ARBITRARY LAGRANGIAN-EULERIAN METHOD AND ITS APPLICATION IN SOLID MECHANICS

By

Jin Wang

B. A. Sc., M. A. Sc. University of Science and Technology Beijing

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE FACULTY OF GRADUATE STUDIES

DEPARTMENT OF MECHANICAL ENGINEERING

We accept this thesis as conforming

to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

April 1998

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Mechanical Engineering

The University of British Columbia
Vancouver, Canada

Date April 16, 1998

Abstract

The applicability and accuracy of existing finite element formulation methods for finite strain deformation and metal forming problems are investigated. It is shown that the existing formulation methods, both Lagrangian and Eulerian type, are not suitable nor efficient for large deformation problems especially when boundary conditions change during deformation, as is the case in most metal forming problems. This creates a need for a more general and efficient type of formulation. An Arbitrary Lagrangian-Eulerian (ALE) method is presented for the general application in solid mechanics and large deformation problems. A consistent ALE formulation is developed from the virtual work equation transformed to an arbitrary computational reference configuration. The formulation presents a general approach to ALE method in solid mechanics applications. It includes load correction terms and it is suitable for both rate-dependent as well as rate-independent material constitutive laws. The proposed formulation reduces to both updated Lagrangian and Eulerian formulations as special cases. The formulation is presented in a form that makes the programming an extension to existing Lagrangian and Eulerian type programs.

An efficient mesh motion scheme for ALE formulation is developed with a procedure for handling boundary motion within the scheme, which can ensure homogeneous mesh results. A practical and more efficient numerical method is presented to handle supplementary constraint equations on element level rather than on the global level. Different numerical algorithms for the integration of the rate type constitutive equation are investigated and coupled with the return mapping algorithm to provide plastic incremental consistency. A numerical procedure for stress integration is developed based on the physical meaning of stress. Jaumann and Truesdell rates are taken as the objective stress rates in the constitutive equation. An alternative numerical treatment for rate of deformation tensor ${}^{t}D_{ij}$ is presented to maintain incremental objectivity of the tensor. It is shown by numerical examples that the use of Truesdell stress rate with a developed numerical integration procedure gives consistently more accurate results than other procedures presented. An algorithm for updating material associated properties is presented and applied in simulation of various metal forming problems.

A 2D finite element program, ALEFE, based on the presented formulation is developed and tested. The program may reduce to an updated Lagrangian or Eulerian methods as special cases. The mesh motion for the whole domain is controlled by the motion of the boundary nodes. The program can handle unsymmetric stiffness matrices and coupled displacements/velocity boundary conditions. The input data is designed to be similar to available commercial finite element codes, so that the data generation phase may be directly imported from these programs. The output data format is designed to be compatible with general graphic simulation and data processing commercial softwares, so that contour, x-y and deformed mesh plots may be easily created from the output data of ALEFE.

Various benchmark and practical problems are simulated by the developed program. Practical simulation cases include flat punch forging process, sheet metal extrusion process, necking bifurcation of a bar in tension, a steady strip rolling and compression between wedge-shaped dies. Numerical results are compared with analytical solutions or experimental results available in literature.

Table of Contents

List of Tables

List of Figures

Nomenclature

General Variables:

a_I, b_I	coefficients in relation of material motion and mesh motion
A_e	area of an element
b	dimension of a square element
c_i	coordinates of a point P
c_i^0	old coordinates of a point P
C_{ijkl}	material constitutive tensor
d_p	punch displacement
D_{ij}	components of rate of deformation tensor
D_{ij}^p	plastic part of D_{ij}
${}_{t+\Delta t}e_{ij}$	components of deformation tensor
E	Young's modulus
f	a general function or a physical quantity
f^c	the value of f at computational referential point
f_i^B	components of body forces
f_i^S	components of surface tractions
f_α	unsmoothed quantity at node α
$\overline{f}_\alpha$	smoothed property at node α
F	external force
$\mathbf{F}, F_{ij}$	deformation gradient tensor
$\mathbf{G}, G_{ij}$	the gradient of displacement from time t to $t + \Delta t$
h	height of the specimen in punch indentation

h_{iI}	values of shape function for displacements in i direction at D.O.F. I
$\overline{h}_\alpha$	smoothing shape function at node α
H	hardening modulus
i	index for directions / coordinate axes ($i = 1, 2, 3$ or x, y, z)
I	index for D.O.F.
$\mathbf{I}$	unit matrix
J	ratio of material volume to the volume in referential state
J^c	Jacobian matrix
k, k_0	current and initial shear yield stresses, respectively
$\mathbf{K}, k_{IJ}$	stiffness matrix related to ${}^t\mathbf{v}$
$\mathbf{K}^c, k_{IJ}^c$	stiffness matrix related to ${}^t\mathbf{v}^c$
$\overline{\mathbf{K}}$	equivalent stiffness matrix
L	length of the elastic-plastic bar
m	modification term for D_{ij}
n	number of D.O.F in an element
n_n	number nodes per element
n_i	components of the unit vector normal to boundary surface
N	total number of D.O.F in the model
N_s	number of sampling points in an element
p	normal pressure
p_I	external nodal force at D.O.F. I
$p_I^{(k)}$	external nodal force in k^{th} iteration at D.O.F. I
$\overline{\mathbf{p}}$	equivalent load vector
q	distributed tensile force
q_I	equivalent nodal force at D.O.F. I

$q_I^{(k)}$	equivalent nodal force in k^{th} iteration at D.O.F. I
r	normalised parametric coordinate $0 \leq r \leq 1$
r_I	residual nodal force at D.O.F. I
$r_I^{(k)}$	residual nodal force in k^{th} at D.O.F. I
$\mathbf{R}, R_{ij}$	rotation tensor
s	normalised parametric coordinate $0 \leq s \leq 1$
S	surface of deformed body on which tractions are prescribed
S_{ij}	2^{nd} Piola Kirchhoff stress
$S_{ij}^{(I)}$	2^{nd} Piola Kirchhoff stress at an intermediate state
t	time
t_d, t_f, t_e	displacement, force and energy tolerances, respectively
$\mathbf{T}$	transformation matrix
U_r	total radial displacement
U_x	total displacement in x direction
u_i	displacement in i direction
u_I	material displacement at D.O.F. I from time t to $t + \Delta t$
u_I^c	mesh point displacement at D.O.F. I from time t to $t + \Delta t$
v	velocity of punch or die
$\mathbf{v}$	material particle velocity vector
v_i	material velocity in direction i
v_I	material velocity at D.O.F. I
$\mathbf{v}^c$	computational reference velocity vector
v_i^c	mesh velocity in direction i
$v_{i(\alpha)}$	material velocity of node α at direction i in global coordinates
$v'_{i(\alpha)}$	material velocity of node α at direction i in local coordinates

v_I^c	mesh velocity at D.O.F. I
$\mathbf{v}'$	material particle velocity vector in local coordinate system
${}^tV$	volume of deformed body at time t
w	diameter of the circular punch in punch indentation
W	diameter of the circular specimen in punch indentation
$\mathbf{W}, W_{ij}$	spin tensor
$\mathbf{x}$	position vector of material coordinate
$\mathbf{x}^*$	position vector of imaginary material coordinate
x_i	material particle coordinate in axis i
α	coefficient in implicit integration algorithm
β	contact angle in rolling
δ_{ij}	Kronecker delta
Δt	time increment
$\Delta u_I^{(k)}$	displacement increment in k^{th} iteration at D.O.F. I
$\Delta_{\mathbf{x}} f$	change of physical quantity due to material motion
$\Delta \sigma_{ij}$	increment of Cauchy stress
$\Delta_{\chi} f$	change of physical quantity due to computational reference system motion
θ	angle for coupled displacements
η	tool angle for wedge shaped dies
ε_{eq}^p	equivalent plastic strain,
$\xi_i(r, s)$	boundary velocity of a geometric domain
σ_{ij}	components of Cauchy stress tensor
$\sigma_1, \sigma_2, \sigma_3$	the principal Cauchy stresses
$\sigma_{yield}, \sigma_{yield0}$	current and initial yield stresses, respectively
$\sigma_{ij}^{\cdot J}$	Jaumann rate of Cauchy stresses

$\sigma_{ij}^{\cdot T}$	Truesdell rate of Cauchy stresses
σ'_{ij}	deviatoric Cauchy stress tensor
$\sigma_{ij}^{(I)}$	Cauchy stresses at an intermediate state I
$\sigma_{eq}^{(I)}$	equivalent stresses at an intermediate state I
τ	friction stress
$\tau^{\cdot J}$	Jaumann rate of Kirchoff stress
μ	friction coefficient
ν	Poisson's ratio
ϕ	objective function in the least square method
$\phi_i(r,s)$	boundary coordinate of a geometric domain
$\boldsymbol{\chi}$	position vector of referential coordinate
χ_i	mesh point coordinate in axis i
ω	angular velocity
∇f	gradient of f

Left Superscripts:

t	the quantity is the value at time t
$t + \Delta t$	the quantity is the value at time $t + \Delta t$

Left Subscripts:

t	the quantity is with respect to the configuration at time t

Superimposed Symbols:

$\cdot$	material time rate of a quantity
$\wedge$	computational reference time rate of a quantity
T	transformation of a matrix

Acknowledgement

I would like to express my deep appreciation to my supervisor, Dr. Mohamed S. Gadala, for his invaluable help, judicious advice and understanding throughout my graduate studies. His patient guidance and continuous encouragement effectively helped me to proceed with this work.

I express my gratitude and appreciation to Dr. Henry Vaughan for his invaluable suggestions and advice. I would also like to thank Dr. Stanley G. Hutton and Dr. Reza Vaziri for their help and valuable discussions and suggestions at various stages of this work.

I acknowledge the financial support of Natural Science and Engineering Research Council of Canada, of UBC-UGF award and, finally, the financial support from my supervisor.

Special thanks are also due to my wife Chunmei and my parents who offered incessant moral support and encouragement. To them and to my new born son, Roy, I dedicate this thesis.

Chapter 1

INTRODUCTION

1.1 NONLINEAR FINITE ELEMENT METHODS

Although the application of finite element methods to many nonlinear problems has been successfully carried out, there are numerous areas of concern and investigation in that regard. One such area, which we consider here, deals with the solution of large strain plasticity and metal forming problems with pseudo type boundary nonlinearities. A number of critical difficulties arise in the finite element analysis of such problems. Among these difficulties are; proper formulation, mesh distortion, proper modelling of the contact boundary conditions, incorporation of the plastic incompressibility condition and accounting for plastic anisotropy.

Foundations of large strain analysis of elastic-plastic solids may be traced back to the early work of Hill [1]. It took some time, however, until Hibbitt, Marcal and Rice [2] introduced the first finite element formulation for large strain problems. In their approach they used a total Lagrangian formulation (TLF). Later on, McMeeking and Rice [3] pioneered the use of updated Lagrangian formulation (ULF) in the same area of applications. The two formulation methods have been widely used for both steady and non steady static large plastic strain problems. On the other hand, Eulerian formulation (EF) has been initially introduced for finite element applications in the fluid mechanics area. Several trials aiming at adapting this formulation to large strain and metal forming problems were attempted [4, 5, 6, 7]. Owing to the difficulties in obtaining material time derivatives in spatial reference frame, no generally accepted Eulerian formulation is available for such problems. Also, since the mesh is spatially fixed in EF, it is not easy to simulate non-steady static or dynamic behavior. Trials have been made to relate the moving material points (in terms of FE Gauss points) to the fixed spatial mesh [6, 7]. Much work is still required, however, to refine and establish this approach.

This chapter investigates the applicability and accuracy of existing formulation methods to the finite strain deformation problems. The objective is to examine some of the above aspects through the use of available FE commercial programs. The programs used in the analysis are DEFORM2 [8], ANSYS [9] and NISA [10]. It is required to provide an assessment for the existing formulation of such problems in order to propose required modifications or new formulation. The applications considered in this regard are flat punch indentation, plane strain extrusion and forging problems.

1.2 LAGRANGIAN FORMULATIONS

1.2.1 Sample Examples

(i) Flat Punch Indentation

Flat punch indentation into a slab is mathematically the same as the compression of a specimen between two flat parallel punches[11]. In the axisymmetric compression model, w and W are the diameters of a circular punch and solid circular specimen, respectively and h is the height of the specimen. The ratios $W/w = 2.7$ and $h/w = 1.7$ are used in the simulation. The material properties are Young's modulus $E = 6.9 \times 10^4\ MPa$, Poisson's ratio $\nu = 0.33$, hardening modulus $H = 138\ MPa$ with initial yield stress $\sigma_{yield0} = 90\ MPa$. Axi-symmetric 8-node isoparametric elements together with 2-node (node-to-node) gap elements are used in the model. Assuming a smooth punch, an indentation is made up to a punch displacement $d_p = 0.15w$. The simulation was carried out using both NISA and ANSYS programs. The mesh topology before and after deformation is presented in Figure 1.1. The radial displacement U_r, the second principle stress σ_2, and the equivalent plastic strain ε^p_{eq} are used for comparison of the results. The results obtained from the two programs showed close agreement in displacement and strains but revealed noticeable discrepancies when comparing stress values. Also, there is noticeable disagreement between the results from both programs and similar ones reported in the literature [11].

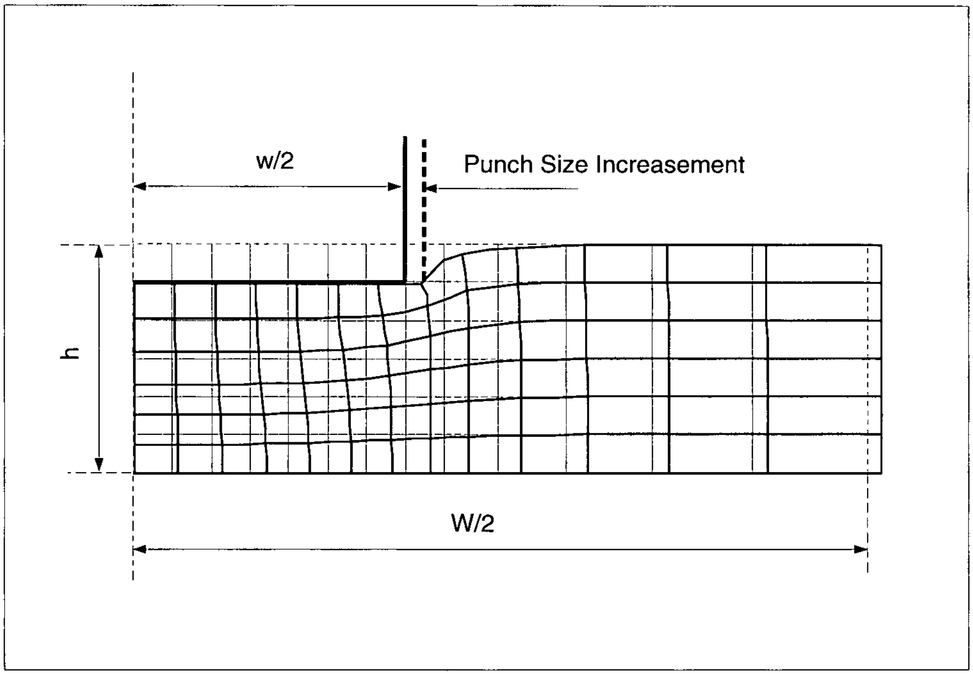


Figure 1.1: The meshes before and after deformation

(ii) Metal Extrusion

Another problem analyzed with NISA and ANSYS is the steady state deformation process of plane-strain metal extrusion. Metal is forced into a symmetric die after sliding between smooth rigid plates. The die is approximated with a straight line and produces a 25% thickness reduction over a distance $1.2a$, where a is half the thickness of the original sheet, as shown by grey lines in Figure 1.2. The material properties are Young's modulus $E = 6.9 \times 10^4\ MPa$, Poisson's ratio $\nu = 0.3$, strain hardening $H = 1.1 \times 10^3\ MPa$ and initial yield stress $\sigma_{yield0} = 4.0 \times 10^2\ MPa$.

The metal extrusion problem is modelled by plane strain 8-node isoparametric elements and

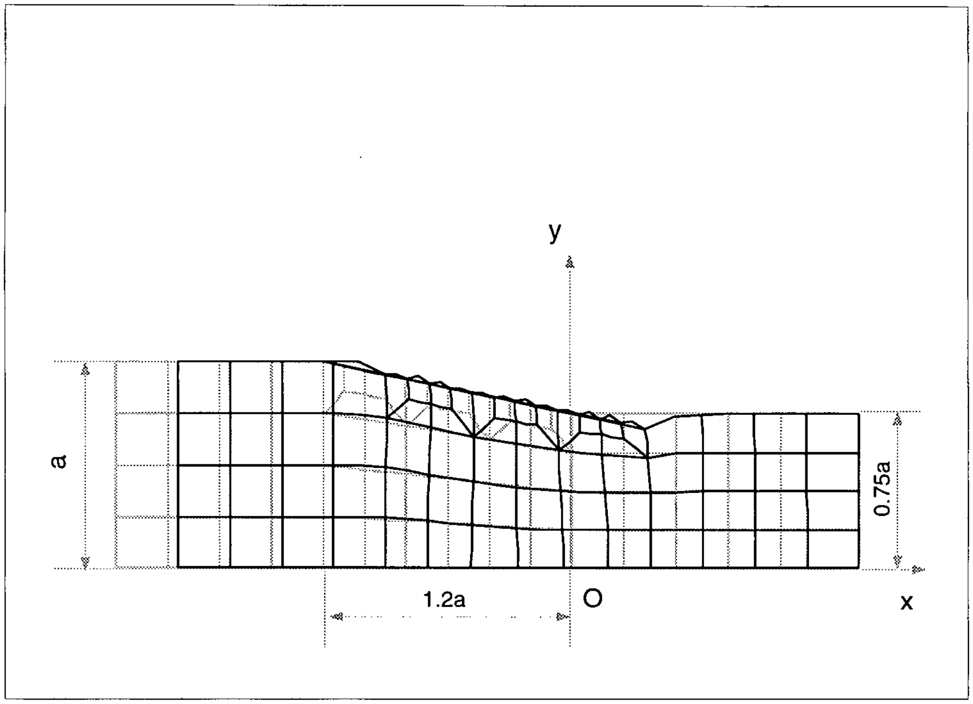


Figure 1.2: Not updated boundary condition

2-node gap elements. The nodal and deformed shape after the piston moves $0.3a$ are shown in Figure 1.2 by grey and black lines, respectively. The distributions of displacement U_x and stress σ_x, both in the extrusion direction, are examined. The distributions of U_x and σ_x from NISA and ANSYS are very similar and the values of U_x are close to each other with a difference of less than 0.4%. The values of σ_x are, however, quite different; and show a difference of up to 25%. Once again, the results obtained here are quite different from those reported in the literature [12].

For frictionless die-workpiece interface, the results with and without gap elements, in both NISA and ANSYS, are significantly different. As an example, distribution of σ_x on the outer surface of the work-piece is shown in Figure 1.3. It is not expected that the introduction of node-to-node gap elements should influence the frictionless die case. The obtained results show, however, a noticeable effect of adding gap elements to the frictionless die case.

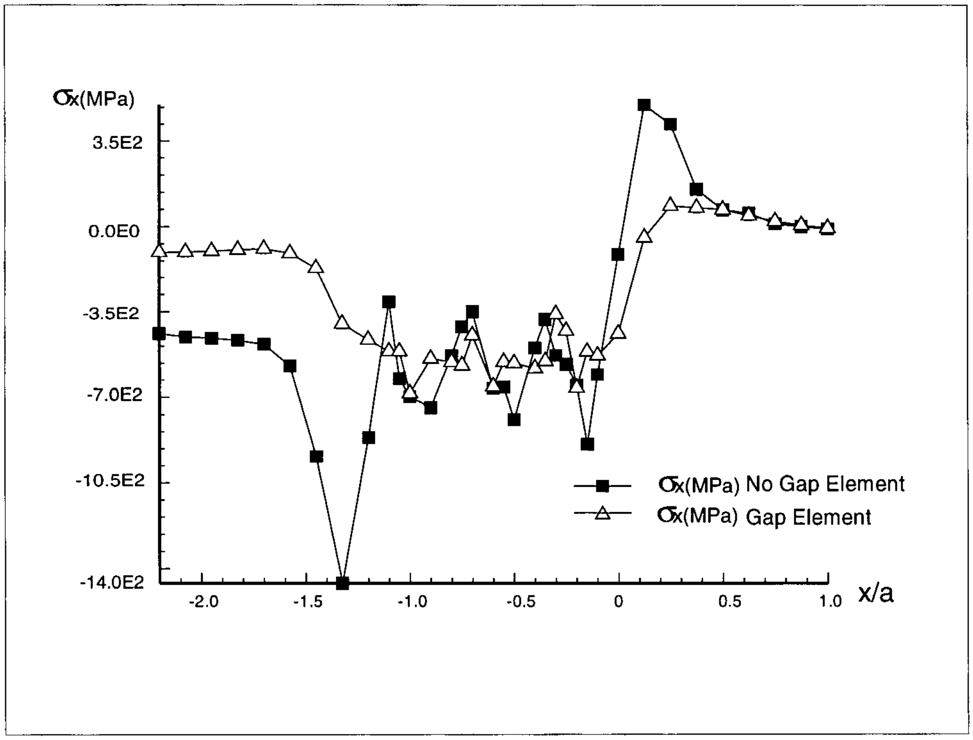


Figure 1.3: Distribution of stress σ_x on outer surface

In simulation of the punch indentation, once the material point under the punch corner moves out of the corner, it should be allowed to move up. However, the Lagrangian formulation can not update the boundary condition automatically. This is an important feature needed in handling most of metal forming problems. Also, it may not be feasible to manually update boundary conditions after each incremental step. This deficiency has the effect of increasing the punch size during the deformation, as shown in Figure 1.1. For metal extrusion, the situation is worse. Figure 1.2 shows a typical deformed mesh (superimposed on the original one) obtained from such programs. The reduction in thickness depicted by the program does not agree with the specified value because of the absence of geometric boundary updating in the program. Thus the predicted deformation is unacceptable.

(iii) Plane Strain Forging

Plane strain forging is also simulated by NISA and ANSYS. The material model is chosen to be an elasto-plastic one with Young's modulus $E = 200.0\ GPa$, Poisson's ratio $\nu = 0.30$, initial yield stress $\sigma_{yield0} = 250.0\ MPa$ and plastic modulus $H = 1.0\ GPa$. The body is deformed by a rigid frictionless tool with a prescribed vertical velocity. Only a quarter of the domain is studied because of symmetry, and plane strain condition is assumed. The finite element model used in NISA and ANSYS is shown in Figure 1.4. The analysis is performed for up to 60% reduction in the height of the original piece to show the effect of large deformation.

Figure 1.5 shows the results from ANSYS [9] and NISA [10]. Because the boundary conditions are not updated automatically, the tool or punch width will be actually increasing with the deformation process as in the indentation example. It is also noted that elements at the punch corner are highly distorted because of large deformation. Increasing the deformation level causes more severe element distortions and the program terminates the analysis. The use of more advanced capabilities such as the point-to-surface gap element in ANSYS may eliminate some of the above problems such as the punch size one. These elements have their own

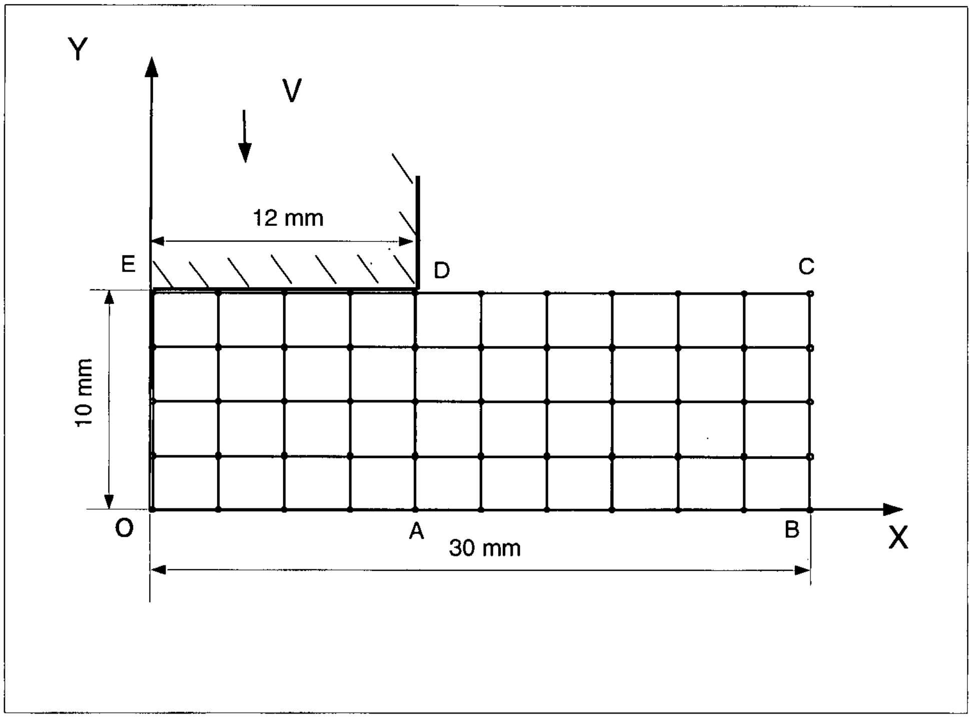


Figure 1.4: The finite element model

problems, however, such as convergence difficulties, user definition of stiffness parameters and violation of local equilibrium at the punch corner. For this particular example, contact elements were used on the tool work-piece interface and several trial cases with different reduction ratios, different tolerance limits and different stiffness parameters were performed. All cases tried ceased to give successful convergence beyond a reduction of 30%.

The same process is simulated by DEFORM2 [8] program, which is one of the most widely used metal forming programs because of its capability of automatic remeshing and updating contact boundary conditions. The finite element model used in DEFORM2 is shown in Figure 1.6. A much denser mesh was required to achieve convergence for the same deformation level obtained in NISA and ANSYS. Figure 1.7 shows the deformed shape from DEFORM2. Although the punch size increment is eliminated by remeshing, it may be seen that the mesh around the punch is highly distorted and the punch corner is "cut off". More importantly,

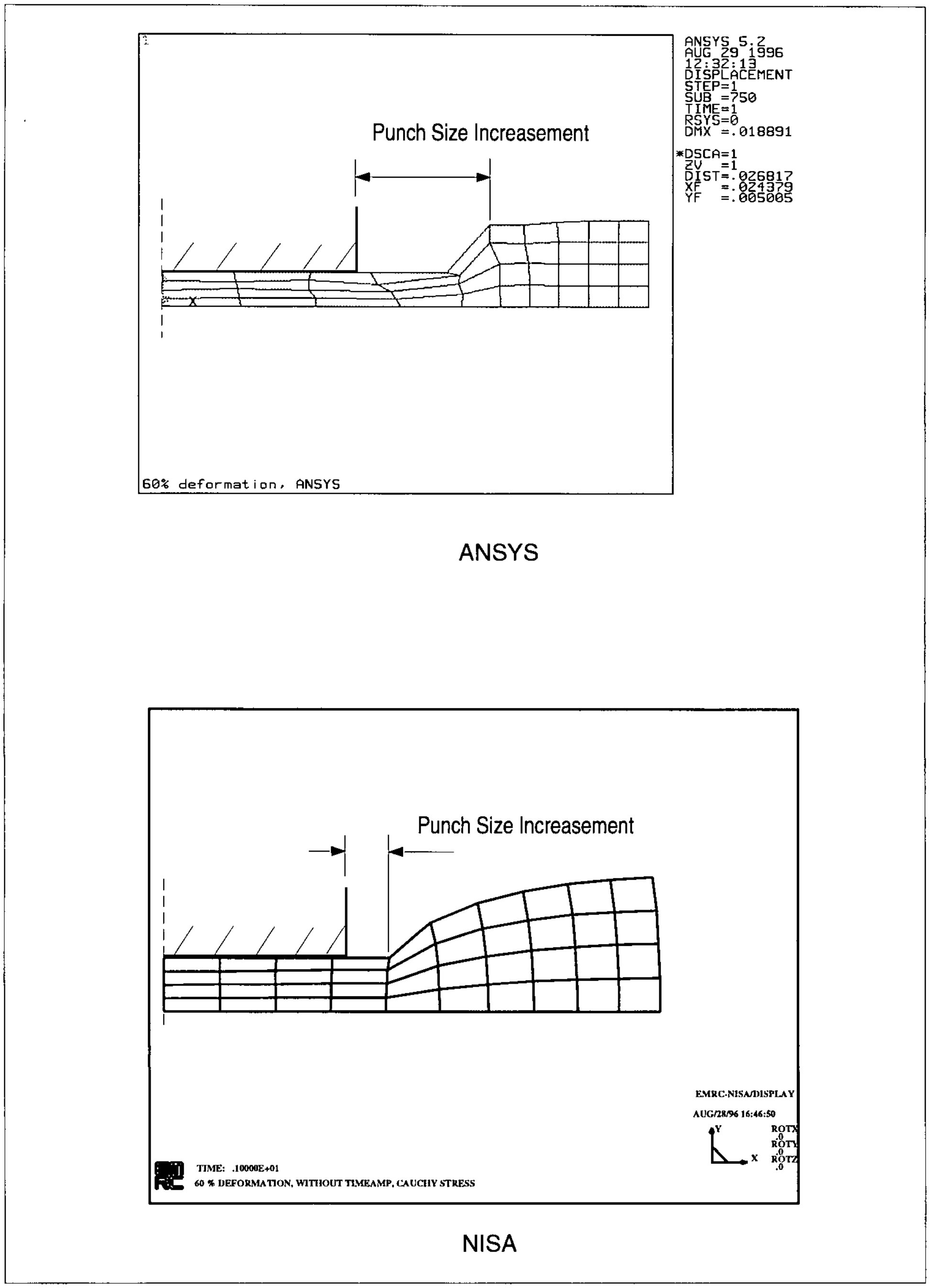


Figure 1.5: The deformed shapes by updated Lagrangian method

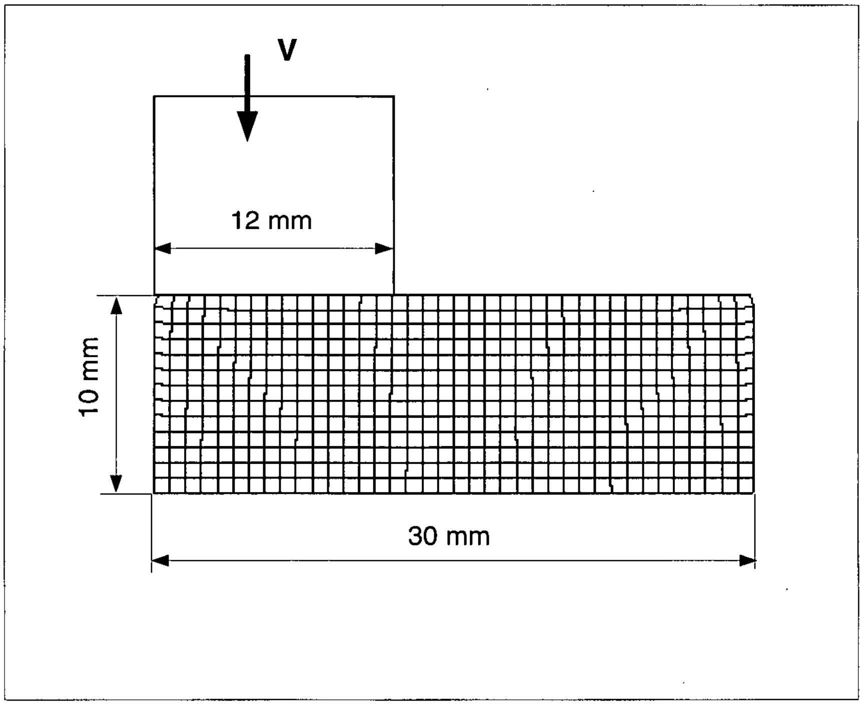


Figure 1.6: Finite element model used in DEFORM2 simulation

the predicted deformed boundary of the free surface near the punch corner is unusual and not rational. Another shortcoming is the fluctuations in loads predicted by Lagrangian method. Figure 1.8 shows a plot for the reduction of the specimen height versus the applied load, from DEFORM2. The result shows noticeable load fluctuation that is pertinent to the updated Lagrangian formulation. Similar fluctuation is observed in the plane strain metal extrusion example. The extrusion pressure increases steadily up to the state when the billet fills the die. As the billet exits the die, the interface nodes that are constrained to move on the die are released. This causes a drop in the strain energy of the billet which in turn causes a drop in the extrusion pressure. The studies performed by Lee [12] and Voyiadjis [13] confirm the existence of these fluctuations.

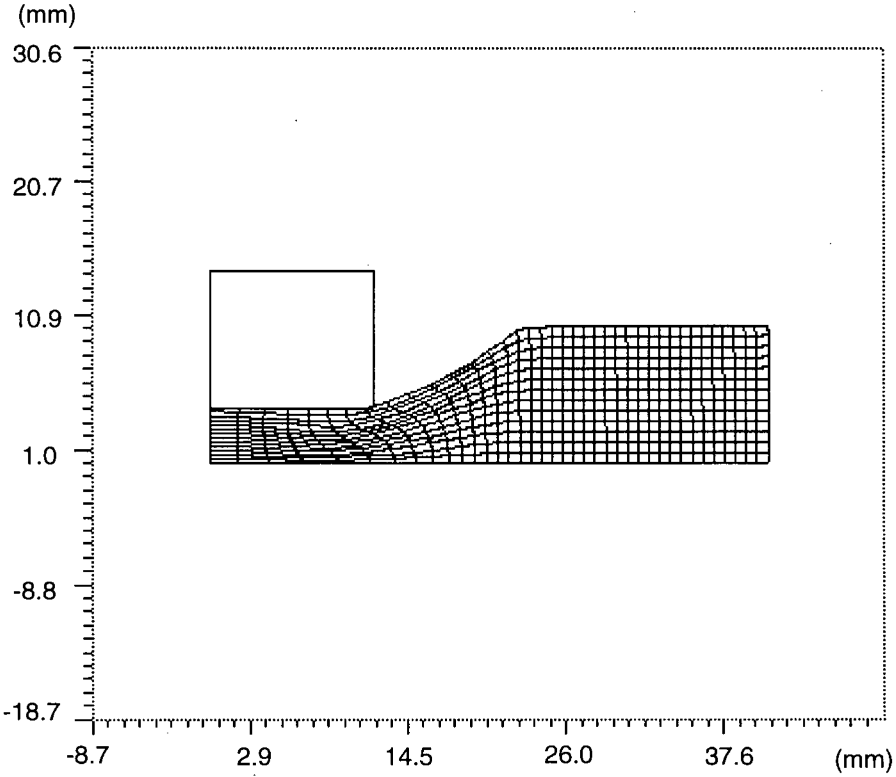


Figure 1.7: Deformed mesh from DEFORM2

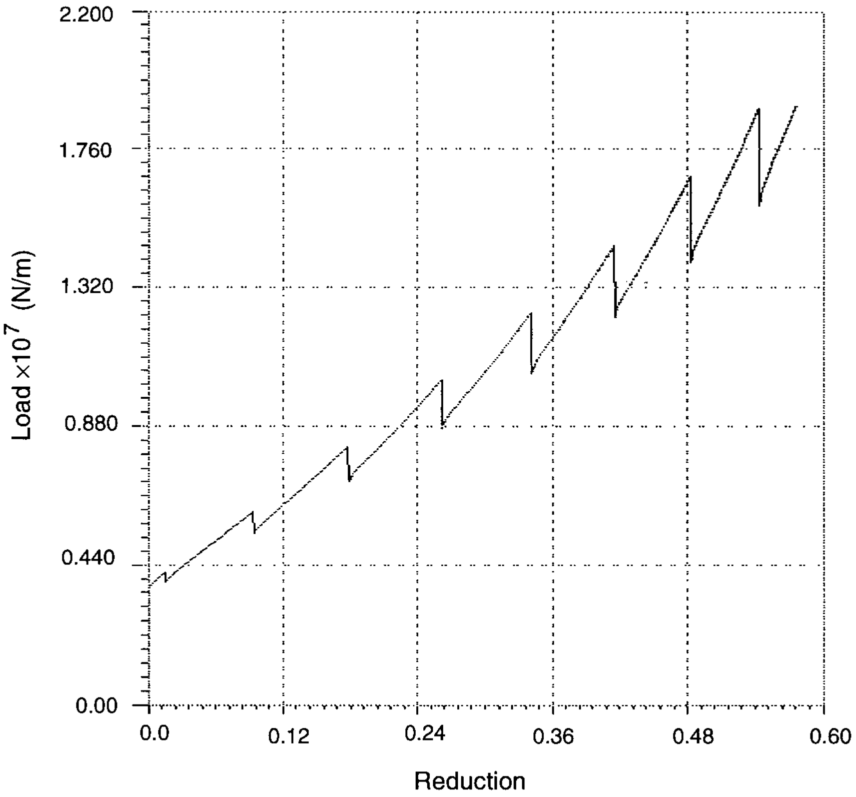


Figure 1.8: Load-height reduction result from DEFORM2

1.2.2 Limitation of Lagrangian Type Formulation

Most of the available commercial FE programs are based on Lagrangian type of formulation. Such formulations are efficient and quite suitable for handling nonlinear problems in which small strains prevail, boundary condition nonlinearities do not change with the course of deformation and where mesh distortion is not a critical factor in the analysis. Unfortunately, all of the above restrictions are violated in any metal forming process.

The above examples reveal some of the main shortcomings in the application of Lagrangian type of formulations to metal forming problems. It is shown that large plastic strains with contact boundary conditions are not suitable to be analyzed by such formulations. Contact boundary conditions with sharp edges or corners can not be applied precisely and load fluctuations are evident with large strains. The problem of mesh distortions and element entanglement pose a serious drawback on the use of such formulation. It is not always feasible to update the mesh manually, and even if it is possible to do so, it will involve major interaction and time involvement by the user of such schemes.

On the other hand, some automatic mesh rezoning methods are developed for Lagrangian finite element analyses of metal forming problems [14, 15, 16]. However, these methods are not so robust or efficient to remedy the mesh distortions. Furthermore, they do not eliminate load fluctuations. These remeshing methods are based on rediscretizing the deformed configuration after certain special deformation. Obviously, the determination of the special deformation is very important and can only be based on user's experience. The remesh has to be applied on the whole domain, which is actually not necessary. In general, these methods can not change the mesh topologies, so they may not improve the overall accuracy of the finite element calculations. The locations of nodes in the new mesh are calculated by algebraic interpolation method or by solving differential equations. The algebraic interpolation will introduce curve-fitting error, especially when the boundary shape is complicated. Solving differential equation can generate higher quality finite element meshes, but it is time consuming and, more significantly, may still produce poorly shaped meshes near boundaries in some cases [17].

The problems of contact and friction boundary conditions pose another serious concern in the solution of metal forming applications. Contact condition, which is defined from the geometrical compatibility on the contact surface or the impenetrability condition, has been introduced by the direct method, elements with special mechanical properties, the Lagrangian multiplier method and the penalty function method [18]. Since the contact area is a prior unknown, the boundary conditions of the contact problems are determined as part of the solution.

Although the contact surfaces obtained by the Lagrangian multiplier method satisfy the contact condition in the integral sense, additional unknowns are required and the total number of unknowns in the system equations increases. In addition, the associated tangent matrix may be indefinite and has zero diagonal entries that pose some difficulties in the solution steps [19]. For the penalty function method, the solution results satisfy the contact conditions only approximately. The accuracy of the approximate solution depends strongly on the penalty parameter, so the correct choice for these parameters is the essence of the algorithm. When the penalty parameter is chosen to be too large, it leads to numerical problems in the form of loss of accuracy in the solution. On the other hand, a too small choice, results in unacceptable penetration of one body into the other.

The node to node or node to surface gap or contact elements have been recently introduced in commercial FE packages [9, 10]. Such an approach is quite natural to the formulation of the FE method because the overall stiffness assembly process is unchanged; the contact elements can be treated as a separate material property group such as plasticity, creep, etc. In spite of their various advantages, these contact or gap elements exhibit several important shortcomings [20]. They undergo anomalous response behavior when employed in situations where large deformation kinematics are needed to generate closure, that is reflected in improper stiffness characterizations, i.e., poorly conditioned Jacobians. They are also awkward to apply in situations requiring friction effects and they require significant amount of equilibrium iterations. Furthermore, parameters of the contact element have to be chosen by the user, and the convergence speed and the accuracy of the solution are quite dependent on the choice of these parameters. Sometimes, especially when friction effect is incorporated, convergence may be difficult to achieve, as indicated by the experience from solving the plane strain forging problem.

1.3 EULERIAN FORMULATION

The main difference between the Eulerian formulation and Lagrangian ones is that the deformation of the material moving through a fixed region in space is determined as a function of the current position and the time t instead of determining the deformation of the material element by following its motion in space [6]. The independent variables in the Eulerian description are current position $\mathbf{x}$ of the body-point $\mathbf{X}$ and time t. For the material motion, $\mathbf{x}$ itself becomes dependent on the time t, which complicates material time derivatives and other relations when approached strictly by Eulerian, or spatial formulation. Although many authors have treated the Lagrangian and the updated Lagrangian formulation, there has been little effort concerning the development of a consistent Eulerian formulation in solid mechanics applications. Some developments, e.g. Gadala et al. [6], give the proper material time derivative in spatial description but original geometric parameters are included in the final equilibrium equation. Therefore, such approach may not be strictly considered as purely Eulerian formulation.

In the Eulerian method, it is necessary to properly determine the state of the material-associated properties (e.g., strain, stress) of the material points momentarily occupying the integration points at the beginning of each step. The method of updating such material-associated properties is not well developed and further investigations need to be considered. Abo-Elkhier [21] utilized an imaginary finite element mesh for updating, but no discussion nor proper assessment of the accuracy of the method is given. Derbalian [22] discussed a conjugate scheme for interpolation of the stress and applied the procedure to simulate steady state static metal extrusion. By constructing a set of conjugate shape functions (bi-orthogonal), a consistent approximation for the stress field may be obtained which is continuous across inter-element boundaries and involves less mean error. However, this method introduces a large system of equations which may be solved iteratively during each incremental step. Also, the conjugate method is originally proposed for improving the accuracy of incremental stress fields, rather than the total stress [23].

Eulerian formulation is generally more suitable for the study of flow problems in a fixed region of known shape. Therefore, the analysis of steady state metal forming processes may be achieved by Eulerian formulation. Because it introduces other difficulties like appropriate representation of free body, it is less suited for domains whose boundaries or interfaces move substantially and it is not easy to simulate non-steady static or dynamic behavior within the frame work of this formulation.

1.4 ARBITRARY LAGRANGIAN-EULERIAN METHOD

From the above discussion, it is believed that a new formulation type which combines the advantages of the Lagrangian and Eulerian methods is essential for the accurate simulation of metal forming processes [24, 25, 26]. This new type of formulation is called Arbitrary Lagrangian Eulerian (ALE) formulation. The key point in differentiating the ALE formulation from Lagrangian or Eulerian type formulation is that in ALE a reference computation domain that can move arbitrarily and independently from the material is introduced. The movement of the reference domain is represented by a set of grid points, which may be interpreted as the movement of a finite element mesh. Therefore, in an ALE formulation, the finite element mesh need not adhere to the material or be fixed in space but may be moved arbitrarily relative to the material. A proper ALE formulation should reduce to Lagrangian formulation if we choose to use the same motion for the computational and material meshes. On the other hand, if we choose to fix the computational mesh, an ALE formulation should reduce to Eulerian formulation.

Combining the merits of both Lagrangian and Eulerian formulation, ALE is more suitable to handle mesh distortion and entanglement and special boundary condition changes in metal forming problems. If the nodes on tool-workpiece interface are specified as Eulerian points in tangential direction, it may eliminate load fluctuations, may describe precisely any contact boundary conditions and make boundary condition updating no longer necessary after each incremental step.

1.5 OBJECTIVES AND SCOPE

ALE method has the potential to eliminate problems caused by Lagrangian or Eulerian methods in simulation of general finite deformation problems. Although the concept of the ALE method was first proposed in mid-seventies [27], its use in solid mechanics problems has been restricted mainly because of the additional effort required in satisfying the deformation history-dependent stress-strain relationship and updating complexities. Relatively little attempts have been made to generalize and present ALE formulation in a general way and to provide a clear connection between ALE and Lagrangian as well as Eulerian approach. New aspects pertinent to the arbitrary mesh motion in ALE have not been investigated intensively.

The objective and steps of this research may be summarized in the following:

- Survey of existing ALE schemes with emphasis on their pertinent characteristics, reason for differences and their applicability to large deformation metal forming problems.

- Develop a complete ALE formulation that may be applied to general finite deformation cases and degenerate to either updated Lagrangian or Eulerian formulation as extreme cases [28]. Emphasise on the generality of the developed formulation to use various material laws, stress rates and on its special application to metal forming problems.
- Investigate an efficient motion scheme for the ALE method and a numerical algorithm to process the supplementary equations arising from mesh motion. Existing schemes tend to introduce higher degree of complexity in the implementation and require more CPU processing time.
- Study the effect of various stress integration methods on objectivity and plastic consistency of results and improve the numerical algorithms. Investigate the combination of stress updating algorithms with the usual return mapping schemes for small deformation. Also investigate approximate numerical integration procedures to ensure objectivity of stress rates and special tensors used in the formulation.
- Investigate a practical and more efficient method to update material associated properties for ALE approach. This is a key point in the formulation because of the arbitrary mesh motion in ALE. The scheme should be based on proper continuum mechanics equations rather than on numerical interpolation and approximation.
- Implement the developed formulation and methods to develop a general 2D finite element program and to simulate bench mark problems and metal forming processes to show the features of ALE method.

Chapter 2

ARBITRARY LAGRANGIAN EULERIAN (ALE) FORMULATION

2.1 SURVEY OF ALE METHODS

The concept of ALE was first proposed in the mid-seventies [27] under the name of "coupled Eulerian-Lagrangian", and used in conjunction with a finite difference scheme to solve two-dimensional hydrodynamics problems with moving fluid boundaries. Later, the ALE method was introduced into the finite element method [29] from fluid mechanics for modelling the solid-fluid interaction and free-surface problems, and introduced in [30] to solve nonlinear analysis of nuclear safety. Since then, ALE method has mainly been used in fluid and linear-path independent solids, where stress states are solely determined by the instantaneous displacement or velocity fields [31, 32]. Only recently the ALE method has been applied to finite strain deformation problems in solid mechanics [24, 25, 28].

Huetink [33, 34] introduced a finite element method to simulate metal forming process under the name of "combined Eulerian-Lagrangian formulation". In his formulation, the material rate of change of the equation of virtual power, with changing integration area, was used to derive the final equation. However, in the final discretized equilibrium equations, only material velocities were included. As in the updated Lagrangian method, the material velocities have to be calculated from the final equations, then the new finite element mesh can be updated by arbitrarily moving the old finite element mesh in the material domain. On the boundaries, however, the mesh can only move in the tangential direction of the material domain. Following this, is the updating of the material-associated properties. In this process, material motion and mesh motion do not occur at the same time and the discretized equilibrium equations do not include simultaneously mesh velocities and material velocities as unknowns. In this regard, Huetink formulation may be considered as an updated Lagrangian formulation (ULF) coupled with remeshing after each incremental step.

Benson [35] described a method called simple ALE. It seems that the method was developed mainly for fluid mechanics. The method was introduced to reduce the computer time of large scale analyses involving finite deformation through the use of explicit finite element schemes. The time step size of the explicit program is, therefore, limited by element mesh dimensions. In order to modify the mesh, the ALE method was implemented and the equations were derived by substituting the relationship between the material time derivative and reference configuration time derivative into the governing equations for a continuum in a Lagrangian coordinate system. An operator split was advocated to decouple the equations so that for every increment, two steps were needed. First, a Lagrangian step was performed, in which the mesh moved with the material during the step. The solution was then mapped from the Lagrangian mesh to the reference one in order to complete the ALE step. This method may be also considered as an updated Lagrangian formulation coupled with remeshing after increment.

