Computational modelling of critical points and structural discontinuities.

Curiel Sosa, Jose Luis

CIVIL AND COMPUTATIONAL ENGINEERING CENTRE
UNIVERSITY OF WALES SWANSEA

COMPUTATIONAL MODELLING OF CRITICAL POINTS AND STRUCTURAL DISCONTINUITIES

José Luis Curiel Sosa
Master of Research, School of Engineering, University of Wales Swansea
Ingeniero Superior Industrial, School of Engineering, University of Seville, Spain

Dissertation submitted to the University of Wales Swansea in fulfilment of
the requirements for the degree of Doctorate of Philosophy of Engineering
University of Wales Swansea, 2006

Civil and Computational Engineering Research Centre (C 2 EC), UWS

CENTRES OF EXCELLENCE
CANOLFANNAU RHAGORIAETH

Centres of Excellence Wales

Summary

An algorithm combining the numerical execution in time of implicit and explicit methods of solution is presented in this thesis. The algorithm swaps between both methods as required for the analysis. The whole mesh is solved for an unique method at once, i.e. there are no partitions of the mesh for separate implicit or explicit treatment of the solution. The combination is in-time, in such a manner that if the implicit method diverges at some time-point the explicit one is initiated by appropriate conditions of transition.

A subcycling algorithm is proposed to improve the solution of nonlinearities associated to the interface in reinforced materials. Dividing the execution by two subcycles, the smallest critical time step does not control the other one. Thus, the computational cost is reduced and allows an efficient model of the interfacial discontinuity. In addition, a beam element is developed for the modelling of the reinforcement.

Acknowledgements

I would like to express my gratitude to my supervisor Prof DRJ Owen for his efforts in directing and supervising this research. His advice and keeping the goals in expectative are very much appreciated.

A big thanks to Dr EA de Souza Neto for his help and guidance. I am particularly grateful to Prof J Bonet for his generous help. To Prof D Peric for such a great module in *Computational Plasticity.* To Dr P Nithiarasu for many useful advices and fantastic module in *Computational Fluid Dynamics.* To Dianne Cook for solution of problems in the systems. To Jackie and L Jenkins for help with administrative issues.

Special thanks are reserved to my wife, Mariló, for support and help, without it, this thesis would have not been possible, and our children, Iván and Mónica. Their births provoked such an enthusiasm and extra motivation. Also, to my parents, to Manuel Vaz and Lola for understanding.

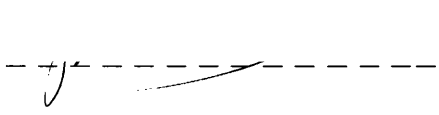
Personal thanks to Antonio Gil, for sharing his knowledge, Arturo Carneiro, for technical chats and making me laugh, to Javier Silla, for many coffee breaks and for proving such an outstanding contender at the gym, to José Luis (Padilla), for being always there, and Raimundo, for smart conversation. Their friendship made my stay in the C^2EC enjoyable and comfortable.

To Aurelio Arranz for friendship and mathematical issues. To Alejandro for fantastic walks in the Swansea's surroundings. To Liébana for friendship in the distance. To Daniel Belloso for his particular way of seeing things.

To Cristina, Africa, Bernita, Clare, Jay, David, Chris, Leon, Prem, Belen, Nick, Belgim, Leigh and Angeles, for good moments in *The Rhyddings*, *The Uplands Tavern* or at parties. To Miguelón, Antonio Vergara, Mari Carmen, Dani, Yoli, Leo and Conchi for many good moments on holidays.

DECLARATION

This work has not previously been accepted for any degree and is not being concurrently submitted in candidature for any degree.



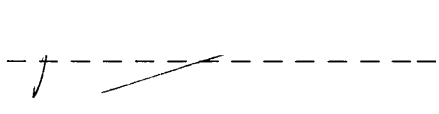
31/10/2006

Candidate Date

STATEMENT I

This thesis is the result of my own investigations, except where otherwise stated. Where correction services have been used, the extent and nature of the correction is clearly marked in a footnote(s).

Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

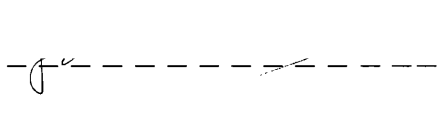


31/10/2006

Candidate Date

STATEMENT II

I hereby give consent for my dissertation, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.



31/10/2006

Candidate Date

To Mariló,

Iván and Mónica

Contents

List of Figures

List of Tables

Chapter 1

Introduction

'...but I'm only interested in things that don't work'.

M. A. Crisfield

1.1 Motivation and objectives

A new in-time implicit/explicit algorithm

The main motivation for this research comes out from the necessity of solving convergence problems in algorithms used by finite element solvers (FES) putting emphasis in the computational cost that these imply. The source of such a divergence is associated generally to critical points. Thus, a review of current methods and their behaviour in the presence of critical points in structural analysis has been initially carried out. Physically, a structure is uncontrollable or partially controllable in these singularities in which the relationship between characteristic load and the associated deflection is not unique -e.g. buckling points such as snap-through or snap-back . It is remarkable that there is not element-partitioning or nodal partitioning in the mesh for sepa-

rate treatment of the solution. Therefore, that is what makes it a novelty as the vast amount of techniques, reading as implicit-explicit or explicit-explicit, considers one or several partitions of the mesh by subdomains and a method of solution is assumed for every subdomain. The proposed I/E algorithm does not make any partitions of the mesh. The entire mesh is solved in a computational time point by either an explicit or an implicit method of solution. That is the reason to name it *in-time*. Many commercial FE software packages have chosen numerical methods of solution integrators in time conditionally stable. Of course, these codes may solve highly nonlinear problems with a variety of singularities. However, the CPU time is often expanded. This aspect has been taken in account in the algorithms developed in this research, i.e. a combination of efficiency and time-saving techniques.

A novel subcycling technique for reinforced materials

The modeling of reinforced materials is a problem of big interest in engineering practice, in particular by the economic effort made in the development of prototypes or physical models to test specific structural components or systems. Among these reinforced materials we can find composites in which the fibre plays a role of high strength to traction forces such as the case of tanks reinforced helicoidally with glass fibre and with a matrix made by polyester resin. There are also hybrid composites that combine the traditional composite layers with layers of metal alloys, extensively used in the aerospace industry. And of course, the reinforced concrete of utmost importance in civil engineering. The last chapter of this dissertation will be centered in this one. However, the subcycling algorithm presented in Chapter 5 may be treated in a general manner for other reinforced materials changing the chosen material model for

matrix and reinforcement. This makes the algorithm developed attractive for the modeling of these composite materials. Other type of nonlinear problems containing boundaries between two distinct materials in contact or interface are also targeted in this study. These are of interest in the solution of structures formed by reinforced materials. The challenge offered by these materials, in terms of nonlinearity introduced and sources of divergence, is due to the geometric nonlinear interface, the nonlinear constitutive law bond stress vs. relative displacement in the constituent surfaces and the jump induced by the different nature of the materials involved. To solve this, an algorithm divided in execution-subcycles one for the matrix and the another one for the reinforcement has been developed. The entire domain of the composite materials is partitioned in two groups of elements. Each material, which is considered to be a subdomain, may be formed by different elements and material models. Thus, matrix and reinforcement constitute two separate systems of solution. They are connected through a new link interfacial model. The proposed approach by subcycles permits a computationally uncoupled treatment of the solution. Thus, the instability, caused by the interface in the transfer zone, is avoided. An additional advantage of this scheme is that two different solvers may be used for each of the subdomains. Then, the use of existent codes or algorithms for matrix and reinforcement is possible[1]. Thus, the novel in-time implicit/explicit algorithm (I/E) presented in Chapter 3 is now used for the matrix. An explicit method (E) is used for the reinforcement. The reinforcement is modelled by continuum-based beam elements which are convenient from the point of view of stability and velocity of convergence. These elements

[1]The subcycling algorithm may referred to in some parts of this dissertation by the characters I/E-E.

are a modification of those proposed by [11]. A complete formulation for these elements is exposed in Chapter 4. Additionally, a constitutive law is proposed for the modelling of the interface. This law is formed by an initial nonlinear part followed by a residual bond stress:

1. The interface in reinforced materials such as reinforced concrete is considered a transfer zone of kinematic and kinetic variables. Physically, there is not a specific material constituting it. However, we need still to take in account the link conditions that the interface causes.

2. The solutions for each subdomain are separate and executed with different computational time steps. Therefore, there must be links interconnecting the subcycles and enforcing the conservation laws in the interface.

3. The interface does not embody material mass as the surface (3D) or thickness (2D) associated to the interface is negligible. Interface forces are represented by an array of link forces which influence the governing equations of every material.

These algorithms are exposed in two-dimensional formulations. A three-dimensional formulation does not change the core of the algorithms developed. Instead, it would bring more effort without any additional progress except, of course, to validate in 3D. Such a development has been avoided for time reasons and focus is addressed to other areas considered more relevant to this investigation.

1.2 Layout

In the following a brief resume of the contents of every chapter is outlined:

- Chapter 2, description of basic principles of continuum mechanics, fundamentals of finite element methods (FEM) and computational aspects involved are summarized including some remarks and demonstrations, in which the theoretical framework for the remaining research is established.

- Chapter 3, an in-time implicit-explicit algorithm and its formulation is developed. In the first part of the chapter, the algorithm is presented and its constituents or sub-algorithms are described. Analysis concerning the role of damping in static or quasi-static explicit finite element implementations are studied. In addition, a brief review of the constitutive models, such as Von-mises, used for the algorithm is included. It is continued by key numerical examples involving buckling in a range of structures and materials. Analyses concerning the role of damping in static or quasi-static explicit finite element implementations are studied.

- Chapter 4, a beam element based in a continuum quadrilateral finite element is formulated. This will be called, henceforth, continuum-based beam element (CBE). Its development is justified because of the necessity of beam elements for the reinforcing bars which will be used in the next chapter in the modelling of the reinforcement. Numerical tests over bars and cantilever beams are exposed at the end of the chapter.

- Chapter 5, a subcycling algorithm for the resolution of reinforced materials is proposed. It involves the separation of processing by subcycles, for the matrix and reinforcement. The matrix is modeled either as an elastic material or with the Mohr-Coulomb constitutive model. Benchmark tests are conducted with these two material models. These are compared with tests proceeding from other numerical and experimental results. An

interface constitutive law has been developed and compared with others from the literature.

- Chapter 6, remarks the results drawn from the research, outlines the limitations of the algorithms presented and suggests new related lines of investigation.

Chapter 2

Computation of Finite Element Methods (FEM) in Solid Mechanics

'Space, time, and force are a priori *forms; they can be derived only from contemplation and from general principles of research. Their common relation to each other in mechanisms must be regarded as something inspired indeed by experience but in its generality fixed by convention'.* **Hamel**; *Elementare Mechanik*; 1912.

2.1 Introduction

A description of the basic principles, which are the fundament of this study, is essential. The problem may be decomposed in levels as follows:

- In a first stage, statement of the differential equations and boundary conditions that govern the physical problem . That is usually designated

as *Strong Form* of the problem in the literature [11]. This form is the start point for the *Weak Formulation.*

- *Weak Form* of the problem through *The Principle of Virtual Work.*
- Discretisation of the weak form by finite elements in order to implement numerically in a computer for later analysis. Discretising it in finite elements and proceeding to direct temporal integration with an explicit method.
- In order to deal with certain nonlinearities, a combined implicit/explicit scheme in time is developed and exposed in next chapter.

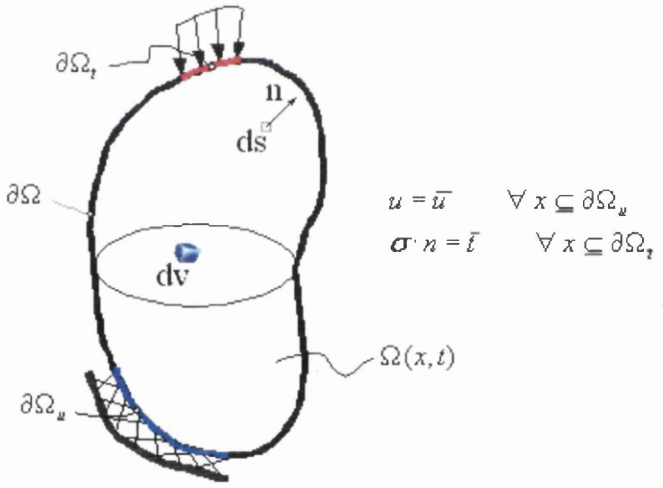


Figure 2.1: Generic domain body and boundary conditions

2.2 An overview on finite element methodologies

The literature concerning this topic is wide and has dramatically increased year after year. Some of the reviewed studies are indicated below. A good first introductions in linear finite element methods are [23, 71, 83, 91]. Basic references undertaking a wide group of topics related to FEM are K-J. Bathe[5], P. E. Lewis et al. [71], O. C. Zienkiewicz and R. L. Taylor [131, 130], T. Belytschko et al. [11] or T.J.R. Hughes [58] amongst others. The first volume of Zienkiewicz[131] treats basic linear problems. The second one touches special problems in solid mechanics as material nonlinear problems or large deformation analysis. In Belytschko's text, we can find some guidance to program the FEM either in an implicit or an explicit form. A good introduction to nonlinear aspects is performed through the lagrangian, eulerian, and also the updated lagrangian finite elements discretisation (convenient in large deformation analysis when the configuration is changing dramatically, every variable is updated each iteration; but the computational cost is dramatically increased).

T.J.R Hughes [58] is devoted to questions of stability, so important in the configuration of the explicit implementation of the method in the present study, as the convergence depends on it. An analysis is performed in the dynamic field, which deals with the aspects of calculation of the critical time step, giving a sort of them for different elements. Irons [61] presents a critical time step expression for plane strain, which seems to have been used in many works. However, this constant value for the critical time step usually drives to perform constant time step along the execution which is not the most efficient if we want the smallest time of computer execution. Thus, in Section(3.7.4), an adaptive time step is

formulated together with details of implementation. Bicanic [14] presents the time discretisation of the dynamic equilibrium equations using isoparametric elements, and presenting stability limits depending on the type of element.

A practical reference treating with programming of the FEM in detail is I. M. Smith and D. V. Griffiths [110] and involving plasticity coding are the works of D.R.J.Owen [85] or E. de Souza et al. [116] . A work devoted to geometrically nonlinear problems and large deformation of hyperelastic materials doing emphasis in programming aspects is the text of J. Bonet et al. [17] or Holzapfel [57]. In Simo and Hughes' text [106], material nonlinearities are treated including the seminal works over integration algorithms for the updating of stresses and internal variables in incremental problems. Basic theory to understand the continuum media treating nonlinear issues can be found in the texts of Malvern [74] or C.A. Felippa [41].

2.3 Deformation and motion

Our body Ω is defined as a group of particles or material points bounded by $\partial\Omega$ in the Euclidean space E^d. The displacement field is defined in the vectorial space $\aleph$ and a tensor will represent a linear transformation between two linear spaces. The placement of our body is a regular region $\Omega_0 \subset \aleph$ in the direction pointed in refs. [80] or [122]. A three-dimensional regular region is a closure of an open set whose boundary is the union of a finite number of surfaces with continuous normal field. A standard deformation of our body would be given by the mapping $\Upsilon : \Omega \longrightarrow E^d$, any material point is associated with its new position in E^d, $\Upsilon(\mathbf{X})$. It is assumed that Υ is continuous and, at least, piecewise continuously differentiable. The determinant of the deforma-

tion gradient $\nabla\Upsilon = \frac{\partial\Upsilon(\mathbf{X})}{\partial\mathbf{X}}$ is assumed positive to guarantee the transformation is preserving the orientation. The displacement of a point of the body is the difference between the current position and the reference position. Motions are time dependent, hence the full description of the displacement is rigourously expressed in Eq(2.1). Υ is assumed to be twice differentiable in time.

$$\mathbf{x} = \Upsilon(\mathbf{X}, t)$$

$$\mathbf{u}(\mathbf{X}, t) = \mathbf{x} - \mathbf{X} \tag{2.1}$$

At this stage it is convenient to introduce the two types of configurations generally adopted to describe the motion: the lagrangian or reference configuration and the Eulerian or spatial description. In the lagrangian configuration the independent variables are the reference configuration X and the time. Thus, an observer placed in a material point would move together at any time. In a Eulerian description, the observer would be situated in a point of the Euclidean space E^d, and would be able to follow the motion of the material points of the body $\Omega(t)$ without being fixed to any of the material points. The latter one is generally adopted in flow problems. That is the reason why spatial description is usually used in Computational Fluid Dynamics (CFD). Time and current configuration are the independent variables in a Eulerian description. In a finite element context, the mesh resulting from the discretisation deforms and moves with the material body in a lagrangian description. In the Eulerian one, the material points of the body are not fixed to the nodes, elements, etcetera, of the mesh. The description assumed in this dissertation is the lagrangian as evaluation of stresses at quadrature points will be always in the same material point independently of the motion. Therefore, velocities

and acceleration fields are defined as stated below (they are supposed to be sufficiently smooth).

$$\dot{\mathbf{u}}(\mathbf{X},t) = \frac{\partial \mathbf{u}(\mathbf{X},t)}{\partial t} \tag{2.2}$$

$$\ddot{\mathbf{u}}(\mathbf{X},t) = \frac{\partial^2 \mathbf{u}(\mathbf{X},t)}{\partial t^2} \tag{2.3}$$

The velocity is the rate of change of current placement of the material point, and the acceleration the rate of change of the velocity of the particle. These rates would be given in eulerian description through the inverse motion $\Upsilon^{-1}(\mathbf{x},t) = \mathbf{X}$ (see Eqns (2.4) (2.5)).

$$\mathbf{v}(\mathbf{x},t) = \dot{\mathbf{x}}_s(\Upsilon^{-1}(\mathbf{x},t),t) \tag{2.4}$$

$$\mathbf{a}(\mathbf{x},t) = \ddot{\mathbf{x}}_s(\Upsilon^{-1}(\mathbf{x},t),t) \tag{2.5}$$

where the subscript s is only to emphasize that they are defined by a spatial configuration. There are other types of configuration less general than the described, as updated lagrangian (where the derivatives are with respect to the eulerian coordinates and the integrals in the weak form are in the current or deformed domain)or corotational (used usually for large rotations) [11]. A corotational formulation is used for a formulation for beam elements in Chapter 4.

2.4 Strain and velocity-strain measures

In our problem, we shall deal with nonlinear problems, either geometrically or materially nonlinearities and also large deformation or finite strains analysis

have been carried out. In finite strains, the term $\nabla \mathbf{u}^T \nabla \mathbf{u}$ is a substantial quantity with respect to the others. The Green-Lagrange strain measure takes into account that term and keep the first requirement for any strain measure: it states any strain must vanish in any motion as rigid body. The Green or the Green-Lagrange strain $\mathbf{E}$ is defined in Eq(2.6).

$$ds^2 - dS^2 = 2d\mathbf{X} \cdot \mathbf{E} \cdot \mathbf{X} \tag{2.6}$$

where ds^2 and dS^2 are the infinitesimal squared lengths of an arch in the current and reference configurations respectively. The change in length that occurs in the deformation of the body is described by the elemental material vector $d\mathbf{X}$. Green strain can be written in terms of the deformation gradient as

$$\mathbf{E} = \frac{1}{2}(\mathbf{F}^T\mathbf{F} - \mathbf{I}) \tag{2.7}$$

with $\mathbf{F}$ the deformation gradient. This is demonstrated as follows. The deformation gradient is defined as the rate of change between the current and the initial or reference,

$$\mathbf{F} = \frac{d\mathbf{x}}{d\mathbf{X}}$$

and being

$$ds^2 = d\mathbf{x} \cdot d\mathbf{x} \qquad and \qquad dS^2 = d\mathbf{X} \cdot d\mathbf{X}$$

writing now in notation with indices

$$ds^2 = dx_i dx_i = F_{il} dX_l F_{ik} dX_k = dX_l F_{il} F_{ik} dX_k = \quad \longrightarrow$$
$$= dX_l F^T_{li} F_{ik} dX_k = d\mathbf{X} \cdot \mathbf{F}^T\mathbf{F} \cdot d\mathbf{X}$$

$$dS^2 = d\mathbf{X} \cdot d\mathbf{X} = d\mathbf{X} \cdot \mathbf{I} \cdot d\mathbf{X}$$

taking in account Eq(2.6),

$$d\mathbf{X} \cdot \mathbf{F}^T\mathbf{F} \cdot d\mathbf{X} - d\mathbf{X} \cdot \mathbf{I} \cdot d\mathbf{X} = 2d\mathbf{X} \cdot \mathbf{E} \cdot \mathbf{X}$$

and that must be true for all $d\mathbf{X}$, thus, it finishes demonstrating Eq(2.7),

$$\mathbf{F}^T\mathbf{F} - \mathbf{I} - 2\mathbf{E} = 0$$

This strain measure can be written in function of the displacement gradients (see Eq(2.8))

$$\mathbf{E} = \frac{1}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T + \nabla\mathbf{u}^T \cdot \nabla\mathbf{u}\right) \tag{2.8}$$

If an infinitesimal strain has been enforced in the body, the term $\nabla\mathbf{u}^T \cdot \nabla\mathbf{u}$ is small enough with respect to the other ones than can be dropped. The strain tensor for small strains, in well-known form, is then obtained. Analogously, in Eulerian or spatial description the strains are represented by the Almansi tensor where the same scalar product is now referenced to the spatial vector dx,

$$d\mathbf{x} \cdot d\mathbf{x} - d\mathbf{X} \cdot d\mathbf{X} = 2d\mathbf{x} \cdot \mathbf{e} \cdot d\mathbf{x}$$

$$\mathbf{e} = \frac{1}{2}\left(\mathbf{I} - (\mathbf{F}\mathbf{F}^T)^{-1}\right)$$

Remark: the strain measures in spatial and reference configurations can be related through an operator Ξ [11, 17]

$$\mathbf{e} = \Xi(\mathbf{E}) = \mathbf{F}^T \cdot \mathbf{E} \cdot \mathbf{F}^{-1} \longrightarrow \quad 'push - forward'$$

$$\mathbf{E} = \Xi^{-1}(\mathbf{e}) = \mathbf{F}^T \cdot \mathbf{e} \cdot \mathbf{F} \longrightarrow \quad 'pull - back'$$

Remark: to show that the Green-Lagrange strain measure vanishes in a rigid body motion, we will consider a rigid rotation of our body Ω without deformation, i.e. without change in volume and/or distortion. The angular velocity of this motion will be called $\dot{\alpha}$. The current position of Ω, at time t,is

$$\mathbf{x}(t) = \mathbf{R}\mathbf{X}(t)$$

where R is the rotation matrix at time t,

$$\mathbf{R} = \begin{pmatrix} cos(\dot{\alpha}t) & -sin(\dot{\alpha}t) \\ sin(\dot{\alpha}t) & cos(\dot{\alpha}t) \end{pmatrix}$$

Then, the Green-Lagrange strain is (taking in account the definition given above and the orthogonality of the rotation tensor),

$$\mathbf{E} = \frac{1}{2}\left(\mathbf{F}^T\mathbf{F} - \mathbf{I}\right) = \frac{1}{2}\left(\left(\frac{\partial \mathbf{x}}{\partial \mathbf{X}}\right)^T\left(\frac{\partial \mathbf{x}}{\partial \mathbf{X}}\right) - \mathbf{I}\right) = \frac{1}{2}\left(\mathbf{R}^T\mathbf{R} - \mathbf{I}\right) = 0$$

The velocity gradient is defined in Eq(2.9) where v is as stated in Eq(2.2) for lagrangian description, and the velocity-strain in Eq(2.10) which is measure of the rate of square length change Eq(2.11).

$$\mathbf{L} = \frac{\partial \mathbf{v}}{\partial \mathbf{x}} = (\nabla \mathbf{v})^T \tag{2.9}$$

$$\mathbf{D} = \frac{1}{2}(\mathbf{L} + \mathbf{L}^T) \tag{2.10}$$

$$\frac{\partial}{\partial t}(ds^2) = \frac{\partial}{\partial t}(d\mathbf{x}(\mathbf{X}, t) \cdot d\mathbf{x}(\mathbf{X}, t)) = 2d\mathbf{x} \cdot \mathbf{D} \cdot d\mathbf{x} \tag{2.11}$$

2.5 Forces

The former section describes the deformation. The concepts of velocity gradient, strain measures or deformation gradient have been introduced without reference to the force, traction or pressure. We can distinguish the following types of interactions,

- Forces per unit area applied in the boundaries of the body. They are interactions between the interior of the body and the exterior and, hence, transmitted through the boundary of the body.
- Body forces such as electromagnetic fields or gravitational fields which interact over all material points of the body under the effect of these fields.
- Internal forces due to the molecular interactions.

2.6 Stress measures

2.6.1 Cauchy's axiom

This axiom states that *at a point* $\mathbf{x}$ *the force per unit area over a surface* S *of the body and normal* $\mathbf{n}$*, depends on* S *only through its normal.* Therefore, any surfaces with normal $\mathbf{n}$ at $\mathbf{x}$ possess the same force per unit area. This force per unit area is called the Cauchy stress vector $\mathbf{t}(\mathbf{x}, \mathbf{n}, t)$. If this surface belongs to the boundary then it represents the force exerted by the surroundings of the body such as contact forces.

2.6.2 The Cauchy stress tensor

As an implication of the momentum balance (see Section 2.7.2), the surface force vector $\mathbf{t}$ is linearly dependent on the normal $\mathbf{n}$ to the surface in which is applied $\mathbf{t}$. It implies that exists a symmetric stress tensor (*Cauchy stress tensor or true stress tensor*) $\boldsymbol{\sigma}(\mathbf{x})$ such that the stress vector $\mathbf{t}(\mathbf{x}, \mathbf{n})$ is given by Eq (2.12).

$$\mathbf{t}(\mathbf{x}, \mathbf{n}) = \boldsymbol{\sigma}(\mathbf{x}) \cdot \mathbf{n} \tag{2.12}$$

Remark: Forces are transmitted trough discrete atomic interactions, hence, the continuum mathematical representation by means of a stress tensor is only valid for a representative scale to enforce the assumptions of continuum mechanics. This scale is often represented by a representative volume element.

$$\boldsymbol{\sigma}(\mathbf{x}) = \sigma_{\mathbf{ij}}\, \mathbf{e_i} \otimes \mathbf{e_j} \tag{2.13}$$

The Cauchy stress tensor in an orthonormal basis $\{\mathbf{e_1}, \mathbf{e_2}, \mathbf{e_3}\}$ is represented by Eq (2.13). The component σ_{ij} is the norm of the projection of the vector $\boldsymbol{\sigma}(\mathbf{x})\mathbf{e_i})$ (i.e. force per unit area over a surface whose normal is the unit vector $\mathbf{e_i}$) in the direction of $\mathbf{e_j}$.A practical representation of the Cauchy stress tensor is that in the principal directions. In these directions, the shear stresses or those out-of-diagonal components vanish. The normal components are the eigenvalues of the tensor. The eigenvectors define the principal stress directions $\{\mathbf{e_1^*}, \mathbf{e_2^*}, \mathbf{e_3^*}\}$.

$$\sigma_{ij} = (\boldsymbol{\sigma}(\mathbf{x})\mathbf{e_i}) \cdot \mathbf{e_j} \tag{2.14}$$

The forces in principal stress directions are strictly perpendicular to the surfaces defined by the these eigenvectors. Another practical consideration convenient in constitutive modelling is the division of the stress tensor in hydrostatic

or spherical tensor and deviatoric stress tensor Eq (2.15).

$$\boldsymbol{\sigma} = \frac{1}{3} tr\boldsymbol{\sigma}\mathbf{I} + \boldsymbol{\sigma}' \tag{2.15}$$

2.6.3 The First Piola-Kirchhoff stress tensor (PK1)

The Cauchy stress tensor is defined over a unit surface on the deformed configuration. The PK1 $\mathbf{P}$ is the counterpart defined over the undeformed configuration. Thus, if we denote by ds an infinitesimal area with normal $\mathbf{n}$ in the deformed configuration and by ds_o an infinitesimal area with normal $\mathbf{n_o}$ in the undeformed or reference configuration, the counterpart of the Cauchy stress vector $\mathbf{t}$ is the PK1 stress vector $\mathbf{t_{PK1}}$ Eq(2.16).

$$\mathbf{t_{PK1}}\, ds_o = \mathbf{t}\, ds = \boldsymbol{\sigma}\, \mathbf{n}\, ds \tag{2.16}$$

If the infinitesimal reference (or undeformed) surface is defined by two infinitesimal vectors $d\mathbf{q}$ and $d\mathbf{h}$, the normal in the deformed configuration through the mapping introduced by the deformation gradient has the next expression,

$$\begin{aligned} \mathbf{n_o}\, ds_o &= d\mathbf{q} \times d\mathbf{h} \\ \mathbf{n}\, ds &= \mathbf{F}\, d\mathbf{q} \times \mathbf{F}\, d\mathbf{h} \end{aligned}$$

It can be demonstrated that the relationship between PK1 $\mathbf{P}\,\mathbf{n}_o = \mathbf{t_{PK1}}$ and the Cauchy stress tensor is given by Eq (2.17). See [17] for more details about this.

$$\mathbf{P} = det[\mathbf{F}]\, \boldsymbol{\sigma} \mathbf{F}^{-T} \tag{2.17}$$

Remark: The First Piola-Kirchhoff stress tensor is unsymmetric in contrast to the Cauchy stress tensor. The PK1 stress tensor is also called the nominal stress tensor.

A symmetric tensor is often defined as $\boldsymbol{\tau} = det[\mathbf{F}]\,\boldsymbol{\sigma}$. It is called the Kirchhoff stress tensor. Furthermore, other stress measure by a symmetric tensor is the Second Piola-Kirchhoff stress tensor given by,

$$\mathbf{S} = det[\mathbf{F}]\,\mathbf{F}^{-1}\,\boldsymbol{\sigma}\mathbf{F}^{-T} \tag{2.18}$$

2.7 Balance principles in lagrangian configuration

The conservation principles must be satisfied at any continuum solid or fluid and at any time. These principles are the physical and compulsory basis in the whole continuum formulation of this study and final computational implementation of it.

2.7.1 Conservation of Mass Principle

It states that the mass may not be destroyed or created [1].

$$\rho_o(X) = \rho(x(X,t))J(X,t) \tag{2.19}$$

where ρ_o is the reference mass density[2], ρ the spatial one during a motion $x(X,t)$ and $J(X,t)$ the jacobian which is defined in 2.20.

$$J(X,t) = \frac{\partial v}{\partial V} = det\mathbf{F} \tag{2.20}$$

[1] In *Relativistic Physics* may occur sources or sinks of mass, but not in non-relativistic case.

[2] The term *reference* refers to the material description, i.e. it corresponds to the initial density of the objective body if a motion or deformation from the initial configuration happened

with $\mathbf{F} = \frac{\partial x}{\partial X}$ the deformation gradient of the transformation, V the reference volume and v the spatial volume of the body.

2.7.2 Linear Momentum Balance

If we have a body Ω subjected to traction forces on its boundary $\mathbf{t}(\mathbf{x}, \mathbf{n})$ and body forces $\mathbf{b}(\mathbf{x})$ acting in the interior of the body, the momentum balance reads,

$$\int_{\partial\Omega} \mathbf{t}(\mathbf{n})\, ds + \int_{\Omega} \rho(\mathbf{x})\, \mathbf{b}\, \mathbf{dv} = \int_{\Omega} \rho(\mathbf{x})\, \ddot{\mathbf{u}}\, \mathbf{dv} \tag{2.21}$$

It establishes the conservation of 'quantity of motion' [3] or momentum. In differential form this balance can be expressed as follows,

$$Div\mathbf{P} + \mathbf{B} = \rho_o \ddot{\mathbf{u}} \qquad in\ \ \Omega$$

$$\mathbf{t} = \mathbf{P}\mathbf{n} \qquad in\ \ \partial\Omega$$

where Div denotes the material divergence, $\mathbf{P}$ the first Piola-Kirchhov stress tensor, $\mathbf{B}$ the body forces measured respect to the unit reference volume, $\mathbf{t}$ the traction vector and $\mathbf{n}$ the outwards unit normal to the boundary $\partial\Omega$.

2.7.3 Angular momentum balance

It is stated in Eq (2.22) and asserts the conservation of the angular moment.

$$\int_{\partial\Omega} \mathbf{x} \times \mathbf{t}(\mathbf{n})\, \mathbf{ds} + \int_{\Omega} \mathbf{x} \times \rho(\mathbf{x})\, \mathbf{b}\, \mathbf{dv} = \int_{\Omega} \mathbf{x} \times \rho(\mathbf{x})\, \ddot{\mathbf{u}}\, \mathbf{dv} \tag{2.22}$$

2.8 Basic weak form for FE discretisation

The posing of a weak form for the obtention of the momentum equation for the discretised system of finite elements is essential. In our case, the *Principle*

[3] In spatial description is the well-known $div\sigma + \mathbf{b} = \rho_o \ddot{\mathbf{u}}$.

of Virtual Work (PVW) has been elected. In other approaches, *Principle of Minimum Energy* or *Principle of Virtual Power* are taken as weak forms leading to a equilibrium equation. For more details about how it is implemented the reader is referred to the work of De Souza Neto et al.[116]. The PVW must hold for any finite element in which is discretised the domain as they must remain in equilibrium if the system is static or in dynamic equilibrium in the contrary case[4]. Finally, assembly for all elements of the domain gives the resulting system of equations to be solved. Then, a numerical approximation for the displacement field is reached if a convenient strategy of solution converges to it. In this thesis, the different schemes to solve the system of generally non linear equations are exposed. Thus, consider the generic domain of a body Ω (this domain may correspond to the domain of a generic body and , hence, to the domain of a finite element resulting of a discretisation in space). The field of admissible displacements from a kinematic point of view is denoted by $\mathbf{u}$ and the virtual displacements field by $\tilde{\mathbf{u}}$. Virtual displacements obey the continuity conditions and vanish in the boundary where prescribed displacements are enforced ($\partial\Omega_u$). Multiplying the momentum equation by a virtual displacement (or test function) and integrating for the whole domain, is obtained that,

$$\int_\Omega \tilde{\mathbf{u}} \cdot (div\, \boldsymbol{\sigma} + \mathbf{b})\, dv = 0 \qquad \forall\, \tilde{\mathbf{u}} \in \Omega \tag{2.23}$$

Using that,

$$div\,(\boldsymbol{\sigma}\tilde{\mathbf{u}}) = div\, \boldsymbol{\sigma} \cdot \tilde{\mathbf{u}} + \boldsymbol{\sigma} : \nabla\tilde{\mathbf{u}}$$

And substituting in Eq(2.23) and using the Gauss theorem,it follows that

$$\int_{\partial\Omega} \mathbf{n} \cdot \boldsymbol{\sigma}\tilde{\mathbf{u}}\, ds - \int_\Omega \boldsymbol{\sigma}{:}\nabla\tilde{\mathbf{u}}\, dv + \int_\Omega \mathbf{b} \cdot \tilde{\mathbf{u}}\, dv = 0 \tag{2.24}$$

[4]In this case the terms of inertial forces and damping forces appear in the momentum equation.

Taking in account that the virtual strain is $\tilde{\varepsilon} = \nabla\tilde{\mathbf{u}}$, it is clear that,

$$\int_{\partial\Omega} \mathbf{t} \cdot \tilde{\mathbf{u}}\, ds - \int_{\Omega} \boldsymbol{\sigma} : \tilde{\boldsymbol{\varepsilon}}\, dv + \int_{\Omega} \mathbf{b} \cdot \tilde{\mathbf{u}}\, dv = 0 \tag{2.25}$$

As Eq(2.25) must be true for any virtual displacement, the weak form reads: given body forces, prescribed displacements and traction in the boundary, the problems reduces to find the displacement field $\mathbf{u}$ that solve Eq(2.25). This is the basic equation for the FE discretisation. It is applied in the domain of a finite element for a latter assembly operation of all elements for the whole domain of the body.

2.9 Finite Element Discretisation

2.9.1 Interpolation

In this section, the discretisation of the body domain carried out by finite elements is described. The Finite Element Method (FEM) used is displacement-based, i.e. the displacement of a particle of the domain is interpolated through the values of displacements in the nodes of the element which the particle point belongs to. This FEM is, in this study, concerned with the solution of quasi-static structural systems, therefore, a lagrangian description of the motion is particularly convenient.

