatrix} -0.3910E+00 & 0.0000E+00 & 0.0000E+00 & -0.4657E+00 & -0.1433E+00 \\ 0.0000E+00 & -0.1000E+01 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & -0.1000E+01 & 0.0000E+00 & 0.0000E+00 \\ -0.4650E+00 & 0.0000E+00 & 0.0000E+00 & -0.6444E+00 & 0.1094E+00 \\ -0.1428E+00 & 0.0000E+00 & 0.0000E+00 & 0.1092E+00 & -0.9664E+00 \end{bmatrix}$$

$$L_{si} = \begin{bmatrix} -0.5227E-01 & 0.0000E+00 & 0.0000E+00 & -0.6364E-01 & -0.1958E-01 \\ 0.0000E+00 & -0.1355E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & -0.1355E+00 & 0.0000E+00 & 0.0000E+00 \\ -0.6364E-01 & 0.0000E+00 & 0.0000E+00 & -0.8683E-01 & 0.1497E-01 \\ -0.1958E-01 & 0.0000E+00 & 0.0000E+00 & 0.1497E-01 & -0.1309E+00 \end{bmatrix}$$

(A.3)

$$c = 10^{-8}$$

$$L_{dd} = \begin{bmatrix} 0.2998E+03 & 0.0000E+00 & 0.0000E+00 & 0.3619E+03 & 0.1132E+03 \\ 0.0000E+00 & 0.7683E+03 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & 0.7683E+03 & 0.0000E+00 & 0.0000E+00 \\ 0.3619E+03 & 0.0000E+00 & 0.0000E+00 & 0.4907E+03 & -0.8438E+02 \\ 0.1132E+03 & 0.0000E+00 & 0.0000E+00 & -0.8438E+02 & 0.7416E+03 \end{bmatrix}$$

$$L_{ds} = \begin{bmatrix} -0.2220E+01 & 0.0000E+00 & 0.0000E+00 & 0.2220E+01 & 0.2220E+01 \\ 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ -0.4441E+01 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & -0.2220E+01 \\ 0.2220E+01 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \end{bmatrix}$$

$$L_{ss} = \begin{bmatrix} 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & -0.4441E+01 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & 0.0000E+00 \\ -0.4441E+01 & 0.0000E+00 & 0.0000E+00 & -0.2220E+01 & -0.6661E+01 \\ 0.0000E+00 & 0.0000E+00 & 0.0000E+00 & -0.6661E+01 & 0.0000E+00 \end{bmatrix}$$

(A.4)

It is seen in eq.(A.2) to eq.(A.4) that the results of the numerical algebraic are largely dependent on the value of c. When $c = 10^{-2}$, it gives good results of L_{ss}, but bad ones of L_{dd} and L_{ds}, while when $c = 10^{-8}$, it gives excellent results of L_{dd}, but the results of L_{ds} and L_{ss} are disastrous in comparison with the analytical one in eq.(A.1). Only the choice of $c = 10^{-4}$ enable the numerical one to yield good results for all the differentials. Therefore the best choice of c for these specific $\mathbf{s}$ and $\dot{\mathbf{e}}$ is 10^{-4}. The difficulty is that we do not know which value of c should be the best choice for every deviatoric stress $\mathbf{s}$ and strain rate $\dot{\mathbf{e}}$, while there is no such a problem at all for analytical arithmetic. Let us look at the computing time needed for both the algebraic on SUN sparc workstation.

Table 1 Computing time of two differential algebraic

computing time(seconds)	1,000 times	10,000 times
analytical arithmetics	1.198 seconds	12.00 seconds
numerical arithmetics	38.77 seconds	387.90 seconds

do same numbers of differentials. Together with the higher accuracy and numerical stability, the analytical algebraic obviously proves to be the better approach for the functional Π_{rgd}.

The similar conclusion exists in the differentiations of functional $\Pi_{ep\text{-}p}$ for elasto-plasticity in chapter six.

APPENDIX B
Line-Search Method

B.1 Golden-Section Search Method

Golden section search[122] is a well-known method among one-dimensional optimisation procedures. It requires fewest conditions of a functional Π to perform the optimizing, that is the value of the functional is finite and predictable, and a proper bracket in which a minimum exists is given. No continuity condition of Π is needed. The last merit of this approach is very attractive to us since in chapter six the functional Π_{III} for elasto-plasticity will be no longer continuous.

A minimum of a function $f(x)$ is known to be bracketed only when there is a triplet of points, $a<b<c$, such that f(b) is less than both f(a) and f(c). In this case we know that the function (if it is not singular) has a minimum in the interval (a, c).

Choose a new point x, either between a and b or between b and c. Suppose, to be specific, that we make the latter choice. Then we evaluate f(x). If f(b) < f(x), then the new bracketing triplet of points is $a<b<x$; Contrariwise, if f(b) > f(x), then the new bracketing triplet of points is $b<x<c$. In all cases the middle point of the new triplet is the abscissa whose ordinate is the best minimum so far. We continue the process of bracketing until the distance between the two outer points of the triplet is tolerably small.

It remains to decide on a strategy for choosing the new point x for given a, b, c. In ref[122], it is suggested that the optimal bracketing interval $a<b<c$ has its middle point b a fractional distance 0.38197 from one end (say, a), and 0.61803 from the other end (say b). These fractions are those of the so-called *golden mean* or *golden section*. This optimal method of function minimization is thus called the golden section search method.

B.2 Brent's Method

A golden section search is designed to handle the worst possible case of function

minimization. If the function is nicely parabolic near to the minimum——surely the generic case for sufficiently smooth functions——then the parabola fitted through any three points should take us a single leap to the minimum, or at least very near to it, as seen in fig.B.1.

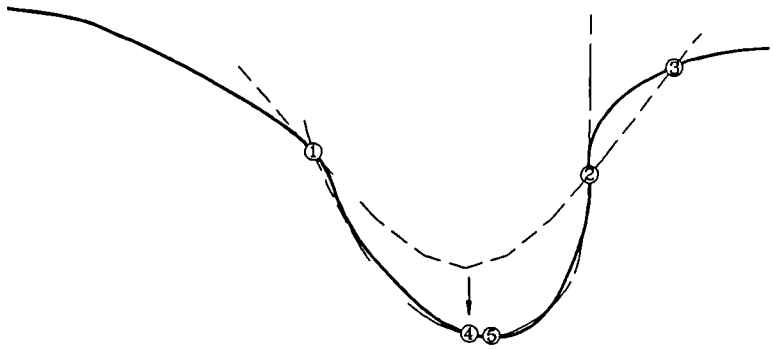


Fig.B.1 The schematic configuration of golden-section search.

Since we want to find an abscissa rather than an ordinate, the procedure is technically called *inverse parabolic interpolation.* The formula for the abscissa x which is the minimum of a parabola through three points f(a), f(b), and f(c) is

$$x = b + \frac{1}{2}\frac{(b-a)^2[\mathrm{f}(b)-\mathrm{f}(c)]-(b-c)^2[\mathrm{f}(b)-\mathrm{f}(a)]}{(b-a)[\mathrm{f}(b)-\mathrm{f}(c)]-(b-c)[\mathrm{f}(b)-\mathrm{f}(a)]} \tag{B.1}$$

This formula fails only if the three points are collinear, in which case the denominator is zero (minimum of the parabola is definitely far away). Note, however, that eq.(5.35) can jump to a parabolic maximum as well. No minimization scheme that depends solely on eq.(B.1) is likely to succeed in practice.

A marvellous scheme is to rely on a sure-but-slow technique like golden-section search when a function is not cooperative, but switch over to eq.(B.1) when the function allows. However a few difficulties need to be overcome[122]

a) The housekeeping needed to avoid unnecessary function evaluations in switching between the two methods can be complicated;
b) Careful attention must be given to the "end game", where the function is being evaluated very near to the round-off limit of eq.(B.1);
c) The scheme for detecting a cooperative versus non-cooperative function must be very robust.

Brent's method[122] (1973) is a excellent scheme to achieve such a task. It is keeping track of six function points, a, b, u, v, w, and x, not all distinct. Initially (a, b) is the interval on which f is defined, and

$$v = w = x = a + (\frac{3-\sqrt{5}}{2})(b-a)$$

the magic number $\frac{3-\sqrt{5}}{2} = 0.381966...$ is rather arbitrarily chosen so that the first step is the same as for a golden-section search.

At the start of a cycle the points a, b, u, v, w, and x always serve as follows: a local minimum lies in $[a, b]$; of all the points at which f has been evaluated, x is the one with the least value of f, or the point of the most recent evaluation if there is a tie; w is the point with the next lowest value of f; v is the previous value of w, and u is the last point of which f has been evaluated.

Let $m = \frac{1}{2}(a+b)$ be the mid-point of the interval known to contain the minimum. If $|x-m| \leq 2\,\mathrm{tol} - \frac{1}{2}(b-a)$, then the procedure terminated with x as the approximate position of the minimum. Otherwise, a inverse parabolic interpolation will be used on the three point $(v, f(v))$, $(w, f(w))$ and $(x, f(x))$. Modified slightly to avoid the collinear case of v, w, x, eq.(B.1) now is presented by p and q,

$$x' = x + \frac{1}{2}p/q \tag{B.2}$$

where

$$p = (x-v)^2[f(x) - f(w)] - (x-w)^2[f(x) - f(v)]$$
$$q = (x-v)[f(x) - f(w)] - (x-w)[f(x) - f(v)]$$

If two or more of three points coincide, or the parabola degenerates to a straight line, then $q = 0$, the program will automatically turns to perform golden section search. Normally the correction p/q should be small if x is close to a minimum. Let e be the value of p/q at the second last cycle. If $|e| \leq \mathrm{Tol}$ or $|p/q| \geq 1/2|e|$, then a golden section search step is performed.

APPENDIX C

Translation between results from von Mises and Tresca

In the analytical solutions it is quite normal to use Tresca's yield condition for a elasto-plastic or rigid plastic problem. On the other hand, von Mises's yield condition is proven to be the better one in the most situations and is extensively used in numerical solutions, such as in finite element method. In order to compare the results obtained from these methods, a proper interpretation is necessary.

In most situations, the direct relation between these two yield conditions is not clear. However, if the problem concerned is in plane strain condition, a simple relation does exist.

Let three principal stresses be σ_1, σ_2 and σ_3. From Hooke's law, we have

$$\varepsilon_{zz} = \frac{1}{E}[\sigma_{zz} - \nu(\sigma_{xx} + \sigma_{yy})] \tag{C.1}$$

Plane strain condition requires that $\varepsilon_{zz} = 0$, which leads to

$$\sigma_{zz} - \nu(\sigma_{xx} + \sigma_{yy}) = 0 \tag{C.2}$$

or

$$\sigma_{zz} = \nu(\sigma_{xx} + \sigma_{yy})$$

and $\tau_{xz} = \tau_{yz} = 0$, which means that σ_{zz} is one of the principal stresses, say σ_3. It is known that in plane problems $(\sigma_{xx} + \sigma_{yy})$ is equal to the sum of the rest two principal stress, i.e. $(\sigma_1 + \sigma_2)$[110]. Remembering that in a plastic region Poisson's ratio $\nu = 0.5$, eq.(C.2) is then equivalent to

$$\sigma_3 = \frac{1}{2}(\sigma_1 + \sigma_2) \tag{C.3}$$

On the other hand, we have the second deviatoric stress invariant J_2' expressed as

$$J_2' = \frac{1}{6}[(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2] \tag{C.4}$$

Substituting eq.(C.3) into eq.(C.4), it follows

$$J_2' = \frac{1}{4}(\sigma_1 - \sigma_2)^2 \tag{C.5}$$

von Mises yield condition can be expressed in terms of J_2' as

$$\sqrt{3J_2'} = \sigma_Y \tag{C.6}$$

With expression of J_2' in eq.(C.5), it is straightforward to describe von Mises yield condition in terms of two principal stresses as

$$\frac{\sqrt{3}}{2}(\sigma_1 - \sigma_2) = \sigma_Y \qquad \text{or} \qquad \sigma_1 - \sigma_2 = \frac{2}{\sqrt{3}}\sigma_Y \tag{C.7}$$

On the other hand, Tresca yield condition can also be described by those principal stresses as

$$\sigma_1 - \sigma_2 = \sigma_Y \equiv 2K \tag{C.8}$$

The only difference between eq.(C.7) and eq.(C.8) is that yield stress is expressed differently: in von Mises condition, it is $\sqrt{3}/2\,\sigma_Y$ while in Tresca condition, it is σ_Y.

Above arguments enable us to translate results solved in Tresca condition to those in von Mises condition. The process is very simple: replace $\sqrt{3}/2\,\sigma_Y$ for σ_Y, then solve it in Tresca condition. The results obtained is equivalent to those solved in von Mises condition.

For 3D problems, such a simple translation is not normally available except for the uniformly expansion of a sphere where σ_{rr}, σ_{zz} and $\sigma_{\theta\theta}$ are naturally three principal stresses σ_1, σ_2 and σ_3. They are not independent but

$$\sigma_2 = \sigma_3 \tag{C.9}$$

Substituting eq.(C.9) into J_2' expression in eq.(C.4) the von Mises yield condition turns out to be identical to eq.(C.8), which is Tresca yield condition. Therefore, the conclusion is that for uniformly expansion of a sphere elasto-plastic solution under von Mises yield condition yields identical results to those under Tresca yield condition.

APPENDIX D

Thermo-Mechanical Simulation of the Spot-Welding Process

D.1 REVIEW OF THE STUDIES IN SPOT-WELDING

D.1.1 Experimental and Numerical studies of the Thermal Problem

A number of experimental efforts have been made on gaining greater insight into nugget formation by monitoring the temperatures involved in and around the weld. Bentley et al[7], Lee and Nagel[57], Cho and Cho[22], and Sheppard[100] related metallurgical changes in a number of semicompleted welds to the peak temperatures seen at locations in the weld region at various time in the weld cycle. Lee and Nagel[57] as well as Kim and Eager[50,52] made a number of high speed films of half welds painted with thermo-sensitive paint in order to measure the movement of isotherms during welding. This technique vividly shows temperature patterns. Also, Kim and Eager[50,52] used infrared monitoring of the surface of both the electrode and sheet. More recently Han[37] used a slotted weld specimen that allowed for a thermal couple to be placed in *heat effective zone(HAZ)*. Alcini[2,3] used a half-weld technique which uses multiple bendless microthermal couples for temperature measurements.

These empirical studies are able to provide valuable temperature data. However the experimental techniques do have their limitations. Unsealed half welds lack the ability to retain molten metal along the exposed surface, and therefore cease to emulate a full weld after welding has initiated; surface techniques do not allow for faying surface or internal temperature to be monitored; thermocouple data gives temperatures only at selected points. Also, the metallurgical technique uses phase changes to flag the passage of a particular temperature, but is not able to give a description of the overall temperature field. Finally, several of the techniques require expensive equipment. These shortcomings make computer simulation an attractive tool for complementing experimental temperature studies[101].

Rice and Funk[95] developed a one-dimensional finite difference simulation with which these researchers got the functional relationship between electrical contact

resistance and temperature when two thin sheets are welded together. A combined analytical-experimental approach was used in developing this relationship. Kaiser et al[49] used the same approach as ref.[95] for estimating the contact resistance for the one-dimensional model. Houchens, Page and Yang[43] developed a one-dimensional finite difference method to study electrode temperatures in the hope of improving electrode life. In 1987, Gould[35] presented a combined experimental and analytical study on nugget formation where a one dimensional finite difference formulation was used again. Although information on the nugget depth could be extracted from the one-dimensional analysis, the effect of the diameter which is important for the mechanical strength of the weld could not be deduced. A two-dimensional finite difference simulation of spot welding was developed by Greenwood[36]. It assumed: there is no electrical contact resistance; material properties are constant with temperature; and no heat is lost from free surface. Houchens et al[43] later extended the same approach to include the effect of temperature dependant properties. In 1984, a two-dimensional finite element method was used by Nied[75] to compute the temperature distribution from an electro-thermal simulation.

More recently Cho and Cho[22], Han et al[37], Tsai et al[114], Reddy and Sharma[91], and Kim[51] presented numerical studies of the thermal cycling in resistance spot welding of sheet steel. Works [22], [37] and [91] were based on the finite difference formulation, while [114] and [51] used the finite element method. All studies assumed that the size of the contact area at the faying surface remain constant throughout thermal cycling and the truncated copper electrode were selected which means no change in the contact area of the electrode-sheet surface occurs.

D.1.2 Thermo-Mechanical Simulation by FE Method

In the work mentioned above, it is evident that the thermo-mechanical coupling of the resistance spot welding process was inadequately explored. Most of the mathematical models were devoted to analyzing the thermal behaviour of the process under different sets of parameters, while neglecting the major role of the mechanical and thermal stresses involved in the process.

In 1984, Nied[75] used a finite element package known as ANSYS to introduce an

axisymmetric model which included the geometry of electrode and workpiece, and accounted for temperature dependant thermal properties, melting and Joule heating. Predictions of electrode and sheets deformations were illustrated and stress distribution along the interfaces were also obtained. It solved the thermo-mechanical problem in a coupled manner but did not change the contact area in the faying surface, and all the mechanical solution was based on elasticity. Tsai[114] used a similar method in studying nugget geometry.

In 1990, Kim et al[53] used a coupled electro-thermo-mechanical finite element method to investigate the fundamental parameters controlling the weld quality. It was a very successful model where elasto-plasticity and changes of the contact area were accounted for. Unfortunately, it can only be used for the truncated electrode. For the curved electrode, the numerical approach would be much more complicated.

So far, although the studies mentioned before involved both electro-thermal and thermo-mechanical aspects, the main purpose was to study the thermal phenomenon rather than the mechanical one, and mechanical simulation is used as a tool to get the pressure and contact area for thermal studies.

Where good fatigue performance[101] is a requirement it is important to be able to estimate any residual stresses. Residual stresses are introduced into a part whenever nonuniform heating and cooling result in plastic deformation. These stresses are often difficult to measure and their presence can largely influence fatigue performance. For example, Lawrence, Corten and McMahon[56] experimented with post-weld coining of spot welded joints which introduces compressive residual stresses in the HAZ and found life improvement of joints when compared with joints in the as-welded condition. Measured values of residual stress in the weld region fell from 92% of the tensile yield strength in the as-welded condition to 74% of the compressive yield strength in the coined condition. Widmann[121] found that tensile-shear plug weld specimens that had been heat treated subsequent to welding shown improved fatigue life. Concluding that the presence of residual stress partially (if not totally) results in this change in fatigue performance is supported by Bolton[9] who found that post weld heat treatment could reduce the residual stresses in spot welding by as much as 50%. Schoepfel[99] found that water quenching the *HAZ* of a tensile-shear plug weld immediately after weld termination also improved fatigue performance by more than doubling the life.

However the residual stress studies of spot welding by numerical methods are rarely found in the public literature.

D.1.3 Researches in Spot-Welding Group

The **Spot Welding in Aluminium group** in the Newcastle University has devoted itself to numerical simulation of both thermal and mechanical behaviour during spot welding and is supported by **Alcan International Ltd.** The difference of the current studies from others exists in that **Finite Difference Method(FDM)** is used for the electro-thermal solution, while **Finite Element Method(FEM)** for the thermo-mechanical response. Therefore advantages of finite difference method in electro-thermal solution and finite element method in thermo-mechanical stress studies can be fully utilised. In addition, because we used the curved electrode in the FEM simulation, it allows us to use the electrode curvature as a additional control parameter in the spot welding process.

In order to handle this electrical, thermal and mechanical coupling problem efficiently, the research work was divided into two parts;

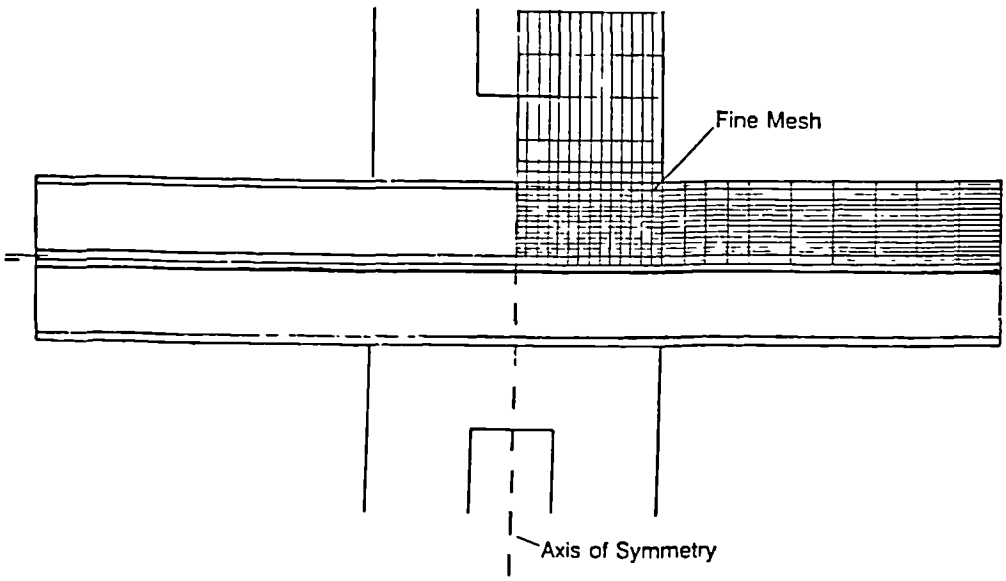


Fig. D.1 Finite difference model for the electric-thermal simulation of the spot-welding[11].

a. Electric-thermal simulation

This work has been done by Browne[13,14] where a axisymmetric finite difference model has been developed and a large number of results have been obtained concerned with the effect of thermal conductivity, contact resistance and water cooling position on the nugget formation. The electric current, its duration, material properties of electrode and sheets, and the geometry must be supplied for this model. The main output is the temperature distribution in the sheets and electrodes during the welding process. The finite different mesh used in this study is shown in Fig.D.1[13]

b. Thermo-mechanical simulation

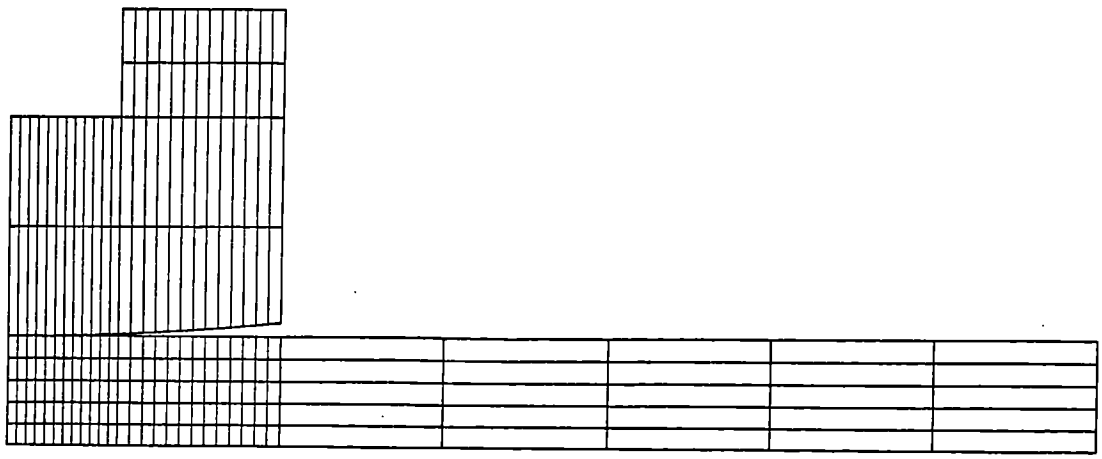


Fig. D.2 Finite element model for the thermo-mechanical simulation of the spot-welding.

An axisymmetric elasto-plastic finite element model was developed for the thermo-mechanical simulation. The contact area and pressure of electrode-sheet and faying surfaces at any stage of welding can be obtained by giving the temperature dependant material properties from DATA FILE and the temperature distribution data from the FDM solution. The finite element model is shown in Fig.D.2, where only one quarter of the whole geometry is needed for the model because of the symmetry conditions of geometry and loading.

These two programs are combined in one integrated package. An interface was designed to transmit the required data between these two program. Eventually the package will output the temperature field, the deformation of electrode and sheet, the stress in electrode and sheet in hot condition and the residual stress of sheets after terminating the welding for various cooling processes.

The source code for the finite difference model(FDM) was written by Browne for a *Pascal* compiler operating in UNIX system on SUN Spark station. The program for the FE model was written by author for a *Fortran* compiler also operating in UNIX on SUN Spark station. The integrated program runs in the manner in which FDM uses FEM as a single procedure.

This appendix will only be concerned with the finite element simulation of the thermo-mechanical problem.

c. Simulation of the adhesive between aluminium sheets

The interest of simulating the adhesive between aluminium sheets arises from the practical welding. It is found in some welds, there are remains of the adhesive at the central area of the faying surface. This may lead to a poor weld due to lack direct contact between two aluminium sheets.

As far as the mathematics is concerned, there should always be an **entrapment** of fluid when a viscous fluid is pressed by a flat disc[74]. In other words, it is impossible to entirely rule out the entrapment theoretically.

Our study is aimed at providing some ideas about how to control the size of entrapment such that it will not effect welding in engineering practise. A **Fluid-FE model** is proposed to analyse the deformation of the adhesive under squeeze force.