An improved version of ALE formulation was presented by Haber [36]. In Haber's work two displacement variables were considered to be primary unknowns, i.e., the Lagrangian and Eulerian type of displacements. This allowed some distinction between the material and the computational motion. The new technique was applied to large-deformation frictional contact and fracture mechanics. It was demonstrated that the mixed Eulerian-Lagrangian description (i.e., ALE) may be used to vary the crack length in a continuous way and may easily handle contact boundary condition. The paper does not, however, clearly indicate which displacement stands for mesh motion and there is no discussion about how to move the computational finite element mesh. Furthermore, one of the most demanding aspects in ALE, the updating of material-associated properties was not addressed. It is also assumed that the formulation is limited to linear elastic material and the load is independent of displacement, therefore, the final formulation may not be directly applicable to non-linear material and deformation-dependent loads.

Similar to the above work, Yamada introduced two types of displacements called the material and spatial incremental displacements and derived an ALE formulation for plane deformation of hyperelastic material [37]. For such kind of materials, the stress can be uniquely determined from the strain energy density and it is independent of strain histories. Thus, no state quantity of the particles needs to be introduced except the deformation gradient tensor. The application of formulation to only path-independent hyperelastic material models eliminated the need for rigorously addressing the problem of calculating and updating material associated properties, and makes the formulation not applicable to the general solid mechanics deformation problems.

An ALE formulation specifically derived for solid mechanics was described by Schreurs [38, 39]. In [39], the difference between the CRS (Computational Reference System) derivative and the MRS (Material Reference System) derivative of a physical quantity was clearly stated. Starting from the equilibrium equation in material domain, the weak form was set up by the principle of weighted residuals. The final discretized equilibrium equations were obtained by transforming integration over material domain to a reference domain. However, the physical meaning of the reference domain was not clearly addressed. A computational mesh moving method was created by assuming the grid point as a material point of a linear isotropic fictitious body. Assuming one shape of a stress-free element being optimal, linear set of algebraic equations were set up to have the deformed mesh elements recover to the optimal shape. By solving the equations, the grid nodal displacements were determined. This method is time consuming and does not guarantee an optimal mesh because the real material is generally elastic-plastic. Follower points or cells are used to represent the material associated properties and deformation history. These properties at new integration points decided by CRS points were updated by extrapolation and interpolations from the follower points.

Hughes [40] elaborated on some basic concepts related to ALE. An important one is the relationship between the material time derivative ${}^tf^{\cdot}$ and referential time derivative ${}^tf^{\wedge}$ of a physical quantity ${}^tf$. These should satisfy the following relation:

$$ {}^tf^{\cdot} \;=\; {}^tf^{\wedge} + \frac{\partial^t f}{\partial^t x_i}({}^tv_i - {}^t v_i^c) \tag{2.1}$$

where ${}^tx_i$ is material coordinate, ${}^tv_i$ and ${}^tv_i^c$ are material and mesh velocities individually.

Liu [41] applied Equation (2.1) to the momentum equation and obtained an equation with respect to arbitrary reference volume. Petrov-Galerkin formulation was then utilized to set up the final discretized equilibrium equation. A stress updating procedure was developed to calculate the stress values at the quadrature and nodal points. The constitutive law was reformulated by introducing a stress-velocity product; the Petrov-Galerkin finite element method was used to set up the equivalent weak form equations of the constitutive law. These equations were solved simultaneously with the final equilibrium equations to get the solution. With this scheme, some wave propagation problems and elastic-plastic dynamic deformation processes have been simulated. In the same paper, Liu also transformed the integration extending over the material domain to referential domain and derived the referential time derivative of internal virtual work. In a later development [42], Liu considered the frictional interface of plane deformation problems and derived equilibrium equations for such case. The Laplace differential equation and fourth order differential equation were proposed as mesh generator to manage mesh movement. The application of proposed ALE algorithm to metal rolling simulation was presented. It was shown that for small rollers, both Lagrangian and ALE meshes are feasible and the agreement between the two is quite good. For realistic roller size, a Lagrangian mesh failed to complete the simulation whereas the ALE mesh performed quite well.

Extended from fluid mechanics, an ALE formulation for solid mechanics was reported by Ghosh et al. [24, 43, 44]. Different from the methods discussed previously, the Reynolds transport theorem widely used in fluid mechanics was applied to an arbitrary moving control volume to obtain the field equations of mechanics with respect to an arbitrarily moving grid point. The weak form of the differential equations were obtained by using appropriate weighting functions and integrating over the current grid volume. In Ghosh's work it was assumed that the motion of the material with respect to the grid is quasi-static, i.e., $\frac{\partial^t v_i}{\partial t}\Big|_{\boldsymbol{\chi}} = 0$. In the opinion of the author, this assumption is not warranted. In quasi-static problems, it is the material point acceleration, as expressed by Equation (2.2), that may be neglected instead of the motion of the material with respect to the grid points, because the term contributes only partially to the material point acceleration.

$$\left.\frac{\partial^t v_i}{\partial t}\right|_{^t\mathbf{x}} = \left.\frac{\partial^t v_i}{\partial t}\right|_{^t\boldsymbol{\chi}} + ({}^t v_j -{}^t v_j^c)\frac{\partial^t v_i}{\partial^t x_j} = 0 \qquad (2.2)$$

On the other hand, the exclusive characteristics of the ALE, that the grid points can move arbitrarily, is restricted so much by this assumption. Using implicit time integration scheme, Ghosh integrated the weak form equation over the current control volume. This makes the calculation laborious because, unlike transient field problems, the integration volume here is changing with time. In [43] Ghosh described a way to update variables to nodal points of arbitrary motion. Pseudo-material elements were constructed first and material-associated variables were evaluated by interpolation. Local algebraic and elliptic mesh generator was implemented to perform mesh management in highly local deformation areas [44]. Although Ghosh indicated that the specific cases of ALE formulation should be either updated Lagrangian or Eulerian formulation, no verification is presented in his work showing that the formulation may reduce to these special cases.

A particular ALE implementation; ALE3D, in metal forming simulations is recently introduced by Cough et al. [25]. The basic computational cycle consists of a Lagrangian step followed by an advection step. At the end of Lagrangian phase of the cycle the velocities and nodal positions are updated. At this point, the user has several options. If the user opts to run a pure Lagrangian mode, no further action is taken and the code proceeds to the next time step. If a pure Eulerian calculation is desired, the nodes are placed back in their original positions. The user has options available to tailor the evolution of the mesh in order to maximize either efficiency or accuracy. The mesh updating scheme implemented in ALE3D is a finite element-based equipotential method. For the constitutive law, the Jaumann rate is used for the stress tensor and von Mises yield condition is applied. In the paper [25], only program features are described and no formulation or rigorous algorithm is given.

An overall description of ALE is presented in [45] by Huerta and Casadei. It is clearly indicated that the most important challenge for the ALE technique lies in its extension to solid mechanics problems in general, and, in particular, to non-linear solid mechanics where path dependent material behavior is fairly common. It is pointed out that the best choice for the mesh motion or velocities and a low cost algorithm for updating the material-related properties constitute the major problems. However, no particular schemes are given or presented. Some primary governing issues, e.g., the conservation laws, constitutive equations and boundary conditions, are presented, but no complete formulation is given. The applications considered in the paper are some elasto-plastic problems, with concentration on fast-transient solid dynamics showing the effectiveness of ALE to impact problems when the explicit integration scheme is used.

Although several forms of ALE schemes have been discussed in the literature, it is noted, however, that these formulations always concentrate on certain aspects of the problem which makes the outcome formulation incomplete and suitable only for specific applications. Crucial points in the formulation, e.g., the final expression of equilibrium equations in ALE frame, the relation between computational and material points, evaluation of material related data, are not clearly addressed in the literature and need further investigation.

2.2 CONSISTENT ALE FORMULATION

2.2.1 Geometric and Kinematic Description

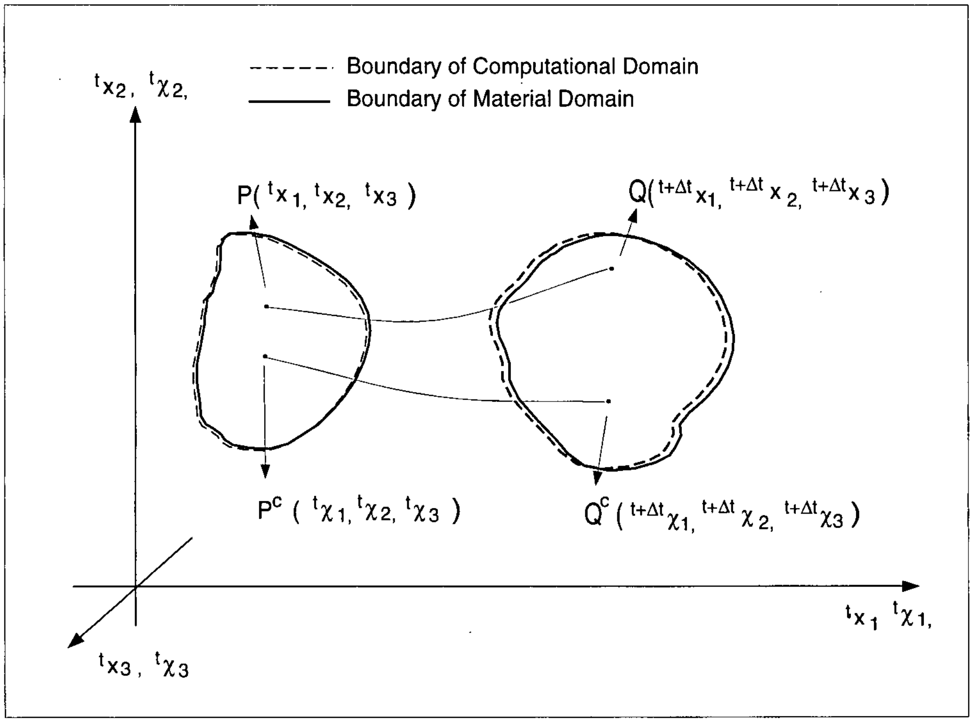

$^tx_2$, $^t\chi_2$,
Boundary of Computational Domain
Boundary of Material Domain
$P(^tx_1, {}^tx_2, {}^tx_3)$
$Q(^{t+\Delta t}x_1, {}^{t+\Delta t}x_2, {}^{t+\Delta t}x_3)$
$P^c(^t\chi_1, {}^t\chi_2, {}^t\chi_3)$
$Q^c(^{t+\Delta t}\chi_1, {}^{t+\Delta t}\chi_2, {}^{t+\Delta t}\chi_3)$
$^tx_1$ $^t\chi_1$,
$^tx_3$, $^t\chi_3$

Figure 2.1: Schematic diagram for domains and mapping in ALE description

Assuming a material particle $P(^tx_1,^t x_2,^t x_3)$ in a material reference system (MRS) at time t has velocity ${}^tv_i(i = 1,2,3)$. The particle moves to $Q(^{t+\Delta t}x_1,^{t+\Delta t} x_2,^{t+\Delta t} x_3)$ at time $t + \Delta t$. Similarly, a grid point $P^c(^t\chi_1,^t \chi_2,^t \chi_3)$ in the corresponding computational reference system (CRS) at time t has velocity ${}^tv_i^c(i = 1,2,3)$ and moves to $Q^c(^{t+\Delta t}\chi_1,^{t+\Delta t} \chi_2,^{t+\Delta t} \chi_3)$ at time $t + \Delta t$, as shown in Figure 2.1. In ALE description, the motion of the finite element grid need not adhere to the material particle and may be controlled by the finite element user in accordance with users judgement. However, we should ensure a one-to-one mapping between the material domain and computational domain at any time t, i.e., for each:

$$ {}^t x_i = x_i({}^t\chi_1, {}^t\chi_2, {}^t\chi_3) \qquad (i = 1,2,3) \tag{2.3} $$

we have only one:

$$ {}^t\chi_i = \chi_i({}^t x_1, {}^t x_2, {}^t x_3) \qquad (i = 1,2,3) $$

The necessary and sufficient condition for the inverse relation to exist, is that the determinant of the Jacobian transformation,

$$ {}^t J^c = \left| \frac{\partial^t x_i}{\partial^t \chi_j} \right| \tag{2.4} $$

is non-vanishing. A chosen mesh motion scheme should satisfy the above requirement. In addition, at every stage of motion, the boundary of the deformed configuration of the material body (solid line in Figure 2.1) must coincide with the boundary of the configuration of the computational reference domain (dashed line in Figure 2.1), i.e., on boundary:

$$ ({}^t v_i - {}^t v_i^c)\,{}^t n_i = 0 \tag{2.5} $$

where ${}^t n_i$ is the component of the unit vector normal to the boundary. The physical interpretation of Equation (2.5) is that no normal convective velocity occurs across the boundary if the surface particles remain on the surface.

It is then possible to derive the governing equations with respect to the referential domain. Since ${}^t\chi_i$ is our computational reference coordinate system, it is necessary to express the material time derivative of a function f in terms of the time derivative with respect to ${}^t\chi_i$. Assume that the material time derivative and the computational reference time derivative are denoted by a superscripted dot "$\cdot$" and a cap "$\wedge$", respectively. The relationship between the two derivatives is given by Equation (2.1) (See Appendix A for details). Equation (2.1) is important since it makes it possible to track the material deformation history when ALE is used.

2.2.2 ALE Formulation

Many problems in solid mechanics may be treated as quasi-static, so that the continuity equation can be satisfied automatically. Starting from the principle of virtual work at time $t + \Delta t$, the equilibrium equation may be expressed as:

$$\int_{^{t+\Delta t}V} {}^{t+\Delta t}\sigma_{ij}\delta_{t+\Delta t}e_{ij}d^{t+\Delta t}V = \int_{^{t+\Delta t}V} {}^{t+\Delta t}f_i^B\delta u_i d^{t+\Delta t}V + \int_{^{t+\Delta t}S} {}^{t+\Delta t}f_i^S\delta u_i d^{t+\Delta t}S \quad (2.6)$$

Notation similar to those used by Bathe [46, 47] are adopted. Referring to a general motion of a body in a fixed Cartesian coordinate system, as shown in Figure 2.1, the left superscripts indicate the configuration at which the quantity occurs, and the left subscripts indicate the configuration with respect to which the quantity is measured. The left subscript may not be used if the quantity under consideration occurs in the same configuration in which it is measured. A quantity with no left superscript or subscript indicates an increment from time t to $t + \Delta t$. Therefore, ${}^{t+\Delta t}\sigma_{ij}$ in Equation 2.6 is the Cartesian component of Cauchy stress referred to the configuration at time $t + \Delta t$. The deformation tensor ${}_{t+\Delta t}e_{ij}$ is defined by:

$$ {}_{t+\Delta t}e_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial^{t+\Delta t}x_j} + \frac{\partial u_j}{\partial^{t+\Delta t}x_i}\right) \quad (2.7)$$

where u_i are admissible incremental displacement vector and δ in Equation (2.6) means a variation. The body force and the surface traction at time $t + \Delta t$ are given by ${}^{t+\Delta t}f_i^B$ and ${}^{t+\Delta t}f_i^S$ in Equation (2.6), respectively. ${}^{t+\Delta t}V$ is the volume of deformed body and ${}^{t+\Delta t}S$ is the surface of deformed body at time $t + \Delta t$ on which surface tractions are prescribed.

Utilizing the following relations:

$$\begin{aligned}
\int_{^{t+\Delta t}V} {}^{t+\Delta t}\sigma_{ij}\delta_{t+\Delta t}e_{ij}d^{t+\Delta t}V &= \int_{^{t+\Delta t}V} {}^{t+\Delta t}\sigma_{ij}\frac{1}{2}\left(\frac{\partial u_i}{\partial^{t+\Delta t}x_j} + \frac{\partial u_j}{\partial^{t+\Delta t}x_i}\right)d^{t+\Delta t}V \\
&= \int_{^{t+\Delta t}V} {}^{t+\Delta t}\sigma_{ij}\frac{\partial u_i}{\partial^{t+\Delta t}x_j}d^{t+\Delta t}V \\
{}^{t+\Delta t}f_i^S &= {}^{t+\Delta t}\sigma_{ij}{}^{t+\Delta t}n_j \\
{}^{t+\Delta t}n_j d^{t+\Delta t}S &= d^{t+\Delta t}A_j
\end{aligned}$$

where $d^{t+\Delta t}A_j$ is the projection of $d^{t+\Delta t}S$ on the coordinate plane having axis ${}^t x_j$ or ${}^{t+\Delta t}x_j$ as normal direction, ${}^{t+\Delta t}n_j$ are the direction cosines of area element $d^{t+\Delta t}S$ with axis ${}^t x_j$ or ${}^{t+\Delta t}x_j$, the Equation (2.6) may be written in the following form:

$$\int_{{}^{t+\Delta t}V} {}^{t+\Delta t}\sigma_{ij} \frac{\partial u_i}{\partial^{t+\Delta t}x_j} d^{t+\Delta t}V = \int_{{}^{t+\Delta t}V} {}^{t+\Delta t}f_i^B \delta u_i d^{t+\Delta t}V + \int_{{}^{t+\Delta t}A_j} {}^{t+\Delta t}\sigma_{ij} \delta u_i d^{t+\Delta t}A_j \quad (2.8)$$

In order to solve Equation (2.8), all quantities are transformed into the known computational reference configuration at time t, which is particularly chosen as the material reference domain at time t, i.e., making P^c and P in Figure 2.1 be the same point. This transformation is only due to the change in or motion of the computational reference domain. We have,

$$\frac{\partial \delta u_i}{\partial^{t+\Delta t}x_j} = \frac{\partial \delta u_i}{\partial^t x_k} \frac{\partial^t x_k}{\partial^{t+\Delta t}x_j} \quad (2.9)$$

$${}^{t+\Delta t}x_k = {}^t x_k + u_k^c \quad (2.10)$$

where u_k^c is the mesh point or CRS displacement from time t to $t + \Delta t$. Substituting in Equation (2.9) and reordering,

$$\frac{\partial \delta u_i}{\partial^{t+\Delta t}x_j} - \frac{\partial \delta u_i}{\partial^t x_j} = -\frac{\partial \delta u_i}{\partial^t x_k} \frac{\partial u_k^c}{\partial^{t+\Delta t}x_j}$$

dividing the above equation by Δt and let $\Delta t \to 0$, we obtain,

$$\left(\frac{\partial \delta u_i}{\partial^t x_j}\right)^{\wedge} = -\frac{\partial \delta u_i}{\partial^t x_k} \frac{\partial^t v_k^c}{\partial^t x_j}$$

which gives,

$$\begin{aligned} \frac{\partial \delta u_i}{\partial^{t+\Delta t}x_j} &\approx \left.\frac{\partial \delta u_i}{\partial^t x_j}\right|_{{}^t\chi} + \left(\frac{\partial \delta u_i}{\partial^t x_j}\right)^{\wedge} \Delta t \\ &= \left.\frac{\partial \delta u_i}{\partial^t x_j}\right|_{{}^t\chi} - \frac{\partial \delta u_i}{\partial^t x_k} \frac{\partial^t v_k^c}{\partial^t x_j} \Delta t \end{aligned} \quad (2.11)$$

The volume element $d^{t+\Delta t}V$ is related to $d^t V|_{{}^t\chi}$ through:

$$d^{t+\Delta t}V = d^t V\Big|_{{}^t\chi} + (d^t V)^{\wedge} \Delta t \quad (2.12)$$

The computational reference time rate of volume is [48]:

$$(d^tV)^\wedge \;=\; \frac{\partial^t v^c_k}{\partial^t x_k} d^tV$$

and it follows that:

$$d^{t+\Delta t}V = d^tV\Big|_{{}^t\chi} + \frac{\partial^t v^c_k}{\partial^t x_k}\Delta t d^tV \tag{2.13}$$

The stress component in Equation (2.8) may be expressed by:

$$ {}^{t+\Delta t}\sigma_{ij} = {}^t\sigma_{ij}\Big|_{{}^t\chi} + \left({}^t\sigma_{ij}\right)^\wedge \Delta t = {}^t\sigma_{ij}\Big|_{{}^t\chi} + \left({}^t\dot{\sigma}_{ij} + ({}^tv^c_k -{}^t v_k)\frac{\partial^t\sigma_{ij}}{\partial^t x_k}\right)\Delta t \tag{2.14}$$

where, $({}^t\sigma_{ij})^\wedge = \left({}^t\dot{\sigma}_{ij} + ({}^tv^c_k -{}^t v_k)\frac{\partial^t\sigma_{ij}}{\partial^t x_k}\right)$ denotes the computational reference time rate of ${}^t\sigma_{ij}$. Since the computation reference domain is chosen to be the material reference domain at time t, i.e., ${}^tx_i ={}^t \chi_i$, the following relations may be obtained:

$$\begin{aligned}
\frac{\partial\delta u_i}{\partial^t x_j}\Big|_{{}^t\chi} &= \frac{\partial\delta u_i}{\partial^t x_j}\Big|_{{}^t\mathbf{x}} = \frac{\partial\delta u_i}{\partial^t x_j} \\
d^tV\Big|_{{}^t\chi} &= d^tV\Big|_{{}^t\mathbf{x}} = d^tV \\
{}^t\sigma_{ij}\Big|_{{}^t\chi} &= {}^t\sigma_{ij}\Big|_{{}^t\mathbf{x}} ={}^t \sigma_{ij}
\end{aligned}$$

Substituting Equations (2.11), (2.13), (2.14) and the above equations into the left hand side of Equation (2.8), the following linearized form of the internal virtual work may be obtained:

$$\begin{aligned}
LHS &= \int_{{}^{t+\Delta t}V} {}^{t+\Delta t}\sigma_{ij}\frac{\partial\delta u_i}{\partial^{t+\Delta t}x_j} d^{t+\Delta t}V \\
&= \int_{{}^tV}\left({}^t\sigma_{ij} + {}^t\dot{\sigma}_{ij}\Delta t + ({}^tv^c_p -{}^t v_p)\frac{\partial^t\sigma_{ij}}{\partial^t x_p}\Delta t\right)\left(\frac{\partial\delta u_i}{\partial^t x_j} - \frac{\partial\delta u_i}{\partial^t x_k}\frac{\partial^t v^c_k}{\partial^t x_j}\Delta t\right) \\
&\times \left(d^tV + \frac{\partial^t v^c_l}{\partial^t x_l}\Delta t d^tV\right) \\
&\approx \int_{{}^tV} {}^t\sigma_{ij}\frac{\partial\delta u_i}{\partial^t x_j} d^tV + \int_{{}^tV}\left(-{}^t\sigma_{ij}\frac{\partial\delta u_i}{\partial^t x_k}\frac{\partial^t v^c_k}{\partial^t x_j} +{}^t \sigma_{ij}\frac{\partial\delta u_i}{\partial^t x_j}\frac{\partial^t v^c_l}{\partial^t x_l}\right)\Delta t d^tV \\
&+ \int_{{}^tV}\left({}^t\dot{\sigma}_{ij}\frac{\partial\delta u_i}{\partial^t x_j} + ({}^tv^c_p -{}^t v_p)\frac{\partial^t\sigma_{ij}}{\partial^t x_p}\frac{\partial\delta u_i}{\partial^t x_j}\right)\Delta t d^tV
\end{aligned}$$

$$\begin{aligned} &= \int_{^tV} {}^t\sigma_{ij}\frac{\partial\delta u_i}{\partial^t x_j}d^tV + \int_{^tV}\frac{\partial\delta u_i}{\partial^t x_j}\left(-\frac{\partial^t v_j^c}{\partial^t x_k}{}^t\sigma_{ik} + {}^t\sigma^{\cdot}_{ij}\right)\Delta t d^tV \\ &+ \int_{^tV}\frac{\partial\delta u_i}{\partial^t x_j}\left(({}^tv_k^c -{}^t v_k)\frac{\partial^t\sigma_{ij}}{\partial^t x_k} +{}^t \sigma_{ij}\frac{\partial^t v_k^c}{\partial^t x_k}\right)\Delta t d^tV \end{aligned} \tag{2.15}$$

All terms with order higher than linear term of Δt are ignored. Considering the right hand side of Equation (2.8), we have:

$$^{t+\Delta t}f_i^B = {}^tf_i^B\Big|_{^t\chi} +{}^t f_i^{B\wedge}\Delta t \tag{2.16}$$

From Equation (2.1),

$$^tf_i^{B\wedge} ={}^t f_i^{B\cdot} + ({}^tv_p^c -{}^t v_p)\frac{\partial^t f_i^B}{\partial^t x_p}$$

Where ${}^tf_i^{B\cdot}$ is the material time rate of ${}^tf_i^B$ at time t. Substituting into Equation (2.16) and considering ${}^tx_i ={}^t \chi_i$

$$^{t+\Delta t}f_i^B ={}^t f_i^B + \left({}^tf_i^{B\cdot} + ({}^tv_k^c -{}^t v_k)\frac{\partial^t f_i^B}{\partial^t x_k}\right)\Delta t \tag{2.17}$$

Similarly, the area element $d^{t+\Delta t}A_j$ may be described as:

$$d^{t+\Delta t}A_j = d^tA_j\Big|_{^t\chi} + d^tA_j^\wedge\Delta t$$

The computational reference time rate of the area is [48]:

$$d^tA_j^\wedge = \frac{\partial^t v_k^c}{\partial^t x_k}d^tA_j - \frac{\partial^t v_k^c}{\partial^t x_j}d^tA_k$$

so that:

$$d^{t+\Delta t}A_j = d^tA_j\Big|_{^t\chi} + \left(\frac{\partial^t v_k^c}{\partial^t x_k}d^tA_j - \frac{\partial^t v_k^c}{\partial^t x_j}d^tA_k\right)\Delta t \tag{2.18}$$

Using Equations (2.13), (2.14), (2.17) and (2.18) and considering the relations ${}^tx_i = {}^t\chi_i$, the right hand side of Equation (2.8) may be written as:

$$RHS = \int_{^{t+\Delta t}V} {}^{t+\Delta t}f_i^B\delta u_i d^{t+\Delta t}V + \int_{^{t+\Delta t}A_j} {}^{t+\Delta t}\sigma_{ij}\delta u_i d^{t+\Delta t}A_j$$

$$\begin{aligned}
&= \int_{{}^tV} \delta u_i \left({}^tf_i^B + \left({}^tf_i^{B\cdot} + ({}^tv_p^c -{}^t v_p)\frac{\partial^t f_i^B}{\partial^t x_p}\right)\Delta t\right)\left(d^tV + \frac{\partial^t v_k^c}{\partial^t x_k}\Delta t d^tV\right) \\
&+ \int_{{}^tA_j} \delta u_i \left({}^t\sigma_{ij} + \left({}^t\sigma^{\cdot}_{ij} + ({}^tv_k^c -{}^t v_k)\frac{\partial^t \sigma_{ij}}{\partial^t x_k}\right)\Delta t\right) \\
&\times \left(d^tA_j + \left(\frac{\partial^t v_k^c}{\partial^t x_k}d^tA_j - \frac{\partial^t v_k^c}{\partial^t x_j}d^tA_k\right)\Delta t\right) \\
&\approx \int_{{}^tV} \delta u_i \left({}^tf_i^B +{}^t f_i^B\frac{\partial^t v_k^c}{\partial^t x_k}\Delta t + \left({}^tf_i^{B\cdot} + ({}^tv_p^c -{}^t v_p)\frac{\partial^t f_i^B}{\partial^t x_p}\right)\Delta t\right) d^tV \\
&+ \int_{{}^tA_j} \delta u_i \left({}^t\sigma_{ij}d^tA_j +{}^t \sigma_{ij}\left(\frac{\partial^t v_k^c}{\partial^t x_k}d^tA_j - \frac{\partial^t v_k^c}{\partial^t x_j}d^tA_k\right)\Delta t\right) \\
&+ \int_{{}^tA_j} \delta u_i \left({}^t\sigma^{\cdot}_{ij} + ({}^tv_k^c -{}^t v_k)\frac{\partial^t \sigma_{ij}}{\partial^t x_k}\right)\Delta t d^tA_j \\
&= \int_{{}^tV} \delta u_i^t f_i^B d^tV + \int_{{}^tA_j} \delta u_i^t \sigma_{ij} d^tA_j \\
&+ \int_{{}^tV} \delta u_i \left({}^tf_i^{B\cdot} +{}^t f_i^B\frac{\partial^t v_k^c}{\partial^t x_k} + ({}^tv_k^c -{}^t v_k)\frac{\partial^t f_i^B}{\partial^t x_k}\right)\Delta t d^tV \\
&+ \int_{{}^tA_j} \delta u_i \left({}^t\sigma^{\cdot}_{ij} +{}^t \sigma_{ij}\frac{\partial^t v_k^c}{\partial^t x_k} -{}^t \sigma_{ik}\frac{\partial^t v_j^c}{\partial^t x_k} + ({}^tv_k^c -{}^t v_k)\frac{\partial^t \sigma_{ij}}{\partial^t x_k}\right)\Delta t d^tA_j
\end{aligned}$$

where all higher order terms of Δt are neglected. Since $d^tA_j ={}^t n_j d^tS$ and ${}^t\sigma^{\cdot t}_{ij} n_j ={}^t f_i^{S\cdot}$ is the traction (or load) rate with the displacements and gradients held fixed, corresponding to the $\Delta_{load}T_i^0$ in [2], we have:

$$\begin{aligned}
RHS &= \int_{{}^tV} \delta u_i^t f_i^B d^tV + \int_{{}^tA_j} \delta u_i^t \sigma_{ij} d^tA_j \\
&+ \int_{{}^tV} \delta u_i \left({}^tf_i^{B\cdot} +{}^t f_i^B\frac{\partial^t v_k^c}{\partial^t x_k} + ({}^tv_k^c -{}^t v_k)\frac{\partial^t f_i^B}{\partial^t x_k}\right)\Delta t d^tV \\
&+ \int_{{}^tS} \delta u_i \left({}^tf_i^{S\cdot} +{}^t f_i^S\frac{\partial^t v_k^c}{\partial^t x_k} -{}^t \sigma_{ik}\frac{\partial^t v_j^c}{\partial^t x_k}{}^t n_j\right)\Delta t d^tS \\
&+ \int_{{}^tS} \delta u_i ({}^tv_k^c -{}^t v_k)\frac{\partial^t \sigma_{ij}}{\partial^t x_k}{}^t n_j \Delta t d^tS
\end{aligned} \tag{2.19}$$

Finally, using LHS Equation (2.15) and RHS Equation (2.19), and applying the relation:

$$\int_{{}^tV} {}^t\sigma_{ij}\frac{\partial u_i}{\partial^t x_j}d^tV = \int_{{}^tV} {}^tf_i^B \delta u_i d^tV + \int_{{}^tA_j} {}^t\sigma_{ij}\delta u_i d^tA_j$$

We obtain the final ALE formulation in the form:

$$\begin{aligned}
& \int_{{}^tV} \frac{\partial \delta u_i}{\partial^t x_j} \left({}^t\sigma^{\cdot}_{ij} - {}^t\sigma_{ik} \frac{\partial^t v^c_j}{\partial^t x_k} + {}^t\sigma_{ij} \frac{\partial^t v^c_k}{\partial^t x_k} + ({}^t v^c_k - {}^t v_k) \frac{\partial^t \sigma_{ij}}{\partial^t x_k} \right) d^tV \\
- & \int_{{}^tV} \delta u_i \left({}^t f^B_i \frac{\partial^t v^c_k}{\partial^t x_k} + ({}^t v^c_k - {}^t v_k) \frac{\partial^t f^B_i}{\partial^t x_k} \right) d^tV \\
- & \int_{{}^tS} \delta u_i \left({}^t f^S_i \frac{\partial^t v^c_k}{\partial^t x_k} - {}^t\sigma_{ik} {}^t n_j \frac{\partial^t v^c_j}{\partial^t x_k} + ({}^t v^c_k - {}^t v_k) {}^t n_j \frac{\partial^t \sigma_{ij}}{\partial^t x_k} \right) d^tS \\
= & \int_{{}^tV} \delta u_i {}^t f^{B\cdot}_i d^tV + \int_{{}^tS} \delta u_i {}^t f^{S\cdot}_i d^tS
\end{aligned} \tag{2.20}$$

The above formulation is a complete and general one in the sense that it may be applied to various types of finite deformation problems with generalized types of loadings and boundary conditions. The formulation is also suitable for implementing various types of rate dependent and rate-independent material constitutive relations. Another unique feature in the formulation, is the existence of generalized load correction terms that facilitates handling of deformation-dependent loads, e.g., follower loads. These features are discussed in the following subsection.

2.3 DISCUSSION AND CHARACTERISTICS OF PROPOSED FORMULATION

The above formulation has special features and may differ from similar ones in the literature in the following main aspects:

- No specific assumptions were introduced after the principle of virtual work. Also, velocities, instead of displacements, are used as primary variables. This makes it more straight forward to implement rate-dependent material laws.
- Consistent load correction term to handle deformation-dependent loads are introduced.
- The integral volume and surface have clear physical meanings which are the material domain after the last incremental step. This is arrived at by choosing the arbitrary reference domain coincident with the material domain after each incremental step.

- The formulation represents a strict ALE method because the mesh motion and deformation can occur independently at the same time, i.e., both material velocity and mesh velocity are included in Equation (2.20).

- Different objective stress rates can be applied to the formulation. If Jaumann stress rate [47],

$$ {}^t\sigma^{\cdot J}_{ij} ={}^t \sigma^{\cdot}_{ij} - {}^t\sigma_{ik}\left(\frac{1}{2}\left(\frac{\partial^t v_j}{\partial^t x_k} - \frac{\partial^t v_k}{\partial^t x_j}\right)\right) - {}^t\sigma_{jk}\left(\frac{1}{2}\left(\frac{\partial^t v_i}{\partial^t x_k} - \frac{\partial^t v_k}{\partial^t x_i}\right)\right) \tag{2.21}$$

  is introduced, the formulation (2.20) takes the following particular form:

$$\begin{aligned}
& \int_{{}^tV} \frac{\partial \delta u_i}{\partial^t x_j}\left({}^t\sigma^{\cdot J}_{ij} + \frac{{}^t\sigma_{ik}}{2}\left(\frac{\partial^t v_j}{\partial^t x_k} - \frac{\partial^t v_k}{\partial^t x_j}\right) + \frac{{}^t\sigma_{jk}}{2}\left(\frac{\partial^t v_i}{\partial^t x_k} - \frac{\partial^t v_k}{\partial^t x_i}\right)\right) d^tV \\
+ & \int_{{}^tV} \frac{\partial \delta u_i}{\partial^t x_j}\left(-{}^t\sigma_{ik}\frac{\partial^t v^c_j}{\partial^t x_k} + {}^t\sigma_{ij}\frac{\partial^t v^c_k}{\partial^t x_k} + ({}^tv^c_k -{}^t v_k)\frac{\partial^t \sigma_{ij}}{\partial^t x_k}\right) d^tV \\
- & \int_{{}^tV} \delta u_i \left({}^tf^B_i\frac{\partial^t v^c_k}{\partial^t x_k} + ({}^tv^c_k -{}^t v_k)\frac{\partial^t f^B_i}{\partial^t x_k}\right) d^tV \\
- & \int_{{}^tS} \delta u_i \left({}^tf^S_i\frac{\partial^t v^c_k}{\partial^t x_k} - {}^t\sigma_{ik}{}^tn_j\frac{\partial^t v^c_j}{\partial^t x_k} + ({}^tv^c_k -{}^t v_k){}^tn_j\frac{\partial^t \sigma_{ij}}{\partial^t x_k}\right) d^tS \\
= & \int_{{}^tV} \delta u^t_i f^{B\cdot}_i d^tV + \int_{{}^tS} \delta u^t_i f^{S\cdot}_i d^tS
\end{aligned} \tag{2.22}$$

  If Truesdell stress rate [49],

$$ {}^t\sigma^{\cdot T}_{ij} ={}^t \sigma^{\cdot}_{ij} - {}^t\sigma_{kj}\frac{\partial^t v_i}{\partial^t x_k} - {}^t\sigma_{ik}\frac{\partial^t v_j}{\partial^t x_k} + {}^t\sigma_{ij}\frac{\partial^t v_k}{\partial^t x_k} \tag{2.23}$$

  is applied, Equation (2.20) takes the form:

$$\begin{aligned}
& \int_{{}^tV} \frac{\partial \delta u_i}{\partial^t x_j}\left({}^t\sigma^{\cdot T}_{ij} +^t \sigma_{kj}\frac{\partial^t v_i}{\partial^t x_k} +^t \sigma_{ik}\frac{\partial^t v_j}{\partial^t x_k} -^t \sigma_{ij}\frac{\partial^t v_k}{\partial^t x_k}\right) d^tV \\
+ & \int_{{}^tV} \frac{\partial \delta u_i}{\partial^t x_j}\left(-{}^t\sigma_{ik}\frac{\partial^t v^c_j}{\partial^t x_k} + {}^t\sigma_{ij}\frac{\partial^t v^c_k}{\partial^t x_k} + ({}^tv^c_k -{}^t v_k)\frac{\partial^t \sigma_{ij}}{\partial^t x_k}\right) d^tV \\
- & \int_{{}^tV} \delta u_i \left({}^tf^B_i\frac{\partial^t v^c_k}{\partial^t x_k} + ({}^tv^c_k -{}^t v_k)\frac{\partial^t f^B_i}{\partial^t x_k}\right) d^tV \\
- & \int_{{}^tS} \delta u_i \left({}^tf^S_i\frac{\partial^t v^c_k}{\partial^t x_k} - {}^t\sigma_{ik}{}^tn_j\frac{\partial^t v^c_j}{\partial^t x_k} + ({}^tv^c_k -{}^t v_k){}^tn_j\frac{\partial^t \sigma_{ij}}{\partial^t x_k}\right) d^tS \\
= & \int_{{}^tV} \delta u^t_i f^{B\cdot}_i d^tV + \int_{{}^tS} \delta u^t_i f^{S\cdot}_i d^tS
\end{aligned} \tag{2.24}$$

- The formulation reduces to updated Lagrangian one when the material velocity is chosen equal to mesh velocity, i.e., when $v_i = v_i^c$. Equation (2.22) takes the form:

$$
\begin{aligned}
& \int_{{}^tV} \frac{\partial \delta u_i}{\partial^t x_j} \left({}^t\sigma_{ij}^{\cdot J} - \frac{{}^t\sigma_{ik}}{2} \left(\frac{\partial^t v_j}{\partial^t x_k} + \frac{\partial^t v_k}{\partial^t x_j} \right) + \frac{{}^t\sigma_{jk}}{2} \left(\frac{\partial^t v_i}{\partial^t x_k} - \frac{\partial^t v_k}{\partial^t x_i} \right) +^t \sigma_{ij} \frac{\partial^t v_k}{\partial^t x_k} \right) d^tV \\
- & \int_{{}^tV} \delta u_i \left({}^tf_i^B \frac{\partial^t v_k}{\partial^t x_k} \right) d^tV - \int_{{}^tS} \delta u_i \left({}^tf_i^S \frac{\partial^t v_k}{\partial^t x_k} - {}^t\sigma_{ik}{}^tn_j \frac{\partial^t v_j}{\partial^t x_k} \right) d^tS \\
= & \int_{{}^tV} \delta u_i^t f_i^{B\cdot} d^tV + \int_{{}^tS} \delta u_i^t f_i^{S\cdot} d^tS \qquad (2.25)
\end{aligned}
$$

which is the same as the updated Lagrangian formulation given by McMeeking [3], as verified in Appendix B, except for the load correction terms:

$$\int_{{}^tV} \delta u_i \left({}^tf_i^B \frac{\partial^t v_k}{\partial^t x_k} \right) d^tV$$

and

$$\int_{{}^tS} \delta u_i \left({}^tf_i^S \frac{\partial^t v_k}{\partial^t x_k} - {}^t\sigma_{ik}{}^tn_j \frac{\partial^t v_j}{\partial^t x_k} \right) d^tS$$

- On the other hand, if mesh velocity is chosen to be equal to zero i.e., $v_i^c = 0$, the formulation (2.20) reduces to a general Eulerian formulation, as shown in the following equation:

$$
\begin{aligned}
& \int_{{}^tV} \frac{\partial \delta u_i}{\partial^t x_j} \left({}^t\sigma_{ij}^{\cdot} -^t v_k \frac{\partial^t \sigma_{ij}}{\partial^t x_k} \right) d^tV = \int_{{}^tV} \delta u_i \left({}^tf_i^{B\cdot} -^t v_k \frac{\partial^t f_i^B}{\partial^t x_k} \right) d^tV \\
+ & \int_{{}^tS} \delta u_i \left({}^tf_i^{S\cdot} - {}^tv_k{}^tn_j \frac{\partial^t \sigma_{ij}}{\partial^t x_k} \right) d^tS \qquad (2.26)
\end{aligned}
$$

- The problem of mesh distortion is easier to handle in ALE method. The ALE method can not only handle the mesh distortion but also improve the calculation accuracy. Because the motion of the mesh is arbitrary, the new mesh is always chosen to give an optimal results.

- In the application of Equation (2.20), various domains in the structure may be specified as Lagrangian, Eulerian or ALE ones. A practical example for the importance of this point may be seen when we consider the interface between the tool and work-piece in a metal forming problem. In this case, two specific directions are considered, perpendicular and parallel to the tool interface. In the direction perpendicular to the tool interface, the nodes will be kept as Lagrangian points, i.e., nodes will be moving with the tool at same speed. In the other direction, however, nodes will be kept as Eulerian points, i.e., nodes will be fixed parallel to the tool interface. Contact boundary condition may be accurately described, therefore, in any general form of tool workpiece interface. The updating of boundary condition is no longer necessary. The "punch size increasement", and load fluctuations discussed in Chapter 1, should generally not occur. This will undoubtedly increase the efficiency and accuracy of the simulation of metal forming problems.

Chapter 3

NUMERICAL IMPLEMENTATION OF ALE

3.1 INTRODUCTION AND BRIEF SURVEY

3.1.1 Mesh Motion and Stiffness Matrix Processing

In general, the stiffness matrix in ALE formulation may be rectangular and nonsymmetric. The rectangular nature of the matrix results from the fact that there are two unknowns for each degree of freedom, one related to the material and the another related to the mesh, while only one equation may be derived from the equilibrium for each degree of freedom. There are generally two ways to get a solvable set of equations in an ALE formulation. One is to specify the mesh displacements or velocities before solving the linearized equilibrium equations. In each incremental step, the mesh motion is decided by certain algorithms and data from previous increments. It is generally independent of the deformation in the current step. This method is straight forward and has been applied by many researchers [24, 39, 44]. The method, however, is not a general one and may not be capable of considering the effect of the current material motion. Another method is to set up supplementary constraint equations, i.e., relations between material displacements or velocities and mesh displacements or velocities. In this method, the mesh motion is coupled with the material motion in the current step by the constraint equations. This method is a more general one and has the potential of producing higher quality mesh, but it will generally double the number of unknowns and increase the calculation time significantly.

The mesh motion algorithm is an important and critical aspect in ALE formulation and it affects the computation efficiency and accuracy. Schreurs [39] introduced a mesh motion technique for an ALE formulation that requires the solution of a simultaneous set of equations. Other researchers applied regular finite element mesh generation techniques to move the computational mesh in ALE. For example, Benson [35] used relaxation stencil method derived from Laplace's equation and Ghosh [44] employed local elliptic and algebraic mesh generation methods. Generating mesh by solving differential equations may create a high quality mesh, especially when applying Poisson's equation [50] or biharmonic equation [51, 52], but it is time consuming and may introduce other types of errors in the process. Other techniques utilize algebraic interpolation and introduce a set of algebraic equations. Algebraic interpolation methods are generally simpler to apply, but they introduce curve fitting errors in the description of regions whose boundary may not be exactly described by polynomials of the same order as those appearing in the interpolation functions [53].

In this chapter, a new method is introduced to handle the supplementary constraint equations that are produced in a mesh motion algorithm [54]. In this method, the equations are processed and incorporated in the stiffness matrix on an element rather than global level. The element equations are then reduced before assembly so that the number of unknowns on the global level are kept same as updated Lagrangian method. This reduces the need for larger computer space and CPU time significantly. A mesh motion algorithm [55] is developed based on transfinite mapping method [53]. This technique provides a homogeneous mesh and matches the boundary of a given region at an infinite number of points, so that no curve fitting errors may be introduced. The new mesh location is directly determined by an explicit formula without solving any equation, which also expedites the calculation speed. Another distinct advantage of the transfinite mapping technique is that it allows the discrete representation of boundary curves and discontinuity of slope of boundary curves. This makes the method convenient and efficient since in ALE, the region boundaries would be normally expressed as discrete curves formed by nodes. Unlike the regular transfinite mapping scheme, the new procedure may take account of the motion of the boundary and may also adjust the nodes on boundaries to obtain a higher quality mesh.

3.1.2 Stress Integration

In finite deformation analysis, due to the continuous change in material configuration, the constitutive equation is usually expressed as a relation between some objective rates of stress and the rate of deformation tensor. The stress integration algorithms have to keep incremental objectivity, i.e., it has to be invariant with respect to superimposed rigid body motions within a given increment. For small deformation problems, the standard time-discretization procedures may be applied to rate constitutive equations without causing too much error, which makes the integration simple and straight forward. For finite deformation analysis, however, the standard time-discretization only achieves objectivity in the limit of vanishingly small time steps [3]. This may lead to excessive error accumulation in practice when a finite time increment is used. An algorithm for integrating rate constitutive equation where Jaumann rate is used is presented in [56] and later employed in ABAQUS [57]. One of the weak-points of the method is the requirement of additional computation of rotation tensor ${}_{t}^{t+\Delta t}\mathbf{R}$ from t to $t + \Delta t$. A more general discussion about the requirements that should be satisfied by an algorithm for the integration of rate constitutive equation is given by Pinsky [58]. A theoretical derivation of a general implicit integration algorithm is also presented in [58] by pure mathematical mapping between deformed and undeformed configuration. The consistency and incremental objectivity of the integration algorithms are clearly verified. Details of the numerical implementations are not, however, discussed.

Another aspect of the integration of rate constitutive equations is to satisfy the material plastic constitutive relations (e.g. Prandtl-Reuss equations) and the incremental plastic consistency, i.e., to keep the state of stress within the elastic domain or on the yield surface. In small deformation analysis, this may be achieved by a general return mapping method [59, 60].

For large deformation analysis, the return mapping methods should, however, be applied consistently with the integration algorithm. Numerical investigation of this point and consistent procedures for applying these algorithms are discussed in this chapter. The presented algorithms are combined consistently with the general return mapping method to satisfy the above mentioned requirements.

The incremental objectivity of stress integration algorithms needs the rate of deformation tensor ${}^{t}D_{ij}$ to be zero when only rigid body motion occurs [58]. It is generally achieved in implicit integration algorithms by using a central difference integration scheme, i.e., $\alpha = 0.5$ [56, 58], where α is the coefficient applied in the implicit integration schemes. In an explicit integration algorithm, a numerical treatment based on forward difference method ($\alpha = 0$) is necessary to keep the incremental objectivity. In this chapter, two explicit numerical integration algorithm for stress rate equation are derived based on the physical definition of stress instead of the purely mathematical tensor transformation presented in [58]. It is verified that the developed algorithms are equivalent to integration of Truesdell stress rate equation. One algorithm is exactly the same as the equivalent algorithm in [58], and another one is slightly different. Practically, however, it is shown that the two algorithms give almost exactly the same results [61].

