

UNIVERSITY OF CAPE TOWN

**FINITE ELEMENT ALGORITHMS FOR
THE STATIC AND DYNAMIC ANALYSIS OF
TIME-DEPENDENT AND TIME-INDEPENDENT PLASTIC BODIES**

A DISSERTATION SUBMITTED TO
THE FACULTY OF ENGINEERING
IN CANDIDACY FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF MECHANICAL ENGINEERING

BY

MODIFY A E KAUNDA

CAPE TOWN, SOUTH AFRICA

JUNE, 1994

The University of Cape Town has been given
the right to reproduce this thesis in whole
or in part. Copyright is held by the author.

The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

University of Cape Town

ABSTRACT

Continuum and finite element formulations of the static and dynamic *initial-boundary-value* evolution (elastoplastic) problems are considered in terms of both the classical and internal variable frameworks. The latter framework is employed to develop algorithms in the form of convex mathematical programming and Newton-Raphson schemes. This latter scheme is shown to be linked to the former in the sense that it expresses the conditions under which the convex non-linear function can be minimised.

A Taylor series expansion in time and space is extensively employed to derive integration schemes which include the generalised trapezoidal rule and a generalised Newton-Raphson scheme. This approach provides theoretical foundations for the generalised trapezoidal rule and the generalised Newton-Raphson scheme that have some geometrical insights as well as an interpretation in terms of finite differences and calculus. Conventionally, one way of interpreting the generalised trapezoidal rule is that it uses a weighted average of values (such as velocity or acceleration) at the two ends of the time interval.

In this dissertation, the generalised trapezoidal scheme is shown to be a special case of the *forward-backward* difference scheme for solving first order differential equations. It includes the Euler forward and backward difference schemes as special cases when the integration parameters are set to $\alpha = 0$ and $\alpha = 1$, respectively. For the solution of second order differential equations, two schemes are developed, termed the *FB1* and *FB2* schemes. The generalised Newton-Raphson scheme includes the conventional Newton-Raphson scheme as a special case when its integration scalars are set to $\beta = 0$ for an *implicit* version or $\beta = 1$ for an *explicit* version.

To consolidate the parametric studies of the integration schemes developed, stability analyses are performed using the energy and spectral stability methods. Both methods reveal that the *FB1* and *FB2* schemes yield conditionally and unconditionally stable algorithms depending on the choices of integration parameters.

Finite element numerical examples in the form of plane stress, plane strain and axisymmetric models are used to evaluate the performance of the algorithms; which demonstrate that the generalised Newton-Raphson scheme can considerably enhance convergence of the conventional Newton-Raphson scheme by converging quadratically even when a large time-step, (violating incremental strain laws) is used, without loss of accuracy, thus providing considerable advantages over the conventional scheme by reducing the costs of computations associated with an incremental process.

DECLARATION

I, Modify Andrew Elton Kaunda, declare that this dissertation is essentially my own work and that it has not been submitted for a degree at any other university.

Signed by candidate

M A E Kaunda

20. June, 94



ACKNOWLEDGEMENTS

I would like to thank my supervisor, Professor J B Martin, for his invaluable guidance and encouragement during my research.

I am also grateful to my sponsor, the South African Embassy (Malawi), for funding my studies. The financial support from the Foundation for Research Development (FRD) and the Centre for Research in Computational and Applied Mechanics (CERECAM) is greatly appreciated.

I am much indebted to Dr H T Pearce, of the Mechanical Engineering Department, who made arrangements for my admission to the Faculty of Engineering and made me feel at home during my stay in South Africa.

I wish to acknowledge use of a Personal Computer Finite Element Analysis Program (PCFEAP) provided by Professor R L Taylor of the University of California at Berkeley. I was also privileged with a free autographed book: Finite Elements in Plasticity, Theory and Practice, by Owen D R J and Hinton E, from Professor Owen of the University College of Swansea.

Lastly, but not least, I would like to thank my colleagues at CERECAM, especially, Dr G P Mitchell and Professor Daya Reddy, for invaluable discussions, the staff of the Department of Mechanical Engineering for keeping me going, and not forgetting the Dean's Office for its hospitality.

TABLE OF CONTENTS

	Page
ABSTRACT	III
DECLARATION.....	iv
ACKNOWLEDGEMENTS	v
LIST OF FIGURES	ix
NOMENCLATURE.....	x
EPIGRAPH.....	xii
CHAPTER 1 INTRODUCTION	1
REFERENCES	3
CHAPTER 2 THEORETICAL ASPECTS OF INTEGRATION SCHEMES.....	6
2.1 INTRODUCTION.....	6
2.2 EQUATIONS OF THE FIRST OR HIGHER ORDER	7
2.3 SOLUTION ALGORITHMS.....	7
2.3.1. <i>1st and 2nd Order Ordinary Differential Equations</i>	9
2.3.2 <i>Generalised Newton-Raphson (GNR) Scheme</i>	12
2.3.3 <i>Some Algebraic Manipulations</i>	17
2.4. ESTIMATION OF DISCRETISATION ERROR.....	19
2.5. CONCLUSIONS.....	21
REFERENCES	22
CHAPTER 3 THE EVOLUTION PROBLEM: CLASSICAL FRAMEWORK.....	23
3.1 INTRODUCTION.....	23

3.2 THE CONTINUUM PROBLEM OF EVOLUTION.....	23
3.2.1. <i>Quasi-static Elastoplasticity</i>	23
3.2.2. <i>Dynamic Elastoplasticity</i>	26
3.3 FINITE ELEMENT EQUATIONS AND ALGORITHMS.....	26
3.3.1 <i>Finite Element Equations</i>	27
3.3.2 <i>Solution Algorithms</i>	28
3.3.3 <i>Generalised Newton-Raphson Algorithm</i>	34
3.3.4 <i>Predictor-Corrector Steps</i>	35
3.3.5. <i>Stress Update</i>	38
3.3.6. <i>Stability Analysis</i>	40
3.4. CONCLUSIONS.....	41
REFERENCES.....	41
CHAPTER 4 THE EVOLUTION PROBLEM: INTERNAL VARIABLE FRAMEWORK.....	43
4.1. INTRODUCTION.....	43
4.2. THE CONTINUUM PROBLEM OF EVOLUTION.....	44
4.2.1. <i>Elastoplastic Constitutive Equations</i>	44
4.2.2. <i>Dissipation Function and Hardening Models</i>	46
4.3. FINITE ELEMENT EQUATIONS AND ALGORITHMS.....	47
4.3.1. <i>Quasi-static Elastoplasticity</i>	47
4.3.2. <i>Dynamic Elastoplasticity</i>	49
4.3.3. <i>The Generalised Newton-Raphson Scheme</i>	51
4.3.4. <i>Stability Analysis</i>	55
4.4. CONCLUSIONS.....	61
REFERENCES.....	61
CHAPTER 5 PARAMETRIC STUDIES OF INTEGRATION SCHEMES.....	64
5.1. INTRODUCTION.....	64

5.2. SOLUTION OF TRANSIENT LINEAR AND NON-LINEAR PROBLEMS	64
5.2.1. <i>Transient Linear Problems</i>	65
5.2.2. <i>Transient Non-linear Problems</i>	67
5.3. ACCURACY	70
5.4. STABILITY	70
5.4.1. <i>Spectral Stability Analysis</i>	71
5.4.2. <i>The Energy Method</i>	73
5.5. NUMERICAL EXAMPLES.....	77
5.5.1. <i>Plane Stress Problem</i>	78
5.5.2. <i>Plane Strain Problem</i>	84
5.5.3. <i>Axisymmetric Problem</i>	90
5.6. CONCLUSIONS.....	95
REFERENCES	96
CHAPTER 6 CONCLUSIONS.....	98

LIST OF FIGURES

	Page
Figure 1 (i) Forward difference scheme, $[x \rightarrow x_*]$; (ii) Backward difference scheme, $[x_* \rightarrow x + \Delta x]$	8
Figure 2 Assumptions: (i) $\partial M / \partial a \ll \partial P / \partial a$ and (ii) $\partial C / \partial a \ll \partial P / \partial a$	32
Figure 3 Perforated tension strip: (a) boundary restraints and applied load, (b) finite element mesh showing node numbering, (c) showing element numbering.....	78
Figure 4 Perforated tension strip (plane stress): 22-stress contours for times 0.04 to 0.1 (s).....	79
Figure 5 Perforated tension strip (plane stress): single time-step, $\Delta t=0.1$ (s); (a) 22-stress contours, (b) nodal displacement vectors.....	81
Figure 6 Perforated tension strip (plane stress): Graphs of Norms Vs Iteration No. showing convergence of iteration process.....	82
Figure 7 Thick cylinder subjected to internal pressure (plane strain): (a) boundary restraints and applied load, (b) finite element mesh showing node numbering, (c) showing element numbering.....	87
Figure 8 Thick cylinder subjected to internal pressure (plane strain): stress contours of yield level for times 12 to 19 (s).....	88
Figure 9 Thick cylinder subjected to internal pressure (plane strain): stress contours of yield level at a single time-step, $\Delta t = 19$ (s).	89
Figure 10 Thick cylinder subjected to internal pressure (plane strain): showing; (a) displacement of inner surface with increasing applied pressure, (b) circumferential (hoop) stress distributions across the thickness of the cylinder for various times during loading.....	90
Figure 11 Thick cylinder subjected to internal pressure (axisymmetric): (a) boundary restraints and applied load, (b) finite element mesh showing node numbering, (c) showing element numbering.....	91
Figure 12 Thick cylinder subjected to internal pressure (axisymmetric): stress contours of yield level for times 12 to 19 (s).	92
Figure 13 Thick cylinder subjected to internal pressure (axisymmetric): (a) stress contours of yield level at a single time-step, $\Delta t = 19$ (s), (b) corresponding displacement vectors.....	94
Figure 14 Thick cylinder subjected to internal pressure (axisymmetric): Circumferential (hoop) stress distribution across the cylinder thickness.....	94

NOMENCLATURE

Special symbols

\dot{x}	time differentiation: = dx / dt	$[.]^T$	matrix transpose
d	differential operator	$[.]^{-1}$	matrix inverse
∂	partial differential operator	$ $	absolute value of
Δ	increment in: = $(.)_{n+1} - (.)_n$	$ $	matrix or vector norm
∇	gradient operator	$O(.)$	of order
$\nabla \bullet$	divergence operator	\emptyset	empty set
$[.]$	a matrix		

Lowercase symbols

$\mathbf{a}, \dot{\mathbf{a}}, \ddot{\mathbf{a}}$	nodal or global displacement/velocity/acceleration vectors
$\mathbf{u}, \dot{\mathbf{u}}, \ddot{\mathbf{u}}$	displacement/velocity/acceleration vectors
\mathbf{e}	error vector
f	Helmholtz free energy
\mathbf{f}	applied load vector
k	counter
q	generalised co-ordinate
t	time

Uppercase symbols

$\mathbf{C}, \mathbf{G}, \mathbf{H}$	matrices of material properties defined in the text		
D	dissipation function	\mathbf{M}	mass matrix
\mathbf{B}	strain displacement matrix	N	(displacement) shape function
\mathbf{D}	elasticity matrix	\mathbf{P}	internal force vector
E	Young's modulus	T	kinetic energy
\mathbf{I}	identity matrix	U	strain energy
\mathbf{K}, \mathbf{S}	stiffness matrices	V	volume of body

Greek symbols

ε	strain vector		
σ	stress vector conjugate to ε		
λ	vector of internal variables		
χ	thermodynamic force vector conjugate to λ		
Ω	any domain	κ	hardening parameter
$\partial\Omega$	boundary of Ω	λ	Lagrange multiplier
α, β, γ	integration parameters	ν	Poisson's ratio
ϕ	convex yield function	ψ	residual vector
$\eta, \dot{\eta}$	preassigned vectors	ρ	mass density

Superscripts

i	iteration counter
e	elastic
p	plastic
q	index

Subscripts

ep	elastoplastic
n	time counter
j	index
o	initial condition index

Abbreviations

CAA	Constant average acceleration	GNR	generalised Newton-Raphson
CNR	conventional Newton-Raphson	GMP	generalised mid-point
FB	forward-backward	GT	generalised trapezoidal
FE	finite element	PC	personal computer
GN22	generalised Newmark scheme	PCFEAP	PC FE Analysis Program

EPIGRAPH

...when you can measure what you are speaking about and express it in numbers you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind.

Lord Kelvin

University of Cape Town

CHAPTER 1

INTRODUCTION

Recent work [1-10] at the FRD/UCT Centre for Research in Computational and Applied Mechanics has focused on the choice of algorithms for time discretisation in the finite element (FE) analysis of bodies subjected to both static and dynamic loading, and composed of materials which exhibit plastic or creep behaviour. One goal which is continuously receiving attention is that of looking for unifying concepts which relate the choice of integration parameters to material properties. Nevertheless, no single concept has completely satisfied the governing constitutive behaviour, principles of balance of linear and angular momenta and energy as well as best accuracy, stability, and overshoot considerations. A trade-off between these attributes becomes necessary. The aim of this work is to investigate some unifying concepts that cover a number of the desirable properties.

Through the use of an internal variable framework, a link is to be explored between a consistent mathematical programming scheme involving continuum and finite element formulations of the static and dynamic *initial-boundary-value* evolution problems and the Newton-Raphson scheme.

Solution algorithms for the incremental time-dependent and time-independent problems in elastoplasticity will be developed using a systematic approach based on a Taylor series

expansion in time and space. Without loss of generality, the investigation will be confined to problems:

- that consist of an elastoplastic von Mises material with linear hardening.
- characterised by the small strain rate-independent plasticity under isothermal conditions. Geometric non-linearity, associated with non-linear strain-displacement relationship that is characterised by large displacement or strain, will not be considered.

The outline of the dissertation is as follows:

Chapter 2 considers some theoretical aspects of integration schemes. In particular, the generalised trapezoidal rule and generalised Newton-Raphson scheme are derived using a systematic approach based on a Taylor series expansion in time and space. Some algebraic definitions useful in stability analysis are defined, and discretisation errors resulting from the recurrence equations are discussed.

In **Chapter 3**, continuum and finite element formulations of the static and dynamic initial-boundary-value evolution problems are considered in the classical framework and solution algorithms are developed using the generalised trapezoidal rule and generalised Newton-Raphson scheme of **Chapter 2**. A brief discussion of the stability analyses of the algorithms is given.

Continuum and finite element formulations of the static and dynamic initial-boundary-value evolution problems are further considered using the internal variable framework in **Chapter 4**. Solution algorithms are also developed using the generalised trapezoidal rule

and generalised Newton-Raphson scheme of **Chapter 2**. A detailed discussion of the stability analysis of the algorithms is presented using energy-like norms that form functions similar to the Lyapunov's functions [11-12].

Finally, in **Chapter 5**, solutions of transient linear and non-linear problems are considered with the goal of comparing the algorithms of **Chapter 2** with the conventional algorithms reported in various papers [13-22], books [23-30] and other literature. Additional algorithms are developed in the form of extensions to the generalised Newton-Raphson scheme, and further stability analyses are performed to consolidate parametric studies of integration schemes. Numerical examples are given in this chapter to evaluate the performance of algorithms.

REFERENCES

- [1] Martin J B, *Plasticity, Fundamentals and General Results*, MIT Press, 1975.
- [2] Martin J B, "A note on the implications of thermodynamic stability in the internal variable theory of inelastic solids", *Int. J. Solids and Structures*, **11**, 247-253, 1975.
- [3] Martin J B, "An internal variable approach to the formulation of finite element problems in plasticity", In *Physical Non-linearities in Structural Analysis*, ed. J Hult and J Lemaitre, Springer-Verlag Berlin Heidelberg, 165-176, (1981).
- [4] Martin J B, Reddy B D, Griffin T B, Bird W W, "Application of mathematical programming concepts to incremental elastic-plastic analysis", *Engng Struct.*, **9**, 171-176, (1987).
- [5] Martin J B, "A complementary work bounding principle for forward integration along the path of loading for elastic-plastic bodies", *J. Appl Mech.*, **109**, 341-345, (1987).
- [6] Martin J B, Reddy B D, "Variational principles and solution algorithms for internal variable formulations of problems in plasticity", In *Omaggio a Giulio: Note*

-
- Scientifiche in Occasione del 70^o Compleanno*, ed. U Andreaus et al., Universita di Roma 'La Sapienza'(Roma), 465-477, (1988).
- [7] Eve R A, Reddy B D, Rockafellar R T, "An internal variable theory of elastoplasticity based on maximum plastic work inequality", *Quart of Appl Math.*, **48**, 59-83, (1990).
- [8] Reddy B D, Martin J B, "Algorithms for the solution of internal variable problems in plasticity", *Computer Methods in Applied Mechanics and Engineering*, **93**, 253-273, (1991).
- [9] Martin J B, Nappi A, "An internal variable formulation for perfectly plastic and linear hardening relations in plasticity", *European Journal of Mechanics and Solids*, **9**, 107-131, (1990).
- [10] Martin J B, Reddy B D, "Piecewise smooth dissipation and yield functions in plasticity", In *Proc. XIth National Congress of Theoretical and Applied Mechanics(AIMETA), Trento, September, (1992)*.
- [11] D'Azzo J J, Houpis C H, *Linear Control System Analysis and Design, Conventional and Modern*, 2nd Edition, 1981.
- [12] Huseyin K, *Multiple Parameter Stability Theory and its Applications, Bifurcations, Catastrophes, Instabilities, ...*, Clarendon Press, Oxford, 1986.
- [13] Houbolt J C, "A recurrence matrix solution for the dynamic response of elastic aircraft", *J. Aero. Sci.*, **17**, 540-550, (1950).
- [14] Newmark N M, "A method of computation for structural dynamics", *J. Engng Mech. Div.*, ASCE, **85**, EM3, 67-94, (1959).
- [15] Wilson E L, "A computer program for the dynamic stress analysis of underground structures", *S.E.L Report 68-1*, University of California, Berkeley, (1968).
- [16] Hilber H M, Hughes T J R, Taylor R L, "Improved numerical dissipation for time integration algorithms in structural dynamics", *Int. J. Earthquake Eng. Struct. Dynam.*, **5**, 283-292, (1977).
- [17] Zienkiewicz O C, "A new look at the Newmark, Houbolt and other time-stepping formulas. A weighted residual approach", *Int. J. Earthquake Eng. Struct. Dynam.*, **5**, 413-418, (1977).

-
- [18] Hilber H M, Hughes T J R, “Collocation, dissipation and ‘overshoot’ for time integration schemes in structural dynamics”, *Int. J. Earthquake Eng. Struct. Dynam.*, **6**, 99-117, (1978).
- [19] Wood W L, “A further look at Newmark, Houbolt, etc., time-stepping formulae”, *Int. J. Num. Meth. Eng.*, **20**, 1009-1017, (1984).
- [20] Zienkiewicz O C, Wood W L, Hine N W, Taylor R L, “A unified set of single-step algorithms, Part 1: general formulation and applications”, *Int. J. Num. Meth. Eng.*, **20**, 1529-1552, (1984).
- [21] Wood W L, “A unified set of single-step algorithms, Part 2: theory”, *Int. J. Num. Meth. Eng.*, **20**, 2303-2309, (1984).
- [22] Katona M G, Zienkiewicz O C, “A unified set of single-step algorithms, Part 3: the beta-m method, a generalisation of the Newmark scheme”, *Int. J. Num. Meth. Eng.*, **21**, 1345-1359, (1985).
- [23] Henrici P, *Discrete Variable Methods in Ordinary Differential Equations*, John Wiley & Sons, Inc., 1962.
- [24] Henrici P, *Elements of Numerical Analysis*, John Wiley & Sons, Inc., 1964.
- [25] Owen D R J, Hinton E, *Finite Elements in Plasticity: Theory and Practice*, Pineridge Press Limited, 1980.
- [26] Bathe K J, *Finite Element Procedures in Engineering Analysis*, Prentice Hall, Inc., 1982.
- [27] Hughes T J R, *The Finite Element Method, Linear Static and Dynamic Finite Element Analysis*, Prentice Hall, Inc., 1987.
- [28] Wood W L, *Practical Time-setting Schemes*, Clarendon Press, Oxford, 1990.
- [29] Zienkiewicz O C, Taylor R L, *The Finite Element Method, Volume 1, Basic Formulations and Linear Problems*, 4th Edition, 1989.
- [30] Zienkiewicz O C, Taylor R L, *The Finite Element Method, Volume 2, Solid and Fluid Mechanics, Dynamics and Non-linearity*, 4th Edition, 1991.

CHAPTER 2

THEORETICAL ASPECTS OF INTEGRATION SCHEMES

2.1 INTRODUCTION

This chapter considers theoretical aspects of the generalised trapezoidal (GT) rule and generalised Newton-Raphson (GNR) scheme resulting from a systematic approach based on a Taylor series expansion. The GT rule provides means for integrating differential equations whereas the GNR scheme, usually in conjunction with some suitable integration scheme, such as GT and mid-point rules, for solving non-linear equations. Mid-point rules, multiple-step schemes and variants of the Newton-Raphson schemes are beyond the scope of this work and will not be considered.

Section 2.2 defines first and higher order differential equations that characterise most applications of interest in engineering. Section 2.3 provides solution algorithms based on the GT rule, and defines non-linear equations whose definitions are equivalent to those of Section 2.2, but which enable the direct use of the GNR scheme. The scheme includes the conventional Newton-Raphson scheme as a special case. This section also defines some algebraic definitions useful in stability analysis. Section 2.4 discusses discretisation errors resulting from the recurrence equations of the GT rule and GNR scheme.

2.2 EQUATIONS OF THE FIRST OR HIGHER ORDER

Consider a general system of p equations of order q , where $q \geq 1$ denotes an integer, and $\mathbf{f}, \mathbf{y}^1, \mathbf{y}^2, \dots, \mathbf{y}^q$ denote vectors with $p \geq 1$ components and assume vector-valued functions $\mathbf{f}(x, \mathbf{y}^1, \mathbf{y}^2, \dots, \mathbf{y}^q)$ defined for $x \in [a, b]$ and arbitrary vectors $\mathbf{y}^1, \dots, \mathbf{y}^q$. In particular, a system of p differential equations of order q

$$(1) \quad \mathbf{y}^{(q)} = \mathbf{f}(x, \mathbf{y}, \mathbf{y}', \dots, \mathbf{y}^{(q-1)}) ,$$

for the purpose of this section, means, any vector-valued function $\mathbf{y}(x)$ which is q times differentiable and satisfies the identity

$$(2) \quad \mathbf{y}^{(q)}(x) = \mathbf{f}(x, \mathbf{y}(x), \mathbf{y}'(x), \dots, \mathbf{y}^{(q-1)}(x)), \quad x \in [a, b] ,$$

with initial values

$$(3) \quad \mathbf{y}(a) = \boldsymbol{\eta}, \quad \mathbf{y}'(a) = \boldsymbol{\eta}', \quad \dots, \quad \mathbf{y}^{(q-1)}(a) = \boldsymbol{\eta}^{(q-1)} ,$$

where $\boldsymbol{\eta}, \boldsymbol{\eta}', \dots, \boldsymbol{\eta}^{(q-1)}$ are preassigned vectors. A solution $\mathbf{y}(x)$ of the posed *initial value problem* is required.

2.3 SOLUTION ALGORITHMS

In this section, approximate solutions for the posed initial value problem are developed using a systematic approach based on a Taylor series expansion. The procedure consists of

estimating the vectors $\mathbf{y}, \mathbf{y}', \dots, \mathbf{y}^{(q-1)}$ of the given function. Let $\mathbf{y} = [\mathbf{y}, \mathbf{y}', \dots, \mathbf{y}^{(q-1)}]^T$ represent the general variable. Expanding this variable using Taylor series gives

$$(4) \quad \begin{aligned} \mathbf{y}(x + \Delta x) &= e^{\{\Delta x(d/dx)\}} \cdot \mathbf{y}(x) \\ &= \left(1 + \Delta x \frac{d}{dx} + \frac{(\Delta x)^2}{2} \frac{d^2}{dx^2} + \dots\right) \cdot \mathbf{y}(x) \end{aligned}$$

where $e^{\{\Delta x(d/dx)\}}$ operates on $\mathbf{y}(x)$, and e is the natural base. The increment Δx , may be subdivided into two increments of $(1 - \alpha) \cdot \Delta x$ for the first interval, $[x_n, x_*]$, and $\alpha \cdot \Delta x$ for the second interval, $[x_*, x_{n+1}]$. The geometric implications are illustrated graphically in Figure 1.

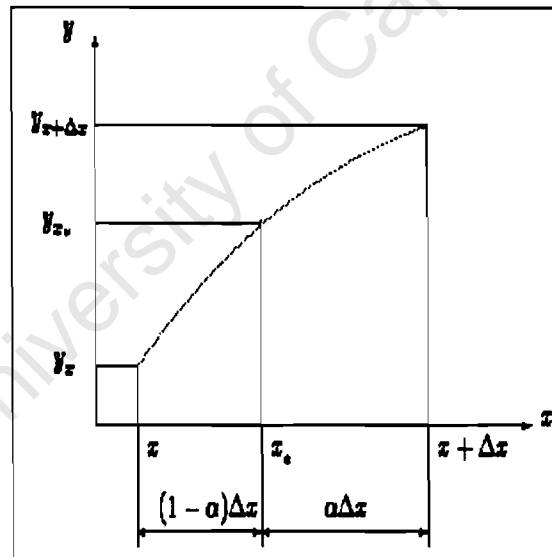


Figure 1 (i) Forward difference scheme, $[x \rightarrow x_*]$; (ii) Backward difference scheme, $[x_* \rightarrow x + \Delta x]$.

The scalar $\alpha \in [0, 1]$ is an integration parameter. The variable is then expanded about the pivot x_n , in a forward difference expansion for the first interval and about the pivot x_{n+1} , in a backward difference expansion for the second interval, giving

$$(5) \quad \mathbf{y}(x_*) = e^{\{(1-\alpha)\Delta x(d/dx)\}} \cdot \mathbf{y}(x) = e^{\{-\alpha\Delta x(d/dx)\}} \cdot \mathbf{y}(x + \Delta x) .$$

In discrete form, this simplifies to a single-step recurrence equation

$$(6) \quad \mathbf{y}(x_{n+1}) = \mathbf{y}(x_n) + \sum_{k=1}^{k=\infty} \frac{\Delta x^k}{k!} \left[(1-\alpha)^k \left(\frac{d^k \mathbf{y}}{dx^k} \right) \Big|_{x_n} - (-1)^k \alpha^k \left(\frac{d^k \mathbf{y}}{dx^k} \right) \Big|_{x_{n+1}} \right] .$$

The simplest algorithms result from taking the summation interval as $\sum_{k=1}^{k=q}$. This recurrence equation is the basic form of the *forward-backward* (FB) difference scheme [1,2,3].

2.3.1. 1st and 2nd Order Ordinary Differential Equations

Consider a first order differential equation with $q = 1$ whose initial value problem satisfies the identity

$$(7) \quad \mathbf{y}'(x) = \mathbf{f}(x, \mathbf{y}(x)), \quad \mathbf{y}(a) = \boldsymbol{\eta}, \quad x \in [a, b],$$

where $\boldsymbol{\eta}$ represents an initial vector. The solution of the initial value first order differential equation consists of the following pair of simultaneous equations:

$$(8) \quad \mathbf{y}(x_{n+1}) = \mathbf{y}(x_n) + \Delta x \cdot [(1-\alpha) \cdot \mathbf{y}'(x_n) + \alpha \cdot \mathbf{y}'(x_{n+1})], \quad n = 0, 1, \dots,$$

$$(9) \quad \mathbf{y}'(x_{n+1}) = \mathbf{f}(x_{n+1}, \mathbf{y}(x_{n+1})).$$

The first recurrence equation is derived from the FB scheme by expanding $y(x_{n+1})$ with the summation interval taken as $\sum_{k=1}^{k=1}$. The above algorithm constitutes the generalised trapezoidal (GT) rule. With $\alpha = 0$, the algorithm is equivalent to the explicit Euler forward difference scheme, whereas if $\alpha = 1$, it is equivalent to the implicit Euler backward difference scheme. An algorithm is implicit if $\alpha > 0$.

The initial value second order differential equation that has $q = 2$ satisfies the conditions

$$(10) \quad y''(x) = f(x, y(x), y'(x)), \quad y(a) = \eta, \quad y'(a) = \eta', \quad x \in [a, b],$$

where η and η' represent initial vectors. The solution of the initial value second order differential equation consists of the following three simultaneous equations:

$$(11) \quad y(x_{n+1}) = y(x_n) + \Delta x \cdot [(1-\alpha) \cdot y'(x_n) + \alpha \cdot y'(x_{n+1})] \\ + \frac{\Delta x^2}{2} \cdot [(1-\alpha)^2 \cdot y''(x_n) - \alpha^2 \cdot y''(x_{n+1})], \quad n = 0, 1, \dots$$

$$(12) \quad y'(x_{n+1}) = y'(x_n) + \Delta x \cdot [(1-\alpha) \cdot y''(x_n) + \alpha \cdot y''(x_{n+1})], \quad n = 0, 1, \dots$$

$$(13) \quad y''(x_{n+1}) = f(x_{n+1}, y(x_{n+1}), y'(x_{n+1})).$$

The first recurrence equation (11) is derived from the FB scheme by expanding $y(x_{n+1})$ with the summation interval taken as $\sum_{k=1}^{k=2}$, whereas the second (12) is derived from the FB scheme by expanding $y'(x_{n+1})$ with the summation interval taken as $\sum_{k=1}^{k=1}$. With $\alpha = 0$, the algorithm is known as the explicit forward difference scheme whereas if $\alpha = 1$, the implicit backward difference scheme. From

now on, this algorithm is termed the *FBI scheme*. A second version of the forward-backward (FB) difference scheme will be presented and termed the *FB2 scheme*.

