TIME INTEGRATION SCHEMES FOR PIECEWISE LINEAR PLASTICITY

by

L.J. RENCONTRE

A thesis submitted for the degree of Doctor of Philosophy in the Faculty of Engineering.

Department of Mechanical Engineering

University of Cape Town

September 1991

ABSTRACT

The formulation of a generalized trapezoidal rule for the integration of the constitutive equations for a convex elastic-plastic solid is presented. This rule, which is based on an internal variable description, is consistent with a generalized trapezoidal rule for creep. It is shown that by suitable linear extrapolation, the standard backward difference algorithm can lead to this generalized trapezoidal rule or to a generalized midpoint rule. In either case, the generalized rules retain the symmetry of the consistent tangent modulus.

It is also shown that the generalized trapezoidal and midpoint rules are fully equivalent in the sense that they lead to the establishment of the same minimum principle for the increment. The generalized trapezoidal rule thus inherits the notion of B-stability and both rules offer the opportunity to exploit the second order rate of convergence for $\alpha = \frac{1}{2}$. However, in the generalized trapezoidal rule, the equilibrium and constitutive equations are fully satisfied at the end of the time increment. This may be more convenient than the generalized midpoint rule, in which equilibrium and plastic consistency are satisfied at the generalized midpoint.

A backward difference return algorithm for piecewise linear yield surfaces is then formulated, with attention restricted to an associated flow rule and isotropic material behavior. Both the Tresca and Mohr-Coulomb yield surfaces with perfectly plastic and linear hardening rules are considered in detail. The algorithm has the advantage of being fully linked to the governing principles and avoids the inherent problems associated with corners on the yield surface. It is fully consistent in that no heuristic assumptions are made.

The algorithm is extended to include the generalized trapezoidal rule in such a way that the general structure of the backward difference algorithm is maintained. This allows both for the computational advantages of the generalized trapezoidal rule to be utilized, and for a basis for comparison between this algorithm and existing backward difference algorithms to be established. Using this fully consistent algorithm, the return paths in stress space for the Tresca and Mohr-Coulomb yield surfaces with perfectly plastic and linear hardening rules are identified. These return paths thus provide a basis against which heuristically developed algorithms can be compared.

DECLARATION

I, Leslie John Rencontré, declare that this thesis is essentially my own work and that it has not been submitted for a degree at any other University.

Signed by candidate

10/9/1991

L.J. Rencontre.

Date.

ACKNOWLEDGMENTS

I would like to express my gratitude to the following people and organizations:

My supervisor, Professor Martin for his invaluable guidance and encouragement during my research.

The Foundation for Research Development (FRD) and the Centre for Research in Computational and Applied Mechanics (CERECAM) for financial support.

My colleagues at the Centre for Research in Computational and Applied Mechanics, especially Mike Snyman, Tony Pretorius and Greg Mitchell, for helpful discussions.

My mother, for her continuous support and interest in my studies.

TABLE OF CONTENTS

NOMENCLATURE

This is a list of the symbols used in the main text of this thesis.

Special Symbols

$\dot{}$	(accent) differential with respect to a time scale
$\hat{}$	(accent) incremental change in
$\tilde{}$	(accent) for the interval as defined in the text
-1	(superscript) inverse of a vector or matrix
T	(superscript) transpose of a vector or matrix
d	differentiation with respect to
∂	partial differentiation with respect to
Δ	increment in
∇	gradient of
O	of the order of

Lowercase symbols

c	cohesion (Mohr-Coulomb)
f	strain energy
k	scalar value (Tresca)
t	time
$\boldsymbol{e}$	deviatoric strain
$\boldsymbol{e}^p$	deviatoric plastic strain
$\boldsymbol{m}$	unit normal vector defined in the text
$\boldsymbol{n}$	unit normal vector defined in the text
$\boldsymbol{r}$	plastic flow direction
$\boldsymbol{s}$	deviatoric stress
$\boldsymbol{s}^E$	elastically predicted deviatoric stress
$\boldsymbol{u}$	nodal displacements

Uppercase symbols

D	dissipation function
E	Young's modulus
G	shear modulus
G_T	tangent shear modulus
H	scalar hardening constant
U	convex potential function
U_p	convex potential function
V	volume of the body
$\boldsymbol{B}$	discretized compatibility matrix
$\widehat{\boldsymbol{C}}$	matrix defined in the text
$\boldsymbol{D}$	elasticity matrix
$\widehat{\boldsymbol{D}}$	consistent tangent
$[\boldsymbol{K}]$	tangent modulus
$\boldsymbol{K}_c$	consistent tangent modulus
$\boldsymbol{C}$	matrix defined in the text
$\boldsymbol{E}$	matrix defined in the text
$\boldsymbol{H}$	matrix defined in the text
$\boldsymbol{K}$	matrix defined in the text
$\boldsymbol{L}$	matrix defined in the text
$\widehat{\boldsymbol{H}}$	reduced matrix defined in the text
$\widehat{\boldsymbol{L}}$	reduced matrix defined in the text
$\boldsymbol{N}$	matrix defined in the text
$\boldsymbol{P}$	external nodal force vector
$\boldsymbol{R}$	out of balance residual force vector
$\boldsymbol{T}$	transformation matrix

Greek Symbols

α	time integration parameter
β	angle defined in the text
γ	scalar hardening value defined in the text
θ	angle of internal friction (Mohr-Coulomb)
κ	hardening parameter
λ	scalar multiplier
ν	Poisson's ratio
ϕ	yield function/surface

χ_0	scalar defined in the text
$\boldsymbol{\epsilon}$	total strain
$\boldsymbol{\epsilon}^e$	elastic strain
$\boldsymbol{\epsilon}^p$	plastic strain
$\boldsymbol{\lambda}$	internal variable
$\Delta\boldsymbol{\Lambda}$	vector defined in the text
$\boldsymbol{\sigma}$	stress
$\hat{\boldsymbol{\sigma}}$	stress update algorithm
$\boldsymbol{\sigma}^E$	elastically predicted stress
$\boldsymbol{\sigma}_c$	contact stress
$\boldsymbol{\chi}$	conjugate force
$\boldsymbol{\chi}_0$	vector defined in the text

Superscripts

(i)	iteration number
o	trapezoidal rule indicator (forward step)
p	trapezoidal rule indicator (backward step)

Subscripts

n	time increment number
N	time increment number
e	element
p	number of

CHAPTER 1
INTRODUCTION

The formulation and solution of the incremental problem for a convex elastic-plastic solid is a fundamental problem in rate-independent computational plasticity. At some point within the finite element solution of the problem, it becomes necessary to integrate the constitutive equations governing the material behaviour. This calculation, which is carried out by an integration algorithm based on an appropriate time integration scheme, directly affects the overall accuracy of the analysis.

Classical approaches to the formulation of the integration algorithm have been essentially heuristic in that they are not fully linked to the governing principles[4]. This is especially so in the case of piecewise linear plasticity, where the algorithm needs to be formulated within the confines of Koiter's rule.

Due to the advantages associated with the backward difference integration scheme, there has been an increasing acceptance of this time integration scheme for piecewise linear plasticity[1,2,3,7,11]. Further, recent research has concentrated both on the improvement of time integration schemes (in terms of the accuracy and rate of convergence of the overall analysis) and on understanding the links between these integration schemes and the governing mechanical principles[6,8,10,12].

By exploiting the links between the classical and the consistent mathematical programming formulations provided by the internal variable approach[5,9], a consistent backward difference integration algorithm can be formulated. The algorithm is written in the form of a mathematical programming problem and is fully consistent in that no heuristic assumptions are made. In the application to piecewise linear plasticity, it has the advantages of being fully linked to the governing principles and avoiding the inherent problems associated with corners on the yield surface under the classical formulation. It thus provides a basis against which heuristically developed algorithms can be compared.

The algorithm can be improved by including a generalized trapezoidal rule for the integration of the constitutive equations. The formulation of this rule is also based on the internal variable approach and is fully linked to the governing mechanical principles. The generalized trapezoidal rule shares with the generalized midpoint rule of Simo *et al* [10,12] the notion of B-stability and the opportunity to exploit the second order rate of convergence for $\alpha = \frac{1}{2}$, while retaining the symmetry of the consistent tangent modulus. It may however be regarded as more convenient

than its generalized midpoint counterpart, in the sense that the equilibrium and constitutive equations are fully satisfied at the end of each interval rather than at the generalized midpoint.

The thesis is set out as follows: In chapter 2 the formulation and solution of the incremental elastic-plastic problem is cast in terms of the classical approach. A number of important computational aspects related to the choice of time integration schemes and their implementation in piecewise linear plasticity are considered.

Chapter 3 deals with the formulation and solution of the incremental problem using the internal variable approach. A time discretization in terms of a dual sequence of discrete instants is introduced together with a backward difference time integration scheme. The formulation provides the basic framework for a finite element model of the elastic-plastic problem and is used in the following chapters.

The generalized trapezoidal rule is introduced in chapter 4 and it is shown that by suitable linear extrapolation, the standard backward difference algorithm can lead to either this rule or to the generalized midpoint rule of Simo *et al* [10,12]. Further, it is shown that the generalized trapezoidal and generalized midpoint rules are fully equivalent in the sense that they lead to the establishment of the same minimum principle for the increment.

In chapter 5, an internal variable formulation of a consistent backward difference algorithm for the corrector step is considered. The algorithm is developed for piecewise linear plasticity and is restricted to isotropic materials with an associated flow rule. It is first specialized to the Tresca and Mohr-Coulomb yield surfaces with linear hardening and then extended to include the generalized trapezoidal rule. The extension is carried out in such a way that the general structure of the backward difference algorithm is maintained in that the same convex quadratic programming problem is established for both cases.

Finally, the quadratic programming problems for the Tresca and Mohr-Coulomb yield surfaces with perfectly plastic and linear hardening material behaviour are developed in chapter 6. These are then used to develop the said basis of comparison in the form of the classical return mapping algorithm by identifying the return paths in principal stress space associated with an elastically predicted stress.

REFERENCES

1. Crisfield, M.A., Plasticity computations using the Mohr-Coulomb yield criterion, *Engineering Computations*, 4, 300-308, 1987.

2. Crisfield, M.A., Consistent Schemes for Plasticity Computation with the

Newton-Raphson Method, Proc. 1st Int. Conf. *Computational Plasticity: Models, Software and Applications*, (D.R.J. Owen, E. Hinton & E. Onate *eds.*), Part 1, Pineridge Press, 133-159, 1987.

3. De Borst, R., Integration of Plasticity Equations for Singular Yield Functions, *Computers and Structures*, **26**, 823-829, 1987.

4. Maier, G. & Munro, J., Mathematical Programming Application to Engineering Plastic Analysis, *Applied Mechanics Reviews*, **35**, 1631-1643, 1982.

5. Martin, J.B., An Internal Variable Approach to the Formulation of Finite Element Problems in Plasticity, *Physical Nonlinearities in Structural Analysis*, eds. J. Holt & J. Lemaitre, Springer-Verlag, Berlin, 165-176, 1981.

6. Ortiz, M. & Martin, J.B., Symmetry-Preserving Return Mapping Algorithms and Incrementally Extremal Paths: a Unification of Concepts, *International Journal for Numerical Methods in Engineering*, **28**, 1839-1853, 1989.

7. Pankaj & Bićanić, N., On Multivector Stress Returns in Mohr-Coulomb Plasticity, *Computational Plasticity: Models, Software and Applications*, (D.R.J. Owen, E. Hinton & E. Onate *eds.*), Part 1, Pineridge Press, 421-436, 1989.

8. Reddy, B.D. & Martin, J.B., Algorithms for the Solution of Internal Variable Problems in Plasticity, *Computer Methods in Applied Mechanics and Engineering*, (to appear).

9. Rice, J.R., Inelastic Constitutive Relations for Solids: an Internal Variable Theory and its Application to Metal Plasticity, *Journal of Mechanics and Physics of Solids*, **19**, 433-455, 1971.

10. Simo, J.C. & Govindjee, S., Nonlinear B-Stability and Symmetry Preserving Return Mapping Algorithms for Plasticity and Viscoplasticity, *International Journal for Numerical Methods in Engineering*, **31**, 151-176, 1991.

11. Simo, J.C., Kennedy, J.G. & Govindjee, S., Non-Smooth Multisurface Plasticity and Viscoplasticity. Loading/Unloading Conditions and Numerical Algorithms, *International Journal for Numerical Methods in Engineering*, **26**, 2161-2185, 1988.

12. Simo, J.C. & Taylor, R.L., A Return Mapping Algorithm for Plane Stress Elastoplasticity, *International Journal for Numerical Methods in Engineering*, **22**, 649-670, 1986.

CHAPTER 2
FINITE ELEMENT SOLUTION FOR PLASTICITY

2.1 INTRODUCTION

In this chapter, the formulation and solution of the rate-independent elastic-plastic problem is cast in terms of the classical framework. Attention is restricted to an associated flow rule and an isotropic material with linear hardening. Various important computational aspects in rate-independent computational plasticity are introduced, and particular attention is paid to the choice of time integration schemes to integrate the constitutive rate equations.

Secondly, the elastic-plastic problem in piecewise linear plasticity is considered and solutions in terms of the classical framework discussed.

2.2 BASIC EQUATIONS FOR ELASTO-PLASTICITY

In small strain plasticity, under isothermal conditions, the total strain ϵ can be decomposed into elastic and plastic components

$$\epsilon = \epsilon^e + \epsilon^p \quad . \tag{2.1}$$

The stress σ is related to the elastic strain component by means of the symmetric elasticity matrix $\boldsymbol{D}$

$$\sigma = \boldsymbol{D}\epsilon^e \quad . \tag{2.2}$$

The plastic strain rate is given by the flow rule

$$\dot{\epsilon^p} = \lambda \boldsymbol{r}(\sigma, \kappa) \quad , \tag{2.3}$$

where λ is the scalar multiplier which denotes the magnitude of the plastic strain rate, $\boldsymbol{r}$ is the plastic flow direction and κ is a hardening parameter.

The onset of plastic flow is defined by a convex yield function $\phi(\boldsymbol{\sigma}, \kappa)$ which is homogeneous and of degree one in $\boldsymbol{\sigma}$ and delimits the elastic response region. This yield function is traditionally visualized as a yield surface in Cauchy stress space, its position in this space dependent on the hardening parameter κ.

For an associated flow rule, the plastic flow direction $\boldsymbol{r}$ is normal to the yield surface in stress space (normality condition)

$$\boldsymbol{r}(\boldsymbol{\sigma}, \kappa) = \frac{\partial \phi(\boldsymbol{\sigma}, \kappa)}{\partial \boldsymbol{\sigma}} \quad , \tag{2.4}$$

and eqn. (2.3) can be written as

$$\dot{\boldsymbol{\epsilon}}^p = \lambda \frac{\partial \phi(\boldsymbol{\sigma}, \kappa)}{\partial \boldsymbol{\sigma}} \quad . \tag{2.5}$$

Eqn.(2.5) is the basic plastic constitutive equation which relates the yield function and the plastic strain rates.

The scalar multiplier λ is evaluated using the loading-unloading conditions. These can be expressed in the Kuhn-Tucker form as the requirement that the following constraints be simultaneously satisfied at all times:

$$\phi(\boldsymbol{\sigma}, \kappa) \leq 0 \tag{2.6a}$$

$$\lambda \geq 0 \tag{2.6b}$$

$$\phi \lambda = 0 \tag{2.6c}$$

This in turn implies the plastic consistency condition, which must be satisfied during plastic flow and has the effect of confining the stress trajectory to the yield surface.

$$\dot{\phi} = \frac{\partial \phi}{\partial \boldsymbol{\sigma}} \dot{\boldsymbol{\sigma}} + \frac{\partial \phi}{\partial \kappa} \dot{\kappa} = 0 \tag{2.7}$$

In computational analysis it becomes necessary to integrate the constitutive equations given by eqns. (2.1)-(2.6). This is now described within the framework of the finite element method.

2.3 FINITE ELEMENT SOLUTION PROCEDURE

In the displacement finite element method, the elastic-plastic body is spatially discretized. External forces $\boldsymbol{P}(t)$, which are given functions of time, act on the unconstrained nodes. Displacements, represented by the vector $\boldsymbol{u}(t)$, are measured at the unconstrained nodes and both the strain field $\boldsymbol{\epsilon}(t)$ and the stress field $\boldsymbol{\sigma}(t)$ are defined at points where the constitutive equations are integrated (Gauss points).

The rate problem, described by eqns. (2.1)-(2.6), is recast in terms of an incremental problem by dividing the time domain into a sequence of discrete instants in time. We consider the time interval $(t_n, t_{n+1} = t_n + \Delta t)$ defined by two such instants. The initial conditions $\boldsymbol{\sigma}_n, \boldsymbol{\epsilon}_n, \boldsymbol{\epsilon}^p_n, \kappa_n$, defined at the instant t_n, as well as the strain increment for the interval $\Delta\boldsymbol{\epsilon}_{n+1}$, are assumed given. A solution $\boldsymbol{\sigma}_{n+1}, \boldsymbol{\epsilon}^p_{n+1}, \kappa_{n+1}$ is sought at the end of the interval at instant t_{n+1}.

In seeking the solution, the equilibrium equations are weakly satisfied at time t_{n+1}

$$\Sigma_e \int_{V_e} \boldsymbol{B}^T \boldsymbol{\sigma}_{n+1} dV = \boldsymbol{P}_{n+1} \quad . \tag{2.8}$$

Subscript [$e = 1, ..., M$ elements] denotes the spatial discretization of the volume V, while $\boldsymbol{B}$ is the matrix relating strains at the Gauss points to displacements at the nodes.

A stress update algorithm, which incorporates the integration of the incremental (rather than the rate) constitutive equations, is defined in the following form:

$$\boldsymbol{\sigma}_{n+1} = \hat{\boldsymbol{\sigma}}(\Delta\boldsymbol{\epsilon}_{n+1}, \boldsymbol{\sigma}_n, \boldsymbol{\epsilon}_n, \boldsymbol{\epsilon}^p_n, \kappa_n) \tag{2.9}$$

The stress update algorithm, in conjunction with the discretized compatibility equation, allows the equilibrium equations to be written in terms of the nodal displacement increments.

$$\Sigma_e \int_V \boldsymbol{B}^T \hat{\boldsymbol{\sigma}}(\boldsymbol{B}(\Delta\boldsymbol{u}_{n+1}), \boldsymbol{\sigma}_n, \boldsymbol{\epsilon}_n, \boldsymbol{\epsilon}^p_n, \kappa_n) dV = \boldsymbol{P}_{n+1} \tag{2.10}$$

The non-linear equations defined by eqn. (2.10) are solved by an iterative procedure based on the Newton Raphson method[15,29]. Each iteration within a particular time interval consists of what will be referred to as a predictor and a corrector step.

2.3.1 The Predictor Step

In the predictor step, new estimates of the nodal displacement increments are obtained by an appropriate linearisation of the incremental constitutive equations. If the linearization is *consistent* with the stress-update algorithm, the consistent tangent $\widehat{\boldsymbol{D}}$ of Simo and Taylor[25] is obtained.

$$\widehat{\boldsymbol{D}} = \frac{\partial \hat{\boldsymbol{\sigma}}(\Delta\boldsymbol{\epsilon}_{n+1}, \boldsymbol{\sigma}_n, \boldsymbol{\epsilon}_n, \boldsymbol{\epsilon}_n^p, \kappa_n)}{\partial \boldsymbol{\epsilon}_{n+1}} \tag{2.11}$$

For a trial solution after the (i)-th iteration, the predictor step, for the $(i+1)$-th iteration, then reduces to the solution of the following set of linear equations in $\Delta\hat{\boldsymbol{u}}$

$$[\boldsymbol{K}]\,\Delta\hat{\boldsymbol{u}} = \left[\Sigma_e \int_V \boldsymbol{B}^T \widehat{\boldsymbol{D}} \boldsymbol{B} dv\right] \Delta\hat{\boldsymbol{u}} = \boldsymbol{P}_{n+1} - \Sigma_e \int_V \boldsymbol{B}^T \boldsymbol{\sigma}_{n+1}^{(i)} dV \quad , \tag{2.12}$$

where the term in brackets is the tangent modulus and the term on the right hand side is the out-of-balance nodal force vector or residual. The new estimates of the nodal displacement increments is given by

$$\Delta\boldsymbol{u}_{n+1}^{(i+1)} = \Delta\boldsymbol{u}_{n+1}^{(i)} + \Delta\hat{\boldsymbol{u}} \quad . \tag{2.13}$$

2.3.2 The Corrector Step

In the corrector step, new estimates of the strain increments at the Gauss points are obtained directly from the nodal displacement increments using the discretized compatibility equation. The stress update algorithm of eqn. (2.9) is then used, to determine the stresses associated with the new estimates from the full (rather than linearised) incremental constitutive equations. The updated stresses lead to the residual (eqn. 2.12) which is the starting point of the next iteration.

2.4 THE STRESS UPDATE ALGORITHM

Two distinct approaches to the stress update algorithm exist, namely formulations based on incremental holonomic methods and those based on the concept of return mapping algorithms. The second of these approaches, although essentially heuristic[8], is treated in detail as it forms an integral part of the classical framework.

2.4.1 Incremental Holonomic Methods

The development of incremental holonomic methods is based on the concepts of incrementally extremal paths[5,18,20,21] and has the advantage that it is closely linked to the governing mechanical principles. A distinct correspondence exists between these methods and the internal variable formulation given in Chapter 3[10].

2.4.2 Return Mapping Algorithms

Return mapping algorithms have been the subject of a considerable amount of study since finite element incremental analysis was first attempted[11,22,28]. Traditionally, as illustrated in Figure 2.1, the stress update algorithm has been conceived in stress space as a two-step algorithm. The stress at the end of the n-th increment, $\boldsymbol{\sigma}_n$, is known, as is the total strain increment estimated for the $(i+1)$-th iteration of the $n+1$-th increment, $\Delta\boldsymbol{\epsilon}_{n+1}^{(i+1)}$. At each point where the stresses are calculated, it is first assumed that there is no plastic deformation in the increment, and the stress increment is computed as

$$\Delta\boldsymbol{\sigma}_{n+1}^{E\,(i+1)} = \boldsymbol{D}\Delta\boldsymbol{\epsilon}_{n+1}^{(i+1)} \quad . \tag{2.14}$$

If the estimated updated stress (known as the elastically predicted stress)

$$\boldsymbol{\sigma}_{n+1}^{E\,(i+1)} = \boldsymbol{\sigma}_n + \Delta\boldsymbol{\sigma}_{n+1}^{E\,(i+1)} \tag{2.15}$$

lies within the yield surface, the assumption is correct, the increment is indeed elastic, and the final stress point $\boldsymbol{\sigma}_{n+1}^{(i+1)}$ is given by the elastically predicted stress. If not, the final stress point $\boldsymbol{\sigma}_{n+1}^{(i+1)}$ on the yield surface and the plastic strain increment $\Delta\boldsymbol{\epsilon}^{p\,(i+1)}_{n+1}$ is computed, such that $\Delta\boldsymbol{\epsilon}^{p\,(i+1)}_{n+1}$ is normal to the yield surface according to some time integration scheme. Incremental plastic consistency (algebraic counterpart of eqn. 2.7) is satisfied by enforcing the yield criterion $\phi(\boldsymbol{\sigma},\kappa) = 0$ at the end of the time increment. Thus

$$\boldsymbol{\sigma}_{n+1}^{(i+1)} = \boldsymbol{\sigma}_n + \boldsymbol{D}(\Delta\boldsymbol{\epsilon}_{n+1}^{(i+1)} - \Delta\boldsymbol{\epsilon}^{p\,(i+1)}_{n+1}) \quad , \tag{2.16a}$$

$$\Delta\boldsymbol{\epsilon}_{n+1}^{p\,(i+1)} = \Delta\lambda\boldsymbol{r}(\boldsymbol{\sigma},\kappa) \quad , \tag{2.16b}$$

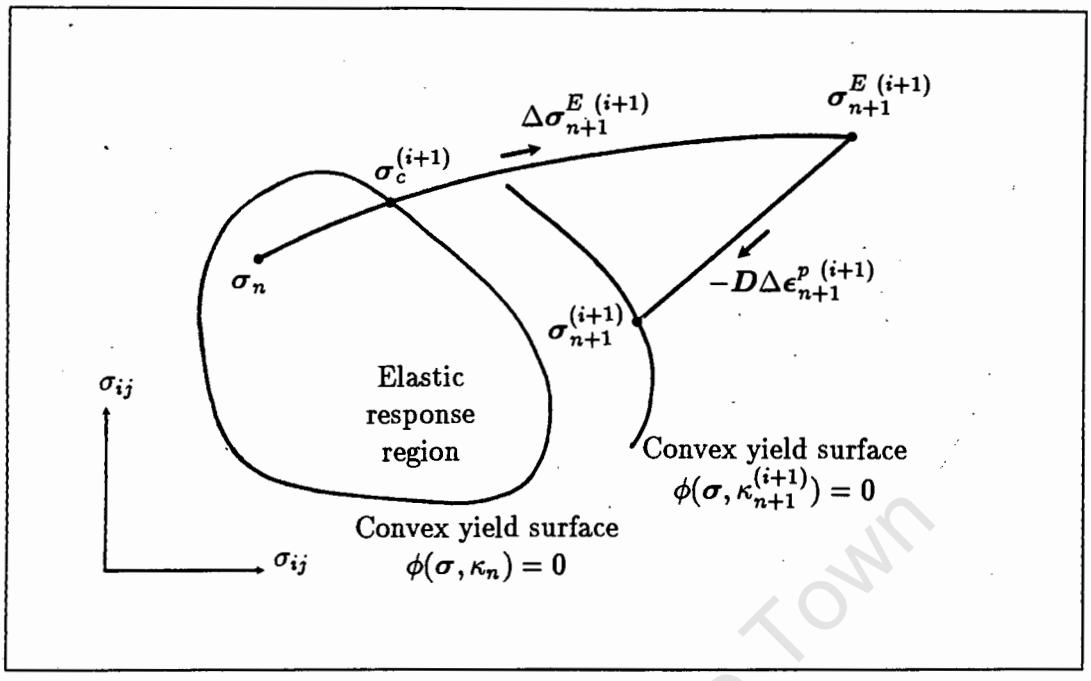

Figure 2.1: Geometric illustration of the return mapping algorithm

where $\boldsymbol{r}$ is evaluated via the normality condition (eqn. 2.4) at some point within the increment.

If the initial stress $\boldsymbol{\sigma}_n$ lies within the yield surface, as depicted in Figure 2.1, a contact stress state $\boldsymbol{\sigma}_c^{(i+1)}$ on the yield surface first needs to be computed[11,16]. However, for simplicity, we choose the initial stress state on the yield surface (i.e. $\boldsymbol{\sigma}_c^{(i+1)} = \boldsymbol{\sigma}_n$), which in no way restricts the generality of the discussion.

For hardening solids the yield surface may also change during the increment. Eqn. (2.16a) describes the movement of the elastically predicted stress point $\boldsymbol{\sigma}_{n+1}^{E\ (i+1)}$ back onto the appropriate yield surface, and is illustrated in Figure 2.1. Set in this framework, the stress update algorithm has been termed the return mapping algorithm and the numerical integration of the constitutive equations reduces to the solution of eqns. (2.16).

2.5 COMPARATIVE STUDIES

As the precision in which the constitutive equations are integrated within the stress update algorithm impacts directly on the accuracy of the overall analysis, comparisons of various algorithms have been made[7,14]. Three criteria for comparison have been established, namely the accuracy of the algorithm, the stability of the algo-

rithm and the symmetry of the tangent moduli. Each of these criteria is considered briefly.

2.5.1 Accuracy

A numerically integrated solution $\boldsymbol{\sigma}_{n+1}, \boldsymbol{\epsilon}_{n+1}, \boldsymbol{\epsilon}^p_{n+1}, \kappa_{n+1}$ is accurate if it agrees with the exact solution $\boldsymbol{\sigma}(t_{n+1}), \boldsymbol{\epsilon}(t_{n+1}), \boldsymbol{\epsilon}^p(t_{n+1}), \kappa(t_{n+1})$ obtained using the full rate constitutive equations (eqns. 2.1-2.6).

As the numerical integration of the constitutive equations is carried out for a given strain increment $\Delta\boldsymbol{\epsilon}_{n+1}$, the total updated strain

$$\boldsymbol{\epsilon}_{n+1} = \boldsymbol{\epsilon}_n + \Delta\boldsymbol{\epsilon}_{n+1} \tag{2.17}$$

is a function of the time increment Δt. The numerical solution is implicitly defined via the integration of the constitutive equations and thus $\boldsymbol{\sigma}_{n+1}, \boldsymbol{\epsilon}^p_{n+1}, \kappa_{n+1}$ are also functions of Δt. Further, by choosing ϕ to be sufficiently smooth, $\boldsymbol{\sigma}_{n+1}, \boldsymbol{\epsilon}^p_{n+1}, \kappa_{n+1}$ are differentiable functions of Δt (the implicit function theorem) and may be expanded using a Taylor series.

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_n + \Delta t \left.\frac{\partial(\boldsymbol{\sigma}_{n+1})}{\partial \Delta t}\right|_{\Delta t=0} + \frac{\Delta t^2}{2} \left.\frac{\partial^2(\boldsymbol{\sigma}_{n+1})}{\partial \Delta t^2}\right|_{\Delta t=0} + O(\Delta t^3) \tag{2.18a}$$

$$\boldsymbol{\epsilon}^p_{n+1} = \boldsymbol{\epsilon}^p_n + \Delta t \left.\frac{\partial(\boldsymbol{\epsilon}^p_{n+1})}{\partial \Delta t}\right|_{\Delta t=0} + \frac{\Delta t^2}{2} \left.\frac{\partial^2(\boldsymbol{\epsilon}^p_{n+1})}{\partial \Delta t^2}\right|_{\Delta t=0} + O(\Delta t^3) \tag{2.18b}$$

$$\kappa_{n+1} = \kappa_n + \Delta t \left.\frac{\partial(\kappa_{n+1})}{\partial \Delta t}\right|_{\Delta t=0} + \frac{\Delta t^2}{2} \left.\frac{\partial^2(\kappa_{n+1})}{\partial \Delta t^2}\right|_{\Delta t=0} + O(\Delta t^3) \tag{2.18c}$$

The stress update algorithm is first order accurate (or consistent with the constitutive equations) if the above numerically integrated values agree with their exact values to within second-order terms in Δt. Similarly, the algorithm is second order accurate if the numerically integrated values agree with their exact values to within third-order terms in Δt. Any algorithm that is second-order accurate incorporates first-order accuracy.

2.5.2 Stability

First order accuracy and stability of the stress update algorithm are necessary and sufficient conditions for convergence as the time increment Δt tends to zero[14]. Thus an acceptable algorithm needs to satisfy these two criteria.

Ortiz and Popov[14] introduced a general methodology for numerical stability of the elasto-plastic problem in which the yield surface is smooth. Within this methodology, an algorithm is stable in the energy norm if infinitesimal perturbations in the initial conditions are attenuated.

More recently, Simo and Govindjee[23] introduced the notion of B-stability[31,32,33,34] in which the algorithm is contractive relative to the natural norm defined by the complementary Helmholtz free energy function. The notion of B-stability is the appropriate notion of non-linear stability for rate independent plasticity. It is independent of the smoothness of the yield surface but relies on the convexity of ϕ (which is assumed) and the normality condition (eqn.2.4).

2.5.3 Symmetry of the Tangent Modulus

In order to preserve the quadratic rate of asymptotic convergence of the Newton Raphson method, the linearization of the constitutive equations in the predictor step must be consistent with the stress update algorithm, leading to the consistent tangent of Simo and Taylor[25]. These consistent tangents may however not preserve constitutive symmetries: i.e. the tangent modulus (eqn.2.12) may not be symmetric. The symmetry of the tangent modulus is necessary for numerous regularity properties and bounding theorems of plasticity[9] and thus it is important that it is preserved by the numerical procedure. As the consistent tangent is dependent on the stress update algorithm (eqn. 2.9), a desirable feature of such an algorithm is that it ensures symmetry of the consistent tangents.