For all elements e of the mesh in which is discretised the domain, a shape or interpolation function is defined in each node of the element. Thus, an element with n nodes will have n shape functions. A linear combination of these functions and the numerical values of displacements obtained in the nodes will give the interpolated approximation to any point inside the element Eq(2.26). The interpolation function has the characteristic of being unity in

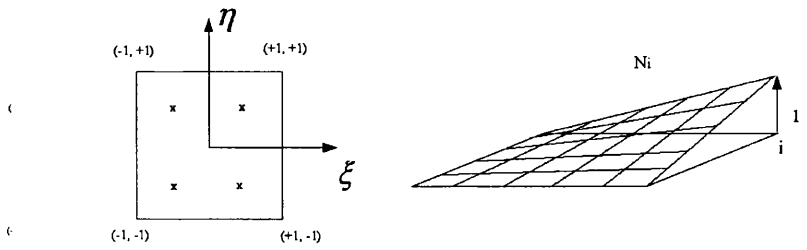


Figure 2.2: Parent element and shape function

the node in which is defined, and zero in the rest of them. In some examples of this dissertation, quadratic shape functions (defined at eigth-nodes finite element have been used, see Section 3.9.4. Thus, a more exact approximation is performed.

$$N_i^{(e)}(\mathbf{x}_j) = \begin{cases} 1 & if \quad i = j \\ 0 & \forall i \neq j \end{cases} \tag{2.26}$$

Any function defined over the domain of the element can be approximated through the shape functions. In Eq(2.27), the displacement in a generic point of the element is displayed.

$$\mathbf{u}(\mathbf{x}) = \sum_{i=1}^{n} N_i(\mathbf{x})\ \mathbf{u}_i \tag{2.27}$$

The displacement of a point on the element (e) is interpolated through the global displacements in the nodes by the global interpolation shape functions. This is represented, for two degrees of freedom per node, by Eq (2.28).

$$\mathbf{u}^{p^{(e)}}(\mathbf{x}) = \mathbf{N}^g(\mathbf{x})\ \mathbf{u}$$

$$\mathbf{u}^{p^{(e)}}(\mathbf{x}) = \begin{bmatrix} N_1^{(e)} & 0 & \dots & N_j^{(e)} & 0 & \dots & N_{n_{node}}^{(e)} & 0 \\ 0 & N_1^{(e)} & \dots & 0 & N_j^{(e)} & \dots & 0 & N_{n_{node}}^{(e)} \end{bmatrix} \cdot \begin{bmatrix} u_1^1 \\ u_2^1 \\ \vdots \\ u_1^j \\ u_2^j \\ \vdots \\ u_1^{n_{node}} \\ u_2^{n_{node}} \end{bmatrix} \tag{2.28}$$

Analogously, we may represent the virtual displacements of a generic point on the element (e) by Eq (2.29).

$$\tilde{\mathbf{u}}^{p^{(e)}}(\mathbf{x}) = \mathbf{N}^g(\mathbf{x})\, \tilde{\mathbf{u}}$$

$$\tilde{\mathbf{u}}^{p^{(e)}}(\mathbf{x}) = \begin{bmatrix} N_1^{(e)} & 0 & \dots & N_j^{(e)} & 0 & \dots & N_{n_{node}}^{(e)} & 0 \\ 0 & N_1^{(e)} & \dots & 0 & N_j^{(e)} & \dots & 0 & N_{n_{node}}^{(e)} \end{bmatrix} \cdot \begin{bmatrix} \tilde{u}_1^1 \\ \tilde{u}_2^1 \\ \vdots \\ \tilde{u}_1^j \\ \tilde{u}_2^j \\ \vdots \\ \tilde{u}_1^{n_{node}} \\ \tilde{u}_2^{n_{node}} \end{bmatrix} \tag{2.29}$$

$$\boldsymbol{\varepsilon} = \mathbf{B} \cdot \mathbf{u} \tag{2.30}$$

The matrix $\mathbf{B}$ which relates strain and global displacements, is often called the *discrete symmetric gradient operator* or simply strain-displacement matrix. In

two-dimensional problems $\mathbf{B}$ reaches the format of Eq (2.31), where the comma indicates derivative in the direction of the corresponding degree of freedom.

$$\mathbf{B}^{(e)} = \begin{bmatrix} N^{(e)}_{1,1} & 0 & N^{(e)}_{2,1} & 0 & \dots & N^{(e)}_{n_{node},1} & 0 \\ 0 & N^{(e)}_{1,2} & 0 & N^{(e)}_{2,2} & \dots & 0 & N^{(e)}_{n_{node},2} \\ N^{(e)}_{1,2} & N^{(e)}_{1,1} & N^{(e)}_{2,2} & N^{(e)}_{2,1} & \dots & N^{(e)}_{n_{node},2} & N^{(e)}_{n_{node},1} \end{bmatrix} \tag{2.31}$$

$$\tilde{\boldsymbol{\varepsilon}} = \mathbf{B} \cdot \tilde{\mathbf{u}} \tag{2.32}$$

The stress tensor is often substituted by a vector format taking advantage of the symmetry of the stress tensor. Thus, the stress is represented by Eq (2.33) for two-dimensional analyses or in Eq(2.34) for 3D.

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{bmatrix} \tag{2.33}$$

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} & \sigma_{22} & \sigma_{33} & \sigma_{12} & \sigma_{23} & \sigma_{13} \end{bmatrix}^T \tag{2.34}$$

2.9.2 The discrete problem

The principle of virtual work exposed by Eq (2.25) becomes after introduction of the interpolation functions Eq (2.35)

$$\int_{\partial\Omega} \mathbf{t} \cdot \mathbf{N}^g(\mathbf{x})\, \tilde{\mathbf{u}}\, ds - \int_{\Omega} \boldsymbol{\sigma} : \mathbf{B} \cdot \tilde{\mathbf{u}}\, dv + \int_{\Omega} \mathbf{b} \cdot \mathbf{N}^g(\mathbf{x})\, \tilde{\mathbf{u}}\, dv = 0 \tag{2.35}$$

Rearranging and, for clarity, dropping the upper-script ^{g} on the shape functions,

$$\left\{ -\int_{\Omega} \mathbf{B}^T \cdot \boldsymbol{\sigma}\, dv + \int_{\Omega} \mathbf{N}^T(\mathbf{x}) \cdot \mathbf{b}\, dv + \int_{\partial\Omega} \mathbf{N}^T(\mathbf{x}) \cdot \mathbf{t}\, ds \right\} \cdot \tilde{\mathbf{u}} = 0 \tag{2.36}$$

As the Eq (2.36) must be satisfied for any virtual displacement, it follows that,

$$-\int_{\Omega} \mathbf{B}^T \cdot \boldsymbol{\sigma}\, dv + \int_{\Omega} \mathbf{N}^T(\mathbf{x}) \cdot \mathbf{b}\, dv + \int_{\partial\Omega} \mathbf{N}^T(\mathbf{x}) \cdot \mathbf{t}\, ds = 0 \tag{2.37}$$

The problem reduces to find the global vector of nodal displacements that obey the boundary conditions,

$$-\mathbf{f}^{(int)}(\mathbf{u}) + \mathbf{f}^{(ext)} = 0 \tag{2.38}$$

2.9.3 Assembly

The assembly operator $\bigwedge_{e=1}^{nelem}$ is now introduced. It implies that each component of the vector of global forces associated to a node is obtained as the sum of the corresponding contributions from the element force vectors of all elements that share that global node.

$$\mathbf{f}^{int}(\boldsymbol{u}) = \int_{\Omega} \mathbf{B}^T \cdot \boldsymbol{\sigma}\, dv = \bigwedge_{e=1}^{nelem} \left\{ \int_{\Omega^{(e)}} \mathbf{B}^T \cdot \boldsymbol{\sigma}\, dv \right\} \tag{2.39}$$

$$\mathbf{f}^{ext} = \int_{\Omega} \mathbf{N}^T \cdot \boldsymbol{b}\, dv + \int_{\partial\Omega} \mathbf{N}^T \cdot \boldsymbol{t}\, ds = \bigwedge_{e=1}^{nelem} \left\{ \int_{\Omega^{(e)}} \mathbf{N}^T \cdot \boldsymbol{b}\, dv + \int_{\partial\Omega^{(e)}} \mathbf{N}^T \cdot \boldsymbol{t}\, ds \right\} \tag{2.40}$$

2.9.4 Numerical integration

The exact calculation of the integrals of the forces are replaced by numerical approximations in quadrature points. For instance, a function in a generic domain Γ is approximated as follows,

$$\int_{\Gamma} f(\boldsymbol{\xi}) d\boldsymbol{\xi} \approx \sum_{k=1}^{n_{gauss}} w_k\, f(\boldsymbol{\xi}_k) \tag{2.41}$$

where w_k are the weights associated to each gaussian or quadrature point. If we map a generic function g in the element domain, the standard domain Γ

onto the element domain is approximated in the following manner,

$$\int_{\partial\Omega^{(e)}} g(\mathbf{x})d\mathbf{x} = \int_{\Gamma} g(\mathbf{x}(\boldsymbol{\xi}))\, J(\boldsymbol{\xi})d\boldsymbol{\xi} \approx \sum_{k=1}^{n_{gauss}} w_k\, g_k\, J_k \tag{2.42}$$

where $J(\boldsymbol{\xi}) = det[\frac{\partial \mathbf{x}}{\partial \boldsymbol{\xi}}]$, $J_k = J(\boldsymbol{\xi}_k)$ and $g_k = g(\mathbf{x}(\boldsymbol{\xi}_k))$. Now Eqs (2.39, 2.40) are represented in a ready-to-be-programmed manner as follows,

$$\mathbf{f}^{int}(\boldsymbol{u}) = \bigwedge_{e=1}^{nelem} \left\{ \sum_{k=1}^{n_{gauss}} w_k\, \mathbf{B}_k^T \cdot \boldsymbol{\sigma}_k\, J_k \right\}^{(e)}$$

$$\mathbf{f}^{ext} = \bigwedge_{e=1}^{nelem} \left\{ \sum_{k=1}^{n_{gauss}} w_k \mathbf{N}_k^T \cdot \boldsymbol{b}_k\, J_k + \sum_{i=1}^{n_{gaussb}} w_i\, \mathbf{N}_i^T \cdot \boldsymbol{t}_i\, J_i^b \right\}^{(e)}$$

where the surface integrals have been approximated by gaussian quadratures on the boundary of the domain in an analogous manner to that for the numerical approximation in the interior of the domain.

2.10 Direct integration of momentum equations

The evolution of the approximate solution may be determined by integrating the semidiscretised momentum balance equation with respect to time. The direct integration methods usually adopted belong to the class of incremental Euler schemes, where approximate solutions are determined at discrete time increments.

Related Euler integration schemes are recovered depending on when equilibrium is sought within a time increment –timestep. Explicit integration schemes consider equilibrium at the beginning of each increment, thus enabling the projected solution to be explicitly defined in terms of known quantities. Implicit

integration schemes establish equilibrium during the time increments resulting in the projected solution being implicitly defined.

The explicit central difference integration scheme is adopted in the in some of the algorithms proposed in this dissertation, see Chapters 3. As a conditionally stable method, the time step needs to be bounded. Proof of this conditional stability can be found in many textbooks [5, 11, 7, 58]. In Chapter 3, further details are provided, such as critical time step for convergence or temporal discretised equations. A formulation of the method is presented in the following section.

2.10.1 Central Difference Method (CDM)

The semidiscretisation by finite elements leads to a system of second order differential equations Eq(2.43).

$$\mathbf{M}\,\ddot{\mathbf{u}}_n + \mathbf{C}\,\dot{\mathbf{u}}_n + \mathbf{f}^{int}(\mathbf{u}_n) = \mathbf{f}^{ext} \tag{2.43}$$

where n indicates the n^{th} time step, $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ the damping matrix[5], $\mathbf{f}^{int}(\mathbf{u}_n)$ the internal forces vector, $\mathbf{f}^{ext}$ the external forces vector, and $\ddot{\mathbf{u}}(t_n), \dot{\mathbf{u}}(t_n), \mathbf{u}_n$ are, respectively, the accelerations, velocities and displacements vectors. The following central differences are utilized for the approximation of thee corresponding derivatives,

$$\dot{\mathbf{u}}_{n-\frac{1}{2}} = \frac{\mathbf{u}_n - \mathbf{u}_{n-1}}{\Delta t_{n-\frac{1}{2}}} \tag{2.44}$$

$$\ddot{\mathbf{u}}_n = \frac{\dot{\mathbf{u}}_{n+\frac{1}{2}} - \dot{\mathbf{u}}_{n-\frac{1}{2}}}{\Delta t_n} \tag{2.45}$$

[5]Some techniques,such as dynamic relaxation, use an adaptive damping matrix and, hence, it is relying upon the time step. This optimises the attenuation of the transient response when solving a quasi-static problem

$$\dot{\mathbf{u}}_n = \frac{\dot{\mathbf{u}}_{n+\frac{1}{2}} + \dot{\mathbf{u}}_{n-\frac{1}{2}}}{2} \tag{2.46}$$

where Δt_n has been set in a general form as an adaptive time each increment (Section 3.7.4). However, there are other works that traditionally have used constant time step with the only constraint of the stability requirement. Performing the substitution of Eqs (2.45, 2.46) in the momentum equilibrium Eq (2.43) results in the following expressions for the velocity,

$$\dot{\mathbf{u}}_{n+\frac{1}{2}} = \frac{2\mathbf{M} - \Delta t_n \mathbf{C}}{2\mathbf{M} + \Delta t_n \mathbf{C}} \dot{\mathbf{u}}_{n-\frac{1}{2}} + \frac{\mathbf{f}^{ext}(t_n) - \mathbf{f}^{int}(t_n)}{2\mathbf{M} + \Delta t_n \mathbf{C}} \tag{2.47}$$

With this expression the displacement at step $n+1$ is simply given by,

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t_{n+\frac{1}{2}} \, \dot{\mathbf{u}}_{n+\frac{1}{2}} \tag{2.48}$$

The mid-time increments are calculated as follows,

$$\Delta t_{n-\frac{1}{2}} = \frac{\Delta t_n + \Delta t_{n-1}}{2} \tag{2.49}$$

To preserve an explicit iterative scheme the mass and damping matrices must be diagonal. A special lumping technique to diagonalize the mass matrix is performed in Chapter 3. The damping matrix may be chosen proportional to the mass matrix. There are other forms, such as *Rayleigh Damping*, which ensures a diagonal damping matrix. This due to the use of an approximated diagonal stiffness matrix. The resultant damping matrix is, then, stiffness and mass proportional. This damps a wide range of frequencies. Therefore, it results in a optimized strategy when pursuing the steady state of the transient system. Thus, if a mass proportional damping, $\mathbf{C} = p\mathbf{M}$, is elected and substituted into Eq (2.47) the velocities and displacements may be expressed as in Eq (2.50).

$$\dot{\mathbf{u}}_{n+\frac{1}{2}} = \frac{2 - \Delta t_n p}{2 + \Delta t_n p} \dot{\mathbf{u}}_{n-\frac{1}{2}} + 2\Delta t_n \mathbf{M}^{-1} \frac{\mathbf{f}^{ext}(t_n) - \mathbf{f}^{int}(t_n)}{2 + \Delta t_n p} \tag{2.50}$$

$\mathbf{M}^{-1}$ is diagonal as the mass matrix is diagonal ($(\mathbf{M}^{-1})_{ii} = \frac{1}{m_{ii}}$). The system is formed now by a set of uncoupled algebraic equations. Each time step is relying on the former time step. Therefore, it renders an explicit method and as all other explicit methods is conditionally stable.

$$\dot{u}_{i,n+\frac{1}{2}} = \frac{2-\Delta t_n p}{2+\Delta t_n p}\dot{u}_{i,n-\frac{1}{2}} + 2\Delta t_n \frac{f_i^{ext}(t_n) - f_i^{int}(t_n)}{(2+\Delta t_n p)m_{ii}} \tag{2.51}$$

The step is completed with the update of displacements by Eq (2.48) which is written in index notation in Eq (2.52).

$$u_{i,n+1} = u_{i,n} + \Delta t_n \, \dot{u}_{i,n+\frac{1}{2}} \tag{2.52}$$

Initiation of CDM

The Eqs (2.50,2.51) need the not-known velocity at $t_{-\frac{1}{2}}$. We need other initial condition to start the iterations. Thus, using Eq(2.46) and $\dot{\mathbf{u}}_0 = 0$ is obtained that,

$$\dot{\mathbf{u}}_{-\frac{1}{2}} = -\dot{\mathbf{u}}_{\frac{1}{2}} \tag{2.53}$$

For the first iteration, the Eqs (2.50,2.51) become,

$$\dot{\mathbf{u}}_{\frac{1}{2}} = \Delta t_n \mathbf{M}^{-1} \frac{\mathbf{f}^{ext}(t_0) - \mathbf{f}^{int}(t_0)}{2} \tag{2.54}$$

$$\dot{u}_{i,\frac{1}{2}} = \Delta t_n \frac{f_i^{ext}(t_0) - f_i^{int}(t_0)}{2m_{ii}} \tag{2.55}$$

2.10.2 Dynamic Relaxation

The problem is reduced to optimize mass matrix, damping matrix and time step to get the maximum velocity of convergence to obtain the displacements such that internal and external forces are in equilibrium. This can be achieved by the use of dynamic relaxation. The static solution is obtained as the attenuation of the transient response of the structural dynamic system, i.e. the

steady state. Thus, it should be noted that only the internal and external forces vectors represent the physical problem. Therefore, mass and damping matrices have not necessarily to represent the physical reality. The update and dynamic relaxation quotient are given by,

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t_{n+\frac{1}{2}} \dot{\mathbf{u}}_{n+\frac{1}{2}} \tag{2.56}$$

$$\mathbf{C}_\beta = \sqrt{\frac{\mathbf{u}_n^T \mathbf{K}_n \mathbf{u}_n}{\mathbf{u}_n^T \mathbf{M} \mathbf{u}_n}} \tag{2.57}$$

$$K_{ii} = \frac{f_i(\mathbf{u}_n) - f_i(\mathbf{u}_{n-1})}{\Delta t_{n-\frac{1}{2}} \dot{u}_{n-\frac{1}{2}}} \tag{2.58}$$

Observe that the stiffness matrix is approximated each iteration for the unique purpose of updating of the damping matrix.

Chapter 3

A Novel In-Time Implicit-Explicit Algorithm for Nonlinear Finite Element Analysis

'....Ray Clough told me that the Finite Element Method is a good thing. We both don't know anything about it. Let's try..'. **O.C. Zienkiewicz**; 1962.

3.1 Introduction

The *Finite Element Method* (FEM) has proved to be a numerical procedure convenient to solve differential equations systems corresponding to physical models. The direct integration of the momentum equations [1] strong form or the

[1] These equations are based in a finite element discretisation of the domain of the body as explained in chapter(2)

solution of weak form through implicit solvers (e.g.*Newton-Raphson*, ...) have been carried out according to the type of problem. However, some problems in finite element analysis may not be readily solved by implicit methods of solution such as contact between corner and curved concave surface [16, 69, 84, 93]. In such cases of non-smooth contact, e.g. where a sharp-pointed edge contacts a concave surface, or critical points, such as buckling points, explicit (EXP) or direct integration of the momentum equations is an alternative scheme of solution, Figure (3.1). However, this last situation would involve a high computational cost in time (as its stability is conditioned by means of a critical time step) in domains with a high number of elements. At this stage, the idea of executing an implicit method until divergence arises and in that point starting to solve with an explicit method is served [86]. Additionally, the execution may return back to the implicit scheme (IMP) once the divergence source has been passed and the external loading is not totally applied.

Other problems difficult to solve with implicit methods include discontinuities emerging in the physical body or when the load applied in large deformation analysis is so high that execution starts to differ from the real solution if the load factor applied is not reduced . It leads to divergence too. Explicit methods, such as the *Central Differences Method*, obtain the solution always when the time step is smaller than a critical value. Lax's Theorem states that stability[2] and consistency[3] are sufficient and necessary conditions for convergence. Therefore, the critical time step must not be exceeded in order to obtain stability of the numerical method (conditional stability).

It is important to remark that no division of the mesh, element-partitioning

[2]The numerical does not diverge

[3]The approximation tends, in the limit,($\Delta t \rightarrow 0$) to the differential equation

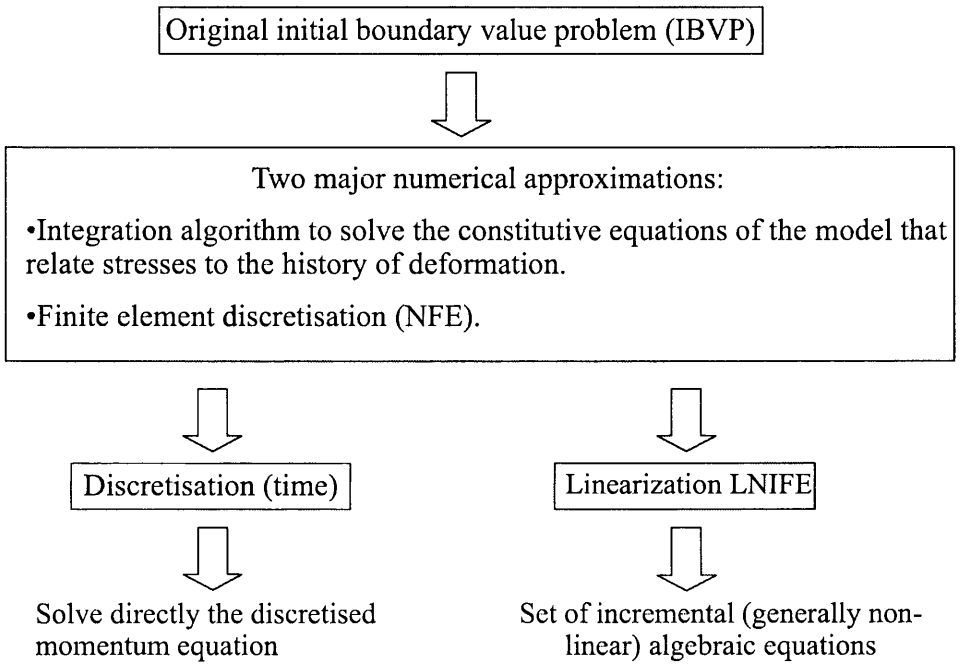


Figure 3.1: Numerical approximations of the schemes considered. *Linearised Non-linear Incremental Finite Element equations (LNIFE)* (implicit) and direct time integration by *Central Difference Method* (explicit) after FE discretisation.

or nodal partitioning for separate treatment of the solution is made as this is widely studied: a first method with element partitioning is described in [59] among others such as[9, 51, 60]. The aim of this chapter is to illustrate the implicit/explicit (I/E) algorithm in-time without going into contact problems at this time. A connection between the full Newton-Raphson Method with implicit backward Euler pseudo-integration (integration algorithm) nested (implicit, IMP) and the Central Differences Method (explicit, EXP) has been elected. A description of the implicit and explicit schemes coded and their connection is presented. Buckling analysis including snap-through has been carried out to validate this in-time implicit/explicit (I/E) code as this sort of problems does not converge with Newton-Raphson unless arc-length procedures are employed. Nevertheless, arc-length methods find difficulties of convergence in snap-back analysis if an appropriate path prediction is not provided [43, 45, 44, 115]. Furthermore, analysis corroborating the explicit code ,alone ,has been executed in order to guarantee the entire program will run properly. This study has also been published in the *'Communications in Numerical Methods in Engineering'* journal [31] and in abbreviated form in the proceedings of the *'3^{rd} MIT Conference on Computational Fluid and Solid Mechanics'* [30].

3.2 Sources of instabilities and nonlinearities in structures

Nonlinearities are classified due to their source in mathematical terms of the *Continuum Mechanics* and related to the physical problem of the structure. The general form to interpret nonlinearities are either load vs. deflection

(scalars) or force vs. displacement (vectorial) relationships. Henceforth, load vs. deflection or components of the force-displacement relationships will be used to plot in a two-dimensional graph. These curves are assessed in the *special points* that characterize the nonlinear behaviour. The response of the structure following these curves is usually named *equilibrium path* . A convenient classification of these special points is [40, 41]:

- *Turning points*, the equilibrium path suddenly stops or jumps because of a physical failures in the structure. This is generally due to the material of the structure. These failure either may be instantaneous and jump to other state of equilibrium or can be catastrophic and the structure does not reach the equilibrium any more.

- *Critical points*, these points can affect methods of solution. They will be central part in this chapter as the proposed algorithm is performed to solve the problems generated by them.

3.2.1 Critical points

Snap-through

The first linear part of the curve is suddenly dropped by softening after the first limit point. In this softening regime, the response has a negative stiffness and generates instability. After this softening, the slope of the curve turns positive and the structure hardens (positive stiffness)Figure[3.2]. This type of behaviours may be observed in shallow arches loaded in the mid-span.

Snap-back

In this case the curve turns back after the first limit point, *i.e. onset of softening regime* causing the appearance of turning points Figure[3.2]. The equilibrium between limit points may be stable, however from our perspective the limit points may cause numerical instability when simulating these behaviours with a computer method.

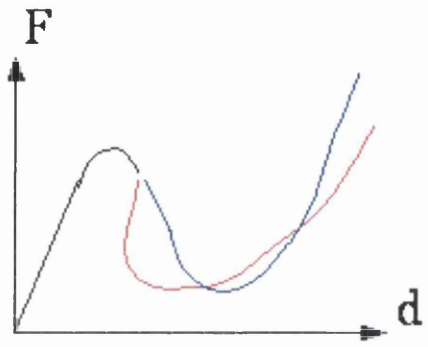


Figure 3.2: Snap-back (*in red*) and snap-through (*in blue*) responses

3.3 Elasto-plastic constitutive model

Von-Mises constitutive model is used for small strain problems. A brief review for two-dimensional elasto-plastic constitutive models is presented in this section and the Von-Mises constitutive model in Section [3.4]. The basic constituents of the elasto-plastic model are,

1. Decomposition of strain in elastic and plastic parts.

2. Elastic evolution.

3. Yield criterion which is geometrically expressed by a yield surface.

4. Evolution of the plastic strain: plastic flow rule.

5. Evolution of the yield border: hardening law.

3.3.1 Yield surface

This surface limits the elastic domain. It has a negative value for elastic deformations and zero when plastic strains occur, *i.e. plastic flow has started.* It is a surface in the space of stresses. The elastic domain E can be defined as,

$$E = \{\boldsymbol{\sigma} \,\|\, Y(\boldsymbol{\sigma}, \boldsymbol{\alpha}) < 0\} \tag{3.1}$$

And the yield surface is defined as,

$$P = \{\boldsymbol{\sigma} \,\|\, Y(\boldsymbol{\sigma}, \boldsymbol{\alpha}) = 0\} \tag{3.2}$$

3.3.2 Plastic flow rule and hardening law

The evolution of the plastic strains and state variables are chosen by the modeller. Thus, work hardening or accumulated plastic strain are generally accepted as internal variables. In the case of existence of damage because of growth of voids and coalescence of them, a damage variable varying between zero and the unity and degrading the stresses [70] is needed as internal variable. In a general way the plastic flow rule and the hardening law can be postulated as,

$$\dot{\boldsymbol{\varepsilon}}^p = \dot{\gamma}\, \mathbf{N}(\boldsymbol{\sigma}, \boldsymbol{\alpha}) = \dot{\gamma}\, \frac{\partial \Psi}{\partial \boldsymbol{\sigma}} \tag{3.3}$$

where $\mathbf{N}(\boldsymbol{\sigma}, \boldsymbol{\alpha})$ is the vector of flow. The hardening law is as follows,

$$\dot{\boldsymbol{\alpha}} = \dot{\gamma}\, \mathbf{H}(\boldsymbol{\sigma}, \boldsymbol{\alpha}) \tag{3.4}$$

where $\mathbf{H}(\boldsymbol{\sigma}, \boldsymbol{\alpha})$ defines the evolution of the hardening variables and is called the *generalized hardening modulus[116].* $\mathbf{N}$ and $\mathbf{H}$ can calculated from the existence of a flow potential. If this potential is the yield function the flow is called *associative.* Metals models are usually chosen as associative.

3.3.3 Loading/unloading criterion

Evolution of the plastic flow Eq 3.3 and hardening Eq 3.4 is complemented with the loading/unloading criterion,

$$\Phi \leq 0 \qquad \dot{\gamma} \geq 0 \qquad \Phi\,\dot{\gamma} = 0 \tag{3.5}$$

3.4 Von-Mises constitutive model

3.4.1 Von-Mises yield criterion

This criterion states than the plastic yielding starts for a critical value (σ^y) of the deviatoric stress invariant $J_2(\mathbf{s})$ where $\mathbf{s}$ is the deviatoric stress tensor. This invariant is a function of the state variables. The hydrostatic pressure does not influence the yield criterion definition such as happens with the Tresca yield criterion. In a state of pure shear, i.e. outer Mohr circle centered in the origin, $\sigma_1 = -\sigma_2 > 0$, $\sigma_3 = 0$ and, hence, $\sqrt{J_2(\mathbf{s})} = \tau_{max} = \sigma_1$. The yield function for pure shear is,

$$\Phi(\boldsymbol{\sigma}) = \sqrt{J_2(\mathbf{s})} - \tau^y$$

. In the case of uniaxial stress the criterion is postulated as,

$$\Phi(\boldsymbol{\sigma}) = \sqrt{3\,J_2(\mathbf{s})} - \sigma^y$$

where σ^y is the uniaxial (tension or compression) yield stress. $\sqrt{3\,J_2(\mathbf{s})}$ is known as the Von-Mises effective stress or Von-Mises equivalent stress. The

relation between the uniaxial yield stress and the shear yield stress is given by,

$$\sigma^y = \sqrt{3}\,\tau^y$$

The multiaxial Von-Mises yield surface $\Phi(0)$ is represented in the space of principal stresses as a infinite cylinder with the hydrostatic line as axis of the cylinder. The Von-Mises yield surface fits better than Tresca yield surface (hexagon) for most metals. Both surfaces, Von-Mises and Tresca coincide in shear (pure shear occurs at an angle of 30^o with the principal axes of stresses).

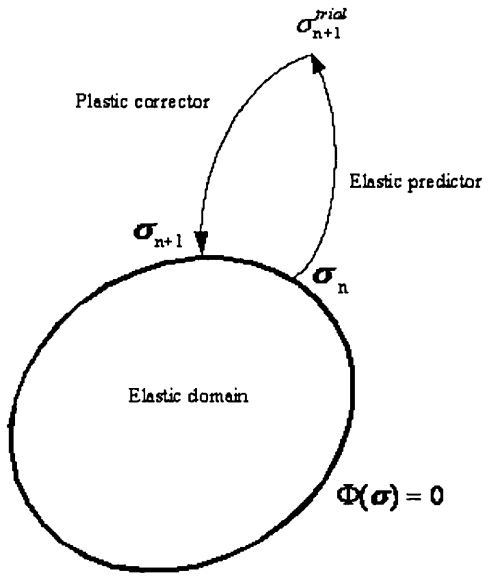


Figure 3.3: Detail of pseudointegration in perfectly plastic material

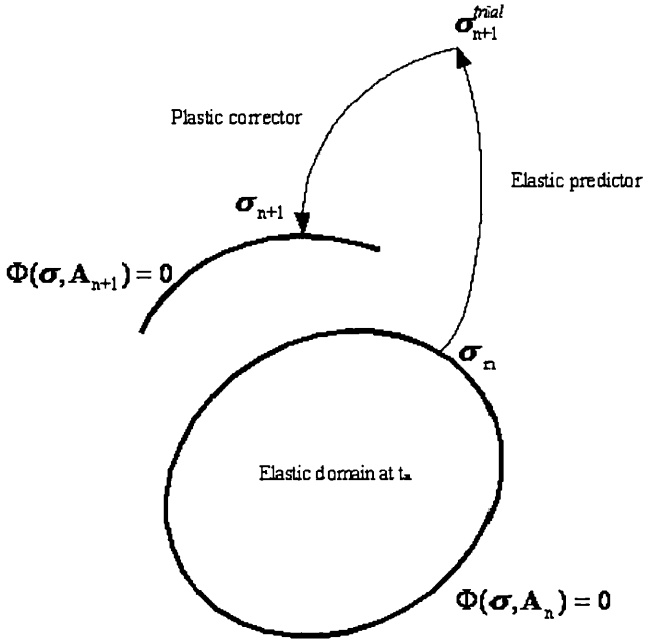


Figure 3.4: Elastic prediction - plastic correction pseudointegration for hardening

3.5 Integration algorithm for the Von Mises model with isotropic hardening

In a time increment $[t_n, t_{n+1}]$, the increment of strain is given by Eq (3.6). The state variables are the elastic strain ε^e_n and the accumulated plastic strain $\bar{\varepsilon}^p_n$ at the beginning of the time interval $[t_n, t_{n+1}]$. The trial state is defined in Eq (3.7).

$$\Delta\varepsilon = \varepsilon_{n+1} - \varepsilon_n \tag{3.6}$$

$$\varepsilon^{e\,trial}_{n+1} = \varepsilon^e_n + \Delta\varepsilon \tag{3.7}$$

$$\bar{\varepsilon}^{p\,trial}_{n+1} = \bar{\varepsilon}^p_n \tag{3.8}$$

The trial stress tensor is computed assuming elastic evolution in Eq (3.9), Figure (3.4). The trial yield stress depends on the accumulated plastic strain

at t_n Eq (3.10).

$$\boldsymbol{\sigma}_{n+1}^{trial} = \mathbf{D}^e : \boldsymbol{\varepsilon}_{n+1}^{e\,trial} \tag{3.9}$$

$$\sigma_{y\,n+1}^{trial} = \sigma_y(\bar{\varepsilon}_n^p) \tag{3.10}$$

If $\boldsymbol{\sigma}_{n+1}^{trial}$ lies inside of the trial yield surface Eq (3.11), the evolution is purely elastic within the time interval $[t_n, t_{n+1}]$ and the trial state is the solution.

$$\Phi(\boldsymbol{\sigma}_{n+1}^{trial}, \sigma_{y\,n}) \leq 0 \tag{3.11}$$

The computational update is simply performed as follows,

$$\boldsymbol{\varepsilon}_{n+1}^e = \boldsymbol{\varepsilon}_{n+1}^{e\,trial} \tag{3.12}$$

$$\bar{\varepsilon}_{n+1}^p = \bar{\varepsilon}_{n+1}^{p\,trial} = \bar{\varepsilon}_n^p \tag{3.13}$$

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_{n+1}^{trial} \tag{3.14}$$

$$\sigma_{y\,n+1} = \sigma_{y\,n+1}^{trial} = \sigma_{y\,n} \tag{3.15}$$

Otherwise, the evolution is elasto-plastic and the trial state lies outside of the elastic domain defined by the yield surface. Therefore, a return mapping, Figure (3.4), is conducted. The implicit return mapping equations for the Von-Mises model are,

$$\boldsymbol{\varepsilon}_{n+1}^e = \boldsymbol{\varepsilon}_{n+1}^{e\,trial} - \Delta\gamma\sqrt{\frac{2}{3}}\frac{\mathbf{s}_{n+1}}{\|\mathbf{s}_{n+1}\|} \tag{3.16}$$

$$\bar{\varepsilon}_{n+1}^p = \bar{\varepsilon}_n^p + \Delta\gamma \tag{3.17}$$

$$0 = \sqrt{3\,J_2(\mathbf{s}_{n+1})} - \sigma_y(\bar{\varepsilon}_{n+1}^p) \tag{3.18}$$

This set of algebraic non-linear equations has to be solved for $\boldsymbol{\varepsilon}_{n+1}^e$, $\bar{\varepsilon}_{n+1}^p$ and $\Delta\gamma$. $\mathbf{s}_{n+1}$ is the deviatoric stress tensor Eq (3.19).

$$\mathbf{s}_{n+1} = 2G\,dev[\boldsymbol{\varepsilon}_{n+1}^e] \tag{3.19}$$

The system can be simplified in number of equations with the *single-equation return mapping* having the plastic multiplier $\Delta\gamma$ as variable. This reduction makes the process more computationally efficient. For more information the interested reader is referred to [114].

3.6 Implicit sub-algorithm

The solution of the weak form of the momentum equations by Newton-Raphson Method (NRM) is used. In nonlinear analysis a convenient scheme [4] to integrate the rate constitutive equation is required. The algorithm will depend on the type of material [5] and the consideration of large deformation or not. In this study, the implicit algorithm is devoted to the general case of path-dependent materials and large deformation, taking into account that the objective of this work is the connection of this implicit algorithm and the explicit one and not the success in a particular type of material model or geometry.