From accuracy consideration, the truncation error depends on the integration parameter α and is different for each recurrence equation. This is evident when the Taylor series is written with the remainder term as:

$$(14) \quad y(x+h) = y(x) + hy'(x) + \frac{h^2}{2!}y''(x) + \dots + \frac{h^k}{k!}y^{(k)}(x + \xi h), \quad 0 \leq \xi \leq 1.$$

According to Taylor's theorem, the series is exact, if the final derivative is evaluated at some point not known *a priori* within the jump. For a good approximate termination, the final derivative may be evaluated at a point a fraction $1/(k+1)$ along the jump [4] as follows:

$$(15) \quad y(x+h) \cong y(x) + hy'(x) + \frac{h^2}{2!}y''(x) + \dots + \frac{h^k}{k!}y^{(k)}\left(x + \frac{h}{k+1}\right).$$

This implies that $\xi \cong 1/(k+1)$. Therefore, a different integration parameter is necessary for each recurrence equation. Taking this into consideration, the *FBI* scheme consists of the following simultaneous equations:

$$(16) \quad \begin{aligned} y(x_{n+1}) = & y(x_n) + \Delta x \cdot [(1 - \alpha_2) \cdot y'(x_n) + \alpha_2 \cdot y'(x_{n+1})] \\ & + \frac{\Delta x^2}{2} \cdot [(1 - \alpha_2)^2 \cdot y''(x_n) - \alpha_2^2 \cdot y''(x_{n+1})], \quad n = 0, 1, \dots \end{aligned}$$

$$(17) \quad y'(x_{n+1}) = y'(x_n) + \Delta x \cdot [(1 - \alpha_1) \cdot y''(x_n) + \alpha_1 \cdot y''(x_{n+1})], \quad n = 0, 1, \dots,$$

$$(18) \quad y''(x_{n+1}) = f(x_{n+1}, y(x_{n+1}), y'(x_{n+1})).$$

The *FB2* scheme consists of recurrence equations where all but the last implicit terms of derivatives in the right hand side (RHS) are eliminated. The RHS of the recurrence equations contain only implicit terms of $\mathbf{y}^{(q)}$. For the second order differential equations, the recurrence equations contain only implicit terms of \mathbf{y}'' . This is achieved by substituting the recurrence equation (12) representing the first derivative into the other recurrence equation (11). The resulting *FB2* scheme consists of the following simultaneous equations:

$$(19) \quad \mathbf{y}(x_{n+1}) = \mathbf{y}(x_n) + \Delta x \cdot \mathbf{y}'(x_n) + \frac{\Delta x^2}{2} \cdot [(1 - \alpha_2^2) \cdot \mathbf{y}''(x_n) + \alpha_2^2 \cdot \mathbf{y}''(x_{n+1})],$$

$$n = 0, 1, \dots$$

$$(20) \quad \mathbf{y}'(x_{n+1}) = \mathbf{y}'(x_n) + \Delta x \cdot [(1 - \alpha_1) \cdot \mathbf{y}''(x_n) + \alpha_1 \cdot \mathbf{y}''(x_{n+1})], \quad n = 0, 1, \dots,$$

$$(21) \quad \mathbf{y}''(x_{n+1}) = \mathbf{f}(x_{n+1}, \mathbf{y}(x_{n+1}), \mathbf{y}'(x_{n+1})).$$

The *FB2* scheme is identical to the Newmark method if $\alpha_2^2 = 2\beta$ and $\alpha_1 = \gamma$, where β and γ are the Newmark [5] integration parameters.

2.3.2 Generalised Newton-Raphson (GNR) Scheme

In this section, the equation

$$(22) \quad \Psi(x, \mathbf{y}, \mathbf{y}', \dots, \mathbf{y}^{(q)}) = \mathbf{0},$$

represents the system of differential equations of order q . This definition is equivalent to that of Section 2.2. Any vector-valued function $\mathbf{y}(x)$ which is q times differentiable and satisfies the identity

$$(23) \quad \Psi(x, \mathbf{y}(x), \mathbf{y}'(x), \dots, \mathbf{y}^{(q)}(x)) = \mathbf{0}, \quad x \in [a, b],$$

subject to initial values

$$(24) \quad \mathbf{y}(a) = \boldsymbol{\eta}, \quad \mathbf{y}'(a) = \boldsymbol{\eta}', \quad \dots, \quad \mathbf{y}^{(q-1)}(a) = \boldsymbol{\eta}^{(q-1)},$$

where $\boldsymbol{\eta}, \boldsymbol{\eta}', \dots, \boldsymbol{\eta}^{(q-1)}$ are preassigned vectors, is a solution of the posed *initial value problem*. The equations might be non-linear in which case the generalised Newton-Raphson (GNR) scheme may be applicable to furnish approximate solutions. In this scheme, the function representing the system of non-linear differential equations of order q is expanded using Taylor series, giving

$$(25) \quad \Psi(\mathbf{x}_{n+1}, \mathbf{y}(\mathbf{x}_{n+1}), \mathbf{y}'(\mathbf{x}_{n+1}), \dots, \mathbf{y}^{(q)}(\mathbf{x}_{n+1})) = e^{\left\{ \Delta \mathbf{x} \frac{\partial}{\partial \mathbf{x}} + \Delta \mathbf{y}^T \frac{\partial}{\partial \mathbf{y}} + \Delta \mathbf{y}'^T \frac{\partial}{\partial \mathbf{y}'} + \dots + \Delta \mathbf{y}^{(q)T} \frac{\partial}{\partial \mathbf{y}^{(q)}} \right\}} \cdot \Psi(\mathbf{x}_n, \mathbf{y}(\mathbf{x}_n), \mathbf{y}'(\mathbf{x}_n), \dots, \mathbf{y}^{(q)}(\mathbf{x}_n)),$$

where $\partial / \partial \mathbf{y}, \partial / \partial \mathbf{y}', \dots, \partial / \partial \mathbf{y}^{(q)}$, are partial differential operators, operating on Ψ_n . From now on, Ψ_n denotes approximate quantity of $\Psi(\dots)_n$; and all other variables with such subscripts imply similar definitions. Assuming Ψ does not depend on x explicitly, the expansion simplifies to

$$(26) \quad \Psi_{n+1} = \Psi_n + \sum_{k=1}^{k=\infty} \frac{1}{k!} \left[\Delta \mathbf{y}^T \frac{\partial}{\partial \mathbf{y}} + \Delta \mathbf{y}'^T \frac{\partial}{\partial \mathbf{y}'} + \dots + \Delta \mathbf{y}^{(q)T} \frac{\partial}{\partial \mathbf{y}^{(q)}} \right]^k \cdot \Psi_n.$$

Using the technique of the FB scheme, the expansion takes the implicit form

$$(27) \quad \begin{aligned} \Psi_{n+1} &= \Psi_n + \\ &\sum_{k=1}^{k=\infty} \frac{1}{k!} \left[\Delta \mathbf{y}^T (1 - \beta_1) \frac{\partial}{\partial \mathbf{y}} + \Delta \mathbf{y}'^T (1 - \beta_2) \frac{\partial}{\partial \mathbf{y}'} + \dots + \Delta \mathbf{y}^{(q)T} (1 - \beta_{q+1}) \frac{\partial}{\partial \mathbf{y}^{(q)}} \right]^k \Psi_n \\ &- \sum_{k=1}^{k=\infty} \frac{(-1)^k}{k!} \left[\Delta \mathbf{y}^T \beta_1 \frac{\partial}{\partial \mathbf{y}} + \Delta \mathbf{y}'^T \beta_2 \frac{\partial}{\partial \mathbf{y}'} + \dots + \Delta \mathbf{y}^{(q)T} \beta_{q+1} \frac{\partial}{\partial \mathbf{y}^{(q)}} \right]^k \Psi_{n+1}, \\ &\beta_j \in [0, 1] \end{aligned}$$

The generalised Newton-Raphson (GNR) scheme results when the linearised form of the above expression is equated to zero, with the upper limit of the summation interval set to $k=1$.

For a *zero order differential* or *algebraic/transcendental* equation, the GNR scheme simplifies to the solution algorithm that satisfies the following pair of equations:

$$(28) \quad \Psi_{n+1} = \Psi_n + \Delta \mathbf{y}^T \left\{ (1 - \beta_1) \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_n + \beta_1 \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_{n+1} \right\} = \mathbf{0} ,$$

$$(29) \quad \mathbf{y}_{n+1} = \mathbf{y}_n + \Delta \mathbf{y} \quad n = 0, 1, \dots$$

The coefficient of $\Delta \mathbf{y}^T$ is the generalised Jacobean. The conventional Newton-Raphson (CNR) scheme is recovered from the GNR scheme by setting $\beta_1 = 0$.

For a *first order differential* equation, the GNR scheme simplifies to the solution algorithm that satisfies the following set of equations:

$$(30) \quad \Psi_{n+1} = \Psi_n + \Delta \mathbf{y}^T \cdot \left[(1 - \beta_1) \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_n + \beta_1 \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_{n+1} \right] \\ + \Delta \mathbf{y}'^T \cdot \left[(1 - \beta_2) \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_n + \beta_2 \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_{n+1} \right] = \mathbf{0}$$

$$(31) \quad \mathbf{y}_{n+1} = \mathbf{y}_n + \Delta \mathbf{y} \quad n = 0, 1, \dots ,$$

$$(32) \quad \mathbf{y}'_{n+1} = \mathbf{y}'_n + \Delta \mathbf{x} \cdot [(1 - \alpha_1) \cdot \mathbf{y}'_n + \alpha_1 \cdot \mathbf{y}'_{n+1}] \quad n = 0, 1, \dots$$

The above equations can be combined to form the GNR as follows:

$$\begin{aligned}
 & \Psi_{n+1} = \\
 & \Psi_n + \Delta \mathbf{y}^T \left\{ (1 - \beta_1) \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_n + \beta_1 \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_{n+1} + \right. \\
 (33) \quad & \left. \frac{1}{\alpha_1 \Delta x} \left[(1 - \beta_2) \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_n + \beta_2 \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_{n+1} \right] \right\} \\
 & - \frac{1}{\alpha_1} \left[(1 - \beta_2) \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_n + \beta_2 \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_{n+1} \right] \cdot \mathbf{y}'_n = \mathbf{0}
 \end{aligned}$$

$$(34) \quad \mathbf{y}_{n+1} = \mathbf{y}_n + \Delta \mathbf{y} \quad n = 0, 1, \dots$$

The coefficient of $\Delta \mathbf{y}^T$ is the generalised Jacobean, and the GNR scheme for solving first order differential equations is implicit in β_1 and β_2 . The CNR scheme is recovered from the GNR scheme by setting $\beta_1 = \beta_2 = 0$.

Similarly, for a *second order differential* equation, the GNR scheme simplifies to the solution algorithm that satisfies the following set of equations:

$$\begin{aligned}
 & \Psi_{n+1} = \Psi_n + \Delta \mathbf{y}^T \cdot \left[(1 - \beta_1) \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_n + \beta_1 \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_{n+1} \right] \\
 (35) \quad & + \Delta \mathbf{y}'^T \cdot \left[(1 - \beta_2) \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_n + \beta_2 \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_{n+1} \right] \\
 & + \Delta \mathbf{y}''^T \cdot \left[(1 - \beta_3) \left(\frac{\partial \Psi}{\partial \mathbf{y}''} \right) \Big|_n + \beta_3 \left(\frac{\partial \Psi}{\partial \mathbf{y}''} \right) \Big|_{n+1} \right] = \mathbf{0}
 \end{aligned}$$

$$(36) \quad \mathbf{y}_{n+1} = \mathbf{y}_n + \Delta \mathbf{y} \quad n = 0, 1, \dots,$$

$$\begin{aligned}
 & \mathbf{y}_{n+1} = \mathbf{y}_n + \Delta x \cdot [(1 - \alpha_2) \cdot \mathbf{y}'_n + \alpha_2 \cdot \mathbf{y}'_{n+1}] \\
 (37) \quad & + \frac{\Delta x^2}{2} \cdot [(1 - \alpha_2)^2 \cdot \mathbf{y}''_n - \alpha_2^2 \cdot \mathbf{y}''_{n+1}], \text{ FB1 scheme}
 \end{aligned}$$

$$(38) \quad \mathbf{y}_{n+1} = \mathbf{y}_n + \Delta x \cdot \mathbf{y}'_n + \frac{\Delta x^2}{2} \cdot [(1 - \alpha_2^2) \cdot \mathbf{y}''_n + \alpha_2^2 \cdot \mathbf{y}''_{n+1}], \text{ FB2 scheme ,}$$

$$(39) \quad \mathbf{y}'_{n+1} = \mathbf{y}'_n + \Delta x \cdot [(1 - \alpha_1) \cdot \mathbf{y}''_n + \alpha_1 \cdot \mathbf{y}''_{n+1}] \quad n = 0, 1, \dots$$

The GNR scheme for solving second order differential equations is implicit in β_1 , β_2 and β_3 . The CNR scheme is recovered from the GNR scheme by setting $\beta_1 = \beta_2 = \beta_3 = 0$.

The above set of equations may be combined to form the GNR-FB1 scheme as follows:

$$(40) \quad \begin{aligned} \Psi_{n+1} = & \Psi_n + \Delta \mathbf{y}^T \left\{ (1 - \beta_1) \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_n + \beta_1 \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_{n+1} \right. \\ & + \frac{2\alpha_1}{\Delta x \alpha_2 (2\alpha_1 - \alpha_2)} \left[(1 - \beta_2) \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_n + \beta_2 \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_{n+1} \right] \\ & \left. + \frac{2}{\Delta x^2 \alpha_2 (2\alpha_1 - \alpha_2)} \left[(1 - \beta_3) \left(\frac{\partial \Psi}{\partial \mathbf{y}''} \right) \Big|_n + \beta_3 \left(\frac{\partial \Psi}{\partial \mathbf{y}''} \right) \Big|_{n+1} \right] \right\} \\ & - \left[(1 - \beta_2) \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_n + \beta_2 \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_{n+1} \right] \times \\ & \left[\frac{2\alpha_1}{\alpha_2 (2\alpha_1 - \alpha_2)} \mathbf{y}'_n - \Delta x \left(1 - \frac{\alpha_1}{\alpha_2 (2\alpha_1 - \alpha_2)} \right) \mathbf{y}''_n \right] \\ & - \left[(1 - \beta_3) \left(\frac{\partial \Psi}{\partial \mathbf{y}''} \right) \Big|_n + \beta_3 \left(\frac{\partial \Psi}{\partial \mathbf{y}''} \right) \Big|_{n+1} \right] \times \\ & \left[\frac{2}{\Delta x \alpha_2 (2\alpha_1 - \alpha_2)} \mathbf{y}'_n + \frac{1}{\alpha_2 (2\alpha_1 - \alpha_2)} \mathbf{y}''_n \right] = 0 \quad , \end{aligned}$$

$$(41) \quad \mathbf{y}_{n+1} = \mathbf{y}_n + \Delta \mathbf{y} \quad n = 0, 1, \dots$$

The coefficient of $\Delta \mathbf{y}^T$ is the generalised Jacobean. The corresponding GNR-FB2 scheme for solving the second order differential equations consists of the following pair of equations:

$$\begin{aligned}
\Psi_{n+1} = & \Psi_n + \Delta \mathbf{y}^T \left\{ (1-\beta_1) \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_n + \beta_1 \left(\frac{\partial \Psi}{\partial \mathbf{y}} \right) \Big|_{n+1} \right. \\
& + \frac{2\alpha_1}{\Delta x \alpha_2^2} \left[(1-\beta_2) \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_n + \beta_2 \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_{n+1} \right] \\
(42) \quad & \left. + \frac{2}{\Delta x^2 \alpha_2^2} \left[(1-\beta_3) \left(\frac{\partial \Psi}{\partial \mathbf{y}''} \right) \Big|_n + \beta_3 \left(\frac{\partial \Psi}{\partial \mathbf{y}''} \right) \Big|_{n+1} \right] \right\} \\
& - \left[(1-\beta_2) \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_n + \beta_2 \left(\frac{\partial \Psi}{\partial \mathbf{y}'} \right) \Big|_{n+1} \right] \cdot \left[\frac{2\alpha_1}{\alpha_2^2} \mathbf{y}'_n - \Delta x \left(1 - \frac{\alpha_1}{\alpha_2^2} \right) \mathbf{y}''_n \right] \\
& - \left[(1-\beta_3) \left(\frac{\partial \Psi}{\partial \mathbf{y}''} \right) \Big|_n + \beta_3 \left(\frac{\partial \Psi}{\partial \mathbf{y}''} \right) \Big|_{n+1} \right] \cdot \left[\frac{2}{\Delta x \alpha_2^2} \mathbf{y}'_n + \frac{1}{\alpha_2^2} \mathbf{y}''_n \right] = 0
\end{aligned}$$

$$(43) \quad \mathbf{y}_{n+1} = \mathbf{y}_n + \Delta \mathbf{y} \quad n = 0, 1, \dots$$

Similarly, the coefficient of $\Delta \mathbf{y}^T$ is the generalised Jacobean. While the procedure for developing the generalised Newton-Raphson schemes for first and second order differential equations is as simple as for the algebraic or transcendental equations, the resulting equations are more complicated.

2.3.3 Some Algebraic Manipulations

This section defines some algebraic manipulations useful in stability analysis. Let

$$\langle \rangle_n = \frac{1}{2} \left[()_{n+1} + ()_n \right] \quad \text{and} \quad \Delta () = ()_{n+1} - ()_n,$$

represent short hand notation for the average and difference of two variables, respectively. Consider two variables, y_n and y_{n+1} . Each variable can be written in terms of the average and difference of the two variables as follows:

$$(44) \quad y_n = \frac{1}{2}(y_{n+1} + y_n) - \frac{1}{2}(y_{n+1} - y_n) = \langle y_n \rangle - \frac{\Delta y}{2},$$

$$(45) \quad y_{n+1} = \frac{1}{2}(y_{n+1} + y_n) + \frac{1}{2}(y_{n+1} - y_n) = \langle y_n \rangle + \frac{\Delta y}{2}.$$

With these definitions, it is possible to express recurrence equations in terms of averages and differences of variables that serve to simplify algorithms. Results of these manipulations are useful in writing energy equations and stability analysis via energy methods.

For the algorithms developed above, the recurrence equations that require transforming are:

$$(46) \quad y_{n+1} = y_n + \Delta x \cdot [(1 - \alpha_2) \cdot y'_n + \alpha_2 \cdot y'_{n+1}] + \frac{\Delta x^2}{2} \cdot [(1 - \alpha_2)^2 \cdot y''_n - \alpha_2^2 \cdot y''_{n+1}], \text{ FB1 scheme}$$

$$(47) \quad y_{n+1} = y_n + \Delta x \cdot y'_n + \frac{\Delta x^2}{2} \cdot [(1 - \alpha_2^2) \cdot y''_n + \alpha_2^2 \cdot y''_{n+1}], \text{ FB2 scheme ,}$$

$$(48) \quad y'_{n+1} = y'_n + \Delta x \cdot [(1 - \alpha_1) \cdot y''_n + \alpha_1 \cdot y''_{n+1}] \quad n = 0, 1, \dots$$

The corresponding transformations are:

$$(49) \quad \Delta x \langle y'_n \rangle = \Delta y - \frac{\Delta x^2}{2} (2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2) \Delta y'', \quad \text{FB1 scheme ,}$$

$$(50) \quad \Delta x \langle y'_n \rangle = \Delta y - \frac{\Delta x^2}{2} (\alpha_2^2 - \alpha_1) \Delta y'', \quad \text{FB2 scheme ,}$$

$$(51) \quad \Delta y' = \Delta x \langle y''_n \rangle + (\alpha_1 - \frac{1}{2}) \Delta x \Delta y''.$$

These equations may serve to eliminate $\langle y'_n \rangle$ and $\Delta y'$ from algorithms as required in energy stability analysis. In mechanics, energy equations may be formulated in terms of $\langle \dot{y}_n \rangle$ and $\Delta \dot{y}$, which denote average of and increment in velocity. In the

stability analysis via energy methods, elimination of these variables is an important operation. The next section briefly outlines discretisation errors resulting from the recurrence equations described in Sections 2.2 and 2.3.

2.4. ESTIMATION OF DISCRETISATION ERROR

The truncation or discretisation error of a recurrence equation plays an important role in estimating the accuracy of algorithms. Each recurrence equation that approximates the exact one is terminated after a certain number of terms in the Taylor series expansion. The truncation error is a measure of accuracy and rate of convergence. Algorithm round-off and inherited errors are beyond the scope of this work. These errors are discussed in [6] and references contained therein.

The truncation errors for the recurrence equations developed may be estimated by comparing the equations (46-48) with the corresponding full Taylor series. Let

$$(52) \quad y(x_{n+1}) = y(x_n) + \sum_{k=1}^{k=\infty} \frac{\Delta x^k}{k!} \left(\frac{d^k y}{dx^k} \right) \Bigg|_{x_n},$$

$$(53) \quad y'(x_{n+1}) = y'(x_n) + \sum_{k=1}^{k=\infty} \frac{\Delta x^k}{k!} \left(\frac{d^{k+1} y}{dx^{k+1}} \right) \Bigg|_{x_n},$$

represent the exact solutions for the initial value problem. The recurrence equations (46-47) for the *FB1* and *FB2* schemes may be expanded so that terms with subscripts $(_{n+1})$ do not appear on the RHS of the equations as follows [3]:

$$\begin{aligned}
(54) \quad y_{n+1} &= y_n + \Delta x \left[(1 - \alpha_2) y'_n + \alpha_2 \left(y'_n + \sum_{k=1}^{k=\infty} \frac{\Delta x^k}{k!} \left(\frac{d^{k+1} y}{dx^{k+1}} \right) \Big|_n \right) \right] + \\
&\quad \frac{\Delta x^2}{2} \left[(1 - \alpha_2)^2 y''_n - \alpha_2^2 \left(y''_n + \sum_{k=1}^{k=\infty} \frac{\Delta x^k}{k!} \left(\frac{d^{k+2} y}{dx^{k+2}} \right) \Big|_n \right) \right] \\
&= y_n + \Delta x y'_n + \frac{\Delta x^2}{2} y''_n + \sum_{k=3}^{k=\infty} \left(\frac{\alpha_2}{(k-1)!} - \frac{\alpha_2^2}{2(k-2)!} \right) \Delta x^k \left(\frac{d^k y}{dx^k} \right) \Big|_n \quad \text{FB1 scheme}
\end{aligned}$$

$$\begin{aligned}
(55) \quad y_{n+1} &= y_n + \Delta x y'_n + \frac{\Delta x^2}{2} \left[(1 - \alpha_2^2) y''_n + \alpha_2^2 \left(y''_n + \sum_{k=1}^{k=\infty} \frac{\Delta x^k}{k!} \left(\frac{d^{k+2} y}{dx^{k+2}} \right) \Big|_n \right) \right] \\
&= y_n + \Delta x y'_n + \frac{\Delta x^2}{2} y''_n + \sum_{k=3}^{k=\infty} \left(\frac{\alpha_2^2}{2(k-2)!} \right) \Delta x^k \left(\frac{d^k y}{dx^k} \right) \Big|_n \quad \text{FB2 scheme}
\end{aligned}$$

Then, the following equations define error estimates of the recurrence equations for the *FB1* and *FB2* schemes:

$$(56) \quad \mathbf{e} = y_{n+1} - y(x_{n+1}),$$

$$(57) \quad \mathbf{e} = \sum_{k=3}^{k=\infty} \left(\frac{\alpha_2}{(k-1)!} - \frac{\alpha_2^2}{2 \cdot (k-2)!} - \frac{1}{k!} \right) \Delta x^k \left(\frac{d^k y}{dx^k} \right) \Big|_n, \quad \text{FB1 scheme},$$

$$(58) \quad \mathbf{e} = \sum_{k=3}^{k=\infty} \left(\frac{\alpha_2^2}{2 \cdot (k-2)!} - \frac{1}{k!} \right) \Delta x^k \left(\frac{d^k y}{dx^k} \right) \Big|_n, \quad \text{FB2 scheme}.$$

For the recurrence equation of the *FB1* scheme, the truncation error is $O(\Delta x^3)$,

$\forall \alpha_2 \in [0, 1]$, but the error is a minimum when $\alpha_2 = \frac{1}{2}$. The truncation error for the *FB2* scheme is $O(\Delta x^4)$ for $\alpha_2^2 = \frac{1}{3}$, and $O(\Delta x^3)$ otherwise. The recurrence equation

$$(59) \quad y'_{n+1} = y'_n + \Delta x \cdot [(1 - \alpha_1) \cdot y''_n + \alpha_1 \cdot y''_{n+1}],$$

may also be expanded [3] so that terms with subscripts $(_{n+1})$ do not appear on the RHS of the equation as follows:

$$(60) \quad \begin{aligned} y'_{n+1} &= y'_n + \Delta x \left[(1 - \alpha_1) y''_n + \alpha_1 \left(y''_n + \sum_{k=1}^{k=\infty} \frac{\Delta x^k}{k!} \left(\frac{d^{k+2} y}{dx^{k+2}} \right) \Big|_n \right) \right] \\ &= y'_n + \Delta x y''_n + \alpha_1 \sum_{k=2}^{k=\infty} \frac{\Delta x^k}{(k-1)!} \left(\frac{d^{k+1} y}{dx^{k+1}} \right) \Big|_n \end{aligned}$$

The truncation error becomes

$$(61) \quad \mathbf{e} = y'_{n+1} - y'(x_{n+1}),$$

$$(62) \quad \mathbf{e} = \sum_{k=2}^{k=\infty} \left(\frac{\alpha_1}{(k-1)!} - \frac{1}{k!} \right) \Delta x^k \left(\frac{d^{k+1} y}{dx^{k+1}} \right) \Big|_n.$$

This gives a truncation error $O(\Delta x^2)$ for $\alpha_1 \neq \frac{1}{2}$ and $O(\Delta x^3)$ for $\alpha_1 = \frac{1}{2}$. The choice of $\alpha_1 = \frac{1}{2}$ results in best accuracy for all the algorithms considered above [3].

The truncation error for the GNR scheme is $O(\Delta y^2)$ when used as the conventional Newton-Raphson scheme [3]. The errors may be estimated in a similar way by comparing the equation representing the GNR scheme with the corresponding full Taylor series.

2.5. CONCLUSIONS

Theoretical aspects of the generalised trapezoidal (GT) rule and generalised Newton-Raphson (GNR) scheme have been considered in a systematic approach based on a Taylor series expansion..

First and higher order differential equations that characterise most applications of interest in engineering have been considered and single-step solution algorithms based on the GT rule, have been proposed. Non-linear equations have also been defined in a way that enables the direct use of the GNR scheme. The GNR scheme includes the conventional Newton-Raphson scheme as a special case.

Also defined in this chapter are some algebraic operations useful in stability analysis. Brief discussions on discretisation errors resulting from the recurrence equations of the GT rule and GNR scheme have been outlined.

In the next chapters, the techniques developed will be readily applied to *initial-boundary value* problems.

REFERENCES

- [1] Kaunda M A E, Martin J B, "A forward-backward difference time-integration scheme based on Taylor series", In *Proc. 11th Symposium on Finite Element Methods in South Africa (FEMSA)*, 287-296, (1992).
- [2] Kaunda M A E, Martin J B, "On single-step time-integrators", In *Proc. 18th South African Symposium on Numerical Mathematics (SANUM)*, 78-101, (1992).
- [3] Kaunda M A E, Martin J B, "Finite element solution algorithms for non-linear time-dependent/independent problems in solid mechanics", In *Proc. 19th South African Symposium on Numerical Mathematics (SANUM)*, 95-117, (1993).
- [4] Irons B M, Shrive N G, *Numerical Methods in Engineering and Applied Science: Numbers are Fun*, Ellis Horwood Limited, 1987.
- [5] Newmark N M, "A method of computation for structural dynamics", *J. Engng Mech. Div.*, ASCE, **85**, EM3, 67-94, (1959).
- [6] Crandall S H, *Engineering Analysis, A Survey of Numerical Procedures*, McGraw-Hill Book Company, Inc., 1956.

CHAPTER 3

THE EVOLUTION PROBLEM: CLASSICAL FRAMEWORK

3.1 INTRODUCTION

Continuum and finite element formulations of the static and dynamic *initial-boundary value* evolution problems will be considered in the classical framework and solution algorithms will be developed using the generalised trapezoidal (GT) rule and generalised Newton-Raphson (GNR) scheme developed in **Chapter 2**.

Sections **3.2.1** and **3.2.2** consider the continuum problem of evolution in terms of the classical framework, whereas in Section **3.3.1** the finite element formulations are treated. Solution algorithms are developed in Section **3.3.2**, and the generalised Newton-Raphson algorithm is presented in Section **3.3.3**. Detailed analysis of solution algorithms is presented in Sections **3.3.4** to **3.3.5**, and Section **3.3.6** is a brief discussion of the stability analysis of the algorithms in the context of earlier results.

3.2 THE CONTINUUM PROBLEM OF EVOLUTION

This section considers quasi-static and dynamic *initial-boundary value* continuum problems of evolution characterised by classical small strain rate-independent plasticity under isothermal conditions.