2.6 TIME INTEGRATION SCHEMES

The time integration scheme forms an integral part of the return mapping algorithm of eqn. (2.16). Ortiz and Popov[14] have generalized the concept of return mapping algorithms and postulated two generalized time integration schemes which categorize a wide range of these algorithms. These generalized schemes are consistent with the return mapping algorithm described in section 2.4.2, in that incremental plastic consistency is enforced at t_{n+1}. The algorithms associated with these generalized

schemes are first order accurate and in general do not ensure symmetry of the tangent modulus except for the specific case of the backward difference scheme[13].

Simo and Taylor[26] and Simo and Govindjee[23] have proposed an alternate generalized midpoint scheme in which incremental plastic consistency is enforced at a generalized midpoint. This scheme, which also leads to first order accurate return mapping algorithms, has the advantage that symmetry of the tangent modulus is ensured. A further advantage of this scheme is that it leads to a second order accurate algorithm if the true midpoint is chosen.

2.6.1 The Generalized Trapezoidal rule of Ortiz and Popov

For the generalized trapezoidal rule eqn. (2.16b) can be written as

$$\Delta\epsilon^{p\,(i+1)}_{n+1} = \Delta\lambda \boldsymbol{r}^{(i+1)}_{n+\alpha} \quad , \tag{2.19a}$$

$$\boldsymbol{r}^{(i+1)}_{n+\alpha} = (1-\alpha)\boldsymbol{r}_n + \alpha \boldsymbol{r}^{(i+1)}_{n+1} \qquad 0 \leq \alpha \leq 1 \quad , \tag{2.19b}$$

where

$$\boldsymbol{r}_n = \boldsymbol{r}(\boldsymbol{\sigma}_n, \kappa_n) \quad , \tag{2.19c}$$

$$\boldsymbol{r}^{(i+1)}_{n+1} = \boldsymbol{r}(\boldsymbol{\sigma}^{(i+1)}_{n+1}, \kappa^{(i+1)}_{n+1}) \quad . \tag{2.19d}$$

The return in stress space, associated with this time integration scheme, is a single step return in direction $\boldsymbol{r}^{(i+1)}_{n+\alpha}$ and is illustrated in Figure 2.2. Alternately, the return can be visualized as consisting of two substeps, the first in the initial plastic flow direction $\boldsymbol{r}_n$ (normal to the yield surface at $\boldsymbol{\sigma}_n, \kappa_n$) and the second in the final plastic flow direction $\boldsymbol{r}^{(i+1)}_{n+1}$ (normal to the yield surface at $\boldsymbol{\sigma}^{(i+1)}_{n+1}, \kappa^{(i+1)}_{n+1}$). The parameter α gives the relative proportions of these substeps.

For $\alpha = 0$ the return mapping algorithm is explicit and for $\alpha > 0$ it is implicit. For $\alpha = \frac{1}{2}$ and the von Mises yield function, the algorithm coincides with the mean normal procedure of Rice and Tracy[22]; and for $\alpha = 1$, it coincides with the backward difference or closest point algorithm (a generalization of the radial return algorithm of Wilkins[28] for the von Mises yield function).

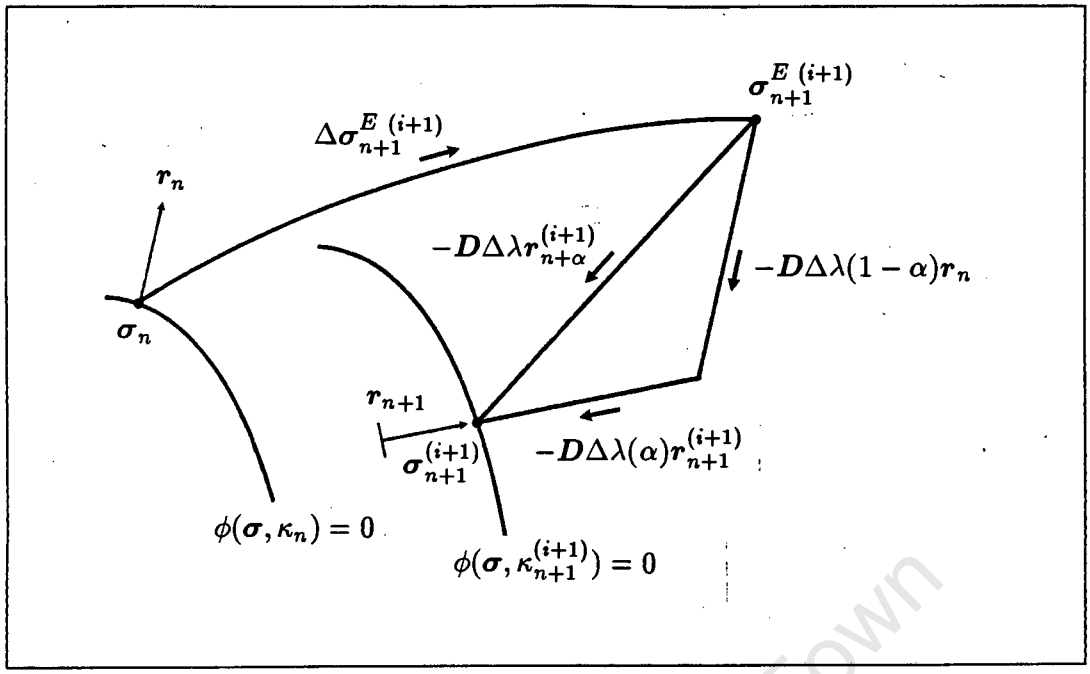

Figure 2.2: The generalized trapezoidal rule of Ortiz and Popov

For a yield surface of constant curvature (eg von Mises) and for $\alpha \geq \frac{1}{2}$, the algorithm is unconditionally stable in the energy norm. The algorithm is also second-order accurate for $\alpha = \frac{1}{2}$.

2.6.2 The Generalized Midpoint rule of Ortiz and Popov

For the generalized midpoint rule eqn. (2.16b) can be written as

$$\Delta\epsilon_{n+1}^{p\,(i+1)} = \Delta\lambda r_{n+\alpha}^{(i+1)} \quad , \tag{2.20a}$$

$$r_{n+\alpha}^{(i+1)} = r((1-\alpha)\sigma_n + \alpha\sigma_{n+1}^{(i+1)}, (1-\alpha)\kappa_n + \alpha\kappa_{n+1}^{(i+1)}) \qquad 0 \leq \alpha \leq 1. \tag{2.20b}$$

The return in stress space, associated with this time integration scheme, is a single step return in the plastic flow direction $r_{n+\alpha}^{(i+1)}$ (normal to the yield surface at $\sigma_{n+\alpha}^{(i+1)}, \kappa_{n+\alpha}^{(i+1)}$) and is illustrated in Figure 2.3.

For $\alpha = 1$, this rule coincides with the generalized trapezoidal rule. The two rules also coincide for all α for the von Mises yield function with elastic perfectly plastic and linear kinematic hardening.

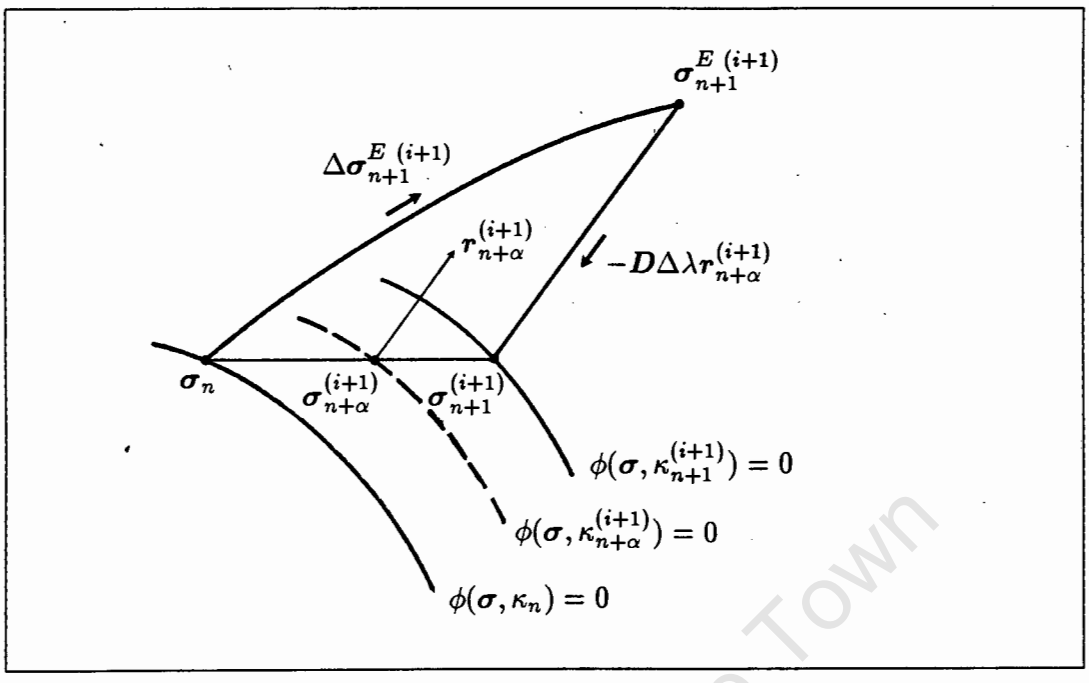

Figure 2.3: The generalized midpoint rule of Ortiz and Popov

Unlike the generalized trapezoidal rule, the generalized midpoint rule is unconditionally stable in the energy norm for $\alpha \geq \frac{1}{2}$ irrespective of the yield surface. The generalized midpoint rule is also second-order accurate for $\alpha = \frac{1}{2}$.

2.6.3 The Generalized Midpoint rule of Simo *et al*

The essential difference of this rule is that incremental plastic consistency is enforced at a generalized midpoint $t_{n+\alpha}$. This implies that the plastic strain increment $\Delta\epsilon_{n+1}^{p\,(i+1)}$ is computed such that the stress point $\sigma_{n+\alpha}^{(i+1)}$, defined at the generalized midpoint, lies on the yield surface. Eqn. (2.16b) thus become

$$\Delta\epsilon_{n+1}^{p\,(i+1)} = \frac{1}{\alpha}\Delta\lambda r_{n+\alpha}^{(i+1)} \quad , \tag{2.21a}$$

$$r_{n+\alpha}^{(i+1)} = r((1-\alpha)\sigma_n + \alpha\sigma_{n+1}^{(i+1)}, (1-\alpha)\kappa_n + \alpha\kappa_{n+1}^{(i+1)}) \qquad 0 \leq \alpha \leq 1. \tag{2.21b}$$

Eqn. (2.21b) coincides with eqn. (2.20b). The geometric interpretation of this return mapping algorithm is given in Figure 2.4, in which the enforcement of incremental plastic consistency at $t_{n+\alpha}$ is clearly illustrated.

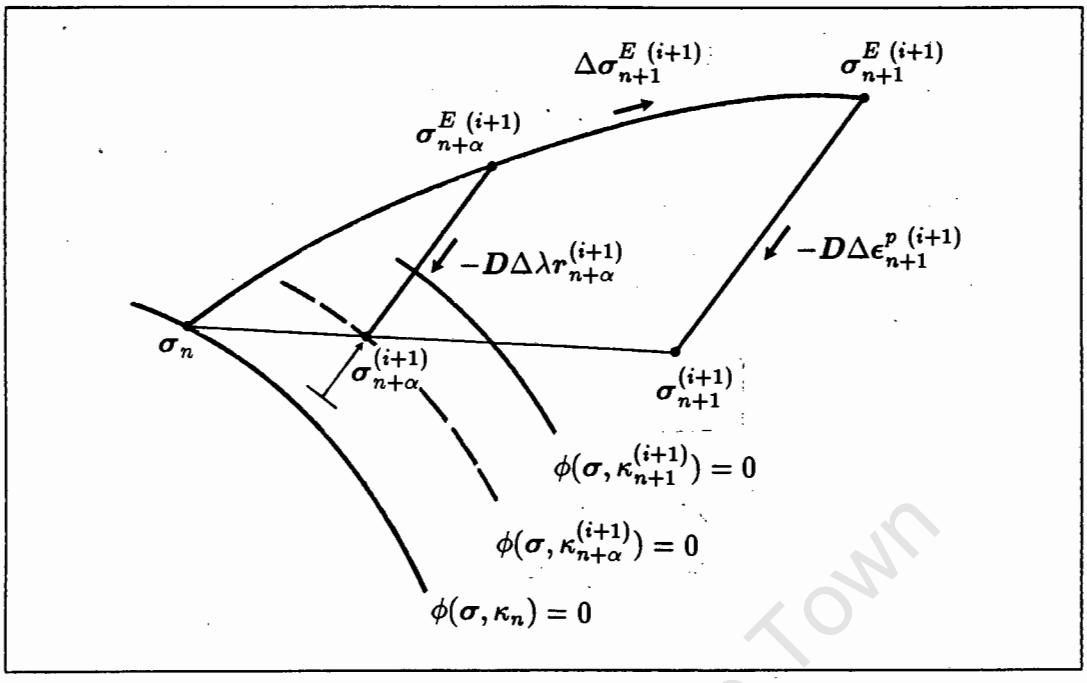

Figure 2.4: The generalized midpoint rule of Simo *et al*

This rule preserves the symmetry of the tangent modulus for $0 \leq \alpha \leq 1$ and is second order accurate for $\alpha = \frac{1}{2}$. Simo and Govindjee[23] have also shown that the algorithm is B-stable for $\alpha \geq \frac{1}{2}$.

2.6.4 The Backward Difference Scheme

The backward difference scheme is a special case of the generalized trapezoidal and both generalized midpoint rules for $\alpha = 1$, in which all three rules coincide. This scheme implies that the plastic strain increment has the direction of the normal to the yield surface at the updated stress point (ie $\Delta\boldsymbol{\epsilon}^{p(i+1)}_n$ has direction $\boldsymbol{r}^{(i+1)}_{n+1} = \boldsymbol{r}(\boldsymbol{\sigma}^{(i+1)}_{n+1}, \kappa^{(i+1)}_{n+1})$ in the context of the integration rules quoted above). This ensures that consistency is enforced at time $t_{n+\alpha} = t_{n+1}$. This is the only time integration scheme in which the contact stress $\boldsymbol{\sigma}^{(i+1)}_c$ need never be calculated for an initial stress $\boldsymbol{\sigma}_n$ not on the yield surface.

The backward difference algorithm, although only first order accurate, is the only member of both generalized rules of Ortiz and Popov that preserves the symmetry of the tangent modulus[13,26]. It is also the only member of the trapezoidal rule that is unconditionally stable in the energy norm for a yield surface with corners. Further, the algorithm, as a member of the generalized midpoint rule of Simo *et al* [23,26], is B-stable.

Ortiz and Martin[13] put forward arguments to show why the backward difference algorithm meets sufficient conditions for symmetry, while the generalized trapezoidal rule of Ortiz and Popov[14] does not. Further, they show that the backward difference algorithm can be brought into correspondence with algorithms formulated using holonomic methods.

2.7 PIECEWISE LINEAR PLASTICITY

Piecewise linear yield functions of the Tresca and Mohr-Coulomb type are widely used in metals and soils plasticity. These yield functions are distinguishable by the fact that they describe piecewise linear (singular) yield surfaces. We shall briefly consider the implications and methods of solution associated with piecewise linear plasticity.

2.7.1 Difficulties in Piecewise Linear Plasticity

When the incremental problem, formulated with a piecewise linear yield function, is solved using the finite element method, numerical singularities may occur in the corrector step. This is due to the fact that the normal to the yield surface at a singularity (a corner or apex) lies within the fan (or singular region) defined by adjacent normals at the singularity. The plastic flow direction $\boldsymbol{r}$, a necessary piece of information for the numerical integration of the constitutive equations, is thus not uniquely defined at this point.

The ambiguity of the plastic flow direction is removed by Koiter's generalization, in which piecewise linear yield functions are considered to be made up of a number of continuously differentiable yield functions[6]. The elastic response region is then given by

$$\phi_p(\boldsymbol{\sigma}, \kappa) < 0 \qquad p = 1, 2, \ldots, N \tag{2.22}$$

where N is the number of continuously differentiable yield functions.

The flow rule (eqn. 2.5) becomes

$$\dot{\boldsymbol{\epsilon}}^p = \sum_{p=1}^{N} \lambda \frac{\partial \phi_p(\boldsymbol{\sigma}, \kappa)}{\partial \boldsymbol{\sigma}} \tag{2.23}$$

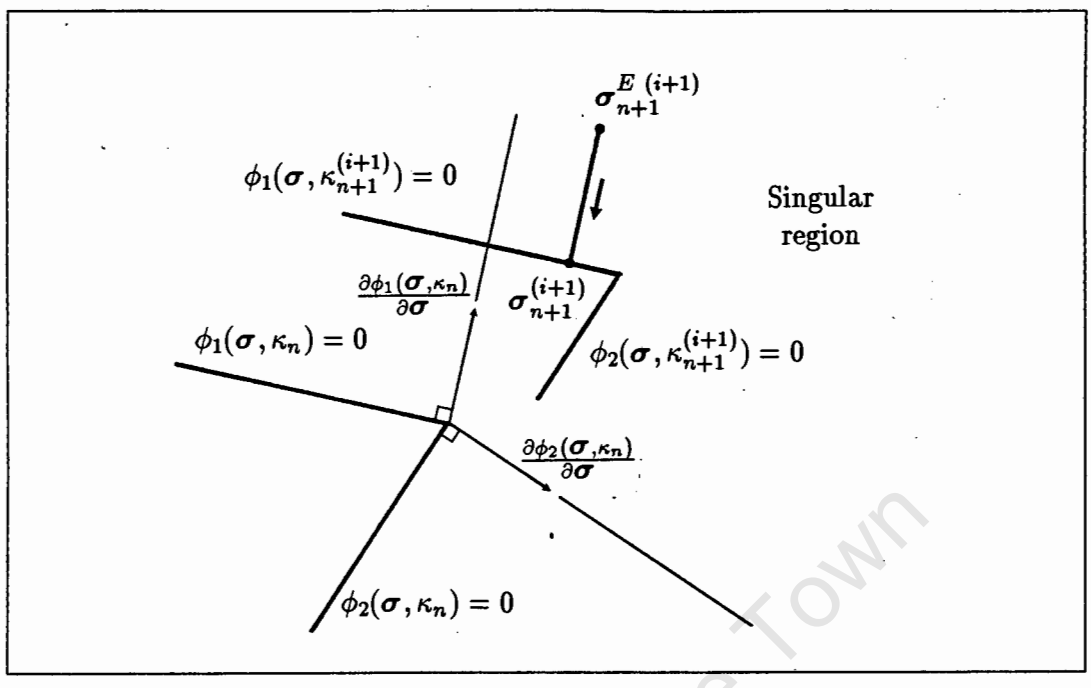

Figure 2.5: Return at a singular region

and the loading-unloading criterion (eqn. 2.6) apply for any ϕ_p.

This generalization allows the plastic flow direction to be constructed from the adjacent normals at the singularity. It also allows for multiple yield surfaces to become *active* at the same time (i.e. $\phi_p > 0, \quad p \geq 2$) and leads to the concept of multivector return paths in the framework of the return mapping algorithm[2,3,4,11,17,19,24].

Singularity indicators have been developed to determine whether the elastically predicted stress point is in a singular region and a multivector return is necessary to return the stress to the corner [4,17,24]. Problems can be encountered during hardening when the elastically predicted stress lies in a singular region but does not return to the corner as illustrated in Figure 2.5.

The problem is further complicated by the fact that piecewise linear yield functions of the Tresca and Mohr-Coulomb type are expressed most naturally in terms of the principal stresses. Nayak and Zienkiewicz[12] introduced a form of stress invariants which allows for a compact description of the yield functions, but these invariants still need to be calculated at each iteration.

A mention of other approaches, not utilizing Koiter's generalization is appropriate at this point. These include; using the normal of the smooth yield surface that passes through the singularity[15] (an approach which results in a discontinuity in the normal at this point), rounding the yield surface locally in the singular regions[27]

and rounding the entire yield surface[30].

2.7.2 Classification of Solution Algorithms

Once again, it is convenient to classify solution algorithms employing Koiter's generalization in terms of the integration schemes used. Due to the advantages associated with the backward difference scheme there has been an increasing acceptance of this scheme for the return mapping algorithm[2,3,4,17,24]. The essential differences between these algorithms is located in the singularity indicators used.

Marques[11], however, uses an explicit scheme with subincrementation which allows the singular regions to be identified. De Borst[4] has shown that this formulation does not satisfy incremental plastic consistency for all active yield surfaces at a singularity.

Crisfield[2,3], De Borst[4] and Pankaj and Bićanić[17] all apply a backward difference scheme and enforce incremental plastic consistency for all active yield surfaces at a singularity. Each uses a different singularity indicator and considers the Mohr-Coulomb yield function with elastic perfectly plastic behaviour. All advocate an *a posteriori* check for hardening.

Simo *et al.*[24] and Pramono and Willam[19] consider hardening and employ an iterative return mapping based on the backward difference scheme. Simo *et al*[24] propose a procedure that assumes all yield surfaces violated by the elastically predicted stress (ie $\phi_p(\boldsymbol{\sigma}_{n+1}^{E\,(i+1)}, \kappa_n) \geq 0$) are active. In the iterative return, active yield surfaces are deactivated one by one when the corresponding plastic multiplier becomes negative. Pramono and Willam have proposed an opposite approach which encompasses softening behaviour as well. Here, the yield surfaces are activated one by one when they are violated during the iterative procedure.

2.8 CONCLUSIONS

The incremental elastic-plastic problem has been formulated in the classical framework of the finite element method. Various time integration schemes, the relative merits of each scheme and the incremental problem and solution in piecewise linear plasticity have also been introduced.

Next we wish to explore the formulation and solution of the same problem in terms of an internal variable formulation.

REFERENCES

1. Carter, P. & Martin, J.B., Work Bounding Functions for Plastic Materials, *Journal of Applied Mechanics*, **43**, 434-438, 1976.

2. Crisfield, M.A., Plasticity computations using the Mohr-Coulomb yield criterion, *Engineering Computations*, **4**, 300-308, 1987.

3. Crisfield, M.A., Consistent Schemes for Plasticity Computation with the Newton-Raphson Method, Proc. 1st Int. Conf. *Computational Plasticity: Models, Software and Applications*, (D.R.J. Owen, E. Hinton & E. Onate *eds.*), Part 1, Pineridge Press, 133-159, 1987.

4. De Borst, R., Integration of Plasticity Equations for Singular Yield Functions, *Computers and Structures*, **26**, 823-829, 1987.

5. Griffin, T.B., Reddy, B.D. & Martin, J.B., A Numerical Study of Holonomic Approximations to Problems in Plasticity, *International Journal for Numerical Methods in Engineering*, **26**, 1449-1466, 1988.

6. Koiter,W.T., Stress-Strain Relations, Uniqueness and Variational Theorems for Elastic-Plastic Materials with a Singular Yield Surface, *Quarterly of Applied Mathematics*, **11**, 350-354, 1953.

7. Krieg, R.D. & Krieg, D.B, Accuracies of Numerical Solution Methods for the Elastic-Perfectly Plastic Model, *Journal of Pressure Vessel Technology, ASME*, **99**, 510-515, 1977.

8. Maier, G. & Munro, J., Mathematical Programming Application to Engineering Plastic Analysis, *Applied Mechanics Reviews*, **35**, 1631-1643, 1982.

9. Martin, J.B., *Plasticity: Fundamentals and General Results*, MIT Press, Cambridge, 1975.

10. Martin, J.B. & Reddy, B.D., Variational Principles and Solution Algorithms for Internal Variable Formulations of Problems in Plasticity, *Omaggio a Giulio Ceradini*, (eds. U. Andreaus et al), Roma, Università di Roma "La Sapienza", 465-477, 1988.

11. Marques, J.M.M.C., Stress Computations in Elastoplasticity, *Engineering Computations*, **1**, 42-51, 1984.

12. Nayak, G.C. & Zienkiewicz, O.C.,, Convenient Form Of Stress Invariants For Plasticity, *Journal of the Structural Division. Proceedings of ASCE*, **98**, 949-953, 1972.

13. Ortiz, M. & Martin, J.B., Symmetry-Preserving Return Mapping Algorithms and Incrementally Extremal Paths: a Unification of Concepts, *International Journal for Numerical Methods in Engineering*, **28**, 1839-1853, 1989.

14. 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.

15. Owen, D.R.J. & Hinton, E., *Finite Elements in Plasticity - Theory and Practice*, Pineridge Press, Swansea, 1980.

16. Pankaj & Bićanić, N., Exact Evaluation of Contact Stress State in Computational Elasto-plasticity, *Engineering Computations*, **6**, 67-73, 1989.

17. Pankaj & Bićanić, N., On Multivector Stress Returns in Mohr-Coulomb Plasticity, *Computational Plasticity: Models, Software and Applications*, (D.R.J. Owen, E. Hinton & E. Onate *eds.*), Part 1, Pineridge Press, 421-436, 1989.

18. Ponter, A.R.S. & Martin, J.B., Some Extremal Properties and Energy Theorems for Inelastic Materials and their Relationship to the Deformation Theory of Plasticity, *Journal of the Mechanics and Physics of Solids*, **20**, 281-300, 1972.

19. Pramono, E. & Willam, K., Implicit integration of composite yield surfaces with corners, *Engineering Computations*, **6**, 186-197, 1989.

20. Reddy, B.D. & Griffin, T.B., Variational Principles and Convergence of Finite Element Approximations of a Holonomic Elastic-Plastic Problem, *Numerische Mathematik*, **52**, 101-117, 1988.

21. Reddy, B.D., Martin, J.B. & Griffin, T.B., Extremal Paths and Holonomic Constitutive Laws in Elastoplasticity, *Quarterly of Applied Mathematics*, **45**, 487-502, 1987.

22. Rice, J.R. & Tracey, D.M., Computational Fracture Mechanics, *Numerical and Computer Methods in Structural Mechanics*, eds. S.J. Fenves *et al*, Academic Press , New york, 585-623, 1973.

23. Simo, J.C. & Govindjee, S., Nonlinear B-Stability and Symmetry Preserving Return Mapping Algorithms for Plasticity and Viscoplasticity, *International Journal for Numerical Methods in Engineering*, **31**, 151-176, 1991.

24. Simo, J.C., Kennedy, J.G. & Govindjee, S., Non-Smooth Multisurface Plasticity and Viscoplasticity. Loading/Unloading Conditions and Numerical Algorithms, *International Journal for Numerical Methods in Engineering*, **26**, 2161-2185, 1988.

25. Simo, J.C. & Taylor, R.L., Consistent Tangent Operators for Rate-Independent Elastoplasticity, *Computer Methods in Applied Mechanics and Engineering*, **48**, 101-118, 1985.

26. Simo, J.C. & Taylor, R.L., A Return Mapping Algorithm for Plane Stress Elastoplasticity, *International Journal for Numerical Methods in Engineering*, **22**, 649-670, 1986.

27. Sloan, S.W & Booker, J.R., Removal of Singularities in Tresca and Mohr-Coulomb Yield Functions, *Communications in Applied Numerical Methods*, **2**, 173-179, 1986.

28. Wilkins, M.L., Calculation of Elastic-Plastic Flow, *Methods of Computational Physics*, eds. B. Alder *et al*, **3**, Academic Press, New York, 1964.

29. Zienkiewicz, O.C., *The Finite Element Method*, McGraw-Hill, London, 1977.

30. Zienkiewicz, O.C. & Pande, G.N., Some Useful Forms of Isotropic Yield Surfaces for Soil and Rock Mechanics, *Finite Elements in Geomechanics*, ed. G. Gudehus, J. Wiley & Sons., 179-190, 1977.

31. Burrage, K. & Butcher, J.C., Stability Criteria for Implicit Runge-Kutta Methods, *Saim Journal of Numerical Mathematics*, **16**, 46-57, 1979.

32. Burrage, K. & Butcher, J.C., Nonlinear Stability of a General Class of Differential Equation Methods, *BIT*, **20**, 185-203, 1980.

33. Butcher, J.C., A Stability Property of Implicit Runge-Kutta Methods, *BIT*, **15**, 358-361, 1975.

34. Dahlquist, G., A Special Stability Problem for Linear Multistep Methods, *BIT*, **3**, 27-43, 1963.

CHAPTER 3
AN INTERNAL VARIABLE FORMULATION

3.1 INTRODUCTION

In this chapter, an internal variable formulation for the classical time-independent elastic-plastic problem is provided. The formulation provides the basic framework for a finite element model of this problem and allows for clear comparison of existing numerical techniques. This is a specific form of a general thermodynamically based formulation of the form initially given by Rice[9] and Martin[4].

In this work, attention is restricted to associated plasticity and simple linear hardening rules. Further, a backward difference scheme is chosen, which assures that a minimum principle, or mathematical programming problem can be written for this formulation[3,6].

3.2 BASIC EQUATIONS FOR THE INTERNAL VARIABLE FORMULATION

Consider a finite element model of an elastic-plastic body under isothermal conditions. The unconstrained nodal displacements are represented by the vector $\boldsymbol{u}(t)$, where t represents time. External forces $\boldsymbol{P}(t)$ act on the unconstrained nodes, and are given functions of time. Internal variables are defined at Gauss points, and are represented by the global vector $\boldsymbol{\lambda}(t)$.

The strain energy f of the body is assumed to be a homogeneous quadratic function of $\boldsymbol{u}(t)$ and $\boldsymbol{\lambda}(t)$ and is given by

$$f = \tfrac{1}{2}\boldsymbol{u}(t)^T\boldsymbol{K}\boldsymbol{u}(t) + \tfrac{1}{2}\boldsymbol{u}(t)^T(\boldsymbol{L}+\boldsymbol{L}^T)\boldsymbol{\lambda}(t) + \tfrac{1}{2}\boldsymbol{\lambda}(t)^T\boldsymbol{H}\boldsymbol{\lambda}(t) \quad . \tag{3.1}$$

The equilibrium equations can then be written in the following form:

$$\frac{\partial f}{\partial \boldsymbol{u}(t)} = \boldsymbol{K}\boldsymbol{u}(t) + \boldsymbol{L}\boldsymbol{\lambda}(t) = \boldsymbol{P}(t) \quad , \tag{3.2a}$$

$$\frac{\partial f}{\partial \boldsymbol{\lambda}(t)} = \boldsymbol{L}^T \boldsymbol{u}(t) + \boldsymbol{H}\boldsymbol{\lambda}(t) = -\boldsymbol{\chi}(t) \quad . \tag{3.2b}$$

The internal forces $\boldsymbol{\chi}(t)$ are conjugate to the internal variables $\boldsymbol{\lambda}(t)$. The matrices $\boldsymbol{K}, \boldsymbol{L}, \boldsymbol{H}$ are fixed, and may be formulated by standard methods[4]. The stiffness matrix $\boldsymbol{K}$ is assumed to be positive definite, and $\boldsymbol{H}$ is assumed to be positive semi-definite.

A convex global dissipation function $D(\dot{\boldsymbol{\lambda}})$ which is positive, homogeneous and of degree one in the components of $\dot{\boldsymbol{\lambda}}$ is introduced. The basic plastic constitutive relation then relates the internal forces $\boldsymbol{\chi}$ and the internal variable rates $\dot{\boldsymbol{\lambda}}$:

$$\boldsymbol{\chi} = \frac{\partial D}{\partial \dot{\boldsymbol{\lambda}}} \tag{3.3}$$

The dissipation function D is assumed to be zero if and only if $\dot{\boldsymbol{\lambda}} = 0$ and to be positive for $\dot{\boldsymbol{\lambda}} \neq 0$. The dissipation function (illustrated Figure 3.1) is thus a generalized cone in a $D - \dot{\boldsymbol{\lambda}}$ space, formed by radial generators. The derivatives of D will certainly be discontinuous at the origin, and may be discontinuous along radial lines in the $\dot{\boldsymbol{\lambda}}$ space. At such points of discontinuity, $\partial D/\partial \dot{\boldsymbol{\lambda}}$ is taken to represent any value within the fan defined by adjacent values.