The implicit algorithm is based upon a pseudo-time discretisation (Simo and Taylor [107]) considering the transition of deformation between two time points. The implicit backward Euler method coupled with the Newton-Raphson iterative scheme is utilized [92, 106, 116]. Thus, if a time increment $[t_n, t_{n+1}]$ and set of internal variables α_n at t_n are given, the deformation tensor $\boldsymbol{\varepsilon}(t_{n+1})$ must determine the stresses $\boldsymbol{\sigma}(t_{n+1})$ and internal variables only through the integration algorithm, i.e.:

$$\boldsymbol{\sigma}(t_{n+1}) = \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1}) \tag{3.20}$$

$$\boldsymbol{\alpha}(t_{n+1}) = \hat{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1}) \tag{3.21}$$

[4] Integration algorithms like for example the return mapping algorithm

[5] In general, path-dependent materials.

After discretisation of the domain into finite elements, the problem is committed to find displacements $\boldsymbol{u}_{n+1}$ at time t_{n+1}, so that the incremental nonlinear FE equation, (5.17) is satisfied.

$$\boldsymbol{R}(\boldsymbol{u}_{n+1}) = \mathbf{f}^{int}(\boldsymbol{u}_{n+1}) - \mathbf{f}^{ext}_{n+1} = 0 \tag{3.22}$$

where internal and external forces vectors are obtained by

$$\mathbf{f}^{int}(\boldsymbol{u}_{n+1}) = \bigwedge_{e=1}^{nelem} \left\{ \int_{\Omega^{(e)}} \mathbf{B}^T \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}(\boldsymbol{u}_{n+1}))\, \mathrm{d}v \right\} \tag{3.23}$$

$$\mathbf{f}^{ext}_{n+1} = \bigwedge_{e=1}^{nelem} \left\{ \int_{\Omega^{(e)}} \mathbf{N}^T \boldsymbol{b}_{n+1} \mathrm{d}v + \int_{\partial\Omega^{(e)}} \mathbf{N}^T \boldsymbol{q}_{n+1}\, \mathrm{d}s \right\} \tag{3.24}$$

where $N(\xi, \eta)$ are bilinear shape functions, $\mathbf{b}_{n+1}$ are the body forces, $\mathbf{q}_{n+1}$ the traction forces applied over the boundary of the body and $\mathbf{B}$ is the linear strains operator which has the next format (in plane stress/strain analysis) for the generic element (e) (the first subindex denotes number of local node and the comma derivative):

$$\mathbf{B} = \begin{bmatrix} N^{(e)}_{1,1} & 0 & N^{(e)}_{2,1} & 0 & \dots & N^{(e)}_{n_{node},1} & 0 \\ 0 & N^{(e)}_{1,2} & 0 & N^{(e)}_{2,2} & \dots & 0 & N^{(e)}_{n_{node},2} \\ N^{(e)}_{1,2} & N^{(e)}_{1,1} & N^{(e)}_{2,2} & N^{(e)}_{2,1} & \dots & N^{(e)}_{n_{node},2} & N^{(e)}_{n_{node},1} \end{bmatrix}$$

Equation (5.17) needs to be linearized in order to enable a numerical procedure. Details of this may be found in de Souza et al.[116].

3.6.1 Solution to the implicit incremental problem

As stated above the NRM has been utilized in the solution of equation (5.17) (note that, in elastic materials, solution is immediate with some algebraic method as Frontal [85]) because of its quadratic rate of convergence. It consists

in solving within each iteration the linearized version of equation (5.17) for the incremental global displacement $\delta \boldsymbol{u}^{(k)}$:

$$\mathbf{K}_T \delta \boldsymbol{u}^{(k)} = -\boldsymbol{R}^{(k-1)}(\boldsymbol{u}_{n+1}) \tag{3.25}$$

where $\mathbf{K}_T$ is the global tangent stiffness matrix given as,

$$\mathbf{K}_T = \left. \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{u}_{n+1}} \right|_{\boldsymbol{u}_{n+1}^{(k-1)}} \tag{3.26}$$

which is obtained by assemble of element stiffness matrices:

$$\mathbf{k}_T^{(e)} = \int_{\Omega^{(e)}} \mathbf{B}^T \hat{\mathbf{D}} \mathbf{B} \, \mathrm{d}v \tag{3.27}$$

where $\hat{\mathbf{D}}$ is the consistent tangent stiffness matrix (de Souza et al. [116]):

$$\hat{\mathbf{D}} = \left. \frac{\partial \hat{\boldsymbol{\sigma}}}{\partial \boldsymbol{\varepsilon}_{n+1}} \right|_{\boldsymbol{\varepsilon}_{n+1}^{(k-1)}} \tag{3.28}$$

The implicit scheme in compact form is presented below. Note at point 7 of Box(3.1) that the flow of execution is diverted to explicit if needed.

As indicated below the initial values (displacement and residual) to initialize the explicit algorithm are the solution obtained at the last load increment that converged.

3.6.2 Formulation for finite strains

In the case of finite elasticity, the material is independent of the path and, hence, internal variables are not needed in the estimation of stresses. Thus, they can be evaluated without any numerical integration algorithm.

The former statement (general case of path-dependent materials) is also applied here to update stresses and other state variables through a numerical integration algorithm. Moreover, the next formulation in the case of finite

BOX 3.1: IMPLICIT INCREMENTAL PROBLEM

1. Initiate (k=0)

$$\boldsymbol{u}_{n+1}^{(0)} = \boldsymbol{u}_n$$

$$\boldsymbol{R} = \boldsymbol{f}^{int}(\boldsymbol{u}_n) - \boldsymbol{f}^{ext}$$

2. For all elements, calculate consistent tangent stiffness matrix.

$$\hat{\mathbf{D}} = \frac{\partial \hat{\boldsymbol{\sigma}}}{\partial \boldsymbol{\varepsilon}_{\mathbf{n+1}}}$$

3. Assembly of stiffness matrices

$$\mathbf{k}_T^{(e)} = \sum_{j=1}^{n_{gaus}} \xi_j \, \mathbf{B}_j^T \, \hat{\mathbf{D}}_j \, \mathbf{B}_j$$

4. Increment iteration counter (k=k+1), assembly, solve the linearized equilibrium equation (5.20) and update stresses and internal variables:

$$\boldsymbol{u}_{n+1}^{(k)} = \boldsymbol{u}_{n+1}^{(k-1)} + \delta \boldsymbol{u}^{(k)}$$

$$\boldsymbol{\varepsilon}_{n+1}^{(k)} = \mathbf{B} \, \boldsymbol{u}_{n+1}^{(k)}$$

$$\boldsymbol{\sigma}_{n+1}^{(k)} = \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1}^{(k)})$$

$$\boldsymbol{\alpha}_{n+1}^{(k)} = \hat{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_n, \boldsymbol{\varepsilon}_{n+1}^{(k)})$$

5. New internal forces at each element

$$\boldsymbol{f}_{(e)}^{int} = \sum_{j=1}^{n_{gaus}} \xi_j \, J_j \mathbf{B}_j^T \, \boldsymbol{\sigma}_{n+1,\,j}^{(k)}$$

6. Gathering of element internal forces vector and updating residual.

(Continues box 3.1)

7. If iterations diverge then go to the EXP scheme (see Fig 3.11), else:

   (a) If $\frac{\|\boldsymbol{f}^{ext}-\boldsymbol{f}^{int}\|}{\|\boldsymbol{f}^{ext}\|} \leq \epsilon$ then the solution for current external load is reached and values for this load are from the last iteration $(\bullet)_{n+1} = (\bullet)_{n+1}^{(k)}$

   (b) else go to (2).

8. If the total load is not completely applied, increment the external load, else exit.

strains are considered in the analysis.

The stresses are given by $\boldsymbol{\sigma}_{n+1} = \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \mathbf{F}_{n+1})$ (right-hand side is named *algorithmic incremental constitutive function*) with $\mathbf{F}_{n+1}$ being the deformation gradient at the end of the interval $[t_n, t_{n+1}]$.

Now, the load vectors are based on the deformed configuration:

$$\mathbf{f}^{int}(\boldsymbol{u}_{n+1}) = \bigwedge_{e=1}^{nelem} \left\{ \int_{\varphi_{n+1}(\Omega^{(e)})} \mathbf{B}^T \boldsymbol{\sigma}(\boldsymbol{\alpha}_n, \mathbf{F}(\boldsymbol{u}_{n+1}))\, \mathrm{d}v \right\} \tag{3.29}$$

$$\mathbf{f}_{n+1}^{ext} = \bigwedge_{e=1}^{nelem} \left\{ \int_{\varphi_{n+1}(\Omega^{(e)})} \mathbf{N}^T \boldsymbol{b}_{n+1} \mathrm{d}v + \int_{\partial\varphi_{n+1}(\Omega^{(e)})} \mathbf{N}^T \boldsymbol{q}_{n+1}\, \mathrm{d}s \right\} \tag{3.30}$$

where $\varphi_{n+1}(\Omega^{(e)})$ is the current deformed domain. For details of linearisation see the work of de Souza et al.

A generic iteration of NRM is, as before (see above), applied to solve the

standard linear system.

$$\mathbf{K}_T \delta \boldsymbol{u}^{(k)} = -\boldsymbol{R}^{(k-1)} \tag{3.31}$$

where $\mathbf{K}_T$ is now obtained Eq(3.32) through $\mathbf{G}$ which is the *discrete (full) spatial gradient operator.* $\mathbf{G}$, in plane stress/strain analysis, has the format given in Eq(3.33).

$$\mathbf{K_T} = \bigwedge_{e=1}^{\text{nelem}} \left\{ \int_{\varphi_{n+1}(\Omega^{(e)})} \mathbf{G}^T \mathbf{a} \mathbf{G} \, \mathrm{d}v \right\} \tag{3.32}$$

$$\mathbf{G} = \begin{bmatrix} N^{(e)}_{1,1} & 0 & N^{(e)}_{2,1} & 0 & \dots & N^{(e)}_{n_{node},1} & 0 \\ 0 & N^{(e)}_{1,1} & 0 & N^{(e)}_{2,1} & \dots & 0 & N^{(e)}_{n_{node},1} \\ N^{(e)}_{1,2} & 0 & N^{(e)}_{2,2} & 0 & \dots & N^{(e)}_{n_{node},2} & 0 \\ 0 & N^{(e)}_{1,2} & 0 & N^{(e)}_{2,2} & \dots & 0 & N^{(e)}_{n_{node},2} \end{bmatrix} \tag{3.33}$$

The fourth order tensor $\mathbf{a}$ is the *consistent spatial tangent modulus* and, in cartesian components, is defined by eq.(3.34) at the end of iteration $(k-1)$.

$$a_{ijkl} = \frac{1}{J} \frac{\partial \tau_{ij}}{\partial F_{km}} F_{lm} - \sigma_{il} \delta_{jk} \tag{3.34}$$

The main modifications for the case of finite strains are displayed in Box(3.2).

3.7 Explicit sub-algorithm

The explicit scheme (*The Central Differences Method (CDM)*) integrates directly the spatially discretised dynamic equilibrium equation at time t_n Eq(5.23). Therefore, there is a discretisation of the domain by finite elements and a discretisation in time by finite differences around a central time-point.

$$\mathbf{M}\,\ddot{\mathbf{u}}(t_n) + \mathbf{C}\,\dot{\mathbf{u}}(t_n) + \mathbf{f}^{int}(\mathbf{u}_n) = \mathbf{f}^{ext} \tag{3.35}$$

BOX 3.2: IMPLICIT INCREMENTAL PROBLEM FOR FINITE STRAINS

1. Computation of *consistent spatial tangent moduli*

$$\hat{a}_{ijkl} = \frac{1}{J}\frac{\partial \hat{\tau}_{ij}}{\partial F_{km}} F_{lm} - \sigma_{il}\delta_{jk}$$

2. Assembly element stiffness matrices

$$\mathbf{k}_T^{(e)} = \sum_{i=1}^{n_{gaus}} \omega_i \, j_i \, \mathbf{G}_i^T \, \hat{\mathbf{a}}_i \, \mathbf{G}_i$$

3. Updating of deformation gradient

$$\mathbf{F}_{n+1}^{(k)} = (\mathbf{I} - \nabla_{\mathbf{x}} \boldsymbol{u}_{\mathbf{n+1}}^{(\mathbf{k})})^{-\mathbf{1}}$$

4. Use of *constitutive integration algorithm* to update the stress and other state variables

$$\boldsymbol{\sigma}_{n+1}^{(k)} = \hat{\boldsymbol{\sigma}}(\boldsymbol{\alpha}_n, \mathbf{F}_{n+1}^{(k)})$$

$$\boldsymbol{\alpha}_{n+1}^{(k)} = \hat{\boldsymbol{\alpha}}(\boldsymbol{\alpha}_n, \mathbf{F}_{n+1}^{(k)})$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ the damping matrix, $\mathbf{f}^{int}(\mathbf{u}_n)$ the internal forces vector, $\mathbf{f}^{ext}$ the external forces vector, and $\ddot{\mathbf{u}}(t_n), \dot{\mathbf{u}}(t_n), \mathbf{u}_n$ are, respectively, the accelerations, velocities and displacements vectors. The approximations of these are carried out, as indicated above, by *central differences.* Algorithmic details of the CDM are presented in Section 2.10.1. An adaptive time step is adopted. Thus, in each temporal iteration, a long step may be intelligently elected if the stability criterion allows it, saving computational time (see Section 3.7.4).

$$\dot{\mathbf{u}}(t_{n-1/2}) = \frac{\mathbf{u}(t_n) - \mathbf{u}(t_{n-1})}{\Delta t_n} \tag{3.36}$$

Firstly, the assembly of matrices and vectors is needed for all nodes and degrees

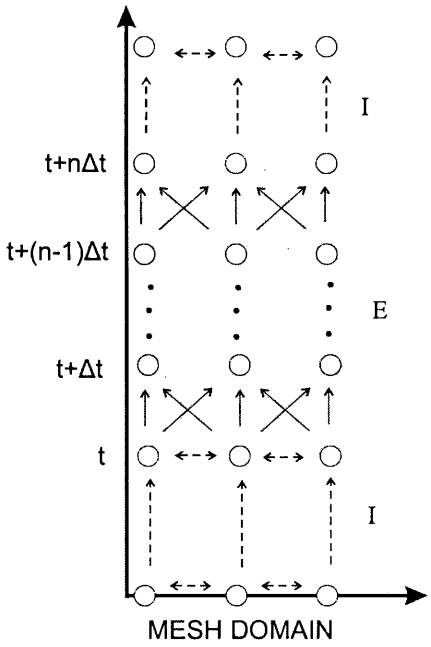


Figure 3.5: Schematic representation of the evolution of a solution executed by the I/E algorithm.

of freedom. Boundary conditions are easily managed as these are transferred

directly as displacements and/or velocities in the nodes of the boundary. Initial conditions in our combined in-time implicit/explicit algorithm are given in the section *Transition implicit/explicit.* If option 2 is elected (solving with the explicit algorithm from the beginning and no use of the implicit/explicit scheme is carried out), then initial displacements and velocities are set to zero.

3.7.1 Lumped mass matrix

A lumped mass matrix is elected because a diagonal matrix results in an uncoupled system of equations which is computationally convenient. The mass matrix is evaluated in the undeformed configuration, before the beginning of the explicit iterations.

$$\mathbf{M} = \bigwedge_{e=1}^{nelem} \int_{\Omega_{(e)}} \rho_0 \, \mathbf{N}_{(e)}^T \, \mathbf{N}_{(e)} \, d\omega \tag{3.37}$$

An approximation needs to be assumed to obtain a diagonal mass matrix. That is, to use a special lumping technique which ensures mass conservation i.e. the sum of masses in each point of the mesh must be the total mass of the body [15].Thus, one third of the mass of a triangular element is assigned to each node or one fourth to each node of a quadrilateral. Obviously, the mass of an element is the the total mass of the body by the proportion $\frac{areaofelement}{totalareaofdomain}$[6].

3.7.2 Damping matrix

The explicit in-time integration of the discretised space equations of dynamic equilibrium can exhibit strange and annoying oscillations. These can be controlled by damping values. Furthermore, in highly nonlinear materials it is convenient to use a damping which can vary with the stiffness [6]. The damping

[6]This is introduced to allow unstructured-mesh-domains to preserve the total body mass

matrix is elected proportional to diagonal matrix $\mathbf{L}$ to provide computational advantage. This leads to a system formed by uncoupled equations.

$$\mathbf{C} = \alpha \mathbf{L} \tag{3.38}$$

Most commonly used is lumped mass matrix proportional damping ([94] or [64]) $\mathbf{C} = \alpha \mathbf{M}$. There are other options such as Munjiza's proposal [78] which includes stiffness and mass relation to obtain damping values.

$$\mathbf{L} = \mathbf{M}(\mathbf{M}^{-1}\mathbf{K})^m \tag{3.39}$$

If $m = 1$ a stiffness proportional damping is introduced, which results in high frequencies being damped. However, the critical time step decreases with increasing damping, leading to increasing computational cost. All frequencies are damped effectively if $m = 0.5$, and we can tune m to adapt to our problem requirements. In this section it is almost obligation to mention Rayleigh damping which is widely employed in engineering applications and is a linear combination of mass and stiffness. This damping can be tuned to damp high or low frequencies. There are many references treating this damping, among them are [94] or [64].

Mass Proportional Damping

An effective manner to develop mass proportionality has been studied and implemented in the explicit code. This consists in calculating the natural frequencies of each node and degree of freedom (named i by simplicity in the equations) as a function of a stiffness approximation to each of them as indicated in Eq (3.40).A discussion about stabilization of numerical computations via the introduction of artificial damping is presented in [90].

$$C_i = 2\xi\omega_i M_i \tag{3.40}$$

where ω_i is given (after homogeneous system of equations solution; see Section 3.7.4) by

$$\omega_i^2 = K_i/M_i \tag{3.41}$$

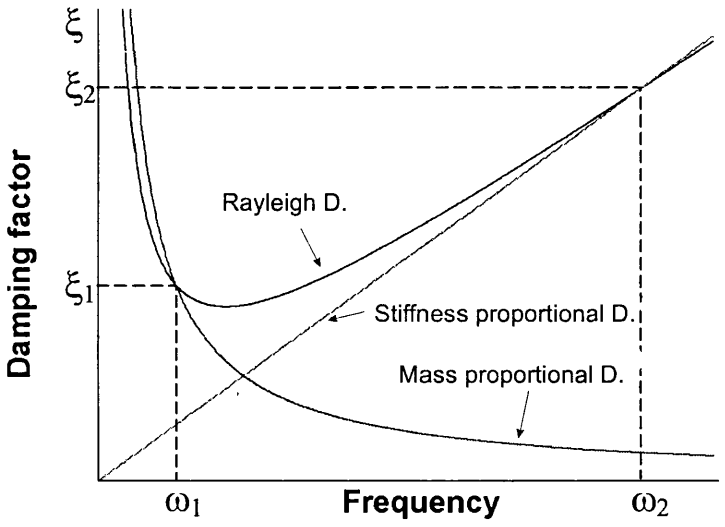


Figure 3.6: Dampings

Damping proportionality constant

Numerical damping proportionality constant α can be chosen from the dynamic relaxation method. This is convenient for problems with high non-linear material and/or geometric behavior [94]. In addition, this damps a wide range of frequencies with equivalent efficiency [7]. The dynamic relaxation method,

Section 2.10.2, computes every deformation mode to define an equivalent frequency. This method chooses the smallest frequency in order to achieve a critical proportionality constant as follows:

$$\omega_{min} = min \begin{cases} \frac{\mathbf{u^T K u}}{\mathbf{u M u}} \\ \\ \frac{1}{\Delta t} \end{cases}$$

Then, the damping is defined under minimum frequency proportionality:

$$\mathbf{C} = 2\,\omega_{min}\,\xi\,\mathbf{B} \tag{3.42}$$

3.7.3 Damping and velocity of convergence

Why damping is important? this question should be answered from two points of view. Firstly, physically, damping is the term associated with the loss of energy in the vibration of a mechanical system or body [56]. This loss of energy is difficult to measure experimentally. It is generally accepted that a force which is dependent on the velocity of the system response will dissipate energy. All above consider dynamic systems, in our problem we are interested in the solution of a quasi-static problem considering damping and inertial forces in the dynamic momentum equation. This seems artificial, posing dynamic equations for static systems, but it permits solution of complicated problems provided that the time step was under a critical value as it will be described in this thesis. Dynamic terms must vanish in the converged static solution. For that a basic form of damping is the viscous one which is velocity proportional form of representation of that loss of energy. Viscous damping has been ex-

tensively used in this study because we model mechanical system [7] and ,also , by its simplicity. Secondly, from a pure mathematical perspective, damping is an factor that influences the velocity of convergence, when used in a system of ordinary differential equations corresponding to a system in equilibrium. Latter on, an artificial and conveniently elected damping will be used in the formulation of a quasi-static system of differential equations. In the rest of chapters of this study and in its programming by explicit methods, a initial study of every particular problem has been carried out in order to determine the range of optimum damping. This range can be calculated taking relevant equations[8] and testing this single equation in a simplified form with different damping coefficients. Although, this system is quasi-static, the accelerations and velocities terms are introduced and will vanish in the static solution. The damping role is getting an optimum velocity of convergence. If the damping is over-damped the solution to the quasi-static problem can be of such slowness that computationally is not recommendable. The reverse , under-damped system, maintains similar with oscillations that ,in very under-damped systems, seem they never will stabilize in a solution to a static equilibrium problem. To test the above stated, we can use the mechanism with one degree of freedom (see Figure(3.7)) compounded for a spring ($k = 5 \cdot 10^8 N/m$), a damper (with variable damping coefficient), a particle ($m = 100kg$) and a constant external force ($F = 2000N$). It is clear that the solution is static as the force is not increasing and there are not other loadings considered. The differential equation is shown in Eq(3.43) and the solution is carried out by finite differences. The scheme is represented below.

[7] Also, energy loss by acoustic radiation or fluid-structure interaction can be effectively modeled by viscous damping [56]

[8] E.g, the equation corresponding to the variable associated with a nonlinearity

$$m\ddot{x} + c\dot{x} + kx = F \tag{3.43}$$

where x is the distance between the particle and the wall.

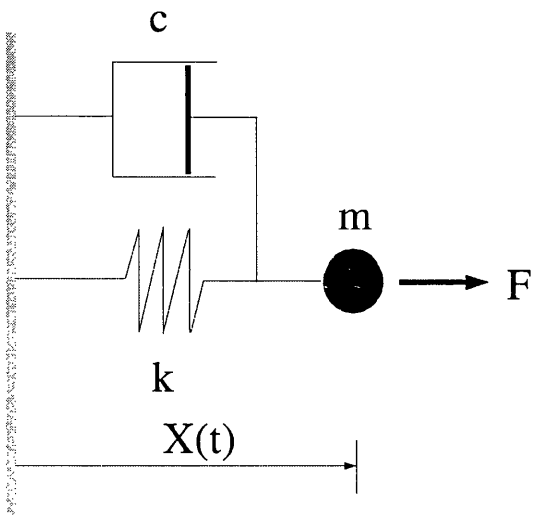


Figure 3.7: System with a one degree of freedom

Several damping were used to see the rapidness in which the static solution to the system of Figure(3.7) was reached. The results are plotted in Figure(3.8). Observing the plots for damping coefficients in the range $[10^5, 10^6]$ and overall for the right boundary of this interval, the static solution is reached between 0.008 and $0.009s$. A less value of damping decreases oscillations peaks slowly and in complex systems of equations it could lead to an unconditioned system. For excessive values of damping (e.g. 10^8), the value of x in a static (in a fluid, it would be the solution to the steady flow) convergency to the solution of equilibrium might take a lifetime!!.

BOX 3.3: EXPLICIT ITERATIVE PROCESS FOR THE ONE D.O.F. SYSTEM

(i)-Loop over time steps

$$t_{n+1} = t_n + \triangle t; \qquad c_2 = \frac{m}{\triangle t} + \frac{c}{2}; \qquad c_3 = \frac{m}{\triangle t} - \frac{c}{2}$$

$$\triangle t = \alpha \sqrt{\frac{k}{m}} \qquad f^{int}(t_n) = k\, x(t_n)$$

(ii)-Solve by central velocities

$$\dot{x}(t_{n+\frac{1}{2}}) = \frac{F - f^{int}(t_n) + c_3\, \dot{x}(t_{n-\frac{1}{2}})}{c_2}$$

$$\dot{x}(t_n) = \frac{\dot{x}(t_{n+\frac{1}{2}}) + \dot{x}(t_{n+\frac{1}{2}})}{2}$$

(iii)-Update displacement

$$x(t_{n+1}) = x(t_n) + \triangle t\, \dot{x}(t_{n+\frac{1}{2}})$$

(iv)-Prepare next iteration

$$t_n = t_{n+1}; \qquad x(t_n) = x(t_{n+1}); \qquad \dot{x}(t_n) = \dot{x}(t_{n+1})$$

(v)-IF stop criterium is satisfied THEN exit, ELSE go to (i).

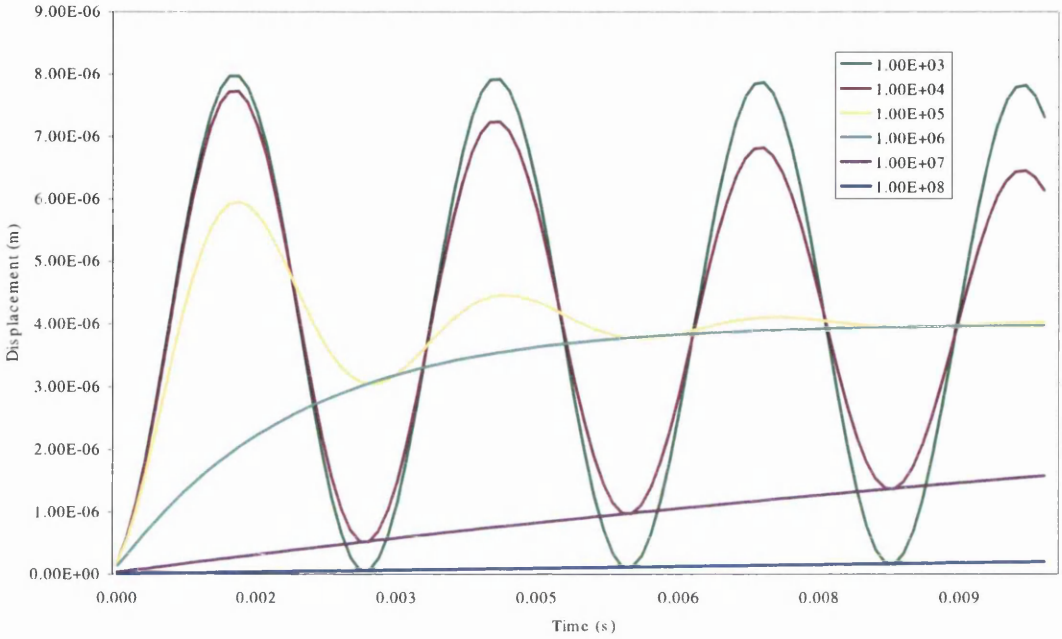


Figure 3.8: Solution by finite differences of the one d.o.f. mass-damping-spring mechanism with different damping coefficients

3.7.4 Time step criteria

The step time for the CDM must be less than a specific value to guarantee the stability of the scheme and therefore its convergence. This value is bounded by the natural frequencies and damping ratio relationships as stated in Eq (3.44).

$$\Delta t \leq min \frac{2}{\omega_i}(-\xi_i + \sqrt{1+\xi_i^2}) \tag{3.44}$$

where ω_i are the natural frequencies and ξ_i the fraction of critical damping at each node i of the mesh. This inequality is satisfied if the maximum frequency in the mesh is elected. The maximum frequency can be calculated knowing the maximum eigenvalue of the mesh as $\omega_{max} = \sqrt{\lambda_{max}^{(mesh)}}$. Moreover, the maximum eigenvalue can be bounded by the maximum element eigenvalue $\lambda_{max}^{(mesh)} \leq \lambda_{max}^{(e)}$, Irons et al. [61],[62]. The eigenvalue computation adds more

processor time, so an alternative practical way is to invoke formulas which can bound adequately the step time for some problems. Thus Eq (3.45) is obtained for linear problems. When nonlinearities emerge a reduction of this upper-bound must be considered (typically the range goes from 80 to 98 per cent, [11]) taking into account the type of problem.

$$\Delta t \leq min \frac{L_e}{c_e} \tag{3.45}$$

$$c_e = \sqrt{\frac{E^{(e)}}{\rho^{(e)}}} \tag{3.46}$$

where L_e is a characteristic element length which is frequently adopted as the minimum length in the smallest element of the mesh and c_e the wave-speed Eq (3.46). Thus, we may see that critical the time step decreases with mesh refinement and increasing stiffness of the material. The ratio of critical time step needed is called Courant's number in the literature. Therefore, the time cost of execution depends basically on the element size and nonlinearity of material. The frequencies for all nodes and degrees of freedom in the mesh are approximately calculated by the code. The maximum frequency bounds the critical time step Eq(3.49). A particular expression is usually chosen, depending of the type of analysis. Eq (3.47) for plane strain problems or Eq (3.48) for plane stress.

$$\Delta t \leq L_e \sqrt{\frac{\rho(1+\nu)(1-2\nu)}{E(1-\nu)}} \tag{3.47}$$

$$\Delta t \leq L_e \sqrt{\frac{\rho(1-\nu^2)}{E}} \tag{3.48}$$

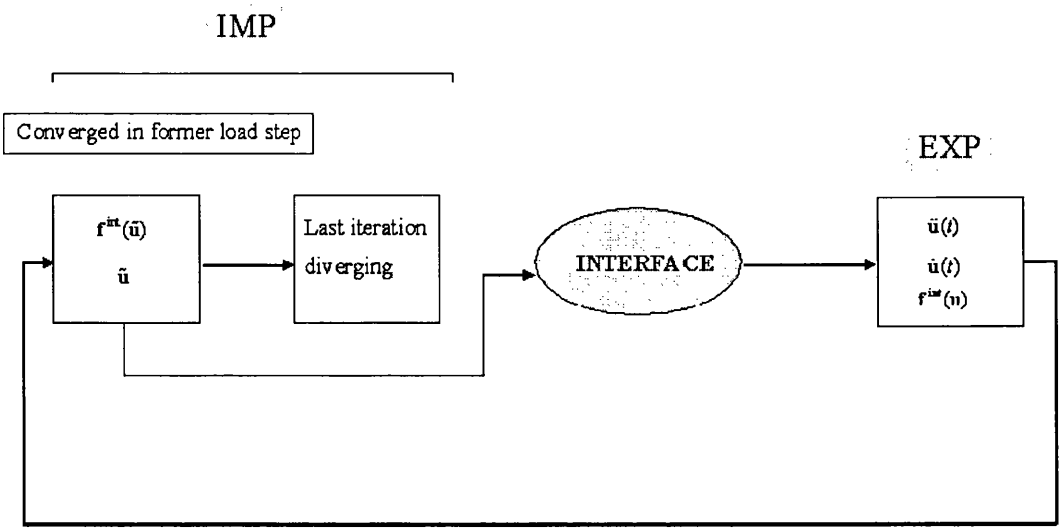


Figure 3.9: Schematic details of the strategy where the swapping between sub-algorithms is remarked

These steps of time are very restrictive and not adapted to new conditions of the next iteration. Step time adaptivity has been utilized in the programming, it attempts to perform a proper step time under the conditions of the natural frequencies of the system at this time point [9]. Thus the critical time step is selected as,

$$\Delta t\,(t_{n+1}) = \frac{2}{max_i\{\omega_i(t_n)\}} \tag{3.49}$$

This statement has given place to a faster performance than using a constant time step as Eqs (3.47) and (3.48). The natural frequencies are determined from homogeneous problem Eq (3.50). Its analytical solution is in the form $u(t) = \widetilde{u}e^{-j\omega t}$ $(j = \sqrt{-1})$, substituting in Eq (3.50) an eigenvalue problem is accomplished.

[9] A similar scheme to calculate critical time steps is given in [82] for central differences and contact problems. The critical time step is shown as a function of the contact conditions in this approach

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{K}\mathbf{u} = 0 \tag{3.50}$$

introducing the analytical solution form,

$$\mathbf{u}(t) = \tilde{\mathbf{u}} \exp(-j\omega t)$$

$$\dot{\mathbf{u}}(t) = -j\omega\tilde{\mathbf{u}} \exp(-j\omega t)$$

$$\ddot{\mathbf{u}}(t) = -\omega^2\tilde{\mathbf{u}} \exp(-j\omega t)$$

which gives after introduction to :

$$-\omega^2\mathbf{M}(cos(\omega t) - jsin(\omega t)) + \mathbf{K}(cos(\omega t) - jsin(\omega t)) = 0$$

which leads to the classical eigenvalue problem

$$|-\omega^2\mathbf{M} + \mathbf{K}| = 0$$

where ω^2 are the eigenvalues of the system that provide the natural frequencies for each node and degree of freedom, i.e. each variable.
In this explicit code the stiffness matrix is never computed and an approximation is done in order to know the natural frequencies,

$$K_{ii}(t_n) \simeq \frac{f_i^{int}(u_n) - f_i^{int}(u_{n-1})}{\dot{u}_i^{n-1/2}\Delta t_n} \tag{3.51}$$

with this approximation the stiffness matrix becomes diagonal and the computational cost of the eigenvalue problem is saved because the natural frequencies may be easily calculated from Eq(3.52).

$$\omega_i(t_n) = \sqrt{\frac{K_i(t_n)}{M_i}} \tag{3.52}$$

and the critical step time is selected as stated in 3.49.

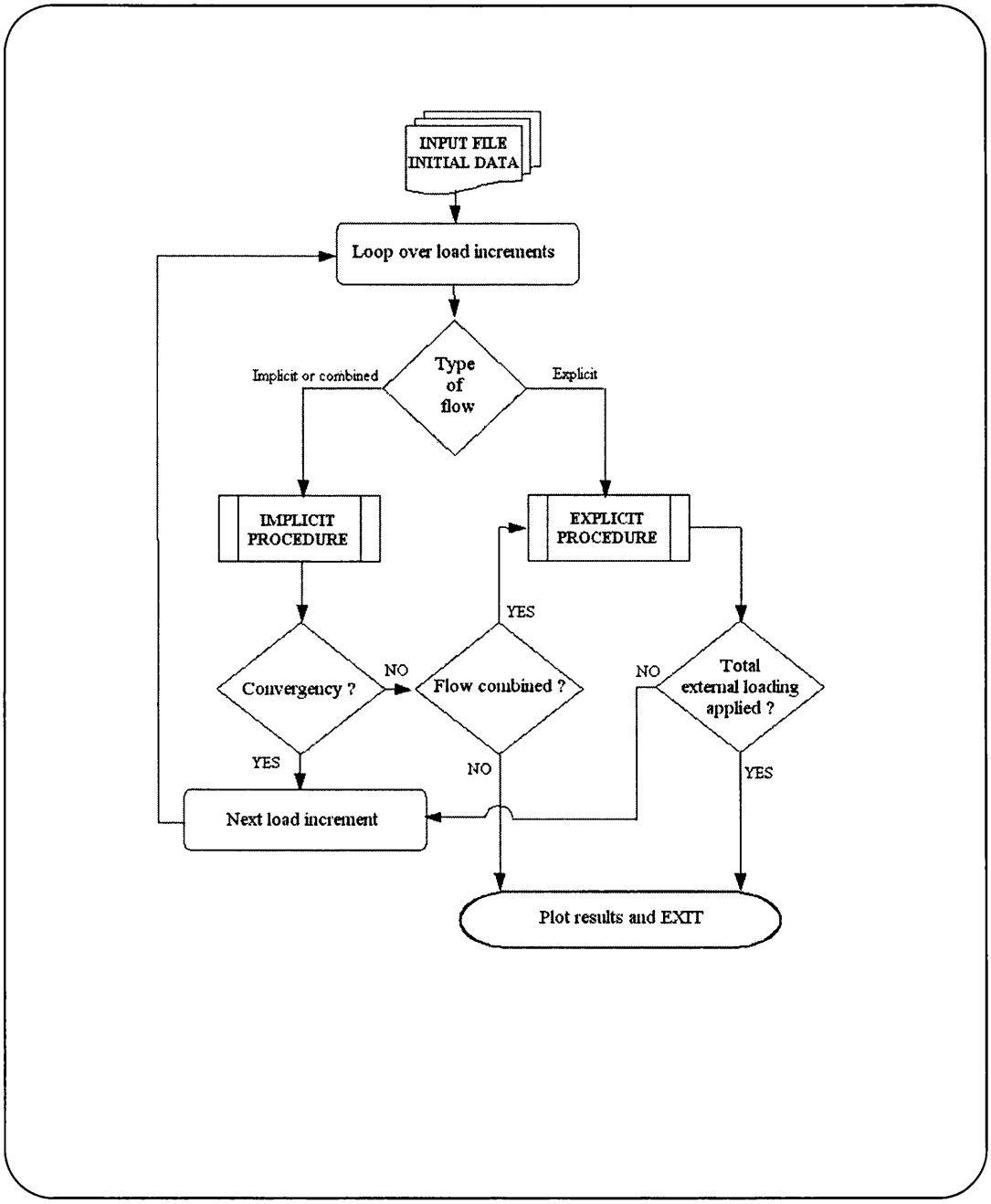


Figure 3.10: Simplified flowchart of the in-time implicit/explicit algorithm

3.8 Transition implicit/explicit

The last converged iteration at IMP (displacement $\tilde{\mathbf{u}}$) is transferred to EXP as initial conditions. That means we have displacements at $t = 0$ equal to the displacements obtained in the last iteration of the implicit method which generally will not be an equilibrium point. Thus, the final internal forces and displacements, from IMP, are used to determine initial accelerations and velocities for EXP.

$$\begin{aligned} \mathbf{f}^{int}(\tilde{\mathbf{u}}) &\rightarrow \mathbf{f}^{int}(0) \\ \tilde{\mathbf{u}} &\rightarrow \mathbf{u}(0) \end{aligned}$$

Initial velocities for EXP are approximated as follows:

$$\begin{aligned} \ddot{u}_i(0) &= \frac{f_i^{ext} - f_i^{int}(0)}{M_{ii}} \\ \dot{u}_i(0) &= \ddot{u}_i(0)\,\Delta t(0) + \dot{u}_i^{-}(0) \\ \dot{u}_i^{-}(0) &= 0.0 \end{aligned}$$

where $\Delta t(0)$ is the initial time step. After the first iteration, an adaptive time step is carried out. Diagram of the I/E algorithm is represented in Figure (3.11). Once the solution is reached the flow is returned back to the implicit sub-algorithm if the external load is not still totally applied (otherwise, the execution is ended). Obviously, the converged solution values of the explicit sub-algorithm are passed on to the implicit one. An update of configuration is also carried out in the case that large deformations have been produced.

