ADVANCES IN ROBUST METHODS FOR LIMIT LOAD ANALYSIS

IHAB F.Z. FANOUS


CENTRE FOR NFLD. STUDIES
MEMORIAL UNIVERSITY
OF NEWFOUNDLAND

ADVANCES IN ROBUST METHODS FOR LIMIT LOAD ANALYSIS

By

Ihab F. Z. Fanous ©

A thesis submitted to the School of Graduate Studies
In partial fulfillment of the requirements For the Degree of
Doctorate of Philosophy

Faculty of Engineering and Applied Science
Memorial University of Newfoundland

May 2008
St. John's, NL, Canada

To my father, my mother, my brothers and their families for their help and continuous support to accomplish this work.

ABSTRACT

Limit load analysis is an essential tool in engineering analysis. Several methods were developed for both the upper and lower bound limit load multipliers. Several methods are developed with the objective of having a simplified analysis procedure to evaluate the limit load without the use of complex inelastic analysis. The recently developed lower bound solutions are either conservative or have some limitations in their applications. The redistribution node method was developed earlier as a lower bound limit load solution using the iterative elastic finite element analysis. It was applied to several two dimensional problems.

In the present work, the iterative R-Node method is introduced as a tool to calculate the lower bound limit load of a component. The method interprets the redistribution of the stress to find the reference stress which is used to calculate the limit load. The applicability of the iterative R-Node method to complex three dimensional problems is investigated. This includes applications with three dimensional shell and solid brick elements. Single and multiple loads are also applied to. Also, the results are used to help in the stress classification of the finite element analysis results according to the American Society of Mechanical Engineers codes.

Finally, the reference volume limit load analysis was developed in previous research using the m^0 upper bound solution. It was shown that it has a high convergence rate when compared to the other analysis methods. In this work, the method is redeveloped using the classical upper bound multiplier. The applicability of the method is verified for complex three dimensional geometries modeled using shell and solid elements.

ACKNOWLEDGMENTS

The author wishes to thank his supervisor, Dr. R. Seshadri, for his unparalleled support and continuous help. He has always been available for discussion despite his duties and limited time.

Also, the financial support of the Faculty of Engineering & Applied Science and the School of Graduate studies are gratefully acknowledged.

TABLE OF CONTENTS

Abstract iii

Acknowledgments iv

Table of Contents v

List of Figures viii

List of Tables xii

List of Symbols xiii

Chapter 1 Introduction 1

1.1 General Background 1

1.2 Limit Load Analysis 2

1.3 Objective of the thesis 3

1.4 Organization of the thesis 4

Chapter 2 Literature Review 6

2.1 Overview 6

2.2 Robust Methods in Stress Distribution Analysis 7

2.2.1 The Elastic Compensation Method (ECM) 7

2.2.2 The Equivalent Strain Energy Density (ESED) Method 8

2.3 Limit Load Analysis 10

2.3.1 Mura's Lower Bound Theorem 10

2.3.2 Classical upper bound multiplier 14

2.3.3 The m^0 Upper Bound Multiplier 15

2.3.4 The m_α-Method 17

2.3.5 The GLOSS R-Node Method 21

2.4 Stress Classification 23

2.4.1 Definition of Stress Classification 23

2.4.2 Method of Classifying Stresses 26

2.4.3 Stress Linearization in ABAQUS 29
2.5 Summary 32

Chapter 3 Theoretical Background 33
3.1 Overview 33
3.2 Reference Stress 33
3.3 Sample Problems 37
3.3.1 Cantilever beam 37
3.3.2 Thick cylinder 40
3.4 The Redistribution Node Method 44
3.5 Summary 46

Chapter 4 Iterative R-Node Analysis method 48
4.1 Overview 48
4.2 Iterative R-Node Analysis Algorithm 53
4.3 Applications 56
4.3.1 Thick Plane-Strain Cylinder 56
4.3.2 Indeterminate Beam 58
4.3.3 Oblique Nozzle 62
4.4 Summary 64

Chapter 5 Sequential Multiple Loads 66
5.1 Overview 66
5.2 Iterative Limit-Load Analysis 66
5.3 Applications 69
5.3.1 Cantilever Beam Model 70
5.3.2 Pipe Bend Model 72
5.4 Summary 78

Chapter 6 Stress Classification 79
6.1 Overview 79
6.2 Stress Classification Techniques 79
6.2.1 Stress Linearization in FE Programs 79
6.2.2 R-Node Analysis 81

6.2.3 Elastic-Plastic Analysis 84
6.3 Proposed Methodology 84
6.4 Applications 85
6.4.1 Indeterminate Beam 85
6.4.2 Axisymmetric Pressure Vessel 89
6.4.3 Straight Nozzle 91
6.4.4 Comparison of Analysis methods 93
6.5 Summary 94

Chapter 7 The Reference Volume Concept 95
7.1 Overview 95
7.2 Applications 99
7.2.1 Indeterminate Beam 99
7.2.2 Thick Plane-Strain Cylinder 102
7.2.3 Axisymmetric Pressure Vessel 104
7.2.4 Oblique Nozzle 105
7.3 Summary 109

Chapter 8 Conclusion 110

References 113

Appendix A User-Defined Material 116
A.1 Plane stress 116
A.2 Plane strain 117
A.3 Axisymmetric 119
A.4 Layered shell 121
A.5 Solid 122

Appendix B Post-processing scripts 125
B.1 Plane Stress and Plane Strain 125
B.2 Axisymmetric 127
B.3 Layered Shell 131
B.4 Solid 135

LIST OF FIGURES

Fig. 2.2: Schematic of the ESED method 9
Fig. 2.3: The variation of the upper bound and the lower bound multipliers with the iteration variable 18
Fig. 2.1: GLOSS diagram 22
Fig. 2.4: Schematic diagram of sample results of stress analysis. 24
Fig. 2.5: Methods for eliminating peak stress. 27
Fig. 2.6: Definition of bending and peak stress. 28
Fig. 2.7. Coordinates of Cross Section 29
Fig. 2.8: (a) Axisymmetric Cross-Section and (b) Geometry used for Axisymmetric Evaluations 31
Fig. 3.1: Convergence of the stress towards the stead state. 34
Fig. 3.2: Steady state stress distribution across a beam. 36
Fig. 3.3: Thick walled cylinder subjected to internal pressure 40
Fig. 3.4: Distribution of the hoop stress through the thickness of the thick cylinder for different values of the creep exponent. 42
Fig. 3.5: Distribution of the normalized equivalent stress through the thickness of the thick cylinder for different values of the creep exponent. 43
Fig. 4.1: (a) Schematic diagram of the indeterminate beam. (b) Expected distribution of the R-Node stress along the beam for the initial and an intermediate iteration. (c) Two-bar model representing the collapse of the indeterminate beam. 51
Fig. 4.2: (a) Torispherical head configuration. (b) R-Node stress distribution for the first and second iteration [19] 52
Fig. 4.3: Flow-chart of UMAT subroutine 53
Fig. 4.4: Schematic diagram showing the procedure for calculating the R-Node stress and location in a plane element using the stress distribution within the element. 55
Fig. 4.5: Meshing of the plane-strain cylinder with 6 inches inner diameter and 18 inches outer diameter. 57

Fig. 4.6: Stress redistribution through the cylinder wall using elastic compensation methods. 57
Fig. 4.7: Comparison of the convergence of the limit load multipliers calculated for the thick cylinder. 58
Fig. 4.8: Schematic diagram of the indeterminate beam with length of 20 inches and unit height and depth subjected to distribution load. 59
Fig. 4.9: Comparison of equivalent stress distribution along the height of the beam in the initial iteration to that in the last iteration of stress redistribution. 60
Fig. 4.10: R-Node stress distribution along the beam for the first and last iteration of the redistribution analysis. 61
Fig. 4.11: Comparison of the convergence of the limit load multipliers of the indeterminate beam. 61
Fig. 4.12: Schematic diagram of the oblique nozzle as modeled by Sang et al [20]. 63
Fig. 4.13: Mesh of the oblique nozzle using shell elements. 63
Fig. 4.14: Comparison of the convergence of the limit load multipliers of the oblique nozzle. 64
Fig. 5.1: Schematic diagram of the cantilever beam with length of 20 inches and unit height and depth subjected to distribution load and axial force 67
Fig. 5.2: Iterative-limit load analysis illustrated on the limit curve of the determinate beam. 68
Fig. 5.3: Iterative-limit load analysis with an improved estimate of the initial value illustrated on the limit curve of the determinate beam. 69
Fig. 5.4: Limit load multiplier of the cantilever beam subjected to 90 N/m distributed load and 2000 N axial load. 71
Fig. 5.5: Limit curve of the cantilever beam. 72
Fig. 5.6: Schematic diagram of the pipe bend. 74
Fig. 5.7: Meshing of the pipe bend. 75
Fig. 5.8: Limit load of the pipe bend. 76
Fig. 5.9: The limit curves of the in-plane closing moment versus the internal pressure of the pipe bend. 77

Fig. 5.10: The limit curves of the out-of-plane moment versus the internal pressure of the pipe bend. 77
Fig. 6.1: Coordinates of Cross Section 80
Fig. 6.2: GLOSS diagram. 81
Fig. 6.3: Bending stress distribution of the initial elastic and redistribution analyses..... 83
Fig. 6.4: Membrane plus bending stress distribution of the initial elastic and redistribution analyses. 83
Fig. 6.5: Schematic diagram of the indeterminate beam. 86
Fig. 6.6: Primary stress distribution along the beam. 88
Fig. 6.7: Primary stress for combined membrane and bending loads. 89
Fig. 6.8: Schematic of the geometry with the expected hinge locations..... 90
Fig. 6.9: Primary stress distribution along the vessel wall. 91
Fig. 6.10: Schematic diagram of the straight nozzle with $Rm = 50\text{mm}$ and $rm = 15\text{mm}$. 92
Fig. 7.1: Variation of m_u with Elastic Iterations 96
Fig. 7.2: Variation of m_u and m' with linear elastic iterations [2] 97
Fig. 7.3: Determination of Reference Volume..... 98
Fig. 7.4: Schematic diagram of the indeterminate beam. 99
Fig. 7.5: Variation of the mu with the volume at different iterations 100
Fig. 7.6: The upper bound multiplier of the indeterminate beam calculated based on selected partial volumes. 101
Fig. 7.7: Shaded diagram of the FE model of the indeterminate beam showing the reference volume (black area) and the remainder volume (gray area.) 101
Fig. 7.8: Comparison of the convergence of the limit load using reference volume method with other limit load analysis methods. 102
Fig. 7.9: Meshing of the plane-strain cylinder..... 103
Fig. 7.10: Results of the analysis of the thick cylinder..... 103
Fig. 7.11: Schematic of the geometry with the expected hinge locations..... 104
Fig. 7.12: Meshing of the plane-strain cylinder..... 105
Fig. 7.13: Schematic diagram of the cross-section of an oblique nozzle used for experimental analysis by Sang et al. [20]. 106

Fig. 7.14: Meshing of the oblique nozzle. 107
Fig. 7.15: Upper bound multiplier based on partial volumes. 108
Fig. 7.16: Comparison of the limit load multipliers..... 108

LIST OF TABLES

Table 2.1: Portion of ASME Section III – Table NB-3217-1 25
Table 4.1: Oblique nozzle analysis results 64
Table 5.1: Number of R-Node iterations versus pressure 78
Table 6.1: Indeterminate beam analysis 87
Table 6.2: Axisymmetric pressure vessel analysis. 91
Table 6.3: Straight nozzle analysis. 93
Table 6.4: Analysis times (seconds). 93

LIST OF SYMBOLS

B, n Creep parameters of the power law

$E_{n,i}$ Elastic modulus of integration point n for iteration i

m' Mura's lower-bound limit load multiplier

m_L Classical lower-bound limit load multiplier

m_u Classical upper-bound limit load multiplier

Δm Difference between the upper-bound limit load multiplier based on the total and the that based on the reference volume

M_0 Moment caused by a plastic hinge

M_y Moment to cause initial yielding

N_T Total number of elements in a model

N_R Number of elements in the reference volume

P Axial load, internal pressure or applied load

P_L Limit load

P_y Axial force to cause initial yielding

q elastic modulus adjustment index

r Radius within the walls of a cylinder

r_i Inner radius of a cylinder

r_o Outer radius of a cylinder

S_m Allowable stress defined by the ASME code

V_k Volume of element k

V_P A partial volume of the model

V_R Reference volume

V_T Total volume

$\dot{W}_{\min}$ Minimum work rate

$\dot{W}_{ref}$ Minimum work rate at steady state

W_y Distributed load to cause initial yield

W_{h1} Distributed load to cause first plastic hinge

W_{LL} Distributed load to cause plastic collapse

$\varepsilon_{e,i}$ Equivalent strain for iteration i at the centroid

ζ Linear elastic iteration variable

σ_a Arbitrary stress for modulus adjustment

$\sigma_{en,i}$ Equivalent stress for iteration i at every material calculation point

$\sigma_{e,i}$ Equivalent stress for iteration i at the centroid

$\bar{\sigma}_n$ Combined R-Node stress

σ_r Radial stress

σ_{ref} Reference stress

σ_{ss} Steady state stress

σ_y Yield stress limit

σ_θ Hoop stress

CHAPTER 1

INTRODUCTION

1.1 General Background

Limit load analysis is an important tool in the design process of mechanical components to ensure their functionality within their operating conditions. It determines the load that would cause plastic collapse. In addition, it helps in providing an assessment of the behavior of the component for other modes of failure. Also, by applying the appropriate boundary conditions and geometrical behavior, limit analysis can provide an assessment of the integrity of a mechanical component during operation.

Limit loads are mainly determined using elastic-plastic analysis. For simple cases of loadings and geometrical configurations, exact limit loads can be determined using analytical approaches. For complex problems, some assumptions are made to ultimately attempt an approximate analytical solution. Such procedures are categorized into lower-bound and upper-bound solutions. More complex problems are solved using iterative numerical methods such as the finite element analysis. Although the elastic-plastic analysis gives a relatively accurate solution, it consumes a huge amount of time and requires advanced computing resources. In addition, considerable input and experience are required in defining the convergence criteria of the solution and the conditions for the limit load.

The complexity of the inelastic numerical solution and the significance of the limit load calculated in the design of mechanical components motivated the development of alternative simplified methods. Several approaches have been developed to calculate the limit load. The basic and simple analysis procedure is the linear elastic analysis which is used in several approaches to find the upper and lower bound limit loads. The ASME

codes have set some guidelines to interpret the results of a linear elastic analysis and categorize the stresses in primary, secondary and peak stresses. These categories are used to find a design load that will avoid most of the failure modes. Inelastic analysis may be used to verify the results of the categorization.

Other robust methods are being developed to find the exact limit load solution by using iterative elastic analyses. These methods have been developed extensively over the past decades. They utilize the approximate approaches of elastic modulus adjustment procedures to investigate the stress redistribution until the distribution corresponding to collapse state is reached. The main concern in the repeated elastic methods is their convergence. It is necessary to be able to confidently use a tool guaranteeing a result within an acceptable range of error and minimum computational effort. In addition, the simplified tools should be applicable to all types of geometries and load types.

1.2 Limit Load Analysis

Knowing the stress field at any stage of the redistribution, Mura [1] has developed the lower bound multipliers m' which was then used to develop multipliers m_1^0 and m_2^0 as upper bound solution and m_α and m_β as lower bound solution. The most commonly used multiplier is the classical lower bound multiplier (m_L) in which the limit load is calculated based on the maximum stress and the yield criteria. Mura used the upper bound m_1^0 and m_L to calculate m' which was shown to be a lower bound solution. Similarly, Mangalaramanan and Seshadri [2] developed the m_α method, and Seshadri and Indermohan [3] developed the m_β method as lower bound solutions.

Seshadri and Fernando [4] developed the R-Node as a tool to find the reference stresses in a component and their locations. This is done by comparing the overall stress distribution in every redistribution analysis iteration to the original elastic analysis.

Hence, the limit load would be directly proportional to the reference stress as explained in earlier research. The reference stress has an advantage of being in direct equilibrium with the externally applied load. Therefore, it is shown to be a lower bound solution.

At the state of collapse, a component could have large dead zones (zero stress) and part of it will have both elastic and plastic stresses. This dead zone might not necessarily be evident in the initial stress analysis. However, due to the softening of the volume that has stresses beyond the yield value, the remaining elastic zone will tend to relax. Seshadri and Mangalaramanan [5] observed that the limit load multipliers calculated based on the portion of the total volume (the reference volume) that has non-zero stress at the state of collapse would be equal for every iteration starting from the first elastic analysis. Hence, they developed a procedure in which the reference volume is calculated starting from the second iteration of the redistribution analysis.

1.3 Objective of the thesis

The purpose of this research is to extend the application of the robust limit load methods. The objectives are:

1. Verification of the applicability of the R-Node method as a lower-bound solution to various types of problems. A complete computational algorithm for the R-Node determination is developed such that it can be incorporated in commercial finite element codes for 2D and 3D geometries.

2. The development of the R-Node method in finding the limiting value of a single load in a component subjected to multiple loads. This is achieved by performing several successive limit load analyses using the R-Node procedure which is made feasible through the fast convergence behavior of the R-Node analysis.

3. Propose the use of the R-Node stress as a stress classification tool for the ASME codes by virtue of the method being a lower bound technique for the reference stresses. The reference stresses are classified as primary stresses, and their locations are used to define the stress classification lines for the ASME guidelines.

4. The concept of the reference volume is derived using the classical upper bound solution and its use as the most accurate solution among the different methods is verified.

1.4 Organization of the thesis

Chapter 1 illustrates the significance of the limit load analysis and the currently used methods for calculating the limit load multipliers. The objectives and organization of this thesis are presented.

Chapter 2 details a literature review of the methods used in stress redistribution and limit load multiplier calculations. Their basis and assumption are clearly stated. The elastic modulus adjustment procedure which is used in the R-Node analysis method is demonstrated. The stress classification procedure according to the ASME codes is explained in details. The advantages and disadvantages of the above methods are illustrated.

The reference stress concept is the basis of the R-Node method. The concept is illustrated in details in chapter 3. The use of the reference stress in the assessment of the creep deformation is illustrated and applied to sample problems. The derivation of the R-Node analysis method using the reference stress concept is explained.

In chapter 4, the iterative R-Node method is introduced. Details of the suggested method are illustrated. In addition, the algorithm for the pre-processing, solution and

post-processing numerical procedures of the analysis are explained. The procedures are used to solve some problems showing its applicability to different geometries, element types and levels of complexity.

In general, pressure components are subjected to several loads simultaneously. The use of the R-Node analysis method is extended in chapter 5 to find the limit value of a single load in a system of other fixed loads is explained. The procedure is verified for a cantilever beam modeled using plane elements. Hence, a pipe bend subjected to internal pressure and bending moments is analyzed to find the limit moment for different values of the internal pressure.

The reference volume concept was introduced in previous work as a tool for calculating the limit load. The method calculate the upper bound multiplier, m^0, based on the active volume of the considered component that undergoes plastic deformation. This accelerates the convergence of the limit load analysis. In chapter 6, the reference volume concept is redeveloped using the classical upper bound multiplier, m_u.

Chapter 7 illustrates the use of the R-Node method in stress classification. The advantages of the method are explained in comparison to the presently used stress classification procedure. The method is applied to several problems modeled using different element types. The results of the analysis are presented to clarify the benefits of using the R-Node method in the ASME stress classification procedure.

Finally, chapter 8 summarizes the advantages of the proposed methods and the original contributions of the thesis are listed at the end of the chapter. Suggestions are also provided for carrying out future work along the lines of this thesis.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview

Several approaches have been developed in order to calculate the limit load of components. The common method is to perform an elastic-plastic analysis of the component, and to determine the response to the applied load. The ASME code has some guidelines for calculating the limit load from the results of the elastic-plastic analysis. However, the inelastic methods of stress analysis can have several complexities. Therefore, robust methods are desirable in order to simplify the stress analysis procedure and for limit load estimation.

In the past decades, the Elastic Modulus Adjustment Procedure (EMAP) was developed in which linear elastic analysis is employed to find the inelastic-like stress distribution due to a given applied loading. The method was used mainly for the estimation of the limit load. The GLOSS method was also developed by utilizing the elastic modulus modification concept to find the stress redistribution due to multiaxial creep relaxation. Hence, the stress distribution is used to calculate the reference stress, which can be used to calculate the limit load.

2.2 Robust Methods in Stress Distribution Analysis

2.2.1 The Elastic Compensation Method (ECM)

The aim of ECM is to establish an inelastic-like stress field by modifying the local elastic modulus in order to obtain the necessary stress redistribution. Numerous sets of statically and kinematically admissible distributions can be generated in this manner, which enable calculation of both lower and upper bounds limit loads.

Mackenzie and Boyle [6] used the concept behind the GLOSS R-Node method to develop a procedure for modifying the elastic modulus in several iterations to reach a statically admissible stress field equivalent to that of the limit case. At the limit state, further modulus modification in the elastic modulus will not affect the stress distribution. The elastic modulus of each element in the linear elastic finite element scheme is modified as

$$E_k^{i+1} = \left(\frac{\sigma_{arb}}{\sigma_{ek}^i} \right) E_k^i \tag{2.1}$$

where σ_{arb} is an arbitrary non-zero stress value less than the maximum stress in the structure, σ_{ek}^i is the equivalent stress and i is the iteration index ($i=1$ for the initial elastic analysis). To guarantee the convergence of the stress redistribution, the arbitrary stress σ_{arb}^i is calculated using the expression

$$\sigma_{arb}^i = \left[\frac{\int_{V_T} \left(\sigma_e^i \right)^2 dV}{V_T} \right]^{1/2} \tag{2.2}$$

In order to make use of equation (2.2) for the FEA solution, it can be written as

$$\sigma_{arb}^i = \left[\frac{\sum_{k=1}^{N} \left(\sigma_{ek}^i \right)^2 V_k}{V_T} \right]^{1/2} \tag{2.3}$$

This formula describes how the elastic modulus at a location with the equivalent stress σ_{ek}^{j} is updated every iteration. This procedure continues until suitable convergence of a subsequent iteration is achieved.

2.2.2 The Equivalent Strain Energy Density (ESED) Method

Molski and Glinka [7] used the concept of total strain energy to find the plastic stress and strain in notch root equivalent to the elastic stress-strain field. The strain energy per unit volume in an elastic stress distribution is given by

$$U = \frac{\sigma^2}{2E} \tag{2.4}$$

When the stress at the notch root increases beyond yield, plastic deformation occurs. For a given load, it is assumed that the ratio of the energy absorbed in an elastic response to that in an elastic-plastic response does not change due to small plastic region. The relatively high volume of the elastic material surrounding the small plastic zone controls the amount of strain energy absorbed by the plastic zone. However, the material stress-strain relationship is used to determine the strain energy absorbed at the notch root. If elastic-perfectly plastic material model is used, the total strain in a plastic zone will be given as

$$\varepsilon_t = \varepsilon_e + \varepsilon_p \tag{2.5}$$

The strain energy per unit volume in the plastic zone will be

$$U = \frac{\sigma_y^2}{2E} + \sigma_y \varepsilon_p \tag{2.6}$$

Hence, equating the total strain energies of the elastic stress distribution shown in equation (2.4) and the plastic redistribution shown in equation (2.6), the plastic strain can be expressed as

$$\varepsilon_p = \frac{\sigma^2 - \sigma_y^2}{2\sigma_y E} \tag{2.7}$$

The secant modulus E_s in the plastic zone is now expressed as the ratio of the yield stress to the total strain ε_t resulting in the following equation

$$E_s = \frac{2\sigma_y^2}{\sigma_y^2 + \sigma^2} E \tag{2.8}$$

where E is the actual elastic modulus. Adibi-Asl et al [8] used equation (2.8) to modify the elastic modulus in the EMAP. The finite element implementation of the equation is expressed as

$$E_k^{i+1} = \frac{2\sigma_{arb}^2}{\sigma_{arb}^2 + \left(\sigma_{ek}^i\right)^2} E_k^i \tag{2.9}$$

where E_k^i is the elastic modulus of element k at increment i. σ_{arb} is similar to equation (2.1). Figure 2.2 shows a graphical interpretation of the ESED method.

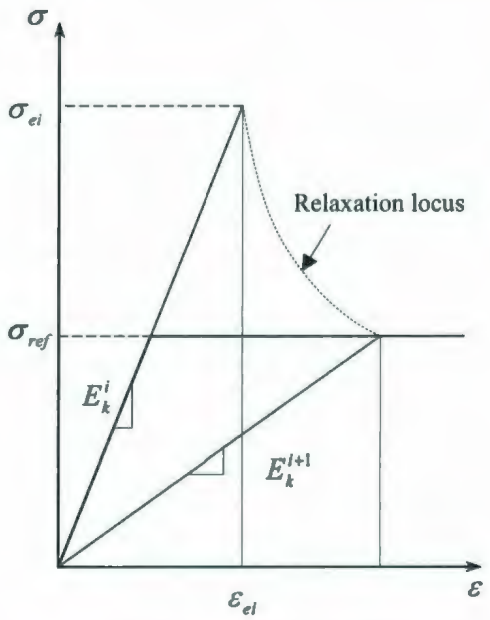


Fig. 2.2: Schematic of the ESED method

2.3 Limit Load Analysis

The limit load analysis is performed using either linear elastic or elastic-plastic analysis. In the linear elastic analysis, the load is applied to a component with an arbitrary value, and a multiplier is calculated using the generated stress field. Hence, the limit load is given by

$$P_L = mP \tag{2.10}$$

where m is the limit load multiplier and P is the applied load. There are several methods that have been developed to calculate the limit load multiplier as an upper or a lower bound. The most commonly used method is the classical lower-bound multiplier, which is given by

$$m_L = \frac{\sigma_y}{[\sigma_e]_{\max}} \tag{2.11}$$

where $[\sigma_e]_{\max}$ is the maximum equivalent stress calculated when applying the arbitrary load P. Hence, the limit load will generate a stress distribution that will be all below the yield limit.

2.3.1 Mura's Lower Bound Theorem

Mura et al [1] have utilized the variational principles to evaluate a lower bound multiplier for a component subjected to prescribed surface tractions. The solution is based on the assumption of a perfectly plastic material. The classical lower bound solution is based on the concept of having a statically admissible stress field within the yield surface. The solution based on Mura's theorem eliminates this by using the concept of the integral mean of yield criterion. Mura et al [1] showed that a lower bound solution for the safety factor m can be achieved by minimizing the functional

$$F\left[v_i, s_{ij}, \sigma, R_i, m, \mu, \varphi\right] = \int_V s_{ij} \tfrac{1}{2}\left(v_{i,j} + v_{j,i}\right) dV + \int_V \sigma \delta_{ij} v_{i,j} dV - \int_{S_V} R_i v_i dS - m\left(\int_{S_T} T_i v_i dS - 1\right) - \int_V \mu\left[f\left(s_{ij}\right) + \varphi^2\right] dV \tag{2.12}$$

where v_i is the velocity, s_{ij} is the stress deviation for the actual solution of a limit case, T_i is the traction on the surface denoted by S_T, and σ, R_i, m, μ and φ are the Lagrangian multipliers. σ is a point-function defining the mean stress, R_i is a point-function defining the reaction on the surface S_V, m is the safety factor, μ is the scalar of proportionality, and φ is the yield parameter. $f\left(s_{ij}\right)$ is the yield function defined as

$$f\left(s_{ij}\right) = \tfrac{1}{2} s_{ij} s_{ij} - k^2 \tag{2.13}$$

where $k^2 = \sigma_y^2/3$. The Lagrangian multipliers are employed in order to determine the minimal conditions of the functional. Taking the variation of the functional yields the natural conditions

$$\tfrac{1}{2}\left(v_{i,j} + v_{j,i}\right) = \mu \frac{\partial f}{\partial s_{ij}} \text{ in } V \tag{2.14}$$

$$\mu \geq 0 \tag{2.15}$$

$$\left(s_{ij} + \delta_{ij}\sigma\right)_{,i} = 0 \text{ in } V \tag{2.16}$$

$$\left(s_{ij} + \delta_{ij}\sigma\right) n_i = m T_i \text{ in } S_T \tag{2.17}$$

$$\left(s_{ij} + \delta_{ij}\sigma\right) n_i = R_i \text{ in } S_V \tag{2.18}$$

$$f\left(s_{ij}\right) + \varphi^2 = 0 \text{ in } V \tag{2.19}$$

$$\mu\varphi = 0 \text{ in } V \tag{2.20}$$

$$\delta_{ij} v_{i,j} = 0 \text{ in } V \tag{2.21}$$

$$v_i = 0 \text{ in } S_V \tag{2.22}$$

$$\int_{S_T} T_i v_i dS = 1 \tag{2.23}$$

Equation (2.14) is the plastic flow potential, equations (2.16) to (2.18) are the equilibrium conditions, and equations (2.21) to (2.23) define a kinematically admissible velocity field. Conditions (2.19) and (2.20) define the admissible domain of the stress field, i.e.

$$f\left(s_{ij}\right) = 0 \text{ if } \mu > 0 \tag{2.24}$$

$$f\left(s_{ij}\right) \le 0 \text{ if } \mu = 0 \tag{2.25}$$

Hence, for example, a stress field that is at the limit state would satisfy condition (2.24).

Considering an arbitrary solution expressed in terms of the actual solution as $v_i^0 = v_i + \delta v_i$, $s_{ij}^0 = s_{ij} + \delta s_{ij}$, ..., the equilibrium equations being a requirement for a statically admissible stress field can be written as

$$\left(s_{ij}^0 + \delta_{ij}\sigma^0\right)_{,j} = 0 \text{ in } V \tag{2.26}$$

$$\left(s_{ij}^0 + \delta_{ij}\sigma^0\right) n_i = m^0 T_i \text{ in } S_T \tag{2.27}$$

$$\left(s_{ij}^0 + \delta_{ij}\sigma^0\right) n_i = R_i^0 \text{ in } S_V \tag{2.28}$$

Hence, using the above equilibrium equations, the functional F can be found for an arbitrary stress field using the actual solution expressed as

$$F = m - \int_V \mu\left[\tfrac{1}{2}\delta s_{ij}\delta s_{ij} + \left(\delta\varphi\right)^2\right] dV - \int_V \delta\mu\left[f\left(s_{ij}^0\right) + \left(\varphi^0\right)^2\right] dV \tag{2.29}$$

Also substituting with the arbitrary solution v_i^0, s_{ij}^0, ... into the funcational and integrating yields

$$F = m^0 - \int_V \mu\left[f\left(s_{ij}^0\right) + \left(\varphi^0\right)^2\right] dV \tag{2.30}$$

The integral mean of yield criterion for an arbitrary solution can be expressed as

$$\int_V \mu^0 \left[f\left(s_{ij}^0\right) + \left(\varphi^0\right)^2 \right] dV = 0 \tag{2.31}$$

Hence,

$$F = m^0 \tag{2.32}$$

Since the first integral in equation (2.29) is a definite positive value, it follows that

$$F \le m - \int_V \delta\mu \left[f\left(s_{ij}^0\right) + \left(\varphi^0\right)^2 \right] dV \tag{2.33}$$

Since $\mu^0 = \mu + \delta\mu$, equation (2.31) can be written as

$$-\int_V \delta\mu \left[f\left(s_{ij}^0\right) + \left(\varphi^0\right)^2 \right] dV = \int_V \mu \left[f\left(s_{ij}^0\right) + \left(\varphi^0\right)^2 \right] dV \tag{2.34}$$

Substituting the equation (2.34) into (2.33) and taking the maximum of the integrand gives

$$F \le m + \max\left[f\left(s_{ij}^0\right) + \left(\varphi^0\right)^2 \right] \int_V \mu dV \tag{2.35}$$

Since

$$\begin{aligned} m &= \int_{S_T} T_i v_i dS = \int_S \left(s_{ij} + \delta_{ij}\sigma\right) n_i v_i dS = \int_V \left(s_{ij} + \delta_{ij}\sigma\right) dS + \int_V \left(s_{ij} + \delta_{ij}\sigma\right) v_{i,j} dS \\ &= \int_V s_{ij} \tfrac{1}{2}\left(v_{i,j} + v_{j,i}\right) dV = \int_V s_{ij} \mu s_{ij} dV = 2k^2 \int_V \mu dV \end{aligned} \tag{2.36}$$

Rearranging equation (2.36)

$$\int_V \mu dV = \frac{m}{2k^2} \tag{2.37}$$

Substituting equation (2.37) into equation (2.35)

$$m^0 \le m + \frac{\max\left[f\left(s_{ij}^0\right) + \left(\varphi^0\right)^2 \right]}{2k^2} \tag{2.38}$$

Rearranging equation (2.38) gives a lower bound multiplier m' expressed as

$$m' = \frac{m^0}{1 + \frac{\max\left[f\left(s_{ij}^0\right) + \left(\varphi^0\right)^2\right]}{2k^2}} \leq m \tag{2.39}$$

2.3.2 Classical upper bound multiplier

The classical upper bound solution is based on the comparison of the response of an assumed solution to a postulated collapse mechanism achieved by a kinematically admissible solution. Considering a body subjected to some distribution of tractions T, the factor m by which the loading can be increased before the solid collapses (m is effectively the factor of safety) is estimated. It is assumed that the component will collapse when subjected to loading mT. Assuming s_{ij} is the deviatoric stress of the exact solution and $\dot{\varepsilon}_{ij}$ is a kinematically admissible solution for the given problem, it can be deduced using Schwarz's inequality that

$$0 \leq s_{ij}\dot{\varepsilon}_{ij} \leq \sqrt{s_{ij}s_{ij}}\sqrt{\dot{\varepsilon}_{ij}^p \dot{\varepsilon}_{ij}^p} \tag{2.40}$$

On the basis that $\sigma_y = \sqrt{3s_{ij}s_{ij}/2}$ for the exact solution, equation (2.40) becomes

$$s_{ij}\dot{\varepsilon}_{ij}^p \leq \sqrt{\frac{2}{3}}\sigma_y \sqrt{\dot{\varepsilon}_{ij}^p \dot{\varepsilon}_{ij}^p} = \sigma_y \dot{\varepsilon}_e \tag{2.41}$$

Using the principle of virtual work and integrating both sides gives

$$\int_V \sigma_y \dot{\varepsilon}_e dV - \int_{S_F} mT_i \dot{u}_i dA \geq 0 \tag{2.42}$$

Assuming an elastic stress field s_{ij}^* calculated using the elastic modulus adjustment procedures corresponding to the loading T, the energy balance using the principle of virtual work yields

$$\int_V s_{ij}^* \dot{\varepsilon}_{ij}^p dV - \int_{S_F} T_i \dot{u}_i dA = 0 \tag{2.43}$$

Substituting equation (2.43) into equation (2.42)

$$\int_V \sigma_y \dot{\varepsilon}_{ij}^p dV - m \int_V s_{ij}^* \dot{\varepsilon}_{ij}^p dV \geq 0 \tag{2.44}$$

Rearranging gives

$$m \leq \frac{\int_V \sigma_y \dot{\varepsilon}_e dV}{\int_V s_{ij}^* \dot{\varepsilon}_{ij}^p dV} = m_u \tag{2.45}$$

This equation is translated to the finite elements form as

$$m_u = \sigma_y \frac{\sum_{k=1}^{N} \varepsilon_{ek} V_k}{\sum_{k=1}^{N} \sigma_{ek} \varepsilon_{ek} V_k} \tag{2.46}$$

2.3.3 The m^0 Upper Bound Multiplier

Mangalaramanan and Seshadri [2] have developed an upper-bound multiplier from Mura's formulation. Considering an arbitrary state of stress, equation (2.12) can be written as

$$F = m^0 - \int_{V_T} \mu^0 \left[f\left(s_{ij}^0\right) + \left(\varphi^0\right)^2 \right] dV \tag{2.47}$$

It was shown that the Von Mises yield criteria can be expressed as

$$f\left(s_{ij}^0\right) = \frac{1}{3}\left[\left(m^0 \sigma_e^0\right)^2 - \sigma_y^2 \right] \tag{2.48}$$

Substituting equation (2.48) into equation (2.47) gives

$$F = m^0 - \int_{V_T} \frac{\mu^0}{3}\left[\left(m^0\sigma_e^0\right)^2 - \sigma_y^2 + 3\left(\varphi^0\right)^2\right]dV \tag{2.49}$$

Applying the stationary conditions for the above functional given by $\delta F = 0$ leads to

$$\frac{\partial F}{\partial m^0} = 0; \frac{\partial F}{\partial \mu^0} = 0; \frac{\partial F}{\partial \varphi^0} = 0 \tag{2.50}$$

The expression for m^0 can be obtained as

$$m_1^0 = \sqrt{\frac{\int_{V_T} \sigma_y^2 dV}{\int_{V_T}\left(\sigma_e^0\right)^2 dV}} = \sigma_y \sqrt{\frac{V_T}{\int_{V_T}\left(\sigma_e^0\right)^2 dV}} \tag{2.51}$$

This equation is translated into a summation form in order to be applied using the results of a finite element analysis. The finite element form of equation (2.51) is

$$m_1^0 = \sigma_y \sqrt{\frac{V_T}{\sum_{k=1}^{N}\left(\sigma_{ek}\right)^2 V_k}} \tag{2.52}$$

where σ_{ek} and ε_{ek} are the stress and strain at the centroid of element k in a model comprising N elements.