3.1.3 Updating Material Associated Properties

In ALE simulation, the material associated properties, such as strains, stresses have to be updated after each incremental step because the mesh motion is independent of the material motion. Such updating is necessary in Eulerian method and updated Lagrangian method if re-meshing is applied. One method of achieving this in Eulerian formulation is to create an imaginary finite element mesh through the material points at time $t + \Delta t$ and use this mesh to calculate properties at integration or nodal points by interpolation [7]. This process is very time consuming and requires different interpolation logic every step and certain simplifications may be applied to the procedure [21, 22].

In updated Lagrangian formulation with remeshing capabilities, the updating of material associated properties is done through three steps [16]. First, properties at the old mesh nodal points are obtained from the old mesh integration points by extrapolation. This step is not consistent when integration points undergo material nonlinearity. A weighted nodal averaging may then be obtained. Normally, a direct relation between old and new meshes is not known. Therefore, in the second step, the old mesh element to which a new mesh node belongs is found. This is done by checking iteratively for the normalized local coordinates of the new mesh node in old mesh elements. If the absolute magnitudes of all the normalized coordinates are less than or equal to 1, then the old mesh element in which a new mesh node belongs may be deciphered. After that, the values of variables at the new mesh nodes may be defined by simple interpolation within the old mesh element. In the third step, the values at the integration points of the new mesh elements are determined by interpolation from the new nodal values.

In ALE formulation, researchers use an updating algorithm called “follower point " or “pseudo-material element " [39, 43]. In this method, all integration point variables are extrapolated to the nodal points of the “pseudo material elements" and a weighted averaging may be used. After that, iterations are used to find the “pseudo material element" to which mesh points will be related. The new nodal and integration point values are then calculated by interpolation within the “pseudo material element". The main disadvantage of this method is the use of one interpolation plus one extrapolation schemes. This may greatly degrade the accuracy of simulation and the calculated parameters. This procedures is basically similar to the one used after re-meshing in updated Lagrangian method.

Interpolation is necessary for various methods of updating. It is usually done by interpolation functions similar to those used for displacement or velocity in the finite element discretization [62]. A slightly different scheme is to create a continuous field for all variables by a method called “local smoothing with triangulation" [16], then, interpolate the variables linearly within one triangle. A different procedure for interpolation is developed by Derbalian [22] using conjugate approximation method [63]. By constructing a set of conjugate shape functions (bi-orthogonal) to those used to represent the displacement or velocity field, a consistent approximation for the stress field may be obtained which is continuous across inter-element boundaries and which involves less mean error. Because this method introduces a large system of equations, it is usually applicable to Eulerian formulation since the large system of equation need to be computed only once. For other cases, the interpolated elements are different in each step and the large system of equations must be solved iteratively during each incremental step.

In this research, an updating scheme based on continuum mechanics equations is used to update material associated properties. The relation between mesh motion and material motion is employed to avoid iterations required in previous schemes. The procedure is described in Appendix A. The method is more accurate, simpler to implement and eliminates the necessity of interpolations and extrapolations mentioned above.

3.2 STIFFNESS MATRIX PROCESSING

Using the standard finite element procedure to discretize the ALE Equation (2.20), we get,

$$ {}^t\mathbf{K}{}^t\mathbf{v} +{}^t \mathbf{K}^{c}{}^t\mathbf{v}^c = {}^t\mathbf{p}^{\cdot} \tag{3.1}$$

where, ${}^t\mathbf{v}$ is the vector of material velocities at nodal points, ${}^t\mathbf{v}^c$ is the vector of mesh velocities at nodal points, ${}^t\mathbf{K}$ is the stiffness matrix related to ${}^t\mathbf{v}$, ${}^t\mathbf{K}^c$is the stiffness matrix related to ${}^t\mathbf{v}^c$, ${}^t\mathbf{p}^{\cdot}$ is the external loading rate vector. In Equation (3.1), the total number of unknowns is twice the total number of D.O.F. and the equation number is only half of the number of unknowns. In order to solve the equations by conventional methods, supplementary equations equal to number of total D.O.F. are needed. These equations are supplied by the relations between ${}^t\mathbf{v}$ and ${}^t\mathbf{v}^c$, i.e., an explicit mesh motion scheme in ALE.

The basic idea is to avoid setting up the supplementary constraint equations to solve Equation (3.1) on a global level. Instead, we set up the constraint on the element level and reduce the equations by condensing out the mesh motion variables before assembly. This will reduce the global D.O.F. to be only the material velocities. On element level, Equation (3.1) may be written as:

$$ {}^t k_{IJ} {}^t v_J + {}^t k^c_{IJ} {}^t v^c_J = {}^t p^{\cdot}_I \qquad (I = 1, 2, ...n) \tag{3.2} $$

where, n is the number of D.O.F. in the element. We use a transfinite mapping for mesh velocities, ${}^t v^c_J$ in the following general form:

$$ {}^t v^c_J = a_J + b_{(J)} {}^t v_{(J)} \qquad (J = 1, 2, ...n) \tag{3.3} $$

where, no summation on “J" is observed.

It should be noted that the mesh motion scheme of Equation (3.3) guarantees one-to-one mapping between material domain and computational reference domain, as verified in Appendix C and that the choice of the coefficients a_J and b_J will identify the type of formulation as follow:

If $a_J = b_J = 0$, the formulation reduces to Eulerian one,

If $a_J = 0$ and $b_J = 1$, the formulation reduces to Lagrangian, and

Otherwise, the formulation will be a general ALE one.

The above three cases may be mixed in a given problem so that certain regions of the model may have different formulation. It is also important to realize that at each node, different D.O.F.s are allowed to have different formulation type. This property makes handling contact boundary condition much easier. The only apparent constraint here is that the boundary points should be always kept as Lagrangian type in the direction normal to the boundary so that the mesh and material points will always represent the same boundary.

Substituting Equation (3.3) into Equation (3.2), we get:

$$ ({}^t k_{IJ} + {}^t k^c_{I(J)} b_{(J)}) {}^t v_J = {}^t p^{\cdot}_I - {}^t k^c_{IJ} a_J \tag{3.4} $$

where there is no summation on (J) in ${}^t k^c_{I(J)} b_{(J)}$. In matrix form, the equation may be written as:

$$ {}^t\overline{\mathbf{K}}\,{}^t\mathbf{v} = {}^t\overline{\mathbf{p}^{\cdot}} \qquad (3.5)$$

where ${}^t\overline{\mathbf{K}}$ is equivalent stiffness matrix and ${}^t\overline{\mathbf{p}^{\cdot}}$ is equivalent load rate vector. In Equation (3.4), the only unknowns are the material velocities ${}^t v_J$. Thus, by introducing the supplementary constraint equations on the element level and modifying the elemental stiffness matrices, the mesh velocities may be condensed out of the element equilibrium equations, so that the number of unknowns will be finally the same as in the traditional finite element formulation.

In Equation (3.4), the equivalent stiffness matrix is ${}^t k_{IJ} + {}^t k^c_{I(J)} b_{(J)}$ and equivalent element load is ${}^t f^{\cdot}_I - {}^t k^c_{IJ} a_J$. The conventional finite element assembly and elimination method may now be applied directly to solve for material velocities ${}^t v_J$. This procedure is more efficient and the usual finite element solution routines may be used with minor changes. The only limitation to the procedure is that the mesh velocities may be coupled with the material velocities only at the same D.O.F.. In practice, such limitation is trivial, because more complicated relations between material and mesh velocities may not guarantee a better mesh quality, but will generally increase the computation time significantly. If for some particular reason, the mesh velocity ${}^t v^c_J$ has to be coupled with the D.O.F. in other elements, the above method is still applicable. Only difference is that processing would create “fictitious" nodes whose D.O.F. is coupled by the mesh velocity equations and which are not connected with the element physically.

3.3 MESH MOTION SCHEME IN ALE

3.3.1 Transfinite Mapping Method

To complete the mesh motion procedure and finalize the supplementary constraint equations, it is essential to decide the values of the coefficient a_I and b_I in Equation (3.3). As discussed above, on boundaries of a deformation domain, the nodes have to be Lagrangian type in the normal direction in order to keep a one-to-one mapping between the material and computational reference domain, i.e., $a_I = 0, b_I = 1$. These boundary points may move, however, in the tangential direction with a velocity different from the mesh velocity. All the internal nodes are assumed as general ALE points moving at speed ${}^t v_I^c = a_I$, which is specified by transfinite mapping method, and the coefficients b_I are set to zero for these general points.

The transfinite mapping method is employed to decide the internal nodal speed [55]. Originally, transfinite mapping is applied to create a mesh on a geometric domain when the boundaries are specified [53]. Assuming a region with four boundary curves specified as $\phi_i(r,0), \phi_i(r,1), \phi_i(0,s)$ and $\phi_i(1,s)$ as shown schematically in Figure 3.1, the mesh coordinate c_i is given by:

$$\begin{aligned} c_i(r,s) &= (1-s)\phi_i(r,0) + s\phi_i(r,1) + (1-r)\phi_i(0,s) + r\phi_i(1,s) \\ &- (1-r)(1-s)\phi_i(0,0) - (1-r)s\phi_i(0,1) - rs\phi_i(1,1) \\ &- r(1-s)\phi_i(1,0) \qquad (3.6) \\ &\qquad 0 \le r \le 1 \qquad\qquad 0 \le s \le 1 \end{aligned}$$

where r, s are normalized coordinates, $i = x, y$, i.e., $\phi_x(r,0)$ and $\phi_y(r,0)$ etc., represent the x, y coordinates of boundary curves, and $c_x(r,s)$, $c_y(r,s)$ are nodal x and y coordinates of a point, respectively.

The transfinite mapping Equation (3.6) will create a homogeneous mesh when the nodes on the boundaries are distributed homogeneously, and it will allow the boundaries to be represented in a discrete form, e.g., with nodal coordinates on the boundaries. This is a characteristic that is quite convenient and useful in ALE formulation because at each step of simulation, the deformed boundaries are actually described by discrete nodes. The only requirement of the discrete

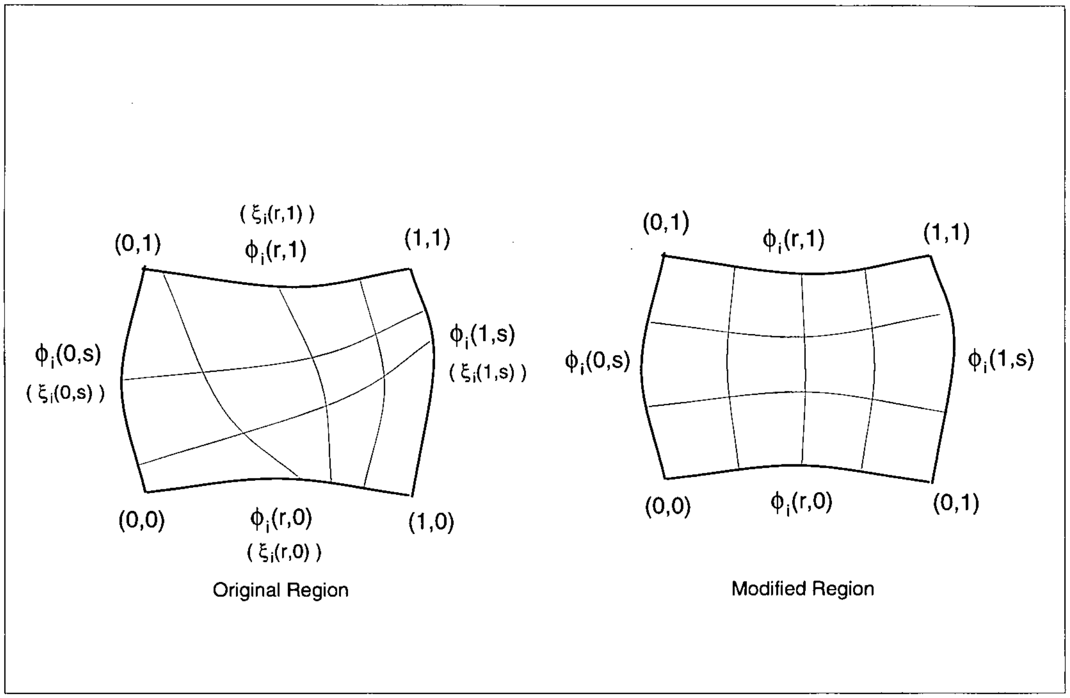


Figure 3.1: Transfinite mapping

representations in the transfinite method is to have normalized parametric coordinates assigned to each node on boundary curves. This process is automated by assigning the coordinate $\frac{i-1}{k}$ to the *ith* point on a curve containing $(k+1)$ points. The location of internal nodal points are completely determined by the position of boundary nodes. The boundary curves may be described by as many nodes as needed, so that curve fitting error in parametric mapping can be minimized or avoided. In general, we assume that at a given incremental step, at time t, the finite element mesh is distorted. In the next incremental step from time t to $t + \Delta t$, the mesh will be automatically moved by the program to get a homogeneous one. The new internal nodal coordinates are calculated by Equation (3.6). If the old (distorted) coordinates are $c_i^0(r,s)$, then the mesh velocity is given by:

$$ {}^{t}v_I^{ct} = \frac{c_i(r,s) - c_i^0(r,s)}{\Delta t} \tag{3.7} $$

$$ a_I = {}^{t}v_I^{ct} \tag{3.8} $$

Figure 3.2 gives an example for distorted and automatically modified mesh obtained by applying the above method. It should be noted that, even when the boundaries have discontinuous slopes, for example at A, B and C in Figure 3.2, the method produced a homogeneous mesh.

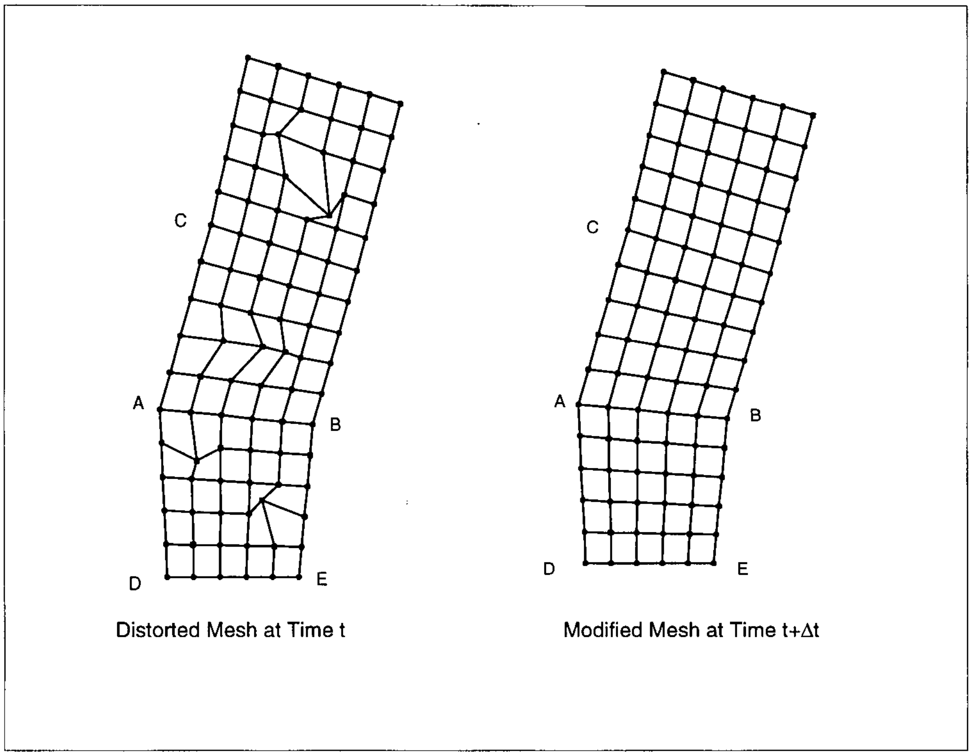


Figure 3.2: Mesh modification

3.3.2 Consideration of Boundary Motion

The above mesh motion scheme will guarantee a homogeneous mesh when boundaries do not move. In a general ALE formulation, however, mesh motion and loading increment occur simultaneously and the boundaries should be allowed to move. In such case, in order to achieve a higher quality mesh, Equation (3.8), has to be modified. First we assume that the boundary velocities are expressed by $\xi_i(r,0), \xi_i(r,1), \xi_i(0,s)$ and $\xi_i(1,s)$ corresponding to boundaries

$\phi_i(r,0), \phi_i(r,1), \phi_i(0,s)$ and $\phi_i(1,s)$ individually, as shown in Figure 3.1. We use the same function as in Equation (3.6) to interpolate the boundary velocities and find the modification term to be used in Equation (3.8);

$$\begin{aligned} {}^t v_I^{cb} &= (1-s)\xi_i(r,0) + s\xi_i(r,1) + (1-r)\xi_i(0,s) + r\xi_i(1,s) \\ &- (1-r)(1-s)\xi_i(0,0) - (1-r)s\xi_i(0,1) - rs\xi_i(1,1) \\ &- r(1-s)\xi_i(1,0) \end{aligned} \qquad (3.9)$$

$$0 \le r \le 1 \qquad\qquad 0 \le s \le 1$$

To ensure that the boundary nodes have to be Lagrangian type in the normal direction, the mesh velocities $\xi_i(r,0), \xi_i(r,1), \xi_i(0,s)$ and $\xi_i(1,s)$ have to be equal to the material velocities ${}^t v_i(r,0),{}^t v_i(r,1),{}^t v_i(0,s)$, and ${}^t v_i(1,s)$; or at least their normal components have to be equal to each other. These material velocities at time t , however, may be unknown, for example as on free surfaces. In order to avoid solving implicit equations and to calculate the normal components, we apply the material velocities at time $(t-\Delta t)$ to approximate the values at time t during the mesh motion. It should be indicated that such assumption is only for mesh motion and does not have any direct effect on the accuracy of solution. Therefore, for Lagrangian boundary points;

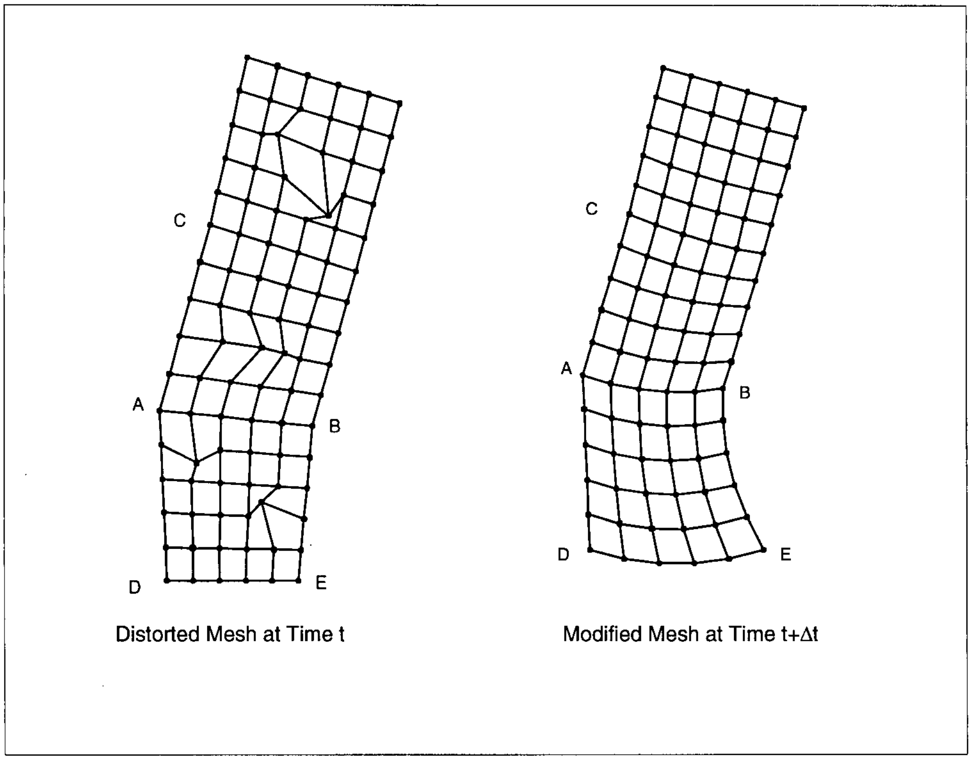
$$\begin{aligned} &\xi_i(r,0) = {}^{t-\Delta t}v_i(r,0) &&\xi_i(r,1) = {}^{t-\Delta t}v_i(r,1) \\ &\xi_i(0,s) = {}^{t-\Delta t}v_i(0,s) &&\xi_i(1,s) = {}^{t-\Delta t}v_i(1,s) \\ &(i = x, y \qquad 0 \le r \le 1 &&0 \le s \le 1) \end{aligned} \qquad (3.10)$$

If higher quality mesh is required, some iterations may be introduced, i.e., using the new material velocities to replace the velocities at time $(t-\Delta t)$ in Equation (3.10) until the difference between the material velocities from the two iterations is less than a given tolerance value. In practical applications, this iteration was found to be not necessary since the mesh quality created by

Equation (3.10) is good enough for most of practical problems. Finally the mesh motion for the nodes in the interior of the domains is given by:

$$ {}^t v_I^c = {}^t v_I^{ct} + {}^t v_I^{cb} = a_I \tag{3.11} $$

For the example shown in Figure 3.2, if the distorted mesh at time t has boundaries DE and BE moving at some inhomogeneous speed, using the above mesh motion scheme, we get the mesh at time $t + \Delta t$ as shown in Figure 3.3. So, even when boundaries move at inhomogeneous speeds, which is the usual case in finite element large deformation problems, the scheme may still give homogeneous meshes.



Distorted Mesh at Time t

Modified Mesh at Time t+Δt

Figure 3.3: The mesh motion in ALE

3.3.3 The Motion of Nodes on Boundaries

The boundaries of deformation domain may be classified as one of two categories. The first kind posses no change in boundary shape during the deformation and the shape is known a prior. An example to this type is a boundary with prescribed velocity or displacement conditions, e.g., the tool work-piece interface in indentation or punch forging problems. In the second kind, the deformed boundary shape is unknown in advance and changes during deformation. The boundaries with applied traction forces often lie in this category. Motion of nodes on the first type of boundaries is easy to decide. Because the normal direction of boundaries is known, the mesh velocity in the normal direction should be equal to material velocity, i.e., $a_I = 0, b_I = 1$ in Equation (3.3). In the tangential direction, the node may be moved arbitrarily in order to achieve calculation efficiency. Without losing generality, b_I is set to zero and a_I may be determined in a way specified by the user, or usually in such a way that the nodes are evenly distributed along the boundaries.

On the second type of boundaries, alternative methods have to be employed due to unknown velocity in the direction normal to the boundary. One way of handling this is to transform the second type of boundaries into first type by assuming the normal direction. The disadvantage here is that iterations have to be introduced to make the difference between the assumed normal direction and the calculated one from equilibrium equation less than the preset tolerance. Another method specifies the nodes on the second type of boundaries as Lagrangian points in all directions, i.e., $a_I = 0$ and $b_I = 1$ in Equation (3.3). This may result in unexpected or inhomogeneous spacing because of the deformation. In order to have homogeneous mesh pattern, the boundary nodes have to be adjusted in the tangential direction of the boundary. This kind of adjustment can not be done during the incremental step and has to be done after the deformed boundary shape is determined. In this work, the nodes are relocated on the boundaries according to nodal spacing specified by user or are positioned to get an equal distance between each node.

3.3.4 Implementation and Discussion

The procedures of mesh motion for an incremental step from time t to time $t + \Delta t$ may be summarized in the following points:

- Get mesh coordinates at time t and the boundary velocities at time $t - \Delta t$,
- From the input data, identify the Lagrangian and Eulerian direction for each boundary node and decide the coefficients "a_I" and "b_I" for each D.O.F.,
- Calculate the value "a_I" for each internal node according to the boundary coordinates and velocities, using Equations (3.6)-(3.11),
- Create the element stiffness matrices, ${}^t k_{IJ}$ and ${}^t k^c_{IJ}$,
- Perform the transformation given by Equation (3.4) to get the equivalent stiffness matrix and load vector,
- Assemble the equivalent element stiffness matrices and load vector and solve the assembled equations to obtain material velocities, displacements, stresses, etc.,
- Adjust the position of nodes on the second type boundaries,
- Update the material associated properties and mesh coordinates.

The application of transfinite mapping method in ALE for mesh motion has many advantages. In comparison with other methods discussed above, the transfinite mapping is efficient, simpler to implement and does not introduce curve fitting errors beyond the discretization error that is inherent to the finite element method. The method allows the boundary curves to be represented in discrete form and to have discontinuous slopes. These properties make the method directly applicable to domain with irregular polygon boundaries. In transfinite mapping method, the interior node locations are calculated from boundary nodes. The curvatures of the boundary curves and spacing of boundary nodes are accurately reflected in the mesh, as may be shown in Figure 3.3. Furthermore, the boundary curves do not need to be simple functions. Cusps and inflection points provide no special problems.

The node motion scheme used in the research for the second type of boundary is simple and more accurate than assuming a normal direction. It is not limited by boundary curve types and it may be combined with transfinite mapping method to handle the mesh motion in the whole deformation domain. Node adjustment in tangential direction on the second kind boundaries is necessary to keep a higher quality mesh.

Stiffness matrix processing discussed above is a general method. That may be used with various material models, i.e., rate dependent or independent.

3.4 INTEGRATION OF STRESS RATE

3.4.1 The Integration Algorithm

In large deformation analysis, because of the fact that the configuration of the body is changing continuously, the Cauchy stress can not be integrated simply by adding the stresses and their increments due to deformation directly [47]. The consideration of configuration changing can be incorporated by purely mathematical tensor transformation as given in [58], or alternatively by utilizing the physical definition of Cauchy stress as presented in the following.

Assume the material configuration at time t is ${}^{t}C$ and the corresponding Cauchy stress is ${}^{t}\sigma_{ij}$. At time $(t+\Delta t)$, the configuration is ${}^{t+\Delta t}C$ and the Cauchy stress is ${}^{t+\Delta t}\sigma_{ij}$. As shown in Figure 3.4, the 2nd Piola-Kirchhoff stress at $(t+\Delta t)$ with respect to the configuration at time

t , ${}_{t}^{t+\Delta t}S_{ij}$, is related to Cauchy stress ${}_{t}^{t+\Delta t}\sigma_{ij}$, at time $t + \Delta t$ by [47]:

$$^{t+\Delta t}\sigma_{ij} = \frac{1}{\left|{}_{t}^{t+\Delta t}\mathbf{F}\right|} \frac{\partial^{t+\Delta t}x_i}{\partial^t x_m} {}_{t}^{t+\Delta t}S_{mn} \frac{\partial^{t+\Delta t}x_j}{\partial^t x_n} \tag{3.12}$$

where ${}_{t}^{t+\Delta t}F_{ij} = \frac{\partial^{t+\Delta t}x_i}{\partial^t x_j}$ is the deformation gradient tensor or the Jacoban matrix.

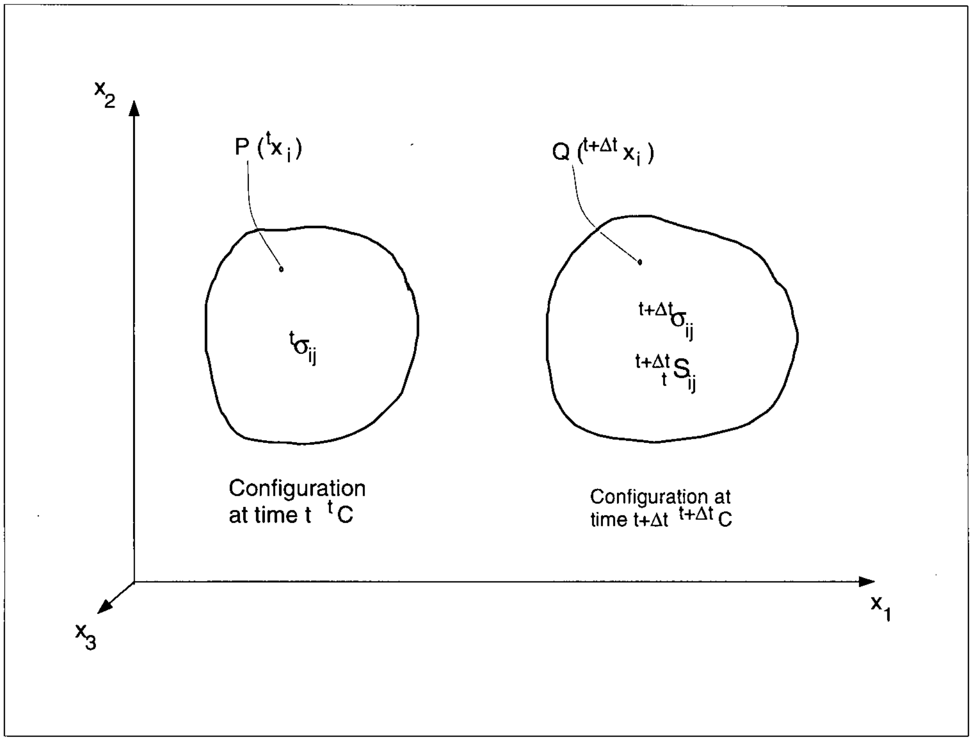


Figure 3.4: The material configurations

Consider the deformation to be decomposed of two steps; first one is only a rigid body motion and the second one is only straining. Then after the first step, the stress relation may be written as:

$$^{t+\Delta t}\sigma_{ij}^{(I)} = \frac{1}{\left|{}_{t}^{t+\Delta t}\mathbf{F}\right|} \frac{\partial^{t+\Delta t}x_i}{\partial^t x_m} {}_{t}^{t+\Delta t}S_{mn}^{(I)} \frac{\partial^{t+\Delta t}x_j}{\partial^t x_n} \tag{3.13}$$

and since this step is only a rigid body motion, then,

$${}_{t}^{t+\Delta t}S_{mn}^{(I)} = {}_{t}^{t}S_{mn} = {}^{t}\sigma_{mn}$$

where ${}^{t+\Delta t}\sigma_{ij}^{(I)}$ and ${}^{t+\Delta t}_{t}S_{mn}^{(I)}$ are Cauchy and 2nd Piolar-Kirchhoff stress at an imaginary intermediate state between t and $t+\Delta t$, respectively, so that Equation (3.13) may be written in the form:

$$ {}^{t+\Delta t}\sigma_{ij}^{(I)} = \frac{1}{\left|{}^{t+\Delta t}_{t}\mathbf{F}\right|} \frac{\partial^{t+\Delta t}x_i}{\partial^{t}x_m} {}^{t}\sigma_{mn} \frac{\partial^{t+\Delta t}x_j}{\partial^{t}x_n} $$

i.e., this step only considers the effect of configuration change on Cauchy stress tensor.

In the second step, due to straining, the Cauchy stress would have an increment of

$$ \Delta\sigma_{ij} = C_{ijkl}{}^{t}D_{kl}\Delta t \tag{3.14} $$

and the stress at time $(t+\Delta t)$ would be:

$$ {}^{t+\Delta t}\sigma_{ij} = {}^{t+\Delta t}\,\sigma_{ij}^{(I)} + \Delta\sigma_{ij} $$

i.e.,

$$ {}^{t+\Delta t}\sigma_{ij} = \frac{1}{\left|{}^{t+\Delta t}_{t}\mathbf{F}\right|} \frac{\partial^{t+\Delta t}x_i}{\partial^{t}x_m} {}^{t}\sigma_{mn} \frac{\partial^{t+\Delta t}x_j}{\partial^{t}x_n} + \Delta\sigma_{ij} \tag{3.15} $$

or,

$$ {}^{t+\Delta t}\sigma_{ij} = \frac{1}{\left|{}^{t+\Delta t}_{t}\mathbf{F}\right|} \frac{\partial^{t+\Delta t}x_i}{\partial^{t}x_m} {}^{t}\sigma_{mn} \frac{\partial^{t+\Delta t}x_j}{\partial^{t}x_n} + C_{ijkl}{}^{t}D_{kl}\Delta t \tag{3.16} $$

where, C_{ijkl} is the element constitutive tensor,

Δt is the time increment,

${}^{t}D_{kl}$ is the rate of deformation tensor, ${}^{t}D_{kl} = \frac{1}{2}\left(\frac{\partial^{t}v_k}{\partial^{t}x_l} + \frac{\partial^{t}v_l}{\partial^{t}x_k}\right)$.

In decomposion of the deformation, if the first step and the second step are switched, i.e., if the first step is only straining and the second one is only rigid body motion, we get another integration scheme as shown in Appendix D.

3.4.2 Consistency of the Integration Algorithm

Integration algorithms have to be consistent with the constitutive equations. It may be shown that the integration algorithm developed in Equation (3.16) is equivalent to using Truesdell stress rate in the rate constitutive equation.

We consider the determinant of the deformation gradient tensor as:

$$\left|{}^{t+\Delta t}_{t}\mathbf{F}\right| \approx \left|{}^{t}_{t}\mathbf{F}\right| + d\left|{}^{t}_{t}\mathbf{F}\right| = \left|{}^{t}_{t}\mathbf{F}\right| + \left|{}^{t}_{t}\mathbf{F}\right| \frac{\partial u_k}{\partial^t x_k}$$

where, $|{}^{t}_{t}\mathbf{F}| = 1$ and $u_k = {}^{t+\Delta t}x_k - {}^t x_k$ is the displacements from time t to $t + \Delta t$, so that;

$$\frac{1}{\left|{}^{t+\Delta t}_{t}\mathbf{F}\right|} = \frac{1}{1 + \frac{\partial u_k}{\partial^t x_k}} = \frac{1 - \frac{\partial u_k}{\partial^t x_k}}{1 - \frac{\partial u_m}{\partial^t x_m}\frac{\partial u_n}{\partial^t x_n}} \approx 1 - \frac{\partial u_k}{\partial^t x_k}$$

with the higher orders of u_i being ignored. Substituting the above equation into Equation (3.16), gives:

$${}^{t+\Delta t}\sigma_{ij} = \left(1 - \frac{\partial u_k}{\partial^t x_k}\right) \frac{\partial\left({}^t x_i + u_i\right)}{\partial^t x_m} {}^t\sigma_{mn} \frac{\partial\left({}^t x_j + u_j\right)}{\partial^t x_n} + C_{ijkl}{}^t D_{kl}\Delta t$$

Rearranging the above equation and neglecting all the higher order terms of u_i; we obtain:

$$\begin{aligned}
{}^{t+\Delta t}\sigma_{ij} &= \frac{\partial^t x_i}{\partial^t x_m}{}^t\sigma_{mn}\frac{\partial^t x_j}{\partial^t x_n} + \frac{\partial u_i}{\partial^t x_m}{}^t\sigma_{mn}\frac{\partial^t x_j}{\partial^t x_n} + \frac{\partial^t x_i}{\partial^t x_m}{}^t\sigma_{mn}\frac{\partial u_j}{\partial^t x_n} \\
&- \frac{\partial^t x_i}{\partial^t x_m}{}^t\sigma_{mn}\frac{\partial^t x_j}{\partial^t x_n}\frac{\partial u_k}{\partial^t x_k} + C_{ijkl}{}^t D_{kl}\Delta t \\
&= \delta_{im}{}^t\sigma_{mn}\delta_{nj} + \frac{\partial u_i}{\partial^t x_m}{}^t\sigma_{mn}\delta_{nj} + \delta_{im}{}^t\sigma_{mn}\frac{\partial u_j}{\partial^t x_n} \\
&- \delta_{im}{}^t\sigma_{mn}\delta_{nj}\frac{\partial u_k}{\partial^t x_k} + C_{ijkl}{}^t D_{kl}\Delta t
\end{aligned}$$

or,

$$\frac{{}^{t+\Delta t}\sigma_{ij} - {}^t\sigma_{ij}}{\Delta t} = \frac{1}{\Delta t}\left(\frac{\partial u_i}{\partial^t x_m}{}^t\sigma_{mj} + {}^t\sigma_{in}\frac{\partial u_j}{\partial^t x_n} - {}^t\sigma_{ij}\frac{\partial u_k}{\partial^t x_k}\right) + C_{ijkl}{}^t D_{kl}$$

which, upon taking the limit when $\Delta t \to 0$, gives:

$${}^t\dot{\sigma}_{ij} = \frac{\partial^t v_i}{\partial^t x_k}{}^t\sigma_{kj} + \frac{\partial^t v_j}{\partial^t x_k}{}^t\sigma_{ik} - \frac{\partial^t v_k}{\partial^t x_k}{}^t\sigma_{ij} + C_{ijkl}{}^t D_{kl} \tag{3.17}$$

where ${}^{t}v_i$ is the velocity component in the ith direction of the coordinates. By definition, the Truesdell rate of Cauchy stress may be given by Equation (2.23). Therefore, we may write Equation (3.17) in the following form:

$$ {}^{t}\sigma_{ij}^{T\cdot} = C_{ijkl}{}^{t}D_{kl} \tag{3.18} $$

The above simple development indicates that the stress integration algorithm derived from the physical definition of the Cauchy stress is actually the same as the integration of Truesdell stress rate equation.

3.4.3 Alternative Bases for Integration Schemes

(i) Integration of Truesdell Stress Rate

The material configuration at time t is denoted by ${}^{t}C$ and at time $t + \Delta t$ by ${}^{t+\Delta t}C$. A material point in both configuration may be given by the position vectors $P({}^{t}x_1, {}^{t}x_2, {}^{t}x_3)$ and $Q({}^{t+\Delta t}x_1, {}^{t+\Delta t}x_2, {}^{t+\Delta t}x_3)$, as shown in Figure 3.4, respectively. Tensor quantities in both configurations ${}^{t}C$ and ${}^{t+\Delta t}C$ may be related to each other by proper transformation. Starting from this point, Pinsky et al. [58] derived the implicit integration algorithms for Truesdell stress rate, Jaumann rate and Green-Naghdi rate. The derivation is purely mathematical and according to their development for Truesdell rate, the algorithm is generally implicit and may be expressed as follow:

$$
\begin{aligned}
{}^{t+\Delta t}\sigma_{ij} &= \frac{1}{\left|{}^{t+\Delta t}_{t}\mathbf{F}\right|}\frac{\partial^{t+\Delta t}x_i}{\partial^{t}x_m}{}^{t}\sigma_{mn}\frac{\partial^{t+\Delta t}x_j}{\partial^{t}x_n} \\
&+ \frac{\Delta t}{\left|{}^{t+\Delta t}_{t+\alpha\Delta t}\mathbf{F}\right|}\frac{\partial^{t+\Delta t}x_i}{\partial^{t+\alpha\Delta t}x_m}\, C_{mnkl}\big|_{t+\alpha\Delta t}\,{}^{t}_{t+\alpha\Delta t}D_{kl}\frac{\partial^{t+\Delta t}x_j}{\partial^{t+\alpha\Delta t}x_n}
\end{aligned} \tag{3.19}
$$

where,

$$ {}^{t+\alpha\Delta t}x_i = \alpha^{t+\Delta t}x_i + (1-\alpha)^{t}x_i $$

$$
\begin{aligned}
{}^{t+\Delta t}_{t+\alpha\Delta t}F_{ij} &= \frac{\partial^{t+\Delta t}x_i}{\partial^{t+\alpha\Delta t}x_j} \\
{}^{t}_{t+\alpha\Delta t}D_{ij} &= \frac{1}{2}\left(\frac{\partial^{t}v_i}{\partial^{t+\alpha t}x_j} + \frac{\partial^{t}v_j}{\partial^{t+\alpha t}x_i}\right)
\end{aligned}
$$

when $\alpha = 0$, it will be explicit and may be expressed as:

$$
{}^{t+\Delta t}\sigma_{ij} = \frac{1}{\left|{}^{t+\Delta t}_{t}\mathbf{F}\right|}\frac{\partial^{t+\Delta t}x_i}{\partial^{t}x_m}{}^{t}\sigma_{mn}\frac{\partial^{t+\Delta t}x_j}{\partial^{t}x_n} + \frac{1}{\left|{}^{t+\Delta t}_{t}\mathbf{F}\right|}\frac{\partial^{t+\Delta t}x_i}{\partial^{t}x_m}\Delta\sigma_{mn}\frac{\partial^{t+\Delta t}x_j}{\partial^{t}x_n} \tag{3.20}
$$

where, $\Delta\sigma_{mn}$ is defined as in Equation (3.14). Upon comparing Equation (3.20) to Equation (3.15), we find that the only difference is in the second term of the right hand side and if the higher order term of u_i is neglected, the difference will vanish. Numerical experiments show, however, that both equations give almost identical results. In addition, it is necessary to indicate that by assuming that the first step in the deformation is straining and the second is a rigid body motion, as shown in Appendix D, we would get the exact same result as Equation (3.20).

(ii) Integration of Jaumann Stress Rate

An integration algorithm for Jaumann rate is proposed by Hughes [56]. The purpose of the development in Hughes's work is to keep the "objectivity" of the integration. The derivation is based on purely geometrical concepts and the result may be expressed by :

$$
{}^{t+\Delta t}\sigma_{ij} = {}^{t+\Delta t}_{t}R_{im}\,{}^{t}\sigma_{mn}\,{}^{t+\Delta t}_{t}R_{jn} + \Delta\sigma_{ij} \tag{3.21}
$$

where, ${}^{t+\Delta t}_{t}R_{ij}$ is the rotation tensor for the increment from time t to $t + \Delta t$;
$\Delta\sigma_{ij}$ is the stress increment given by:

$$
\Delta\sigma_{ij} = C_{ijkl}\,{}^{t}_{t+0.5\Delta t}D_{kl}\Delta t
$$

The ${}^{t}_{t+0.5\Delta t}D_{kl}$ is calculated by central difference method:

$$
{}^{t}_{t+0.5\Delta t}D_{kl} = \frac{1}{2}\left(\frac{\partial^{t}v_k}{\partial^{t+0.5\Delta t}x_l} + \frac{\partial^{t}v_l}{\partial^{t+0.5\Delta t}x_k}\right) \tag{3.22}
$$

and

$$^{t+0.5\Delta t}x_i = 0.5^{t+\Delta t}x_i + 0.5^{t}x_i$$

If displacement $u_i = {}^{t+\Delta t}x_i - {}^{t}x_i$ and the gradient of displacement with respect to ${}^{t+0.5t}x_i$ is:

$$G_{ij} = \frac{\partial u_i}{\partial^{t+0.5\Delta t}x_j}$$

${}^{t+\Delta t}_{\ t}\mathbf{R}$ can be expressed in matrix form as [56]:

$${}^{t+\Delta t}_{\ t}\mathbf{R} = \left(\mathbf{I} + \frac{\mathbf{G} - \mathbf{G}^T}{4}\right)\left(\mathbf{I} - \frac{\mathbf{G} - \mathbf{G}^T}{4}\right)^{-1}$$

where, $\mathbf{I}$ is unit matrix and $\mathbf{G}$ has components of G_{ij}.

Upon comparing Equation (3.21) and Equation (3.15), it is important to note that in Equation (3.21) the stress at time t is transformed by a rotation tensor ${}^{t+\Delta t}_{\ t}R_{ij}$ whereas in Equation (3.15), it is transformed by deformation gradient tensor, ${}^{t+\Delta t}_{\ t}F_{ij}$. For a pure rigid body motion, the Truesdell rate is the same as Jaumann rate, the deformation gradient tensor reduces to the rotation tensor and determinant of deformation gradient tensor is equal to unity, so that Equation (3.15) and Equation (3.21) lead to the same results. The objectivity of the algorithms have already been verified in [56] and [58].

3.5 INCREMENTAL OBJECTIVITY AND NUMERICAL TREATMENT OF ${}^{t}\mathrm{D}_{ij}$

3.5.1 General Considerations

The rate of deformation tensor is not only used in integration of rate type constitutive equations as in Equations (3.15, 3.20 and 3.21), but it is also used in other applications, such as visco-elasticity and creep involving large strains and rotations. The incremental objectivity of ${}^{t}D_{ij}$ means that under finite incremental step, its components will be independent of any rigid body rotation. For large deformation analysis, this is shown to be achieved when the integration scheme is used with a value $\alpha = 0.5$ (central difference method) in implicit stress integration algorithms [58]. In many applications, however, ${}^tD_{ij}$ is employed separately, not with stress integration, or is used with an explicit stress integration, e.g., Equation (3.16). In this section it is shown that using the conventional forward difference method in such analyses may only achieve objectivity when the time steps are very small and it may lead to excessive error accumulation in practice. A forward difference method with a correction term is presented to keep the incremental objectivity in situations where rigid body rotation is presented.

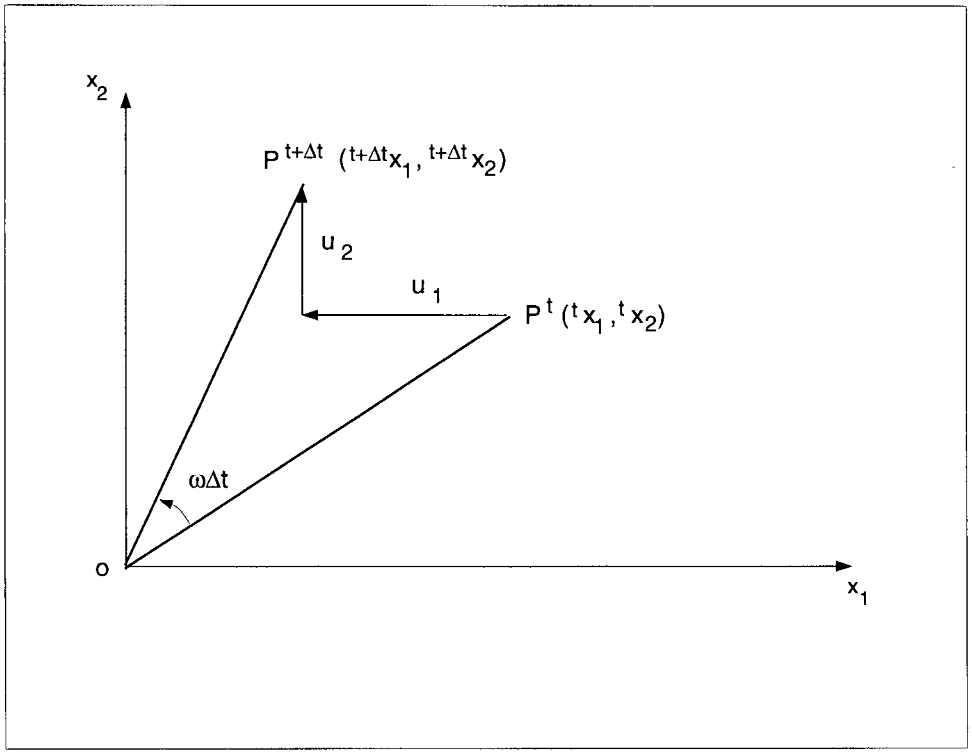

x_2
$P^{t+\Delta t}$ $({}^{t+\Delta t}x_1, {}^{t+\Delta t}x_2)$
u_2
u_1
P^t $({}^tx_1, {}^tx_2)$
$\omega\Delta t$
o
x_1

Figure 3.5: Rigid body rotation of point P^t

It may be instructive to show first that analytically, the components of the rate of deformation tensor do identically vanish in the case of rigid body rotation. This may be achieved by considering a general two-dimensional rigid body rotation with an angular velocity ω. If the rotation center O is chosen as the origin of a fixed Cartesian coordinate system, a material particle located at $P^t({}^tx_1, {}^tx_2)$ at time t moves to a new position $P^{t+\Delta t}({}^{t+\Delta t}x_1, {}^{t+\Delta t}x_2)$ at time $t + \Delta t$, as shown in Figure 3.5. The new position may be expressed in the following relations:

$$\begin{aligned} {}^{t+\Delta t}x_1 &= {}^tx_1 cos(\omega\Delta t) - {}^tx_2 sin(\omega\Delta t) \\ {}^{t+\Delta t}x_2 &= {}^tx_1 sin(\omega\Delta t) + {}^tx_2 cos(\omega\Delta t) \end{aligned} \tag{3.23}$$

so that, the displacements are given by:

$$\begin{aligned} u_1 &= {}^tx_1(cos(\omega\Delta t) - 1) - {}^tx_2 sin(\omega\Delta t) \\ u_2 &= {}^tx_1 sin(\omega\Delta t) + {}^tx_2(cos(\omega\Delta t) - 1) \end{aligned} \tag{3.24}$$

Taking derivatives with respect to Δt and calculating the limit values when $\Delta t \rightarrow 0$, gives the velocities at time t as:

$${}^tv_1 = -\omega {}^tx_2 \qquad\qquad {}^tv_2 = \omega {}^tx_1$$

From the definition of ${}^tD_{ij}$, the components of the rate of deformation tensor ${}^t\mathbf{D}$ at time t is given by:

$$\begin{aligned} {}^tD_{11} &= \frac{\partial {}^tv_1}{\partial {}^tx_1} = 0 \\ {}^tD_{22} &= \frac{\partial {}^tv_2}{\partial {}^tx_2} = 0 \\ {}^tD_{12} &= \frac{1}{2}\left(\frac{\partial {}^tv_1}{\partial {}^tx_2} + \frac{\partial {}^tv_2}{\partial {}^tx_1}\right) = 0 \end{aligned} \tag{3.25}$$

This verifies that the analytical values of the rate of deformation tensor are identically zero for the case of rigid body rotation, i.e., the tensor ${}^t\mathbf{D}$ is objective with respect to rigid body rotation.