D.2 FE MODELLING OF SPOT WELDING

Simulations of the spot welding process can be classified as a 3D thermal-elasto-plastic, large-strain problem. To simplify this problem, an 2D axisymmetric elasto-plastic, small-strain finite element model was developed for the cold condition and later developed to include the effect of temperature on the yield strength and the thermal

strain.

In the case of sheets with equal thickness and equal electrode geometry, only one quarter of the model has to be constructed due to geometric symmetry, as shown in fig.D.2. 4-noded isoparameteric elements were used to represent the electrode and the aluminium sheet.

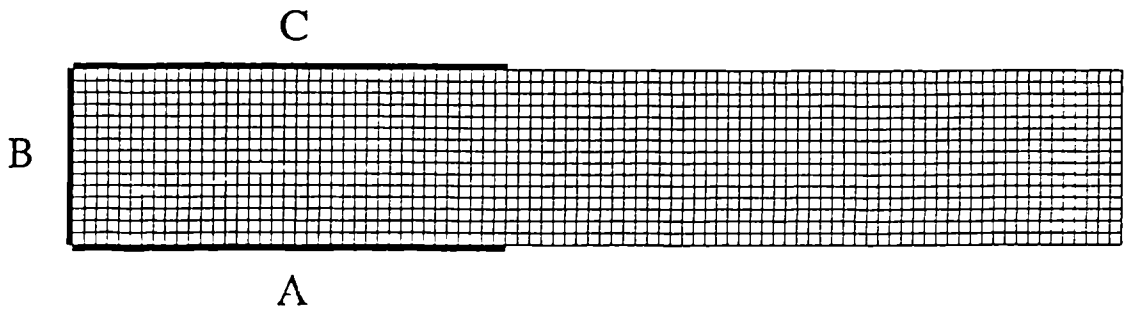


Fig. D.3 The boundaries in the FE mesh representing the aluminium sheet.

In order to discuss the boundary condition, the FE mesh for aluminium sheet is presented in fig.D.3. On boundary **B** the radial displacements are all zero while the vertical ones are free. On boundary **A**, the faying surface, the vertical displacements can only be positive or zero because the sheet-sheet surface can part but can not overlap, and no tensile traction exists in vertical direction. The boundary conditions on C are more complicated and will be discussed in next section in detail.

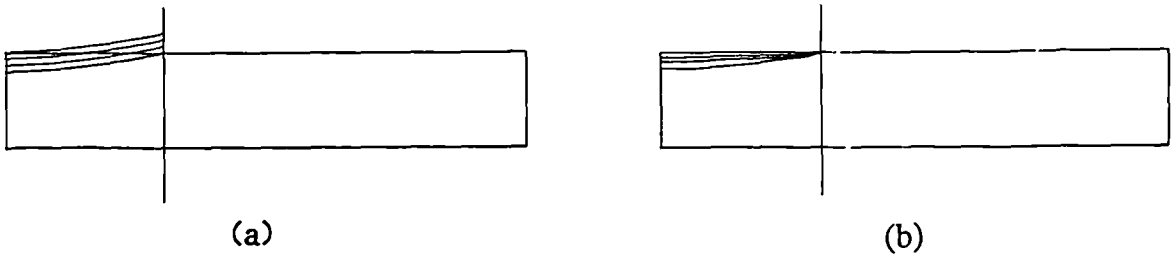
The material model used for representing the elasto-plastic problem is Von Mises linear isotropic hardening model. Modified Newton's method was used to solve the nonlinear system equation sets for this elasto-plasticity.

Because the curved electrode was used in the simulation, the modifications concerned with the constraints on boundary C must be made to handle it as *contact problem*. Two treatments used in this study will be presented below.

D.2.1 Simple Contact Treatment at Surface A and C

The assumptions used in this simple model are, a) electrodes are rigid enough that there is no deformation occurred; b) as long as the final configuration of sheet coincides with the electrode, the actual history of deformation in the contact area is not important since

there is no unloading occurred in the sheets when they are cold.

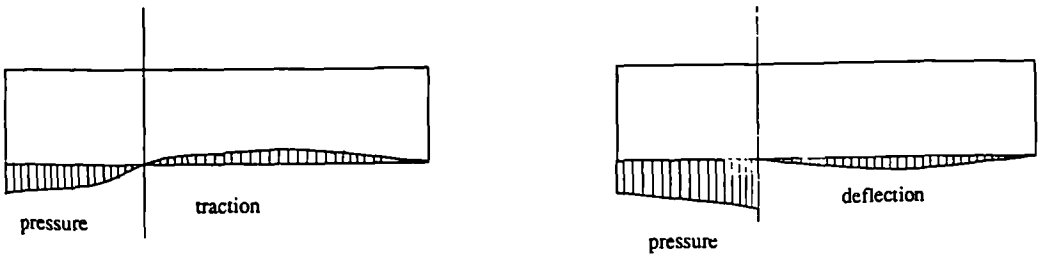


(a) (b)

Fig.D.4 The actual contact pattern and the one used in the FE model with simple contact treatments.

Obviously, with above assumptions the contact problem can be solved by specifying the displacement loading on the sheet with the same shape of the electrode by a usual FEM. No electrode appears in the FE mesh. When solved by an incremental method, the final displacement load was divided into a number of steps as shown in Fig.D.4(b). Clearly the actual process should be like the case in Fig.D.4(a). We think the differences between the cases in Fig.D.4(a) and Fig.D.4(b) are small due to the assumption b).

The boundary conditions on the faying surface must be specified carefully. Firstly, it is assumed that all nodes on the faying surface are contact, which means these nodes are constrained. Then the reaction forces of the nodes on the faying surface can be obtained. Since it is not true for every node to be contact, some tensile force will be found on some nodes, which should be released by allowing them to go free as in Fig.D.5(a).



(a) (b)

Fig.D.5 Handling the boundary conditions at the faying surface.

Secondly, after we adjust the contact nodes during the first stage, the model is solved

and then the contact condition will be checked again. However at this time both reaction forces and deflection of the nodes at the faying surface need to be used. When all the reaction forces are compressive (this is possible when contact area is not large enough), the deflections of the nodes should be checked. If any node cross the faying surface is found, as shown in Fig.D.5(b), additional constraints should be imposed.

The above steps will form small iterations, and final solution will be reached when there are both no tensile force and no negative displacement on the faying surface. After the solution, the contact area can be obtained by counting the number of contacting nodes at the faying surface. The reaction forces of the nodes at the faying surface were added together and formed inversely the external loading required on the electrode to achieve such contact situation.

There are two models used to represent the situation of the interface between the electrode and the sheet: *Slip model* and *No slip model*

- Slip There is no friction existed on the electrode-sheet surface. The displacement in radial direction is totally free.
- No slip The friction on the electrode-sheet surface is so large that there exists no relative movement on such surfaces.

The main advantage of this contact model is that any conventional finite element program can be used to solve the contact problem in the spot welding. However the shortcomings of it are obvious. Usually, we give the load by forces rather than displacements. When you first give the displacement loading, you can calculate the external loads from the reaction forces, which is often not equal to the force you intend to apply. Thus you have to adjust the displacement loading again and again to meet the actual load condition. Furthermore, this method cannot give the deformation and stress state of the electrode, and the rigid electrode assumption is often not correct.

D.2.2 Full Contact Treatment

Consider two bodies **A** and **B**. We discretize problems into finite element models by the standard method, except for the contact surface where pairs of nodes are designated

as candidates for contact as shown in Fig.D.6. The assumptions are as follows:

- displacements and strains are small,
- the contact surface is continuous line or curve,
- the frictional force acting at the contact surface follows the Coulomb type criterion for friction.

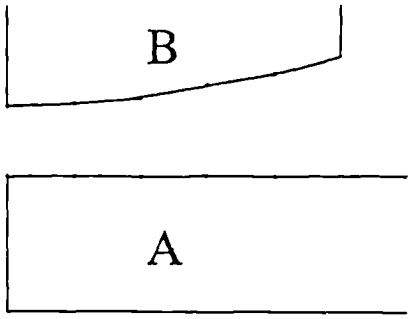


Fig.D.6 Two contact bodies A and B

Let q_{ji} and R_{ji} be the nodal displacement and nodal force at the contact surface, where the subscript j indicates the body identification and i the direction. The displacement and force increments must satisfying the following equilibrium equations and continuity conditions for different contact conditions, which can be expressed as the following mathematical formulations:

1. Open condition: gap remains or becomes open

$$\begin{aligned} &\Delta R_{Ai} = \Delta R_{Bi} \equiv \Delta R_p \leq 0 \\ &\Delta q_{Ai} - \Delta q_{Bi} \equiv \Delta l_i \qquad (i=x,y) \end{aligned}$$

2. Stick condition: gap remains closed and no slipping occurs,

$$\begin{aligned} &\Delta R_{Ai} = -\Delta R_{Bi} \equiv \Delta R_p > 0 \\ &\Delta q_{Ai} - \Delta q_{Bi} + \delta_i = 0 \qquad (i=x,y) \end{aligned}$$

3. Slipping condition: gap remains closed and **slipping occurs.**

$$\begin{aligned} &\Delta R_{Ai} = -\Delta R_{Bi} \equiv \Delta R_y > 0 \\ &\Delta q_{Ai} - \Delta q_{Bi} + \delta_x = 0 \end{aligned} \quad ; \quad \begin{aligned} &\Delta R_{Ax} = -\Delta R_{Bx} = \pm\mu\Delta R_y \\ &\Delta q_{Ax} - \Delta q_{By} + \delta_y = 0 \end{aligned}$$

(when slipping is in the x-direction)

Here, δ_i is the initial relative displacement which is zero except for the initial state, Δl_i

is the relative displacement between the contact node pairs. $\Delta R_i \geq 0$ means that body A and body B are pushing each other while $\Delta R_i < 0$ means they are pulling each other.

Above contact conditions were imposed on a normal FE program by means of *quadratic penalty function*. This leads to a program which can treat spot welding as a real contact process, as shown in fig.D.4(a). At the moment, the program used in this study cannot deal with Coulomb friction yet.

The finite element mesh with full contact treatment is shown as Fig.D.2. The loads are applied incrementally by distributing pressure on the upper end of electrode. Very small loading steps were chosen for the requirement of contact simulation.

In addition to a simple curve shaped, an electrode with any profile can be simulated by this program and the thermo-mechanical behaviour of the electrode can be obtained at the same time.

D.3 TEMPERATURE EFFECT ON THE SOLUTION

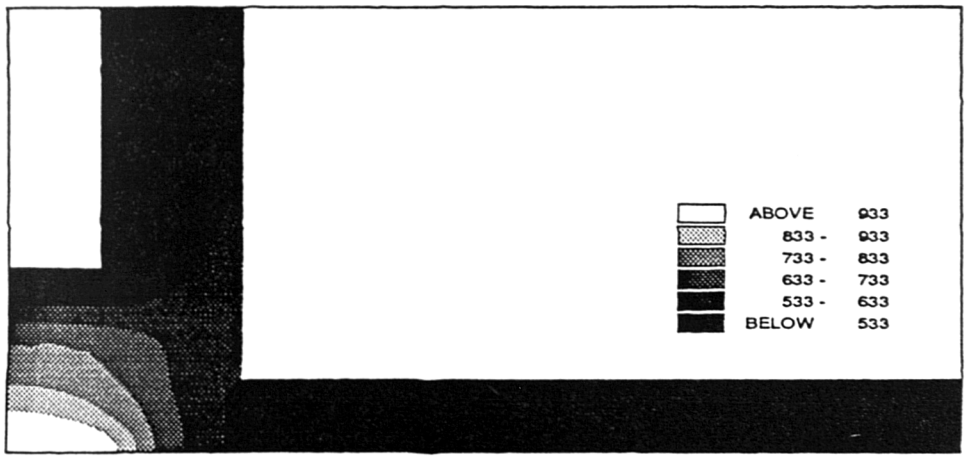


Fig.D.7 Typical temperature distribution of the sheet and the electrode during spot-welding process[12]

In the previous discussion there is no temperature mentioned. The real spot welding occurs inevitably at high temperatures. Thus temperature effects are crucial factors in welding. The typical temperature distribution used in this report is shown as Fig.D.7

in [ref.14].

The highest temperature is near the centre of the faying surface, which may be higher than 1000° K and the temperature gradient is significant over the whole area of sheets and electrodes.

For the numerical simulation used in this study, the temperature effects were accounted for by two aspects,

D.3.1 Temperature dependant material properties

The material properties concerned with thermo-elasto-plastic are:

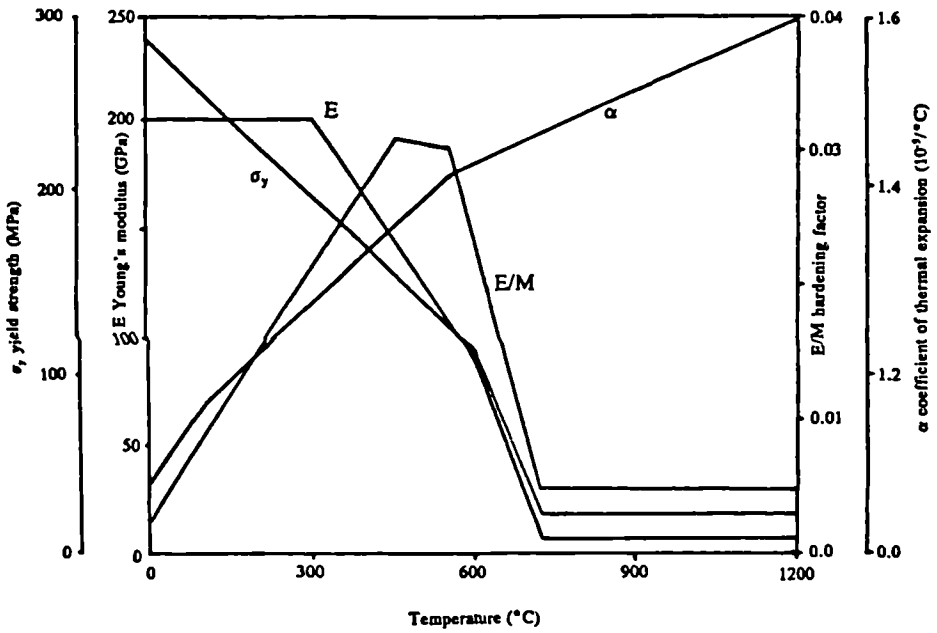
E: Young's modules

ν: Poisson's ratio

H/M: Hardening modules

σy: Yield stress

α: Coefficient of linear thermal expansion



D.8 The temperature dependant properties of low carbon steel.

Except Poisson's ratio that is thought to be independent of temperature, all other properties are functions of the temperature. For low carbon steel the functions are

shown by Fig.D.8[53].

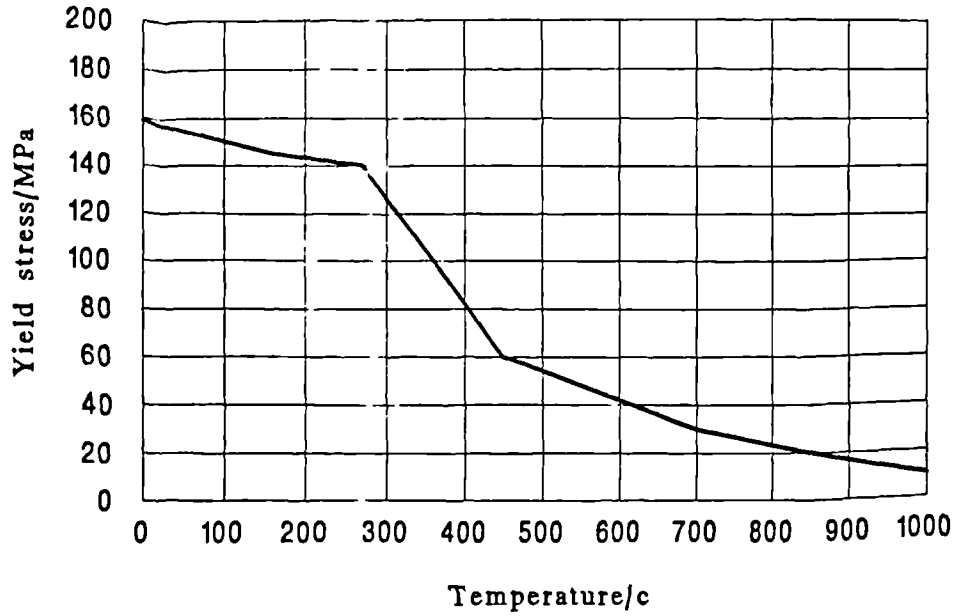


Fig.D.9 The temperature dependant yield stress of aluminium.

About aluminium, unfortunately, we have not got so much information. We used only temperature dependant yield stress in the program, which is thought to affect greatest on plastic strain and residual stress. The data was supplied by Alcan ltd and is shown as Fig.D.9.

D.3.2 Thermal strain effects

The thermal strain is usually interpreted as "initial strain". The Hooke's law can be written in elasticity by[81]

$$\sigma^* = D(\epsilon - \epsilon^0) = \sigma - \sigma^0$$

to handle the thermal problem.

When the incremental method is used to solve the elasto-plastic problem, the initial stress status caused by thermal strain is computed before any external force is applied. There maybe exists plastic strain. The thermo-mechanical stress under external force is then superimposed upon the thermal stress and strain just obtained.

For plane stress situation, thermal strains are simply presented by

$$\epsilon_x^0 = \alpha T; \qquad \epsilon_{xy}^0 = 0$$
$$\epsilon_y^0 = \alpha T; \qquad \epsilon_{xz}^0 = 0$$
$$\epsilon_z^0 = \alpha T; \qquad \epsilon_{yz}^0 = 0$$

where α is the coefficient of linear thermal expansion and T is the temperature measured from an arbitrary data. It is required for plane stress situation that initial stress component in the through-thickness direction σ_z^0 be zero.

For plane strain, the through-thickness initial stress σ_z^0 is non-zero but the corresponding strain component ϵ_z^0 is required to vanish. Consequently,

$$\epsilon_x^0 = (1 + \nu)\,\alpha T; \qquad \epsilon_{xy}^0 = 0$$
$$\epsilon_y^0 = (1 + \nu)\,\alpha T; \qquad \epsilon_{xz}^0 = 0$$
$$\sigma_z^0 = -E\,\alpha T; \qquad \epsilon_{yz}^0 = 0$$

where ν is the Poisson's ratio for the material.

D.4 RESULTS AND DISCUSSION

As discussed in the last section, the **full contact model** has to be used when unloading occurs at high temperatures. Therefore the thermo-mechanical simulation will be performed by using this full contact model. In studying geometrical effect of the electrode on mechanical contact in cold conditions, the **simple contact model** will be used to simplify this slightly more complicated problem.

D.4.1 Geometric and Material Parameters

Basically, the aluminum sheets are modelled as a rectangle with a thickness H and radius R_s. The electrode is constructed as a curved head with the curvature of R_h and the radius of electrode is R_e. In the centre there is a space for water cooling with radius R_c. The height of the cooling area and the electrode are H_{cl} and H_{el} respectively.

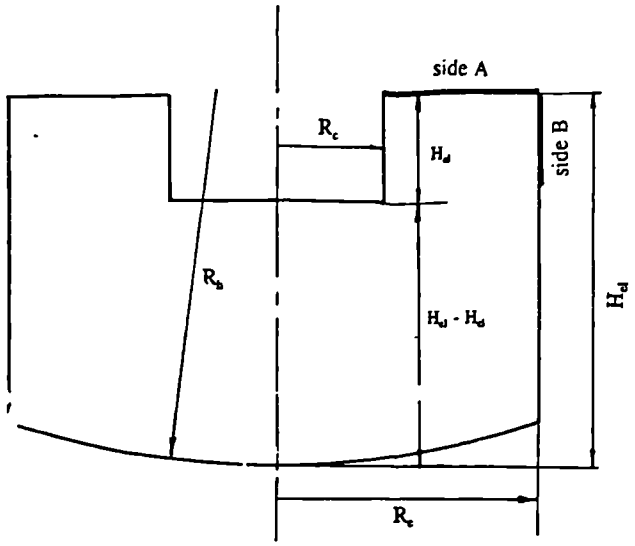


Fig.D.10 Schematic structure of a electrode in spot welding.

The force is applied by distributing a pressure on side **A** of the electrode in our model instead of the shear force on side **B** as observed in practice. The difference between these loading configurations is thought to be very little since either side **A** or side **B** is far away from the location concerned. The schematic figure of the geometry is shown in Fig.D.10. The constants concerning geometry and material properties of both copper and aluminium are listed as follows,

Tab.D.1 Material and geometrical constants

Material constants of aluminium	Young's Modular	Poisson's ratio	Yield stress(cold)	Plastic Modular
	70000MPa	0.25	158MPa	294MPa
Material constants of copper	Young's Modular	Poisson's ratio	Yield stress(cold)	Plastic Modular
	126000MPa	0.25	560MPa	294MPa
Geometric constants	Sht. thickness	Sht. radius	Elc. radius	Col. radius
	H = 2.00mm	R_s = 20.0mm	R_e =5.0mm	R_c = 2.0mm
	Elc. curvature	Elc. height	Col. height	Distance
	R_h = 50.0	H_{el} = 6.0mm	H_{cl} = 2.0mm	H_{el}-H_{cl}=4.0mm

D.4.2 Deformation and Stress of Electrode and Sheet

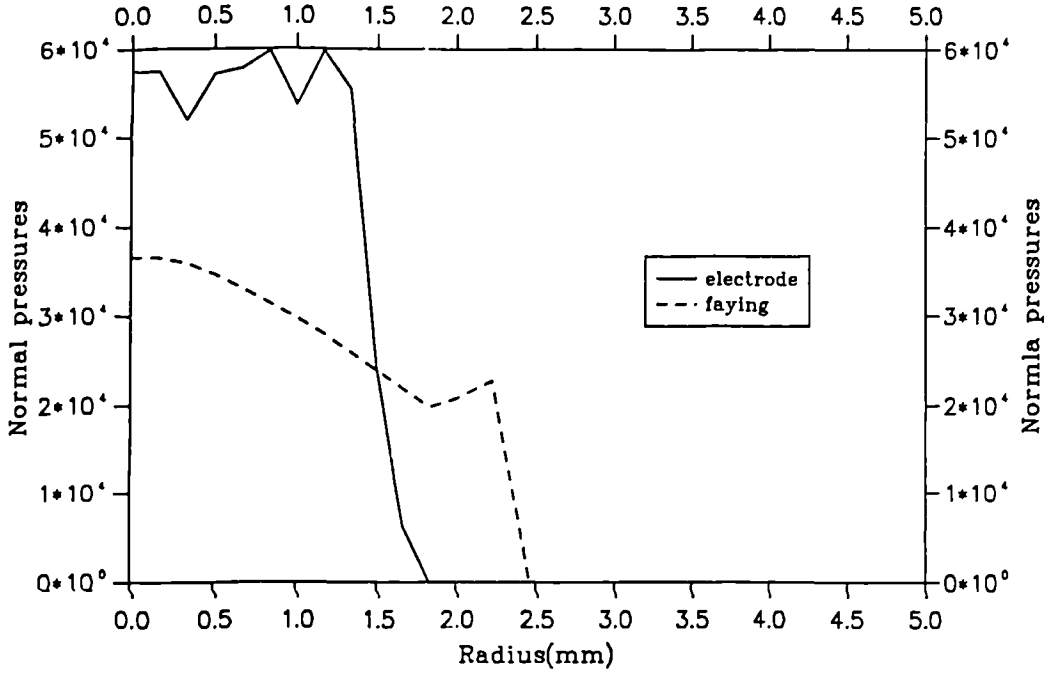
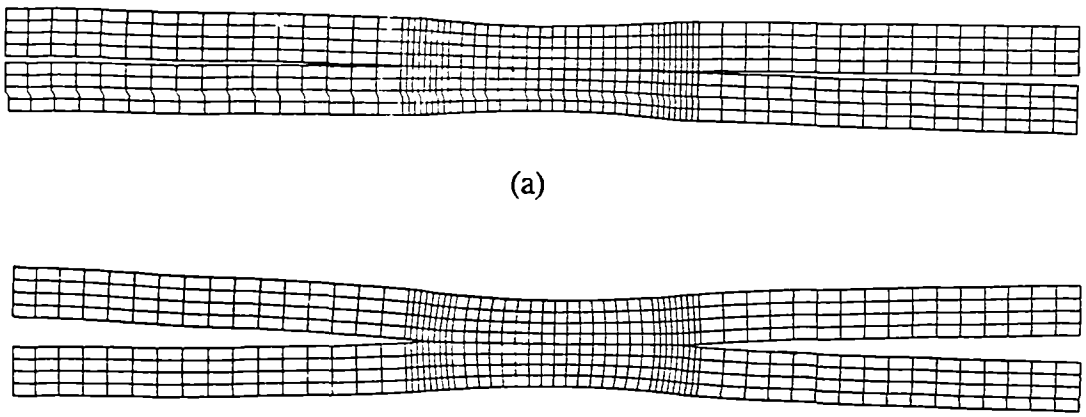


Fig.D.11 Pressure at the faying surface and at the electrode in cold condition.



(a)

(b)

Fig.D.12 Deformations of aluminium sheets under squeeze force. (a) Predicted by **slip model**; (b) by **no slip model** under cold condition.

In cold condition the pressures at the faying surface and the electrode-sheet surface are shown in Fig.D.11. The initial contact areas at both surface are defined by their radii, which are: r_{el} = 2.23mm and r_{fay} = 2.69mm. The whole deformed aluminium sheets

are reconstructed from the deformed FE mesh which is only a quarter of it, as seen in fig.D.12.