Pan and Seshadri [9] have modified the above equation to account for the effect of variable flow parameter, μ. The new multiplier is given by

$$m_2^0 = \sigma_y \sqrt{\frac{\sum_{k=1}^{N} V_k / E_{sk}}{\sum_{k=1}^{N} \sigma_{ek}^2 V_k / E_{sk}}} \tag{2.53}$$

Substituting for the secant modulus $E_s = \sigma_e/\varepsilon_e$, m_2^0 becomes

$$m_2^0 = \sigma_y \sqrt{\frac{\sum_{k=1}^{N} \varepsilon_{ek} V_k / \sigma_{ek}}{\sum_{k=1}^{N} \varepsilon_{ek} \sigma_{ek} V_k}} \tag{2.54}$$

2.3.4 The m_α-Method

The m_α-method is an improved technique for obtaining the lower bound limit load that was developed by Seshadri and Manglaramanan [2] using Mura's lower bound formulation. It is based on the idea of finding an intermediate multiplier between the lower (m') and upper (m^0) bound solutions. Mura's lower bound multiplier (m') is given by

$$m' = \frac{2m^0}{1 + \left(\frac{m^0}{m_L}\right)^2} \leq m \tag{2.55}$$

Using the EMAP or the ECM methods, both (m') and upper (m^0) are calculated for every iteration of stress redistribution. Therefore, the closer the stress distributions are to the limit state, the closer the multipliers will be to the exact solution.

An independent iteration variable ζ is assumed to characterize continous redistribution. Figure 2.3 is a schematic plot of the multiplier as a function of the iteration variable.

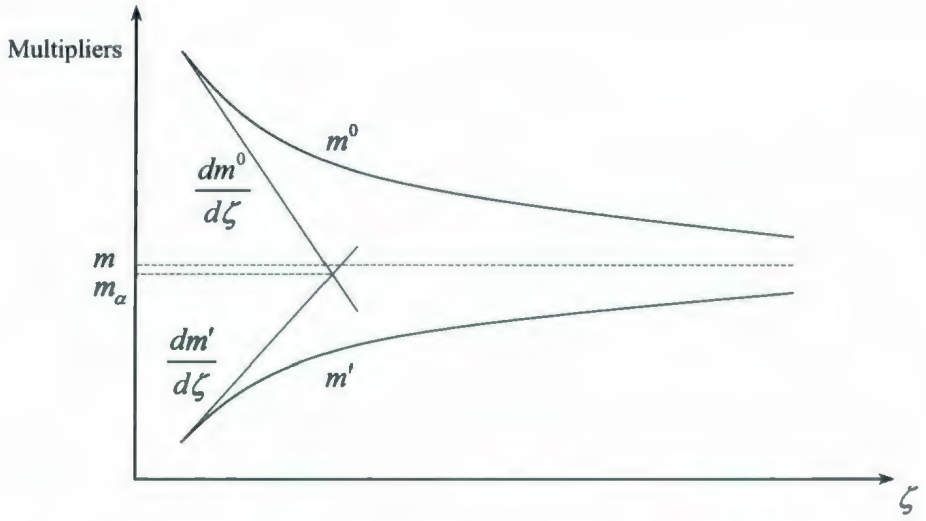


Fig. 2.3: The variation of the upper bound and the lower bound multipliers with the iteration variable

The lower bound multiplier expressed in terms of finite differences is given by

$$\Delta m' = \left(\frac{\partial m'}{\partial m^0}\right)_{\zeta_i} \Delta m^0 + \left(\frac{\partial m'}{\partial \frac{1}{m_L}}\right)_{\zeta_i} \Delta \frac{1}{m_L} \tag{2.56}$$

From equation (2.55) the partial derivatives are expressed as

$$\frac{\partial m'}{\partial m^0} = \frac{2}{1+\left(\frac{m^0}{m_L}\right)^2} - \frac{4}{\left(1+\left(\frac{m^0}{m_L}\right)^2\right)^2}\left(\frac{m^0}{m_L}\right)^2$$

$$= 2\frac{1+\left(\frac{m^0}{m_L}\right)^2}{\left(1+\left(\frac{m^0}{m_L}\right)^2\right)^2} - \frac{4}{\left(1+\left(\frac{m^0}{m_L}\right)^2\right)^2}\left(\frac{m^0}{m_L}\right)^2$$

$$= \frac{2-2\left(\frac{m^0}{m_L}\right)^2}{\left(1+\left(\frac{m^0}{m_L}\right)^2\right)^2}$$

$$\frac{\partial m'}{\partial \frac{1}{m_L}} = -\frac{2m^0}{\left(1+\left(\frac{m^0}{m_L}\right)^2\right)^2} \cdot 2\left(\frac{m^0}{m_L}\right) \cdot m^0$$

$$= -\frac{4}{\left(1+\left(\frac{m^0}{m_L}\right)^2\right)^2} \cdot \frac{\left(m^0\right)^3}{m_L} \tag{2.57}$$

The finite differences in equation (2.56) are expressed as

$$\Delta m' = m_a - m' = m_a - \frac{2m^0}{2+\left(\frac{m^0}{m_L}\right)^2}$$

$$\Delta m^0 = m_a - m^0 \tag{2.58}$$

$$\Delta\frac{1}{m_L} = \frac{1}{m_a} - \frac{1}{m_L}$$

Substituting equations (2.57) and (2.58) into equation (2.56) gives

$$m_\alpha - \frac{2m^0}{1+\left(\frac{m^0}{m_L}\right)^2} = \left(\frac{2-2\left(\frac{m^0}{m_L}\right)^2}{\left(1+\left(\frac{m^0}{m_L}\right)^2\right)^2}\right)\left(m_\alpha - m^0\right) - \frac{4}{\left(1+\left(\frac{m^0}{m_L}\right)^2\right)^2}\cdot\frac{\left(m^0\right)^3}{m_L}\left(\frac{1}{m_\alpha}-\frac{1}{m_L}\right) \tag{2.59}$$

Rearranging the coefficients of m_α gives

$$Am_\alpha^2 + Bm_\alpha + C = 0 \tag{2.60}$$

where

$$A = \left(\frac{m^0}{m_L}\right)^4 + 4\left(\frac{m^0}{m_L}\right)^2 - 1,\ B = -8m^0\left(\frac{m^0}{m_L}\right)^2,\ C = 4\frac{\left(m^0\right)^3}{m_L} \tag{2.61}$$

Solving the above polynomial for the larger positive value of m_α gives

$$\begin{aligned} m_\alpha &= 2m^0\frac{2\left(\frac{m^0}{m_L}\right)^2 \pm \sqrt{\left(-\left(\frac{m^0}{m_L}\right)^4 + 4\left(\frac{m^0}{m_L}\right)^3 - 4\left(\frac{m^0}{m_L}\right)^2 + 1\right)\left(\frac{m^0}{m_L}\right)}}{\left(\frac{m^0}{m_L}\right)^4 + 4\left(\frac{m^0}{m_L}\right)^2 - 1} \\ &= 2m^0\frac{2\left(\frac{m^0}{m_L}\right)^2 \pm \sqrt{\frac{m^0}{m_L}\left(\frac{m^0}{m_L}-1\right)^2\left(1+\sqrt{2}-\frac{m^0}{m_L}\right)\left(\frac{m^0}{m_L}-1+\sqrt{2}\right)}}{\left(\left(\frac{m^0}{m_L}\right)^2+2-\sqrt{5}\right)\left(\left(\frac{m^0}{m_L}\right)^2+2+\sqrt{5}\right)} \end{aligned} \tag{2.62}$$

2.3.5 The GLOSS R-Node Method

The Generalized Local Stress Strain (GLOSS) R-Node analysis, developed by Seshadri (1991), is a simple systematic method for inelastic evaluation of components and structures on the basis of two linear elastic finite element analyses. The component is divided in to "local" and "remainder" regions. The local region undergoes inelastic deformation and the remainder region of the component remains elastic. This method relates the inelastic multiaxial stress redistribution in the local region to the uniaxial stress relaxation process, and assumes that the relation locus is linear for small to moderate plastic zone size. Inelastic response of the local region due to plasticity is simulated by artificially lowering its stiffness. Then the inelastic strain can be estimated from the two analysis results per point on the effective stress-strain curve.

The GLOSS analysis is based on the follow-up analysis due to creep relaxation. The creep and elastic strain rates are is given by

$$\dot{\varepsilon}_c = B\sigma^n,\ \dot{\varepsilon}_e = \frac{1}{E_0}\left(\frac{d\sigma_e}{dt}\right) \tag{2.63}$$

Since the total strain is fixed during creep relaxation, it can be deduced that

$$\frac{d\sigma_e}{dt} + \bar{\lambda} BE\sigma_e^n = 0 \tag{2.64}$$

where $\bar{\lambda}$ is the constraint or follow-up parameter. Hence, the relaxation modulus can be expressed in terms of $\bar{\lambda}$ as

$$\bar{E}_r = \frac{\bar{\lambda}}{\bar{\lambda} - 1} \tag{2.65}$$

where $\bar{E}_r = E_r / E_0$. The relaxation modulus is calculated using two linear elastic finite element analyses. The first analysis is carried out for a given component configuration that is subjected to various mechanical and thermal loadings, on the assumption that the entire material is linear elastic. The second analysis is then carried out after artificially

reducing the elastic modulii of all elements that exceed the yield stress, assuming an elastic-perfectly plastic constitutive relationship as follows

$$E_s = \left(\frac{\sigma_y}{\sigma_e}\right) E_0 \tag{2.66}$$

This is based on the assumption that the inelastic elements soften in a deformation controlled mode although, in reality, some follow-up might be present. All other elements in the component are left unchanged.

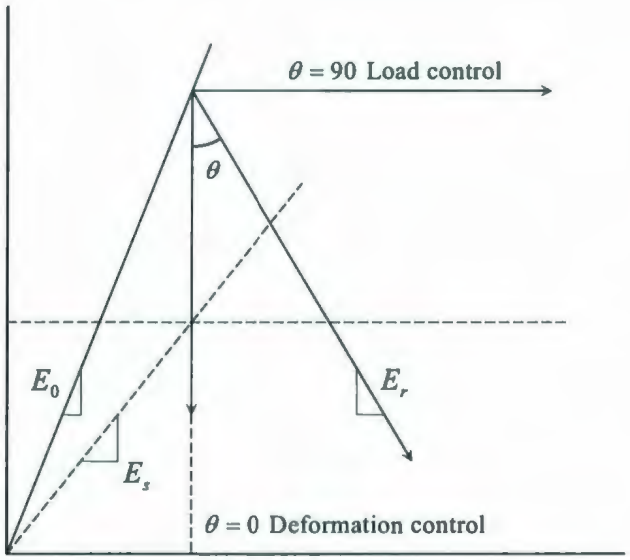


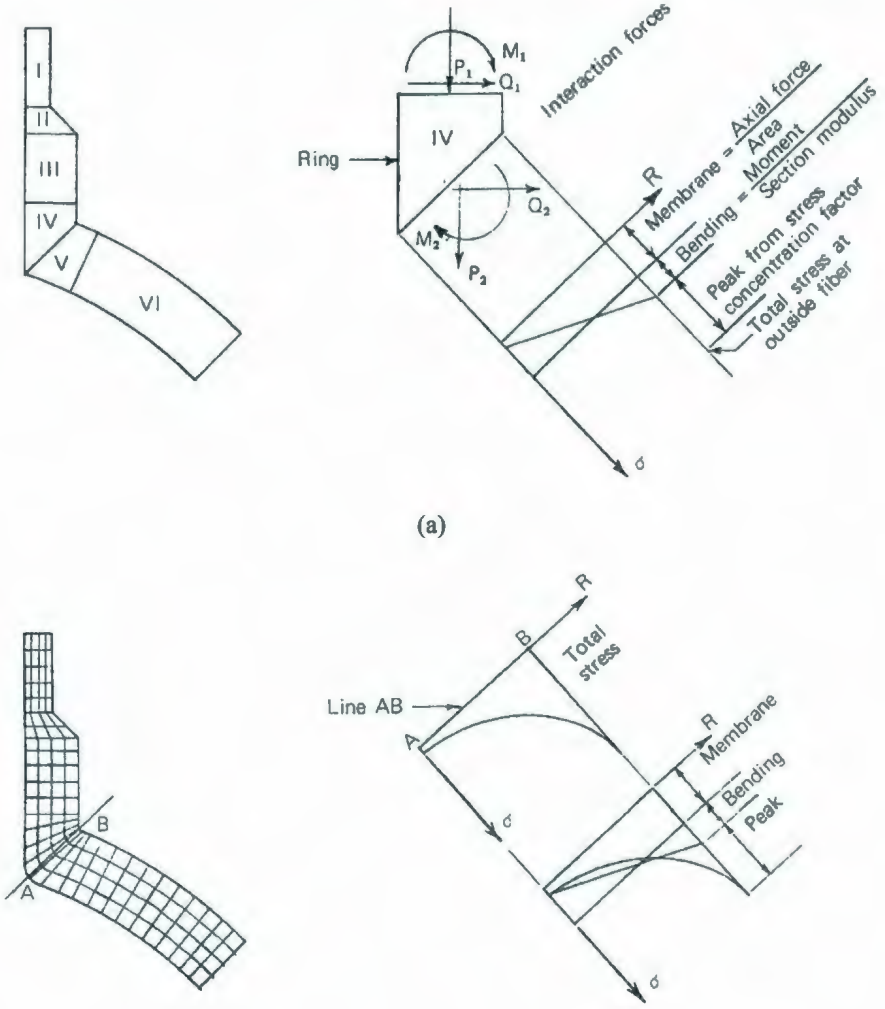
Fig. 2.1: GLOSS diagram

2.4 Stress Classification

2.4.1 Definition of Stress Classification

Overview

The stress fields obtained by linear elastic finite element analyses are estimated on the basis of the combined applied load and the reaction forces. Also, the generated stresses must be interpreted according to ASME Section III [10] which has outlined some guidelines for dividing the stresses into several parts according to the applied loadings and/or geometrical conditions. Each part is compared to an allowable limit. ASME has specified three different levels of allowable stresses which are given as S_m, $\frac{3}{2}S_m$, and $3S_m$ where S_m is the basic allowable stress calculated according to the material properties and a design safety factor. Each part or a combination of parts of the total stress is compared to one of the allowable stress levels. The main stress categories in pressure components are primary, secondary and peak stresses. The primary stress category is further divided into primary membrane and primary bending. ASME Section III [10] gives the allowable limit for each of these categories, and their combinations. Hence, it is necessary to identify a clear procedure to divide the total stresses obtained by finite elements analysis into the various stress categories defined by the ASME. Figure 2.4(a) shows a typical stress distribution across the thickness of an axisymmetric finite element model of a nozzle connected to a spherical head. It shows how the total stress is divided into the different categories of stress, where the total stress distribution calculated based on elastic analysis.



(a)

(b)

Fig. 2.4: Schematic diagram of sample results of stress analysis.

Using the finite element analysis results, the total stress is monitored at certain points and the distribution is then plotted as shown in Fig. 2.4(b). Hence, in order to find the values of the different categories of stress, "stress classification procedures" have been developed in conjunction with the ASME Section III guidelines in order to compare the results with the suitable allowable limits.

Criteria for Stress Categories

The ASME Section III includes the basic stress classification procedure according to the location, origin, and type. Hence, for the general pressure components, it describes the different stress categories according to these three aspects. Table 2.1 is part of Table NB-3217-1 of ASME Section III which describes the categorization of some selected components.

Investigation of these criteria allows the assignment of the stress to the proper classification, i.e., P_m general primary-membrane stress; P_L local primary-membrane stress; $P_L + P_b$ primary membrane plus primary bending stress; $P_L + P_b + Q$, primary plus secondary stress; and $P_L + P_b + Q + F$, total stress. The influence of location can be demonstrated by the following example.

Table 2.1: Portion of ASME Section III – Table NB-3217-1

Vessel Component	Location	Origin of Stress	Type of Stress	Classification
Cylindrical or spherical shell	Shell plate remote from discontinuities	Internal pressure	General membrane	P_m
			Gradient thru plate thickness	Q
		Axial thermal gradient	Membrane	Q
			Bending	Q
	Junction with head or flange	Internal pressure	Membrane	P_L
			Bending	Q
Nozzle	Nozzle wall	Internal pressure	General membrane	P_m
			General membrane	P_L
			Bending	Q
			Peak	F
		Differential expansion	Membrane	Q
			Bending	Q
			Peak	F

Outline of Two-Dimensional Stress Classification

The actual classification for two-dimensional stresses is carried out in three steps. First, the stress components are calculated on any desired plane. Second, the total stress is divided into membrane, bending, and peak and labeled primary, secondary, or peak according to location, origin, and type as discussed above. Finally, having classified the components, the principal stresses and stress intensities are calculated.

2.4.2 Method of Classifying Stresses

Stresses Calculated on a Line

The first step in the classification of stresses resulting from an axisymmetric solution using quadrilateral elements based on constant strain triangles is calculating stresses on a desired plane in the axisymmetric model. The plane is represented by a line in the cross section being modeled and will be called a "stress classification line" or simply a "stress line" in what follows. The stress line is described either by two nodes on opposite surfaces of the vessel or by the coordinates of two such points. Stresses in the global coordinate directions are calculated at evenly spaced points along the stress line by extrapolation or interpolation.

The final step in the presentation of stresses for classification is to rotate them to a local coordinate system which is parallel and perpendicular to the stress line. Having stresses on any desired plane in the vessel, the next step is to divide the total finite element stress into membrane, bending, and peak categories. Several unsuccessful methods tried will be discussed before listing the methods being used.

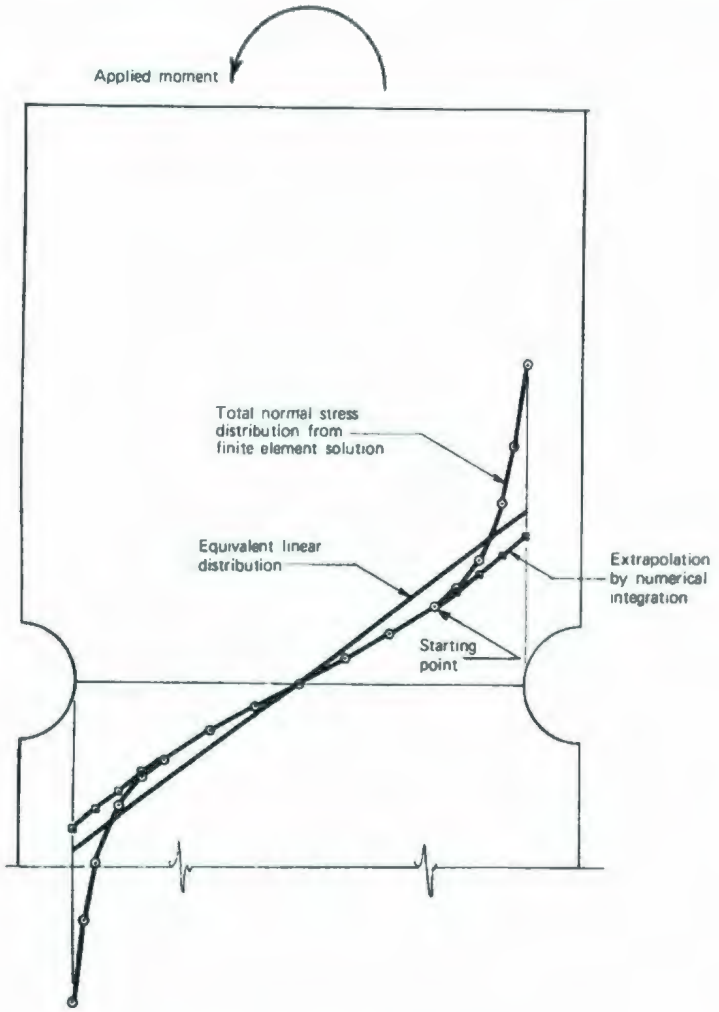


Fig. 2.5: Methods for eliminating peak stress.

After arriving at definitions for normal membrane, bending, and peak stresses, the actual calculations are straightforward. The following definitions were used for normal stresses:

Membrane stress - The constant portion of normal stress such that pure moment acts on a plane after the membrane is subtracted from the total stress.

Bending stress - The variable portion of normal stress equal to the equivalent linear stress or equal to the total stress minus membrane in areas where no peak stresses exist.

Peak stress - The portion of the normal stress which exists after subtracting membrane and bending from the total stress.

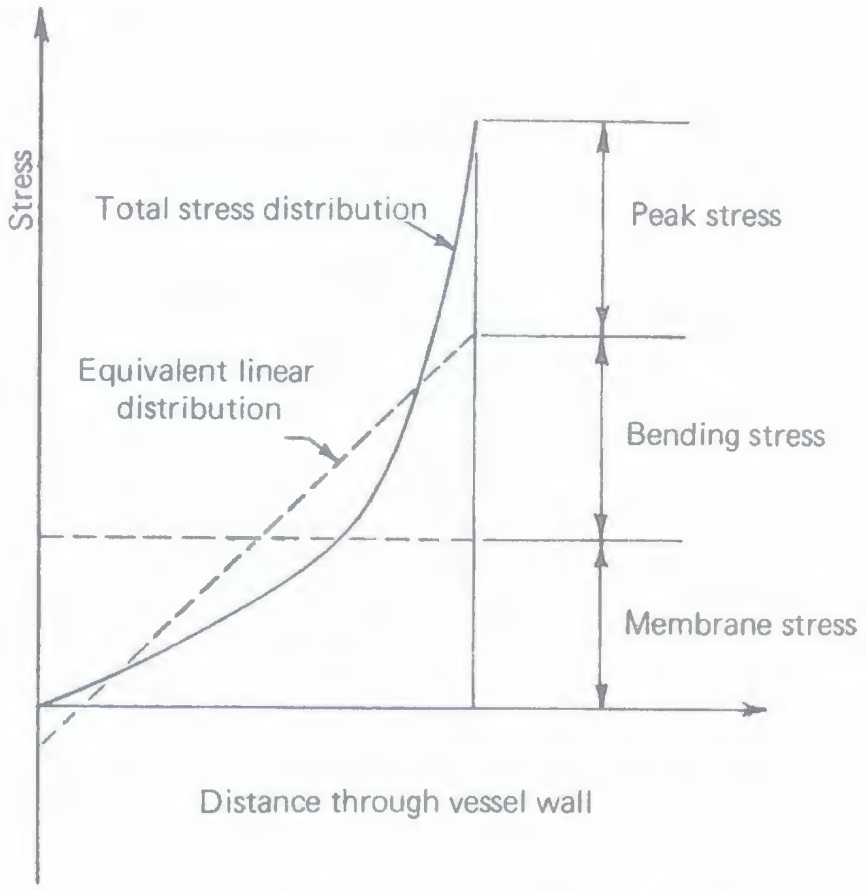


Fig. 2.6: Definition of bending and peak stress.

The value of the membrane stress is calculated by dividing the volume under the total stress distribution by the area over which the stress acts. Since the total stress distribution is described by discrete points, the volume is calculated by integration of parabolas which pass through each three consecutive stress values and extend one radian in the circumferential direction as shown in Fig. 2.6. With the above definition of membrane stress, the positive and negative volumes under the curve of total stress minus membrane are equal and the resultant moment of the stress distribution can be calculated by summing moments about any point. It is to be noted that the bending stress definition depends on the presence or absence of peak stress. If peak stresses are present, the bending stress distribution is equal to the equivalent linear stress distribution. As defined by ASME Section III, the equivalent linear stress is "the linear stress distribution which

has the same net bending moment as the actual distribution." If peak stresses are not present, the bending stress is equal to the total stress minus membrane stress. The equivalent linear distribution is demonstrated in Fig. 2.6. Also demonstrated in this figure is the calculation of peak stress. For shear stress, the membrane portion is defined and calculated the same as the normal membrane stress. A bending stress is not calculated. The peak stress is set equal to the total stress minus membrane stress. These procedures are deficient in that there is a lack of a procedure for linearizing shear stress.

2.4.3 Stress Linearization in ABAQUS

An option is available in the ABAQUS commercial code that performs a stress classification along a predefined path. The path defined by two nodes within the model as shown in Fig. 2.7.

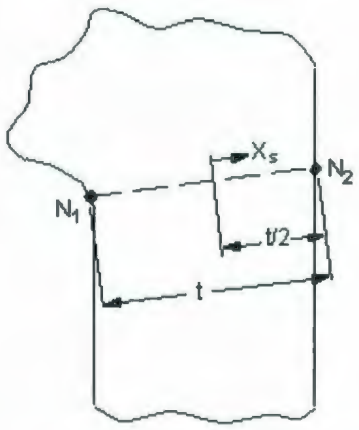


Fig. 2.7. Coordinates of Cross Section

The procedure is split into main routines, one for the non-axisymmetric (cartesian) cases and one for the axisymmetric cases. The program splits the stresses into membrane (constant), bending (linear slope along the path) stresses and peak stresses. For the cartesian case, the membrane stress is given by

$$\sigma_i^m = \frac{1}{t}\int_{-1/2}^{1/2} \sigma_i dx_s \tag{2.67}$$

where σ_i is a stress component, t is the length of the path and x_s is the coordinate along the chosen path. The magnitude of bending stress at the extreme points of the path is given by

$$\sigma_i^b = -\frac{6}{t^2}\int_{-1/2}^{1/2} \sigma_i x_s dx_s \tag{2.68}$$

It must be noted that the bending stress at the extremes will be opposite in sign. Hence, the peak stress at any point along the path will be

$$\sigma_i^p = \sigma_i - \left(\sigma_i^m + \sigma_i^b\right) \tag{2.69}$$

where σ_i is the total stress calculated in the finite element analysis.

As for the axisymmetric case, the membrane and the bending stresses are calculated in the same manner taking into account the curvature about the axis of symmetry and the local curvature within the component. Hence, in this case, the stress components do not have similar equations as it was in the non-axisymmetric case. Figure 2.8 shows the direction notations and the geometry used in calculated the linearized stresses.

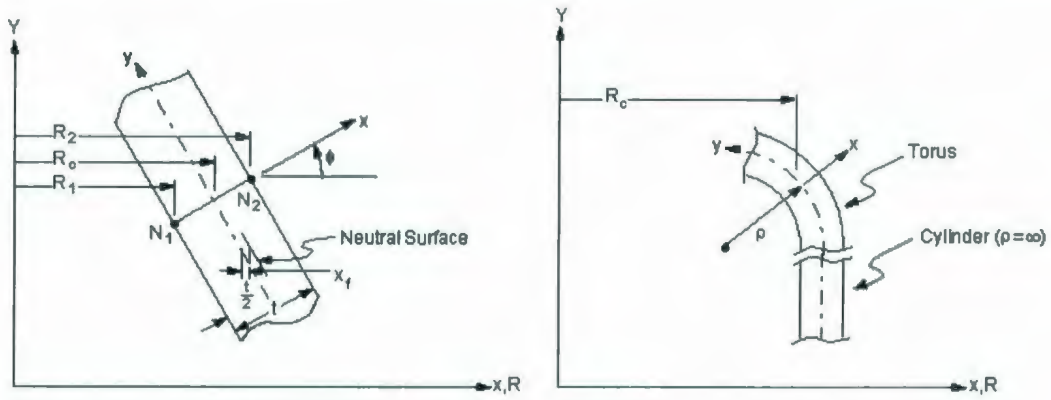


Fig. 2.8: (a) Axisymmetric Cross-Section and (b) Geometry used for Axisymmetric Evaluations

The membrane and bending stress classes in the y-direction are given by

$$\sigma_y^m = \frac{1}{R_c t}\int_{-t/2}^{t/2} \sigma_y R dx \ , \ \sigma_y^b = \frac{x - x_f}{R_c t\left(\frac{t^2}{12} - x_f^2\right)}\int_{-t/2}^{t/2} (x - x_f)\sigma_y R dx \tag{2.70}$$

where

$$x_f = \frac{t^2 \cos(\phi)}{12 R_c} \tag{2.71}$$

is the distance of the neutral bending surface from the center line. If the bending effect along the path is ignored, the membrane component is given by

$$\sigma_x^m = \frac{1}{t}\int_{-t/2}^{t/2} \sigma_x dx \tag{2.72}$$

Finally, in the hoop direction, the stresses are given by

$$\sigma_h^m = \frac{1}{t}\int_{-t/2}^{t/2} \sigma_h\left(1 + \frac{x}{\rho}\right) dx \ , \ \sigma_h^b = \frac{x - x_f}{t\left(\frac{t^2}{12} - x_f^2\right)}\int_{-t/2}^{t/2} (x - x_f)\sigma_h\left(1 + \frac{x}{\rho}\right) dx \tag{2.73}$$

The peak stress for the axisymmetric cases is calculated as in the non-axisymmetric case using equation (2.69).

2.5 Summary

There are several limit load calculation methods that are based on linear elastic finite element analysis. Most of these methods are upper bound that converges to the exact solution after several redistribution iterations. The convergence issue was addressed with development of different redistribution algorithms. The classical lower bound solution will always give a safe limit load. However, in cases with high geometrical discontinuities, it would give a highly conservative solution. In addition, the classical lower bound solution is very sensitive to the redistribution algorithm as it was observed in previous research. It is necessary to establish a lower bound solution that would overcome the problems of the classical lower bound solution. In the next chapter, the concept of the reference stress is illustrated and used to demonstrate the use of the R-Node method as one that gives a lower bound solution.

CHAPTER 3

THEORETICAL BACKGROUND

3.1 Overview

A major concern in the calculations of limit loads is the accuracy of the estimates and their reliability. It is shown that the upper bound solutions are robust and provide accurate solutions for the limit loads after several iterations of stress redistribution. However, lower bound solutions are required for design. In this chapter, the reference stress method that is used in the assessment of creep behavior of a component is illustrated. Hence, the reference stress is used as a basis to explain the R-Node concept which is used to develop a robust lower bound limit load method.

3.2 Reference Stress

The Reference Stress Method (RSM) has been proven to be successful by its extensive use in the various integrity assessments of components and structures with and without defects for both below and within the creep range of temperatures.

One of the approximate methods of reference stress determination relies on prior knowledge of limit loads for various configurations and loadings. This is shown by interpreting the results of creep analysis and tests. Initially, a mechanical component will behave elastically in response to an applied load. It is assumed to undergo creep deformation according to Norton's (power law) constitutive relation

$$\dot{\varepsilon}_c = B\sigma^n \tag{3.1}$$

The total strain rate is expressed as

$$\begin{aligned}\dot{\varepsilon} &= \dot{\varepsilon}_e + \dot{\varepsilon}_c \\ &= \frac{\dot{\sigma}}{E} + B\sigma^n\end{aligned} \tag{3.2}$$

where $\dot{\varepsilon}_e$ is the elastic, $\dot{\varepsilon}_c$ is the creep strain rates and B and n are the parameters of the creep law. Hence, the stress rate can be expressed as

$$\dot{\sigma} = E\left(\dot{\varepsilon} - B\sigma^n\right) \tag{3.3}$$

$\dot{\varepsilon}$ is derived using the deformation of the component, which is expressed as a function of the applied load and the stiffness of the component. The latter depends on the geometry and the material properties. The solution of equation (3.3) yields a function of stress versus time, having the creep law exponent n as a parameter, which will have the form shown in Fig. 3.1.

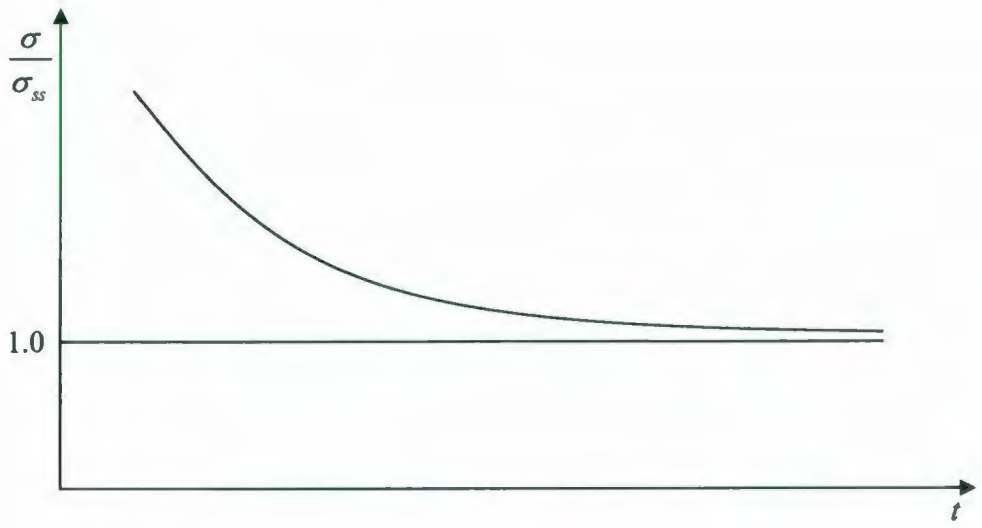


Fig. 3.1: Convergence of the stress towards the stead state.