3.2.1. Quasi-static Elastoplasticity

Consider a quasi-static *initial-boundary value* problem defined by the equilibrium equation

$$(1) \quad \nabla \cdot \boldsymbol{\sigma} + \mathbf{f} = \mathbf{0} ,$$

where $\nabla \cdot \boldsymbol{\sigma}$ denotes the divergence of the Cauchy stress; and \mathbf{f} the given body force. The constitutive equation associated with the classical small strain rate-independent plasticity under isothermal conditions is given by

$$(2) \quad \dot{\boldsymbol{\sigma}} = \mathbf{D}(\dot{\boldsymbol{\varepsilon}} - \dot{\boldsymbol{\varepsilon}}^p) .$$

where \mathbf{D} is a positive definite symmetric matrix of elasticity constants, and

$$(3) \quad \boldsymbol{\varepsilon} = \nabla \mathbf{u} = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T) ,$$

is the total strain field. The plastic strain rate is given by the associated flow rule

$$(4) \quad \dot{\boldsymbol{\varepsilon}}^p = \lambda \cdot \frac{\partial \phi(\boldsymbol{\sigma}, \boldsymbol{\kappa})}{\partial \boldsymbol{\sigma}} ,$$

and the associated hardening law is given by

$$(5) \quad \dot{\boldsymbol{\kappa}} = -\lambda \cdot \mathbf{H} \frac{\partial \phi(\boldsymbol{\sigma}, \boldsymbol{\kappa})}{\partial \boldsymbol{\kappa}} ,$$

where \mathbf{H} is a positive definite matrix of hardening constants; and λ , a plastic Lagrange multiplier that obeys the Kuhn-Tucker loading and unloading relations

$$(6) \quad \lambda \geq 0, \quad \phi(\boldsymbol{\sigma}, \boldsymbol{\kappa}) \leq 0, \quad \lambda \cdot \phi(\boldsymbol{\sigma}, \boldsymbol{\kappa}) = 0 ,$$

along with the consistency condition

$$(7) \quad \lambda \cdot \dot{\phi}(\boldsymbol{\sigma}, \boldsymbol{\kappa}) = 0 ,$$

where $\phi(\sigma, \kappa)$ is a convex yield function, defining the onset of plastic flow, that is homogeneous and of degree one in its arguments; and κ , a hardening parameter. Under the above conditions, the plastic Lagrange multiplier may be computed [1] as follows:

$$(8) \quad \lambda = \begin{cases} \left\langle \frac{\partial \phi}{\partial \sigma} : \mathbf{D} \cdot \dot{\boldsymbol{\varepsilon}} \right\rangle / \left(\frac{\partial \phi}{\partial \sigma} : \mathbf{D} \cdot \frac{\partial \phi}{\partial \sigma} + \frac{\partial \phi}{\partial \kappa} \cdot \mathbf{H} \frac{\partial \phi}{\partial \kappa} \right) & \text{if } \phi = 0 \\ 0 & \text{if } \phi < 0 \end{cases},$$

where the non-smooth function is defined as

$$(9) \quad \left\langle \frac{\partial \phi}{\partial \sigma} : \mathbf{D} \cdot \dot{\boldsymbol{\varepsilon}} \right\rangle = \frac{1}{2} \left[\left(\frac{\partial \phi}{\partial \sigma} : \mathbf{D} \cdot \dot{\boldsymbol{\varepsilon}} \right) + \left| \left(\frac{\partial \phi}{\partial \sigma} : \mathbf{D} \cdot \dot{\boldsymbol{\varepsilon}} \right) \right| \right].$$

The solution to the quasi-static *initial-boundary value* problem consists of finding a displacement vector field $\mathbf{u}(x, t)$, and a stress field $\boldsymbol{\sigma}(x, t)$, that satisfy the above equations for all $x \in \Omega$ and $t \in [0, T]$, $T > 0$, such that

$$(10) \quad \mathbf{u}(x, 0) = \mathbf{u}(x), \quad x \in \Omega,$$

$$(11) \quad \boldsymbol{\sigma}(x, 0) = \boldsymbol{\sigma}(x), \quad x \in \Omega,$$

$$(12) \quad \mathbf{u}(x, t) = \mathbf{g}(x), \quad x \in \partial\Omega_1, \quad t \in [0, T],$$

$$(13) \quad \mathbf{n}(x) \cdot \boldsymbol{\sigma}(x, t) = \mathbf{h}(x), \quad x \in \partial\Omega_2, \quad t \in [0, T],$$

where $\mathbf{u}(x)$ and $\boldsymbol{\sigma}(x)$ represent the given initial data; Ω , a bounded region with piecewise smooth boundary $\partial\Omega$; $\mathbf{g}(x)$ and $\mathbf{h}(x)$, the given boundary data; $\mathbf{n}(x)$, the unit normal vector to $\partial\Omega$; and $\partial\Omega_1$ and $\partial\Omega_2$ are subregions of $\partial\Omega$ satisfying $\partial\Omega_1 \cup \partial\Omega_2 = \partial\Omega$ and $\partial\Omega_1 \cap \partial\Omega_2 = \emptyset$.

3.2.2. Dynamic Elastoplasticity

A dynamic *initial-boundary value* problem consists of the governing equations that satisfy the conservation of momentum, known as the equations of motion, and may be written as

$$(14) \quad \nabla \cdot \boldsymbol{\sigma} + \mathbf{f} = \rho \cdot \ddot{\mathbf{u}} ,$$

where ρ is the mass density and $\ddot{\mathbf{u}}$, the acceleration vector field. The problem includes constitutive equations and boundary data defined for the quasi-static *initial-boundary value* problem with additional initial data

$$(15) \quad \mathbf{u}(x, 0) = \mathbf{u}(x), \quad \dot{\mathbf{u}}(x, 0) = \dot{\mathbf{u}}(x), \quad x \in \Omega ,$$

where $\dot{\mathbf{u}}(x)$ is the initial velocity vector field. The solution to the dynamic *initial-boundary value* problem consists of finding a displacement vector field $\mathbf{u}(x, t)$, and a stress field $\boldsymbol{\sigma}(x, t)$, that satisfy the above equations for all $x \in \Omega$ and $t \in [0, T]$, $T > 0$.

3.3 FINITE ELEMENT EQUATIONS AND ALGORITHMS

In the finite element semi-discretisation procedure, the continuum bounded region Ω , is subdivided into elements Ω_e , and the displacement field in each element is approximated by shape functions $\mathbf{N}(x)$ that are functions of space alone, and nodal displacements $\mathbf{a}(t)$ as follows:

$$(16) \quad \mathbf{u} = \mathbf{N}(x) \cdot \mathbf{a}(t) .$$

The strain field then becomes

$$(17) \quad \boldsymbol{\varepsilon} = \mathbf{B}\mathbf{a} = \frac{1}{2}(\mathbf{B}\mathbf{a} + (\mathbf{B}\mathbf{a})^T),$$

where $\mathbf{B} = \nabla \mathbf{N}$ is the strain-displacement matrix.

3.3.1 Finite Element Equations

Matrix ordinary differential equations for the quasi-static and dynamic *initial-boundary value* problems may be written as follows [2]:

$$(18) \quad \mathbf{P}(\mathbf{a}) + \mathbf{f} = \mathbf{0} \quad (\text{Quasi - static}),$$

$$(19) \quad \mathbf{M}(\mathbf{a})\ddot{\mathbf{a}} + \mathbf{P}(\mathbf{a}) + \mathbf{f} = \mathbf{0} \quad (\text{Dynamic}),$$

where \mathbf{f} represents the equivalent nodal forces composed of body forces and applied loads, and the mass matrix and internal force vector are given by

$$(20) \quad \mathbf{M}(\mathbf{a}) = \int \mathbf{N}^T \rho \cdot \mathbf{N} dV, \quad \mathbf{P}(\mathbf{a}) = \int \mathbf{B}^T \boldsymbol{\sigma} dV.$$

The stiffness matrix is given by

$$(21) \quad \mathbf{S} = \frac{\partial \mathbf{P}(\mathbf{a})}{\partial \mathbf{a}} = \int \mathbf{B}^T \mathbf{D}_{ep} \mathbf{B} dV,$$

where $\mathbf{D}_{ep} = \partial \boldsymbol{\sigma} / \partial \boldsymbol{\varepsilon}$ is the elastoplastic stress-strain matrix. The boundary conditions are weakly enforced in the finite element equations.

3.3.2 Solution Algorithms

The generalised trapezoidal (GT) rule and generalised Newton-Raphson (GNR) scheme developed in **Chapter 2** (see also ref. [3]) will be readily applied to solve the posed quasi-static and dynamic *initial-boundary value* problems.

Consider the quasi-static equilibrium equation that represents a system of *zero order differential or algebraic/transcendental* equations

$$(22) \quad \Psi(t, \mathbf{a}) = \mathbf{P}(\mathbf{a}) + \mathbf{f} = \mathbf{0}, \quad \mathbf{a}(t_n) = \boldsymbol{\eta}, \quad t \in [t_n, t_{n+1}],$$

where $\boldsymbol{\eta}$ represents the initial displacement vectors and $\Psi(t, \mathbf{a})$ is the residual vector for the quasi-static problem. The solution to the finite element problem is sought at discrete instants of time. Initial conditions are assumed known at time t_n , and in the time interval $[t_n, t_{n+1}]$, the time increment Δt , and load increment $\Delta \mathbf{f}$, are given. The problem consists of finding a solution at time t_{n+1} . Therefore, the function Ψ does not depend on time explicitly. *Note* also that for the quasi-static problem, time is simply used as a counter for a solution sequence of events, and Δt is a pseudo-time scalar multiplier.

The non-linear time-independent equation may be rewritten in the discrete form as

$$(23) \quad \Psi(\mathbf{a}_{n+1}) = \mathbf{P}(\mathbf{a}_{n+1}) + \mathbf{f}_{n+1} = \mathbf{0}.$$

The solution algorithm provided by the generalised Newton-Raphson (GNR) scheme developed in **Chapter 2** (equations (28) and (29)) satisfies the following pair of equations:

$$(24) \quad \Psi_{n+1}^{i+1} = \Psi_{n+1}^i + \Delta \mathbf{a}^T \left\{ (1-\beta) \left(\frac{\partial \Psi}{\partial \mathbf{a}} \right) \Big|_{\mathbf{a}_{n+1}^i} + \beta \left(\frac{\partial \Psi}{\partial \mathbf{a}} \right) \Big|_{\mathbf{a}_{n+1}^{i+1}} \right\} = \mathbf{0}, \quad \beta \in [0,1],$$

$$(25) \quad \mathbf{a}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i + \Delta \mathbf{a}, \quad i = 0, 1, \dots$$

The superscript $(.)^i$ denotes an iteration counter within the time step counter $(.)_n$, and the coefficient of $\Delta \mathbf{a}^T$ is the generalised Jacobean that may be written as

$$(26) \quad \mathbf{K}_{n+1}^{i+1} = (1-\beta) \mathbf{S}_{n+1}^i + \beta \mathbf{S}_{n+1}^{i+1},$$

where the stiffness matrix is given by

$$(27) \quad \mathbf{S} = \frac{\partial \Psi(\mathbf{a})}{\partial \mathbf{a}} = \frac{\partial \mathbf{P}(\mathbf{a})}{\partial \mathbf{a}}.$$

With the exception of the case when the elastoplastic body is undergoing flow, the stiffness matrix will be positive definite; it then follows that

$$(28) \quad \mathbf{a}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i - [\mathbf{K}_{n+1}^{i+1}]^{-1} \Psi_{n+1}^i, \quad i = 0, 1, \dots$$

The generalised Newton-Raphson (GNR) scheme is implicit in β , and the conventional Newton-Raphson (CNR) scheme is a special case of the former and may be recovered when $\beta = 0$. The CNR scheme converges quadratically providing the initial trial values are close to the solution values. The CNR scheme may be written as

$$(29) \quad \mathbf{a}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i - [\mathbf{S}_{n+1}^i]^{-1} \Psi_{n+1}^i, \quad i = 0, 1, \dots,$$

where \mathbf{S}_{n+1}^i is the consistent tangent stiffness or Jacobean matrix.

Consider the dynamic *initial-boundary value* problem that satisfies the boundary data together with the conditions

$$(30) \quad \Psi(t, \mathbf{a}, \ddot{\mathbf{a}}) = \mathbf{M}(\mathbf{a})\ddot{\mathbf{a}} + \mathbf{P}(\mathbf{a}) + \mathbf{f} = \mathbf{0}, \quad \mathbf{a}(t_n) = \boldsymbol{\eta}, \quad \dot{\mathbf{a}}(t_n) = \dot{\boldsymbol{\eta}}, \quad t \in [t_n, t_{n+1}],$$

where $\boldsymbol{\eta}$ and $\dot{\boldsymbol{\eta}}$ represent the initial displacement and velocity vectors and $\Psi(t, \mathbf{a}, \ddot{\mathbf{a}})$ is the residual vector for the dynamic problem. The non-linear time-dependent equation may be rewritten in the discrete form as

$$(31) \quad \Psi(\mathbf{a}_{n+1}, \ddot{\mathbf{a}}_{n+1}) = \mathbf{M}(\mathbf{a}_{n+1})\ddot{\mathbf{a}}_{n+1} + \mathbf{P}(\mathbf{a}_{n+1}) + \mathbf{f}_{n+1} = \mathbf{0}.$$

The following two methods for solving second order differential equations are discussed in **Chapter 2**:

- The generalised Newton-Raphson (GNR) scheme combined with the first of the forward-backward (FB) difference schemes, known as the GNR-FB1 scheme, and
- The generalised Newton-Raphson (GNR) scheme combined with the second of the forward-backward (FB) difference schemes, known as the GNR-FB2 scheme.

1. The GNR-FB1 Scheme

The solution algorithm provided by the GNR-FB1 scheme developed in **Chapter 2** (equations (35) to (39)) satisfies the following pair of equations:

$$\begin{aligned}
 \Psi_{n+1}^{i+1} = & \Psi_{n+1}^i + \Delta \mathbf{a}^T \left\{ (1-\beta_1) \left(\frac{\partial \Psi}{\partial \mathbf{a}} \right) \Big|_{\mathbf{a}_{n+1}^i} + \beta_1 \left(\frac{\partial \Psi}{\partial \mathbf{a}} \right) \Big|_{\mathbf{a}_{n+1}^{i+1}} + \right. \\
 & \left. \frac{2}{\Delta t^2 \alpha_2 (2\alpha_1 - \alpha_2)} \left[(1-\beta_3) \left(\frac{\partial \Psi}{\partial \ddot{\mathbf{a}}} \right) \Big|_{\ddot{\mathbf{a}}_{n+1}^i} + \beta_3 \left(\frac{\partial \Psi}{\partial \ddot{\mathbf{a}}} \right) \Big|_{\ddot{\mathbf{a}}_{n+1}^{i+1}} \right] \right\} \\
 (32) \quad & - \left[(1-\beta_3) \left(\frac{\partial \Psi}{\partial \ddot{\mathbf{a}}} \right) \Big|_{\ddot{\mathbf{a}}_{n+1}^i} + \beta_3 \left(\frac{\partial \Psi}{\partial \ddot{\mathbf{a}}} \right) \Big|_{\ddot{\mathbf{a}}_{n+1}^{i+1}} \right] \times \\
 & \left[\frac{2}{\Delta t \alpha_2 (2\alpha_1 - \alpha_2)} \dot{\mathbf{a}}_{n+1}^i + \frac{1}{\alpha_2 (2\alpha_1 - \alpha_2)} \ddot{\mathbf{a}}_{n+1}^i \right] = \mathbf{0},
 \end{aligned}$$

$$\beta_1, \beta_3 \in [0, 1],$$

$$(33) \quad \mathbf{a}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i + \Delta \mathbf{a}, \quad i = 0, 1, \dots$$

The velocity and acceleration vectors are updated using the FB1 scheme developed in **Chapter 2** as described in Section 3.3.4 below.

For problems of practical consequence, the change of mass with respect to displacement is considerably less than the change of internal forces with respect to displacement, so that

$$(34) \quad \frac{\partial \Psi}{\partial \mathbf{a}} \cong \frac{\partial \mathbf{P}(\mathbf{a})}{\partial \mathbf{a}} = \mathbf{S}, \quad \frac{\partial \mathbf{M}(\mathbf{a})}{\partial \mathbf{a}} \ll \frac{\partial \mathbf{P}(\mathbf{a})}{\partial \mathbf{a}},$$

and $\frac{\partial \Psi}{\partial \ddot{\mathbf{a}}} = \mathbf{M}$. For a uniaxial example, this is illustrated graphically in Figure 2.

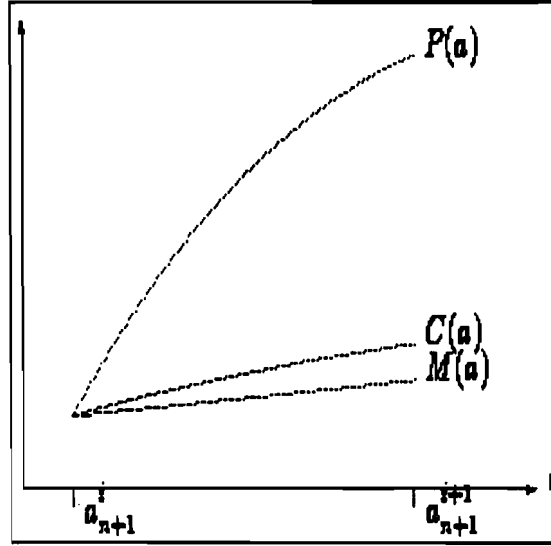


Figure 2 Assumptions: (i) $\partial \mathbf{M} / \partial \mathbf{a} \ll \partial \mathbf{P} / \partial \mathbf{a}$ and (ii) $\partial \mathbf{C} / \partial \mathbf{a} \ll \partial \mathbf{P} / \partial \mathbf{a}$ where $\mathbf{C} \equiv$ viscous damping.

The assumption made is valid for most applications, especially in solid mechanics where $\mathbf{M}(\mathbf{a}_{n+1}^i) = \mathbf{M}(\mathbf{a}_{n+1}^{i+1})$.

Substituting these matrices into the above equation (32), the generalised Jacobean may be written as

$$(35) \quad \mathbf{K}_{n+1}^{i+1} = \left\{ (1 - \beta_1) \mathbf{S}_{n+1}^i + \beta_1 \cdot \mathbf{S}_{n+1}^{i+1} + \frac{2}{\Delta t^2 \alpha_2 (2\alpha_1 - \alpha_2)} [(1 - \beta_3) \mathbf{M}_{n+1}^i + \beta_3 \cdot \mathbf{M}_{n+1}^{i+1}] \right\},$$

and the GNR-FB1 scheme reduces to

$$(36) \quad \Psi_{n+1}^{i+1} = \Psi_{n+1}^i + \Delta \mathbf{a}^T \mathbf{K}_{n+1}^{i+1} - [(1 - \beta_3) \mathbf{M}_{n+1}^i + \beta_3 \cdot \mathbf{M}_{n+1}^{i+1}] \times \left[\frac{2}{\Delta t \alpha_2 (2\alpha_1 - \alpha_2)} \dot{\mathbf{a}}_{n+1}^i + \frac{1}{\alpha_2 (2\alpha_1 - \alpha_2)} \ddot{\mathbf{a}}_{n+1}^i \right] = \mathbf{0}$$

$$(37) \quad \mathbf{a}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i + \Delta \mathbf{a}, \quad i = 0, 1, \dots$$

When the stiffness matrix is positive definite or semi-definite and the mass matrix is positive definite the generalised Jacobean has an inverse, and the equations (36) and (37) may be combined as follows:

$$(38) \quad \mathbf{a}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i - [\mathbf{K}_{n+1}^{i+1}]^{-1} \cdot \left\{ \Psi_{n+1}^i - \left[(1 - \beta_3) \mathbf{M}_{n+1}^i + \beta_3 \cdot \mathbf{M}_{n+1}^{i+1} \right] \times \left[\frac{2}{\Delta t \alpha_2 (2\alpha_1 - \alpha_2)} \dot{\mathbf{a}}_{n+1}^i + \frac{1}{\alpha_2 (2\alpha_1 - \alpha_2)} \ddot{\mathbf{a}}_{n+1}^i \right] \right\},$$

$$i = 0, 1, \dots$$

where Ψ_{n+1}^{i+1} is set to zero. The GNR-FB1 scheme is implicit in β_1 and β_3 , and the CNR-FB1 scheme may be recovered from the former by setting $\beta_1 = \beta_3 = 0$, resulting in

$$(39) \quad \mathbf{a}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i - [\mathbf{K}_{n+1}^i]_{\mathcal{T}}^{-1} \times \left\{ \Psi_{n+1}^i - \mathbf{M}_{n+1}^i \cdot \left[\frac{2}{\Delta t \alpha_2 (2\alpha_1 - \alpha_2)} \dot{\mathbf{a}}_{n+1}^i + \frac{1}{\alpha_2 (2\alpha_1 - \alpha_2)} \ddot{\mathbf{a}}_{n+1}^i \right] \right\}, \quad i = 0, 1, \dots$$

where $[\mathbf{K}_{n+1}^i]_{\mathcal{T}}$ is the consistent tangent stiffness or Jacobean matrix for the CNR-FB1 scheme, given by

$$(40) \quad [\mathbf{K}_{n+1}^i]_{\mathcal{T}} = \mathbf{S}_{n+1}^i + \frac{2}{\Delta t^2 \alpha_2 (2\alpha_1 - \alpha_2)} \mathbf{M}_{n+1}^i.$$

2. The GNR-FB2 Scheme

Using similar techniques, the GNR-FB2 scheme developed in **Chapter 2** (equations (35) to (39)) may be written as

$$(41) \quad \mathbf{a}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i - [\mathbf{K}_{n+1}^{i+1}]^{-1} \cdot \left\{ \Psi_{n+1}^i - \left[(1 - \beta_3) \mathbf{M}_{n+1}^i + \beta_3 \cdot \mathbf{M}_{n+1}^{i+1} \right] \cdot \left[\frac{2}{\Delta t \alpha_2^2} \dot{\mathbf{a}}_{n+1}^i + \frac{1}{\alpha_2^2} \ddot{\mathbf{a}}_{n+1}^i \right] \right\}, \quad i = 0, 1, \dots$$

with the corresponding CNR-FB2 scheme obtained by setting $\beta_1 = \beta_3 = 0$,

$$(42) \quad \mathbf{a}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i - [\mathbf{K}_{n+1}^i]_T^{-1} \cdot \left\{ \Psi_{n+1}^i - \mathbf{M}_{n+1}^i \cdot \left[\frac{2}{\Delta t \alpha_2^2} \dot{\mathbf{a}}_{n+1}^i + \frac{1}{\alpha_2^2} \ddot{\mathbf{a}}_{n+1}^i \right] \right\}, \quad i = 0, 1, \dots$$

where $[\mathbf{K}_{n+1}^i]_T$ is the consistent tangent stiffness or Jacobean matrix for the CNR-FB2 scheme, given by

$$(43) \quad [\mathbf{K}_{n+1}^i]_T = \mathbf{S}_{n+1}^i + \frac{2}{\Delta t^2 \alpha_2^2} \mathbf{M}_{n+1}^i.$$

3.3.3 Generalised Newton-Raphson Algorithm

The GNR algorithm for the quasi-static and dynamic *initial-boundary value* problems may be described by the following steps [3]:

- 1 Specify $\alpha_1, \alpha_2, \beta_1, \beta_3$.
- 2 Specify Δt .
- 3 Specify time, the number of time steps and set $i = 0$.
- 4 For each time step, t_{n+1} :
 - (i) Compute predictors: $\bar{\mathbf{a}}_{n+1}^{i+1}, \bar{\dot{\mathbf{a}}}_{n+1}^{i+1}, \bar{\ddot{\mathbf{a}}}_{n+1}^{i+1}$.
 - (ii) Compute $\bar{\Psi}_{n+1}^{i+1}$.
 - (iii) In the first iteration ($i = 0$), compute \mathbf{K}_{n+1}^1 ; $\beta_1 = \beta_3 = 0$. In the second and subsequent iterations, compute \mathbf{K}_{n+1}^{i+1} using specified scalars (β_j).
 - (iv) Solve for $\Delta \mathbf{a}$.
 - (v) Update $\mathbf{a}_{n+1}^{i+1}, \dot{\mathbf{a}}_{n+1}^{i+1}, \ddot{\mathbf{a}}_{n+1}^{i+1}$.
- 5 Check convergence:
 - (i) If satisfied terminate iteration.
 - (ii) If not satisfied set $i = i + 1$ and repeat step 4.
- 6 Output solution information if needed.
- 7 Check time limit:
 - (i) If $n \geq$ maximum number, stop; else:
 - (ii) If $n <$ maximum number, go to step 3.

Implemented in the *explicit* form, the GNR algorithm includes the conventional Newton-Raphson (CNR) scheme as a special case with $\beta_1 = \beta_3 = 1$, unlike in the *implicit* mode where the CNR scheme is recovered from the GNR scheme when $\beta_1 = \beta_3 = 0$. For definitions of the terms *explicit* and *implicit*, see **Chapter 5**, Section **5.2.2**.

The approximate solution to the quasi-static and dynamic *initial-boundary value* problems in the form of a displacement vector field $\mathbf{u}(x, t)$, and a stress field $\sigma(x, t)$ is provided in Steps 4 (v)-6, together with all other updates. Details of the variables in the algorithm are described in Section **3.3.4** below.

3.3.4 Predictor-Corrector Steps

The derivations of the predictor-corrector steps for the generalised and conventional Newton-Raphson (GNR and CNR) algorithms are now described in detail. Given [3] the predicted displacement, velocity and acceleration vectors for the GNR/CNR-FB1 scheme as:

$$(44) \quad \bar{\mathbf{a}}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i,$$

$$(45) \quad \bar{\dot{\mathbf{a}}}_{n+1}^{i+1} = \left(1 - \frac{2\alpha_1}{\alpha_2(2\alpha_1 - \alpha_2)}\right) \dot{\mathbf{a}}_{n+1}^i + \Delta t \left(1 - \frac{\alpha_1}{\alpha_2(2\alpha_1 - \alpha_2)}\right) \ddot{\mathbf{a}}_{n+1}^i,$$

$$(46) \quad \bar{\ddot{\mathbf{a}}}_{n+1}^{i+1} = -\frac{2}{\Delta t \alpha_2(2\alpha_1 - \alpha_2)} \dot{\mathbf{a}}_{n+1}^i + \left(1 - \frac{1}{\alpha_2(2\alpha_1 - \alpha_2)}\right) \ddot{\mathbf{a}}_{n+1}^i,$$

and the predicted displacement, velocity and acceleration vectors for the GNR/CNR-FB2 scheme as:

$$(47) \quad \bar{\mathbf{a}}_{n+1}^{i+1} = \mathbf{a}_{n+1}^i$$

$$(48) \quad \bar{\dot{\mathbf{a}}}_{n+1}^{i+1} = \left(1 - \frac{2\alpha_1}{\alpha_2}\right) \dot{\mathbf{a}}_{n+1}^i + \Delta t \left(1 - \frac{\alpha_1}{\alpha_2}\right) \ddot{\mathbf{a}}_{n+1}^i,$$

$$(49) \quad \bar{\ddot{\mathbf{a}}}_{n+1}^{i+1} = -\frac{2}{\Delta t \alpha_2} \dot{\mathbf{a}}_{n+1}^i + \left(1 - \frac{1}{\alpha_2}\right) \ddot{\mathbf{a}}_{n+1}^i,$$

the total residual may be written in the expanded predictor-corrector form as

$$(50) \quad \begin{aligned} \Psi_{n+1}^{i+1} = & \bar{\Psi}_{n+1}^{i+1} + \Delta \mathbf{a}^T \left[(1 - \beta_1) \left(\frac{\partial \Psi}{\partial \mathbf{a}} \right) \Big|_{\bar{\mathbf{a}}_{n+1}^{i+1}} + \beta_1 \cdot \left(\frac{\partial \Psi}{\partial \mathbf{a}} \right) \Big|_{\mathbf{a}_{n+1}^{i+1}} \right] \\ & + \Delta \dot{\mathbf{a}}^T \left[(1 - \beta_3) \left(\frac{\partial \Psi}{\partial \dot{\mathbf{a}}} \right) \Big|_{\bar{\dot{\mathbf{a}}}_{n+1}^{i+1}} + \beta_3 \cdot \left(\frac{\partial \Psi}{\partial \dot{\mathbf{a}}} \right) \Big|_{\dot{\mathbf{a}}_{n+1}^{i+1}} \right] = \mathbf{0}, \\ & \beta_1, \beta_3 \in [0, 1] \end{aligned}$$

where the residual vector for the predictor phase may be written in the form

$$(51) \quad \bar{\Psi}_{n+1}^{i+1} = \mathbf{M}_{\bar{\mathbf{a}}_{n+1}^{i+1}} \bar{\mathbf{a}}_{n+1}^{i+1} + \mathbf{P}_{\bar{\dot{\mathbf{a}}}_{n+1}^{i+1}} \bar{\dot{\mathbf{a}}}_{n+1}^{i+1} + \mathbf{f}_{n+1}.$$

Note that the external forces remain fixed within the time step. The corrector displacement, velocity and acceleration vectors for the GNR/CNR-FB1 scheme are given [3] by

$$(52) \quad \mathbf{a}_{n+1}^{i+1} = \bar{\mathbf{a}}_{n+1}^{i+1} + \Delta \mathbf{a},$$

$$(53) \quad \dot{\mathbf{a}}_{n+1}^{i+1} = \bar{\dot{\mathbf{a}}}_{n+1}^{i+1} + \frac{2\alpha_1}{\Delta t \alpha_2 (2\alpha_1 - \alpha_2)} \Delta \mathbf{a} = \bar{\dot{\mathbf{a}}}_{n+1}^{i+1} + \Delta \dot{\mathbf{a}},$$

$$(54) \quad \ddot{\mathbf{a}}_{n+1}^{i+1} = \bar{\ddot{\mathbf{a}}}_{n+1}^{i+1} + \frac{2}{\Delta t^2 \alpha_2 (2\alpha_1 - \alpha_2)} \Delta \mathbf{a} = \bar{\ddot{\mathbf{a}}}_{n+1}^{i+1} + \Delta \ddot{\mathbf{a}},$$

and the corrector displacement, velocity and acceleration vectors for the GNR/CNR-FB2 scheme are given by

$$(55) \quad \mathbf{a}_{n+1}^{i+1} = \bar{\mathbf{a}}_{n+1}^{i+1} + \Delta \mathbf{a},$$

$$(56) \quad \dot{\mathbf{a}}_{n+1}^{i+1} = \bar{\dot{\mathbf{a}}}_{n+1}^{i+1} + \frac{2\alpha_1}{\Delta t \alpha_2^2} \Delta \mathbf{a} = \bar{\dot{\mathbf{a}}}_{n+1}^{i+1} + \Delta \dot{\mathbf{a}},$$

$$(57) \quad \ddot{\mathbf{a}}_{n+1}^{i+1} = \bar{\ddot{\mathbf{a}}}_{n+1}^{i+1} + \frac{2}{\Delta t^2 \alpha_2^2} \Delta \mathbf{a} = \bar{\ddot{\mathbf{a}}}_{n+1}^{i+1} + \Delta \ddot{\mathbf{a}}.$$

Substituting these corrector vectors into the equation for the total residual results in

$$(58) \quad \Psi_{n+1}^{i+1} = \bar{\Psi}_{n+1}^{i+1} + \Delta \mathbf{a}^T \left\{ \left[(1-\beta_1) \left(\frac{\partial \Psi}{\partial \mathbf{a}} \right) \Big|_{\bar{\mathbf{a}}_{n+1}^{i+1}} + \beta_1 \cdot \left(\frac{\partial \Psi}{\partial \mathbf{a}} \right) \Big|_{\mathbf{a}_{n+1}^{i+1}} \right] \right. \\ \left. + C1 \times \left[(1-\beta_3) \left(\frac{\partial \Psi}{\partial \dot{\mathbf{a}}} \right) \Big|_{\bar{\dot{\mathbf{a}}}_{n+1}^{i+1}} + \beta_3 \cdot \left(\frac{\partial \Psi}{\partial \dot{\mathbf{a}}} \right) \Big|_{\dot{\mathbf{a}}_{n+1}^{i+1}} \right] \right\} = \mathbf{0}, \quad \beta_1, \beta_3 \in [0, 1]$$

where C1 is given by

$$(59) \quad C1 = \frac{2}{\Delta t^2 \alpha_2 (2\alpha_1 - \alpha_2)}, \quad \text{FB1 scheme},$$

$$(60) \quad C1 = \frac{2}{\Delta t^2 \alpha_2^2}, \quad \text{FB2 scheme}.$$

The coefficient of $\Delta \mathbf{a}^T$ is the generalised Jacobean for the generalised Newton-Raphson (GNR) predictor-corrector scheme.