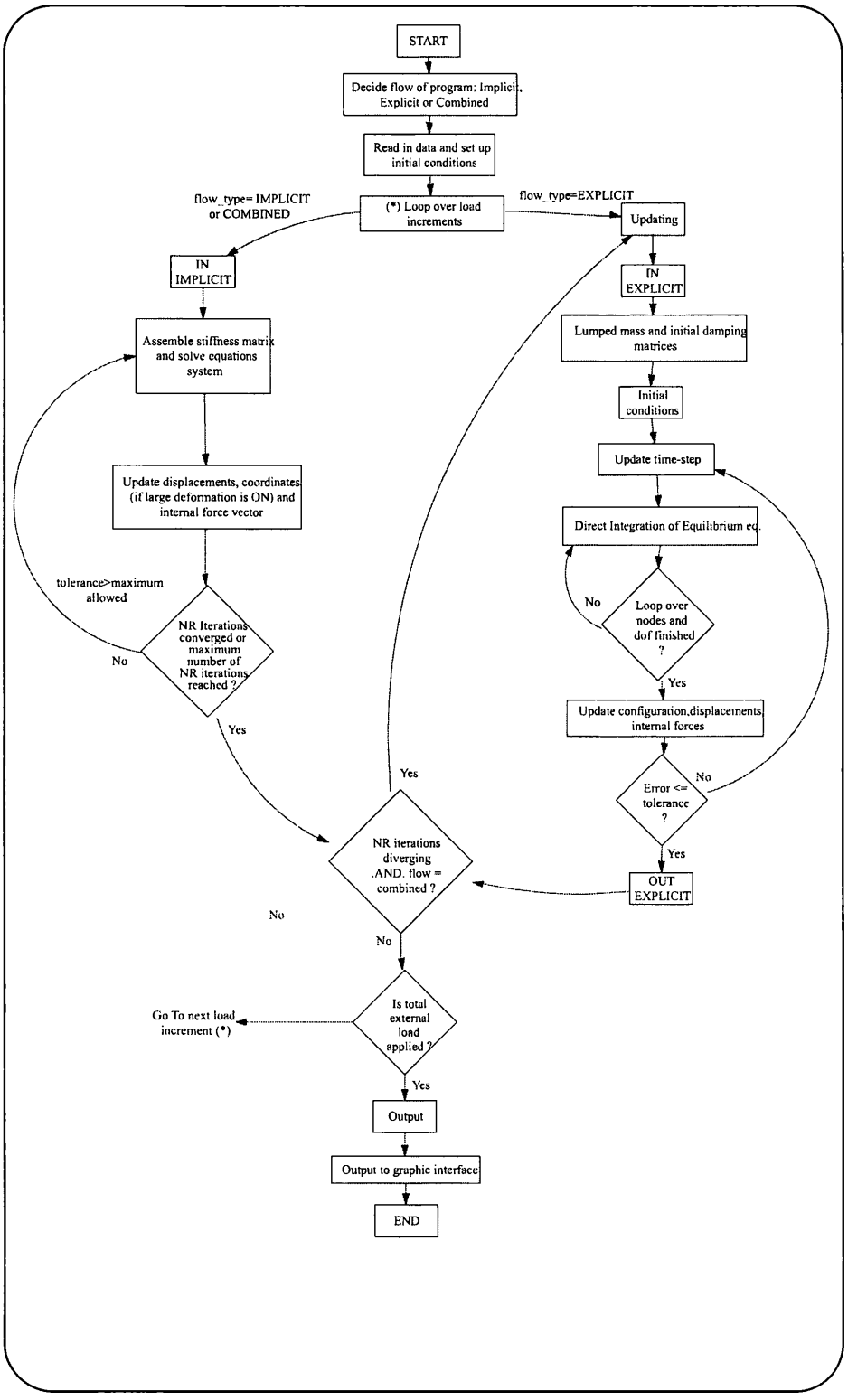


Figure 3.11: Combined implicit/explicit algorithm.

3.9 Numerical tests

A set of examples in 2D have been analyzed in order to confirm the code, among them a plate under traction (fig.3.12) and Cook's membrane. In the first one, undamped and damped cases are studied. The plate was uniquely solved by the explicit code to validate the explicit sub-algorithm with this example, and the remaining tests by the combined explicit-implicit scheme in-time (I/E).

3.9.1 Test 1: Elastic undamped plate

This example was one of the tests used to validate the explicit code. Therefore, the solutions displayed in this section refers to explicit only. The geometry of Figure 3.12 (2D-plane strain) is loaded by a constant force in the right-side corners. The structure has the following properties and characteristics: density $\rho = 10^3 Kg/m^3$, Young modulus $E = 10^4 MN/m^2$, Poisson ratio $\nu = 0.0$, cross-section area $A = 0.1m^2$.

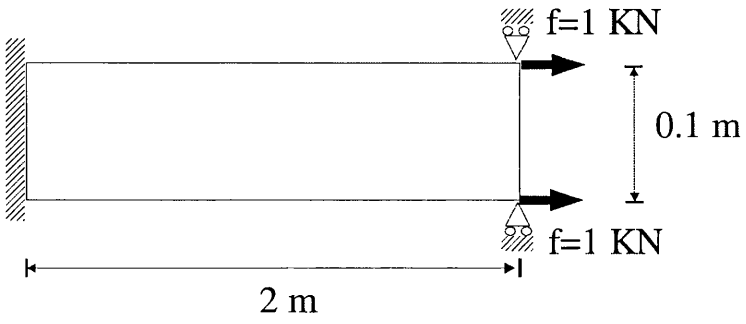


Figure 3.12: Plate under traction forces.

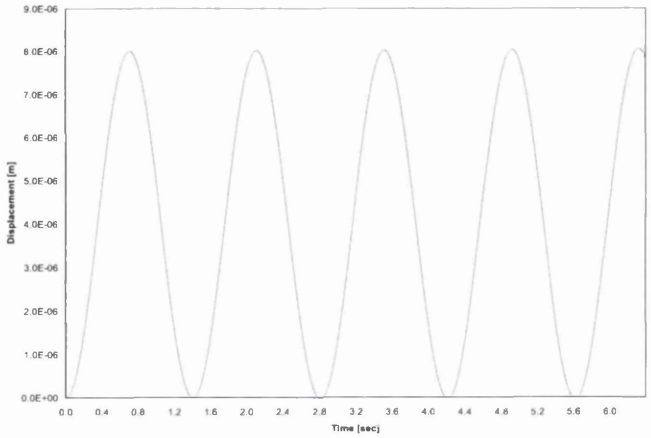


Figure 3.13: Right-side displacement (undamped system).

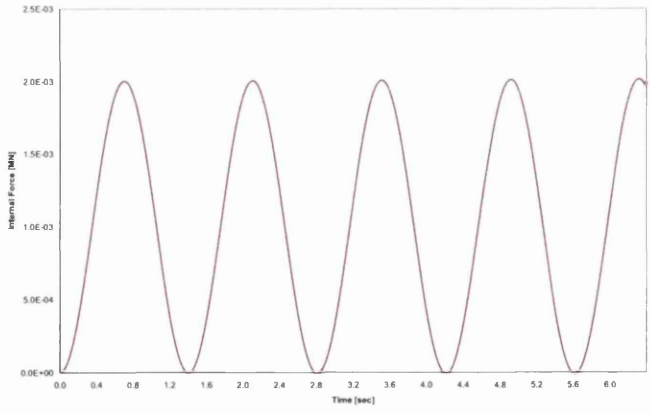


Figure 3.14: Nodal internal force (undamped system).

3.9.2 Test 2: Analysis of plate considering damping

A damping is used in the same structure to obtain the final solution of the quasi-static problem. It is a mass proportional damping as stated above. As is observed in Figure (3.15) the amplitude is decreasing with time converging to the final solution. Right-side displacement solution is reached at $0.004mm$ (i.e. nodes situated at right side).

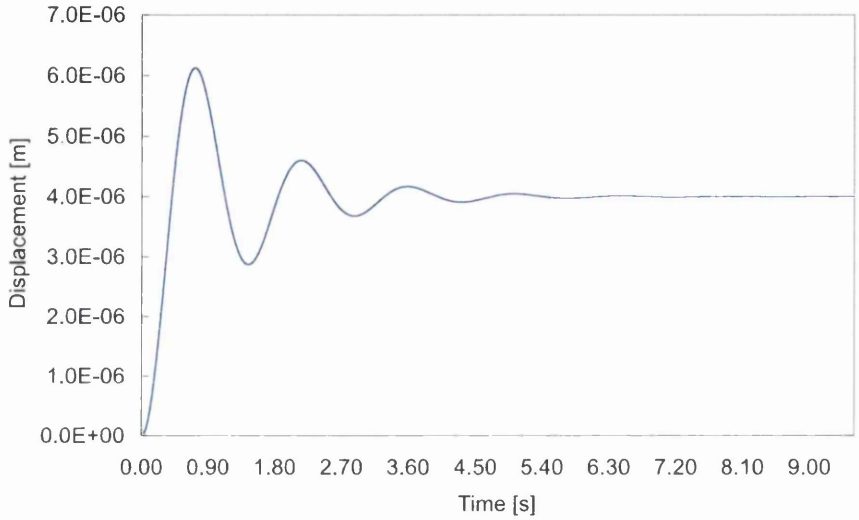


Figure 3.15: Right-side displacement (mass proportional damping).

A distribution of displacements is given by the interface program HYPTOGID (which has been created by the author of this thesis in order to obtain image results from the numerical output of the principal program).

3.9.3 Test 3:Large deformation analysis. Cook's membrane

A non-linear material model such as Ogden model [10] for hyperelastic –rubber-like– materials, is an option to assess a code with these characteristics as divergency of implicit methods is almost guaranteed if there is no load control. In our problem, the load is totally applied from the first time point. The implicit code demonstrates divergence if the load is not divided in smaller increments. Thus, the code switches to the explicit scheme after detecting that the implicit solution is diverging according to the criterion displayed in Box (3.1). The geometry in Figure (3.16) corresponds to Cook's membrane problem which is generally used to assess the convergence in elements which are nearly incompressible under shear and bending strains [105] or [1]. Obviously, this is not our problem, however it is an useful exercise to use the I/E algorithm because if the distributed shearing force applied on the right edge is not divided in partial loadings as stated above, NRM diverges, and at that point execution by EXP sub-algorithm may be carried out. Thus, the I/E algorithm can be assessed.

The membrane is clamped in the left edge and a shearing force of $100N$ is distributed on the right side, Figure (3.16). Ogden material is assumed with bulk modulus $\kappa = 40.0942\text{x}10^4$ and shear modulus $\mu = 80.1938$. Two simulations were principally considered. Firstly, use of IMP sub-algorithm alone allowing partial loadings (to avoid divergence) until the total load is applied. And secondly, total load applied from the beginning by I/E. Thus, four IMP iterations (Newton-Raphson Method) were executed and, then, the EXP sub-

[10]This model is only used here. Interested readers are refered to [81].

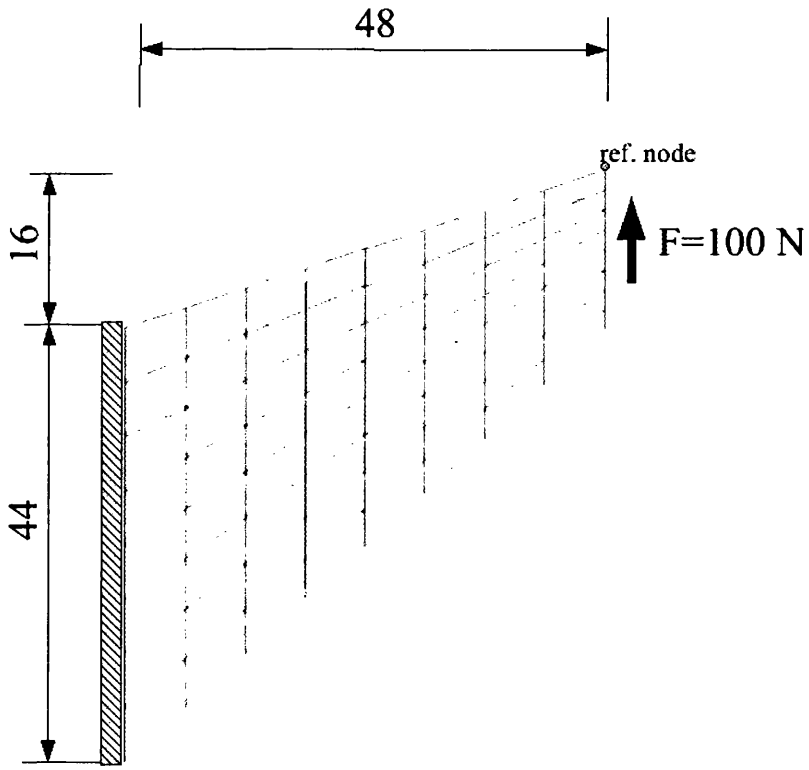


Figure 3.16: Cook's membrane (dimensions in mm). Geometry and finite element mesh in the undeformed configuration

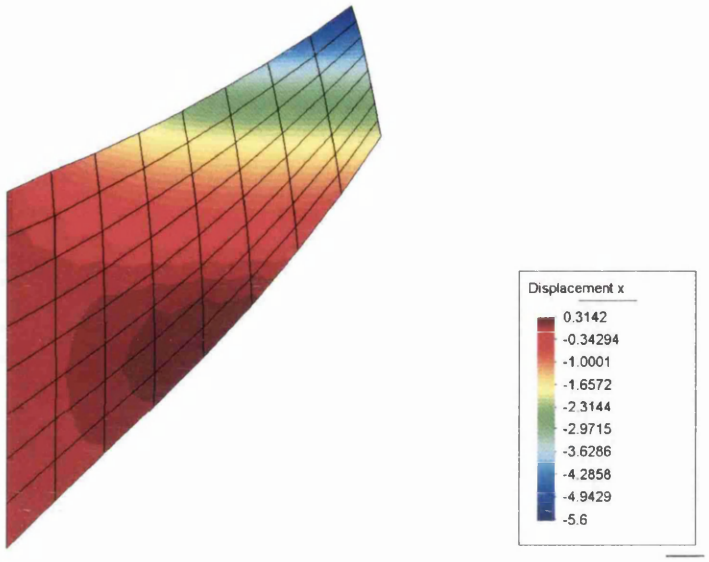


Figure 3.17: Displacement $u_x\,(mm)$ by I/E

algorithm run to reach solution with an error estimator equal to 0.2% . The solutions obtained by I/E for displacements and stresses may be visualized in Figures (3.17, 3.19, 3.21, 3.23, 3.25). Note that explicit iterations were stopped when the residual error-norm was 0.2% (this may be observed in Figure (3.26) where nodal horizontal velocities have an appreciable value). Our interest is assessing the transition between sub-algorithms more than to reach a really low error, for that to allow more run-time is simply needed. Therefore slight differences appear in the plots.

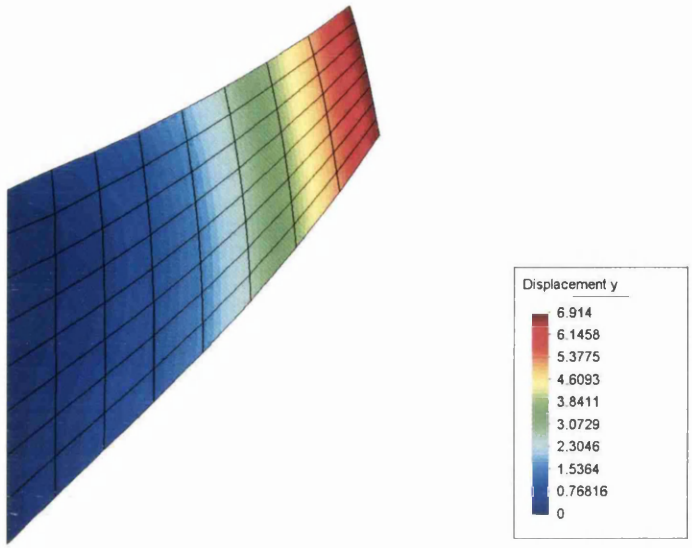


Figure 3.18: Vertical displacement $u_y\,(mm)$ by IMP

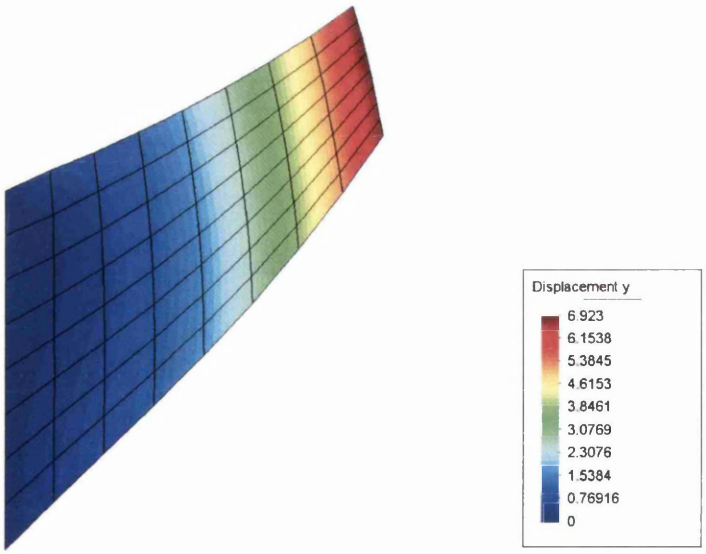


Figure 3.19: Vertical displacement $u_y\,(mm)$ by I/E

The test result is in good agreement with the IMP solution and with the

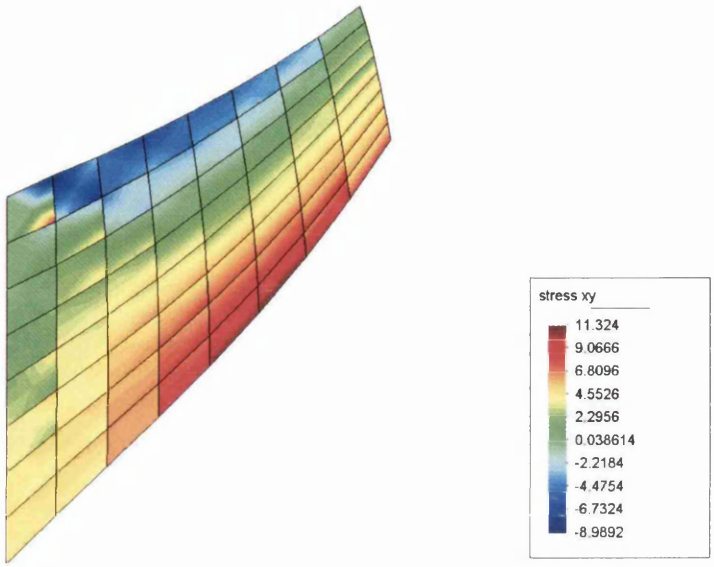


Figure 3.20: Shear stress $\sigma_{xy}\,(N/mm^2)$ by IMP (division of total external loads by increments).

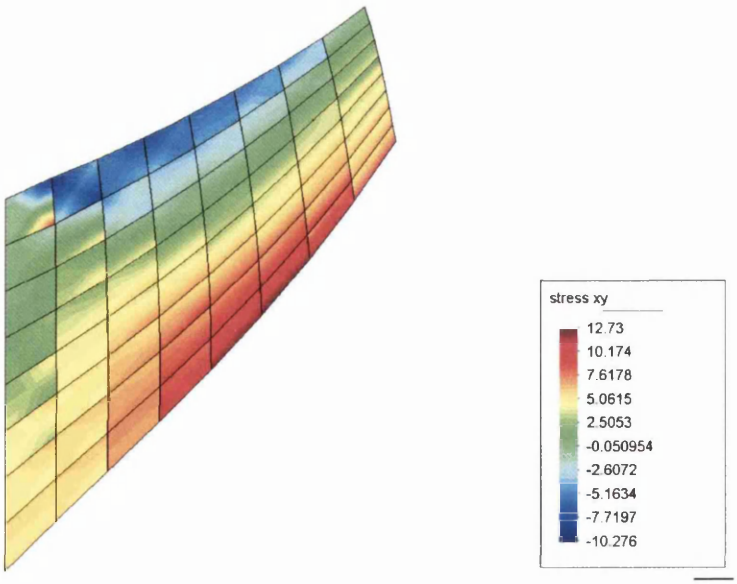


Figure 3.21: Shear stress $\sigma_{xy}\,(N/mm^2)$ by I/E

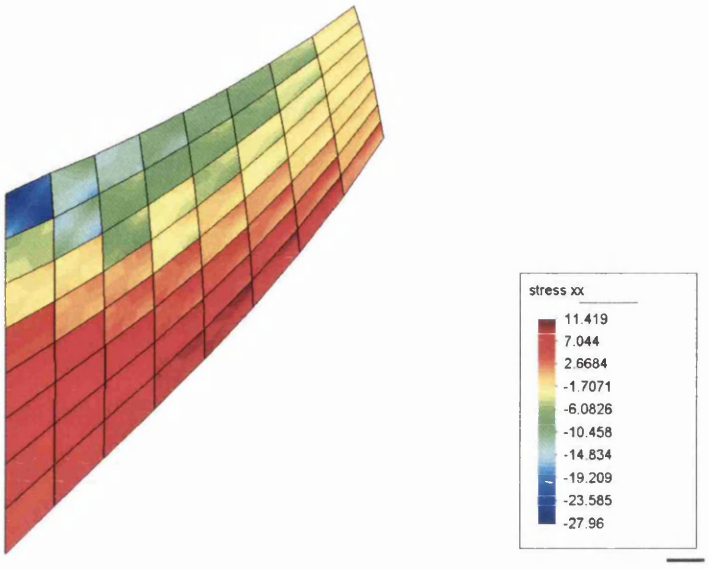


Figure 3.22: Stress $\sigma_{xx}\,(N/mm^2)$ by implicit method

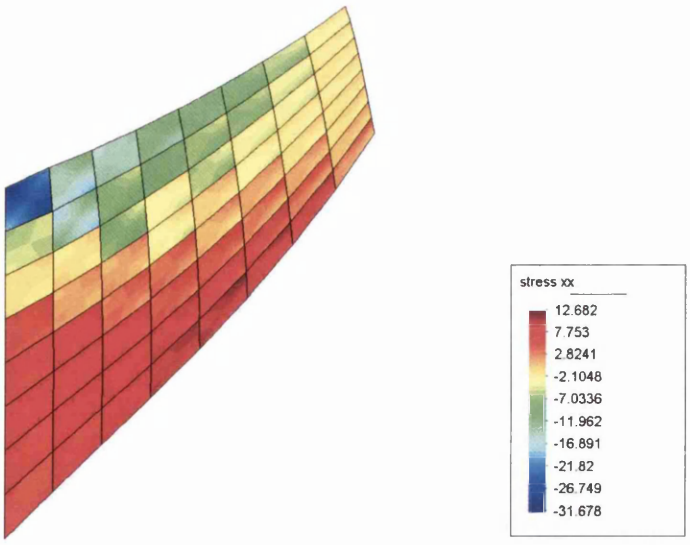


Figure 3.23: Stress $\sigma_{xx}\,(N/mm^2)$ by I/E

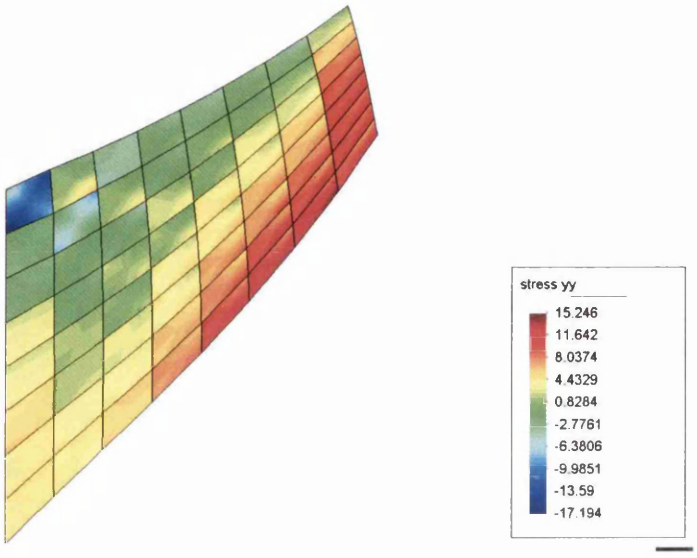


Figure 3.24: Stress $\sigma_{yy}\,(N/mm^2)$ by implicit method

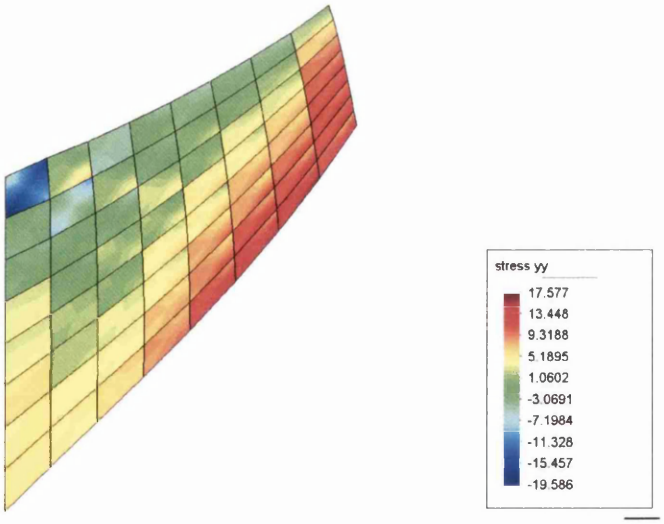


Figure 3.25: Stress $\sigma_{yy}\,(N/mm^2)$ by I/E

reviewed literature, see for example [1]. Simo et al. [105] uses an IMP algorithm with control load of $\Delta F = 1.0\,(elastic)$ $\Delta F = 0.5\,(elastic - plastic)$ ranging from 0 to 100 obtaining a displacement at the ref. node (right top corner in Figure (3.16) of approximately $6.9\,mm$.

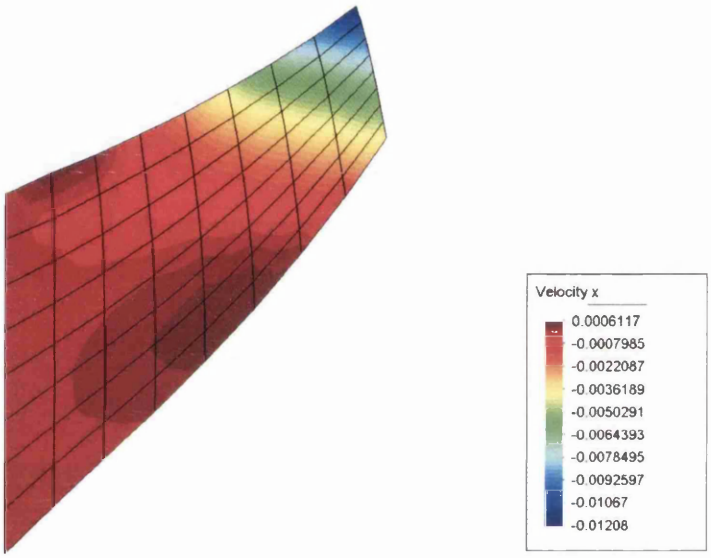


Figure 3.26: Velocity $v_x\,(mm/s)$ by I/E

3.9.4 Test 4: Buckling of a hyperelastic arch

Analyses that diverge with NRM are presented in order to validate the I/E algorithm. Thus, the buckling of an aluminium alloy (Young modulus $E = 6.895 \cdot 10^4\,MPa$, poisson ratio $\nu = 0.34$ and density $\rho = 2700\,kg/m^3$) arch is presented. An external point load (up to a magnitude $F = 4000\,N$ that causes the snap-through of the arch) is applied in the middle of the arch as shown in Figure (3.27). Tests with other point forces are also presented. The range of

the external loads includes buckling and no-buckling behaviours. The arch is totally clamped at the two edges. Other geometric values are cross-section area $A = 25,806 \cdot 10^{-4}\, m^2$, inertia moment $I = 5.54 \cdot 10^{-7}\, m^4$, thickness $t = 0.0508\, m$, and arch radius $R = 5.08\, m$.

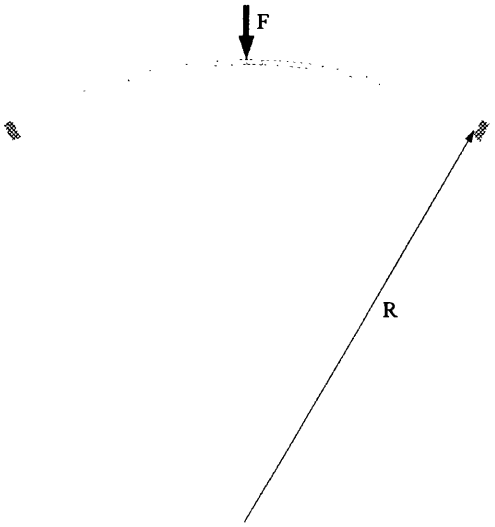


Figure 3.27: Geometry of the fixed-at-edges arch

This problem has been widely studied. Calhoun et al. [21] or Wen et al. [126] simulated it in the pre-buckling behavior. A further development was performed by Yong-Lin Pi and N. S. Trahair [95] carrying out simulations after the buckling point. The central load is increased up to $4000\, N$. In the simulations, NRM began to diverge at deflection $|\delta_{crit}| = 0.076\, m$ (which corresponds to an internal nodal force in the vertical direction $\mathbf{f}^{int}_{central\, node,y} = 2781.917 N$), and ,hence, the EXP sub-algorithm is initiated (with the last converged solution of NRM indicated at Fig.(3.28) as point (3)) and run.

Flow	i/t_n	Error(%)	Max. Resid.	$\mathbf{f}_y^{int}(k_{cn})$	$\mathbf{u}_y(k_{cn})$	Stage
IMP	1	3,118.820	197,857.00	-	-	-
IMP	2	59.346	3,767.75	-7,220.6	-0.061	diverging
IMP	3	3,116.590	202,416.00	-4,015.2	-0.074	swap
EXP	0.0001	73.404	1,616.89	-3,477.9	-0.074	oscillat.
EXP	0.001	64.204	785.12	-4,389.4	-0.075	oscillat.
EXP	0.1	2.386	336.19	-3,971.2	-0.466	oscillat.
EXP	0.5	0.092	29.74	-3,998.5	-1.193	converging
EXP	0.641	0.033	9.98	-3,999.3	-1.17	converging
EXP	1.0	0.007	2.016	-4,001.27	-1.17	converging
EXP	1.5	10^{-3}	0.293	-4,000.16	-1.17	converging
EXP	2.0	$4.4 \cdot 10^{-5}$	0.017	-4,000.01	-1.17	solution

Table 3.1: Relative residual norm(%) (error(%) in the table) and maximum residual. Vertical internal force (N) and displacement (m) at central node of the arch(k_{cn}) . In the second column, the iteration number(i) is shown for the IMP sub-algorithm and the time (t_n) for the EXP sub-algorithm

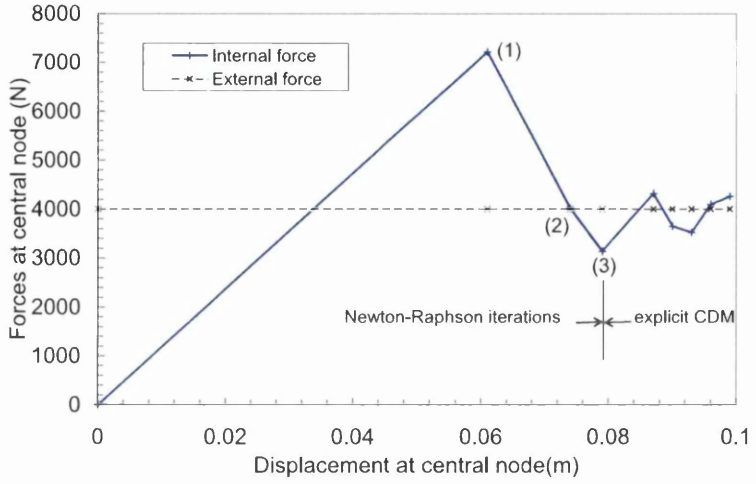


Figure 3.28: Transition after three NRM iteration. In the fourth one, it diverges and the EXP sub-algorithm is initiated (central displacement in absolute value)

The execution of the explicit code may be observed in Figure (3.29). Convergence to solution $|\delta_{sol}| = 1.17\,m$ (absolute value of deflection at the central node) is reached. Furthermore, several numerical tests were executed for smaller external forces than $4000\,N$. The resultant load-displacement curve is plotted in Figure (3.31), where the snap-through effect due to buckling is observed. Each marked point corresponds to an applied external force. Obviously, the algorithm did not swap from implicit to explicit sub-algorithms while no buckling had occurred i.e. in the first part of the curve, before the maximum. The swap, as mentioned above, is only carried out if the implicit sub-algorithm is not able to converge. The connection of these discrete points plots the snap-through curve numerically obtained with the I/E algorithm.

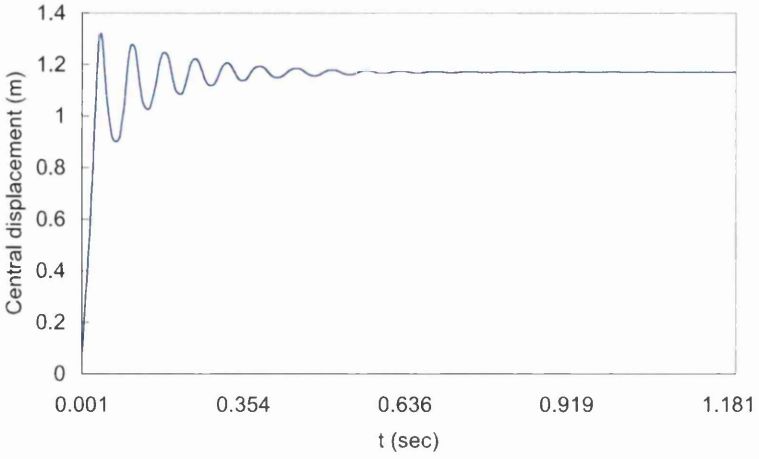


Figure 3.29: Absolute value of deflection at the central node. It starts when EXP is initiated after three IMP iterations (see Fig 3.28)

3.9.5 Test 5: Elasto-plastic circular arch

A circular arch, loaded at the centre and clamped at the two edge-supports, is analysed with the I/E code. No rotation is permitted at the supports. The finite element model consists of 20 quadrilateral elements with four nodes. The numerical integration is performed on 4 gauss points in the finite element. A plane stress state is considered with the uniform thickness of the arch set to $10\,cm$. The geometry is shown in Figure (3.35). The material is a Von-Mises perfectly plastic material (see material characteristics in Tables (3.2,3.3)).