σ_{ss} is the steady state stress field that is approached due to creep deformation and is dependent on n. The steady state stress field can be derived by setting $\dot{\sigma}$ to zero implying no changes in the stress field. Hence, equation (3.3) becomes

$$\dot{\varepsilon} - B\sigma^n = 0 \tag{3.4}$$

Having the total strain rate $\dot{\varepsilon}$ expressed in terms of the applied load, the stead state stress is found with n as the only parameter.

The total work done by an external load applied to surface S can be equated to the internal strain energy rate of the total volume V of the component and is expressed as

$$\dot{W} = \int_S P\dot{u}dS = \int_V \sigma\dot{\varepsilon}dV \tag{3.5}$$

Since the steady state stress minimizes the work done, σ_{ss} can be found by minimizing $\dot{W}$ for all stress fields that are in equilibrium with the externally applied load.

The value of n controls that amount of redistribution of stresses within the component. $n=1$ simulates the pure elastic behavior of the material. At higher values of the exponent, peak stresses vanishes as a result of creep deformation, while smaller values of the stress increase to balance the externally applied load. Figure 3.2 shows a sample steady state stress distribution across the section of beam subjected to bending for different values of n [31].

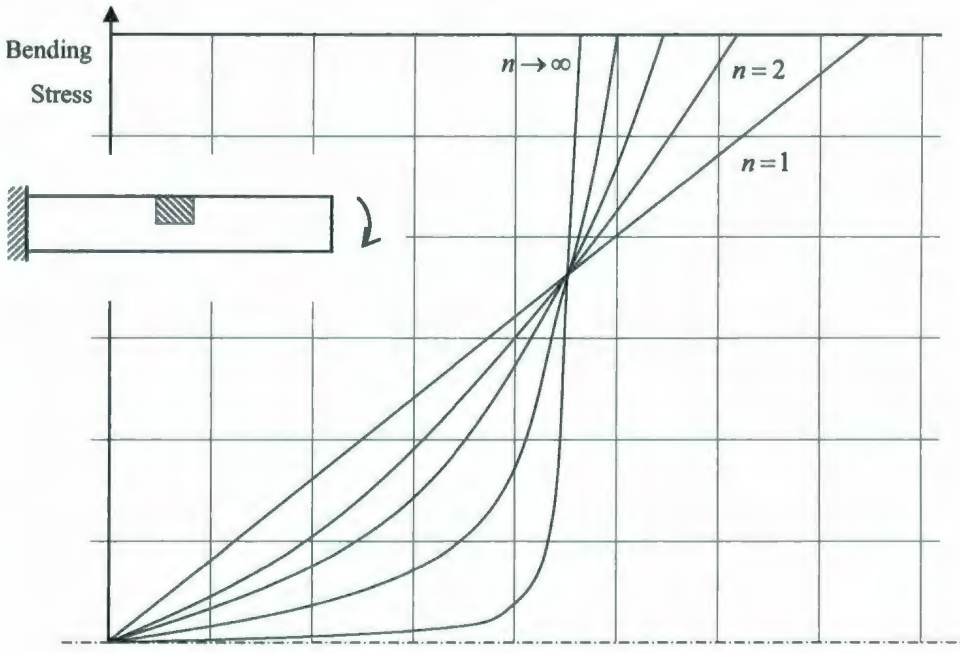


Fig. 3.2: Steady state stress distribution across a beam.

It can be noted that the distributions of the bending stress illustrated in Fig. 3.2 intersect at the vicinity of the same point at which the stress remains constant throughout the redistribution process. This point is referred to as the "skeletal point" and the corresponding stress is the "reference stress" (σ_{ref}). Using this concept, Sim [11] has observed that the steady state stress field for $n = \infty$ in a creep test has the same shape as that in the case of plastic collapse. In this case, just before plastic collapse, the stress is constant across the section as at the yield limit and in direct equilibrium with the load, which is the case with the creep deformation problem having the σ_{ref} as the constant stress across the section. This yields the relation between the reference stress and the limit load for a perfectly plastic material which is expressed as

$$\sigma_{ref} = \frac{P}{P_L}\sigma_y \tag{3.6}$$

where P is the applied load and P_L is the limit load. Analysis has shown that the steady stress is approached after a period of redistribution of

$$t_{ref} = \frac{\sigma_{ref}}{E\dot{\varepsilon}^c_{ref}} \tag{3.7}$$

The reference stress was first calculated by Soderberg [12] in 1941 in which the multiaxial creep behavior was related to the uniaxial creep behavior. It was observed that there were some points where the stress did not change with the different redistribution of the stresses based on the creep exponent. The reference stress was found to be in direct equilibrium with the applied load. Subsequently, several analytical methods were developed to calculate these reference stresses.

3.3 Sample Problems

3.3.1 Cantilever beam

A detailed analysis of the beam problem is shown to verify the reference stress method and its use to assess the creep behavior of a component. In order to derive the stress rate function using equation (3.3), the strain rate is first evaluated as a function of the applied load. Assuming that the plane sections of the beam remain plane, the total strain rate can be expressed as

$$\dot{\varepsilon} = \dot{\kappa} z \tag{3.8}$$

where κ is the curvature of the beam which is time dependant. Webster and Ainsworth [13] showed that, from the equilibrium of the stress with the externally applied load, it can be deduced that

$$\dot{\kappa} = \frac{3A}{d^3} \int_0^d \sigma^n z dz \tag{3.9}$$

Hence, substituting in equation (3.8), the stress rate is expressed as

$$\dot{\sigma} = \left(\frac{3EA}{d^3} \right) z \int_0^d \sigma^n z dz - EA\sigma^n \tag{3.10}$$

The steady state stress field may be obtained directly from equation (3.10). Since the stress is constant in the steady state phase, the stress rate in equation (3.10) will be zero, which eliminates that elastic component of the strain rate in equation (3.2). Solving for the steady state stress gives

$$\sigma_{ss} = \left(\frac{M}{Bd^2} \right) \left(1 + \frac{1}{2n} \right) \left(\frac{z}{d} \right)^{1/n} \tag{3.11}$$

It is evident that the stress distribution is only influenced by the exponent n. At $n = \infty$, σ_{ss} becomes σ_{ref}, which yields

$$\sigma_{ref} = \frac{M}{Bd^2} \tag{3.12}$$

This indicates that, knowing the reference stress, the steady state stress distribution can be assessed without a complete creep analysis. Using equation (3.11), the creep strain rate at the steady state stage is expressed as

$$\dot{\varepsilon}^c = A\sigma_{ref}^n \left(1+\frac{1}{2n}\right)^n \left(\frac{z}{d}\right) \tag{3.13}$$

Or, in terms of the creep strain rate at the skeletal point for $n \to \infty$, expressed as

$$\dot{\varepsilon}^c = \left(\frac{3}{2}\right)\dot{\varepsilon}_{ref}^c \left(\frac{z}{d}\right) \tag{3.14}$$

Considering the creep law, the work rate in the steady state can be expressed using equation (3.5) as

$$\begin{aligned} \dot{W}_{min} &= B\int_{-d}^{d} \sigma_{ss}\dot{\varepsilon}dz = 2B\int_{0}^{d} \sigma_{ss}\dot{\varepsilon}dz \\ &= 2B\int_{0}^{d} A\sigma_{ss}^{n+1}dz \end{aligned} \tag{3.15}$$

Using equation (3.11) of the steady state stress, the minimum work rate is expressed as

$$\dot{W}_{ref} = 2BdA\sigma_{ref}^{n+1}\left[\frac{1}{2}\left(1+\frac{1}{2n}\right)^n\right] \tag{3.16}$$

Another method to find the minimum work rate equivalent to the steady state is to consider the stress field of σ_{ref} corresponding to $n \to \infty$. In addition, a uniform distribution of the creep strain rate across the beam section of $\dot{\varepsilon}_{ref}$. Substituting in equation (3.3) yields

$$\begin{aligned}\dot{W}_{sp} &= B\int_{-d}^{d}\sigma_{ref}\dot{\varepsilon}_{ref}dz \\ &= 2B\int_{0}^{d}A\sigma_{ref}^{n+1}dz \\ &= 2BdA\sigma^{n+1}\end{aligned} \tag{3.17}$$

Comparison of $\dot{W}_{ref}$ and $\dot{W}_{sp}$ with $\dot{W}_{min}$ shows that equation (3.16) is a conservative solution for the work rate that is calculated using the reference stress.

Knowing the reference stress and the equivalent creep strain rate, the total strain after time t during the steady state at the skeletal point can be expressed as

$$\begin{aligned}\varepsilon &= \frac{\sigma_{ref}}{E} + \int_{0}^{t}\dot{\varepsilon}_{ref}^{c}dt \\ &= \frac{\sigma_{ref}}{E} + \varepsilon_{ref}^{c}\end{aligned} \tag{3.18}$$

Having the plane section remain plane during deformation, the total strain is assumed to be varying linearly across the beam, with the creep strain at the skeletal point as ε_{ref}^{c}. The maximum total strain may be expressed as

$$\varepsilon_{max} = \left(\frac{3}{2}\right)\frac{\sigma_{ref}}{E} + \left(\frac{3}{2}\right)\varepsilon_{ref}^{c} \tag{3.19}$$

For $n \to \infty$ the elastic stress at the outer fibers will be σ_{ref} and the elastic strain will be σ_{ref}/E. This yields the maximum creep strain to be

$$\varepsilon_{max} = \left(\frac{1}{2}\right)\frac{\sigma_{ref}}{E} + \left(\frac{3}{2}\right)\varepsilon_{ref}^{c} \tag{3.20}$$

The difference between equation (3.14) and (3.20) shows that the redistribution period leads to an extra creep strain of $\sigma_{ref}/2E$.

3.3.2 Thick cylinder

The thick cylinder problem illustrates the use of the reference stress methods in axisymmetric bodies. Figure 3.3 shows a schematic of the problem considered. The cylinder is subjected to internal pressure and p. The internal and external radii are r_i and r_o, respectively.

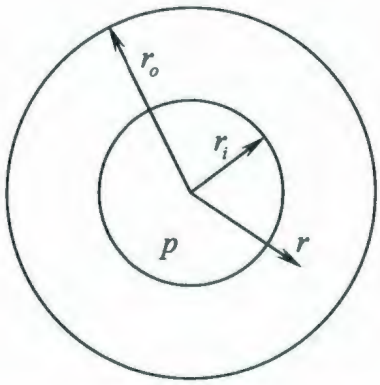


Fig. 3.3: Thick walled cylinder subjected to internal pressure

The equilibrium of small element yields

$$\frac{\partial \sigma_r}{\partial r} = \frac{\sigma_\theta - \sigma_r}{r} \tag{3.21}$$

The axial stress is balanced with internal pressure applied ends of the cylinder. Hence,

$$2\pi \int_{r_i}^{r_o} \sigma_a r dr = \pi r_i^2 p \tag{3.22}$$

For small deformation, the hoop and radial strain rates are expressed in terms of deformation as

$$\dot{\varepsilon}_\theta^c = \frac{\dot{w}}{r} \tag{3.23}$$

$$\dot{\varepsilon}_r^c = \frac{\partial \dot{w}}{\partial r} \tag{3.24}$$

where w is the radial displacement. In addition, the condition for constant volume yields

$$\dot{\varepsilon}_{\theta}^{c} = -\dot{\varepsilon}_{r}^{c} \tag{3.25}$$

Substituting equations (3.23) and (3.24) into equation (3.25) gives the solution for the deformation rate as

$$\dot{w} = \frac{C}{r} \tag{3.26}$$

where C is a constant that does not depend on the radius but is a function of time. Webster and Ainsworth [13] have shown that the time strain hardening for the cylinder is expressed as

$$\dot{\varepsilon}_{\theta}^{c} = -\dot{\varepsilon}_{r}^{c} = \frac{\sqrt{3}}{2} A \bar{\sigma}^{n} F(t) \tag{3.27}$$

where $\bar{\sigma}$ is the von Mises stress which is expressed as

$$\bar{\sigma} = \frac{\sqrt{3}}{2}(\sigma_{\theta} - \sigma_{r}) \tag{3.28}$$

The elastic component is ignored to find the steady state stress as shown in the cantilever beam. Substituting equations (3.28) and (3.26) in equation (3.27), Webster and Ainsworth [13] gives

$$\sigma_{\theta} - \sigma_{r} = C_{1} r^{-2/n} \tag{3.29}$$

where $\sigma_{\theta} - \sigma_{r} = C_{1} r^{-2/n}$ is a constant which is readily obtained by integrating the equilibrium equation (3.21) and applying the boundary conditions for the radial stress given as

$$\begin{aligned} \sigma_{r}\big|_{r=r_{i}} &= -p \\ \sigma_{r}\big|_{r=r_{o}} &= 0 \end{aligned} \tag{3.30}$$

This yields the radial stress σ_{r} and the hoop stress σ_{r} expressed as

$$\sigma_r = -p\frac{\left[(r_i/r)^{2/n} - (r_i/r_o)^{2/n}\right]}{\left[1-(r_i/r_o)^{2/n}\right]} \tag{3.31}$$

$$\sigma_\theta = p\frac{\left[(r_i/r_o)^{2/n} - (1-2/n)(r_i/r)^{2/n}\right]}{\left[1-(r_i/r_o)^{2/n}\right]} \tag{3.32}$$

It can be deduced that the radial stress distribution is not strongly influenced by the value of n because of the imposed boundary conditions, while the hoop stress distribution changes from negative slope for $n<2$ to positive slope for $n>2$ with a constant value at $n=2$. Figure 3.4 shows the radial distribution of the hoop stress calculated using equation (3.32) for different values of n. As it was the case in the beam section shown in Fig. 3.2, the stress distributions intersect at the skeletal point defining the reference stress.

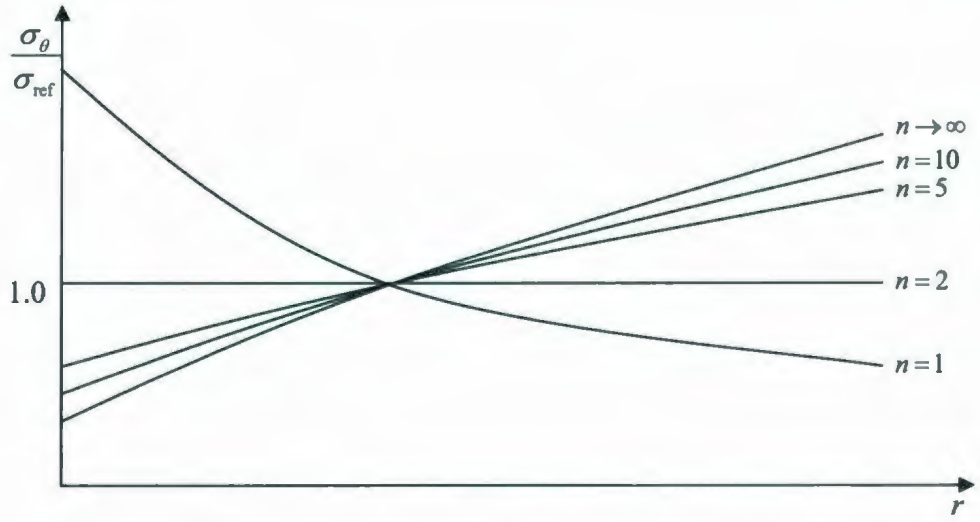


Fig. 3.4: Distribution of the hoop stress through the thickness of the thick cylinder for different values of the creep exponent.

The equivalent von Mises stress derived by substituting equations (3.31) and (3.32) in equation (3.28) which yields

$$\bar{\sigma} = \frac{\sqrt{3}}{2} p \left[\frac{\left[(r_i/r_o)^{2/n} - (1-2/n)(r_i/r)^{2/n} \right]}{\left[1-(r_i/r_o)^{2/n} \right]} + \frac{\left[(r_i/r)^{2/n} - (r_i/r_o)^{2/n} \right]}{\left[1-(r_i/r_o)^{2/n} \right]} \right]$$
$$= \frac{\sqrt{3} p}{n} \frac{\left[(r_i/r)^{2/n} \right]}{\left[1-(r_i/r_o)^{2/n} \right]} \tag{3.33}$$

The distribution of the equivalent stress through the thickness is similar to that of the hoop stress, except that the stress distribution tends to constant for $n \to \infty$ as shown in Fig. 3.5.

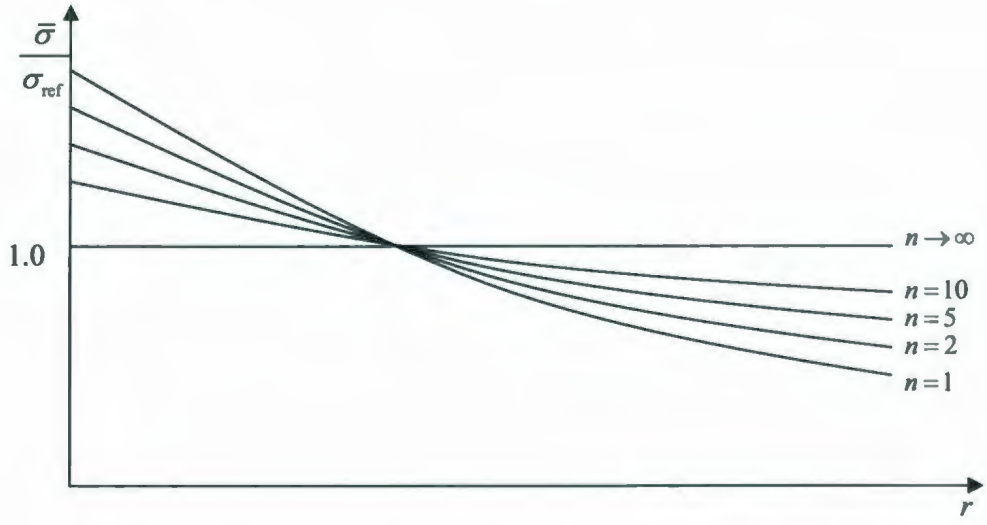


Fig. 3.5: Distribution of the normalized equivalent stress through the thickness of the thick cylinder for different values of the creep exponent.

The creep strain rate is then derived using equations (3.27) and (3.33) as

$$\dot{\varepsilon}_\theta^c = -\dot{\varepsilon}_r^c = \frac{\sqrt{3}}{2} A \left(\frac{\sqrt{3} p}{n} \frac{\left[(r_i/r)^{2/n} \right]}{\left[1-(r_i/r_o)^{2/n} \right]} \right)^n F(t) \tag{3.34}$$

The limit load of the cylinder is derived for a perfectly plastic material as

$$P_L = \left(\frac{2\sigma_y}{\sqrt{3}}\right)\ln\left(\frac{r_o}{r_i}\right) \tag{3.35}$$

Using equation (3.6), the reference stress is derived as

$$\sigma_{ref} = \frac{\sqrt{3}}{2\ln(r_o/r_i)}p \tag{3.36}$$

The location of the skeletal point is found by setting the equivalent stress in equation (3.33) for $n=1$ equal to the reference stress in equation (3.36). This yields the radius of the skeletal point as

$$r_{sp} = r_i\sqrt{\frac{2\ln(r_o/r_i)}{\left[1-(r_i/r_o)^2\right]}} \tag{3.37}$$

Using the reference stress, the total strain can be expressed in form shown in equation (3.18) in the beam problem. The strain is assumed to vary inversely to the square of radius in the cylinder wall. Hence, using the accumulated reference creep strain and its location, the maximum strain at the inner radius is expressed as

$$\varepsilon_{max} = \left(\frac{r_{sp}}{r_i}\right)^2\left(\frac{\sigma_{ref}}{E}+\varepsilon^c_{ref}\right) \tag{3.38}$$

which consists of both the elastic component (σ_{ref}/E) and the creep component. Again, this indicates excessive deformation due to stress redistribution

3.4 The Redistribution Node Method

Understanding the significance of the reference stress, it can be observed that the analytical methods may not be possible in complex problems. Besides, using Sim's approximation given by equation (3.6) requires a prior knowledge of the limit load which

may not be readily available for all structures. Seshadri and Fernando [4] showed that the insensitivity of the reference stress to the creep parameters has more to do with its load-control nature. The deformation-control stresses are redistributed throughout the component with the spread of inelastic action.

Seshadri and Fernando [4] have used the reference stress concept to develop the R-Node method utilizing the GLOSS analysis [4] and the EMAP [6] to redistribute the stresses. The R-Node stresses are load-controlled stress and, therefore, are directly proportional to the applied load. Hence, any two stress distributions satisfying equilibrium with the externally applied loads will intersect at the R-Nodes locations. This is because redistribution of stresses reduces deformation-control stresses, at locations such as geometrical discontinuities, and increases other zones to preserve the equilibrium of the applied loads.

Therefore, the R-Node analysis method compares the von Mises equivalent stress field of the initial elastic analysis to the redistributed stress and the points where the equivalent stress remains constant after the redistribution are set to be the redistribution nodes (R-Nodes). The elastic modulus adjustment is performed using the equation

$$E_{k,i+1} = \left(\frac{\sigma_a}{\sigma_{ek,i}}\right)^q E_{k,i} \tag{3.39}$$

Where σ_a is an arbitrary stress selected to be within the range of stresses in the component. This adjustment is made to the whole element, $\sigma_{ek,i}$ is the equivalent von Mises stress at the centroid of element k at iteration i and $E_{k,i}$ is the elastic modulus of element k at iteration i. The exponent q controls the amount of redistribution within the element. A detailed development of these formal basis for the elastic modulus adjustment and related procedures has been provided by Ponter and co-workers [14–16]. The generalized approach has similarities to the elastic modulus adjustment procedures and can be better described as "linear matching methods" where a sequence of linear solutions is matched to the nonlinear problem. The elastic modulus adjustment methods rely on the convergence of the specific moduli adjustment procedure. This problem was

also addressed by Ponter et al. [15], who showed that the convergence of suitable matching methods is theoretically guaranteed for practically important yield functions.

Another analysis is performed using the modified elastic modulus. This step would be repeated several times to go closer to the stress distribution of the limit solution due to the applied load. In every step of the EMAP, the stress distribution is compared with that of the initial elastic analysis. Hence, in each stage, the R-Node stress will be the intersection of the stress distribution with the initial one. Using the load-controlled nature, the R-Node stresses are expressed as

$$\sigma_{R-Node} = \alpha P \tag{3.40}$$

where α is a proportionality constant that depends on the geometrical properties and the nature of the applied load. Since the stress redistribution using the elastic modulus adjustment generates a stress field almost equivalent to that at the state of collapse, the R-Node stress in equilibrium with an arbitrary load is equivalent to the yield stress in equilibrium with the limit load. Hence,

$$\sigma_y = \alpha P_L \tag{3.41}$$

Eliminating the proportionality constant, the limit load is expressed as

$$P_L = m_{R-Node} P, \quad m_{R-Node} = \frac{\sigma_y}{\sigma_{R-Node}} \tag{3.42}$$

Mangalaramanan and Seshadri [2] have used the R-Node analysis to find the limit load of different components. Seshadri [17] has used the R-Node analysis to generate the limit curve of a cantilever beam subjected to both an axial force and a bending moment, which was done by estimating the limit value of both loads at specific ratios.

3.5 Summary

The Reference Stress Method (RSM) is a useful tool in the evaluation of the creep deformation of components. It is shown that the RSM is a simple and an effective method

for finding the steady state stress and the total creep deformation without the complex creep analysis. It gave results for the work done and energy dissipation within 10% error. This gave an evaluation of the accuracy of determining the steady state stress and strain.

The skeletal points are those at which the stress remains constant after redistribution independent on the constitutive parameters. Hence, the stress at the skeletal point for small values of the creep law exponent is equal to that for high values (near infinity) of the exponent. Since, the stress field at high values of the creep exponent represents the reference stress field, the stress at skeletal point, which is constant, represent the reference stress for any value of the constitutive parameters.

The reference stress method initiated the R-Node concept for limit load calculation using finite element analysis in which redistribution of the stress is performed using linear elastic analysis. This is achieved by modifying the modulus of elasticity at the integration points in the component simulating the softening of the material due to creep deformation. Hence, comparing two simple elastic analyses, the R-Nodes are located and the values of the R-Node stress are found which helps in calculating the limit load of the component. In the next chapter, the algorithm of the R-Node method is explained in details showing its capability of being integrated in commercial finite element analysis programs.

CHAPTER 4

ITERATIVE R-NODE ANALYSIS METHOD

4.1 Overview

The R-Node stresses are those that are in equilibrium with the externally applied load and, thus, are load controlled stresses as described by Seshadri and Marriott [18], and are linearly proportional to the externally applied load. A single R-Node stress through the thickness of a pressure component indicates a dominant membrane stress as the case of a thick cylinder shown in the previous chapter, while a pair of R-Node stresses indicates a dominant bending stress.

A statically determinate structure will collapse with the formation of a single plastic hinge, while an indeterminate structure requires higher number of plastic hinges, depending on its degree of indeterminacy, to collapse. The distribution of the R-Node stresses has several peaks throughout the domain. The average of these peaks remains almost constant with the redistribution of the stresses since the R-Node stresses are load-controlled. The peaks of the R-Node stresses are at the hinge locations or plasticity initiation location defining the collapse state.

Seshadri and Fernando [4] demonstrated the collapse of an indeterminate beam to illustrate the concept behind the R-Node peaks. Figure 4.1 (a) shows the indeterminate beam with the expected plastic hinge locations. The collapse in the beam is approached first by the formation of a plastic hinge at point A, followed by another plastic hinge at point B. Figure 4.1 (b) shows the distribution of the R-Node stresses along the beam. The collapse of the beam can be represented by the two bar model shown in Fig. 4.1 (c). Hence, the load that would cause the first plastic hinge in the indeterminate beam will be that which will cause the short bar in the two-bar model to collapse, and the load to cause

the second plastic hinge will be that which will cause both bars to collapse. The stress in the first bar will be the first R-Node peak stress σ_{n1} at point A, and the that of the second bar is the other R-Node peak stress σ_{n2}. The cross section sectional area of the bars are selected to satisfy the equilibrium equation

$$\sigma_{n1}A_1 + \sigma_{n2}A_2 = P \tag{4.1}$$

At the state of collapse, the equilibrium requirement is expressed as

$$\bar{\sigma}_n\left(A_1 + A_2\right) = P \tag{4.2}$$

where $\bar{\sigma}_n$ is a combined R-Node effective stress. It can be deduced from equations (4.1) and (4.2) that

$$\bar{\sigma}_n = \mu_1\sigma_{n1} + \mu_2\sigma_{n2} \tag{4.3}$$

where $\mu_1 = A_1/\left(A_1 + A_2\right)$ and $\mu_2 = A_2/\left(A_1 + A_2\right)$. Using the condition that the deflection of the two bars is equal, the R-Node stresses in the bars are expressed as

$$\begin{aligned}\sigma_{n1} &= \frac{E_1/L_1}{E_1A_1/L_1 + E_2A_2/L_2}P \\ \sigma_{n2} &= \frac{E_2/\mathrm{L}_2}{E_1A_1/L_1 + E_2A_2/L_2}P\end{aligned} \tag{4.4}$$

Substituting equation (4.4) into (4.3), the limit load can be expressed as

$$P_{\mathrm{L}} = \left[\frac{E_1A_1/\mathrm{L}_1 + E_2A_2/\mathrm{L}_2}{\mu_1E_1/\mathrm{L}_1 + \mu_2E_2/\mathrm{L}_2}\right]\sigma_{\mathrm{y}} \tag{4.5}$$

Seshadri and Fernando [4] have studied two cases for equation (4.5) to derive the R-Node effective stress $\bar{\sigma}_n$. In case 1, it is assumed that $E_1/\mathrm{L}_1 = E_2/\mathrm{L}_2$. This yields $P_{\mathrm{L}} = \left[A_1 + A_2\right]\sigma_{\mathrm{y}}$ from equation (4.5). Hence, from equations (4.4), the R-Node stresses will be equal and expressed as

$$\sigma_{n1} = \sigma_{n2} = \frac{1}{A_1 + A_2}P \tag{4.5a}$$

This can be considered to be a trivial solution since there is no unique R-Node stress for every bar.

In case 2, it is assumed that $A_1 = A_2 = A$. The R-Node stresses from equations (4.4) can be expressed as

$$\sigma_{n1} = \frac{1}{A}\left[\frac{E_1/L_1}{E_1/L_1 + E_2/L_2}\right]P$$
$$\sigma_{n2} = \frac{1}{A}\left[\frac{E_2/L_2}{E_1/L_1 + E_2/L_2}\right]P \qquad (4.5b)$$

Substituting equation (4.5b) in equation (4.5), the R-Node effective stress can be expressed as the arithmetic average of the R-Node stresses given by

$$\bar{\sigma}_n = \frac{\sigma_{n1} + \sigma_{n2}}{2} \qquad (4.6)$$

This illustrates the fact that, since at collapse the stress at the plastic hinges is the yield stress for an elastic-perfectly plastic material, therefore the redistribution analysis makes the peak R-Node stress approach the same average value as shown in Fig. 4.1 (b). Hence, for a general problem, the R-Node effective stress can be expressed as

$$\bar{\sigma}_n = \frac{\sum_{i=1}^{N} \sigma_{ni}}{N} \qquad (4.7)$$

where σ_{ni} is the R-Node stress at peak i and N is the number of peaks. Hence, the limit load can be expressed as

$$P_{LL} = \frac{\sigma_y}{\bar{\sigma}_n} P \qquad (4.8)$$

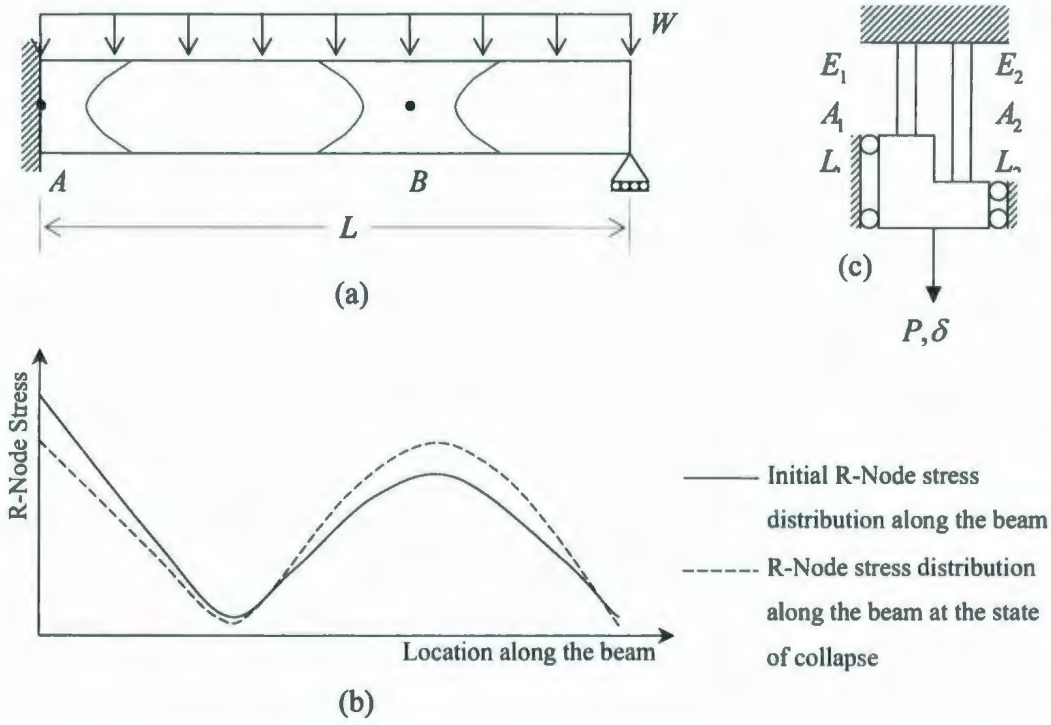
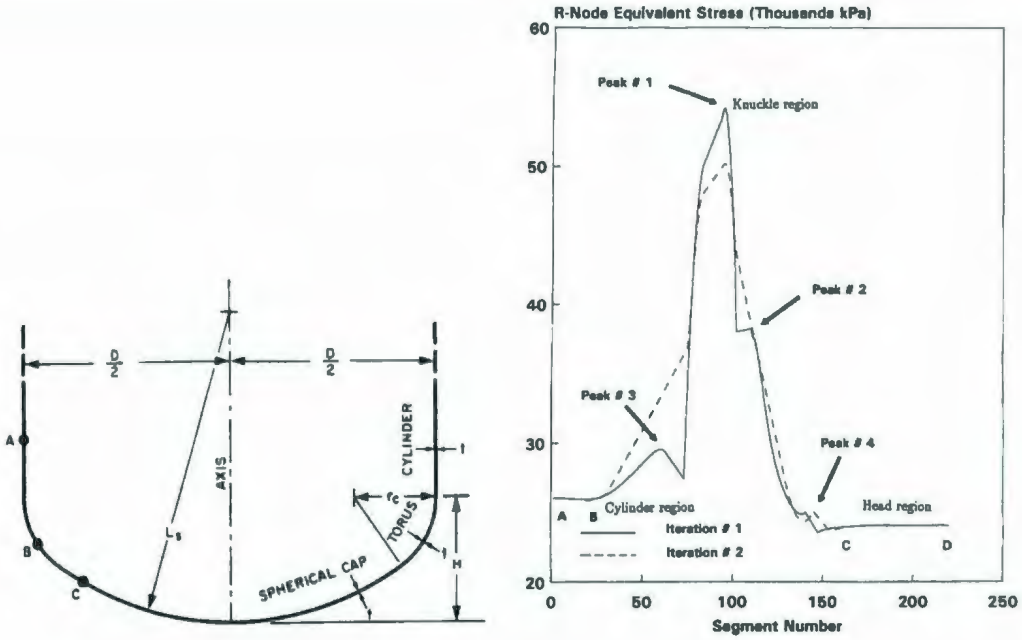


Fig. 4.1: (a) Schematic diagram of the indeterminate beam. (b) Expected distribution of the R-Node stress along the beam for the initial and an intermediate iteration. (c) Two-bar model representing the collapse of the indeterminate beam.