Substitution of eqn. (3.3) into eqns. (3.2) leads to

$$\boldsymbol{K}\boldsymbol{u}(t) + \boldsymbol{L}\boldsymbol{\lambda}(t) = \boldsymbol{P}(t) \quad , \tag{3.4a}$$

$$\boldsymbol{L}^T \boldsymbol{u}(t) + \boldsymbol{H}\boldsymbol{\lambda}(t) + \left.\frac{\partial D}{\partial \dot{\boldsymbol{\lambda}}}\right|_{\dot{\boldsymbol{\lambda}}(t)} = 0 \quad . \tag{3.4b}$$

The dissipation function D will in fact be the sum of contributions from individual Gauss points. Eqn. (3.4b) can be subdivided into a sequence of independent equations for each Gauss point; the equations are independent in the sense that the components of $\boldsymbol{\lambda}$ and $\dot{\boldsymbol{\lambda}}$ which appear in any one Gauss point equation are only those associated with that Gauss point.

Eqns. (3.4) provide the basic framework of the equations for a finite element model of an elastic-plastic body for a wide class of linear kinematic hardening materials, including perfect plasticity. Suitable generalization can be incorporated to extend the model to cover isotropic hardening[5], but the model described by eqns. (3.4) is sufficiently general for our present purposes. The generalization to isotropic hardening is incorporated at Gauss point level in Chapter 5.

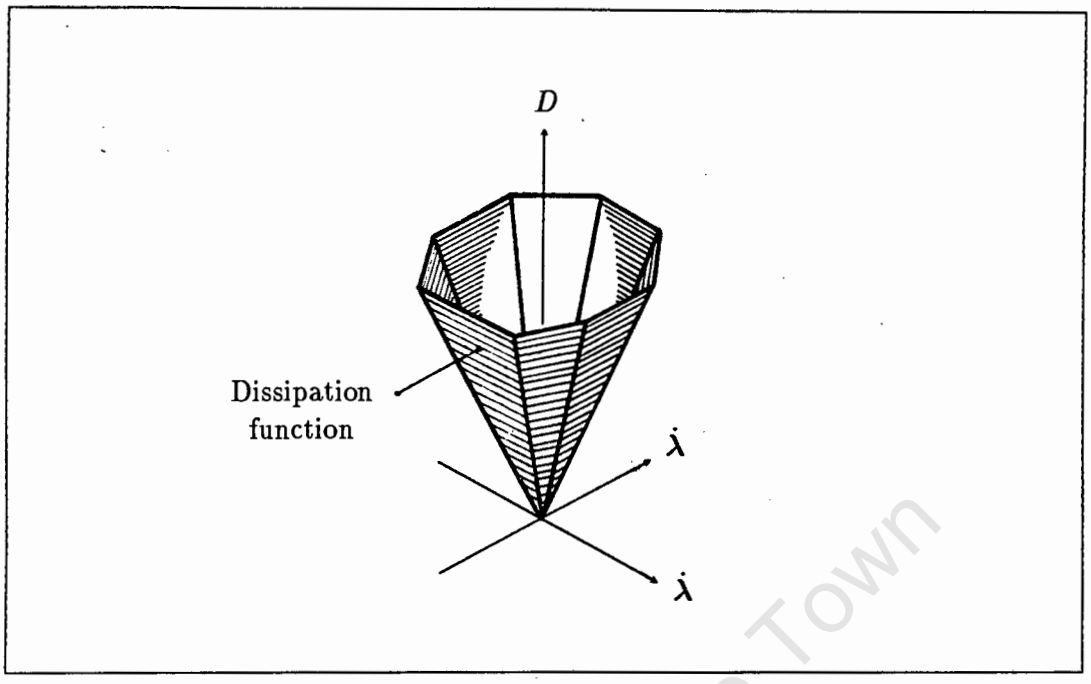

Figure 3.1: The dissipation function

3.3 TIME DISCRETIZATION

In the incremental problem, the time history is divided into discrete intervals by defining discrete instants in time. Two such sequences of discrete instants are introduced. The first sequence is denoted by $t_0 = 0, t_1, \ldots, t_{n-1}, t_n, t_{n+1}, \ldots$, and for simplicity it is assumed that the interval between the discrete instants is fixed, and is given by Δt. The second sequence will be denoted by $T_1, \ldots, T_{n-1}, T_n, T_{n+1}, \ldots$. Again, the interval between these instants is fixed at Δt and $t_n < T_{n+1} \leq t_{n+1}$. The two sequences are shown diagrammatically in Figure 3.2.

Consider the interval (t_n, T_{n+1}). For this interval

$$T_{n+1} - t_n = \alpha \Delta t \tag{3.5a}$$

where $0 < \alpha \leq 1$, and

$$t_{n+1} - T_{n+1} = (1 - \alpha)\Delta t \quad . \tag{3.5b}$$

Values of the variables $\boldsymbol{u}(t), \boldsymbol{\lambda}(t), \boldsymbol{P}(t), \boldsymbol{\chi}(t)$ at time t_n will be denoted by $\boldsymbol{u}_n, \boldsymbol{\lambda}_n, \boldsymbol{P}_n, \boldsymbol{\chi}_n$, and at time T_n by $\boldsymbol{u}_N, \boldsymbol{\lambda}_N, \boldsymbol{P}_N, \boldsymbol{\chi}_N$.

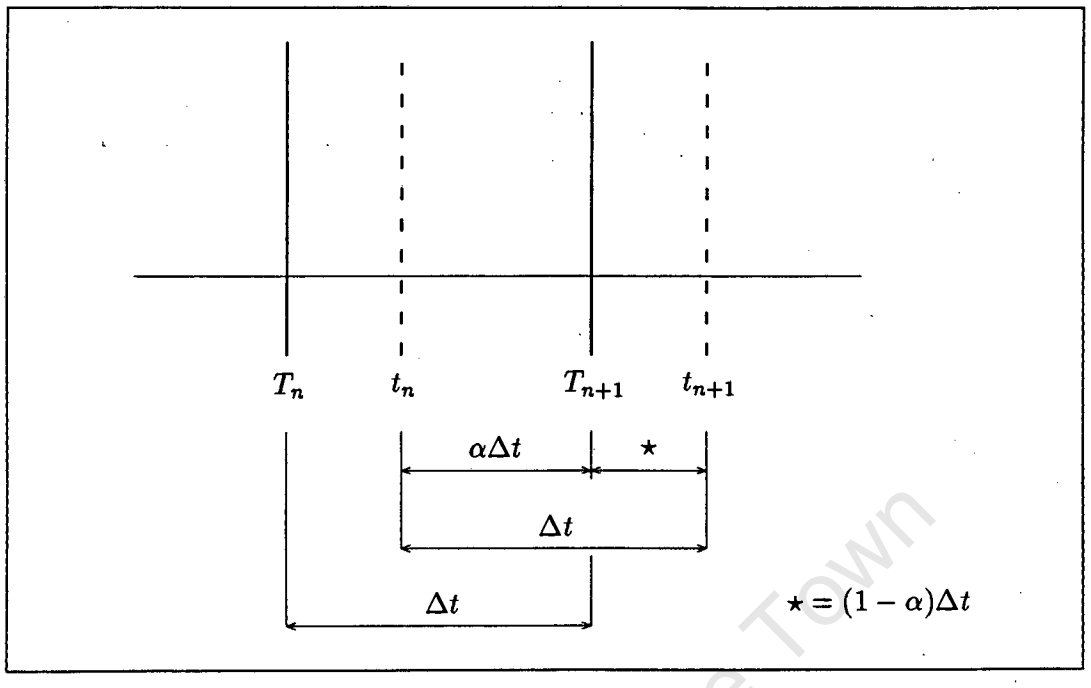

Figure 3.2: The time sequences t and T

The governing equations must be satisfied at the end of the discrete interval (t_n, T_{n+1}), so that at time T_{n+1}

$$\boldsymbol{K}\boldsymbol{u}_{N+1} + \boldsymbol{L}\boldsymbol{\lambda}_{N+1} = \boldsymbol{P}_{N+1} \quad , \tag{3.6a}$$

$$\boldsymbol{L}^T\boldsymbol{u}_{N+1} + \boldsymbol{H}\boldsymbol{\lambda}_{N+1} + \left.\frac{\partial D}{\partial \dot{\boldsymbol{\lambda}}}\right|_{\dot{\boldsymbol{\lambda}}_{N+1}} = 0 \quad . \tag{3.6b}$$

The term involving $\dot{\boldsymbol{\lambda}}_{N+1}$ in eqn. (3.6b) must be expressed in terms of $\boldsymbol{\lambda}_{N+1}$ and previous values $\boldsymbol{\lambda}_n$, $\boldsymbol{\lambda}_N$, $\boldsymbol{\lambda}_{n-1}, \cdots$. This is achieved by applying the classical backward difference assumption over the interval (t_n, T_{n+1}). This assures a choice of a minimum work path for the change in stress from $\boldsymbol{\sigma}_n$ to $\boldsymbol{\sigma}_{n+1}^{(i+1)}$ [1], and in turn the existence of the minimum principle[3,6]. In this assumption $\dot{\boldsymbol{\lambda}}(t)$ is fixed in direction over the interval (t_n, T_{n+1}), and thus

$$\dot{\boldsymbol{\lambda}}(t) = \dot{\boldsymbol{\lambda}}_{N+1} = \frac{1}{\alpha\Delta t}[\boldsymbol{\lambda}_{N+1} - \boldsymbol{\lambda}_n] = \frac{1}{\alpha\Delta t}\Delta\tilde{\boldsymbol{\lambda}}_{n+1} \quad , \tag{3.7}$$

where

$$\Delta\tilde{\boldsymbol{\lambda}}_{n+1} = \boldsymbol{\lambda}_{N+1} - \boldsymbol{\lambda}_n \quad . \tag{3.8}$$

Since D is homogeneous and of degree one, the derivative $\partial D/\partial\dot{\boldsymbol{\lambda}}$ is homogeneous and of degree zero. This implies that $\partial D/\partial\dot{\boldsymbol{\lambda}}$ depends only on the direction of $\dot{\boldsymbol{\lambda}}$, and not on its magnitude. Since $\alpha > 0$, this implies that

$$\left.\frac{\partial D}{\partial\dot{\boldsymbol{\lambda}}}\right|_{\dot{\boldsymbol{\lambda}}_{N+1}} = \left.\frac{\partial D}{\partial\dot{\boldsymbol{\lambda}}}\right|_{\Delta\tilde{\boldsymbol{\lambda}}_{n+1}} \quad . \tag{3.9a}$$

It is also convenient to regard D as a function of $\Delta\boldsymbol{\lambda}$, rather than $\dot{\boldsymbol{\lambda}}$, and we can thus put

$$\left.\frac{\partial D(\dot{\boldsymbol{\lambda}})}{\partial\dot{\boldsymbol{\lambda}}}\right|_{\dot{\boldsymbol{\lambda}}_{N+1}} = \left.\frac{\partial D(\Delta\boldsymbol{\lambda})}{\partial\Delta\boldsymbol{\lambda}}\right|_{\Delta\tilde{\boldsymbol{\lambda}}_{n+1}} \quad . \tag{3.9b}$$

Introducing the increment

$$\Delta\tilde{\boldsymbol{u}}_{n+1} = \boldsymbol{u}_{N+1} - \boldsymbol{u}_n \quad , \tag{3.10}$$

the governing equations (eqns. 3.6) may then be recast in incremental form:

$$\boldsymbol{K}\Delta\tilde{\boldsymbol{u}}_{n+1} + \boldsymbol{L}\Delta\tilde{\boldsymbol{\lambda}}_{n+1} = \boldsymbol{P}_{N+1} - (\boldsymbol{K}\boldsymbol{u}_n + \boldsymbol{L}\boldsymbol{\lambda}_n) \quad , \tag{3.11a}$$

$$\boldsymbol{L}^T\Delta\tilde{\boldsymbol{u}}_{n+1} + \boldsymbol{H}\Delta\tilde{\boldsymbol{\lambda}}_{n+1} + \left.\frac{\partial D}{\partial\Delta\boldsymbol{\lambda}}\right|_{\Delta\tilde{\boldsymbol{\lambda}}_{n+1}} = -(\boldsymbol{L}^T\boldsymbol{u}_n + \boldsymbol{H}\boldsymbol{\lambda}_n) \quad . \tag{3.11b}$$

The solution of eqns. (3.11) can also be regarded as the minimization of a convex function $U_p(\Delta\boldsymbol{u}, \Delta\boldsymbol{\lambda})$, given by

$$\begin{aligned} U_p = {} & \tfrac{1}{2}\left\{\begin{array}{c}\Delta\boldsymbol{u}\\ \Delta\boldsymbol{\lambda}\end{array}\right\}^T \left[\begin{array}{cc}\boldsymbol{K} & \boldsymbol{L}\\ \boldsymbol{L}^T & \boldsymbol{H}\end{array}\right]\left\{\begin{array}{c}\Delta\boldsymbol{u}\\ \Delta\boldsymbol{\lambda}\end{array}\right\} + D(\Delta\boldsymbol{\lambda}) \\ & - \Delta\boldsymbol{u}^T(\boldsymbol{P}_{N+1} - \boldsymbol{K}\boldsymbol{u}_n - \boldsymbol{L}\boldsymbol{\lambda}_n) + \Delta\boldsymbol{\lambda}^T(\boldsymbol{L}^T\boldsymbol{u}_n + \boldsymbol{H}\boldsymbol{\lambda}_n) \quad . \end{aligned} \tag{3.12}$$

3.4 ITERATIVE SOLUTION ALGORITHM

In a finite element implementation, the incremental problem is solved by an iterative procedure based on the Newton Raphson method[7,10]. The classical Newton-Raphson algorithm can be considered to be a two-step iteration made up of what will be referred to as a predictor step and a corrector step.

Let the trial solution after the i-th iteration be $\Delta\tilde{\boldsymbol{u}}_{n+1}^{(i)}, \Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{(i)}$. In the predictor step, eqns. (3.11) are approximated by expanding $\partial D/\partial\Delta\boldsymbol{\lambda}$ to first order about the trial solution, and solved for a new trial value of the displacement increment $\Delta\tilde{\boldsymbol{u}}_{n+1}^{(i+1)}$. This new value is then substituted into eqn. (3.11b), which is solved for the new trial value of the internal variable increment $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{(i+1)}$.

The expansion of $\partial D/\partial\Delta\boldsymbol{\lambda}$ to first order is equivalent to the expansion of $D(\Delta\boldsymbol{\lambda})$ in the convex function U_p (eqn. 3.12) to second order. In order to avoid difficulties, D is expanded only at Gauss points where $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{(i)}$ is non-zero; these Gauss points, associated with the expansion, are defined as active for this iteration. Inactive Gauss points are ignored, eliminating the contributions for these Gauss points from eqns. (3.11a) and (3.11b). Reduced matrices $\hat{\boldsymbol{L}}, \widehat{\boldsymbol{H}}$ are thus defined.

The solution of the approximate, linearised form of eqns. (3.11) is denoted by $\Delta\tilde{\boldsymbol{u}}_{n+1}^{(i+1)}, \Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{i+\frac{1}{2}}$. The vector $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{i+\frac{1}{2}}$, defined only at the active Gauss points, is an intermediate value which is not computed. It is convenient to define the incremental changes

$$\Delta\hat{\boldsymbol{u}} = \Delta\tilde{\boldsymbol{u}}_{n+1}^{(i+1)} - \Delta\tilde{\boldsymbol{u}}_{n+1}^{(i)} \quad , \tag{3.13a}$$

$$\Delta\hat{\boldsymbol{\lambda}} = \Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{i+\frac{1}{2}} - \Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{(i)} \quad . \tag{3.13b}$$

The linearised form of eqns. (3.11) can then be written as

$$\boldsymbol{K}\Delta\hat{\boldsymbol{u}} + \hat{\boldsymbol{L}}\Delta\hat{\boldsymbol{\lambda}} = \boldsymbol{P}_{N+1} - \boldsymbol{K}(\boldsymbol{u}_n + \Delta\tilde{\boldsymbol{u}}_{n+1}^{(i)}) - \boldsymbol{L}(\boldsymbol{\lambda}_n + \Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{(i)}) \quad , \tag{3.14a}$$

$$\hat{\boldsymbol{L}}^T\Delta\hat{\boldsymbol{u}} + (\widehat{\boldsymbol{H}} + \hat{\boldsymbol{C}})\Delta\hat{\boldsymbol{\lambda}} = -\boldsymbol{\chi}_{N+1}^{(i)} - \hat{\boldsymbol{L}}^T(\boldsymbol{u}_n + \Delta\tilde{\boldsymbol{u}}_{n+1}^{(i)}) - \widehat{\boldsymbol{H}}(\boldsymbol{\lambda}_n + \Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{(i)}) = 0. \tag{3.14b}$$

The right hand side of eqn. (3.14b) is zero as a consequence of the application of

the corrector step in the i-th iteration; this point will be referred to later. In this equation

$$\chi_{N+1}^{(i)} = \left.\frac{\partial D}{\partial \Delta\lambda}\right|_{\Delta\tilde{\lambda}_{n+1}^{(i)}} \tag{3.14c}$$

at active Gauss points, and

$$\hat{C} = \left.\frac{\partial^2 D}{\partial \Delta\lambda \partial \Delta\lambda}\right|_{\Delta\tilde{\lambda}_{n+1}^{(i)}} . \tag{3.14d}$$

Except for situations in which unrestricted plastic flow may occur in the body (which are not of direct interest), the matrix $(\widehat{H} + \hat{C})$ is positive definite[8]. Thus $\Delta\hat{\lambda}$ may be eliminated using eqn. (3.14b), and eqn. (3.14a) may be written as

$$\boldsymbol{K}_c \Delta\hat{\boldsymbol{u}} = \tilde{\boldsymbol{R}}_{n+1}^{(i+1)} \quad , \tag{3.15a}$$

where

$$\boldsymbol{K}_c = \boldsymbol{K} - \hat{\boldsymbol{L}}(\widehat{\boldsymbol{H}} + \hat{\boldsymbol{C}})^{-1}\hat{\boldsymbol{L}}^T \quad , \tag{3.15b}$$

$$\tilde{\boldsymbol{R}}_{n+1}^{(i+1)} = \boldsymbol{P}_{N+1} - \boldsymbol{K}(\boldsymbol{u}_n + \Delta\tilde{\boldsymbol{u}}_{n+1}^{(i)}) - \boldsymbol{L}(\boldsymbol{\lambda}_n + \Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{(i)}) \quad . \tag{3.15c}$$

The matrix $\boldsymbol{K}_c$ is the consistent tangent modulus defined at structural level, and is symmetric and positive definite. The vector $\tilde{\boldsymbol{R}}_{n+1}^{(i+1)}$ is the residual at the end of the i-th iteration.

In the corrector step, the value

$$\Delta\tilde{\boldsymbol{u}}_{n+1}^{(i+1)} = \Delta\tilde{\boldsymbol{u}}_{n+1}^{(i)} + \Delta\hat{\boldsymbol{u}} \tag{3.16}$$

is adopted as the new trial displacement increment, and eqn. (3.11b) is written as

$$\boldsymbol{H}\Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{(i+1)} + \left.\frac{\partial D}{\partial \Delta\boldsymbol{\lambda}}\right|_{\Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{(i+1)}} = -\boldsymbol{L}^T(\boldsymbol{u}_n + \Delta\tilde{\boldsymbol{u}}_{n+1}^{(i+1)}) - \boldsymbol{H}\boldsymbol{\lambda}_n \quad . \tag{3.17}$$

These equations can be solved independently at each Gauss point. Note that the concept of active and inactive Gauss points falls away at the end of the predictor step; eqn. (3.17) is applied at every Gauss point.

Eqn. (3.17) can also be written as

$$\chi_{N+1}^{(i+1)} = -\boldsymbol{L}^T(\boldsymbol{u}_n + \Delta\tilde{\boldsymbol{u}}_{n+1}^{(i+1)}) - \boldsymbol{H}(\boldsymbol{\lambda}_n + \Delta\tilde{\boldsymbol{\lambda}}_{n+1}^{(i+1)}) \quad . \tag{3.18}$$

This justifies setting the right hand side of eqn. (3.14b) to zero.

The predictor and corrector steps defined in this way are exactly equivalent to the consistent tangent predictors and the return algorithms classically used in Newton-Raphson schemes in finite element analysis[2,5,8]. Note that in the first iteration, the starting trial values are taken as $\Delta\tilde{\boldsymbol{u}}_{n+1} = 0, \Delta\tilde{\boldsymbol{\lambda}}_{n+1} = 0$, and by definition all Gauss points are inactive in the predictor step.

If the time sequences coincide i.e. $t_{n+1} = T_{n+1}$ or $\alpha = 1$ in eqns. (3.5), the formulation of the problem and the predictor and corrector steps reduces to the classical backward difference incremental formulation (i.e. for the interval (t_n, t_{n+1})).

3.5 CONCLUSIONS

An internal variable formulation for the finite element solution of the incremental elastic-plastic problem is presented. It is shown that the choice of a backward difference assumption leads to the standard Newton-Raphson algorithm with the consistent tangent modulus. The advantages linked to the use of the consistent tangent modulus, as discussed in the previous chapter, thus apply.

REFERENCES

1. Carter, P. & Martin, J.B., Work Bounding Functions for Plastic Materials, *Journal of Applied Mechanics*, **43**, 434-438, 1976.

2. Bird, W.W. & Martin, J.B., Consistent Predictors and the Solution of the Piecewise Holonomic Incremental Problem in Elasto-plasticity, *Engineering Structures*, **12**, 9-14, 1990.

3. Maier, G. Quadratic Programming and Theory of Elastic-Perfectly Plastic Structures, *Meccanica*, **3**, 265-273, 1968.

4. Martin, J.B., An Internal Variable Approach to the Formulation of Finite Element Problems in Plasticity, *Physical Nonlinearities in Structural Analysis*, eds. J. Holt & J. Lemaitre, Springer-Verlag, Berlin, 165-176, 1981.

5. Martin, J.B. & Nappi, A., An Internal Variable Formulation of Perfectly Plastic and Linear Kinematic and Isotropic Hardening Relations with a Von Mises Yield Condition, *European Journal of Mechanics, A-Solids*, **9**, 107-131, 1990.

6. Martin, J.B., Reddy, B.D., Griffin, T.B. & Bird, W.W., Applications of Mathematical Programming Concepts to Incremental Elastic-Plastic Analysis, *Engineering Structures*, **9**, 171-176, 1987.

7. Owen, D.R.J. & Hinton, E., *Finite Elements in Plasticity - Theory and Practice*, Pineridge Press, Swansea, 1980.

8. Reddy, B.D. & Martin, J.B., Algorithms for the Solution of Internal Variable Problems in Plasticity, *Computer Methods in Applied Mechanics and Engineering*, (to appear).

9. Rice, J.R., Inelastic Constitutive Relations for Solids: an Internal Variable Theory and its Application to Metal Plasticity, *Journal of Mechanics and Physics of Solids*, **19**, 433-455, 1971.

10. Zienkiewicz, O.C., *The Finite Element Method*, McGraw-Hill, London, 1977.

CHAPTER 4
TIME INTEGRATION SCHEMES FOR PLASTICITY

4.1 INTRODUCTION

It has been established that the symmetry of the consistent tangent modulus is an important criterion of integration algorithms for the finite element analysis of convex elastic-plastic problems[2]. The implicit backward difference algorithm leads to such symmetry[6,2], but is only first order accurate[3]. The question is thus to find an algorithm which is second order accurate and which preserves the symmetry of the consistent tangent modulus.

Simo and Taylor[7] and Simo and Govindjee[5] consider a generalized midpoint rule in which the governing equations (equilibrium and the constitutive law) are satisfied at the generalized midpoint[4]. They show that this rule preserves symmetry and further, when the true midpoint is chosen, provides second order accuracy.

In this chapter, a generalized trapezoidal rule, that is symmetry preserving and contains a second order accurate algorithm, is introduced. This rule is not the same as that of Ortiz and Popov[3], but appears to be more consistent with a generalized trapezoidal rule for creep.

The relation between the generalized trapezoidal rule, the backward difference rule and the generalized midpoint rule of Simo *et al*[5,7] is examined in terms of the internal variable formulation of chapter 3. It is shown that, by suitable linear extrapolation, we may generalize the backward difference algorithm while preserving symmetry of the consistent tangent modulus. If results from the previous step are incorporated, the generalized trapezoidal algorithm is recovered, while by projecting forward, the generalized midpoint algorithm of Simo and Govindjee[5] is obtained.

4.2 THE GENERALIZED TRAPEZOIDAL RULE

Utilizing the time discretization introduced in chapter 3, we consider the time sequence T_n, T_{n+1} to be the prime sequence. The backward difference formulation of chapter 3 provided $\boldsymbol{u}_{N+1}, \boldsymbol{\lambda}_{N+1}$ at time T_{n+1}, given $\boldsymbol{u}_{n+1}, \boldsymbol{\lambda}_{n+1}$ at time t_{n+1}. The problem is now framed in terms of the interval (T_n, T_{n+1}); i.e. find $\boldsymbol{u}_{N+1}, \boldsymbol{\lambda}_{N+1}$,

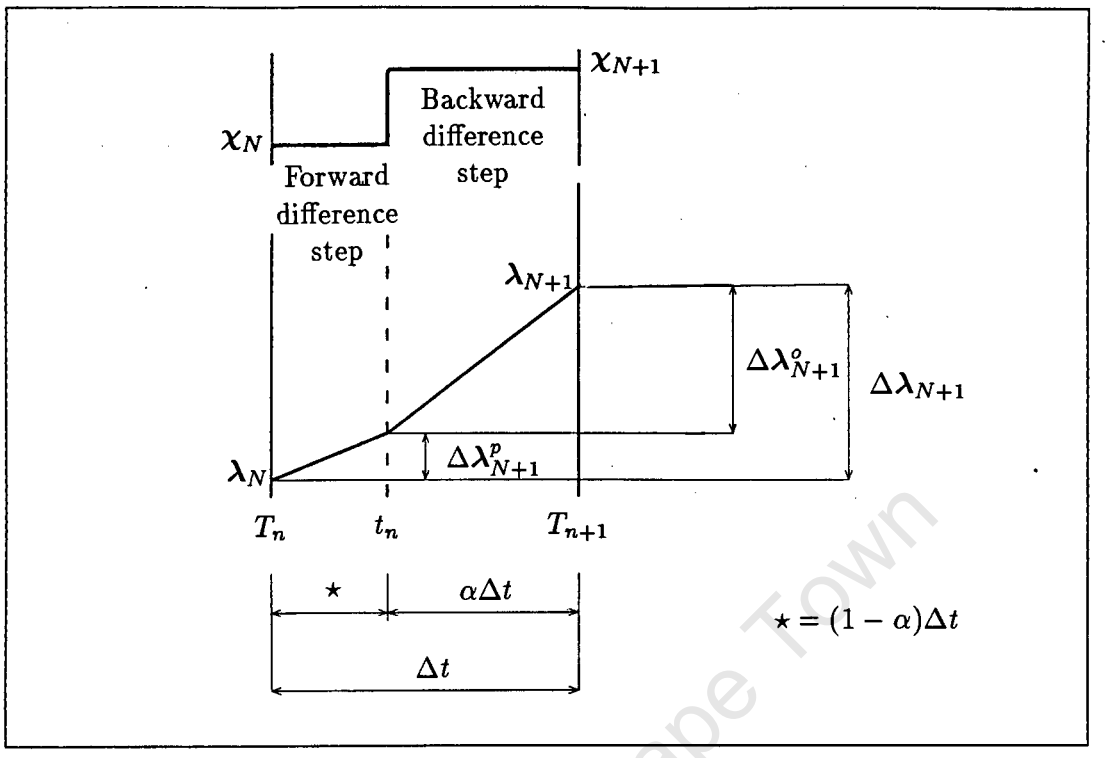

Figure 4.1: The generalized trapezoidal rule

given $\boldsymbol{u}_N, \boldsymbol{\lambda}_N$.

We apply a generalized trapezoidal rule to the increment

$$\Delta\boldsymbol{\lambda}_{N+1} = \boldsymbol{\lambda}_{N+1} - \boldsymbol{\lambda}_N \quad . \tag{4.1}$$

Thus, for $0 \leq \alpha \leq 1$

$$\Delta\boldsymbol{\lambda}_{N+1} = \alpha\dot{\boldsymbol{\lambda}}_{N+1}\Delta t + (1-\alpha)\dot{\boldsymbol{\lambda}}_N\Delta t \quad . \tag{4.2}$$

Eqn.(4.2) is consistent with the creep problem in which the creep rates are known. However, due to the non-real time nature of the elastic-plastic problem, only the plastic increments and not the rates are known and eqn.(4.2) can be written as

$$\Delta\boldsymbol{\lambda}_{N+1} = \Delta\boldsymbol{\lambda}^o_{N+1} + \Delta\boldsymbol{\lambda}^p_{N+1} \quad , \tag{4.3}$$

where

$$\Delta\boldsymbol{\lambda}^{o}_{N+1} = \alpha\dot{\boldsymbol{\lambda}}_{N+1}\Delta t \quad , \tag{4.4a}$$

$$\Delta\boldsymbol{\lambda}^{p}_{N+1} = (1-\alpha)\dot{\boldsymbol{\lambda}}_{N}\Delta t \quad = \frac{(1-\alpha)}{\alpha}\Delta\boldsymbol{\lambda}^{o}_{N} \quad . \tag{4.4b}$$

Eqn. (4.3) is the incremental form of the generalized trapezoidal rule. This can be can be visualized as a two step integration scheme and is illustrated in Figure 4.1. First, a forward difference integration scheme is applied over part of the interval (T_n, t_n). Then a backward difference scheme is applied over the remainder of the interval (t_n, T_{n+1}). The relative proportions of these two steps are governed by the parameter α.

From the basic framework, $\boldsymbol{\chi}_{N+1}$ and hence $\boldsymbol{\chi}_N$ will lie on or inside the yield surface. The increment $\Delta\boldsymbol{\lambda}_{N+1}$, in keeping with the notion of a two step integration, is divided into two parts. The component $\Delta\boldsymbol{\lambda}^{p}_{N+1}$ is associated with $\boldsymbol{\chi}_N$: if $\boldsymbol{\chi}_N$ lies on the yield surface, $\Delta\boldsymbol{\lambda}^{p}_{N+1}$ will be normal to the yield surface at $\boldsymbol{\chi}_N$, while if $\boldsymbol{\chi}_N$ lies within the yield surface, $\Delta\boldsymbol{\lambda}^{p}_{N+1}$ will be zero. Similarly, $\Delta\boldsymbol{\lambda}^{o}_{N+1}$ is associated with $\boldsymbol{\chi}_{N+1}$.

This is a distinct difference from the formulation suggested by Ortiz and Popov[3], where the direction but not the magnitude of $\Delta\boldsymbol{\lambda}^{p}_{N+1}$ is assumed known if $\boldsymbol{\chi}_N$ lies on the yield surface. If $\boldsymbol{\chi}_N$ lies within the yield surface, unlike this formulation, the direction of $\Delta\boldsymbol{\lambda}^{p}_{N+1}$ is then evaluated at the contact stress and not set equal to zero.