When electrical current passes through the sheets, the temperature builds up and the yield stress of the material drops. This causes the redistribution of the stresses and the changes in the contact area. The modification of the contact area can in turn affect the electrical resistance of the welding.

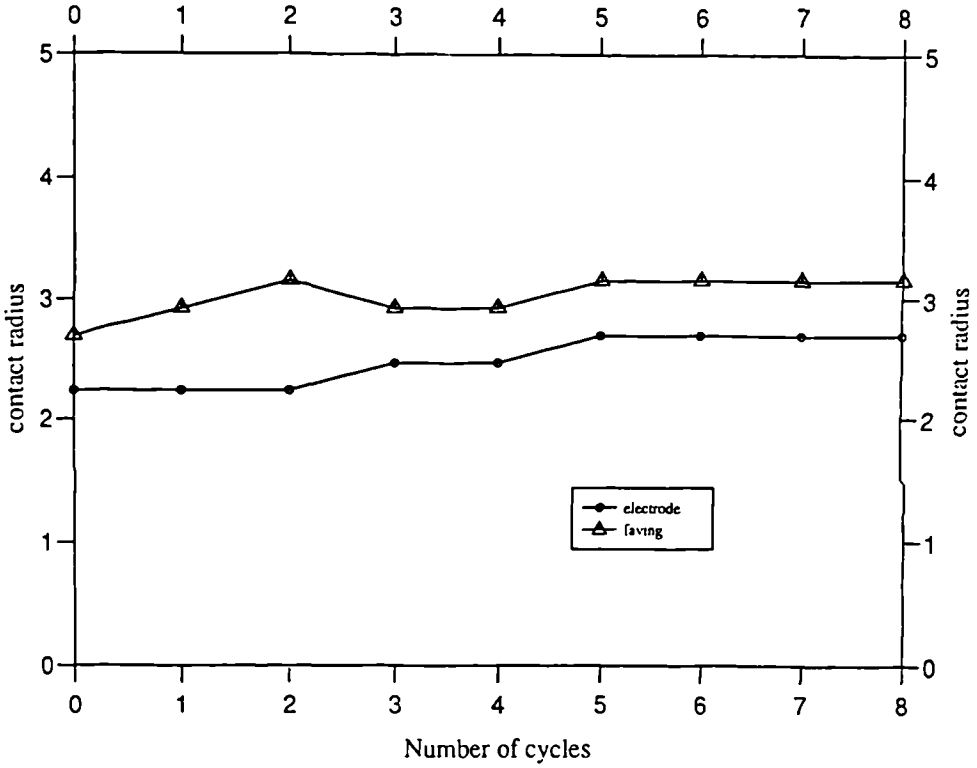


Fig.D.13 The contact areas at both the faying surface and electrode surface at each cycle of the current.

The contact areas of both the faying and electrode surfaces at each cycle of current are shown in Fig.D.13. The pressures on both surfaces at cycle 0, 4 and 8 are shown in Fig.D.14 and Fig.D.15 respectively.

In Fig.D.13, the contact area of both the faying and the electrode surfaces have a tendency to increase with the increasing number of cycles. The changes are larger in the beginning since there are tremendous changes in material properties as temperature increases. These changes are observed until at the fifth cycle. After this, both contact areas remain almost constant. This is probably because no further significant

redistribution of stress and strain takes place during these cycles.

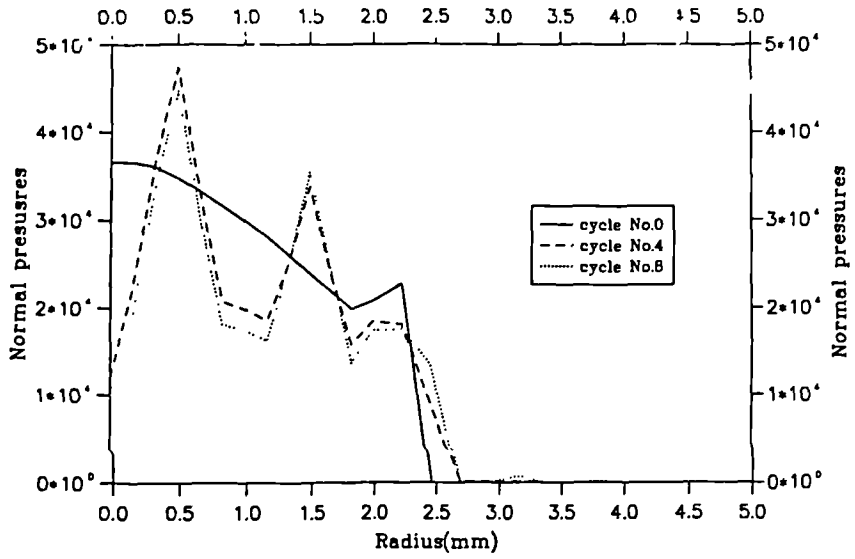


Fig.D.14 Pressures at the faying surface for some cycles of the current.

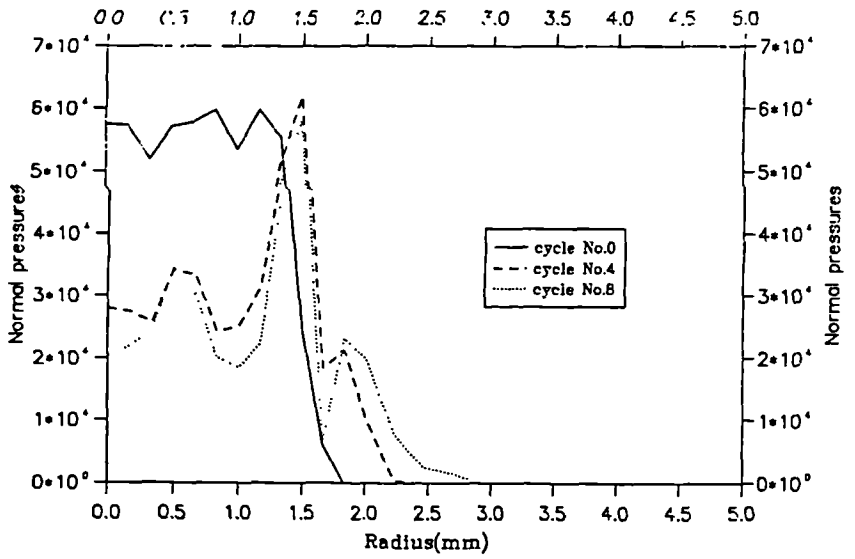


Fig.D.15 The pressures at the electrode surface for some cycles of the current.

At the faying surface, we also find some decrease of the contact area, which can be explained as the larger plastic deformation at the centre area causes the two sheets to separate more than before, thus reduce the contact area. Obviously this will not occur

at the electrode surface.

Pressures at the faying surface in Fig.D.14 are well behaved when everything is cold. However at cycle No.4 and cycle No.8, some oscillation of the pressures is found. In the central area the pressures are released because of the decreased yield stress in this high temperature area. The compressive area at the faying surface is also seen to be larger than at cycle No.0.

For a numerical simulation, the situation for the electrode surface is more difficult to handle because the surface condition is more complicated. This is why even in cold condition, the pressure at the electrode surface proves to be unstable, as found in Fig.D.11 and Fig.D.15. However the results in Fig.D.15 do show reduction of the pressures after heating in central area of welding, and the maximum pressures at cycle No.4 and cycle No.5 are located at about 1.5mm away from the centre of the electrode. The reason why the maximum pressures move outwards is partly because the outer material is colder than at the centre, and thus it has a higher yield stress. As a result, damage of the electrode surface will be more likely in these areas.

D.4.3 Effects of the Profile of the Electrode

The profile of the electrode plays a very important role in the welding process. It can affect the contact area of both the faying and electrode surfaces which may cause changes in the electrical resistance. The shear stresses on both surfaces, are also a vital factor. They break down the electrically insulating layers on the Aluminium sheets and enable welding to start.

However, it is quite clear that at the faying surface, which is the symmetric plane, no shear exists. Of course, the real welding process is more complicated and cannot be fully modelled by a purely mathematical system. But the model can give us some idea how to modify the profile of electrodes such that shear stresses at the faying surface will be produced.

Three pairs of electrodes, male-male model, male-flat model and male-female model, aimed at studying shear stress at the faying surface are proposed as shown in Fig.D.16.

This time we use the **simple contact model** to handle this situation. No

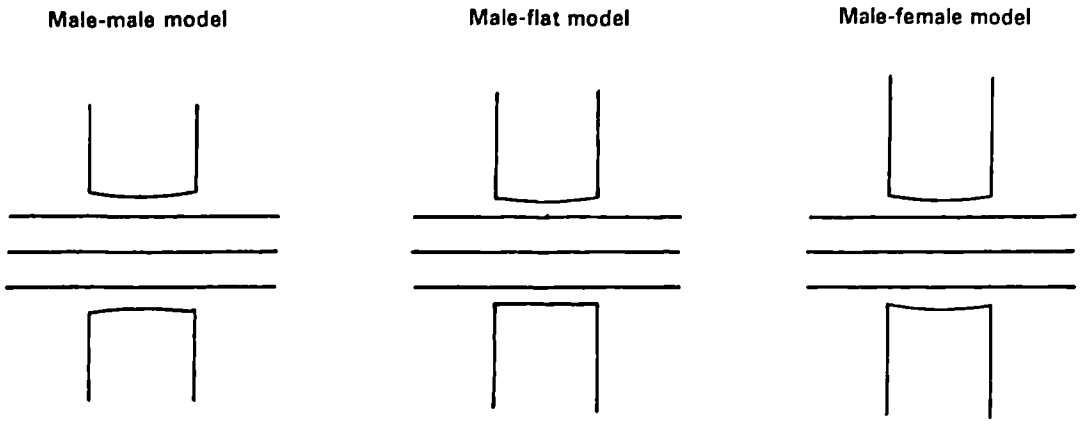


Fig.D.16 Three electrode models for creating shear stress at the faying surface.

temperature effect was included in the study. The shear stresses and normal stresses are presented in Fig.D.17 and Fig.D.18.

The shear stress in the male-male model, which is most close to the real one, is insignificant, while the male-female has largest shear stress among them. The male-flat model has fairly big shear and has about the same normal pressure as that in male-male model, as shown in Fig.D.18.

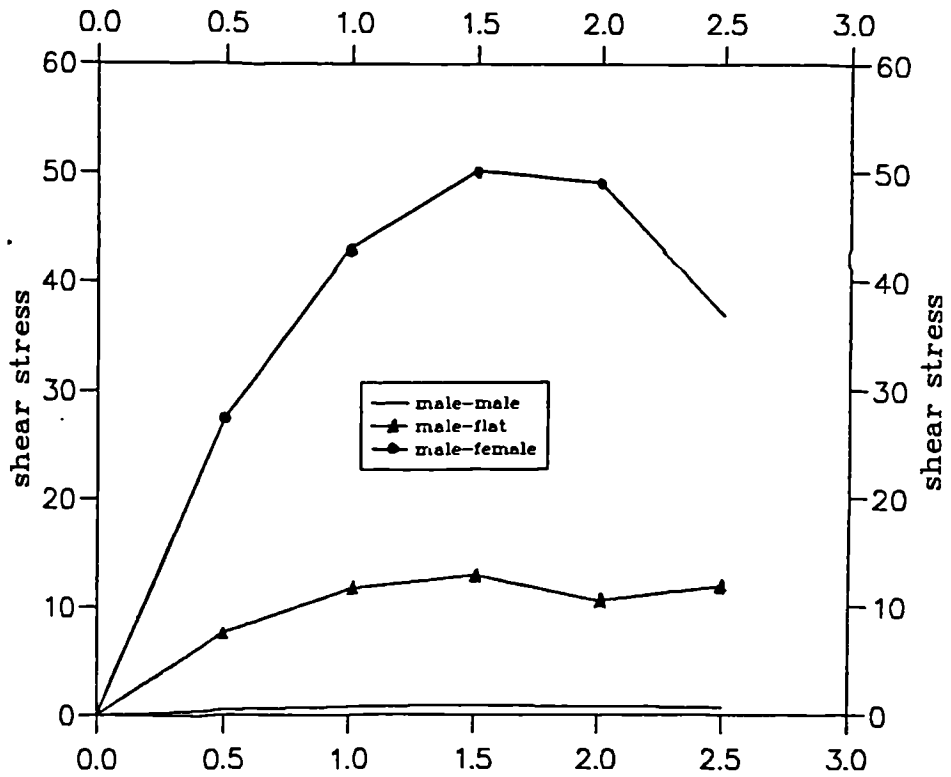


Fig.D.17 Shear stresses at the faying surface.

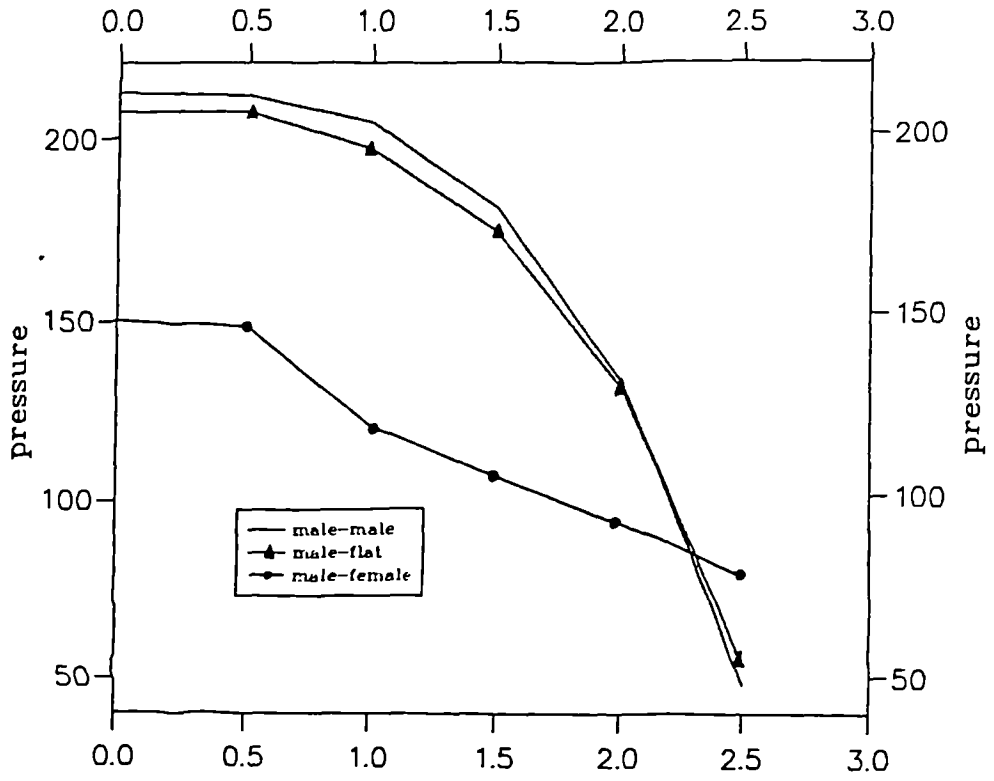


Fig.D.18 Normal stresses at the faying surface.

Compared with the male-male and male-flat models, the male-female model has only about half the contact pressure, which means that it may lack the compressive force to keep the two sheets in full contact. Therefore, to achieve a best weld, our study suggest that the best pair of electrode should be the male-flat model.

In addition to choose a optimum profile of the electrode, the mathematic model can also give us the information about what and how the welding process is influenced by a worn electrode.

Fig.D.19 shows two models of the electrode in the cold condition: **Flat-curved electrode** and **Slope-curved electrode.** The former one is to simulate the electrode which is worn and becomes flat in the centre area.

The pressures on the faying surface by this model are shown in Fig.D.20, where four different r_0 are used to represent different degrees of wearing of the electrode. The pressures change tremendously but there are always pressures in the central area, which means that the sheets are kept together no matter how much wear the electrode has undergone. In the meantime, the contact areas at the faying surface increase largely when r_0 increase, which may affect the temperature distribution because the electrical

resistance is varied.

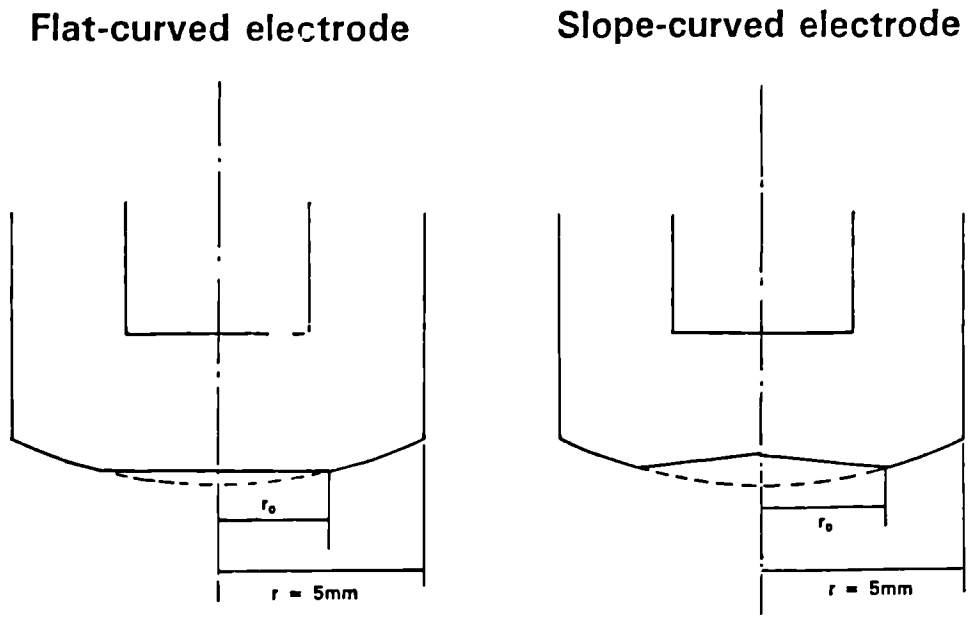


Fig.D.19 Two models for studying of a worn electrode

On the other hand, the **Slope-curved electrode** is an extreme case of a damaged electrode, which is designed to show whether it is possible for the two aluminium sheets to separate in the central area when a badly damaged electrode is used.

Fig.D.21 shows pressures under four different r_0. When r_0 increases under external load of 8kn, the pressures in the central area decrease. If r_0 = 4.0mm, i.e. 80% of the radius of the electrode R_e, the pressure at the centre is zero. This means that separation may occur between two aluminium sheets. This can also take place when less external load (3.5kn) is applied while r_0 = 3.5mm, i.e. 70% of R_e.

However these extreme cases are unlikely to happen since in Fig.D.15, the most probable position for an damage at the electrode tip is at about 30% of R_e at the electrode surface.

D.4.4 Deformation of the Adhesive under Cold Condition

Say an adhesive is used to attach two aluminium sheets together with viscosity μ at cold condition. The deformation of the adhesive under the squeeze force is affected by following parameters:

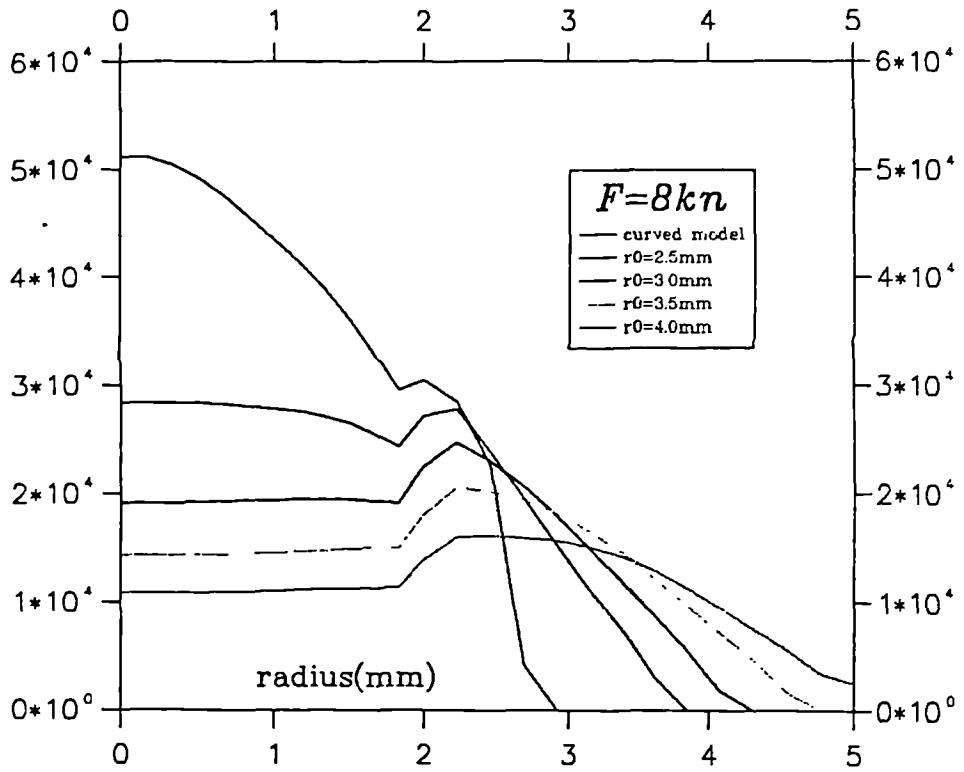


Fig.D.20 Pressures at the faying surface in the Flat-curved model.

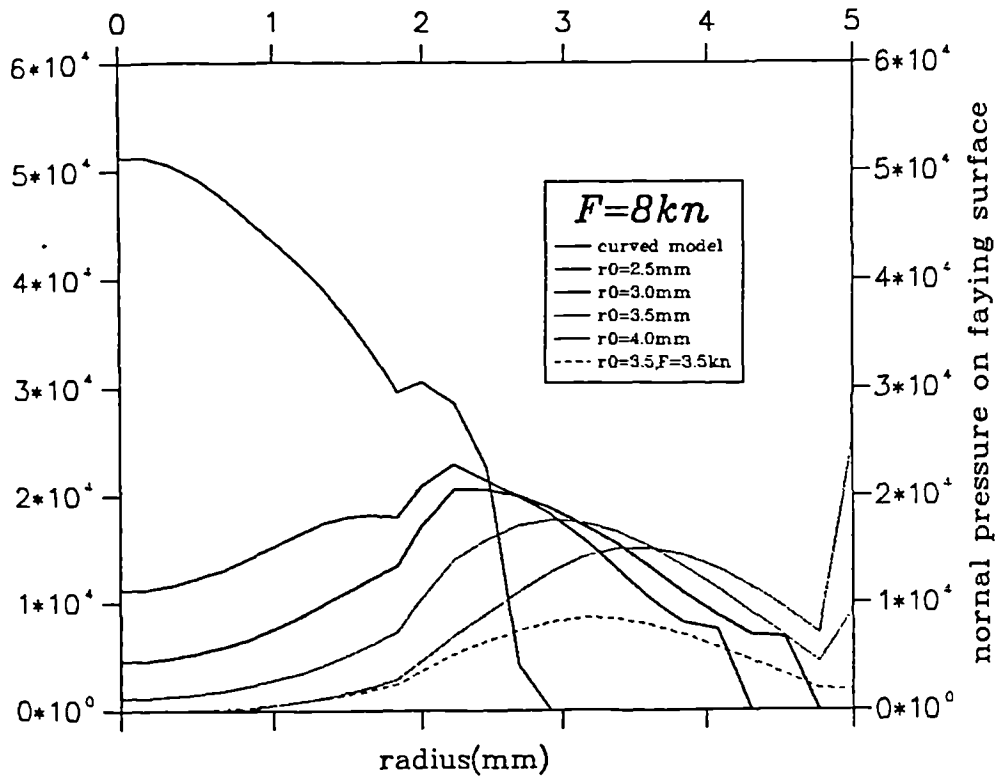


Fig.D.21 Pressures at the faying surface in the Slope-curved model.

- Loading profile: $F(t)$ determined by F_T and $\beta = F_T / F_{½T}$;

- Viscosity of the adhesive: μ;
- Yield stress of aluminium sheet: σ_y;
- Thickness of aluminum sheet: H;
- Radius of the electrode curvature:R_h.

The size of the entrapment is represented in terms of volume of the shaded area in fig.D.22.

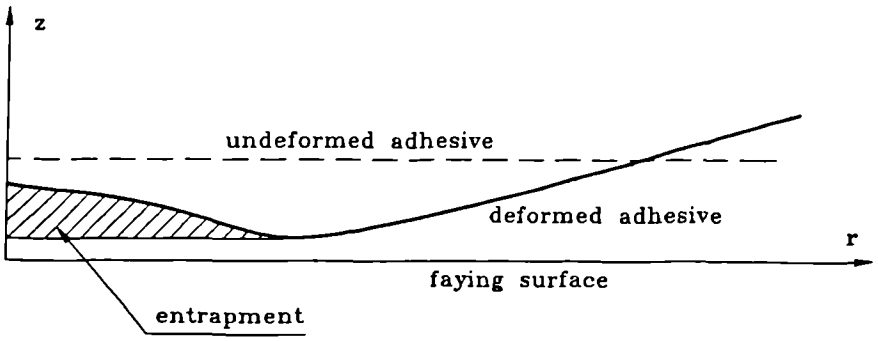


Fig.D.22 The size of the entrapment is presented by the volume of the shaded area.