The R-Node stress distribution may not be always easy to interpret as it is the case for the indeterminate beam shown in Fig. 4.1(b). The approximate extrapolation of the stresses within an element may cause the R-Node stress distribution to have some virtual peaks at several locations in the component. Seshadri [19] illustrates the distribution of the R-Node stress in a torispherical head shown in Fig. 4.2. The comparison of the first and second analysis showed to have 4 peaks for the R-Node stresses. However, further redistribution of the stresses eliminated the virtual R-Nodes and real ones remained. This shows the transient nature of the virtual R-Nodes.



(a) (b)

Fig. 4.2: (a) Torispherical head configuration. (b) R-Node stress distribution for the first and second iteration [19].

The iterative R-Node limit load analysis method developed in the present work uses the maximum R-Node peak instead of the average of all the peaks. The use of the maximum R-Node has two advantages:

1 – Guarantees having a real R-Node stress peak avoiding any inaccuracy due to virtual R-Node stress peaks.

2 – From equation (4.7), the average R-Node stress has the feature

$$\bar{\sigma}_n \leq \sigma_{n1} \tag{4.8a}$$

Hence, the limit load based on the average R-Node stress is always greater than or equal that based on the maximum R-Node peak. This guarantees a lower bound solution for the limit load.

4.2 Iterative R-Node Analysis Algorithm

The R-Node stress values and locations are computed by comparing the stress distribution of two elastic analyses. In the first analysis, the stresses are computed using the actual modulus of elasticity of the material used. In the second analysis, the elastic modulus is adjusted using any of the methods explained earlier in chapter 2 and a new stress distribution is calculated. In this research, a code is written for the iterative R-Node method to be used in ABAQUS. In the developed algorithm, the elastic modulus is adjusted at the integration points as opposed to the centroid used earlier in the R-Node analysis method. This gives a better definition of the stress variation within the element and thus improves the results for coarse mesh. The iterative R-Node method is used with different types of elements to verify its applicability.

In implementing the R-Node analysis in ABAQUS, a user-defined FORTRAN subroutine UMAT for the material is created. Figure 4.3 shows a flow-chart describing the subroutine.

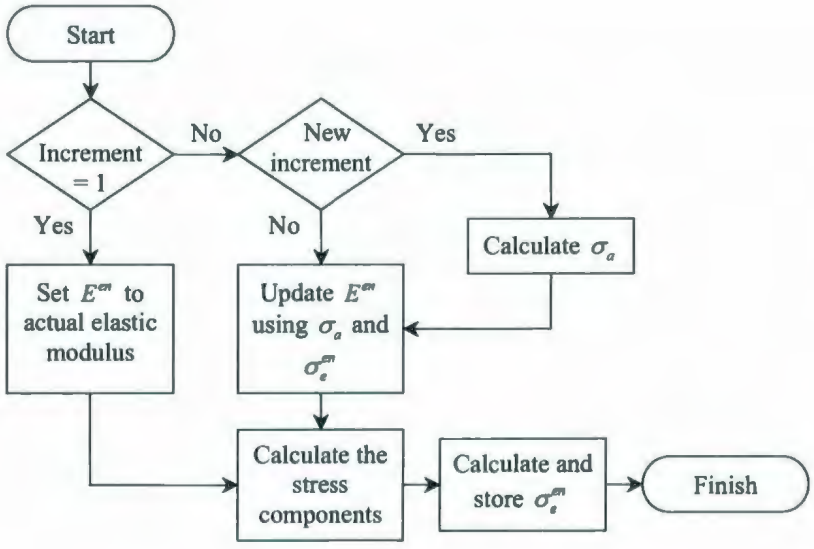


Fig. 4.3: Flow-chart of UMAT subroutine.

The UMAT subroutine is called for every Gaussian integration point. The total and incremental strains are passed as parameters to the subroutine, the stress components are calculated according to Hooke's law and the equivalent stress is calculated according to von Mises criteria. At the beginning of the analysis, during the first increment, there is no change in the elastic modulus. The calculated equivalent stress and the initial value of the elastic modulus are stored for every Gaussian integration point. If it is not the first increment, the program checks if it is the first call for the subroutine in that increment. If so, the subroutine calculates the arbitrary stress according to equation (2.3) using the stored stress which would be that of the previous increment. In any case, the subroutine proceeds to calculate the new elastic modulus for the associated material calculation point. This is done by the equation

$$E_{n,i+1} = \left(\frac{\sigma_a}{\sigma_{en,i}} \right)^q E_{n,i} \tag{4.9}$$

where $\sigma_{en,i}$ is the equivalent stress at the integration point n, $E_{n,i}$ is the elastic modulus stored during the previous increment and σ_a is the arbitrary stress calculated at the first call of the subroutine during the current increment. q is a redistribution constraint factor which controls the amount of redistribution for every iteration. A value of zero suppresses the redistribution and a value of 1.0 adjusts the elastic modulus according to the actual value of the stress. For the iterative R-Node analysis method, the value of q is assumed to be 1.0.

Once the analysis is performed, a Python script is called to read the results and find the R-Node locations and stress value. The script reads the values of the extrapolated stress at the nodes for every element. It compares the stresses calculated in the first iteration with that in the other iterations. Figure 4.4 shows a plane quadrilateral element illustrating the distribution of the stress within the element for the first and second iterations. The comparison is done within the adjacent nodes in the elements. Hence, for the element shown in Fig. 4.4, an R-Node is found between nodes 1 and 2, and between nodes 3 and 4.

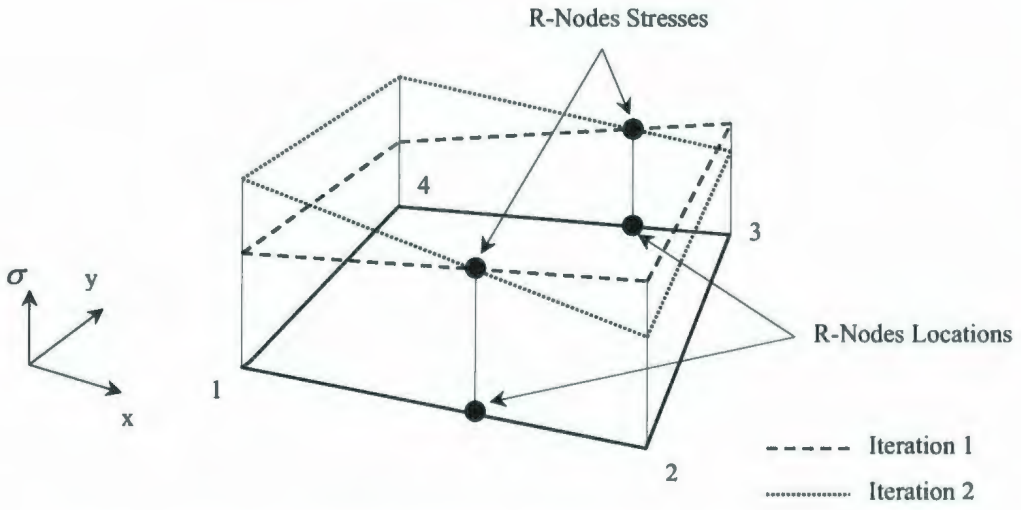


Fig. 4.4: Schematic diagram showing the procedure for calculating the R-Node stress and location in a plane element using the stress distribution within the element.

The procedure is also used to find the R-Node stresses and locations in a solid brick element by handling every pair of adjacent nodes separately.

Mangalaramanan and Seshadri [2] have showed that the factor m^0 can be useful in assessing whether or not limit stress distribution is being approached during successive elastic iterations. m^0 should monotonically decrease and converge for a stress redistribution that is approaching a limit state. Hence,

$$m_i^0 - m_{i+1}^0 > 0 \tag{4.10}$$

Should this not occur, i.e., if there is an increase in the value of m^0 as compared to its value during the previous iteration, then the theorem of nesting surfaces ([2]) would be violated implying that the stress distributions are not on a redistribution path leading to limit state. In this case, based on several trials, a slower redistribution rate would improve achieve condition (4.10). A value of q in equation (4.9) less than 1.0 will slow down the redistribution and, hence, the convergence rate. The response of a redistribution analysis

to the change in q varies according to the geometry and the applied load. Hence, the selection of the value of q is based on trials and observation.

4.3 Applications

The simple problem of a thick cylinder modeled using plane strain elements is also analyzed using the iterative R-Node method. In addition, to illustrate the effect of indeterminacy of the structure, a detailed analysis of the indeterminate beam is illustrated. Finally, an oblique nozzle studied experimentally by Sang et al [20] is analyzed using the R-Node method and the results are compared to the experimental values. The oblique nozzle is modeled using shell elements.

4.3.1 Thick Plane-Strain Cylinder

The thick cylinder is modeled using 4-noded plain strain elements. The inside diameter of the cylinder is assumed to be 6 in. and the outside diameter is 18 in. The elastic modulus is 30×10^6 psi and Poisson's ratio is 0.49. The value of the Poisson's ratio is selected near 0.5 to account for the incompressibility of the material due to plastic expected plastic deformation. An arbitrary pressure of 100 psi is applied. A quarter of the cylinder is modeled as shown in Fig. 4.5. Figure 4.6 shows the redistribution of the stress through the wall thickness. It can be seen that the stresses approach a single value which is the expected distribution at collapse. Figure 4.7 shows the convergence of the limit load multipliers indicating the exact solution, upper bound, lower bound and the iterative R-Node analysis. It can be observed that the R-Node solution converged efficiently to the solution derived using equation (3.35), and faster than the classical lower bound solution.

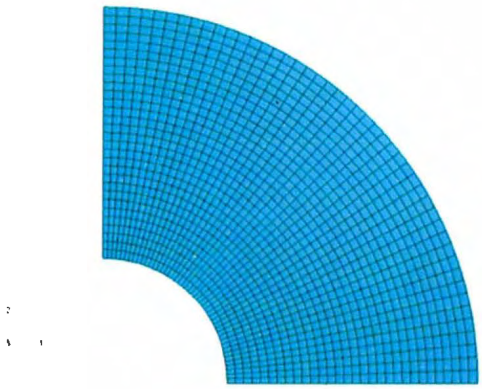


Fig. 4.5: Meshing of the plane-strain cylinder with 6 inches inner diameter and 18 inches outer diameter.

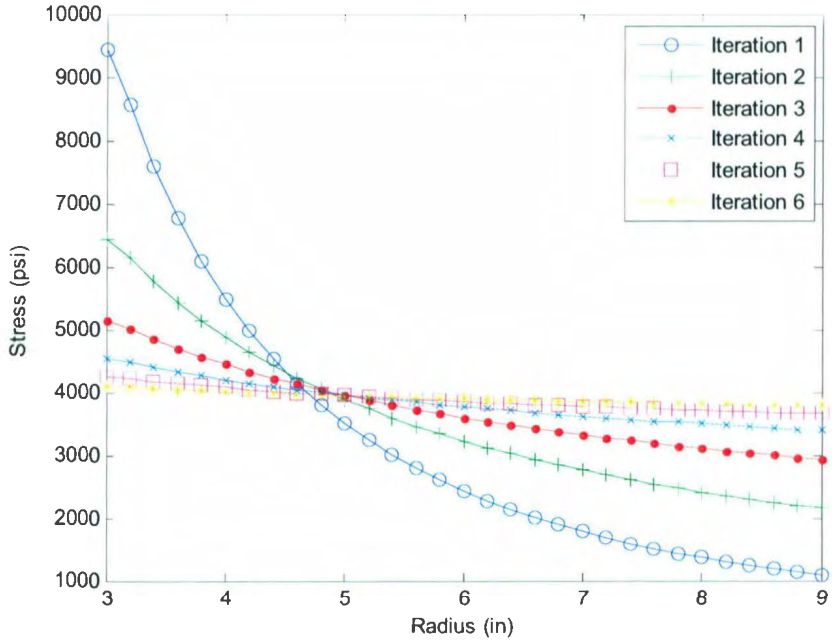


Fig. 4.6: Stress redistribution through the cylinder wall using elastic compensation methods.

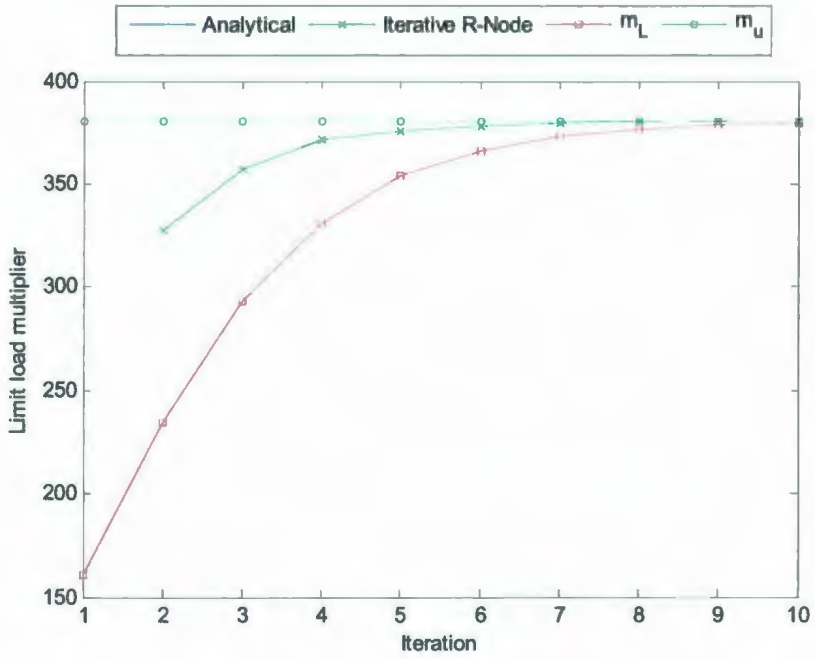


Fig. 4.7: Comparison of the convergence of the limit load multipliers calculated for the thick cylinder.

4.3.2 Indeterminate Beam

The beam modeled in this example is solved both analytically and using the R-Node method to find the collapse load. The length of the beam is 20 inches with unit height and width. The beam is subjected to a distributed load as shown in Fig. 4.8. For the finite element mesh, the plane stress elements are used.

For the R-Node analysis, the user-defined subroutine UMAT developed earlier is used with the modification of the stress-strain relation to account for the plane stress formulation to define the behavior of a user material.

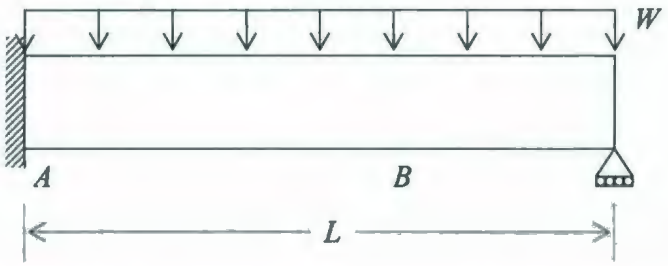


Fig. 4.8: Schematic diagram of the indeterminate beam with length of 20 inches and unit height and depth subjected to distribution load.

The problem is solved analytically by Mendelson [21]. The bending moment due to the distributed load is expressed as

$$M(x) = -\frac{wx^2}{2} + \frac{5wL}{8}x - \frac{wL^2}{8} \tag{4.10a}$$

The equation for the load to cause the initial yield is given by

$$W_y = \frac{8M_y}{L^2} = \frac{4\sigma_y}{3L^2} \tag{4.11}$$

for a unit height and depth. In addition, the load to form the first hinge is given by

$$W_{h1} = \frac{8M_0}{L^2} = \frac{12M_y}{L^2} = \frac{2\sigma_y}{L^2} \tag{4.12}$$

and the load to cause collapse is given by

$$W_{LL} = \frac{11.65M_0}{L^2} = \frac{17.48M_y}{L^2} = \frac{2.91\sigma_y}{L^2} \tag{4.13}$$

where M_y is the moment that causes initial yielding and M_0 is the moment that causes a full plastic hinge in the beam.

Figure 4.9 shows the von Mises stress distribution through the beam for the first and last iteration at point A. It is observed that the stress distribution at the last iteration tends to have a uniform value corresponding to the hinge formation at the state of collapse. The point of intersection with the initial stress distribution indicates the R-Node location and stress value at the specified section. Figure 4.10 shows the distribution of the R-Node stress along the beam. This is done by checking for the R-Node stress at every

section along the beam. It is observed that there are two peaks for the R-Node stress. In the first iteration, the peaks are not equal indicating a sequence of hinge formation in a collapse mechanism. Hence, a hinge will be formed first at point A, and then another hinge is formed at point B.

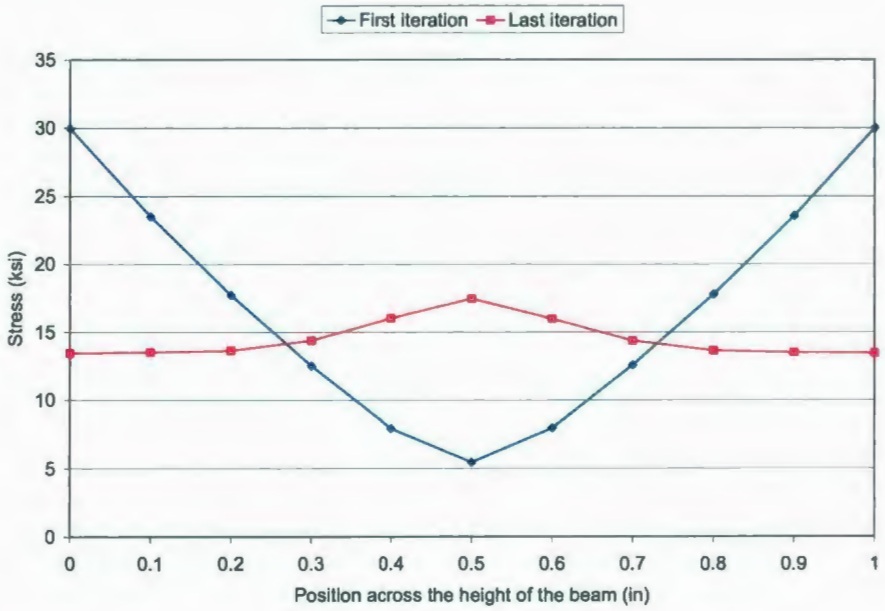


Fig. 4.9: Comparison of equivalent stress distribution along the height of the beam in the initial iteration to that in the last iteration of stress redistribution.

For a perfectly plastic material, when the hinge is formed the stress remains constant at the yield limit σ_y. Therefore, at the collapse state, the stress at the hinges will have the same value. This is indicated by the R-Node stress distribution at the last iteration shown in Fig. 4.10 as R-Node stress peaks approach an average value. Considering only maximum R-Node peak, Fig. 4.11 shows the convergence of the limit load using classical upper bound solution, the classical lower bound solution and the iterative R-Node method, compared to the analytical solution expressed by equation (4.13).

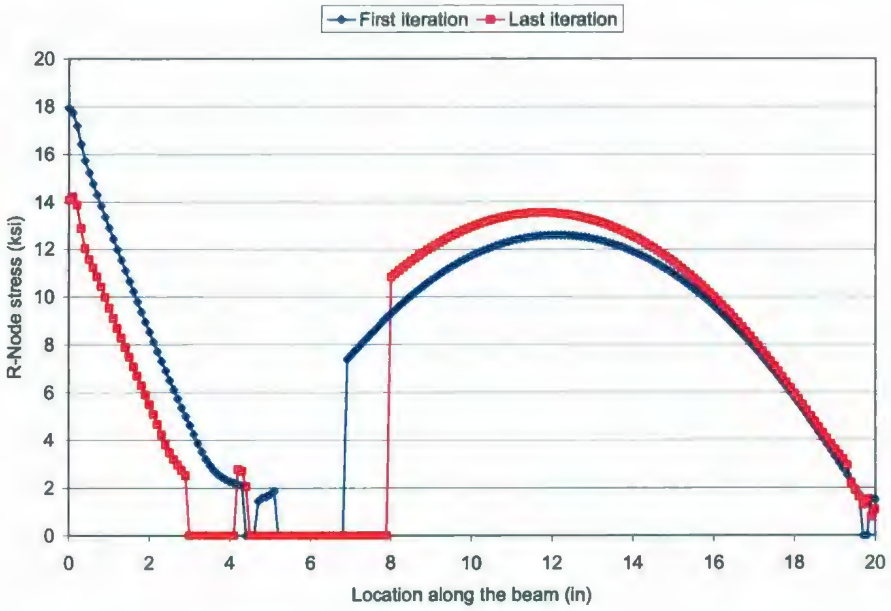


Fig. 4.10: R-Node stress distribution along the beam for the first and last iteration of the redistribution analysis.

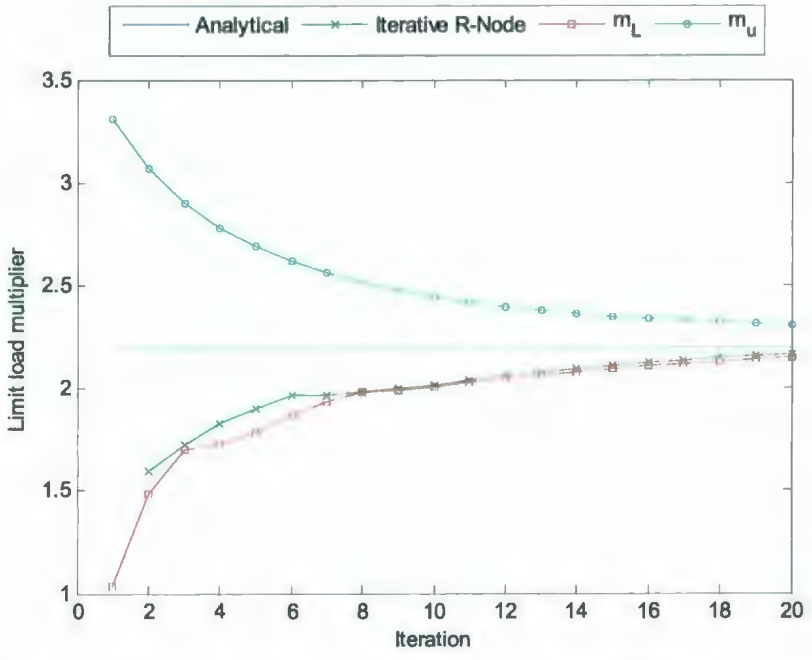


Fig. 4.11: Comparison of the convergence of the limit load multipliers of the indeterminate beam.

4.3.3 Oblique Nozzle

The oblique nozzle has numerous parameters in its geometry which affect its overall strength. Figure 4.12 shows a cross-section of an oblique nozzle connected to a pressure vessel that was used for experimental and numerical analysis of the stress distribution and the limit load by Sang et al [20]. The basic geometrical parameters are the nominal diameters and thickness of the vessel, the diameter and thickness of the nozzle and the oblique angle. The model is analyzed for its limit pressure and compared with the experimental results presented by Sang et al [20].

The geometry consists of a pressure vessel with two torispherical heads, with a closed nozzle connected at 30° angle as shown in Fig. 4.12. The inner diameter of the vessel is 60 cm and the outer diameter of the nozzle is 32.5 cm. The length of the shell is 2.4 meters. The length of the nozzle along the center line is 60 cm. The thickness of the vessel and the nozzle is 6 mm. The geometry is meshed using S4 4-noded shell elements with 4 integration points as shown in Fig. 4.13. The model is analyzed using the iterative R-Node method, the results of which are compared to experimental values from Sang et al [20], the classical upper-bound and classical lower-bound solutions.

The subroutine used in this problem is the same as that used in the previous example. The limit load calculated using the iterative R-Node analysis compares well with the experimental value as shown in Fig. 4.14 and summarized Table 4.1. The discrepancy in the value of the load to cause the initial yield is due to the fact that, in Sang et al [20], the stress is measured at a point that is slightly farther from the point of maximum stress.

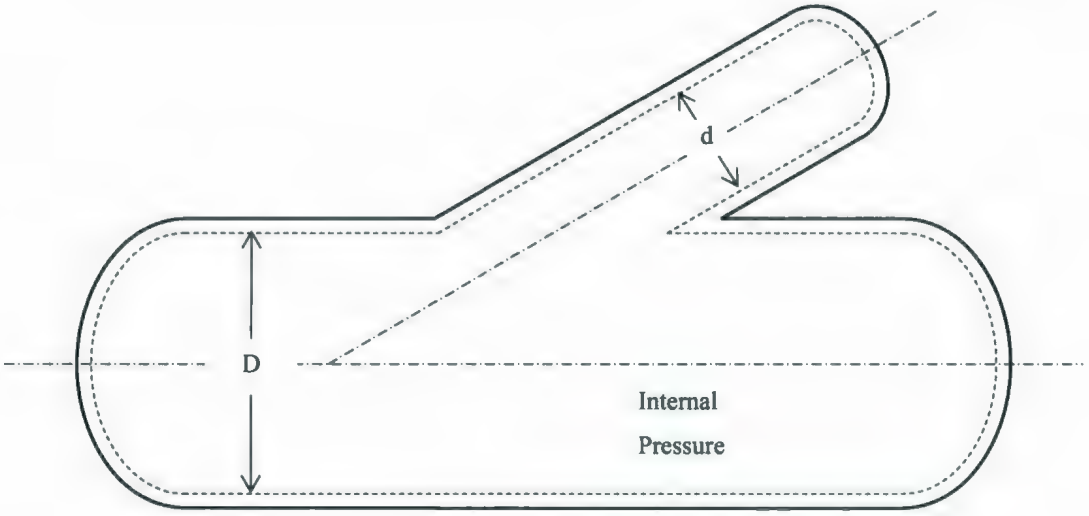


Fig. 4.12: Schematic diagram of the oblique nozzle as modeled by Sang et al [20].

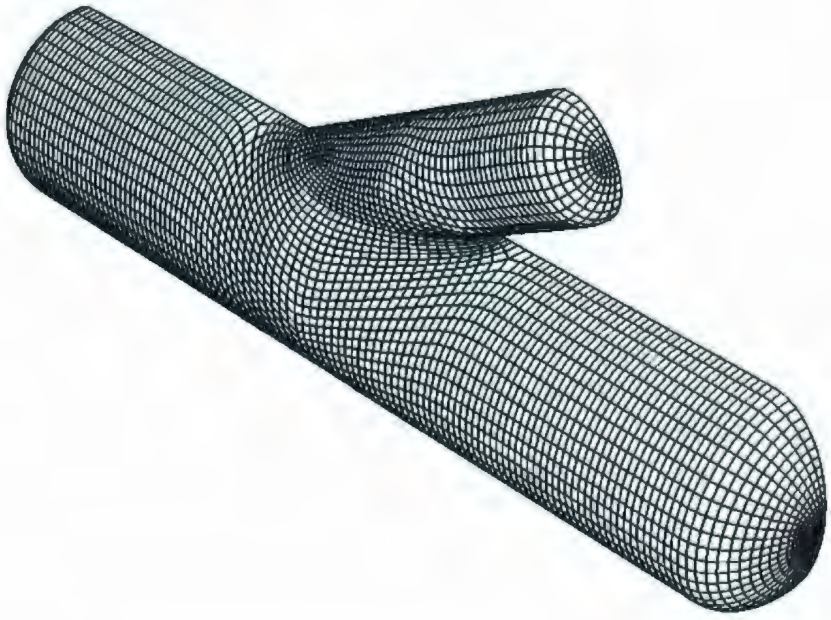


Fig. 4.13: Mesh of the oblique nozzle using shell elements.

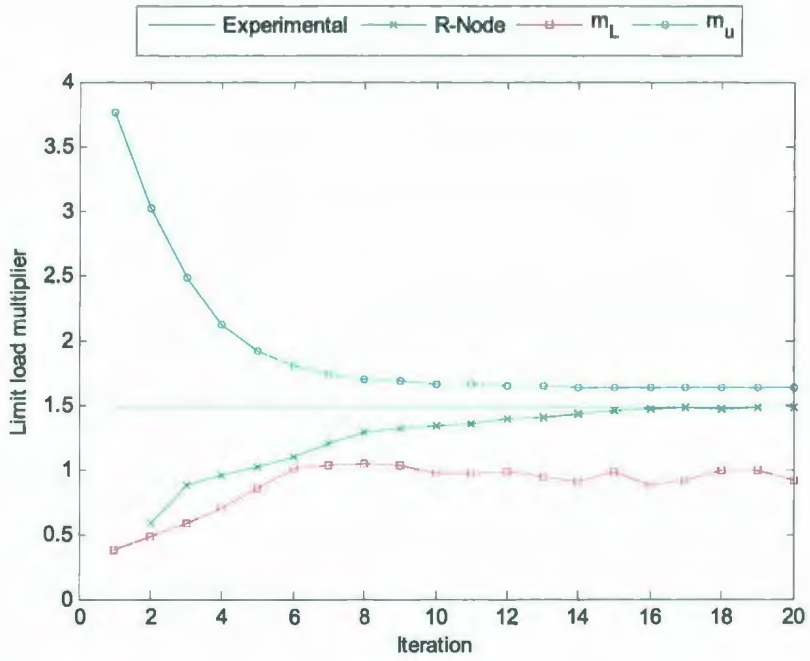


Fig. 4.14: Comparison of the convergence of the limit load multipliers of the oblique nozzle.

Table 4.1: Oblique nozzle analysis results

Load	Iterative R-Node	Elastic-Plastic FE
P_y (MPa)	0.40	1.00
P_{LL} (MPa)	1.45	1.48*

$m^0_{i=1} = 5.299, m^0_{i=2} = 5.286 \Rightarrow m^0_{i=1} - m^0_{i=2} > 0$

$P_{LL}/1.5 = 0.98$ MPa

* Experimental result from [20]

4.4 Summary

The iterative R-Node method is implemented in the ABAQUS commercial finite element analysis program through a user defined material. In the developed code, the constitutive relation is calculated at the integration points according to the stress history. Thus, a redistribution of the stresses is calculated at several increments until a steady

state is reached showing the stress distribution at the state of collapse. The R-Nodes location and stress values are calculated after the redistribution analysis is performed by comparing the stress field of each increment to that of the first increment which is performed using the actual material properties.

The developed algorithm is applied to several problems of different forms of finite element simulations. It is applied first to a thick cylinder modeled using plane strain elements and an indeterminate beam problem modeled using plane strain elements, the results of which are compared to the respective analytical solution. The method is also applied to a complex oblique nozzle problem modeled using shell elements, the results of which are compared to experimental results. It is shown that the R-Node analysis results compared well with other limit load analysis methods using the same simplified linear analysis method. In the next chapter, the R-Node analysis method is applied to further complex problems of multiple loads.

CHAPTER 5

SEQUENTIAL MULTIPLE LOADS

5.1 Overview

If multiple loads are applied, the limit value of each load would have the same form as equation (2.10) provided that the R-Node stress would be evaluated on the basis of the application of the combined loads. If it is required to determine the limit of one of the loads while fixing the others, the analysis is performed several times with different trials of the loads until the required fixed load is achieved. A simple procedure is illustrated that directs the trials towards the required solution.

5.2 Iterative Limit-Load Analysis

A sample problem of the cantilever beam subjected to two different loads is considered to help illustrate the procedure. Figure 5.1 illustrates the beam being subjected to both an axial force and a distributed load. The reaction moment at point A is expressed as

$$M = \frac{WL^2}{2} \tag{5.1}$$

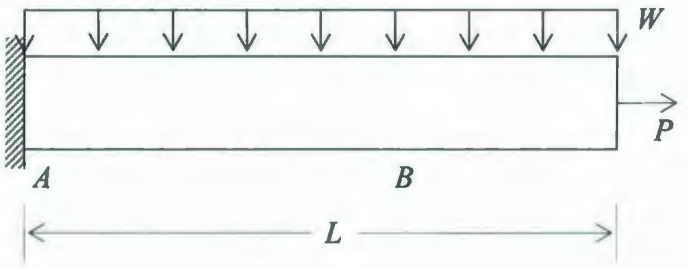


Fig. 5.1: Schematic diagram of the cantilever beam with length of 20 inches and unit height and depth subjected to distribution load and axial force.

Figure 5.2 illustrates the limit curve of the beam. To find the limit distributed load for a fixed axial force P_1, an initial R-Node analysis step will be attempted using an arbitrary distributed load combined with the required fixed force which is indicated by point A. This will give a value for m which is the multiplier for this combination of loads to give the combined limit load which would be at point A'. This value of m will be used in another R-Node analysis using the same fixed axial force P_1, but with the limit distributed load calculated at point A'. This loading is indicated by point B. The procedure will be applied to move from point B to points C and D. The more the iterations, the closer one gets to the exact solution. However, by several trials of load combinations, it was found that about 4 to 5 iterations would give an appropriate result.

If the fixed axial force is small, the initial distributed load value used in the R-Node analysis would affect the number of iterations required to reach the final answer. If the initial value of the distributed load is small, the number of iterations will be large. This is because initial value of the combination of the two loads is far from the actual limit curve. To overcome this problem, an initial value must be estimated for the distributed load that would be close enough to the final answer in order to have a minimum number of iterations. A good estimate can be obtained by initially assuming a linear limit line as shown in Fig. 5.3. The linear limit line is found by calculating the limit distributed load and axial force separately using two R-Node analyses. Point A is the intersection of the vertical axis at $P' = P_1$, which is the fixed axial force, and the linear

limit line. Knowing point *A*, the iterative R-Node procedure could be applied. It can be seen from Fig. 5.3 that the number of iterations is reduced when compared to the scheme shown in Fig. 5.2, reaching a solution very close to the analytical value.

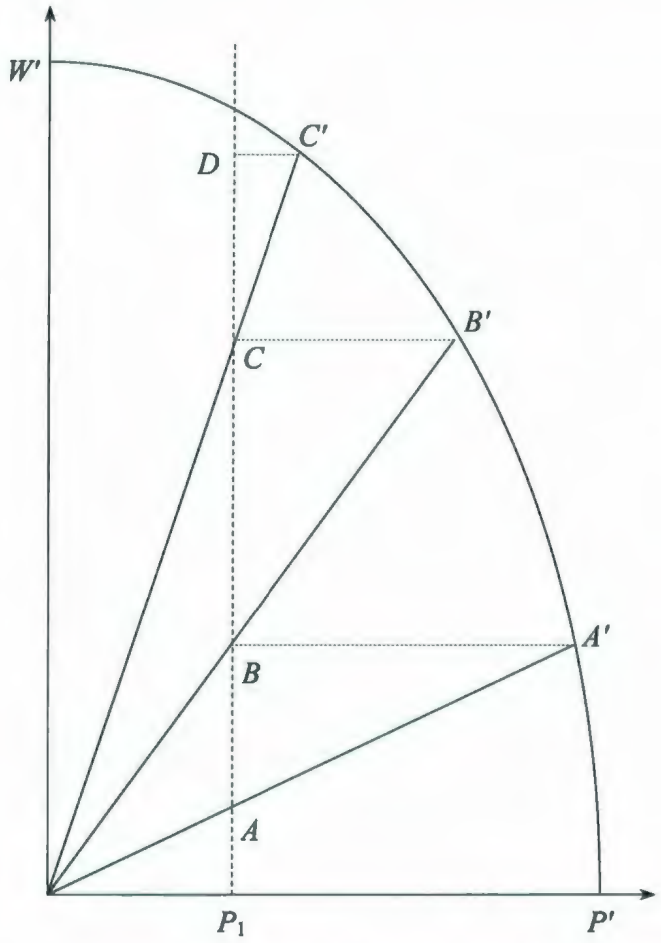


Fig. 5.2: Iterative-limit load analysis illustrated on the limit curve of the determinate beam.