Material properties	
Density	$7,860 \cdot 10\ \frac{kg}{cm^3}$
Young Modulus	$210 \cdot 10^5\ \frac{N}{cm^2}$
Poisson ratio	0.3

Table 3.2: Material properties of the elastic-plastic material

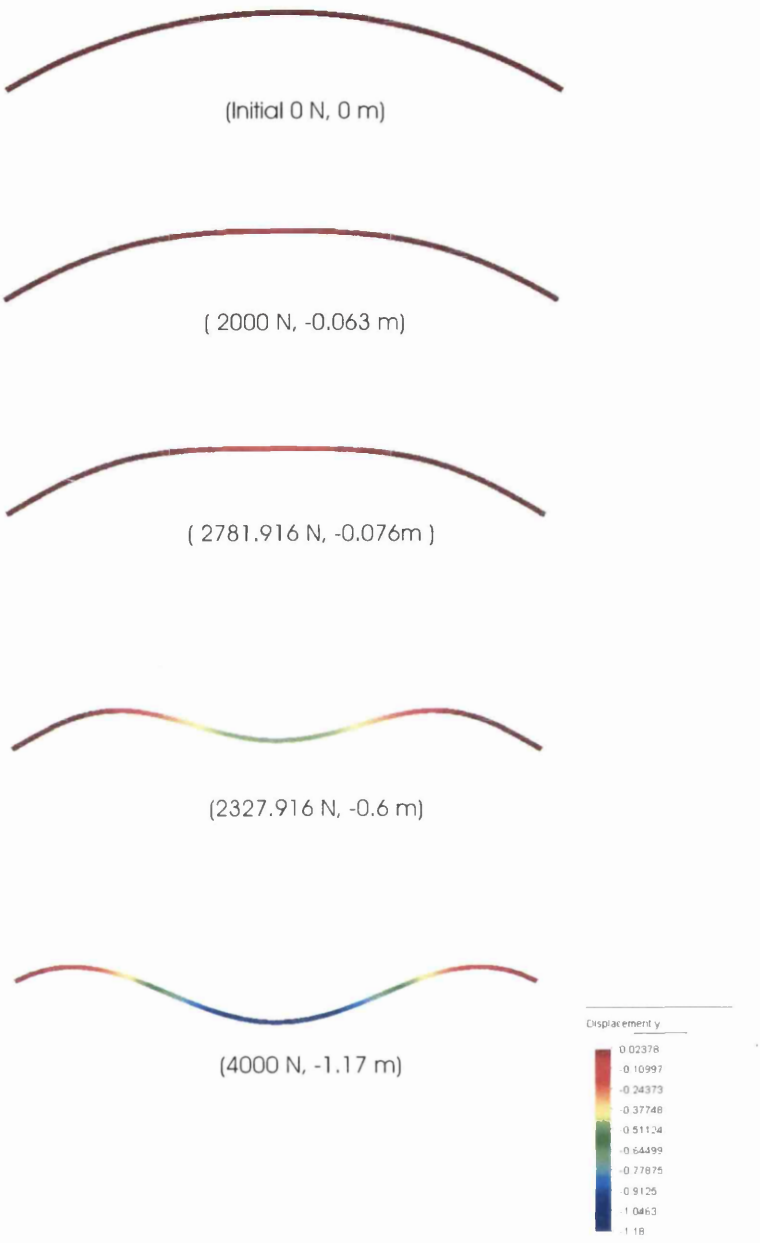


Figure 3.30: Deformed configurations and y-displacement field for different external point loads at midpoint. Nodal internal force and deflection at central node in brackets

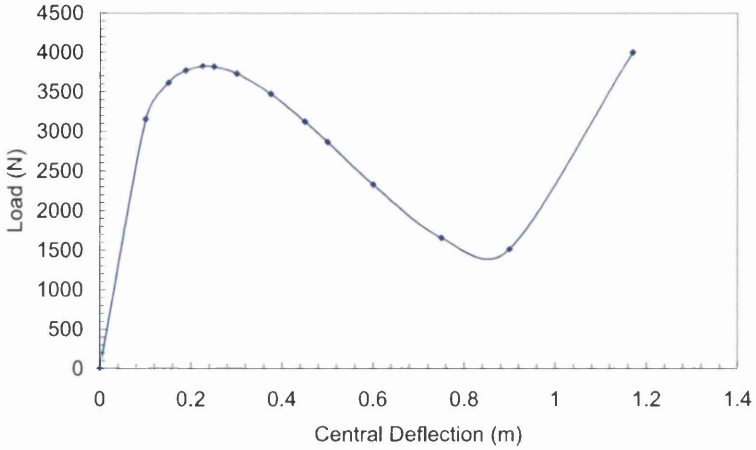


Figure 3.31: Snap-through curve obtained with the I/E algorithm

Figure 3.32: Contours of horizontal displacement $u_x\,(m)$ by I/E when external point load is set to $4000\,N$

ε^p	$\sigma(\frac{N}{cm^2})$
0.0	$24,500.0$
0.001167	$24,500.0$

Table 3.3: Material parameters for the plastic evolution.

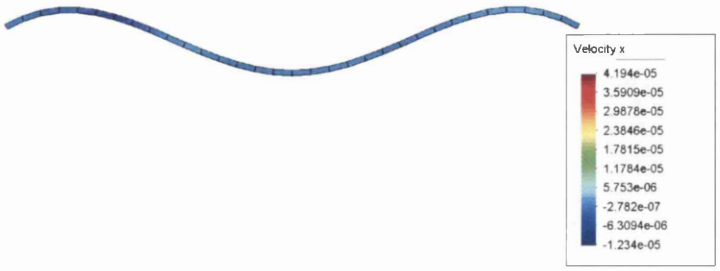


Figure 3.33: Contours of horizontal velocities $v_x\,(m/s)$ by I/E when external point load is set to $4000\,N$. It is apparent that their nodal values are practically zero from a physical point of view which correspond to the quasi-static solution

Figure 3.34: Contours of vertical velocities $v_y\,(m/s)$ by I/E when external point load is set to $4000\,N$

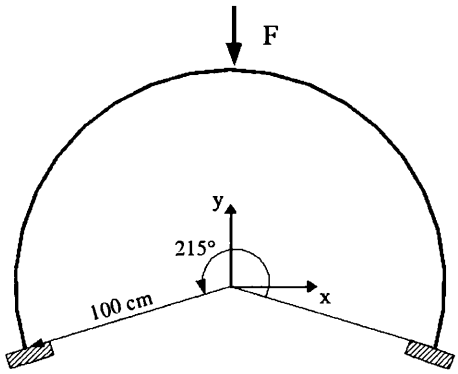


Figure 3.35: Circular arch clamped at the edge-supports

When the arch is loaded in the central-top node with $6.6\,N$ the IMP sub-algorithm diverges in the fourth iteration. At that point EXP sub-algorithm is automatically initiated (see Table (3.4)). The final vertical displacement field, over the undeformed configuration, is represented in Figure (3.37).

Flow	i/t_n	Error(%)	Max. Resid.	$\mathbf{f}_y^{int}(k_{cn})$	$\mathbf{u}_y(k_{cn})$	Stage
IMP	1	47.48	237.70	-	-	initial
IMP	2	25.72	163.08	-7430.40	-0.605	-
IMP	3	1.479	7336.16	-29695.92	-1.078	diverging
IMP	4	862.37	$0.44 \cdot 10^{+7}$	-62912.7	-1.119	swap
EXP	0.3943256	972567	5493540	-5542582	-1.625	oscillat.
EXP	3.33	0.802	2633.40	-63371.6	-1.522	-
EXP	6.28	0.45	1885.83	-64114.2	-1.703	-
EXP	99.99	0.000188	0.0096	-65998.2	-1.043	converging
EXP	150	0.000008	0.0406	-65999.97	-1.044	converging
EXP	170.56	$1.0 \cdot 10^{-6}$	0.008	-65999.99	-1.044	solution

Table 3.4: Relative residual norm(%) (error(%) in the table) and maximum residual. Vertical internal force (N) and displacement (cm) at central node of the arch(k_{cn}) . In the second column, the iteration number(i) is shown for the IMP sub-algorithm and the time (t_n) for the EXP sub-algorithm

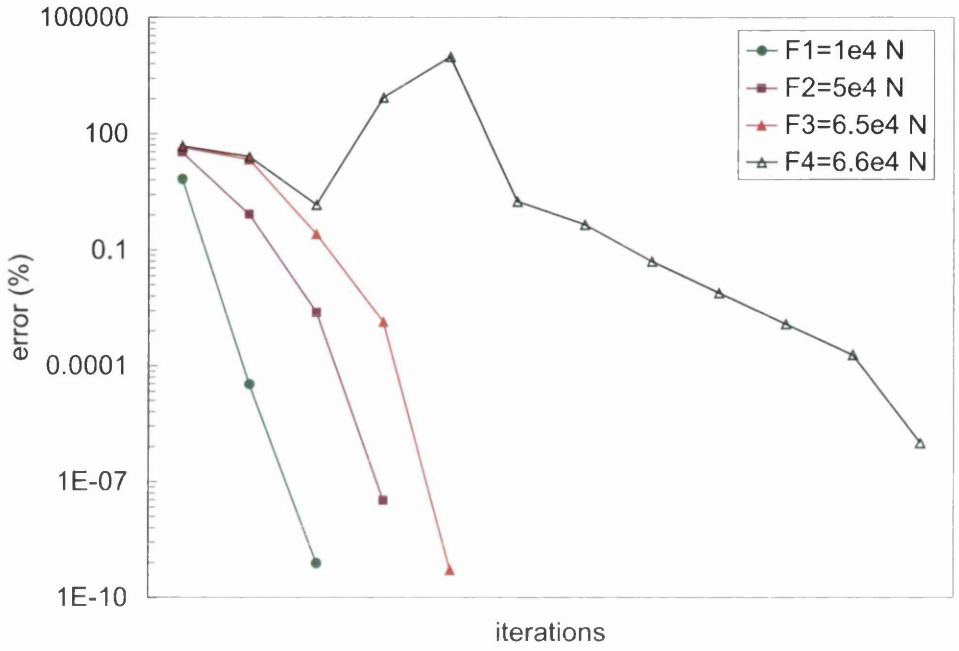


Figure 3.36: Relative residual norm error (%) on a logarithmic scale (circular elasto-plastic arch)

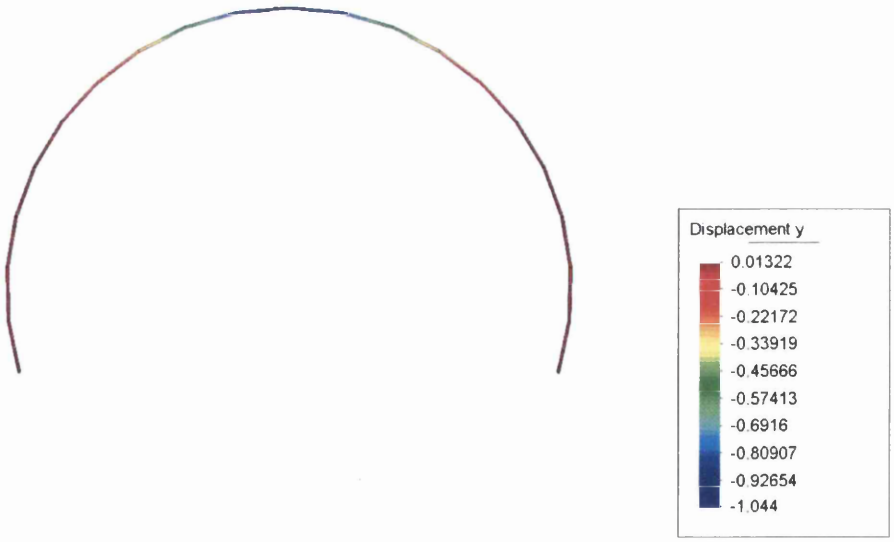


Figure 3.37: Contours of vertical displacement in the elastic-plastic circular arch deformed. Central node loaded with point force $6.6 \cdot 10^4\, N$

3.10 Concluding remarks

In this chapter an algorithm for nonlinear finite element problems has been presented. It included:

- An explicit scheme by CDM. A proper damping was studied. The critical time step is analysed in order to enforce stability to the numerical method. An adaptive time step demonstrated as being computationally more efficient than the usual formulas used for the calculation of constant time steps.

- An in-time implicit/explicit algorithm to solve finite element problems has been described. This chapter has been devoted to illustrating the algorithm and the steps followed in its programming. In particular, for analysis with finite strains where geometrically nonlinear buckling results in divergence of NRM. Practical application of this algorithm might also include the solution of non-smooth contact problems in which ,for instance, sharp-pointed edges and curved concave surface are involved.

Chapter 4

Development of a Structural Continuum-based Beam Element (CBE)

'...the beam could support twice as much load at the center as at the free end because the same moment was produced at the fixed end.'. **Galileo**; *Dialogue on Two New Sciences*; 1638.

4.1 Introduction and reasons for a CBE

A structural beam element based in a continuum rectangular element (CBE) is developed next. CBE formulation has been successfully programmed by the FORTRAN language. An original version of this element is partially indicated in Belytschko [11]. Some modifications to this formulation has been carried out, which will be shown at the point they appear in the analytical development below. The usefulness of this sort of element arises from the need to solve

beam structures that are sufficiently thin (diameter<<<length), for two main reasons:

- Continuum elements models imply smaller time steps than a structural one and, hence, a higher computational cost. Moreover, if thin structures are modelled, the time step must already be quite small to obtain conditional stability and, therefore, convergence (*Lax's Theorem*). This aspect is of utmost importance as the computational cost can sensibly be reduced.

- As it will be seen, CBE elements need half of the nodes (with respect to continuum elements) to compute the solution.

The continuum-based approach is founded in a rectangular continuum element on which structural conditions are imposed and the solver only will take master nodes into account for the solution of the system. Its formulation, coding as well as its preprocessing and postprocessing for plotting results in GiD have been completely programmed by the author. The solution of the discretised system is carried out in space by finite CBE and in time by finite differences using central velocities.

4.2 Application of external loading

Momentum equation for explicit solution at master nodes

$$\mathbf{M}\,\ddot{\mathbf{u}}(t_n) + \mathbf{C}\,\dot{\mathbf{u}}(t_n) + \mathbf{f}^{int}(\mathbf{u}_n) = \mathbf{f}^{ext} \tag{4.1}$$

where approximation is made by central velocities, being

$$\dot{\mathbf{u}}^T = \left\{ \begin{array}{ccccccccccc} \dot{u}_{x1} & \dot{u}_{y1} & \dot{\theta}_1 & \ldots & \dot{u}_{x\kappa} & \dot{u}_{y\kappa} & \dot{\theta}_\kappa & \ldots & \dot{u}_{x\,nmast} & \dot{u}_{y\,nmast} & \dot{\theta}_{nmast} \end{array} \right\}$$

with three degrees of freedom per master node $x\,y$ and the rotation of the cross-section around the master node. Subscript κ will be used in the remainder of this chapter for the master node and *nmast* will be the total number of master nodes in the mesh.

External forces are updated if they are applied on slave nodes because of the change of position in large rotations or/and deformations. Thus, for a constant loading, the module of the external force vector is the same at every iteration of the explicit method used in the solution. Therefore, it is needed to apply these slave nodal forces to the masters.

$$\begin{Bmatrix} f_{x\kappa} \\ f_{y\kappa} \\ m_{\kappa} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ y_{\kappa} - y_{\kappa^-} & x_{\kappa^-} - x_{\kappa} & y_{\kappa} - y_{\kappa^+} & x_{\kappa^+} - x_{\kappa} \end{bmatrix} \begin{Bmatrix} f_{x\kappa^-} \\ f_{y\kappa^-} \\ f_{x\kappa^+} \\ f_{x\kappa^+} \end{Bmatrix}$$

where κ^- and κ^+ are the slaves nodes at the bottom and at the top of the element and the master node κ. Writing in compact form for every master node κ where this is needed,

$$\mathbf{f}_{\kappa}^{(ext)} = \mathbf{T}_{\kappa}^{T} \cdot \mathbf{f}_{\kappa^-\,\kappa^+}^{(ext)} \tag{4.2}$$

where the superscript (ext) denotes external forces. Nevertheless, the tensor $\mathbf{T}$ refers to the application of any array from master nodes to slaves nodes.

4.3 Mass and damping matrices

A procedure, known as physical lumping has been used to obtain a diagonal mass matrix. The diagonalisation is suitable in order to formulate a system of uncoupled equations. Below is represented the mass matrix for a rectangular

element with four slaves nodes and two masters,

$$\mathbf{M} = \frac{\rho_o\, h_o\, l_o\, a_o}{420} \begin{bmatrix} 210 & 0 & 0 & 0 & 0 & 0 \\ 0 & 210 & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha\, h_o^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 210 & 0 & 0 \\ 0 & 0 & 0 & 0 & 210 & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha\, h_o^2 \end{bmatrix} \tag{4.3}$$

where $h_o\, l_o\, a_o$ are respectively the thickness, the length and the depth of the element at the initial configuration.

The damping matrix has been selected proportional to the mass matrix with proportionality constant $2\,\xi\,\omega_j$ with ξ being the damping ratio and ω_j the natural frequency of the degree of freedom of the master node considered,

$$\mathbf{C} = \begin{cases} C_{ij} = 2\,\xi\,\omega_j\, M_{ij} & i = j \\ C_{ij} = 0 & i \neq j \end{cases} \tag{4.4}$$

4.4 Kinematics

Displacements at the slaves nodes are calculated from the displacements in the masters. Rearrangement of the arrays are performed in order to formulate by slaves. Thus, the velocities at master κ are taken out of the global vector $\dot{u}$,

$$\dot{\mathbf{u}}_\kappa^{mast} = \begin{Bmatrix} \dot{u}_{x\kappa} \\ \dot{u}_{y\kappa} \\ \dot{\theta}_\kappa \end{Bmatrix} \tag{4.5}$$

The director vector of the material section at master κ is $\mathbf{p}_\kappa(t_n)$. This section rotates with the material and, therefore, θ_κ is the rotation angle which

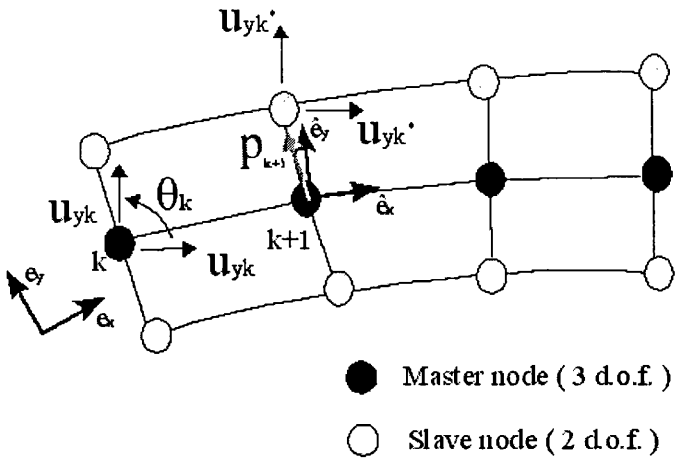


Figure 4.1: Continuum-based beam element (CBE) with four master nodes and eight slaves

defines the director vector,

$$\mathbf{p}_\kappa(t_n) = cos(\theta_\kappa(t_n))\,\mathbf{e}_x + sin(\theta_\kappa(t_n))\,\mathbf{e}_y \tag{4.6}$$

The coordinates of the slaves nodes are readily calculated from the position of the master $\mathbf{x}_\kappa(t_n)$ and the director vector,

$$\mathbf{x}_{\kappa^+}(t_n) = \mathbf{x}_\kappa(t_n) + \frac{1}{2}\,h^0_\kappa\,\mathbf{p}_\kappa(t_n)$$

$$\mathbf{x}_{\kappa^-}(t_n) = \mathbf{x}_\kappa(t_n) - \frac{1}{2}\,h^0_\kappa\,\mathbf{p}_\kappa(t_n)$$

where h^0_κ is the thickness of the CB-beam element at the master κ in the initial configuration. In the remainder of this chapter, the index for time and number of iterations of the explicit method t_n is dropped unless it is considered useful to highlight some point. The velocities in the slave nodes are,

$$\mathbf{v}_{\kappa^+} = \mathbf{v}_\kappa + \frac{1}{2}\,h_{\kappa^0}\,\boldsymbol{\Omega}_\kappa \times \mathbf{p}_\kappa$$

$$\mathbf{v}_{\kappa^-} = \mathbf{v}_\kappa - \frac{1}{2} h_{\kappa^0} \boldsymbol{\Omega}_\kappa \times \mathbf{p}_\kappa$$

where $\boldsymbol{\Omega}_\kappa$ is the angular velocity at master κ and is defined as,

$$\boldsymbol{\Omega}_\kappa = \frac{D\theta_\kappa}{Dt} \mathbf{e}_z \tag{4.7}$$

designating $\frac{D}{Dt}$ the material time derivative.

4.4.1 Laminar configuration

A corotational basis $\{\hat{\mathbf{e}}_x, \hat{\mathbf{e}}_y\}$ may be defined at each point of the element which rotates in such a manner that the unit vector $\mathbf{e}_x$ is always tangent to the fiber at that point. The element is defined by fibers and laminas. Fibers are imagined along the longitudinal section of the element. One of these fibers is the mid-line of the element. Observe that the direction of vector $\mathbf{e}_y$ is not the same as the vector director and ,therefore, the angle of the rotation of the corotational (or laminar) base is not θ either. Thus, the base is defined as,

$$\hat{\mathbf{e}}_x = \frac{x_{,\xi}\ \mathbf{e}_x + y_{,\xi}\ \mathbf{e}_y}{\sqrt{x_{,\xi}^2 + y_{,\xi}^2}} \tag{4.8}$$

$$\hat{\mathbf{e}}_y = \frac{-y_{,\xi}\ \mathbf{e}_x + x_{,\xi}\ \mathbf{e}_y}{\sqrt{x_{,\xi}^2 + y_{,\xi}^2}} \tag{4.9}$$

where $\mathbf{x}^T = (\ x \quad y\ )$is the current position of a point in the element and the comma indicates the derivatives with respect to the parent element coordinate ξ. The interpolation is carried out by shape functions, thus the position of a point in the element may be interpolated as a function of the position of the slaves nodes (placed in the corners of the continuum element and designated either by κ^* for all of them in the element or by κ^- for the slave placed just at the bottom of the master node κ and κ^+ for the one situated at the top of the

master [1]; see Figure 4.4) and the shape function evaluated in the corresponding parent element coordinates $(\xi\,,\,\eta)$,

$$\mathbf{x} = \sum_{\kappa^*=1}^{nslave} \mathbf{x}_{\kappa^*}\, N_{\kappa^*}(\xi\,,\,\eta) \tag{4.10}$$

The shape functions selected for an element with four slaves nodes are,

$$N_{\kappa^*}(\xi\,,\,\eta) = \frac{1}{4}(1+\xi_{\kappa^*}\,\xi)(1+\eta_{\kappa^*}\,\eta) \tag{4.11}$$

with $(\xi_{\kappa^*}\,,\,\eta_{\kappa^*})$ being the corresponding coordinates of the slaves nodes in the parent element. Thus, for example, for the node 3^* are, obviously, $(\xi_{3^*}\,,\,\eta_{3^*}) = (1,1)$. These shape functions fulfil the property that they are equal to the unity in its definition node and zero in the remainder. Then, the derivatives with respect to the parent element coordinates are,

$$\left\{ \begin{array}{c} x_{,\xi} \\ y_{,\xi} \end{array} \right\} = \sum_{\kappa^*=1}^{nslave} \mathbf{x}_{\kappa^*}\, \frac{\partial N_{\kappa^*}}{\partial \xi} \tag{4.12}$$

This array corresponds to the first column of the jacobian of the transformation between the parent and current configuration, given by,

$$\mathbf{x}_{,\xi} = \left[\begin{array}{cc} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{array} \right] \tag{4.13}$$

In the remainder of the chapter the partial derivatives are designated with commas instead of fractional partial derivative symbol. Thus, for example, $\frac{\partial N_{\kappa^*}}{\partial \xi}$ will be stated as $N_{\kappa^*,\xi}$.

The rotation matrix between the reference $\{\mathbf{e}_x, \mathbf{e}_y\}$ and the corotational or laminar base $\{\hat{\mathbf{e}}_x, \hat{\mathbf{e}}_y\}$ is stated as,

$$\mathbf{R} = \left[\begin{array}{cc} \mathbf{e}_x \cdot \hat{\mathbf{e}}_x & \mathbf{e}_x \cdot \hat{\mathbf{e}}_y \\ \mathbf{e}_y \cdot \hat{\mathbf{e}}_x & \mathbf{e}_y \cdot \hat{\mathbf{e}}_y \end{array} \right] \tag{4.14}$$

[1] The line formed by the slaves and the corresponding master is not, in general, perpendicular to the midline of the element

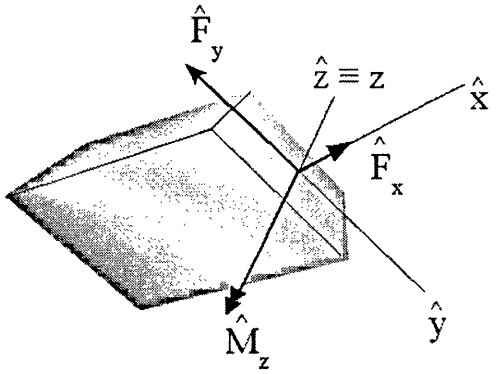


Figure 4.2: Detail of forces and momentum on the laminar configuration

The velocity of slave node κ^* in laminar coordinates $\hat{\mathbf{v}}_{\kappa^*}^T = \begin{pmatrix} \hat{v}_{x\kappa^*} & \hat{v}_{y\kappa^*} \end{pmatrix}$,

$$\hat{\mathbf{v}}_{\kappa^*} = \mathbf{R}^T \cdot \mathbf{v}_{\kappa^*} \tag{4.15}$$

But in a more general manner $\hat{\mathbf{v}}_{\kappa^*}$ is used as an array representing velocities in all slave nodes of the element,

$$\hat{\mathbf{v}}_{\kappa^*} = \begin{bmatrix} \hat{v}_{x1^*} & \hat{v}_{x2^*} & \hat{v}_{x3^*} & \dots & \hat{v}_{x\,nslave^*} \\ \hat{v}_{y1^*} & \hat{v}_{y2^*} & \hat{v}_{y3^*} & \dots & \hat{v}_{y\,nslave^*} \end{bmatrix} \tag{4.16}$$

In the programme an element of four slave nodes is taken ($nslave = 4$).

4.4.2 Velocity gradient and rate of deformation

We need the jacobian matrix in laminar coordinates to compute the derivatives of shape functions in laminar components $\mathbf{N}_{\kappa^*,\hat{x}}$ and later on the velocity gradient $\hat{\mathbf{L}}$,

$$\hat{\mathbf{x}}_{,\xi} = \mathbf{R}^T \cdot \mathbf{x}_{,\xi} \cdot \mathbf{R} \tag{4.17}$$

The shape function derivatives array in a corotational base are,

$$\mathbf{N}^T_{\kappa^*,\hat{x}} = \mathbf{N}^T_{\kappa^*,\xi} \cdot \hat{\mathbf{x}}^{-1}_{,\xi} \tag{4.18}$$

The velocity gradient is given by,

$$\hat{\mathbf{L}} = \hat{\mathbf{v}}_{\kappa^*} \cdot \mathbf{N}^T_{\kappa^*,\hat{x}} \tag{4.19}$$

which may be written in matrix form for four slave nodes per element as follows,

$$\begin{bmatrix} \hat{L}_{xx} & \hat{L}_{xy} \\ \hat{L}_{yx} & \hat{L}_{yy} \end{bmatrix} = \begin{bmatrix} \hat{v}_{x1^*} & \hat{v}_{x2^*} & \hat{v}_{x3^*} & \hat{v}_{x4^*} \\ \hat{v}_{y1^*} & \hat{v}_{y2^*} & \hat{v}_{y3^*} & \hat{v}_{y4^*} \end{bmatrix} \cdot \begin{bmatrix} N_{1^*,\hat{x}} & N_{1^*,\hat{y}} \\ N_{2^*,\hat{x}} & N_{2^*,\hat{y}} \\ N_{3^*,\hat{x}} & N_{3^*,\hat{y}} \\ N_{4^*,\hat{x}} & N_{4^*,\hat{y}} \end{bmatrix} \tag{4.20}$$

Finally, the rate of deformation $\hat{\mathbf{D}}$ or velocity strains in laminar components are obtained from,

$$\hat{\mathbf{D}} = \frac{1}{2}(\hat{\mathbf{L}}^T + \hat{\mathbf{L}}) \tag{4.21}$$

4.5 Computation of stress at quadrature points

The rate of stress is computed through the constitutive law of the material considered and the velocity strains.

$$\frac{D\hat{\sigma}}{Dt} = \Xi(\hat{\mathbf{D}}) \tag{4.22}$$

The material time derivative of stress $\frac{D\hat{\sigma}}{Dt}$ is calculated as follows: firstly, the time derivatives are computed as any other derivative above i.e. by central differences in time and, secondly, the gradient of stress tensor is approximated by central differences in space in a vicinity small enough around the point

BOX 4.1: UPDATE OF MASTER NODAL INTERNAL FORCES

1. For each master node of the current element, do:

   (a) Extract displacements and velocities in the master of the current element from global arrays

   (b) Update position of master nodes:

$$\mathbf{x}_\kappa(t_{n+1}) = \mathbf{x}_\kappa(t_n) + \mathbf{u}_\kappa(t_n)$$

   (c) Director vectors at masters,

$$\mathbf{p}_\kappa(t_n) = cos(\theta_\kappa(t_n))\,\mathbf{e}_x + sin(\theta_\kappa(t_n))\,\mathbf{e}_y$$

   (d) Coordinates and velocities of slaves nodes through the director vector and the updated position of master nodes:

$$\mathbf{x}_{\kappa^\pm}(t_n) = \mathbf{x}_\kappa(t_n) \pm \frac{1}{2} h^0_\kappa\, \mathbf{p}_\kappa(t_n)$$

$$\mathbf{v}_{\kappa^\pm} = \mathbf{v}_\kappa \pm \frac{1}{2} h_{\kappa^0}\, \boldsymbol{\Omega}_\kappa \times \mathbf{p}_\kappa$$

2. Reset slave nodal internal force array

3. Loop over quadrature points: Box 4.2

4. Compute master nodal forces and return

$$\mathbf{f}^{(ext)}_\kappa = \mathbf{T}^T_\kappa \cdot \mathbf{f}^{(ext)}_{\kappa^-\,\kappa^+}$$

of quadrature at iteration-time t_n(see Eq 4.23). In the examples tested, the gradient of stresses resulted too small to be compared with the variations of stresses in time. However, the computation of these gradients is always carried out in the simulations because they can reach high values in nonlinear problems for two main reasons:

- The nonlinearity source itself causes high fluctuations in the stresses field in some parts of the domain.
- The smaller time step required for this sort of analysis provokes the magnitude of variation of stresses in time might be of the same order as the variation in space.

$$\frac{D\hat{\boldsymbol{\sigma}}}{Dt} = \frac{\partial \hat{\boldsymbol{\sigma}}}{\partial t} + \hat{\mathbf{v}} \cdot \nabla \hat{\boldsymbol{\sigma}} \tag{4.23}$$

where the approximations are carried out as,

$$\frac{\partial \hat{\sigma}_{ij}}{\partial t} \approx \frac{\triangle \hat{\sigma}_{ij}}{\triangle t} = \frac{\hat{\sigma}_{ij}(t_{n+\frac{1}{2}}) - \hat{\sigma}_{ij}(t_{n-\frac{1}{2}})}{\triangle t_n}$$

$$\hat{v}_k \cdot \frac{\partial \hat{\sigma}_{ij}}{\partial \hat{x}_k} = \hat{v}_x \cdot \frac{\partial \hat{\sigma}_{ij}}{\partial \hat{x}} + \hat{v}_y \cdot \frac{\partial \hat{\sigma}_{ij}}{\partial \hat{y}} \approx \hat{v}_x \cdot \frac{\triangle \hat{\sigma}_{ij}}{\triangle \hat{x}} + \hat{v}_y \cdot \frac{\triangle \hat{\sigma}_{ij}}{\triangle \hat{y}}$$

the values of the increments $\triangle \hat{x}$ and $\triangle \hat{y}$ are defined in relation to the smallest length in the element l, such as $\triangle \hat{x}/l \leq \zeta$ where ζ depends on the nonlinearity of the analysis. In the simplest case, an elastic material, the constitutive law is immediate.

4.6 Numerical tests of CBE

All representations below are plotted with the slaves nodes to highlight the properties and distribution fields, although only the master nodes are used in

BOX 4.2a: STRESS COMPUTATION AT QUADRATURE POINT

1. Compute jacobian at current quadrature point:

$$\mathbf{x}_{,\xi} = \sum_{\kappa^*=1}^{nslave} \mathbf{x}_{\kappa^*} \frac{\partial N_{\kappa^*}}{\partial \xi}$$

2. Set laminar basis at point $\mathbf{x}(\xi, \eta)$,

$$\hat{\mathbf{e}}_x = \frac{x_{,\xi}\ \mathbf{e}_x + y_{,\xi}\ \mathbf{e}_y}{\sqrt{x_{,\xi}^2 + y_{,\xi}^2}} \ ; \ \hat{\mathbf{e}}_y = \frac{-y_{,\xi}\ \mathbf{e}_x + x_{,\xi}\ \mathbf{e}_y}{\sqrt{x_{,\xi}^2 + y_{,\xi}^2}}$$

3. Rotation matrix $R(\mathbf{x}(\xi, \eta))$,

$$\mathbf{R} = \begin{bmatrix} \mathbf{e}_x \cdot \hat{\mathbf{e}}_x & \mathbf{e}_x \cdot \hat{\mathbf{e}}_y \\ \mathbf{e}_y \cdot \hat{\mathbf{e}}_x & \mathbf{e}_y \cdot \hat{\mathbf{e}}_y \end{bmatrix}$$

4. New positions and velocities of the slave nodes κ^* in laminar basis:

$$\hat{\mathbf{v}}_{\kappa^*} = \mathbf{R}^T \cdot \mathbf{v}_{\kappa^*}$$

$$\hat{\mathbf{x}}_{\kappa^*} = \mathbf{R}^T \cdot \mathbf{x}_{\kappa^*}$$

5. Jacobian in laminar components,

$$\hat{\mathbf{x}}_{,\xi} = \mathbf{R}^T \cdot \mathbf{x}_{,\xi} \cdot \mathbf{R}$$

6. Calculate jacobian inverse $\hat{\mathbf{x}}_{,\xi}^{-1}$

BOX 4.2b: STRESS COMPUTATION AT QUADRATURE POINT (continued)

7. The shape function gradient is obtained as follows,

$$\mathbf{N}^T_{\kappa^*,\hat{x}} = \mathbf{N}^T_{\kappa^*,\xi} \cdot \hat{\mathbf{x}}^{-1}_{,\xi}$$

8. Velocity gradient,

$$\hat{\mathbf{L}} = \hat{\mathbf{v}}_{\kappa^*} \cdot \mathbf{N}^T_{\kappa^*,\hat{x}}$$

9. Rate of deformation,

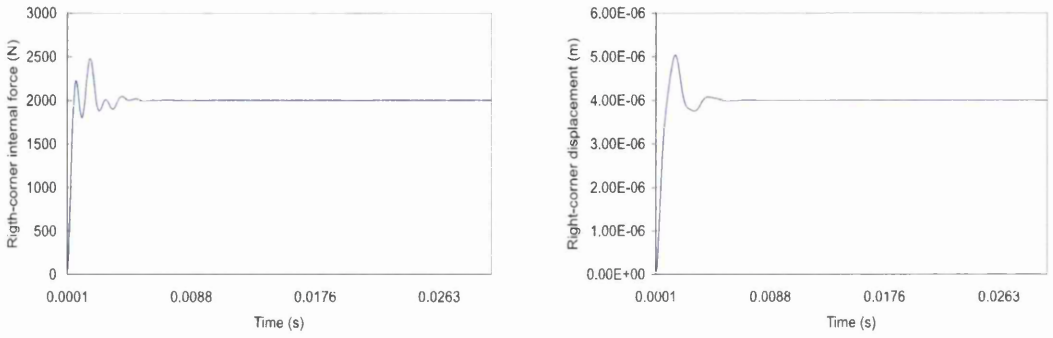
$$\hat{\mathbf{D}} = \frac{1}{2}(\hat{\mathbf{L}}^T + \hat{\mathbf{L}})$$

10. Update of rate of stress and integration Section(4.5)

11. Add force-contribution to the nodal slave forces vector

12. Go to next quadrature point or end if quadrature points for this element completed.

the explicit computation of displacements and, hence, used as classical beam elements. The difference remains in the computation of nodal internal forces which is made in the slaves nodes (through numerical integration by quadrature points) and transferred to the masters correspondingly.