Note:

- The conventional Newton-Raphson predictor-corrector scheme is recovered from the above scheme by setting $\beta_1 = \beta_3 = 0$.

- The quasi-static problem is recovered from the dynamic one by setting $\mathbf{M} = \mathbf{0}$ and $\dot{\mathbf{a}} = \ddot{\mathbf{a}} = \mathbf{0}$, so that only one algorithm need be discussed.

The similarities in the vectors for the GNR/CNR-FB1 and GNR/CNR-FB2 schemes ensure that one algorithm can be used for both schemes. This facility made it possible for the algorithm to be successfully implemented in **PCFEAP** [2], a PC-based Finite Element Analysis Program.

3.3.5. Stress Update

During loading or neutral loading, the plastic consistency may be enforced at the end of the time step [4,5] according to the Kuhn-Tucker loading and unloading relations written in the form

$$(61) \quad \Delta\lambda \geq 0, \quad \phi(\boldsymbol{\sigma}_{n+1}, \boldsymbol{\kappa}_{n+1}) \leq 0, \quad \Delta\lambda \cdot \phi(\boldsymbol{\sigma}_{n+1}, \boldsymbol{\kappa}_{n+1}) = 0,$$

and the stress field is updated in a procedure known as the *return mapping* algorithm as follows:

$$(62) \quad \boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_n + \mathbf{D}[\Delta\boldsymbol{\varepsilon} - \Delta\lambda \cdot \nabla\phi(\boldsymbol{\sigma}_{n+1}, \boldsymbol{\kappa}_{n+1})],$$

where the current total strain increment is known and is given by

$$(63) \quad \Delta\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_{n+1} - \boldsymbol{\varepsilon}_n.$$

Alternatively, the plastic consistency may be enforced at the generalised mid-point [1] in a generalised mid-point (GMP) rule according to the Kuhn-Tucker loading and unloading relations written in the form

$$(64) \quad \Delta\lambda \geq 0, \quad \phi(\boldsymbol{\sigma}_{n+\alpha}, \boldsymbol{\kappa}_{n+\alpha}) \leq 0, \quad \Delta\lambda \cdot \phi(\boldsymbol{\sigma}_{n+\alpha}, \boldsymbol{\kappa}_{n+\alpha}) = 0, \quad \alpha \in [0, 1],$$

with the stress updated by the relation

$$(65) \quad \boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_n + \mathbf{D}[\Delta\boldsymbol{\varepsilon} - \Delta\lambda \cdot \nabla\phi(\boldsymbol{\sigma}_{n+\alpha}, \boldsymbol{\kappa}_{n+\alpha})].$$

This algorithm is second order accurate for $\alpha = \frac{1}{2}$. The Euler backward difference algorithm is a special case of the GMP algorithm with $\alpha = 1$, and if implemented according to [1], inherits the properties of the GMP algorithm, except that it is first order accurate. The algorithms presented in [4,5] also result in the Euler backward difference algorithm when $\alpha = 1$ and are also second order accurate for $\alpha = \frac{1}{2}$. However, the algorithm given in [4] results in a non-symmetric tangent modulus except when $\alpha = 1$.

The plastic consistency may also be enforced at the end of the time step in the generalised trapezoidal rule according to the Kuhn-Tucker loading and unloading relations written in the form

$$(66) \quad \lambda \geq 0, \quad \phi(\boldsymbol{\sigma}_{n+1}, \boldsymbol{\kappa}_{n+1}) \leq 0, \quad \lambda \cdot \phi(\boldsymbol{\sigma}_{n+1}, \boldsymbol{\kappa}_{n+1}) = 0.$$

The plastic strain increment may be approximated by the generalised trapezoidal (GT) scheme developed herein as follows:

$$(67) \quad \begin{aligned} \Delta\boldsymbol{\varepsilon}^{ep} &= \Delta t \cdot [(1-\alpha)\dot{\boldsymbol{\varepsilon}}_n^{ep} + \alpha \cdot \dot{\boldsymbol{\varepsilon}}_{n+1}^{ep}], \quad \alpha \in [0, 1] \\ &= \Delta t (\dot{\boldsymbol{\varepsilon}}_n^{ep} + \alpha \cdot \Delta\dot{\boldsymbol{\varepsilon}}^{ep}) \end{aligned},$$

where the plastic strain rates and strain rate increment are given by

$$(68) \quad \dot{\boldsymbol{\varepsilon}}_n^{ep} = \lambda_n \left. \frac{\partial \phi}{\partial \boldsymbol{\sigma}} \right|_n, \quad \dot{\boldsymbol{\varepsilon}}_{n+1}^{ep} = \lambda_{n+1} \left. \frac{\partial \phi}{\partial \boldsymbol{\sigma}} \right|_{n+1}, \quad \Delta\dot{\boldsymbol{\varepsilon}}^{ep} = \dot{\boldsymbol{\varepsilon}}_{n+1}^{ep} - \dot{\boldsymbol{\varepsilon}}_n^{ep},$$

resulting in the stress update equation

$$(69) \quad \sigma_{n+1} = \sigma_n + \mathbf{D} \left[\Delta \varepsilon - (1 - \alpha) \tilde{\lambda}_n \nabla \phi(\sigma_n, \kappa_n) - \alpha \cdot \tilde{\lambda}_{n+1} \nabla \phi(\sigma_{n+1}, \kappa_{n+1}) \right], \\ \alpha \in [0, 1]$$

where $\tilde{\lambda} = \Delta t \lambda$ is interpreted as a Lagrange multiplier over the time interval and all the quantities at time t_n are known. This algorithm resembles the one implemented in [5], and may be identical to the generalised mid-point rule if the identity

$$(70) \quad \Delta \lambda \nabla \phi(\sigma_{n+\alpha}, \kappa_{n+\alpha}) \equiv (1 - \alpha) \tilde{\lambda}_n \nabla \phi(\sigma_n, \kappa_n) + \alpha \cdot \tilde{\lambda}_{n+1} \nabla \phi(\sigma_{n+1}, \kappa_{n+1}),$$

can be established even solely based on the mode of implementation. The LHS of the equation was derived under the incremental form of the constitutive equations and may be manipulated to yield the expression on the RHS of the equation which has been derived under the rate form of the constitutive equations. The mathematical proof [5] is beyond the scope of this work. Clearly, the implementation of the mid-point rule implies the satisfaction of

$$(71) \quad \nabla \phi(\sigma_{n+\alpha}, \kappa_{n+\alpha}) = (1 - \alpha) \nabla \phi(\sigma_n, \kappa_n) + \alpha \cdot \nabla \phi(\sigma_{n+1}, \kappa_{n+1}), \\ (n + \alpha) \in [n, n + 1],$$

where the quantity on the LHS of the equation has been linearised in the interval.

3.3.6. Stability Analysis

The formulation and non-linear stability analysis of the generalised mid-point algorithm for plasticity (and viscoplasticity) are given in [1]. The algorithm is *B*-stable for $\alpha \in [\frac{1}{2}, 1]$. The static [6] and dynamic [7] analyses of elastoplastic systems are also dealt with in the context of the internal variable framework. Both papers

consider the generalised mid-point algorithm, and explore the notion of B -stability among other things. The stability analysis of the generalised trapezoidal rule discussed herein will also be carried out in terms of the internal variable formulations in the next chapter.

3.4. CONCLUSIONS

Finite element formulations of the static and dynamic *initial-boundary value* evolution problems have been considered in the classical framework and solution algorithms have been developed using the generalised trapezoidal (GT) rule and generalised Newton-Raphson (GNR) scheme.

The next chapter considers the internal variable framework of the static and dynamic *initial-boundary value* evolution problems and, among other things, the stability analysis of the generalised trapezoidal rule will be carried out.

REFERENCES

- [1] Simo J C, Govindjee S, "Non-linear B-stability and symmetry preserving return mapping algorithms for plasticity and viscoplasticity", *International Journal for Numerical Methods in Engineering*, **31**, 151-176, (1991).
- [2] Zienkiewicz O C, Taylor R L, *The Finite Element Method, Volume 2, Solid and Fluid Mechanics Dynamics and Non-linearity*, 4th Edition, 1991.
- [3] Kaunda M A E, Martin J B, "Finite element solution algorithms for non-linear time-dependent/independent problems in solid mechanics", In *Proc. 19th South African Symposium on Numerical Mathematics (SANUM)*, 95-117, (1993).
- [4] Ortiz M, Popov E P, "Accuracy and stability of integration algorithms for elastoplastic constitutive relations", *International Journal for Numerical Methods in Engineering*, **21**, 1561-1576, (1985).

- [5] Rencontre L J, Caddemi S, Martin J B, “The relationship between the generalised mid-point and trapezoidal rules in incremental elasto-plasticity”, *Comp. Meth. Appl. Mech.*, **96**, 201-212, (1992).
- [6] Reddy B D, Martin J B, “Algorithms for the solution of internal variable problems in plasticity”, *Computer Methods in Applied Mechanics and Engineering*, **93**, 253-273, (1991).
- [7] Corigliano A, Perego U, “Unconditionally stable mid-point time integration in elastic-plastic dynamics”, *Rendi. Acc. Lincei*, s. 9, **1**, 367-376, (1990).

CHAPTER 4

THE EVOLUTION PROBLEM: INTERNAL VARIABLE FRAMEWORK

4.1. INTRODUCTION

Continuum and finite element formulations of the static and dynamic *initial-boundary value* evolution problems will be considered using the internal variable framework. An internal variable approach describes the constitutive behaviour in terms of a set of conjugate static and kinematic internal variables. A number of pioneering articles [1-8] have provided the framework of internal variables with a sound thermodynamic basis. This chapter will consider some of the recent works [9-16] and apply the time-discretisation and stability analyses. Without loss of generality, attention will be focused on problems that consist of an elastoplastic von Mises material with linear kinematic hardening.

Solution algorithms will be developed using the generalised trapezoidal (GT) rule and generalised Newton-Raphson (GNR) scheme developed in **Chapter 2**.

Section 4.2 considers the continuum problem of evolution in terms of the internal variable framework, whereas Section 4.3, the finite element equations and algorithms. Solution algorithms are developed in Sections 4.3.1 and 4.3.2 and the generalised Newton-Raphson algorithm is presented in Section 4.3.3. Section 4.3.4 is a discussion of the stability analysis of the algorithms using energy-like norms that form functions similar to the Lyapunov's functions [17-18].

4.2. THE CONTINUUM PROBLEM OF EVOLUTION

This section considers quasi-static and dynamic *initial-boundary value* continuum problems of evolution, using the internal variable framework, characterised by the small strain rate-independent plasticity under isothermal conditions. Geometric non-linearity, associated with a non-linear strain-displacement relationship that is characterised by large displacement or strain will not be considered.

4.2.1. Elastoplastic Constitutive Equations

Consider an element of a structure that may be infinitesimal or a conventionally defined finite element. Let the natural strains be $\boldsymbol{\varepsilon}$, and the conjugate generalised stresses be $\boldsymbol{\sigma}$. A stiffness matrix \mathbf{C} , which is assumed symmetric and positive definite, characterises the linear elastic isothermal behaviour. An additional vector of internal variables $\boldsymbol{\lambda}$ describes plastic behaviour [7,9-11,14-16]. A free energy density, $f = f(\boldsymbol{\varepsilon}, \boldsymbol{\lambda})$, that is assumed homogeneous and quadratic in $\boldsymbol{\varepsilon}$ and $\boldsymbol{\lambda}$ is introduced. This may be described as the sum of the elastic strain energy and stored strain energy due to structural rearrangements at the micro-scale which represents the Helmholtz free energy as follows:

$$(1) \quad f = \frac{1}{2} \boldsymbol{\varepsilon}^T \mathbf{C} \boldsymbol{\varepsilon} + \boldsymbol{\varepsilon}^T \mathbf{G} \boldsymbol{\lambda} + \frac{1}{2} \boldsymbol{\lambda}^T \mathbf{H} \boldsymbol{\lambda} .$$

The matrices \mathbf{C} , \mathbf{G} , and \mathbf{H} , defined below, are material properties. The stress, $\boldsymbol{\sigma}$, conjugate to the strain, $\boldsymbol{\varepsilon}$, may be described through

$$(2) \quad \boldsymbol{\sigma} = \frac{\partial f}{\partial \boldsymbol{\varepsilon}} = \mathbf{C} \boldsymbol{\varepsilon} + \mathbf{G} \boldsymbol{\lambda} .$$

The relationship describing the thermodynamic force χ , conjugate to the internal variable λ is given by

$$(3) \quad -\chi = \frac{\partial f}{\partial \lambda} = \mathbf{G}^T \boldsymbol{\varepsilon} + \mathbf{H} \lambda .$$

Under these circumstances the second law of thermodynamics requires that

$$(4) \quad \dot{\lambda}^T \chi \geq 0 ,$$

which represents the specific dissipation function. The matrices describing material properties are as follows:

$$(5) \quad \mathbf{C} = \frac{\partial \sigma}{\partial \boldsymbol{\varepsilon}} = \frac{\partial^2 f}{\partial \boldsymbol{\varepsilon} \partial \boldsymbol{\varepsilon}}, \quad \mathbf{H} = -\frac{\partial \chi}{\partial \lambda} = \frac{\partial^2 f}{\partial \lambda \partial \lambda} ,$$

$$(6) \quad \mathbf{G} = \frac{\partial \sigma}{\partial \lambda} = \frac{\partial^2 f}{\partial \lambda \partial \boldsymbol{\varepsilon}}, \quad \mathbf{G}^T = -\frac{\partial \chi}{\partial \boldsymbol{\varepsilon}} = \frac{\partial^2 f}{\partial \boldsymbol{\varepsilon} \partial \lambda} ,$$

where \mathbf{C} is a positive definite symmetric matrix of elasticity constants; and \mathbf{H} , a positive semi-definite matrix of hardening constants. In addition, it is necessary to specify the evolution law of λ , in the form:

$$(7) \quad \dot{\lambda} = \mathbf{F}(\chi) .$$

This describes the rate of change of the internal variable, $\dot{\lambda}$, as a function of χ only [11-12,14]. For rate independent plasticity, the evolution law [11-12,14] takes the form

$$(8) \quad \chi \in \partial D(\dot{\lambda}) .$$

D is the dissipation function which is assumed convex and positively homogeneous and of degree one. If D is smooth at $\dot{\lambda}$, the sub-differential becomes

$$(9) \quad \partial D(\dot{\lambda}) = \{ \nabla D(\dot{\lambda}) \} .$$

Conventionally, the postulated evolution law takes the form:

$$(10) \quad \dot{\lambda} = v \frac{\partial \phi}{\partial \chi} .$$

$\phi(\chi) = k$ is the yield surface, k being a positive constant, and v is a non-negative scalar.

4.2.2. Dissipation Function and Hardening Models

A pseudo-potential or dissipation function describes the kinematic relationship between the rate of change of internal variables $\dot{\lambda}$ and the internal forces χ . It is directly related to the entropy production rate and is required by the second law of thermodynamics to be non-negative. Various illustrations of the dissipation function are given in [7,11-12]. Mechanical models for plastic behaviour are also presented in the references, as follows:

- Elastic, perfectly plastic,
- Linear kinematic hardening, and
- Linear isotropic hardening.

Detailed derivations of the piece-wise smooth dissipation and yield functions are given in [15-16]. The von Mises yield criterion for linear kinematic hardening takes the form:

$$(11) \quad \phi(\boldsymbol{\chi}) = |\boldsymbol{\chi}| \leq k ,$$

where k is a positive constant, and the dissipation function is given by

$$(12) \quad D(\dot{\boldsymbol{\lambda}}) = \chi_0 |\dot{\boldsymbol{\lambda}}| ; \quad \chi_0 = k .$$

Under the above conditions the rate of the internal variable becomes,

$$(13) \quad \dot{\boldsymbol{\lambda}} = \nu \frac{\boldsymbol{\chi}}{|\boldsymbol{\chi}|} .$$

Reference [14] shows how to ensure that $\dot{\boldsymbol{\lambda}}$ and $\boldsymbol{\lambda}$ represent the deviatoric plastic strain rate and plastic strain, respectively.

4.3. FINITE ELEMENT EQUATIONS AND ALGORITHMS

In this section, finite element formulations are considered, and the Helmholtz free energy is described in terms of vectors of nodal displacements (and internal variables) instead of strains (and internal variables). Nodal displacements are denoted by \mathbf{a} , and for dynamics, vectors of nodal velocities and accelerations are denoted by $\dot{\mathbf{a}}$, $\ddot{\mathbf{a}}$ respectively.

4.3.1. Quasi-static Elastoplasticity

In the finite element context, the Helmholtz free energy may be defined by

$$(14) \quad \hat{f} = \frac{1}{2} \mathbf{a}^T \mathbf{K} \mathbf{a} + \mathbf{a}^T \mathbf{G} \lambda + \frac{1}{2} \lambda^T \mathbf{H} \lambda ,$$

and for simplicity the symbol, $f \equiv \hat{f}$ will be used from now on. The generalised stiffness matrix is given by

$$(15) \quad \mathbf{K} = \frac{\partial^2 f}{\partial \mathbf{a} \partial \mathbf{a}} ,$$

and other generalised material properties are given by

$$(16) \quad \mathbf{G}' = \frac{\partial^2 f}{\partial \lambda \partial \mathbf{a}} , \quad \mathbf{G}'^T = \frac{\partial^2 f}{\partial \mathbf{a} \partial \lambda} , \quad \mathbf{H}' = \frac{\partial^2 f}{\partial \lambda \partial \lambda} .$$

Note that these quantities defined in the finite element context (in terms of nodal displacements), are similar to those defined in the continuum context (in terms of strains) in Section 4.2.1. For simplicity, the primes will be dropped and from now on $\mathbf{G}' \equiv \mathbf{G}$ and $\mathbf{H}' \equiv \mathbf{H}$.

Consider the Helmholtz free energy, $f(\mathbf{a} + \Delta \mathbf{a}, \lambda + \Delta \lambda)$, expanded using a Taylor series:

$$(17) \quad \begin{aligned} f(\mathbf{a} + \Delta \mathbf{a}, \lambda + \Delta \lambda) = & f(\mathbf{a}, \lambda) + \Delta \mathbf{a}^T \left(\frac{\partial f}{\partial \mathbf{a}} \right) \Big|_{(\mathbf{a}, \lambda)} + \Delta \lambda^T \left(\frac{\partial f}{\partial \lambda} \right) \Big|_{(\mathbf{a}, \lambda)} \\ & + \frac{1}{2} \Delta \mathbf{a}^T \left(\frac{\partial^2 f}{\partial \mathbf{a} \partial \mathbf{a}} \right) \Big|_{(\mathbf{a}, \lambda)} \Delta \mathbf{a} + \Delta \mathbf{a}^T \left(\frac{\partial^2 f}{\partial \mathbf{a} \partial \lambda} \right) \Big|_{(\mathbf{a}, \lambda)} \Delta \lambda + \frac{1}{2} \Delta \lambda^T \left(\frac{\partial^2 f}{\partial \lambda \partial \lambda} \right) \Big|_{(\mathbf{a}, \lambda)} \Delta \lambda , \end{aligned}$$

where \mathbf{a} denotes generalised nodal displacement vector; λ , vector of generalised internal variable; $\mathbf{f} = \partial f / \partial \mathbf{a}$, generalised applied force; $-\chi = \partial f / \partial \lambda$, generalised thermodynamic force conjugate to generalised internal variable λ .

From the equilibrium equations defined in the finite element context, the following collocation differential equation may be defined:

$$(18) \quad \begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{H} \end{bmatrix} \begin{Bmatrix} \mathbf{a}_j \\ \lambda_j \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_j \\ -\chi_j \end{Bmatrix} \quad j = n, (n+1) .$$

Substituting the collocation differential equation into the expanded equation (17) of the Helmholtz free energy results in a convex function:

$$(19) \quad f(\Delta \mathbf{a}, \Delta \lambda) = \frac{1}{2} \begin{Bmatrix} \Delta \mathbf{a} \\ \Delta \lambda \end{Bmatrix}^T \begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{H} \end{bmatrix} \begin{Bmatrix} \Delta \mathbf{a} \\ \Delta \lambda \end{Bmatrix} + \begin{Bmatrix} \Delta \mathbf{a} \\ \Delta \lambda \end{Bmatrix}^T \begin{Bmatrix} \mathbf{f}_n \\ -\chi_n \end{Bmatrix} ,$$

From this equation, a non-linear mathematical programming problem may be defined by the statement

$$(20) \quad \text{Minimise: } f = f(\Delta \mathbf{a}, \Delta \lambda) .$$

4.3.2. Dynamic Elastoplasticity

Consider the dynamic elastoplastic equation that consists of the Helmholtz free energy f , and kinetic energy T (without viscous damping). The total energy, $E = E(\mathbf{a}, \dot{\mathbf{a}}, \lambda) = T + f$, takes the form:

$$(21) \quad E(\mathbf{a}, \dot{\mathbf{a}}, \lambda) = \frac{1}{2} \dot{\mathbf{a}}^T \mathbf{M} \dot{\mathbf{a}} + \frac{1}{2} \mathbf{a}^T \mathbf{K} \mathbf{a} + \mathbf{a}^T \mathbf{G} \lambda + \frac{1}{2} \lambda^T \mathbf{H} \lambda ,$$

where $\dot{\mathbf{a}} = d\mathbf{a}/dt$, denotes a vector of generalised nodal velocity and $\mathbf{M} = \partial^2 T / \partial \dot{\mathbf{a}} \partial \dot{\mathbf{a}}$, a positive definite generalised mass matrix. Lagrange's equations help to formulate a set of dynamic elastoplastic collocation differential equations, that is:

$$(22) \quad \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\mathbf{q}}_j} \right) - \frac{\partial T}{\partial \mathbf{q}_j} + \frac{\partial f}{\partial \mathbf{q}_j} = \mathbf{Q}_j; \quad \mathbf{q}_1 = \mathbf{a}, \mathbf{q}_2 = \lambda; \mathbf{Q}_1 = \mathbf{f}, \mathbf{Q}_2 = -\chi,$$

where \mathbf{q} denotes generalised co-ordinate; $\dot{\mathbf{q}} = d\mathbf{q}/dt$, generalised velocity; \mathbf{Q} , generalised force; \mathbf{f} , generalised applied force and χ , generalised thermodynamic force conjugate to generalised internal variable λ . The resulting dynamic elastoplastic collocation differential equations are as follows:

$$(23) \quad \begin{Bmatrix} \ddot{\mathbf{a}}_j \\ \ddot{\lambda}_j \end{Bmatrix}^T \begin{bmatrix} \mathbf{M} \\ \mathbf{0} \end{bmatrix}_j + \begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{H} \end{bmatrix} \begin{Bmatrix} \mathbf{a}_j \\ \lambda_j \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_j \\ -\chi_j \end{Bmatrix} \quad j = n, (n+1).$$

The symbol $\ddot{\mathbf{a}} = d^2\mathbf{a}/dt^2$, denotes a vector of generalised acceleration and $\ddot{\lambda} = d^2\lambda/dt^2$.

In discrete form, the total energy is $E_{n+1} = E(\mathbf{a} + \Delta\mathbf{a}, \dot{\mathbf{a}} + \Delta\dot{\mathbf{a}}, \lambda + \Delta\lambda)$. Writing this function in a Taylor series expansion yields

$$(24) \quad E_{n+1} = E_n + E(\Delta\mathbf{a}, \Delta\dot{\mathbf{a}}, \Delta\lambda),$$

where following the technique used in the quasi-static case, the increment in energy is given by

$$(25) \quad E(\Delta\mathbf{a}, \Delta\dot{\mathbf{a}}, \Delta\lambda) = \langle \dot{\mathbf{a}}_n \rangle^T \mathbf{M} \Delta\dot{\mathbf{a}} + \frac{1}{2} \begin{Bmatrix} \Delta\mathbf{a} \\ \Delta\lambda \end{Bmatrix}^T \begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{H} \end{bmatrix}_n \begin{Bmatrix} \Delta\mathbf{a} \\ \Delta\lambda \end{Bmatrix} + \Delta\mathbf{a}^T \{ \mathbf{K}\mathbf{a}_n + \mathbf{G}\lambda_n \} - \Delta\lambda^T \chi_n$$

and $\langle \dot{\mathbf{a}}_n \rangle = \frac{1}{2} [\dot{\mathbf{a}}_{n+1} + \dot{\mathbf{a}}_n]$. The following statement defines the unconstrained non-linear mathematical programming problem resulting from the above equations:

$$(26) \quad \text{Minimise: } E = E(\Delta\mathbf{a}, \Delta\dot{\mathbf{a}}, \Delta\lambda).$$

Using the dynamic elastoplastic collocation differential equations, the following substitution simplifies the above equation:

$$(27) \quad \mathbf{K}\mathbf{a}_n + \mathbf{G}\lambda_n = \mathbf{f}_n - \mathbf{M}\ddot{\mathbf{a}}_n; \quad \mathbf{K}\Delta\mathbf{a} + \mathbf{G}\Delta\lambda = \Delta\mathbf{f} - \mathbf{M}\Delta\ddot{\mathbf{a}},$$

so that the increment in energy becomes

$$(28) \quad E(\Delta\mathbf{a}, \Delta\dot{\mathbf{a}}, \Delta\lambda) = \langle \dot{\mathbf{a}}_n \rangle^T \mathbf{M} \Delta\dot{\mathbf{a}} - \Delta\mathbf{a}^T \mathbf{M} \langle \ddot{\mathbf{a}}_n \rangle + \begin{Bmatrix} \Delta\mathbf{a} \\ \Delta\lambda \end{Bmatrix}^T \begin{Bmatrix} \langle \mathbf{f}_n \rangle \\ -\langle \boldsymbol{\chi}_n \rangle \end{Bmatrix},$$

where $\langle \cdot \rangle_n \equiv \frac{1}{2} [(\cdot)_{n+1} + (\cdot)_n]$ and $\Delta(\cdot) \equiv (\cdot)_{n+1} - (\cdot)_n$ as defined in **Chapter 2**,

Section 2.3.3.

4.3.3. The Generalised Newton-Raphson Scheme

In this subsection, the solutions of the dynamic elastoplastic equations are considered. The solutions of the quasi-static equations may be obtained from those of the dynamic ones and hence these will not be considered here. The procedure for solving the differential equation consists of forming a residual force, which is the out of balance force between the external, internal and D'Alembert's inertia forces. Initially, these forces are balanced so that the residual force is equal to zero as follows:

$$(29) \quad \Psi(\mathbf{q}, \ddot{\mathbf{q}}) = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{P}(\mathbf{q}) - \mathbf{Q} = \mathbf{0},$$

where the following definitions are taken from the previous sections:

$$(30) \quad \mathbf{q} = \{\mathbf{a} \quad \lambda\}^T; \quad \ddot{\mathbf{q}} = \{\ddot{\mathbf{a}} \quad \ddot{\lambda}\}^T; \quad \mathbf{Q} = \{\mathbf{f} \quad -\chi\}^T; \quad \mathbf{M} = \{\mathbf{M} \quad \mathbf{0}\}^T;$$

$$\mathbf{P}(\mathbf{q}) = \begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{H} \end{bmatrix} \begin{Bmatrix} \mathbf{a} \\ \lambda \end{Bmatrix}; \quad \mathbf{S} = \nabla \mathbf{P}(\mathbf{q}) = \begin{bmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{H} \end{bmatrix}$$

At the beginning of a new time-step, the residual is the out of balance force due to the increment or decrement of the external forces ($\mathbf{f}_{n+1} = \mathbf{f}_n + \Delta \mathbf{f}$). The residual is expanded in a Taylor series based on the *forward-backward* (FB) difference scheme (see ref. [19-21] and **Chapter 2**, Section 2.3.2) as follows:

$$(31) \quad \Psi_{n+1}^{i+1} = \Psi_{n+1}^i + \Delta \mathbf{q}^T \left[(1-\beta_1) \left(\frac{\partial \Psi}{\partial \mathbf{q}} \right) \Big|_{\mathbf{q}_{n+1}^i} + \beta_1 \left(\frac{\partial \Psi}{\partial \mathbf{q}} \right) \Big|_{\mathbf{q}_{n+1}^{i+1}} \right]$$

$$+ \Delta \ddot{\mathbf{q}}^T \left[(1-\beta_3) \left(\frac{\partial \Psi}{\partial \ddot{\mathbf{q}}} \right) \Big|_{\ddot{\mathbf{q}}_{n+1}^i} + \beta_3 \left(\frac{\partial \Psi}{\partial \ddot{\mathbf{q}}} \right) \Big|_{\ddot{\mathbf{q}}_{n+1}^{i+1}} \right] \quad 0 \leq \beta_j \leq 1$$

The superscript $(.)^i$ denotes the iteration counter within the time interval $[n, n+1]$.