In keeping with the formulation of Ortiz and Popov, the governing equations (equilibrium and plastic consistency) are enforced at time T_{n+1}. First, by defining the increment

$$\Delta\boldsymbol{u}_{N+1} = \boldsymbol{u}_{N+1} - \boldsymbol{u}_{N} \quad , \tag{4.5}$$

the governing equations (eqns. 3.6) can be written in incremental form

$$\boldsymbol{K}\Delta\boldsymbol{u}_{N+1} + \boldsymbol{L}\Delta\boldsymbol{\lambda}^{o}_{N+1} = \boldsymbol{P}_{N+1} - \boldsymbol{K}\boldsymbol{u}_{N} - \boldsymbol{L}(\boldsymbol{\lambda}_{N} + \Delta\boldsymbol{\lambda}^{p}_{N+1}) \quad , \tag{4.6a}$$

$$\boldsymbol{L}^{T}\Delta\boldsymbol{u}_{N+1} + \boldsymbol{H}\Delta\boldsymbol{\lambda}^{o}_{N+1} + \left.\frac{\partial D}{\partial\dot{\boldsymbol{\lambda}}}\right|_{\dot{\boldsymbol{\lambda}}_{N+1}} = -\boldsymbol{L}^{T}\boldsymbol{u}_{N} - \boldsymbol{H}(\boldsymbol{\lambda}_{N} + \Delta\boldsymbol{\lambda}^{p}_{N+1}). \tag{4.6b}$$

Noting that $D(\dot{\boldsymbol{\lambda}})$ is homogeneous and of degree one in its argument, we can put

$$\left.\frac{\partial D(\dot{\boldsymbol{\lambda}})}{\partial \dot{\boldsymbol{\lambda}}}\right|_{\dot{\boldsymbol{\lambda}}_{N+1}} = \left.\frac{\partial D(\Delta\boldsymbol{\lambda})}{\partial \Delta\boldsymbol{\lambda}}\right|_{\Delta\boldsymbol{\lambda}^{\circ}{}_{N+1}} \quad , \tag{4.7}$$

and eqns. (4.6) become

$$\boldsymbol{K}\Delta\boldsymbol{u}_{N+1} + \boldsymbol{L}\Delta\boldsymbol{\lambda}^{\circ}_{N+1} = \boldsymbol{P}_{N+1} - \boldsymbol{K}\boldsymbol{u}_N - \boldsymbol{L}(\boldsymbol{\lambda}_N + \Delta\boldsymbol{\lambda}^{p}_{N+1}) \quad , \tag{4.8a}$$

$$\boldsymbol{L}^T\Delta\boldsymbol{u}_{N+1} + \boldsymbol{H}\Delta\boldsymbol{\lambda}^{\circ}_{N+1} + \left.\frac{\partial D}{\partial \Delta\boldsymbol{\lambda}}\right|_{\Delta\boldsymbol{\lambda}^{\circ}{}_{N+1}} = -\boldsymbol{L}^T\boldsymbol{u}_N - \boldsymbol{H}(\boldsymbol{\lambda}_N + \Delta\boldsymbol{\lambda}^{p}_{N+1}). \tag{4.8b}$$

If we set $\alpha = 1$ in eqn. (4.4b), we recover a backward difference formulation for the interval (T_n, T_{n+1}).

To further explore the relation between the generalized trapezoidal rule and the backward difference scheme, the displacement increment, in a similar way to the internal variable increment $\Delta\boldsymbol{\lambda}_{N+1}$, is divided into two parts:

$$\Delta\boldsymbol{u}_{N+1} = \boldsymbol{u}_{N+1} - \boldsymbol{u}_N = \Delta\boldsymbol{u}^{p}_{N+1} + \Delta\boldsymbol{u}^{\circ}_{N+1} \quad . \tag{4.9}$$

First, we associate $\Delta\boldsymbol{u}^{\circ}_{N+1}, \Delta\boldsymbol{\lambda}^{\circ}_{N+1}$ with the increments of our basic formulation

$$\Delta\boldsymbol{u}^{\circ}_{N+1} = \Delta\tilde{\boldsymbol{u}}_{n+1} \quad , \qquad \Delta\boldsymbol{\lambda}^{\circ}_{N+1} = \Delta\tilde{\boldsymbol{\lambda}}_{n+1} \quad , \tag{4.10}$$

and adopt the governing equations (eqns. 3.11) for the backward difference formulation for the interval (t_n, T_{n+1}). Second we assume that $\Delta\boldsymbol{u}^{p}_{N+1}, \Delta\boldsymbol{\lambda}^{p}_{N+1}$ are determined by projecting forward from the previous interval. Specifically, we put

$$\Delta\boldsymbol{u}^{p}_{N+1} = \frac{(1-\alpha)}{\alpha}\Delta\boldsymbol{u}^{\circ}_{N} \quad , \tag{4.11a}$$

$$\Delta\boldsymbol{\lambda}^{p}_{N+1} = \frac{(1-\alpha)}{\alpha}\Delta\boldsymbol{\lambda}^{\circ}_{N} \quad . \tag{4.11b}$$

It follows then that $\Delta\boldsymbol{u}^{p}_{N+1}, \Delta\boldsymbol{\lambda}^{p}_{N+1}$ are known *a priori* in the incremental problem. To formulate the problem in terms of the increments $\Delta\boldsymbol{u}^{p}_{N+1}, \Delta\boldsymbol{\lambda}^{p}_{N+1}, \Delta\boldsymbol{u}^{\circ}_{N+1}, \Delta\boldsymbol{\lambda}^{\circ}_{N+1}$ we put

$$\boldsymbol{u}_n = \boldsymbol{u}_N + \Delta\boldsymbol{u}^p_{N+1} \quad , \tag{4.12a}$$

$$\boldsymbol{\lambda}_n = \boldsymbol{\lambda}_N + \Delta\boldsymbol{\lambda}^p_{N+1} \quad , \tag{4.12b}$$

and substitute into the governing equations (eqns. 3.11). This gives

$$\boldsymbol{K}\Delta\boldsymbol{u}^\circ_{N+1} + \boldsymbol{L}\Delta\boldsymbol{\lambda}^\circ_{N+1} = \boldsymbol{P}_{N+1} - \boldsymbol{K}(\boldsymbol{u}_N + \Delta\boldsymbol{u}^p_{N+1}) - \boldsymbol{L}(\boldsymbol{\lambda}_N + \Delta\boldsymbol{\lambda}^p_{N+1}), \tag{4.13a}$$

$$\begin{aligned}\boldsymbol{L}^T\Delta\boldsymbol{u}^\circ_{N+1} + \boldsymbol{H}\Delta\boldsymbol{\lambda}^\circ_{N+1} + \left.\frac{\partial D}{\partial\Delta\boldsymbol{\lambda}}\right|_{\Delta\boldsymbol{\lambda}^\circ{}_{N+1}} &= -\boldsymbol{L}^T(\boldsymbol{u}_N + \Delta\boldsymbol{u}^p_{N+1}) \\ &\quad -\boldsymbol{H}(\boldsymbol{\lambda}_N + \Delta\boldsymbol{\lambda}^p_{N+1}) \quad .\end{aligned} \tag{4.13b}$$

On inspecting these equations, it is evident that it is not necessary to distinguish between $\Delta\boldsymbol{u}^p_{N+1}$ and $\Delta\boldsymbol{u}^\circ_{N+1}$; these terms can be combined, and eqns. (4.13) are rewritten as

$$\boldsymbol{K}\Delta\boldsymbol{u}_{N+1} + \boldsymbol{L}\Delta\boldsymbol{\lambda}^\circ_{N+1} = \boldsymbol{P}_{N+1} - \boldsymbol{K}\boldsymbol{u}_N - \boldsymbol{L}(\boldsymbol{\lambda}_N + \Delta\boldsymbol{\lambda}^p_{N+1}) \quad , \tag{4.14a}$$

$$\boldsymbol{L}^T\Delta\boldsymbol{u}_{N+1} + \boldsymbol{H}\Delta\boldsymbol{\lambda}^\circ_{N+1} + \left.\frac{\partial D}{\partial\Delta\boldsymbol{\lambda}}\right|_{\Delta\boldsymbol{\lambda}^\circ{}_{N+1}} = -\boldsymbol{L}^T\boldsymbol{u}_N - \boldsymbol{H}(\boldsymbol{\lambda}_N + \Delta\boldsymbol{\lambda}^p_{N+1}). \tag{4.14b}$$

These equations retain symmetry and convexity in $\Delta\boldsymbol{u}_{N+1}, \Delta\boldsymbol{\lambda}^\circ_{N+1}$ and are equivalent to eqns. (4.8).

In considering the predictor step, we put

$$\Delta\hat{\boldsymbol{u}}_{N+1} = \Delta\boldsymbol{u}^{(i+1)}_{N+1} - \Delta\boldsymbol{u}^{(i)}_{N+1} \quad . \tag{4.15}$$

Substituting into eqns. (3.15), we write

$$\boldsymbol{K}_c\Delta\hat{\boldsymbol{u}}_{N+1} = \boldsymbol{R}^{(i+1)}_{N+1} \tag{4.16}$$

where $\boldsymbol{K}_c$ is defined as in eqn. (3.15b). It should be noted, however, that a Gauss point is active if $\Delta\boldsymbol{\lambda}^{\circ\,(i)}_{N+1}$ is non-zero. The residual $\boldsymbol{R}^{(i+1)}_{N+1}$ is given by

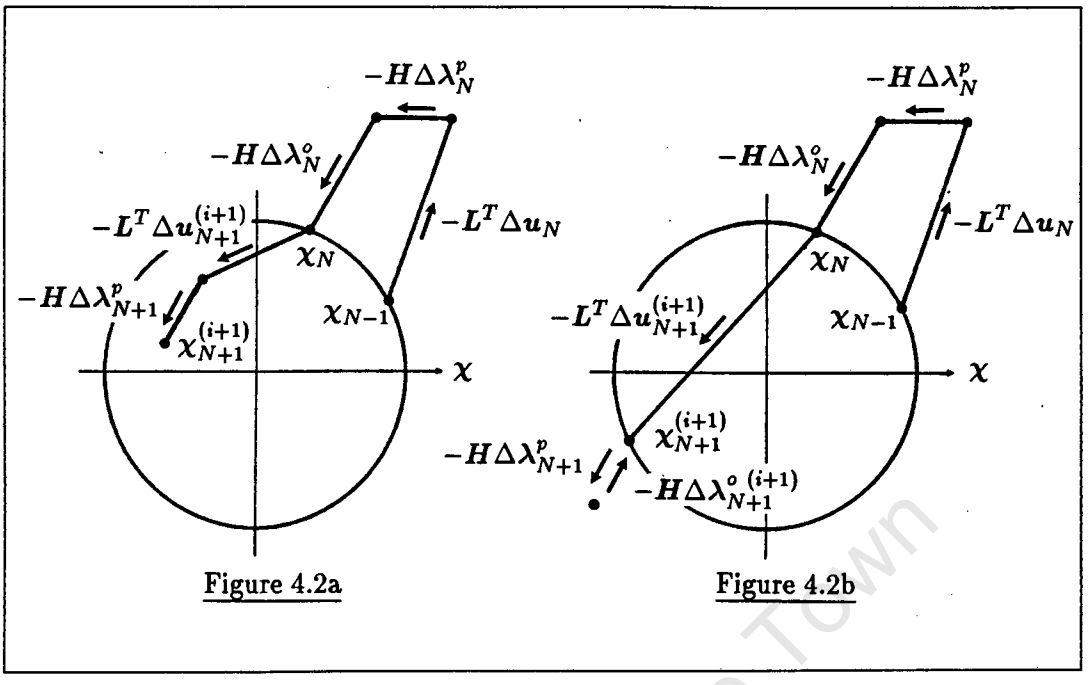

Figure 4.2a Figure 4.2b

Figure 4.2: The corrector step for the generalized trapezoidal rule

$$\boldsymbol{R}_{N+1}^{(i+1)} = \boldsymbol{P}_{N+1} - \boldsymbol{K}(\boldsymbol{u}_N + \Delta\boldsymbol{u}_{N+1}^{(i)}) - \boldsymbol{L}(\boldsymbol{\lambda}_N + \Delta\boldsymbol{\lambda}_{N+1}^{(i)}) \quad , \tag{4.17a}$$

with

$$\Delta\boldsymbol{\lambda}_{N+1}^{(i)} = \Delta\boldsymbol{\lambda}_{N+1}^{p} + \Delta\boldsymbol{\lambda}_{N+1}^{\circ\,(i)} \quad . \tag{4.17b}$$

The corrector step is written, from eqn. (3.17), as

$$\boldsymbol{H}\Delta\boldsymbol{\lambda}_{N+1}^{\circ\,(i+1)} + \frac{\partial D}{\partial\Delta\boldsymbol{\lambda}}\bigg|_{\Delta\boldsymbol{\lambda}_{N+1}^{\circ\,(i+1)}} = -\boldsymbol{L}^T(\boldsymbol{u}_N + \Delta\boldsymbol{u}_{N+1}^{(i+1)}) - \boldsymbol{H}(\boldsymbol{\lambda}_N + \Delta\boldsymbol{\lambda}_{N+1}^{p}). \tag{4.18}$$

The essential features of the backward difference formulation are thus retained, in the sense that both the predictor modulus $\boldsymbol{K}_c$ is symmetric, and the return algorithm is applied in the same way.

The corrector step is shown diagrammatically as a return map in $\boldsymbol{\chi}$ space in Figure 4.2. Eqn. (4.18) is written as

$$\boldsymbol{\chi}_{N+1}^{(i+1)} = (\boldsymbol{\chi}_N - \boldsymbol{L}^T\Delta\boldsymbol{u}_{N+1}^{(i+1)}) - \boldsymbol{H}\Delta\boldsymbol{\lambda}_{N+1}^{p} - \boldsymbol{H}\Delta\boldsymbol{\lambda}_{N+1}^{\circ\,(i+1)} \tag{4.19a}$$

where

$$\boldsymbol{\chi}_{N+1}^{(i+1)} = \left.\frac{\partial D}{\partial \Delta\boldsymbol{\lambda}}\right|_{\Delta\boldsymbol{\lambda}_{N+1}^{o\,(i+1)}} \quad . \tag{4.19b}$$

The vector $(\boldsymbol{\chi}_N - \boldsymbol{L}^T \Delta\boldsymbol{u}_{N+1}^{(i+1)})$ is the equivalent elastically predicted value of $\boldsymbol{\chi}_{N+1}^{(i+1)}$, the vector $-\boldsymbol{H}\Delta\boldsymbol{\lambda}_{N+1}^{p}$ is the equivalent return associated with the forward difference scheme and the vector $-\boldsymbol{H}\Delta\boldsymbol{\lambda}_{N+1}^{o\,(i+1)}$ is the equivalent return associated with the backward difference scheme.

As $\Delta\boldsymbol{\lambda}_{N+1}^{p}$ is determined fully from the result of the previous increment, some apparently paradoxical features are encountered. If the N-th increment involves plastic deformation, and $(\boldsymbol{\chi}_N - \boldsymbol{L}^T \Delta\boldsymbol{u}_{N+1}^{(i+1)})$ is inside the yield surface (unloading), $\Delta\boldsymbol{\lambda}_{N+1}^{(i+1)}$ may not be zero, although $\Delta\boldsymbol{\lambda}_{N+1}^{o\,(i+1)}$ will be zero. Further, for the specific unloading case illustrated in Figure 4.2b, $\Delta\boldsymbol{\lambda}_{N+1}^{o\,(i+1)}$ will not be zero and will cancel $\Delta\boldsymbol{\lambda}_{N+1}^{p}$ to ensure plastic consistency is enforced at T_{n+1}. This behaviour is consistent with the limiting case of the generalized trapezoidal rule applied to Norton creep[1].

Alternatively, the vector $(\boldsymbol{\chi}_N - \boldsymbol{L}^T \Delta\boldsymbol{u}_{N+1}^{(i+1)}) - \boldsymbol{H}\Delta\boldsymbol{\lambda}_{N+1}^{p}$ can be taken to represent the equivalent elastically predicted value of $\boldsymbol{\chi}_{N+1}^{(i+1)}$. The vector $-\boldsymbol{H}\Delta\boldsymbol{\lambda}_{N+1}^{o\,(i+1)}$ is then the equivalent backward difference return with $\Delta\boldsymbol{\lambda}_{N+1}^{o\,(i+1)}$ normal to the yield surface at $\boldsymbol{\chi}_{N+1}^{(i+1)}$. Recast in this form, the corrector step takes on the classical form of the backward difference return mapping algorithm of chapter 2.

4.3 THE GENERALIZED MIDPOINT RULE

The relation between the generalized midpoint rule of Simo *et al* [5,7] and the backward difference scheme is now examined. An internal variable formulation of the generalized midpoint rule is given by Reddy and Martin[4].

The time sequence t_n, t_{n+1} is now considered to be the prime sequence. The backward difference formulation given in chapter 3 provides $\boldsymbol{u}_{N+1}, \boldsymbol{\lambda}_{N+1}$ at time T_{n+1}, given $\boldsymbol{u}_n, \boldsymbol{\lambda}_n$ at time t_n. We choose to project forward linearly, as shown diagrammatically in Figure 4.3, to obtain $\boldsymbol{u}_{n+1}, \boldsymbol{\lambda}_{n+1}$.

Thus, we put

$$\frac{\boldsymbol{u}_{N+1} - \boldsymbol{u}_n}{\alpha\Delta t} = \frac{\boldsymbol{u}_{n+1} - \boldsymbol{u}_n}{\Delta t} \tag{4.20a}$$

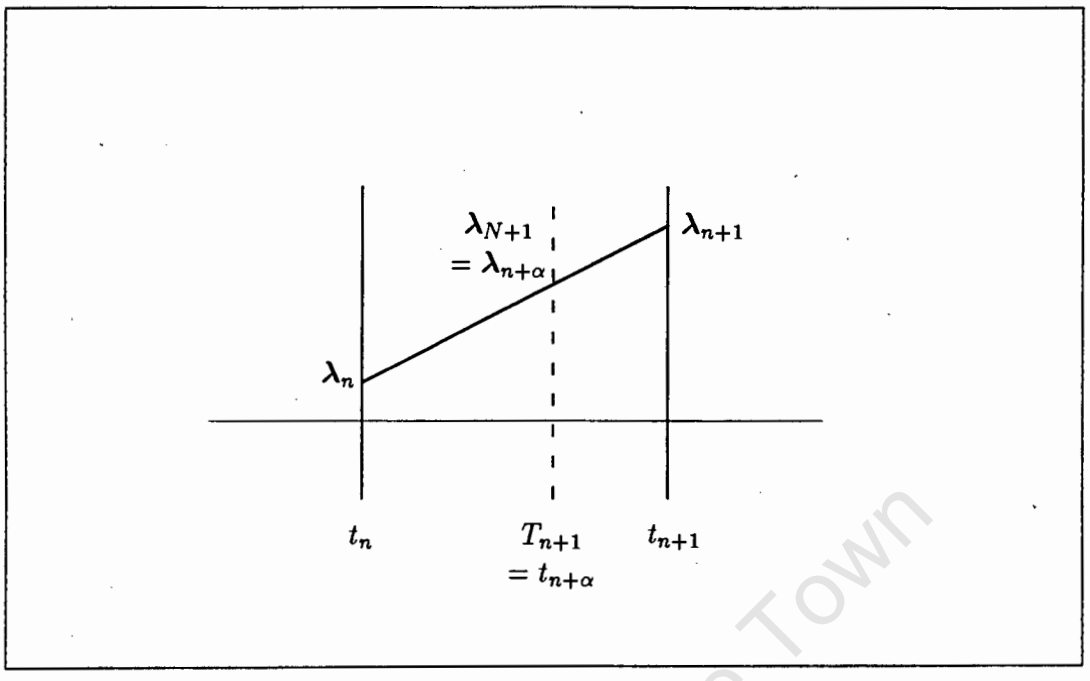

Figure 4.3: The generalized midpoint rule

or

$$\boldsymbol{u}_{n+1} = \boldsymbol{u}_n + \frac{1}{\alpha}(\boldsymbol{u}_{N+1} - \boldsymbol{u}_n) = \boldsymbol{u}_n + \frac{1}{\alpha}\Delta\tilde{\boldsymbol{u}}_{n+1} \quad . \tag{4.20b}$$

Similarly

$$\boldsymbol{\lambda}_{n+1} = \boldsymbol{\lambda}_n + \frac{1}{\alpha}(\boldsymbol{\lambda}_{N+1} - \boldsymbol{\lambda}_n) = \boldsymbol{\lambda}_n + \frac{1}{\alpha}\Delta\tilde{\boldsymbol{\lambda}}_{n+1} \quad . \tag{4.20c}$$

It follows also from eqns. (4.20) that

$$\boldsymbol{u}_{N+1} = (1-\alpha)\boldsymbol{u}_n + \alpha\boldsymbol{u}_{n+1} \quad , \tag{4.21a}$$

$$\boldsymbol{\lambda}_{N+1} = (1-\alpha)\boldsymbol{\lambda}_n + \alpha\boldsymbol{\lambda}_{n+1} \quad . \tag{4.21b}$$

This is recognizable as the form of the generalized midpoint rule, with the governing equations being satisfied at time

$$T_{n+1} = t_{n+\alpha} = t_n + \alpha\Delta t \quad . \tag{4.22}$$

It is instructive to rewrite the governing equations in terms of the variables

$$\Delta \boldsymbol{u}_{n+1} = \boldsymbol{u}_{n+1} - \boldsymbol{u}_n \quad , \tag{4.23a}$$

$$\Delta \boldsymbol{\lambda}_{n+1} = \boldsymbol{\lambda}_{n+1} - \boldsymbol{\lambda}_n \quad . \tag{4.23b}$$

It follows, on comparing eqns. (4.20) and (4.23), that

$$\Delta \tilde{\boldsymbol{u}}_{n+1} = \alpha \Delta \boldsymbol{u}_{n+1} \quad , \qquad \Delta \tilde{\boldsymbol{\lambda}}_{n+1} = \alpha \Delta \boldsymbol{\lambda}_{n+1} \quad . \tag{4.24}$$

In view of the fact that $\partial D / \partial \Delta \boldsymbol{\lambda}$ is homogeneous and of degree zero in the components of $\Delta \boldsymbol{\lambda}$, and that $\alpha > 0$,

$$\left.\frac{\partial D(\Delta \boldsymbol{\lambda})}{\partial \Delta \boldsymbol{\lambda}}\right|_{\Delta \tilde{\boldsymbol{\lambda}}_{n+1}} = \left.\frac{\partial D(\Delta \boldsymbol{\lambda})}{\partial \Delta \boldsymbol{\lambda}}\right|_{\alpha \Delta \boldsymbol{\lambda}_{n+1}} = \left.\frac{\partial D(\Delta \boldsymbol{\lambda})}{\partial \Delta \boldsymbol{\lambda}}\right|_{\Delta \boldsymbol{\lambda}_{n+1}} \quad . \tag{4.25}$$

Hence the governing equations (3.11) can be written as

$$\alpha \boldsymbol{K} \Delta \boldsymbol{u}_{n+1} + \alpha \boldsymbol{L} \Delta \boldsymbol{\lambda}_{n+1} = \boldsymbol{P}_{N+1} - (\boldsymbol{K} \boldsymbol{u}_n + \boldsymbol{L} \boldsymbol{\lambda}_n) \quad , \tag{4.26a}$$

$$\alpha \boldsymbol{L}^T \Delta \boldsymbol{u}_{n+1} + \alpha \boldsymbol{H} \Delta \boldsymbol{\lambda}_{n+1} + \left.\frac{\partial D}{\partial \Delta \boldsymbol{\lambda}}\right|_{\Delta \boldsymbol{\lambda}_{n+1}} = -(\boldsymbol{L}^T \boldsymbol{u}_n + \boldsymbol{H} \boldsymbol{\lambda}_n) \quad . \tag{4.26b}$$

If we define

$$\Delta \hat{\boldsymbol{u}}_{n+1} = \Delta \boldsymbol{u}_{n+1}^{(i+1)} - \Delta \boldsymbol{u}_{n+1}^{(i)} \quad , \tag{4.27a}$$

it follows that

$$\Delta \hat{\boldsymbol{u}} = \Delta \tilde{\boldsymbol{u}}_{n+1}^{(i+1)} - \Delta \tilde{\boldsymbol{u}}_{n+1}^{(i)} = \alpha \Delta \hat{u}_{n+1} \quad . \tag{4.27b}$$

The predictor equations (3.15a) then become

$$\alpha \boldsymbol{K}_c \Delta \hat{\boldsymbol{u}}_{n+1} = \tilde{\boldsymbol{R}}_{n+1}^{(i+1)} \quad , \tag{4.28}$$

where $\boldsymbol{K}_c$ and $\tilde{\boldsymbol{R}}_{n+1}^{(i+1)}$ are defined as in eqns. (3.15b) and (3.15c).

The corrector step eqn. (3.17) becomes

$$\alpha \boldsymbol{H} \Delta \boldsymbol{\lambda}_{n+1}^{(i+1)} + \left. \frac{\partial D}{\partial \Delta \boldsymbol{\lambda}} \right|_{\Delta \boldsymbol{\lambda}_{n+1}^{(i+1)}} = -\boldsymbol{L}^T (\boldsymbol{u}_n + \alpha \Delta \boldsymbol{u}_{n+1}^{(i+1)}) - \boldsymbol{H} \boldsymbol{\lambda}_n \quad . \tag{4.29}$$

We may thus formulate the problem directly in terms of the increments $\Delta \boldsymbol{u}_{n+1}, \Delta \boldsymbol{\lambda}_{n+1}$, and we retain the essential structure of the backward difference formulation, in the sense that the predictor modulus $\alpha \boldsymbol{K}_c$ is symmetric, and that the corrector step, or return algorithm, is applied in a similar way.

The comparison between the generalized midpoint rule and the backward difference formulation for the interval (t_n, t_{n+1}) can be better understood by recasting the equations slightly. First, if the time sequence t_n, t_{n+1} is the prime sequence, it would be consistent to assume that the data for the problem is given at these instants, so that we are given $\boldsymbol{P}_n, \boldsymbol{P}_{n+1}$. If this is so, we must interpolate to find $\boldsymbol{P}_{N+1}$. It is consistent to interpolate linearly, and hence we put

$$\boldsymbol{P}_{N+1} = (1-\alpha)\boldsymbol{P}_n + \alpha \boldsymbol{P}_{n+1} = \boldsymbol{P}_n + \alpha \Delta \boldsymbol{P}_{n+1} \tag{4.30a}$$

where

$$\Delta \boldsymbol{P}_{n+1} = \boldsymbol{P}_{n+1} - \boldsymbol{P}_n \quad . \tag{4.30b}$$

The equilibrium equation (4.26a) then becomes

$$\alpha \boldsymbol{K} \Delta \boldsymbol{u}_{n+1} + \alpha \boldsymbol{L} \Delta \boldsymbol{\lambda}_{n+1} = \alpha \Delta \boldsymbol{P}_{n+1} + (\boldsymbol{P}_n - \boldsymbol{K} \boldsymbol{u}_n - \boldsymbol{L} \boldsymbol{\lambda}_n) \quad . \tag{4.31}$$

If equilibrium is satisfied at the beginning of the step, the term in parentheses on the right hand side will be zero. This will not of course generally be so, since the iterative scheme is terminated with some residual, but acceptable, error. Nevertheless, if we assume that equilibrium is satisfied at time t_n, eqn. (4.31) reduces to

$$\boldsymbol{K} \Delta \boldsymbol{u}_{n+1} + \boldsymbol{L} \Delta \boldsymbol{\lambda}_{n+1} = \Delta \boldsymbol{P}_{n+1} \quad . \tag{4.32}$$

It then follows that

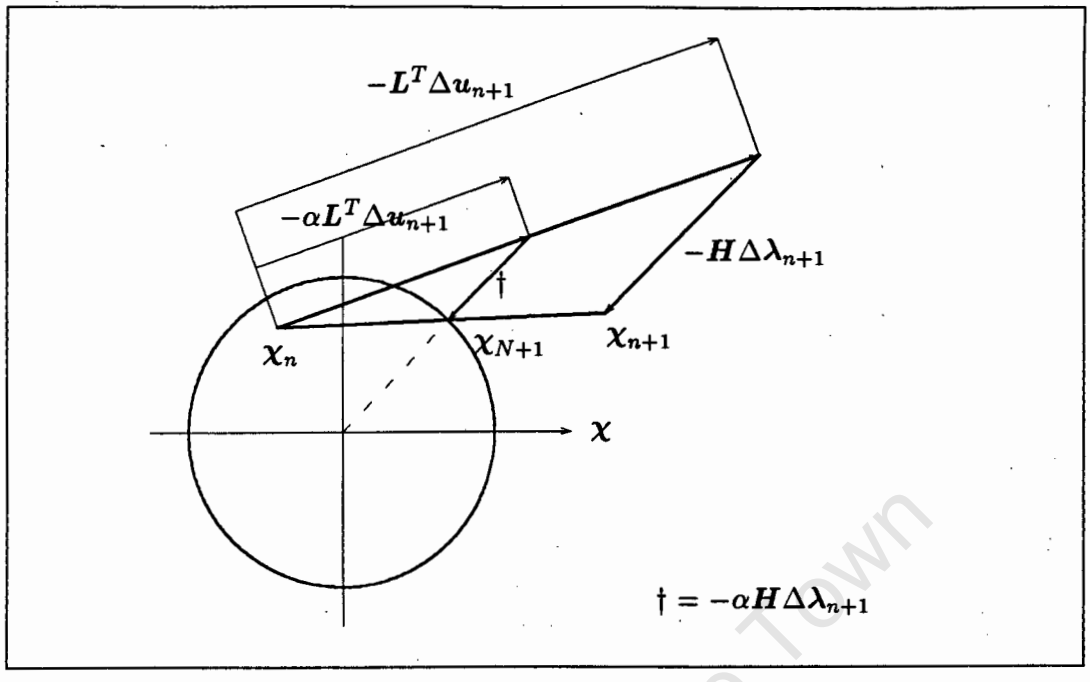

Figure 4.4: The corrector step for the generalized midpoint rule

$$\boldsymbol{K}\boldsymbol{u}_{n+1} + \boldsymbol{L}\boldsymbol{\lambda}_{n+1} = \boldsymbol{P}_{n+1} \quad , \tag{4.33}$$

i.e. equilibrium is satisfied at the end of the step.

Continuing with this assumption, the residual $\tilde{\boldsymbol{R}}_{n+1}^{(i+1)}$ (eqn. 3.15c) can be written as

$$\tilde{\boldsymbol{R}}_{n+1}^{(i+1)} = \alpha(\Delta\boldsymbol{P}_{n+1} - \boldsymbol{K}\Delta\boldsymbol{u}_{n+1}^{(i)} - \boldsymbol{L}\Delta\boldsymbol{\lambda}_{n+1}^{(i)}) + (\boldsymbol{P}_n - \boldsymbol{K}\boldsymbol{u}_n - \boldsymbol{L}\boldsymbol{\lambda}_n) = \alpha\boldsymbol{R}_{n+1}^{(i+1)} \tag{4.34a}$$

where

$$\boldsymbol{R}_{n+1}^{(i+1)} = \Delta\boldsymbol{P}_{n+1} - \boldsymbol{K}\Delta\boldsymbol{u}_{n+1}^{(i)} - \boldsymbol{L}\Delta\boldsymbol{\lambda}_{n+1}^{(i)} \quad . \tag{4.34b}$$

The predictor equations (4.28) then become

$$\boldsymbol{K}_c\Delta\hat{\boldsymbol{u}}_{n+1} = \boldsymbol{R}_{n+1}^{(i+1)} \quad . \tag{4.35}$$

Note that this equation does not depend on α, and is thus identical for the backward difference algorithm ($\alpha = 1$) and the generalized midpoint rule ($0 < \alpha \leq 1$).

The corrector step is shown diagrammatically as a return map in χ space in Figure 4.4. Eqn. (4.29) is written as

$$\chi_{N+1}^{(i+1)} = \chi_n - \alpha \boldsymbol{L}^T \Delta \boldsymbol{u}_{n+1}^{(i+1)} - \alpha \boldsymbol{H} \Delta \boldsymbol{\lambda}_{n+1}^{(i+1)} \tag{4.36a}$$

where

$$\chi_{N+1}^{(i+1)} = \left. \frac{\partial D}{\partial \Delta \boldsymbol{\lambda}} \right|_{\Delta \boldsymbol{\lambda}_{n+1}^{(i+1)}} \tag{4.36b}$$

The vector ($\chi_n - \alpha \boldsymbol{L}^T \Delta \boldsymbol{u}_{n+1}^{(i+1)}$) is an elastic value of $\chi_{N+1}^{(i+1)}$; if this vector lies within the yield surface, it provides $\chi_{N+1}^{(i+1)}$, and $\Delta \boldsymbol{\lambda}_{n+1}^{(i+1)}$ is zero. If it lies outside the yield surface, we must find $\chi_{N+1}^{(i+1)}$ on the yield surface and $\Delta \boldsymbol{\lambda}_{n+1}^{(i+1)}$ such that eqn. (4.36a) is satisfied and $\Delta \boldsymbol{\lambda}_{n+1}^{(i+1)}$ is normal to the yield surface at $\chi_{N+1}^{(i+1)}$. The diagram in Figure 4.4 is drawn assuming that $\boldsymbol{\lambda}$ and $\boldsymbol{H\lambda}$ have the same direction; this is not necessarily the case, but it simplifies the diagram. Then

$$\chi_{n+1}^{(i+1)} = \chi_n + \frac{1}{\alpha}(\chi_{N+1}^{(i+1)} - \chi_n) \tag{4.37}$$

It is clear from this diagram that when $\alpha = 1$ we recover the classical backward difference return algorithm. It is also clear that, for $\alpha \neq 1$, $\chi_{n+1}^{(i+1)}$ will not in general lie on the yield surface. It should also be noted that, for $\alpha \neq 1$, χ_n may lie within or beyond the yield surface and $\chi_{n+1}^{(i+1)}$ may thus lie within or beyond the yield surface.