(a) (b)

Figure 4.3: Tension test: internal force (a) and displacement (b) of the master node in the loaded edge

4.6.1 Tension test

The tension and compression tests are the most simple tests to assess the CBE element formulation. They are an almost obligatory tests to perform for beam elements. A bar, formed by 2 CBE elements, is loaded by a traction force in this test. The external force (2000 N) is applied distributed in the two slaves nodes on the right side and constrained at left side. The geometry dimensions and material parameters are shown in Table (4.1). This simulation and compression test were also carried out by [94]. The results agree exactly with the analytical solution, i.e. an elongation of $4 \cdot 10^{-6} m$ and axial stress of $20000 Pa$ (see Figure (4.4)).

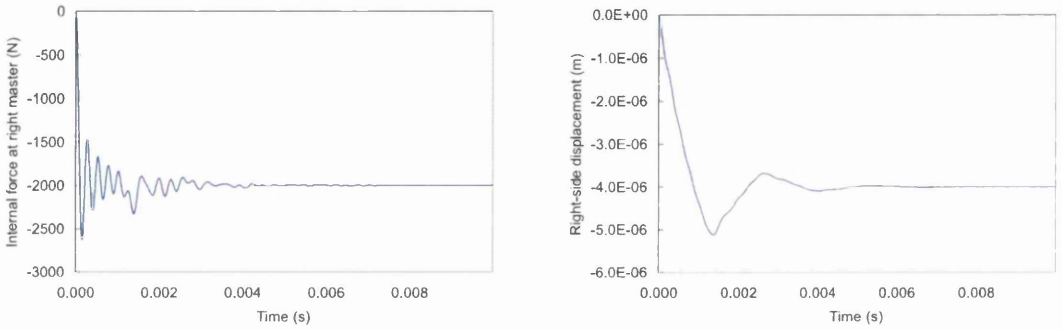
Geometry and material properties	
Young Modulus	$1000\,MPa$
Poisson Ratio	0.3
Mass density	$2700\,Kg/m^3$
Length	$2\,m$
Depth	$1\,m$
Thickness	$0.1\,m$

Table 4.1: Material properties for the compression and traction tests

Figure 4.4: Horizontal displacements $u_x[m]$ due to traction

4.6.2 Compression test

A bar, formed by 8 CBE elements, is compressed in this test. The external force (-2000 N) is applied distributed in the two slaves nodes on the right side and constrained at left side. The geometry dimensions and material parameters are shown in Table (4.1). The results agree exactly with the analytical solution, i.e. an elongation of $-4 \cdot 10^{-6}\, m$ and axial stress of $-20000 Pa$ (see Figures 4.6 and 4.7).



(a) (b)

Figure 4.5: Compression test: $1.0e{+}05$ iterations for the internal force (a) and displacement (b) of the master node in the loaded edge

4.6.3 Cantilever beam

In this test, a cantilever beam is loaded at its edge with a vertical load P. The beam is formed by 8 CBE elements and geometry and material characteristics are displayed at Table (4.2). The maximum deflection, i.e. at the free edge, agrees well with the analytical solution for the loads specified in the following subsections. Selective-reduced integration was carried out in the first test, $P = 20,000\, N$, and nine quadrature point were used in the second one,

Figure 4.6: Axial stresses $\sigma_{xx}[N/m^2]$ due to compression

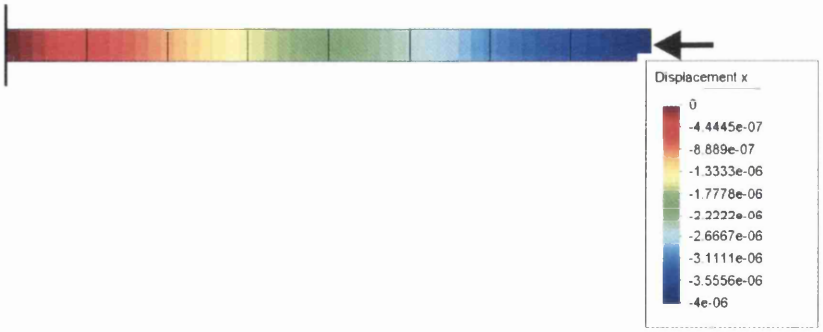


Figure 4.7: Horizontal displacements $u_x[m]$ due to compression

$P = 200,000\,N$.

Geometry and material properties	
Young Modulus	$71000\,MPa$
Poisson Ratio	0.34
Mass density	$2700\,Kg/m^3$
Length	$1\,m$
Depth	$0.1\,m$
Thickness	$0.1\,m$

Table 4.2: Material properties for cantilever beam

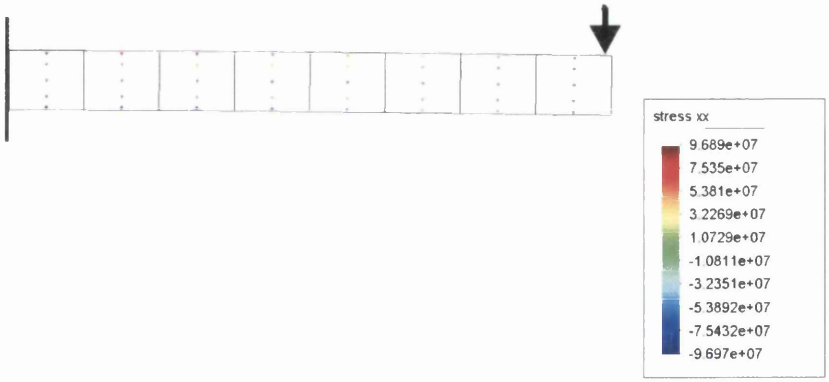


Figure 4.8: Cantilever beam ($P = 2e4N$): axial stress $\sigma_{xx}[N/m^2]$ in through-thickness quadrature points (selective-reduced integration)

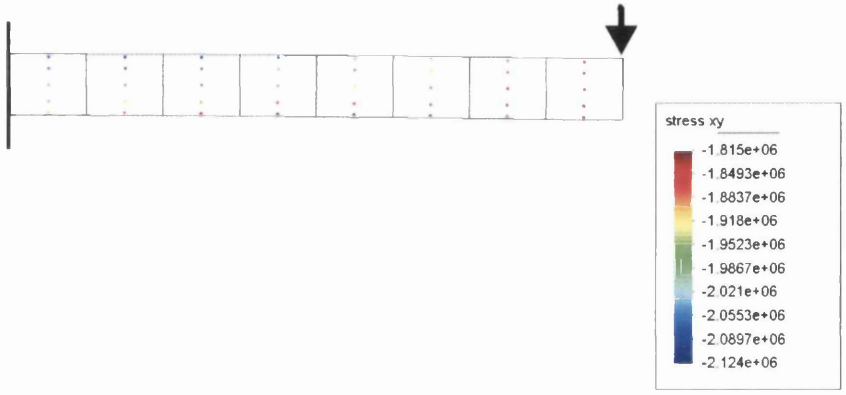
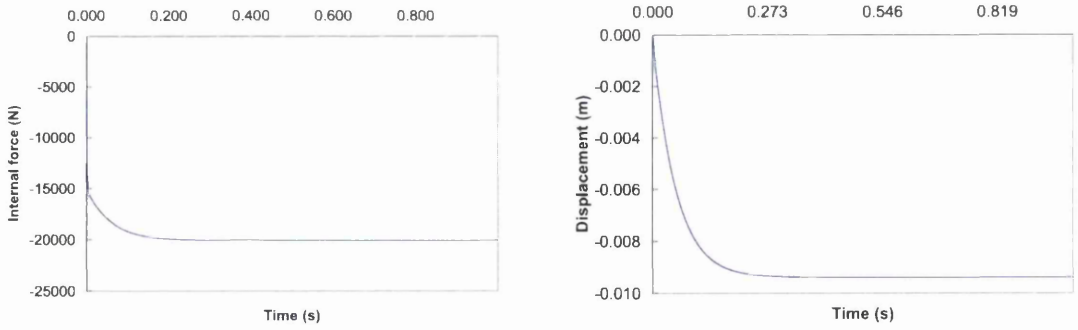


Figure 4.9: Cantilever beam ($P = 2e4N$): shear stress $\sigma_{xy}[N/m^2]$ in through-thickness quadrature points (selective-reduced integration)



(a) (b)

Figure 4.10: Cantilever beam ($P = 2e4N$): attenuation of explicit iterations for the internal force (a) and displacement (b) of the master node in the loaded edge

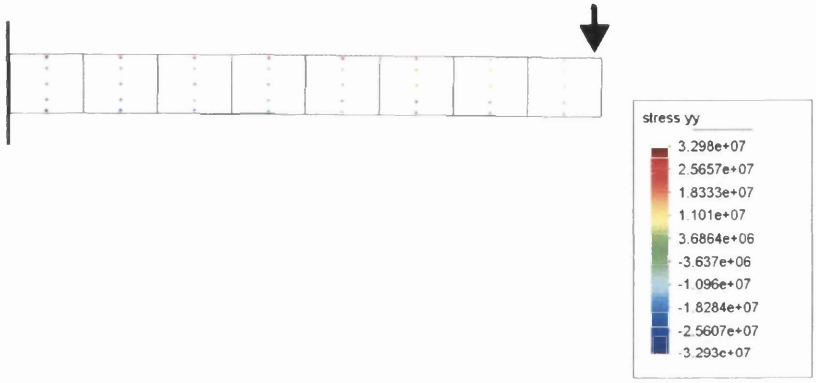


Figure 4.11: Cantilever beam ($P = 2e4N$): transverse stress $\sigma_{yy}[N/m^2]$ in through-thickness quadrature points (selective-reduced integration)

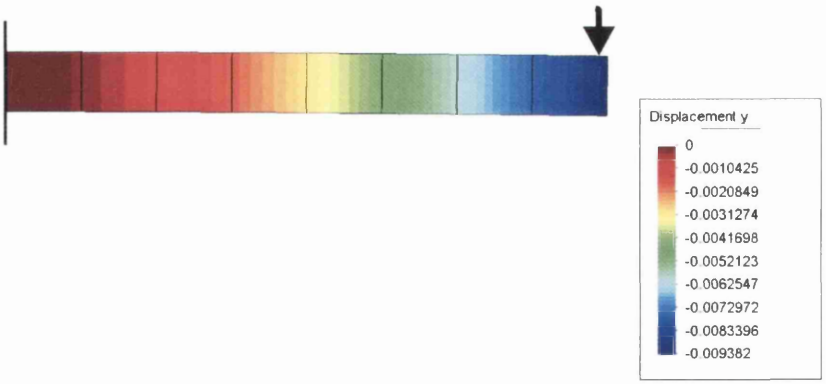


Figure 4.12: Cantilever beam ($P = 2e4N$): vertical displacement $u_y[m]$

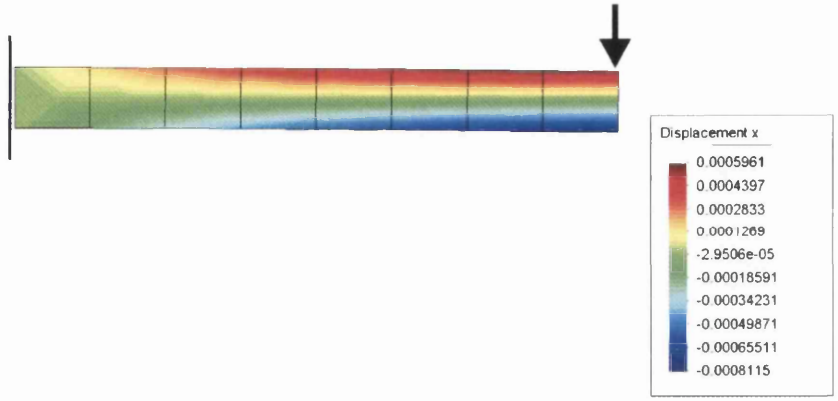


Figure 4.13: Cantilever beam ($P = 2e4N$): horizontal displacement $u_x[m]$

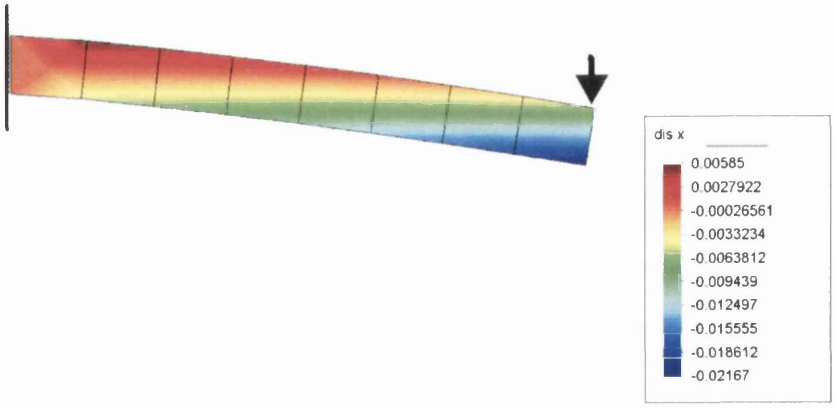


Figure 4.14: Cantilever beam ($P = 2e5N$): horizontal displacement $u_y[m]$

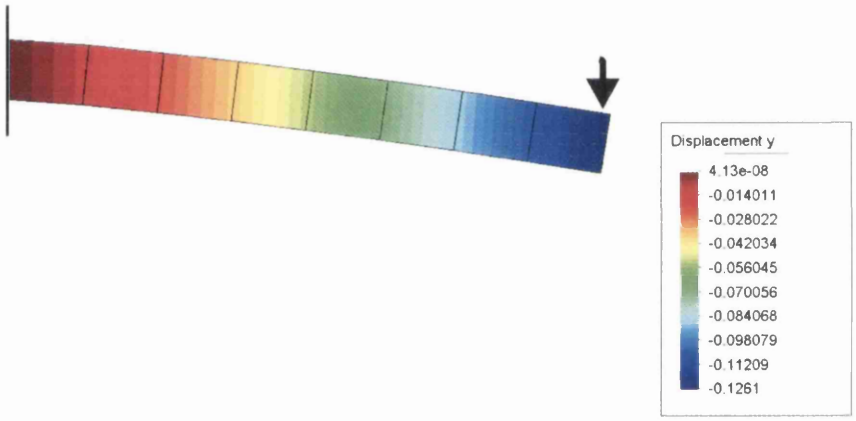


Figure 4.15: Cantilever beam ($P = 2e5N$): vertical displacement $u_y[m]$

4.6.4 Supported beam

The geometry and material parameters are those from the Table (4.2). The beam is loaded as represented in Figure (4.16).

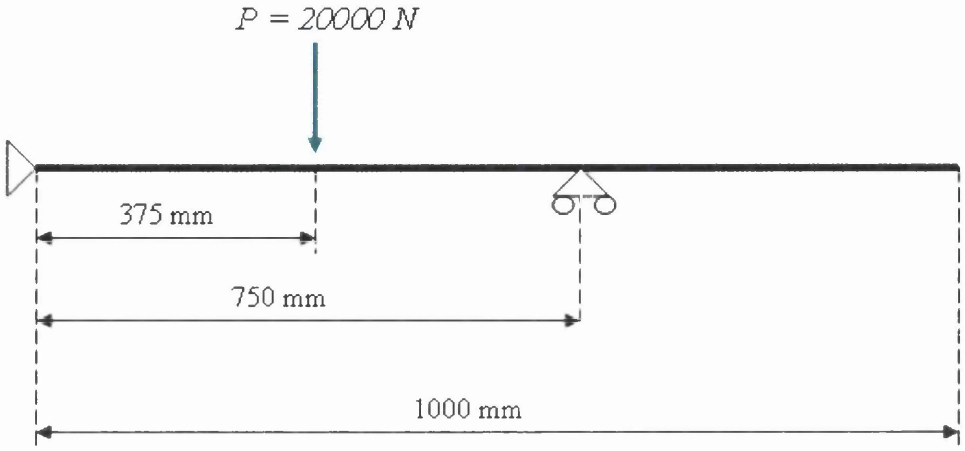


Figure 4.16: Supported beam geometry

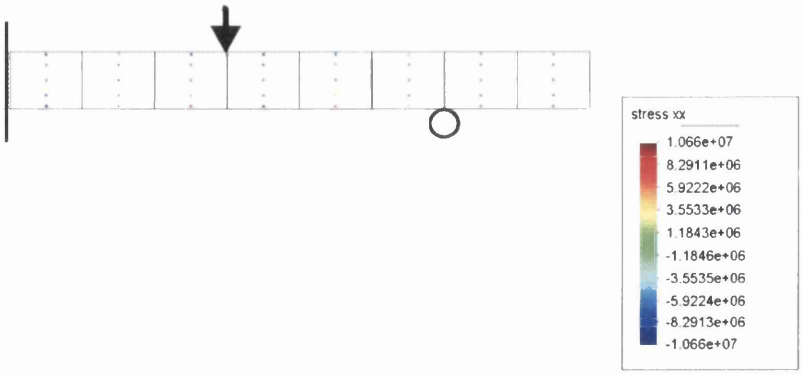


Figure 4.17: Beam : axial stress $\sigma_{xx}[N/m^2]$ in through-thickness quadrature points (selective-reduced integration)

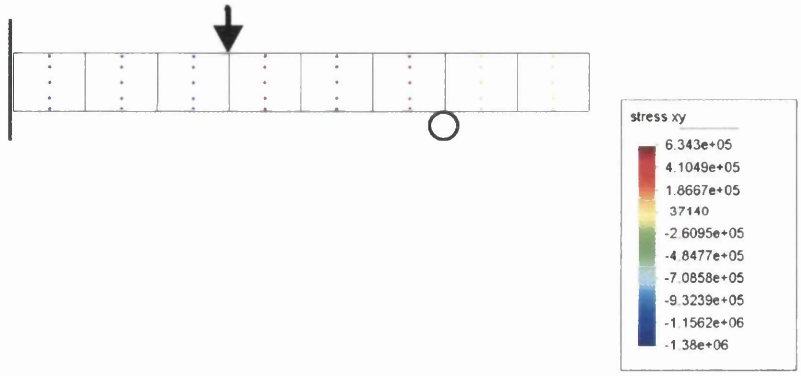


Figure 4.18: Beam : shear stress $\sigma_{xy}[N/m^2]$ in through-thickness quadrature points (selective-reduced integration)

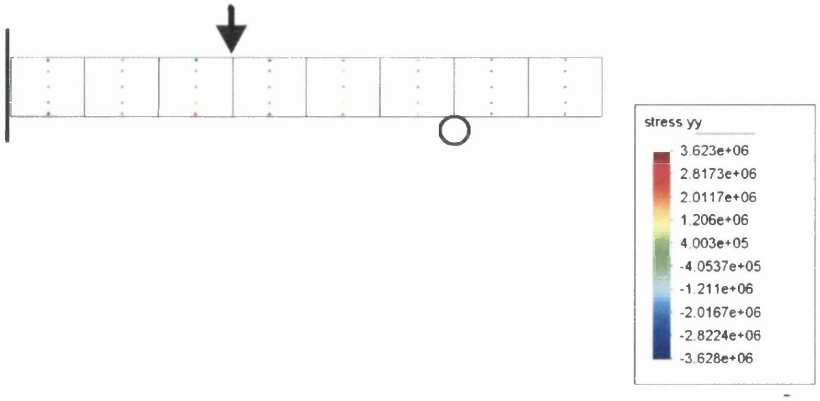


Figure 4.19: Beam : transverse stress $\sigma_{yy}[N/m^2]$ in through-thickness quadrature points (selective-reduced integration)

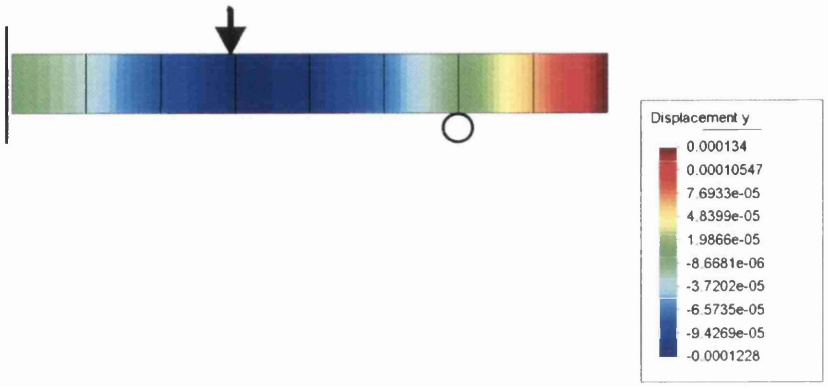


Figure 4.20: Beam : vertical displacement $u_y[m]$

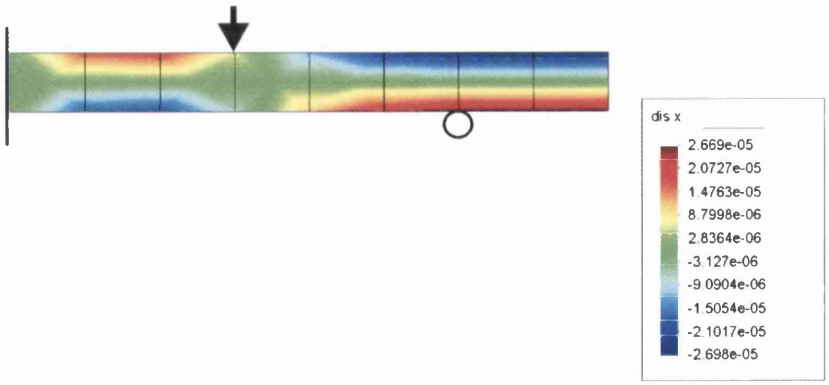


Figure 4.21: Beam : horizontal displacement $u_x[m]$

Chapter 5

Modeling of Reinforced Materials using a Subcycling Algorithm

'The method of Science depends upon our attempts to describe the world with simple theories: theories that are complex may become unstable, even if they happen to be true. Science must be described as the act of systematic simplifications'. **K. Popper**; *The Open Universe.*

5.1 Introduction

An algorithm formed by two subcycles [1]: one for the matrix and another one for the reinforcement is developed in this chapter. Reinforced materials are difficult to model computationally because of the *jump* introduced by the in-

[1]The word subcycle is used here indicating an execution of a group of elements with a solver - *implicit or explicit* - and, also, a different processing time

terface. The use of subcycles is carried out towards solving the constituents of the material, reinforcement and matrix separately, and enforcing conservation laws in the interface. The highest frequency determines the critical time step in explicit methods such as Central Differences. Then, the entire mesh is ruled by this critical time step. Those parts of the mesh with smaller frequencies, for example with larger elements, might be solved with a larger time increment without altering the conditional stability. Therefore, the different frequencies associated with the mesh of matrix and reinforcement motivate the present approach. Thus, a subcycling algorithm is developed. The subcycling algorithm is applicable for the in-time implicit-explicit algorithm for the matrix (see Chapter 3,[30, 31]) and the explicit *Central Differences Method* for the reinforcement (see Chapter 4). An abbreviated form I/E-E to label the subcycling algorithm is adopted. In Figure (5.4.3) a schematic representation of the evolution of the different parts of the mesh constituents is shown. In Figure (5.4.2) is represented the evolution in case of non-divergent matrix execution. The reason to use the implicit-explicit algorithm (for the matrix) is to take advantage of the quadratic rates of convergence of the implicit method proposed (Chapter 3) and in case of divergence solve the matrix with an explicit method. Furthermore, the implicit method (see Chapter 3) is unconditionally stable. Stability of the subcycling algorithm requires an accurate application of the conservation laws at the interface [10, 34, 42].

The coupling between explicitly and implicitly integrated partitions plays a key role. There is a classification of implicit-explicit or explicit-explicit subcycling algorithms related to the type of coupling: *weak coupling or strong coupling* [8]. For instance, weak couplings perform a simple transmission of information at the interface. Their stability needs to be carefully studied de-

pending on the different approaches undertaken [8]. On the contrary, strong couplings approaches include interface elements. However, this last approach needs to assign a material space for the interface element. In our problem, the jump-interface is a surface where kinematics and kinetics conditions must be enforced without any relevant thickness and, therefore, the introduction of interface elements is not the most convenient. At this stage, the following scheme is adopted:

(i) The nodes of the matrix and reinforcement share initially the same spatial position.

(ii) Enforcing the conservation laws in the structural discontinuity or jump in the bimaterial interface.

(iii) It permits the solution by a subcycling algorithm as described in this

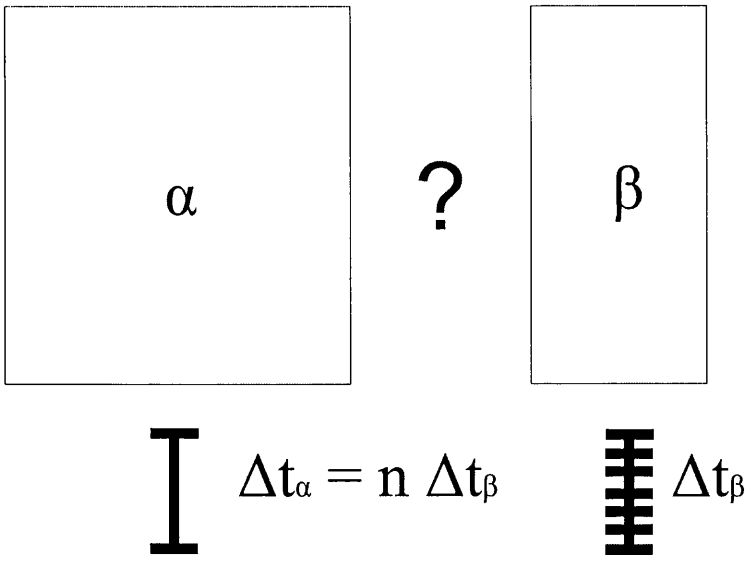


Figure 5.1: Schematic representation of subcycling. The time step in one of the subdomains is an integer number of time steps in the other.

work. This is convenient as the instability caused by the jump in the interface is avoided.

Belytschko [79] proposed a subcycling algorithm based in two explicit subcycles. The mesh was divided into group of nodes and each group required a different time step. This scheme enhances the computational efficiency when the elements are of different sizes. Roberts (1999) presents another explicit-explicit subcycling applied to the solution of the system of rockbolt-rock in tunnels. An uncoupled approach of reinforcement and matrix is carried out in [22] in a two-dimensional context. A major issue generated for this last approach is the proper modelling of the interface to ensure stability in the numerical processing.

5.2 Reinforced materials

The literature concerning the computational modeling of reinforced concrete is quite numerous, which reflects the enormous importance of this material. Research topics go from the softening material modeling for the concrete matrix and the elasto-plastic material model for the reinforcing bars to the models treating cracks, crack growth, static or dynamic loading, cycling loading, fatigue, etc.

One of the most referenced studies in the literature is the seminal work of the doctoral thesis by Rots [100] that performed a different approach by introducing interface elements. Rots used them firstly in the generation and development of cracks in a softening material, but the basis for using this type of elements in other applications such as interface-jump was developed.

The understanding of the mechanics of bond is of absolute importance in the

physical interface between the concrete and the reinforcing bars, as this effect affects the global behaviour of this RM. Various experimental works highlight this effect,

- Tasnimi et al. [118] presents a set of compression tests (axial compression tests over cylinders of concrete) used for the generation of a mathematical model for the curve of bond stress-strain.

- Weathersby [125] highlights that the major limitation of the FE modeling of response of RC is the accurate modelling of the interaction between the concrete and the steel. Experimental impact loading and static loading were performed to gain a better understanding of the influence of the loading rates -static, dynamic and impact loadings-. FE analysis was developed using the experimental parameters to determine the value of the chemical adhesion and to compare the experimental results with the analytical values.

In some studies a perfect bond model was assumed for the interface, e.g. May et al. [75], when this is limited to a narrow range of problems. The imperfect bond is frequently modeled by a shear bond stress versus relative displacement or slip relationship. It is generally assumed that there is no normal bond stress as the bar is confined in the continuum. Some of the works treating imperfect bond are summarized next:

- Wang et al. [124] proposed an analytical model to describe the bond strength considering the ribs in a partly cracked thick-walled cylinder exposed to radial external pressure. The bond stress-strain curve is composed of an initial linear part followed by softening after the peak in the bond stress-strain curve

- Sluys and de Borst [108] highlighted the key role of the bond-slip relation in predicting crack spacing in reinforced elements

- Brighenti [18] performs a rigorous mathematical interpretation of the interface in reinforced materials. in a simple and computationally effective way. A physically based nonlinear constitutive law is used. Transmission to the fibres is made via shear bond stress in the axial fiber direction. The amount of sliding is calculated through an energetic approach. The main effect of the fibres is more intense in the pre-debonding stage and the limitation of the plastic area of the fibres embedded into the matrix material which heavily depends on the interface mechanical characteristics.

- In Desir et al. [36], standard continuum plasticity is extended to the interface to account for bond deterioration. A return mapping algorithm is used for the integration of relative plastic displacement.

- In Ghosh et al. [49] interfacial separation is conducted by cohesive elements in multiple fiber reinforced composites. The connection force is carried out by normal and tangential springs.

An interesting approach is presented by [3, 36] in which the relative displacement in the interface is included into a finite element as a slip degree of freedom. Some other authors [12, 26, 36] included an elastoplastic model for the interface. All these approaches consider a displacement-based finite element formulation for the treatment of the interface. Some works [101] prove that the force-based approaches for elements with bond-slip lead to non-symmetric tangent operators, or may not completely simulate the material behaviour.

5.2.1 Mechanics of bond

Bond is the physical phenomenon by which two constituents or composite material components are adhered. In this section,the parameters that affect the bond response to loading are described. The different bond models used in finite element implementations aim to create a relationship between the relative displacement of the constituents and the shear bond stress caused on the contacting surfaces. One of the simplest bond model is the Coulomb friction model in which the shear bond stress is considered proportional to the normal stress in the contacting surfaces. In this case, the shear stress is non dependant of the slip or relative displacement between the surfaces in contact. For this reason, the Coulomb bond friction model is unrealistic because the complex behavior in the interface needs to consider the slip of surfaces in contact in order to get a reasonable measurement of the bond stresses along the interface. In reinforced materials with fine fibres, debonding and friction forces may appear at the intersections of fibre and cracks in the matrix due to the crack opening under tensile strain [128]. The fracture mechanics associated with the matrix is out of the scope of this study. Thus, the bridging effect, in the rebar, caused by a crack is simulated through the inclusion of a notch in the matrix, see the example in Section 5.6.2.

Micro behavior around reinforcing bars or fibres

The bond consists of three basic phases: adhesion, friction and mechanical interlock. Friction does not develop until adhesion has failed. The chemical adhesion is usually neglected as its effect is small. The main adhesion or first stage of the bond mechanism is present provided that ribs exist in the reinforcing bars. For the sake of simplicity the terms reinforcing bars or fibres are

used alternatively although they are associated with very different composite materials. In the case of ribbed bars, the onset of bond failure can physically be seen as the transfer of forces from the ribs to the surrounding continuum and the contacting surfaces between ribs are debonded. This effect creates circumferential tensile forces around the bar in the continuum. Failure occurs if these tensile stresses reach the yield stress of the surrounding continuum [124]. The confinement or circumferential stress σ_r is related to the shear stresses τ_{bond} through the slope formed between the ribs and the axial direction of the bar [119]. Thus, it is stated as $\sigma_r = \tau_{bond} * \tan(\theta)$, where θ is the angle between the rib side perpendicular direction and the tangential direction or direction of the contacting surfaces.

5.3 Constitutive models for the matrix

The Mohr-Coulomb constitutive model is used to describe the behaviour of the matrix when subjected to infinitesimal strains. More details of the model used are described below. Also, some elastic-matrix simulations have been conducted in order to see the bridging effect of the re-bars when subjected to finite strains.