The simplified load process is presented by an exponential function with different rates of β, as shown in fig.D.23. Various deformations of adhesive under different load profiles are also presented in this fig.

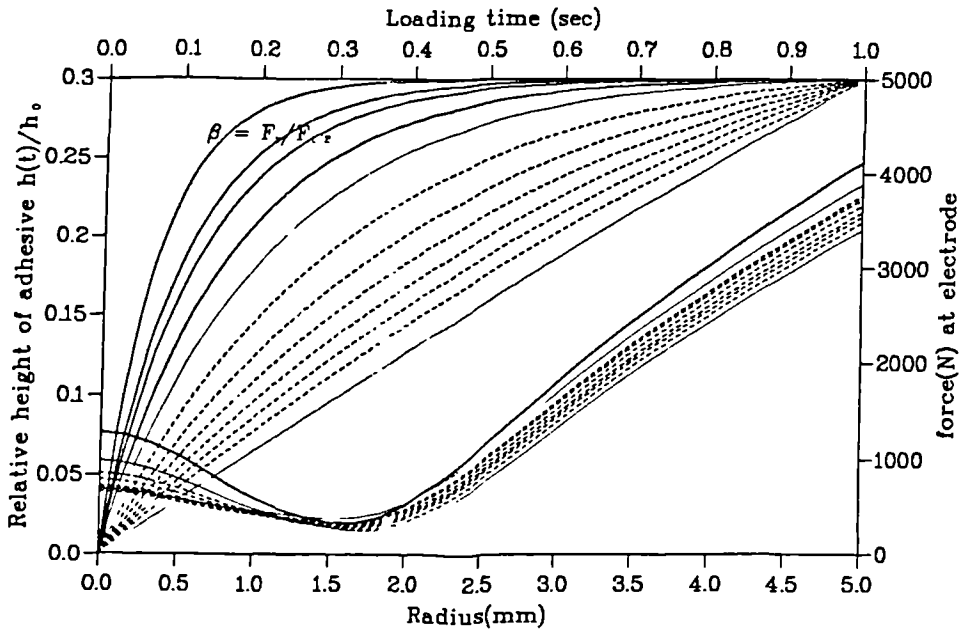


Fig.D.23 Load profile $F(t)$ assumed as a exponential function. The deformations with different β under F_T=-5000kn, μ=1000PaS, σ_y=1.58MPa.

It is clearly found that the minium thickness of the adhesive after deformation is at about 1.5mm away from centre. Therefore if two aluminium sheets eventually contact, there is a space that "traps" a small amount of adhesive.

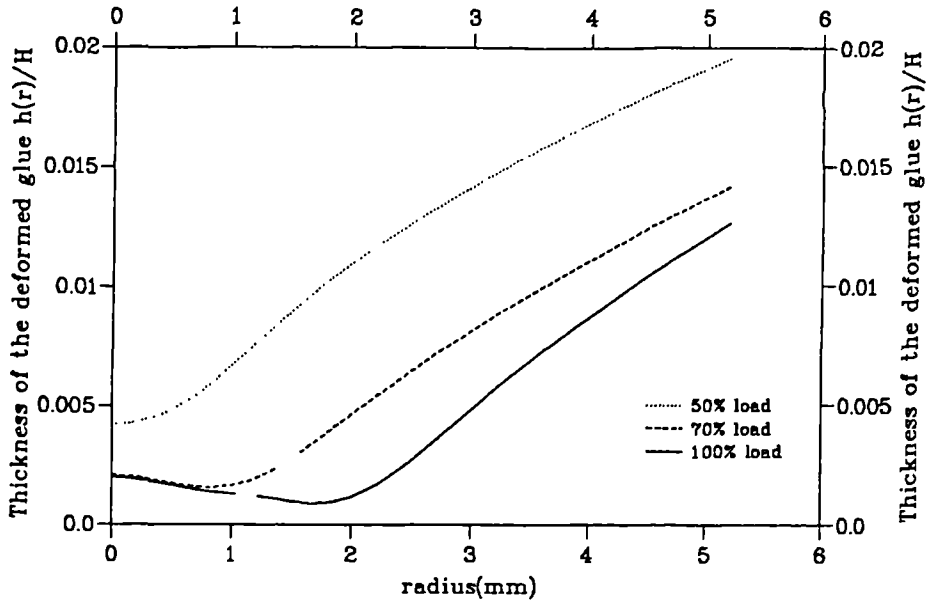


Fig.D.24 Formation of the entrapment of the adhesive at the same condition as fig.D.23.

Fig.D.24 shows that how the "entrapment" is built up. When 30% of F_T is achieved, no entrapment is found. This is because that the curve electrode presses the centre part of the aluminium sheet more than outer range. With further increase of the load to 70%, the entrapment can be identified. This may cause by yielding of the centre part of the aluminium sheet. Finally at 100% of F_T, more deformation of adhesive is caused. The volume of trapped adhesive is increased too.

It can be anticipated that the yield stress of the aluminium and the viscosity of the adhesive play a important role of the entrapment. In fig.D.25, it is shown that entrapment increases with the higher viscosities of adhesive. With regard to the yield stress, the situation is that the entrapment decreases with the lower σ_y. It is also noticed that when very high σ_y is used, i.e. 200MPa, there is virtually no entrapment found.

Other parameters such as the thickness of the aluminium sheet H and the radius of the electrode curvature R_h also have effects on the entrapment in some degree. Our

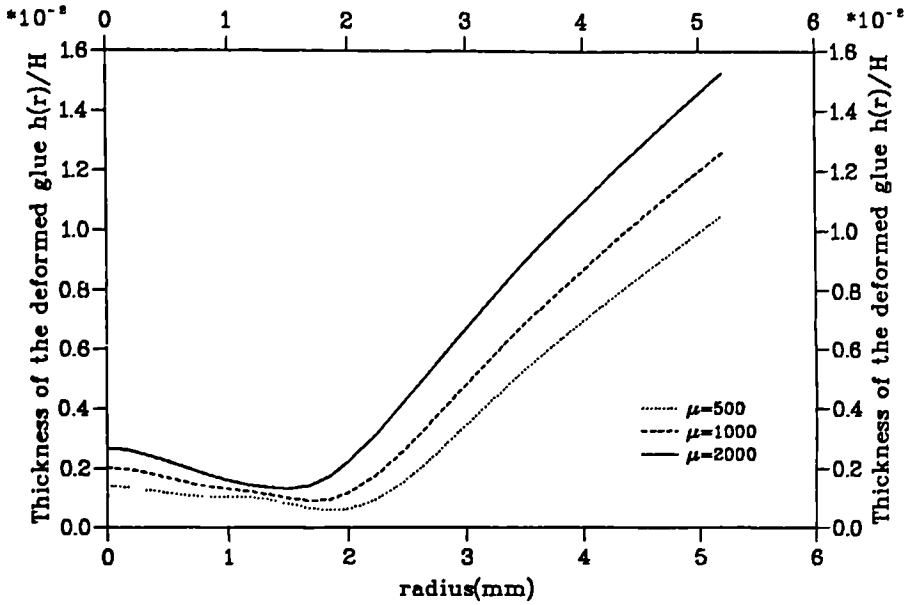


Fig.D.25 The effects of viscosity on the deformation of the adhesive. F_T=-5000kn, σ_y=1.58MPa, H=2.0mm and R_h=50.0mm.

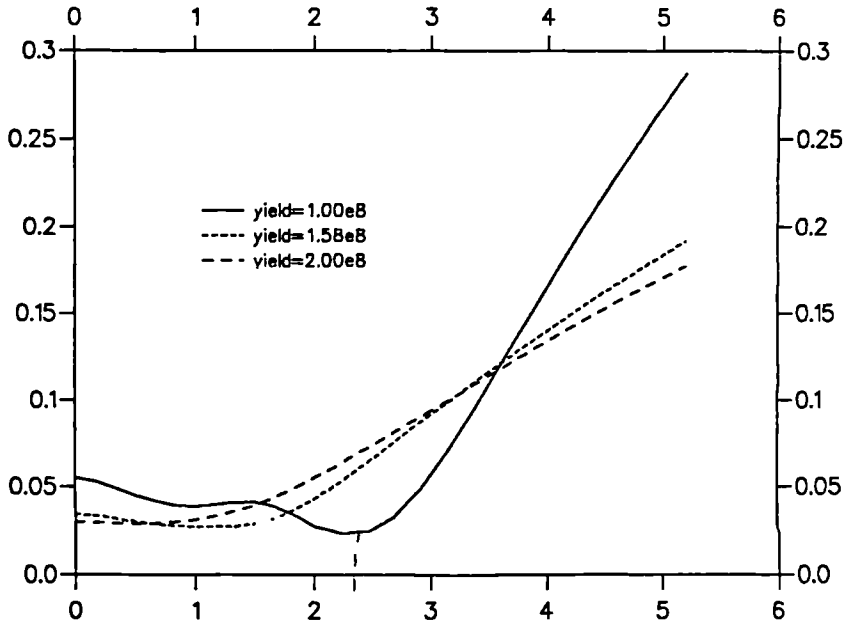


Fig.D.26 The effects of yield stress on the deformation of the adhesive. F_T=-5000kn, μ=1000PaS, H=2.0mm and R_h=50.0mm.

task is to find a parameter which can include all the parameters and also directly related to the size of the entrapment.

Say α is designated as such a dimensionless parameter and v^* is dimensionless

volume, i.e.

$$v^* = \frac{v}{R_h^3} \quad ; \quad \alpha = \frac{F_T}{\sigma_y R_h H}$$

If dimensionless parameter α calculated with three different value for each parameters is plotted against the dimensionless v^*, then a very interesting relationship between is found, as seen in fig.D.27. v^* varies with α nearly parabolically. In other words, to minimize the entrapment can be achieved by carefully choosing a set of F_T, R_h, H and σ_y such that α is small enough.

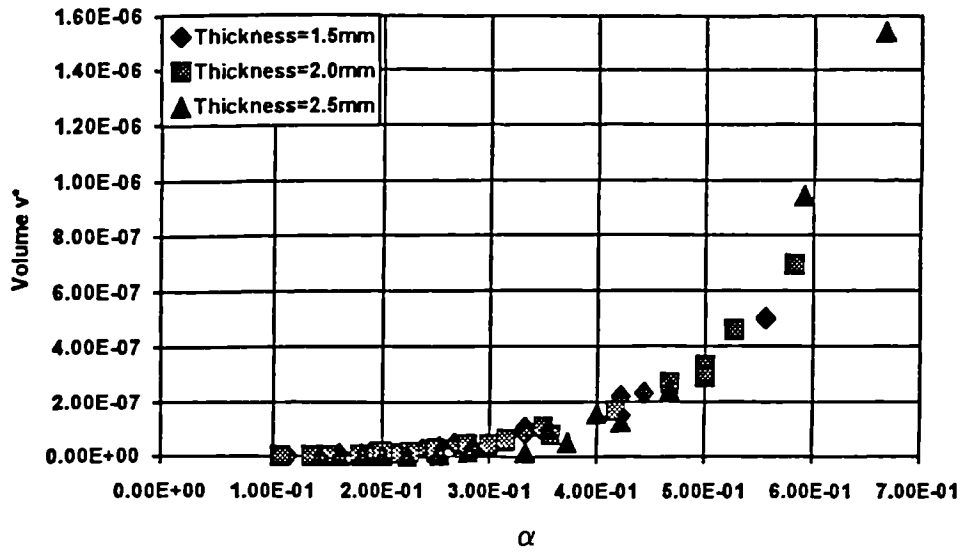


Fig.D.27 Dimensionless volume v^* vary with dimensionless parameter α.