3.5.2 Forward Difference Algorithm to Calculate ${}^t\mathbf{D}_{ij}$

In numerical and finite element analysis, it may not be possible to get the analytical values as indicated above and, in general, a time-discretization procedure is applied to calculate ${}^tD_{ij}$. The conventional method in finite element analysis is to use the average velocity from time t to time $t + \Delta t$ as velocity at time t, and employ this to calculate ${}^tD_{ij}$ as the rate of deformation tensor at time t. The calculated value of ${}^tD_{ij}$ is also used as the average value of the tensor from time t to time $t + \Delta t$. Mathematically, this is equivalent to a forward difference method. Thus, the velocities are given by;

$$\begin{aligned} {}^tv_1 &= \frac{u_1}{\Delta t} = \frac{{}^tx_1(cos(\omega\Delta t) - 1) - {}^tx_2 sin(\omega\Delta t)}{\Delta t} \\ {}^tv_2 &= \frac{u_2}{\Delta t} = \frac{{}^tx_1 sin(\omega\Delta t) + {}^tx_2(cos(\omega\Delta t) - 1)}{\Delta t} \end{aligned} \tag{3.26}$$

the ${}^tD_{ij}$ components are then obtained by taking derivatives with respect to ${}^tx_1$ and ${}^tx_2$ as follow;

$$\begin{aligned} {}^tD_{11} &= \frac{cos(\omega\Delta t) - 1}{\Delta t} \\ {}^tD_{22} &= \frac{cos(\omega\Delta t) - 1}{\Delta t} \\ {}^tD_{12} &= 0 \end{aligned} \tag{3.27}$$

which shows that ${}^tD_{11}$ and ${}^tD_{22}$ are not equal to zero except when the time step is very small. A finite time increment may cause errors in the calculation of the components of the tensor ${}^t\mathbf{D}$, especially when the numerical values of these components are of the same order as the right hand side of Equation (3.27).

3.5.3 Forward Difference with a Modification Term

In order to remedy the above problem, a modification term is needed [61]. From Equation (3.27), it may be seen that the modification term should be a function of the rigid body rotation $\omega\Delta t$ or, in general, be a function of the components of the spin tensor ${}^t\mathbf{W}$. In general large deformation , the skew-symmetric spin tensor is defined by;

$$ {}^tW_{ij} = \frac{1}{2}\left(\frac{\partial^t v_i}{\partial^t x_j} - \frac{\partial^t v_j}{\partial^t x_i}\right) \tag{3.28}$$

and the modification term for ${}^tD_{ij}$ components for general 2D large deformation process may be considered as:

$$\frac{\sqrt{1-({}^tW_{12}\Delta t)^2}-1}{\Delta t}$$

The above modification guarantees the objectivity of ${}^tD_{ij}$ in the case of rigid body rotation. With the modification term, numerical calculation of the ${}^tD_{ij}$ components are given by:

$$ {}^tD_{ij} = \frac{1}{2}\left(\frac{\partial^t v_i}{\partial^t x_j} + \frac{\partial^t v_j}{\partial^t x_i}\right) - \delta_{ij}{}^tm \tag{3.29}$$

where,

$$ {}^tm = \frac{\sqrt{1-({}^tW_{12}\Delta t)^2}-1}{\Delta t} = \frac{\sqrt{1-\left(\frac{1}{2}\left(\frac{\partial^t v_i}{\partial^t x_j} - \frac{\partial^t v_j}{\partial^t x_i}\right)\Delta t\right)^2}-1}{\Delta t} \tag{3.30}$$

and δ_{ij} is the Kronecker delta.

For the general rigid body rotation case given above, the spin tensor ${}^tW_{ij}$ is calculated in a way similar to Equation (3.27) and the components are given by:

$$\begin{aligned} {}^tW_{11} &= 0 \\ {}^tW_{22} &= 0 \\ {}^tW_{12} &= -{}^tW_{21} = -\frac{sin(\omega\Delta t)}{\Delta t} \end{aligned}$$

Substituting the above values in Equation (3.30) and (3.29), we have:

$$\begin{aligned} {}^tD_{11} &= 0 \\ {}^tD_{22} &= 0 \\ {}^tD_{12} &= 0 \end{aligned}$$

Therefore, the explicit integration algorithm, Equation (3.16), will satisfy incremental objectivity if ${}^tD_{ij}$ is modified as shown above. The ${}^tD_{ij}$ modification presented above may be considered as an alternative way to keep the incremental objectivity of integration algorithms developed by Pinsky [58] and Hughes [56]. The above modification is practically useful when explicit integration algorithms are utilized. More detailed numerical examples on the effectiveness of the above procedures is given in Chapter 5.

3.6 STRESS TRANSFORMATION AND RETURN MAPPING

3.6.1 General Considerations

It may be seen from the above development that in order to keep objectivity of the integration algorithms, the stress at a given time may be first transformed with an appropriate transformation tensor and then it should be added to the stress increment due to pure straining. For integrating the constitutive equation, the algorithm also has to satisfy the material constitutive relations (e.g., Prandtl-Reuss equations) and the incremental plastic consistency.

The return mapping algorithm [59, 60] is a numerical method used mostly in small deformation problems to satisfy the material constitutive relations and incremental plastic consistency. The method is generally based on two steps. The first one is to calculate an elastic predictor, which obtains the stresses at the end of the increment from the use of the elastic stress-strain relations. The second step is to subsequently map the obtained stresses onto a suitably updated yield surface and restore plastic consistency. This mapping step may be further divided into two steps; one explicit by projecting along the initial plastic flow direction and the other implicit by projecting along the updated plastic flow direction. Although some research has been done on objectivity of stress integration algorithms under finite deformation cases and the algorithms to keep incremental plastic consistency in small deformation cases are well developed, not too much effort has been made to satisfy incremental plastic consistency in finite deformation conditions.

In the case of large deformation, although the constitutive relation is not the same as for small deformation, the elastic predictor may still be calculated by assuming that the deformation in the current increment is completely elastic. Then the predictor is mapped onto a suitably updated yield surface using the appropriate large strain constitutive relation, thus restoring plastic consistency. For von-Mises material the equivalent stress at time t is defined by:

$$^{t}\sigma_{eq} = \left(\frac{3}{2}{}^{t}\sigma'_{ij}{}^{t}\sigma'_{ij}\right)^{\frac{1}{2}} \tag{3.31}$$

where, ${}^{t}\sigma'_{ij} = {}^{t}\sigma_{ij} - \frac{1}{3}\delta_{ij}{}^{t}\sigma_{kk}$ is the deviatoric stress and δ_{ij} is Kronecker delta. The accumulated equivalent plastic strain is calculated by [64]:

$$\varepsilon^{p}_{eq} = \int_{t_0}^{t}\left(\frac{2}{3}{}^{\tau}D^{p}_{ij}{}^{\tau}D^{p}_{ij}\right)^{\frac{1}{2}} d\tau \tag{3.32}$$

where; t_0 is the initial time, t is the current time of the deformation and ${}^{\tau}D^{p}_{ij}$is the plastic part of the rate of deformation tensor.

For rate-independent materials, the time t is trivial. An important point to be discussed in large deformation analysis is whether the return mapping should be performed before or after the stress transformation. If the return mapping is to be performed before the stress transformation, we should maintain the plastic consistency after the transformation. To investigate this, two different numerical schemes are presented and compared.

3.6.2 Scheme A: Transformation after Return Mapping

If at time t, the converged stress is ${}^{t}\sigma_{ij}$ and material velocity is ${}^{t}v_i$, with other quantities as shown in Figure 3.4, the procedure may be summarized as follow:

(i) input data for integration: ${}^{t}\sigma_{ij}$, ${}^{t}v_i$, Δt, ${}^{t}x_i$;

(ii) update material coordinate:${}^{t+\Delta t}x_i = {}^{t}x_i + {}^{t}v_i\Delta t$; then, calculate C_{ijkl} by assuming the increment to be completely elastic;

(iii) calculate the elastic stress increments $\Delta\sigma_{ij}$ by the use of Equation (3.14);

(iv) use stress quantities ${}^{t}\sigma_{ij}$ and $\Delta\sigma_{ij}$ as input to the return mapping algorithm and calculate an intermediate stress state ${}^{t}\sigma_{ij}^{(I)}$ that satisfies the incremental plastic consistency; then, calculate deformation history-dependent parameters, such as the current yield surface σ_{yield} and plastic strain ε_{eq}^{p};

(v) calculate the stress increments by $\Delta\sigma_{ij} = {}^{t+\Delta t}\sigma_{ij}^{(I)} - {}^{t}\sigma_{ij}$, use Equations (3.15), (3.20) or (3.21) to update stresses and obtain a second intermediate stress state ${}^{t+\Delta t}\sigma_{ij}^{(II)}$ and its corresponding equivalent stress ${}^{t+\Delta t}\sigma_{eq}^{(II)}$ by Equation (3.31);

(vi) scale down the equivalent stress point ${}^{t}\sigma_{ij}^{(II)}$ to the yield surface by using the relation ${}^{t+\Delta t}\sigma_{ij} = \frac{\sigma_{yield}}{{}^{t+\Delta t}\sigma_{eq}^{(II)}}\,{}^{t+\Delta t}\sigma_{ij}^{(II)}$

In the procedure above, it may be seen that step (vi) is necessary to keep the final stress point ${}^{t+\Delta t}\sigma_{ij}$ plastically consistent, i.e., on the yield surface.

3.6.3 Scheme B: Transformation Before Return Mapping

The second procedure may be summarized as follow:

(i) input data for stress integration: ${}^{t}\sigma_{ij}$, ${}^{t}v_i$, Δt, ${}^{t}x_i$;

(ii) update material coordinates:${}^{t+\Delta t}x_i = {}^{t}x_i + {}^{t}v_i\Delta t$; then, calculate C_{ijkl} by assuming the increment to be completely elastic;

(iii) calculate the intermediate stress state ${}^{t}\sigma_{ij}^{(I)}$ by the use of Equations (3.15), (3.20) or (3.21);

(iv) calculate the "elastic" stress increments by $\Delta\sigma_{ij} = {}^{t+\Delta t}\sigma_{ij}^{(I)} - {}^{t}\sigma_{ij}$;

(v) use ${}^{t}\sigma_{ij}$ and $\Delta\sigma_{ij}$ as the input stress components to regular return mapping algorithm and obtain the mapped stress ${}^{t+\Delta t}\sigma_{ij}$;

(vi) calculate the deformation history-dependent parameters, such as current yield stress σ_{yield} and plastic strain ε_{eq}^{p}.

Although this method projects the final stress point on the yield surface, the "elastic" stress increments include the incremental stress due to proper transformation, i.e. taking into account of the change in configuration. The return mapping procedure has to start with ${}^t\sigma_{ij}$ instead of ${}^t\sigma_{ij}^{(I)}$, because all history dependent parameters such as the current yield stress and the plastic strain are only updated to the state at time t.

The above numerical algorithms for keeping plastic incremental consistency, have been combined with the integration algorithms presented in Section 3.4.1. It is found that the combination of integration of Truesdell rate with Scheme A generally gives better results than other cases. Details of this development and calculation are given in Chapter 5.

3.7 UPDATING OF MATERIAL ASSOCIATED PROPERTIES

3.7.1 Basic Updating Scheme

In an ALE formulation, the relation between mesh motion and material motion at each point is given by:

$$u_i^c = a_i + b_{(i)}u_{(i)} \tag{3.33}$$

where, u_i is material displacement in i direction, u_i^c is mesh displacement in i direction, and no summation on i is observed. Therefore, location of new mesh points and integration points may be easily traced if isoparametric elements are used.

For any physical quantity f, the change due to material motion $\Delta_{\mathbf{x}}f$ and the change due to mesh motion $\Delta_{\boldsymbol{\chi}}f$ may be related to each other by (see Appendix A for details):

$$\Delta_{\boldsymbol{\chi}}f = \Delta_{\mathbf{x}}f + \frac{\partial^{t+\Delta t}f}{\partial^{t+\Delta t}x_i}(u_i^c - u_i) \tag{3.34}$$

where ${}^{t+\Delta t}f = f({}^{t+\Delta t}\mathbf{x}, t+\Delta t)$ is the value of f at material point at time $t+\Delta t$. The material increment $\Delta_{\mathbf{x}}f$ may be obtained as in the regular finite element calculations. If a material particle ${}^t\mathbf{x}$ at time t moves to ${}^{t+\Delta t}\mathbf{x}$ at time $t+\Delta t$, then:

$${}^{t+\Delta t}f = {}^tf + \Delta_{\mathbf{x}}f \tag{3.35}$$

Similarly, when mesh point ${}^{t}\chi$ at time t moves to ${}^{t+\Delta t}\chi$ at time $t + \Delta t$, at ${}^{t+\Delta t}\chi$, we have:

$$ {}^{t+\Delta t}f^{c} = {}^{t}f^{c} + \Delta_{\chi} f \tag{3.36} $$

But at time t, ${}^{t}\mathbf{x} = {}^{t}\chi$, so that ${}^{t}f^{c} = {}^{t}f = f({}^{t}\mathbf{x}, t)$. Therefore, Equations (3.34)-(3.36) may have the form:

$$ \begin{aligned} {}^{t+\Delta t}f^{c} &= {}^{t}f^{c} + \Delta_{\mathbf{x}} f + \frac{\partial^{t+\Delta t} f}{\partial^{t+\Delta t} x_i}(u_i^c - u_i) \\ &= {}^{t+\Delta t}f + \frac{\partial^{t+\Delta t} f}{\partial^{t+\Delta t} x_i}(u_i^c - u_i) \end{aligned} \tag{3.37} $$

Equation (3.37) effectively updates any physical quantity f from a material point to a mesh point. The feature that the relation between mesh motion and material motion is known prior is utilized. This procedures effectively eliminates iterations and interpolations or extrapolations used by other researchers [39, 43].

It is important to indicate that the above method may be utilized in updating for Lagrangian and Eulerian method. In updated Lagrangian method, the iteration to detect the relation between new mesh point and old mesh point may still be necessary. However, the updating may be done by Equation (3.37) if the old mesh points are treated as material point in ALE and new mesh points are treated as mesh points in ALE. For Eulerian method, the updating may be carried directly if the imaginary elements are taken as material elements in ALE and mesh points in Eulerian method as mesh points in ALE [7] .

3.7.2 Material Associated Properties at Nodal Points

The material associated properties are available at Gaussian integration points by the conventional incremental finite element method and the updating scheme presented in the previous section. Nodal values are important, however, for output consideration and for ALE calculation (e.g., Equation (2.20)). In order to obtain these nodal values, local smoothing procedure [62] is applied.

In this method, assuming the smoothed property $\overline{f}_\alpha$ and the smoothing shape function $\overline{h}_\alpha$ at nodal point α, then:

$$\overline{f} = \overline{h}_\alpha \overline{f}_\alpha \tag{3.38}$$

where $\overline{f}$ is the smoothed property and convention of summation about α is applied. If the unsmoothed property, calculated directly from deformation history is f, a local smoothing equation, by least square method, is applied to minimize the following equation on element level:

$$\phi = \int\int_{A_e} \left(\overline{h}_\alpha \overline{f}_\alpha - f\right)^2 dxdy \tag{3.39}$$

whereA_e is the area of the element and the following condition should be satisfied:

$$\frac{\partial \phi}{\partial \overline{f}_\beta} = 0, \qquad \beta = 1, 2, ..n_n \tag{3.40}$$

where, n_n is the number of nodes per element. So the smoothed properties may be obtained from the expression:

$$\int\int_{A_e} \left(\overline{h}_\alpha \overline{h}_\beta dxdy\right) \overline{f}_\alpha = \int\int_{A_e} \overline{h}_\beta f dxdy \tag{3.41}$$

The above equation may be evaluated using numerical integration procedures. Since the f values at integration points are already known, the values $\overline{f}_\alpha$ can be obtained, if the smoothing function is specified. The values from different elements sharing the same node are averaged and taken as the final property value at the node.

For rectangular and parallelogram elements, if $\overline{h}_\alpha$ is a bilinear shape function and a 2 × 2 Gaussian integration rule is adopted in evaluating Equation (3.41), it can be shown that the smoothed nodal properties $\overline{f}_1, \overline{f}_2, \overline{f}_3, \overline{f}_4$ may be expressed as:

$$\left\{ \begin{array}{c} \overline{f}_1 \\ \overline{f}_2 \\ \overline{f}_3 \\ \overline{f}_4 \end{array} \right\} = \begin{bmatrix} 1+\frac{\sqrt{3}}{2} & -\frac{1}{2} & 1-\frac{\sqrt{3}}{2} & -\frac{1}{2} \\ -\frac{1}{2} & 1+\frac{\sqrt{3}}{2} & -\frac{1}{2} & 1-\frac{\sqrt{3}}{2} \\ 1-\frac{\sqrt{3}}{2} & -\frac{1}{2} & 1+\frac{\sqrt{3}}{2} & -\frac{1}{2} \\ -\frac{1}{2} & 1-\frac{\sqrt{3}}{2} & -\frac{1}{2} & 1+\frac{\sqrt{3}}{2} \end{bmatrix} \left\{ \begin{array}{c} f_I \\ f_{II} \\ f_{III} \\ f_{IV} \end{array} \right\} \tag{3.42}$$

where, f_I, f_{II}, f_{III} and f_{IV} are the unsmoothed values at the Gauss points.

For general element shapes and shape functions, solving Equation (3.41) may be costly. To overcome this problem, discrete smoothing is introduced as described in the following. The function in Equation (3.39) is replaced by:

$$\phi = \sum_{K=1}^{N_s} \left(\left(\overline{h}_\alpha \overline{f}_\alpha \right) |_K - f \,|_K \right)^2 dxdy \tag{3.43}$$

where N_s is the number of sampling points within the element. For 4-node quadrilateral elements, taking the sampling points to be same as the 2 × 2 Gauss points and using bilinear shape functions for $\overline{h}_\alpha$, the following expression may be obtained,

$$\begin{bmatrix} \sum_{K=1}^{IV} \left(\overline{h}_1 \overline{h}_1\right) |_K & \cdots & \sum_{K=1}^{IV} \left(\overline{h}_1 \overline{h}_4\right) |_K \\ \cdots & \cdots & \cdots \\ \sum_{K=1}^{IV} \left(\overline{h}_4 \overline{h}_1\right) |_K & \cdots & \sum_{K=1}^{IV} \left(\overline{h}_4 \overline{h}_4\right) |_K \end{bmatrix} \begin{Bmatrix} \overline{f}_1 \\ \vdots \\ \overline{f}_4 \end{Bmatrix} = \begin{Bmatrix} \sum_{K=1}^{IV} \left(\overline{h}_1 f\right) |_K \\ \vdots \\ \sum_{K=1}^{IV} \left(\overline{h}_4 f\right) |_K \end{Bmatrix} \tag{3.44}$$

In the above equation, $\overline{h}_1 \,|_K, \ldots, \overline{h}_4 \,|_K$ are values of shape functions at Gaussian points I, II, III and IV, $f \,|_K$ $(K = I, II, \ldots, IV)$ is the unsmoothed properties at Gaussian points. Solving for $\overline{f}_1$, $\overline{f}_2$, $\overline{f}_3$ and $\overline{f}_4$, the above equation reduces to Equation (3.42). In this case, it should be noted that $\overline{f} = \overline{h}_\alpha \overline{f}_\alpha$ is an exact least square fit to the unsmoothed stress values f_I, f_{II}, f_{III} and f_{IV} for any shape of element.

The procedures of updating material associated properties may be summarized as following steps:

- Input the properties at material integration points ${}^{t+\Delta t}f$, u_i, u_i^c and ${}^{t+\Delta t}x_i = {}^t x_i + u_i$;

- If $u_i = u_i^c$ for all D.O.F. in the element, go to next step; otherwise, calculate the values at mesh integration points using Equation (3.37), then let ${}^{t+\Delta t}f = {}^{t+\Delta t}f^c$;

- Take the values of ${}^{t+\Delta t}f$ at different integration points I, II, III and IV, as the values f_I, f_{II}, f_{III} and f_{IV}, in Equation (3.42), and get smoothed variables $\overline{f}_1$, $\overline{f}_2$, $\overline{f}_3$ and $\overline{f}_4$ for nodes 1, 2, 3 and 4 individually.

- Add up the values from different elements sharing the same node and divided by the number of elements sharing the node to get the final property values at the mesh node.

Chapter 4

TWO-DIMENSIONAL ALE PROGRAM (ALEFE)

4.1 PROGRAM ALEFE

4.1.1 Features of ALEFE

Based on the formulation and the numerical algorithms presented in previous chapters, a general 2D finite element program ALEFE is developed with emphasis of application to metal forming problems. The basic features of the developed program may be summarized in the following:

- Three formulation options are available: general ALE, updated Lagrangian and Eulerian formulation. The user control parameters are a_I and b_I in the relation between mesh motion and material motion,

  $$ {}^{t}v_I^c = a_I + b_{(I)}{}^{t}v_{(I)} \qquad (I = 1, \ldots, N) \tag{4.1}$$

  where, no summation on I is observed in the above equation. ${}^{t}v_I$ and ${}^{t}v_I^c$ are material and mesh velocities at D.O.F. I respectively, and N is the total number of D.O.F. in the model. If $a_I = b_I = 0$ for all I, the program reduces to Eulerian one; If $a_I = 0$, $b_I = 1$ for all I, the program reduces to updated Lagrangian one; otherwise, the program will use a general ALE formulation.

- The program incorporates both Jaumann stress rate and Truesdell stress rate in stiffness calculation. The corresponding stress integration algorithms are employed in computing stress.

- For general ALE case, the mesh motion for interior nodes is controlled by the nodal motion on boundaries. User only needs to choose the a_I and b_I values in Equation (4.1) for boundary nodes. The internal nodes are specified by the program automatically to keep a homogeneous mesh. This is a particularly important features for users and for practical use of the program. It is relatively easy and straight forward to identify the formulation type required, and hence the coefficient a_I, b_I, for the boundary nodes.

- The program handles unsymmetric stiffness matrices and coupled displacements/velocity boundary conditions. The direct methods for coupled D.O.F. is applied so that no approximation is introduced in this aspect.

- Convergence criteria include displacement, residual force and iterative energy. User may choose one, two or all criteria for a given problem.

- Rate dependent and rate independent material models are included in the program. For plastic deformation, multi-linear hardening models can be chosen.

- Type of elements available is four node quadrilateral element.

- The input data format is designed similar to available commercial finite element codes, so that the data generation phase may be directly imported from these programs.

- The output data format is designed to be compatible with general graphic simulation and data processing commercial softwares, so that contour, x-y and deformed mesh plots may be easily created from the output data file of ALEFE.

- Full and modified Newton-Raphson methods are available.

4.1.2 Flow Chart of the Main Program

To discuss applications of numerical algorithms in program ALEFE, a general flow chart is shown in Figure (4.1). Two main flow schemes have been tried in the structure of program ALEFE. Both schemes have been implemented in the program, numerically tested and, finally the more efficient one was adopted. These are described briefly in the following.

The first scheme for the program flow is summarized in the following steps:

1. Create stiffness matrices from ${}^T\sigma_{ij}$, ${}^t x_i$;
2. Assemble the matrices and solve for ${}^t v_i$ and ${}^t v_i^c$;
3. Calculate ${}^t v_i \Delta t$, ${}^t v_i^c \Delta t$ and use ${}^T\sigma_{ij}$, ${}^t v_i \Delta t$ to calculate $\Delta\sigma_{ij}$;
4. Use ${}^T\sigma_{ij}$ and $\Delta\sigma_{ij}$ as input to stress integration and return mapping algorithm to get the new stress ${}^T\sigma_{ij}$ after the current iteration;
5. Check convergence. If step is not converged, apply the residuals divided by Δt as new incremental load and go to step 1; otherwise, leave the iteration loop.

The second scheme for the program is summarized in the following steps:

1. Create stiffness matrices from ${}^T\sigma_{ij}$, ${}^t x_i$;
2. Assemble the matrices and solve for ${}^t v_i$ and ${}^t v_i^c$;
3. Calculate accumulated displacements in the current load increment step $u_i = u_i + {}^t v_i \Delta t$ and $u_i^c = u_i^c + {}^t v_i^c \Delta t$ and use ${}^t\sigma_{ij}$, u_i to calculate $\Delta\sigma_{ij}$;
4. Use ${}^t\sigma_{ij}$ and $\Delta\sigma_{ij}$ as input to stress integration and return mapping algorithm to get the new stress ${}^T\sigma_{ij}$ after the current iteration;
5. Check convergence. If step is not converged, apply the residuals divided by Δt as new incremental load and go to step 1; otherwise, end the iteration loop.

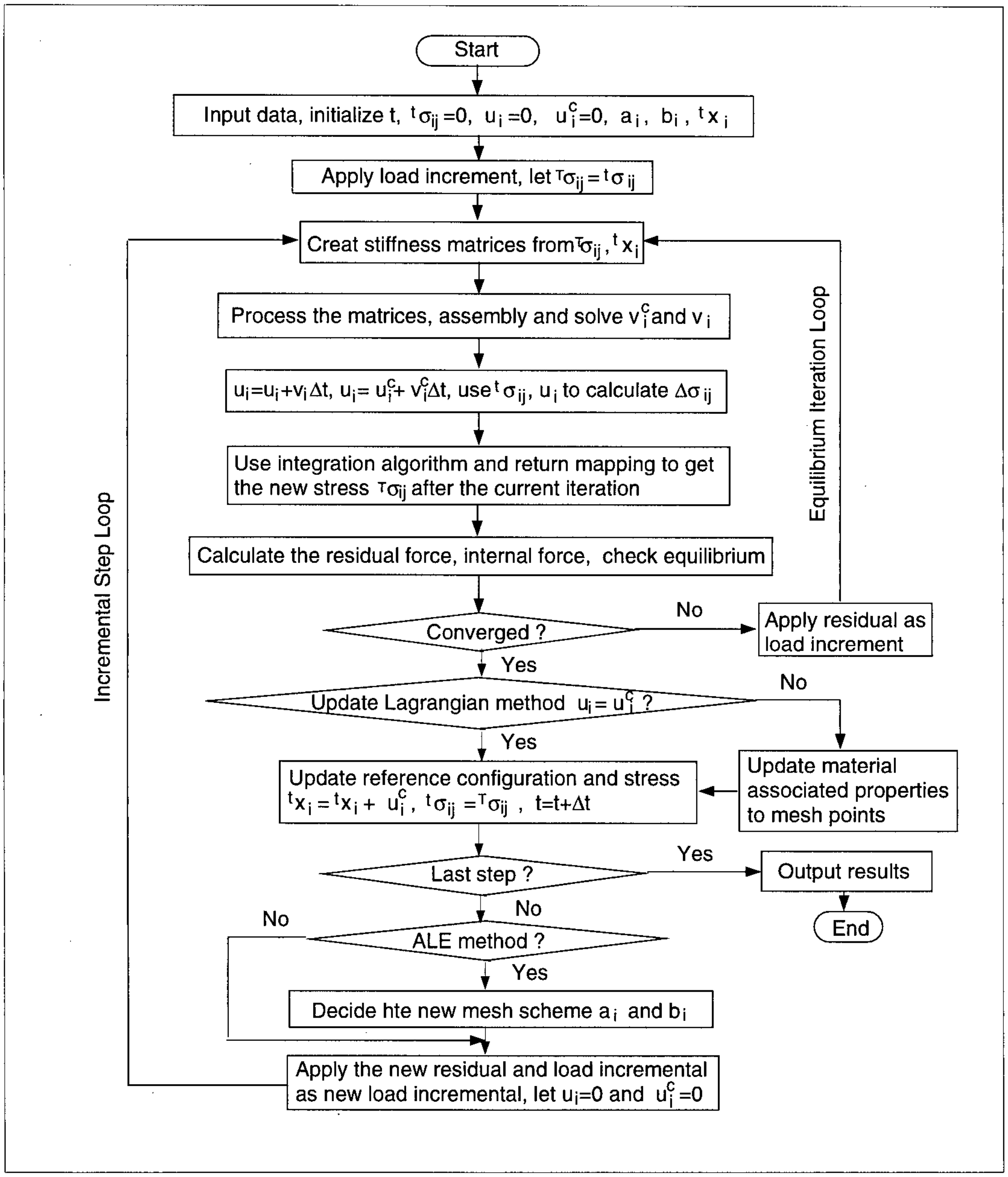

Start
Input data, initialize t, ${}^{t}\sigma_{ij}$=0, u_i=0, u_i^c=0, a_i, b_i, ${}^{t}x_i$
Apply load increment, let ${}^{T}\sigma_{ij} = {}^{t}\sigma_{ij}$
Creat stiffness matrices from ${}^{T}\sigma_{ij}$, ${}^{t}x_i$
Process the matrices, assembly and solve v_i^c and v_i
u_i=u_i+$v_i\Delta$t, u_i= u_i^c+ $v_i^c\Delta$t, use ${}^{t}\sigma_{ij}$, u_i to calculate $\Delta\sigma_{ij}$
Use integration algorithm and return mapping to get the new stress ${}^{T}\sigma_{ij}$ after the current iteration
Calculate the residual force, internal force, check equilibrium
Converged ?
No
Apply residual as load increment
Yes
Update Lagrangian method u_i= u_i^c ?
No
Yes
Update reference configuration and stress ${}^{t}x_i = {}^{t}x_i$ + u_i^c, ${}^{t}\sigma_{ij} = {}^{T}\sigma_{ij}$, t=t+Δt
Update material associated properties to mesh points
Last step ?
Yes
Output results
End
No
No
ALE method ?
Yes
Decide hte new mesh scheme a_i and b_i
Apply the new residual and load incremental as new load incremental, let u_i=0 and u_i^c =0
Incremental Step Loop
Equilibrium Iteration Loop

Figure 4.1: Flow chart of the main program

The difference between the two flow schemes is mainly in steps 3 and 4. In first scheme, the displacements of the current iteration ${}^{t}v_i\Delta t$ and the updated stress ${}^{T}\sigma_{ij}$ are taken as basic variables to calculate the stress increment $\Delta\sigma_{ij}$, so that $\Delta\sigma_{ij}$ is only the increment in current iteration and is caused by ${}^{t}v_i\Delta t$. The return mapping and stress integration starts from the stress state after last iteration, ${}^{T}\sigma_{ij}$. This state may not be generally a converged stress state and may not reflect a proper deformation history. In the second scheme, however, the accumulated displacement, $u_i = u_i + {}^{t}v_i\Delta t$, up to the current iteration, i.e., from the beginning of the current incremental step, and the stress at beginning of the current increment ${}^{t}\sigma_{ij}$ are applied to calculate the stress increment $\Delta\sigma_{ij}$, so that $\Delta\sigma_{ij}$ is the "accumulated" increment from the beginning of the current step. The return mapping and stress integration starts from the stress state after the last incremental step ${}^{t}\sigma_{ij}$. This stress state is always a converged state and reflects a proper deformation history. Another disadvantage of the first method is that the current yield surface or yield stress may be overestimated, which may lead to wrong results. To explain this, we consider an iteration load that takes the current yield surface at a point to a higher level. If this iteration load is not right, but is then corrected during the increment, the yield surface at the point will still be registered at the higher value obtained from the iteration. This may, in general, not reflect a true deformation history. In the second scheme, the integration and return mapping are always performed from the last converged or accepted stress state. Thus the final converged results are not affected by errors or overshoots that may occur in a specific iteration.

Numerical experiments show that the first scheme is less robust from the convergence point of view and gives less accurate results upon comparing to experiments or published data than the second scheme. Consequently, the second scheme is employed in program ALEFE.

4.2 MAIN SUBROUTINES

4.2.1 Functions of Subroutines

A total of twenty-three subroutines are compiled with the main program to perform the ALE simulation. The function of each subroutine is described as:

MAIN Input analysis data such as total number of incremental steps, tolerances, maximum iteration number; specify the boundary nodes as either Lagrangian or general ALE nodes. Output results according to user's requirement.

INDATA Input initial data from the text file created by other commercial finite element codes, e.g., NISA. The data include finite element model, material properties and boundary conditions.

INCRL Evaluates the incremental load vector according to the total load, total number of incremental steps and residual forces from previous iteration.

GAUSSQ Sets up the integration point position in natural coordinate system and calculates weighting factors for numerical integration.

SHADRI Calculates shape function values and their derivatives with respect to natural coordinates at a particular point.

JACOBD Calculates Cartesian coordinates of a Gaussian points, Jacobian determinant and the Cartesian derivatives of shape functions at Gaussian points.

DORDEP Evaluates the constitutive matrix C_{ijkl} for plane strain problems.

STIFMA Evaluates ALE stiffness matrices for each element.

FRONT Frontal solver for unsymmetric stiffness matrix.

SRS1N Evaluates stresses, plastic strains and current yield stress at Gaussian points and calculates residual forces at nodal points.

ELEPL2 Calculates the stress increment due to elastic deformation and the part due to plastic deformation.

INV2 Evaluates stress invariants and the equivalent stresses from given stress components.

POPV2 Uses return mapping method to update the stress increment due to plastic deformation.

STRN2 Calculates the rate of deformation tensor.

STRUPD Updates the material associated properties from material points to mesh points.

ALEDEC Assigns mesh motion for boundary nodes according to the user's specification.

MESMOT Calculates mesh motion for all interior nodes.

BOUADH Adjusts boundary nodes after each incremental step to keep homogeneous distribution of boundary nodes.

CONVER Calculates convergence parameters and checks convergence.

DECONF Stores the output data in a format compatible with other commercial graphic processing software for post processing.

WRIDA Write data files for further processing.

READA Restores data from direct access files.

CHADIM Converts stress, rate of deformation from one-dimension array to two dimension array.

MMATX Calculates the product of two matrices.

4.2.2 Flow Chart of Mesh Motion

In ALEFE, mesh motion is completed by three subroutines, ALEDEC, MESMOT and BOUADH which, individually, calculates the a_I and b_I values for boundary and interior nodes and adjusts boundary nodes after each incremental step. User only needs to specify each D.O.F. on boundaries as Lagrangian, Eulerian or general ALE. Users may specify particular areas as Lagrangian, Eulerian or ALE ones. Such specifications will be maintained during the whole simulation. Figure 4.2 shows a flow chart for the mesh motion scheme.

4.3 CONVERGENCE CRITERIA

4.3.1 General Discussion

In an incremental solution strategy based on iterative methods, realistic convergence criteria are necessary for the termination of the iteration. If the convergence tolerance is too loose, inaccurate results may be obtained, whereas if the tolerance is too tight, much computational effort is spent to obtain needless accuracy [65].

ANSYS [9] uses displacements and residual forces as convergence criteria. The maximum value, absolute value and Euclidean norm options are included. In residual force check, the total applied external force instead of the increments of external forces are employed as the base. In NISA [10], displacements, residual forces and iterative energy are checked in each iteration. In the displacement criterion, a load step will be assumed converged if the ratio of the maximum absolute iterative displacement to the maximum absolute displacement at first iteration in the current step is less than or equal to the preset tolerance. As in ANSYS, Euclidean norm of residual forces is checked, but the base is the incremental external forces. In the energy criterion, a load step will be assumed converged if the ratio of the iterative energy to the energy at first iteration is less than or equal to the preset tolerance. In [66] residual force criterion similar to ANSYS and energy criterion (ratio of the sum of absolute values of work done by the

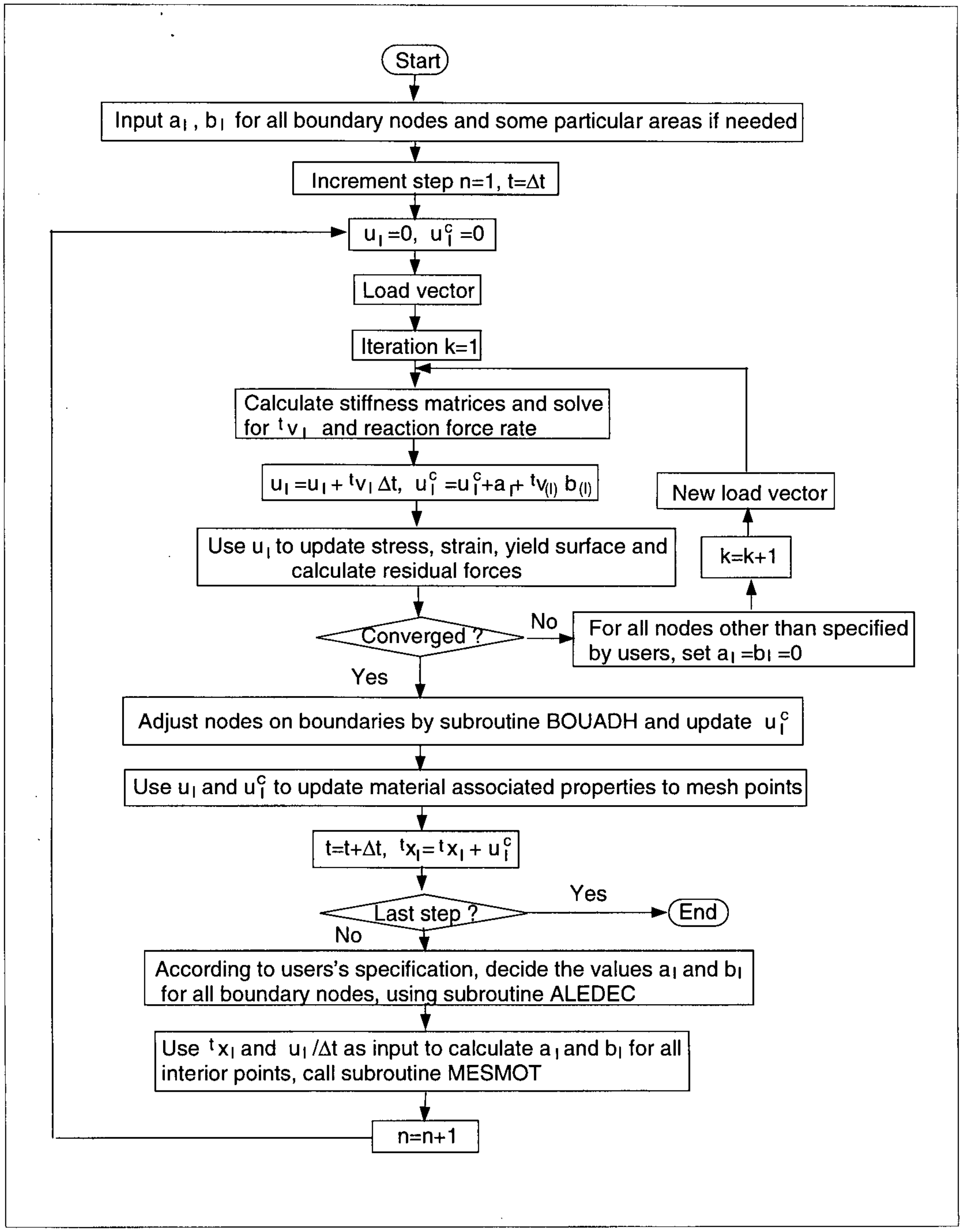

Start
Input a_I, b_I for all boundary nodes and some particular areas if needed
Increment step n=1, t=Δt
u_I=0, u_I^c=0
Load vector
Iteration k=1
Calculate stiffness matrices and solve for ${}^tv_I$ and reaction force rate
$u_I = u_I + {}^tv_I \Delta t$, $u_I^c = u_I^c + a_I + {}^tv_{(I)} b_{(I)}$
Use u_I to update stress, strain, yield surface and calculate residual forces
Converged ?
No
Yes
For all nodes other than specified by users, set $a_I = b_I = 0$
k=k+1
New load vector
Adjust nodes on boundaries by subroutine BOUADH and update u_I^c
Use u_I and u_I^c to update material associated properties to mesh points
$t = t + \Delta t$, ${}^tx_I = {}^tx_I + u_I^c$
Last step ?
Yes
End
No
According to users's specification, decide the values a_I and b_I for all boundary nodes, using subroutine ALEDEC
Use ${}^tx_I$ and $u_I/\Delta t$ as input to calculate a_I and b_I for all interior points, call subroutine MESMOT
n=n+1

Figure 4.2: The flow chart for mesh control

iterative residual forces to the sum of absolute values of work done by the accumulated external forces in the first iteration) are used. In program ALEFE, displacements, residual forces and energy criteria are employed for checking convergence. Brief description of these criteria is described in the following.

4.3.2 Displacement Criterion

In this criterion, it is assumed that the ratio of the norm of the current iterative displacement to the norm of the displacement in first iteration be within a preset tolerance. This may be written as:

$$\frac{\sqrt{\Sigma_{I=1}^{N}(\Delta u_I^{(k)})^2}}{\sqrt{\Sigma_{I=1}^{N}(\Delta u_I^{(1)})^2}} \leq t_d \tag{4.2}$$

where N is total number of D.O.F. in the model, $\Delta u_I^{(k)} = {}^{t}v_I \Delta t$ is the displacement in k^{th} iteration, ${}^{t}v_I$ is the material velocity in the k^{th} iteration at global D.O.F., I, t_d is displacement tolerance. Although this criteria is effective in some analyses, numerical experience showed that other criteria should be, in general, incorporated with this one [65].

4.3.3 Force Criterion

Another criterion is based on the out of balance forces. Substituting Equation (2.7) and the shape function approximation into Equation (2.6), we get the equivalent nodal force at D.O.F. I:

$${}^{t+\Delta t}q_I = \int_{{}^{t+\Delta t}V} \frac{1}{2} {}^{t+\Delta t}\sigma_{ij} \left(h_{iI,j} + h_{jI,i}\right) d^{t+\Delta t}V$$

and the external force at D.O.F. I:

$${}^{t+\Delta t}p_I = \int_{{}^{t+\Delta t}V} {}^{t+\Delta t}f_i^B h_{iI} d^{t+\Delta t}V + \int_{{}^{t+\Delta t}S} {}^{t+\Delta t}f_i^S h_{iI} d^{t+\Delta t}S$$

Therefore, the residual force at D.O.F. I can be expressed as:

$$^{t+\Delta t}r_I = {}^{t+\Delta t}p_I - {}^{t+\Delta t}q_I$$

where h_{iI} is the value of shape function in i direction at D.O.F. I. If the variables in the expressions for internal equivalent nodal force and external nodal force are calculated at the current iteration k, we get the residual force at the k^{th} iteration in the incremental step expressed as:

$$^{t+\Delta t}r_I^{(k)} = {}^{t+\Delta t}p_I^{(k)} - {}^{t+\Delta t}q_I^{(k)} \tag{4.3}$$

A force convergence criterion is chosen as the ratio of the norm of the iterative residual force to the load increment, i.e.,

$$\frac{\sqrt{\Sigma_{I=1}^{N}({}^{t+\Delta t}r_I^{(k)})^2}}{\sqrt{\Sigma_{I=1}^{N}({}^{t+\Delta t}p_I^{(0)} - {}^{t}q_I)^2}} \leq t_f \tag{4.4}$$

where$^{t}q_I$ is the converged equivalent nodal force at D.O.F. I at time t, and t_f is force tolerance.

In order that the above criterion does not become too restrictive for small load increments, the maximum load increment during the course of deformation up to the current increment is used. Equation (4.4) becomes:

$$\frac{\sqrt{\Sigma_{I=1}^{N}({}^{t+\Delta t}r_I^{(k)})^2}}{max(\sqrt{\Sigma_{I=1}^{N}({}^{t+\Delta t}p_I^{(0)} - {}^{t}q_I)^2})} \leq t_f \tag{4.5}$$

It should be noted that one of the disadvantages in using a force criterion is the possible existence of inconsistent units in the force vector, e.g., forces and moments in beam elements.

4.3.4 Energy Criterion

In order to provide some indication of when both the displacements and forces are near to equilibrium values, the iterative energy, i.e., the work done by the iterative residual forces, is compared to the initial internal energy increment, so that:

$$\frac{\Sigma_{I=1}^{N}(\Delta u_I^{(k)\,t+\Delta t}r_I^{(k-1)})}{\Sigma_{I=1}^{N}({}^{t+\Delta t}p_I^{(0)} - {}^{t}q_I)\Delta u_I^{(1)}} \leq t_e \tag{4.6}$$

where ${}^{t+\Delta t}p_I^{(0)}$ is the accumulated external force at D.O.F. I at $t + \Delta t$.

It is generally accepted that a combination of any two criteria provide an effective convergence criteria for most engineering problems [65].

4.4 COUPLED DISPLACEMENT CONDITION

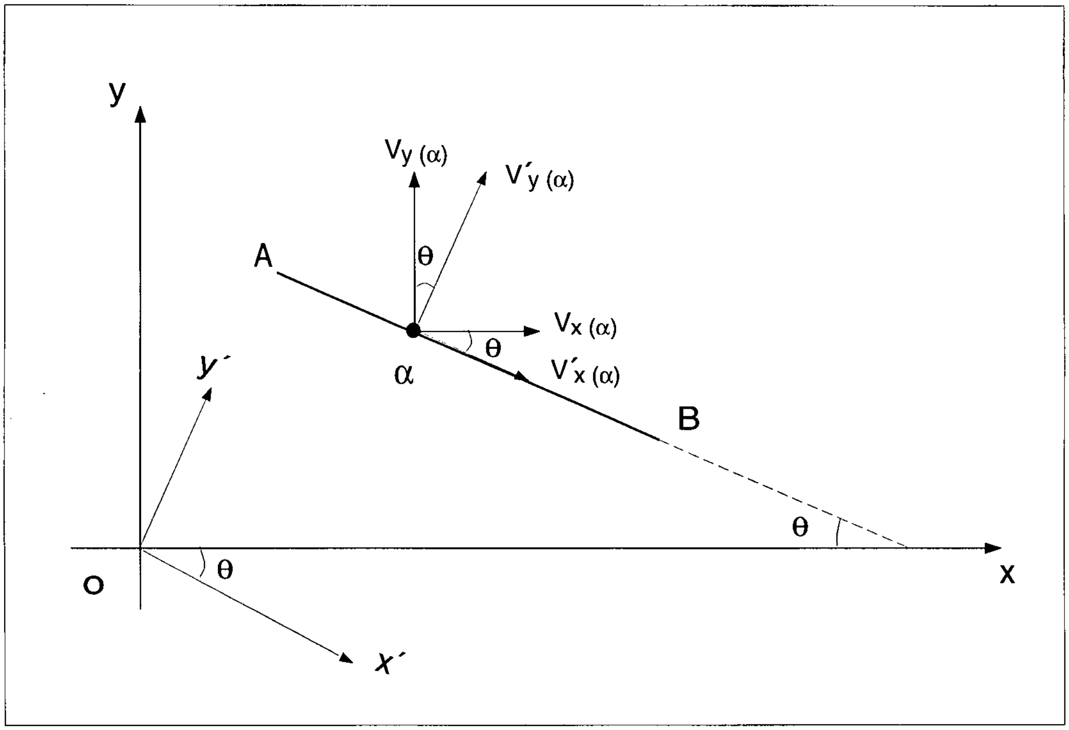


Figure 4.3: The relation between global and local velocities

The direct transformation method is applied to impose coupled displacement conditions. As shown in Figure 4.3, if a nodal point α is confined to move along a surface AB, we have the following relation between local and global displacement at node α:

$$\left\{\begin{array}{c} {}^t v_{x(\alpha)} \\ {}^t v_{y(\alpha)} \end{array}\right\} = \left[\begin{array}{cc} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{array}\right] \left\{\begin{array}{c} {}^t v'_{x(\alpha)} \\ {}^t v'_{y(\alpha)} \end{array}\right\} \tag{4.7}$$

In the local coordinates x', y', the boundary condition will be ${}^t v'_{y(\alpha)} = 0$ and the conventional elimination algorithm may be employed. It is first necessary to transform the stiffness matrix for the pertinent node using the transformation tensor:

$$\left[\begin{array}{cc} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{array}\right] = \mathbf{T}_{(\alpha)} \tag{4.8}$$

In matrix form, the following equations are utilized:

$${}^t\mathbf{v} = \mathbf{T}{}^t\mathbf{v}' \tag{4.9}$$

where ${}^t\mathbf{v}$ is the global velocity vector, ${}^t\mathbf{v}'$ is local velocity vector and $\mathbf{T}$ is transformation matrix. Substituting Equation (4.9) into the final finite element equations (Equation (3.5)), we have:

$${}^t\overline{\mathbf{K}}\mathbf{T}{}^t\mathbf{v}' = {}^t\overline{\mathbf{p}} \tag{4.10}$$

The above equation may be solved for ${}^t\mathbf{v}'$ by the conventional assembly and elimination procedure. It should be noted that the stiffness matrix resulted in Equation (4.10), i.e., ${}^t\overline{\mathbf{K}}\mathbf{T}$, is an unsymmetric one. To keep the symmetry characteristic, Equation (4.10) should be premultiplied by T^T.. This treatment is not necessary in general ALE formulation since ${}^t\overline{\mathbf{K}}$ is originally unsymmetric. Figure 4.4 shows a flow chart for the frontal solution scheme.