5.3.1 Mohr-Coulomb yield surface

This criterion assumes that the plastic yielding is the result of friction between particles. It is a generalisation of *Coulomb's Friction Law.* It states that *'the onset of plastic yielding begins when the function of shear stress and normal stress defined by Eq (5.1) reaches a critical value'.*

$$\tau = c - \sigma_n \tan\phi \tag{5.1}$$

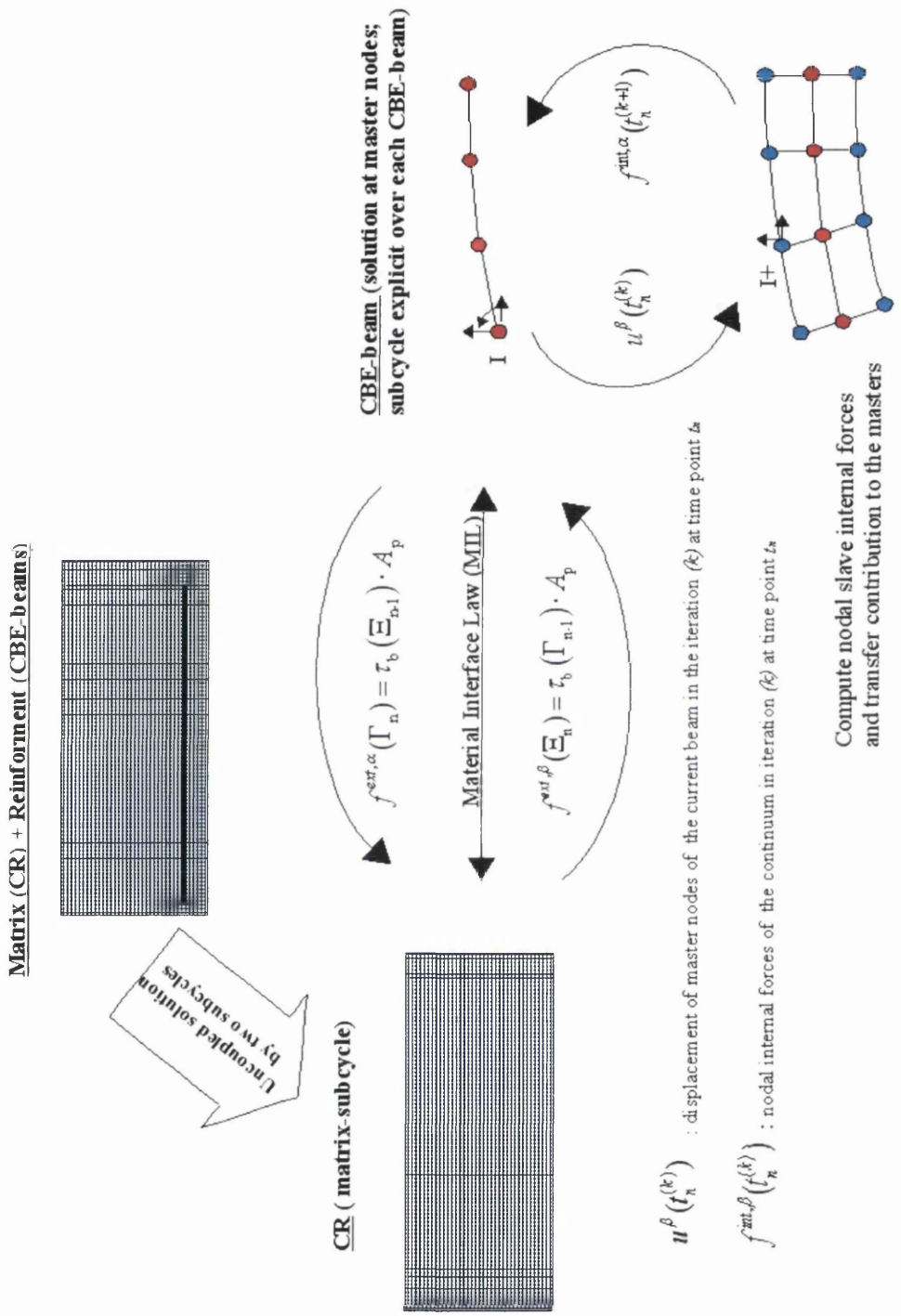


Figure 5.2: Resume of the implemented scheme

where c is the cohesion, σ_n the normal stress, τ the shear stress and ϕ the angle of internal friction. This defines six planes converging in an apex in the space of stresses. In terms of principal stresses the criterion can be expressed as in Eq (5.2).

$$\frac{\sigma_I - \sigma_{III}}{2}\cos\phi = c - \left(\frac{\sigma_I + \sigma_{III}}{2} + \frac{\sigma_I - \sigma_{III}}{2}\sin\phi\right)\tan\phi \qquad (5.2)$$

where $\sigma_I \geq \sigma_{II} \geq \sigma_{III}$. Eq(5.2) can be rearranged as follows,

$$(\sigma_I - \sigma_{III}) + (\sigma_I + \sigma_{III})\sin\phi = 2c\cos\phi \qquad (5.3)$$

In view of this, the yield surface can be expressed as the set of stresses that make $\Phi = 0$.This defines a pyramidal surface with the apex located at $c = \cot\phi$ in the tensile side of the hydrostatic axis. Its pyramidal shape differs from the prism that defines the Tresca yield surface. This makes the Mohr-Coulomb yield surface pressure-sensitive. It is interesting to remark that both criteria coincide in the absence of internal friction when $\phi = 0$. The apex defines the strength of the material in a tensional state of stress. Limited resistance in tension is a typical characteristic of some materials such as concrete or soils, and for this reason has been chosen to represent concrete material in the present study

$$\Phi(\boldsymbol{\sigma}, c) = (\sigma_I - \sigma_{III}) + (\sigma_I + \sigma_{III})\sin\phi - 2c\cos\phi \qquad (5.4)$$

The six plane surfaces that enclose the elastic domain as a function of a generic state of stress are written next. Each plane of the pyramid is defined by one

of these equations with $\Phi_i(\boldsymbol{\sigma}, c) = 0$.

$$\Phi_1(\boldsymbol{\sigma}, c) = (\sigma_I - \sigma_{III}) + (\sigma_I + \sigma_{III})\sin\phi - 2c\cos\phi$$

$$\Phi_2(\boldsymbol{\sigma}, c) = (\sigma_{II} - \sigma_{III}) + (\sigma_{II} + \sigma_{III})\sin\phi - 2c\cos\phi$$

$$\Phi_3(\boldsymbol{\sigma}, c) = (\sigma_{II} - \sigma_I) + (\sigma_{II} + \sigma_I)\sin\phi - 2c\cos\phi$$

$$\Phi_4(\boldsymbol{\sigma}, c) = (\sigma_{III} - \sigma_I) + (\sigma_{III} + \sigma_I)\sin\phi - 2c\cos\phi$$

$$\Phi_5(\boldsymbol{\sigma}, c) = (\sigma_{III} - \sigma_{II}) + (\sigma_{III} + \sigma_{II})\sin\phi - 2c\cos\phi$$

$$\Phi_6(\boldsymbol{\sigma}, c) = (\sigma_I - \sigma_{II}) + (\sigma_I + \sigma_{II})\sin\phi - 2c\cos\phi$$

Analogously, the Mohr-Coulomb criterion can be expressed as a function of the stress invariants Eq (5.5) [27].

$$\Phi = \Big(\cos\theta - \frac{1}{\sqrt{3}}\sin\theta\sin\phi\Big)\sqrt{J_2(\mathbf{s})} + p(\boldsymbol{\sigma})\sin\phi - c\cos\phi \tag{5.5}$$

where θ is the Lode angle defined in Eq (5.6)which is a function of the deviatoric stresses through of its second and third invariants.

$$\theta(\mathbf{s}) := \frac{1}{3}\arcsin\left(\frac{-3\sqrt{3}J_3(\mathbf{s})}{2\big(J_2(\mathbf{s})\big)^{\frac{2}{3}}}\right) \tag{5.6}$$

5.3.2 Associative and non-associative plastic flow rules for Mohr-Coulomb

In the associative laws the yield function is adopted as the flow potential. The flow rule requires the consideration of singularities such as at the apex and the intersection of the faces of the pyramid or edges. Plastic flow can be initiated from an edge, a face or the apex. In the derivation of the flow rules at faces or edges, the analysis can be performed in a sextant (with principal stresses $\sigma_1 \geq \sigma_2 \geq \sigma_3$) of a cross-section of the Mohr-Coulomb pyramid Figure (5.3.2). The vectors N_a and N_b are not entirely deviatoric as they have a non-zero value

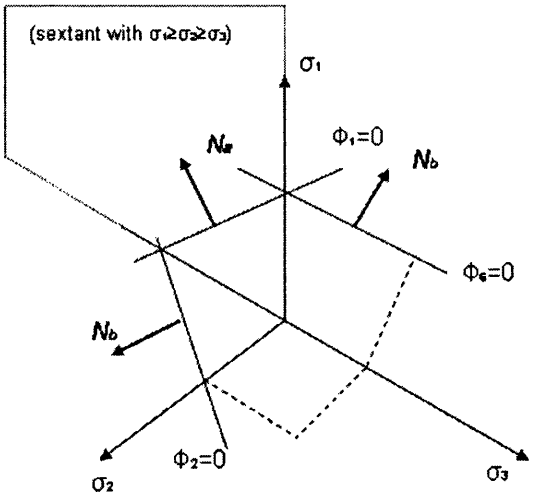


Figure 5.3: Sextant of a cross-section of the Mohr-Coulomb pyramid. Mohr-Coulomb flow rule

of the component along the hydrostatic axis. The vectors in Figure (??) are deviatoric projections as we are cutting the hydrostatic axis perpendicularly. The flow rule for plastic evolution from a face is,

$$\dot{\varepsilon}^p = \dot{\gamma}\mathbf{N}_a \tag{5.7}$$

where $\mathbf{N}_a$ is perpendicular to the plane $\Phi_1 = 0$, i.e.,

$$\mathbf{N}_a = \frac{\partial \Phi_1}{\partial \boldsymbol{\sigma}} = (1 + sin\phi)\,\mathbf{e_1} \otimes \mathbf{e_1} - (1 - sin\phi)\,\mathbf{e_3} \otimes \mathbf{e_3} \tag{5.8}$$

At the corners the flow rule is given by,

$$\dot{\varepsilon}^p = \dot{\gamma}_a\mathbf{N}_a + \dot{\gamma}_b\mathbf{N}_b \tag{5.9}$$

At the intersection of $\Phi_1 = 0$ and $\Phi_6 = 0$, $\mathbf{N}_b$ is expressed by,

$$\mathbf{N}_b = \frac{\partial \Phi_1}{\partial \boldsymbol{\sigma}} = (1 + sin\phi)\,\mathbf{e_1} \otimes \mathbf{e_1} - (1 - sin\phi)\,\mathbf{e_2} \otimes \mathbf{e_2} \tag{5.10}$$

At the intersection of $\Phi_1 = 0$ and $\Phi_2 = 0$, $\mathbf{N}_b$ is expressed by,

$$\mathbf{N}_b = \frac{\partial \Phi_1}{\partial \boldsymbol{\sigma}} = (1 + sin\phi)\,\mathbf{e_2} \otimes \mathbf{e_2} - (1 - sin\phi)\,\mathbf{e_3} \otimes \mathbf{e_3} \tag{5.11}$$

At the apex of the pyramid, the six planes intersect and, hence, six normals are defined. The plastic flow rule in this singular region is a combination of these six normals Eq (5.12). It is interesting to remark that this associative plastic flow rule predicts a non-zero volumetric plastic strain, Eq (5.13), due to the pressure sensitivity of the Mohr-Coulomb criterion.

$$\dot{\varepsilon}^p = \sum_{k=1}^{6} \mathbf{N}_k \tag{5.12}$$

Eq (5.13) is positive as $\dot{\gamma}_k \geq 0 \ \forall k$ and, therefore, dilatant. This dilatancy predicted by the associative Mohr-Coulomb plastic flow rule is often excessive [116]. To overcome this problem, a non-associative plastic flow rule can be performed. It adopts, as the flow potential, a Mohr-Coulomb yield function with a frictional angle ϕ which is replaced by an usually smaller angle ψ. The dilatancy predicted is proportional to $\sin\psi$. For $\psi = 0$ the plastic flow rule reduces to the associative Tresca law at all points of the yield surface except at the apex as the plastic flow becomes purely deviatoric.

$$\dot{\varepsilon}_v^p = (2\sin\phi)\sum_{k=1}^{6} \dot{\gamma}_k \tag{5.13}$$

5.3.3 Integration algorithm for the Mohr-Coulomb material model

The update of stress is performed in the principal stress space. Therefore, the spectral decomposition of stresses after the trial stress computation is carried out. Thus, the principal stresses in the elastic prediction are arranged as follows,

$$\sigma_I^{trial} \geq \sigma_{II}^{trial} \geq \sigma_{III}^{trial}$$

Then, the yielding is checked with the consistency inequality, from Eq (5.3),

$$\Phi^{trial} \equiv (\sigma_I^{trial} - \sigma_{III}^{trial}) + (\sigma_I^{trial} + \sigma_{III}^{trial}) \sin\phi \leq 2c(\bar{\varepsilon}_n^p) \cos\phi \tag{5.14}$$

If inequality Eq (5.14) is satisfied, then, the step is purely elastic and the variables are directly updated with the trial values. On the contrary a plastic correction needs to be carried out. The general return mapping Eq (5.15) for the stress tensor is used to correct the stresses values which lie now on the softened yield surface.

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_{n+1}^{trial} - \mathbf{D}^e : \Delta\boldsymbol{\varepsilon}^p \tag{5.15}$$

Taking in account the expressions Eqs (5.7,5.9,5.12), Eq (5.15) reaches the expression,

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_{n+1}^{trial} - \Delta\gamma \mathbf{D}^e : \mathbf{N}_{n+1} \tag{5.16}$$

There are several return mappings depending on the situation of stress on the principal space. This is due to the singularities of the Mohr-Coulomb yield surface (see Section 5.3.2).

5.4 Subcycling

5.4.1 Governing equations for the matrix

Henceforth, two-phases of the reinforced material will be considered. It is formed by two constituents: matrix or continuum reference (α) and reinforcing bars or reinforcement (β). The residual equations, resulting from finite element discretisation of the continuum, (5.17) are linearised (5.20) in order to accomplish a Newton-Raphson procedure. The matrix is modeled either by the

Mohr-Coulomb material model [116] or an elastic model. A return mapping algorithm for the integration of the stress $\hat{\boldsymbol{\sigma}}$ is conducted [106].

$$\boldsymbol{R}(\boldsymbol{u}_n^\alpha) = \mathbf{f}_\alpha^{int}(\boldsymbol{u}_n^\alpha) - \mathbf{f}_\alpha^{ext} = 0 \tag{5.17}$$

where the force vectors are calculated as,

$$\mathbf{f}_\alpha^{int}(\boldsymbol{u}_n) = \bigwedge_{e \in \Im_\alpha} \left\{ \int_{\Omega^{(e)}} \mathbf{B}^T \hat{\boldsymbol{\sigma}}(\boldsymbol{u}_n^\alpha)\, \mathrm{d}v \right\} \tag{5.18}$$

$$\mathbf{f}_\alpha^{ext} = \bigwedge_{e \in \Im_\alpha} \left\{ \int_{\Omega^{(e)}} \mathbf{N}^T \boldsymbol{b} \mathrm{d}v + \int_{\partial\Omega^{(e)}} \mathbf{N}^T \boldsymbol{q}\, \mathrm{d}s \right\} \tag{5.19}$$

where $\bigwedge_{e \in \Im_\alpha}$ is the assembly operator for the elements in the subdomain of the matrix $\Im_\alpha$, $N(\xi, \eta)$ are bilinear shape functions, $\mathbf{b}$ are the body forces, $\mathbf{q}$ the traction forces applied over the boundary of the body and $\mathbf{B}$ is the linear strain operator. The integrals are approximated by gaussian quadrature.

$$\mathbf{K}_T \delta \boldsymbol{u}^{\alpha\,(k)} = -\boldsymbol{R}^{(k-1)}(\boldsymbol{u}_{n+1}^\alpha) \tag{5.20}$$

where $\mathbf{K}_T$ is the global tangent stiffness matrix given as,

$$\mathbf{K}_T = \left. \frac{\partial \boldsymbol{R}}{\partial\, \boldsymbol{u}^\alpha} \right|_{\boldsymbol{u}_{n+1}^{(k-1)}} \tag{5.21}$$

k is the Newton-Raphson integration step, and n refers to the external load increment. If the solution would diverge the algorithm swaps to an explicit solution as described in [31]. The external force is updated at the beginning of the matrix-subcycle with the bond forces vector. This vector is formed for the longitudinal bond forces (Section 5.4.3).

$$\mathbf{f}_\alpha^{ext\,(j+1)} = \mathbf{f}_\alpha^{ext\,(0)} + \mathbf{f}_{\beta \rightleftharpoons \alpha}^{link\,(j)} \tag{5.22}$$

5.4.2 Governing equations for the reinforcement

The finite element discretisation by beam elements of the momentum equations renders the following system of second order differential equations,

$$\mathbf{M}_\beta \, \ddot{\mathbf{u}}^\beta(t_m) + \mathbf{C}_\beta \, \dot{\mathbf{u}}^\beta(t_m) + \mathbf{f}_\beta^{int}(\mathbf{u}_m^\beta) = \mathbf{f}_\beta^{ext} + \mathbf{f}_{\beta \rightleftharpoons \alpha}^{link} \tag{5.23}$$

where $\mathbf{M}_\beta$ is a lumped mass matrix, $\mathbf{C}_\beta$ is the mass proportional damping matrix, $\mathbf{f}_\beta^{ext}$ are the external forces applied directly to the reinforcing bar and $\mathbf{f}_\beta^{int}(\mathbf{u}_m^\beta)$ the internal forces vector.

$$\mathbf{f}_\beta^{int}(\boldsymbol{u}_m^\beta) = \bigwedge_{e \in \Im_\beta} \left\{ \int_{\Omega^{(e)}} \mathbf{B}^T \, \hat{\boldsymbol{\sigma}}(\boldsymbol{u}_m^\beta) \, \mathrm{d}v \right\} \tag{5.24}$$

$$\mathbf{f}_\beta^{ext} = \bigwedge_{e \in \Im_\beta} \left\{ \int_{\Omega^{(e)}} \mathbf{N}^T \, \boldsymbol{b} \mathrm{d}v + \int_{\partial\Omega^{(e)}} \mathbf{N}^T \, \boldsymbol{q} \, \mathrm{d}s \right\} \tag{5.25}$$

The system is solved by *dynamic relaxation* (Chapter 3) and Central Difference Method as described below. The static solution is obtained from the attenuation of the transient response, leaving the static solution for the applied load. Initiation of the subcycle ($m = 0$) is given by,

$$\dot{\mathbf{u}}_{\frac{1}{2}}^\beta = \mathbf{M}_\beta(\mathbf{f}_\beta^{ext}(t_0) + \mathbf{f}_{\beta \rightleftharpoons \alpha}^{link}(t_0) - \mathbf{f}_\beta^{int}(t_0)) \frac{\Delta t_1}{2} \tag{5.26}$$

$$\mathbf{u}_0^\beta = \mathbf{u}_n^\alpha \tag{5.27}$$

$$\mathbf{u}_1^\beta = \mathbf{u}_0^\beta + \Delta t_{\frac{1}{2}} \dot{\mathbf{u}}_{\frac{1}{2}}^\beta \tag{5.28}$$

and following iterations by,

$$\dot{\mathbf{u}}_{m+\frac{1}{2}}^\beta = \frac{2\mathbf{M}_\beta - \Delta t_m \mathbf{C}_\beta}{2\mathbf{M}_\beta + \Delta t_m \mathbf{C}_\beta} \dot{\mathbf{u}}_{m-\frac{1}{2}}^\beta + \frac{\mathbf{f}_\beta^{ext}(t_m) + \mathbf{f}_{\beta \rightleftharpoons \alpha}^{link}(t_m) - \mathbf{f}_\beta^{int}(t_m)}{2\mathbf{M}_\beta + \Delta t_m \mathbf{C}_\beta} \tag{5.29}$$

The update and dynamic relaxation quotient are given by,

$$\mathbf{u}_{m+1}^{\beta} = \mathbf{u}_{m}^{\beta} + \Delta t_{m+\frac{1}{2}} \dot{\mathbf{u}}_{m+\frac{1}{2}}^{\beta} \tag{5.30}$$

$$\mathbf{C}_{\beta} = \sqrt{\frac{\mathbf{u}_{m}^{\beta T} \mathbf{K}_{m}^{\beta} \mathbf{u}_{m}^{\beta}}{\mathbf{u}_{m}^{\beta T} \mathbf{M}_{\beta} \mathbf{u}_{m}^{\beta}}} \tag{5.31}$$

$$K_{ii}^{\beta} = \frac{f_i^{\beta}(\mathbf{u}_{m}^{\beta}) - f_i^{\beta}(\mathbf{u}_{m-1}^{\beta})}{\Delta t_{m-\frac{1}{2}} \dot{u}_{m-\frac{1}{2}}^{\beta}} \tag{5.32}$$

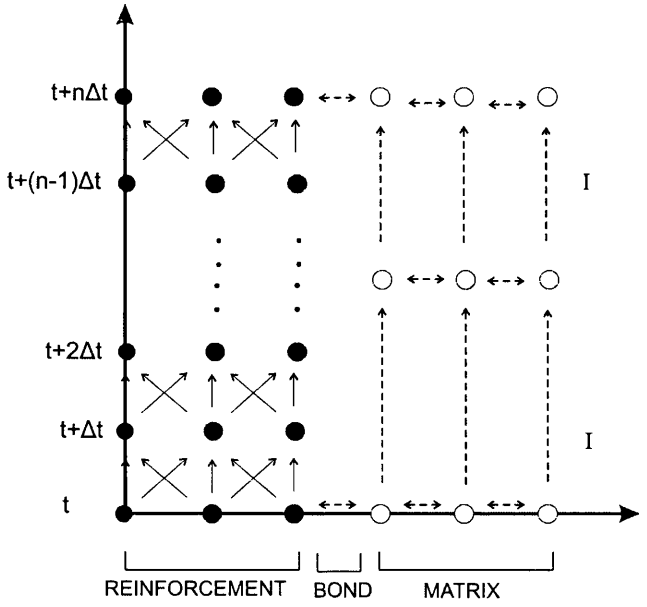


Figure 5.4: Schematic representation of a cycle in which the reinforcement subcycle is executed by an explicit method and the matrix by an implicit method.

5.4.3 Kinetic link

The interface is defined as the zone of nodal transmission of forces and displacements. The displacements are used to calculate the slip or relative dis-

placement between the matrix and reinforcement. Initially, the nodes of the matrix and reinforcement in contact coincide in the same spatial point. Once that a relative displacement is produced an interpolation by linear shape functions between two nodes is carried out. The bond stress depends upon the relative displacement between matrix and reinforcement.

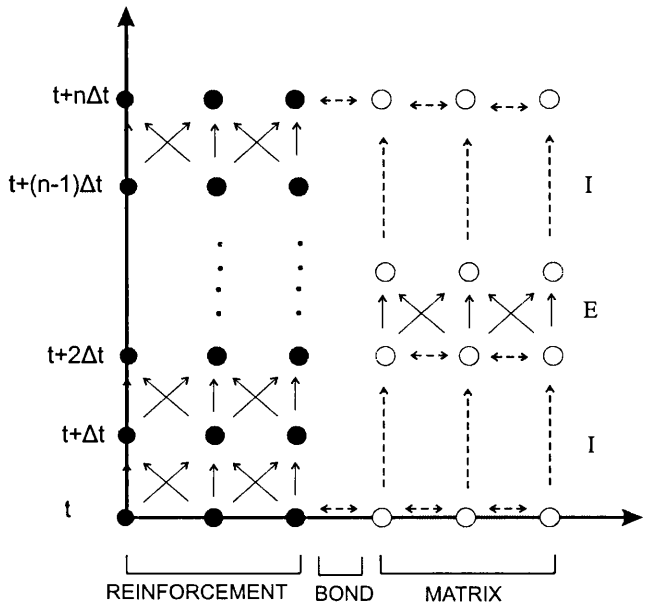


Figure 5.5: Schematic representation of a cycle. Reinforcement subcycle is solved by an explicit method and the matrix by I/E algorithm.

The bond stress is calculated through the interface constitutive law ϑ, from the reinforcement to the matrix:

$$\tau_b(\Im_\beta^{(j)}) = \vartheta(s_x(\Im_\alpha^{(j)})) \tag{5.33}$$

The slip is simply given by,

$$s_i(\Im_\alpha^{(j)}) = u_i^{\alpha\,(j-1)} - u_i^{\beta\,(j)} \tag{5.34}$$

where $u_i^{\alpha\,(j-1)}$ is the tangential displacement (of the matrix node with component i) to the axis of the current reinforcing bar from subcycle $(j-1)$. $u_i^{\beta\,(j)}$

is the longitudinal displacement of a node in the reinforcing bar. i refers to a local node in the interface. They do not correspond to global numbers of d.o.f. either of the matrix or of the reinforcement. The relative displacement, in a point x between nodes i and $i+1$, is interpolated as follows,

$$s_x(\Im_\alpha^{(j)}) = s_i(\Im_\alpha^{(j)})\, N_i(x) + s_{i+1}(\Im_\alpha^{(j)})\, N_{i+1}(x) \tag{5.35}$$

where $N_i(x)$ is a linear interpolation function. The external force for the next subcycle is calculated through equilibrium balance in a slice considering the reinforcing bar, the matrix and the interface. If A_p is the perimetric area surrounding the reinforcing bar between two nodes, the bond force which is going to be treated as an external force for the matrix subcycle (from the bond stress computed in the reinforcement subcycle) is:

$$f_{\alpha\, i}^{bond} = \tau_b(\Im_\beta^{(j)})\, A_p \tag{5.36}$$

This component will be part of the assembled external forces array for the next matrix subcycle. The forces, which are perpendicular to the re-bar, are also assembled in this array. The kinematic continuity is enforced from α to β as an initial condition for the explicit method Eq(5.27). But this is not enforced from β to α.

5.5 Interface constitutive law

Several laws or relationships between bond stress and relative displacement or slip for materials such as reinforced concrete have been proposed in the literature, see for instance [99],[108]. Henceforth, the matrix is considered to be concrete and the reinforcement is a steel reinforcing bar in order to specify a more realistic reinforced material. The simplest ones suggest an initial linear

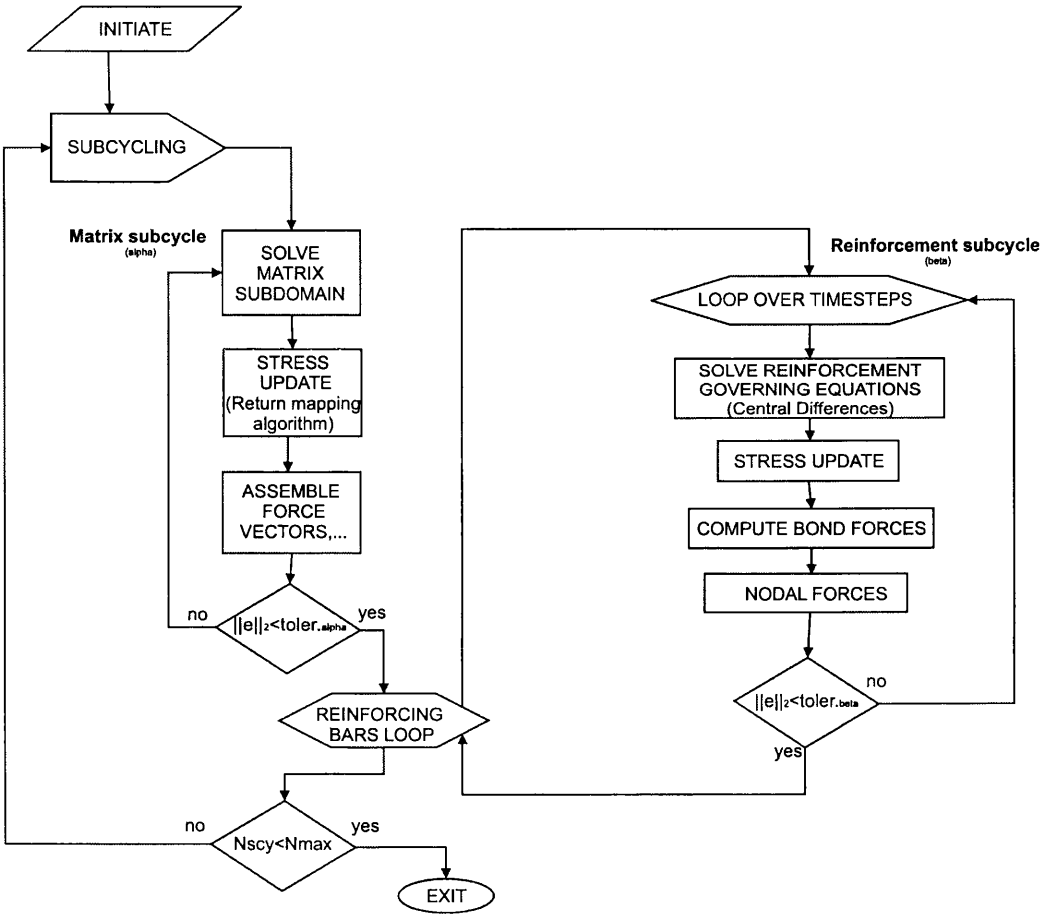


Figure 5.6: Flowchart of the subcycling scheme developed. For sake of simplicity, only the implicit flowchart, for the matrix (α), is represented. Although, the matrix is solved explicitly in case of divergence of the implicit method

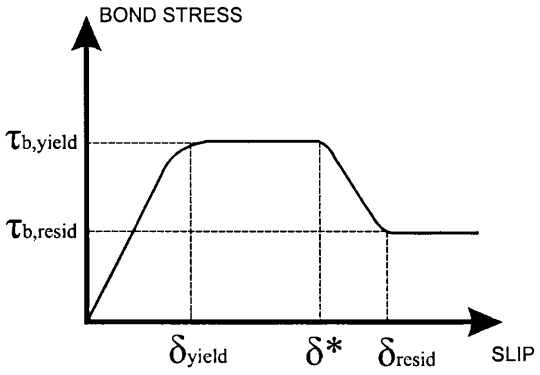


Figure 5.7: Relationship at the interface in which the bond stress is given as a function of the slip at the interface

part followed by a constant value of the bond stress after a threshold value. These would not model the effect of maximum slip when the ribs of the reinforcing bar have sufficiently eroded the interface in the concrete, leaving a minimum residual bond stress in the system. This residual bond stress is represented in the Figure 5.7. As a result of reaching a critical slip δ_* a softening in the interface is developed. This effect is physically due to the deterioration that the ribs of the rebar induce on the surrounding continuum until reaching a residual bond stress $\tau_{b,resid}$.

In a pullout test, the reinforcing bar embedded in concrete is loaded in the axial direction at the free edge. An empirical relationship based in experimental pullout tests [39] is used in the implementation of the I/E-I algorithm to compare with realistic experiments. In the experimental pullout tests conducted by [39], each bar was instrumented with seven or eight strain gauges in the axial direction. The shear stress was calculated as $\tau_{1-2} = \frac{E}{r}(\epsilon_1 - \epsilon_2)\, 2\, l_{1-2}$, where l_{1-2} is the distance between gauges, r is the bar radius, E the modulus of elasticity of the bar material and ϵ_i the strain measures provided by the

gauges. A relationship $\vartheta(s)$ is proposed based in the experimental evidence and analytical results,

$$\vartheta(s) = \begin{cases} 0.2\,K_{bond}\,\frac{s}{s_o} + 0.1\,\tau_b^{max}\left(5\frac{s}{s_o} - 4.5\left(\frac{s}{s_o}\right)^2 + 1.4\left(\frac{s}{s_o}\right)^3\right), \\ \qquad\qquad \forall \quad |\frac{s}{s_o}| < \delta^* \\ sign(s)\cdot\tau_b^{max}, \qquad \forall \quad |\frac{s}{s_o}| \geq \delta^* \end{cases}$$

where s is the relative displacement (or slip) between the master nodes of the reinforcing bars and the nodes of the matrix in contact with the reinforcing bar, s_o is a material parameter indicating the deformation at which perfect slip occurs [2] and $\delta^* = \frac{\tau_b^{max}}{K_{bond}}$.

5.6 Numerical tests

5.6.1 Pullout test

This type of test is chosen to validate I/E-I because of the limit load that may be applied, although this sort of load directly applied on the bar is not usual in practical structures. The bar is loaded ,as described above, on the free edge with a point force P. The embedded edge and the remaining nodes of the bar do not have any constraint to the motion except those imposed by the matrix.

Details of the geometry are expressed in Figure (5.8). The matrix nodes at the boundary (except those on the side in which the exterior edge of the bar is located) are constrained in all their degrees of freedom.

The matrix mesh is formed by 250 quadrilateral elements with a thickness of $78.5\,mm$. The reinforcing bar is composed of 8 CBE elements (see chapter

[2] [108] suggests a value $s_o = 0.06\,mm$ for a typical reinforced concrete

4) of cross-section area $A_{cs} = \pi \cdot 10^{-4}\, m^2$. The cross-section area is assumed to be square for the purposes of analysis. Obviously, the experimental results conducted by [39] were performed with cylindrical metallic bars. The total length of the embedded rebar is $l_{emb} = 0.35\, m$. The material properties are given in Table (5.1). The matrix has been modelled with a Mohr-Coulomb law in which friction and dilatancy angles coincide (associative flow). The interface law $\vartheta(s)$ used is described in Section 5.7 with $\delta^* = 0.00032\, m$ and $\tau_b^{max} = 8\, MPa$.

	Matrix	Reinforcing bar
Young Modulus	$14\, GPa$	$180\, GPa$
Poisson Ratio	0.1	0.33
Yield stress		$300\, MPa$
Mass density	$2400\, Kg/m^3$	
Friction angle (degrees)	20.0	
Dilatancy angle (degrees)	20.0	

Table 5.1: Material properties of reinforced material used in the pullout tests .

The reinforcing bar is loaded up to an external force of $100\, KN$ in the longitudinal direction. The matrix undergoes elongation induced by the axial

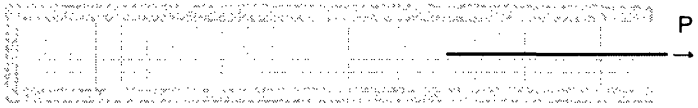


Figure 5.8: Mesh and geometry of the tested pullout specimen

deformation of the reinforcing bar at the interface. The behaviour of bond stress along the bar is highly non-linear Figure (5.11). It is observed that the level of relative displacements is sufficiently high to reach the yield bond stress ($8\,MPa$) near to the loaded edge.

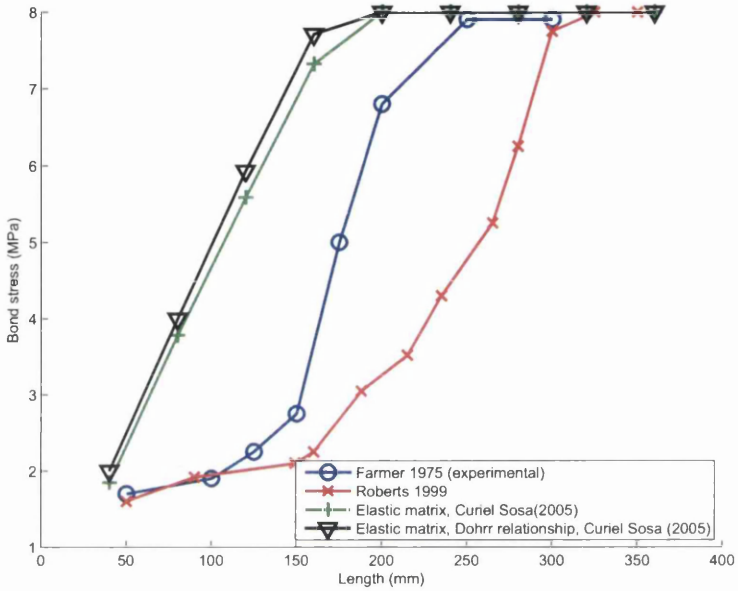


Figure 5.9: Numerical versus experimental bond stress along the reinforcement. Chart for Curiel Sosa(2005) considered reinforcing bar embedded in an elastic matrix . External force applied in the outside edge = $100\,KN$. Material parameters taken from the Table (5.1)

In Figure (5.11) the close agreement between predicted and experimental values is observed over almost the entire length of the bar. This results agree closer than other results from the literature [99] along the most of the rebar length. However, the predicted bond stress near the embedded edge is clearly in disagreement with experiments (approximately one third of the total length). The discrepancy in this part is easily detected in the left region of Figure (5.11).

This is probably due to the interface constitutive model adopted as the

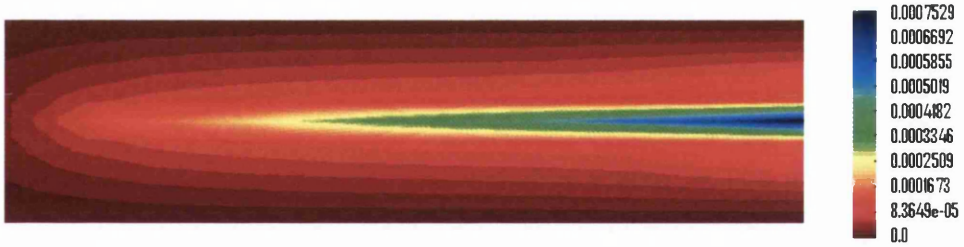


Figure 5.10: Displacement field contours (m) for half sample. Matrix was considered elastic in this simulation.

algorithm is in agreement in this region only when perfect bond is assumed in the interface. Nevertheless, the elastic matrix do not describe the physical nature of the concrete and, hence, fails to agree with experimental evidence along the remaining of the rebar, Figure (5.9). Contours fields of displacements in the elastic case can be observed in Figure (5.10)

ε^p	*cohesion*
0.00142	$0.20 \cdot 10^{+8}$
0.002	$0.15 \cdot 10^{+8}$

Table 5.2: Softening curve for the Mohr-Coulomb model of the matrix

5.6.2 Bridging effect

I/E-E is primarily designed to provide an useful research tool to solve the jump-discontinuity caused by the interface. With an aim to validate this program, simulations on flexural behaviour of beams and the effect of the reinforcement are undertaken. The simulations were performed over a range of materials for the matrix. Tests were conducted on notched beams with longi-

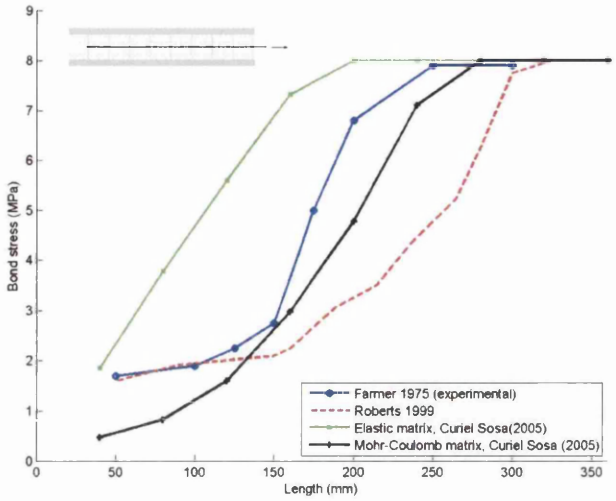


Figure 5.11: Numerical versus experimental comparison of bond stress along the reinforcement. External force = $100\,KN$. Matrix was modeled with Mohr-Coulomb material model with the data specified at Tables 5.1,5.2

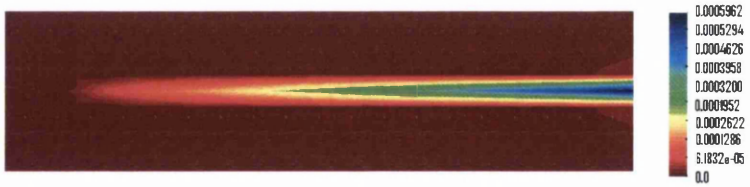


Figure 5.12: Displacement field contours (m) (half sample) for imperfect bond between re-bar and concrete. Matrix was modeled with Mohr-Coulomb material model with the data specified at Tables 5.1,5.2

tudinal reinforcement to assess the bridging effect of the reinforcing bar over the notch. In addition, contour plots of displacements are provided. In the next set of numerical examples, the efficiency of the proposed subcycling algorithm is demonstrated in the simulation of problems with interfacial jumps and singularities. The tip of the notch causes a well-known singularity. Singularities are undesirable as they cause numerical instabilities and divergences. Therefore, these examples conduct a critical assessment of the numerical scheme presented. The proposed strategy is demonstrated to be efficient for this type of problem. The examples include two-dimensional three-point bending tests over a notched reinforced specimen Figure (5.13). The external loading is applied either displacement-driven or force-driven to demonstrate the capabilities of the algorithm. Linear triangular finite elements with three Gauss points for the numerical integration were adopted. A refinement of the mesh is conducted around of the tip of the notch, see Figure (5.14). The reinforcing bar is modelled by beam elements. One of the two supports of the specimen allows motion in the x-direction (horizontal). A perfect bond is assumed between the reinforcing bar and the surrounding continuum.