APPENDIX E

Fortran Program MIXEP.F for ELASTO-PLASTICITY

```
c     program penalty function,tape5=input
c    *output,tape8=output, tape4,tape13)
c*****************************************************************L
c        *******   elastoplastic program  ******
c                          SUN & HP version
c     first order stress function and displacement variable----
c        mixed-fem package for plane and axisymmetric
c      using Newton's method in incremental form
c      and changed the integration method in axisymmetric
c      situation. program has been tested.
c
c      variational equation:
c  when ps>=0
c    L= s*d - sqrt(s*s)sqrt(d*d) - rho*ps^2 - rho*e^2
c  when ps<0
c    L= -2G*d*d - k*e^2
c  The general variable B is dimensionless by replacing F by F/2G
c                designed by J.Wen
c*************************************************************
      implicit double precision(a-h,o-z)
        parameter (nmax=4000, tol=1.0e-4)
        external sign, dmax, dmin, delt, times
      dimension ifpre(5,1000),lnods(500,9),leqns(45,500),coord(1000,2),
     *          force( 4000),props(10, 5) ,mhigh( 4000), matno(500),
     *          dispt( 4000),posgp(4) ,weigp(4),stifi(200000),
     *          maxai( 4000),bline(4,500), presc(3,100), gpcod(2,4500)
        common stres(5,4500),strin(5,4500),strdd(5,4500)
        common /dfunc/pcom(nmax), xicom(nmax), ncom
        read(5,*)trial,niter
    open(5,file='mixep.in',status='old')
    open(8,file='mixep.ou',status='unknown')
    open(4,file='fort04',status='unknown')
    open(13,file='fort.13',status='unknown')
        call contol (ndofn ,nelem ,nmats ,npoin )
c
        call inputd (coord ,presc ,lnods ,matno ,ndime ,ndofn ,nelem ,
     *               ngaus ,nmats ,nnode ,npoin ,nstre ,ntype ,posgp ,
     *               props ,weigp ,nbdis ,nline ,bline )
        ncom=npoin*ndofn
        do 20 i=1,ncom
        dispt(i)=0.0
20      pcom(i)=0.0
         rho=1.0e20
c
        call linkin (force ,ifpre ,npoin ,leqns ,lnods ,maxai ,
     *               nwktl ,mhigh ,ndofn ,nelem ,neqns ,nnode )
c
        do 1000 iiter=1,niter
        write(6,'(1x,6hiiter=,i5)')iiter
        write(4,'(1x,6hiiter=,i5)')iiter
        call gstiff (coord ,stifi ,leqns ,lnods ,matno ,maxai ,
     *               nwktl ,ndime ,ndofn ,nelem ,ngaus ,nmats ,
     *               nnode ,npoin ,nstre ,ntype ,props ,weigp ,
     *               posgp ,dispt ,neqns ,nbdis ,nline ,bline ,
     *               ifpre ,rho   ,presc ,force ,gpcod ,iiter ,
     *               trial)
        do 100 icom=1,ncom
          xicom(icom)=dispt(icom)
100     continue
```

```
        write(6,'(1x,6hgstiff)')
c
          ax=1.0
          bx=2.0
          call mnbrak(ax,  bx, cx, fa ,fb, fc,   rho,
     *                lnods, coord, nnode, nevab, ndofn,
     *                npoin, nelem, ndime, ntype, nstre, ngaus,
     *                props, nmats, matno, weigp, posgp,
     *                presc, bline, ifpre, nbdis, nline,iiter)
          write(6,'(1x,6hmnbrak)')
          write(6,'(1x,3e12.4)')ax,bx,cx
          write(6,'(1x,3e12.4)')fa,fb,fc
C
       call brent(ax, bx, cx, tol, xmin,  vm ,rho,
     *             lnods, coord, nnode, nevab, ndofn,
     *             npoin, nelem, ndime, ntype, nstre,ngaus,
     *             props, nmats, matno, weigp, posgp,
     *             presc, bline, ifpre, nbdis, nline,iiter)
          write(6,'(1x,6hbrent ,2e12.4)')xmin,vm
          do 200 j=1,ncom
            pcom(j)=pcom(j)+xicom(j)*xmin
            dispt(j)=pcom(j)
200       continue
c
           call output (dispt,npoin,nelem,nstre,ngaus,ndofn,gpcod,
     *                 coord)
          if(dabs(xmin).lt.1.0e-5)stop
1000      continue
          close(8)
c
       stop
       end
c
       subroutine contol (ndofn ,nelem ,nmats ,npoin )
c*****************************************************************
c
c***  read control data and check for dimension
c
c*****************************************************************
       read(5,*  ) npoin,nelem,ndofn,nmats
       if(nelem.gt.  500) go to 200
       if(npoin.gt. 1000) goto 200
       if(nmats.gt.   10) goto 200
       goto 210
200    write(8,120)
       stop
120    format(/'set dimension exceeded - contol check '/)
110    format(16i5)
210    continue
       return
       end
c
       subroutine inputd (coord ,presc ,lnods ,matno ,ndime ,ndofn ,
     *                    nelem ,ngaus ,nmats ,nnode ,npoin ,nstre ,
     *                    ntype ,posgp ,props ,weigp ,nbdis ,nline ,
     *                    bline )
c*********************************************************************
c
c***  input routine
c
c*********************************************************************
       implicit double precision(a-h,o-z)
       dimension coord(npoin,*) ,presc(3,*) ,weigp(*) ,matno(*) ,
     *           lnods(nelem,*) ,props(nmats,*) ,posgp(*) ,title(10),
     *           bline(4,*)
       read(5,913) title
913    format(20a4)
```

```
      write(8,914) title
914   format(//,5x,20a4)
c
c***  read the first data card, and echo it immediately.
c
      read (5,*  ) nbdis,nline,ntype,nnode,nprop,ngaus,ndime,nstre
      write(8,901) npoin,nelem,ntype,nnode,ndofn,nmats,nprop,
     *             ngaus,ndime,nstre,nbdis,nline
901   format (/5x,18hcontrol parameters/
     *        /5x,8h npoin =,i10,5x,8h nelem =,i10,5x,8h ntype =,i10,
     *        /5x,8h nnode =,i10,5x,8h ndofn =,i10,5x,8h nmats =,i10,
     *        /5x,8h nprop =,i10,5x,8h ngaus =,i10,5x,8h ndime =,i10,
     *        /5x,8h nstre =,i10,5x,8h nbdis =,i10,5x,8h nline =,i10/)
900   format(16i5)
c
c***  read the element nodal connections, and the property numbers
c
      write (8,902)
902   format(//5x,8h element,3x,8hproperty,6x,12hnode numbers)
      do 530 ielem=1,nelem
      read (5,*  ) num,matno(num),(lnods(num,inode),inode=1,nnode)
      write(13,915) (lnods(ielem,i),i=1,nnode)
530   write(8,903) matno(ielem),(lnods(ielem,inode),inode=1,nnode)
903   format(6x,i9,6x,10i5)
915   format(1x,8(i5,','))
c
c***  read some nodal coordinates, finishing with the last node of all
c
904   format(//5x,5h node,9x,1hx,9x,1hy,5x)
200   read (5,*  ) (l,(coord(l,idime),idime=1,ndime),i=1,npoin)
      write(8,906) (i,(coord(i,idime),idime=1,ndime),i=1,npoin)
      write(13,'(1x,2e15.4)')((coord(i,idime),idime=1,ndime),i=1,npoin)
905   format(i5,6f10.5)
906   format(5x,i5,2e15.4)
c
c***  read the available selection of element properties
c
      write(8,910)
910   format(//5x,19hmaterial properties)
      do 520 imats=1,nmats
      read(5,*  ) numat
      read(5,*  ) (props(numat,iprop),iprop=1,nprop)
      write(8,911) numat
911   format(/5x,11hmaterial no,i5)
520   write(8,912) (props(numat,iprop),iprop=1,nprop)
912   format(/5x,13hyoung modulus,g12.4/5x,13hpoisson ratio,g12.4/
     *        5x,13hthickness    ,g12.4/5x,13hreference no ,g12.4/
     *        5x,13hhardening par,g12.4/)
      read(5,*) ((presc(i,ibdis),i=1,3),ibdis=1,nbdis)
      write(8,121)
121   format(5x,'displacement boundry information'/)
      write(8,120)((presc(i,ibdis),i=1,3),ibdis=1,nbdis)
120   format(1x,f8.2,4x,2e12.4)
      read(5,*) ((bline(i,iline),i=1,4),iline=1,nline)
      write(8,123)
123   format(5x,'stress boundry information'/)
      write(8,124)((bline(i,iline),i=1,4),iline=1,nline)
124   format(3x,2f8.1,5x,2e12.4)
c
        call gaussq (ngaus,posgp,weigp)
      return
      end
c
        subroutine linkin (force ,ifpre ,npoin ,leqns ,lnods ,maxai ,
     *                     nwktl ,mhigh ,ndofn ,nelem ,neqns ,nnode )
c*********************************************************************
c
```

```
c***  links with profile solver
c
c*********************************************************************
      implicit double precision(a-h,o-z)
       dimension lnods(nelem,*),leqns(45,*),maxai(*),leqnn(45),
     *           ifpre(ndofn,*) ,force(*), mhigh(*)
c
       nevab=nnode*ndofn
c
c***   number of unknowns
c
       neqns=0
       do 100 ipoin=1,npoin
       do 150 idofn=1,ndofn
         neqns=neqns+1
         ifpre(idofn,ipoin)=neqns
150    continue
100    continue
       meqns=1+neqns
c
c***   connectivity array leqns
c
       do 70 ielem=1,nelem
       do 70 ievab=1,nevab
70     leqns(ievab,ielem)=0
       do 50 ielem=1,nelem
       ievab=1
       do 80 inode=1,nnode
       ident=lnods(ielem,inode)
       do 80 idofn=1,ndofn
       leqns(ievab,ielem)=ifpre(idofn,ident)
80     ievab=ievab+1
c      write(8,6) ielem,(leqns(ievab,ielem),ievab=1,nevab)
50     continue
6      format(i10,24i3)
7      format(4i10)
8      format(8e12.4)
c
c**    loop over all elements
c
250    do 190 ielem=1,nelem
       do 300 le=1,nevab
         leqnn(le)=leqns(le,ielem)
300    continue
c
       call colmht  (mhigh, nevab,leqnn)
190    continue
c
c***   addresses of diagonal elements- maxa array
c
       call addres(maxai,mhigh,neqns,nwktl,mkoun)
       write(8,920)neqns,nwktl
920    format(/5x,'neqns=',i5,5x,'nwktl=',i5/)
       if(nwktl.gt.200000) goto 210
       goto 220
210    write(8,910)
       stop
220    continue
910    format(/'set dimension exceeded - check linkin '/)
       return
       end
c
       subroutine  gstiff (coord ,stifi ,leqns ,lnods ,matno ,maxai ,
     *                     nwktl ,ndime ,ndofn ,nelem ,ngaus ,nmats ,
     *                     nnode ,npoin ,nstre ,ntype ,props ,weigp ,
     *                     posgp ,dispt ,neqns ,nbdis ,nline ,bline ,
     *                     ifpre ,rho   ,presc ,force ,gpcod ,iiter ,
```

```
     *                        trial)
c*****************************************************************
c
c     evaluates linear stiffness matrix
c     for mixed extrieme mothed
c
c*****************************************************************
      implicit double precision(a-h,o-z)
      double precision ldd(5,5),lss(5,5),lds(5,5),ld1(5),ls1(5)
      dimension coord(npoin,*) ,elcod(2,9) ,lnods(nelem,*) ,cartd(2,9),
     *          props(nmats,*) ,leqns(45,*),estif(    820) ,
     *          deriv(2,9) ,shape(9) ,leqnn(45), bline(4,*),
     *          amatx(5,30),bmatx(5,20),asmat(5,30),bsmat(5,20),
     *          ammat(30), bmmat(20),gpcod(2,*),ifpre(ndofn,*),
     *          presc(3,*),force(*)
        common stres(5,4500),strin(5,4500),strdd(5,4500)
c
      dimension stifi(*),maxai(*),posgp(*),dispt(*),weigp(*),matno(*),
     *          s(5),d(5),dt(5)
c
      twopi=6.283185307179586
      kgaus=0
c
c***  loop over each element
c
      nstr1=5
      nevab=ndofn*nnode
      do 500 iwktl=1,nwktl
500   stifi(iwktl)=0.0
      do 70 ielem=1,nelem
      lprop=matno(ielem)
c
c***  evaluate the coordinates of the element nodal points
c
      do 10 inode=1,nnode
      lnode=lnods(ielem,inode)
      do 10 idime=1,ndime
10    elcod(idime,inode)=coord(lnode,idime)
      young=props(lprop, 1)
      poiss=props(lprop, 2)
      thick=props(lprop, 3)
        y0 =props(lprop, 4)*0.816497
        y  =props(lprop, 5)*0.816497
       shear=young/2.0/(1.0+poiss)
       xkcon=young/3.0/(1.0-2.0*poiss)
       factr=young/props(1,1)
c
c***  initialaze the element stiffness matrix 820=nevab*(nevab+1)/2
c
      do 20 isize=1,820
20    estif(isize)=0.0
c
c***  enter loops for area numerical integration
c
      do 50 igaus=1,ngaus
      exisp=posgp(igaus)
      do 50 jgaus=1,ngaus
      etasp=posgp(jgaus)
      kgaus=kgaus+1
      call sfr2   (deriv,nnode,shape,exisp,etasp)
      call jacob2 (cartd,deriv,djacb,elcod,gpcod,
     *             ielem,kgaus,nnode,shape)
      dvolu=djacb*weigp(igaus)*weigp(jgaus)
      radiu=gpcod(1,kgaus)
      if(ntype.eq.3) dvolu=dvolu*twopi*radiu
      if(ntype.eq.1) dvolu=dvolu*thick
c
```

```
c***  evaluate the a,b,as,bs,am and bm matrices
c
        if(iiter.eq.1)then
          if(ntype.ne.3) then
c
c****     for rigid plastic poiss=0.5
            call matrip(amatx, bmatx, asmat, bsmat,ammat, bmmat,
     *                  radiu, shape, cartd, poiss,nnode,ntype,0)
          else
            call matria(amatx, bmatx, asmat, bsmat,ammat, bmmat,
     *                  radiu, shape, cartd, poiss,nnode,ntype,0)
          end if
        end if
c
c***  calculate the differential of l
c
      if(iiter.gt.1) then
        sigma=(stres(1,kgaus)+stres(4,kgaus)+stres(5,kgaus))/3.0
        epcilt= strin(1,kgaus)+strin(4,kgaus)+strin(5,kgaus)
        epcild= strdd(1,kgaus)+strdd(4,kgaus)+strdd(5,kgaus)
        s(1)=stres(1,kgaus)-sigma
        dt(1)=strin(1,kgaus)-epcilt/3.0
        d(1)=strdd(1,kgaus)-epcild/3.0
        s(2)=stres(2,kgaus)
        dt(2)=strin(2,kgaus)
        d(2)=strdd(2,kgaus)
        s(3)=stres(3,kgaus)
        dt(3)=strin(3,kgaus)
        d(3)=strdd(3,kgaus)
        s(4)=stres(4,kgaus)-sigma
        dt(4)=strin(4,kgaus)-epcilt/3.0
        d(4)=strdd(4,kgaus)-epcild/3.0
        s(5)=stres(5,kgaus)-sigma
        dt(5)=strin(5,kgaus)-epcilt/3.0
        d(5)=strdd(5,kgaus)-epcild/3.0
c
        call differ2(s,d,dt,ldd,lss,lds,ld1,ls1,y0,y,ps)
c
c****     for rigid plastic poiss=0.5
c
          plasp=poiss
c          if(ps.ge.0)plasp=0.49
          if(ntype.ne.3) then
            call matrip(amatx, bmatx, asmat, bsmat,ammat, bmmat,
     *                  radiu, shape, cartd, plasp,nnode,ntype,0)
          else
            call matria(amatx, bmatx, asmat, bsmat,ammat, bmmat,
     *                  radiu, shape, cartd, plasp,nnode,ntype,0)
          end if
        call differ1(force,bsmat,asmat,ammat,bmmat,lnods,nelem,
     *               nnode, ndofn,nstre,ntype,ielem,dvolu,ld1,ls1,
     *               epcild,epcilt,shear,xkcon,ps,dt)
        end if
c***  calculate the element stiffness
c
      call estifn(estif,amatx,bmatx,asmat,bsmat,ammat,bmmat,
     *            nnode,shear,xkcon,dvolu,kgaus,ndofn,ntype,
     *            ldd  ,lss  ,lds  ,ps   ,iiter, trial, factr)
50    continue
c
c***  generates global stiffness matrix in compacted coln form
c
      do 18 le=1,nevab
       leqnn(le)=leqns(le,ielem)
18    continue
c
      call addban (stifi,maxai,estif,leqnn,nevab)
```

```
 70    continue
 c
 c****    add boundary conditions
 c
         if(ntype.ne.3) then
           call boundp(stifi,maxai,ifpre,nbdis,presc,nline,thick,
      *              bline,ndofn,rho,coord,npoin,force,dispt,shear)
         else
           call bounda(stifi,maxai,ifpre,nbdis,presc,nline,thick,
      *              bline,ndofn,rho,coord,npoin,force,dispt,shear)
         end if
 c
       do 100 ieqns=1,neqns
         dispt(ieqns)=force(ieqns)
         force(ieqns)=0.0
 100   continue
 c
 c*******  do decomposition for global stiffness matrix
 c
       call decomp (stifi ,maxai ,neqns ,   1  )
 c
       call redbak (stifi,dispt,maxai,neqns)
       return
       end
 c
         subroutine matrip(amatx, bmatx, asmat, bsmat,ammat, bmmat,
      *              radiu, shape, cartd, poiss, nnode ,ntype, key )
 c*************************************************
 c       this is a routine to form a, b, as
 c       bs, am, bm for plane problem using shape function n
 c*************************************************
         implicit double precision(a-h,o-z)
         dimension amatx(5,* ), bmatx(5,* ), asmat(5, *), bsmat(5,* ),
      *            ammat( *  ), bmmat(*   ), cartd(2,9 ), shape(*)
 c
         icont=0
         do 10 inode=1,nnode
 c
 c****       a matrix
             amatx(1,icont+1)=0.0
             amatx(1,icont+2)= cartd(2,inode)
             amatx(2,icont+1)=-cartd(2,inode)
             amatx(2,icont+2)=0.0
             amatx(3,icont+1)=0.0
             amatx(3,icont+2)=-cartd(1,inode)
             amatx(4,icont+1)= cartd(1,inode)
             amatx(4,icont+2)=0.0
 c
             if(ntype.eq.1) then
               amatx(5,icont+1)=0.0
               amatx(5,icont+2)=0.0
             end if
             if(ntype.eq.2) then
               amatx(5,icont+1)=poiss*cartd(1,inode)
               amatx(5,icont+2)=poiss*cartd(2,inode)
             end if
             if (key.eq.1) goto 20
 c
 c*****    b matrix
            bmatx(1,icont+1)= cartd(1,inode)
            bmatx(1,icont+2)= 0.0
            bmatx(2,icont+1)=0.5*cartd(2,inode)
            bmatx(2,icont+2)=0.5*cartd(1,inode)
            bmatx(3,icont+1)=0.5*cartd(2,inode)
            bmatx(3,icont+2)=0.5*cartd(1,inode)
            bmatx(4,icont+1)=0.0
            bmatx(4,icont+2)= cartd(2,inode)
```

```
            if(ntype.eq.1)then
              bmatx(5,icont+1)=-poiss*cartd(1,inode)/(1.0-poiss)
              bmatx(5,icont+2)=-poiss*cartd(2,inode)/(1.0-poiss)
            end if
c
            if(ntype.eq.2) then
              bmatx(5,icont+1)=0.0
              bmatx(5,icont+2)=0.0
            end if
c
c****    am matrix and bm matrix
              ammat(icont+1)=(amatx(1,icont+1)+amatx(4,icont+1)
     *                       +amatx(5,icont+1))/3.0
              ammat(icont+2)=(amatx(1,icont+2)+amatx(4,icont+2)
     *                       +amatx(5,icont+2))/3.0
c
              bmmat(icont+1)=bmatx(1,icont+1)+bmatx(4,icont+1)
     *                       +bmatx(5,icont+1)
              bmmat(icont+2)=bmatx(1,icont+2)+bmatx(4,icont+2)
     *                       +bmatx(5,icont+2)
c
c****  as matrix
c
            asmat(1,icont+1)=amatx(1,icont+1) -ammat(icont+1)
            asmat(1,icont+2)=amatx(1,icont+2) -ammat(icont+2)
            asmat(2,icont+1)=amatx(2,icont+1)
            asmat(2,icont+2)=amatx(2,icont+2)
            asmat(3,icont+1)=amatx(3,icont+1)
            asmat(3,icont+2)=amatx(3,icont+2)
            asmat(4,icont+1)=amatx(4,icont+1) -ammat(icont+1)
            asmat(4,icont+2)=amatx(4,icont+2) -ammat(icont+2)
            asmat(5,icont+1)=amatx(5,icont+1) -ammat(icont+1)
            asmat(5,icont+2)=amatx(5,icont+2) -ammat(icont+2)
c
c*****   bs matrix
            bsmat(1,icont+1)=bmatx(1,icont+1) -bmmat(icont+1)/3.0
            bsmat(1,icont+2)=bmatx(1,icont+2) -bmmat(icont+2)/3.0
            bsmat(2,icont+1)=bmatx(2,icont+1)
            bsmat(2,icont+2)=bmatx(2,icont+2)
            bsmat(3,icont+1)=bmatx(3,icont+1)
            bsmat(3,icont+2)=bmatx(3,icont+2)
            bsmat(4,icont+1)=bmatx(4,icont+1) -bmmat(icont+1)/3.0
            bsmat(4,icont+2)=bmatx(4,icont+2) -bmmat(icont+2)/3.0
            bsmat(5,icont+1)=bmatx(5,icont+1) -bmmat(icont+1)/3.0
            bsmat(5,icont+2)=bmatx(5,icont+2) -bmmat(icont+2)/3.0
20          icont=icont+2
10        continue
          return
          end
c
          subroutine matria(amatx, bmatx, asmat, bsmat,ammat, bmmat,
     *              radiu, shape, cartd, poiss, nnode ,ntype, key )
c************************************************
c       this is a routine to form a, b, as
c       bs, am, bm for axisymmetric problem using shape function n
c************************************************
          implicit double precision(a-h,o-z)
          dimension amatx(5,*), bmatx(5,* ), asmat(5,* ), bsmat(5,* ),
     *            ammat(*   ), bmmat(*   ), cartd(2,9 ), shape(*)
c
          icont=0
          do 10 inode=1,nnode
c
c****       a matrix
            amatx(1,icont+1)=0.0
            amatx(1,icont+2)= cartd(2,inode)
            amatx(1,icont+3)=shape(inode)/radiu
```

```
            amatx(2,icont+1)=-cartd(2,inode)
            amatx(2,icont+2)=0.0
            amatx(2,icont+3)=0.0
            amatx(3,icont+1)=0.0
            amatx(3,icont+2)=-cartd(1,inode)
            amatx(3,icont+3)=0.0
            amatx(4,icont+1)= cartd(1,inode)+shape(inode)/radiu
            amatx(4,icont+2)=0.0
            amatx(4,icont+3)=0.0
            amatx(5,icont+1)=0.0
            amatx(5,icont+2)=cartd(2,inode)
            amatx(5,icont+3)=cartd(1,inode)
c****    am matrix and
              ammat(icont+1)=(amatx(1,icont+1)+amatx(4,icont+1)
     *                       +amatx(5,icont+1))/3.0
              ammat(icont+2)=(amatx(1,icont+2)+amatx(4,icont+2)
     *                       +amatx(5,icont+2))/3.0
              ammat(icont+3)=(amatx(1,icont+3)+amatx(4,icont+3)
     *                       +amatx(5,icont+3))/3.0
c
c****  as matrix
c
            asmat(1,icont+1)=amatx(1,icont+1) -ammat(icont+1)
            asmat(1,icont+2)=amatx(1,icont+2) -ammat(icont+2)
            asmat(1,icont+3)=amatx(1,icont+3) -ammat(icont+3)
            asmat(2,icont+1)=amatx(2,icont+1)
            asmat(2,icont+2)=amatx(2,icont+2)
            asmat(2,icont+3)=amatx(2,icont+3)
            asmat(3,icont+1)=amatx(3,icont+1)
            asmat(3,icont+2)=amatx(3,icont+2)
            asmat(3,icont+3)=amatx(3,icont+3)
            asmat(4,icont+1)=amatx(4,icont+1) -ammat(icont+1)
            asmat(4,icont+2)=amatx(4,icont+2) -ammat(icont+2)
            asmat(4,icont+3)=amatx(4,icont+3) -ammat(icont+3)
            asmat(5,icont+1)=amatx(5,icont+1) -ammat(icont+1)
            asmat(5,icont+2)=amatx(5,icont+2) -ammat(icont+2)
            asmat(5,icont+3)=amatx(5,icont+3) -ammat(icont+3)
10        icont=icont+3
          icont=0
          do 20 inode=1,nnode
c
c*****    b matrix
            bmatx(1,icont+1)= cartd(1,inode)
            bmatx(1,icont+2)= 0.0
            bmatx(2,icont+1)=0.5*cartd(2,inode)
            bmatx(2,icont+2)=0.5*cartd(1,inode)
            bmatx(3,icont+1)=0.5*cartd(2,inode)
            bmatx(3,icont+2)=0.5*cartd(1,inode)
            bmatx(4,icont+1)=0.0
            bmatx(4,icont+2)= cartd(2,inode)
            bmatx(5,icont+1)=shape(inode)/radiu
            bmatx(5,icont+2)=0.0
            bmmat(icont+1)=bmatx(1,icont+1)+bmatx(4,icont+1)
     *                     +bmatx(5,icont+1)
            bmmat(icont+2)=bmatx(1,icont+2)+bmatx(4,icont+2)
     *                     +bmatx(5,icont+2)
c
c*****   bs matrix
            bsmat(1,icont+1)=bmatx(1,icont+1) -bmmat(icont+1)/3.0
            bsmat(1,icont+2)=bmatx(1,icont+2) -bmmat(icont+2)/3.0
            bsmat(2,icont+1)=bmatx(2,icont+1)
            bsmat(2,icont+2)=bmatx(2,icont+2)
            bsmat(3,icont+1)=bmatx(3,icont+1)
            bsmat(3,icont+2)=bmatx(3,icont+2)
            bsmat(4,icont+1)=bmatx(4,icont+1) -bmmat(icont+1)/3.0
            bsmat(4,icont+2)=bmatx(4,icont+2) -bmmat(icont+2)/3.0
            bsmat(5,icont+1)=bmatx(5,icont+1) -bmmat(icont+1)/3.0
```

```
            bsmat(5,icont+2)=bmatx(5,icont+2) -bmmat(icont+2)/3.0
20          icont=icont+2
          return
          end
c
        subroutine output (dispt,npoin,nelem,nstre,ngaus,ndofn,
     *                     gpcod,coord)
c*********************************************************************
c
c       ouput routine for displacements and streses
c
c*********************************************************************
        implicit double precision(a-h,o-z)
        dimension dispt(*),gpcod(2,*),coord(npoin,*)
        common stres(5,2700)
        mgaus=ngaus*ngaus
        kgaus=nelem*mgaus
        write(8,900)
900     format(3x,'node:', 8x,'displacement=',10x,'stress vector='/)
        do 600 ipoin=1,npoin
          in1=(ipoin-1)*ndofn+1
          in2=(ipoin-1)*ndofn+2
          in3=(ipoin-1)*ndofn+3
          in4=(ipoin-1)*ndofn+4
          in5=(ipoin-1)*ndofn+5
c        write(10,911)coord(ipoin,2),dispt(in2)
        if(ndofn.eq.4)
     *  write(8,910)ipoin,dispt(in1),dispt(in2),dispt(in3),dispt(in4)
        if(ndofn.eq.5)
     *  write(8,910)ipoin,dispt(in1),dispt(in2),dispt(in3),dispt(in4)
     *             ,dispt(in5)
        write(13,911)dispt(in1),dispt(in2)
910     format(1x,i5, 2e12.4,2x,'|',3e12.4)
911     format(1x,2e15.4)
600     continue
c
        write(8,920)
920     format(/10x,'**********element stresses*********'//,
     *          5x,'stress-xx',5x,'stress-xy',5x,'stress-yx',
     *          5x,'stress-yy',5x,'stress-zz'/)
        ielem=0
        do 100 igaus=1,kgaus
             write(4,980)gpcod(1,igaus),gpcod(2,igaus),
     *                  (stres(istre,igaus),istre=1,nstre)
980          format(1x,7e15.6)
110       continue
          if((igaus-1)/mgaus*mgaus.eq.(igaus-1)) then
            ielem=ielem+1
            write(8,940)ielem
940         format(/15x,'ielem=:',i4/)
          end if
          write(8,930)(stres(istre,igaus),istre=1,nstre)
930       format(1x,5e14.5)
100     continue
        return
        end
c
        subroutine estifn(estif,amatx,bmatx,asmat,bsmat,ammat,bmmat,
     *                    nnode,shear,xkcon,dvolu,kgaus,ndofn,ntype,
     *                    ldd  ,lss  ,lds  ,ps ,iiter, trial, factr)
c*************************************************
c       this routine is to form a stiffness
c       matrix at each gauss point.  when iiter=1 for elstic
c                                    when iiter>1 for rigid plastic
c*************************************************
        implicit double precision(a-h,o-z)
        double precision kmatx(40,40),ldd(5,5),lss(5,5),lds(5,5)
```

```
      dimension amatx(5,* ), bmatx(5,* ), asmat(5,* ), bsmat(5,* ),
     *          ammat(*   ), bmmat(*   ), estif(*   )
c
      ndof1=2
      ndof2=2
      if(ntype.eq.3)ndof2=3
      nstre=5
      plt=50.0
      g1=shear*2.0
      nevab=ndofn*nnode
    alfa=2.0*shear/xkcon
      do 1000 inode=1,nnode
         do 1000 jnode=1,nnode
c
c*****    bs^*ldd*bs + 2p*bm^bm or 2G(bs^bs + k/2G*bm^*bm)
         do 10 idofn=1,ndof1
           ivar=(inode-1)*ndof1+idofn
           igar=(inode-1)*ndofn+idofn
           do 10 jdofn=1,ndof1
             jvar=(jnode-1)*ndof1+jdofn
             jgar=(jnode-1)*ndofn+jdofn
             kmatx(igar,jgar)=0.0
             if(iiter.eq.1.or.ps.lt.0.0) then
               do 21 i=1,nstre
                 kmatx(igar,jgar)=kmatx(igar,jgar)+bsmat(i,ivar)