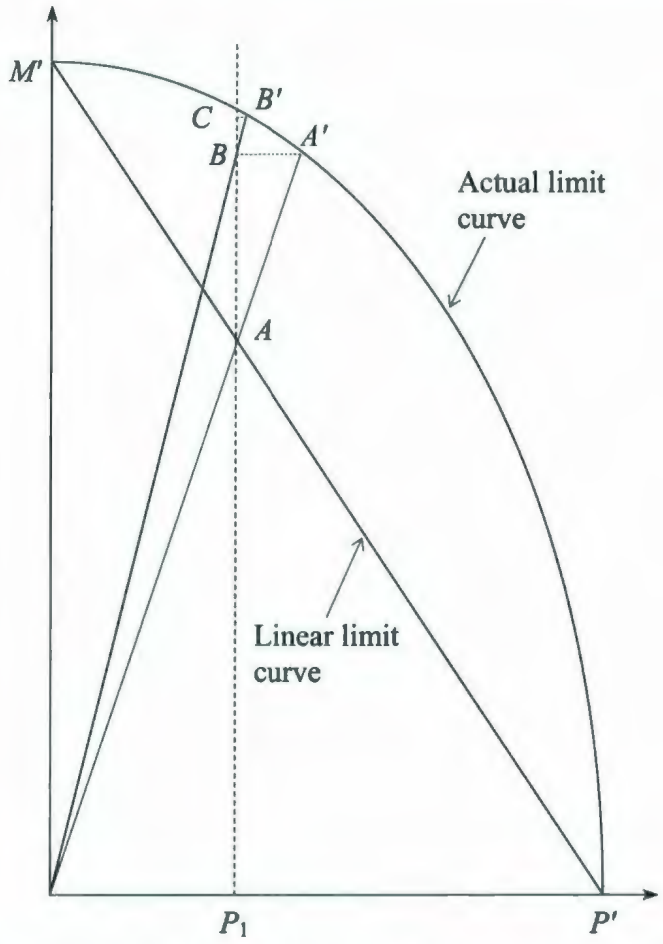


Fig. 5.3: Iterative-limit load analysis with an improved estimate of the initial value illustrated on the limit curve of the determinate beam.

5.3 Applications

The procedure of the iterative limit load analysis is applied to the cantilever beam as illustrated in the previous section. Its results are compared to the analytical solution. Another example that is used to illustrate the use of the procedure is the problem of a pipe bend subjected to both internal pressure and bending moment. The pipe bend has been a problem of great interest in many researches. It has a very complex response to in-plane and out-of-plane moments. Also, the variation of the limit moment with the change in the internal pressure has been of great interest and has been studied numerically and

analytically. In the present work, the pipe bend is analyzed for in-plane bending moment using the R-Node and the iterative R-Node analysis. The results of the analysis is compared to previous work done by Shalaby and Younan [22] and Mourad and Younan [23] who analyzed the pipe bend as a stand alone component checking for the effect of the internal pressure on the in-plane and out-of-plane limit moment, respectively. Also, the results of the R-Node analysis will be compared to that of Chattopadhyay [24] who analyzed the pipe bend analytically, and developed a limit equation for the moment and the internal pressure.

5.3.1 Cantilever Beam Model

In this model, a sample cantilever beam illustrated in Fig. 5.1 is modeled having a length of 20 inches, a height of 1 inch and a depth of 1 inch. The elastic modulus is 30×10^6 psi, the yield strength is 30×10^3 psi and the Poisson's ratio is 0.49. The model is developed using 4-node plane stress elements. On the basis of several trials, it was found that the R-Node stress can be found with an acceptable accuracy using 11 nodes across the height of the beam. Therefore, the beam is divided into 200×10 elements.

The nodes at the fixed end are constrained in the x-direction, and only the mid point along that side is constrained in the y-direction. The nodes at the free end are coupled together in a way such that the slopes of the lines between the adjacent nodes would be equal. A variable pressure is applied along the free edge equivalent to the designated combination of moment and axial force.

First, different combinations of the bending moment and axial force are applied to the beam to demonstrate the iterative R-Node analysis and generate the limit curve of the beam. Also, the limit load is calculated using the classical upper-bound multipliers given by equation (2.46) and the classical lower-bound multiplier given by equation (2.11) that use the same results of the finite element analysis. The results of the finite element analysis are compared to the exact solution which is governed by the equation

$$\frac{2}{3}\left(\frac{M}{M_y}\right)+\left(\frac{P}{P_y}\right)^2=1 \tag{5.2}$$

where M_y is the moment to cause initial yielding in bending and P_y is axial force to cause initial yielding in tension.

Then the iterative limit load analysis using the iterative R-Node method is used to find the limit moments for different value of the axial force. These results are compared to that of equation (5.2). The problem is analyzed using the Elastic Modulus Adjustment Procedure by Seshadri and Fernando [4]. Figure 5.4 shows a sample calculation for the limit multiplier of the beam subjected to a distributed load of 90 N/m and an axial force 2000 N. The results show that the R-Node analysis converges effectively as a lower bound towards the exact solution, which is also approached by the upper bound solution.

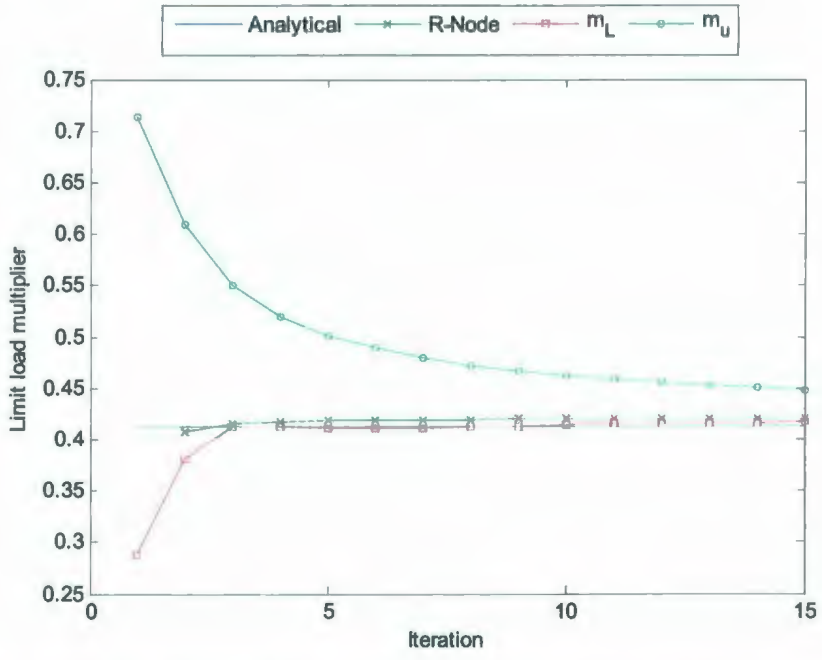


Fig. 5.4: Limit load multiplier of the cantilever beam subjected to 90 N/m distributed load and 2000 N axial load.

Figure 5.5 shows the limit curve comparing the analytical solution to the R-Node and iterative limit load analysis. The R-Node analysis is performed for different combinations of the axial force and the bending moment. The iterative limit load analysis is performed for arbitrarily selected values of the axial force to find the limit moment. The figure illustrates the convergence of the iterative R-Node analysis as well as the iterative limit load analysis to the analytical solution of the limit curve for all combinations of the load.

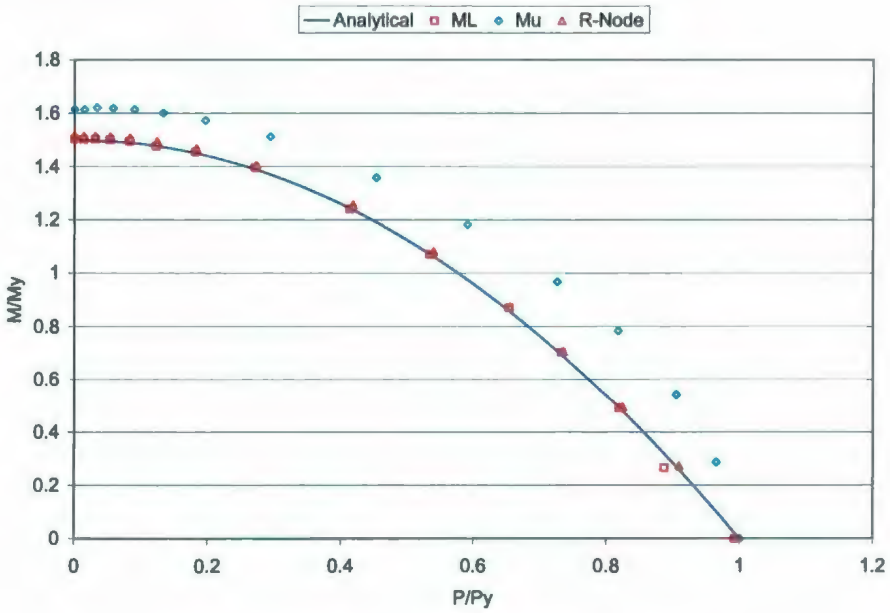


Fig. 5.5: Limit curve of the cantilever beam.

5.3.2 Pipe Bend Model

The pipe bend to be analyzed in the present work is similar to that presented by Mourad and Younan [23]. The bend factor h of the elbow is defined as

$$h = \frac{Rt}{r^2} \tag{5.3}$$

where R is the radius of curvature of the elbow's center-line, t is the pipe wall thickness and r is the mean pipe radius. The bend factor of the elbow presented in the present work

is 0.1615, which is the one selected by Mourad and Younan [23], the nominal pipe radius is 8 in, the thickness is 0.4307 in and the bend radius is 24 in. Mourad and Younan [23] have modeled the elbow standalone using the special 2-node elbow element. However, in the present work, it is modeled using shell elements to be able to apply the Elastic Modulus Adjustment Procedures. In order to be able to apply a moment to circular edge of the elbow, the nodes at the edge must be coupled together to a single point at which the load is applied. However, this would prevent the edge from deforming freely (either warping or ovalization). Therefore, two pipes connected to the elbow, as illustrated in Fig. 5.6, are modeled so that the load would be applied to the end of the pipe and, thus, transferred to the elbow allowing its the edge to deform without any constraints. The length of the pipes would be four times the diameter in order to reduce the effect of the pipe ends on the response of the elbow. It must be noted that a standalone elbow would give more conservative results for the limit load of the bending moment as Mourad and Younan [23] have indicated.

The model used for elastic-plastic material is meshed using 4-node shell element with 5 section points along the thickness. Figure 5.7 shows the mesh for the whole geometry. The model is fixed at one end as indicated in Fig. 5.6. The material model used is elastic-perfectly plastic Stainless Steel 304 having an elastic modulus of 28.1×10^6 psi, a yield stress of 39.44×10^3 psi and a Poisson's ratio of 0.28 at room temperature. The model used for the EMAP has the same configuration except that the shell element shall have 10 layers and a Poisson's ratio of 0.49. The layers are introduced to be able to find the R-Node stress through the thickness as explained earlier in Section 4.2.

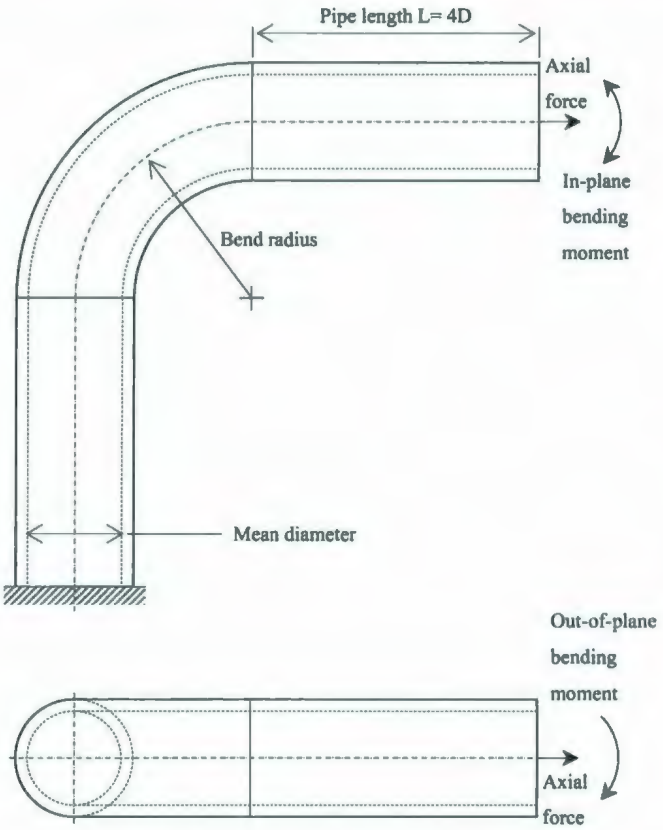


Fig. 5.6: Schematic diagram of the pipe bend.

The model is analyzed for the limit in-plane closing and out-of-plane moments, as indicated in Fig. 5.6, at various internal pressure levels. The assumed pressure range would start from zero to the pressure to cause initial yielding. The limit loads are calculated using the elastic-plastic analysis, iterative R-Node and the other limit load analysis methods used in the beam problem to verify the convergence of the R-Node analysis for combined loading of the component. Then, the limit curve for in-plane and out-of-plane moments is generated using elastic-plastic analysis and R-Node analysis for comparison. The iterative limit load analysis is used to find the limit moments at specific values of the internal pressure. The results of the iterative R-Node analysis are compared to the generated limit curve.

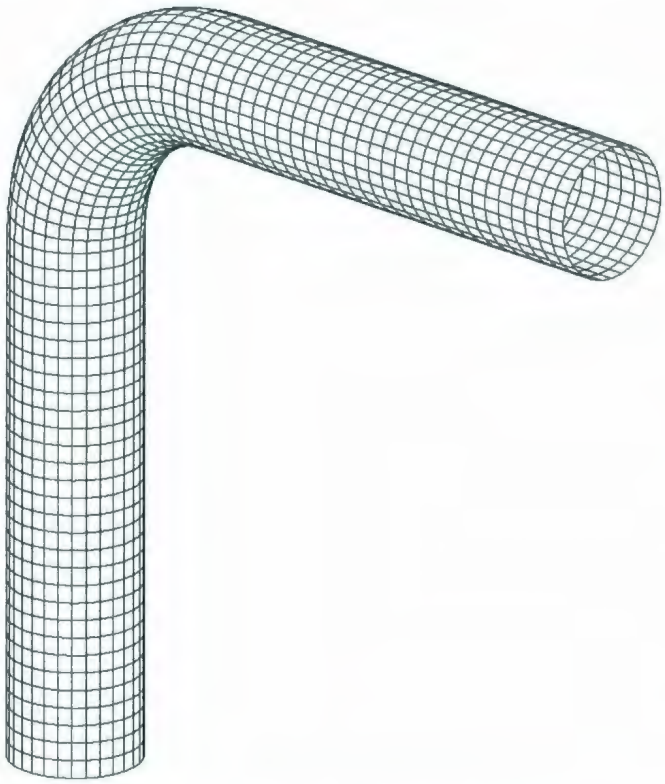


Fig. 5.7: Meshing of the pipe bend.

The results of the R-Node analysis of the pipe bend compared well with the other analysis methods. Figure 5.8 shows the limit load multipliers calculated from the results of EMAP analysis. The R-Node analysis converged as a lower bound to the solution estimated by the elastic-plastic analysis and the upper bound multipliers. Also, it was observed that the classical lower bound solution tends to diverge just after the second iteration because of several sources of local high stress points in the complex structure due to the complexity of the geometry.

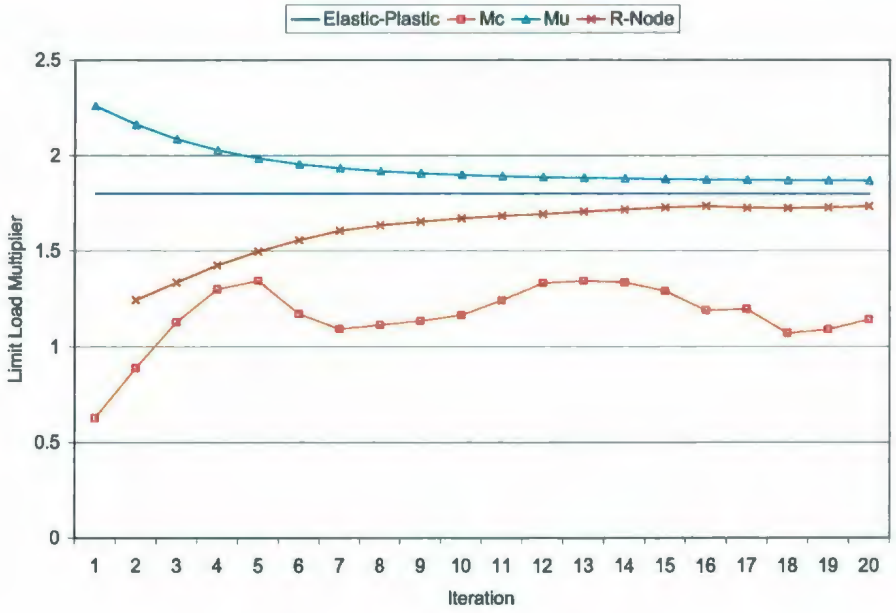


Fig. 5.8: Limit load of the pipe bend.

As it is expected, the limit moment decreases with an increase in the internal pressure. This is indicated in Figs. 5.9 and 5.10 showing the limit curve of the in-plane and out-of-plane moment, respectively, versus the internal pressure for the pipe bend estimated using the R-Node and plastic analysis. The results of the iterative limit load analysis are also shown illustrating their match with the plastic analysis. Table 5.1 shows the number of iterations used to calculate the in-plane and out-of-plane moments using the iterative limit load analysis. It can be observed from Table 5.1 that the solution converges faster for small values of the pressure and, consequently, small values of the slope of the limit curve. The same observation was made from the results of the limit load analysis of the in-plane and out-of-plane moment.

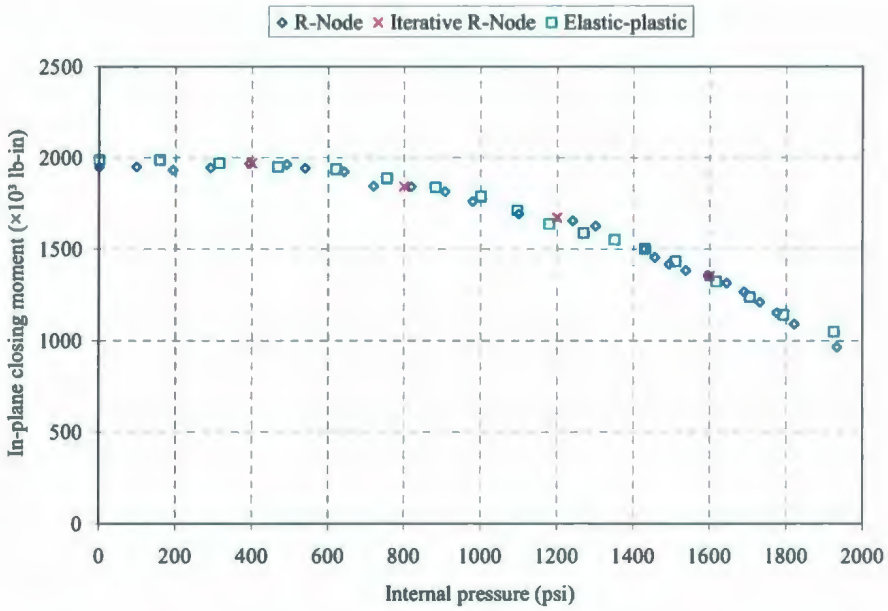


Fig. 5.9: The limit curves of the in-plane closing moment versus the internal pressure of the pipe bend.

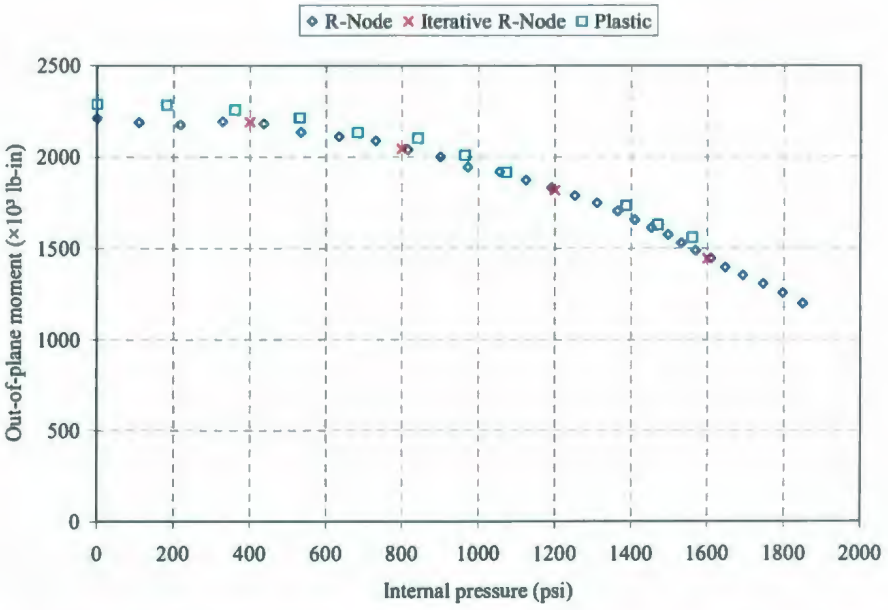


Fig. 5.10: The limit curves of the out-of-plane moment versus the internal pressure of the pipe bend.

Table 5.1: Number of R-Node iterations versus pressure

	Number of iterations	
Internal pressure (psi)	In-plane	Out-of-plane
400	2	3
800	3	4
1200	3	4
1600	4	5

5.4 Summary

Limit load analysis methods determine the multipliers for all loads that are applied to a component. In cases where the component is subjected to other fixed loads in addition to the applied load, the solution to the limit multiplier will not be accurate if all the loads are considered in a single analysis. Hence, the iterative limit load algorithm is suggested in which several analyses are made while changing the load being studied and fixing the other loads. The procedure is verified through the analytical solution of the cantilever beam subjected to bending and axial load and the non-linear solution of the pipe bend subjected to internal pressure along with in-plane and out-of-plane moments. The R-Node is used in these problems being a lower bound solution. In the next chapter, another application of the R-Node method in the design of pressure components is illustrated in which it is used as a tool for stress classification to find the design according to the ASME codes.

CHAPTER 6

STRESS CLASSIFICATION

6.1 Overview

The ASME Code Section III and Section VIII (Division 2) provide stress classification guidelines to interpret the results of a linear elastic finite element analysis. These guidelines enable the splitting of the generated stresses into primary, secondary and peak stress. The code gives some examples to explain the suggested procedures. Although these examples may reflect a wide range of applications in the field of pressure vessel and piping, the guidelines are difficult to use with complex geometries. Hence, it would of great advantage to have a detailed general procedure for stress classification. In the present work, the R-Node method illustrated earlier is used to investigate the primary stresses and their locations in both simple and complex geometries. The method is verified using the plane beam and axisymmetric torispherical head. Also, the method is applied to analyze 3D straight and oblique nozzle modeled using both solid and shell elements. The results of the analysis of the oblique nozzle are compared with recently published experimental data.

6.2 Stress Classification Techniques

6.2.1 Stress Linearization in FE Programs

An option is available in the ABAQUS [25] commercial code that performs a stress classification along a predefined path. The path defined by two nodes within the model as shown in Fig. 6.1.

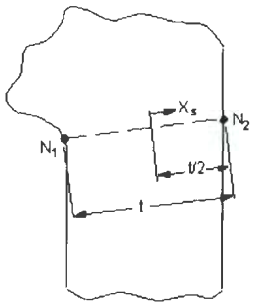


Fig. 6.1: Coordinates of Cross Section

The procedure is divided into main routines for planar and 3D applications. The program splits the stresses into membrane (uniform), bending (linear slope along the path) stresses and peak stresses. For the cartesian system of coordinates, the membrane stress is given by

$$\sigma_i^m = \frac{1}{t}\int_{-t/2}^{t/2} \sigma_i dx_s \tag{6.1}$$

where σ_i is a stress component, t is the length of the path (thickness) and x_s is the coordinate along the path. The magnitude of bending stress at the extreme points of the path is given by

$$\sigma_i^b = -\frac{6}{t^2}\int_{-t/2}^{t/2} \sigma_i x_s dx_s \tag{6.2}$$

It must be noted that the bending stress at the extremes will be opposite in sign. Hence, the peak stress at any point along the path will be

$$\sigma_i^p = \sigma_i - \left(\sigma_i^m + \sigma_i^b\right) \tag{6.3}$$

where σ_i is the total stress calculated in the finite element analysis.

6.2.2 R-Node Analysis

The R-Node analysis determines the reference stress in a component using two linear elastic stress analyses. An initial elastic analysis is performed from which the results are used for the second analysis. The modulus of elasticity is modified throughout the entire structure at every Gaussian integration point of every element. The modification is made using the EMAP [6] formula which is expressed as

$$E_{n,i+1} = \left(\frac{\sigma_a}{\sigma_{en,i}} \right)^q E_{n,i} \tag{6.4}$$

Subsequently, a second linear elastic analysis is performed using the modified elastic modulus. The result of the elastic analyses shows the redistribution of stress as explained by Seshadri [17]. Figure 6.2 shows the GLOSS analysis diagram.

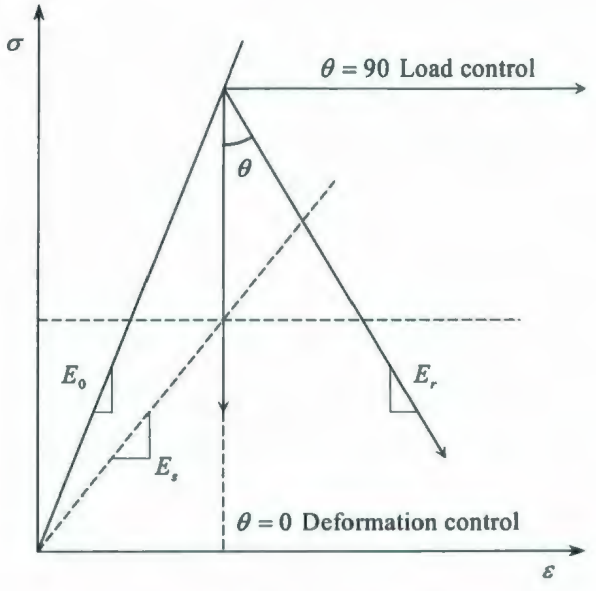


Fig. 6.2: GLOSS diagram.

The points where the stress is not affected by redistribution of the stresses are in direct equilibrium with the externally applied load, or are load controlled. These points are defined as the R-Nodes. Seshadri [17] has shown that, in order to ensure that the R-

Node stress is converging as a lower bound solution, successive distributions should meet the following conditions

$$\max\left[(\sigma_e)_{r-node}\right] \leq S_m \tag{6.5}$$

$$m_{i=1}^0 - m_{i=2}^0 \geq 0 \tag{6.6}$$

where

$$m_i^0 = \sigma_y \sqrt{\frac{V_T}{\sum_{k=1}^{N_T}\left(\sigma_{eik}^2 V_k\right)_i}} \tag{6.7}$$

The magnitude of R-Node stresses in a component tends to be high at the locations of plastic hinges or plasticity initiation, known as R-Node stress peaks, and becomes smaller elsewhere. Hence, a system that fails due to the formation of a single hinge can be represented by a one-bar model in which collapse occurs when the R-Node stress reaches the yield limit. In the case of a system with more than one hinge, it can be represented by a multi-bar model. In this case, collapse occurs when the average of the R-Node peaks reaches the yield limit [4]. When several iterations of the redistribution are performed, the R-Node peak stresses all approach the same value. The final value of the R-Node stress peaks after several redistribution iteration approaches the reference stress corresponding to plastic collapse.

Since the R-Node stresses are load-controlled, it is equivalent to the primary stress calculated using the ASME guidelines. In the case of pure membrane stress, there will be no redistribution in the stresses, which makes the R-Node stress to be equal to the membrane stress. In the case of pure bending, the R-Node stress will be less than the maximum elastic value according to the "shape factor" as indicated in Fig. 6.3.

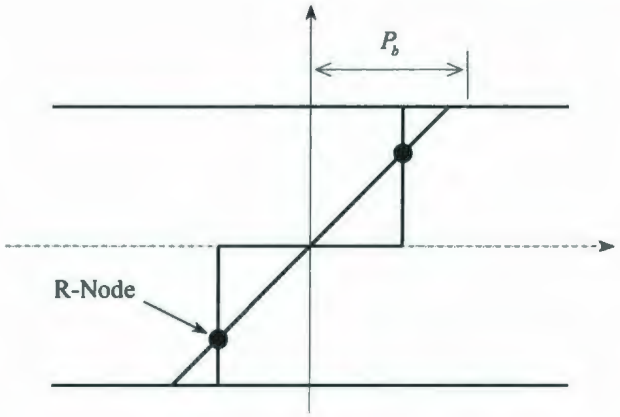


Fig. 6.3: Bending stress distribution of the initial elastic and redistribution analyses.

In the case of a combined membrane and bending stresses, the location of the R-Node shifts from the location found using the pure bending as shown in Fig. 6.4. Therefore, the R-Node can be used to find the primary stress as an equivalent quantity by combining both the membrane and the bending components. In this case, the R-Node will be limited to the allowable stress S_m according to equations (6) and (7).

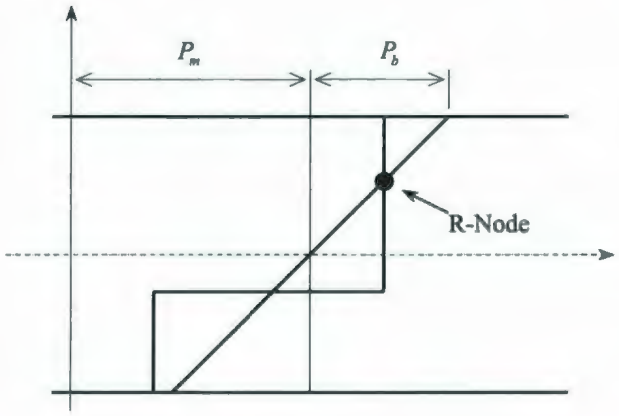


Fig. 6.4: Membrane plus bending stress distribution of the initial elastic and redistribution analyses.

6.2.3 Elastic-Plastic Analysis

The elastic-plastic analysis is used to verify the results of the stress-classification methods (stress linearization and R-Node analysis). This is achieved by finding the load that causes the first hinge from the non-linear analysis as compared to the maximum equivalent primary membrane stress P_{eq} and the maximum R-Node peak stress. In addition, the collapse load enables an evaluation of the acceptable design load.

6.3 Proposed Methodology

It appears that the current stress classification tools may have their limitations in the analysis of complex geometries. Therefore, it is suggested that the R-Node method be added as a tool to find the primary stress within a component in conjunction with other classification tools. In this paper, an analysis procedure is proposed to illustrate the use of the R-Node method.

1. A linear elastic FE analysis is carried out to find the load that causes initial yielding.
2. The results of the linear elastic analysis are used in conjunction with the stress linearization tool in ABAQUS post-processor to find P_m, P_b and F.
3. Another linear elastic analysis is carried out in conjunction with EMAP [6] to locate the R-Nodes in the pressure component. The maximum R-Node equivalent stress is identified and limited to S_m. This ensures satisfaction of both P_m and P_b.
4. A complete elastic-plastic analysis is carried out in order to compare results obtained by the foregoing methods.

The proposed methodology is applied to four problems for the purpose of verification and to illustrate its applications.

6.4 Applications

The example of the indeterminate beam is used to verify the procedure. Subsequently, it is used to analyze an axisymmetric model of a pressure vessel with torisperical head, and full 3D shell model of an oblique nozzle that is welded to a pressure vessel.

The stress classification methods outlined earlier in both ASME and ABAQUS are demonstrated mainly with plane geometries (plane stress, plane strain and axisymmetric problems). In the present work, two geometries are used to demonstrate the use of the R-Node in stress classification. The first is for a thick cylinder created using plane strain elements and the second is for a nozzle connected at 90° to spherical shell. The models are analyzed for their elastic responses. The results are then used to perform the stress classification according to the ASME guidelines and the ABAQUS linearization procedure. An R-Node analysis is performed for the two models to check for the primary stresses and compare the results. The detailed analysis of the problem of the nozzle is published by Kroenke [26] illustrating the stress classification lines and the detailed analysis according to the ASME guidelines.

6.4.1 Indeterminate Beam

The analytical solution of the indeterminate beam problem is shown in section 4.3.2 where the loads that cause initial yield, first hinge and the collapse load are found. The length of the beam is 20 in with unit height and width. The beam is subjected to a distributed load as shown in Fig. 6.5. The model is meshed using plane stress elements for the elastic modulus adjustment procedure.

The model is also analyzed using the R-Node method. The results of the initial elastic analysis are used for stress linearization carried out using the ABAQUS code. Also, in implementation of the R-Node analysis, the user-defined subroutine UMAT

developed earlier to define the behavior of a user material is used. In this subroutine, the modulus of elasticity is adjusted for every Gaussian integration point according its value in the previous increment using equation (2.3). Since it is required to find the R-Node stress with no redistribution in the stresses, the value of q is assumed to be 0.1 in order to minimize the stress redistribution.

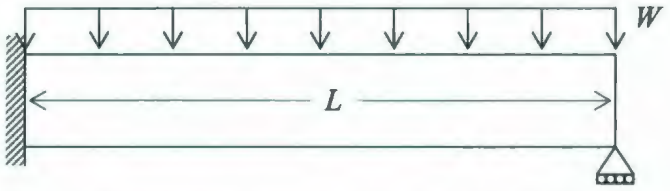


Fig. 6.5: Schematic diagram of the indeterminate beam.

The R-Node location is found by comparing the stress distribution of the initial elastic analysis with that of the next EMAP iteration. The comparison is carried out for each pair of adjacent nodes of every element. Thus, whenever there is an intersection, it would represent an R-Node location.