The scalars β_j , $j=1,3$ determine what proportions of the forward/backward difference schemes participate in the expansion of the residual with respect to the variables, and may be used to control the accuracy and stability of the algorithm.

From the definitions given in the above equations

$$(32) \quad \frac{\partial \Psi}{\partial \mathbf{q}} = \mathbf{S}; \quad \frac{\partial \Psi}{\partial \ddot{\mathbf{q}}} = \mathbf{M},$$

and the expanded residual equation becomes:

$$(33) \quad \Psi_{n+1}^{i+1} = \Psi_{n+1}^i + \Delta \mathbf{q}^T \left[(1-\beta_1) \mathbf{S}_{n+1}^i + \beta_1 \mathbf{S}_{n+1}^{i+1} \right] + \Delta \ddot{\mathbf{q}}^T \left[(1-\beta_3) \mathbf{M}_{n+1}^i + \beta_3 \mathbf{M}_{n+1}^{i+1} \right].$$

Using time discretisation (see **Chapter 2**, Section 2.3.2 and reference [21]), the change in the acceleration vector may be written as

$$(34) \quad \Delta \ddot{\mathbf{q}} = \frac{2}{\Delta t^2 \alpha_2 (2\alpha_1 - \alpha_2)} \Delta \mathbf{q} - \frac{2}{\Delta t \alpha_2 (2\alpha_1 - \alpha_2)} \dot{\mathbf{q}}_{n+1}^i - \frac{1}{\alpha_2 (2\alpha_1 - \alpha_2)} \ddot{\mathbf{q}}_{n+1}^i,$$

for the **GNR-FB1** scheme (**Chapter 2**, Section 2.3.2) and as

$$(35) \quad \Delta \ddot{\mathbf{q}} = \frac{2}{\Delta t^2 \alpha_2^2} \Delta \mathbf{q} - \frac{2}{\Delta t \alpha_2^2} \dot{\mathbf{q}}_{n+1}^i - \frac{1}{\alpha_2^2} \ddot{\mathbf{q}}_{n+1}^i,$$

for the **GNR-FB2** scheme. Substituting these equations into the expanded residual equation results in the *generalised* Newton-Raphson (GNR) scheme as follows:

$$(36) \quad \mathbf{K}_T^{i+1} \Delta \mathbf{q} = \Psi_{n+1}^{i+1},$$

where the *generalised* Jacobean matrix for the **GNR-FB1** scheme is defined as;

$$(37) \quad \mathbf{K}_T^{i+1} = (1 - \beta_1) \mathbf{S}_{n+1}^i + \beta_1 \mathbf{S}_{n+1}^{i+1} + \frac{2}{\Delta t^2 \alpha_2 (2\alpha_1 - \alpha_2)} [(1 - \beta_3) \mathbf{M}_{n+1}^i + \beta_3 \cdot \mathbf{M}_{n+1}^{i+1}],$$

with the corresponding residual force

$$(38) \quad \Psi_{n+1}^{i+1} = -\Psi_{n+1}^i + [(1 - \beta_3) \mathbf{M}_{n+1}^i + \beta_3 \mathbf{M}_{n+1}^{i+1}] \cdot \left(\frac{2}{\Delta t \alpha_2 (2\alpha_1 - \alpha_2)} \dot{\mathbf{q}}_{n+1}^i + \frac{1}{\alpha_2 (2\alpha_1 - \alpha_2)} \ddot{\mathbf{q}}_{n+1}^i \right).$$

The *generalised* Jacobean matrix for the **GNR-FB2** scheme is defined as;

$$(39) \quad \mathbf{K}_T^{i+1} = (1 - \beta_1) \mathbf{S}_{n+1}^i + \beta_1 \mathbf{S}_{n+1}^{i+1} + \frac{2}{\Delta t^2 \alpha_2^2} [(1 - \beta_3) \mathbf{M}_{n+1}^i + \beta_3 \cdot \mathbf{M}_{n+1}^{i+1}],$$

with the corresponding residual force

$$(40) \quad \Psi_{n+1}^{i+1} = -\Psi_{n+1}^i + \left[(1-\beta_3)\mathbf{M}_{n+1}^i + \beta_3 \mathbf{M}_{n+1}^{i+1} \right] \cdot \left(\frac{2}{\Delta t \alpha_2^2} \dot{\mathbf{q}}_{n+1}^i + \frac{1}{\alpha_2^2} \ddot{\mathbf{q}}_{n+1}^i \right).$$

The iterations in each algorithm are continued until the following convergence criterion [22] is satisfied:

$$(41) \quad \left| \Delta \mathbf{q}^T (\Psi_{n+1}^{i+1} + \Psi_{n+1}^i) \right| < c \hat{E}_{n+1}^i, \quad c \ll 1,$$

where c is some small tolerance and for each algorithm, Ψ is a residual vector discussed above in this subsection and \hat{E} is an energy-like norm [22] discussed in Section 4.3.4 below.

Note:

- The generalised Newton-Raphson scheme expresses the conditions that the increment in energy, $E = E(\Delta \mathbf{a}, \Delta \dot{\mathbf{a}}, \Delta \lambda)$, defined in Section 4.3.2, takes on its minimum value, which is written as a non-linear mathematical programming problem.
- The generalised Newton-Raphson schemes presented in this chapter are identical to those of **Chapter 2**, Section 2.3.2 and **Chapter 3**, Section 3.3.2 and 3.3.4.
- The predictor-corrector steps for the generalised Newton-Raphson schemes presented in this chapter are identical to those of **Chapter 3**, Section 3.3.4.

- The solutions of the static elastoplastic equations may be obtained from those of the dynamic elastoplastic equations by setting $\mathbf{M} = \mathbf{0}$ and $\dot{\mathbf{q}} = \ddot{\mathbf{q}} = \mathbf{0}$.
- The conventional Newton-Raphson (NR) scheme is recovered from the GNR scheme by setting $\beta_1 = \beta_3 = 0$.

4.3.4. Stability Analysis

Algorithm stability is classified as conditional or unconditional. In the former, stability is achieved with a restriction on the time step-size, unlike in the latter where the step-size is related purely to accuracy. This section discusses stability of elastoplastic equations using energy methods similar to the Lyapunov's functions [17-18], in the form presented by Wood [22]. The Lyapunov method provides a means for determining the stability of a system without explicitly solving for the trajectories in the state space. This method is applicable for determining the behaviour of first and higher order systems that may be forced or unforced, linear or non-linear, time-invariant or time-varying, and deterministic or stochastic. A summary of the Lyapunov global asymptotic stability conditions follows:

A system is globally asymptotically stable if there is only one stable equilibrium point and there exists a scalar function $V(\mathbf{x})$ such that:

1. $V(\mathbf{x})$ is continuous and has continuous first partial derivatives in the whole state space.
2. $V(\mathbf{x}) > 0$ for $\mathbf{x} \neq \mathbf{0}$.
3. $V(\mathbf{0}) = 0$.
4. $V(\mathbf{x}) \rightarrow \infty$ as $\|\mathbf{x}\| \rightarrow \infty$.
5. $\dot{V}(\mathbf{x}) = \langle \nabla V(\mathbf{x}), \dot{\mathbf{x}} \rangle \leq 0$.

6. Either $V(\mathbf{x}) \neq 0$ except at $\mathbf{x} = \mathbf{0}$ or any locus in the state space where $V(\mathbf{x}) = 0$ is not a trajectory of the system.

These Lyapunov stability conditions motivate the energy stability methods [14,17-18,22-26] in use.

Non-linear B-Stability

The notion of B-stability was introduced by Simo and Govindjee [25], who showed that the evolution equations for hardening plasticity and for viscoplasticity both exhibit the property of contractivity. This notion states that the evolution equations for plasticity or viscoplasticity are contractive [25] or non-expansive [14] relative to the norm defined by the complementary Helmholtz free energy. Application of this notion to dynamic elastoplasticity is covered in reference [13]. The results of these papers, in context of stability analysis, can be summarised as follows:

$$(42) \quad \frac{d}{dt}(f(\mathbf{x})) = \langle \nabla f(\mathbf{x}), \dot{\mathbf{x}} \rangle \leq 0 \quad \mathbf{x} = \mathbf{x}(\mathbf{a}, \lambda) \quad \text{quasi-static ,}$$

$$(43) \quad \frac{d}{dt}(f + T) = \langle \nabla f(\mathbf{x}), \dot{\mathbf{x}} \rangle + \langle \nabla T(\dot{\mathbf{a}}), \ddot{\mathbf{a}} \rangle \leq 0 \quad \mathbf{x} = \mathbf{x}(\mathbf{a}, \lambda) \quad \text{dynamic .}$$

- These results are precisely the Lyapunov's conditions for stability. Stability analysis for the dynamic elastoplastic equations will be considered in this chapter using energy methods discussed by Wood [22].

Consider the dynamic elastoplastic equations presented in Section 4.3.2 and rewritten here for convenience:

$$(44) \quad E_{n+1} = E_n + E(\Delta \mathbf{a}, \Delta \dot{\mathbf{a}}, \Delta \lambda) ,$$

$$(45) \quad E(\Delta \mathbf{a}, \Delta \dot{\mathbf{a}}, \Delta \lambda) = \langle \dot{\mathbf{a}}_n \rangle^T \mathbf{M} \Delta \dot{\mathbf{a}} - \Delta \mathbf{a}^T \mathbf{M} \langle \ddot{\mathbf{a}}_n \rangle + \begin{Bmatrix} \Delta \mathbf{a} \\ \Delta \lambda \end{Bmatrix}^T \begin{Bmatrix} \langle \mathbf{f}_n \rangle \\ -\langle \chi_n \rangle \end{Bmatrix}.$$

1. Stability Analysis of FBI Algorithm:

Combining the above equations and making use of the relations developed for the FBI Algorithm, in Chapter 2, Section 2.3.3 result in:

$$(46) \quad E_{n+1} = E_n + \left[\Delta \mathbf{a} - \frac{\Delta t^2}{2} (2\alpha_1 \alpha_2 - \alpha_1 - \alpha_2^2) \Delta \ddot{\mathbf{a}} \right]^T \mathbf{M} [\langle \ddot{\mathbf{a}}_n \rangle + (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}] - \Delta \mathbf{a}^T \mathbf{M} \langle \ddot{\mathbf{a}}_n \rangle + \Delta \mathbf{a}^T \langle \mathbf{f}_n \rangle - \Delta \lambda^T \langle \chi_n \rangle.$$

Multiplying out the brackets and using the dynamic elastoplastic collocation differential equations (23) of Section 4.3.2, leads to the following equation:

$$(47) \quad E_{n+1} = E_n - \frac{\Delta t^2}{2} (2\alpha_1 \alpha_2 - \alpha_1 - \alpha_2^2) \langle \ddot{\mathbf{a}}_n \rangle^T \mathbf{M} \Delta \ddot{\mathbf{a}} - \frac{\Delta t^2}{2} (2\alpha_1 \alpha_2 - \alpha_1 - \alpha_2^2) (\alpha_1 - \frac{1}{2}) \Delta \ddot{\mathbf{a}}^T \mathbf{M} \Delta \ddot{\mathbf{a}} - (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}^T \mathbf{K} \Delta \mathbf{a} - (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}^T \mathbf{G} \Delta \lambda + \Delta \mathbf{a}^T \{ \langle \mathbf{f}_n \rangle + (\alpha_1 - \frac{1}{2}) \Delta \mathbf{f} \} - \Delta \lambda^T \langle \chi_n \rangle.$$

The above equation suggests an energy norm [22] of the form

$$(48) \quad \hat{E}_n = \frac{1}{2} \frac{\Delta t^2}{2} (2\alpha_1 \alpha_2 - \alpha_1 - \alpha_2^2) \ddot{\mathbf{a}}_n^T \mathbf{M} \ddot{\mathbf{a}}_n + E_n,$$

where the true energy E_n , given by $E = E(\mathbf{a}, \dot{\mathbf{a}}, \lambda) = T + f$ in Section 4.3.2, is recovered only when $(2\alpha_1 \alpha_2 - \alpha_1 - \alpha_2^2) = 0$. For this energy norm to be

legitimate, the condition, $(2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2) \geq 0$ must be satisfied at all times. The

FBI algorithm reduces to:

$$(49) \quad \begin{aligned} \hat{E}_{n+1} = \hat{E}_n & - \frac{\Delta t^2}{2} (2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2) (\alpha_1 - \frac{1}{2}) \Delta \ddot{\mathbf{a}}^T \mathbf{M} \Delta \ddot{\mathbf{a}} \\ & - (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}^T \mathbf{K} \Delta \mathbf{a} - (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}^T \mathbf{G} \Delta \lambda \\ & + \Delta \mathbf{a}^T \{ \langle \mathbf{f}_n \rangle + (\alpha_1 - \frac{1}{2}) \Delta \mathbf{f} \} - \Delta \lambda^T \langle \chi_n \rangle \end{aligned}$$

The following condition must be satisfied for *unconditional stability* of the *FBI* algorithm:

$$(50) \quad \hat{E}_{n+1} \leq \hat{E}_n .$$

Note that the above statement, which states that the energy is non-increasing, is closely related to the Lyapunov conditions for stability, and is true for the following conditions:

$$(51) \quad \alpha_1 \geq \frac{1}{2}; (2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2) \geq 0; \Delta \lambda^T \langle \chi \rangle \geq 0; \text{ no external loading .}$$

Unconditional stability of this algorithm is only possible when $\alpha_1 = \alpha_2 = 1$, and this coincides with $\hat{E}_n = E_n$, that is, the energy norm defined is equal to the true energy.

However, under these circumstances, energy is dissipated due to numerical damping as shown by the condition $\hat{E}_{n+1} < \hat{E}_n$ for $\alpha_1 \neq \frac{1}{2}; \forall \Delta \lambda^T \langle \chi \rangle \geq 0$. Also with unconditional stability, the work done by the external loads reduces to $\Delta \mathbf{a}^T \mathbf{f}_n$.

Therefore, if the loading is bounded, the energy norm is also bounded, and the algorithm converges. The energy associated with the dissipation function, $\Delta \lambda^T \langle \chi \rangle \geq 0$, is independent of the integration parameters. It is always

positive as a consequence of the second law of thermodynamics and is not restricted to a specific yield criterion.

2. Stability Analysis of FB2 Algorithm:

Similarly, making use of the relations developed for the *FB2* Algorithm, in

Chapter 2, Section 2.3.3 results in:

$$(52) \quad E_{n+1} = E_n + \left[\Delta \mathbf{a} - \frac{\Delta t^2}{2} (\alpha_2^2 - \alpha_1) \Delta \ddot{\mathbf{a}} \right]^T \mathbf{M} [\langle \ddot{\mathbf{a}}_n \rangle + (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}] - \Delta \mathbf{a}^T \mathbf{M} \langle \ddot{\mathbf{a}}_n \rangle + \Delta \mathbf{a}^T \langle \mathbf{f}_n \rangle - \Delta \lambda^T \langle \chi_n \rangle$$

Multiplying out the brackets and using the dynamic elastoplastic collocation differential equations (23) of Section 4.3.2, leads to the following equation:

$$(53) \quad E_{n+1} = E_n - \frac{\Delta t^2}{2} (\alpha_2^2 - \alpha_1) \langle \ddot{\mathbf{a}}_n \rangle^T \mathbf{M} \Delta \ddot{\mathbf{a}} - \frac{\Delta t^2}{2} (\alpha_2^2 - \alpha_1) (\alpha_1 - \frac{1}{2}) \Delta \ddot{\mathbf{a}}^T \mathbf{M} \Delta \ddot{\mathbf{a}} - (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}^T \mathbf{K} \Delta \mathbf{a} - (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}^T \mathbf{G} \Delta \lambda + \Delta \mathbf{a}^T \{ \langle \mathbf{f}_n \rangle + (\alpha_1 - \frac{1}{2}) \Delta \mathbf{f} \} - \Delta \lambda^T \langle \chi_n \rangle$$

The above equation suggests that an energy norm be defined [22] in the following manner:

$$(54) \quad \hat{E}_n = \frac{1}{2} \frac{\Delta t^2}{2} (\alpha_2^2 - \alpha_1) \ddot{\mathbf{a}}_n^T \mathbf{M} \ddot{\mathbf{a}}_n + E_n,$$

where the true energy E_n , given by $E = E(\mathbf{a}, \dot{\mathbf{a}}, \lambda) = T + f$ in Section 4.3.2, is recovered only when $(\alpha_2^2 - \alpha_1) = 0$. For this energy norm to be legitimate, the

condition, $(\alpha_2^2 - \alpha_1) \geq 0$ must be satisfied at all times. The *FB2* algorithm reduces to:

$$(55) \quad \begin{aligned} \hat{E}_{n+1} = \hat{E}_n &- \frac{\Delta t^2}{2} (\alpha_2^2 - \alpha_1) (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}^T \mathbf{M} \Delta \ddot{\mathbf{a}} \\ &- (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}^T \mathbf{K} \Delta \mathbf{a} - (\alpha_1 - \frac{1}{2}) \Delta \mathbf{a}^T \mathbf{G} \Delta \lambda . \\ &+ \Delta \mathbf{a}^T \{ \langle \mathbf{f}_n \rangle + (\alpha_1 - \frac{1}{2}) \Delta \mathbf{f} \} - \Delta \lambda^T \langle \chi_n \rangle \end{aligned}$$

The following condition must be satisfied for *unconditional stability* of the *FB2* algorithm:

$$(56) \quad \hat{E}_{n+1} \leq \hat{E}_n .$$

This statement is true for the following conditions:

$$(57) \quad \alpha_1 \geq \frac{1}{2}; (\alpha_2^2 - \alpha_1) \geq 0; \Delta \lambda^T \langle \chi \rangle \geq 0; \text{ no external loading .}$$

Unconditional stability of this algorithm is possible for all the cases shown in the above statement, and $\hat{E}_n = E_n$ for $\alpha_2^2 = \alpha_1$, that is, the energy norm defined is equal to the true energy. The algorithm does not possess numerical damping if $\alpha_1 = \frac{1}{2}$, that is, $\hat{E}_{n+1} = \hat{E}_n$ when there is no external loading and no energy is dissipated. Since this algorithm is identical to the Newmark [27] algorithm for $\alpha_2^2 = 2\beta$; $\alpha_1 = \gamma$, this condition also confirms the popular result that the unconditionally stable trapezium rule or constant average acceleration (CAA) method does not possess numerical damping. In contrast with the *FBI* scheme discussed above, under these conditions the work done by the external loads reduces to $\Delta \mathbf{a}^T \langle \mathbf{f}_n \rangle$. Therefore, if the loading is bounded, the energy norm is also bounded, and the algorithm converges. The energy associated with the dissipation

function, $\Delta\lambda^T \langle \chi \rangle \geq 0$, is also independent of the integration parameters, and is always positive and not restricted to a specific yield criterion as already proved.

4.4. CONCLUSIONS

Finite element formulations of the static and dynamic *initial-boundary value* evolution problems have been considered using the internal variable framework and solution algorithms have been developed using the generalised trapezoidal (GT) rule and generalised Newton-Raphson (GNR) scheme.

The stability analyses for the dynamic elastoplastic equations have been performed using energy-like norms that form functions similar to the Lyapunov's functions. These norms are further used in the algorithms convergence tests. In common with the Newmark scheme, the *FB2* algorithm is unconditionally stable for $\alpha_1 \geq \frac{1}{2}$ and $(\alpha_2^2 - \alpha_1) \geq 0$, and with $\alpha_2^2 = \alpha_1 = \frac{1}{2}$, the algorithm is equivalent to the constant average acceleration (CAA) or trapezium rule, and does not possess numerical damping. The *FBI* algorithm is unconditionally stable for $\alpha_1 = \alpha_2 = 1$ and possesses numerical damping as a consequence of $\alpha_1 \neq \frac{1}{2}$.

REFERENCES

- [1] Kestin J, Rice J R, "Paradoxes in the application of thermodynamics to strained solids", *A Critical Review of Thermodynamics*, ed. E B Stuart, Gal-or and A J Brainard, Mono Book Corp., Baltimore, 275-298, (1970).
- [2] Moreau J, "Sur les lois de frottement, de viscosity et plasticite'", *C R Acad Sc.*, 271, 608-611, (1970).

-
- [3] Rice J R, "Inelastic constitutive relations for solids: an internal variable theory and its application to metal plasticity", *J of Mechanics and Physics of Solids*, 19, 433-455, (1971).
- [4] Martin J B, *Plasticity, Fundamentals and General Results*, MIT Press, 1975.
- [5] Martin J B, "A note on the implications of thermodynamic stability in the internal variable theory of inelastic solids", *Int. J. Solids and Structures*, 11, 247-253, 1975.
- [6] Halphen B, Nguyen Q S, "Sur les mate'riaux standards ge'ne'ralize's", *J de Me'canique*, 14, 39-63, (1975).
- [7] Martin J B, "An internal variable approach to the formulation of finite element problems in plasticity", In *Physical Non-linearities in Structural Analysis*, ed. J Hult and J Lemaitre, Springer-Verlag Berlin Heidelberg, 165-176, (1981).
- [8] Lemaitre J, Chaboche J L, *Me'canique des mate'riaux solides*, Dunod, Paris, 1975.
- [9] Martin J B, Reddy B D, Griffin T B, Bird W W., "Application of mathematical programming concepts to incremental elastic-plastic analysis", *Engng Struct.*, 9, 171-176, (1987).
- [10] Martin J B, "A complementary work bounding principle for forward integration along the path of loading for elastic-plastic bodies", *J. Appl. Mech.*, 109, 341-345, (1987).
- [11] Martin J B, Reddy B D, "Variational principles and solution algorithms for internal variable formulations of problems in plasticity", In *Omaggio a Giulio: Note Scientifiche in Occasione del 70^o Compleanno*, ed. U Andreaus et al., Universita di Roma 'La Sapienza' (Roma), 465-477, (1988).
- [12] Eve R A, Reddy B D, Rockafellar R T, "An internal variable theory of elastoplasticity based on maximum plastic work inequality", *Quart of Appl. Math.*, 48, 59-83, (1990).
- [13] Corigliano A, Perego U, "Unconditionally stable mid-point time integration in elastic-plastic dynamics", *Rendi. Acc. Lincei*, s. 9, 1, 367-376, (1990).
- [14] Reddy B D, Martin J B, "Algorithms for the solution of internal variable problems in plasticity", *Computer Methods in Applied Mechanics and Engineering*, 93, 253-273, (1991).

-
- [15] Martin J B, Nappi A, “An internal variable formulation for perfectly plastic and linear hardening relations in plasticity”, *European Journal of Mechanics and Solids*, 9, 107-131, (1990).
- [16] Martin J B, Reddy B D, “Piecewise smooth dissipation and yield functions in plasticity”, In Proc. XIth National Congress of Theoretical and Applied Mechanics (AIMETA), Trento, September, (1992).
- [17] D’Azzo J J., Houpis C H, *Linear Control System Analysis and Design, Conventional and Modern*, 2nd Edition, 1981.
- [18] Huseyin K, *Multiple Parameter Stability Theory and its Applications, Bifurcations, Catastrophes, Instabilities, ...*, Clarendon Press, Oxford, 1986.
- [19] Kaunda M A E, Martin J B, “A forward-backward difference time-integration scheme based on Taylor series”, In Proc. 11th Symposium on Finite Element Methods in South Africa (FEMSA), 287-296, (1992).
- [20] Kaunda M A E, Martin J B, “On single-step time-integrators”, In Proc. 18th South African Symposium on Numerical Mathematics (SANUM), 78-101, (1992).
- [21] Kaunda M A E, Martin J B, “Finite element solution algorithms for non-linear time-dependent/independent problems in solid mechanics”, In Proc. 19th South African Symposium on Numerical Mathematics (SANUM), 95-117, (1993).
- [22] Wood W L, *Practical Time-stepping Schemes*, Clarendon Press, Oxford, 1990.
- [23] Hughes T J R, “A note on the stability of Newmark’s algorithm in non-linear structural dynamics”, *Int. J. for Num. Meth. in Eng.*, 11, 383-386, (1977).
- [24] Simo J C, Wong K K., “Unconditionally stable algorithms for rigid body dynamics that exactly preserve energy and momentum”, *Int. J. Num. Meth. Eng.*, 31, 19-52, (1991).
- [25] Simo J C, Govindjee S, “Non-linear B-stability and symmetry preserving return mapping algorithms for plasticity and viscoplasticity”, *Int. J. Num. Meth. Eng.*, 31, 151-176, (1991).
- [26] Simo J C, Tarnow N, Wong K K., “Exact energy-momentum conserving algorithms and symplectic schemes for non-linear dynamics”, *Computer Methods in Applied Mechanics and Engineering*, 100, 63-116, (1992).
- [27] Newmark N M, “A method of computation for structural dynamics”, *J. Engng Mech. Div., ASCE*, 85, EM3, 67-94, (1959).

CHAPTER 5

PARAMETRIC STUDIES OF INTEGRATION SCHEMES

5.1. INTRODUCTION

In this chapter, solutions of transient linear and non-linear problems are considered with a goal of comparing algorithms presented in **Chapter 2** with the conventional algorithms reported in various papers [1-10], books [11-16] and other literature. The discussion is confined to single-step time-integration schemes because they have the desirable property of allowing variable time-steps.

Section 5.2 reviews the algorithms developed in **Chapter 2** and compares them with the well known Newmark scheme in the solution of transient linear and non-linear problems. Extensions to the generalised Newton-Raphson (GNR) scheme not reported in earlier chapters are proposed in Section 5.2.2. Section 5.3 summarises the accuracy of each recurrence equation developed so that a comparison can easily be made with the corresponding one from the Newmark scheme. Section 5.4 looks at the stability of the algorithms to consolidate parametric studies of integration schemes. Numerical examples are given in Section 5.5 to evaluate the performance of algorithms.

5.2. SOLUTION OF TRANSIENT LINEAR AND NON-LINEAR PROBLEMS

Only the second order differential (dynamic) equation is dealt with because it includes the static one as a special case with $\mathbf{M} = \mathbf{C} = \mathbf{0}$, and $\ddot{\mathbf{a}} = \dot{\mathbf{a}} = \mathbf{0}$; see equation (1) below.

5.2.1. Transient Linear Problems

Single-step time-integration algorithms for solving the differential equation

$$(1) \quad \mathbf{M}\ddot{\mathbf{a}} + \mathbf{C}\dot{\mathbf{a}} + \mathbf{K}\mathbf{a} = \mathbf{f}, \quad t \in [c, d],$$

with initial values

$$(2) \quad \mathbf{a}(c) = \mathbf{a}_0, \quad \dot{\mathbf{a}}(c) = \dot{\mathbf{a}}_0,$$

date back to 1959 when Newmark [2] developed his well known algorithm:

$$(3) \quad \mathbf{M}\ddot{\mathbf{a}}_{n+1} + \mathbf{C}\dot{\mathbf{a}}_{n+1} + \mathbf{K}\mathbf{a}_{n+1} = \mathbf{f}_{n+1},$$

$$(4) \quad \mathbf{a}_{n+1} = \mathbf{a}_n + \Delta t \dot{\mathbf{a}}_n + \frac{\Delta t^2}{2} \left[(1 - 2\beta) \ddot{\mathbf{a}}_n + 2\beta \ddot{\mathbf{a}}_{n+1} \right],$$

$$(5) \quad \dot{\mathbf{a}}_{n+1} = \dot{\mathbf{a}}_n + \Delta t \left[(1 - \gamma) \ddot{\mathbf{a}}_n + \gamma \ddot{\mathbf{a}}_{n+1} \right],$$

where \mathbf{M} denotes the mass matrix; \mathbf{C} , the viscous damping matrix; \mathbf{K} , the stiffness matrix; \mathbf{f} , the applied nodal loading vector; \mathbf{a} , $\dot{\mathbf{a}}$ and $\ddot{\mathbf{a}}$, are vectors of generalised displacement, velocity and acceleration, respectively, and scalars $\gamma, \beta \in [0,1]$ are integration parameters that determine the stability and accuracy of the algorithm. A number of researchers [3-10] have looked closely at the algorithm for a number of reasons; the most important ones being

- Accuracy considerations.
- Better control of algorithm stability characteristics.
- Control of algorithm *overshoot* behaviour.

- Preservation of symmetry of tangent moduli (for non-linear equations).