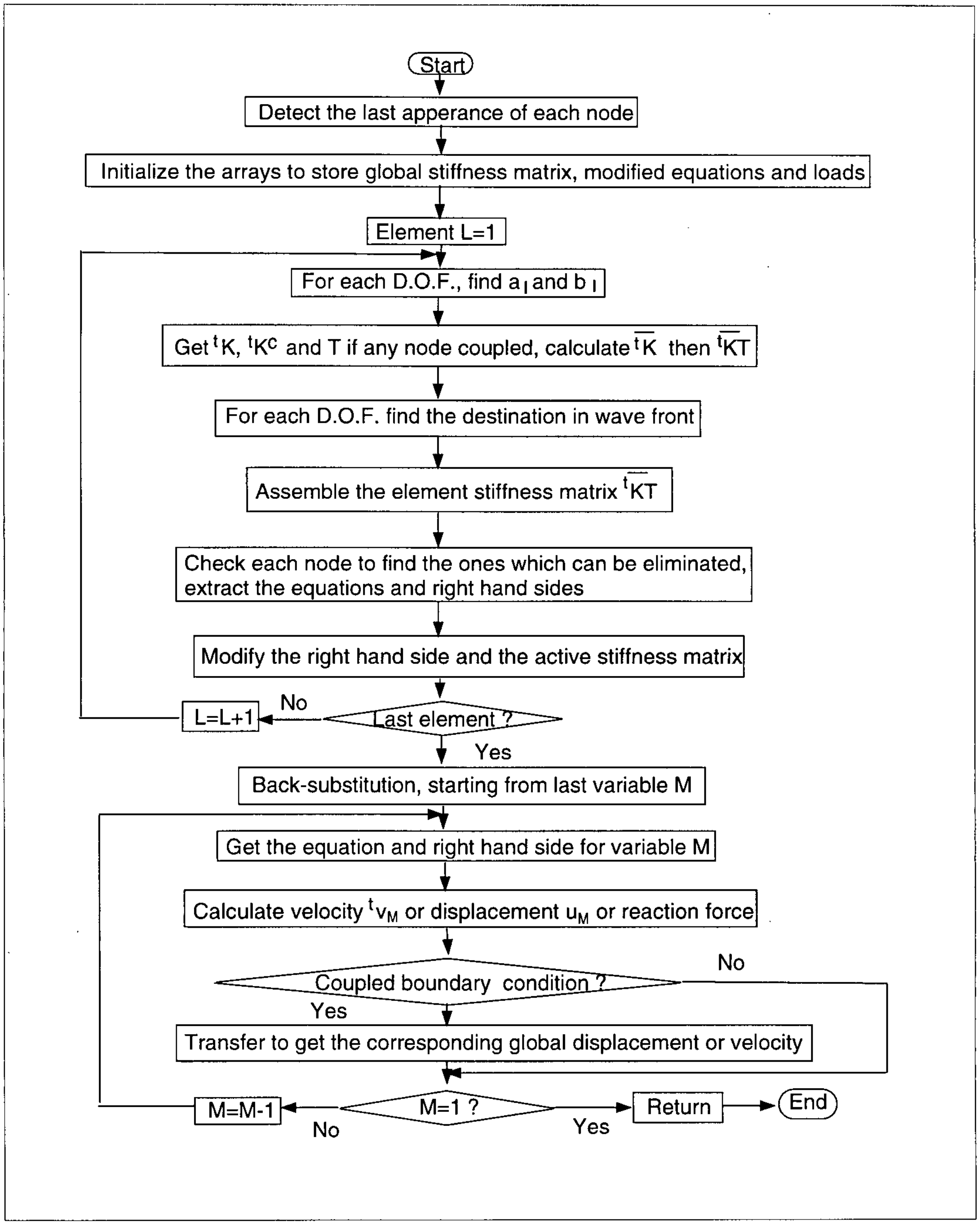

Start
Detect the last apperance of each node
Initialize the arrays to store global stiffness matrix, modified equations and loads
Element L=1
For each D.O.F., find a_I and b_I
Get ${}^tK$, ${}^tK^C$ and T if any node coupled, calculate ${}^t\overline{K}$ then ${}^t\overline{KT}$
For each D.O.F. find the destination in wave front
Assemble the element stiffness matrix ${}^t\overline{KT}$
Check each node to find the ones which can be eliminated, extract the equations and right hand sides
Modify the right hand side and the active stiffness matrix
Last element ?
No
L=L+1
Yes
Back-substitution, starting from last variable M
Get the equation and right hand side for variable M
Calculate velocity ${}^tv_M$ or displacement u_M or reaction force
Coupled boundary condition ?
No
Yes
Transfer to get the corresponding global displacement or velocity
M=1 ?
No
M=M-1
Yes
Return
End

Figure 4.4: The flow chart for frontal solution

Chapter 5

NUMERICAL EXAMPLES

5.1 OVERVIEW

Numerical examples are employed to demonstrate the correctness of the developed formulation, numerical algorithm and its implementation and to highlight features of ALE in comparison with other Lagrangian and Eulerian formulation. All examples are plane strain deformation problems and may be divided into two categories. The first category is sample problems which include rotation of an elastically deformed element, elastic-plastic small deflection analysis of a square element and pure bending of rectangular plate. This kind of problem is invoked to verify the correctness and accuracy of the program and the numerical algorithms. The second category of problems is more practical in nature and includes punch forging process, tensile necking of a elastic-plastic bar, sheet metal extrusion through curved and straight die, rolling and compression between wedge-shaped dies. These problems are generally more complicated and are applied to investigate the advantages of ALE in simulation of real industrial problems. All examples are simulated by the developed ALEFE program and one or more available commercial codes. Results are compared with available numerical and/or experimental ones from literature.

5.2 ROTATION OF AN ELASTICALLY DEFORMED ELEMENT

A combined extension and rotation of single element is designed to show the necessity of the special numerical treatment of rate of deformation tensor ${}^{t}D_{ij}$. The two-dimensional plane strain element shown in Figure 5.1 at time $t = 0$, is subjected to displacement in the x direction, $u_x = 4.5 \times 10^{-4} b$ and displacement in the y direction, $u_y = -1.125 \times 10^{-4} b$, where b is the dimension of the square element. Choosing a value of Poisson's ratio $\nu = 0.2$, means that the stress σ_y will be equal to zero at time $t = 0$. A rigid body rotation of 90 degrees is imposed on the element as shown in the figure through a constant angular velocity ω. The material's Young's modulus E is taken as 200 GPa. In this example, small deformation but large rotation and displacement occur. The analytical stress solution may be obtained from Hooke's law and Mohr's circle, and it is used as a reference to test the general structure and the correctness of the program.

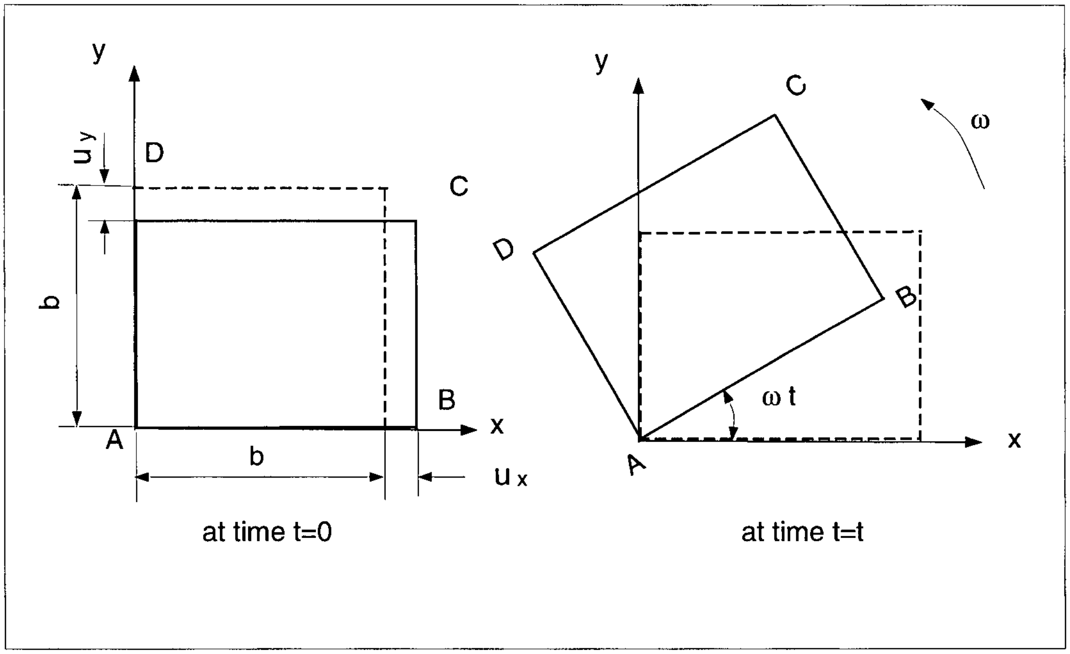


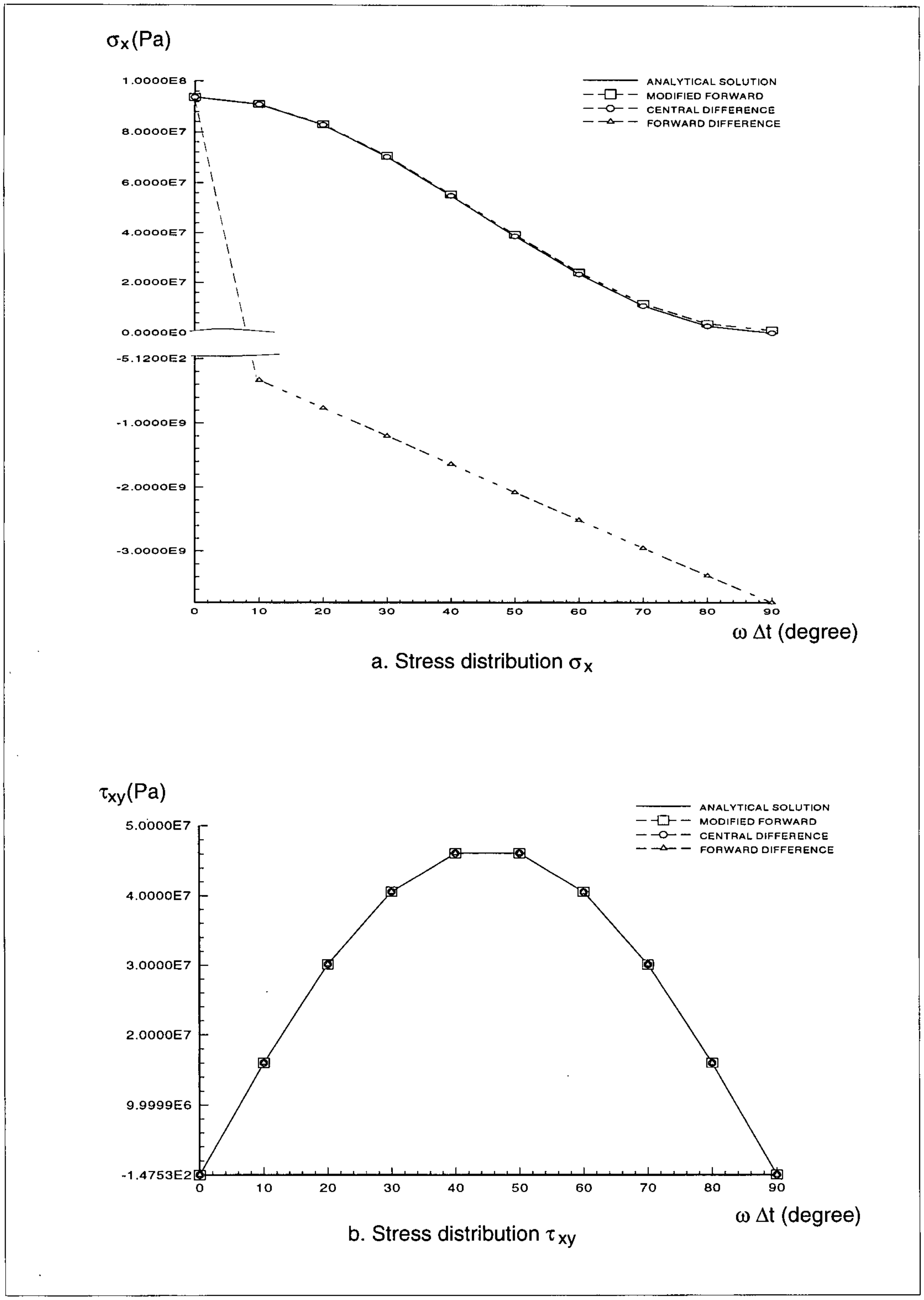
Figure 5.1: Element with combined extension and rigid body rotation

In the numerical solution, the three different algorithms discussed in Chapter 4 are applied to calculate the ${}^tD_{ij}$. The stress is integrated using Equation (3.21). Stress results of the above example are shown in Figure 5.2. It is clear that the predictions of the forward difference formula with modification term and the central difference algorithm agree very well with the analytical values, even for large rotation of up to 90 degrees. The conventional forward difference algorithm predicts, however, incorrect values for stress σ_x. Because the non-zero term in Equation (3.27) is a constant, it brings a constant increment for the normal stresses and in this particular example, the stress increment owing to the rotation is much less than this value, so that the stress σ_x versus the rotation angle $\omega\Delta t$ is a straight line. It is found that reducing the incremental step in conventional forward difference algorithm has very little or no effect on the improvement of accuracy of σ_x. This may be due to the accumulation of the value from non-zero term in Equation (3.27).

The above example shows that applying conventional time discretization or forward difference method in calculation of ${}^tD_{ij}$ or stress integration may cause significant errors, especially when the boundary conditions are only displacement boundary conditions. The errors can be eliminated by using the central difference method in implicit stress integration. In explicit integration algorithm, however, or when ${}^tD_{ij}$ is employed separately, e.g., in visco-plastic deformation, the modification term developed in Chapter 3 may be applied to keep the objectivity of ${}^tD_{ij}$.

5.3 ELASTIC-PLASTIC SMALL DEFLECTION ANALYSIS OF A SQUARE ELEMENT

The example is a simple verification problem used by many commercial programs [10]. In this problem, a bar of square cross section is incrementally loaded by tension on two opposite edges, as shown in Figure 5.3. Material properties are taken as Young's modulus $E = 18.4$ MPa $(8/3\ ksi)$, Poisson's ratio $\nu = 0.5$, initial yield stress $\sigma_{yield0} = 55.1\ KPa$ $(8.0\ psi)$ and hardening modulus $H = 2.6\ MPa$ $(8/21\ ksi)$ and the material is assumed to have a bilinear isotropic hardening behavior. The final pressure is increased to 137.8 KPa $(20\ psi)$ in 20 steps. Because the geometry and loading are symmetric, only a quarter is modelled with a 4-node



a. Stress distribution σ_x

b. Stress distribution τ_{xy}

Figure 5.2: The stress results of single element under rotation

quadrilateral element. To compare the results, the exact same problem is also simulated by NISA and ANSYS.

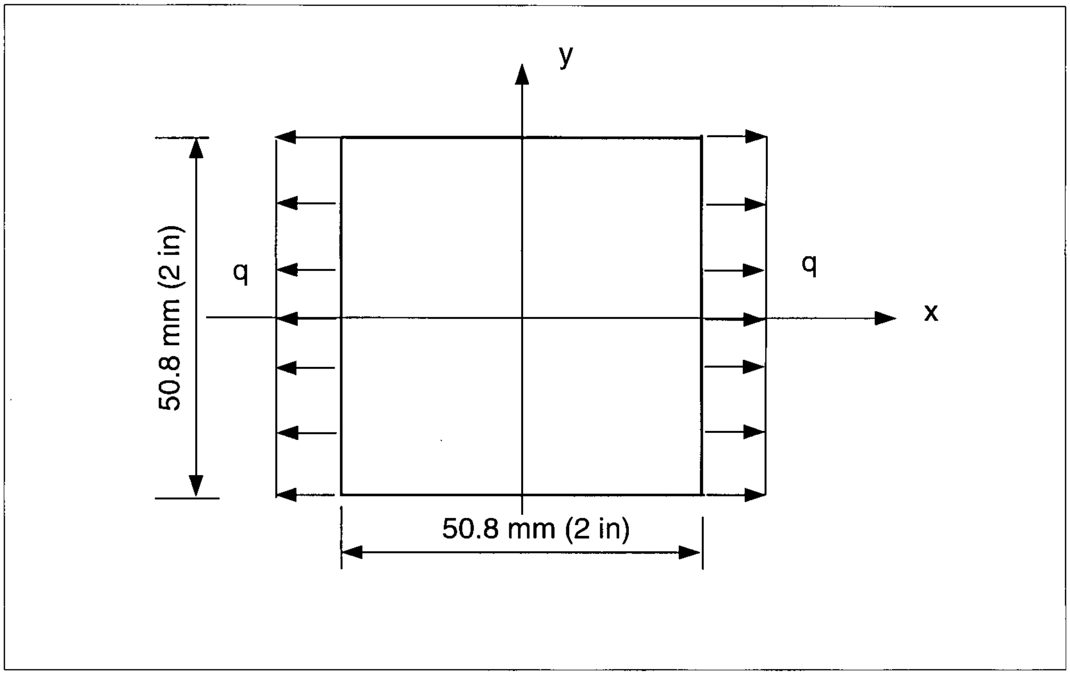


Figure 5.3: Elastic-plastic small deflection model

Figure 5.4 is the curve of tensile load versus displacement in x direction u_x. Before a load level of approximate 68.9 KPa (10 psi), results from the developed program, NISA and ANSYS are almost identical. Discrepancy of results starts at a load level of 68.9 KPa (10 psi) and increases with the increasement of load or deformation. This may be attributed to different level of approximation used in each program for handling the effect of finite deformation. Generally, however, the solution from ALEFE is in good agreement with the ones from NISA and ANSYS programs.

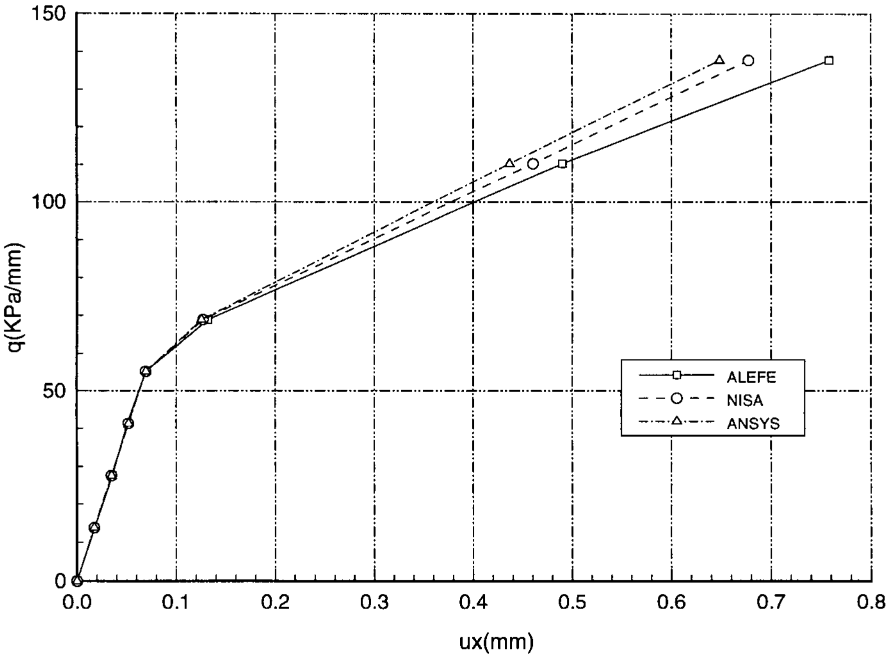


Figure 5.4: Load versus displacement in x direction

5.4 PURE BENDING OF A RECTANGULAR PLATE

5.4.1 Problem Description

This example is designed to investigate the effect of distorted mesh on the accuracy of simulated results and to show the necessity of mesh motion in ALE. The example is also analysed by NISA. In this example, a rectangular plane strain plate is subjected to pure bending, as shown in Figure 5.5. The deformation is pure elastic to eliminate effects caused by the differences in handling plasticity related aspects. Young's modulus E is 200 GPa, Poisson's ratio $\nu = 0.3$ and concentrated force F is 2000 N/mm. The problem is solved using 5 incremental steps.

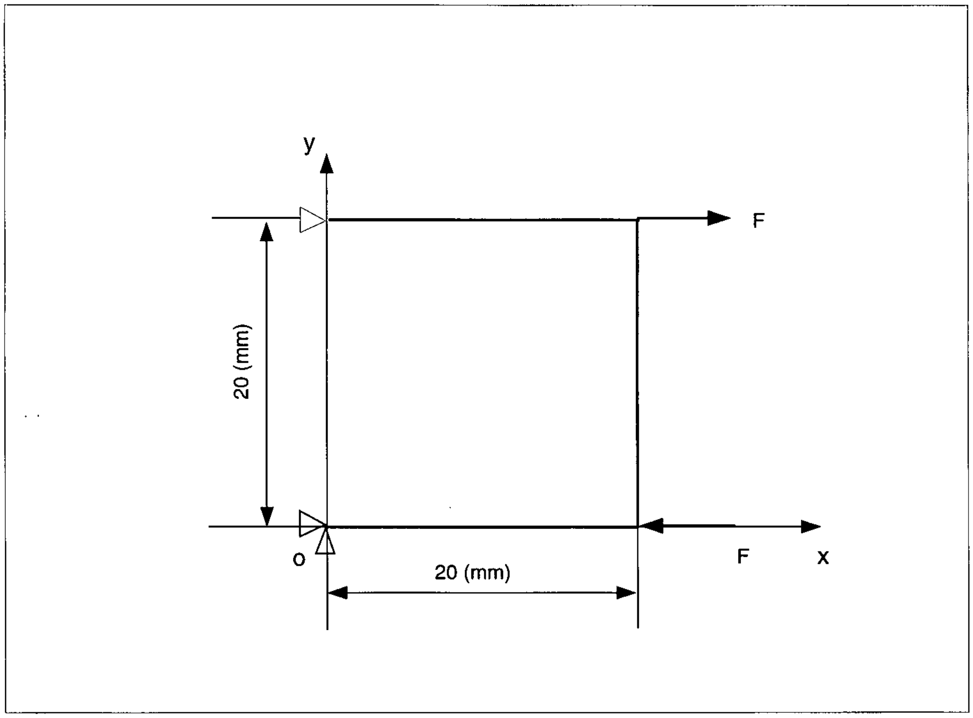


Figure 5.5: Pure bending of a rectangular plate

5.4.2 Finite Element Models

Finite element models used in the analysis are shown in Figure 5.6. Model-a consists of 100 square quadrilateral elements with 10 elements on each edge. This homogeneous and dense mesh is expected to give more accurate results that will be used as reference [67]. Model-b has only 4 square elements. Although still homogeneous, it is much coarser and is applied as "standard" model to compare with model-c and model-d which have distorted meshes. In model-c, although all 4 elements have interior angles less than 180°, so negative Jacobian determinant will not appear, the mesh is severely distorted. Model-d is an extreme case where element 3 has an interior angle larger than 180°.

5.4.3 Results of Lagrangian Formulation

The deflections are simulated first by the updated Lagrangian option of the developed program ALEFE and the commercial code NISA. Models a, b and c are run successfully for a maximum value of F equal to 2000 N/mm. The deflection at the lower-right corner of the plate is shown in Figure 5.7. It may be seen that the fine homogeneous mesh, model-a, gives the largest displacements, the distorted mesh, model-c, gives the least displacements; and the coarse homogeneous mesh, model-b, predicts an intermediate displacement values. In the example, the distorted less dense mesh stiffens the solution. This may be generalized to elastic and elastic-plastic problems when displacement methods are applied in finite element analyses, because solution starts from a kinematically admissible condition and a upper bound solution is expected.

Model-d did not go through both in ALEFE and NISA because of the negative Jacobian determinant in element 3 with both programs reporting a value of -2.6036. The negative Jacobian is due to an interior angle larger than 180°. The only limitation on the shape of the real element is that each interior corner angle must be less than 180°, i.e., the element must be

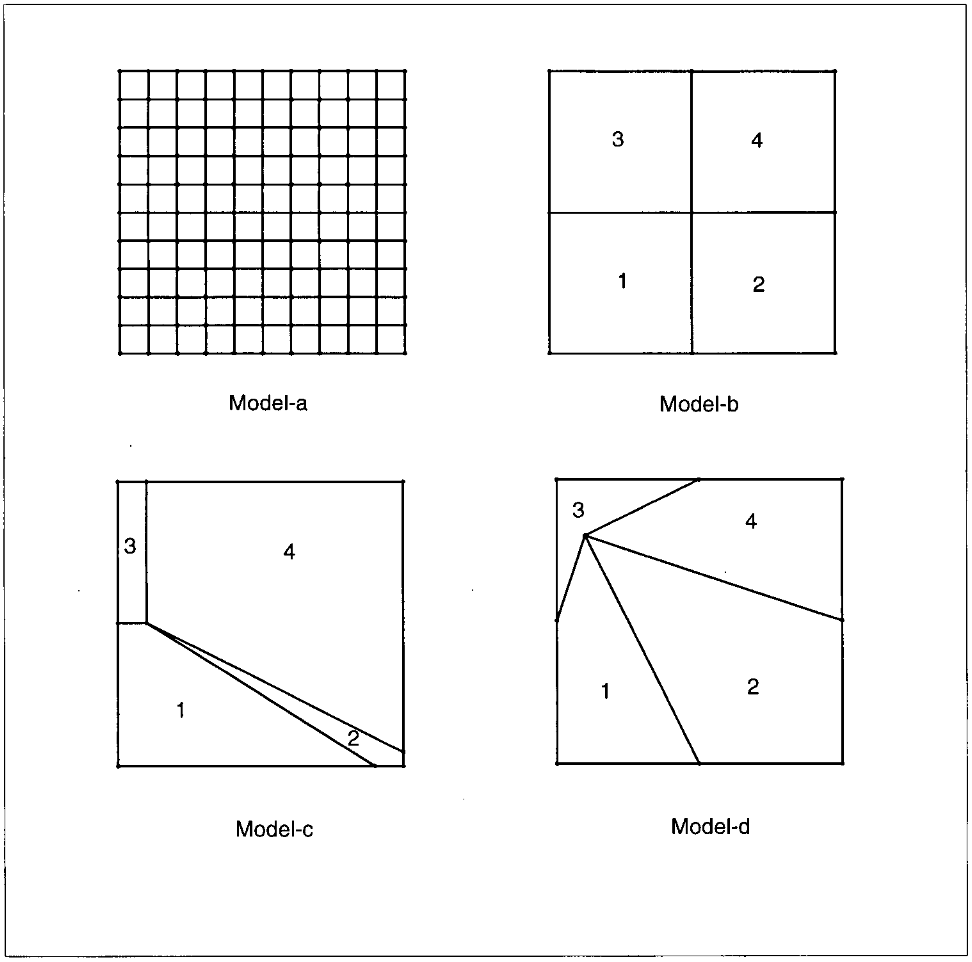
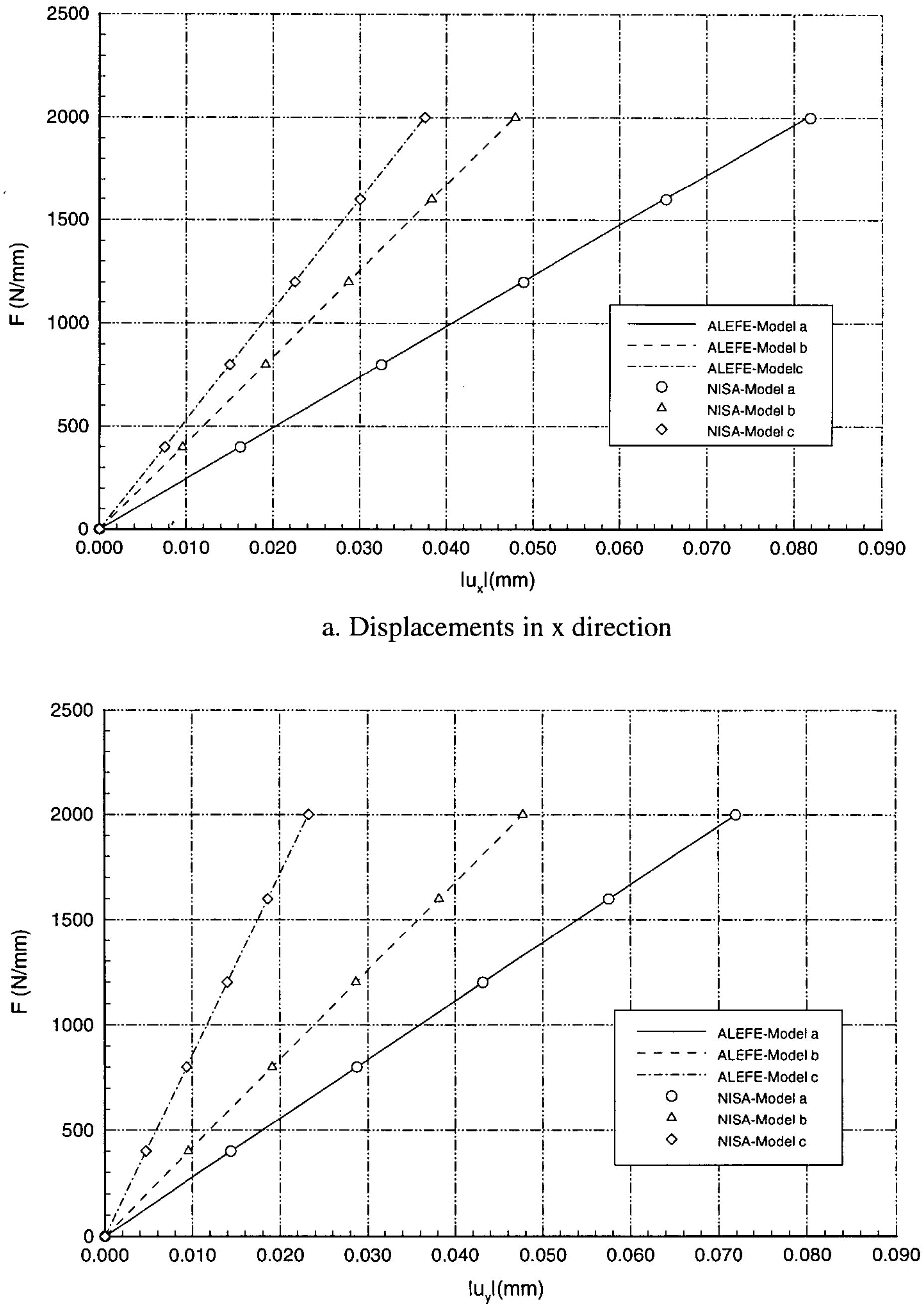


Figure 5.6: Finite element models for the pure bending of a plate



a. Displacements in x direction

b. Displacements in y direction

Figure 5.7: Pure bending of a plate: Lagrangian Formulation Results

convex [68]. Practically, however, to achieve higher accuracy, the optimum elements should have interior corner angles of quadrilaterals not far from 90° [67] and aspect ratios from $1/5$ to 5 [68].

5.4.4 ALE Simulation and Result

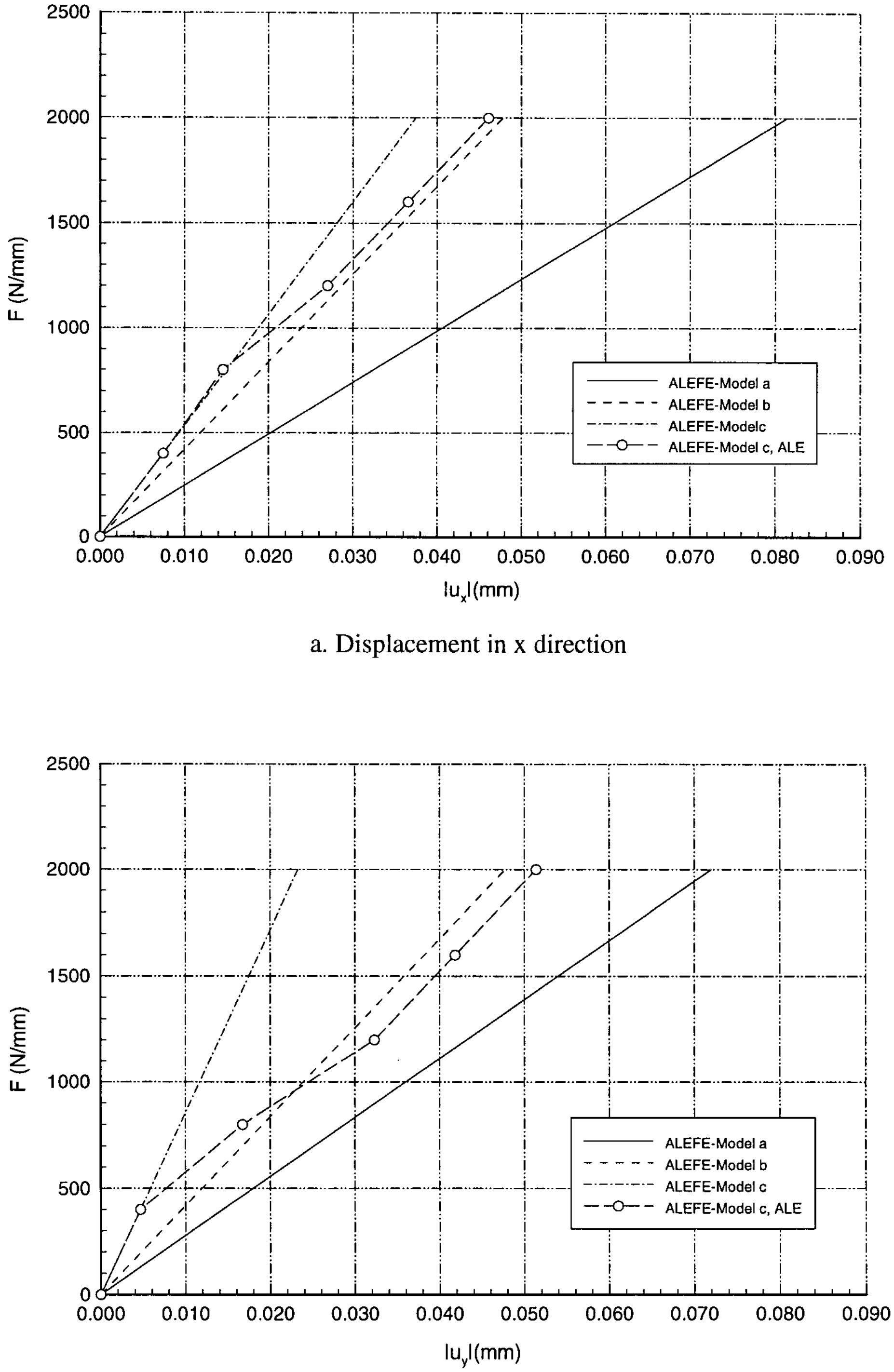
The same problem with model-c is also simulated by the ALE option of the program ALEFE. The results are compared with the Lagrangian ones from model-a, model-b and model-c in Figure 5.8. In the first incremental step, ALE and updated Lagrangian results coincide. This is obvious since ALE always starts with a updated Lagrangian increment for the problem without contact boundaries. In subsequent steps, the mesh begins to move, in ALE method, according to the scheme specified in Chapter 3 and mesh quality starts to improve, so that the results are closer to those from homogeneous mesh, model-b. After three steps, transition stage due to significant mesh motion seem to end and the response from ALE and from updated Lagrangian method are parallel to each other. The discrepancy is owing to the incremental method which causes some history-dependent effects in the simulation.

This example shows that ALEFE and NISA predict identical results when same methods are applied. The mesh distortion has significant effect on the accuracy of the results. Mesh distortion may cause negative Jacobian leading to termination of the simulation, as indicated by model-d. ALE method may eliminate mesh distortion and improve accuracy of simulation.

5.5 PUNCH FORGING PROCESS

5.5.1 Model and Updated Lagrangian Results

A punch forging process discussed in Chapter 1 is employed to demonstrate the capabilities of the ALE method and to compare with conventional Lagrangian methods. To show possible effects of different meshes on ALE and conventional updated Lagrangian method, homogeneous



a. Displacement in x direction

b. Displacement in y direction

Figure 5.8: Pure bending of a plate: ALE Results

and inhomogeneous meshes are used in simulation. The material properties, geometry and boundary condition have been described in Chapter 1. The problem is simulated by the developed program ALEFE and commercial codes NISA [10], ANSYS [9] and DEFORM2 [8]. Analysis is performed for up to 60% reduction in the height of the original piece.

The homogeneous mesh is shown in Figure 1.4 and is composed of square elements in the undeformed geometry. The results from NISA and ANSYS are obtained using updated Lagrangian method and are shown in Figure 1.5. For proper modeling of this problem, the material point under the punch corner should be free once it moves out of the corner. However, the Lagrangian formulation can not update the boundary condition automatically. This is an important feature in handling most of metal forming problems. This deficiency has the effect of increasing the punch size during the deformation, as shown in Figure 1.5. It is also noted that elements at the punch corner are highly distorted because of the large deformation. Increasing the deformation level causes more severe element distortions and the program eventually terminates the analysis.

The deformed shape from DEFORM2 is shown in Figure 1.7. In order to achieve convergence for the same deformation level obtained in NISA and ANSYS, a much denser mesh is required, as shown in Figure 1.6. Although remeshing eliminates the punch size increment, it may be seen that the mesh around the punch is highly distorted and more importantly, the predicted deformed boundary of the free surface near the punch corner is unusual and not rational. Another shortcoming is the fluctuations in loads predicted by Lagrangian method, which could not be eliminated by auto remeshing. Figure 1.8 shows a plot for the reduction of the specimen height versus the applied load, from DEFORM2. The result shows noticeable load fluctuation that is pertinent to the updated Lagrangian formulation.

5.5.2 ALE Simulation

In general, contact boundary condition may be easily handled in ALE by utilizing the feature of arbitrary mesh motion. On the interface of tool and work-piece, two specific directions are considered; perpendicular and parallel to the tool interface. In the direction perpendicular to the tool interface, the nodes will be kept as Lagrangian points, i.e., nodes will be moving with the tool at tool speed. In the other direction, however, nodes will be kept as Eulerian points, i.e., nodes will be fixed parallel to the tool interface. Contact boundary condition may be accurately described, therefore, in any general form of tool workpiece interface. The updating of boundary condition is no longer necessary. The "punch increasement", and load fluctuations discussed above, should generally not occur. This will undoubtedly increase the efficiency and accuracy of the simulation of metal forming problems.

In the particular case of punch forging process, as shown in Figure 1.4, the nodes on ED are kept as Lagrangian points in Y-direction, i.e., have the same velocity as the punch. The same nodes are treated, however, as Eulerian points in X-direction, i.e., have a zero velocity. This ensures a node remains coincident with each punch corner in the whole deformation process. The nodes on EO are Lagrangian in X-direction and general ALE ones in Y-direction where the speed linearly reduces from v at point E to zero at point O. The nodes on OA are Lagrangian in Y-direction and Eulerian in X-direction. On AB, nodes are treated as Lagrangian in Y-direction, and general ALE in X-direction. Boundaries DC and CB are traction free, so all the nodes are Lagrangian in both directions. The mesh motion scheme described in Chapter 3 is employed to control the internal mesh motion. The simulation is successfully carried out for the whole course of deformation.

The results of deformed shape and current yield stress from the ALE method are presented in Figure 5.9. It is obvious that the contact boundary condition for the nodes under the punch is handled automatically and accurately by allowing the degree of freedom in the

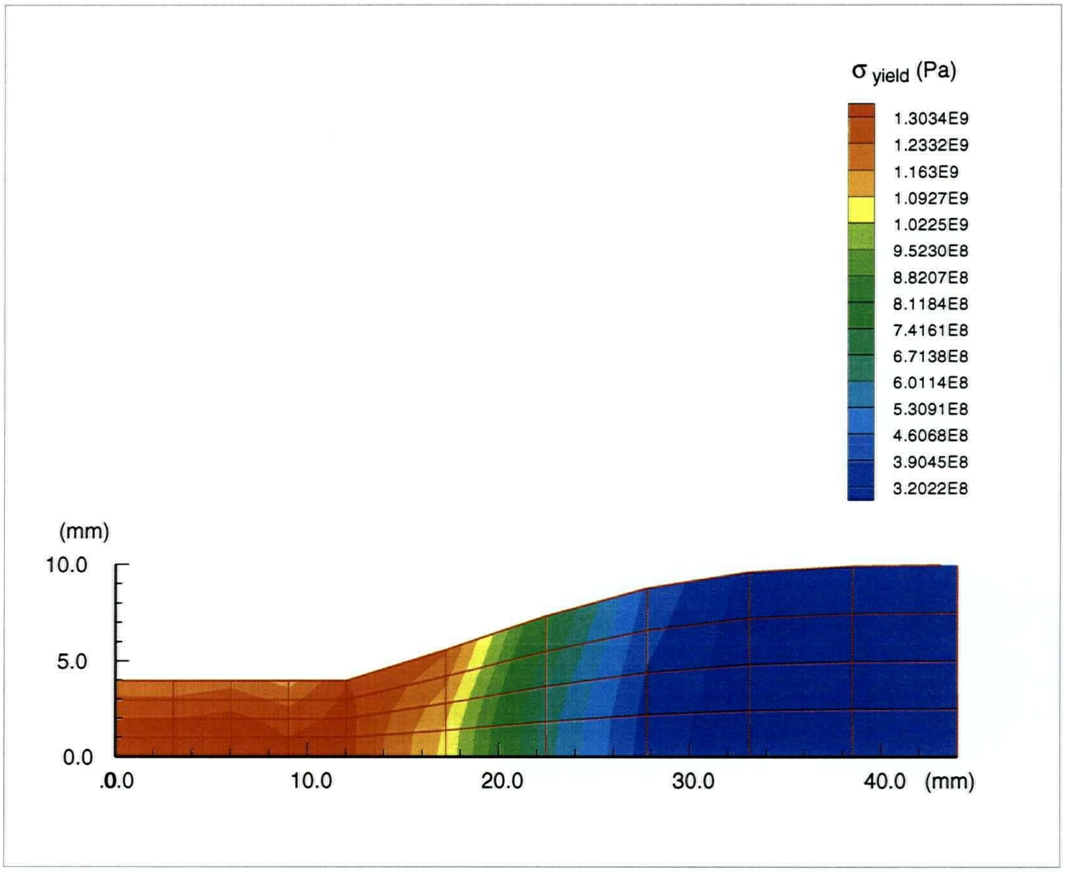


Figure 5.9: The current yield stress from ALE

vertical direction to be Lagrangian while freezing the degree of freedom in the horizontal direction and the punch size increment is eliminated. The deformed shape near the punch corner is acceptable with no element distortion. Both displacements and stresses obtained from the ALE are in agreement with the results reported in [69]. Figure 5.10 shows the load versus reduction curve from the ALE simulation. The load fluctuation caused by updated Lagrangian formulation (Figure 1.8) is completely eliminated in the ALE results. That is believed to be a result of accurately describing the contact boundary condition in the whole deformation process. Figure 5.11 shows contours of normal stress in punch direction, σ_y. At the punch corner, the compression stress is maximum. In this area, stress values change dramatically. For the area under the punch, the compression stress reduces from the surface to the center. On the traction-free surface around corner, a small tensile stress area is noticed and values go down from surface to the center in most of the area.