The algorithm described in this section is identical to that of Simo and Govindjee[5].

4.4 THE RELATION BETWEEN THE GENERALIZED TRAPEZOIDAL RULE AND THE GENERALIZED MIDPOINT RULE

Again, the generalized midpoint rule is applied over the interval (t_n, t_{n+1}) and the generalized trapezoidal rule over the interval (T_n, T_{n+1}).

From a comparison between the two rules, illustrated in Figure (4.5), the following can be noted

$$\Delta \boldsymbol{\lambda}_{N+1}^{o} = \alpha \Delta \boldsymbol{\lambda}_{n+1} \qquad \boldsymbol{\lambda}_N + \Delta \boldsymbol{\lambda}_{N+1}^{p} = \boldsymbol{\lambda}_n \quad , \tag{4.38a}$$

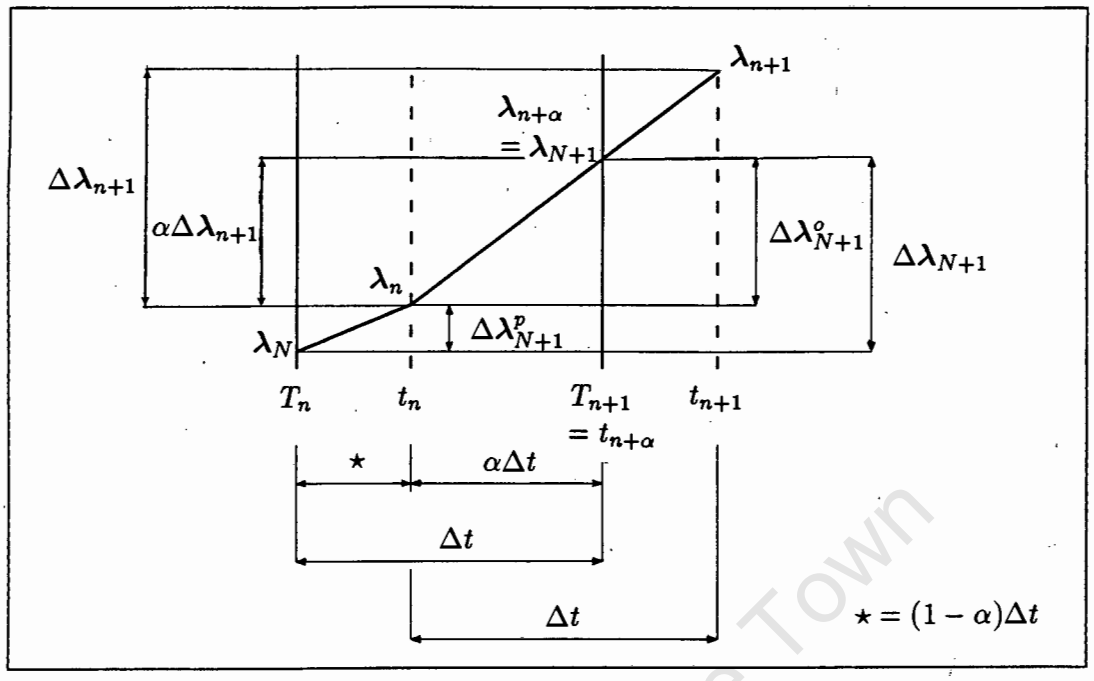

Figure 4.5: Comparison of the generalized rules

$$\Delta \boldsymbol{u}^o_{N+1} = \alpha \Delta \boldsymbol{u}_{n+1} \qquad \boldsymbol{u}_N + \Delta \boldsymbol{u}^p_{N+1} = \boldsymbol{u}_n \quad . \tag{4.38b}$$

$$\left.\frac{\partial D(\Delta\boldsymbol{\lambda}^o)}{\partial \Delta \boldsymbol{\lambda}^o}\right|_{\Delta\boldsymbol{\lambda}^o_{N+1}} = \left.\frac{\partial D(\Delta\boldsymbol{\lambda})}{\partial \Delta \boldsymbol{\lambda}}\right|_{\Delta\boldsymbol{\lambda}_{n+1}} \quad . \tag{4.38c}$$

Substituting the above into the governing equations (eqns. 4.8) for the generalized trapezoidal rule, we obtain

$$\alpha \boldsymbol{K} \Delta \boldsymbol{u}_{n+1} + \alpha \boldsymbol{L} \Delta \boldsymbol{\lambda}_{n+1} = \alpha \Delta \boldsymbol{P}_{n+1} + (\boldsymbol{P}_n - \boldsymbol{K}\boldsymbol{u}_n - \boldsymbol{L}\boldsymbol{\lambda}_n) \quad , \tag{4.39a}$$

$$\alpha \boldsymbol{L}^T \Delta \boldsymbol{u}_{n+1} + \alpha \boldsymbol{H} \Delta \boldsymbol{\lambda}_{n+1} + \left.\frac{\partial D}{\partial \Delta \boldsymbol{\lambda}}\right|_{\Delta\boldsymbol{\lambda}_{n+1}} = -(\boldsymbol{L}^T \boldsymbol{u}_n + \boldsymbol{H}\boldsymbol{\lambda}_n) \quad . \tag{4.39b}$$

Eqns. (4.39) are recognizable as the governing equations for the generalized midpoint rule (eqns. 4.31 and 4.26b). Thus the generalized trapezoidal rule and the generalized midpoint rule lead to the same minimum principle for the increment and in this sense are equivalent.

As the prime sequence for the generalized midpoint rule is t_n, t_{n+1}, it is consistent to assume that the data for the problem is given at these instants. Even though equilibrium is satisfied at these instants, both equilibrium and the constitutive equations

are only satisfied at the generalized midpoint $t_{n+\alpha} = T_{n+1}$. It is thus meaningful to report the results $u_{N+1}, \lambda_{N+1}, \chi_{N+1}$ at the generalized midpoint[5].

In contrast, for the generalized trapezoidal rule, both the equilibrium and constitutive equations are fully satisfied at the end of each interval, where the data for the problem is given. In this sense, it can be argued that the generalized trapezoidal rule is more convenient.

4.5 CONCLUSIONS

First, as the preservation of the symmetry of the consistent tangent modulus is an important criterion, the formulation of the generalized trapezoidal rule should be done on the basis given in this chapter.

Second, the generalized midpoint rule of Simo *et al* [5,7] and the generalized trapezoidal rule as defined in this chapter are fully equivalent. The generalized trapezoidal rule thus inherits the notion of B-stability associated with the generalized midpoint rule. It could also be argued that the generalized trapezoidal rule is more convenient, in the sense that the equilibrium and constitutive equations are fully satisfied at the end of each interval.

Third, the symmetry of the consistent tangent modulus is associated with the backward difference formulation over part of the interval in each case. Thus the generalized midpoint rule and the generalized trapezoidal rule should be regarded as equivalent generalizations of the backward difference formulation, in which the essential advantageous characteristics of the backward difference formulation are retained.

Fourth, both the generalized midpoint rule and the generalized trapezoidal rule offer the opportunity to exploit the second order rate of convergence for $\alpha = \frac{1}{2}$ demonstrated by Simo and Govindjee[5], as opposed to the first order rate of convergence associated with the standard backward difference formulation.

REFERENCES

1. Marais, N.J., Time Integration Algorithms for Finite Element Analysis of Creep Problems, PhD Thesis, University of Cape Town, 1989.

2. Ortiz, M. & Martin, J.B., Symmetry-Preserving Return Mapping Algorithms and Incrementally Extremal Paths: a Unification of Concepts, *International Journal for Numerical Methods in Engineering*, **28**, 1839-1853, 1989.

3. 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.

4. 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.

5. Simo, J.C. & Govindjee, S., Nonlinear B-Stability and Symmetry Preserving Return Mapping Algorithms for Plasticity and Viscoplasticity, *International Journal for Numerical Methods in Engineering*, **31**, 151-176, 1991.

6. Simo, J.C. & Taylor, R.L., Consistent Tangent Operators for Rate-Independent Elastoplasticity, *Computer Methods in Applied Mechanics and Engineering*, **48**, 101-118, 1985.

7. Simo, J.C. & Taylor, R.L., A Return Mapping Algorithm for Plane Stress Elastoplasticity, *International Journal for Numerical Methods in Engineering*, **22**, 649-670, 1986.

CHAPTER 5
FORMULATION OF THE CORRECTOR STEP

5.1 INTRODUCTION

As discussed in chapter 2, the corrector step has traditionally been formulated in the form of a return mapping algorithm. This essentially heuristic approach has been extended to piecewise linear plasticity, together with an increasing acceptance of the backward difference scheme for the integration of the constitutive equations[1,2,3,8,9]. The question is thus to formulate a corrector step for piecewise linear plasticity that is closely linked to the governing principles and utilizes the computational advantages (second order accuracy and B-stability) associated with the generalized time integration rules of chapter 4.

In this chapter, an internal variable formulation of a backward difference algorithm for the corrector step is considered. The backward difference integration scheme is chosen as it assures that a minimum principle can be written[5,7], and it is readily extended to both generalized rules of chapter 4. The resulting algorithm is written in the form of a mathematical programming problem and is fully consistent, in that no heuristic assumptions need to be made.

Suitable generalization to include linear isotropic hardening is introduced and the algorithm is first specialized to encompass piecewise linear plasticity and then specialized for the Tresca and Mohr-Coulomb yield surfaces with linear hardening (examples in chapter 6). The algorithm is also extended to include the generalized trapezoidal rule in such a way that the overall structure of the backward difference algorithm is maintained. This allows us to utilize the computational advantages of the generalized trapezoidal rule.

5.2 FORMULATION OF THE CORRECTOR ALGORITHM FOR A GENERAL YIELD SURFACE

The internal variable formulation of chapter 3 is developed at structural level. For the corrector step, it is now necessary to restate the formulation at Gauss point level. The kinematic variables are thus strain components ϵ and internal variables λ (representing slips within the continuum) defined at an individual Gauss point.

The specific free energy f for an isothermal element is assumed to be a homogeneous quadratic function of ϵ and λ. We thus write

$$f = \tfrac{1}{2}\epsilon^T \boldsymbol{C}\epsilon + \tfrac{1}{2}\epsilon^T(\boldsymbol{E} + \boldsymbol{E}^T)\boldsymbol{\lambda} + \tfrac{1}{2}\boldsymbol{\lambda}^T \boldsymbol{H}\boldsymbol{\lambda} \ . \tag{5.1}$$

In general the matrix $\boldsymbol{C}$ is positive definite, while $\boldsymbol{H}$ is positive semi-definite[6]. The equations of state give the stresses σ and the internal forces χ at the Gauss point:

$$\sigma = \frac{\partial f}{\partial \epsilon} = \boldsymbol{C}\epsilon + \boldsymbol{E}\boldsymbol{\lambda} \ , \tag{5.2a}$$

$$-\chi = \frac{\partial f}{\partial \boldsymbol{\lambda}} = \boldsymbol{E}^T\epsilon + \boldsymbol{H}\boldsymbol{\lambda} \ . \tag{5.2b}$$

The minus sign is introduced into eqn. (5.2b) to define χ as the forces applied by the elastic continuum to the slips rather than the reverse.

A convex, non-negative dissipation function $D(\dot{\boldsymbol{\lambda}})$ which is homogeneous and of degree one in the components of $\dot{\boldsymbol{\lambda}}$ is introduced. Again, the dissipation function is assumed to be zero if and only if $\dot{\boldsymbol{\lambda}} = 0$ and to be positive for $\dot{\boldsymbol{\lambda}} \neq 0$. The dissipation function is thus a generalized cone in a $D - \dot{\boldsymbol{\lambda}}$ space, formed by radial generators. The derivatives of D will be discontinuous at the origin, and may be discontinuous along radial lines in the $\dot{\boldsymbol{\lambda}}$ space. At such points of discontinuity, $\partial D/\partial \dot{\boldsymbol{\lambda}}$ is taken to represent any value within the fan defined by adjacent values. The dissipation function is defined at the individual Gauss point, unlike the global dissipation function of chapter 3, which is a sum of contributions from individual points.

The internal forces χ and the slip rates $\dot{\boldsymbol{\lambda}}$ are related through a kinetic equation:

$$\chi = \frac{\partial D}{\partial \dot{\boldsymbol{\lambda}}} \ . \tag{5.3}$$

Combining eqns. (5.2) and (5.3), leads to the constitutive equations for a class of elastic-plastic solids:

$$\boldsymbol{C}\epsilon + \boldsymbol{E}\boldsymbol{\lambda} = \sigma \ , \tag{5.4a}$$

$$\boldsymbol{E}^T\epsilon + \boldsymbol{H}\boldsymbol{\lambda} + \frac{\partial D}{\partial \dot{\boldsymbol{\lambda}}} = 0 \ . \tag{5.4b}$$

It is pertinent to note that eqn. (5.4b) corresponds to one equation in the sequence of equations defined by eqn. (3.4b).

Utilizing the time discretization introduced in chapter 3, the time sequence t_n, T_{n+1} is again considered to be the prime sequence. The constitutive equations (eqns. 5.4) must be satisfied at time T_{n+1}. Thus

$$\boldsymbol{C}\boldsymbol{\epsilon}_{N+1} + \boldsymbol{E}\boldsymbol{\lambda}_{N+1} = \boldsymbol{\sigma}_{N+1} \quad , \tag{5.5a}$$

$$\boldsymbol{E}^T\boldsymbol{\epsilon}_{N+1} + \boldsymbol{H}\boldsymbol{\lambda}_{N+1} + \left.\frac{\partial D}{\partial \dot{\boldsymbol{\lambda}}}\right|_{\dot{\boldsymbol{\lambda}}_{N+1}} = 0 \quad . \tag{5.5b}$$

We introduce the increments

$$\Delta\tilde{\boldsymbol{\epsilon}}_{n+1} = \boldsymbol{\epsilon}_{N+1} - \boldsymbol{\epsilon}_n \quad , \tag{5.6}$$

$$\Delta\tilde{\boldsymbol{\lambda}}_{n+1} = \boldsymbol{\lambda}_{N+1} - \boldsymbol{\lambda}_n \quad , \tag{5.7}$$

and apply a backward difference assumption over the interval (t_n, T_{n+1}). As D is homogeneous and of degree one in $\dot{\boldsymbol{\lambda}}$, we can put

$$\left.\frac{\partial D}{\partial \dot{\boldsymbol{\lambda}}}\right|_{\dot{\boldsymbol{\lambda}}_{N+1}} = \left.\frac{\partial D(\dot{\boldsymbol{\lambda}})}{\partial \dot{\boldsymbol{\lambda}}}\right|_{\Delta\tilde{\boldsymbol{\lambda}}_{n+1}} = \left.\frac{\partial D(\Delta\boldsymbol{\lambda})}{\partial \Delta\boldsymbol{\lambda}}\right|_{\Delta\tilde{\boldsymbol{\lambda}}_{n+1}} \quad . \tag{5.8}$$

Eqns. (5.5) can then be rewritten as

$$\boldsymbol{C}\Delta\tilde{\boldsymbol{\epsilon}}_{n+1} + \boldsymbol{E}\Delta\tilde{\boldsymbol{\lambda}}_{n+1} = \boldsymbol{\sigma}_{N+1} - (\boldsymbol{C}\boldsymbol{\epsilon}_n + \boldsymbol{E}\boldsymbol{\lambda}_n) \quad , \tag{5.9a}$$

$$\boldsymbol{E}^T\Delta\tilde{\boldsymbol{\epsilon}}_{n+1} + \boldsymbol{H}\Delta\tilde{\boldsymbol{\lambda}}_{n+1} + \left.\frac{\partial D}{\partial \Delta\boldsymbol{\lambda}}\right|_{\Delta\tilde{\boldsymbol{\lambda}}_{n+1}} = \boldsymbol{\chi}_n \quad . \tag{5.9b}$$

Eqns. (5.9) are the incremental constitutive equations for a class of elastic-plastic solids that encompasses perfectly plastic and linear kinematic hardening solids. In order to include isotropic hardening it is necessary to constrain the signs of the components of $\dot{\boldsymbol{\lambda}}$. Alternatively, a simpler modification can be effected by introducing into the expression for f, given by eqn. (5.1), a term representing dissipated work stored elastically but irrecoverably in the solid[6]. For the case of linear isotropic

hardening, the term $H\gamma^2/2$ is added to f, where H is a scalar constant and γ is a scalar value given by

$$\gamma = \int_0^t D(\dot{\boldsymbol{\lambda}})dt \ . \tag{5.10}$$

In view of the backward difference assumption, we note that we can write, for $t_n < t \leq T_{n+1}$,

$$\gamma_{N+1} = \gamma_n + D(\Delta\tilde{\boldsymbol{\lambda}}_{n+1}) \ , \tag{5.11}$$

and hence

$$\frac{\partial\gamma}{\partial\boldsymbol{\lambda}} = \lim_{\Delta t\to 0}\frac{\Delta\gamma}{\Delta\boldsymbol{\lambda}} = \frac{\partial D}{\partial\Delta\boldsymbol{\lambda}} \ . \tag{5.12}$$

Thus, again following through the steps of the previous argument, eqns. (5.9) become

$$\boldsymbol{C}\Delta\tilde{\boldsymbol{\epsilon}}_{n+1} + \boldsymbol{E}\Delta\tilde{\boldsymbol{\lambda}}_{n+1} = \boldsymbol{\sigma}_{N+1} - (\boldsymbol{C}\boldsymbol{\epsilon}_n + \boldsymbol{E}\boldsymbol{\lambda}_n) \ , \tag{5.13a}$$

$$\begin{aligned}\boldsymbol{E}^T\Delta\tilde{\boldsymbol{\epsilon}}_{n+1} + \boldsymbol{H}\Delta\tilde{\boldsymbol{\lambda}}_{n+1} + \frac{\partial}{\partial\Delta\boldsymbol{\lambda}}[(1+H\gamma_n)D(\Delta\boldsymbol{\lambda}) \\ + \tfrac{1}{2}HD^2(\Delta\boldsymbol{\lambda})]\Big|_{\Delta\tilde{\boldsymbol{\lambda}}_{n+1}} = -(\boldsymbol{E}^T\boldsymbol{\epsilon}_n + \boldsymbol{H}\boldsymbol{\lambda}_n) \ .\end{aligned} \tag{5.13b}$$

With suitable choices of the constants $\boldsymbol{C}, \boldsymbol{E}, \boldsymbol{H}$ and H, this model can represent perfectly plastic, linear kinematic and linear isotropic hardening.

Implementing these equations as the corrector step, we note that $\boldsymbol{\epsilon}_n, \boldsymbol{\lambda}_n$ and γ_n are known, and $\Delta\tilde{\boldsymbol{\epsilon}}_{n+1}$ is given. We see then that the problem can be uncoupled. First, eqn. (5.13b) can be solved for $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}$, and then $\boldsymbol{\sigma}_{N+1}$ recovered from eqn. (5.13a), which can be written as

$$\boldsymbol{\sigma}_{N+1} = \boldsymbol{C}(\boldsymbol{\epsilon}_n + \Delta\tilde{\boldsymbol{\epsilon}}_{n+1}) + \boldsymbol{E}(\boldsymbol{\lambda}_n + \Delta\tilde{\boldsymbol{\lambda}}_{n+1}) \ . \tag{5.14}$$

The solution of eqn. (5.13b) can be written as a convex, non-linear programming

problem. Without giving a rigorous argument, we can define a function $U_{N+1}(\Delta\boldsymbol{\lambda})$ whose least value is given by $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}$, where

$$U_{N+1}(\Delta\boldsymbol{\lambda}) = (\Delta\tilde{\boldsymbol{\epsilon}}_{n+1}^T\boldsymbol{E}\Delta\boldsymbol{\lambda} + \boldsymbol{\epsilon}_n^T\boldsymbol{E}\Delta\boldsymbol{\lambda} + \boldsymbol{\lambda}_n^T\boldsymbol{H}^T\Delta\boldsymbol{\lambda}) + \tfrac{1}{2}\Delta\boldsymbol{\lambda}^T\boldsymbol{H}\Delta\boldsymbol{\lambda} + (1 + H\gamma_n)D(\Delta\boldsymbol{\lambda}) + \tfrac{1}{2}HD^2(\Delta\boldsymbol{\lambda}) \quad . \tag{5.15}$$

5.3 SPECIALIZATION TO A PIECEWISE LINEAR YIELD SURFACE

A formal relation can be established between the yield surface in $\boldsymbol{\chi}$ space and a level surface of the dissipation function D in $\dot{\boldsymbol{\lambda}}$ space[4,6]. In view of the backward difference assumption and eqn. (5.3), D may be written in terms of the increments $\Delta\boldsymbol{\lambda}$, and

$$\boldsymbol{\chi} = \frac{\partial D(\Delta\boldsymbol{\lambda})}{\partial\Delta\boldsymbol{\lambda}} \quad . \tag{5.16}$$

A piecewise linear yield surface is associated with a dissipation function with piecewise linear level surfaces, as shown diagrammatically in Figure 5.1. The relationship between these two surfaces is a relatively straightforward one. A flat region on the yield surface, identified by a normal vector $\boldsymbol{n}$, becomes a vertex on a level surface of dissipation lying on the radial line defined by $\boldsymbol{n}$. Similarly, a vertex on the yield surface will become a flat region on a level surface of the dissipation function.

We choose to represent the increment $\Delta\boldsymbol{\lambda}$ in terms of the set of unit vectors $\boldsymbol{n}_1, \boldsymbol{n}_2, ..., \boldsymbol{n}_k$ which identify each of the radial lines in $\Delta\boldsymbol{\lambda}$ space on which vertices of the level surfaces of the dissipation function lie. We then write

$$\Delta\boldsymbol{\lambda} = \Delta\Lambda_1\boldsymbol{n}_1 + \Delta\Lambda_2\boldsymbol{n}_2 + ... + \Delta\Lambda_k\boldsymbol{n}_k = \boldsymbol{N}^T\Delta\boldsymbol{\Lambda} \quad , \tag{5.17a}$$

with

$$\Delta\Lambda_i \geq 0 \ , \ i = 1, ..., k \quad , \tag{5.17b}$$

and where $\Delta\boldsymbol{\Lambda}$ is the vector with components $\Delta\Lambda_i$, and $\boldsymbol{N}^T$ is the matrix whose columns are the vectors $\boldsymbol{n}_i$.

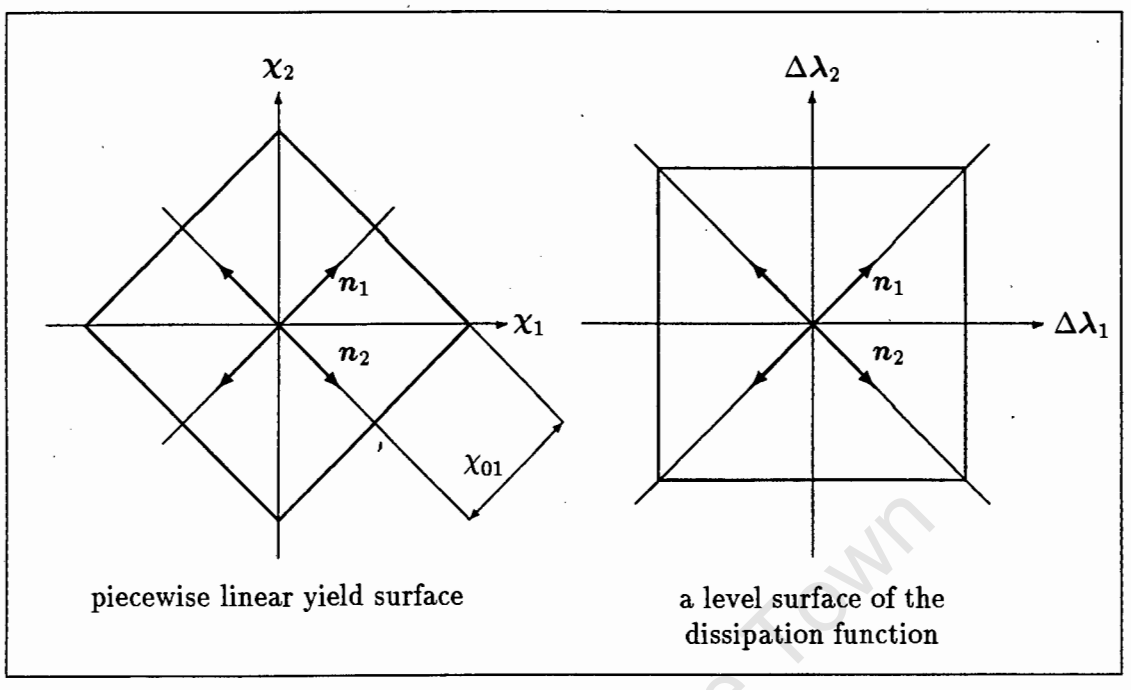

Figure 5.1: A level surface of the dissipation function associated with a piecewise linear yield function

Along the radial line defined by $\boldsymbol{n}_i$ in $\Delta\boldsymbol{\lambda}$ space, the value of the dissipation function is given by

$$D = \chi_{0i}\Delta\Lambda_i \quad , \quad (no\ summation) \tag{5.18}$$

as D is homogeneous and of degree one and $\Delta\Lambda_i$ is non-negative. The coefficient χ_{0i} can be identified as the shortest distance between the plane on the yield surface with normal $\boldsymbol{n}_i$ and the origin in $\boldsymbol{\chi}$ space (Figure 5.1).

The value of D associated with any given $\Delta\boldsymbol{\lambda}$ can be expressed as a linear programming problem[7] ;

$$D(\Delta\boldsymbol{\lambda}) = min(\ \boldsymbol{\chi}_0^T\Delta\boldsymbol{\Lambda}\ :\ \boldsymbol{N}^T\Delta\boldsymbol{\Lambda} = \Delta\boldsymbol{\lambda}\ ,\ \Delta\boldsymbol{\Lambda} \geq 0\ )\ , \tag{5.19}$$

where $\boldsymbol{\chi}_0$ is the vector with components χ_{0i}.

This linear programming problem may now be embedded into the convex non-linear programming problem of eqn. (5.15)

We thus seek

$$min(\ U_{N+1}(\Delta\boldsymbol{\Lambda}) : \Delta\boldsymbol{\Lambda} \geq 0\ ) \tag{5.20a}$$

where

$$\begin{aligned} U_{N+1}(\Delta\boldsymbol{\Lambda}) &= \tfrac{1}{2}\Delta\boldsymbol{\Lambda}^T\boldsymbol{N}\boldsymbol{H}\boldsymbol{N}^T\Delta\boldsymbol{\Lambda} + \Delta\tilde{\boldsymbol{\epsilon}}_{n+1}^T\boldsymbol{E}\boldsymbol{N}^T\Delta\boldsymbol{\Lambda} + (1+H\gamma_n)\boldsymbol{\chi}_0^T\Delta\boldsymbol{\Lambda} \\ &\quad + \tfrac{1}{2}H\Delta\boldsymbol{\Lambda}^T\boldsymbol{\chi}_0\boldsymbol{\chi}_0^T\Delta\boldsymbol{\Lambda} + \boldsymbol{\epsilon}_n^T\boldsymbol{E}\boldsymbol{N}^T\Delta\boldsymbol{\Lambda} + \boldsymbol{\lambda}_n^T\boldsymbol{H}^T\boldsymbol{N}^T\Delta\boldsymbol{\Lambda} \\ &= [(\Delta\tilde{\boldsymbol{\epsilon}}_{n+1}^T\boldsymbol{E} + \boldsymbol{\epsilon}_n^T\boldsymbol{E} + \boldsymbol{\lambda}_n^T\boldsymbol{H}^T)\boldsymbol{N}^T + (1+H\gamma_n)\boldsymbol{\chi}_0^T]\Delta\boldsymbol{\Lambda} \\ &\quad + \tfrac{1}{2}\Delta\boldsymbol{\Lambda}^T[\boldsymbol{N}\boldsymbol{H}\boldsymbol{N}^T + H\boldsymbol{\chi}_0\boldsymbol{\chi}_0^T]\Delta\boldsymbol{\Lambda} \end{aligned} \tag{5.20b}$$

This may be recognized as a convex quadratic programming problem in $\Delta\boldsymbol{\Lambda}$. We denote its solution by $\Delta\tilde{\boldsymbol{\Lambda}}_{n+1}$, and recover $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}$ from the relation

$$\Delta\tilde{\boldsymbol{\lambda}}_{n+1} = \boldsymbol{N}^T\Delta\tilde{\boldsymbol{\Lambda}}_{n+1}\ . \tag{5.21}$$

All the information needed to solve the corrector step as a quadratic programming problem is contained in the dissipation function and the free energy function. The dissipation function contains information about the yield surface while the energy function contains information on the hardening model.

5.4 SPECIALIZATION TO THE TRESCA AND MOHR-COULOMB YIELD SURFACES

5.4.1 A framework for the Tresca and Mohr-Coulomb yield surfaces

For both the Tresca and Mohr-Coulomb yield surfaces, the internal variable increment $\Delta\boldsymbol{\lambda}$ can be construed as the plastic strain increment $\Delta\boldsymbol{\epsilon}^p$, and can equally be expressed in the form of the symmetric cartesian tensor of order 2, $\Delta\lambda_{ij}$.

$$\Delta\boldsymbol{\lambda} = \begin{bmatrix} \Delta\lambda_{11} \\ \Delta\lambda_{22} \\ \Delta\lambda_{33} \\ \Delta\lambda_{12} \\ \Delta\lambda_{13} \\ \Delta\lambda_{23} \end{bmatrix} \qquad \Delta\lambda_{ij} = \begin{bmatrix} \Delta\lambda_{11} & \Delta\lambda_{12} & \Delta\lambda_{13} \\ \Delta\lambda_{12} & \Delta\lambda_{22} & \Delta\lambda_{23} \\ \Delta\lambda_{13} & \Delta\lambda_{23} & \Delta\lambda_{33} \end{bmatrix} . \tag{5.22}$$

Similarly the principal tensor $\Delta\lambda^*_{ij}$ can be expressed in the form of the principal vector $\Delta\boldsymbol{\lambda}^*$.

$$\Delta\boldsymbol{\lambda}^* = \begin{bmatrix} \Delta\lambda_1 \\ \Delta\lambda_2 \\ \Delta\lambda_3 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \Delta\lambda_1 \\ \Delta\lambda_2 \\ \Delta\lambda_3 \end{bmatrix} \quad \Delta\lambda^*_{ij} = \begin{bmatrix} \Delta\lambda_1 & 0 & 0 \\ 0 & \Delta\lambda_2 & 0 \\ 0 & 0 & \Delta\lambda_3 \end{bmatrix} \tag{5.23}$$

where $\Delta\lambda_1$, $\Delta\lambda_2$, $\Delta\lambda_3$ are the principal components of the internal variable increment. The same applies to the conjugate force vector $\boldsymbol{\chi}$.

The Tresca and Mohr-Coulomb yield surfaces, $\phi(\boldsymbol{\chi})$, are only piecewise linear in principal conjugate force space. Thus, for the time interval (t_n, T_{n+1}), the quadratic programming problem of eqn. (5.20) needs to be solved in principal $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}$ space, rather than in the reference space in which the problem is initially stated. It is therefore necessary to establish a framework that facilitates a transformation between these spaces.