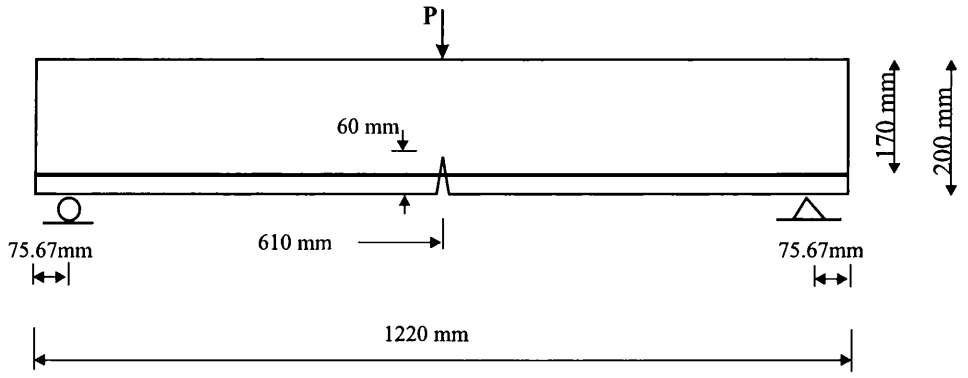


Figure 5.13: Notched sample geometry

Test 5.4: Bridging effect on notched beam

The material properties of the matrix are: elastic modulus $35GPa$ and Poisson ratio 0.2. And for the reinforcing bar: 200 GPa and 0.33 respectively. In that context, we have a reinforcement that is more rigid than the matrix. As a consequence, the jump in the interface is critical for the convergence of the methods. The subcycling algorithm is demonstrated to be particularly efficient in this case taking also into account the presence of the singularity introduced by the tip of the notch. In the simulations, it was observed that a stress localisation occurs as expected in the area around of the notch.

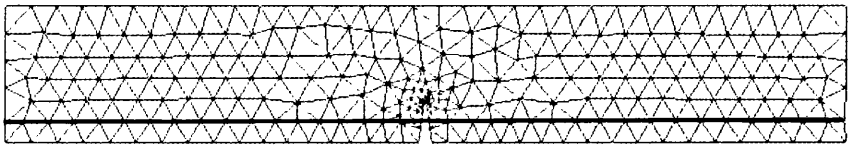


Figure 5.14: Detail of the refinement of the mesh around the notch

In Figure (5.15) or Figure (5.17)is observed that the effect of the top-compression and bottom-tension, of the reinforcing bar close to the bridging region, induces opposite signs of the shear stress (in the elements of the matrix). This expected effect is responsible for debonding in case the system is incrementally loaded to a further stage. Debonding is not simulated in this work. In Figure (5.16), the contours fields of transversal stress are plotted. The reason is to assess that the tip of the notch is effectively a stress raising singularity. This is shown in the zoomed region of the Figure (5.16). In this case, the driven vertical displacement was set to $0.05mm$. A test with larger deformation is represented in Figure (5.18)

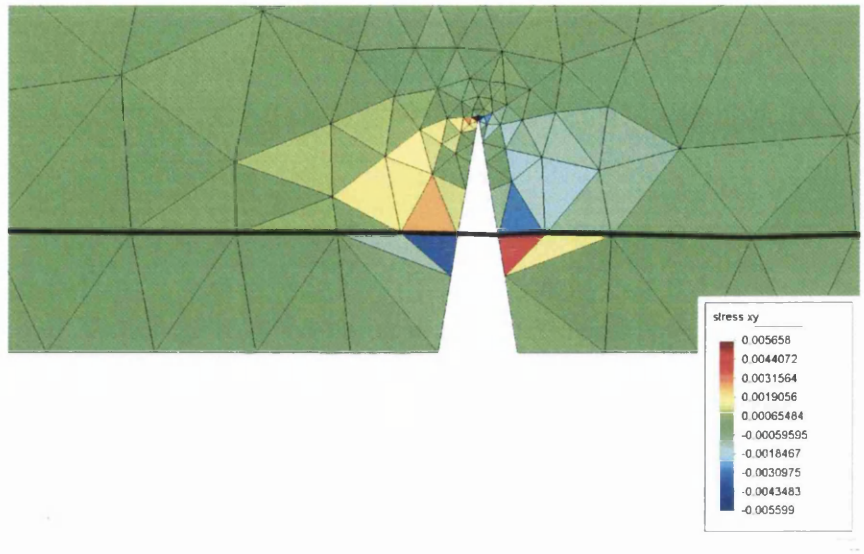


Figure 5.15: Detail of the shear stress (GPa) around the notch. Displacement driven: $0.0001mm$ (top of the beam)

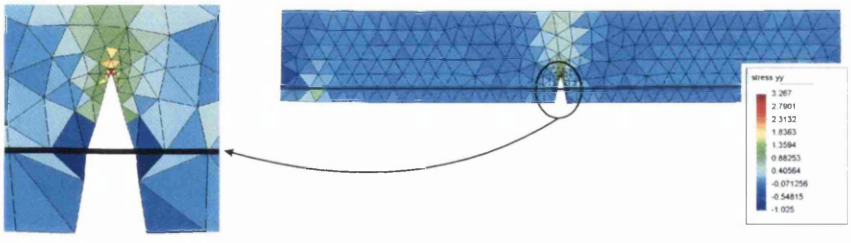


Figure 5.16: Transversal stress (GPa). Displacement driven: $0.05mm$ (top of the beam)

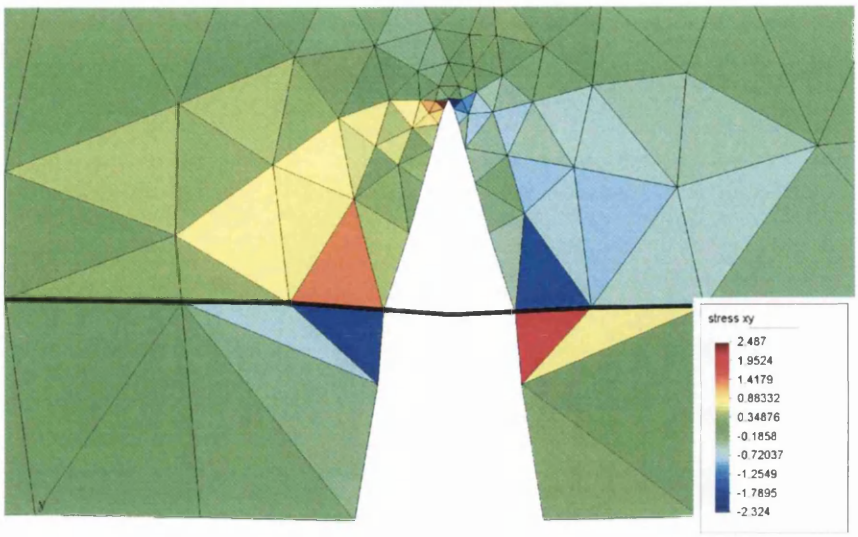


Figure 5.17: Shear stress (GPa) around the notch. Displacement driven: $0.05mm$ (top of the beam)

The top of the beam was displacement driven until a value of $0.1220mm$

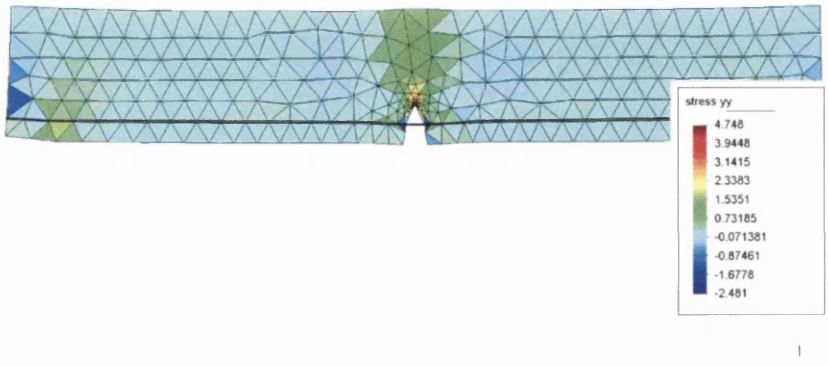


Figure 5.18: Transversal stress (GPa). Displacement driven: $0.1220mm$ (top of the beam)

Chapter 6

Conclusions and Further Research

'Everything of importance has been said before by somebody who did not discover it'. **Alfred N. Whitehead.**

*'So far as the laws of mathematics refer to reality, they are not certain. And so far as they are certain, they do not refer to reality'.***Albert Einstein**

6.1 Concluding remarks and further research

An algorithm that considers the combination in-time of two general strategies of solution (implicit and explicit) has been presented, in particular, for the analysis of large deformations where geometrically non-linear snap-through buckling results in divergence of the implicit method used. This in-time implicit/explicit algorithm (I/E) proved to be efficient and convenient for these type of problems. In the numerical examples conducted, I/E gave results in good agreement with other results from the literature. Several material models,

such as Ogden or Von-Mises, have been included in the codes. The nonlinearity associated to these models required the use of return mapping algorithms for the stress computation. Geometric nonlinearities were also undertaken as described above. It permitted a wide range of numerical tests to validate the I/E code such as large to small deformations and from hyperelastic to elastoplastic materials. Other critical points such as snap-back may also be undertaken by this algorithm. Practical applications of this algorithm might also include the solution of contact problems between complex geometries such as sharp-pointed edges and curved concave surfaces. It is important to remark that there is no partition of the mesh in groups of nodes or elements. The entire mesh is executed with the implicit sub-algorithm and, in case of divergence, the flow of execution is swapped to the explicit sub-algorithm. This is made by appropriate transfer conditions. The scheme is optimized when the I/E code is only executed in a part of the entire mesh (some authors have already made combined meshes ([53] or [85] amongst others) in which a partial mesh was solved by an explicit code (this part of the mesh would include the cause for divergence) and the remaining by an unconditionally stable algorithm.

An interesting further research would be the application of the I/E to non-smooth contact problems. This undesirable-in-simulation contact is frequent in Bioengineering computational problems. Computational Bioengineering is increasing in interest year after year. In this research area, the use of I/E would be an additional tool to save CPU time and to conduct an efficient simulation. Otherwise, the numerical test may be either uneconomical (if only a time-marching technique is carried out very small time steps would be required due to the stability requirements) or inefficient (if the method diverges as a consequence of the problems described above).

In the second part of this research, the development and validation of a two-dimensional computational solution by subcycles are presented. Also, a beam element was developed to model the reinforcing bars. After analysis of the results the main conclusions may be summarised:

1. The accuracy of the finite element computations is preserved due to the uncoupling by subcycles for reinforcement and matrix. The error is assessed independently for each subcycle and, also, for the global cycle. This approach is relevant because of the discontinuity of the stresses in the bimaterial interface.

2. The simulation of pullout tests permitted an excellent prediction of the bond stress vs. slip along most of the length of the interface, taking account of the experimental results. However, this prediction showed a discrepancy with respect to the experimental results in the embedded edge of the re-bar. The main reason is related to the constitutive interface law adopted. Enhancement of this is outside of the main scope of this research and it may be considered as future research given the important role that the interface plays in reinforced materials.

3. The computational model employs a Mohr-Coulomb material model for the matrix. A return mapping algorithm for the integration of the stress is used. Additionally, the first results with an elastic matrix, used in the first validations of the program, are presented.

4. This approach demonstrated a convenient way to enforce the conservation laws. Stability was guaranteed which was a matter of concern in weak couplings by subcycling algorithms. Furthermore, strong coupling by interface elements are not the most convenient option (see Section

5.1) for the presented problems. For those reasons the presented scheme is a mix between weak and strong couplings.

5. The advantage of this scheme lies in the computational efficiency implied by executing reinforcement and continuum calculations at two different rates of execution and solving the jump associated with the interface.

6. Each sub-domain permits treatment by different algorithms and discretisation techniques of well-known efficiency.

7. This partitioned approach permits advantage to be taken of existing codes.

8. As almost always, advantages are not cost free as it requires careful formulation at the interface to avoid degradation in stability when explicit methods are used.

Given the importance that the modelling of reinforced concrete has in the Civil Engineering industry, the development of efficient interface constitutive models continues to be a relevant issue. Further research is needed in that direction. Most commercial codes consider perfect bond in the interface. The enhancement of interface laws would allow better simulations either by using this subcycling scheme or another strategies of simulation dealing with reinforced concrete.

Bibliography

[1] F. M. Andrade Pires, E. A. de Souza Neto, and J. L. de la Cuesta Padilla. An assessment of the average nodal volume formulation for the analysis of nearly incompressible solids under finite strain. *Commun. Numer. Meth. Engng.*, 20:569–583, 2004.

[2] J. Argyris and H-P. Mlejnek. *Dynamics of Structures.* Elsevier, 1991.

[3] S. Balakrishnan and D. W. Murray. Concrete constitutive model for nlfe analysis of structures. *Journal of Structural Engineering*, 114:1449–1466, 1988.

[4] B. Banholzer, W. Brameshuber, and W. Jung. Analytical simulation of pull-out tests - the direct problem. *Cement and Concret Composites*, 27:93–101, 2005.

[5] K. J. Bathe. *Finite Element Procedures.* Prentice-Hall, 1996.

[6] T. Belytschko, R. L. Chiappetta, and H. D. Bartel. Efficient large scale non-linear transient analysis by finite elements. *Int. J. Numer. Meth. Engng.*, 10:579–596, 1976.

[7] T. Belytschko and T.J.R. Hughes, editors. *Computational Methods for Transient Analysis*, The Netherlands, 2001. Elsevier Science B.V.

[8] T. Belytschko and R. Mullen. Mesh partitions of explicit-implicit time integration. In K-J Bathe, J. T. Oden, and W. Wunderlich, editors, *Formulations and Computational Algorithms in finite Element Analysis: U.S. - Germany Symposium*, 2^{nd} edition, pages 673–690, 1978.

[9] T. Belytschko and R. Mullen. Stability of explicit-implicit mesh partitions in time integration. *Int. J. Numer. Meth. Engng.*, 12:1575–1586, 1978.

[10] T. Belytschko and R. Mullen. Mixed methods for time integration. *Comput. Methods Appl. Mech. Engrg.*, 17/18:259–275, 1979.

[11] T. Belytschko, W.K.Liu, and B.Moran. *Nonlinear Finite Elements for Continua and Structures.* John Wiley and Sons, 2001.

[12] M. R. Ben Romdhane and F. J. Ulm. Computational mechanics of the steel-concrete interface. *Int. J. Num. Anal. Meth. Geomech.*, 26:99–120, 2002.

[13] J. F. Besseling and E. Van der Giesser. *Mathematical Modelling of Inelastic Deformation.* CRC Press, 1994.

[14] N. Bicanic. *Nonlinear Finite Element Transient Response of Concrete Structures.* PhD thesis, University of Wales-Swansea, 1978.

[15] J. Bonet. Transient dynamics. Taught master course, University of Wales-Swansea, 2003.

[16] J. Bonet, A. Gil, R. D. Wood, R. Said, and R. V. Curtis. Simulating superplastic forming. *To be published in Comp. Meth. in App. Mech. and Engng.*, 2005.

[17] J. Bonet and R. D. Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis.* Cambridge University Press, 1997.

[18] R. Brighenti. A mechanical model for fiber reinforced composite materials with elasto-plastic matrix and interface debonding. *Computational Materials Science*, 29:475–493, 2004.

[19] I. Bronshtein and K. Semendiaev. *Manual de Matemáticas para Ingenieros y Estudiantes.* MIR Moscu, Alcalá 98, Madrid, Spain, 4th edition, 1988. In Spanish.

[20] P. H. Burgos, M. R. Danta, and J. M. Garcia. *Cinemática y Tensores.* Universidad de Sevilla, Spain, 1989. In Spanish.

[21] P. R. Calhoun and D. A. DaDeppo. Nonlinear finite element analysis of clamped arches. *J. Struct. Engng. ASCE*, 109(3):599–612, 1983.

[22] D. A. Cendón and G. Ruiz. Local fracture and steel-concrete decohesion phenomena studied by means of cohesive and interface elements. In H.A. Mang, F.G. Rammerstorfer, and J. Ebarhardsteiner, editors, *Fifth World Congress on Computational Mechanics, WCCM V*, Vienna, Austria, 7-12 July 2002.

[23] T. Chandrupatla and A. D. Belegundu. *Introduction to finite elements in engineering.* Upper Saddle River: Prentice Hall, New Jersey, 3rd edition, 2002.

[24] A. K. Chopra. *Dynamics of Structures.* Prentice-Hall, 1995.

[25] O. Coussy. *Mechanics of Porous Continua.* Wiley, Chichester, 1995.

[26] J. V. Cox and L. R. Herrmann. Development of a plasticity bond model for steel reinforcement. *Mech. of Cohes.-Fricț. Mater.*, 3:155–180, 1998.

[27] M. A. Crisfield. *Non-linear Finite Element Analysis of Solids and Structures*, volume 2. Wiley, 3 edition, August 2001.

[28] M. A. Crisfield. *Non-linear Finite Element Analysis of Solids and Structures*, volume 1. Wiley, 3 edition, August 2001.

[29] J. L. Curiel Sosa. Testing, finite element analysis and asymptotic expansion homogenization of laminated glass reinforced plastic materials. Master of research, University of Wales Swansea, Singleton Park SA2 8PP, 2002.

[30] J. L. Curiel Sosa, E. A. de Souza Neto, and D. R. J. Owen. In-time implicit-explicit algorithm for nonlinear finite element analysis. In K-J. Bathe, editor, *Third MIT Conference on Computational Fluid and Solid Mechanics*, pages 1090–1094, Cambridge, MA (USA), 14-17 June 2005. MIT, Elsevier.

[31] J. L. Curiel Sosa, E. A. de Souza Neto, and D. R. J. Owen. A combined implicit-explicit algorithm in time for non-linear finite element analysis. *Commun. Numer. Meth. Engng*, 22:63–75, 2006.

[32] J. L. Curiel Sosa, E. A. de Souza Neto, D. R. J. Owen, and N. Petrinic. Modelling of reinforced materials by a subcycling algorithm. In *Proceedings of the III Europpean Conference on Computational Mechanics (ECCM)*, Lisbon, 5-9 June 2006. Associacao Portuguesa de Mecanica Teorica, Aplicada e Computacional.

[33] J. L. Curiel Sosa and D. R. J. Owen. Subcycling model for reinforced concrete. *Computers and Structures*, (X):xxx–xxx, 2007. Submitted.

[34] W. J. T. Daniel. A study of the stability of subcycling algorithms in structural dynamics. *Comput. Methods Appl. Mech. Engrg.*, 156:1–13, 1998.

[35] W. J. T. Daniel. A partial velocity approach to subcycling structural dynamics. *Comput. Methods Appl. Mech. Engrg.*, 192:375–394, 2003.

[36] J. M. Desir, M. R. B. Rhomdane, F. J. Ulm, and E. M. R. Fairbairn. Steel-concrete interface: revisiting constitutive and numerical modeling. *Computers and Structures*, 71:489–503, November 1998.

[37] J. Donea and H. Laval. Nodal partition of explicit finite element methods for unsteady diffusion problems. *Comput. Methods Appl. Mech. Engrg.*, 68:189–204, 1988.

[38] T. M. J. Ellis. *Fortran 77 programming : with an introduction to the Fortran 90 standard.* Addison-Wesley, Wokingham, 2nd edition, 1992.

[39] I. W. Farmer. Stress distribution along a resin grouted rock anchor. *Int. J. Rock Mech. Min. Sci. Geomech.*, 12:347–351, 1975.

[40] C. A. Felippa. Traversing critical points by penalty springs. In M. Nijhoff Pubs, editor, *Proceedings of NUMETA87 Conference, Swansea, Wales*, Dordrecht, Holland, 1987.

[41] C. A. Felippa. Nonlinear finite element methods. Department of Aerospace Engineering Sciences and Center for Space Structures and

Controls University of Colorado Boulder, Colorado 80309-0429, USA, August 2001.

[42] C. A. Felippa, K. C. Park, and C. Farhat. Partitioned analysis of coupled mechanical systems. *Comput. Methods Appl. Mech. Engrg.*, 190:3247–3270, 2001.

[43] Y.T. Feng, D. Peric, and D.R.J. Owen. Determination of travel directions in path-following methods. *Int. J. Math. Comput. Modell.*, 21(7):43–59, 1995.

[44] Y.T. Feng, D. Peric, and D.R.J. Owen. A new criterion for determination of initial loading parameter in arc-length methods. *Computers and Structures*, 58(3):479–485, 1995.

[45] Y.T. Feng, D. Peric, and D.R.J. Owen. On the sign of the determinant of the structural stiffness matrix for determination of loading increment in arc-length algorithms. *Commun. Numer. Meth. Engng*, 13:47–49, 1997.

[46] I. Flood, L. Muszynski, and S. Nandy. Rapid analysis of externally reinforced concrete beams using neural networks. *Computers and Structures*, 79:1553–1559, 2001.

[47] S. J. Foster, B. Budiono, and R. I. Gilbert. Rotating crack finite element model for reinforced concrete structures. *Computers and Structures*, 58(1):43–50, 1996.

[48] Galileo Galilei. *Discorsi e dimostrazioni matematiche intorno a due nuove scienze. English: Dialogues concerning two new sciences.* New York : Dover Publications London, 1952. Translated by Henry Crew and Alfonso de Salvio with an introduction by Antonio Favaro.

[49] S. Ghosh, Y. Ling, B. Majumdar, and R. Kim. Interfacial debonding analysis in multiple fiber reinforced composites. *Mechanics of Materials*, 32:561–591, 2000.

[50] A. J. Gil Ruiz. *Structural analysis of prestressed Saint-Venant Kirchhoff hyperelastic membranes.* PhD thesis, School of Engineering. Universiy of Wales-Swansea, May 2004.

[51] A. Gravouil and A. Combescure. Multi-time-step explicit-implicit method for nonlinear structural dynamics. *Int. J. Numer. Meth. Engng.*, 50:199–225, 2001.

[52] A. Gravouil and A. Combescure. Multi-time-step and two-scale domain decomposition method for non-linear structural dynamics. *Int. J. Numer. Meth. Engng.*, 58:1545–1569, 2003.

[53] K. Han, D.R.J. Owen, and D. Peric. Combined finite-discrete and explicit-implicit simulations of peen forming process. *Engineering Computations*, 19(1):92–118, 2002.

[54] J. M. Hawkes and R. H. Evans. Bond stresses in reinforced concrete columns and beams. *The Structural Engineer*, XXIX(12):323–327, 1951.

[55] R. C. Hibbeler. *Structural Analysis.* Upper Saddle River: Prentice Hall, New Jersey, 4th edition, 1999.

[56] D. Hitchings, editor. *A Finite Element Dynamics Premier.* NAFEMS, Birniehill, East Kilbride, Glasgow G75 0QU, 1992.

[57] G. A. Holzapfel. *Nonlinear Solid Mechanics. A Continuum Approach for Engineering.* Wiley, 2000.

[58] T. J. R. Hughes. *The Finite Element Method.Linear Static and Dynamic Finite Element Analysis.* Dover Publications, Inc., 2nd edition, 2000.

[59] T. J. R. Hughes and W. K. Liu. Implicit-explicit finite elements in transient analysis: implementation and numerical examples. *Journal of Applied Mechanics*, 45:375–378, 1978.

[60] T. J. R. Hughes, K. S. Pister, and R. L. Taylor. Implicit-explicit finite elements in nonlinear transient analysis. *Comput. Methods Appl. Mech. Engrg.*, 17/18(1):159–182, 1979.

[61] B. Irons and S. Ahmad. *Techniques of Finite Elements.* Ellis Horwood Ltd., 1980.

[62] B. M. Irons and C. Treharne. A bound theorem for eigenvalues and its practical application. *Proc. 3rd Conference of Matrix Methods in Structural Mechanics*, AFFDL-TR-71-160:245–254, 1972. Wright-Patterson Air Force Base, Ohio.

[63] B-W Jo, Y-H Shon, and Y-J Kim. The evaluation of elastic modulus for steel fiber reinforced concrete. *Russian Journal of non-destructive testing*, 37(2):152–161, 2001.

[64] P. A. Klerck. *The Finite Element Modelling of Discrete Fracture in Quasi-brittle Materials.* PhD thesis, University of Wales-Swansea, 2000.

[65] D. Kracjcinovic. Damage mechanics. *Mechanics of Materials*, 8:117–197, 1989.

[66] T. Krauthammer and M. Zineddin. Structural concrete slabs under localized impact. technical report, The Pennsylvania State University, USA, 1998.

[67] P. Krysl and Z. Bittnar. Parallel explicit finite element solid dynamics with domain decomposition and message passing: dual partitioning scalability. *Computers and Structures*, 79:345–360, 2001.

[68] H. G. Kwak and F. C. Filippou. Nonlinear fe analysis of r/c structures under monotonic loads. *Computers and Structures*, 65(1):1–16, 1997.

[69] T. A. Laursen and J. C. Simo. A continuum-based finite element formulation for the implicit solution of multibody, large deformation frictional contact problems. *Int. J. Num. Meth. Engng.*, 36(20):3451–3485, 1993.

[70] J. Lemaitre and J-L. Chaboche. *Mechanics of Solid Materials.* Cambridge University Press, 2nd edition, 1994.

[71] P. E. Lewis and J. D. Ward. *The Finite Element Method (Principles and applications).* Adisson Wesley Publishing Comp., 1991.

[72] Love. *The Mathematical Theory of Elasticity.* Cambridge University Press, 4th edition, 1927.

[73] D. Maciel and H. B. Coda. Positional description for nonlinear 2-d static and dynamic frame analysis by fem with reissner kinematics. In K-J. Bathe, editor, *Third MIT Conference on Computational Fluid and Solid Mechanics*, pages 352–356, Cambridge, MA (USA), 14-17 June 2005. MIT, Elsevier.

[74] L. E. Malvern. *Introduction to the Mechanics of a Continuous Media.* Prentice-Hall, 1969.

[75] I. M. May, C. T. Morley, and Y. Y. Chen. Benchmarking of non-linear finite element analyses for reinforced concrete deep beams. In N. Bicanic, R. de Borst, H. Mang, and G. Meschke, editors, *Proc. Euro-C, Conf. Computational Modelling of Concrete Structures*, Mayrhofen, Tyrol, Austria, 2003. A A Balkema Publishers. This paper was also published ,in amended version, in *The Structural Engineer* magazine (UK), 1 February 2005.

[76] C. Michler, E. H. van Brummelen, S. J. Hulshoff, and R. de Borst. The relevance of conservation for stability and accuracy of numerical methods for fluid-structure interaction. *Comput. Methods Appl. Mech. Engrg.*, 192:4195–4215, 2003.

[77] A. Munjiza, D. R. J. Owen, and N. Bicanic. A combined finite-discrete element method in transient dynamics of fracturing solid. *Engineering Computations*, 12:145–174, 1995.

[78] A. Munjiza, D. R. J. Owen, and A. J. L. Crook. An $\mathbf{M}(\mathbf{M}^{-1}\mathbf{K})^{\mathbf{m}}$ proportional damping in explicit integration of dynamic structural systems. *Int. J. Num. Meth. Engng.*, 41(7):1277–1296, 1998.

[79] M. O. Neal and T. Belytschko. Explicit-explicit subcycling with non-integer time step ratios for structural dynamic systems. *Computers and Structures*, 31(6):871–880, 1989.

[80] W. Noll and E. G. Virga. Fit regions and functions with bounded variation. *Arch. Rational Mech. Anal.*, 102:1–21, 1998.

[81] R. W. Ogden. Large deformation isotropic elasticity - on the correlation of theory and experiment for incompressible rubberlike solids. *Proc. R. Soc. Lond. A.*, 326:565–584, 1972.

[82] C. O'Sullivan and J. D. Bray. Selecting a suitable time step for discrete element simulations that use the central difference time integration scheme. *Engineering Computations*, 21(2/3/4):278–303, 2004.

[83] E. Oñate. *Cálculo de Estructuras por el Método de los Elementos Finitos. Análisis estático lineal.* Centro Internacional de Métodos Numéricos en Ingenieria (CIMNE), Barcelona, Spain, 1995. In Spanish.

[84] E. Oñate and F. G. Flores. Advances in the formulation of the rotation-free basic shell triangle. *Comput. Methods Appl. Mech. Engrg.*, 194:2406–2443, 2005.

[85] D. R. J. Owen and E. Hinton. *Finite Elements in Plasticity.* Pineridge Press Ltd., Swansea, 1980.

[86] D.R.J. Owen and J.L. Curiel Sosa. Private communication. Discussions about this work, November 2002.

[87] J. Pamin and R. de Borst. Simulation of crack spacing using a reinforced concrete model with an internal length parameter. *Archive of Applied Mechanics*, 68:613–625, 1998.

[88] M. Papadrakakis. *Solving Large Scale Problems in Mechanics.* Wiley, 1993.

[89] F. Paris Carballo. *Teoria de la Elasticidad.* Escuela Tecnica Superior de Ingenieros Industriales. Universidad de Sevilla, Sevilla, Spain, 1996. In Spanish.

[90] K. C. Park. Practical aspects of numerical time integration. *Computers and Structures*, 7:343–353, 1977.

[91] R. Perera. *Introducción al método de Elementos Finitos.* CIMNE, Barcelona, Spain, 2004. In Spanish.

[92] D. Peric. On a class of constitutive equations in viscoplasticity: formulation and computational issues. *Int. J. Num. Meth. Engng.*, 36(8):1365–1393, 1993.

[93] D. Peric and D R. J. Owen. Computational model for 3-d contact problems with friction based on the penalty method. *Int. J. Num. Meth. Engng.*, 35(6):1289–1309, 1992.

[94] N. Petrinic. *Aspects of Discrete Element Modelling Involving Facet-to-facet Contact Detection and Interaction.* PhD thesis, University of Wales-Swansea, 1996.

[95] Y-L. Pin and N. S. Trahair. Nonlinear buckling and postbuckling of elastic arches. *Engineering Structures*, 20(7):571–579, 1998.

[96] S. Piperno. Explicit/implicit fluid/structure staggered procedures with a structural predictor and fluid subcycling for 2d inviscid aeroelastic simulations. *Int. J. Num. Meth. Engng.*, 25:1207–1226, 1997.

[97] M. V. K. V. Prasad and C. S. Krishnamoorthy. Computational model for discrete growth in plain and reinforced concrete. *Comput. Methods Appl. Mech. Engrg.*, 191:2699–2725, 2002.

[98] M. Richter, I. Lepenies, and B. W. Zastrau. On the application of a slip-based interface model for textile reinforced concrete. *PAMM Proc. Appl. Math. Mech.*, 3:228–229, 2003.

[99] D. Roberts. *Finite Element Modelling of Rockbolts and Reinforcing Elements.* PhD thesis, University of Wales Swansea, Swansea, 1999.

[100] J. G. Rots. *Computational Modeling of Concrete Fracture.* PhD thesis, Delft University of Technology, The Netherlands, Delft, 1988.

[101] M. R. Salari and E. Spacone. Finite element formulations of one-dimensional elements with bond-slip. *Engineering Structures*, 23:815–826, 2001.

[102] H. Schafer. A contribution to the solution of contact problems with the aid of bond elements. *Comput. Methods Appl. Mech. Engrg.*, 6(3):335–354, 1975.

[103] J. Sena Cruz and J. Barros. Modeling of bond between near-surface mounted cfrp laminate strips and concrete. *Computers and Structures*, 82:1513–1521, 2004.

[104] P. Sharifi and E. P. Popov. Nonlinear buckling of sandwich arches. *Proc. ASCE J. Eng. Mech. Div.*, 97(EM5):1397–1412, October 1971.

[105] J. C. Simo and F. Armero. Geometrically non-linear enhanced strain mixed methods and the method of incompatible modes. *Int. J. Num. Meth. Engng.*, 33(7):1413–1449, 1992.

[106] J. C. Simo and T. J. R. Hughes. *Computational Inelasticity.* Springer-Verlag, New York, 1998.

[107] J. C. Simo and R. L. Taylor. Return mapping algorithm for plane stress elastoplasticity. *Int. J. Num. Meth. Engng.*, 22:649–670, 1986.

[108] L. J. Sluys and R. de Borst. Failure in plain and reinforced concrete-an analysis of crack width and crack spacing. *Int. J. Solids Structures*, 33(20-22):3267–3276, 1996.

[109] I. M. Smith. *Programming in FORTRAN 90.* Wiley, Chichester, 1994.

[110] I. M. Smith and D. V. Griffiths. *Programming the Finite Element Method.* Wiley, 3rd edition, 1998.

[111] P. Smolinski. Subcycling integration with non-integer time steps for structural dynamics problems. *Computers and Structures*, 59(2):273–281, 1996.

[112] P. Smolinski and Y-S. Wu. Stability of explicit subcycling time integration with linear interpolation for first-order finite element semidiscretizations. *Comput. Methods Appl. Mech. Engrg.*, 151:311–324, 1998.

[113] E. A. de Souza Neto. *Aspects of continuum modelling and numerical simulation of internal and surface damage in finitely deformed solids.* PhD thesis, University of Wales Swansea, Singleton Park, Swansea, 1994.

[114] E. A. de Souza Neto. A fast, one-equation integration algorithm for the lemaitre ductile damage model. *Commun. numer. meth. Engng.*, 18:541–554, 2002.

[115] E. A. de Souza Neto and Y. T. Feng. On the determination of the path direction for arc-length methods in the presence of bifurcations and 'snap-backs'. *Comput. Methods Appl. Mech. Engrg.*, 179:81–89, 1999.

[116] E. A. de Souza Neto, D. Peric, and D. R. J. Owen. Computational methods for plasticity: Theory and application. In press, 2006.

[117] M. A. Tasdemir, C. Tasdemir, S. Akyüz, A. D. Jefferson, F. D. Lydon, and B. I. G. Barr. Evaluation of strains at peak stresses in concrete: a three-phase composite model approach. *Cement and Concrete Composites*, 20:301–318, 1998.

[118] A. A. Tasnimi. Mathematical model for complete stress-strain curve prediction of normal, light weight and high-strength concretes. *Magazine of Concrete Research*, 56(1):23–34, February 2004.

[119] R. Tepfers. Cracking of concrete cover along anchored deformed reinforcing bars. *Magaz. Concr. Research*, 106:3–12, 1979.

[120] M. Touchaei. *Combined finite/discrete element analysis of impact loading of composite shells.* PhD thesis, University of Wales Swansea, Swansea, 1998.

[121] C. Truesdell. *Rational Thermodynamics.* McGraw-Hill, 1969.

[122] C. Truesdell. *A first Course in Rational Continuum Mechanics.* Academic Press, New York, first edition, 1977.

[123] T. Ueda and J. Dai. Interface bond between frp sheets and concrete substrates: properties, numerical modeling and roles in member behaviour. *Prog. Struct. Engng Mater.*, 7:27–43, 2005.

[124] X. Wang and X. Liu. A strain softening model for steel-concrete bond. *Cement and Concrete Research*, 33:1669–1673, 2003.

[125] J. H. Weathersby. *Investigation of bond-slip between concrete and steel reinforcement under dynamic loading conditions.* PhD thesis, Louisiana State University, USA, May 2003.

[126] R. K. Wen and B. Suhendro. Nonlinear curved-beam element for arch structures. *J. Struct. Engng. ASCE*, 117(11):3496–3515, 1991.

[127] W. L. Wood. *Practical Time-Stepping Schemes.* Clarendon Press, Oxford, 1990.

[128] L. Yan, C.H. Jenkinsb, and R.L. Pendleton. Polyolefin fiber-reinforced concrete composites part ii. damping and interface debonding. *Cement and Concrete Research*, 30:403–410, 2000.

[129] Z. Yaqun, L. Zhongqin, and Z. Weigang. A quadrilateral thin shell element based on area co-ordinate for explicit dynamic analysis. *Commun. Numer. Meth. Engng.*, 19:169–178, 2003.

[130] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method: Solid Mechanics*, volume 2. Butterworth-Heinemann, Oxford, 5th edition, 2000.

[131] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method: The Basis*, volume 1. Butterworth-Heinemann, Oxford, 5th edition, 2000.

Index