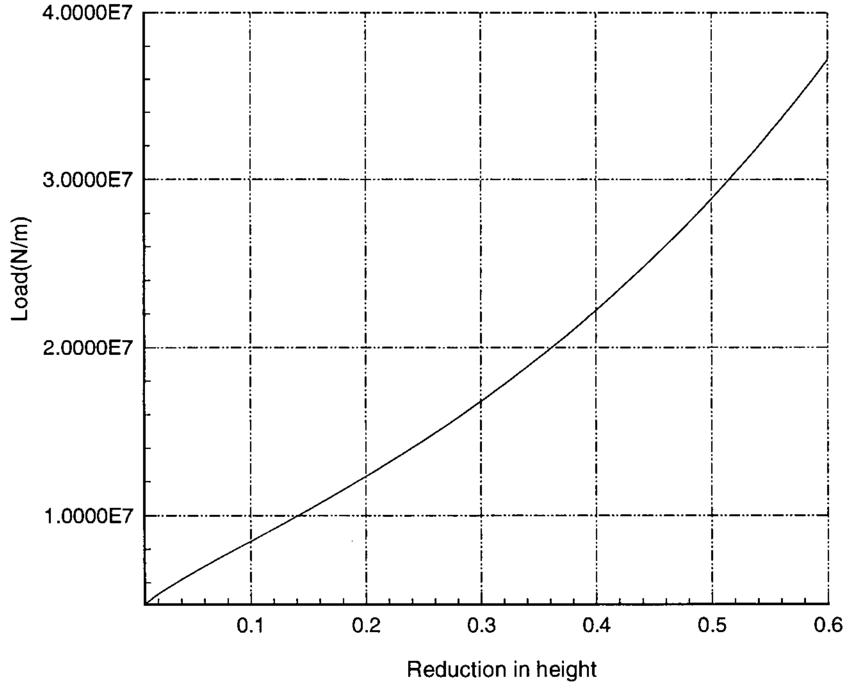


Figure 5.10: Load-height reduction result from ALE

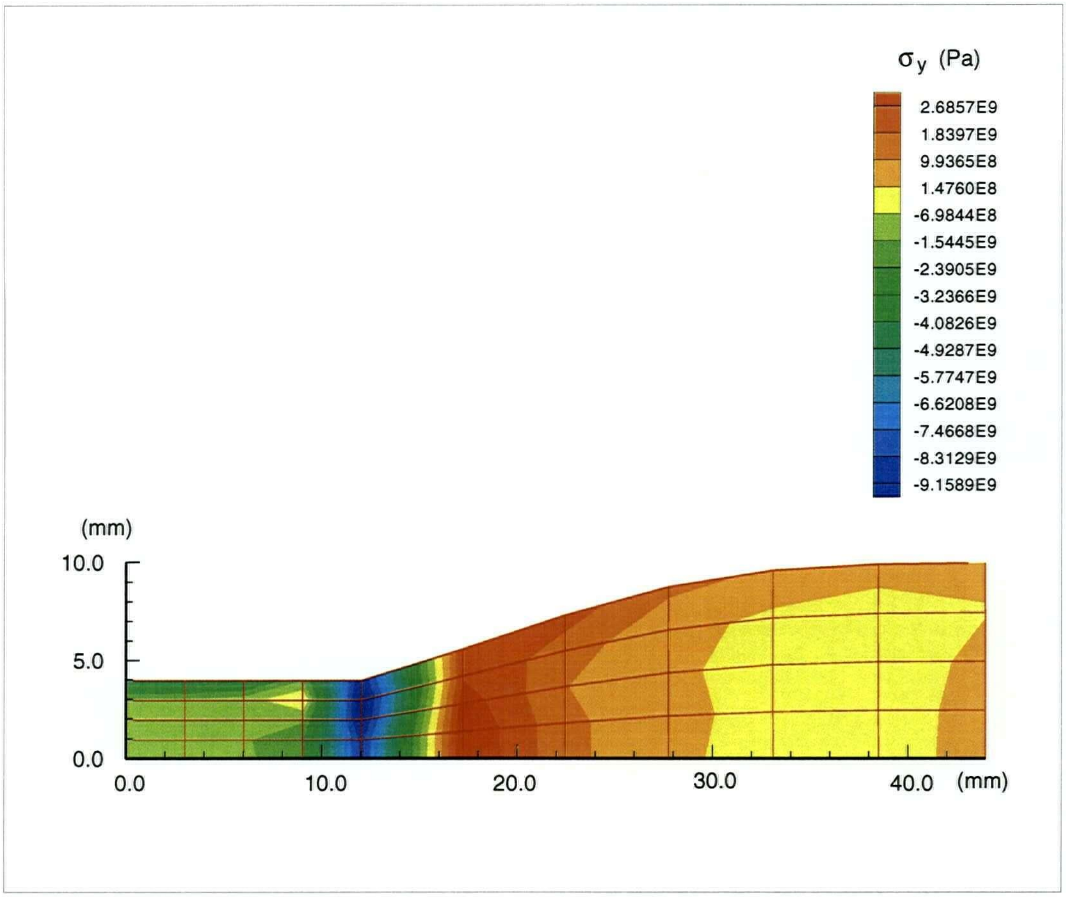


Figure 5.11: Contours of σ_y from ALE

Boundary node adjustment in the mesh motion scheme discussed in Chapter 3 is necessary. Figure 5.12 shows the deformed shape and current yield stress distribution from ALE simulation without the adjustment. It is clear that although the punch size is not increased in the simulation and the mesh is generally homogeneous, the aspect ratio of the first column of elements outside the punch corner may not be acceptable. More importantly, most of nodes on the traction-free surface moved to the far right side and no node is located close enough to the punch corner to represent the boundary shape, where the shape changes dramatically.

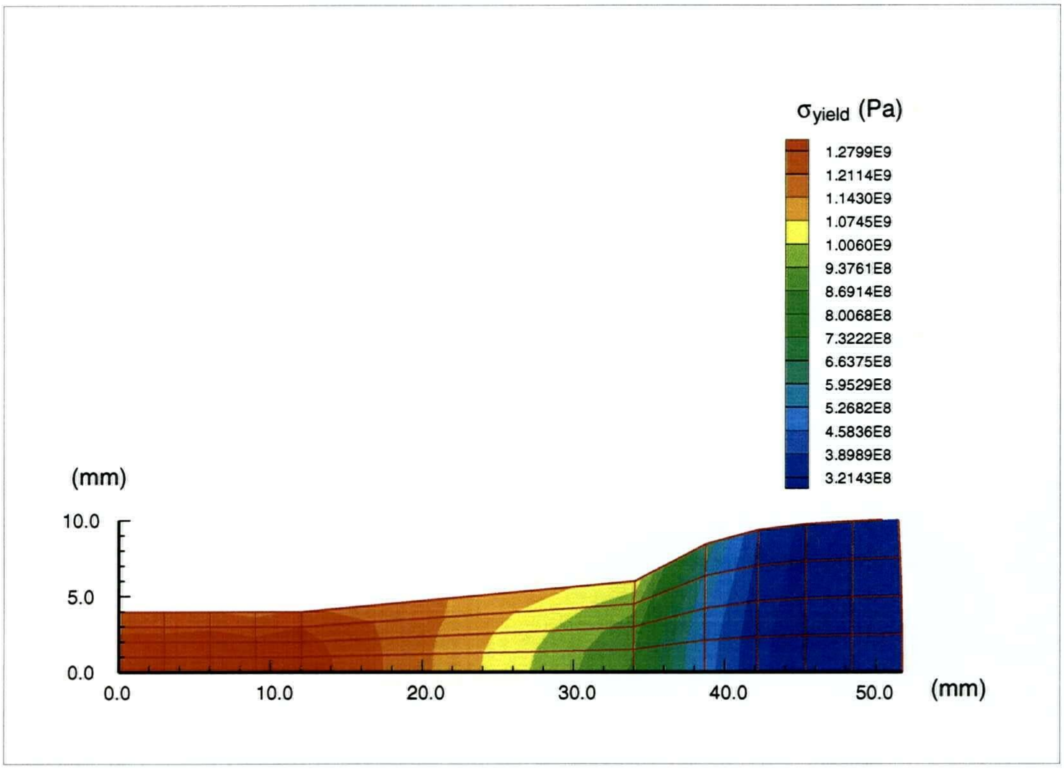


Figure 5.12: The current yield stress without boundary node adjustment

5.5.3 Simulations from an Inhomogeneous Mesh

An inhomogeneous mesh with same connectivity as the homogeneous mesh is applied as shown in Figure 5.13. Although the mesh is inhomogeneous, elements are still optimally shaped. The simulation is carried out using NISA, ANSYS and ALEFE with ALE option. The displacements of the top right corner, Point C, from all simulations are shown in Table 5.1.

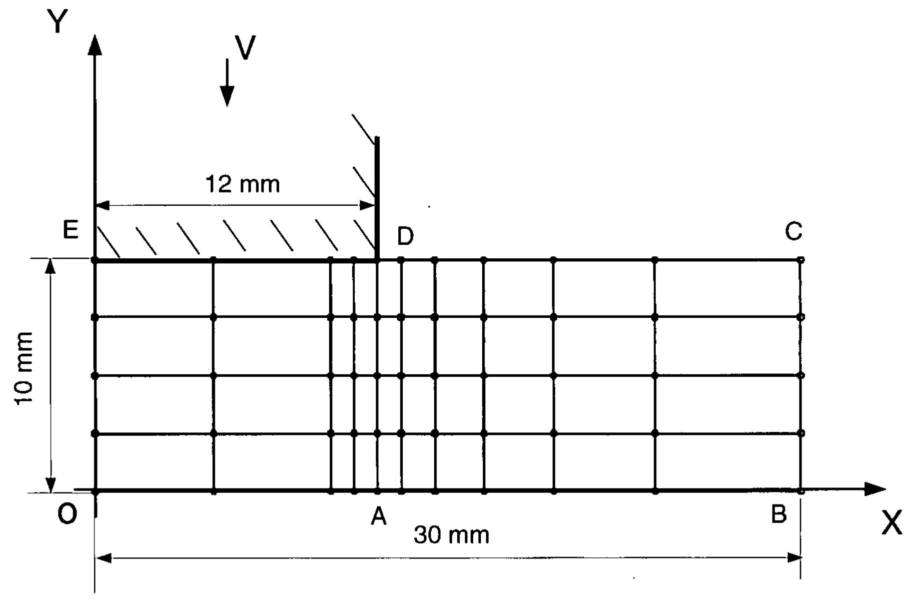


Figure 5.13: Inhomogeneous mesh for punch forging simulation

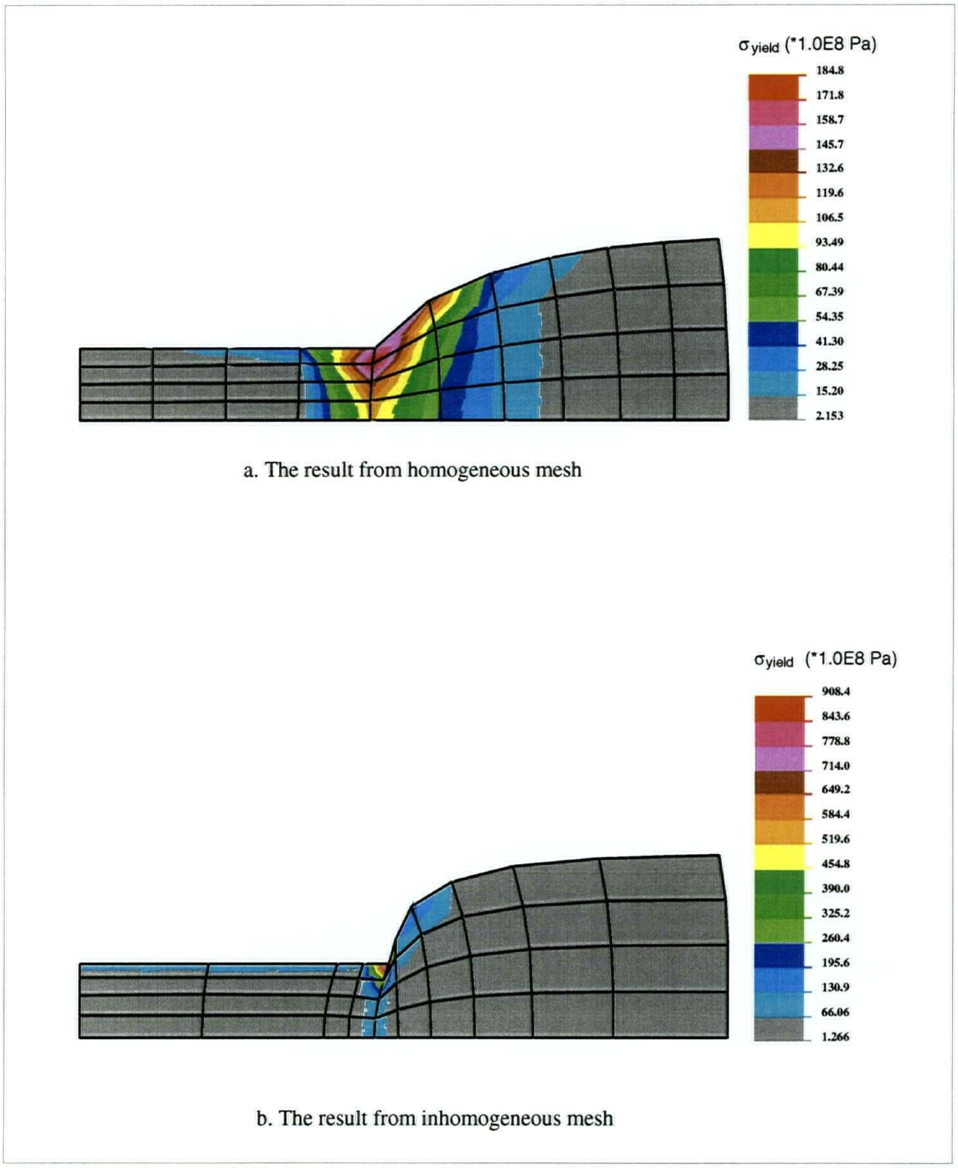
By comparing the stress and displacement distributions from different programs, it is found that although NISA gives similar displacements from homogeneous and inhomogeneous meshes, stress distributions are much different from the two meshes with stresses from the inhomogeneous mesh 2 to 10 times those obtained from homogeneous mesh. As an example, the current yield stresses are illustrated in Figure 5.14, in which the difference is about 5 times.

Table 5.1: Displacements of top right corner from all simulations

Program	u_x (mm)		u_y (mm)	
	Homogeneous	Inhomogeneous	Homogeneous	Inhomogeneous
NISA	15.459	14.583	0.003	0.012
ANSYS	18.741	18.652	0.096	0.021
ALEFE	13.929	12.019	0.010	0.040

ANSYS results show closer stress and displacement values and the predicted deformation shape is not improved. Shape and the mesh at the punch corners are severely distorted. The developed ALEFE program gives not only similar displacements, as indicated in Table 5.1, and quite close stress values, but also a realistic deformation shape. The largest difference of stress in ALE exists in τ_{xy} where the value from inhomogeneous mesh is 1.6 times that from homogeneous mesh. As an example, the current yield stress contour is shown in Figure 5.15. When compared with the result from homogeneous mesh, shown in Figure 5.9, the difference is less than 15%. The nodes are closer to the punch corner on the traction-free surface, so that the deformed shape around the corner is represented better.

The ALE results of the punch forging problem indicates that the "punch size increasement", load fluctuation and mesh distortion caused by traditional updated Lagrangian method may be eliminated by the ALE scheme. The boundary node adjustment in mesh motion is necessary to keep high quality mesh and to achieve high accuracy. Inhomogeneous meshes and changes in mesh density do not affect the result significantly. Accuracy may be improved if mesh has higher density at locations where deformation is highly inhomogeneous, as at the punch corners. In the traditional updated Lagrangian method, however, such density changes may decrease the calculation accuracy and lead to significantly different results. The discrepancy is believed to be caused by the severe mesh distortion which occurrs in dense meshes.



a. The result from homogeneous mesh

b. The result from inhomogeneous mesh

Figure 5.14: Comparison of current yield stresses from NISA

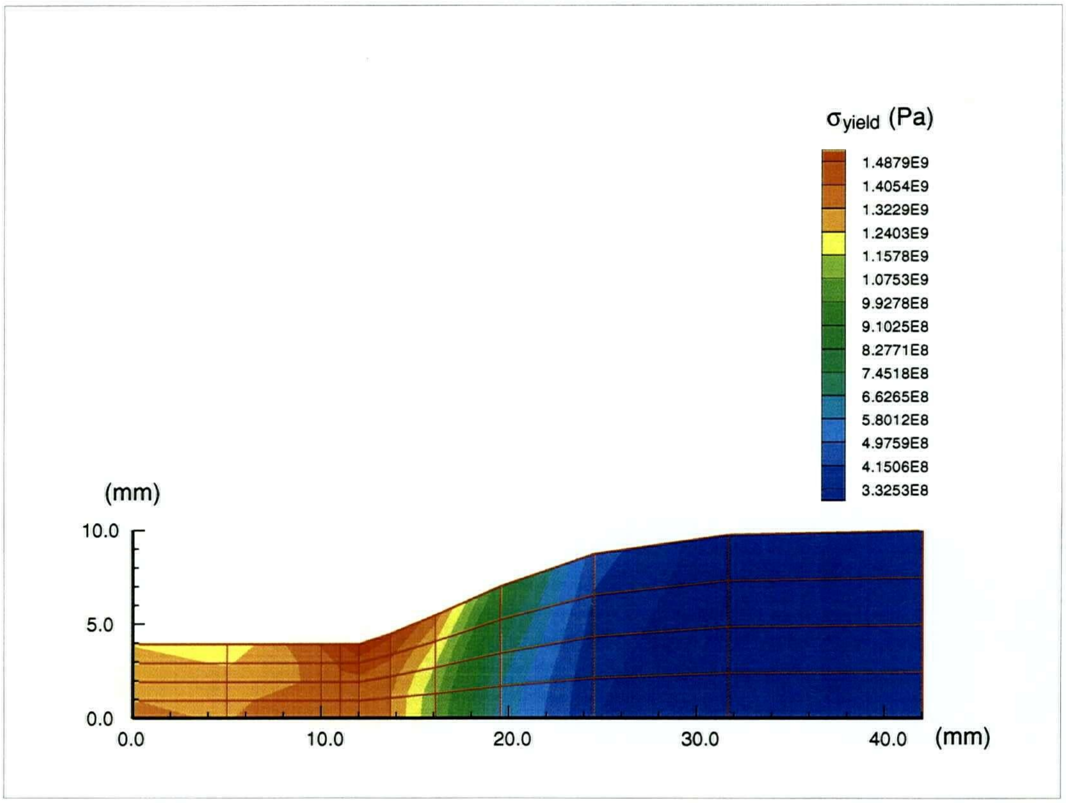


Figure 5.15: The ALE result of current yield stress from inhomogeneous mesh

5.6 TENSILE NECKING ANALYSIS OF AN ELASTIC-PLASTIC BAR

The initiation of tensile necking of an elastic-plastic bar in plane strain is simulated by updated Lagrangian method, which is degenerated from the ALE formulation. The objective is to check the correctness of the method and to compare the integration algorithms discussed in Chapter 3. The different integration algorithms used in this example are: integration algorithm of Truesdell rate discussed in Chapter 3, Equation (3.15), integration algorithm of Truesdell rate developed by Pinsky [58], Equation (3.20), and integration of Jaumann rate developed by Hughes [56], Equation (3.21), combined with transformation A and B discussed in the chapter.

The bar has a length to width ratio of 3 and the central one-sixth of the length is thinned slightly so that the necking analysis may be formulated as a standard deformation problem instead of an eigenvalue bifurcation problem. Only one quarter of the bar is modelled due to symmetry and the finite element model is shown in Figure 5.16. There are 256 quadrilateral 4-node elements and 297 nodes in the model. The material has a Young's modulus to yield stress ratio of 7.5×10^2 , Poisson's ratio of 0.3 and the ratio of hardening modulus to yield stress is constant at 1.25 with a yield stress value of 250 MPa. The length L is taken as 30.0 mm and $W_0 = 10.0$ mm in calculation.

The amplitude δ of the neck is defined by the width reduction at center section normalised by the initial length L [3]. The simulation is carried out until δ is about 0.07. The same problem is also solved using the commercial finite element code ANSYS [9]. Results are compared with the results reported by McMeeking and Rice [3]. The necking parameter δ is plotted against the engineering strain $\gamma = \Delta L/L$. In Figure 5.17, Jaumann stress rate results are labelled as Jaumann-A and Jaumann-B. These correspond to the use of Jaumann rate, Equation (3.21), with transformation-A and transformation-B, respectively. Similarly, Truesdell-A and Truesdell-B are from the integration of Truesdell rate, Equation (3.15), with transformation-A and B, respectively.

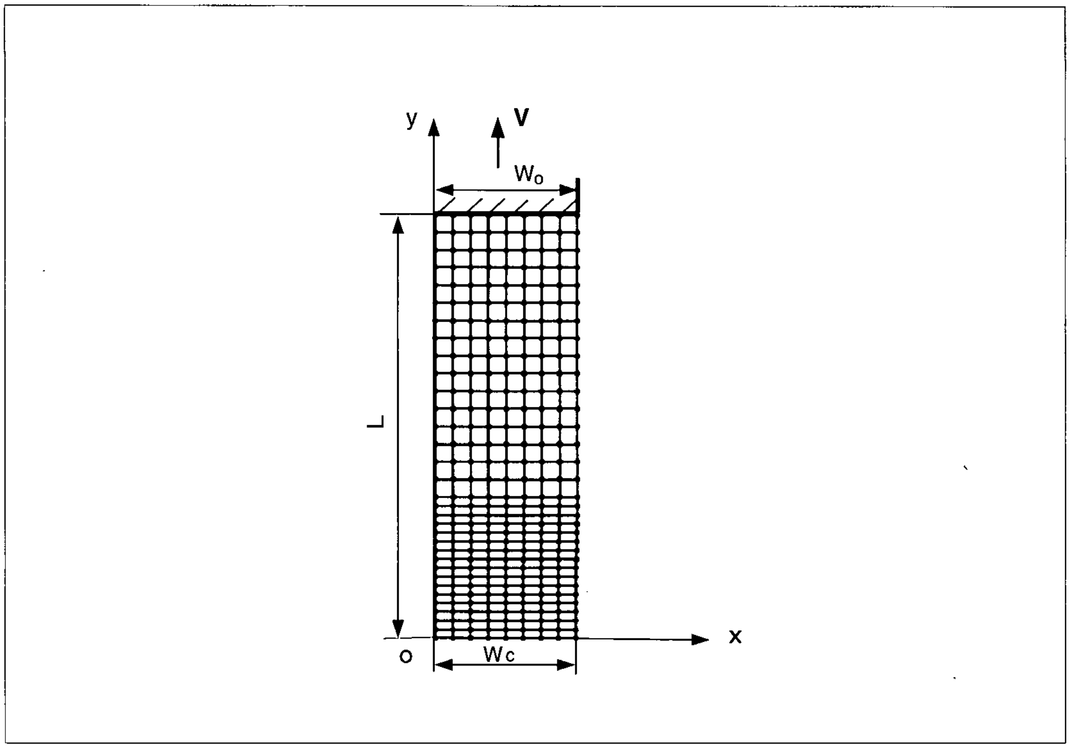


Figure 5.16: The finite element model for tensile necking

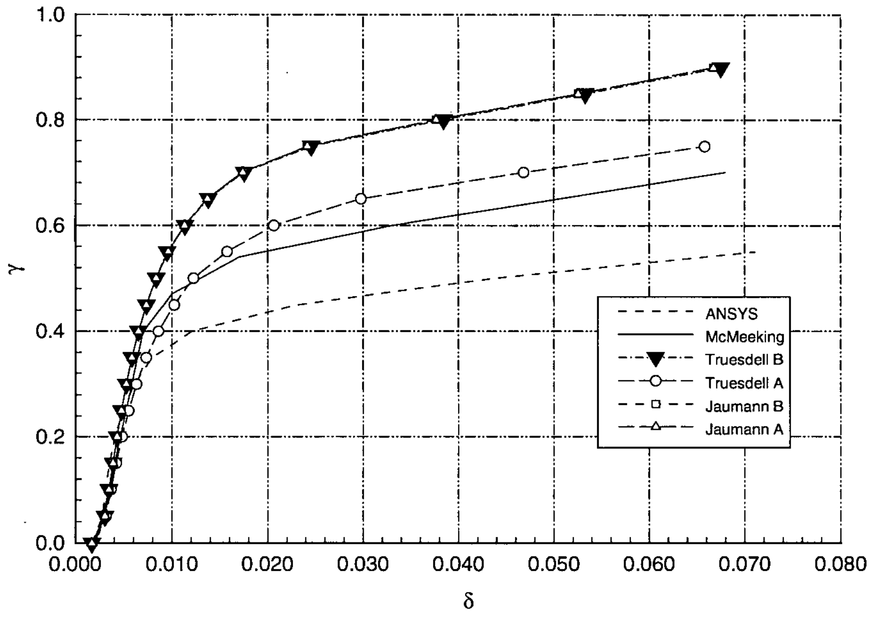


Figure 5.17: The necking δ and engineering straining γ

It can be seen from Figure 5.17 that for integration of Jaumann rate, the transformation algorithm A and B lead to same results. For Truesdell rate, transformation-A and B give quite different results. This indicates that Truesdell rate is more sensitive to the location of stress transformation. It should be mentioned here that the results from Truesdell A scheme have almost same slopes and very close values to those reported by McMeeking [3]. The results from ANSYS show very much different values of necking. The ANSYS run was terminated at $\gamma = 0.57$ because of divergence, whereas all the runs of the developed ALE program have been successful for values up to $\gamma = 1.0$.

Figure 5.18 shows results from different schemes developed by Pinsky et al. [58] for the integration of Truesdell rate, Equation (3.20) in this thesis, (curves: Truesdell-A, P and Truesdell-B, P). Figure 5.18 also shows corresponding results developed in this research, Equation (3.15), (curves: Truesdell-A and Truesdell-B). The results indicate that schemes developed in this research give same results as those developed by Pinsky et al. [58]. This is expected since the difference between Equation (3.15) and Equation (3.20) is only in the second term on right hand side and is caused only by higher order terms of Δt.

Figure 5.19 shows the contours of the equivalent von-Mises stress and the ratio of the equivalent stress to current yield stress at $\gamma = 0.75$. Although no stress distribution is reported in [3] , the shape of unloading zone is discussed. From Figure 5.19, the zone where the ratio of the equivalent stress to current yield stress is less than 1.0 is experiencing unloading. The unloading zone shown here has the same shape and location as that reported in [3].

The example indicates that the transformation algorithm A and B may generally be combined with any integration algorithm. The integration of Jaumann rate is less sensitive to the location of stress transformation. The combination of Truesdell rate with Transformation-A generally gives better results than other cases. Algorithms for integration of Truesdell rate developed in this research gives same results as those developed by Pinsky [58]. Similar conclusions to those reported above have been obtained through simulation of the punch forging

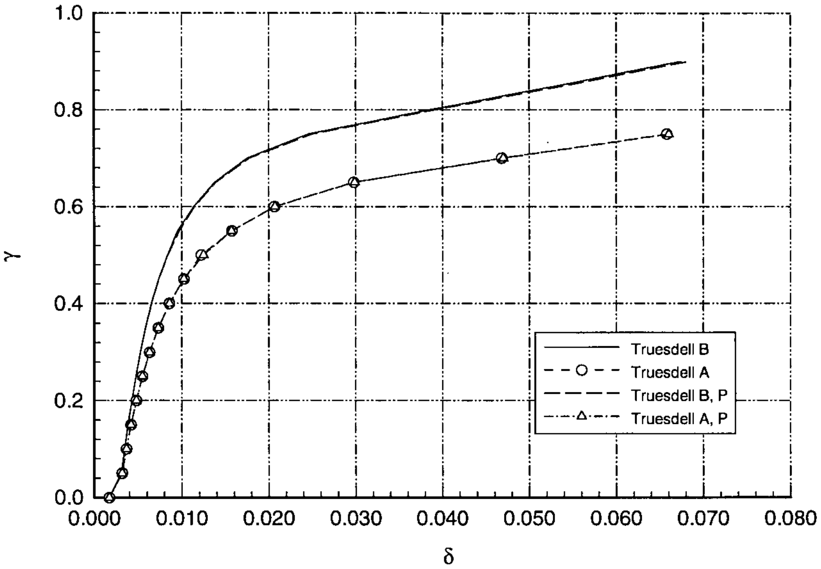


Figure 5.18: Comparison of two algorithms for Truesdell rate

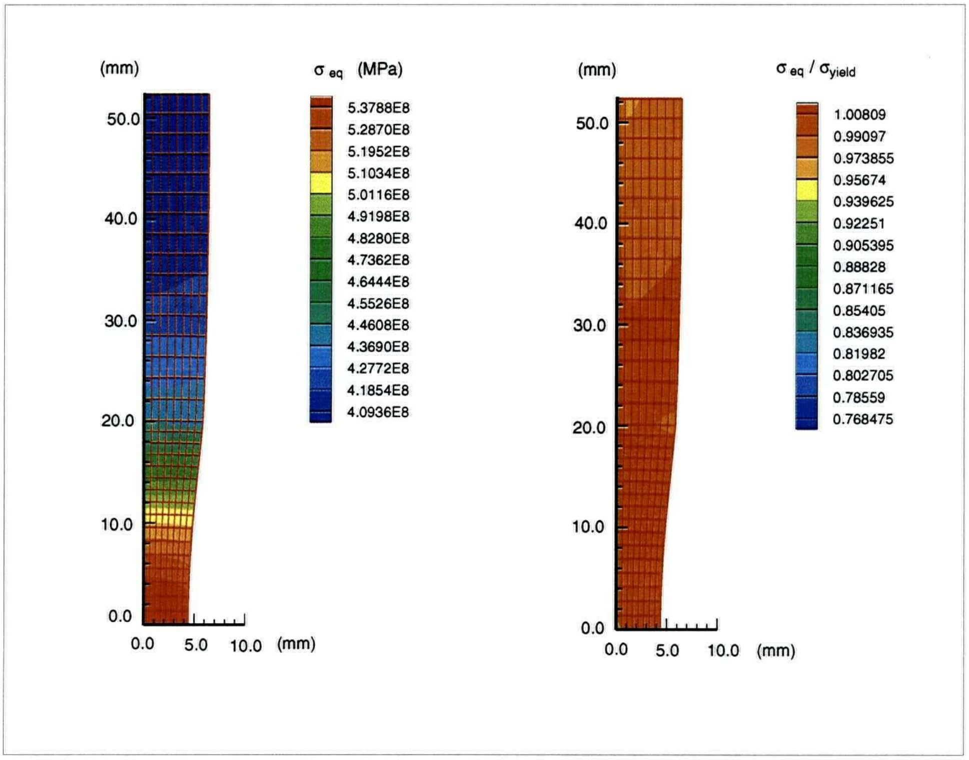


Figure 5.19: Contours of equivalent stress (Truesdell rate, transformation A)

problem by ALE method.

5.7 SHEET METAL EXTRUSION

5.7.1 Extrusion through a Curved Die

Plane strain sheet metal extrusion is another typical metal forming problem in which large strains are expected and boundary condition updating as well as re-meshing are required in traditional updated Lagrangian methods. The ALE method is applied to simulate two extrusion cases to show the advantage of ALE method in such deformation problems where the deformation extent is independent of the displacement of the tool after certain steps, i.e., when piston moves a distance equal to the length of the die. The first simulation (curved die) has been taken as a bench mark problem by many researchers [7, 70] whereas the second simulation has some limited experimental results available in the literature.

An aluminium billet in an initially stress free state is forced into the die while sliding between two smooth rigid plates. A rigid smooth piston pressing against the rear face of the billet and moving with prescribed velocity provides the driving force. The die produces 25% thickness reduction over a distance $1.2a$, where a is the half thickness of the original sheet and taken as 25.4 mm (1 in) as shown in Figure 5.20. Material properties of the Aluminium billet are: Young's modulus $E = 6.89 \times 10^4\ MPa\ (1 \times 10^7\ psi)$; hardening modulus $H = 1.14 \times 10^3\ MPa\ (1.65 \times 10^5\ psi)$, initial yield stress $\sigma_{yield0} = 3.95 \times 10^2\ MPa\ (5.7 \times 10^4\ psi)$, and Poisson's ratio $\nu = 0.3$. The example is the same as the one discussed in Chapter 1, except that the die is shaped in the form of 5th order polynomial curve with zero curvature and slope at the ends. The same process was simulated with updated Lagrangian method by Lee [12] and by Yamada [70], where the billet was assumed to have been initially machined to fit the die. Figure 5.20 shows the geometry and the finite element mesh used in ALE simulation. In the ALE simulation, the mesh and the corresponding nodes confined to the die area are taken

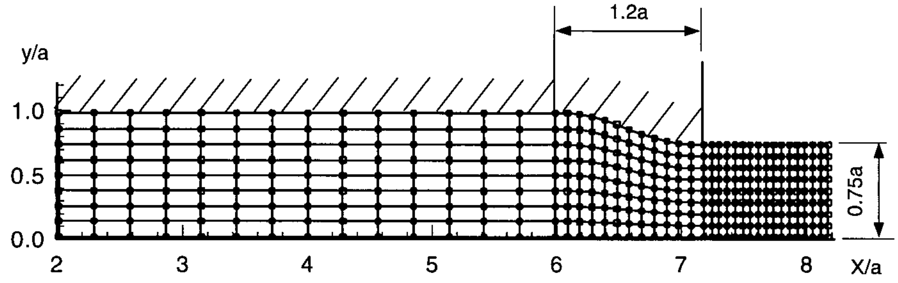


Figure 5.20: Original geometry and mesh in extrusion through a curved die

as Eulerian during the whole course of deformation. The nodes on traction free boundaries are designed to be completely Lagrangian, whereas those on symmetric axes, contacting with piston, or sliding on plate are taken as Lagrangian in the normal direction and the general ALE in tangential direction. All other nodes are assumed to be ALE ones and are controlled by the mesh motion scheme detailed in Chapter 3.

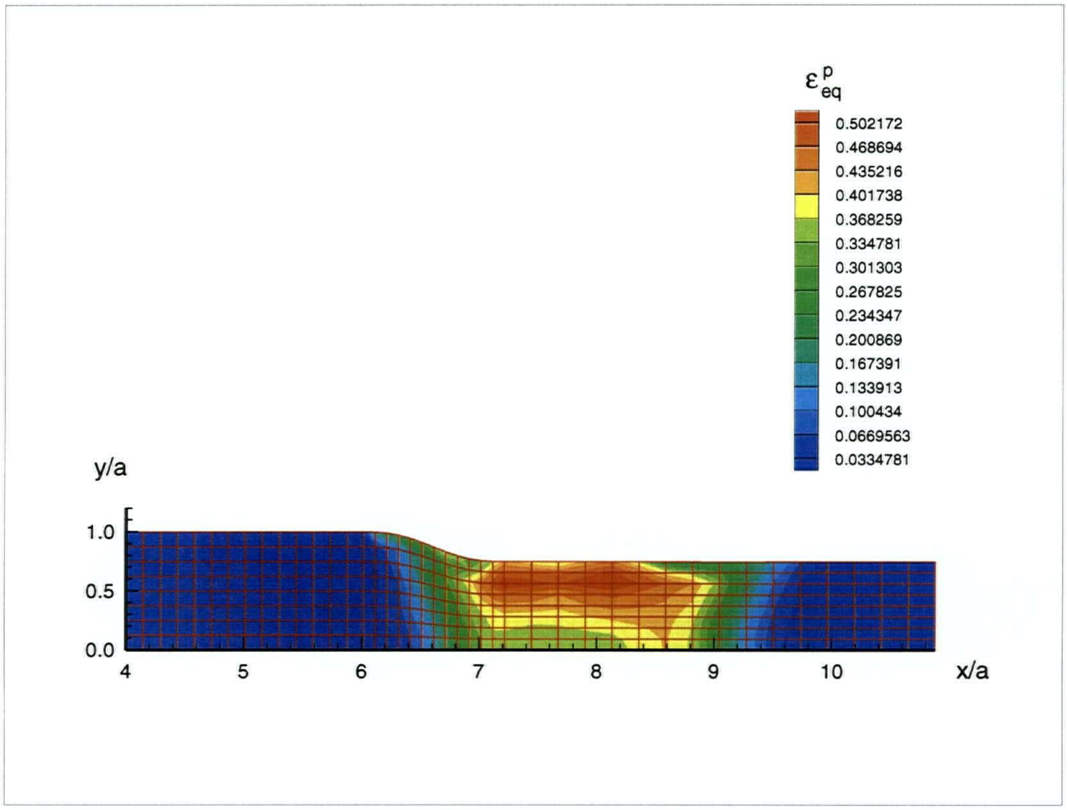


Figure 5.21: Contours of plastic strain after piston moved $2a$

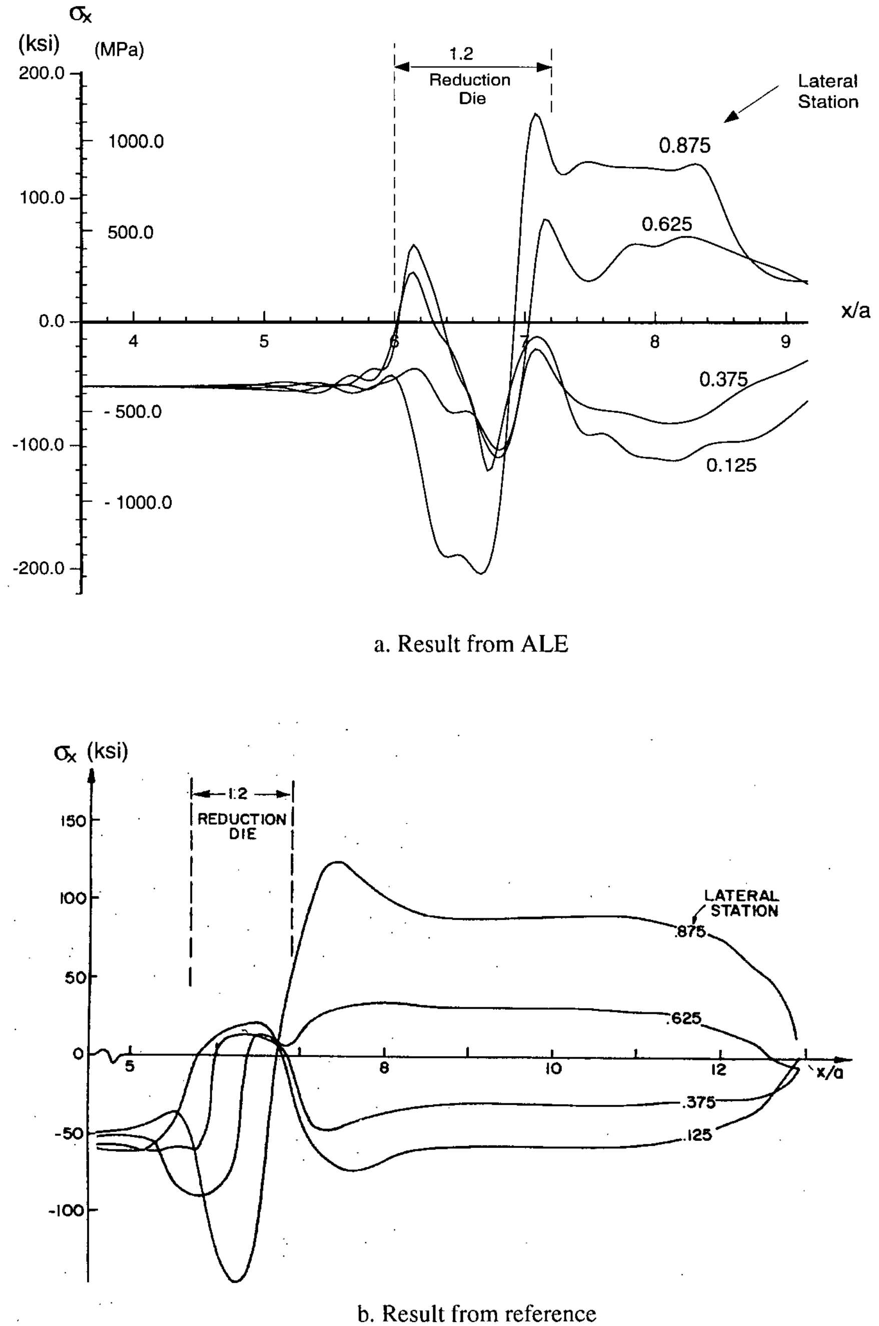
Figure 5.21 shows the plastic strain distribution and the deformed shape after a piston movement of $2a$. As shown from the figure, a piston displacement of $2a$ corresponds to approximately 2.0 times the width of the die. The figure shows the shape of plastic zone and how it is expanding. The distribution of longitudinal stress σ_x at different lateral positions, labelled "lateral levels", is also shown in Figure 5.22. A lateral level corresponds to the initial distance from the central line of the sheet. Similar results from reference [12] are shown in

Figure 5.22 as well. The values of shear stress τ_{xy} obtained from the two methods are illustrated in Figure 5.23. Generally, good agreement is observed. The results of reference [12] were obtained from a sequentially updated Lagrangian scheme with special introduction of initial stress stiffness terms as well as dilatation and rotation terms in the stress updating scheme. Special attention was given when applying the boundary condition at the die face and after the material leaves the die face. Numerical and computational difficulties are reported in [12] in the process of correctly applying and updating the boundary conditions. The treatment described is essentially unique for the given application.

5.7.2 Extrusion through a Straight Die

Plane strain metal extrusion is investigated experimentally and analyzed with slip line field method by Farmer and Oxley [71]. Measuring the die separation force is not practical and would greatly complicate the design of the apparatus. Therefore, only the extrusion force is measured and the friction on the die interface is minimized by effective lubrication. The half die-angle is 30° and the thickness reduction ratio is 50%. The work-piece has cross-section 2.54 mm (0.1 in) thick × 10.16 mm (0.4 in) wide, the thickness being reduced to 1.27 mm (0.05 in) during extrusion. In order to reduce the peak load and to reduce the time taken to reach steady-state condition, the work-piece is pre-formed to the angle of die. The material is 5052-H34 tempered aluminium. A stress /strain curve for this material obtained from compression test is applied to derive the initial yield stress of $\sigma_{yield0} = 267.7\ MPa$ and strain hardening modulus $H = 85.9$ MPa. Other material properties are: Young's modulus $E = 6.9 \times 10^4\ MPa$ and Poisson's ratio $\nu = 0.32$ [72].

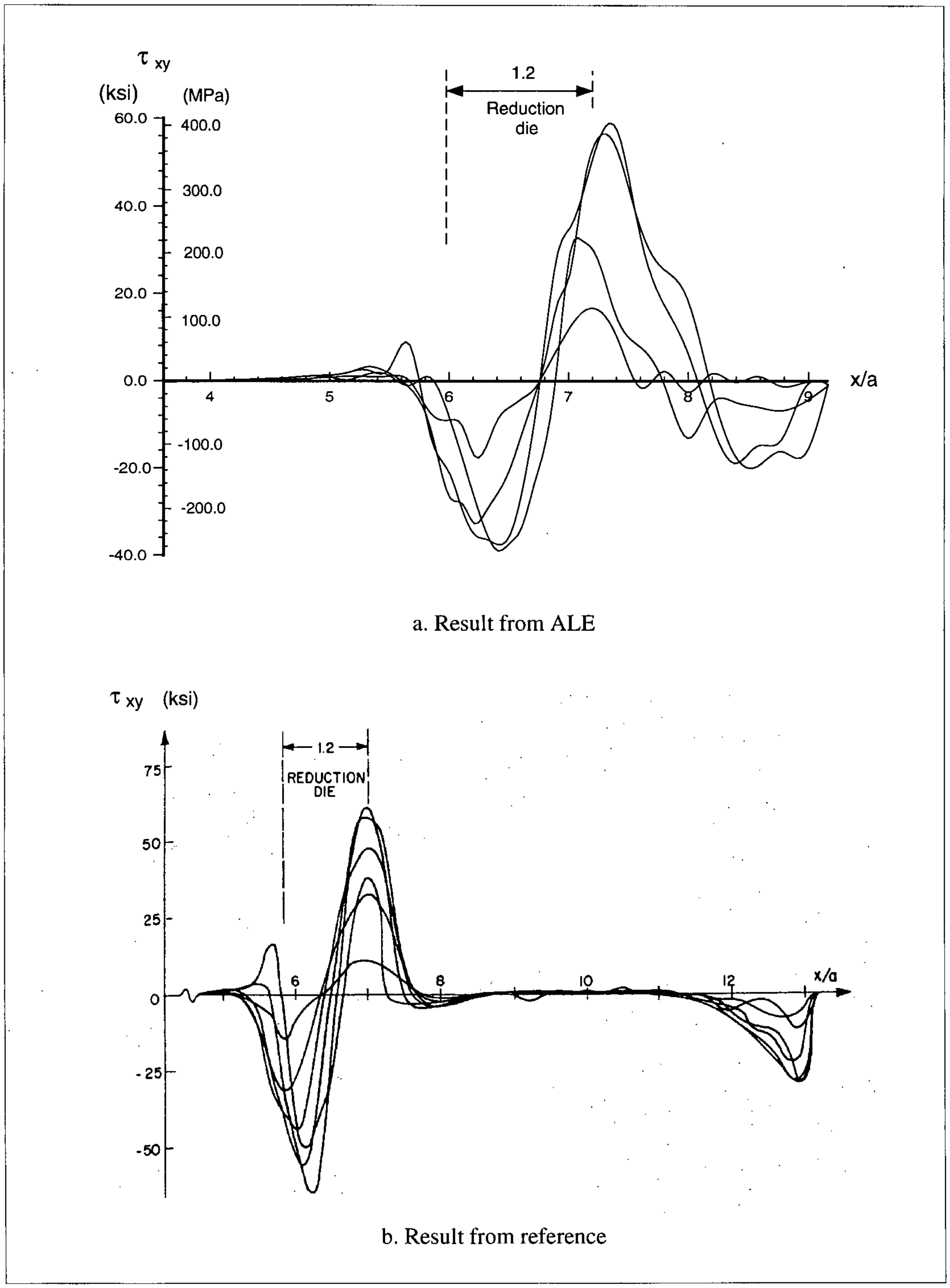
Because the geometry and boundary condition are symmetric, only half of the billet is considered. The finite element model is shown in Figure 5.24. The same problem is simulated by NISA and ANSYS. In the ALE simulation, all the nodes in patch BCFG in Figure 5.24



a. Result from ALE

b. Result from reference

Figure 5.22: Comparison of σ_x from ALE and reference [12]



a. Result from ALE

b. Result from reference

Figure 5.23: Comparison of τ_{xy} from ALE and reference [12]

are kept as Eulerian points, so that the contact boundary conditions of the die work-piece are satisfied throughout the deformation history. The nodes on boundaries AB, CD and HG are kept as Lagrangian points in Y direction and general ALE points in X direction. The nodes on AH, FE and ED are treated as Lagrangian points to keep one-to-one mapping between mesh domain and material domain. Once again, the internal points are handled by the mesh motion scheme discussed in Chapter 3. The piston motion is maintained until some distance after the steady deformation is achieved.

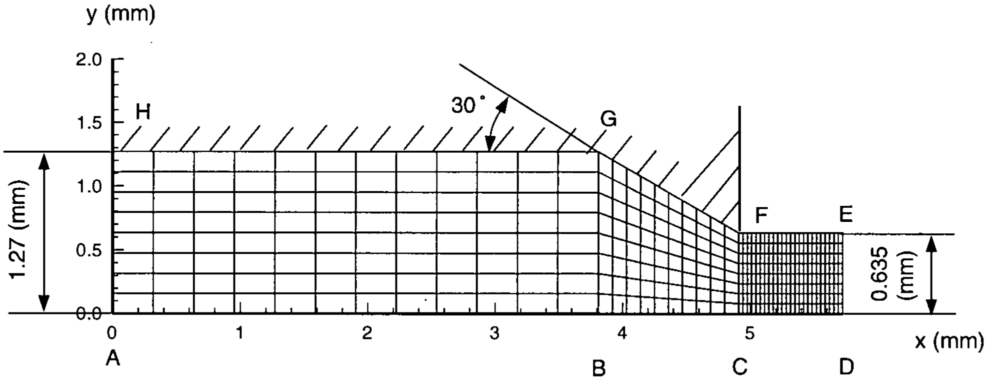


Figure 5.24: The finite element model for straight die extrusion

Figure 5.25 shows contours of current yield stress (plastic zones) and the deformed mesh from ALE after a piston motion of 1.2 mm . It may be seen that the mesh is homogeneous in the whole domain with the highest deformation area just before the exit around the die corner. This is different from the curved die, that was considered in the previous section, where the largest deformation is located just after the exit. This may be attributed to the zero curvature and slope at the exit.