The solution to the linear dynamic problem posed above in equation (1) using the methods developed in **Chapter 2** are rewritten here for convenience:

(i) the **FBI scheme** that consists of the following set of simultaneous equations,

$$(6) \quad \mathbf{M}\ddot{\mathbf{a}}_{n+1} + \mathbf{C}\dot{\mathbf{a}}_{n+1} + \mathbf{K}\mathbf{a}_{n+1} = \mathbf{f}_{n+1},$$

$$(7) \quad \mathbf{a}_{n+1} = \mathbf{a}_n + \Delta t \left[(1 - \alpha_2) \dot{\mathbf{a}}_n + \alpha_2 \dot{\mathbf{a}}_{n+1} \right] + \frac{\Delta t^2}{2} \left[(1 - \alpha_2)^2 \ddot{\mathbf{a}}_n - \alpha_2^2 \ddot{\mathbf{a}}_{n+1} \right],$$

$$(8) \quad \dot{\mathbf{a}}_{n+1} = \dot{\mathbf{a}}_n + \Delta t \left[(1 - \alpha_1) \ddot{\mathbf{a}}_n + \alpha_1 \ddot{\mathbf{a}}_{n+1} \right],$$

and

(ii) the **FB2 scheme**, that consists of a similar set of simultaneous equations,

$$(9) \quad \mathbf{M}\ddot{\mathbf{a}}_{n+1} + \mathbf{C}\dot{\mathbf{a}}_{n+1} + \mathbf{K}\mathbf{a}_{n+1} = \mathbf{f}_{n+1},$$

$$(10) \quad \mathbf{a}_{n+1} = \mathbf{a}_n + \Delta t \dot{\mathbf{a}}_n + \frac{\Delta t^2}{2} \left[(1 - \alpha_2^2) \ddot{\mathbf{a}}_n + \alpha_2^2 \ddot{\mathbf{a}}_{n+1} \right],$$

$$(11) \quad \dot{\mathbf{a}}_{n+1} = \dot{\mathbf{a}}_n + \Delta t \left[(1 - \alpha_1) \ddot{\mathbf{a}}_n + \alpha_1 \ddot{\mathbf{a}}_{n+1} \right].$$

Note that the **FB2** scheme is identical to the Newmark scheme for all $\alpha_1 = \gamma$ and $\alpha_2^2 = 2\beta$, and hence enjoys all the good characteristics of the well known Newmark scheme. The **FBI** scheme is, however, equivalent to the Newmark scheme only when [19] $\alpha_1 = \gamma$ and $\beta = \alpha_2(2\alpha_1 - \alpha_2)/2$ (also written as $\alpha_2 = \gamma \pm \sqrt{\gamma^2 - 2\beta}$).

These results are obtained by comparing the coefficients of equations (4, 7 and 10; 5, 8 and 11) in the algorithms. Given that $\alpha_j, \gamma, \beta \in [0, 1]$, and $\gamma^2 - 2\beta \geq 0$ for non-

complex values of α_2 , the *FBI* scheme is equivalent to the Newmark scheme for a very narrow range of desirable values of integration parameters. In particular, given that the Newmark scheme is unconditionally stable for $2\beta \geq \gamma \geq \frac{1}{2}$, it may be deduced that the *FBI* scheme is unconditionally stable for $\alpha_1 = \alpha_2 = 1$. This fact is proved in stability analyses; see ref. [19] and **Chapter 4, Section 4.3.4**.

5.2.2. Transient Non-linear Problems

In many practical situations non-linearities exist, typically [16] altering equation (1) presented in the previous section to

$$(12) \quad \mathbf{M}(\mathbf{a})\ddot{\mathbf{a}} + \mathbf{C}(\mathbf{a})\dot{\mathbf{a}} + \mathbf{P}(\mathbf{a}) = \mathbf{f}, \quad t \in [c, d].$$

An iterative solution process for this non-linear equation is generally required. Henrici [17-18] gives the theoretical background to the Newton-Raphson method for solving general non-linear equations. This motivates the solution algorithms developed in **Chapters 2-4**. In particular, the generalised Newton-Raphson (GNR) scheme of **Chapter 3, Section 3.3.4**, in conjunction with any suitable set of recurrence equations relating displacement and velocity with acceleration, may be used for solving the non-linear equation (12). The GNR scheme includes the conventional Newton-Raphson (CNR) scheme as a special case with the GNR parameters set to:

- $\beta_1 = \beta_2 = \beta_3 = 0$, when implemented in the *implicit* version as follows: $\Psi_{n+1}^i + \Delta \mathbf{q}^T \left[(1-\beta)(\partial \Psi / \partial \mathbf{q})_{n+1}^i + \beta(\partial \Psi / \partial \mathbf{q})_{n+1}^{i+1} \right] = \mathbf{0}$. For practical reasons, this version is difficult to implement.
- $\beta_1 = \beta_2 = \beta_3 = 1$, when implemented in the *explicit* version as follows: $\Psi_{n+1}^i + \Delta \mathbf{q}^T \left[(1-\beta)(\partial \Psi / \partial \mathbf{q})_{n+1}^{i-1} + \beta(\partial \Psi / \partial \mathbf{q})_{n+1}^i \right] = \mathbf{0}$. This version and its extensions have been implemented, and provide remarkable enhancement on the *global* convergence of the conventional Newton-Raphson scheme without significant impairment of the *quadratic* convergence characteristics. As the solution is approached and $i \rightarrow \infty$, the algorithm becomes $\Psi_{n+1}^i + \Delta \mathbf{q}^T [\partial \Psi / \partial \mathbf{q}]_{n+1}^i = \mathbf{0}$, $\forall \beta \in [0,1]$; that is, the conventional Newton-Raphson scheme which converges quadratically.

Extended *explicit* versions have been investigated in the following ways:

- (i) **Version 1:** which is the original explicit version, that is,

$$\Psi_{n+1}^i + \Delta \mathbf{q}^T \left[(1-\beta)(\partial \Psi / \partial \mathbf{q})_{n+1}^{i-1} + \beta(\partial \Psi / \partial \mathbf{q})_{n+1}^i \right] = \mathbf{0}.$$

- (ii) **Version 2:** which is derived by extending Version 1 by one iteration back within a time step by letting

$$\left(\overline{\partial \Psi} / \partial \mathbf{q} \right)_{n+1}^{i-1} = (1-\beta)(\partial \Psi / \partial \mathbf{q})_{n+1}^{i-2} + \beta(\partial \Psi / \partial \mathbf{q})_{n+1}^{i-1},$$

which is substituted into

the equation for **Version 1**. This results in

$$\Psi_{n+1}^i + \Delta \mathbf{q}^T \left[(1-\beta)^2(\partial \Psi / \partial \mathbf{q})_{n+1}^{i-2} + (1-\beta)\beta(\partial \Psi / \partial \mathbf{q})_{n+1}^{i-1} + \beta(\partial \Psi / \partial \mathbf{q})_{n+1}^i \right] = \mathbf{0}.$$

(iii) **Version 3:** which is derived by extending Version 2 by one iteration back within a time step by letting

$$\left(\frac{\partial \bar{\Psi}}{\partial \mathbf{q}}\right)_{n+1}^{i-2} = (1-\beta)\left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-3} + \beta\left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-2}$$
 which is substituted into

the equation for **Version 2**, resulting in

$$\Psi_{n+1}^i + \Delta \mathbf{q}^T \left[(1-\beta)^3 \left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-3} + (1-\beta)^2 \beta \left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-2} + (1-\beta)\beta \left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-1} + \beta \left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^i \right] = \mathbf{0}.$$

(iv) **Version 4:** which is derived by extending Version 3 by one iteration back within a time step by letting

$$\left(\frac{\partial \bar{\Psi}}{\partial \mathbf{q}}\right)_{n+1}^{i-3} = (1-\beta)\left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-4} + \beta\left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-3}$$
 which is substituted into

the equation for **Version 3**, resulting in

$$\Psi_{n+1}^i + \Delta \mathbf{q}^T \left[(1-\beta)^4 \left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-4} + (1-\beta)^3 \beta \left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-3} + (1-\beta)^2 \beta^2 \left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-2} + (1-\beta)\beta^2 \left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^{i-1} + \beta^2 \left(\frac{\partial \Psi}{\partial \mathbf{q}}\right)_{n+1}^i \right] = \mathbf{0}$$

The extension process can go on up to the desired number of terms and it is not claimed that this is the only way of making extensions. Note that for Version 4, in the first iteration, the algorithm works as the conventional Newton-Raphson scheme. Then the process continues as follows: Version 1 in the second iteration; Version 2 in the third iteration; Version 3 in the fourth iteration; and Version 4 for the rest of the iterations. For performance details of the GNR scheme, see numerical examples in Section 5.5 below.

5.3. ACCURACY

In **Chapter 2**, the truncation or discretisation error of a recurrence equation is used to estimate the accuracy of algorithms. Each recurrence equation that approximates the exact one is terminated after a certain number of terms in the Taylor series expansion. The truncation error is a measure of accuracy and rate of convergence. For instance, the recurrence equation (7) of the *FBI* scheme, has a truncation error $O(\Delta x^3)$, $\forall \alpha_2 \in [0, 1]$, but the error is a minimum when $\alpha_2 = \frac{1}{2}$. The recurrence equation (10) of the *FB2* scheme has a truncation error $O(\Delta x^4)$ for $\alpha_2^2 = \frac{1}{3}$, and $O(\Delta x^3)$ otherwise. Similarly, using the same techniques, the recurrence equation (4) of the Newmark scheme has a truncation error $O(\Delta x^4)$ for $\beta = \frac{1}{6}$, and $O(\Delta x^3)$ otherwise. The recurrence equations (5, 8 and 11) have a truncation error $O(\Delta x^2)$ for $\gamma, \alpha_1 \neq \frac{1}{2}$ and $O(\Delta x^3)$ for $\gamma = \alpha_1 = \frac{1}{2}$. The choice of $\gamma = \alpha_1 = \frac{1}{2}$ provides the best accuracy for all the algorithms considered above; see ref. [19].

5.4. STABILITY

One important observation is that if a solution procedure is stable with no external loading, then it will also be stable if the external force is non zero but bounded. A number of forms of stability analysis has been established, such as:

- Spectral or Fourier stability which examines the effects of a time integration method on a single equation of motion obtained by modally uncoupling the original structure equations.

- Routh-Hurwitz criterion: The procedure involves the application of the so called *z-transformation* which maps the unit circle into the imaginary axis, the interior of the circle into the half-plane; and invoking the Routh-Hurwitz criterion which gives necessary and sufficient conditions for the root of a polynomial to have negative and real parts. For details, see, for example, Wood [14] or Zienkiewicz et al [16].
- Root-locus method (RLM) which follows a similar procedure as the Routh-Hurwitz criterion, and is popular in Control Engineering Systems analysis. For details, see, for example, Park et al [20].
- Energy stability which does not involve modal uncoupling of structure matrices. For details, see, for example, Wood [14] or Chapter 4, Section 4.3.4. The notions of **A**- and **B**-stability are found in Simo et al [21] and its references.

5.4.1. Spectral Stability Analysis

The stability analyses of commonly used formulae have been performed, for example, those of the central difference, generalised trapezoidal and Newmark recurrence equations; see ref. [14] for instance. In this section, the stability analysis of the *FBI* scheme is given.

The procedure consists of eliminating the accelerations from the set of equations (7 and 8) by substituting the accelerations given by:

$$(12) \quad \ddot{\mathbf{a}}_n = \mathbf{M}^{-1}[\mathbf{f}_n - \mathbf{C}\dot{\mathbf{a}}_n - \mathbf{K}\mathbf{a}_n],$$

$$(13) \quad \ddot{\mathbf{a}}_{n+1} = \mathbf{M}^{-1}[\mathbf{f}_{n+1} - \mathbf{C}\dot{\mathbf{a}}_{n+1} - \mathbf{K}\mathbf{a}_{n+1}].$$

Definitions: Let

$$(14) \quad \mathbf{y}_n = \begin{pmatrix} \mathbf{a}_n \\ \dot{\mathbf{a}}_n \end{pmatrix}, \quad \mathbf{y}_{n+1} = \begin{pmatrix} \mathbf{a}_{n+1} \\ \dot{\mathbf{a}}_{n+1} \end{pmatrix},$$

and consider a single equation of motion obtained by modally uncoupling the original structure equations (from now on in this subsection, matrices \mathbf{M} , \mathbf{C} , and \mathbf{K} reduce to scalars m , c , and k respectively). Then, substituting equations (12 and 13) into (7 and 8), and using the definitions given by equation (14) results in

$$(15) \quad \mathbf{y}_{n+1} = \mathbf{A}^{-1}[\mathbf{B} \cdot \mathbf{y}_n + \mathbf{l}_n],$$

where \mathbf{A} and \mathbf{B} are square matrices given by

$$(16) \quad \mathbf{A} = \begin{bmatrix} 1 - \frac{\Delta t^2}{2} \alpha_2^2 \frac{k}{m} & -\Delta t \alpha_2 \left(1 + \frac{\Delta t}{2} \alpha_2 \frac{c}{m}\right) \\ \Delta t \alpha_1 \frac{k}{m} & 1 + \Delta t \alpha_1 \frac{c}{m} \end{bmatrix},$$

$$(17) \quad \mathbf{B} = \begin{bmatrix} 1 - \frac{\Delta t^2}{2} (1 - \alpha_2)^2 \frac{k}{m} & \Delta t (1 - \alpha_2) \left[1 - \frac{\Delta t}{2} (1 - \alpha_2) \frac{c}{m}\right] \\ -\Delta t (1 - \alpha_1) \frac{k}{m} & 1 - \Delta t (1 - \alpha_1) \frac{c}{m} \end{bmatrix},$$

and the vector \mathbf{l} is given by

$$(18) \quad \mathbf{l} = \begin{pmatrix} \frac{\Delta t^2}{2} m^{-1} \left[(1 - \alpha_2)^2 f_n - \alpha_2^2 f_{n+1} \right] \\ \Delta t m^{-1} \left[(1 - \alpha_1) f_n + \alpha_1 f_{n+1} \right] \end{pmatrix}.$$

The product of the matrices \mathbf{A}^{-1} and \mathbf{B} is defined as the *amplification factor*, and for spectral stability

$$(19) \quad \|\mathbf{A}^{-1}\mathbf{B}\|_2 \leq 1.$$

For *unconditional* stability of the system with physical damping, this results [22] in

$$(20) \quad \alpha_1 \geq \frac{1}{2}, \text{ and } \alpha_1(2\alpha_2 - 1) - \alpha_2^2 \geq 0,$$

and for *conditional* stability of the system without physical damping, in

$$(21) \quad \frac{\Delta t^2 k}{2m} \leq \frac{1}{\alpha_2^2 - 2\alpha_1\alpha_2 + \alpha_1}.$$

Note that:

- The *FB1* scheme's unconditional stability result given in equation (20) is confirmed using the energy method in **Chapter 4**, Section 4.3.4 and Section 5.4.2 below. It can be shown that the conditions given by equation (20) reduce to $\alpha_1 = \alpha_2 = 1$. The conditional stability result given in equation (21) is also confirmed in Section 5.4.2 below.
- Since the *FB2* scheme is identical to the Newmark scheme for $\alpha_1 = \gamma$ and $\alpha_2^2 = 2\beta$, its stability analysis (see **Chapter 4**, Section 4.3.4) need not be performed here.

5.4.2. The Energy Method

The basic idea of this method is to formulate an *energy norm* which is a measure of the solution given by an algorithm and which must not increase, that is, satisfies

$$(22) \quad \|\mathbf{a}_{n+1}\|_E \leq \|\mathbf{a}_n\|_E,$$

where \mathbf{a}_{n+1} is the numerical solution at time $(n+1)\Delta t$, and $\|\mathbf{a}_{n+1}\|_E$ is the energy norm. From the dynamic system of equation (1), the kinetic energy may be written as

$$(23) \quad T_n = \frac{1}{2} \dot{\mathbf{a}}_n^T \mathbf{M} \dot{\mathbf{a}}_n,$$

and the potential energy, as

$$(24) \quad U_n = \frac{1}{2} \mathbf{a}_n^T \mathbf{K} \mathbf{a}_n.$$

From *now on*, the algebraic definitions of **Chapter 2**, Section 2.3.3, will be used extensively. The change in kinetic energy may be written as

$$(25) \quad \begin{aligned} \Delta T = T_{n+1} - T_n &= \frac{1}{2} (\dot{\mathbf{a}}_n + \dot{\mathbf{a}}_{n+1})^T \mathbf{M} (\dot{\mathbf{a}}_{n+1} - \dot{\mathbf{a}}_n) \\ &= \langle \dot{\mathbf{a}}_n \rangle^T \mathbf{M} \Delta \dot{\mathbf{a}}, \quad (\mathbf{M} - \text{symmetric}) \end{aligned}$$

and the change in potential energy, as

$$(26) \quad \Delta U = U_{n+1} - U_n = \langle \mathbf{a}_n \rangle^T \mathbf{K} \Delta \mathbf{a}, \quad (\mathbf{K} - \text{symmetric}).$$

Ignoring the effect of viscous damping, the change in the total energy becomes

$$(27) \quad \Delta E = \Delta T + \Delta U = \langle \dot{\mathbf{a}}_n \rangle^T \mathbf{M} \Delta \dot{\mathbf{a}} + \langle \mathbf{a}_n \rangle^T \mathbf{K} \Delta \mathbf{a}.$$

Using the *FBI* scheme's (**Chapter 2**, Section 2.3.3; equations (49 and 51)) recurrence equations results in

$$(28) \quad \Delta E = \left[\Delta \mathbf{a} - \frac{\Delta t^2}{2} (2\alpha_1 \alpha_2 - \alpha_1 - \alpha_2^2) \Delta \ddot{\mathbf{a}} \right]^T \mathbf{M} [\langle \ddot{\mathbf{a}} \rangle + \Delta \ddot{\mathbf{a}}] + \langle \mathbf{a} \rangle^T \mathbf{K} \Delta \mathbf{a}.$$

Making use of the collocation differential equations

$$(29) \quad \mathbf{M}\ddot{\mathbf{a}}_j + \mathbf{K}\mathbf{a}_j = \mathbf{0}, \quad j = n, (n+1),$$

so that $\mathbf{M}\langle\ddot{\mathbf{a}}_n\rangle = -\mathbf{K}\langle\mathbf{a}_n\rangle$, $\mathbf{M}\Delta\ddot{\mathbf{a}} = -\mathbf{K}\Delta\mathbf{a}$, $\Delta\ddot{\mathbf{a}} = -\mathbf{M}^{-1}\mathbf{K}\Delta\mathbf{a}$, results in

$$(30) \quad \begin{aligned} \Delta E = & -\frac{\Delta t^2}{2}(2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2)\Delta\ddot{\mathbf{a}}^T\mathbf{M}\langle\ddot{\mathbf{a}}_n\rangle \\ & -\frac{\Delta t^2}{2}(2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2)(\alpha_1 - \frac{1}{2})\Delta\ddot{\mathbf{a}}^T\mathbf{M}\Delta\ddot{\mathbf{a}}, \\ & -(\alpha_1 - \frac{1}{2})\Delta\mathbf{a}^T\mathbf{K}\Delta\mathbf{a} \end{aligned}$$

or

$$(31) \quad \begin{aligned} \Delta E = & -\frac{\Delta t^2}{2}(2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2)\Delta\ddot{\mathbf{a}}^T\mathbf{M}\langle\ddot{\mathbf{a}}_n\rangle \\ & -(\alpha_1 - \frac{1}{2})\Delta\mathbf{a}^T\left[\mathbf{I} + \frac{\Delta t^2}{2}(2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2)\mathbf{M}^{-1}\mathbf{K}\right]\mathbf{K}\Delta\mathbf{a} \end{aligned}$$

Equation (31) suggests an energy norm of the form

$$(32) \quad \begin{aligned} \hat{E}_n = & \frac{1}{2}\dot{\mathbf{a}}_n^T\mathbf{M}\dot{\mathbf{a}}_n + \frac{1}{2}\mathbf{a}_n^T\mathbf{K}\mathbf{a}_n + \frac{1}{2}\ddot{\mathbf{a}}_n^T\mathbf{M}\ddot{\mathbf{a}}_n \\ = & \frac{1}{2}\dot{\mathbf{a}}_n^T\mathbf{M}\dot{\mathbf{a}}_n + \frac{1}{2}\mathbf{a}_n^T\left[\mathbf{I} + \frac{\Delta t^2}{2}(2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2)\mathbf{M}^{-1}\mathbf{K}\right]\mathbf{K}\mathbf{a}_n \end{aligned}$$

so that

$$(33) \quad \hat{E}_{n+1} = \hat{E}_n - (\alpha_1 - \frac{1}{2})\Delta\mathbf{a}^T\left[\mathbf{I} + \frac{\Delta t^2}{2}(2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2)\mathbf{M}^{-1}\mathbf{K}\right]\mathbf{K}\Delta\mathbf{a}.$$

The energy norm defined in equation (32) is legitimate if

$$(34) \quad (2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2) \geq 0, \quad \text{or} \quad \mathbf{I} + \frac{\Delta t^2}{2}(2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2)\mathbf{M}^{-1}\mathbf{K} \geq \mathbf{0}.$$

From equation (33), the energy is non-increasing ($\hat{E}_{n+1} \leq \hat{E}_n$) if

$$(35) \quad \begin{aligned} & \text{(i) } \alpha_1 - \frac{1}{2} \geq 0 \text{ and } (2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2) \geq 0, \text{ or} \\ & \text{(ii) } \alpha_1 - \frac{1}{2} \geq 0 \text{ and } \mathbf{I} + \frac{\Delta t^2}{2}(2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2)\mathbf{M}^{-1}\mathbf{K} \geq \mathbf{0} \end{aligned}$$

The first of these equations corresponds to *unconditional* stability whereas the second one, *conditional* stability. These results are also obtained using the method of spectral stability in Section 5.4.1 above; see equations (20 and 21). However, using the energy method, the stability conditions are obtained without modal decomposition, and it can be shown that identical results are obtained when viscous damping is included. The resulting energy norm may be defined by

$$(36) \quad \hat{E}_n = \frac{1}{2} \dot{\mathbf{a}}_n^T \mathbf{K} \dot{\mathbf{a}}_n + \frac{1}{2} \ddot{\mathbf{a}}_n^T \left[\mathbf{M} + \frac{\Delta t^2}{2} (2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2) \mathbf{K} \right] \ddot{\mathbf{a}}_n,$$

with the corresponding algorithm written in the form

$$(37) \quad \begin{aligned} \hat{E}_{n+1} = \hat{E}_n & - \frac{1}{\Delta t} \Delta \dot{\mathbf{a}}^T \mathbf{C} \Delta \dot{\mathbf{a}} \\ & - (\alpha_1 - \frac{1}{2}) \Delta \ddot{\mathbf{a}}^T \left[\mathbf{M} + \frac{\Delta t^2}{2} (2\alpha_1\alpha_2 - \alpha_1 - \alpha_2^2) \mathbf{K} \right] \Delta \ddot{\mathbf{a}} \end{aligned}$$

Note that the energy norm does not look like the physical energy. The stability analysis of a system with non-linear material properties is performed in **Chapter 4**, Section 4.3.4.

5.5. NUMERICAL EXAMPLES

In this section, comparative parametric studies of the generalised Newton-Raphson (GNR) scheme are carried out using the integration schemes:

- The Newmark scheme with integration parameters $2\beta \geq \gamma \geq \frac{1}{2}$, termed the $\text{GNR}_k(\beta)$ -GN22(β, γ) scheme, where GN22 stands for the generalised Newmark algorithm [14, 16] using an approximation of degree two for the unknown function in the time step and is applicable to numerical solution of differential equations of order two. Note that the GNR scheme's scalar $\beta \in [0, 1]$ is not related to the Newmark integration parameters $\beta, \gamma \in [0, 1]$. The subscript $(\cdot)_k$ represents the version number as defined in Section 5.2.2 above.
- The *FBI* scheme with integration parameters $\alpha_1 = \alpha_2 = 1$, termed the $\text{GNR}_k(\beta)$ -FBI(α_2, α_1) scheme.

Note that the conventional Newton-Raphson scheme used in combination with the integration schemes will be termed, for example, as the CNR-FBI(α_2, α_1). The *FB2* scheme is the same as the Newmark scheme and only a comparative study between the latter and the *FBI* scheme need be done.

Since only non-linear (elastoplastic) numerical examples are considered, the choice of integration parameters is confined to the *unconditional* stability region. The aim of these studies is to show that for a very large time-step (violating the incremental strain laws), the

above schemes used with the conventional Newton-Raphson (CNR) scheme, may not converge; whereas, if they are used with the GNR scheme, convergence may be enhanced.

5.5.1. Plane Stress Problem

A finite element model taken from Zienkiewicz et al [16] is used to test the algorithms developed; see Figure 3. This is a quarter model of a perforated tension strip. In this example, plane stress conditions are assumed, and the von Mises criterion is used with a constant slope, H , of the uniaxial hardening curve.

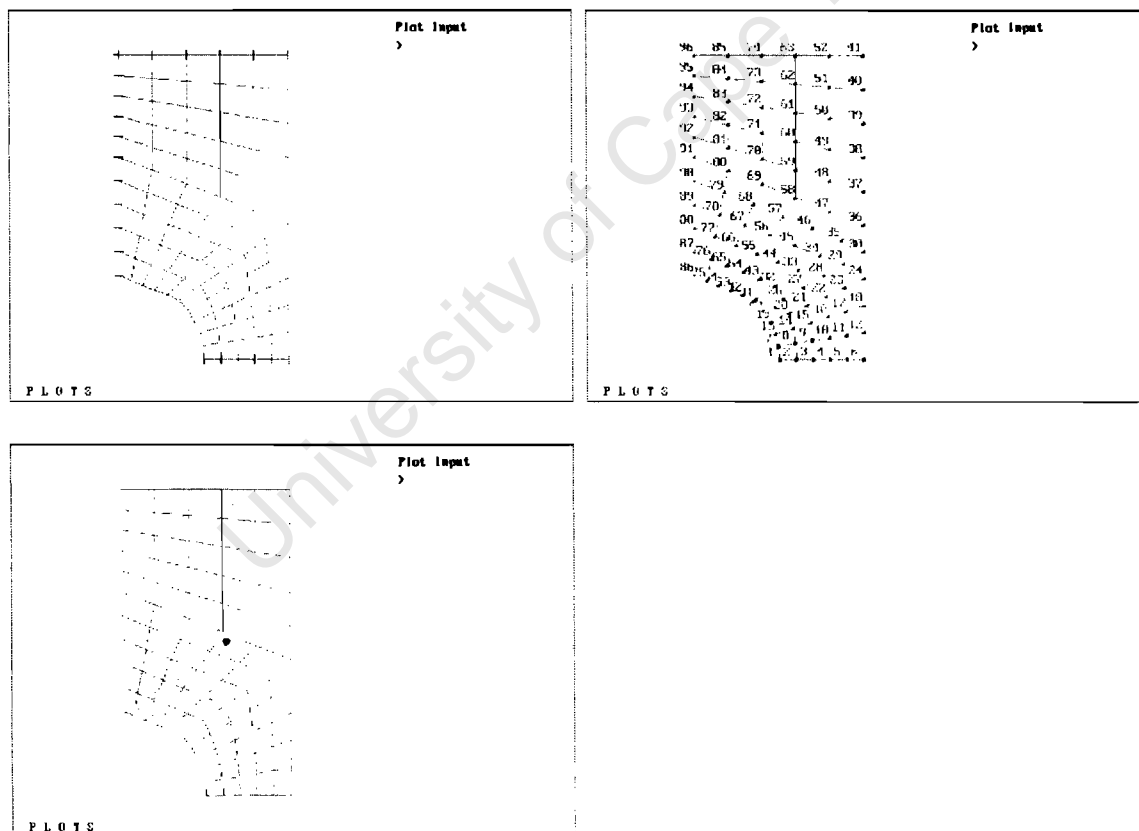


Figure 3 Perforated tension strip: l-r, t-b; (a) boundary restraints and applied load, (b) finite element mesh showing node numbering, (c) showing element numbering.