The results of all the analyses are shown in Table 6.1. The value of the load that causes initial yielding, W_y, is calculated analytically using equation (4.11) and from the results of the linear elastic analysis. It is compared to the value of the calculated design load W_{design} that is evaluated using stress classification and R-Node methods (primary stress $\leq S_m$). Also, the load that causes the first hinge, W_{h1}, is calculated analytically using equation (4.12) and using the results of the elastic-plastic analysis. Besides, the same load is calculated using the stress classification and R-Node methods by limiting the equivalent primary membrane and the R-Node stresses to the yield limit. These results are compared to verify the validity of the R-Node method.

The R-Node and stress linearization analyses are carried out at several sections along the beam. Figure 6.6 shows the result of the analyses as well as the maximum bending stress at every section. It can be seen that the R-Node stress value matches well

with the value of P_{eq}. If the value of the maximum R-Node stress is limited to the yield stress, the load will be 155.6 lb/in, which is the load that causes the first hinge as explained earlier. It compares well with those calculated using the analytical (150 lb/in), stress classification (153.5 lb/in) and elastic-plastic analyses (153.3 lb/in).

Table 6.1: Indeterminate beam analysis.

Load	Analytical	Stress classification	R-Node	Elastic-Plastic FE
W_{design} (lb/in)	-	102.3	103.8	-
W_y (lb/in)	100.0	-	-	98.9
W_{h1} (lb/in)	150.0	153.5	155.6	153.3
W_{LL} (lb/in)	218.3	-	-	222.0

$m^0_{i=1} = 1.668208, m^0_{i=2} = 1.668202 \Rightarrow m^0_{i=1} - m^0_{i=2} > 0$

$W_{LL}/1.5 = 145.53$ lb/in

If the maximum R-Node stress is limited to the allowable stress S_m, the load will be 103.8 lb/in, which compares well with that calculated using the stress classification analysis, as well as the load that will initiate yielding, as shown in Table 6.1.

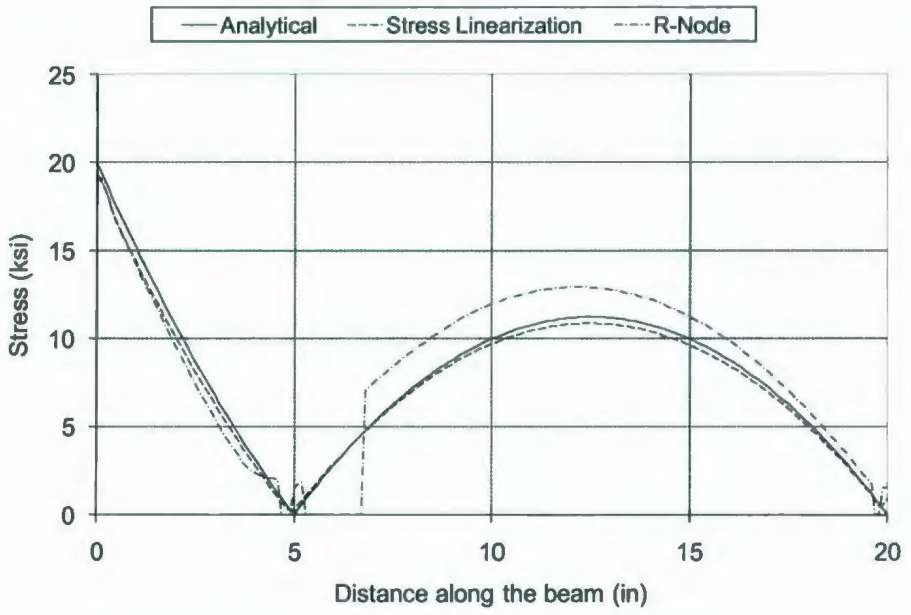


Fig. 6.6: Primary stress distribution along the beam.

In an attempt to extend the understanding of the behavior of the R-Node stress in comparison to that of the stress linearization, the beam is loaded axially to introduce a membrane component to the stress distribution across the section of the beam. The results are observed for different combinations of load values in order to generate the interaction curve between the axial and bending loads. Figure 6.7 shows the generated curves comparing the results of the normalized loads calculated using analytical solution of the bending stress using equation (4.10a), numerical stress linearization and R-Node analysis. It can be seen that the result of the stress classification and linearization generates a discontinuous curve since it is actually generated using two different curves – one for limiting the bending stress and the other for limiting the membrane stress.

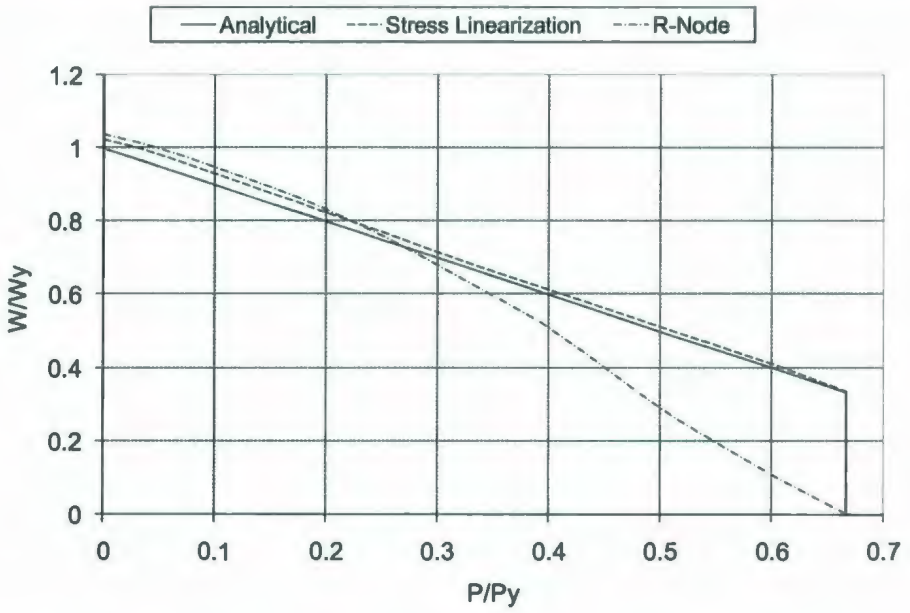


Fig. 6.7: Primary stress for combined membrane and bending loads.

However, the R-Node stress curve depends on the combined loading, and, thus, is a continuous curve having a similar trend joining the two maximum points of loads. This explains the fact that the design load calculated using the R-Node analysis is slightly lower than that using other methods in cases of high bending and membrane loading, which is due to comparing of a discontinuous curve to a continuous one.

6.4.2 Axisymmetric Pressure Vessel

In this problem, an axisymmetric model of a pressure vessel with a torispherical head is developed as shown in Fig. 6.8. It is analyzed using elastic-plastic FEA, stress linearization method and the R-Node method. The vessel is subjected to an internal pressure. As Seshadri and Fernando [4] have explained, this geometry is expected to collapse after the formation of three hinges whose locations are schematically shown in Fig. 6.8. Hence, three R-Node peaks are expected at these locations.

In the R-Node analysis, the same subroutine defined earlier in the beam problem is used after adjusting the various equations to account for the axisymmetric behavior. The R-Nodes are checked for every element as explained earlier. Figure 6.9 shows the maximum elastic stress, equivalent membrane stress and the R-Node stress along the walls of the vessel starting from the crown of the head. It can be observed that the results of the R-Node compares well with that of stress classification. Some discrepancies may occur due to the fact that, at all sections of the vessel walls, there are both membrane and bending stresses. Hence, as discussed earlier in the beam problem, at points where there are high values of both stresses, the R-Node stress would be higher, and, at points of dominating bending stress, the R-Node stress would be slightly lower.

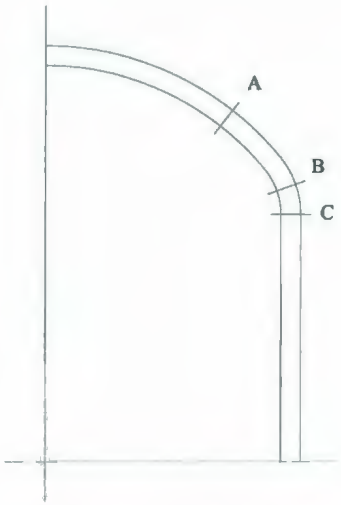


Fig. 6.8: Schematic of the geometry with the expected hinge locations.

Table 6.2 shows the results of the analyses. It is seen that there is a slight discrepancy between loads calculated using the R-Node and the stress classification methods. This arises due to the combination of both high bending and small membrane stresses. Also, the design load calculated using both the R-Node and the stress classification methods are slightly higher than the initial yield because of the existence of secondary stresses in the component.

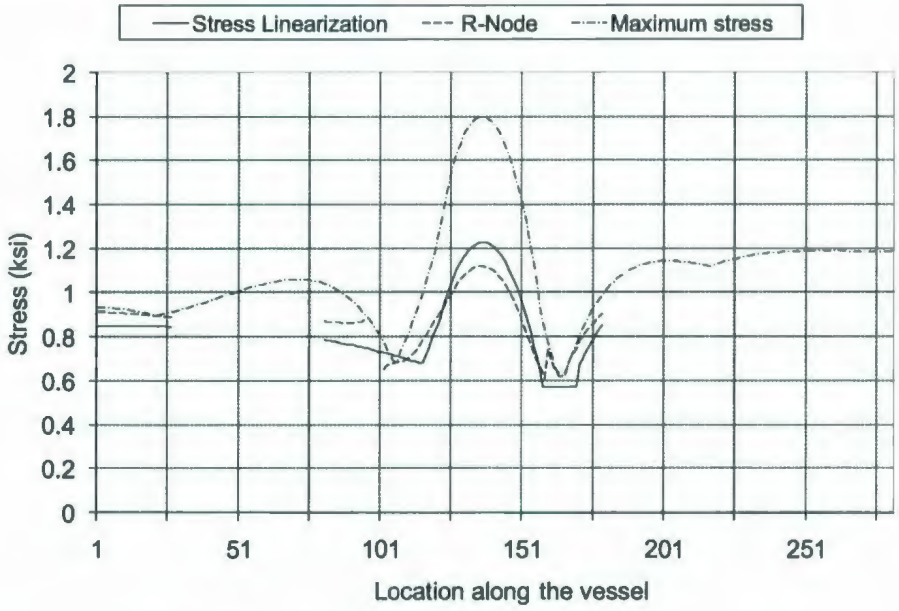


Fig. 6.9: Primary stress distribution along the vessel wall.

Table 6.2: Axisymmetric pressure vessel analysis.

Load	Stress classification	R-Node	Elastic-Plastic FE
P_{design} (psi)	1626.7	1781.2	-
P_y (psi)	-	-	1623.6
P_{h1} (psi)	2440.1	2671.8	2700.0
P_{LL} (psi)	-	-	2833.3

$m^0_{i=1} = 12.653, m^0_{i=2} = 12.561 \Rightarrow m^0_{i=1} - m^0_{i=2} > 0$

$P_{LL}/1.5 = 1888.89$ MPa

6.4.3 Straight Nozzle

The R-Node stress and location is a characteristic of the structure and it is independent on the analysis model. In this problem, a 90° nozzle connected to a cylindrical shell is analyzed using the R-Node method in two different models. In the first model, the geometry is simulated using quadratic solid tetrahedron elements and, in the second model, it is simulated using quadratic layered shell elements. Figure 6.10 shows a

schematic diagram of the geometry. The inside diameter of the shell is 30 mm and that of the nozzle is 100 mm. The wall thickness throughout the geometry is 6 mm. It is intended to have a large thickness in order to be able to have an acceptable mesh of solid element with adequate number of elements through the wall thickness. The layered shell elements are used instead of conventional ones so as to be able to find the R-Node stress and location through the thickness as explained by Fanous et al [27]. In order to have an acceptable number of stress calculation points across the thickness, 6 layers with 3 section points in each layer are used in the shell elements.

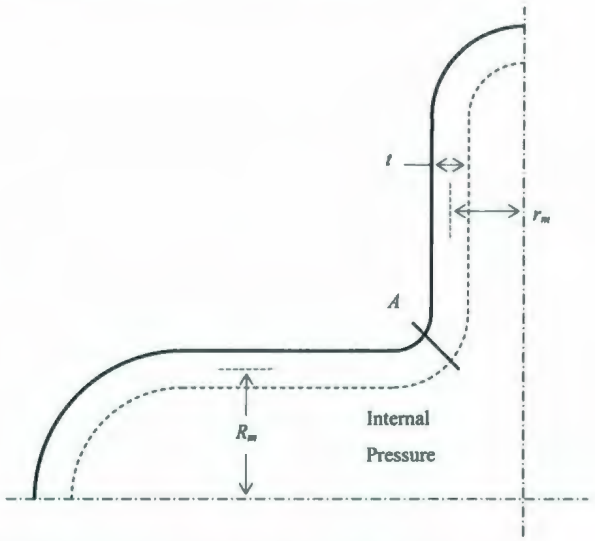


Fig. 6.10: Schematic diagram of the straight nozzle with $R_m = 50\text{mm}$ and $r_m = 15\text{mm}$.

The subroutine used earlier in the beam problem to modify the modulus of elasticity at the Gaussian integration points is similar to that used for the shell with slight modification to account for the number of layers. In this problem, the results of R-Node analysis are compared to limit load values. Hence, R-Node analysis is performed using $q = 0.1$ to find the design load. In addition, iterative R-Node analysis is performed to find the limit load (as explained by Fanous et al [27].)

The R-Node analysis of the first (solid) model showed that the maximum R-Node stress is 13.1 MPa at section A indicated in Fig. 6.10. This was found to compare well with the value of the equivalent membrane stress calculated using the stress classification method. Also, the maximum R-Node stress calculated using the second (shell) model is found to be 14.0 MPa at section A as well. Table 6.3 summarizes the results of the calculated loads.

Table 6.3: Straight nozzle analysis.

	Solid model			Shell model	
Load	Stress classification	R-Node	Elastic-plastic	R-Node	Elastic-plastic
P_{design} (MPa)	17.76	22.90	-	21.42	-
P_y (MPa)	-	-	12.19	-	13.77
P_{LL} (MPa)	-	-	33.51	-	32.75

$m^0_{i=1} = 4.44, m^0_{i=2} = 4.38 \Rightarrow m^0_{i=1} - m^0_{i=2} > 0$

$P_{LL}/1.5 = 21.83$ MPa

6.4.4 Comparison of Analysis methods

Table 6.4 shows a comparison between the computation times for the elastic-plastic analysis and the R-Node analysis. The linearization procedure involved insignificant computation times but the time consumed in analyzing the problem depends on the experience of the analyst and the ability to locate the appropriate stress classification lines and planes.

Table 6.4: Analysis times (seconds).

Problem	R-Node	Elastic-Plastic FE
Axi symmetric pressure vessel	24	219
Oblique nozzle	89	356
Straight nozzle	427	2070

6.5 Summary

The stress classification methods of the ASME codes require some experience in order to find the suitable locations and directions of the stress classification lines. The R-Node method is illustrated as a tool for stress classification to find the primary stress. The locations and values of the R-Node stresses are found by comparing two linear elastic analyses with minimum redistribution. The suggested analysis method has shown to be very effective in several applications in 2D and 3D modeling. It gave results with high accuracy, especially in cases with single type of stresses, when compared to the currently used stress classification method.

CHAPTER 7

THE REFERENCE VOLUME CONCEPT

7.1 Overview

Seshadri and Mangalaramanan [5] have observed that, if plastic collapse occurs over a localized region of the mechanical component or structure, m^0 will be significantly overestimated if it is calculated on the basis of the total volume, V_T. Furthermore, the corresponding m_L, which is calculated based on a single element that has the maximum equivalent stress in the component, will be underestimated. During local collapse, plastic action is confined to a sub-region of the total volume, and the remainder region, being still elastic, will become a zone with zero stress and strain. Hence, the magnitude of the upper bound multiplier (m^0) would depend on the sub-volume, V_β, where

$$V_\beta = \sum_{k=1}^{\beta} (V_k) \tag{7.1}$$

within which the elements are arranged in the order of

$$\left(\sigma_{e1}^0\right)^2 V_1 > \left(\sigma_{e2}^0\right)^2 V_2 > \cdots > \left(\sigma_{e\beta}^0\right)^2 V_\beta \tag{7.2}$$

where $(\sigma_{ek}^0)^2 V_k$ is the denominator of equation (2.52) of m^0.

Since, the classical upper bound multiplier, m_u, is widely used in many applications, the concept of the reference volume is developed in the present work using the classical upper bound multiplier m_u instead of m^0. With reference to equation (2.46) of the classical upper bound multiplier m_u, it can be deduced that the elements are arranged in the order of

$$\sigma_{e1}^0 \varepsilon_{e1}^0 V_1 > \sigma_{e2}^0 \varepsilon_{e2}^0 V_2 > \cdots > \sigma_{e\beta}^0 \varepsilon_{e\beta}^0 V_\beta \tag{7.2a}$$

An iteration variable ζ is introduced in such a way that infinitesimal changes to the element elastic modulus of the various elements during the second and subsequent linear elastic FEA would induce a corresponding change $\Delta\zeta$. The magnitude of $\Delta\zeta$ would, of course, depend on the nature of the modulus-adjustments.

The value of m_u based on the total value would decrease with increasing ζ as illustrated in Fig. 7.1 while approaching the final solution. It can be assumed that, in every iteration, m_u is split into a constant value and a variable portion that vanishes with increasing ζ. Hence,

$$m_u = m_C + \Delta m \tag{7.3}$$

where m_C is the constant part.

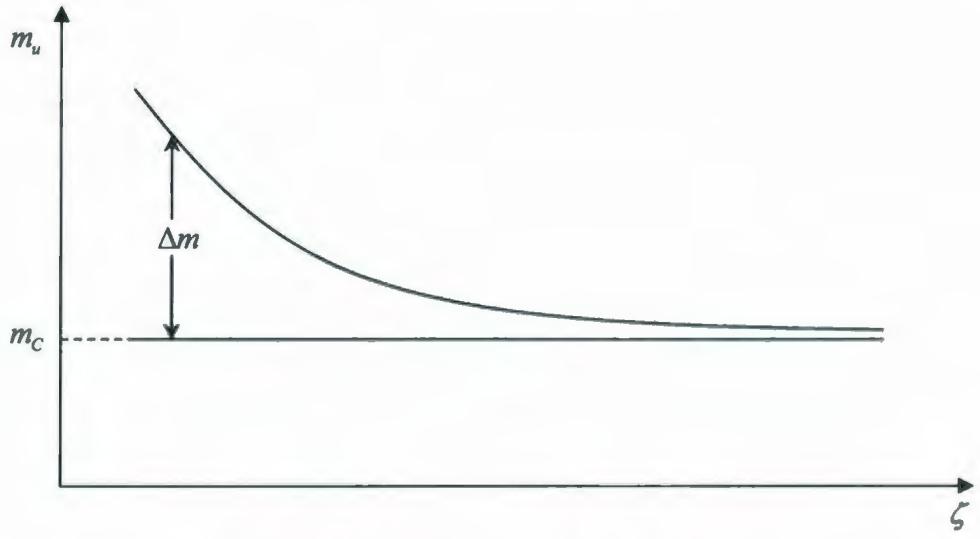


Fig. 7.1: Variation of m_u with Elastic Iterations

It was observed that the vanishing part represents the zone in the component that is not affected by the plastic deformation occurring in the highly stressed zone, and thus, in the

state of collapse, its stress level tends towards zero. When comparing the value of m_u to m_L, it is noticed that the former is calculated based upon the total volume and the latter is calculated based upon an infinitesimal volume with the highest stress value. The schematic of variation of m_u and m' with the iteration variable, ζ, is shown in Fig. 7.2. Therefore, for some volume V_R, where $\Delta V_1 < V_R \leq V_T$, the multiplier m_u would be invariant, i.e., $m_{u,1} = m_{u,2}$. Hence, equation (7.3) can be written as

$$m_u = \sigma_y \left(\frac{\int_{V_R} \varepsilon_e dV}{\int_{V_T} \varepsilon_e \sigma_e dV} + \frac{\int_{V_E} \varepsilon_e dV}{\int_{V_T} \varepsilon_e \sigma_e dV} \right) \tag{7.4}$$

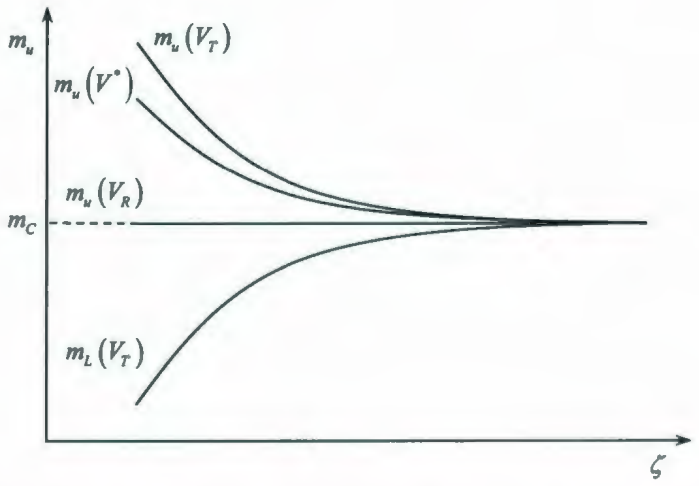


Fig. 7.2: Variation of m_u and m' with linear elastic iterations [2]

The elastic modulus adjustment procedures tend to make the stress within the component to a uniform value equivalent to the selected arbitrary stress. On the other hand, the strain distribution tends to high values in the reference volume zone and vanishes in the remaining volume. Hence, assuming the strain in the remaining volume to be zero, m_u is expressed as

$$m_u = \sigma_y \frac{\int_{V_R} \varepsilon_e dV}{\int_{V_R} \varepsilon_e \sigma_e dV} \tag{7.5}$$

Hence, in terms of finite elements, the multiplier will be

$$m_u = \sigma_y \frac{\sum_{k=1}^{N_R} \varepsilon_{ek} V_k}{\sum_{k=1}^{N_R} \varepsilon_{ek} \sigma_{ek} V_k} \tag{7.6}$$

where N_R is the number of elements in the reference volume.

To find the reference volume, the elements are sorted according to equation (7.2a) and m_u is calculated using equation (7.6) starting with $N_R = 1$ (single element) until $N_R = N_T$ (total volume). This is done for every iteration and m_u is plotted versus the considered volume as shown in Fig. 7.3. The intersecting point between iteration ζ and $\zeta + 1$ will be the solution for iteration $\zeta + 1$.

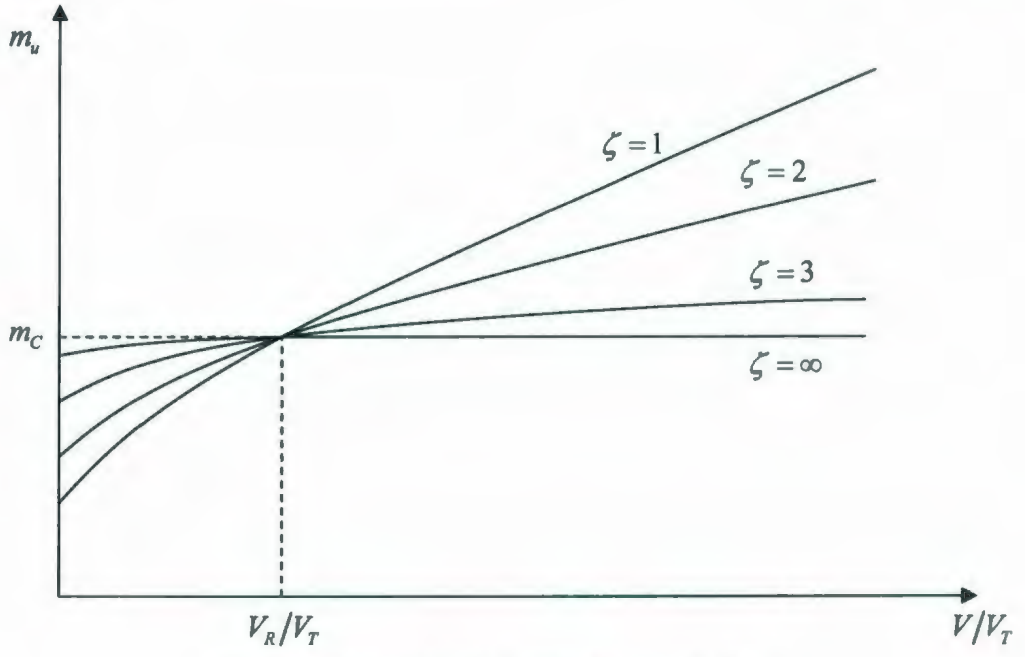


Fig. 7.3: Determination of Reference Volume

7.2 Applications

7.2.1 Indeterminate Beam

Being a complex problem with an analytical solution, the indeterminate beam is selected to illustrate the reference volume analysis. At the state of collapse, the plastic hinges are formed at locations A and B shown in Fig. 7.4. Hence, the regions of the hinges have stresses in the plastic range while the rest of the beam will have a near zero stress distribution.

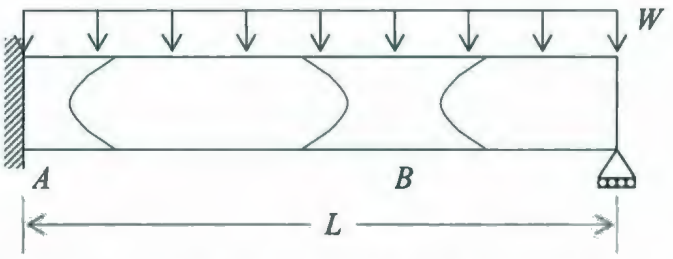


Fig. 7.4: Schematic diagram of the indeterminate beam.

The dimensions and the material used in this problem is the same as that used in section 4.3.2. Using the results of the stress redistribution analysis, the elements are sorted according to equation (7.2a). Hence, the upper bound multiplier is calculate based on several selected partial volumes V_P using the equation

$$m_u = \sigma_y \frac{\int_{V_P} \varepsilon_e dV}{\int_{V_P} \varepsilon_e \sigma_e dV} \tag{7.7}$$

for different ratios of V_P/V_T . A plot of the variation of the m_u with the considered for a number of iterations is shown in Fig. 7.5. It is noticed that the variation of the multiplier tends diminish beyond a certain volume at which all the curves intersect. Figure 7.6 shows a plot of the calculated values of the multiplier versus the iteration variable. It can

be noticed that the curve tends to flatten to a constant value as the volume decrease to a value just below 40% of the total volume. By comparing the first two iterations of the analysis, the reference volume was found to be that which is defined the stress greater that 7,415 psi illustrated in Fig. 7.7 which was found to be 34.5% of the total volume. Also, the value of the multiplier approach the classical lower bound as the partial volume tends to the minimum value which is the volume of the element having the highest stress.

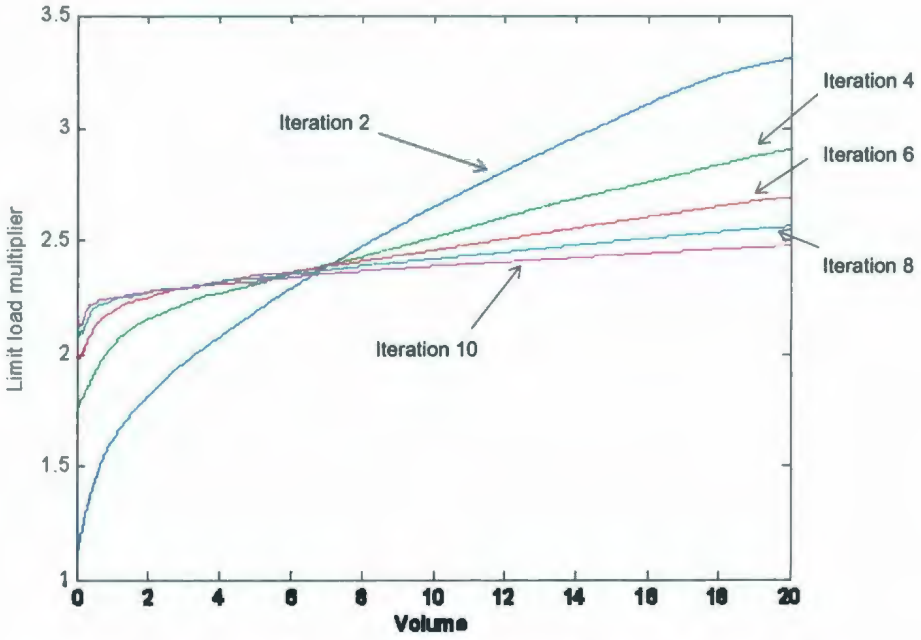


Fig. 7.5: Variation of the m_u with the volume at different iterations

Figure 7.8 shows the results of the analysis comparing the classical upper bound, classical lower bound, R-Node and reference volume solutions. It can be observed how the multiplier calculated using the reference volume converged as an upper bound solution faster than the other methods.

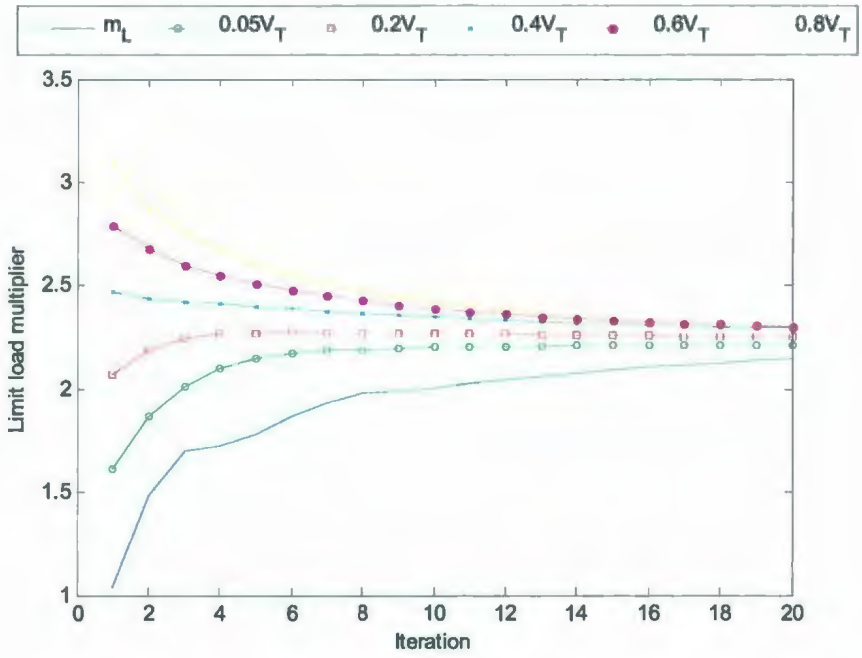


Fig. 7.6: The upper bound multiplier of the indeterminate beam calculated based on selected partial volumes.

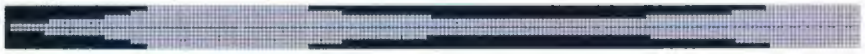


Fig. 7.7: Shaded diagram of the FE model of the indeterminate beam showing the reference volume (black area) and the remainder volume (gray area.)

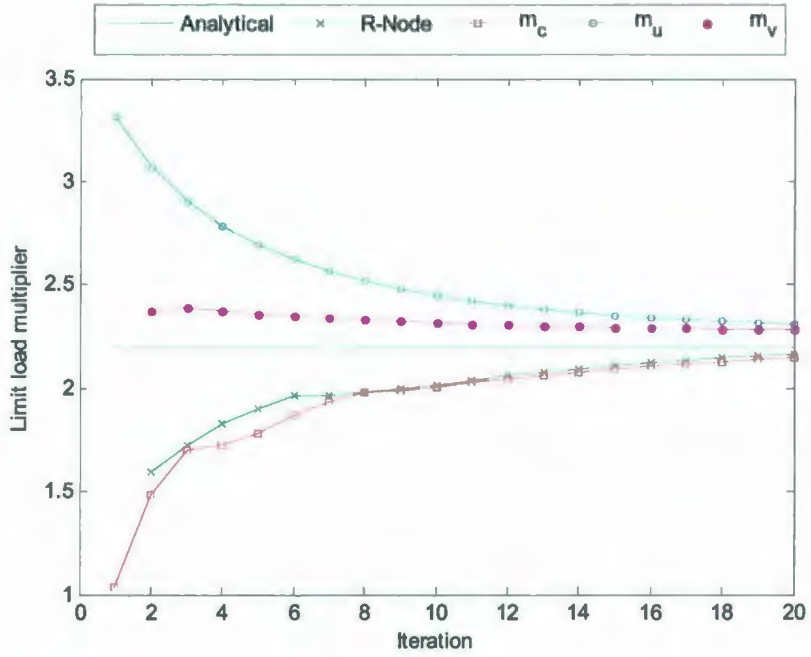


Fig. 7.8: Comparison of the convergence of the limit load using reference volume method with other limit load analysis methods.

7.2.2 Thick Plane-Strain Cylinder

The von Mises stress distribution within the walls of a thick cylinder has a high value at the inner radius and a low value at the outside radius. The redistribution analysis flattens the stress to a uniform value across the thickness. Due to the simplicity of the problem, the upper bound limit load calculated using finite element has a very high accuracy. On the other hand, the R-Node method shows a slightly slower convergence than the upper bound solution. This is observed by slowing down the redistribution using a value of $q = 0.1$ in the elastic modulus adjustment procedure. Figure 7.9 shows the meshing of the thick cylinder. Plane strain elements are used in the meshing. Figure 7.10 shows the results of the analysis of the limit load multiplier using the reference volume compared to the other methods for the thick cylinder described in chapter 4. It can be observed that the reference volume method gave a more accurate and faster solution compared to the R-Node and the classical lower bound methods. This is because the

reference volume is equal to the total volume of the pipe since collapse occurs when the whole thickness undergoes plastic deformation.