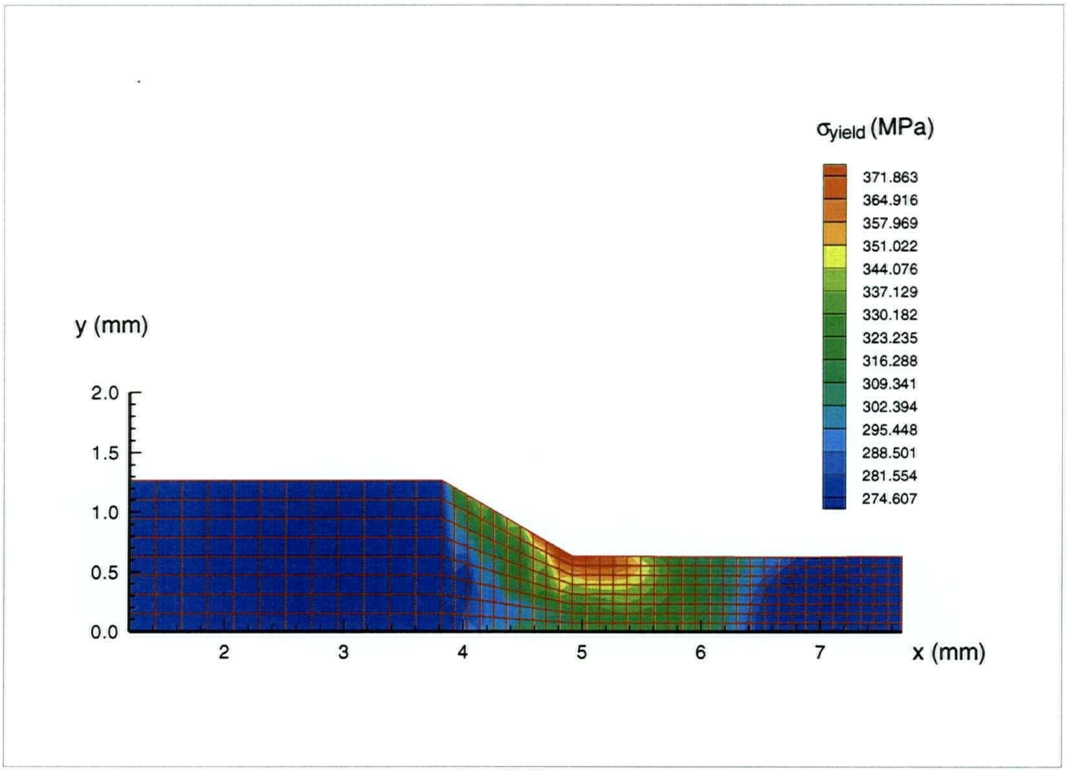


Figure 5.25: The current yield stress (plastic zone) from ALE

With the same mesh used above, both NISA and ANSYS analyses were terminated automatically by the programs after a piston motion of only about 0.18 mm because the mesh was entangled around the exit of the die. Therefore, a coarser mesh is used for NISA and ANSYS simulations, as shown by the gray lines in Figure 5.26. With the new model, NISA results were successfully obtained for a piston motion of 0.54 mm. The final distorted mesh at the end of the run is shown by the black lines in Figure 5.26. It may be noted that most of the distortion is due to the in-ability to update the boundary conditions. For the ANSYS run, the program did not converge at 0.19 mm of piston displacement and the simulation was terminated due to excessive mesh distortion.

Results of the extrusion forces from ALE, ANSYS and NISA are compared with the

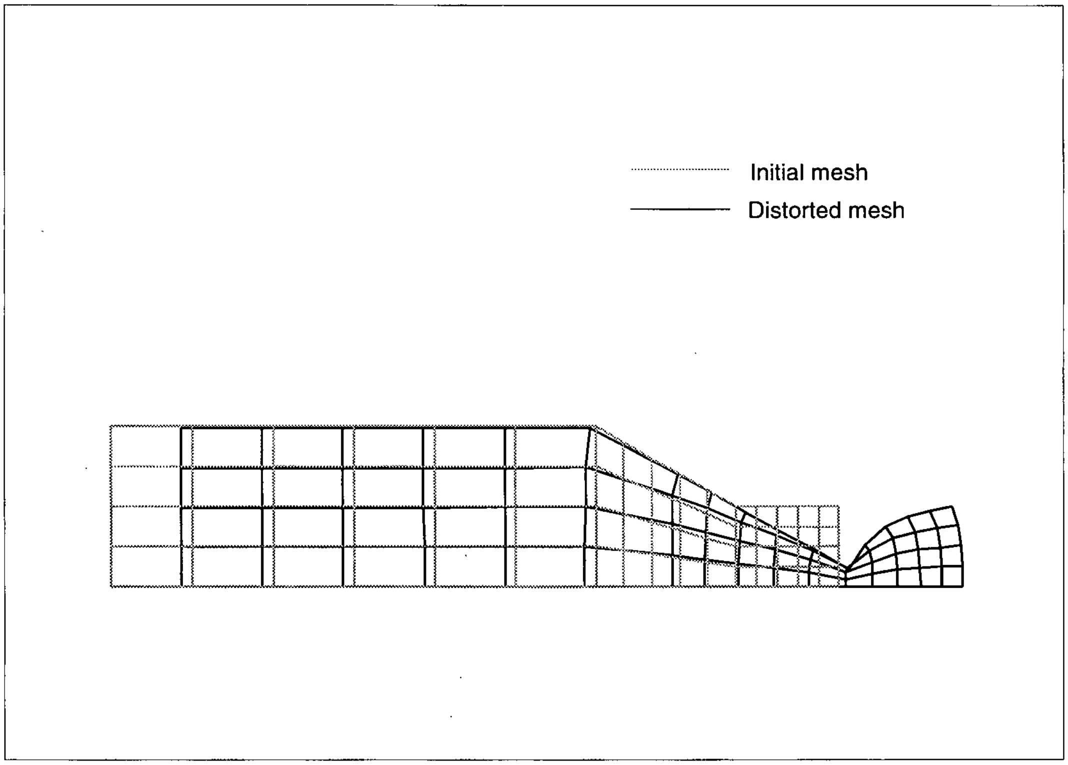


Figure 5.26: The initial and deformed mesh from NISA

experimental one from [71], as shown in Figure 5.27. It is observed that the extrusion force from ALE is in close agreement with the experimental one. The ALE and experimental one show a steady force after the piston moves about 0.2 mm. The extrusion forces from NISA and ANSYS clearly did not show this behavior. An important reason for this is believed to be that the contact boundary condition is not updated automatically, so that the reduction in height increased with the piston movement, which is not practical. Figure 5.28 shows a comparison of the pressure distribution on the piston face from ALE with the one from slip line method [71]. It is observed that the pressure from ALE is in good agreement with the result from slip line theory except at points close to the surface of the work-piece (i.e., $y = 1.27$ mm). The reason of this difference is not clear, but at $y = 1.27mm$, the result from slip line theory give a tension stress, which is not reasonable.

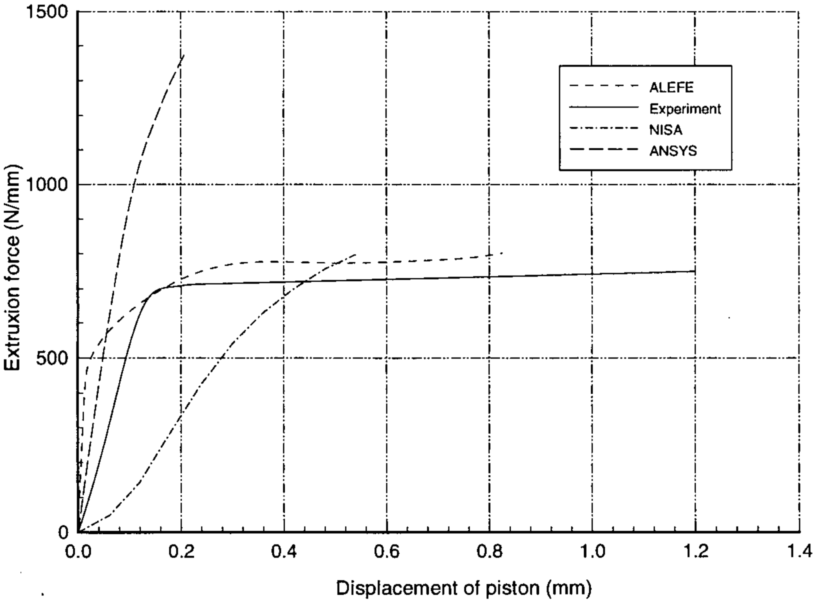


Figure 5.27: Comparison of extrusion force

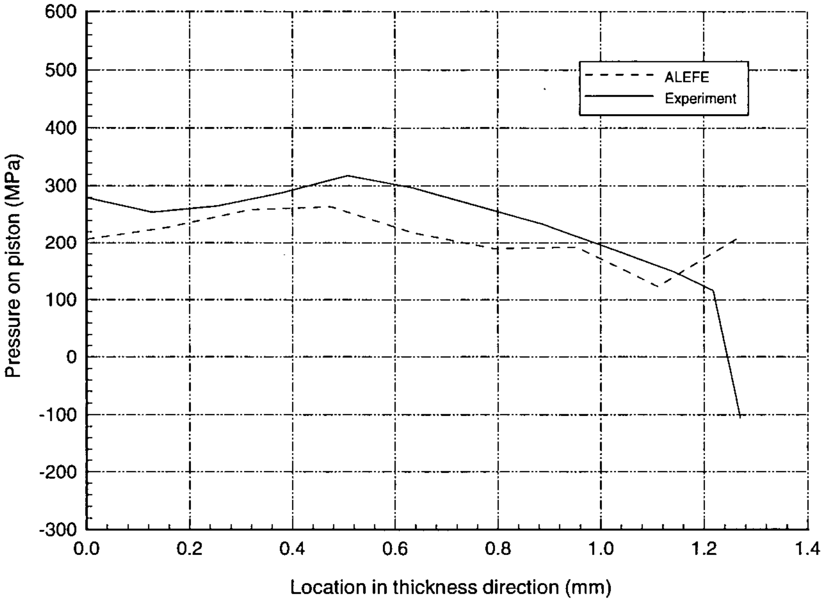


Figure 5.28: Pressure distribution on piston face

5.7.3 Discussion

The metal extrusion examples indicates that ALE formulation is able to handle changes in the kinematic boundary conditions easily and keep mesh homogeneous in the whole deformation process. In traditional updated Lagrangian method, this may only be achieved by particular boundary updating scheme and a specially developed program, as in [12]. The effect of die shape on the plastic zones is clearly reflected in the results. The stress, extrusion force and pressure distribution generally agrees with other numerical results from the literature [12], as well as experiment and slip line [71] results.

5.8 STRIP ROLLING

5.8.1 Particular Difficulties in Rolling Process

Rolling is one of the oldest industrial metal working processes. In view of the tremendous amount and wide variety of rolled products manufactured every year, it may be considered as one of the most important. For more than half a century, numerous investigations, both analytical and experimental, have been carried out on rolling [73]. Slab analysis, slip-line field and upper- and lower-bound methods have been widely used in theoretical analyses of rolling. However, owing to the complexities of deformation involved in rolling, various degrees of simplification and idealization are necessary. These methods have provided useful but limited information in rolling. The increase in quality requirements and the concern for maintaining and improving the competitiveness of rolled products have led to demands for more detailed information and understanding of the process. Under more realistic conditions, accurate determinations of detailed metal flow in rolling processes became possible only when the finite element method was introduced into the analysis.

Numerical simulation of rolling is challenging. Besides boundary condition updating and mesh distortion due to large deformation, friction on the roll work-piece contact boundary is more important and complicated than other forming processes. Because of the change in relative speed between the roll and the work-piece, the friction force along the contact surface (arc GF in Figure 5.29) changes its direction at the neutral point N, as shown in Figure 5.29. The location of neutral point changes during the non-steady state deformation and becomes steady at somewhere between the exit section and the halfway point of the contact surface. The location of the neutral point depends mainly on the friction condition and is not known a prior. A difficulty arises when attempting to implement Coulomb friction model in which the tangential frictional stress distribution is dependent on the normal pressure distribution. Some researchers developed particular formulations to handle the friction [74]; others employ friction layer or interface elements [42, 75]. In this research, we do not introduce the particular formulation to handle rolling simulation. Instead, an approximate method based on the general program used above is applied. Rolling process is simulated by Eulerian method reduced from the general ALE formulation.

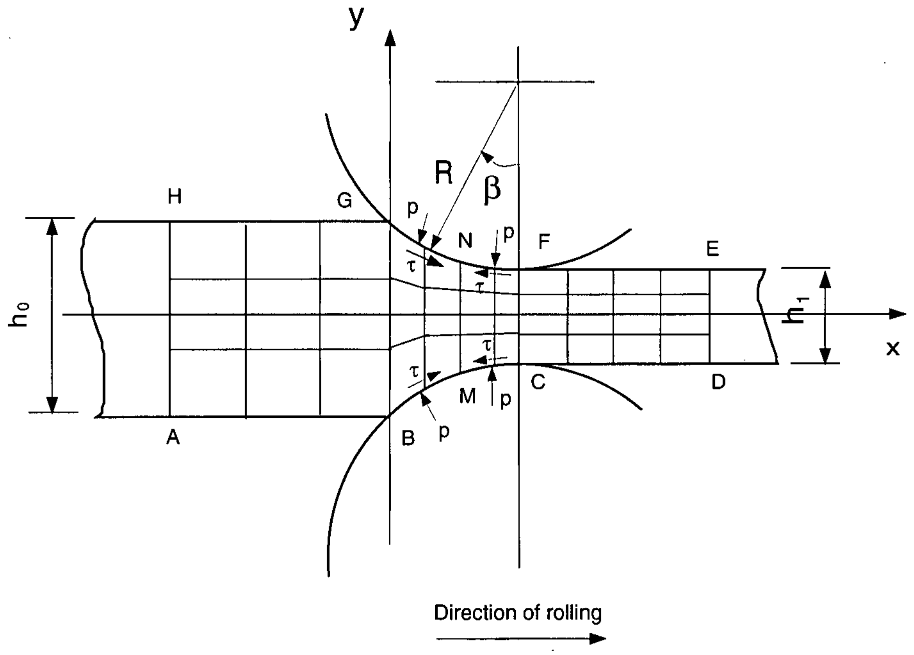


Figure 5.29: The definition of geometry in rolling

Table 5.2: Geometric dimensions of rolling process, (Length unit: mm)

Case No.	Initial Height h_0	Final Height h_1	R/h_0	Reduction
1	6.274	5.385	12.5	14.17 %
2	6.274	4.902	12.5	21.86 %
3	6.274	4.445	12.5	29.40 %
4	6.274	4.115	12.5	34.41 %
5	2.057	1.588	39.0	22.83 %

5.8.2 Deformation Conditions in Rolling

A series of cases of strip rolling of aluminium with different dimensions have been numerically analyzed with the developed program ALEFE. For comparison, the material properties and geometry are chosen to be the same as in reference [76]. The material is elasto-plastic with Young's modulus $E = 6.894 \times 10^4\ MPa$, Poisson's ratio $\nu = 0.30$ and initial yield stress $\sigma_{yield0} = 50.3\ MPa$. The material has isotropic strain hardening with the following relation:

$$\sigma_{yield} = \sigma_{yield0}\left(1 + \frac{\varepsilon^p_{eq}}{0.05}\right)^{0.26} \tag{5.1}$$

where, σ_{yield} is current yield stress,

ε^p_{eq} is equivalent plastic strain.

In the simulation, a piece-wise linear model for the plastic hardening modulus H is used to approximate Equation (5.1). The geometric dimensions of simulated rolling process are shown in Table 5.2. Two values of the ratio of the roll diameter to the initial height of work-piece, Case 1 and Case 5, are employed to show the different types of the normal stress distribution on the contact surface. A series of reduction under same ratio R/h_0 are applied to examine the change of rolling force and torque with reductions.

The friction condition on the interface is determined by experiment [76]. It can be described by Coulomb law:

$$\tau = \mu p \tag{5.2}$$

where, τ is friction stress; p is normal pressure; and μ is equivalent friction coefficient, ($\mu = 0.1$ for the studied cases).

The roller is assumed rigid and the elastic deformation of work-piece before getting into and after leaving the roll gap is ignored. The deformation is simulated under plane-strain condition. Because of symmetry, only one-half of the model is analyzed. The finite element model for case-1 is shown in Figure 5.30. The model consists of 480 four-node quadrilateral isoparametric elements.

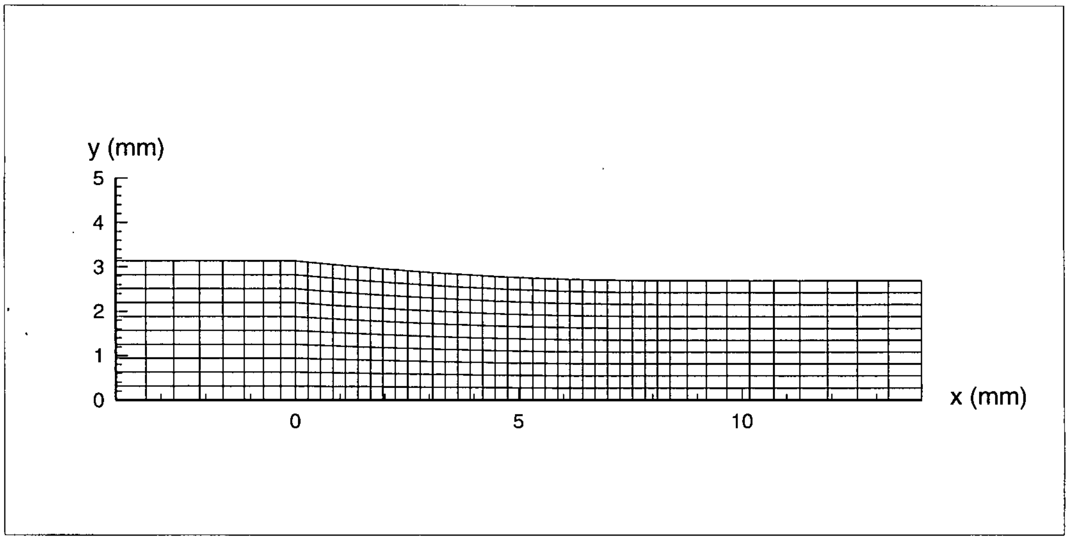


Figure 5.30: Finite element model (Case 1: $R/h_0 = 12.5$, Reduction 14.17%)

5.8.3 Numerical Treatments in the Simulation

The steady state strip rolling process is simulated by Eulerian formulation Equation (2.26) which is degenerated from the general ALE formulation Equation (2.20). Referring to Figure 5.29, it is assumed that there is a spatially fixed boundary, ABCDEFGH, and the material crosses this boundary and goes out of the rolls. New material being continuously convected into the fixed region has prescribed initial state, usually no deformation and stresses. The simulation is carried on until the steady state is achieved, in which all the variables, i.e., velocities, stresses and nodal forces etc., do not change as the new material enters the rollers.

The Coulomb friction is calculated from the normal pressure distribution in the previous incremental step. At the first incremental step, the friction stress is assumed to be homogeneous on the whole contact surface and given by:

$$\tau = 0.02k_0 \tag{5.3}$$

where, k_0 is the initial shear yield strength, $k_0 = \sigma_{yield0}/\sqrt{3}$.
Such treatment should not impact the final results, since the simulation does not finish until differences of all variables between two consecutive iterations, including normal pressure, are less than the preset tolerances. This treatment, however, makes the handling of the friction effect easier and more straight forward. Specification of boundary condition on contact surfaces is now discussed. All the nodes on the contact surface are sliding except the neutral points which may spread over a small area. A tangential stress equal to "μp" is applied to sliding nodes. The direction is dependent on whether the node is in forward sliding area or backward sliding area. The velocity boundary condition in the neutral area is that the neutral point has a circumferential speed equal to that of the roller, i.e., there is no relative slip between the roller and the neutral point. However, a prescribed velocity boundary condition may cause significant reaction force in tangential direction which is not rational. An alternative method is to specifiy a prescribed horizontal velocity at the boundary HA, as shown in Figure 5.29. The velocity is calculated by:

$${}^{t}v_{h_0} = \frac{h_n v_{rh}}{h_0} \tag{5.4}$$

where, h_n is the height of work-piece at neutral point,

v_{rh} is the horizontal component of roll speed.

Equation (5.4) assumes homogeneous ${}^{t}v_x$ distribution on the cross section passing through the neutral point NM and a volume consistency between NM and HA. This assumption, is trivial and should not impact the simulation results. Another requirement to be considered here is the total reaction force caused on edge HA. This force should be much smaller than the reaction forces elsewhere if no other external traction is applied in the rolling. In simulation, the above two ways of applying the velocity boundary condition are employed and compared.

5.8.4 Simulation of Neutral Point Location

The existence of a neutral point is the unique feature of rolling. Its location directly affects the results of simulation, so it is important to correctly detect or simulate it. However, finding the neutral point is not easy. The neutral point has the same speed as the roller, so it is natural to think that the neutral point should be the point on the contact surface that has a velocity equal or close to the tangential roller velocity. To find such a point, an additional iteration loop needs to be nested outside the equilibrium iteration in each incremental step. At the beginning of a new step, we apply the boundary conditions according to the results from the last converged step, then find the new location of the neutral point from the calculated nodal velocities. If the difference between the new and old neutral points is not within the specified tolerance, the boundary conditions have to be updated according to the new neutral point. The process continues until convergence occurs. Results of the neutral point simulation by the above procedure are shown in Figure 5.31. The figure shows the x-location of the neutral point as a function of the incremental step. It is observed that the neutral points for all five-reduction cases did not change from the initial guess throughout the deformation process and that different initial guess lead to different results. Two different initial guesses are made for Case 1 and shown in Figure 5.31, labelled as Case 1a and Case 1b.

The distribution of rolling pressure and friction stress on the contact surface with respect to angle β in Figure 5.29 is also examined. When velocity boundary condition is applied at neutral point, both rolling pressure and friction stresses show a sudden "jump". The results from Case

1 are shown in Figure 5.32 (curve ALE-NEU). The "jump" occurs at the neutral point. It is believed that this is due to the reaction resulting from the application of the boundary condition. On the other hand, if the velocity boundary condition is applied on the external boundary HA, the neutral point fluctuates and may not in general converge. Figure 5.32 (curve ALE-BOU), shows the distribution of rolling pressure and friction stress for Case 1 from such simulation. We see that the "jumps" are eliminated, but the distribution patterns are quite different. Another important point to be considered here is that the reaction force on HA is quite high (108 N/mm) and may not, in general, be ignored if compared to the total rolling force (845 N/mm). In addition to these points, we see that the above simulations of neutral points may not maintain equilibrium of external forces.

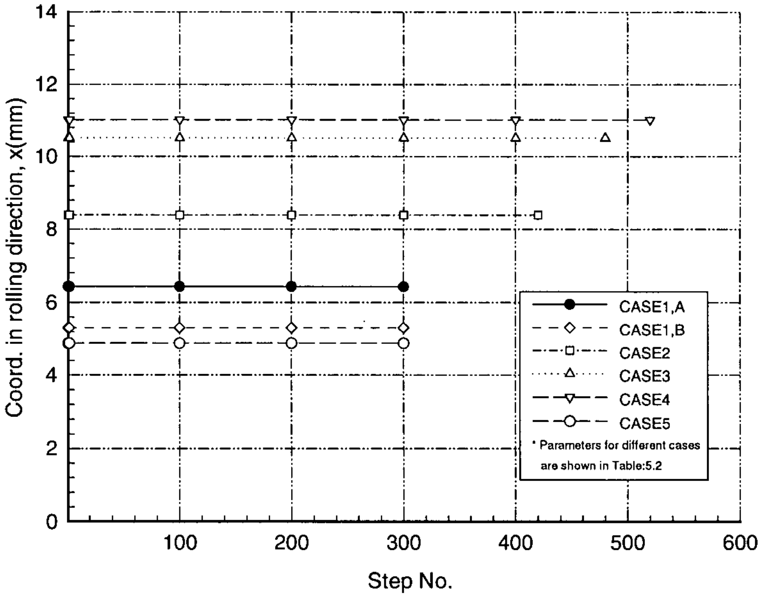
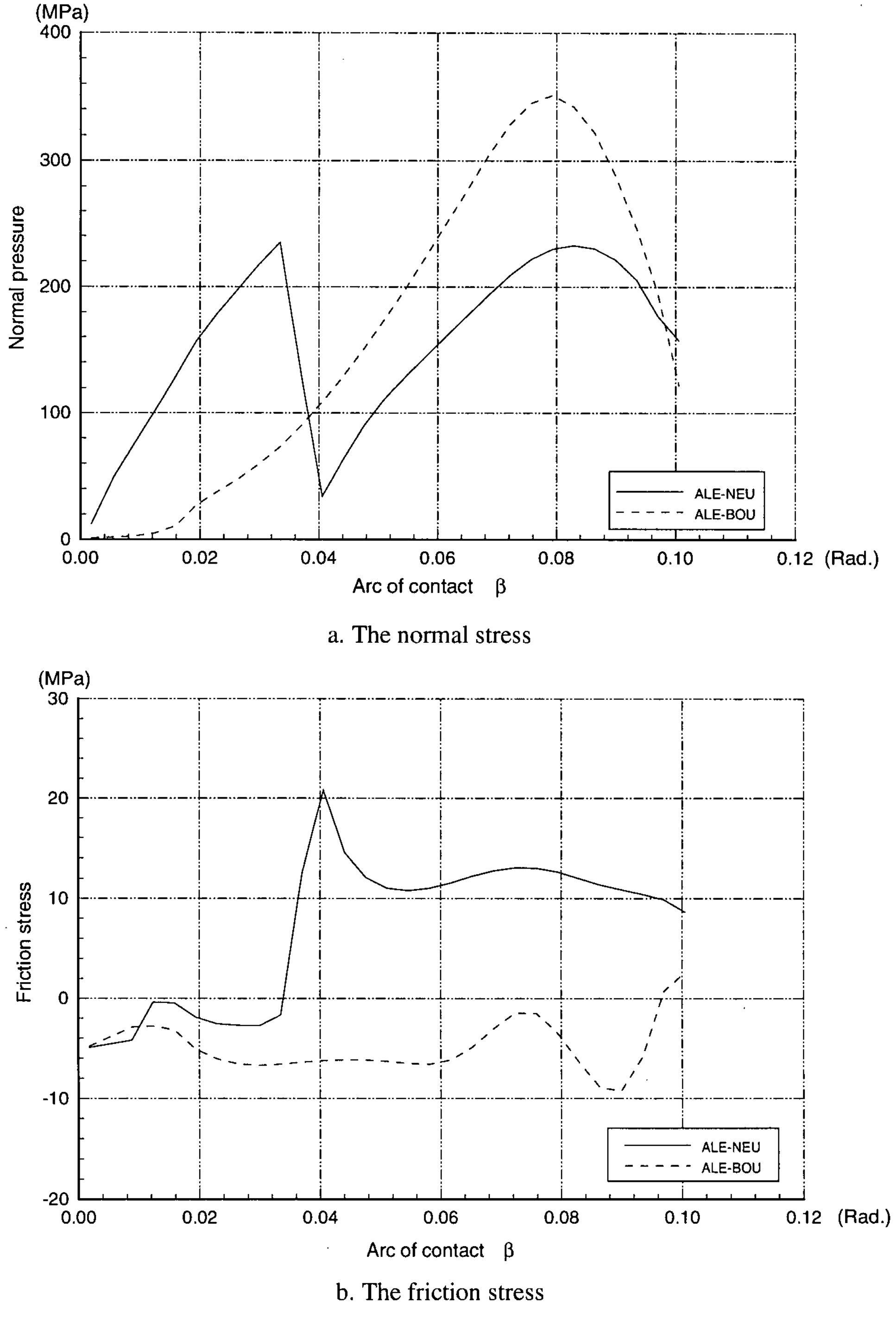


Figure 5.31: Locations of neutral points when velocity criterion is applied

The correct and final location of the neutral point should keep the external forces self-balanced. That is a unique feature of the rolling process and generally makes the simulation of



a. The normal stress

b. The friction stress

Figure 5.32: Rolling pressure and friction stress when velocity criterion is applied

rolling more complicated. The above methods of deciding neutral points introduced resultant forces on either the free edge HA or at the neutral point itself. These forces are not present in the real system and so the methods are unacceptable. A more rational definition of the neutral point is thought to be given by the following force balance equation in the rolling direction [77]:

$$\int_{\beta_N}^{\beta_G} (\tau R cos\beta - pR sin\beta)\, d\beta - \int_0^{\beta_N} (\tau R cos\beta + pR sin\beta)\, d\beta = 0 \tag{5.5}$$

where, β is the angle from the vertical line to the line connecting the contact point and the roll center as shown in Figure 5.29, β_N is the value of β at the neutral point N, and β_G is the value of β at point G.

In this proposed simulation of the neutral point, the velocity boundary condition may be either applied at the neutral point itself or on the external boundary HA as discussed above. Figure 5.33 shows how the location of the neutral point moves from the initial guess to the final steady state location when this simulation is used. It is observed that the neutral points moved from the initial guesses to steady state locations for all reduction cases, and the difference in the initial guess has no impact on the final result. To show this, two different initial guesses are considered for Case 1 and the results are shown in Figure 5.33 as Case 1a and Case 1b. It is observed that the final neutral point has the same position after the 2nd step. Applying velocity boundary condition on the stress-free boundary HA gives the same results for neutral point in all 5 cases. Figure 5.34 shows the impact on rolling pressure when applying the velocity boundary conditions on the stress-free boundary HA (curve ALE-BOU) or on the neutral point itself (curve ALE-NEU). It is obvious that the two methods give almost identical results and the "jumps" caused by the previous methods are eliminated. It is also important to investigate the reaction forces in this new simulation. The reaction force at HA, when the velocity boundary condition is applied at the same boundary, is 2.31 N/mm, which is quite small and may be ignored if compared to the rolling force of 875.11 N/mm. The above results suggest that this method is more reliable and more accurate than the ones previously discussed.

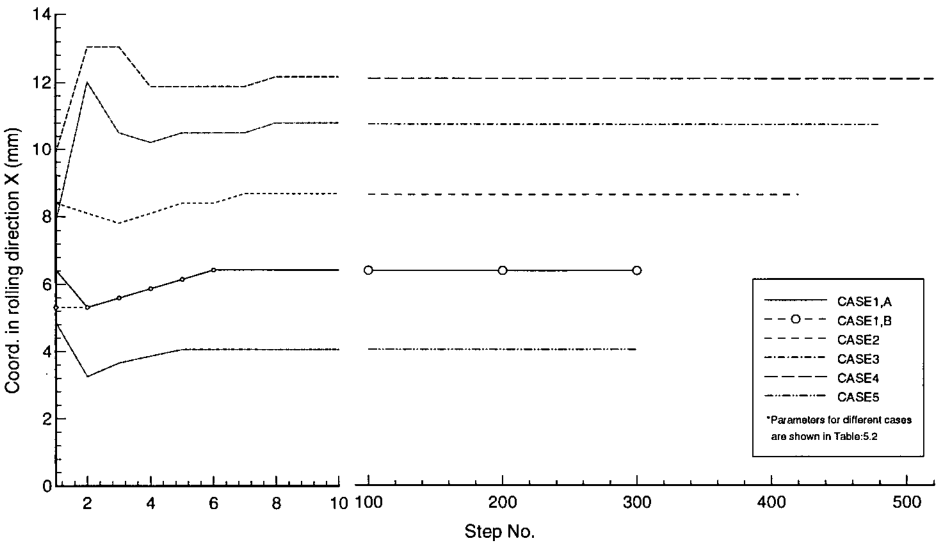
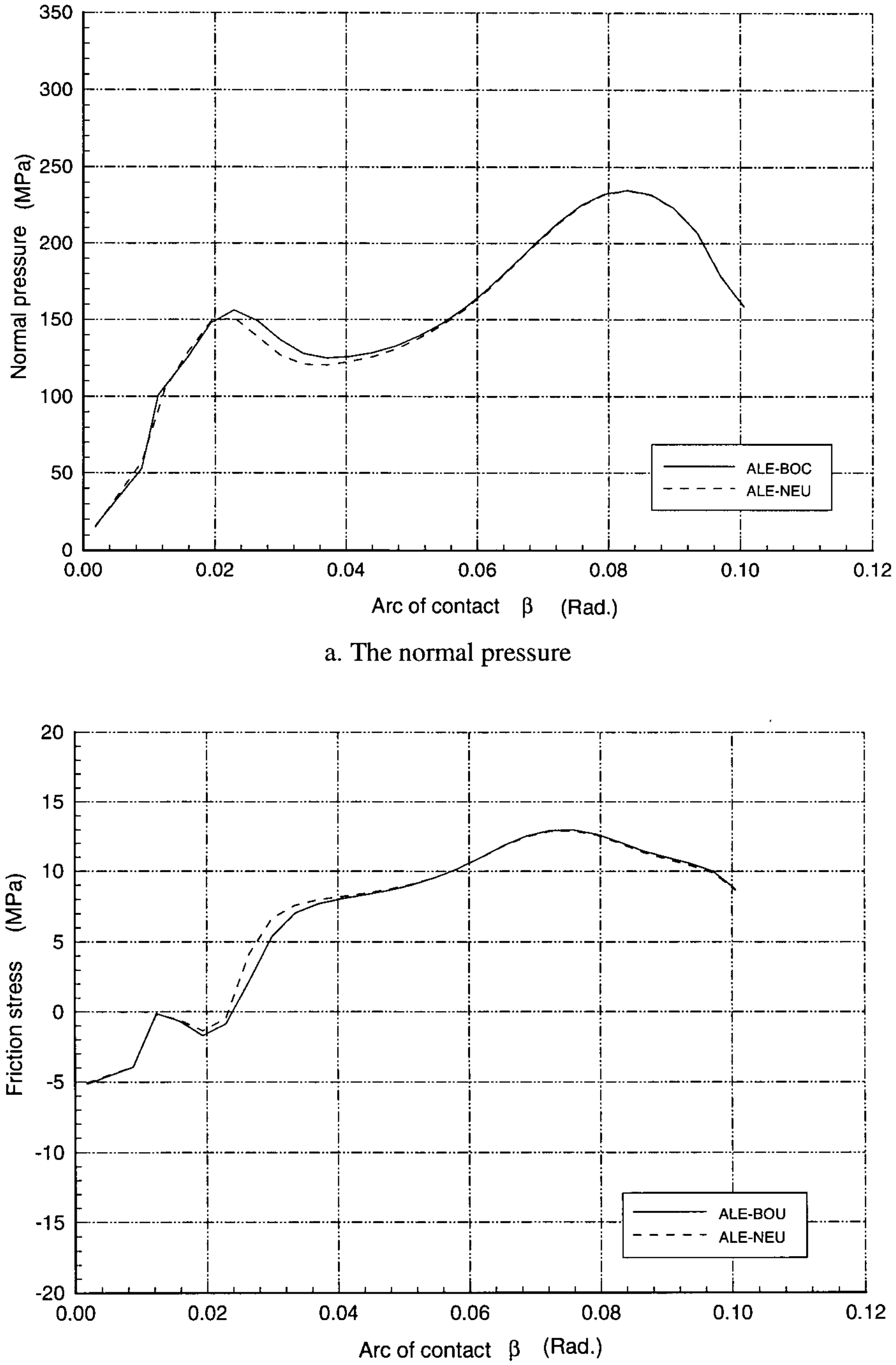


Figure 5.33: Locations of neutral points when force condition is applied

5.8.5 Simulation of Rolling Pressures

As far as the rolling pressure distribution patterns is concerned, the classical slab method gives a distribution with maximum pressure, known as "friction hill", at the neutral point, on both sides of which the pressure decreases monotonically, as shown in Figure 5.36. However, some circumstances were observed experimentally [76] in which the pressure distribution possess double peaks. In the present computed results, double-peak pressure distribution curves, as well as "friction hill" distribution are obtained depending on the parameters involved. Figure 5.35 and Figure 5.36 show the pressure variations for two typical cases, Case 1 and Case 5, respectively. For small roll ratios, R/h_0, two-peak roll pressures are obtained with maximums near entrance and exit and with pressure drop in the middle of contact arc. For cases of large R/h_0, Figure 5.36 shows a friction-hill distribution. It may be observed that the computed results are, generally,



a. The normal pressure

b. The friction stress

Figure 5.34: Rolling pressure and friction stress when force criterion is applied

in good qualitative agreement with the experimental results. The discrepancies at the entrance and exit points may be attributed to inaccurate description of friction, the roll flattening and the spring back of work-piece after exit. The experimental results show significant increase in contact angle at both entrance and exit, while the contact angle in the numerical simulation is smaller, obtained only from pure geometric relations and has the same value as in a classical slab method.

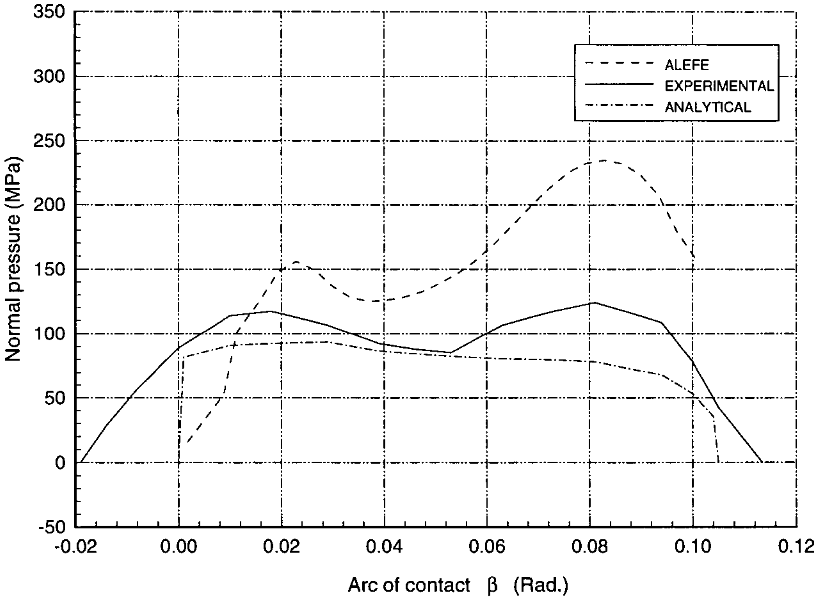


Figure 5.35: Comparison of rolling pressure, small R/h_0

5.8.6 Simulation of Rolling Force and Torque

The roll separating force and the roll torque under different deformation reduction but same R/h_0 values form Case 1 to Case 4, are simulated by applying the velocity boundary condition at neutral point and entrance. Figure 5.37 and Figure 5.38 show a comparison of the numerically obtained results and experimental ones from reference [76]. Once again, the

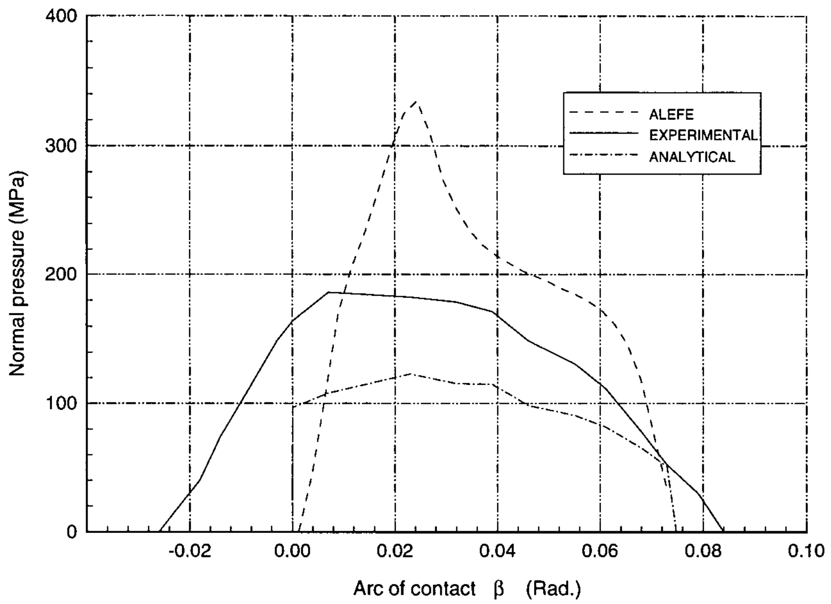


Figure 5.36: Comparison of rolling pressure, large R/h_0

two boundary conditions (ALE-BOU and ALE-NEU) gave almost identical results as in the roll pressure distribution. The simulated results for rolling force are smaller than the measured values for some reductions and larger for others whereas the simulated torque values generally overestimate the experimental values. The difference between the computed and measured values may be attributed to the assumption of rigid roll in the finite element analysis, to the uncertainty in friction modeling and measurements and to the strain hardening modeling and measurements.

Figure 5.39 is the distribution of equivalent plastic strain for Case 1. It shows the spread of plastic zones within the strip. Prior to entry, a large plastic zone is observed and yielding is indicated to take place before the point of contact. At the center, yielding occurs very near to the point of contact. Continued elasto-plastic flow is observed after yielding. Figure 5.40 shows contours of current yield stress for Case 5 at different steps. It may be observed that the maximum values are moving horizontally from entrance to the exit. The figure also shows how

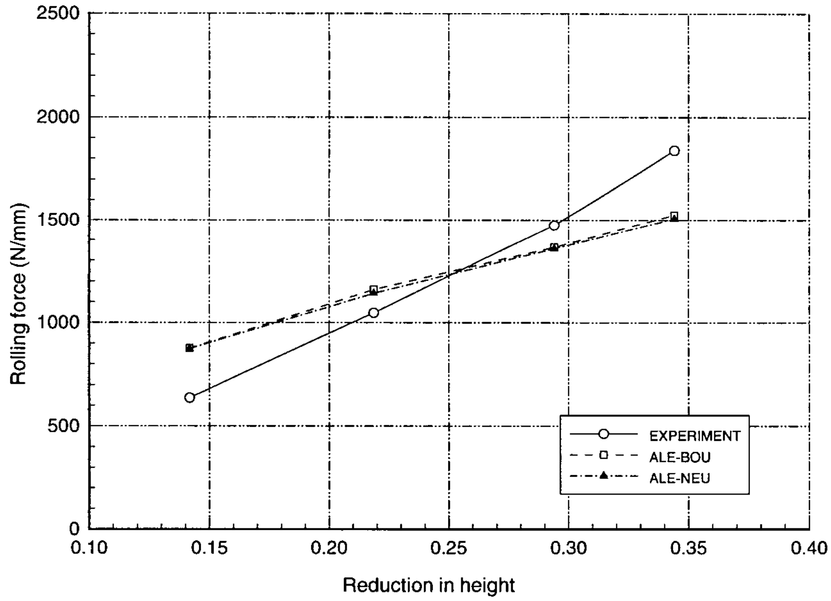


Figure 5.37: Comparison of rolling force

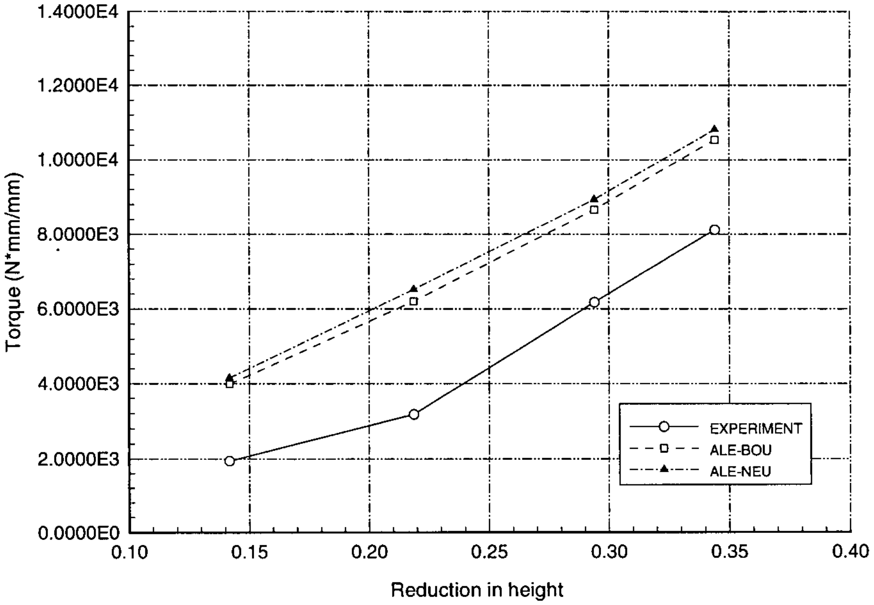


Figure 5.38: Comparison of rolling torque

the plastic zone is developed. In the simulation, a steady state condition was reached after 300 incremental steps.

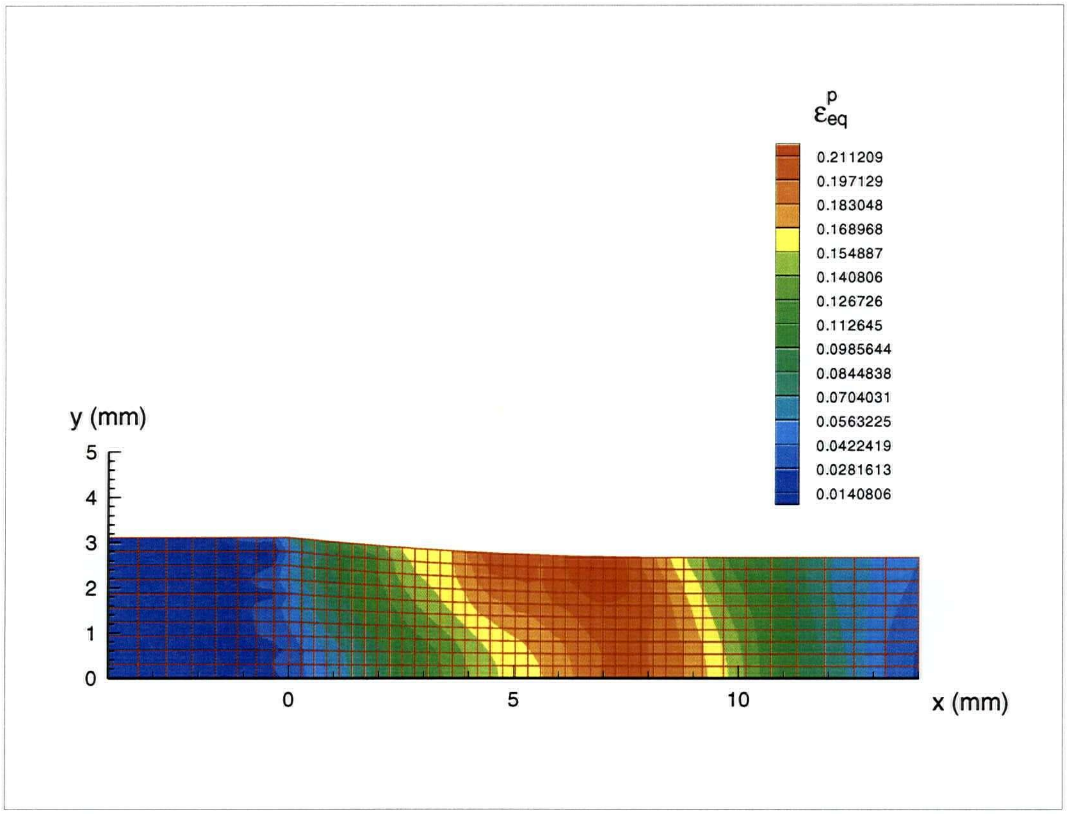
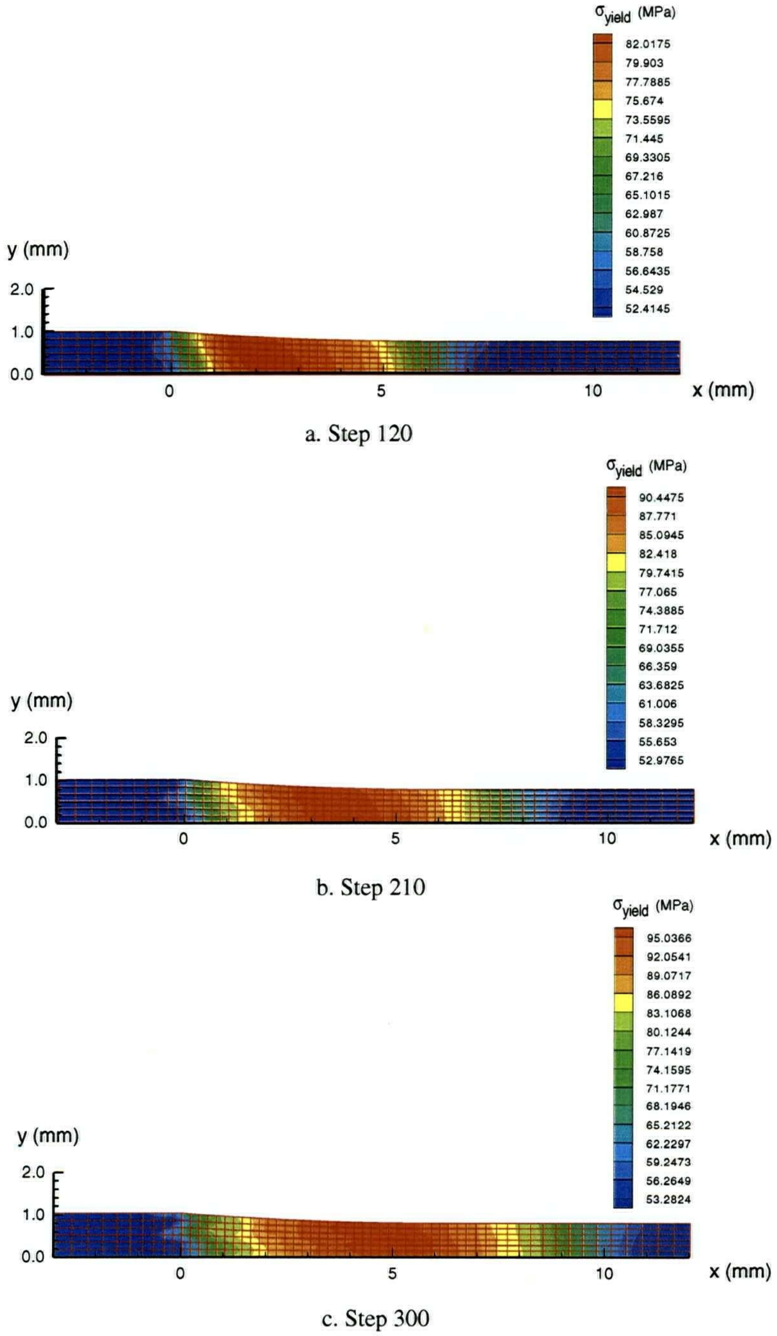


Figure 5.39: Equivalent plastic strain (Case 1: $R/h_0 = 12.5$, $Reduction = 14.17\%$)

5.8.7 Concluding Remarks

One of the most complicated deformation processes in metal forming: the rolling process, is successfully simulated by the Eulerian method reduced from the general ALE formulation developed in this research. Program ALEFE is directly applied to simulate the process without special developments or enhancements. Various methods for applying correct boundary conditions and locating neutral points are investigated and proved to be helpful in simulating such kind of problems efficiently and accurately. The method used in handling friction forces in this



a. Step 120

b. Step 210

c. Step 300

Figure 5.40: Current yield stress at different steps (Case 5: $R/h_0 = 39.0$, $Reduction = 22.83\%$)

work is simple and proved to be effective in conjunction with the Eulerian simulation. The computed results showed both double-peak as well as "friction hill" distributions for rolling pressure. In general, the results are in good agreement with experimental ones.