     *                           *bsmat(i,jvar)
21             continue
               kmatx(igar,jgar)=kmatx(igar,jgar)+
     *                          bmmat(ivar)*bmmat(jvar)/alfa
            else
               do 20 i=1,nstre
               do 20 j=1,nstre
                 kmatx(igar,jgar)=kmatx(igar,jgar)+bsmat(i,ivar)
     *                           *ldd(i,j)*bsmat(j,jvar)
20             continue
               kmatx(igar,jgar)=kmatx(igar,jgar)+
     *          bmmat(ivar)*bmmat(jvar)*2.0*plt
             end if
10         continue
c
c*****       as^{ldd/2G/2G-lds/2G+lss}*as+2pam^am/k/k or as^*as/2G +
am^*am/k
           do 30 idofn=1,ndof2
             ivar=(inode-1)*ndof2+idofn
             igar=(inode-1)*ndofn+2+idofn
             do 30 jdofn=1,ndof2
               jvar=(jnode-1)*ndof2+jdofn
               jgar=(jnode-1)*ndofn+2+jdofn
               kmatx(igar,jgar)=0.0
               if(iiter.eq.1.or.ps.lt.0.0) then
                 do 41 i=1,nstre
                   kmatx(igar,jgar)=kmatx(igar,jgar)+asmat(i,ivar)
     *                           *asmat(i,jvar)/factr/factr
41               continue
                 kmatx(igar,jgar)=kmatx(igar,jgar)+ammat(ivar)
     *                          *ammat(jvar)*alfa/factr/factr
     *                         +(asmat(2,ivar)-asmat(3,ivar))
     *                       *(asmat(2,jvar)-asmat(3,jvar))*trial
               else
                 do 40 i=1,nstre
                 do 40 j=1,nstre
                   tmp=(lss(i,j)*g1-2.0*lds(i,j))*g1+ldd(i,j)
                   kmatx(igar,jgar)=kmatx(igar,jgar)+asmat(i,ivar)
     *                           *tmp*asmat(j,jvar)
40               continue
                 kmatx(igar,jgar)=kmatx(igar,jgar)+2.*plt/factr/factr
     *                           *ammat(ivar)*ammat(jvar)*alfa*alfa
```

```
     *                              +(asmat(2,ivar)-asmat(3,ivar))
     *                     *(asmat(2,jvar)-asmat(3,jvar))*trial*g1
                end if
30           continue
c
c****      bs^{-ldd/2G+lds}as-2pbm^am/k or b^*a
              do 50 idofn=1,ndof1
                ivar=(inode-1)*ndof1+idofn
                igar=(inode-1)*ndofn+idofn
                do 50 jdofn=1,ndof2
                  jvar=(jnode-1)*ndof2+jdofn
                  jgar=(jnode-1)*ndofn+2+jdofn
                  kmatx(igar,jgar)=0.0
                  if(iiter.eq.1.or.ps.lt.0.0) then
                    do 61 i=1,nstre
                      kmatx(igar,jgar)=kmatx(igar,jgar)-bmatx(i,ivar)
     *                                 *amatx(i,jvar)/factr
61                  continue
                  else
                    do 60 i=1,nstre
                    do 60 j=1,nstre
                      tmp=-ldd(i,j)+lds(i,j)*g1
                      kmatx(igar,jgar)=kmatx(igar,jgar)+bsmat(i,ivar)
     *                                 *tmp*asmat(j,jvar)
60                  continue
                    kmatx(igar,jgar)=kmatx(igar,jgar)-2.0*plt*alfa
     *                              *bmmat(ivar)*ammat(jvar)/factr
                  end if
                  kmatx(jgar,igar)=kmatx(igar,jgar)
50              continue
1000     continue
c
c****     transfer k to 1-d matrix
          kount=0
          do 2000 ievab=1,nevab
            do 2000 jevab=ievab,nevab
              kount=kount+1
            if(iiter.eq.1.or.ps.lt.0.0) then
              estif(kount)=estif(kount)+kmatx(ievab,jevab)*dvolu*g1
            else
              estif(kount)=estif(kount)+kmatx(ievab,jevab)*dvolu
            end if
2000      continue
         return
         end
c
         subroutine boundp(stifi,maxai,ifpre,nbdis,presc,nline,thick,
     *                     bline,ndofn,rho,coord,npoin,force,dispt,shear)
c*******************************************************
c
c        impose  conditions of displacements
c        and stresses for penalty function
c
c*******************************************************
         implicit double precision (a-h,o-z)
         parameter (pent=1.0e30)
         dimension stifi(*), ifpre(ndofn,*),maxai(*),idrec(2),
     *             presc(3,*), bline(4,*),coord(npoin,2),force(*),
     *             tmatx(4,4),vectr(4),dispt(*)
           alfa=2.0*shear
c
c****     displacement condition
c
         do 10 ibdis=1,nbdis
            ipoin=presc(1,ibdis)
            idrec(1)=(presc(1,ibdis)-ipoin+0.05)*10
            tmp=idrec(1)/10.0
```

```
            idrec(2)=(presc(1,ibdis)-ipoin-tmp+0.005)*100
            do 100 i=1,2
              if(idrec(i).ne.0) then
                ieqns=iabs(ifpre(i,ipoin))
                ifpre(i,ipoin)=-ieqns
                idres=maxai(ieqns)
                stifi(idres)=stifi(idres)+2.0*rho
                force(ieqns)=force(ieqns)+2.0*rho*(presc(i+1,ibdis)
     *                      -dispt(ieqns))
              end if
100         continue
10        continue
c
c******   stress conditions
c
          do 20  iline=1,nline
            ipot1=bline(1,iline)
            ipot2=bline(2,iline)
            if(ifpre(3,ipot1).lt.ifpre(3,ipot2)) then
              ieqn1=ifpre(3,ipot1)
              ieqn2=ifpre(3,ipot2)
              cord1x=coord(ipot1,1)
              cord1y=coord(ipot1,2)
              cord2x=coord(ipot2,1)
              cord2y=coord(ipot2,2)
            else
              ieqn1=ifpre(3,ipot2)
              ieqn2=ifpre(3,ipot1)
              cord1x=coord(ipot2,1)
              cord1y=coord(ipot2,2)
              cord2x=coord(ipot1,1)
              cord2y=coord(ipot1,2)
            end if
c
            do 200 i=1,2
              if(ifpre(i,ipot1).gt.0.or.ifpre(i,ipot2).gt.0) then
               call transp(cord1x,cord1y,cord2x,cord2y,tmatx,vectr,i,i1)
               if(ntype.eq.1)pload=bline(2+i1,iline)/thick/alfa
               if(ntype.eq.2)pload=bline(2+i1,iline)/alfa
c ***     for right-hand side
               irow1=ieqn1
               irow2=ieqn1+1
               irow3=ieqn2
               irow4=ieqn2+1
               pr1=tmatx(1,1)*dispt(irow1)+tmatx(1,2)*dispt(irow2)
     *            +tmatx(1,3)*dispt(irow3)+tmatx(1,4)*dispt(irow4)
               pr2=tmatx(2,1)*dispt(irow1)+tmatx(2,2)*dispt(irow2)
     *            +tmatx(2,3)*dispt(irow3)+tmatx(2,4)*dispt(irow4)
               pr3=tmatx(3,1)*dispt(irow1)+tmatx(3,2)*dispt(irow2)
     *            +tmatx(3,3)*dispt(irow3)+tmatx(3,4)*dispt(irow4)
               pr4=tmatx(4,1)*dispt(irow1)+tmatx(4,2)*dispt(irow2)
     *            +tmatx(4,3)*dispt(irow3)+tmatx(4,4)*dispt(irow4)
c ***     for stiffness matrix
               idr11=maxai(ieqn1)
               idr12=maxai(ieqn1+1)+1
               idr13=maxai(ieqn2)+(ieqn2-ieqn1)
               idr14=maxai(ieqn2+1)+(ieqn2-ieqn1)+1
               idr22=idr12-1
               idr23=idr13-1
               idr24=idr14-1
               idr33=maxai(ieqn2)
               idr34=maxai(ieqn2+1)+1
               idr44=idr34-1
c
               stifi(idr11)=stifi(idr11)+2.0*rho*tmatx(1,1)
               stifi(idr12)=stifi(idr12)+2.0*rho*tmatx(1,2)
               stifi(idr13)=stifi(idr13)+2.0*rho*tmatx(1,3)
```

```
              stifi(idr14)=stifi(idr14)+2.0*rho*tmatx(1,4)
              stifi(idr22)=stifi(idr22)+2.0*rho*tmatx(2,2)
              stifi(idr23)=stifi(idr23)+2.0*rho*tmatx(2,3)
              stifi(idr24)=stifi(idr24)+2.0*rho*tmatx(2,4)
              stifi(idr33)=stifi(idr33)+2.0*rho*tmatx(3,3)
              stifi(idr34)=stifi(idr34)+2.0*rho*tmatx(3,4)
              stifi(idr44)=stifi(idr44)+2.0*rho*tmatx(4,4)
c
              force(irow1)=force(irow1)+2.0*rho*(pload*vectr(1)-pr1)
              force(irow2)=force(irow2)+2.0*rho*(pload*vectr(2)-pr2)
              force(irow3)=force(irow3)+2.0*rho*(pload*vectr(3)-pr3)
              force(irow4)=force(irow4)+2.0*rho*(pload*vectr(4)-pr4)
            end if
200       continue
20      continue
        ic1=ifpre(3,npoin)
        ic2=ifpre(4,npoin)
        stifi(maxai(ic1))=stifi(maxai(ic1))+pent
        stifi(maxai(ic2))=stifi(maxai(ic2))+pent
        return
        end
c
        subroutine bounda(stifi,maxai,ifpre,nbdis,presc,nline,thick,
     *                  bline,ndofn,rho,coord,npoin,force,dispt,shear)
c*******************************************************
c
c       impose  conditions of displacements
c       and stresses for penalty function
c
c*******************************************************
        implicit double precision (a-h,o-z)
        parameter (pent=1.0e30)
        dimension stifi(*), ifpre(ndofn,*),maxai(*),idrec(2),
     *            presc(3,*), bline(4,*),coord(npoin,2),force(*),
     *            tmatx(6,6),vectr(6),dispt(*)
c
         twopi=6.283185307179586
          alfa=2.0*shear
c
c****    displacement condition
c
        do 10 ibdis=1,nbdis
           ipoin=presc(1,ibdis)
           rad=coord(ipoin,1)
           idrec(1)=(presc(1,ibdis)-ipoin+0.05)*10
           tmp=idrec(1)/10.0
           idrec(2)=(presc(1,ibdis)-ipoin-tmp+0.005)*100
           do 100 i=1,2
             if(idrec(i).ne.0) then
               ieqns=iabs(ifpre(i,ipoin))
               ifpre(i,ipoin)=-ieqns
               idres=maxai(ieqns)
               stifi(idres)=stifi(idres)+2.0*rho
               force(ieqns)=force(ieqns)+2.0*rho*
     *                      (presc(i+1,ibdis)-dispt(ieqns))
             end if
100        continue
10      continue
c
c******   stress conditions
c
        do 20  iline=1,nline
          ipot1=bline(1,iline)
          ipot2=bline(2,iline)
          if(ifpre(3,ipot1).lt.ifpre(3,ipot2)) then
            ieqn1=ifpre(3,ipot1)
            ieqn2=ifpre(3,ipot2)
```

```
            cord1x=coord(ipot1,1)
            cord1y=coord(ipot1,2)
            cord2x=coord(ipot2,1)
            cord2y=coord(ipot2,2)
          else
            ieqn1=ifpre(3,ipot2)
            ieqn2=ifpre(3,ipot1)
            cord1x=coord(ipot2,1)
            cord1y=coord(ipot2,2)
            cord2x=coord(ipot1,1)
            cord2y=coord(ipot1,2)
          end if
c
          do 200 i=1,2
            if(ifpre(i,ipot1).gt.0.or.ifpre(i,ipot2).gt.0) then
            call transa(cord1x,cord1y,cord2x,cord2y,tmatx,vectr,i, i1)
            pload=bline(2+i1,iline)/alfa
c ***     for right-hand side
               irow1=ieqn1
               irow2=ieqn1+1
               irow3=ieqn1+2
               irow4=ieqn2
               irow5=ieqn2+1
               irow6=ieqn2+2
               pr1=tmatx(1,1)*dispt(irow1)+tmatx(1,2)*dispt(irow2)
     *            +tmatx(1,3)*dispt(irow3)+tmatx(1,4)*dispt(irow4)
     *            +tmatx(1,5)*dispt(irow5)+tmatx(1,6)*dispt(irow6)
               pr2=tmatx(2,1)*dispt(irow1)+tmatx(2,2)*dispt(irow2)
     *            +tmatx(2,3)*dispt(irow3)+tmatx(2,4)*dispt(irow4)
     *            +tmatx(2,5)*dispt(irow5)+tmatx(2,6)*dispt(irow6)
               pr3=tmatx(3,1)*dispt(irow1)+tmatx(3,2)*dispt(irow2)
     *            +tmatx(3,3)*dispt(irow3)+tmatx(3,4)*dispt(irow4)
     *            +tmatx(3,5)*dispt(irow5)+tmatx(3,6)*dispt(irow6)
               pr4=tmatx(4,1)*dispt(irow1)+tmatx(4,2)*dispt(irow2)
     *            +tmatx(4,3)*dispt(irow3)+tmatx(4,4)*dispt(irow4)
     *            +tmatx(4,5)*dispt(irow5)+tmatx(4,6)*dispt(irow6)
               pr5=tmatx(5,1)*dispt(irow1)+tmatx(5,2)*dispt(irow2)
     *            +tmatx(5,3)*dispt(irow3)+tmatx(5,4)*dispt(irow4)
     *            +tmatx(5,5)*dispt(irow5)+tmatx(5,6)*dispt(irow6)
               pr6=tmatx(6,1)*dispt(irow1)+tmatx(6,2)*dispt(irow2)
     *            +tmatx(6,3)*dispt(irow3)+tmatx(6,4)*dispt(irow4)
     *            +tmatx(6,5)*dispt(irow5)+tmatx(6,6)*dispt(irow6)
c ***     for stiffness matrix
               idr11=maxai(ieqn1)
               idr12=maxai(ieqn1+1)+1
               idr13=maxai(ieqn1+2)+2
               idr14=maxai(ieqn2)+(ieqn2-ieqn1)
               idr15=maxai(ieqn2+1)+(ieqn2-ieqn1)+1
               idr16=maxai(ieqn2+2)+(ieqn2-ieqn1)+2
               idr22=idr12-1
               idr23=idr13-1
               idr24=idr14-1
               idr25=idr15-1
               idr26=idr16-1
               idr33=idr13-2
               idr34=idr14-2
               idr35=idr15-2
               idr36=idr16-2
               idr44=maxai(ieqn2)
               idr45=maxai(ieqn2+1)+1
               idr46=maxai(ieqn2+2)+2
               idr55=idr45-1
               idr56=idr46-1
               idr66=idr46-2
c
               stifi(idr11)=stifi(idr11)+2.0*rho*tmatx(1,1)
               stifi(idr12)=stifi(idr12)+2.0*rho*tmatx(1,2)
```

```
                stifi(idr13)=stifi(idr13)+2.0*rho*tmatx(1,3)
                stifi(idr14)=stifi(idr14)+2.0*rho*tmatx(1,4)
                stifi(idr15)=stifi(idr15)+2.0*rho*tmatx(1,5)
                stifi(idr16)=stifi(idr16)+2.0*rho*tmatx(1,6)
c
                stifi(idr22)=stifi(idr22)+2.0*rho*tmatx(2,2)
                stifi(idr23)=stifi(idr23)+2.0*rho*tmatx(2,3)
                stifi(idr24)=stifi(idr24)+2.0*rho*tmatx(2,4)
                stifi(idr25)=stifi(idr25)+2.0*rho*tmatx(2,5)
                stifi(idr26)=stifi(idr26)+2.0*rho*tmatx(2,6)
c
                stifi(idr33)=stifi(idr33)+2.0*rho*tmatx(3,3)
                stifi(idr34)=stifi(idr34)+2.0*rho*tmatx(3,4)
                stifi(idr35)=stifi(idr35)+2.0*rho*tmatx(3,5)
                stifi(idr36)=stifi(idr36)+2.0*rho*tmatx(3,6)
c
                stifi(idr44)=stifi(idr44)+2.0*rho*tmatx(4,4)
                stifi(idr45)=stifi(idr45)+2.0*rho*tmatx(4,5)
                stifi(idr46)=stifi(idr46)+2.0*rho*tmatx(4,6)
c
                stifi(idr55)=stifi(idr55)+2.0*rho*tmatx(5,5)
                stifi(idr56)=stifi(idr56)+2.0*rho*tmatx(5,6)
c
                stifi(idr66)=stifi(idr66)+2.0*rho*tmatx(6,6)
c
                force(irow1)=force(irow1)+2*rho*(pload*vectr(1)-pr1)
                force(irow2)=force(irow2)+2*rho*(pload*vectr(2)-pr2)
                force(irow3)=force(irow3)+2*rho*(pload*vectr(3)-pr3)
                force(irow4)=force(irow4)+2*rho*(pload*vectr(4)-pr4)
                force(irow5)=force(irow5)+2*rho*(pload*vectr(5)-pr5)
                force(irow6)=force(irow6)+2*rho*(pload*vectr(6)-pr6)
              end if
200         continue
20        continue
          ic1=ifpre(3,npoin)
          ic2=ifpre(4,npoin)
          ic3=ifpre(5,npoin)
c          stifi(maxai(ic1))=stifi(maxai(ic1))+pent
          stifi(maxai(ic2))=stifi(maxai(ic2))+pent
c          stifi(maxai(ic3))=stifi(maxai(ic3))+pent
          return
          end
c
      subroutine decomp (stiff ,maxai ,neqns ,ishot )
c*******************************************************
c
c     factorises (l)*(d)*(l) transpose of stiffness matrix
c
c*******************************************************
      implicit double precision(a-h,o-z)
      dimension stiff(*) ,maxai(*)
c
      if(neqns.eq.1) return
      do 200 ieqns=1,neqns
      imaxa=maxai(ieqns)
      lower=imaxa+1
      kuper=maxai(ieqns+1)-1
      khigh=kuper-lower
      if(khigh) 304,240,210
210   ksize=ieqns-khigh
      icoun=0
      juper=kuper
      do 260 jhigh=1,khigh
      icoun=icoun+1
      juper=juper-1
      kmaxa=maxai(ksize)
      ndiag=maxai(ksize+1)-kmaxa-1
```

```
      if(ndiag) 260,260,270
270   ncolm=min0(icoun,ndiag)
      count=0.
      do 280 icolm=1,ncolm
280   count=count+stiff(kmaxa+icolm)*stiff(juper+icolm)
      stiff(juper)=stiff(juper)-count
260   ksize=ksize+1
240   ksize=ieqns
      bsumm=0.
      do 300 icolm=lower,kuper
      ksize=ksize-1
      jmaxa=maxai(ksize)
      ratio=stiff(icolm)/stiff(jmaxa)
      bsumm=bsumm+ratio*stiff(icolm)
300   stiff(icolm)=ratio
      stiff(imaxa)=stiff(imaxa)-bsumm
304   if(stiff(imaxa)) 310,310,200
310   if(ishot.eq.0) go to 320
      if(stiff(imaxa).eq.0) stiff(imaxa)=-1.e-16
      go to 200
320   write(8,2000) ieqns,stiff(imaxa)
      stop
200   continue
      return
2000  format(//48h stop - stiffness matrix not positive definite ,//
     *32h nonpositive pivot for equation ,i4,//10h pivot =  ,e20.12 )
      end
c
       subroutine redbak (stiff ,force ,maxai ,neqns)
c*************************************************************
c
c      to reduce and back_substitute iteration vector
c
c*************************************************************
      implicit double precision(a-h,o-z)
       dimension stiff(*) ,force(*) ,maxai(*)
c
       do 400 ieqns=1,neqns
       lower=maxai(ieqns)+1
       kuper=maxai(ieqns+1)-1
       if(kuper-lower) 400,410,410
410    jeqns=ieqns
       sumcc=0.0
       do 420 icolm=lower,kuper
       jeqns=jeqns-1
420    sumcc=sumcc+stiff(icolm)*force(jeqns)
       force(ieqns)=force(ieqns)-sumcc
400    continue
c
       do 480 ieqns=1,neqns
       kmaxa=maxai(ieqns)
480    force(ieqns)=force(ieqns)/stiff(kmaxa)
       if(neqns.eq.1) return
       jeqns=neqns
       DO 500 IEQNS=2,NEQNS
       lower=maxai(jeqns)+1
       kuper=maxai(jeqns+1)-1
       if(kuper-lower) 500,510,510
510    keqns=jeqns
       do 520 icolm=lower,kuper
       keqns=keqns-1
520    force(keqns)=force(keqns)-stiff(icolm)*force(jeqns)
500    jeqns=jeqns-1
       return
       end
c
      subroutine addban (stiff,maxai,estif,leqns,nevab)
```

```
c*****************************************************************
c
c     assembly of total stiffness vector
c
c*****************************************************************
      implicit double precision(a-h,o-z)
      dimension stiff(*),maxai(*),estif(*),leqns(*)
c
      kount=0
      do 200 ievab=1,nevab
      ieqns=leqns(ievab)
      if(ieqns) 200,200,100
100   imaxa=maxai(ieqns)
      kevab=ievab
      do 220 jevab=1,nevab
      jeqns=leqns(jevab)
      if(jeqns) 220,220,110
110   ijeqn=ieqns-jeqns
      if(ijeqn) 220,210,210
210   isize=imaxa+ijeqn
      jsize=kevab
      if(jevab.ge.ievab) jsize=jevab+kount
      stiff(isize)=stiff(isize)+estif(jsize)
220   kevab=kevab+nevab-jevab
200   kount=kount+nevab-ievab
      return
      end
c
      subroutine colmht (mhigh ,nevab ,leqns )
c*************************************************************
c
c     evaluates the column height of stiffness matrix
c
c*************************************************************
      implicit double precision(a-h,o-z)
      dimension leqns(*) ,mhigh(*)
      maxam=100000
      do 100 ievab=1,nevab
      if(leqns(ievab)) 110,100,110
110   if(leqns(ievab)-maxam) 120,100,100
120   maxam=leqns(ievab)
100   continue
      do 200 ievab=1,nevab
      ieqns=leqns(ievab)
      if(ieqns.eq.0) go to 200
      jhigh=ieqns-maxam
      if(jhigh.gt.mhigh(ieqns)) mhigh(ieqns)=jhigh
200   continue
      return
      end
c
      subroutine addres (maxai ,mhigh ,neqns ,nwktl ,mkoun )
c*************************************************************
c
c     evaluates adresses of diagonal elements
c
c*************************************************************
      implicit double precision(a-h,o-z)
      dimension maxai(*) ,mhigh(*)
      neqnn=neqns+1
      do 20 ieqnn=1,neqnn
20    maxai(ieqnn)=0
      maxai(1)=1
      maxai(2)=2
      mkoun=0
      if(neqns.eq.1) go to 30
      do 10 ieqns=2,neqns
```

```
      if(mhigh(ieqns).gt.mkoun) mkoun=mhigh(ieqns)
10    maxai(ieqns+1)=maxai(ieqns)+mhigh(ieqns)+1
30    mkoun=mkoun+1
      nwktl=maxai(neqns+1)-maxai(1)
      return
      end
c
      subroutine sfr2(deriv,nnode,shape,exisp,etasp)
c*******************************************************************
c
c**** this subroutine evaluates shape functions and their derivatives
c     for linear,quadratic lagrangian and serendipity
c     isoparametric 2-d elements
c
c*******************************************************************
      implicit double precision(a-h,o-z)
      dimension deriv(2,9) ,shape(9)
      s=exisp
      t=etasp
      if(nnode.gt.4) go to 10
      st=s*t
c
c***  shape functions for 4 nodes element
c
      shape(1)=(1-t-s+st)*0.25
      shape(2)=(1-t+s-st)*0.25
      shape(3)=(1+t+s+st)*0.25
      shape(4)=(1+t-s-st)*0.25
c
c***  shape functions derivatives
c
      deriv(1,1)=(-1+t)*0.25
      deriv(1,2)=(+1-t)*0.25
      deriv(1,3)=(+1+t)*0.25
      deriv(1,4)=(-1-t)*0.25
      deriv(2,1)=(-1+s)*0.25
      deriv(2,2)=(-1-s)*0.25
      deriv(2,3)=(+1+s)*0.25
      deriv(2,4)=(+1-s)*0.25
      return
10    if(nnode.gt.8) go to 30
      s2=s*2.0
      t2=t*2.0
      ss=s*s
      tt=t*t
      st=s*t
      sst=s*s*t
      stt=s*t*t
      st2=s*t*2
c
c***  shape functions for 8 noded element
c
      shape(1)=(-1.0+st+ss+tt-sst-stt)/4.0
      shape(2)=(1.0-t-ss+sst)/2.0
      shape(3)=(-1.0-st+ss+tt-sst+stt)/4.0
      shape(4)=(1.0+s-tt-stt)/2.0
      shape(5)=(-1.0+st+ss+tt+sst+stt)/4.0
      shape(6)=(1.0+t-ss-sst)/2.0
      shape(7)=(-1.0-st+ss+tt+sst-stt)/4.0
      shape(8)=(1.0-s-tt+stt)/2.0
c
c***  shape function derivatives
c
      deriv(1,1)=(t+s2-st2-tt)/4.0
      deriv(1,2)=-s+st
      deriv(1,3)=(-t+s2-st2+tt)/4.0
      deriv(1,4)=(1.0-tt)/2.0
```

```
      deriv(1,5)=(t+s2+st2+tt)/4.0
      deriv(1,6)=-s-st
      deriv(1,7)=(-t+s2+st2-tt)/4.0
      deriv(1,8)=(-1.0+tt)/2.0
      deriv(2,1)=(s+t2-ss-st2)/4.0
      deriv(2,2)=(-1.0+ss)/2.0
      deriv(2,3)=(-s+t2-ss+st2)/4.0
      deriv(2,4)=-t-st
      deriv(2,5)=(s+t2+ss+st2)/4.0
      deriv(2,6)=(1.0-ss)/2.0
      deriv(2,7)=(-s+t2+ss-st2)/4.0
      deriv(2,8)=-t+st
      return
30    continue
      ss=s*s
      st=s*t
      tt=t*t
      s1=s+1.0
      t1=t+1.0
      s2=s*2.0
      t2=t*2.0
      s9=s-1.0
      t9=t-1.0
c
c***  shape functions for 9 noded elemednt
c
      shape(1)=0.25*s9*st*t9
      shape(2)=0.5*(1.0-ss)*t*t9
      shape(3)=0.25*s1*st*t9
      shape(4)=0.5*s*s1*(1.0-tt)
      shape(5)=0.25*s1*st*t1
      shape(6)=0.5*(1.0-ss)*t*t1
      shape(7)=0.25*s9*st*t1
      shape(8)=0.5*s*s9*(1.0-tt)
      shape(9)=(1.0-ss)*(1.0-tt)
c
c***  shape function derivatives
c
      deriv(1,1)=0.25*t*t9*(-1.0+s2)
      deriv(1,2)=-st*t9
      deriv(1,3)=0.25*(1.0+s2)*t*t9
      deriv(1,4)=0.5*(1.0+s2)*(1.0-tt)
      deriv(1,5)=0.25*(1.0+s2)*t*t1
      deriv(1,6)=-st*t1
      deriv(1,7)=0.25*(-1.0+s2)*t*t1
      deriv(1,8)=0.5*(-1.0+s2)*(1.0-tt)
      deriv(1,9)=-s2*(1.0-tt)
      deriv(2,1)=0.25*(-1.0+t2)*s*s9
      deriv(2,2)=0.5*(1.0-ss)*(-1.0+t2)
      deriv(2,3)=0.25*s*s1*(-1.0+t2)
      deriv(2,4)=-st*s1
      deriv(2,5)=0.25*s*s1*(1.0+t2)
      deriv(2,6)=0.5*(1.0-ss)*(1.0+t2)
      deriv(2,7)=0.25*s*s9*(1.0+t2)
      deriv(2,8)=-st*s9
      deriv(2,9)=-t2*(1.0-ss)
20    continue
      return
      end
c
      subroutine jacob2(cartd,deriv,djacb,elcod,gpcod,ielem,kgaus,
     *                  nnode,shape)
c*****************************************************************
c
c***  this subroutine evaluates the jacobian matrix and the cartestian
c     shape functon derivatives
c
```

```
c**********************************************************************
      implicit double precision(a-h,o-z)
      dimension cartd(2,9),deriv(2,9),elcod(2,9),gpcod(2,*),shape(9),
     *          xjaci(2,2),xjacm(2,2)
c
c***  calculate coordinates of sampling point
c
      do 2 idime=1,2
      gpcod(idime,kgaus)=0.0
      do 2 inode=1,nnode
      gpcod(idime,kgaus)=gpcod(idime,kgaus)+elcod(idime,inode)
     *                  *shape(inode)
2     continue
c
c***  create jacobian matrix xjacm
c
      do 4 idime=1,2
      do 4 jdime=1,2
      xjacm(idime,jdime)=0.0
      do 4 inode=1,nnode
      xjacm(idime,jdime)=xjacm(idime,jdime)+deriv(idime,inode)
     *                  *elcod(jdime,inode)
4     continue
c
c***  calculate determinant and inverse of jacobian matrix
c
      djacb=xjacm(1,1)*xjacm(2,2)-xjacm(1,2)*xjacm(2,1)
      if(djacb) 6,6,8
6     write(8,600) ielem,djacb
      stop
8     continue
      xjaci(1,1)=xjacm(2,2)/djacb
      xjaci(2,2)=xjacm(1,1)/djacb
      xjaci(1,2)=-xjacm(1,2)/djacb
      xjaci(2,1)=-xjacm(2,1)/djacb
c
c***  calculate cartesian derivatives
c
      do 10 idime=1,2
      do 10 inode=1,nnode
      cartd(idime,inode)=0.0
      do 10 jdime=1,2
      cartd(idime,inode)=cartd(idime,inode)+xjaci(idime,jdime)*
     *                  deriv(jdime,inode)
10    continue
600   format(//,36h program halted in subroutine jacob2,/,11x,
     *          22h zero or negative area,/5x,'element number',i5,
     *                                    5x,'djacb=',e12.5)
      return
      end
c
      subroutine gaussq (ngaus,posgp,weigp)
c**********************************************************************
c
c*** this subroutine sets up the gauss-legendre integration constants
c
c**********************************************************************
      implicit double precision(a-h,o-z)
      dimension posgp(4) ,weigp(4)
      if(ngaus.gt.2) go to 4
2     posgp(1)=-0.577350269189626
      weigp(1)=1.0
      go to 6
4     posgp(1)=-0.77459666924183
      posgp(2)=0.0
      weigp(1)=0.555555555555556
      weigp(2)=0.888888888888889
```

```
6     kgaus=ngaus/2
      do 8 igash=1,kgaus
      jgash=ngaus+1-igash
      posgp(jgash)=-posgp(igash)
      weigp(jgash)=weigp(igash)
8     continue
      return
      end
c
      subroutine nodxyr (coord,lnods,nelem,nnode,npoin,nrads,ntype)
c*****************************************************************
c
c*** interpolation of midside and center nodes
c
c*****************************************************************
      implicit double precision(a-h,o-z)
      dimension  coord(npoin,*),lnods(nelem,*)
c
      if(ntype.ne.3.or.nrads.eq.0) go to 40
cc
c***  change polar coordinates to cartisian
      do 50 ipoin=1,npoin
      raddi=coord(ipoin,1)
      theta=coord(ipoin,2)
      theta=0.017453292*theta
      coord(ipoin,1)=raddi*dsin(theta)
50    coord(ipoin,2)=raddi*dcos(theta)
c