To define a transformation between reference and principal $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}$ space, we need to compute the principal directions of $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}$ *a priori* (i.e. at time t_n). For a backward difference scheme, the increment $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}$ is evaluated at time T_{n+1} and is a scalar multiple of $\nabla\phi_{N+1}$. As ϕ_{N+1}, and hence $\nabla\phi_{N+1}$, can be described in terms of the principal basis of $\boldsymbol{\chi}_{N+1}$, the principal directions of $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}$ and $\boldsymbol{\chi}_{N+1}$ correspond. Further, the tensor $\boldsymbol{H}$ is isotropic and for a perfectly plastic or linear kinematic hardening rule, eqn. (5.9b) can be written as

$$-(\Delta\tilde{\boldsymbol{\epsilon}}^T_{n+1}\boldsymbol{E} + \boldsymbol{\epsilon}^T_n\boldsymbol{E} + \boldsymbol{\lambda}^T_n\boldsymbol{H}^T) = (\boldsymbol{\chi}_{N+1} + \boldsymbol{H}\Delta\tilde{\boldsymbol{\lambda}}_{n+1}) \ . \tag{5.24}$$

Thus both terms in parentheses in eqn. (5.24) have principal directions corresponding to those of $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}$ and, by determining the principal directions of the left-hand term, we can establish a transformation between reference and principal $\Delta\tilde{\boldsymbol{\lambda}}_{n+1}$ space *a priori*. This is equivalent to first formulating the problem at each iteration as a holonomic problem in $\boldsymbol{\chi}$ space.

Similarly, for a linear isotropic hardening rule, eqn. (5.13b) can be written as

$$-(\Delta\tilde{\epsilon}_{n+1}^{T}\boldsymbol{E} + \epsilon_{n}^{T}\boldsymbol{E} + \lambda_{n}^{T}\boldsymbol{H}^{T}) = (\chi_{N+1} + \boldsymbol{H}\Delta\lambda_{N+1} + \frac{\partial}{\partial\Delta\lambda}[H\gamma_{n}D(\Delta\lambda) + \tfrac{1}{2}HD^{2}(\Delta\lambda)]\Big|_{\Delta\tilde{\lambda}_{n+1}}) \quad . \tag{5.25}$$

As before, both terms in parentheses in eqn. (5.25) have principal directions corresponding to those of $\Delta\tilde{\lambda}_{n+1}$ and the transformation can be established *a priori*.

A framework can now been established where, while the corrector step at each iteration of each timestep is formulated in reference (6-dimensional) space, the associated convex quadratic programming problem is solved in principal (3-dimensional) space. The corrector algorithm for the Tresca and Mohr-Coulomb yield surfaces is obtained by embedding the appropriate quadratic programming problem in this framework, which is illustrated in Figure 5.2.

5.4.2 The Tresca plane stress case

To validate the transformation process, it is necessary to ensure that a normal return in principal χ space corresponds to a normal return in reference χ space. The plane stress case allows for a geometric validation of this sort. The Tresca yield surface is constructed in $\sigma_{11} - \sigma_{22} - \sigma_{12}$ space, which is equivalent to the reference χ space for an elastic perfectly-plastic material. The subspace $\sigma_{11} - \sigma_{22}$ then represents the principal χ space and is shown in Figure 5.3.

The transformation for the plane stress case is a purely rotational transformation about the major axis of the yield surface. Also, each cross-section of the yield surface normal to the major axis can be represented by an appropriate Mohr circle of stress, as shown in Figure 5.3. Thus, as normals to the yield surface lie on radial lines in the Mohr circle representation, a purely rotational transformation ensures that a correspondence exists between normal returns in principal space and reference space.

5.5 EXTENSION TO THE GENERALIZED TRAPEZOIDAL RULE

The corrector step is now extended to incorporate the generalized trapezoidal rule, while maintaining the general structure of the backward difference algorithm. As discussed in chapter 4, this is possible as the trapezoidal rule applied over the interval (T_n, T_{n+1}) is equivalent to a forward projection from T_n to t_n and the application of a backward difference scheme over the interval (t_n, T_{n+1}).

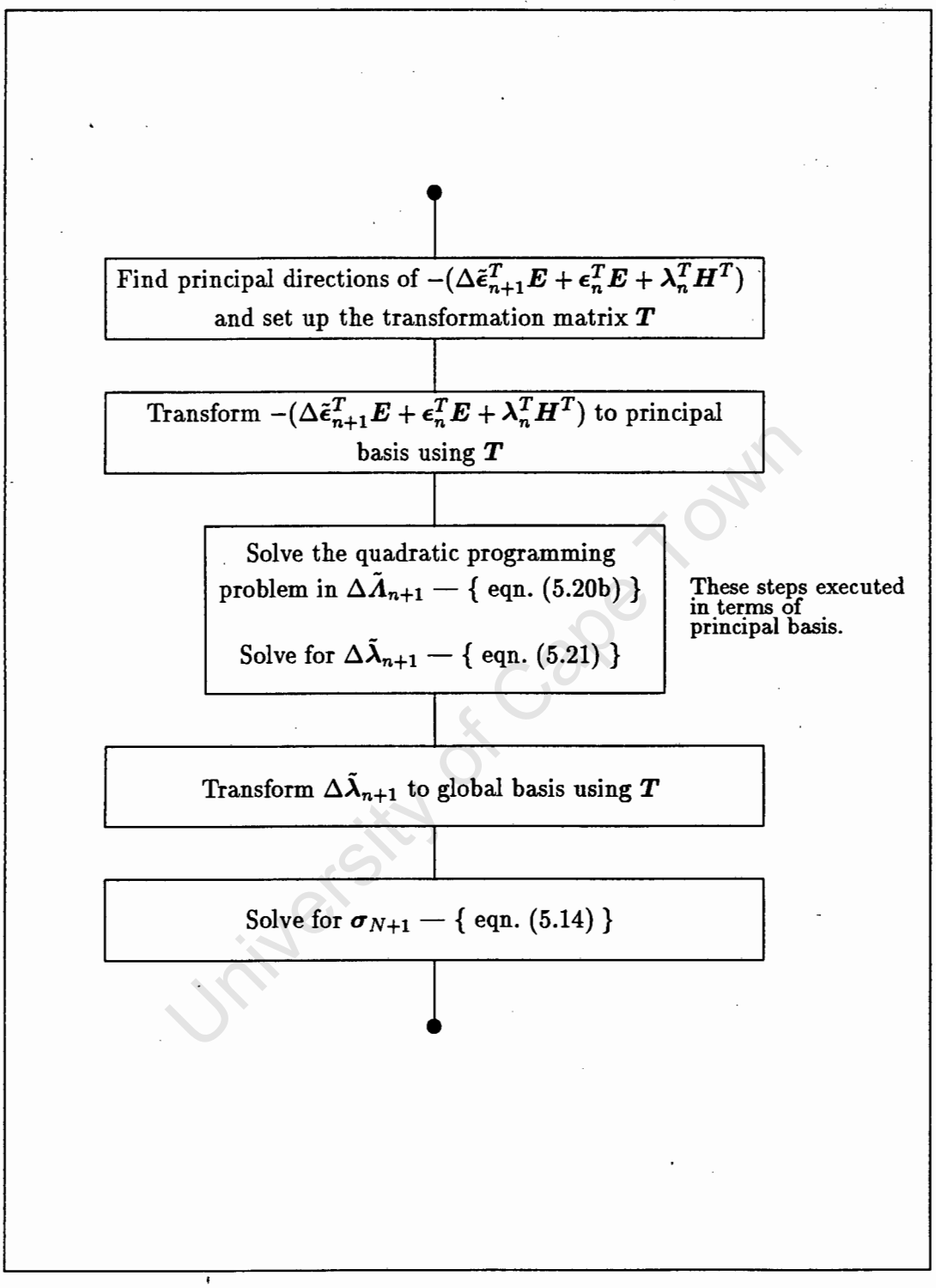

Figure 5.2: Framework for the Tresca and Mohr-Coulomb yield surfaces (backward difference)

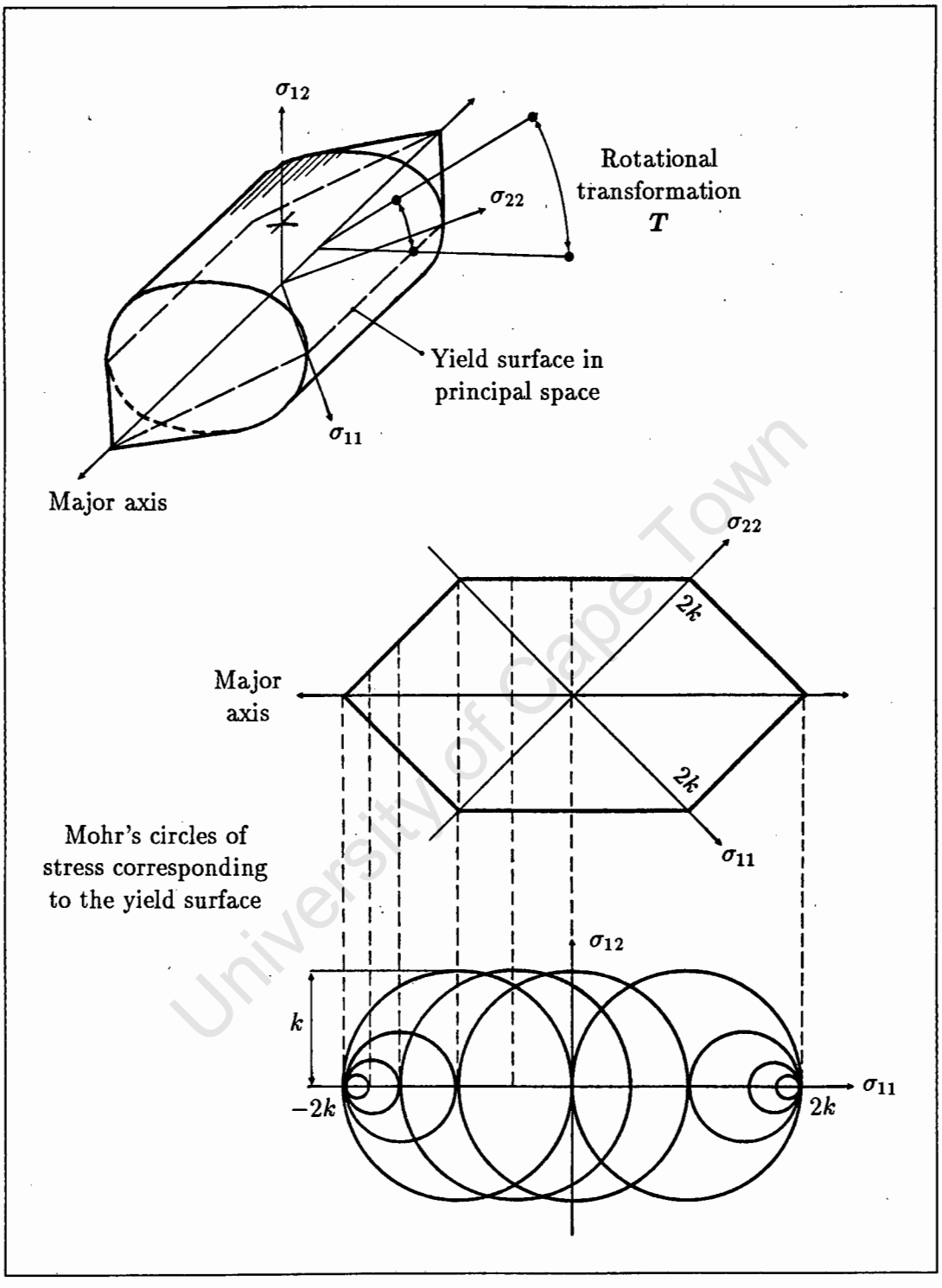

Figure 5.3: The Tresca plane stress case

We now consider the sequence T_n, T_{n+1} to be the prime sequence and apply the trapezoidal rule over the interval (T_n, T_{n+1}). The governing equations (eqns. 4.8) are written at Gauss point level as:

$$\boldsymbol{C}\Delta\boldsymbol{\epsilon}_{N+1} + \boldsymbol{E}\Delta\boldsymbol{\lambda}^{\circ}_{N+1} = \boldsymbol{\sigma}_{N+1} - (\boldsymbol{C}\boldsymbol{\epsilon}_N + \boldsymbol{E}\boldsymbol{\lambda}_N + \boldsymbol{E}\Delta\boldsymbol{\lambda}^{\circ}_{N+1}) \ , \tag{5.26a}$$

$$\boldsymbol{E}^T\Delta\boldsymbol{\epsilon}_{N+1} + \boldsymbol{H}\Delta\boldsymbol{\lambda}^{\circ}_{N+1} + \frac{\partial D}{\partial\Delta\boldsymbol{\lambda}}\bigg|_{\Delta\boldsymbol{\lambda}^{\circ}_{N+1}} = -(\boldsymbol{E}^T\boldsymbol{\epsilon}_N + \boldsymbol{H}\boldsymbol{\lambda}_N + \boldsymbol{H}\Delta\boldsymbol{\lambda}^{p}_{N+1}) \tag{5.26b}$$

where

$$\Delta\boldsymbol{\lambda}_{N+1} = \Delta\boldsymbol{\lambda}^{\circ}_{N+1} + \Delta\boldsymbol{\lambda}^{p}_{N+1} \ , \tag{5.27a}$$

$$\Delta\boldsymbol{\lambda}^{p}_{N+1} = \frac{(1-\alpha)}{\alpha}\Delta\boldsymbol{\lambda}^{\circ}_{N} \ . \tag{5.27b}$$

We introduce the modification to include linear isotropic hardening and decompose the scalar value $\Delta\gamma$ in a consistent manner

$$\Delta\gamma_{N+1} = \gamma_{N+1} - \gamma_N = \Delta\gamma^{\circ}{}_{N+1} + \Delta\gamma^{p}{}_{N+1} \ , \tag{5.28a}$$

$$\Delta\gamma^{p}{}_{N+1} = \frac{(1-\alpha)}{\alpha}\Delta\gamma^{\circ}{}_{N} \ . \tag{5.28b}$$

Following in the steps of previous arguments, the governing equations (eqns. 5.26) then become

$$\boldsymbol{C}\Delta\boldsymbol{\epsilon}_{N+1} + \boldsymbol{E}\Delta\boldsymbol{\lambda}^{\circ}_{N+1} = \boldsymbol{\sigma}_{N+1} - (\boldsymbol{C}\boldsymbol{\epsilon}_N + \boldsymbol{E}\boldsymbol{\lambda}_N + \boldsymbol{E}\Delta\boldsymbol{\lambda}^{p}_{N+1}) \ , \tag{5.29a}$$

$$\begin{aligned}\boldsymbol{E}^T\Delta\boldsymbol{\epsilon}_{N+1} + \boldsymbol{H}\Delta\boldsymbol{\lambda}^{\circ}_{N+1} + \frac{\partial}{\partial\Delta\boldsymbol{\lambda}}\big[&(1 + H\gamma_N + H\Delta\gamma^{p}{}_{N+1})D(\Delta\boldsymbol{\lambda}) \\ &+ \tfrac{1}{2}HD^2(\Delta\boldsymbol{\lambda})\big]\Big|_{\Delta\boldsymbol{\lambda}^{\circ}_{N+1}} = -(\boldsymbol{E}^T\boldsymbol{\epsilon}_N + \boldsymbol{H}\boldsymbol{\lambda}_N + \boldsymbol{H}\Delta\boldsymbol{\lambda}^{p}_{N+1}) \ .\end{aligned} \tag{5.29b}$$

Implementing these equations as the corrector step, we note that $\boldsymbol{\epsilon}_N, \boldsymbol{\lambda}_N$ and γ_N are known, $\Delta\boldsymbol{\lambda}^{p}_{N+1}$ and $\Delta\gamma^{p}{}_{N+1}$ are forward projections and are thus known, and $\Delta\boldsymbol{\epsilon}_{N+1}$

is given. Again the problem can be uncoupled. First, we can solve eqn. (5.29b) for $\Delta\boldsymbol{\lambda}^{\circ}_{N+1}$, and then recover $\boldsymbol{\sigma}_{N+1}$ from eqn. (5.29a), which can be written as

$$\boldsymbol{\sigma}_{N+1} = \boldsymbol{C}(\boldsymbol{\epsilon}_N + \Delta\boldsymbol{\epsilon}_{N+1}) + \boldsymbol{E}(\boldsymbol{\lambda}_N + \Delta\boldsymbol{\lambda}^{p}_{N+1} + \Delta\boldsymbol{\lambda}^{\circ}_{N+1}) \ . \tag{5.30}$$

The solution of eqn. (5.29b) can be written as a convex, non-linear programming problem. The function $U_{N+1}(\Delta\boldsymbol{\lambda})$, whose least value is given by $\Delta\boldsymbol{\lambda}^{\circ}_{N+1}$, is defined as

$$\begin{aligned} U_{N+1}(\Delta\boldsymbol{\lambda}) &= (\Delta\boldsymbol{\epsilon}^{T}_{N+1}\boldsymbol{E} + \boldsymbol{\epsilon}^{T}_{N}\boldsymbol{E} + \boldsymbol{\lambda}^{T}_{N}\boldsymbol{H}^{T} + \Delta\boldsymbol{\lambda}^{p\,T}_{N+1}\boldsymbol{H}^{T})\Delta\boldsymbol{\lambda} \\ &+\tfrac{1}{2}\Delta\boldsymbol{\lambda}^{T}\boldsymbol{H}\Delta\boldsymbol{\lambda} + (1 + H\gamma_N + H\Delta\gamma^{p}_{N+1})D(\Delta\boldsymbol{\lambda}) + \tfrac{1}{2}HD^{2}(\Delta\boldsymbol{\lambda}) \ . \end{aligned} \tag{5.31}$$

It is evident that the Gauss point solution procedures of eqns. (5.30) & (5.31) and eqns. (5.14) & (5.15) are equivalent. Thus, in specializing the corrector step to a piecewise linear yield surface, we may embed the linear programming problem of eqn. (5.19) into the convex non-linear programming problem of eqn. (5.31). The resulting convex quadratic programming problem in $\Delta\boldsymbol{\Lambda}$ for the generalized trapezoidal rule is then given by

$$min(\ U_{N+1}(\Delta\boldsymbol{\Lambda}) : \Delta\boldsymbol{\Lambda} \geq 0\ ) \tag{5.32a}$$

where

$$\begin{aligned} U_{N+1}(\Delta\boldsymbol{\Lambda}) &= \Big[(\Delta\boldsymbol{\epsilon}^{T}_{N+1}\boldsymbol{E} + \boldsymbol{\epsilon}^{T}_{N}\boldsymbol{E} + \boldsymbol{\lambda}^{T}_{N}\boldsymbol{H}^{T} + \Delta\boldsymbol{\lambda}^{p\,T}_{N+1}\boldsymbol{H}^{T})\boldsymbol{N}^{T} + \\ &(1 + H\gamma_N + H\Delta\gamma^{p}_{N+1})\boldsymbol{\chi}^{T}_{0}\Big]\,\Delta\boldsymbol{\Lambda} + \tfrac{1}{2}\Delta\boldsymbol{\Lambda}^{T}[\boldsymbol{N}\boldsymbol{H}\boldsymbol{N}^{T} + H\boldsymbol{\chi}_0\boldsymbol{\chi}^{T}_{0}]\Delta\boldsymbol{\Lambda} \ . \end{aligned} \tag{5.32b}$$

We denote its solution by $\Delta\boldsymbol{\Lambda}^{\circ}_{N+1}$, and recover $\Delta\boldsymbol{\lambda}^{\circ}_{N+1}$ from the relation

$$\Delta\boldsymbol{\lambda}^{\circ}_{N+1} = \boldsymbol{N}^{T}\Delta\boldsymbol{\Lambda}^{\circ}_{N+1} \ . \tag{5.33}$$

In extending the framework for the Tresca and Mohr-Coulomb yield surfaces to the generalized trapezoidal rule, the quadratic programming problem of eqn. (5.32) needs to be solved in principal $\Delta\boldsymbol{\lambda}^{\circ}_{N+1}$ space, rather than in the reference space.

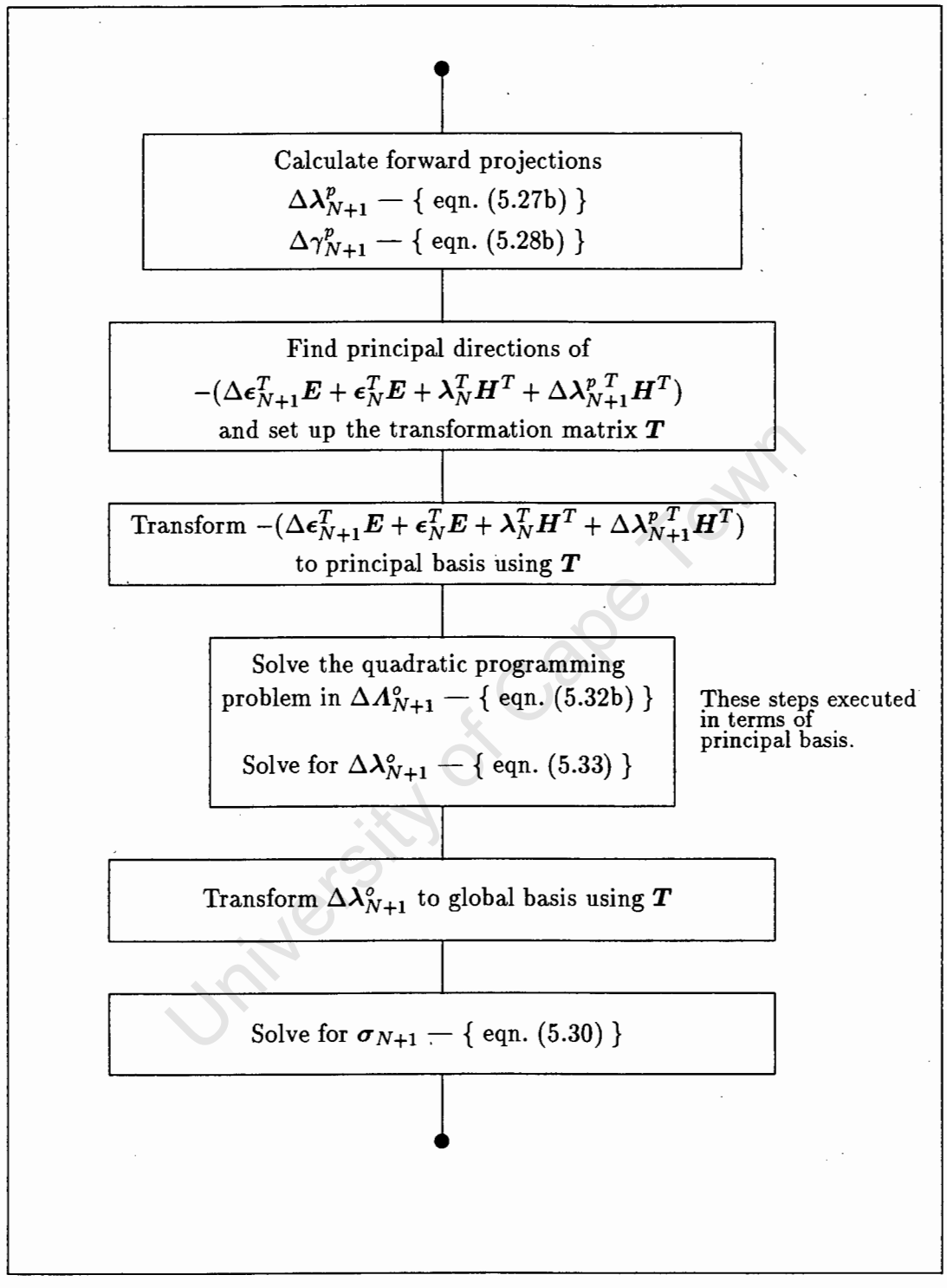

Figure 5.4: Framework for the Tresca and Mohr-Coulomb yield surfaces (generalized trapezoidal)

For a perfectly plastic or linear kinematic hardening rule, eqn. (5.26b) can be written as

$$-(\Delta\epsilon^T_{N+1}\boldsymbol{E} + \epsilon^T_N\boldsymbol{E} + \boldsymbol{\lambda}^T_N\boldsymbol{H}^T + \Delta\boldsymbol{\lambda}^{p\,T}_{N+1}\boldsymbol{H}^T) = (\boldsymbol{\chi}_{N+1} + \boldsymbol{H}\Delta\boldsymbol{\lambda}^\circ_{N+1}) \qquad (5.34)$$

and, for a linear isotropic hardening rule, eqn. (5.29b) can be written as

$$-(\Delta\epsilon^T_{N+1}\boldsymbol{E} + \epsilon^T_N\boldsymbol{E} + \boldsymbol{\lambda}^T_N\boldsymbol{H}^T + \Delta\boldsymbol{\lambda}^{p\,T}_{N+1}\boldsymbol{H}^T) = (\boldsymbol{\chi}_{N+1} + \boldsymbol{H}\Delta\boldsymbol{\lambda}^\circ_{N+1} + \frac{\partial}{\partial\Delta\boldsymbol{\lambda}}[H(\gamma_N + \Delta\gamma^p{}_{N+1})D(\Delta\boldsymbol{\lambda}) + \tfrac{1}{2}HD^2(\Delta\boldsymbol{\lambda})]\Big|_{\Delta\boldsymbol{\lambda}^\circ_{N+1}}) \; . \qquad (5.35)$$

As the increment $\Delta\boldsymbol{\lambda}^\circ{}_{N+1}$ is evaluated at time T_{n+1}, and following a similar argument given in section 5.4.1, both terms in parentheses in eqns. (5.34) and (5.35) have principal directions corresponding to those of $\Delta\boldsymbol{\lambda}^\circ{}_{N+1}$. Thus, by determining the principal directions of the left-hand term, a transformation between reference and principal $\Delta\boldsymbol{\lambda}^\circ{}_{N+1}$ space can be established *a priori*.

This is again equivalent to first formulating the problem at each iteration as a holonomic problem in $\boldsymbol{\chi}$ space, except now the forward projection $\Delta\boldsymbol{\lambda}^{p\,T}_{N+1}\boldsymbol{H}^T$ is included in the holonomic step. The transformation and minimization are then carried out in exactly the same way as for the backward difference algorithm, in the sense that the same quadratic programming problem is solved in each case. This facilitates a simple extension from the backward difference algorithm to an algorithm using the generalized trapezoidal rule. The extended framework for the Tresca and Mohr-Coulomb yield surfaces is given in Figure 5.4.

5.6 CONCLUSIONS

Firstly, a backward difference corrector algorithm for piecewise linear plasticity has been presented. The algorithm is written in the form of a convex quadratic programming problem and is fully consistent, in that no heuristic assumptions are made.

Secondly, for the particular cases of the Tresca and Mohr-Coulomb yield surfaces, a framework is established that is equivalent to first formulating the problem as a holonomic problem in $\boldsymbol{\chi}$ space. This allows the associated convex quadratic programming problem to be solved in principal space, while the corrector step is formulated in reference space.

Thirdly, the algorithm is extended to include the generalized trapezoidal rule. For the Tresca and Mohr-Coulomb cases, this entails a modification of the framework by including the forward projection as part of the holonomic step. The general structure of the backward difference algorithm is thus maintained, in the sense that the same quadratic programming problem is solved in both cases.

REFERENCES

1. Crisfield, M.A., Plasticity computations using the Mohr-Coulomb yield criterion, *Engineering Computations*, **4**, 300-308, 1987.

2. Crisfield, M.A., Consistent Schemes for Plasticity Computation with the Newton-Raphson Method, Proc. 1st Int. Conf. *Computational Plasticity: Models, Software and Applications*, (D.R.J. Owen, E. Hinton & E. Onate *eds.*), Part 1, Pineridge Press, 133-159, 1987.

3. De Borst, R., Integration of Plasticity Equations for Singular Yield Functions, *Computers and Structures*, **26**, 823-829, 1987.

4. Eve, R.A., Reddy, B.D. & Rockafellar, R.T., An Internal Variable Theory of Elastoplasticity based on the Maximum Plastic Work Inequality, *Quarterly of Applied Mathematics* **48**, 59-83, 1990.

5. Maier, G. Quadratic Programming and Theory of Elastic-Perfectly Plastic Structures, *Meccanica*, **3**, 265-273, 1968.

6. Martin, J.B. & Nappi, A., An Internal Variable Formulation of Perfectly Plastic and Linear Kinematic and Isotropic Hardening Relations with a Von Mises Yield Condition, *European Journal of Mechanics, A-Solids*, **9**, 107-131, 1990.

7. Martin, J.B., Reddy, B.D., Griffin, T.B. & Bird, W.W., Applications of Mathematical Programming Concepts to Incremental Elastic-Plastic Analysis, *Engineering Structures*, **9**, 171-176, 1987.

8. Pankaj & Bićanić, N., On Multivector Stress Returns in Mohr-Coulomb Plasticity, *Computational Plasticity: Models, Software and Applications*, (D.R.J. Owen, E. Hinton & E. Onate *eds.*), Part 1, Pineridge Press, 421-436, 1989.

9. Simo, J.C., Kennedy, J.G. & Govindjee, S., Non-Smooth Multisurface Plasticity and Viscoplasticity. Loading/Unloading Conditions and Numerical Algorithms, *International Journal for Numerical Methods in Engineering*, **26**, 2161-2185, 1988.

CHAPTER 6
CORRECTOR STEP ALGORITHMS

6.1 INTRODUCTION

In this chapter, the quadratic programming problems for the Tresca and Mohr-Coulomb yield surfaces with perfectly plastic and linear hardening material behaviour are developed. These quadratic programming problems are solved in principal space and, provided the problem is first formulated as the appropriate holonomic problem in conjugate force space, are common to both the backward difference and generalized trapezoidal rules.

Because the corrector algorithm formulated as a quadratic programming problem is fully consistent, the quadratic programming problems can be used to provide a basis against which heuristically developed algorithms can be compared. This is done in the form of the traditional return mapping algorithm in which the return paths in principal stress space associated with an elastically predicted stress are identified.

6.2 THE QUADRATIC PROGRAMMING PROBLEMS FOR THE TRESCA YIELD SURFACE

The derivation of the quadratic programming problems for the Tresca yield surface with perfectly plastic, linear kinematic and linear isotropic hardening material behaviour, is now considered. For principal conjugate forces χ_1, χ_2, χ_3, the yield surface is the envelope bounded by

$$\begin{aligned} \phi &= \pm\tfrac{1}{2}(\chi_1 - \chi_3) - k = 0 \\ \phi &= \pm\tfrac{1}{2}(\chi_1 - \chi_2) - k = 0 \\ \phi &= \pm\tfrac{1}{2}(\chi_2 - \chi_3) - k = 0 \end{aligned} \tag{6.1}$$

where k is a scalar constant.

Eqns. (6.1) describe the surfaces of an infinitely long regular hexagonal prism in

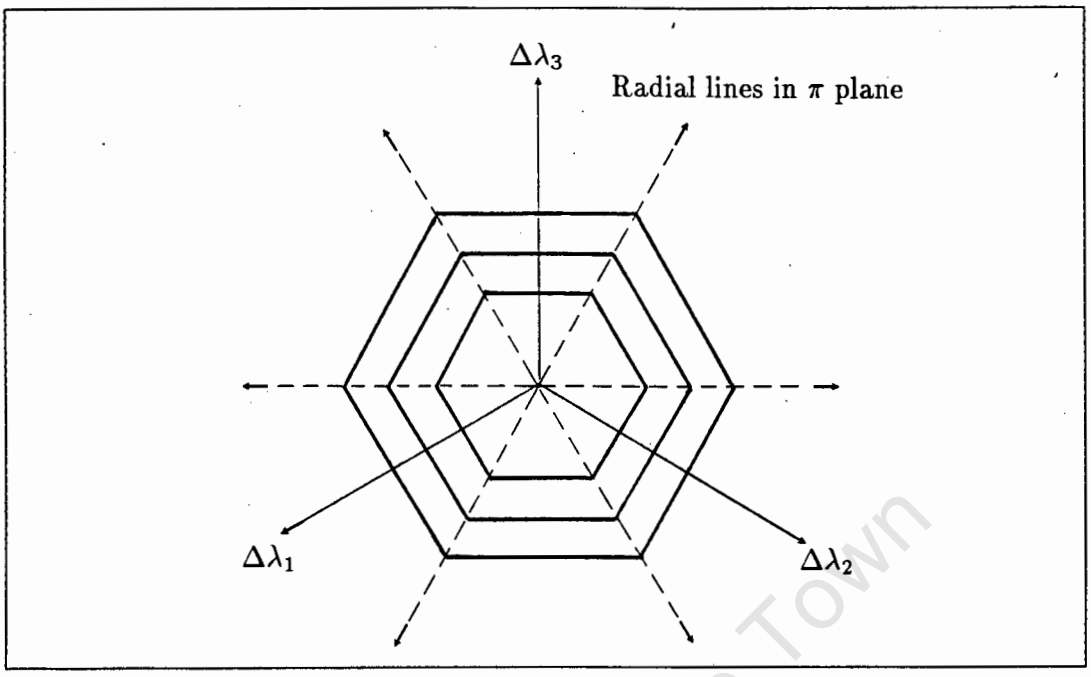

Figure 6.1: Level surfaces of the Tresca dissipation function

principal χ space, with an axis in the hydrostatic direction. Associated with this yield surface is a dissipation function with piecewise linear level surfaces in the principal $\Delta\boldsymbol{\lambda}$ deviator plane, as shown in Figure 6.1.