Figure 4 shows the results (22-stress contours) of an incremental process of loading,

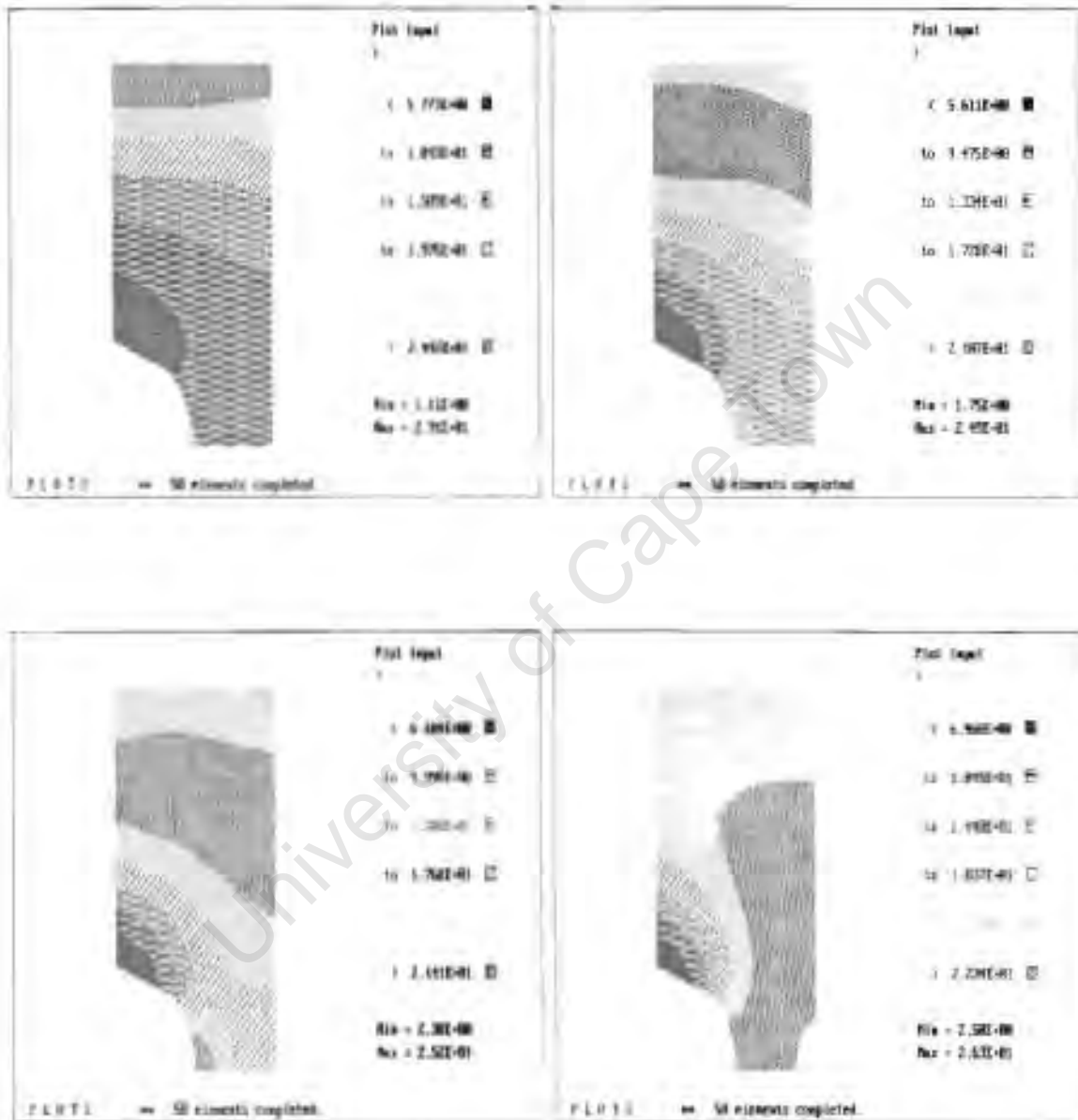


Figure 4 (a) Perforated tension strip (plane stress) l-r, t-b, time = 0.04, 0.05, 0.06, 0.07 (s), $E=7000$ (kg/mm²), $\nu = 0.2$; $\sigma_v = 24.3$ (kg/mm²), $H/E=0.032$ (Strain Hardening)

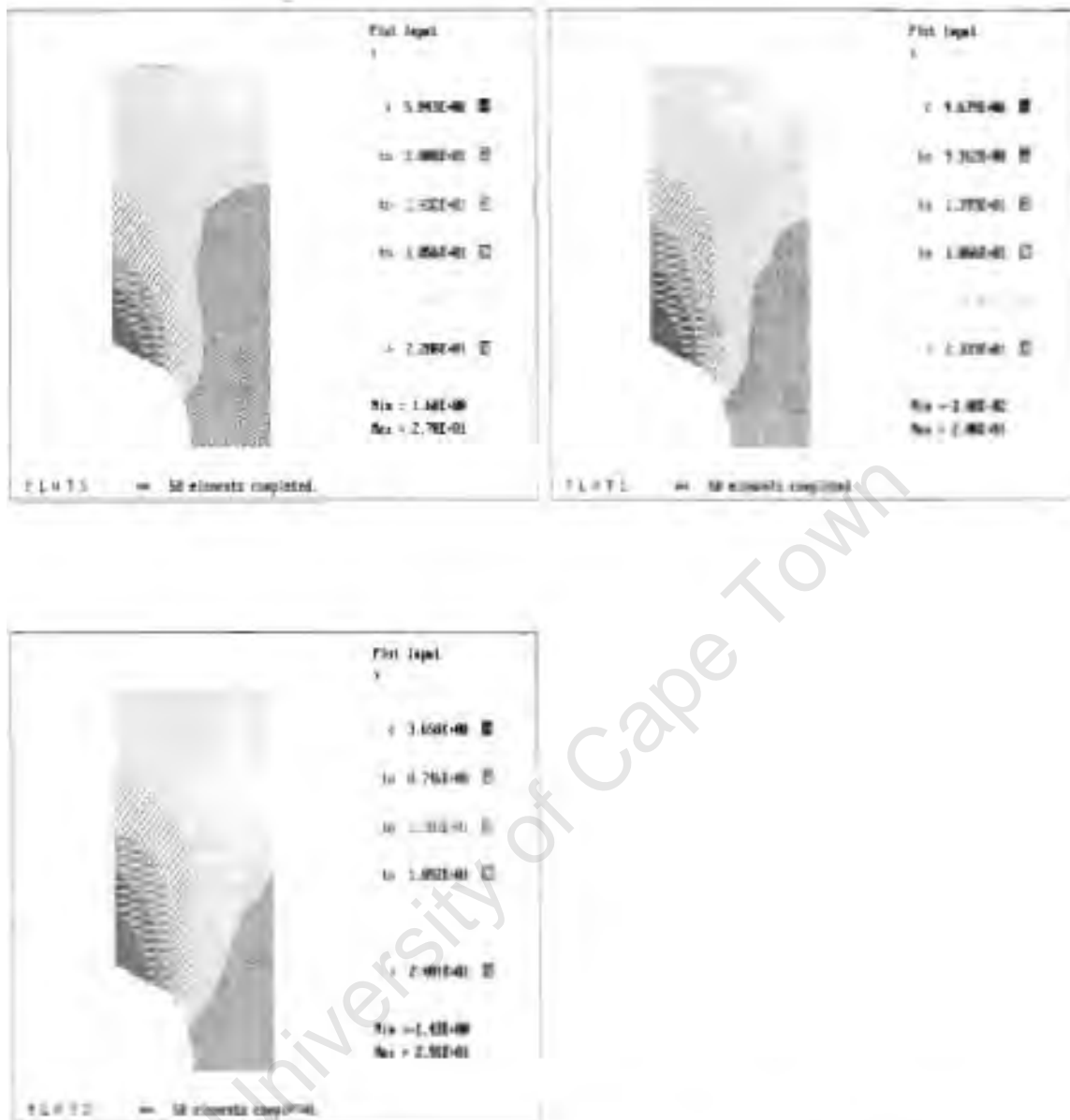


Figure 4 (b) Perforated tension strip (plane stress): l-r, t-b, time = 0.08, 0.09, 0.1 (s); $E=7000$ (kg/mm²), $\nu = 0.2$, $\sigma_y = 24.3$ (kg/mm²), $H/E=0.032$ (Strain Hardening).

whereas Figure 5 shows results (22-stress contours and nodal displacement vectors) of a process where loads are applied in a large single time-step.

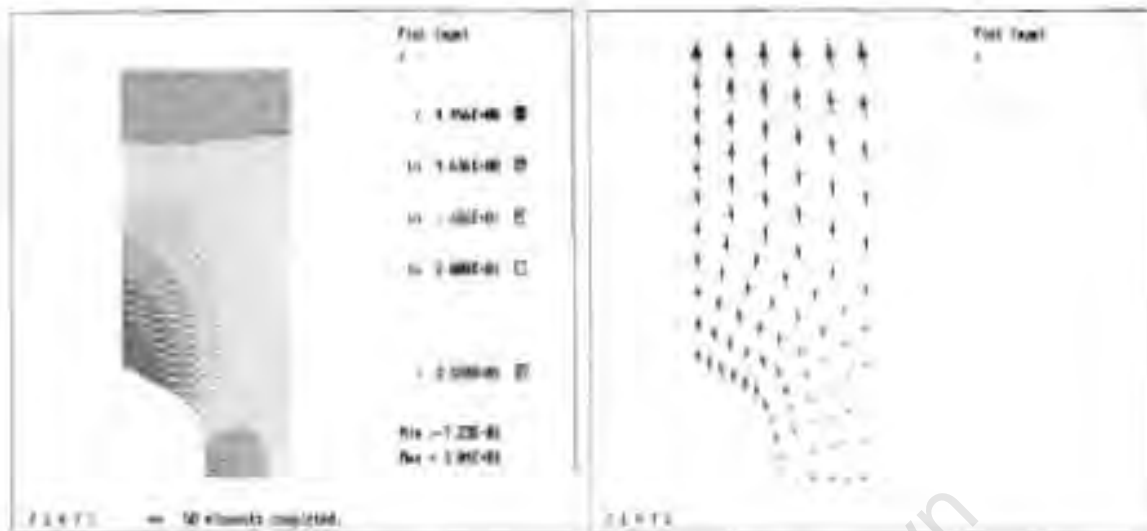


Figure 5 Perforated tension strip (plane stress): single time-step, $\Delta t=0.1$ (s).
 (a) 22-stress contours, (b) nodal displacement vectors.

The $GNR_1(0.9)$ -FB1(1,1) scheme was used. The initial stress process still yields an equilibrating solution and one which does not exceed the yield surface. It is of interest to note that despite the violation of the incremental strain laws, similar results for plastic zones are achieved and that the CNR-FB1(1,1) could not converge for the large single time-step. This clearly shows that the GNR scheme can enhance the convergence of algorithms.

Figure 6 shows results of varying the GNR parameters. Norms of energy and residual force are plotted against the iteration number. At a single time-step of $\Delta t=0.04$ (s) which represents the point of yield, the algorithm converged for integration parameters of $\beta \in [0.4, 1]$. Note that this implies that convergence was

also achieved for the conventional Newton-Raphson scheme ($\beta=1$); see Figure 6

(a).

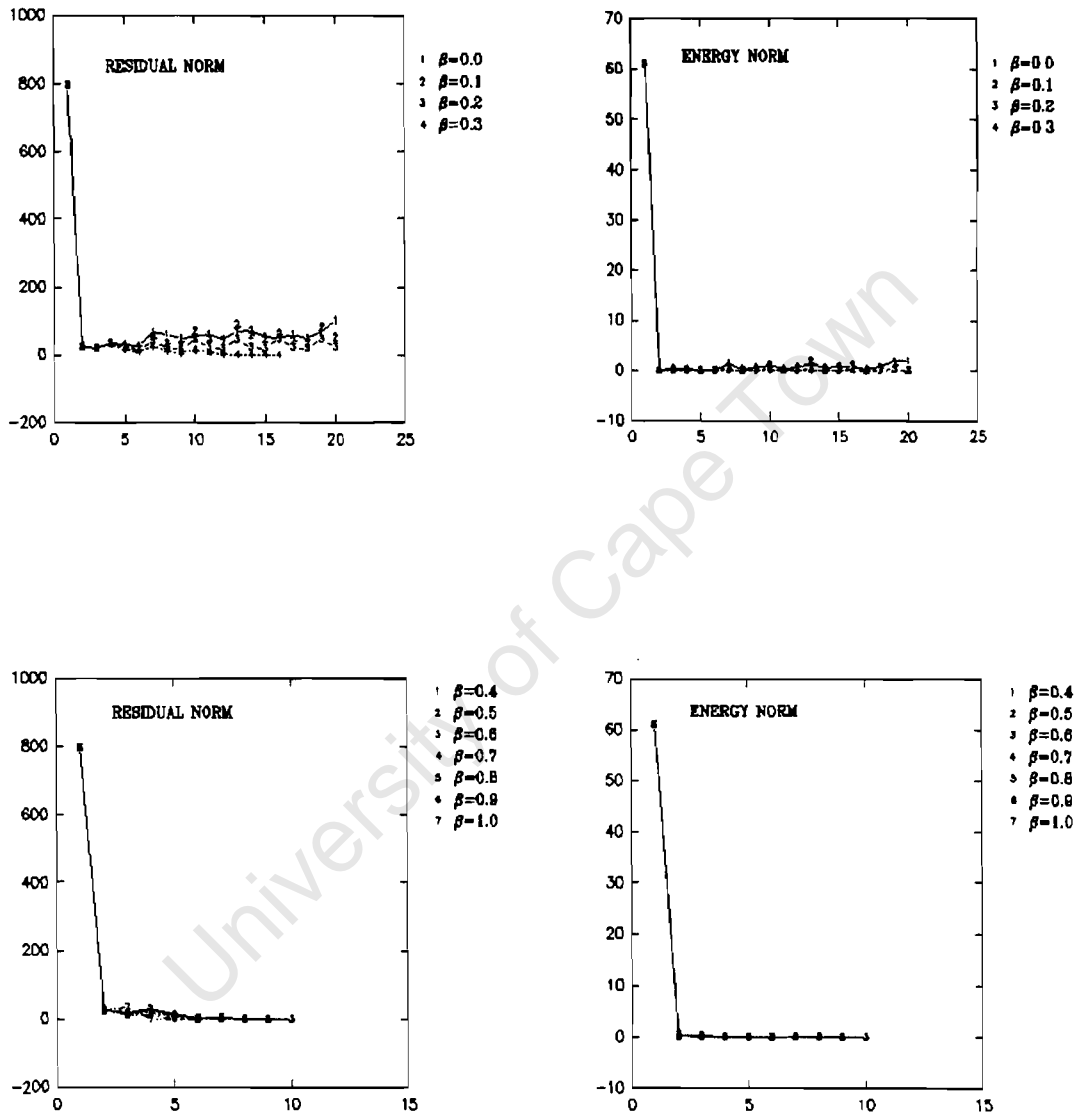


Figure 6 (a) Perforated tension strip (plane stress): Graphs of Norms Vs Iteration No. for $\Delta t=0.4$ (s); showing convergence of iteration process for $0.4 \leq \beta \leq 1$.

In contrast, for a large single time-step of $\Delta t=0.08$ (s), the algorithm converged for a narrower range of integration parameters ($0.5 < \beta < 1$). This implies that convergence was not achieved for the conventional Newton-Raphson scheme ($\beta=1$); see Figure 6 (b).

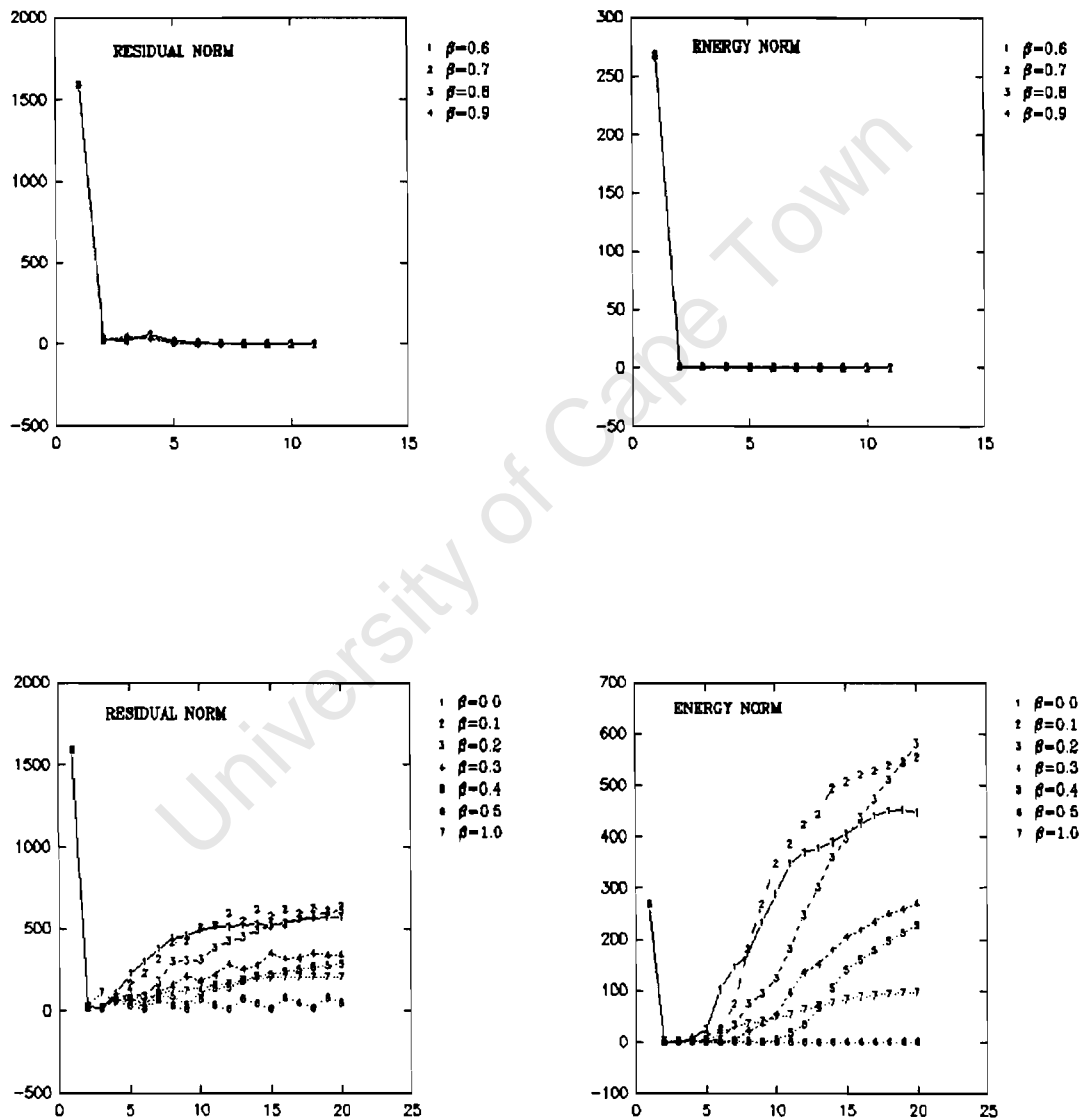


Figure 6 (b) Perforated tension strip (plane stress): Graphs of Norms Vs Iteration No. for $\Delta t=0.8$ (s), showing convergence of iteration process for $\frac{1}{2} < \beta < 1$.

Convergence of the iteration process was achieved for values of $0.5 < \beta < 1$ even for a large single time-step of $\Delta t > 0.1(\text{s})$, whereas the CNR - GN22($\frac{1}{4}, \frac{1}{2}$) scheme documented in ref. [16] failed to converge at a single time-step of $\Delta t = 0.06(\text{s})$ for the static case, and $\Delta t = 0.09(\text{s})$ for the dynamic case. *Note* that 12 or less iterations are typical for a converged iteration process with a tolerance of $(1\text{e-}20)$ in the energy norm.

It is of interest to note that the GNR scheme may be used with any method to enhance the convergence process. However, the $\text{GNR}_1(\beta)$ - FB1(1,1) scheme, failed to converge at a single time-step of $\Delta t = 0.3(\text{s})$. In contrast, the $\text{GNR}_1(\beta)$ - GN22($\frac{1}{4}, \frac{1}{2}$) scheme, failed to converge at a single time-step of $\Delta t = 0.4(\text{s})$. This time-step is more than four times that achieved by the CNR - GN22($\frac{1}{4}, \frac{1}{2}$) scheme. For this problem, the extended explicit versions provided remarkable improvement on the convergence characteristics; for example, the $\text{GNR}_3(\frac{3}{4})$ - FB1(1,1), $\text{GNR}_4(\frac{3}{4})$ - FB1(1,1), $\text{GNR}_3(\frac{3}{4})$ - GN22($\frac{1}{4}, \frac{1}{2}$) and $\text{GNR}_4(\frac{3}{4})$ - GN22($\frac{1}{4}, \frac{1}{2}$) versions converged even for large single time-steps of $\Delta t > 50(\text{s})$! Thus, with these versions, a time-step more than 500 times that achieved using the CNR - GN22($\frac{1}{4}, \frac{1}{2}$) scheme is possible.

5.5.2. Plane Strain Problem

The same finite element problem (perforated tension strip) of Section 5.5.2 is used to test the algorithms developed, but in this case, plane strain conditions are assumed. Similar results are obtained for an incremental process. However, for this

problem, convergence of the iteration process was achieved for values of $0.4 \leq \beta < 1$ for a large single time-step of $\Delta t = 0.2(\text{s})$ using the $\text{GNR}_1(\beta)$ -FB1(1,1) scheme, whereas the CNR - GN22($\frac{1}{4}, \frac{1}{2}$) scheme documented in ref. [16] failed to converge at a single time-step of $\Delta t = 0.2(\text{s})$. For this problem, the typical best overall performances of both schemes are summarised as follows:

- The $\text{GNR}_1(0.4)$ -GN22($\frac{1}{4}, \frac{1}{2}$) failed at a single time-step of $\Delta t = 0.5(\text{s})$; converged at a single time-step of $\Delta t = 0.4(\text{s})$ in 32 iterations (this represents a time-step that is more than twice that achieved by the CNR - GN22($\frac{1}{4}, \frac{1}{2}$) scheme), whereas the $\text{GNR}_1(\beta \in [\frac{3}{4}, 0.9])$ -FB1(1,1) failed at $\Delta t = 0.3(\text{s})$; converged at $\Delta t = 0.2(\text{s})$ in 11 iterations. This shows that by choosing integration parameters carefully, convergence of the algorithms can be enhanced.
- The $\text{GNR}_2(\frac{3}{4})$ -GN22($\frac{1}{4}, \frac{1}{2}$) failed at a single time-step of $\Delta t = 0.5(\text{s})$; converged at a single time-step of $\Delta t = 0.4(\text{s})$ in 12 iterations (again this represents a time-step that is more than twice that achieved by the CNR - GN22($\frac{1}{4}, \frac{1}{2}$) scheme), whereas the $\text{GNR}_2(\frac{3}{4})$ -FB1(1,1) also failed at $\Delta t = 0.5(\text{s})$; converged at $\Delta t = 0.4(\text{s})$ in 12 iterations. These results show that **Version 2** algorithms are superior to **Version 1** algorithms.
- The $\text{GNR}_3(0.9)$ -GN22($\frac{1}{4}, \frac{1}{2}$) failed at a single time-step of $\Delta t = 0.7(\text{s})$; converged at a single time-step of $\Delta t = 0.6(\text{s})$ in 10 iterations (this represents a time-step that is more than three times that achieved by the CNR - GN22($\frac{1}{4}, \frac{1}{2}$))

scheme), whereas the $\text{GNR}_3(\frac{3}{4})\text{-FB1}(1,1)$ failed at $\Delta t = 0.5(\text{s})$; converged at $\Delta t = 0.4(\text{s})$ in 12 iterations. These results reveal that **Version 3** algorithms for the $\text{FB1}(1,1)$ scheme have no advantages over **Version 2** algorithms, whereas **Version 3** algorithms for the $\text{GN22}(1/4,1/2)$ scheme are superior to **Version 2** algorithms.

- The $\text{GNR}_4(0.9)\text{-GN22}(\frac{1}{4},\frac{1}{2})$ failed at a single time-step of $\Delta t = 0.7(\text{s})$; converged at a single time-step of $\Delta t = 0.6(\text{s})$ in 10 iterations, whereas the $\text{GNR}_4(\frac{3}{4})\text{-FB1}(1,1)$ failed at $\Delta t = 0.5(\text{s})$; converged at $\Delta t = 0.4(\text{s})$ in 12 iterations. These results show that **Version 4** and **Version 3** algorithms for both schemes perform equally well in this example.
- In this example, **Version 2** algorithm is the optimal choice for the $\text{FB1}(1,1)$ scheme whereas **Version 3** algorithm is the optimal choice for the $\text{GN22}(1/4,1/2)$ scheme.

Another plane strain problem is shown in Figure 7 which is taken from Owen et al [11]. It is a finite element quarter model of a thick-walled cylinder subjected to internal pressure.

Figure 8 shows the results (yield level stress contours) of an incremental process of loading, whereas Figure 9 shows results (yield level stress contours) of a process where loads are applied in a large single time-step.

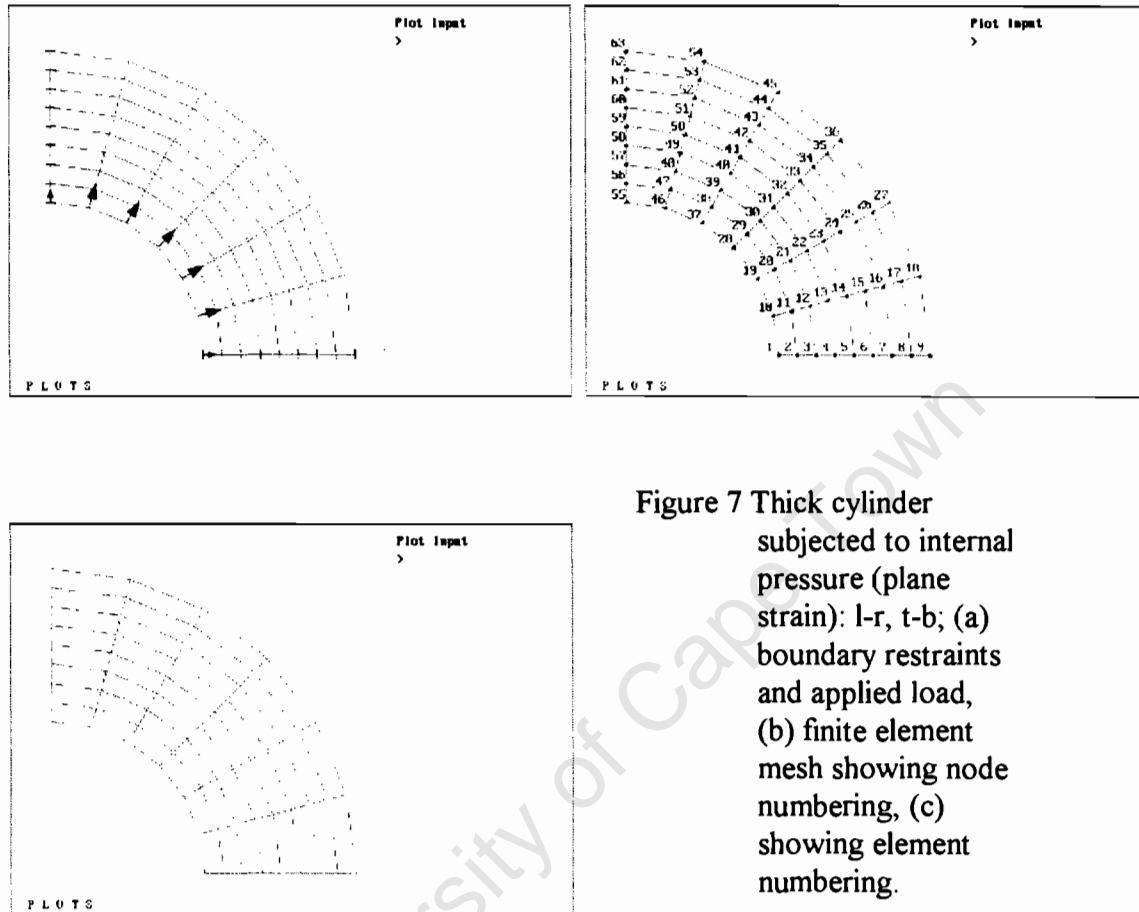


Figure 7 Thick cylinder subjected to internal pressure (plane strain): l-r, t-b; (a) boundary restraints and applied load, (b) finite element mesh showing node numbering, (c) showing element numbering.

University of Capetown

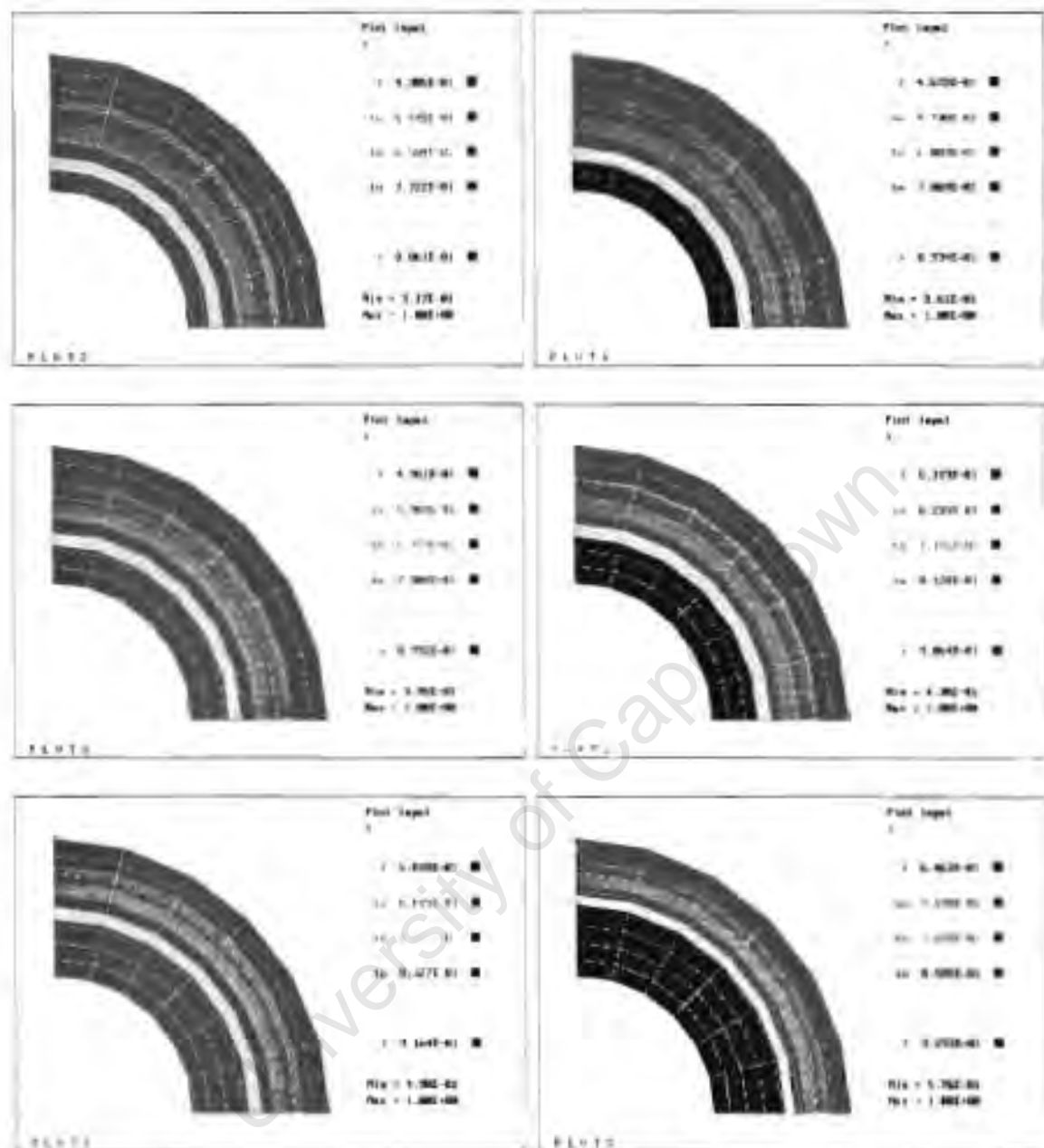


Figure 8 (a) Thick cylinder subjected to internal pressure (plane strain). $E=2.1 \times 10^4$ (dN/mm²), $\nu = 0.3$, $\sigma_y = 24$ (dN/mm²), $H/E=0$. (Strain Hardening) (I) Initially $\Delta t = 12$ (s), then $\Delta t = 1$ (s) for the rest of the process, (II) l-r, t-b; stress contours of yield level at time = 12, 13, 14, 15, 16, 17 (s).