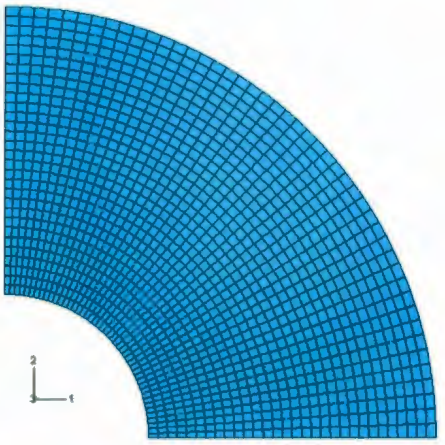


Fig. 7.9: Meshing of the plane-strain cylinder

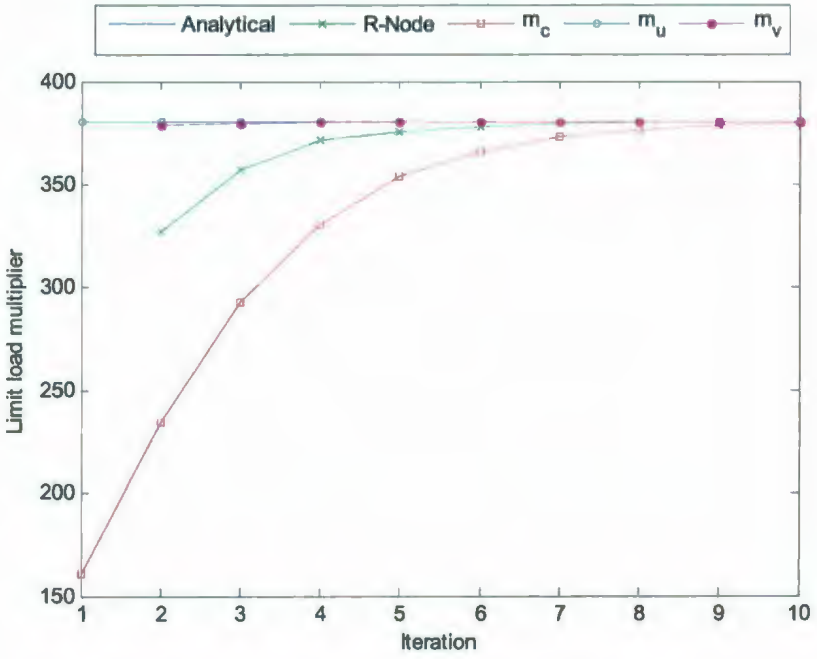


Fig. 7.10: Results of the analysis of the thick cylinder

7.2.3 Axisymmetric Pressure Vessel

The Pressure Vessel problem was illustrated earlier and it was shown to have three hinges form at the collapse state the locations of which are shown in Fig. 7.11. The value of q is selected to be 1 in order to achieve the stress distribution of the collapse state in a small number of iterations. Figure 7.12 shows the results of the analysis comparing the classical limit load multipliers with that calculated using the reference volume and the R-Node methods.

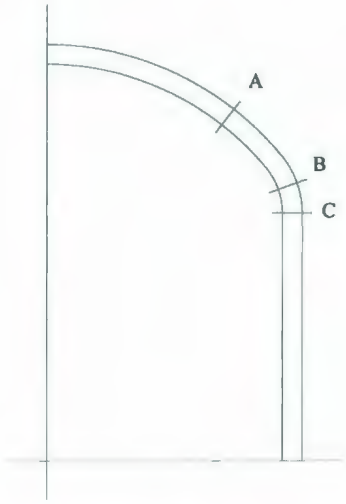


Fig. 7.11: Schematic of the geometry with the expected hinge locations.

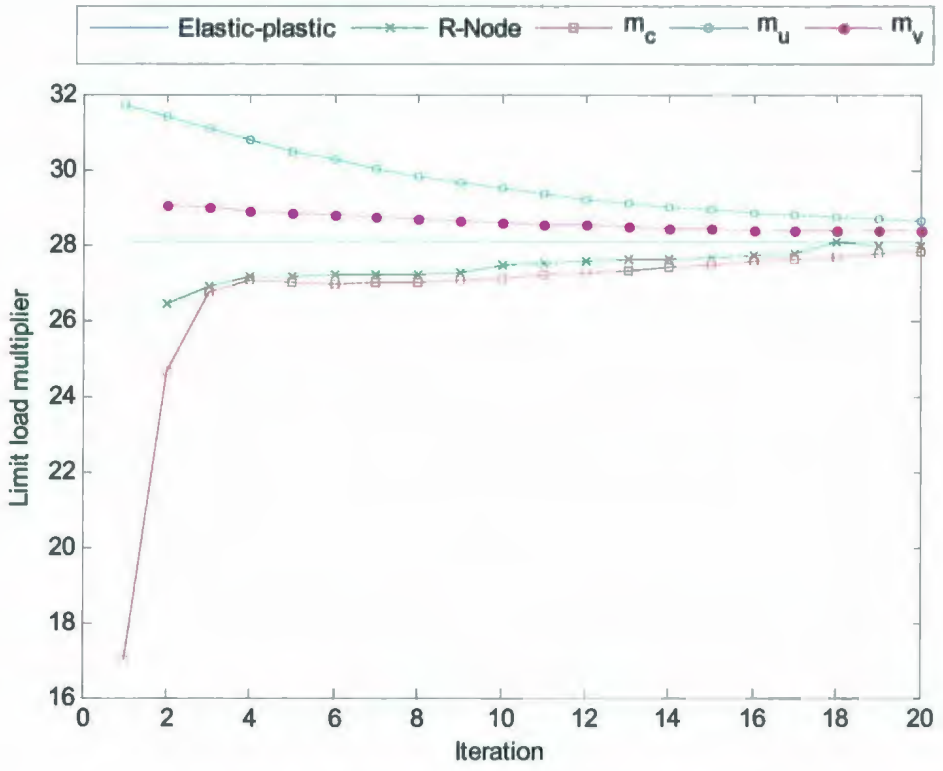


Fig. 7.12: Meshing of the plane-strain cylinder

7.2.4 Oblique Nozzle

A complex problem that has been considered in many researches is the analysis of an oblique nozzle attached to a pressure vessel. Sang et al [20] has conducted a set of experiments and elastic-plastic finite element analysis to monitor the complete stress distribution and find the limit load of the problem. The reference volume procedure is applied to the oblique nozzle presented by Sang et al [20] to compare its results. Also, the problem is solved to using other limit load analysis procedure to observe the convergence of the reference volume analysis. Figure 7.13 shows the geometry of the nozzle.

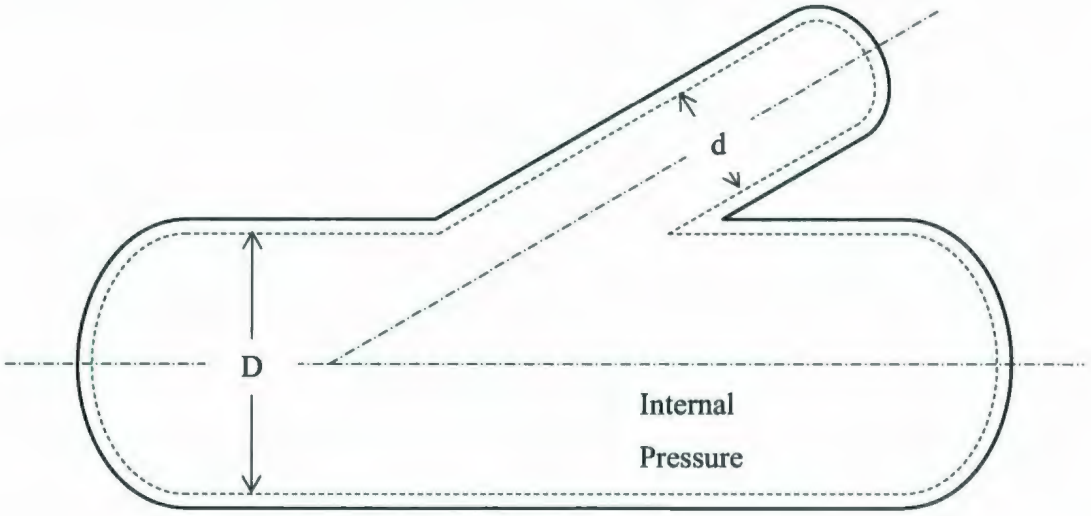


Fig. 7.13: Schematic diagram of the cross-section of an oblique nozzle used for experimental analysis by Sang et al. [20].

The inside diameter and thickness of the vessel are 600 mm and 6 mm, respectively. The outside diameter and thickness of the nozzle are 325 mm and 6 mm, respectively. The oblique angle is 30°. The length of the shell is 2400 mm and the length of the nozzle is 600 mm. The finite element model is prepared using 4-noded layered shell elements with 20 layers. Fig. 7.14 shows the meshing of the described geometry. Half of the geometry is considered making use of the symmetrical characteristic.

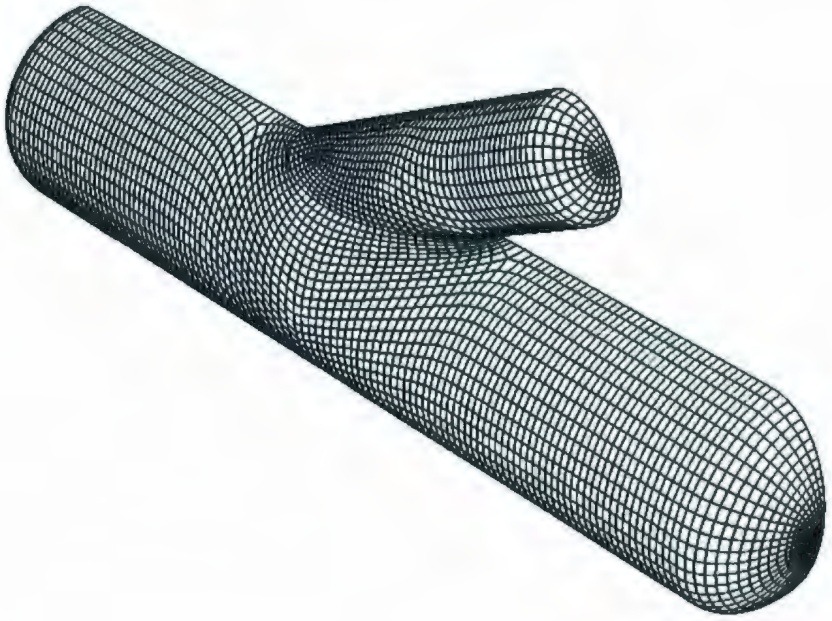


Fig. 7.14: Meshing of the oblique nozzle.

Treating each layer of the shell element as a separate element, the elements and their layers are sorted according to the centroid stress. Figure 7.15 shows a plot of the convergence of the calculated multipliers for various partial volumes. It can be noticed that the reference volume is confined within a very small part of the total volume, which is near 2-5% of the total volume. This is because the collapse of the whole structure will occur when just the joint of the nozzle to the vessel suffers complete plastic deformation. Fig. 7.16 shows that the calculated multiplier compares well with the experimental values estimated by Sang et al [20].

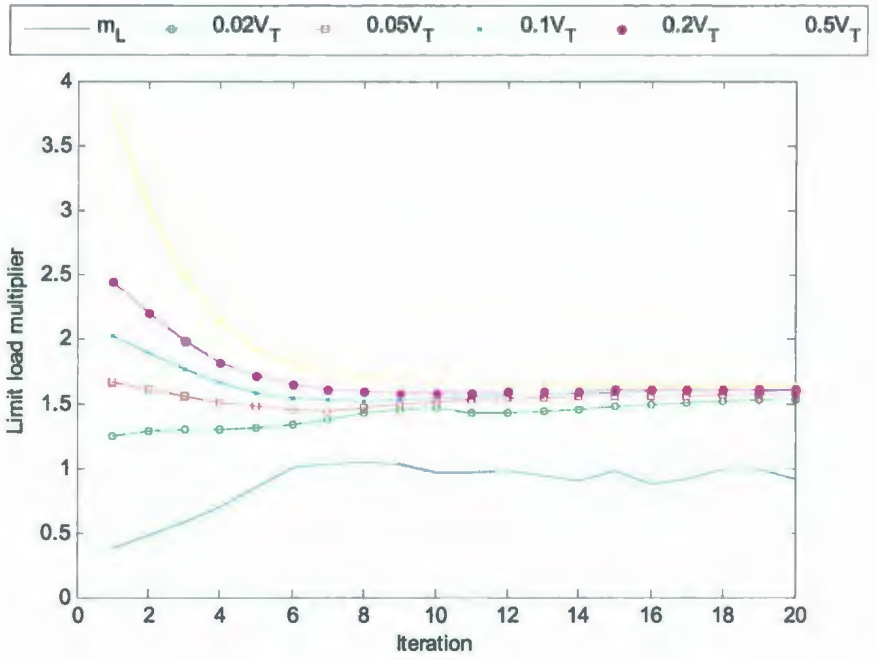


Fig. 7.15: Upper bound multiplier based on partial volumes.

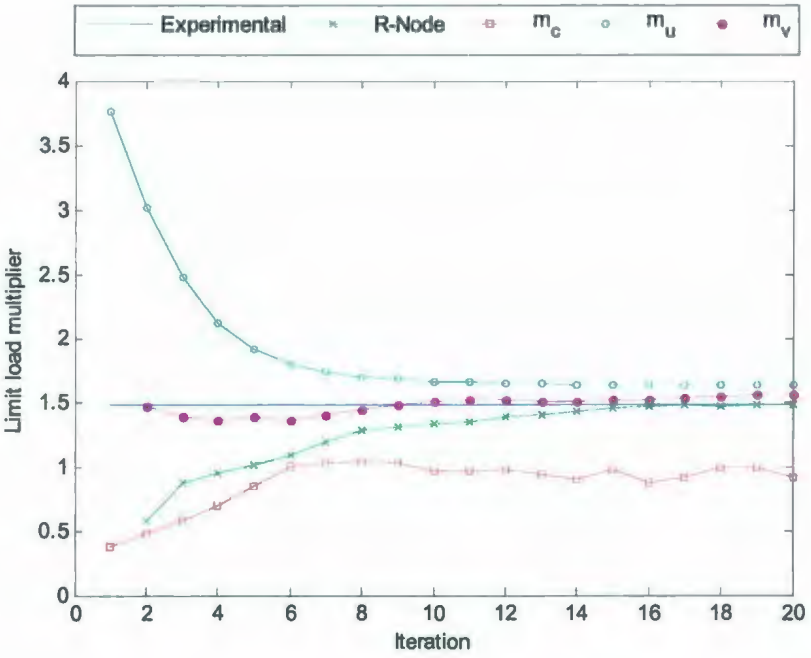


Fig. 7.16: Comparison of the limit load multipliers.

7.3 Summary

The limit load analysis using the finite element method requires several stress redistribution analysis, especially with highly complex problems. Using the fact that only part of the structure suffers full plastic deformation, the concept of the reference volume is introduced in which the upper-bound limit load multiplier is calculated based on part of the total volume. This is performed by comparing the variation of the upper-bound multiplier for two linear elastic analyses with increasing volume starting with the elements having the highest centroid stress. The results of the reference volume analysis method compares well with the analytical and experimental solutions of several problems with high accuracy and convergence rate.

CHAPTER 8

CONCLUSION

Robust methods for the analysis of limit loads are useful tools in the design and assessment of components subjected to mechanical loads. The R-Node method is developed to determine the lower bound limit value of single and multiple loads applied to a component. The applicability of the method in a wide range of problems and its use in the interpretation of FEA results is investigated.

The R-Node method is implemented in the ABAQUS program for several uses and applications. A code is developed for 2D plane, 3D shell and solid elements to perform the elastic modulus adjustment at the Gaussian integration points. A post-processing code is also developed to search for the maximum R-Node peak by comparing the iterations of elastic analysis. The method is used to find the limit load of an plane stress indeterminate beam, thick plane strain cylinder, axisymmetric vessel and an oblique nozzle modeled using shell elements. It was shown that the R-Node method gives a true lower-bound limit load.

An algorithm is formulated to find the limit value of a single load applied in a system of loads. In this algorithm, a number of limit load analysis iterations are performed with a systematic change in the load of interest until the limit value is reached. The algorithm was verified with simple problems and compared to analytical solution, and applied to the problem of bending of a pipe bend under internal pressure being a complex problem in the field of pressure component analysis. The problem is modeled using 3D layered-shell elements. The method showed to be effective and gave accurate solutions in few number of limit load analysis iterations.

The R-Node stresses, shown to be in direct equilibrium with the load, are shown to be an effective tool for stress classification to find the primary stresses component subjected to a single or multiple loads. By performing highly constrained stress redistribution, the R-Node stresses represent the load-controlled at the initial elastic phase. Hence, limiting the maximum R-Node stress to an allowable stress gives a design load. The procedure is applied to 2D plane geometries and 3D shell and solid geometries and the results compared well with that calculated using the ASME guidelines and the stress linearization tools in the ABAQUS commercial FEA program.

The concept of the reference volume is developed as a technique to find the limit load of component by considering the effect of only kinematically active part using the classical upper-bound multiplier. The procedure is shown in details and applied to the indeterminate beam and thick cylinder as plane problems, and the oblique nozzle as complex 3D geometry in the field of pressure vessel and piping. The concept of the reference volume is verified numerically by showing the change convergence rate of the limit load calculated using partial volumes. It is shown the method gives results with very high accuracy compared to the other methods using few numbers of elastic iterations.

In this thesis, several theories are used that have been developed in previous work. The original contributions made in the thesis are:

1. The implementation of the elastic modulus adjustment procedure for stress redistribution using the results at the Gaussian integration points. This is relevant to programming the R-Node analysis method and improving its results

2. The extension of the lower bound solution of the R-Node analysis for finding the limit value of a single load in a system of multiple loads.

3. The use of the R-Node method as a tool to find the primary stress that is equivalent to the membrane and bending primary stress in the ASME stress classification procedures.

4. Application of the reference volume using the classical upper-bound limit load theory as opposed to the use of the m^0 multiplier. This enables calculation of a more accurate limit load solution in a reduced computational time.

The R-Node method is applied and verified using the plane, axisymmetric, full integration 3D shell and solid elements in the present work. For future work, the method can be tested for other types of elements as well as those of higher orders. Further investigations are required to validate the method for orthotropic and anisotropic materials. Finally, further development of the algorithms of the subroutines presented in the present work can help in incorporating the R-Node in commercial codes.

REFERENCES

[1] Mura, T., W. H. Rimawi and S. L. Lee, "Extended Theorems of Limit Analysis"

[2] Mangalaramanan, S. P., and Seshadri, R., ''Limit Loads of Layered Beams and Layered Cylindrical Shells using Reduced Modulus Methods,'' ASME Pressure Vessels and Piping Conference, 1997, Orlando, FL.

[3] Seshadri, R. and H. Indermohan, 2004, "Lower bound limit load determination: The m_β multiplier method." Journal of Pressure Vessel Technology, 126, pp. 237-240.

[4] Seshadri, R., and Fernando, C. P. D., ''Limit Loads of Mechanical Components and Structures Using the GLOSS R-Node Method,'' ASME Journal of Pressure Vessel Technology, 114, 1992, pp. 201-208.

[5] Seshadri, R. and Mangalaramanan, S. P., 2001, "Simplified Inelastic Analysis and Code Perspectives," ASME PVP Conference, PVPD Tutorial.

[6] Mackenzie, D., and Boyle, J. T., 1993, ''A Method of Estimating Limit Loads by Iterative Elastic Analysis I: Simple Examples,'' Int. J. Pressure Vessels Piping, 53, pp. 77–95.

[7] Molski, K and G. Glinka, 1981, "Method of Elastic-Plastic Stress and Strain Calculation at a Notch Root." Materials Science and Engineering, 50, pp. 93-100.

[8] Adibi-Asl, R., Ihab F. Z. Fanous and R. Seshadri, 2006, "Elastic Modulus Adjustment Procedures – Improved Convergence Schemes," International Journal of Pressure Vessels and Piping, 83, 154-160.

[9] Pan, L. and R. Seshadri, 2002, "Limit Load Estimation Using Plastic Flow Parameter in Repeated Elastic Finite Element Analyses," Journal of Pressure Vessel Technology, **124**, 433-439.

[10] ASME Boiler & Pressure Vessel Code, Section III, ASME Press, 1996

[11] Sim, R. F., 1971, "Evaluation of Reference Stress Parameters for Structures Subject to Creep," Journal of Mechanical Engineering Science, **13**, 47-50

[12] Soderberg, C. R., "Interpretation of creep tests on tubes, Trans. ASME, Vol. 63, 1941, pp. 737-748

[13] Webster, G. A. and R. A. Ainsworth, *High Temperature component life assessment*, Chapman and Hall, London, 1994

[14] Ponter, A. R. S., Fuschi, P., and Engelhardt, M., 2000, ''Limit Analysis for a General Class of Yield Conditions,'' Eur. J. Mech. A/Solids, **19**, pp. 401–421.

[15] Ponter, A. R. S., and Engelhardt, M., 2000, ''Shakedown Limit for a General Yield Condition,'' Eur. J. Mech. A/Solids, **19**, pp. 423–445.

[16] Ponter, A. R. S., and Chen, H., 2001, ''A Programming Method for Limit Load and Shakedown Analysis of Structures,'' ASME PVP-Vol. 430, pp. 155–160.

[17] Seshadri, R., "The Generalized Local Stress Strain (GLOSS) Analysis—Theory and Applications," ASME J. Pressure Vessel Technol., 113, 1991, pp. 219–227.

[18] Seshadri, R. and D. L. Marriott, "On Relating the Reference Stress, Limit Load and the ASME Stress Classification Concepts," Int. J. Pres. Ves. & Piping, 56, 1993, pp. 387-408.

[19] Seshadri, R., 1997, "In search of redistribution nodes," International Journal of Pressure Vessel & Piping, **73**, 69-76.

[20] Sang, Z. F., Y. J. Lin, L. P. Xue and G. E. O, Widera, "Limit and Burst Pressures for a Cylindrical Vessel With a 30 deg – Lateral $(d/D \geq 0.5)$," Journal of Pressure Vessel Technology, 127, pp. 61-69

[21] Mendelson, Alexander, 1968, *Plasticity: Theory and Application*, The MacMillan Company, NY.

[22] Shalaby, M. A., and Younan, M. Y. A., ''Limit Loads for Pipe Elbows with Internal Pressure Under In-plane Closing Bending Moment,'' ASME Journal of Pressure Vessel Technology, 120, 1998, pp. 35–42.

[23] Mourad, Hashem M. and Maher Y. A. Younan, "Limit-Load Analysis of Pipe Bends Under Out-of-Plane Moment Loading and Internal Pressure," ASME Journal of Pressure Vessel Technology, 124, 2002, pp. 32-37.

[24] Chattopadhyay, J., D. K. Nathani, B. K. Dutta and H. S. Kushwaha, "Closed-Form Collapse Moment Equations of Elbows Under Combined Internal Pressure

and In-Plane Bending Moment," ASME Journal of Pressure Vessel Technology, 122, 2000, pp. 431-436

[25] Hibbitt, Karlsson, and Sorensen, ''ABAQUS/Standard User's Manual,'' 6.2, 2001.

[26] Kroenke, W. C., "Classification of Finite Element Stresses According to ASME Section III Stress Categories," ASME PVP Publications: Analysis and Computers, NY, 1974, pp. 107-140.

[27] Fanous, Ihab F. Z., R. Adibi-Asl and R. Seshadri, "Limit Load Analysis of Pipe Bend Using the R-Node Method," Journal of Pressure Vessel Technology, **127**, pp. 443-448.

[28] Marriott, D. L., ''Evaluation of Deformation or Load Control of Stresses under Inelastic Conditions using Elastic Finite Element Stress Analysis,'' ASME PVP, Vol. 136, 1988, pp. 3–9

[29] Hollinger, Greg and John Hechmer, 2000, "Three-Dimensional Stress Criteria – Summary of the PVRC Project," Journal of Pressure Vessel Technology, **122**, pp. 105-109.

[30] Pastor, T. P. and John Hechmer, 1997, "ASME Task Group Report on Primary Stress," Journal of Pressure Vessel Technology, **119**, pp. 61–67.

[31] Kraus, H. *Creep Analysis*, John Wiley, 1980

[32] Dhalla, A. K. (ed.), 1991, "Recommended Practice in Elevated Temperature Design: A Compendium of Breeder Reactor Experiences (1970-1987), II, Preliminary Design and Simplified Methods," Welding Research Council

[33] Calladine, C. R., *Engineering Plasticity*, Pergamon Press, Oxford, 1969

APPENDIX A

USER-DEFINED MATERIAL

A.1 Plane stress

```
      SUBROUTINE UMAT(STRESS,STATEV,DDSDDE,SSE,SPD,SCD,
     1 RPL,DDSDDT,DRPLDE,DRPLDT,
     2 STRAN,DSTRAN,TIME,DTIME,TEMP,DTEMP,PREDEF,DPRED,CMNAME,
     3 NDI,NSHR,NTENS,NSTATEV,PROPS,NPROPS,COORDS,DROT,PNEWDT,
     4 CELENT,DFGRD0,DFGRD1,NOEL,NPT,LAYER,KSPT,KSTEP,KINC)
C
      INCLUDE 'ABA_PARAM.INC'
C
      CHARACTER*8 CMNAME
      REAL K(2000, 8)
      DIMENSION STRESS(NTENS),STATEV(NSTATEV),
     1 DDSDDE(NTENS,NTENS),DDSDDT(NTENS),DRPLDE(NTENS),
     2 STRAN(NTENS),DSTRAN(NTENS),TIME(2),PREDEF(1),DPRED(1),
     3 PROPS(NPROPS),COORDS(3),DROT(3,3),DFGRD0(3,3),DFGRD1(3,3),
     4 S(2000, 8), CINC(2000, 8)
C
      PARAMETER (ONE=1.0D0, TWO=2.0D0)
      M=1
      I=0
      IF (KINC .LE. 1) THEN
            K(NOEL, NPT)=1.0
            CINC(NOEL, NPT)=1
      END IF
      NE=PROPS(3)
      NL=PROPS(4)
      IF (KINC .NE. CURRENTINC) THEN
            CURRENTINC=KINC
            ST1=0
            ST2=0
            IF (KINC .GT. 1) THEN
                  DO I=1,NE*NL
                              SS=(S(I,1)+S(I,2)+S(I,3)+S(I,4))/4
                              ST1=ST1+SS*SS
                              ST2=ST2+1
                  ENDDO
            END IF
            SREF=(ST1/ST2)**0.5
            print *,sreF,ST1,ST2
      END IF
      IF (KINC .GT. CINC(NOEL,NPT)) THEN
                  CINC(NOEL,NPT)=KINC
                  SR=SREF
                  S1=S(NOEL,NPT)
```

```
                K(NOEL,NPT)=K(NOEL,NPT)*((SR/S1)**0.75)
                END IF
        END IF
        IF (KINC .GT. 1) THEN
                E=PROPS(1)*K(NOEL,NPT)
        ELSE
                E=PROPS(1)
        END IF
        ANU=E/(1+PROPS(2))/(1-PROPS(2))
        AMU=E/2/(ONE+PROPS(2))
        DO I=1,NTENS
                DO J=1,NTENS
                        DDSDDE(I,J)=0.0D0
                ENDDO
        ENDDO
        DDSDDE(1,1)=ANU
        DDSDDE(2,2)=DDSDDE(1,1)
        DDSDDE(3,3)=AMU
        DDSDDE(4,4)=0
        DDSDDE(5,5)=0
        DDSDDE(6,6)=0
        DDSDDE(1,2)=PROPS(2)*ANU
        DDSDDE(1,3)=0
        DDSDDE(2,3)=0
        DDSDDE(2,1)=DDSDDE(1,2)
        DDSDDE(3,1)=0
        DDSDDE(3,2)=0
C
        DO I=1,NTENS
              STRESS(I)=0
              DO J=1,NTENS
                    STRESS(I)=STRESS(I)+DDSDDE(I,J)*(STRAN(J)+DSTRAN(J))
              ENDDO
        ENDDO
              SEQ=(STRESS(1)-STRESS(2))**2+(STRESS(2)-0)**2
              SEQ=SEQ+(0-STRESS(1))**2
              SEQ=SEQ+6*(STRESS(3)**2+0+0)
              SEQ=(0.5*SEQ)**0.5
              S(NOEL,NPT)=SEQ
        RETURN
        END
```

A.2 Plane strain

```
      SUBROUTINE UMAT(STRESS,STATEV,DDSDDE,SSE,SPD,SCD,
     1 RPL,DDSDDT,DRPLDE,DRPLDT,
     2 STRAN,DSTRAN,TIME,DTIME,TEMP,DTEMP,PREDEF,DPRED,CMNAME,
     3 NDI,NSHR,NTENS,NSTATEV,PROPS,NPROPS,COORDS,DROT,PNEWDT,
     4 CELENT,DFGRD0,DFGRD1,NOEL,NPT,LAYER,KSPT,KSTEP,KINC)
C
      INCLUDE 'ABA_PARAM.INC'
C
```

```
      CHARACTER*8 CMNAME
      REAL K(2000, 8), KK
      DIMENSION STRESS(NTENS),STATEV(NSTATEV),
     1 DDSDDE(NTENS,NTENS),DDSDDT(NTENS),DRPLDE(NTENS),
     2 STRAN(NTENS),DSTRAN(NTENS),TIME(2),PREDEF(1),DPRED(1),
     3 PROPS(NPROPS),COORDS(3),DROT(3,3),DFGRD0(3,3),DFGRD1(3,3),
     4 S(2000, 8), CINC(2000, 8)
C
      PARAMETER (ONE=1.0D0, TWO=2.0D0)
      M=1
      I=0
      IF (KINC .LE. 1) THEN
            K(NOEL, NPT)=1.0
            CINC(NOEL, NPT)=1
      END IF
      NE=PROPS(3)
      NL=PROPS(4)
      IF (KINC .NE. CURRENTINC) THEN
            CURRENTINC=KINC
            ST1=0
            ST2=0
            IF (KINC .GT. 1) THEN
                  DO I=1,NE*NL
                              SS=(S(I,1)+S(I,2)+S(I,3)+S(I,4))/4
                              KK=(K(I,1)+K(I,2)+K(I,3)+K(I,4))/4
                              ST1=ST1+SS*SS/KK
                              ST2=ST2+SS/KK
                  ENDDO
            END IF
            SREF=(ST1/ST2)**1
            print *,sreF,ST1,ST2
      END IF
      IF (KINC .GT. CINC(NOEL,NPT)) THEN
                  CINC(NOEL,NPT)=KINC
                  SR=SREF
                  S1=S(NOEL,NPT)
                  K(NOEL,NPT)=K(NOEL,NPT)*((SR/S1)**0.5)
      END IF
      IF (KINC .GT. 1) THEN
                  E=PROPS(1)*K(NOEL,NPT)
      ELSE
                  E=PROPS(1)
      END IF
      ANU=E/(1+PROPS(2))/(1-TWO*PROPS(2))
      AMU=E/2/(ONE+PROPS(2))
      DO I=1,NTENS
                  DO J=1,NTENS
                              DDSDDE(I,J)=0.0D0
                  ENDDO
      ENDDO
      DDSDDE(1,1)=(1-PROPS(2))*ANU
      DDSDDE(2,2)=DDSDDE(1,1)
      DDSDDE(3,3)=0
      DDSDDE(4,4)=AMU
      DDSDDE(5,5)=0
      DDSDDE(6,6)=0
      DDSDDE(1,2)=PROPS(2)*ANU
```

```
      DDSDDE(1,3)=PROPS(2)*ANU
      DDSDDE(2,3)=PROPS(2)*ANU
      DDSDDE(2,1)=PROPS(2)*ANU
      DDSDDE(3,1)=PROPS(2)*ANU
      DDSDDE(3,2)=PROPS(2)*ANU
C
      DO I=1,NTENS
            STRESS(I)=0
            DO J=1,NTENS
                  STRESS(I)=STRESS(I)+DDSDDE(I,J)*(STRAN(J)+DSTRAN(J))
            ENDDO
      ENDDO
            SEQ=(STRESS(1)-STRESS(2))**2+(STRESS(2)-STRESS(3))**2
            SEQ=SEQ+(STRESS(3)-STRESS(1))**2
            SEQ=SEQ+6*(STRESS(4)**2+0+0)
            SEQ=(0.5*SEQ)**0.5
            S(NOEL,NPT)=SEQ
      RETURN
      END
```

A.3 Axisymmetric

```
      SUBROUTINE UMAT(STRESS,STATEV,DDSDDE,SSE,SPD,SCD,
     1 RPL,DDSDDT,DRPLDE,DRPLDT,
     2 STRAN,DSTRAN,TIME,DTIME,TEMP,DTEMP,PREDEF,DPRED,CMNAME,
     3 NDI,NSHR,NTENS,NSTATEV,PROPS,NPROPS,COORDS,DROT,PNEWDT,
     4 CELENT,DFGRD0,DFGRD1,NOEL,NPT,LAYER,KSPT,KSTEP,KINC)
C
      INCLUDE 'ABA_PARAM.INC'
C
      CHARACTER*8 CMNAME
      REAL K(5000, 8),Q, KK, VK, SS
      DIMENSION STRESS(NTENS),STATEV(NSTATEV),
     1 DDSDDE(NTENS,NTENS),DDSDDT(NTENS),DRPLDE(NTENS),
     2 STRAN(NTENS),DSTRAN(NTENS),TIME(2),PREDEF(1),DPRED(1),
     3 PROPS(NPROPS),COORDS(3),DROT(3,3),DFGRD0(3,3),DFGRD1(3,3),
     4 S(5000, 8), CINC(5000, 8), VOL(5000,4)
C
      PARAMETER (ONE=1.0D0, TWO=2.0D0)
      M=1
      I=0
      IF (KINC .LE. 1) THEN
            K(NOEL, NPT)=1.0
            CINC(NOEL, NPT)=1
      END IF
      NE=PROPS(3)
      NL=PROPS(4)
      IF (KINC .NE. CURRENTINC) THEN
            CURRENTINC=KINC
            ST1=0
            ST2=0
            IF (KINC .GT. 1) THEN
```

```
                DO I=1,NE*NL
                        SS=(S(I,1)+S(I,2)+S(I,3)+S(I,4))/4

      VK=(VOL(I,1)+VOL(I,2)+VOL(I,3)+VOL(I,4))/4
                        KK=(K(I,1)+K(I,2)+K(I,3)+K(I,4))/4
                        ST1=ST1+SS*SS
                        ST2=ST2+1
                ENDDO
            END IF
            SREF=(ST1/ST2)**0.5
            print *,sreF,ST1,ST2
      END IF
      IF (KINC .GT. CINC(NOEL,NPT)) THEN
                CINC(NOEL,NPT)=KINC
                SR=SREF
                S1=S(NOEL,NPT)
C               K(NOEL,NPT)=K(NOEL,NPT)*((SR/S1)**0.75)
C               K(NOEL,NPT)=K(NOEL,NPT)*(2*SR*SR/(SR*SR+S1*S1))
                Q=1
                K(NOEL,NPT)=K(NOEL,NPT)*((SR/S1)**Q)
      END IF
      IF (KINC .GT. 1) THEN
                E=PROPS(1)*K(NOEL,NPT)
      ELSE
                E=PROPS(1)
      END IF
      ANU=E/(1+PROPS(2))/(1-2*PROPS(2))
      AMU=E/2/(ONE+PROPS(2))
      DO I=1,NTENS
                DO J=1,NTENS
                        DDSDDE(I,J)=0.0D0
                ENDDO
      ENDDO
      DDSDDE(1,1)=ANU*(1-PROPS(2))
      DDSDDE(2,2)=DDSDDE(1,1)
      DDSDDE(3,3)=DDSDDE(1,1)
      DDSDDE(4,4)=AMU
      DDSDDE(5,5)=0
      DDSDDE(6,6)=0
      DDSDDE(1,2)=PROPS(2)*ANU
      DDSDDE(1,3)=PROPS(2)*ANU
      DDSDDE(2,3)=PROPS(2)*ANU
      DDSDDE(2,1)=DDSDDE(1,2)
      DDSDDE(3,1)=DDSDDE(1,3)
      DDSDDE(3,2)=DDSDDE(2,3)
C
      DO I=1,NTENS
            STRESS(I)=0
            DO J=1,NTENS
                  STRESS(I)=STRESS(I)+DDSDDE(I,J)*(STRAN(J)+DSTRAN(J))
            ENDDO
      ENDDO
            SEQ=(STRESS(1)-STRESS(2))**2+(STRESS(2)-STRESS(3))**2
            SEQ=SEQ+(STRESS(3)-STRESS(1))**2
            SEQ=SEQ+6*(STRESS(4)**2+0+0)
            SEQ=(0.5*SEQ)**0.5
            S(NOEL,NPT)=SEQ
```

```
                VOL(NOEL,NPT)=COORDS(1)
      RETURN
      END
```