The matrix $\boldsymbol{N}^T$ and the vector $\boldsymbol{\chi}_0^T$ are given by

$$\boldsymbol{N}^T = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} . \tag{6.2}$$

$$\boldsymbol{\chi}_0^T = 2k \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} . \tag{6.3}$$

For this case $\Delta\boldsymbol{\lambda} \equiv \Delta\boldsymbol{\epsilon}^p$ and, as $\Delta\boldsymbol{\lambda}$ lies in the deviator plane, the plastic strains are deviatoric. We can thus divide the incremental problem for the time interval (t_n, T_{n+1}) into its hydrostatic and deviatoric components and solve the hydrostatic component as an elastic problem and the deviatoric component as an elastic-plastic problem. We thus only consider the deviatoric elastic-plastic problem for the three material models.

6.2.1 The perfectly plastic case

The specific free energy is given by

$$f = G(\boldsymbol{e} - \boldsymbol{e}^p)(\boldsymbol{e} - \boldsymbol{e}^p) \tag{6.4}$$

where G is the shear modulus. The deviatoric strains $\boldsymbol{e}$ replace the strain components $\boldsymbol{\epsilon}$ and the plastic strains $\boldsymbol{e}^p$ the internal variables $\boldsymbol{\lambda}$ of the general formulation of chapter 5. The equations of state then give

$$\boldsymbol{s} = \frac{\partial f}{\partial \boldsymbol{e}} = \boldsymbol{C}\boldsymbol{e} + \boldsymbol{E}\boldsymbol{e}^p = 2G(\boldsymbol{e} - \boldsymbol{e}^p) \ , \tag{6.5a}$$

$$-\boldsymbol{\chi} = \frac{\partial f}{\partial \boldsymbol{e}^p} = \boldsymbol{E}^T\boldsymbol{e} + \boldsymbol{H}\boldsymbol{e}^p = -2G(\boldsymbol{e} - \boldsymbol{e}^p) = -\boldsymbol{s} \ , \tag{6.5b}$$

where $\boldsymbol{s}$ is the deviatoric stress, in place of $\boldsymbol{\sigma}$.

Using the arguments given in chapter 5, the convex quadratic programming problem is given by

$$min(\ U_{N+1}(\Delta\boldsymbol{\Lambda}) : \Delta\boldsymbol{\Lambda} \geq 0\ ) \tag{6.6a}$$

where

$$\begin{aligned} U_{N+1}(\Delta\boldsymbol{\Lambda}) = [(-\Delta\tilde{\boldsymbol{e}}^T_{n+1} - \boldsymbol{e}^T_n + \boldsymbol{e}^{pT}_n)2G\boldsymbol{N}^T + \boldsymbol{\chi}^T_0]\Delta\boldsymbol{\Lambda} \\ + \tfrac{1}{2}\Delta\boldsymbol{\Lambda}^T[\boldsymbol{N}2G\boldsymbol{N}^T]\Delta\boldsymbol{\Lambda} \ . \end{aligned} \tag{6.6b}$$

6.2.2 The linear kinematic hardening case

The specific free energy is given by

$$f = G(\boldsymbol{e} - \boldsymbol{e}^p)(\boldsymbol{e} - \boldsymbol{e}^p) + \tfrac{1}{2}\kappa\boldsymbol{e}^p\boldsymbol{e}^p \ , \tag{6.7}$$

where κ is the hardening parameter given by

$$\kappa = \frac{G_T G}{G - G_T} \tag{6.8}$$

and G_T is the tangent shear modulus.

The equations of state give

$$\boldsymbol{s} = \frac{\partial f}{\partial \boldsymbol{e}} = \boldsymbol{C}\boldsymbol{e} + \boldsymbol{E}\boldsymbol{e}^p = 2G(\boldsymbol{e} - \boldsymbol{e}^p) \ , \tag{6.9a}$$

$$-\boldsymbol{\chi} = \frac{\partial f}{\partial \boldsymbol{e}^p} = \boldsymbol{E}^T\boldsymbol{e} + \boldsymbol{H}\boldsymbol{e}^p = -2G\boldsymbol{e} + (2G + \kappa)\boldsymbol{e}^p \ , \tag{6.9b}$$

and the convex quadratic programming problem is given by

$$min(\ U_{N+1}(\Delta\boldsymbol{\Lambda}) : \Delta\boldsymbol{\Lambda} \geq 0\ ) \tag{6.10a}$$

where

$$\begin{aligned} U_{N+1}(\Delta\boldsymbol{\Lambda}) = [\boldsymbol{e}^{pT}_n(2G + \kappa)\boldsymbol{N}^T - (\Delta\tilde{\boldsymbol{e}}^T_{n+1} + \boldsymbol{e}^T_n)2G\boldsymbol{N}^T + \boldsymbol{\chi}^T_0]\Delta\boldsymbol{\Lambda} \\ + \tfrac{1}{2}\Delta\boldsymbol{\Lambda}^T[\boldsymbol{N}(2G + \kappa)\boldsymbol{N}^T]\Delta\boldsymbol{\Lambda} \ . \end{aligned} \tag{6.10b}$$

6.2.3 The linear isotropic hardening case

The specific free energy is given by

$$f = G(\boldsymbol{e} - \boldsymbol{e}^p)(\boldsymbol{e} - \boldsymbol{e}^p) + \frac{H\gamma^2}{2} \ , \tag{6.11}$$

where

$$\gamma = \int_0^t D(\dot{\boldsymbol{e}}^p)dt \ , \tag{6.12}$$

$$H = \frac{\kappa}{k^2} \quad = \frac{G_T G}{k^2(G - G_T)} \quad . \tag{6.13}$$

The equations of state give

$$\boldsymbol{s} = \frac{\partial f}{\partial \boldsymbol{e}} = \boldsymbol{C}\boldsymbol{e} + \boldsymbol{E}\boldsymbol{e}^p = 2G(\boldsymbol{e} - \boldsymbol{e}^p) \quad , \tag{6.14a}$$

$$-\boldsymbol{\chi} = \frac{\partial f}{\partial \boldsymbol{e}^p} = \boldsymbol{E}^T\boldsymbol{e} + \boldsymbol{H}\boldsymbol{e}^p = -2G(\boldsymbol{e} - \boldsymbol{e}^p) + H\gamma\frac{\partial \gamma}{\partial \boldsymbol{e}^p} \quad . \tag{6.14b}$$

For $t_n < t \leq T_{n+1}$,

$$\gamma = \gamma_n + D(\Delta \boldsymbol{e}^p) \quad , \tag{6.15}$$

and the convex quadratic programming problem is given by

$$min(\ U_{N+1}(\Delta \boldsymbol{\Lambda}) : \Delta \boldsymbol{\Lambda} \geq 0\ ) \tag{6.16a}$$

where

$$U_{N+1}(\Delta \boldsymbol{\Lambda}) = [(-\Delta \tilde{\boldsymbol{e}}^T_{n+1} - \boldsymbol{e}^T_n + \boldsymbol{e}^{pT}_n)2G\boldsymbol{N}^T + (1 + \frac{\kappa}{k^2}\gamma_n)\boldsymbol{\chi}^T_0]\Delta \boldsymbol{\Lambda}$$
$$+\tfrac{1}{2}\Delta \boldsymbol{\Lambda}^T[\boldsymbol{N}2G\boldsymbol{N}^T + \frac{\kappa}{k^2}\boldsymbol{\chi}_0\boldsymbol{\chi}^T_0]\Delta \boldsymbol{\Lambda} \quad . \tag{6.16b}$$

6.3 THE QUADRATIC PROGRAMMING PROBLEMS FOR THE MOHR-COULOMB YIELD SURFACE

We next consider the derivation of the quadratic programming problems for the Mohr-Coulomb yield surface with perfectly plastic and linear isotropic hardening material behaviour. For principal conjugate forces χ_1, χ_2, χ_3, the yield surface is the envelope bounded by

$$\begin{aligned}
\phi &= \pm\tfrac{1}{2}(\chi_1 - \chi_3) \pm \tfrac{1}{2}(\chi_1 + \chi_3)\sin\theta - c\cos\theta = 0 \\
\phi &= \pm\tfrac{1}{2}(\chi_1 - \chi_2) \pm \tfrac{1}{2}(\chi_1 + \chi_2)\sin\theta - c\cos\theta = 0 \\
\phi &= \pm\tfrac{1}{2}(\chi_2 - \chi_3) \pm \tfrac{1}{2}(\chi_2 + \chi_3)\sin\theta - c\cos\theta = 0
\end{aligned} \tag{6.17}$$

where c is the cohesion and θ is the angle of internal friction.

Eqns. (6.17) describe the surface of a cone whose normal section at any point is an irregular hexagon in principal $\boldsymbol{\chi}$ space. The axis of the cone coincides with the hydrostatic axis and the apex is at the point given by $\chi_1 = \chi_2 = \chi_3 = c\cot\theta$. Associated with this yield surface is a dissipation function with piecewise linear level surfaces in principal $\Delta\boldsymbol{\lambda}$ space, as shown in Figure 6.2.

The matrix $\boldsymbol{N}^T$ and the vector $\boldsymbol{\chi}_0^T$ associated with this dissipation function are given by

$$\boldsymbol{N}^T = \begin{bmatrix}
\frac{(\sin\theta+1)}{2} & \frac{(\sin\theta-1)}{2} & \frac{(\sin\theta-1)}{2} & \frac{(\sin\theta+1)}{2} & 0 & 0 \\
0 & 0 & \frac{(\sin\theta+1)}{2} & \frac{(\sin\theta-1)}{2} & \frac{(\sin\theta-1)}{2} & \frac{(\sin\theta+1)}{2} \\
\frac{(\sin\theta-1)}{2} & \frac{(\sin\theta+1)}{2} & 0 & 0 & \frac{(\sin\theta+1)}{2} & \frac{(\sin\theta-1)}{2}
\end{bmatrix} \tag{6.18}$$

$$\boldsymbol{\chi}_0^T = \begin{bmatrix} c\cos\theta & c\cos\theta & c\cos\theta & c\cos\theta & c\cos\theta & c\cos\theta \end{bmatrix} . \tag{6.19}$$

6.3.1 The perfectly plastic case

The specific free energy is given by

$$f = \tfrac{1}{2}(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}^p)^T \boldsymbol{D}(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}^p) \ , \tag{6.20}$$

where $\boldsymbol{D}$ is the principal isotropic elastic constitutive tensor given by

$$\boldsymbol{D} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix}
(1-\nu) & \nu & \nu \\
\nu & (1-\nu) & \nu \\
\nu & \nu & (1-\nu)
\end{bmatrix} \tag{6.21}$$

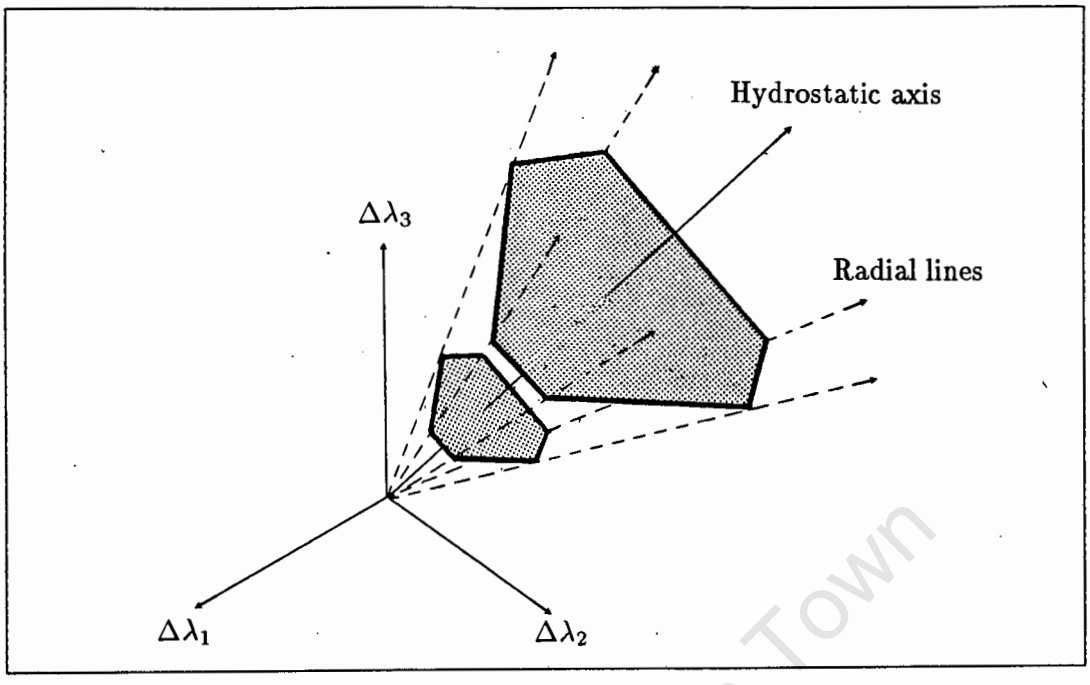

Figure 6.2: Level surfaces of the Mohr-Coulomb dissipation function

and E is Young's modulus and ν is Poisson's ratio.

The plastic strains ϵ^p are identified as the internal variables $\boldsymbol{\lambda}$ of the general formulation and the equations of state then give:

$$\boldsymbol{\sigma} = \frac{\partial f}{\partial \boldsymbol{\epsilon}} = \boldsymbol{C\epsilon} + \boldsymbol{E\epsilon}^p = \boldsymbol{D}(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}^p) \quad , \tag{6.22a}$$

$$-\boldsymbol{\chi} = \frac{\partial f}{\partial \boldsymbol{\epsilon}^p} = \boldsymbol{E}^T\boldsymbol{\epsilon} + \boldsymbol{H\epsilon}^p = -\boldsymbol{D}(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}^p) = -\boldsymbol{\sigma} \quad . \tag{6.22b}$$

Using the arguments given in chapter 5, the convex quadratic programming problem is given by

$$min(\ U_{N+1}(\Delta\boldsymbol{\Lambda}) : \Delta\boldsymbol{\Lambda} \geq 0\ ) \tag{6.23a}$$

where

$$\begin{aligned} U_{N+1}(\Delta\boldsymbol{\Lambda}) = [(-\Delta\tilde{\boldsymbol{\epsilon}}^T_{n+1} - \boldsymbol{\epsilon}^T_n + \boldsymbol{\epsilon}^{pT}_n)\boldsymbol{DN}^T + \boldsymbol{\chi}^T_0]\Delta\boldsymbol{\Lambda} \\ + \tfrac{1}{2}\Delta\boldsymbol{\Lambda}^T[\boldsymbol{NDN}^T]\Delta\boldsymbol{\Lambda} \quad . \end{aligned} \tag{6.23b}$$

6.3.2 The linear isotropic hardening case

It is assumed that the internal friction angle θ remains constant during hardening and that the parameter that varies is the cohesion c. The specific free energy is then given by

$$f = \tfrac{1}{2}(\epsilon - \epsilon^p)^T \boldsymbol{D}(\epsilon - \epsilon^p) + \frac{H\gamma^2}{2} \tag{6.24}$$

where

$$\gamma = \int_0^t D(\dot{\epsilon^p})dt \tag{6.25}$$

$$H = \frac{\kappa}{(c\cos\theta)^2} \quad = \frac{G_T G}{(c\cos\theta)^2(G - G_T)} \ . \tag{6.26}$$

The equations of state give

$$\boldsymbol{\sigma} = \frac{\partial f}{\partial \epsilon} = \boldsymbol{C}\epsilon + \boldsymbol{E}\epsilon^p = \boldsymbol{D}(\epsilon - \epsilon^p) \ , \tag{6.27a}$$

$$-\boldsymbol{\chi} = \frac{\partial f}{\partial \epsilon^p} = \boldsymbol{E}^T\epsilon + \boldsymbol{H}\epsilon^p = -\boldsymbol{D}(\epsilon - \epsilon^p) + H\gamma\frac{\partial \gamma}{\partial \epsilon^p} \ . \tag{6.27b}$$

For $t_n < t \leq T_{n+1}$,

$$\gamma = \gamma_n + D(\Delta\epsilon^p) \tag{6.28}$$

and the convex quadratic programming problem is given by

$$min(\ U_{N+1}(\Delta\boldsymbol{\Lambda}) : \Delta\boldsymbol{\Lambda} \geq 0\ ) \tag{6.29a}$$

where

$$U_{n+1}(\Delta\boldsymbol{\Lambda}) = [(-\Delta\tilde{\boldsymbol{\epsilon}}^T_{n+1} - \boldsymbol{\epsilon}^T_n + \boldsymbol{\epsilon}^{pT}_n)\boldsymbol{D}\boldsymbol{N}^T + (1 + \frac{\kappa}{(c\cos\theta)^2}\gamma_n)\boldsymbol{\chi}^T_0]\Delta\boldsymbol{\Lambda}$$
$$+\tfrac{1}{2}\Delta\boldsymbol{\Lambda}^T[\boldsymbol{N}\boldsymbol{D}\boldsymbol{N}^T + \frac{\kappa}{(c\cos\theta)^2}\boldsymbol{\chi}_0\boldsymbol{\chi}^T_0]\Delta\boldsymbol{\Lambda} \quad . \text{(6.29b)}$$

6.4 UNIAXIAL EXAMPLES

We now consider the material response associated with the quadratic programming problems developed for the Tresca yield surface by carrying out a uniaxial test. As it is convenient to illustrate the material response in stress rather than conjugate force space, the test is carried out along a line of pure shear in principal deviatoric stress space. For the Tresca yield surface, these lines have the same direction as radial lines defined by n_i passing through the vertices of level surfaces of dissipation. The material response for the perfectly plastic, linear kinematic hardening and linear isotropic hardening cases are plotted in deviatoric shear stress/shear strain space in Figure 6.3.

For the linear isotropic hardening case, the increment in the scalar value k can be derived geometrically as

$$\Delta k = \frac{GG_T}{G - G_T}(\Delta e^p_1 - \Delta e^p_3) \quad , \tag{6.30}$$

or

$$\Delta k = \frac{G_T}{G}[(s^E_1 - s^E_3) - k] \quad , \tag{6.31}$$

where $\boldsymbol{s}^E$ is the elastically predicted deviatoric stress.

These equations are the Tresca equivalent of the equations for the Von Mises yield surface for a holonomic incremental problem[2].

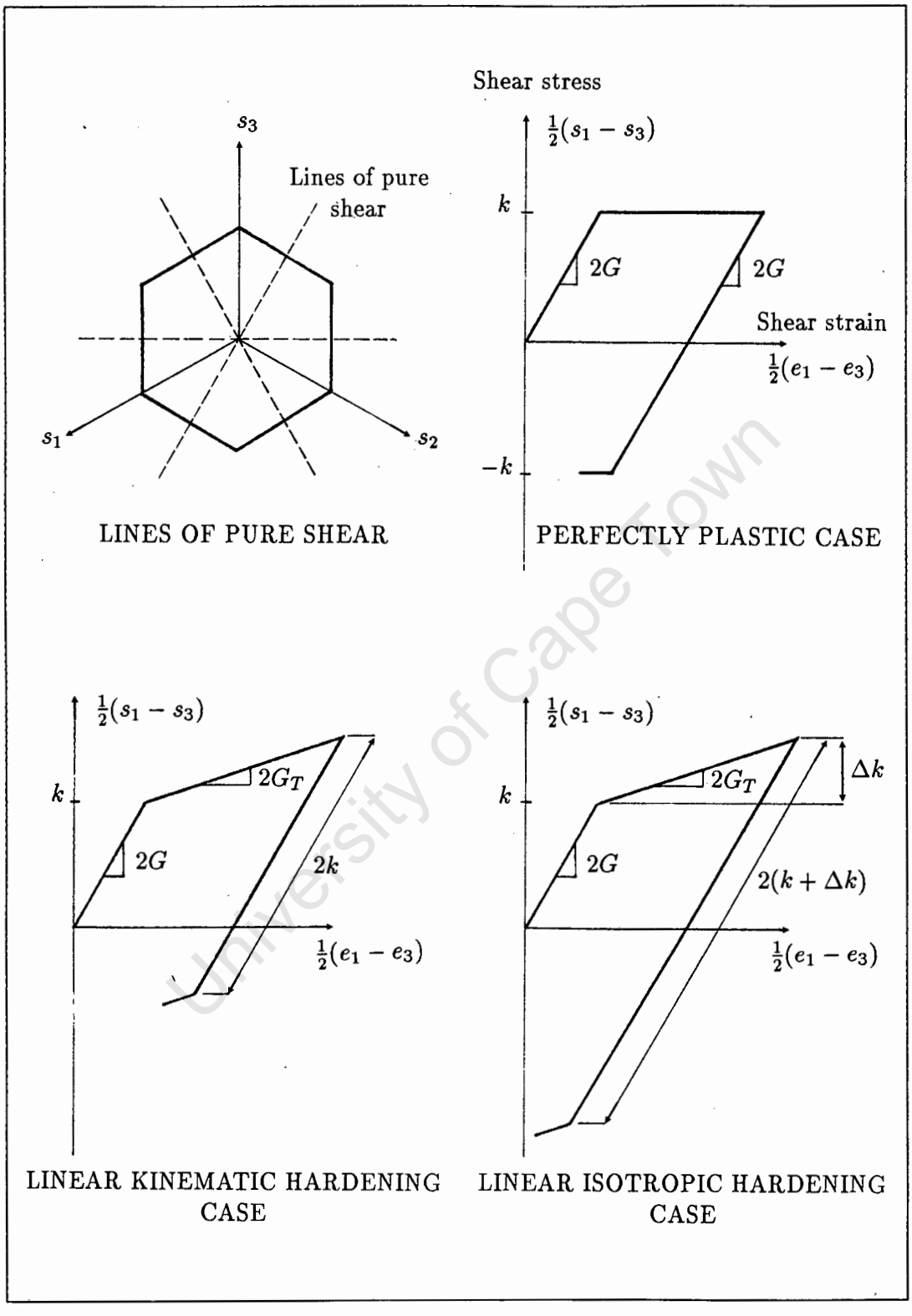

Figure 6.3: Uniaxial examples (Tresca yield surface)

6.5 THE RETURN PATHS IN PRINCIPAL STRESS SPACE

The quadratic programming problems developed in this chapter are solved in principal space and contain no heuristic assumptions. They can thus be used to identify the correct return paths in principal stress space for both a backward difference and generalized trapezoidal integration rule. This is done in the form of the traditional return mapping algorithm described in chapter 2.

In this form, the return path for the $(i+1)$-th iteration of the $n+1$-th increment is from the elastically predicted stress point $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ to the final stress point $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ in stress space. We thus need to identify each $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ corresponding to all given $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$. Further, for hardening solids, the yield surface in stress space may change during the increment and we also need to identify this change.

In adapting the framework for the Tresca and Mohr-Coulomb yield surfaces of chapter 5 to the form of the return mapping algorithm, the elastically predicted stress $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ needs to be brought into correspondence with the holonomic step in conjugate force space. For the backward difference scheme, we write

$$\boldsymbol{\sigma}_{N+1}^{E\,(i+1)} = \sigma_n + \boldsymbol{D}\Delta\tilde{\boldsymbol{\epsilon}}_{n+1}^{(i+1)} \quad , \tag{6.32}$$

and for the generalized trapezoidal rule, we include the forward projection $\Delta\boldsymbol{\lambda}_{N+1}^{p}$ by updating the yield surface from ϕ_N to ϕ_n and writing

$$\boldsymbol{\sigma}_{N+1}^{E\,(i+1)} = \sigma_N + \boldsymbol{D}\Delta\boldsymbol{\epsilon}_{N+1}^{(i+1)} - \boldsymbol{D}\Delta\boldsymbol{\lambda}_{N+1}^{p} \quad . \tag{6.33}$$

Having defined the yield surface and the elastically predicted stress in this way, the return paths in principal stress space for both the backward difference and generalized trapezoidal rules coincide.

It is pertinent to note that, for all time increments, both the Tresca and Mohr-Coulomb yield surfaces remain fixed in conjugate force space for hardening solids, and that all the returns presented below for these surfaces correspond to a closest point return in this space.

For the description of the return paths in principal stress space, however, it is convenient, for a given yield surface and hardening rule, to group the $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ in such a way that the $\Delta\boldsymbol{\epsilon}_{N+1}^{p\,(i+1)}$ associated with the said $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ lie on level surfaces of the dissipation function D. The $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ grouped in this way form concentric piecewise

linear surfaces in principal stress space, with axes coinciding with the current yield surface axis. Only for the linear kinematic hardening case are we unable to construct surfaces of $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ that correspond to level surfaces of D. For this hardening case, the surfaces of $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ defined for the perfectly plastic case are used.

As the surfaces of $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ are concentric, each $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ can only lie on one such surface and thus, we need only consider the change in the yield surface associated with, and the return paths from, the $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on one of these surfaces.

6.5.1 The Tresca yield surface

The piecewise linear surfaces of $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ associated with the Tresca yield surface are infinitely long concentric 12-sided prisms in principal stress space. Further, as the plastic strains are deviatoric, we need only consider the return paths in principal deviator stress space.

We divide the area around the appropriate yield surface in the deviator plane into flat and corner regions, defined by adjacent normals $\boldsymbol{n}_i$ at the corners of the appropriate yield surface. Due to the symmetry of the yield surface, we need only make a distinction between flat and corner regions, and not between individual flat regions or individual corner regions respectively.

The perfectly plastic case

For a perfectly plastic solid the yield surface remains fixed in principal stress space for all time increments. Thus the current and subsequent yield surface is the initial yield surface ϕ_0.

The surface of elastically predicted stress points $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ for this hardening case is illustrated in Figure 6.4, and is constructed with reference to the *initial* yield surface ϕ_0, as are the flat and corner regions. The return paths associated with the $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on this surface are characterized below.

All $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on the surface within a flat region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ on the flat surface of the yield surface ϕ_0 along the normal $\boldsymbol{n}_i$ at $\boldsymbol{\sigma}_{N+1}^{(i+1)}$. All $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on the surface within a corner region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ at the corner of the yield surface ϕ_0. These return paths are illustrated in Figure 6.4.

The linear kinematic hardening case

We consider the same surface of elastically predicted stress points $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ as defined for the perfectly plastic case, except it is now constructed with reference to the *current* rather than the initial yield surface. This surface, which does not correspond to a level surface of D for this hardening case, is illustrated in Figure 6.5, together with the flat and corner regions which are also constructed with reference to the current yield surface ϕ_n.

We identify the stress points $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ that return from $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on this surface and construct a surface of these stress points. The surface of $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ does not coincide with any one yield surface and is illustrated in Figure 6.5 together with the return paths which are characterized below.

All $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on the surface within a flat region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ on the flat surface of the *subsequent* yield surface along the normal $\boldsymbol{n}_i$ at $\boldsymbol{\sigma}_{N+1}^{(i+1)}$. All $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on the surface within a corner region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ at the corner of the *subsequent* yield surface ϕ_{N+1}. This return is along the straight line joining $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ and the corner of ϕ_n.

During the time increment the yield surface translates. This translation is given by $\kappa\Delta\boldsymbol{e}_{N+1}^{p\,(i+1)}$ and is in the opposite direction to the return path $-2G\Delta\boldsymbol{e}_{N+1}^{p\,(i+1)}$. We can thus identify the subsequent yield surfaces on which the stress points lie. These are illustrated in Figures 6.6 and 6.7 for a flat and corner region respectively.

The linear isotropic hardening case

For this hardening case, we consider a surface of elastically predicted stress points $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ constructed with reference to the *subsequent* yield surface ϕ_{N+1}. We also define the flat and corner regions with reference to the subsequent yield surface ϕ_{N+1}. This surface of $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ and the appropriate regions are illustrated in Figure 6.8 together with the return paths which are characterized below.

All $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on the surface within a flat region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ on the flat surface of the subsequent yield surface along the normal $\boldsymbol{n}_i$ at $\boldsymbol{\sigma}_{N+1}^{(i+1)}$. All $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on the surface within a corner region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ at the corner of the subsequent yield surface ϕ_{N+1}. The surface of these stress points $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ coincides with the subsequent yield surface ϕ_{N+1}.

As the yield surface expands in stress space during the time increment, the corner of the subsequent yield surface lies on the radial line through the corner of the current

(and initial) yield surface. This expansion of the yield surface in stress space can be measured by the increment $\Delta k_{N+1}^{(i+1)}$. This increment is linked to a level surface of D ie

$$\Delta k_{N+1}^{(i+1)} \propto D(\Delta \boldsymbol{e}_{N+1}^{p\,(i+1)}) \tag{6.34}$$

and corresponds to the largest of

$$\begin{aligned}
\Delta k_{N+1}^{(i+1)} &= \pm \frac{GG_T}{G - G_T}(\Delta e_{1\,N+1}^{p\,(i+1)} - \Delta e_{2\,N+1}^{p\,(i+1)}) \\
\Delta k_{N+1}^{(i+1)} &= \pm \frac{GG_T}{G - G_T}(\Delta e_{1\,N+1}^{p\,(i+1)} - \Delta e_{3\,N+1}^{p\,(i+1)}) \\
\Delta k_{N+1}^{(i+1)} &= \pm \frac{GG_T}{G - G_T}(\Delta e_{2\,N+1}^{p\,(i+1)} - \Delta e_{3\,N+1}^{p\,(i+1)})
\end{aligned} \tag{6.35}$$

for the flat regions only.

There is a correspondence between eqn. (6.35) and eqn. (6.30) for the uniaxial case. We can make use of eqn. (6.35) to construct the concentric surfaces of $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ about the *current* yield surface. From these we can then identify the subsequent yield surface ϕ_{N+1} and hence the stress point $\boldsymbol{\sigma}_{N+1}^{(i+1)}$.

Traditionally, the flat and corner regions have been constructed with reference to the current yield surface ϕ_n rather than with reference to the subsequent yield surface. The return paths have also been associated with these regions of ϕ_n. We can identify the return paths given above in terms of these regions of ϕ_n by defining two secondary regions within each corner region of ϕ_n. The return paths within the flat regions of ϕ_n and these secondary regions are then characterized by those of a flat region of ϕ_{N+1}. The return paths within the remainder of the corner region of ϕ_n are characterized by those of a corner region of ϕ_{N+1}. These secondary regions can be calculated from the shear and tangent shear moduli by

$$\tan\beta = \frac{G_T}{\sqrt{3}G} \tag{6.36}$$

and are illustrated in Figure 6.9.