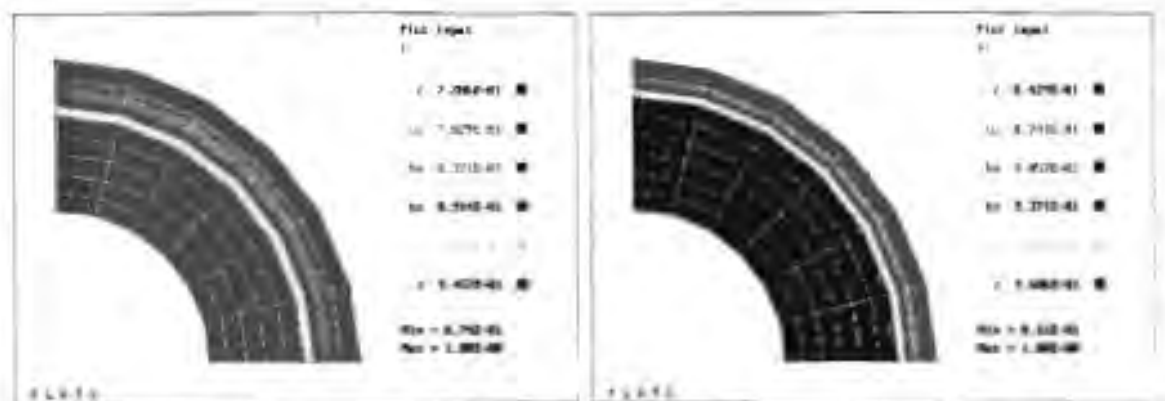


Figure 8 (b) t-r, t-b, stress contours of yield level at time = 18, 19 (s)

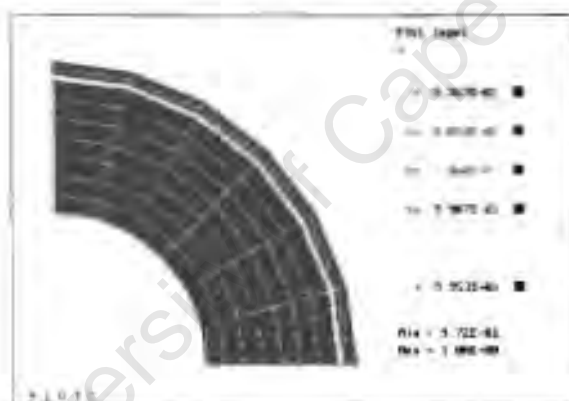


Figure 9 Thick cylinder subjected to internal pressure (plane strain) stress contours of yield level at a single time-step, $\Delta t = 19$ (s)

The $GNR_1(0.9)$ - $FBI(1,1)$ scheme was again used because the conventional Newton-Raphson scheme failed to converge for a large single time-step. It is of interest to note again that despite the violation of the incremental strain laws, similar results for plastic zones are achieved. Figure 10 shows pressure/radial displacement characteristics and circumferential (hoop) stress distributions for

various times across the thickness of the cylinder. The pattern of the distributions, as well as the pressure/radial displacement characteristics are similar to those given in ref. [11].

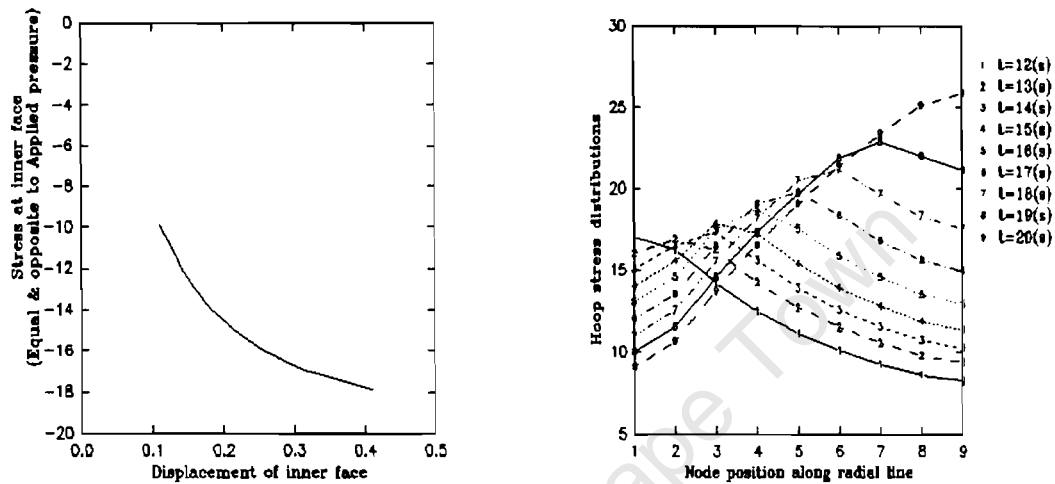


Figure 10 Thick cylinder subjected to internal pressure (plane strain): showing; (a) displacement of inner surface with increasing applied pressure, (b) circumferential (hoop) stress distributions across the thickness of the cylinder for various times during loading.

5.5.3. Axisymmetric Problem

The above numerical example of the thick-walled cylinder subjected to internal pressure is resolved using an axisymmetric finite element model; see Figure 11.

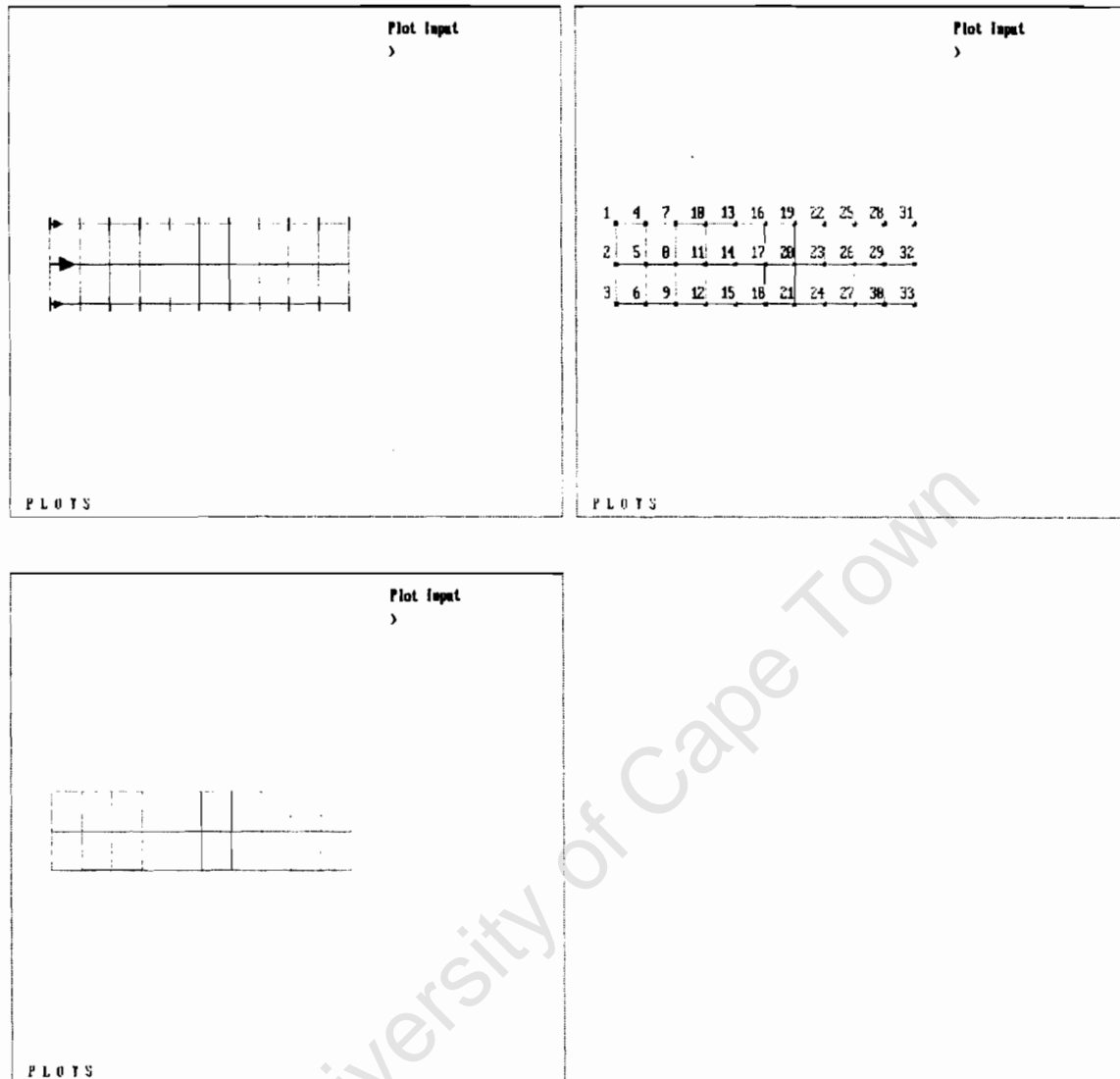


Figure 11 Thick cylinder subjected to internal pressure (axisymmetric): l-r, t-b;
 (a) boundary restraints and applied load, (b) finite element mesh showing node numbering, (c) showing element numbering.

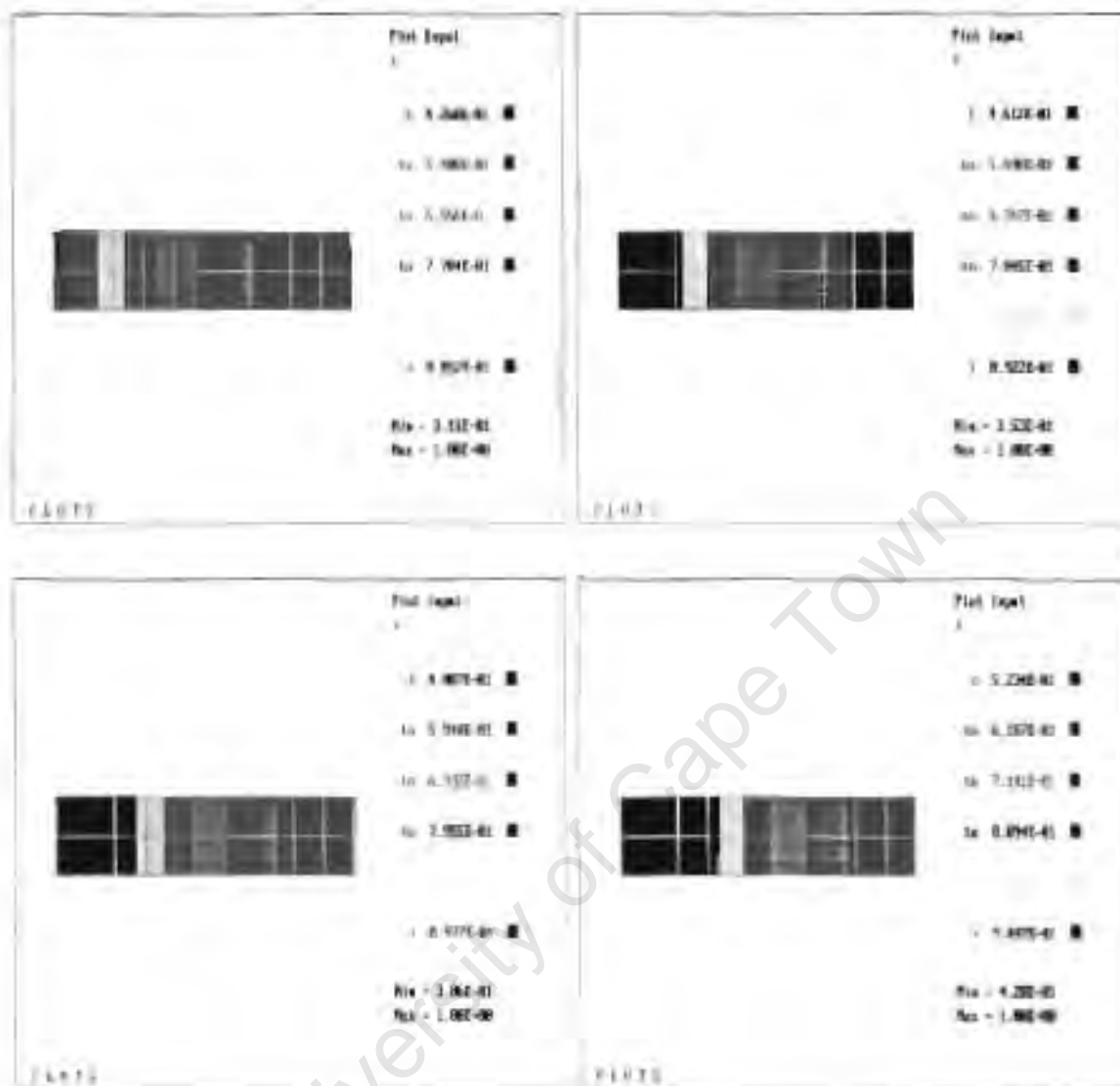


Figure 12 (s) Thick cylinder subjected to internal pressure (axisymmetric):
 $E=2.1 \times 10^4$ (dN/mm²); $\nu = 0.3$; $\sigma_y = 24$ (dN/mm²); $H/E=0$. (Strain Hardening). (I) Initially $\Delta t = 12$ (s), then $\Delta t = 1$ (s) for the rest of the process, (II) l-r, t-b; stress contours of yield level at time = 12, 13, 14, 15 (s).

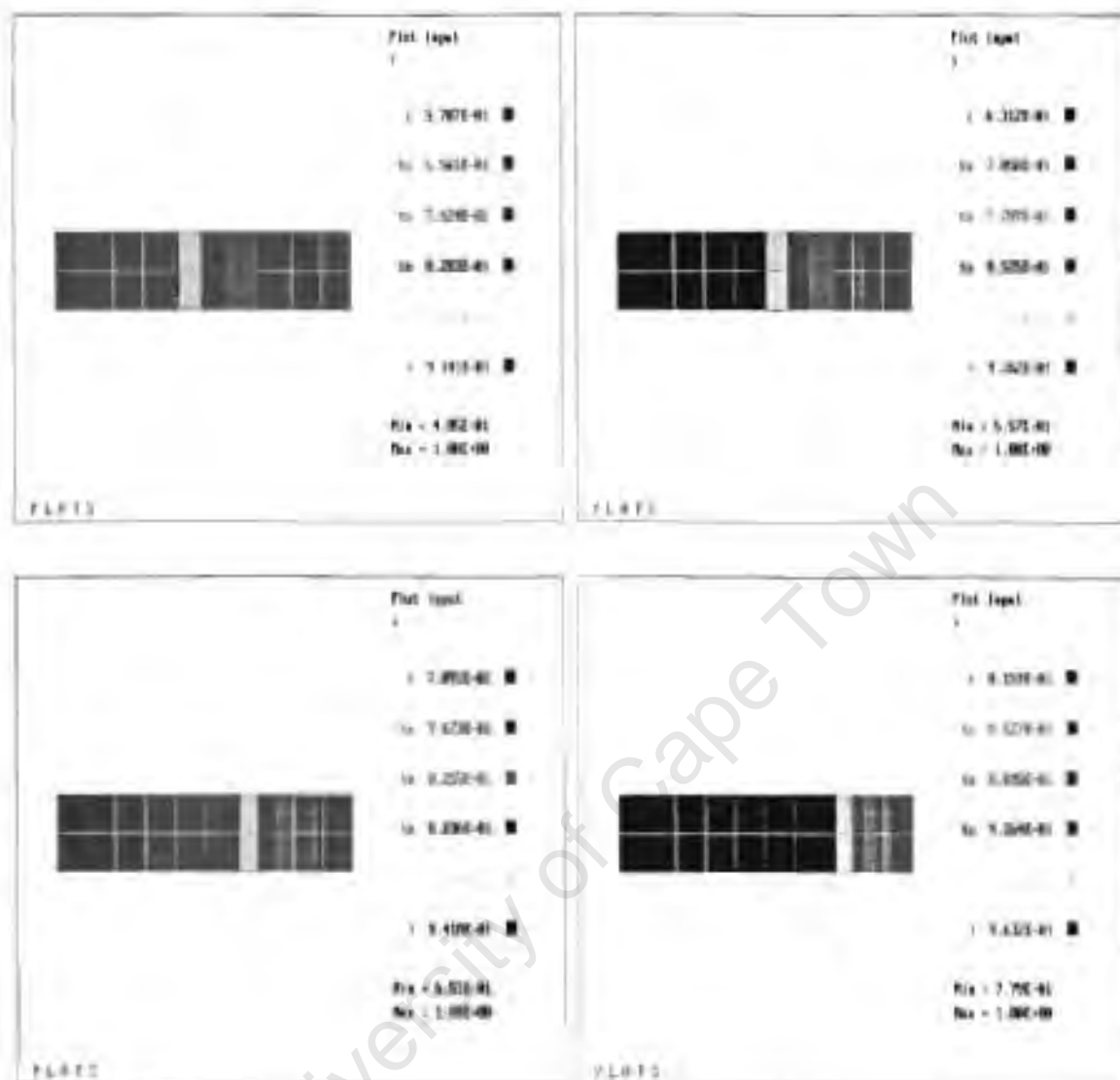


Figure 12 (b) l-r, t-b, stress contours of yield level at time = 16, 17, 18, 19 (s)

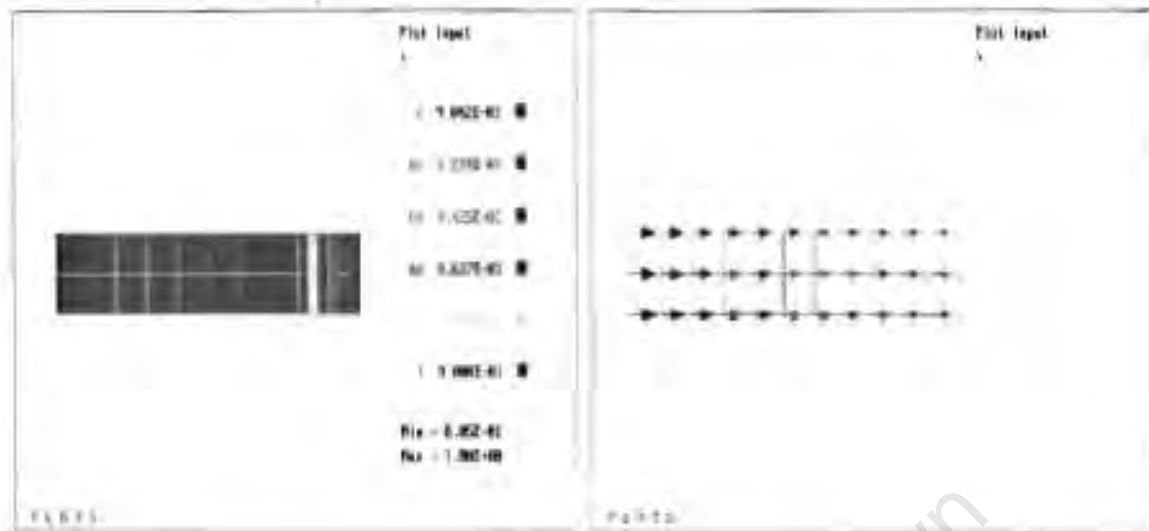


Figure 13 Thick cylinder subjected to internal pressure (axisymmetric) (a) stress contours of yield level at a single time-step, $\Delta t = 19$ (s), (b) corresponding displacement vectors

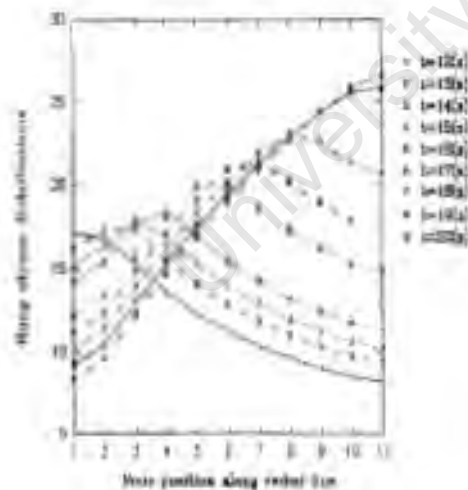


Figure 14 Thick cylinder subjected to internal pressure (axisymmetric) Circumferential (hoop) stress distribution across the cylinder thickness

Note the similarities between results (yield level stress contours) of the plain strain and axisymmetric models:

- The yield levels for various times of loading are similar along the thickness of the cylinder as shown in Figures 8 and 12.
- For both models, the yield levels for incremental and large steps are in good agreement; see for example, at time equal to 19 (s) in Figures 9 and 13. This shows that large time-steps can be used to reduce costs of computations associated with an incremental process.
- The circumferential (hoop) stress distributions for various times across the thickness of the cylinder are similar for both models (also similar to those of ref. [11]); see Figures 10 and 14.
- The similarities in the results for both models verify the fidelity of the solutions.

5.6. CONCLUSIONS

A comparative study of the Newmark scheme and schemes developed in **Chapters 2-4** has been carried out. Additional schemes have been developed in this chapter in the form of extensions to those developed in earlier chapters; see Section **5.2.2**.

To consolidate parametric studies of integration schemes, further stability analyses have been performed using the energy and spectral stability methods. Both methods yield the same results; see Sections **5.4.1** and **5.4.2**.

Note the similarities between results (yield level stress contours) of the plain strain and axisymmetric models:

- The yield levels for various times of loading are similar along the thickness of the cylinder as shown in Figures 8 and 12.
- For both models, the yield levels for incremental and large steps are in good agreement; see for example, at time equal to 19 (s) in Figures 9 and 13. This shows that large time-steps can be used to reduce costs of computations associated with an incremental process.
- The circumferential (hoop) stress distributions for various times across the thickness of the cylinder are similar for both models (also similar to those of ref. [11]); see Figures 10 and 14.
- The similarities in the results for both models verify the fidelity of the solutions.

5.6. CONCLUSIONS

A comparative study of the Newmark scheme and schemes developed in **Chapters 2-4** has been carried out. Additional schemes have been developed in this chapter in the form of extensions to those developed in earlier chapters; see Section **5.2.2**.

To consolidate parametric studies of integration schemes, further stability analyses have been performed using the energy and spectral stability methods. Both methods yield the same results; see Sections **5.4.1** and **5.4.2**.

Numerical examples of Section 5.5 have demonstrated that the generalised Newton-Raphson (GNR) scheme can considerably enhance convergence of the conventional Newton-Raphson (CNR) scheme. The GNR scheme could even converge quadratically when a large single time-step was used (without loss of accuracy); this is a considerable advantage over the CNR scheme, which can reduce costs of computations associated with an incremental process.

REFERENCES

- [1] Houbolt J C, "A recurrence matrix solution for the dynamic response of elastic aircraft", *J. Aero. Sci.*, **17**, 540-550, (1950).
- [2] Newmark N M, "A method of computation for structural dynamics", *J. Engng Mech. Div.*, ASCE, **85**, EM3, 67-94, (1959).
- [3] Wilson E L, "A computer program for the dynamic stress analysis of underground structures", *S.E.L Report 68-1*, University of California, Berkeley, (1968).
- [4] Hilber H M, Hughes T J R, Taylor R L, "Improved numerical dissipation for time integration algorithms in structural dynamics", *Int. J. Earthquake Eng. Struct. Dynam.*, **5**, 283-292, (1977).
- [5] Zienkiewicz O C, "A new look at the Newmark, Houbolt and other time-stepping formulas. A weighted residual approach", *Int. J. Earthquake Eng. Struct. Dynam.*, **5**, 413-418, (1977).
- [6] Hilber H M, Hughes T J R, "Collocation, dissipation and 'overshoot' for time integration schemes in structural dynamics", *Int. J. Earthquake Eng. Struct. Dynam.*, **6**, 99-117, (1978).
- [7] Wood W L, "A further look at Newmark, Houbolt, etc., time-stepping formulae", *Int. J. Num. Meth. Eng.*, **20**, 1009-1017, (1984).
- [8] Zienkiewicz O C, Wood W L, Hine N W, Taylor R L, "A unified set of single-step algorithms, Part 1: general formulation and applications", *Int. J. Num. Meth. Eng.*, **20**, 1529-1552, (1984).
- [9] Wood W L, "A unified set of single-step algorithms, Part 2: theory", *Int. J. Num. Meth. Eng.*, **20**, 2303-2309, (1984).
- [10] Katona M G, Zienkiewicz O C, "A unified set of single-step algorithms, Part 3: the beta-m method, a generalisation of the Newmark scheme", *Int. J. Num. Meth. Eng.*, **21**, 1345-1359, (1985).

-
- [11] Owen D R J, Hinton E, *Finite Elements in Plasticity: Theory and Practice*, Pineridge Press Limited, 1980.
- [12] Bathe K J, *Finite Element Procedures in Engineering Analysis*, Prentice Hall, Inc., 1982.
- [13] Hughes T J R, *The Finite Element Method, Linear Static and Dynamic Finite Element Analysis*, Prentice Hall, Inc., 1987.
- [14] Wood W L, *Practical Time-setting Schemes*, Clarendon Press, Oxford, 1990.
- [15] Zienkiewicz O C, Taylor R L, *The Finite Element Method, Volume 1, Basic Formulations and Linear Problems*, 4th Edition, 1989.
- [16] Zienkiewicz O C, Taylor R L, *The Finite Element Method, Volume 2, Solid and Fluid Mechanics, Dynamics and Non-linearity*, 4th Edition, 1991.
- [17] Henrici P, *Discrete Variable Methods in Ordinary Differential Equations*, John Wiley and Sons, Inc., 1962.
- [18] Henrici P, *Elements of Numerical Analysis*, John Wiley and Sons, Inc., 1964.
- [19] Kaunda M A E, Martin J B, "Finite element solution algorithms for non-linear time-dependent/independent problems in solid mechanics", In *Proc. 19th South African Symposium on Numerical Mathematics (SANUM)*, 95-117, (1993).
- [20] Park K C, Felippa C A, "Partitioned analysis of coupled systems", in *Computational Methods for Transient Analysis*, eds. T Belytschko and T J R Hughes, Elsevier Science Publishers B. V., (1983).
- [21] Simo J C, Govindjee S, "Non-linear B-stability and symmetry preserving return mapping algorithms for plasticity and viscoplasticity", *Int. J. Num. Meth. Eng.*, 151-176, (1991).
- [22] Kaunda M A E, Martin J B, "On single-step time-integrators", In *Proc. 18th South African Symposium on Numerical Mathematics (SANUM)*, 78-101, (1992).

CHAPTER 6

CONCLUSIONS

Continuum and finite element formulations of the static and dynamic *initial-boundary-value* evolution (elastoplastic) problems have been considered in terms of both the classical and internal variable frameworks. In particular, the latter framework has facilitated the development of solution algorithms in the form of the convex mathematical programming and Newton-Raphson schemes. In this development, the latter scheme has been linked to the former in the sense that it expresses the conditions under which the convex non-linear function can be minimised.

Solution algorithms for the dynamic initial-boundary-value evolution problems have been developed using the generalised trapezoidal rule in combination with the generalised Newton-Raphson scheme. Algorithms for the static case are recovered from those of the dynamic case when $\mathbf{M} = \mathbf{C} = \mathbf{0}$ and $\dot{\mathbf{a}} = \ddot{\mathbf{a}} = \mathbf{0}$. A Taylor series expansion in time and space has been extensively employed to derive the integration schemes. This approach has provided:

- some theoretical foundations of the generalised trapezoidal rule and generalised Newton-Raphson scheme that have some geometrical insights as well as an interpretation in terms of finite differences and calculus. Recall that the time interval is sub-divided into two intervals; in the first one, a forward difference scheme is used whereas in the second, a backward difference scheme; see the derivations in **Chapter 2**.

Conventionally, one way of interpreting the generalised trapezoidal rule is that it uses a weighted average of values (such as velocity or acceleration) at the two ends of the time interval.

- some simple estimates of accuracy of recurrence equations in terms of truncation errors.

In this dissertation, the generalised trapezoidal scheme has been shown to be a special case of the *forward-backward* difference scheme for solving first order differential equations. It includes the Euler forward and backward difference schemes as special cases when the integration parameters are set to $\alpha = 0$ and $\alpha = 1$, respectively. For solution of second order differential equations, two schemes have been developed: the *FB1* and *FB2* schemes. The latter is identical to the Newmark scheme for all $\alpha_2^2 = 2\beta$ and $\alpha_1 = \gamma$ where β and γ are the Newmark integration parameters.

The generalised Newton-Raphson scheme has been shown to include the conventional Newton-Raphson scheme as a special case when its integration scalars are set to $\beta = 0$ for an *implicit* version or $\beta = 1$ for an *explicit* version, as discussed in **Chapter 5**.

To consolidate parametric studies of the integration schemes developed, stability analyses have been performed using the energy and spectral stability methods. Both methods have revealed that the *FB1* scheme is unconditionally stable for $\alpha_1 = \alpha_2 = 1$ and conditionally stable for $\Delta t \mathbf{M}^{-1} \mathbf{K} / 2 \leq \mathbf{I} / (\alpha_2^2 - 2\alpha_1\alpha_2 + \alpha_1)$ where, for the spectral analysis, the matrices are replaced by scalars. As a consequence of choosing $\alpha_1 \neq \frac{1}{2}$, the scheme possesses

numerical damping. The *FB2* scheme has stability characteristics identical to those of the Newmark scheme.

Finally, finite element numerical examples in form of plane stress and strain and axisymmetric models have served to evaluate the performance of the algorithms. In particular, they have demonstrated that the generalised Newton-Raphson scheme can considerably enhance convergence of the conventional Newton-Raphson scheme by converging quadratically even when a large single time-step (violating incremental strain laws) is used, without loss of accuracy, thus providing considerable advantages over the conventional scheme by reducing costs of computations associated with an incremental process.