40    if(nnode.eq.4) return
c
      lnode=nnode-1
      do 30 ielem=1,nelem
c*** loop over each element edge
      do 20 inode=1,nnode,2
      if(inode.eq.9) go to 20
c***  compute the node number of the first node
      nodst=lnods(ielem,inode)
      igash=inode+2
      if(igash.gt.lnode) igash=1
c***  compute the node number of the last node
      nodfn=lnods(ielem,igash)
      midpt=inode+1
c***  compute the node number of the intermediate node
      nodmd=lnods(ielem,midpt)
      total=dabs(coord(nodmd,1))+dabs(coord(nodmd,2))
c***  if the coordinates of the intermediate node are both zero
c     intermediate by a straight line
      if(total.gt.0.0) go to 20
      kount=1
10    coord(nodmd,kount)=(coord(nodst,kount)+coord(nodfn,kount))/2.0
      kount=kount+1
      if(kount.eq.2) go to 10
20    continue
30    continue
      return
      end
c
        subroutine transp(cord1x,cord1y,cord2x,cord2y,tmatx,vectr,
     *                    key, opt)
c*****************************************************************
c      form the stress boundary matrix for normal stress
c           and shear stress in "plane problem"
c*****************************************************************
        implicit double precisioN(A-H,O-Z)
        integer opt
        parameter (err=1.0e-5)
        dimension tmatx(4,4),vectr(4)
```

```
          r=dsqrt((cord1x-cord2x)**2+(cord1y-cord2y)**2)
          t=(cord2x-cord1x)/r
          s=(cord2y-cord1y)/r
 c*****         descide the normal and the tangent vector
          opt=key
 c        if(dabs(t).lt.err)opt=key
          if(dabs(s).lt.err)opt=3-key
          ss=s*s
          tt=t*t
          st=s*t
          if(opt.eq.1) then
            vectr(1)=-t
            vectr(2)=-s
            vectr(3)= t
            vectr(4)= s
            do 11 i=1,4
              do 11 j=1,4
                tmatx(i,j)=vectr(i)*vectr(j)/r
 11           continue
          end if
 c
          if(opt.eq.2) then
            vectr(1)=-s
            vectr(2)= t
            vectr(3)= s
            vectr(4)=-t
            do 12 i=1,4
              do 12 j=1,4
                tmatx(i,j)=vectr(i)*vectr(j)/r
 12           continue
          end if
          return
          end
 c
          subroutine transa(cord1x,cord1y,cord2x,cord2y,tmatx,vectr,
      *                     key, opt )
 c*****************************************************************
 c       form the stress boundary matrix for normal stress
 c            and shear stress in "axisymmetric prroblem"
 c*****************************************************************
          implicit double precision(a-h,o-z)
          integer opt
          double precision l
          parameter (err=1.0e-5,pent=1.0e20)
          dimension tmatx(6,6),v(6),vectr(6),p(4),w(4)
          twopi=6.283185307179586
          ng=3
          call gaussq(ng,p,w)
          r=dsqrt((cord1x-cord2x)**2+(cord1y-cord2y)**2)
          t=(cord2x-cord1x)/r
          s=(cord2y-cord1y)/r
 c*****         descide the normal and the tangent vector
          opt=key
          if(dabs(s).lt.err)opt=3-key
          ss=s*s
          tt=t*t
          st=s*t
    do 20 i=1,6
            vectr(i)=0.0
            do 20 j=1,6
 20       tmatx(i,j)=0.0
 c
          do 10 ig=1,ng
            l=0.5*p(ig)
            x=(cord1x+cord2x)/2.0+(cord2x-cord1x)*l
            djcb=0.5*twopi*r*w(ig)
      if(dabs(x).lt.err) then
```

```
            ax=pent
          else
            ax=1.0/x
          end if
          if(opt.eq.1) then
            v(1)=-t/r*x+tt*(0.5-l)
            v(2)=-s/r*x
            v(3)= ss*(0.5-l)
            v(4)= t/r*x+tt*(0.5+l)
            v(5)= s/r*x
            v(6)= ss*(0.5+l)
            do 11 i=1,6
              vectr(i)=vectr(i)+v(i)*djcb
              do 11 j=1,6
11          tmatx(i,j)=tmatx(i,j)+v(i)*v(j)*djcb*ax
          end if
c
          if(opt.eq.2) then
            v(1)=-s/r*x+st*(0.5-l)
            v(2)= t/r*x
            v(3)=-st*(0.5-l)
            v(4)= s/r*x+st*(0.5+l)
            v(5)=-t/r*x
            v(6)=-st*(0.5+l)
            do 12 i=1,6
              vectr(i)=vectr(i)+v(i)*djcb
              do 12 j=1,6
12          tmatx(i,j)=tmatx(i,j)+v(i)*v(j)*djcb*ax
          end if
10  continue
        return
        end
c
        subroutine mvalue (lnods, coord, x    ,  nnode, nevab, ndofn,
     *                     np,    ne,    ndime, ntype, nstre,  ngaus,
     *                     posgp,val,   props, nmats, matno, weigp,
     *                     presc, bline, ifpre, nbdis, nline,rho,iiter)
c*******************************************************
c*
c*      calculate m from obtained stranes  and
c*          stresses for each gauss point
c*
c*******************************************************
        implicit double precision(a-h,o-z)
        double precision k
        parameter (nmax=4000,err=1.0e-5,pent=1.0e20)
       dimensioncoord(np,*),lnods(ne,*),elcod(2,9),cartd(2,9),shape(9),
     *           gpcod(2,2700),deriv(2,9),amatx(5,30),bsmat(5,20),
     *           am(30),bmatx(5,20),bm(20),asmat(5,30),dispt(nmax),
     *           posgp(*), weigp(*),props(nmats,5),matno(*),
     *           ifpre(ndofn,*),presc(3,*),bline(4,*),
     *           id(2),f(2),di(5),dit(5),si(5),sd(5)
        common stres(5,4500),strin(5,4500),strdd(5,4500)
        common /dfunc/pcom(nmax), xicom(nmax),  ncom
        plt=50.0
        do 100 j=1,ncom
          dispt(j)=pcom(j)+x*xicom(j)
100     continue
c
      twopi=6.283185307179586
      ndof1=2
      ndof2=2
      e0=props(1, 1)
      p0=props(1, 2)
      g0=e0/2.0/(1.0+p0)
      if(ntype.eq.3)ndof2=3
c
```

```
c***  loop over each element
        val=0.0
        kcont=0
        do 500 ie=1,ne
          lprop=matno(ie)
           young=props(lprop, 1)
           poiss=props(lprop, 2)
           thick=props(lprop, 3)
            y0 =props(lprop, 4)*0.816497
            y  =props(lprop, 5)*0.816497
           shear=young/2.0/(1.0+poiss)
          xkcon=young/3.0/(1.0-2.0*poiss)
c
c***  evaluate the coordinates of the element nodal points
c
          do 10 inode=1,nnode
            lnode=lnods(ie,inode)
            do 10 idime=1,ndime
10        elcod(idime,inode)=coord(lnode,idime)
c
          do 500 igaus=1,ngaus
          do 500 jgaus=1,ngaus
            kcont=kcont+1
            xpoin=posgp(igaus)
            ypoin=posgp(jgaus)
            call sfr2   (deriv,nnode,shape,xpoin,ypoin)
            call jacob2 (cartd,deriv,djacb,elcod,gpcod,
     *                   ie,kcont,nnode,shape)
            radiu=gpcod(1,kcont)
c
c***  evaluate the a,b,as,bs,am and bm matrices
c
        plasp=poiss
        if(ntype.ne.3) then
          call matrip(amatx, bmatx, asmat, bsmat,   am,    bm,
     *                radiu, shape, cartd, plasp, nnode, ntype, 0)
        else
          call matria(amatx, bmatx, asmat, bsmat,   am,    bm,
     *                radiu, shape, cartd, plasp, nnode, ntype, 0)
        end if
        do 25 istre=1,nstre
          stres(istre,kcont)=0.0
          sd(istre)=0.0
          do 21 jevab=1,nnode*ndof2
            jnode=(jevab-1)/ndof2+1
            jdofn=jevab+2-(jnode-1)*ndof2
            jpoin=lnods(ie,jnode)
            jpott=(jpoin-1)*ndofn+jdofn
            stres(istre,kcont)=stres(istre,kcont)+2.0*g0
     *                        *amatx(istre,jevab)*dispt(jpott)
            sd(istre)=sd(istre)+2.0*g0
     *               *amatx(istre,jevab)*xicom(jpott)*x
21        continue
          sigma=(stres(1,kcont)+stres(4,kcont)+stres(5,kcont))/3.0
          si(1)=stres(1,kcont)-sigma
          si(2)=stres(2,kcont)
          si(3)=stres(3,kcont)
          si(4)=stres(4,kcont)-sigma
          si(5)=stres(5,kcont)-sigma
          ss=dsqrt(times(si,si))
          ps0=ss/y0-1.0
          strin(istre,kcont)=0.0
          strdd(istre,kcont)=0.0
          do 20 ievab=1,nnode*ndof1
            inode=(ievab-1)/ndof1+1
            idofn=ievab-(inode-1)*ndof1
            ipoin=lnods(ie,inode)
```

```
              ipott=(ipoin-1)*ndofn+idofn
              strin(istre,kcont)=strin(istre,kcont)
     *                          +bmatx(istre,ievab)*dispt(ipott)
              strdd(istre,kcont)=strdd(istre,kcont)
     *                          +bmatx(istre,ievab)*xicom(ipott)*x
c     *                           +bmatx(istre,ievab)*dispt(ipott)
20          continue
25        continue
c
c ***     evaluate the residual strain on each Gauiss ponit
c
          if(iiter.eq.1.and.ps0.ge.0.0) goto 340
            strin(1,kcont)=strin(1,kcont)-(stres(1,kcont)-poiss*
     *                    (stres(4,kcont)+stres(5,kcont)))/young
            strin(2,kcont)=strin(2,kcont)-(stres(2,kcont)
     *                    +stres(3,kcont))/4.0/shear
            strin(3,kcont)=strin(2,kcont)
            strin(4,kcont)=strin(4,kcont)-(stres(4,kcont)-poiss*
     *                    (stres(1,kcont)+stres(5,kcont)))/young
            strin(5,kcont)=strin(5,kcont)-(stres(5,kcont)-poiss*
     *                    (stres(1,kcont)+stres(4,kcont)))/young
            if(ntype.eq.1) then
              strin(5,kcont)=strin(5,kcont)+poiss*(stres(1,kcont)
     *                      +stres(4,kcont))/young
            end if
c
c ***     evaluate the residual incremental strain on each Gauiss ponit
c
            strdd(1,kcont)=strdd(1,kcont)-(sd(1)-poiss*
     *                    (sd(4)+sd(5)))/young
            strdd(2,kcont)=strdd(2,kcont)-(sd(2)+sd(3))/4.0/shear
            strdd(3,kcont)=strdd(2,kcont)
            strdd(4,kcont)=strdd(4,kcont)-(sd(4)-poiss*
     *                    (sd(1)+sd(5)))/young
            strdd(5,kcont)=strdd(5,kcont)-(sd(5)-poiss*
     *                    (sd(1)+sd(4)))/young
            if(ntype.eq.1) then
              strdd(5,kcont)=strdd(5,kcont)+poiss*(sd(1)
     *                      +sd(4))/young
            end if
340       continue
c
          epcilt= strin(1,kcont)+strin(4,kcont)+strin(5,kcont)
          epcild= strdd(1,kcont)+strdd(4,kcont)+strdd(5,kcont)
          dit(1)=strin(1,kcont)-epcilt/3.0
          di(1)=strdd(1,kcont)-epcild/3.0
          dit(2)=strin(2,kcont)
          di(2)=strdd(2,kcont)
          dit(3)=strin(3,kcont)
          di(3)=strdd(3,kcont)
          dit(4)=strin(4,kcont)-epcilt/3.0
          di(4)=strdd(4,kcont)-epcild/3.0
          dit(5)=strin(5,kcont)-epcilt/3.0
          di(5)=strdd(5,kcont)-epcild/3.0
c
          k=y0+y*dsqrt(times(dit,dit))
          dd=dsqrt(times(di,di))
          ps=ss/k-1.0
          if(ps.ge.0.0)then
            val=val+(-times(si,di)+dd*ss+plt*(ps*ps+epcild*epcild))
     *         *djacb*weigp(igaus)*weigp(jgaus)
          else
            val=val+(shear*times(dit,dit)+0.5*xkcon*epcilt*epcilt)
     *         *djacb*weigp(igaus)*weigp(jgaus)
          end if
500       continue
c
```

```
c****    displacement penalty terms
c
        do 60 ibdis=1,nbdis
           ipoin=presc(1,ibdis)
           djacb=1.0
c           if(ntype.eq.3)djacb=twopi*coord(ipoin,1)
           id(1)=(presc(1,ibdis)-ipoin+0.05)*10
           tmp=id(1)/10.0
           id(2)=(presc(1,ibdis)-ipoin-tmp+0.005)*100
           do 70 i=1,2
             if(id(i).ne.0) then
               ieqns=iabs(ifpre(i,ipoin))
               ifpre(i,ipoin)=-ieqns
               val=val+rho*(presc(i+1,ibdis)-dispt(ieqns))**2*djacb
             end if
70         continue
60      continue
c
c******   stress penalty terms
c
        do 80  iline=1,nline
          ipot1=bline(1,iline)
          ipot2=bline(2,iline)
          if(ifpre(3,ipot1).lt.ifpre(3,ipot2)) then
            ieqn1=ifpre(3,ipot1)
            ieqn2=ifpre(3,ipot2)
            cord1x=coord(ipot1,1)
            cord1y=coord(ipot1,2)
            cord2x=coord(ipot2,1)
            cord2y=coord(ipot2,2)
          else
            ieqn1=ifpre(3,ipot2)
            ieqn2=ifpre(3,ipot1)
            cord1x=coord(ipot2,1)
            cord1y=coord(ipot2,2)
            cord2x=coord(ipot1,1)
            cord2y=coord(ipot1,2)
          end if
c
          d=dsqrt((cord1x-cord2x)**2+(cord1y-cord2y)**2)
          r=0.5*(cord1x+cord2x)
          c=(cord2x-cord1x)/d
          s=(cord2y-cord1y)/d
          dfx=dispt(ieqn2)-dispt(ieqn1)
          dfy=dispt(ieqn2+1)-dispt(ieqn1+1)
          if(ntype.eq.3) then
            djacb=twopi*r*d
            cc=c*c
            ss=s*s
            sc=s*c
            if(dabs(cord1x).lt.err) cord1x=pent
            if(dabs(cord2x).lt.err) cord2x=pent
            fct=0.5*(dispt(ieqn1+2)/cord1x+dispt(ieqn2+2)/cord2x)
            fxx=0.5*(dispt(ieqn1)/cord1x+dispt(ieqn2)/cord2x)
            f(1)=((dfx*c+dfy*s)/d + ss*fct+cc*fxx)
            f(2)=((dfx*s-dfy*c)/d + sc*(fxx-fct))
          else
            f(1)=(dfx*c+dfy*s)/d
            f(2)=(dfx*s-dfy*c)/d
            djacb=d
          end if
          if(ntype.eq.3.and.iline.eq.1) goto 80
          do 90 k=1,2
            if(ifpre(k,ipot1).gt.0.or.ifpre(k,ipot2).gt.0) then
            i=k
            if(dabs(s).lt.err)i=3-k
            pload=bline(2+i,iline)/2.0/g0
```

```
            val=val+rho*(pload-f(i))**2*djacb
            end if
90        continue
80      continue
        return
        end
c
        subroutine brent(ax, bx, cx, tol, xmin, vm, rho,
     *             lnods, coord, nnode, nevab, ndofn,
     *             np,    ne,    ndime, ntype, nstre, ngaus,
     *             props, nmats, matno, weigp, posgp,
     *             presc, bline, ifpre, nbdis, nline,iiter)
c*************************************************************
c
c       brent's method to find a minimum
c
c*********************************************************************
        implicit double precision(a-h,o-z)
        parameter (itmax=100, cgold=0.3819660, zeps=1.0e-4,nmax=4000)
        dimension coord(np,*),lnods(ne,*),posgp(*),weigp(*),
     *            props(nmats,5), matno(*), ifpre(ndofn,*),
     *            presc(3,*), bline(4,*)
        common /dfunc/pcom(nmax), xicom(nmax),  ncom
        common stres(5,4500),strin(5,4500),strdd(5,4500)
        a=dmin(ax,cx)
        b=dmax(ax,cx)
        v=bx
        w=v
        x=v
        e=0.0
c       fx=func(x)
        call mvalue (lnods, coord, x    ,  nnode, nevab, ndofn,
     *               np,    ne,    ndime, ntype, nstre,  ngaus,
     *               posgp,fx ,   props, nmats, matno, weigp,
     *               presc,bline, ifpre, nbdis, nline, rho,iiter)
        fv=fx
        fw=fx
        do 100 iter=1,itmax
          xm=0.5*(a+b)
          tol1=tol*dabs(x)+zeps
          tol2=2.*tol1
          if(dabs(x-xm).le.(tol2-0.5*(b-a))) goto 3
          if(dabs(e).gt.tol1) then
            r=(x-w)*(fx-fv)
            q=(x-v)*(fx-fw)
            p=(x-v)*q-(x-w)*r
            q=2.0*(q-r)
            if(q.gt.0.0) p=-p
            q=dabs(q)
            etemp=e
            e=d
            if(dabs(p).ge.dabs(0.5*q*etemp).or.p.le.q*(a-x).or.
     *         p.ge.q*(b-x)) goto 1
c***    the above conditons determine the acceptability of the
c       parabolic fit. here it is ok
                d=p/q
                u=x+d
                if(u-a.lt.tol2.or.b-u.lt.tol2) d=sign(tol1,xm-x)
                goto 2
          end if
1         if(x.ge.xm) then
            e=a-x
          else
            e=b-x
          end if
          d=cgold*e
2         if(dabs(d).ge.tol1) then
```

```
            u=x+d
          else
            u=x+sign(toll,d)
          end if
c         fu=func(u)
          call mvalue (lnods, coord, u    ,  nnode, nevab, ndofn,
     *                 np,    ne,    ndime, ntype, nstre,  ngaus,
     *                 posgp,fu ,   props, nmats, matno, weigp,
     *                 presc,bline, ifpre, nbdis, nline, rho,iiter)
c
          if(fu.le.fx)then
            if(u.ge.x) then
              a=x
            else
              b=x
            end if
            v=w
            fv=fw
            w=x
            fw=fx
            x=u
            fx=fu
          else
            if(u.lt.x) then
              a=u
            else
              b=u
            end if
c
            if(fu.le.fw.or.w.eq.x) then
              v=w
              fv=fw
              w=u
              fw=fu
            else if(fu.le.fv.or.v.eq.x.or.v.eq.w) then
              v=u
              fv=fu
            end if
          end if
100     continue
        pause 'brent exceed maximum  iterations.'
3       xmin=x
        vm=fx
        return
        end
c
        subroutine mnbrak(ax,bx,cx,fa,fb,fc,   rho  ,
     *              lnods, coord, nnode, nevab, ndofn,
     *              np,    ne,    ndime, ntype, nstre, ngaus,
     *              props, nmats, matno, weigp, posgp,
     *              presc, bline, ifpre, nbdis, nline,iiter)
        implicit double precision(a-h, o-z)
        parameter( gold=1.618034, glimit=100., tiny=1.0e-20,nmax=4000)
        dimension coord(np,*),lnods(ne,*),posgp(*),weigp(*),
     *            props(nmats,5), matno(*),ifpre(ndofn,*),
     *            presc(3,*), bline(4,*)
        common /dfunc/pcom(nmax), xicom(nmax),  ncom
        common stres(5,4500),strin(5,4500),strdd(5,4500)
          call mvalue (lnods, coord, ax   ,  nnode, nevab, ndofn,
     *                 np,    ne,    ndime, ntype, nstre,  ngaus,
     *                 posgp,fa ,   props, nmats, matno, weigp,
     *                 presc,bline, ifpre, nbdis, nline, rho,iiter)
          call mvalue (lnods, coord, bx   ,  nnode, nevab, ndofn,
     *                 np,    ne,    ndime, ntype, nstre,  ngaus,
     *                 posgp,fb ,   props, nmats, matno, weigp,
     *                 presc,bline, ifpre, nbdis, nline, rho,iiter)
        if(fb.gt.fa) then
```

```
          dum=ax
          ax=bx
          bx=dum
          dum=fb
          fb=fa
          fa=dum
        end if
        cx=bx+gold*(bx-ax)
          call mvalue (lnods, coord, cx    ,  nnode, nevab, ndofn,
     *                 np,    ne,    ndime, ntype, nstre, ngaus,
     *                 posgp,fc ,   props, nmats, matno, weigp,
     *                 presc,bline, ifpre, nbdis, nline, rho,iiter)
1       if(fb.ge.fc) then
          r=(bx-ax)*(fb-fc)
          q=(bx-cx)*(fb-fa)
        u=bx-((bx-cx)*q-(bx-ax)*r)/(2.*sign(dmax(dabs(q-r),tiny),q-r))
          ulim=bx+glimit*(cx-bx)
          if((bx-u)*(u-cx).gt.0) then
          call mvalue (lnods, coord, u     ,  nnode, nevab, ndofn,
     *                 np,    ne,    ndime, ntype, nstre,  ngaus,
     *                 posgp,fu ,   props, nmats, matno, weigp,
     *                 presc,bline, ifpre, nbdis, nline, rho,iiter)
                if(fu.lt.fc) then
                  ax=bx
                  fa=fb
                  bx=u
                  fb=fu
                  goto 1
                else if(fu.gt.fb) then
                  cx=u
                  fc=fu
                  goto 1
                end if
                u=cx+gold*(cx-bx)
         call mvalue (lnods, coord, u     ,  nnode, nevab, ndofn,
     *                 np,    ne,    ndime, ntype, nstre,  ngaus,
     *                 posgp,fu ,   props, nmats, matno, weigp,
     *                 presc,bline, ifpre, nbdis, nline, rho,iiter)
         else if((cx-u)*(u-ulim).gt.0) then
         call mvalue (lnods, coord, u     ,  nnode, nevab, ndofn,
     *                 np,    ne,    ndime, ntype, nstre,  ngaus,
     *                 posgp,fu ,   props, nmats, matno, weigp,
     *                 presc,bline, ifpre, nbdis, nline, rho,iiter)
               if(fu.lt.fc) then
                 bx=cx
                 cx=u
                 u=cx+gold*(cx-bx)
                 fb=fc
                 fc=fu
         call mvalue (lnods, coord, u     ,  nnode, nevab, ndofn,
     *                 np,    ne,    ndime, ntype, nstre,  ngaus,
     *                 posgp,fu ,   props, nmats, matno, weigp,
     *                 presc,bline, ifpre, nbdis, nline, rho,iiter)
               end if
         else if((u-ulim)*(ulim-cx).ge.0) then
               u=ulim
         call mvalue (lnods, coord, u     ,  nnode, nevab, ndofn,
     *                 np,    ne,    ndime, ntype, nstre,  ngaus,
     *                 posgp,fu ,   props, nmats, matno, weigp,
     *                 presc,bline, ifpre, nbdis, nline, rho,iiter)
         else
              u=cx+gold*(cx-bx)
         call mvalue (lnods, coord, u     ,  nnode, nevab, ndofn,
     *                 np,    ne,    ndime, ntype, nstre,  ngaus,
     *                 posgp,fu ,   props, nmats, matno, weigp,
     *                 presc,bline, ifpre, nbdis, nline, rho,iiter)
         end if
```

```
          ax=bx
          bx=cx
          cx=u
          fa=fb
          fb=fc
          fc=fu
          goto 1
        end if
        return
        end
c
        subroutine differ2(si,di,dit,dldd,dlss,dlds,dld1,dls1,y0,y,ps)
c*************************************************************
c
c       calculate the second order differential of l
c       on each gauss point by analytical method
c                         ------- 3 mar. 1992
c*************************************************************
        implicit double precision(a-h,o-z)
        double precision k
        dimension di(5),dit(5),si(5),fdi(5),fdti(5),fsi(5),
     *            pdi(5),psi(5),dldd(5,5),dlss(5,5),dlds(5,5),
     *            dld1(5),dls1(5)
        plt=50.0
        fs=dsqrt(times(si,si))
        fs2=1.0/fs/fs
        fd=dsqrt(times(di,di))
        fd2 =1.0/fd/fd
        fdt=dsqrt(times(dit,dit))
        fdt2=1.0/fdt/fdt
        k=y*fdt+y0
        d= k*fd
        ps=fs/k-1.0
        if(ps.lt.0.0)return
        do 10 i=1,5
          fdi(i)=di(i)/fd
          fdti(i)=dit(i)/fdt
          fsi(i)=si(i)/fs
          pdi(i)=-y*fs*fdti(i)/k/k
          psi(i)=fsi(i)/k
c
          dld1(i)=-si(i)+fs*fdi(i)+2.0*plt*ps*pdi(i)
          dls1(i)=-di(i)+fd*fsi(i)+2.0*plt*ps*psi(i)
10      continue
c
        do 20 i=1,5
          do 20 j=i,5
            fss=(fs*delt(i,j)-si(i)*fsi(j))*fs2
            fdd=(fd*delt(i,j)-di(i)*fdi(j))*fd2
            fddt=(fdt*delt(i,j)-dit(i)*fdti(j))*fdt2
            pdd=y*fs*(2.0*y*fdti(i)*fdti(j)-k*fddt)/k/k/k
            pss=fss/k
            pds=-y*fdti(i)*fsi(j)/k/k
c
            dldd(i,j)=fs*fdd+2*plt*(ps*pdd+pdi(i)*pdi(j))
            dlss(i,j)=fd*fss+2*plt*(ps*pss+psi(i)*psi(j))
            dlds(i,j)=-delt(i,j)+fdi(i)*fsi(j)+2*plt*(ps*pds+
     *      pdi(i)*psi(j))
            dldd(j,i)=dldd(i,j)
            dlss(j,i)=dlss(i,j)
            dlds(j,i)=dlds(i,j)
20      continue
        return
        end
c
        subroutine differ1(force,bsmat,asmat,ammat,bmmat,lnods,nelem,
     *                     nnode,ndofn,nstre,ntype,ielem,dvolu,dld1,dls1,
```

```
     *                        epcild,epcilt,shear,xkcon,ps,dt)
c*****************************************************************
c
c       this routine is to calculate the first derivative
c       of L at each gauss point and form the 'force' term
c
c*****************************************************************
        implicit double precision(a-h,o-z)
    dimension lnods(nelem,*), asmat(5,*),bsmat(5,*), ammat(*),
     *           bmmat(*),dld1(*),dls1(*),force(*),dt(*)
        plt=50.0
    ndof1=2
    ndof2=2
    if(ntype.eq.3) ndof2=3
        g1=2.0*shear
        do 150 inode=1,nnode
          ndofi=(lnods(ielem,inode)-1)*ndofn
          do 165 idofn=1,ndof1
            igar=ndofi+idofn
            ivar=(inode-1)*ndof1+idofn
            do 160 istre=1,nstre
              if(ps.ge.0.0) then
               force(igar)=force(igar)-dld1(istre)*bsmat(istre,ivar)
     *                     *dvolu
              else
               force(igar)=force(igar)-dt(istre)*bsmat(istre,ivar)
     *                     *g1*dvolu
              end if
160         continue
            if(ps.ge.0.0)then
            force(igar)=force(igar)-2.0*plt*epcild*bmmat(ivar)*dvolu
            else
            force(igar)=force(igar)-xkcon*epcilt*bmmat(ivar)*dvolu
            end if
165       continue
          do 175 jdofn=1,ndof2
            jgar=ndofi+ndof1+jdofn
            jvar=(inode-1)*ndof2+jdofn
            do 170 jstre=1,nstre
              if(ps.ge.0.0)then
               tmp=dls1(jstre)*g1-dld1(jstre)
               force(jgar)=force(jgar)-asmat(jstre,jvar)*tmp*dvolu
              else
               force(jgar)=force(jgar)+asmat(jstre,jvar)
     *                     *dt(jstre)*dvolu*g1
              end if
170         continue
            if(ps.ge.0.0)then
            force(jgar)=force(jgar)+2.0*plt*epcild*ammat(jvar)
     *                 *g1/xkcon*dvolu
            else
            force(jgar)=force(jgar)+epcilt*ammat(jvar)*dvolu*g1
            end if
175       continue
150     continue
    return
    end
c
        function times(a,b)
        implicit double precision(a-h,o-z)
        dimension a(5),b(5)
        times=0.0
        do 10 i=1,5
10      times=times+a(i)*b(i)
        return
        end
c
```

```
      function delt(i,j)
      double precision delt
      delt=0.0
      if(i.eq.j)delt=1.0
      return
      end
c
      function sign(a,b)
      implicit double precision(a-h, o-z)
      if(b.gt.0.0) then
         sign=dabs(a)
      else
         sign=-dabs(a)
      end if
      return
      end
c
      function dmax(a,b)
      implicit double precision(a-h, o-z)
      if(a.gt.b) then
        dmax=a
      else
        dmax=b
      end if
      return
      end
c
      function dmin(a,b)
      implicit double precision(a-h, o-z)
      if(a.lt.b) then
        dmin=a
      else
        dmin=b
      end if
      return
      end
```