A.4 Layered shell

```
      SUBROUTINE UMAT(STRESS,STATEV,DDSDDE,SSE,SPD,SCD,
     1 RPL,DDSDDT,DRPLDE,DRPLDT,
     2 STRAN,DSTRAN,TIME,DTIME,TEMP,DTEMP,PREDEF,DPRED,CMNAME,
     3 NDI,NSHR,NTENS,NSTATEV,PROPS,NPROPS,COORDS,DROT,PNEWDT,
     4 CELENT,DFGRD0,DFGRD1,NOEL,NPT,LAYER,KSPT,KSTEP,KINC)
C
      INCLUDE 'ABA_PARAM.INC'
C
      CHARACTER*8 CMNAME
      REAL K(90000, 8)
      DIMENSION STRESS(NTENS),STATEV(NSTATEV),
     1 DDSDDE(NTENS,NTENS),DDSDDT(NTENS),DRPLDE(NTENS),
     2 STRAN(NTENS),DSTRAN(NTENS),TIME(2),PREDEF(1),DPRED(1),
     3 PROPS(NPROPS),COORDS(3),DROT(3,3),DFGRD0(3,3),DFGRD1(3,3),
     4 S(90000, 8), CINC(90000, 8)
C
      PARAMETER (ONE=1.0D0, TWO=2.0D0)
      M=1
      I=0
      NE=PROPS(3)
      NL=PROPS(4)
      NNOEL=NOEL+NE*3*(LAYER-1)+NE*(KSPT-1)
      IF (KINC .LE. 1) THEN
            K(NNOEL, NPT)=1.0
            CINC(NNOEL, NPT)=1
      END IF
      IF (KINC .NE. CURRENTINC) THEN
            CURRENTINC=KINC
            ST1=0
            ST2=0
            IF (KINC .GT. 1) THEN
                  DO I=1,NE
                        DO J=1,NL
                              SS=S(I+NE*3*(J-1)+NE,1)
                              SS=SS+S(I+NE*3*(J-1)+NE,2)
                              SS=SS+S(I+NE*3*(J-1)+NE,3)
                              SS=SS+S(I+NE*3*(J-1)+NE,4)
                              SS=SS/4
                              ST1=ST1+SS*SS
                              ST2=ST2+1
                        ENDDO
                  ENDDO
            END IF
            SREF=(ST1/ST2)**0.5
c           SREF=100000000
            print *,sreF,ST1,ST2
```

```
      END IF
      IF (KINC .GT. CINC(NNOEL,NPT)) THEN
                  CINC(NNOEL,NPT)=KINC
                  SR=SREF
                  S1=S(NNOEL,NPT)
                  K(NNOEL,NPT)=K(NNOEL,NPT)*((SR/S1)**0.5)
      END IF
      IF (KINC .GT. 1) THEN
                  E=PROPS(1)*K(NNOEL,NPT)
      ELSE
                  E=PROPS(1)
      END IF
      ANU=E/(1+PROPS(2))/(1-PROPS(2))
      AMU=E/2/(ONE+PROPS(2))
      DO I=1,NTENS
                  DO J=1,NTENS
                              DDSDDE(I,J)=0.0D0
                  ENDDO
      ENDDO
      DDSDDE(1,1)=ANU
      DDSDDE(2,2)=DDSDDE(1,1)
      DDSDDE(3,3)=AMU
      DDSDDE(4,4)=0
      DDSDDE(5,5)=0
      DDSDDE(6,6)=0
      DDSDDE(1,2)=PROPS(2)*ANU
      DDSDDE(1,3)=0
      DDSDDE(2,3)=0
      DDSDDE(2,1)=DDSDDE(1,2)
      DDSDDE(3,1)=0
      DDSDDE(3,2)=0
C
      DO I=1,NTENS
            STRESS(I)=0
            DO J=1,NTENS
                  STRESS(I)=STRESS(I)+DDSDDE(I,J)*(STRAN(J)+DSTRAN(J))
            ENDDO
      ENDDO

      SEQ=(STRESS(1)-STRESS(2))**2+(STRESS(2)-0)**2
      SEQ=SEQ+(0-STRESS(1))**2
      SEQ=SEQ+6*(STRESS(3)**2+0+0)
      SEQ=(0.5*SEQ)**0.5
      S(NNOEL,NPT)=SEQ

      RETURN
      END
```

A.5 Solid

```
      SUBROUTINE UMAT(STRESS,STATEV,DDSDDE,SSE,SPD,SCD,
     1 RPL,DDSDDT,DRPLDE,DRPLDT,
```

```
     2 STRAN,DSTRAN,TIME,DTIME,TEMP,DTEMP,PREDEF,DPRED,CMNAME,
     3 NDI,NSHR,NTENS,NSTATEV,PROPS,NPROPS,COORDS,DROT,PNEWDT,
     4 CELENT,DFGRD0,DFGRD1,NOEL,NPT,LAYER,KSPT,KSTEP,KINC)
C
      INCLUDE 'ABA_PARAM.INC'
C
      CHARACTER*8 CMNAME
      REAL K(200000, 8)
      DIMENSION STRESS(NTENS),STATEV(NSTATEV),
     1 DDSDDE(NTENS,NTENS),DDSDDT(NTENS),DRPLDE(NTENS),
     2 STRAN(NTENS),DSTRAN(NTENS),TIME(2),PREDEF(1),DPRED(1),
     3 PROPS(NPROPS),COORDS(3),DROT(3,3),DFGRD0(3,3),DFGRD1(3,3),
     4 S(200000, 8), CINC(200000, 8)
C
      PARAMETER (ONE=1.0D0, TWO=2.0D0)
      M=1
      I=0
      IF (KINC .LE. 1) THEN
                  K(NOEL, NPT)=1.0
                  CINC(NOEL, NPT)=1
      END IF
      NE=PROPS(3)
      NL=PROPS(4)
      IF (KINC .NE. CURRENTINC) THEN
            CURRENTINC=KINC
            ST1=0
            ST2=0
            IF (KINC .GT. 1) THEN
c                 DO I=1,NE
c                       SS=S(I,1)
c                       SS=SS+S(I,2)
c                       SS=SS+S(I,3)
c                       SS=SS+S(I,4)
c                       SS=SS/4
c                       ST1=ST1+SS*SS
c                       ST2=ST2+1
c                 ENDDO
c                 SREF=(ST1/ST2)**0.5
                  SREF=20000
            END IF
            print *,sreF,ST1,ST2
      END IF
      IF (KINC .GT. CINC(NOEL,NPT)) THEN
            S(NOEL,NPT)=SEQ
            CINC(NOEL,NPT)=KINC
            SR=SREF
            S1=S(NOEL,NPT)
            K(NOEL,NPT)=K(NOEL,NPT)*((SR/S1)**0.5)
      END IF

      IF (KINC .GT. 1) THEN
            E=PROPS(1)*K(NOEL,NPT)
      ELSE
            E=PROPS(1)
      END IF
      ANU=E/(1+PROPS(2))/(1-2*PROPS(2))
      AMU=E/2/(ONE+PROPS(2))
```

```
      DO I=1,NTENS
        DO J-1,NTENS
           DDSDDE(I,J)=0.0D0
        ENDDO
      ENDDO
      DDSDDE(1,1)=ANU*(1-PROPS(2))
      DDSDDE(2,2)=DDSDDE(1,1)
      DDSDDE(3,3)=DDSDDE(1,1)
      DDSDDE(4,4)=AMU
      DDSDDE(5,5)=AMU
      DDSDDE(6,6)=AMU
      DDSDDE(1,2)=PROPS(2)*ANU
      DDSDDE(1,3)=PROPS(2)*ANU
      DDSDDE(2,3)=PROPS(2)*ANU
      DDSDDE(2,1)=DDSDDE(1,2)
      DDSDDE(3,1)=DDSDDE(1,3)
      DDSDDE(3,2)=DDSDDE(2,3)
C
      DO I=1,NTENS
           STRESS(I)=0
           DO J=1,NTENS
                STRESS(I)=STRESS(I)+DDSDDE(I,J)*(STRAN(J)+DSTRAN(J))
           ENDDO
      ENDDO
      SEQ=(STRESS(1)-STRESS(2))**2+(STRESS(2)-STRESS(3))**2
      SEQ=SEQ+(STRESS(3)-STRESS(1))**2
      SEQ=SEQ+6*(STRESS(4)**2+STRESS(5)**2+STRESS(6)**2)
      SEQ=(0.5*SEQ)**0.5
      S(NOEL,NPT)=SEQ

      RETURN
      END
```

APPENDIX B

POST-PROCESSING SCRIPTS

B.1 Plane Stress and Plane Strain

ABAQUS Python Script

```
from abaqusConstants import *
from odbAccess import *
from string import *
odb=openOdb('beam1.odb')
sy=30000
f1=open('beam1_1.txt','w')
s1=[]
e1=[]
vke=[]
nlayers=1
nelem=len(odb.rootAssembly.instances['PART-1-1'].elements)
elems=odb.rootAssembly.instances['PART-1-1'].elements
nods=odb.rootAssembly.instances['PART-1-1'].nodes
for i in range(nelem):
      s1=s1+[0]
      n1=elems[i].connectivity[0]-1
      n2=elems[i].connectivity[1]-1
      n3=elems[i].connectivity[2]-1
      n4=elems[i].connectivity[3]-1
      x1=nods[n1].coordinates[0]
      y1=nods[n1].coordinates[1]
      z1=nods[n1].coordinates[2]
      x2=nods[n2].coordinates[0]
      y2=nods[n2].coordinates[1]
      z2=nods[n2].coordinates[2]
      x3=nods[n3].coordinates[0]
      y3=nods[n3].coordinates[1]
      z3=nods[n3].coordinates[2]
      x4=nods[n4].coordinates[0]
      y4=nods[n4].coordinates[1]
      z4=nods[n4].coordinates[2]
      l1=((x2-x1)**2+(y2-y1)**2+(z2-z1)**2)**0.5
      l2=((x3-x2)**2+(y3-y2)**2+(z3-z2)**2)**0.5
      l3=((x4-x3)**2+(y4-y3)**2+(z4-z3)**2)**0.5
      l4=((x1-x4)**2+(y1-y4)**2+(z1-z4)**2)**0.5
      l5=((x3-x1)**2+(y3-y1)**2+(z3-z1)**2)**0.5
      v=0.25*((l1+l2+l5)*(-l1+l2+l5)*(l1-l2+l5)*(l1+l2-l5))**0.5
      v=v+0.25*((l3+l4+l5)*(-l3+l4+l5)*(l3-l4+l5)*(l3+l4-l5))**0.5
      vke=vke+[v]
numiter=len(odb.steps['Step-1'].frames)-1
a=str(nelem)+",1,"+str(numiter)+",0,0,0,0\n"
```

```
f1.write(a)
sv1=odb.steps['Step-1'].frames[1].fieldOutputs['S'].values
for j in range(numiter):
      sv=odb.steps['Step-1'].frames[j+1].fieldOutputs['S'].values
      svc=odb.steps['Step-
1'].frames[j+1].fieldOutputs['S'].getSubset(position=CENTROID).values
      evc=odb.steps['Step-
1'].frames[j+1].fieldOutputs['E'].getSubset(position=CENTROID).values
      Sr=0
      s1_1=0
      s1_2=0
      s2_1=0
      s2_2=0
      maxs=0
      print("Iteration "+str(j+1))
      maxsrn=0
      for i in range(nelem):
            vk=vke[i]/nlayers
            for k in range(nlayers):
                  se=svc[nelem*k+i].mises
                  ee=evc[nelem*k+i].mises
                  a=str(se)+","+str(ee)+","+str(vk)+"\n"
                  f1.write(a)

      s1=[sv1[(nelem*k+i)*4+0].mises]+[sv1[(nelem*k+i)*4+1].mises]+[sv1
[(nelem*k+i)*4+2].mises]+[sv1[(nelem*k+i)*4+3].mises]

      s2=[sv[(nelem*k+i)*4+0].mises]+[sv[(nelem*k+i)*4+1].mises]+[sv[(n
elem*k+i)*4+2].mises]+[sv[(nelem*k+i)*4+3].mises]
                  for n1 in range(3):
                        s1_1=s1[n1]
                        s2_1=s2[n1]
                        if maxs<s2_1:
                              maxs=s2_1
                        for n2 in range(3-n1):
                              s1_2=s1[n2+n1+1]
                              s2_2=s2[n2+n1+1]
                              if s2_1<>s1_1:
                                    ds1=(s2_1-s1_1)/abs(s2_1-s1_1)
                              else:
                                    ds1=0
                              if s2_2<>s1_2:
                                    ds2=(s2_2-s1_2)/abs(s2_2-s1_2)
                              else:
                                    ds2=0
                              srn=0
                              a1=s1_2-s1_1
                              b1=s1_1
                              a2=s2_2-s2_1
                              b2=s2_1
                              if abs(a2-a1)>0.0001:
                                    xs=abs((b2-b1)/(a1-a2)-0.5)
                                    if xs<0.75:
                                          srn=(a1*b2-a2*b1)/(a1-a2)
                              else:
                                    if abs(b1-b2)<1:
                                          srn=s1_1
```

```
                                if srn>maxsrn:
                                      maxsrn=srn
                                      rne=i
                                      rnl=k
                                      rns=n1
                        if maxs<s2[13]:
                              maxs=s2[13]
            if j==0:
                  maxsrn=0
            a=str(maxsrn)+","+str(rne)+","+str(rnl)+","+str(rns)+","+str(maxs
)+"\n"
            f1.write(a)
f1.close
odb.close()
```

B.2 Axisymmetric

ABAQUS Python Script

```
from abaqusConstants import *
from odbAccess import *
from string import *
odb=openOdb('nozzle1.odb')
sy=30000
f1=open('nozzle1_1.txt','w')
s1=[]
e1=[]
vke=[]
nlayers=1
nelem=len(odb.rootAssembly.instances['PART-1-1'].elements)
m2=0
elems=odb.rootAssembly.instances['PART-1-1'].elements
nods=odb.rootAssembly.instances['PART-1-1'].nodes
for i in range(nelem):
      s1=s1+[0]
      n1=elems[i].connectivity[0]-1
      n2=elems[i].connectivity[1]-1
      n3=elems[i].connectivity[2]-1
      n4=elems[i].connectivity[3]-1
      x1=nods[n1].coordinates[0]
      y1=nods[n1].coordinates[1]
      z1=nods[n1].coordinates[2]
      x2=nods[n2].coordinates[0]
      y2=nods[n2].coordinates[1]
      z2=nods[n2].coordinates[2]
      x3=nods[n3].coordinates[0]
      y3=nods[n3].coordinates[1]
      z3=nods[n3].coordinates[2]
      x4=nods[n4].coordinates[0]
      y4=nods[n4].coordinates[1]
      z4=nods[n4].coordinates[2]
      l1=((x2-x1)**2+(y2-y1)**2+(z2-z1)**2)**0.5
      l2=((x3-x2)**2+(y3-y2)**2+(z3-z2)**2)**0.5
```

```
      l3=((x4-x3)**2+(y4-y3)**2+(z4-z3)**2)**0.5
      l4=((x1-x4)**2+(y1-y4)**2+(z1-z4)**2)**0.5
      l5=((x3-x1)**2+(y3-y1)**2+(z3-z1)**2)**0.5
      v=0.25*((l1+l2+l5)*(-l1+l2+l5)*(l1-l2+l5)*(l1+l2-l5))**0.5
      v=v+0.25*((l3+l4+l5)*(-l3+l4+l5)*(l3-l4+l5)*(l3+l4-l5))**0.5
      v=v*(x1+x2+x3+x4)/4
      vke=vke+[v]
numiter=len(odb.steps['Step-1'].frames)-1
a=str(nelem)+",1,"+str(numiter)+",0,0,0,0\n"
f1.write(a)
sv1=odb.steps['Step-1'].frames[1].fieldOutputs['S'].values
for j in range(numiter):
      sv=odb.steps['Step-1'].frames[j+1].fieldOutputs['S'].values
      svc=odb.steps['Step-
1'].frames[j+1].fieldOutputs['S'].getSubset(position=CENTROID).values
      evc=odb.steps['Step-
1'].frames[j+1].fieldOutputs['E'].getSubset(position=CENTROID).values
      Sr=0
      s1_1=0
      s1_2=0
      s2_1=0
      s2_2=0
      maxs=0
      print("Iteration "+str(j+1))
      maxsrn=0
      for i in range(nelem):
            vk=vke[i]/nlayers
            for k in range(nlayers):
                  se=svc[nelem*k+i].mises
                  ee=evc[nelem*k+i].mises
                  a=str(se)+","+str(ee)+","+str(vk)+"\n"
                  f1.write(a)

      s1=[sv1[(nelem*k+i)*4+0].mises]+[sv1[(nelem*k+i)*4+1].mises]+[sv1
[(nelem*k+i)*4+2].mises]+[sv1[(nelem*k+i)*4+3].mises]

      s2=[sv[(nelem*k+i)*4+0].mises]+[sv[(nelem*k+i)*4+1].mises]+[sv[(n
elem*k+i)*4+2].mises]+[sv[(nelem*k+i)*4+3].mises]
                  for n1 in range(3):
                        s1_1=s1[n1]
                        s2_1=s2[n1]
                        if maxs<s2_1:
                              maxs=s2_1
                        for n2 in range(3-n1):
                              s1_2=s1[n2+n1+1]
                              s2_2=s2[n2+n1+1]
                              if s2_1<>s1_1:
                                    ds1=(s2_1-s1_1)/abs(s2_1-s1_1)
                              else:
                                    ds1=0
                              if s2_2<>s1_2:
                                    ds2=(s2_2-s1_2)/abs(s2_2-s1_2)
                              else:
                                    ds2=0
                              srn=0
                              a1=s1_2-s1_1
                              b1=s1_1
```

```
                                    a2=s2_2-s2_1
                                    b2=s2_1
                                    if abs(a2-a1)>0.0001:
                                          xs=abs((b2-b1)/(a1-a2)-0.5)
                                          if xs<0.75:
                                                srn=(a1*b2-a2*b1)/(a1-a2)
                                    else:
                                          if abs(b1-b2)<1:
                                                srn=s1_1
                                    if srn>maxsrn:
                                          maxsrn=srn
                                          rne=i
                                          rnl=k
                                          rns=n1
                        if maxs<s2[13]:
                              maxs=s2[13]
            if j==0:
                  maxsrn=0
            a=str(maxsrn)+","+str(rne)+","+str(rnl)+","+str(rns)+","+str(maxs
)+"\n"
            f1.write(a)
f1.close
odb.close()
```

MatLab Graphing

```
st = importdata('beam1_1.txt',',');
nelem=st(1,1);
niter=st(1,2);
nlayer=st(1,2);
niter=st(1,3);
sy=30000;
for i=1:1:niter;
    mu_1=0;
    mu_2=0;
    m(i,1)=2.2;
    m(i,2)=sy/st(1+(nelem+1)*i,1);
    m(i,3)=sy/st(1+(nelem+1)*i,5);
    step(:,:,i)=sortrows(st(2+(nelem+1)*(i-1):2+(nelem+1)*(i-1)+nelem-
1,1:7));
    vt=sum(step(:,3,i));
    vti=0;
    for k=nelem:-1:1
        s=step(k,1,i);
        v=step(k,3,i);
        step(k,2,i)=s/step(k,2,i);
        e=step(k,2,i);
        vti=vti+v;
        step(k,4,i)=s*v/e;
        step(k,5,i)=s*s*v/e;
        mu_1=mu_1+step(k,4,i);
        mu_2=mu_2+step(k,5,i);
        step(k,6,i)=vti;
        step(k,7,i)=sy*((mu_1/mu_2)^1);
    end;
    m(i,4)=step(1,7,i);
```

```
    if i>1
        n1 = 1;
        n2 = 1;
        dif1=sign(step(n2,7,i)-step(n1,7,1));
        v1=vt;
        v2=vt;
        mv1=0;
        while mv1==0
            dif2=sign(step(n2,7,i)-step(n1,7,1));
            if dif1==dif2
                if v2<v1
                    while v2<v1
                        v1=v1-step(n1,3,1);
                        n1=n1+1;
                    end;
                else
                    while v1<=v2
                        v2=v2-step(n2,3,i);
                        n2=n2+1;
                    end;
                end;
            else
                mv1=step(n2,7,i);
            end;
        end;
    else
        mv1=Inf;
    end;
    m(i,5)=mv1;
end;


figure1 = figure;
axes1 = axes('Parent',figure1);
xlabel(axes1,'Iteration');
ylabel(axes1,'Limit load multiplier');
box(axes1,'on');
hold(axes1,'all');
plot1 = plot(m);
set(plot1(2),'Marker','x');
set(plot1(3),...
  'Marker','square',...
  'MarkerSize',3);
set(plot1(4),...
  'Marker','o',...
  'MarkerSize',3);
set(plot1(5),'Marker','.');

%% Create legend
legend1 = legend(...
  axes1,{'Analytical','R-Node','m_c','m_u','m_v'},...
  'Location','NorthOutside',...
  'Orientation','horizontal');
```

B.3 Layered Shell

ABAQUS Python Script

```
from abaqusConstants import *
from odbAccess import *
from string import *
odb=openOdb('nozzle3.odb')
sy=30000
f1=open('nozzle3_1.txt','w')
s1=[]
e1=[]
vke=[]
nlayers=6
nelem=len(odb.rootAssembly.instances['PART-7-1'].elements)
m2=0
elems=odb.rootAssembly.instances['PART-7-1'].elements
nods=odb.rootAssembly.instances['PART-7-1'].nodes
for i in range(nelem):
      s1=s1+[0]
      n1=elems[i].connectivity[0]-1
      n2=elems[i].connectivity[1]-1
      n3=elems[i].connectivity[2]-1
      n4=elems[i].connectivity[3]-1
      x1=nods[n1].coordinates[0]
      y1=nods[n1].coordinates[1]
      z1=nods[n1].coordinates[2]
      x2=nods[n2].coordinates[0]
      y2=nods[n2].coordinates[1]
      z2=nods[n2].coordinates[2]
      x3=nods[n3].coordinates[0]
      y3=nods[n3].coordinates[1]
      z3=nods[n3].coordinates[2]
      x4=nods[n4].coordinates[0]
      y4=nods[n4].coordinates[1]
      z4=nods[n4].coordinates[2]
      l1=((x2-x1)**2+(y2-y1)**2+(z2-z1)**2)**0.5
      l2=((x3-x2)**2+(y3-y2)**2+(z3-z2)**2)**0.5
      l3=((x4-x3)**2+(y4-y3)**2+(z4-z3)**2)**0.5
      l4=((x1-x4)**2+(y1-y4)**2+(z1-z4)**2)**0.5
      l5=((x3-x1)**2+(y3-y1)**2+(z3-z1)**2)**0.5
      v=0.25*((l1+l2+l5)*(-l1+l2+l5)*(l1-l2+l5)*(l1+l2-l5))**0.5
      v=v+0.25*((l3+l4+l5)*(-l3+l4+l5)*(l3-l4+l5)*(l3+l4-l5))**0.5
      v=v*0.006
      vke=vke+[v]
numiter=len(odb.steps['Step-1'].frames)-1
a=str(nelem)+","+str(nlayers)+","+str(numiter)+",0,0,0,0\n"
f1.write(a)
sv1=odb.steps['Step-1'].frames[1].fieldOutputs['S'].values
for j in range(numiter):
      sv=odb.steps['Step-1'].frames[j+1].fieldOutputs['S'].values
      svc=odb.steps['Step-
1'].frames[j+1].fieldOutputs['S'].getSubset(position=CENTROID).values
      evc=odb.steps['Step-
1'].frames[j+1].fieldOutputs['E'].getSubset(position=CENTROID).values
```

```
Sr=0
s1_1=0
s1_2=0
s2_1=0
s2_2=0
maxs=0
print("Iteration "+str(j+1))
maxsrn=0
for i in range(nelem):
      vk=vke[i]/nlayers
      for k in range(nlayers):
            se=svc[nelem*k*3+nelem+i].mises
            ee=evc[nelem*k*3+nelem+i].mises
            a=str(se)+","+str(ee)+","+str(vk*(1+(2.5-k)/6))+"\n"
            f1.write(a)
            for sl in range(4):
                  s1=[sv1[(nelem*4*k*3+nelem*4*0+i*4+sl)].mises]

s1=s1+[sv1[(nelem*4*k*3+nelem*4*1+i*4+sl)].mises]

s1=s1+[sv1[(nelem*4*k*3+nelem*4*2+i*4+sl)].mises]
                  s2=[sv[(nelem*4*k*3+nelem*4*0+i*4+sl)].mises]

s2=s2+[sv[(nelem*4*k*3+nelem*4*1+i*4+sl)].mises]

s2=s2+[sv[(nelem*4*k*3+nelem*4*2+i*4+sl)].mises]
                  for n1 in range(2):
                        s1_1=s1[n1]
                        s2_1=s2[n1]
                        if maxs<s2_1:
                              maxs=s2_1
                        n2=n1+1
                        s1_2=s1[n1+1]
                        s2_2=s2[n1+1]
                        if s2_1<>s1_1:
                              ds1=(s2_1-s1_1)/abs(s2_1-s1_1)
                        else:
                              ds1=0
                        if s2_2<>s1_2:
                              ds2=(s2_2-s1_2)/abs(s2_2-s1_2)
                        else:
                              ds2=0
                        srn=0
                        a1=s1_2-s1_1
                        b1=s1_1
                        a2=s2_2-s2_1
                        b2=s2_1
                        if abs(a2-a1)>0.0001:
                              xs=abs((b2-b1)/(a1-a2)-0.5)
                              if xs<=0.5:
                                    srn=(a1*b2-a2*b1)/(a1-a2)
                        else:
                              if abs(b1-b2)<0.0001:
                                    srn=s1_1
                        if srn>maxsrn:
                              maxsrn=srn
                              rne=i+1
```

```
                                    rn1=k+1
                                    rns=s1+1
                              if maxs<s2[12]:
                                    maxs=s2[12]
                        for s1 in range(3):
                              s1-[sv1[(nelem*4*k*3+nelem*4*s1+i*4+0)].mises]

      s1-s1+[sv1[(nelem*4*k*3+nelem*4*s1+i*4+1)].mises]

      s1=s1+[sv1[(nelem*4*k*3+nelem*4*s1+i*4+2)].mises]

      s1=s1+[sv1[(nelem*4*k*3+nelem*4*s1+i*4+3)].mises]
                              s2-[sv[(nelem*4*k*3+nelem*4*s1+i*4+0)].mises]

      s2=s2+[sv[(nelem*4*k*3+nelem*4*s1+i*4+1)].mises]

      s2=s2+[sv[(nelem*4*k*3+nelem*4*s1+i*4+2)].mises]

      s2=s2+[sv[(nelem*4*k*3+nelem*4*s1+i*4+3)].mises]
                              for n1 in range(4):
                                    s1_1=s1[n1]
                                    s2_1=s2[n1]
                                    if n1<3:
                                          n2=n1+1
                                    else:
                                          n2=0
                                    s1_2=s1[n2]
                                    s2_2=s2[n2]
                                    if s2_1<>s1_1:
                                          ds1=(s2_1-s1_1)/abs(s2_1-s1_1)
                                    else:
                                          ds1=0
                                    if s2_2<>s1_2:
                                          ds2=(s2_2-s1_2)/abs(s2_2-s1_2)
                                    else:
                                          ds2=0
                                    srn=0
                                    a1=s1_2-s1_1
                                    b1=s1_1
                                    a2=s2_2-s2_1
                                    b2=s2_1
                                    if abs(a2-a1)>0.0001:
                                          xs=abs((b2-b1)/(a1-a2)-0.5)
                                          if xs<=0.75:
                                                srn=(a1*b2-a2*b1)/(a1-a2)
                                    else:
                                          if abs(b1-b2)<0.0001:
                                                srn=s1_1
                                    if srn>maxsrn:
                                          maxsrn=srn
                                          rne=i+1
                                          rn1=k+1
                                          rns=s1+1
      if j==0:
            maxsrn=0
      a=str(maxsrn)+","+str(rne)+","+str(rn1)+","+str(rns)+","+str(maxs
)+"\n"
```

```
      f1.write(a)
f1.close
odb.close()
```

MatLab Graphing

```
st = importdata('nozzle3_1.txt',',');
nelem=st(1,1);
nlayer=st(1,2);
niter=st(1,3);
sy=339400000;
for i=1:1:niter;
    mu_1=0;
    mu_2=0;
    m(i,1)=1.48;
    m(i,2)=sy/st(1+(nelem*nlayer+1)*i,1);
    m(i,3)=sy/st(1+(nelem*nlayer+1)*i,5);
    step(:,:,i)=sortrows(st(2+(nelem*nlayer+1)*(i-
1):2+(nelem*nlayer+1)*(i-1)+nelem*nlayer-1,1:7));
    vt=sum(step(:,3,i));
    vti=0;
    for k=nelem*nlayer:-1:1
        s=step(k,1,i);
        v=step(k,3,i);
        step(k,2,i)=s/step(k,2,i);
        e=step(k,2,i);
        vti=vti+v;
        step(k,4,i)=s*v/e;
        step(k,5,i)=s*s*v/e;
        mu_1=mu_1+step(k,4,i);
        mu_2=mu_2+step(k,5,i);
        step(k,6,i)=vti;
        step(k,7,i)=sy*((mu_1/mu_2)^1);
    end;
    m(i,4)=step(1,7,i);
    if i>1
        n1=1;
        n2=1;
        dif1=sign(step(n2,7,i)-step(n1,7,1));
        v1=vt;
        v2=vt;
        mv1=0;
        while mv1==0
            dif2=sign(step(n2,7,i)-step(n1,7,1));
            if dif1==dif2
                if v2<v1
                    while v2<v1
                        v1=v1-step(n1,3,1);
                        n1=n1+1;
                    end;
                else
                    while v1<=v2
                        v2=v2-step(n2,3,i);
                        n2=n2+1;
                    end;
                end;
```

```
            else
                mv1=step(n2,7,i);
            end;
        end;
        m(i,5)=mv1;
    else
        mv1=Inf;
    end;
    m(i,5)=mv1;
end;


figure1 = figure;
axes1 = axes('Parent',figure1);
xlabel(axes1,'Iteration');
ylabel(axes1,'Limit load multiplier');
box(axes1,'on');
hold(axes1,'all');
plot1 = plot(m);
set(plot1(2),'Marker','x');
set(plot1(3),...
  'Marker','square',...
  'MarkerSize',3);
set(plot1(4),...
  'Marker','o',...
  'MarkerSize',3);
set(plot1(5),'Marker','.');

%% Create legend
legend1 = legend(...
  axes1,{'Experimental','R-Node','m_c','m_u','m_v'},...
  'Location','NorthOutside',...
  'Orientation','horizontal');
```

B.4 Solid

```
from abaqusConstants import *
from odbAccess import *
from string import *
odb=openOdb('nozzle4.odb')
sy=30000
f1=open('nozzle4_1.txt','w')
s1=[]
e1=[]
vke=[]
nlayers=1
nelem=len(odb.rootAssembly.instances['PART-7-1'].elements)
elems=odb.rootAssembly.instances['PART-7-1'].elements
nods=odb.rootAssembly.instances['PART-7-1'].nodes
for i in range(nelem):
      s1=s1+[0]
      vke=vke+[1]
```

```
numiter=len(odb.steps['Step-1'].frames)-1
a=str(nelem)+",1,"+str(numiter)+",0,0,0,0\n"
f1.write(a)
sv1=odb.steps['Step-1'].frames[1].fieldOutputs['S'].values
for j in range(numiter):
      sv=odb.steps['Step-1'].frames[j+1].fieldOutputs['S'].values
      svc=odb.steps['Step-
1'].frames[j+1].fieldOutputs['S'].getSubset(position=CENTROID).values
      Sr=0
      s1_1=0
      s1_2=0
      s2_1=0
      s2_2-0
      maxs=0
      print("Iteration "+str(j+1))
      maxsrn=0
      for i in range(nelem):
            vk=vke[i]/nlayers
            for k in range(nlayers):
                  se=svc[nelem*k+i].mises

      s1=[sv1[(nelem*k+i)*4+0].mises]+[sv1[(nelem*k+i)*4+1].mises]+[sv1
[(nelem*k+i)*4+2].mises]+[sv1[(nelem*k+i)*4+3].mises]

      s2=[sv[(nelem*k+i)*4+0].mises]+[sv[(nelem*k+i)*4+1].mises]+[sv[(n
elem*k+i)*4+2].mises]+[sv[(nelem*k+i)*4+3].mises]
                  if i==0:
                        se=(s2[0]+s2[1]+s2[2]+s2[3])/4

      print(str(s2[0]),str(s2[1]),str(s2[2]),str(s2[3]))
                  ee=0
                  a=str(se)+","+str(ee)+","+str(vk)+"\n"
                  f1.write(a)
                  for n1 in range(3):
                        s1_1=s1[n1]
                        s2_1=s2[n1]
                        for n2 in range(3-n1):
                              s1_2=s1[n2+n1+1]
                              s2_2=s2[n2+n1+1]
                              srn-0
                              a1=s1_2-s1_1
                              b1=s1_1
                              a2=s2_2-s2_1
                              b2=s2_1
                              if abs(a2-a1)>0.0001:
                                    xs=abs((b2-b1)/(a1-a2)-0.5)
                                    if xs<0.75:
                                          srn=(a1*b2-a2*b1)/(a1-a2)
                              else:
                                    if abs(b1-b2)<1:
                                          srn=s1_1
                              if srn>maxsrn:
                                    maxsrn=srn
                                    rne=i+1
                                    rnl=k+1
                                    rns=n1
      a=str(maxsrn)+","+str(rne)+","+str(rnl)+","+str(rns)+"\n"
```

```
      f1.write(a)
f1.close
odb.close()
```