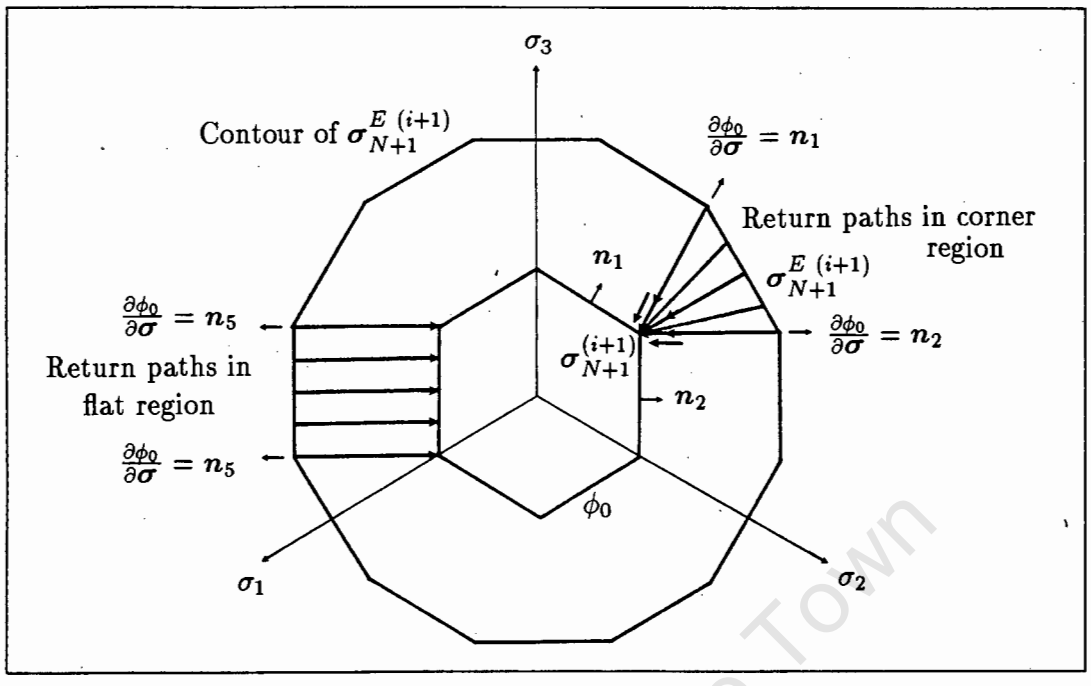

Figure 6.4: Return paths for the Tresca perfectly plastic case

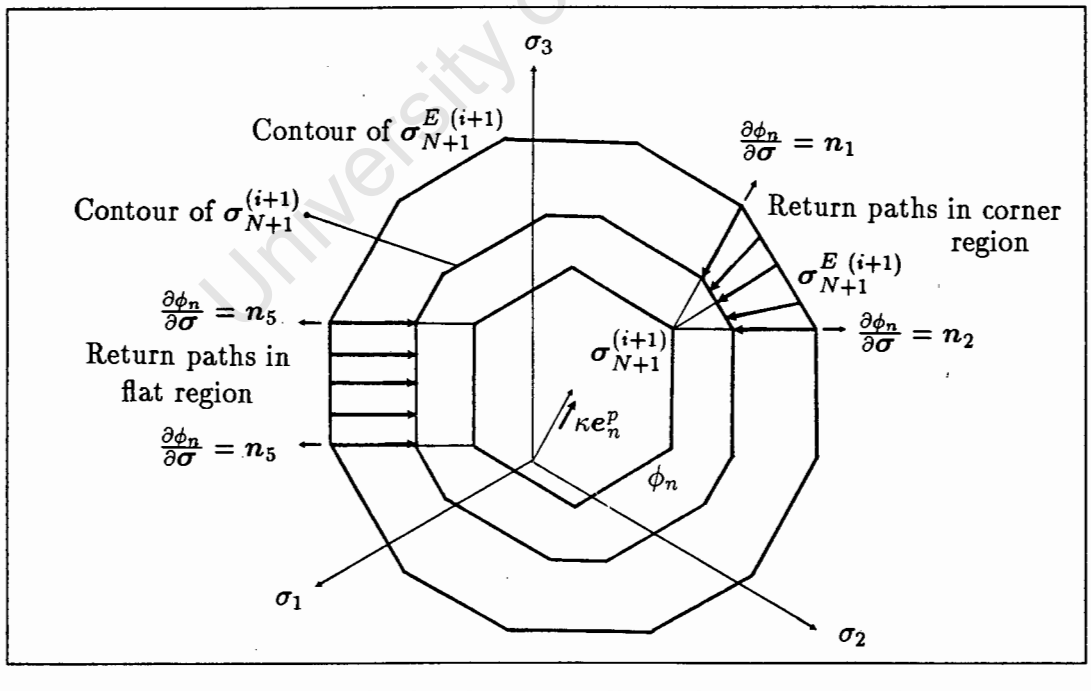

Figure 6.5: Return paths for the Tresca linear kinematic hardening case

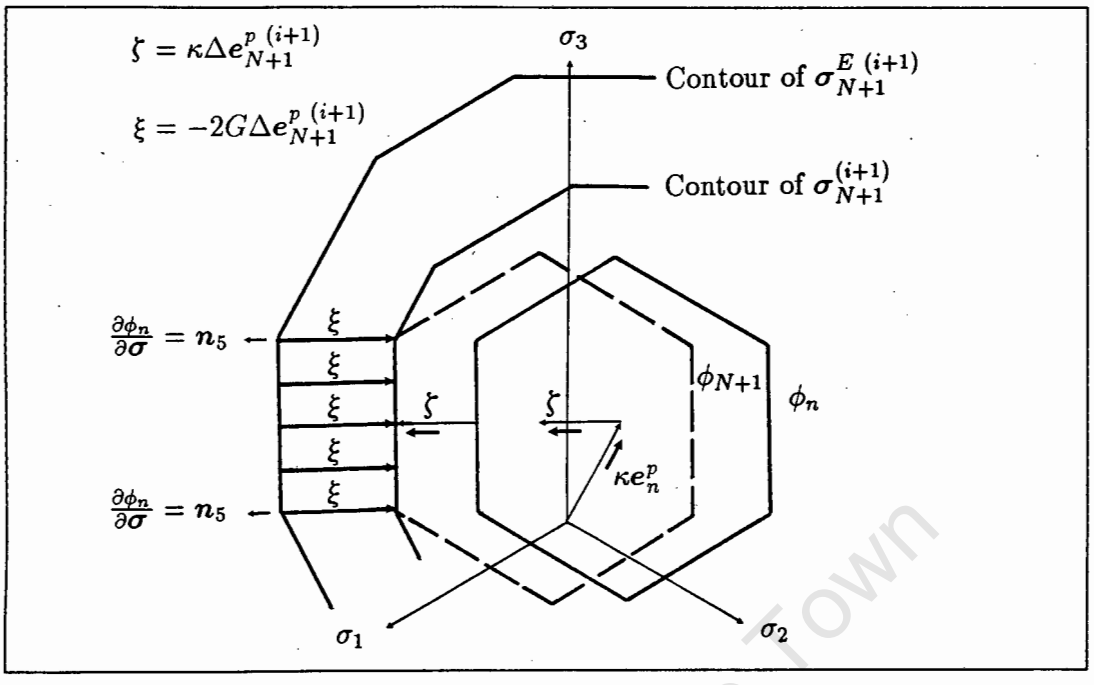

Figure 6.6: Subsequent yield surface for return paths in a flat region (LKH)

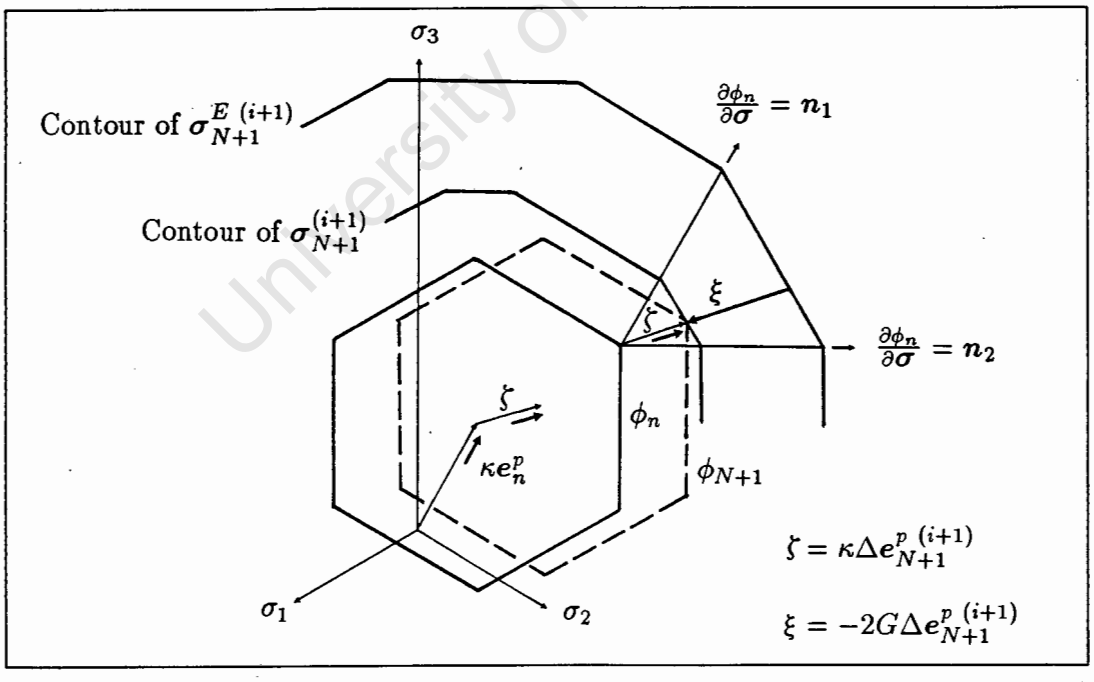

Figure 6.7: Subsequent yield surface for return paths in a corner region (LKH)

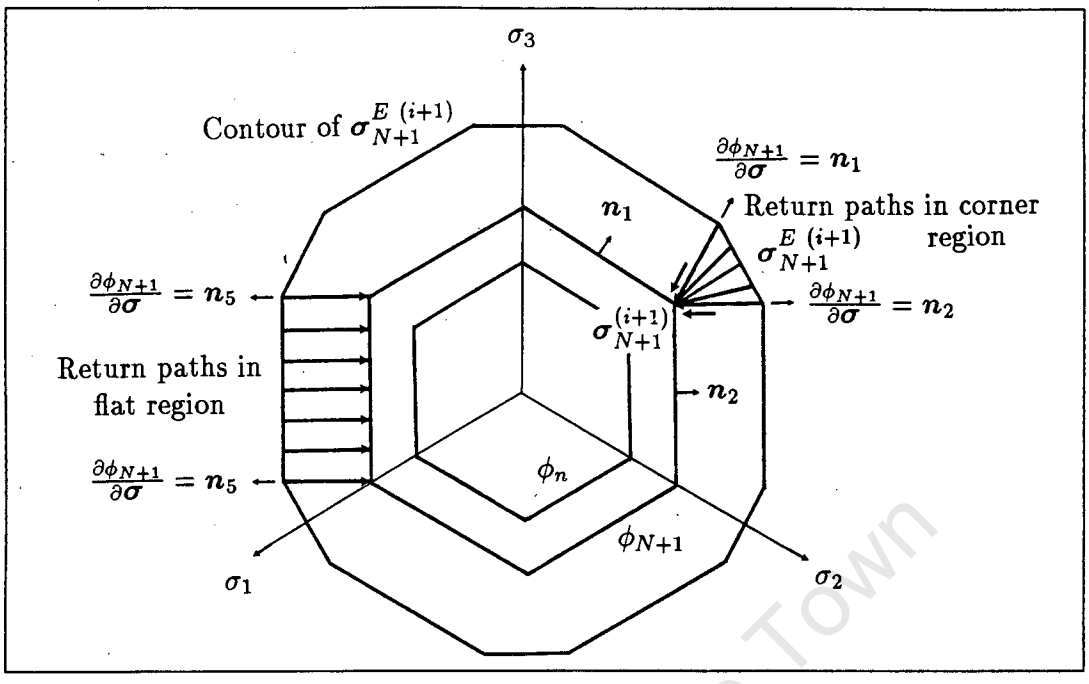

Figure 6.8: Return paths for the Tresca linear isotropic hardening case

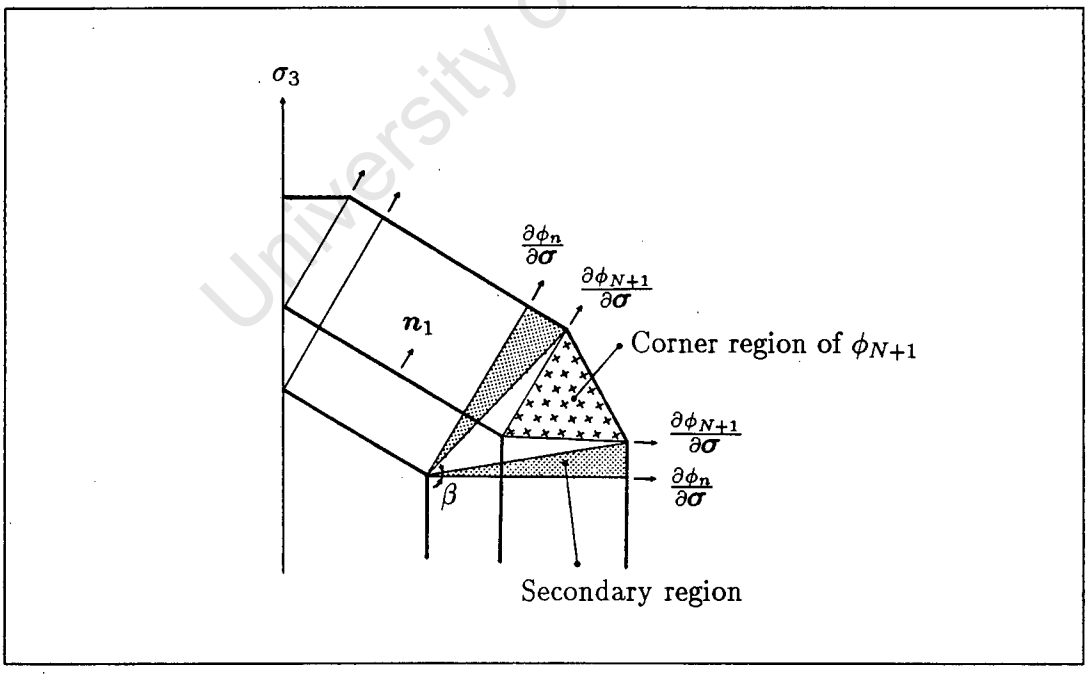

Figure 6.9: Corner detail for the Tresca linear isotropic hardening case

6.5.2 The Mohr-Coulomb yield surface

The piecewise linear surfaces of $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ associated with the Mohr-Coulomb yield surface are infinitely long 12-sided cones in principal stress space with axes coinciding with the current yield surface axis. Again, we divide the area around the appropriate yield surface into flat and corner regions, defined now by vectors $\boldsymbol{m}_i = \boldsymbol{D}\boldsymbol{n}_i$ at the corners of the appropriate yield surface. Due to the symmetry of the yield surface, we need only make a distinction between flat and corner regions, and not between individual flat regions or individual corner regions respectively. We further define an apex region defined by vectors $\boldsymbol{m}_i$ at the apex. These three region types are identified in Figure 6.10.

The perfectly plastic case

For a perfectly plastic solid the yield surface remains fixed in principal stress space for all time increments. Thus the current and subsequent yield surface is the initial yield surface ϕ_0.

A surface of elastically predicted stress points $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ is illustrated in Figure 6.11. This surface, together with the flat, corner and apex regions, is constructed with reference to the *initial* yield surface ϕ_0.

The return paths from this surface of $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ are also illustrated in Figure 6.11 and are characterized as follows. All $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on the surface within a flat region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ on the flat surface of the yield surface ϕ_0 along the vector $\boldsymbol{m}_i$ at $\boldsymbol{\sigma}_{N+1}^{(i+1)}$. All $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on the surface within a corner region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ at the corner of the yield surface ϕ_0 along the vector $\boldsymbol{D}\partial\phi_0/\partial\boldsymbol{\sigma}$ at the corner passing through the particular $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$. All $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ on the surface within the apex region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ at the apex of the yield surface ϕ_0.

These return paths correspond with the return paths, and the apex region coincides with the inverted pyramid, of Pankaj & Bićanić [1].

The linear isotropic hardening case

We now consider a surface of elastically predicted stress points $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ constructed with reference to the *subsequent* yield surface ϕ_{N+1}. We also define the flat, corner and apex regions with reference to the subsequent yield surface ϕ_{N+1}. The return paths from this surface of $\boldsymbol{\sigma}_{N+1}^{E\,(i+1)}$ are illustrated in Figure 6.12 and are characterized below.

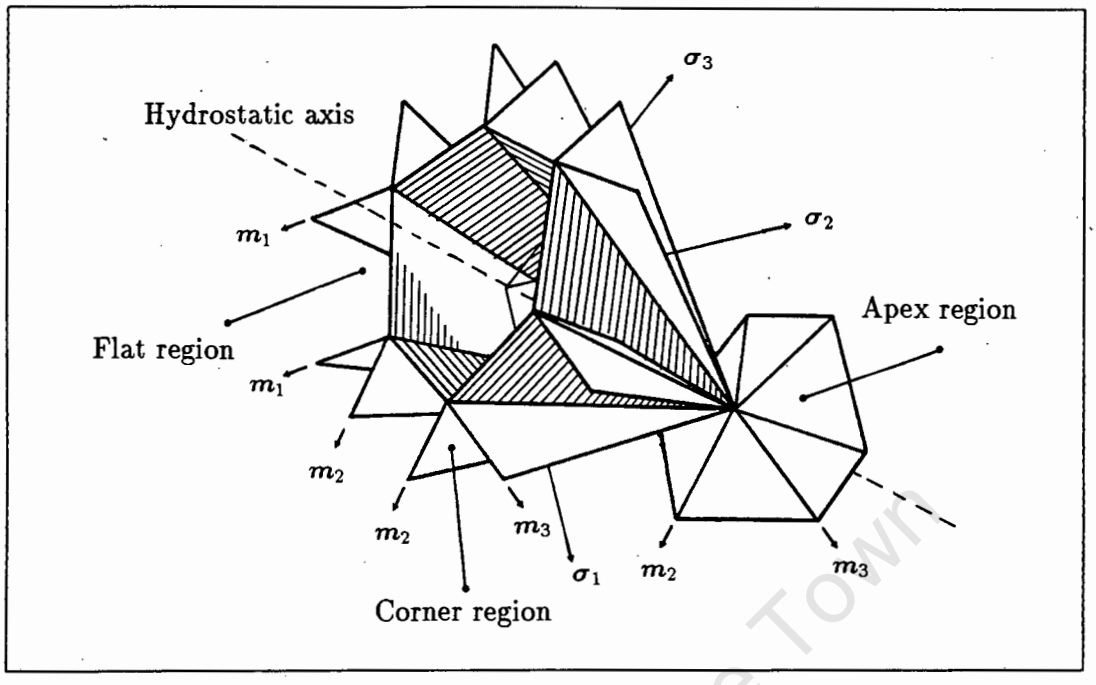

Figure 6.10: Corner, flat and apex regions of the Mohr-Coulomb yield surface

All $\boldsymbol{\sigma}_{N+1}^{E\ (i+1)}$ on the surface within a flat region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ on the flat surface of the subsequent yield surface along the vector $\boldsymbol{m}_i$ at $\boldsymbol{\sigma}_{N+1}^{(i+1)}$. All $\boldsymbol{\sigma}_{N+1}^{E\ (i+1)}$ on the surface within a corner region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ at the corner of the subsequent yield surface ϕ_{N+1} along the vector $\boldsymbol{D}\partial\phi_{N+1}/\partial\boldsymbol{\sigma}$ at the corner passing through the particular $\boldsymbol{\sigma}_{N+1}^{E\ (i+1)}$. All $\boldsymbol{\sigma}_{N+1}^{E\ (i+1)}$ on the surface within the apex region return to $\boldsymbol{\sigma}_{N+1}^{(i+1)}$ at the apex of the yield surface ϕ_{N+1}.

The yield surface expands in stress space during the time increment with no change in the friction angle θ. Thus the corners of the subsequent yield surface lie on radial lines through the corners of the current (and initial) yield surface, and the apex of the subsequent yield surface lies on the hydrostatic axis, but at a now greater distance from the origin.

This expansion of the yield surface in stress space can be measured as an increase in the cohesion $\Delta c_{N+1}^{(i+1)}$ and is quantified as the largest of

$$\Delta c_{N+1}^{(i+1)} = \pm \frac{GG_T}{G - G_T} \left[\frac{(\Delta\epsilon_{1\ N+1}^{p\ (i+1)} - \Delta\epsilon_{2\ N+1}^{p\ (i+1)})}{cos(\theta)} \right]$$

$$\Delta c_{N+1}^{(i+1)} = \pm \frac{GG_T}{G - G_T} \left[\frac{(\Delta\epsilon_{1\ N+1}^{p\ (i+1)} - \Delta\epsilon_{3\ N+1}^{p\ (i+1)})}{cos(\theta)} \right]$$

$$\Delta c_{N+1}^{(i+1)} = \pm \frac{GG_T}{G - G_T} \left[\frac{(\Delta \epsilon_{2\ N+1}^{p\ (i+1)} - \Delta \epsilon_{3\ N+1}^{p\ (i+1)})}{cos(\theta)} \right] \tag{6.37}$$

for the flat regions only.

We can make use of eqn. (6.37) to construct the concentric surfaces of $\boldsymbol{\sigma}_{N+1}^{E\ (i+1)}$ about the current yield surface. From these we can then identify the subsequent yield surface ϕ_{N+1} and hence the stress point $\boldsymbol{\sigma}_{N+1}^{(i+1)}$.

6.6 CONCLUSIONS

Firstly, the quadratic programming problems of both the Tresca and Mohr-Coulomb yield surfaces with perfectly plastic and linear hardening material behavior are considered in detail.

Secondly, the return paths, associated with the traditional return mapping algorithm, are identified in principal stress space from the quadratic programming problems. These return paths provide a correct basis against which heuristically developed algorithms can be compared.

Thirdly, provided the yield surface and the elastically predicted stress are suitably formulated in stress space (equivalent to the appropriate holonomic formulation in conjugate force space), the return paths in principal stress space are common to both the backward difference and generalized trapezoidal rules.

REFERENCES

1. Pankaj & Bićanić, N., On Multivector Stress Returns in Mohr-Coulomb Plasticity, *Computational Plasticity: Models, Software and Applications*, (D.R.J. Owen, E. Hinton & E. Onate *eds.*), Part 1, Pineridge Press, 421-436, 1989.

2. Bird, W.W. & Martin, J.B., A secant approximation for holonomic elastic plastic incremental analysis with a Von Mises Yield Condition, *Engineering computations*, **3**, 192-201, 1986.

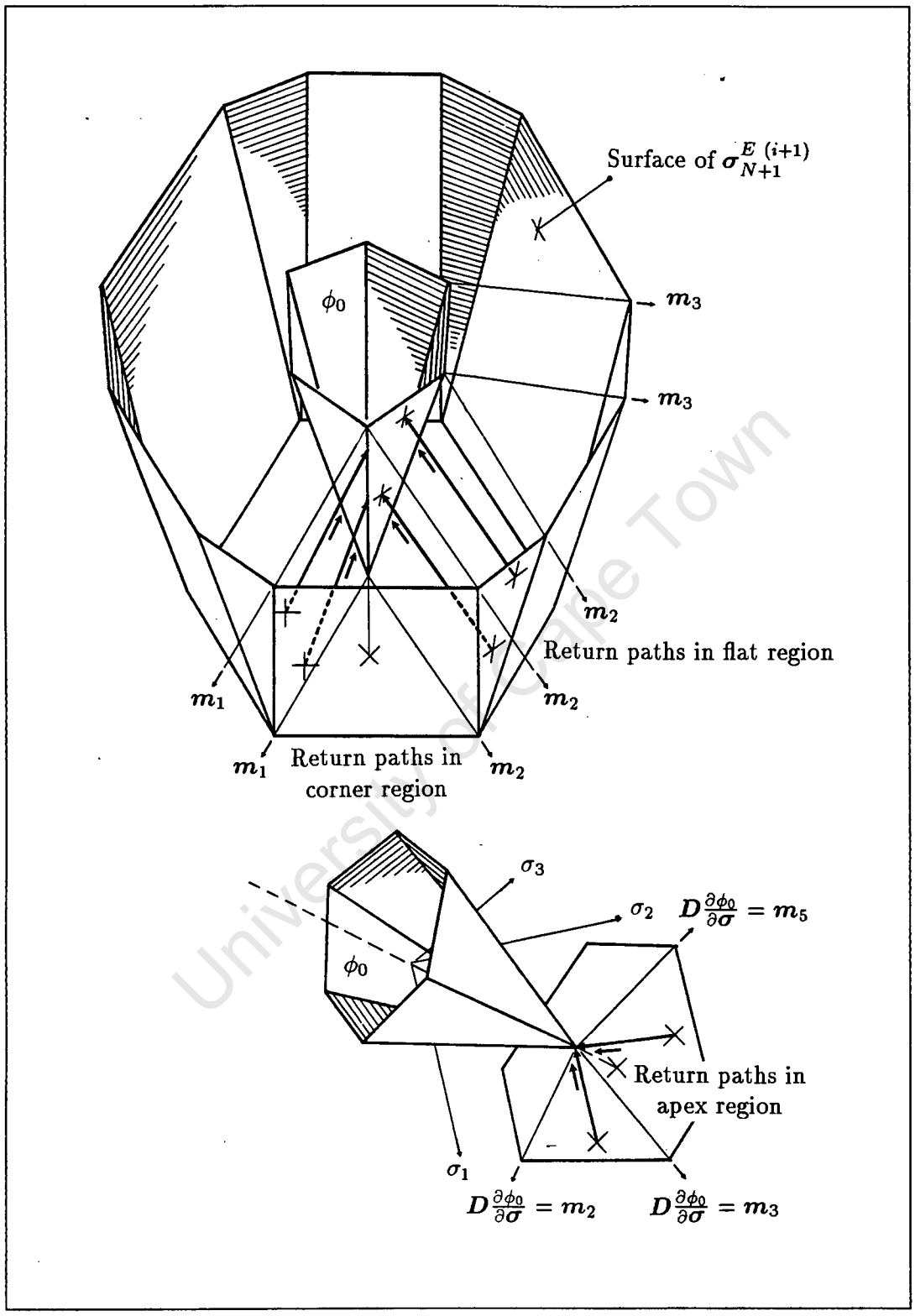

Figure 6.11: Return paths for the Mohr-Coulomb perfectly plastic case

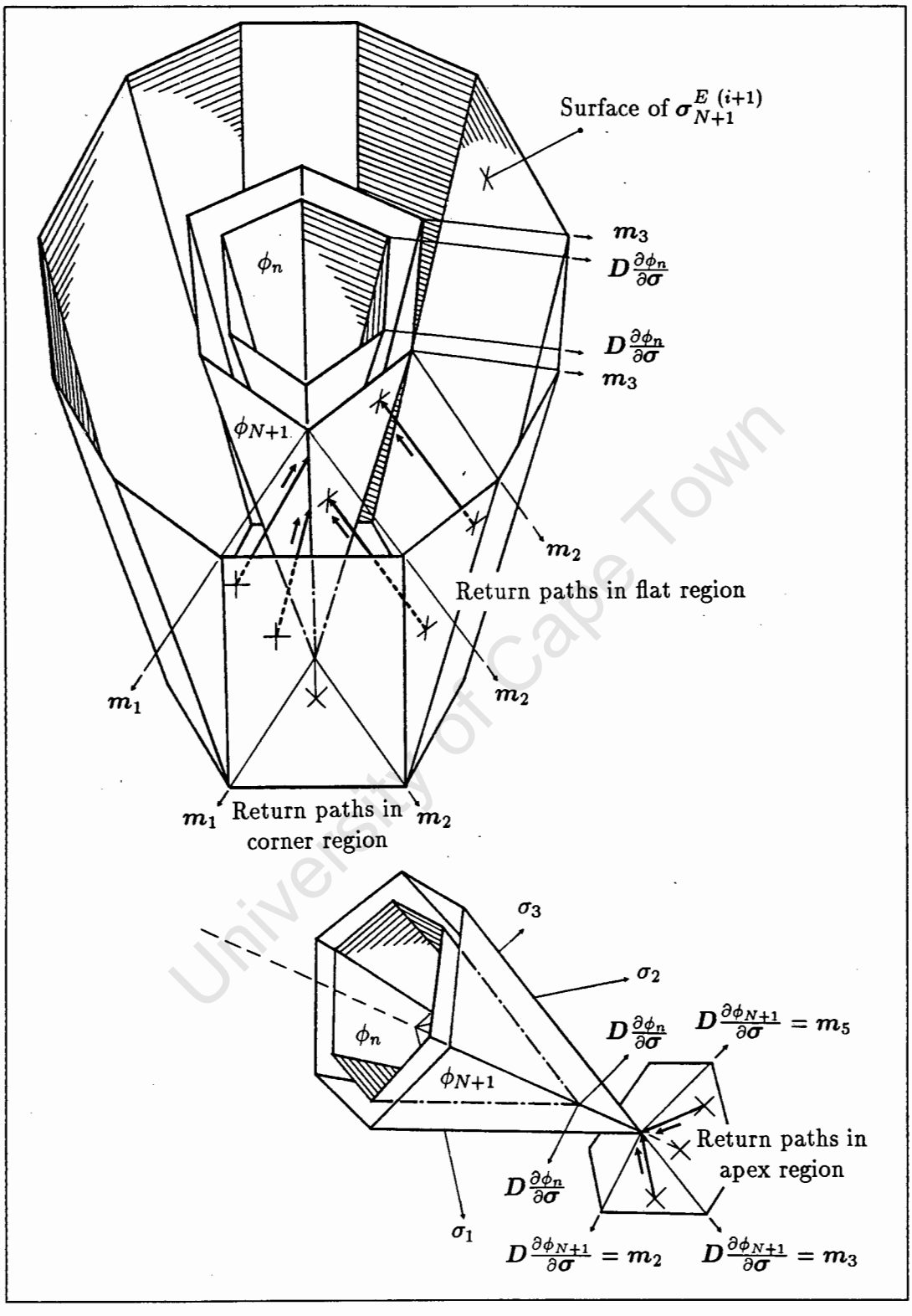

Figure 6.12: Return paths for the Mohr-Coulomb linear isotropic hardening case

CHAPTER 7
CONCLUSION

The internal variable formulation of the incremental problem for convex elastic-plastic materials presented in this thesis has facilitated the development of a generalized trapezoidal rule that is consistent with a generalized trapezoidal rule for creep. Further, it has provided the understanding of the relationship between this and other integration rules in terms of the governing principles. Indeed, it is shown that the generalized trapezoidal rule and the generalized midpoint rule of Simo *et al* [1,2] are fully equivalent in that the same minimum principle is established in each case. The generalized trapezoidal rule thus inherits the notion of B-stability and offers the opportunity to exploit the second order rate of convergence for $\alpha = \frac{1}{2}$ associated with the generalized midpoint rule.

The generalized midpoint rule and the generalized trapezoidal rule can also be regarded as equivalent generalizations of the backward difference scheme. The essential advantageous characteristics of the backward difference formulation (amongst others, the symmetry of the consistent tangent modulus) are thus retained under these rules. However, the generalized trapezoidal rule may be regarded as more convenient than its generalized counterpart, in the sense that the equilibrium and constitutive equations are fully satisfied at the end of each interval rather than at the generalized midpoint.

The internal variable formulation of a backward difference corrector algorithm for piecewise linear plasticity provides us with a consistent algorithm for the integration of the constitutive equations. The algorithm is written in the form of a mathematical programming problem and is fully consistent in that no heuristic assumptions are made. It further has the advantages of being fully linked to the governing principles and avoiding the inherent problems associated with corners under the classical formulation.

For the particular cases of the Tresca and Mohr-Coulomb yield surfaces, by first formulating the problem as a holonomic problem in reference space, a convex quadratic programming problem can be established in principal space. The backward difference algorithm can then be extended to include the generalized trapezoidal rule by simply modifying the holonomic step. This allows the general structure of the backward difference algorithm to be maintained, in the sense that the same quadratic programming problem is solved in both cases. In this way, the computational advantages of the generalized trapezoidal rule can be exploited.

Finally, as the corrector algorithm is fully consistent, the quadratic programming problems developed for the Tresca and Mohr-Coulomb yield surfaces are used to provide a basis against which heuristically developed algorithms can be compared. This is done in the form of the classical formulation of a return mapping algorithm in which the return paths in principal stress space associated with an elastically predicted stress are identified.

REFERENCES

1. Simo, J.C. & Govindjee, S., Nonlinear B-Stability and Symmetry Preserving Return Mapping Algorithms for Plasticity and Viscoplasticity, *International Journal for Numerical Methods in Engineering*, **31**, 151-176, 1991.

2. Simo, J.C. & Taylor, R.L., A Return Mapping Algorithm for Plane Stress Elastoplasticity, *International Journal for Numerical Methods in Engineering*, **22**, 649-670, 1986.