REFERENCES

1. Acheson, D. J., **Elementary Fluid Dynamics,** *Clarendon Press,* Oxford(1990).
2. Alcini, W. V., **Experimental Measurement of Liquid Nugget Heat Convection in Spot Welding,** *Welding Journal,* ppl77-180(1990).
3. Alcini, W. V., **A Measurement Window into Resistance Welding,** *Welding Journal,* pp47-50(1990).
4. Argyris, J. H., **Energy Theorems and Structural Analysis,** Butterworth(1960).
5. Atkin, R. J. and Fox, N, **An Introduction to the Theory of Elasticity,** *Longman,* London(1980).
6. Atluri, S. N., Kobayashi, A. S. and Nakagaki, M., **Application of an Assumed Displacement Hybrid Element Procedure to Two-Dimensional Problems in Fracture Mechanics,** *AIAA/ASME/SAE Fifteenth Structures, Structural Dynamics and Materials Conference,* April 17-19, Las Vegas(1974).
7. Bentley, K. P., Greenwood, J. A., Knowlson, P. and Baker, R. G., **Temperature distributions in spot welds,** *BWRA REPORT,* pp613-619 (1963).
8. Bercovier, M., **Finite Elements for Incompressible or Nearly Incompressible Materials,** *Proceedings of the ADINA Conference, Massachusetts Institute of Technology,*Cambridge, pp.384-400(1977).
9. Bolton, W., **Residual stresses in and around spot welds,** *British Welding Journal,* Vol 8, pp57-60(1961).
10. Brent, R. P., **Algorithms for Minimization without Derivatives,** *England Cliffs, N. J.:Prentice-Hall*(1973).
11. Brezzi, F and Bathe, K. J., **A Discourse on the Stability Conditions for Mixed Finite Element Formulations,** *Computer Methods in Applied Mechanics and Engineering,* Vol. 82, pp.27-57(1990).
12. Brezzi, F. and Fortin, M., **Mixed and Hybrid Finite Element Methods,**

Springer, London(1991).

13. Browne, D., **Computer Model of Nugget Formation in the Spot-Welding of Aluminium**, *A report prepared for Alcan International*(1991).

14. Browne, D., **Computer Modelling of Aluminium Spot Welding**, *A report prepared for Alcan International*(1992).

15. Chandler, H. W., **A Finite Element Implementation of a Mixed Extremum Principle for Linear Elasticity**, *Communications in Applied Numerical Methods*, Vol. 8, pp.9-15(1992).

16. Chandler, H. W., **Preparation of a Report on Mechanical Simulation of Spot-Welding**, *Private conversation with Jian Wen*(1993).

17. Chandler, H. W., **First-order Stress Functions for Axisymmetric problems**, *Private conversation with Jian Wen*(1991).

18. Chandler, H. W., **Homogeneous and Localised Deformation in Granular Materials: a Mechanic Model**, *Int. J. Engng Sci.*, Vol. 28, pp.719-734(1990).

19. Chandler, H. W., **A Plasticity Theory Without Drucker's Postulate, Suitable for Granular Materials**, *J. Mech. Phys. Solids*. Vol ??(1985).

20. Chavent, G., Cohen, G., Jaffre, J. Eymard, R., Dominique, R. and Weiil, L., **Discontinuous and Mixed Finite Elements for Two-Phase Incompressible Flow**, *SPE Reservoir Engineering (Society of Petroleum Engineers)*, Vol.5, No.4, pp567-575(1990).

21. Cescotto, S. and Charlier, R., **Frictional Contact Finite Elements Based on Mixed Variational Principles**, *International Journal for Numerical Methods in Engineering*, Vol.36, pp.1681-1701(1993).

22. Cho, H. S. and Cho, Y. J., **A Study of the Thermal Behaviour in Resistance Spot Welds**, *Weld Journal*, pp236-244(1989).

23. Chouchaoui, B., Shirazi-Adl, A., **Mixed Finite Element Formulation for the Stress Analysis of Composite Structures**, *Computers and Structures*, Vol. 43, No.4, pp687-698(1992).

24. Clough, R. H., **The Finite Element in Plane Stress Analysis**, *Proceedings of the Second Conference on Electronic Computation*, ASCE, pp.345-378(1960).

25. Crandall, S. H., **Engineering Analysis**, *McGraw-Hill*(1956).

26. Day, M. L. and Yang, T. Y., **A mixed Variational Principle for Finite Element Analysis**, *International Journal for Numerical Methods in Engineering*, Vol. 18, pp.1213-1230(1982).

27. de Saint-Venant, B., *Comptes Rendus Acad. Sci. Paris*, Vol. 70(1870).

28. Desai, C. S. and Lightner, J. G., **Mixed Finite Element Procedure for Soil-Structure Interaction and Construction Sequences**, *International Journal for Numerical Methods in Engineering*, Vol. 21, pp.801-824(1985).

29. Drucker, D. C., A **More Fundamental Approach to Plastic Stress-strain Relation**, *Proc. 1st National Congress of Applied Mechanics, ASME*, Chicago, pp.487-491(1951).

30. Dunham, R. S. and Pister, K. S., **A Finite Element Application of the Hellinger-Reissner Variational Theorem**, *Proceedings of the Conference on Matrix Methods in Structural Mechanics*, AFFDL-TR-68-150, pp471-(1968).

31. Finlayson, B. A., **The Method of Weighted Residuals and Variational Principles**, *Academic Press*(1972).

32. Fletcher, R., **Practical Methods of Optimisation**, *2nd Edition*, Wiley(1987).

33. Fraeijs de Veubeke, B., **Displacement and equilibrium models in finite element method**, Chapter 9 of *Stress Analysis*, eds O. C. Zienkiewicz and C. S. Holister, Wiley, pp.145-242(1965).

34. Goleniewski, G., **Low-order Mixed Method finite elements in non-linear elasticity.**, *Communications in Applied Numerical Methods*, Vol. 7, No. 1, pp.57-63(1991).

35. Gould, J. E., **An examination of nugget development during spot welding, using both experimental and analytical technique**, *Welding Journal*, pp1-10(1987).

36. Greenwood, T. A., **Temperature in Spot Welding**, *BWRA Report*, pp3 16-322(1961).

37. Han, Z., Orozco, J., Indacochen, J. E. and Chen, C. H., **Resistance Spot Welding: A Heat Transfer Study**, *Welding Journal*, pp363-371(1989).

38. Hellan, K., **Analysis of Elastic Plates in Flexure by a Simplified Finite Element Method**, *Acta Polytechnica Scandinavica*, Civil Engineering Series

No. 46(1967).

39. Hellinger, E., **Der Allgemeine Ansatz der Mechanik der Kontinua**, *Encyclopedie der Mathematischen Wissenschaften*, Vol. 4, Part 4, pp.602-694(1914).

40. Herrmann, L. R., **A Bending Analysis of Plates**, *Proceedings of the Conference on Matrix Methods in Structural Mechanics*, AFFDL-TR-66-80, pp. 577-604(1966).

41. Heyliger, P. R. and Kriz, R. D., **Stress Intensity Factors by Enriched Mixed Finite Elements**, *International Journal for Numerical Methods in Engineering*, Vol. 28, pp.1461-1473(1989).

42. Hill, R., **The Mathematical Theory of Plasticity**, *Oxford at Clarendon Press*(1950).

43. Houchens, A. F., Page, R. E. and Yang, W. H., **Numerical modelling of resistance spot welding**, *Numerical Modelling of Manufacturing Processes*, eds. Jones, R. F., Armin, H. and Fong, J. T., presented at ASME WAM, pp117-129(1977).

44. Hu, H. C., **On Some Variational Principles in the Theory of Elasticity and the Theory of Plasticity**, *Scientia Sinica*, Vol. 4, pp.33-45(1955).

45. Hutton, S. G., Exeter, M. K., Fussy, D. E., Webster, J. J. and Rigon, C., **Primitive Variable Finite Element Formulations for Steady Viscous Flows**, *International Journal for Numerical Methods in Engineering*, Vol. 15, pp.209-223(1980).

46. Jackson, C. P. and Cliffe, K. A., **Mixed Interpolation in Primitive Variable Finite Element Formulations for Incompressible Flow**, *International Journal for Numerical Methods in Engineering*, Vol. 17, pp.1659-1688(1981).

47. Jenson, S., **P-version of Mixed Finite Element Methods for Stokes-like Problems**, *Computer Methods for Applied Mechanics and Engineering*, Vol.101, pp.27-41(1992).

48. Jetteur, PH. and Cescotto, S., **A Mixed Finite element for the Analysis of Large Inelastic Strains**, *International Journal for Numerical Methods in Engineering*, Vol. 31, pp.229-239(1991).

49. Kaiser, J. G., Dunn, G. J. and Eagar, T. W., **The effect of Electrical**

Resistance on Nugget Formation During Spot Welding, *Welding Journal*, ppl67-174(1982).

50. Kim, E. W. and Eagar, T. W., **Transient Thermal behaviour in resistance spot welding**. *Sheet Metal Welding Conference 111*, Paper No.2, Detroit (1988).
51. Kim, E., **Analysis of Resistance Spot Welding Lobe Curve**, *PhD Thesis*, MIT(1989).
52. Kim, E. W. and Eagar, T. W., **Measurement of Transient Temperature Response during Resistance Spot Welding**, *Welding Journal*, pp303-312(1989).
53. Kim, E. and Eagar, T. W., **Controlling Parameters in Resistance Welding**, *Sheet Metal Welding Conference IV*, Paper No. 17(1990).
54. Klein, B., **A Simple Method of Matrix Structural Analysis**, *Journal of the Aeronautical Sciences*, Vol. 24, No.1, pp.39-46(1957).
55. Krieg, R. D. and Krieg, B. B., **Accuracies of Numerical Solution Methods for the Elasto-perfectly plastic model**, *J. Press. Vess. Technol. ASME*, Vol. 99, pp.510-515(1977).
56. Lawrence, F. V. Jr., Corten, H. T. and McMahon, J. C., **Final report to the American Iron and Steel Institute on the improvement of steel spot weld fatigue resistance**(1985).
57. Lee, A. and Nagel, G. L., **Basic phenomena in resistance spot welding**. *SAE Paper no. 880277*(1988).
58. Levy, M., *Journ. Math. pures et app.*, Vol 16(1871).
59. Liao, C. L. and Tsai, J. S., **A Mixed Finite Element Formulation for Nonlinear Analysis of Plane Problems**, *International Journal for Numerical Methods in Engineering*, Vol. 33, pp.1721-1736(1992).
60. Lightner, J. G., **A Mixed Finite Element Procedure for Soil-Structure Interaction Including Construction Sequences**, *Ph.D Dissertation*, Virginia Polytechnic Institute and State University, Blacksburg(1981).
61. Loula,A. F. D. and Guerreiro, J. N. C., **Finite Element Analysis of Nonlinear Creeping Flows**, *Computer Methods in Applied Mechanics and Engineering*, Vol.79, No.1, pp87-109(1990).

62. Love, A. E. H., **Mathematical Theory of Elasticity**, *4th ed.*, Cambridge University Press, New York, p274-(1927).

63. Lynn, P. P. and Arya, S. K., **Finite Element Formulation by the Weighted Discrete Least Squares Method**, *International Journal for Numerical Methods in Engineering*, Vol. 8, pp.71-161(1974).

64. Mahapatra, R. C. and Dasgupta, S. P., **The Mixed Finite Element Method in Elastic and Elasto-plastic Axisymmetric Problems**, *Computers and Structures*, Vol. 30, pp1047-1065(1988).

65. Mahapatra, R. C. and Dasgupta, S. P., **Elasto-plastic Solution of Some Axially Symmetric Geomechanic Problems by Mixed Finite Elements**, *Computers and Structures*, Vol.41, No.1 pp93-92(1991).

66. Markov, A. A., **On Variational Principles in the Theory of Plasticity**, *Prikladnaia Matematika i Mekhanika*, Vol.11, pp.339-389(1947).

67. Mau, S. T., Pian, T. H. H. and Tong, P., **Vibration Analysis of Laminated Plates and Shells by a Hybrid Stress Element**, *AIAA Journal*, Vol. 11, pp.1450-1452(1973).

68. Meissner, U., **A Mixed Finite Element Model for Use in Potential Flow Problems**, *International Journal for Numerical Methods in Engineering*, Vol. 6, pp.467-473(1973).

69. Melan, E., *Ingenieur-Archiv*, vol. 9(1938).

70. Miklin, S. C., **The Problems of the Minimum of a Quadratic Functional**, *Holden-Day*(1965).

71. Miklin, S. C., **Variational Methods in Mathematical Physics**, *MacMillan*(1964).

72. Marguerre, K. and Woernle H., **Elastic Plates**, *Blaisdell Publishing Company*(1969).

73. Moitinho de Almeida, J. P. and Teixeira de Freitas, J. A., **Continuity Conditions for Finite Element Analysis of Solid**, *International Journal for Numerical Methods in Engineering*, Vol. 33, pp.845-853(1992).

74. Moore, D. F., **The Friction and Lubrication of Elastomers**, *Pergamon Press*(1972).

75. Nied, H. A., **The finite element modelling of the resistance spot welding**

process, *Welding Journal*, ppl23-132(1984).

76. Noor, A. K. and Andersen, C. M., **Mixed Isoparametric Elements models of Laminated Composite Shells**, *Computer Methods in Applied Mechanics and Engineering*, Vol. 11, pp.255-280(1977).

77. Noor, A. K., **State-of-the Art Surveys on Finite Element Technology**, *ASME*, ed. by A. K. Noor and W. D. Pilkey, New York, pp.127-158(1983).

78. Olson, M. D. and Mirza, F. A., **A Mixed Finite Element Method for Calculation of Stress Intensity Factors**, *Numerical Methods in Fracture Mechanics*, ed. by A. R. Luxmore and D. R. J. Owen, University College of Swansea,pp.798-803(1978).

79. Ortiz, M and Popov, E. P., **Accuracy and Stability of Integration Algorithms for Elastoplastic Constitutive Relation**, *International Journal for Numerical Methods in Engineering*, Vol. 21, pp.1561-1576(1985).

80. Ortiz, M and Simo, J. C., **An Analysis of a New Class of the Integration Algorithms for Elastoplastic Constitutive Relations**, *International Journal for Numerical Methods in Engineering*, Vol. 23, pp.353-366(1986).

81. Owen, D. R. J. and Hinton, E., **Finite Elements in Plasticity, Theory and Practice**, *Pineridge Press*(1980).

82. Papadopoulos, P, Taylor, R. L., **Mixed Formulation for the Finite Element Solution of Contact Problems**, *Computer Methods in Applied Mechanics and Engineering*, Vol.94, No.3, pp373-389(1992).

83. Pian, T. H. H., **Derivation of Element Stiffness Matrices by Assumed Stress Distributions**, *JAIAA*, Vol. 2, pp.1333-1338(1964).

84. Pinsky, P. M., Jasti, R. V., **Mixed Finite Element for Laminated Composite plates based on the use of Bubble Functions**, *Engineering Computations (Swansea, Wales)*, Vol.6, No.4, pp316-330(1989).

85. Pinsky, P. M., Jasti, R. V., **On the Use of Lagrange Multiplier Compatible Modes for Controlling Accuracy and Stability of Mixed Shell Finite Elements**, *Computer Methods in Applied Mechanics and Engineering*, Vol.85, No.2, pp151-182(1991).

86. Prager, W., **Recent Developments in the Mathematical Theory of Plasticity**, *Journal of Applied Physics*, Vol.20, pp.235-241(1949).

87. Prager, W. and Synge, J. L., **Approximations in Elasticity Based on the Concept of Function Space**, *Quart. Appl. Math.*, Vol. 5, pp.241-269(1974).

88. Prandtl, L., *Proc. 1st Int. Cong. App. Mech.*, Delft(1924).

89. Prange, G., **Das Extremum der Formanderungsarbeit**, *Habilitation Thesis*, Hannover(1916).

90. Rayleigh, J. W. S., **Theory of Sound**, First edition(1877), revised at Dover, N. Y. (1945).

91. Reddy, G. P. and Sharma, S., **Simulation of spot weld process**, *Recent Trends in Welding Science and Technology*, pp59-68(1989).

92. Reddy, J. N., **Finite Element Analysis of Viscous Incompressible Flows Using Primitive Variables**, *Computers and Structures*, Vol.47, pp857-869(1993).

93. Reissner, E., **On a Variational Theorem in Elasticity**, *Journal of Mathematics and Physics*, Vol. 29, pp.90-95(1950).

94. Reuss, A., *Zeits. ang. Math. Mech.*, Vol.10(1930).

95. Rice, W. and Funk, E. J., **An analytical investigation of the temperature distribution during resistance welding**, *Welding Journal*, ppl75-186(1967).

96. Ritz, W., **Uber eine neue Methode zer Losung gewisser Variations - Probleme der Mathematischen Physik**, *J. Reine Angew. Math.*, Vol. 135, pp.1-61(1909).

97. Sandhu, R. S., Shaw, H. L. and Hong and S. J., **Three-Field Finite Element Procedure for Analysis of Elastic Wave Propagation through Fluid-Saturated Soils**, *Soil Dynamics and Earthquake Engineering*, Vol.9, No.1, pp58-65(1990).

98. Sarigul, N, **Assumed Stress Function Finite Element Method: Two-Dimensional Elasticity**, *International Journal for Numerical Methods in Engineering*, Vol. 28, pp.1577-1598(1989).

99. Schoepfel, A., **Fatigue Performance of Plug Welds**, *Diploma thesis*, Munchen, Technical University Munchen, TUM-MW 14/8805-DA(1988).

100. Sheppard, S., **Residual Stresses in Resistance Spot Welds**, *NSF Status Report*, Grant No. DMC86-18462-A1(1989).

101. Sheppard, S., **Thermal and Mechanical Simulations of Resistance Spot**

Welding, *Sheet Metal Welding Conference IV*, paper no. 20(1990).

102. Shyu, S. C., Chang, T. Y., Saleeb, A. F., **Friction-Contact Analysis Using A Mixed Element Method**, *Computers and Structures*, Vol.32, No.1, pp223-242(1989).

103. Silvester, D. J. and Kechkar, N., **Stabilised Bilinear-Constant Velocity-Pressure Finite Elements for the Conjugate Gradient Solution of the Stokes Problem**, *Computer methods in Applied Mechanics and Engineering*, Vol.79, No.1, pp71-86(1990).

104. Sime, J. C., Kennedy, J. G. and Taylor, R. L., **Complementary Mixed Finite Element Formulation for Elasto-Plasticity**, *Computer methods in Applied Mechanics and Engineering*, Vol.74, No.2, pp177-206(1989).

105. Simo, J. C. and Rifai, M. S., **A Class of Mixed Assumed Strain Methods and The Method of Incompatible Modes**, *International Journal for Numerical Methods in Engineering*, Vol. 29, pp.1595-1638(1990).

106. Simo, J. C. and Armero, F., **Geometrically Nonlinear enhanced Strain Mixed Methods and The Method of Incompatible Modes**, *International Journal for Numerical Methods in Engineering*, Vol. 33, pp.1413-1449(1992).

107. Sohn, J. L. and Heinrich, J. C., **Poisson Equation Formulation fro Pressure Calculation in Penalty Finite Element Models for Viscous Incompressible flows**, *International Journal for Numerical Methods in Engineering*, Vol.30, pp349-361(1990).

108. Spencer, A. J. M., **Continuum Mechanics**, *Longman*, London(1980).

109. Spilker, R. L. and Maxian, T. A., **A Mixed-Penalty Finite Element Formulation of the Linear Biphasic Theory for Soft Tissues**, *International Journal for Numerical Methods in Engineering*, Vol. 30, pp.1063-1082(1990).

110. Timoshenko, S. P. and Goodier, J. N., **Theory of Elasticity**, *McGraw-Hill*, New York(1951).

111. Tong, P., **New Displacement Hybrid Finite Element Models for Solid Continua**, *International Journal for Numerical Methods in Engineering*, Vol. 2, pp.73-83(1970).

112. Tong, P. and Pian, T. H. H. and Lasry, S. J., **A Hybrid Element approach to Crack Problems in Plane Elasticity**, *International Journal for Numerical*

Methods in Engineering, Vol. 7, pp.297-308(1973).

113. Tresca, H., *Comptes Rendus Acad. Sci. Paris*, Vol. 59(1864).

114. Tsai, C. L., Jammal, O. A. and Dickimson, D., **Study of Nugget Formation in Resistance Spot Welding Using the Finite Element Method**, *Recent Trends in Welding Science and Technology*, ppl4-18(1989).

115. Tseng, J. and M. D. Olson, **The Mixed Finite Element Method Applied to Two-dimensional Elastic Contact Problems**, *International Journal for Numerical Methods in Engineering*, Vol. 17, pp.991-1014(1981).

116. Turner, M. J., Clough, R. W., Martin, H. C. and Topp, L. J., **Stiffness and Deflection Analysis of Complex Structures**, *Journal of Aeronautical Sciences*, Vol. 23, pp.805-823(1956).

117. Van den Bogert, P. A., De Borst, R., Luiten, G. t. and Zeilmaker, J., **Robust Finite Elements for 3D-Analysis of Rubber-like Materials**, *Engineering Computations (Swansea, Wales)*, Vol.8, No.1, pp3-17(1991).

118. Von Mises, R., **Göttinger Nachrichten**, Math. Phys. Klasse(1913).

119. Von Mises, R., Zeits. ang. Math. Meth. Vol. 8(1928).

120. Washizu, K., **On the Variational Principles of Elasticity and Plasticity**, *Rep. 25-18, Contract N5 Ori-07833, Aeroelastic and Structures Research Laboratory*, Massachusetts Institute of Technology, Cambridge(1955).

121. Widmann, J., **Fatigue resistance of simulated spot welds status report III**, *a report to S. Sheppard and H. Fuchs*(1988).

122. William, H. P. et al, **Numerical Recipes: The Art of Scientific Computing (Fortran Version)**, *Cambridge University Press*(1989).

123. Wilt, T. E., Saleeb, A. F. and Chang, T. Y., **Mixed Element for Laminated Plates and Shells**, *Computers and Structures*, Vol. 37, No. 4, pp.597-611(1990).

124. Wood, R. D., **Finite Element Analysis of Plane Couple-stress Problems Using First Order Stress Functions**, *International Journal for Numerical Methods in Engineering*, Vol. 26, pp.489-509(1988).

125. Yamada, Y., Ito, K., Yokouchi, Y., Tamano, T. and Ohtsubo, T., **Finite Element Analysis of Steady Fluid and Metal Flow**, *Finite Elements in Fluids*,ed. by R. H. Gallagher, J. T. Oden, C. Taylor and O. C. Zienkiewicz,

Vol. 1, John Wiley and Sons, pp.73-94(1975).

126. Zienkiewicz, O. C. and Taylor, R. L., **The Finite Element Method,** *4th ed., Vol. 1 McGraw-Hill*, London(1988).

127. Zienkiewicz, O. C., Owen, D. R. J. and Lee, K. N., **Least Square Finite Element for Elasto-static Problems - Use of Reduced Integration,** *International Journal for Numerical Methods in Engineering*, Vol. 8, pp.341-399(1974).

128. Zienkiewicz, O. C. and Cheung, Y. K., **Finite Elements in the Solution of Field Problems,** *The Engineer*, pp.507-510(1965).