5.9 COMPRESSION BETWEEN WEDGE-SHAPED DIES

5.9.1 Deformation Conditions

Plane strain compression of pre-shaped material between wedge-shaped dies is simulated by ALE method. The same process is investigated experimentally and analysed with slip line method by Chitkara and Johnson [78]. The dies are considered to be perfectly rough, so sticking friction conditions are applied in the simulation. Another distinct characteristic of the process is that the dies are wedge shaped which brings out two consequences. A more general coupled displacement / velocity boundary condition on the die-metal interfaces is needed because the dies themselves move in the process and the deformation is no longer symmetric about the vertical axis.

The general shape and size of specimen and die are indicated in Figure 5.41. In order to get the curves of mean vertical die pressure and end velocities with respect to strip thickness, different specimen dimensions are employed as shown in Table 5.3. The unique tool angle $\eta = 3°$ is used because most of the comprehensive experimental results are available in literature for this angle [78]. The specimen ABCDEFGP is compressed between two inclined dies BC and FG with angle of inclination η with the horizontal direction. The specimen is pre-shaped such that surface BC and FG are to be matched with the tool face. The dotted lines indicate the approximate shape after deformation. Velocities v_l and v_s are at large and small end of the specimen, respectively, while each tool is advancing with a velocity v.

Experiments were conducted on 300-ton Denson hydraulic testing machine and loads of less than 3.45 MPa (500 lb/in^2) were measured independently by means of two calibrated

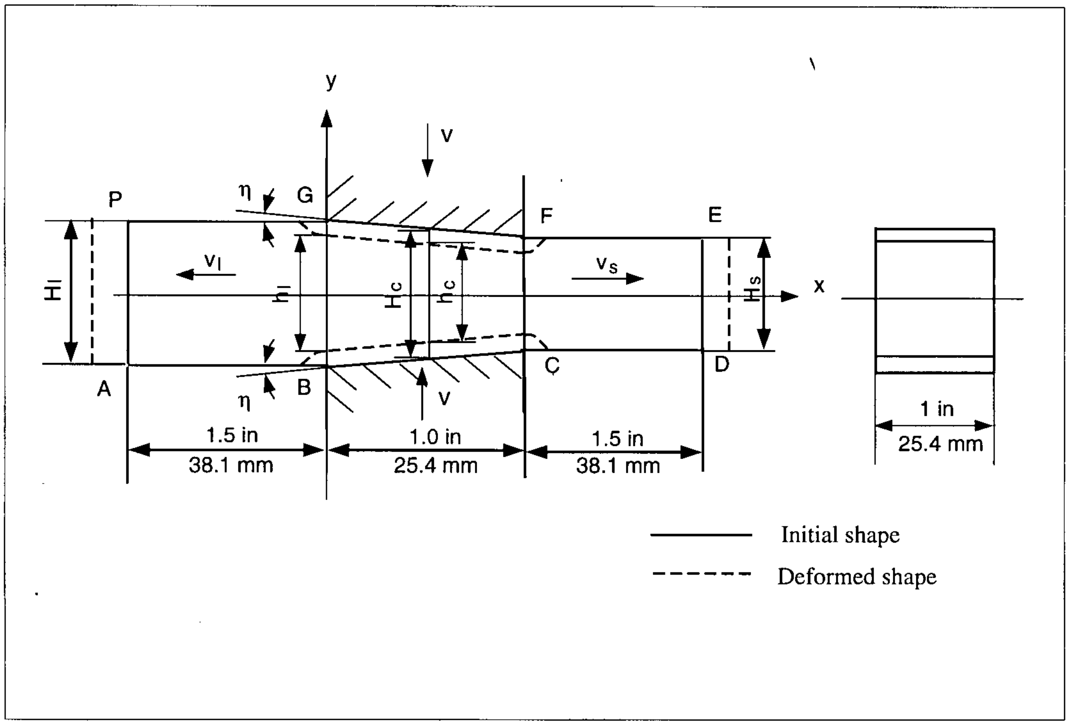


Figure 5.41: General shape of the specimen and die

Table 5.3: Specimen dimensions in compression between wedge-shaped dies

No	η	H_l		H_0	
	degree	mm	in	mm	in
1	3°	25.40	1.000	22.74	0.895
2	3°	22.23	0.875	19.56	0.770
3	3°	19.05	0.750	16.39	0.645
4	3°	15.88	0.625	13.21	0.520
5	3°	12.70	0.500	10.04	0.395
6	3°	9.53	0.375	6.86	0.270
7	3°	6.35	0.250	3.69	0.145
8	3°	4.45	0.175	1.78	0.070

pressure gauges, so that the die load may be recorded. Dial gauges were used for measuring downward displacement of the die and lateral movements of the specimen. Therefore, the deformed specimen thickness and velocities at two ends can be measured. The rough dies were prepared by machining a square grid on the surface and using no lubrication in compression [78]. The material of specimen chosen in ALE simulation is half hard aluminium. The stress-strain curve is obtained by experiment [78]. Using bilinear approximation of the constitutive curve, we use initial yield stress $\sigma_{yield0} = 276\ MPa$ and hardening modulus $H = 25\ MPa$. Other material properties used in the simulation are: Young's modulus $E = 69\ GPa$ and Poisson's ratio $\nu = 0.33$ [72].

5.9.2 Finite Element Model

Due to symmetry about the horizontal axis only upper half of the specimen is discretized. The finite element model for Case 1 is shown in Figure 5.42. A total of 240, 4-node quadrilateral elements are used in the model. The material nodes on horizontal axis are restricted to move vertically, however they are free to move horizontally. The material points on die-metal interface FG are restrained to move horizontally to achieve the sticking friction condition, but they are allowed to move in vertical direction along with the die movement.

In ALE simulation, the mesh nodes on edge FG are treated as Lagrangian points in both horizontal and vertical direction. The mesh nodes on GB and FC are taken as Eulerian points, i.e., fixed in horizontal direction, and general ALE points in vertical direction where the velocity changes linearly from die velocity v at F and G to zero at C and B. The mesh points on AB and CD are kept as Lagrangian points in vertical direction but as general ALE point in horizontal direction where velocity increases linearly from zero at B and C to the material velocities of points A and D at A and D, respectively. The mesh points on BC are Lagrangian points in vertical direction and Eulerian points in horizontal direction. All other boundary nodes

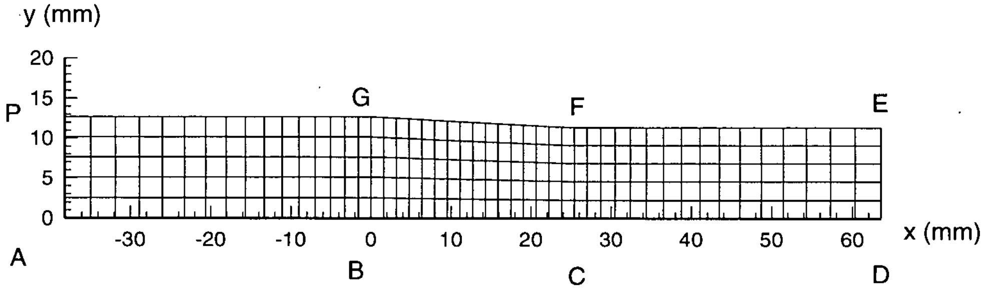


Figure 5.42: The finite element model for Case 1

are on traction-free boundaries, and, therefore, they are Lagrangian points in both horizontal and vertical directions. The motion of internal nodes is specified by the mesh motion scheme discussion in Chapter 3. The compression is performed for up to 50% reduction in initial height in 500 steps.

5.9.3 Slip Line Field Solution

The slip line field solution [78] assumes that centred fans radiate from corners of the dies and that the starting slip lines are at 45° to the plane of symmetry of the specimen. The usual technique of approximating small arcs by their chords is applied to build up the slip-line field. The total normal and net tangential forces on the dies are evaluated. Then, vertical die pressures are obtained by resolving these forces and summing algebraically [78]. It should also be noted that the average yield stress is used in slip line analysis. The velocities of the specimen at the large and small end are achieved from the hodograph constructed according to the slip line field.

5.9.4 Comparison of Results

The results predicted by ALE method are compared with the experimental and slip line field analysis results published by Chitkara and Johnson [78]. Figure 5.43 shows the result of Case 1 giving the variation of load on the die with respect to compressive strain $ln\ H_c/h_c$. The load-strain plot of ALE result matches closely the experimental results when strain is less than 0.30. An obvious discrepancy starts to appear after this level and increases as the strain increases. The discrepancies at large reduction levels may be due to a reduction in friction at the interfaces as the compression proceeds in experiment, which was reported in [78]. Besides, the experiments used a finite width 25.4 mm (1 in) of the specimen, compared with 25.4 mm (1 in) of projected length of die. Therefore, the plane strain conditions might not be achieved experimentally.

Non-dimensionalized mean vertical pressures $p_v/2k$ as a function of the current strip thickness of the specimen at large side h_l is presented in Figure 5.44. The result from ALE matches the experimental values much better than the slip line results. Figure 5.45 shows results of end velocities of specimen at different thickness h_l. The values from ALE and slip line theory agree with the experimental results very well.

5.9.5 Development of Deformed Shapes and Plastic Zones

Figure 5.46 shows the deformed shapes and contours of equivalent plastic strains at steps when reduction in height is 15%, 30% and 45% for Case 1. It is observed that even when reduction is 45%, the mesh is still homogeneous everywhere in the deformation domain. The punch or die size increasement which occurs in updated Lagrangian method does not appear here. From the equivalent plastic strain distributions at different steps, it is may be seen that

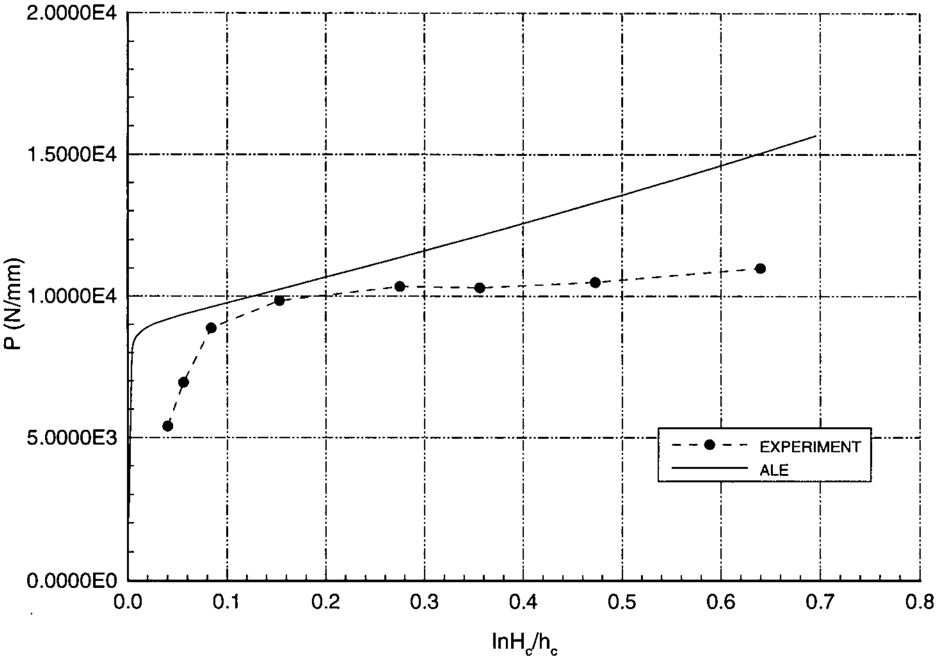


Figure 5.43: The variation of vertical load on the die with compressive strain

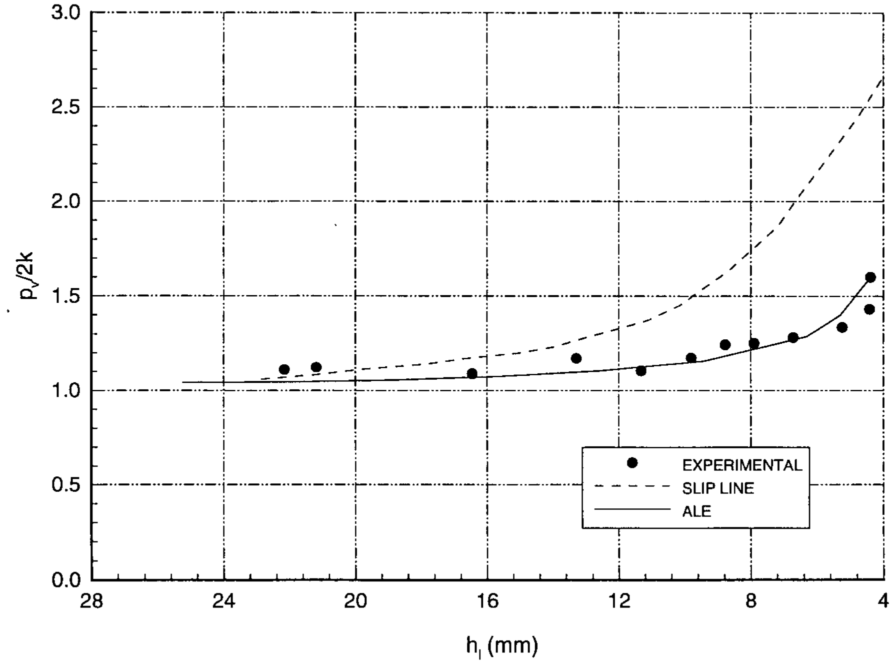
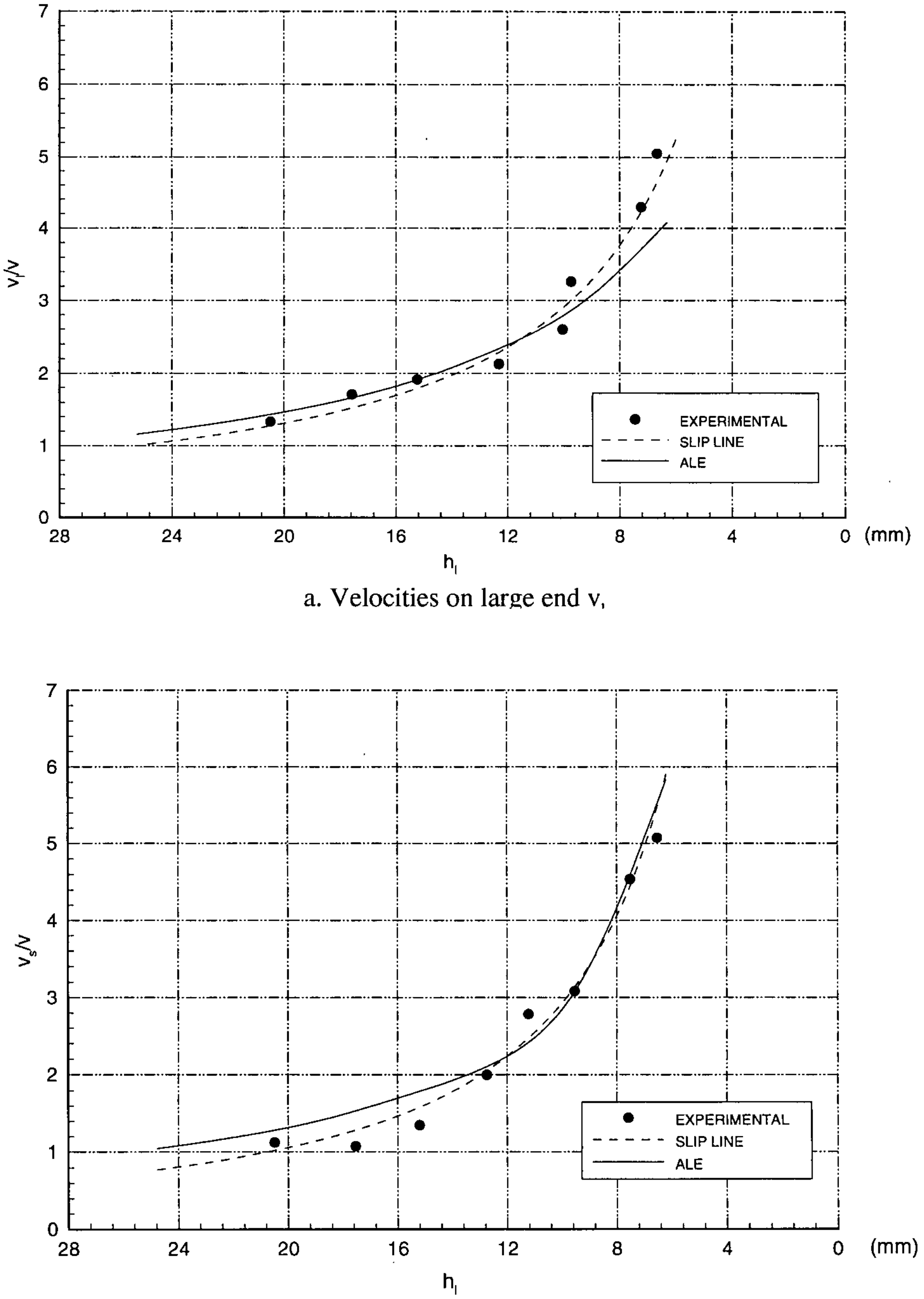


Figure 5.44: The variation of mean vertical die pressure with current strip thickness

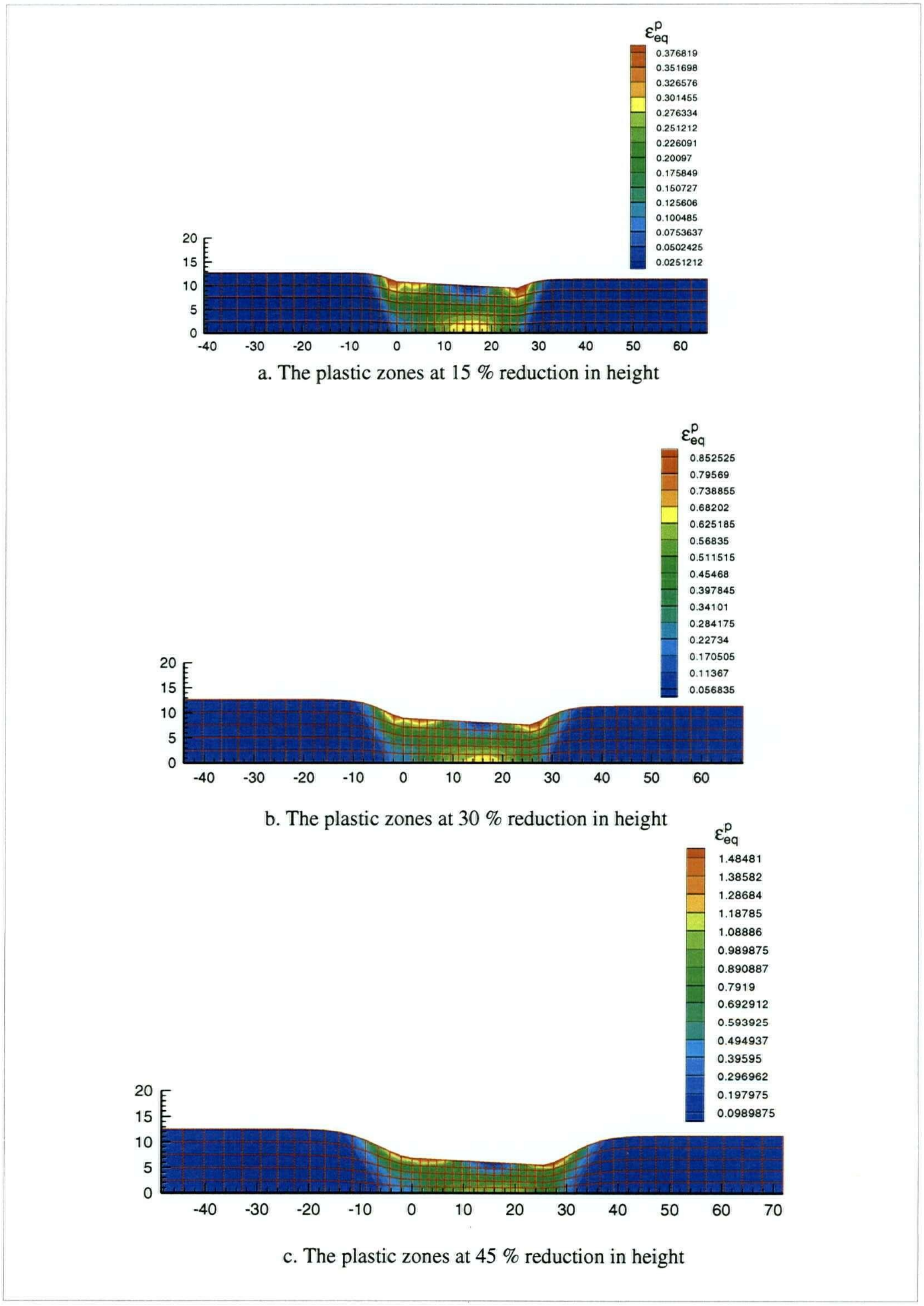


a. Velocities on large end v_l

b. Velocity on small end v_s

Figure 5.45: Distribution of end velocities

the plastic zones are initially formed at the die corners and the strain values are highest through the whole deformation. The central parts of the interfaces have lower values of plastic strain, which means that the deformation is small and is incurred by the non-slip friction condition. This may be the reason why in slip line fields, a rigid body block is usually assumed there. It is interesting to indicate that plastic zones are already formed outside of the dies, but in slip line field constructed by Chitkara and Johnson [78], the plastic deformation begins only inside the dies, which may contribute to the inaccuracy of slip line results. Plastic zones are mainly located between and around the two dies. For most area outside the dies, plastic deformation is almost zero.



a. The plastic zones at 15 % reduction in height

b. The plastic zones at 30 % reduction in height

c. The plastic zones at 45 % reduction in height

Figure 5.46: Equivalent plastic strain at different steps

Chapter 6

CONCLUSIONS AND FUTURE WORK

6.1 CONCLUSIONS

It is shown that when the conventional Lagrangian methods are applied to simulate finite strain deformation processes, many problems arise: extensive mesh distortion or entanglement, load fluctuations and incorrect description of contact boundary conditions with sharp edges or corners. Available remeshing and rezoning techniques may overcome some mesh distortion and boundary description problems but they are generally not robust and they create additional problems in convergence and consume extensive CPU time. In this research, an ALE formulation is shown to have the potential of eliminating the above problems and of providing a promising tool by combining the merits of both Lagrangian and Eulerian schemes. A critical discussion of ALE formulation in solid mechanics is given in this work. It is shown that many authors have concentrated on particular problems that led to a special form or a restricted formulation. A general ALE formulation is proposed in this thesis and it is developed from the principal of virtual work after discretization for a linear increment. The presented formulation is applicable to various material constitutive models and to general loading.

A mesh motion scheme is developed based on the transfinite mapping method and is extended to consider the effect of general boundary motion. The scheme is simple and efficient one that does not cause curve fitting errors. It allows the boundary curves to be represented in discrete form and to have discontinuous slopes. Therefore, it is directly applicable to domains with irregular polygons as boundaries. This scheme is shown to be computationally more efficient and is straight forward to implement in an ALE formulation. In the thesis, motion of nodes on boundaries, the classification of boundary types and the corresponding detailed motion schemes are presented and shown to be general and efficient in obtaining a high quality mesh. The effect of distorted mesh on the accuracy of the results and the necessity for an efficient ALE mesh motion scheme are demonstrated through numerical examples. It is shown that the developed mesh motion scheme maintains a homogeneous mesh under very large deformation levels.

The ${}^tD_{ij}$ components calculated from the conventional forward difference algorithm may not vanish under rigid body rotation when the time increment is finite. This may lead to excessive error accumulation in practice. A modified forward difference algorithm or a central difference one provides an alternative for calculating the components of ${}^tD_{ij}$ in large deformation and large rotation problems. This ensures the objectivity of the ${}^t\mathbf{D}$ tensor with respect to rigid body rotations and maintains objectivity in explicit stress integration algorithm when applied separately. The modification term for the components of the ${}^t\mathbf{D}$ tensor is presented in this thesis and is also shown to maintain objectivity of the tensor with respect to rigid body motions.

An alternative stress integration algorithm developed in this thesis is shown to be equivalent to an integration of Truesdell rate equation. The development is based on physical meaning of 2nd Piola-Kirchhoff stress and finite deformation. The difference between the developed algorithm and the one based on purely mathematical transformation is only in the higher order terms of time increment Δt, and it is demonstrated that both algorithms produce identical numerical results. Two numerical implementation schemes for stress integration coupled with a return mapping method are presented to keep incremental plastic consistency. The proposed schemes may generally be combined with any integration algorithm. It is shown that the integration of Jaumann rate is less sensitive to the location of stress transformation than that of Truesdell rate. The combination of Truesdell rate with transformation A presented in this work generally gives better results than other cases.

In the updating of material associated properties, the relation between material derivative and referential time derivative is applied to create the relation between increment due to mesh motion and increment due to material motion. Using the developed relation, the material associated properties may be updated directly from material points to mesh points without the need for time consuming interpolations and the creation of a hypothetical mesh as was done by other authors. Combined with local smoothing by the least square method, the updating scheme provides an efficient and accurate routine to update material associated properties.

Practical application of the developed ALE formulation is presented in the metal forming area. Simulation of metal extrusion and punch forging processes is performed. It is shown that the load fluctuation and punch size increasement, normally associated with updated Lagrangian formulation, are eliminated. ALE results indicate that the mesh motion algorithm maintains homogeneous mesh throughout the deformation process. It is also shown that description of changing contact boundary conditions is straight forward and easy to perform in ALE formulation. A necking bifurcation of a bar in tension is simulated by an updated Lagrangian method that is reduced from the ALE formulation. Through this example, it is shown that the combination of Truesdell rate with transformation after return mapping gives consistently better results than use of Jaumann rate.

One of the most important and complicated metal forming processes is rolling. The steady state condition of rolling of sheet metals is simulated with the Eulerian option, i.e. the particular case of ALE method when the mesh does not move. Application of correct boundary conditions and simulation of neutral points are investigated. The simulation is performed with the general 2D finite element program ALEFE developed in this research without the need to append special routines for the application. No particular interface elements or friction layers are needed. For different ratio of initial thickness of work piece and roll radius cases, the program predicts both double and single peaks of friction-hill of pressure distribution, as indicated by experiments. Rolling pressure, force and torques are in reasonable agreement with the experimental results.

Compression between wedge-shaped dies is also simulated by the ALE method. It is noted that although the deformation process is not symmetrical, the ALE method still maintains a homogeneous mesh during deformation and eliminates boundary condition problems caused by updated Lagrangian method. Simulated results are in good agreement with experimental ones.

6.2 FUTURE WORK

The ALE formulation presented in this research may be extended to simulate dynamic problems. Explicit time integration schemes should be investigated and applied in this area to achieve higher calculation efficiency. Some effort should be concentrated on mesh motion schemes with dynamic effects. The application of ALE in fracture mechanics is also a promising area since it may facilitate the simulation of crack propagation in a continuous manner. However, some details, e.g., tracing material particles and updating material-associated properties at crack tip, are not well developed.

The combination of transfinite method with other methods, such as isoparametric interpolation method, is another area worth to explore. Such combination makes it possible to take the advantage of both techniques. Therefore, meshes with complicated topologies or excessively distorted meshes may be moved easily to obtain better quality . Other type of elements, like triangular, 8-node quadrilateral element should be incorporated in the program ALEFE, so that user can choose different element types according to the feature of a specific problem. In addition, curved boundaries may be described more accurately.

In metal forming, deformation often occurs at elevated temperature, so that the deformation rate may have significant effect. Large deformation normally produces heat and temperature rise. Therefore, rate dependent material model should be applied in the simulation. Furthermore, thermal effects should be coupled with the deformation analysis to achieve higher accuracy.

Bibliography

[1] Hill, R., "Some Basic Principles in the Mechanics of Solids without a Natural Time", J. Mech. Phy. Solids, Vol.7, pp. 209-225, 1959.

[2] Hibbitt, H. D., Marcal, P. V., Rice, J. R., "A Finite Element Formulation for Problems of Large Strain and Large Displacement", Int. J. Solids Struct., Vol.6, pp. 1069-1086, 1970.

[3] McMeeking, R. M., Rice, J. R., "Finite Element Formulations for Problems of Large Elastic-plastic Deformation", Int. J. Solids Struct., Vol.11, pp. 601-616, 1975.

[4] Maniatty, A. M., Dawson, P. R., Weber, G. G., "An Eulerian Elasto-viscoplastic Formulation for Steady-state Forming Process", Int. J. Mech. Sci., Vol. 33, pp. 361-377, 1991.

[5] Gadala, M. S., Dokainish, M. A., Oravas, G. AE., "Formulation Methods of Geometric and Material Nonlinearity Problems", Int. J. Nume Meth. Engng., Vol.20, pp. 887-914, 1984.

[6] Gadala, M. S., Oravas, G. AE., Dokainish, M. A., "A Consistent Eulerian Formulation of Large Deformation Problems in Static and Dynamics", Int. J. Non-linear Mechanics, Vol.18, pp. 21-35, 1983.

[7] Abo-Elkhier, M., Oravas, G. AE., Dokainish, M. A., "A Consistent Eulerian Formulation for Large Deformation Analysis with Reference to Metal-extrusion Process", Int. J. Non-linear Mechanics, Vol.23, pp. 37-52, 1988.

[8] Scientific Forming Technologies Corporation, Design Environment for Forming Users Manual (Version 4.1), Scientific Forming Technologies Corporation, Columbus, USA, 1995.

[9] Swanson Analysis Systems Inc., Engineering Analysis System User's Manual, Swanson Analysis Systems Inc., Houston, USA, 1992.

[10] Engineering Mechanics Research Corporation, Users Manual for NISA Systems Analysis, Engineering Mechanics Research Corporation, Michigan, USA, 1992.

[11] Lee, C. H., Kobayashi, S., "Elastoplastic Analysis of Plane-strain and Axisymmetric Flat Punch Indentation by the Finite Element Method", Int. J. Mech. Sci., Vol.12, pp. 349-370, 1970.

[12] Lee, E. H., Mallett, R. L., "Stress and Deformation of Metal Extrusion Process", Comp. Mech. Appl. Mech Engrg., Vol.10, pp. 339-353, 1977.

[13] Voyiadjis, C. Z., Foroozesh, M.,"A Finite Strain Total Lagrangian Finite Element Solution for Metal Extrusion Problems", Comp. Appl. Mech. Engrg., Vol.86, pp. 337-370, 1991.

[14] Lush, A. M., "Numerical Grid Generation Used for Remeshing Finite Element Analyses of Metal Forming", In: Numerical Grid Generation in Computational Fluid Mechanics' 88 (Sengupta, S., Hauser, J., Eiseman, P. R., Thompson, J. F. eds), 2nd Conference on Grid Generation in Computational Fluid Dynamics, pp. 1029-1038, Swansea, UK, 1988.

[15] Cheng, J., Kikuchi, N., "A Mesh Re-zoning Technique for Finite Element Simulations of Metal Forming Process", Int. J. Nume. Meth. in Engrg., Vol.23, pp. 219-288, 1986.

[16] Gelten, I. C. J. M., Konter, I. A. W. A., "Application of Mesh-rezoning in the Updated Lagrangian Method to Metal Forming Analyses, In: Numerical Methods in Industrial Forming Processes (Pittman, J. F. T. eds), pp. 511-521, Pineridge Press, Swansea, UK, 1982.

[17] Wang, J., Gadala, M. S.," Remarks on the Finite Element Solution of Large Strain and Forming Problems", Proceedings of Canadian Congress of Applied Mechanics-CANCAM 95, pp. 262-263, Victoria, BC, Canada, 1995.

[18] Kanto, Y., Tagawa, G., "A Dynamic of Contact Buckling Analysis by the Penalty Finite Element Method", Int. J. for Nume. Methods in Engrg., Vol.29, pp. 755-774, 1990.

[19] Nour-Omid, B., Wriggers, P., "A Two-level Iteration Method for Solution of Contact Problems", Comp. Meth. Appl. Mech. Engrg., Vol.54, pp. 131-144, 1986.

[20] Padovan, J., Moscarello, R., "Pantographing Self-adaptive GAP Elements", Computers & Structures, Vol.20, pp. 745-758, 1985.

[21] Abo-Elkhier, M., On the Numerical Solution of Nonlinear Problems in Continuum Mechanics, Ph. D. thesis, McMaster University, Hamilton, Canada, 1985.

[22] Derbalian, K. A., Lee, E. H., Mallet, R. L., McMeeking, R. M, "Finite Element Metal Forming Analysis with Spatially Fixed Mesh", In: Application of Numerical Methods for Forming Process. (Armen, H. eds.) ASME-AMD Vol.28, pp. 39-47, 1978.

[23] Oden, J. T., Braucnli, H. J., " On the Calculation of Consistent Stress Distributions in Finite Element Approximations", Int. J. Nume. Mech. Engrg., Vol.3, pp. 317-325, 1971.

[24] Ghosh, S., Kikuchi, N.," An Arbitrary Lagrangian-Eulerian Finite Element Method for Large Deformation Analysis of Elastic-viscoplastic Solids", Comp. Meth. Appl. Mech. Engrg., Vol.86, pp. 127-188, 1991.

[25] Cough, R., Sharp, R., Otero, I., Tipon, R., McCallen. R., "Application of ALE Techniques to Metal Forming Simulations", Advanced Computational Methods for Materials Modeling, American Society of Mechanical Engineers, Applied Mechanics Division, AMD Vol.180, pp. 133-140, 1993.

[26] Wang, J., Gadala, M. S., "Aspects in the Formulation and Application of Finite Elements to Metal Forming Problems", Proceedings of the International Conference, Modelling and Simulation in Metallurgical Engineering and Material Science, pp. 577-582, Beijing, P. R. China, 1996.

[27] Noh, W. F., "A Time-dependent Two-space-dimensional Coupled Eulerian-Lagrangian Code", In: Methods in Computational Physics (Alderb, B. et al eds), Vol.3, Academic Press, New York, USA, 1964.

[28] Wang, J., Gadala, M.S., "Formulation and Survey of ALE Method in Nonlinear Solid Mechanics", Finite Elements in Analysis and Design, Vol.24, pp. 253-269, 1997.

[29] Donea, J., Fasoli-stella, D. P., Giuliani, S., " Lagrangian and Eulerian Finite Element Techniques for Transient Fluid-structure Interaction Problems", Trans. SMiRT-4, Paper B1/2, pp. 115-19, San Francisco, USA, 1977.

[30] Belytschko, T., Kennedy J. M., "Computer Methods for Subassembly Simulation", Nuclear Engrg. Des., Vol.47, pp. 17-38, 1978.

[31] Lynch, P. R., O'Neill, K., "Continuously Deforming Finite Elements for the Solution of Parabolic Problems with and without Phase Change", Int. J. Nume. Meth. Engrg., Vol.17, pp. 81-96, 1981.

[32] Haber, R. B., Abel, J. F. "Contact-slip Analysis Using Mixed Displacements", J. Engrg. Mech. Vol.109, pp. 441-429, 1983.

[33] Huetink, J., "Analysis of Metal Forming Processes Based on a Combined Eulerian-Lagrangian Finite Formulation", In: Numerical Methods in Industrial Forming Processes (Pittman, J. F. T. eds), pp. 501–509, Pineridge, Swansea, UK, 1982.

[34] Huetink, J., Vreede, P. T., Lugt, J. V. D., "Progress in Mixed Eulerian-Lagrangian Finite Element Simulation of Forming Process", Int. J. Nume. Meth Engng., Vol.30, pp. 1441-1457, 1990.

[35] Benson, D. J., "An Efficient, Accurate, Simple ALE Method for Nonlinear Finite Element Programs", Comp. Meth. Appl. Mech. Engrg., Vol. 72, pp. 305-350, 1989.

[36] Haber, R. B., "A Mixed Eulerian-Lagrangian Displacement Model for Large-deformation Analysis in Solid Mechanics", Comp. Meth. Appl. Mech. Engng., Vol.43, pp. 277-292, 1984.

[37] Yamada, T., Kikuchi, F., "An Arbitrary Lagrangian-Eulerian Finite Element Method for Incompressible Hyperelasticity", Comp. Meth. Appl. Mech. Engrg., Vol.102, pp.149-177, 1993.

[38] Schreurs, P. J. G., Veldpaus, F. E., Brekelmans, W. A. M., "An Arbitrary Eulerian Lagrangian Finite Element Model for the Simulation of Geometric Nonlinear Hyperelastic and Elasto-plastic Deformation Processes", In: Numerical Methods in Industrial Forming Processes (Pittman, J. F. T. eds), pp. 491-500, Pineridge, Swansea, 1982.

[39] Schreurs, P. J. G., Veldpaus, F. E., Brekelmans, W. A. M., "Simulation of Forming Processes, Using the Arbitrary Eulerian-Lagrangian Formulation", Comp. Meth. Appl. Mech. Engng., Vol.58, pp. 19-36, 1986.

[40] Hughes, T. J. R., Liu, W. K., Zimmerman, T. K.: "Lagrangian-Eulerian Finite Element Formulation for Incompressible Viscous Flows", Comp. Meth. Appl. Mech. Engrg., Vol.29, pp. 329-349, 1981.

[41] Liu, W. K., Chang, H., Chen, J., Belytschko, T., "Arbitrary Lagrangian-Eulerian Petrov-Galerkin Finite Elements for Nonlinear Continua", Comp. Meth. Appl. Mech. Engrg., Vol.68, pp. 259-310, 1988.

[42] Liu, W. K., Chen, J. S. Belytschko, T., Zhang, Y. F.," Adaptive ALE Finite Elements with Particular Reference to External Work Rate on Frictional Interface", Comp. Mech. Appl. Mech. Engrg., Vol.93, pp. 189-216, 1991.

[43] Ghosh, S., Kikuchi, N., "Finite Element Formulation for the Simulation of Hot Sheet Metal Forming Process", Int. J. Engrg. Sci., Vol.26, pp. 143-161, 1988.

[44] Ghosh, S., "Finite Element Simulation of Some Extrusion Processes Using the Arbitrary Lagrangian-Eulerian Description", J. Mat. Shaping Technol., Vol.8, pp. 53-64, 1990.

[45] Huerta, A., Casadei, F ., "New ALE Applications in Non-linear Fast-transient Solid Dynamic", Engineering Computations, Vol.11, pp.317-345, 1994.

[46] Bathe, K. J., Ramm, E., Wilson, E. L., "Finite Element Formulation for Large deformation Dynamic Analysis", Int. J. Nume. Meth. Engrg., Vol.9, pp. 353-386, 1975.

[47] Bathe, K., Finite Element in Engineering Analysis, Prentice-Hall Inc., New Jersey, USA, 1982.

[48] Truesdell, C. , Toupin, R., "The Classical Field Theories", In: Encyclopedia of Physics, Vol.III/1, pp. 552-610, Spriner, Berlin-NewYork, 1960.

[49] Truesdell, C., "The Simplest Rate Theory of Pure Elasticity", Communications in Pure and Applied Mathematics, Vol.8, pp. 123-132, 1955.

[50] Dabke, P.D., Haque,I., Srikrishna, M. and Jackson, J.E., "A Knowledge-Based System for Generation and Control of Finite-Element Meshes in Forging Simulation", J. Mat. Eng. and Perf., Vol.1, pp. 415-428, 1992.

[51] Lush, A., "Numerical Grid Generation Used for Remeshing Finite Element Analyses of Metal Forming", In: Numerical Grid Generation in Computational Fluid Mechanics' 88 (Sengupta, S., et al, eds.), pp.1029-1038, Swansea, UK, 1988.

[52] Sparis, P. D., "A Method for Generating Boundary-orthogonal Curvilinear Coordinate Systems Using the Biharmonic Equation", Journal of Computational Physics, Vol.61, pp. 445-462, 1985.

[53] Harber, R., Shephard, M. S., Abel, J. E., Gallagher, R. H., Greenberg, D. P.,"A General Two Dimensional, Graphic Finite Element Preprocessor Utilizing Discrete Transfinite Mapping", Int. J. Nume. Meth. Eng. Vol.17, pp. 1015-1044, 1981.

[54] Gadala, M.S., Wang, J., "A Practical Procedure for Mesh Motion in Arbitrary Lagrangian-Eulerian Method", Submitted for publication in Engineering with Computers July, 1997.

[55] Gadala, M. S., Wang, J., "Numerical Treatment of Mesh Distortion Problems in FEA", Submitted for publication in Int. J. of Computers and Structures, Sept., 1997.

[56] Hughes, T. J. R., Winget, J., "Finite Rotation Effects in Numerical Integration of Rate Constitutive Equations Arising in Large-deformation Analysis", International Journal for Numerical Methods in Engineering, Vol.15, pp. 1862-1867, 1980.

[57] Hibbitt, Karlsson and Sorensen, Inc., ABAQUS Theory Manual, Hibbit, Karlsson & Sorensen, Inc., Pawtucket, USA, 1992.

[58] Pinsky, P. M., Ortiz M., Pister, K. S., "Numerical Integration of Rate Constitutive Equations in Finite Deformation Analysis", Computer Methods in Applied Mechanics and Engineering, Vol.40, pp. 137-158, 1983.

[59] Ortiz, M., Popov, E. P., "Accuracy and Stability of Integration Algorithms for Elastoplastic Constitutive Relation", International Journal for Numerical Methods in Engineering, Vol.21, pp. 1561-1576, 1985.

[60] Ortiz, M., Simo, J. C., "An Analysis of a New Class of Integration Algorithms for Elastoplastic Constitutive Relations", International Journal for Numerical Methods in Engineering, Vol.23, pp. 353-366, 1985.

[61] Wang, J.,Gadala, M.S., "Computational Implementation of Stress Integration in FE Analysis of Elasto-plastic Large Deformation Problems", Submitted for publication in Engineering Computations, March, 1997.

[62] Hinton, E., Campbell, J. S., "Local and Global Smoothing of Discontinuous Finite Element Functions Using a Least Square Method", Int. J. of Nume. Mech. Eng., Vol.8, pp. 461-480, 1974.

[63] Oden, J. T., Braucnli, H. J.,"On the Calculation of Consistent Stress Distributions in Finite Element Application", Int. Jour. Nume. Meth. Engrg., Vol.3, pp. 317-325, 1971.

[64] Malvern, L. E., Introduction to the Mechanics of a Continuous Medium, Prentice-Hall Inc, New Jersey, USA, 1969.

[65] Bathe, K. J., Cimento, A. P., "Some Practical Procedures for the Solution of Nonlinear Finite Element Equations", Computer Methods in Applied Mechanics and Engineering, Vol.22, pp.59-85, 1980.

[66] Owen, D. J. R., Hinton, E., Finite Element in Plasticity, Theory and Practice, Pineridge Press Limited, Swansea, UK, 1980

[67] Cook, R. D., Malkus, D. S., Plesha, M. E., Concepts and Applications of Finite Element Analysis, John Wiley & Sons, New York, USA, 1989.

[68] Burnett, D. S., Finite Element Analysis, from Concepts to Applications, Addison-wesley Publishing Company, Menlo Park, California, USA, 1987.

[69] Huerta, A., Casadei, F., "New ALE Applications in Non-linear Fast-transient Solid Dynamic", Engineering, Computations, Vol.11, pp. 317-345, 1982.

[70] Yamada, Y., Hirakawa, H., "Large Deformation and Instability Analysis in Metal Forming Processing", Application of Numerical Methods to forming Processes, ASME, AMD-Vol.28, (Edited by H. Armen), pp.27-38, 1978.

[71] Farmer, L. E., Oxley, P. L. B., "A Slip Field for Plane Strain Extrusion of a Strain-hardening Material", J. Mech. Phys. Solids, Vol.19, pp.369-388, 1971.

[72] Lyman, T., Metals Handbook, Vol.1, Properties and Selection of Metals, American Society for Metals, Metals Park, Ohio, USA, 1961.

[73] Li ,G., Kobayashi, S., "Rigid-plastic Finite Element Analysis of Plain Strain Rolling", J. of Eng. for Industry, Vol.104, pp.55-64,1982.

[74] Hwang, S.M., Joun, M.S., "Analysis of Hot Strip Rolling by a Penalty Rigid-viscoplastic Finite Element Method", Int. J. Mech. Sci., Vol.34, pp.971-985, 1992.

[75] Liu, C., Hartley, P., Sturgess, C. E. N., Rowe, G. W., "Finite Element Modelling of Deformation and Spread in Slab Rolling", Int. J. Mech. Sci., Vol.29, pp.271-283, 1987.

[76] Al-Salehi, F. A. R. , Fibrank, T. C., Lancaster, P. R., "An Experimental Determination of the Roll Pressure Distributions in Cold Rolling", Int. J. Mech. Sci. Vol.15, pp.693-710, 1973.

[77] Wusatowski, Z., Fundamentals of Rolling, Pergamon Press, New York, USA, 1969.

[78] Chitkara, N. R., Johnson, W., " Plane Strain Compression of pre-shaped material between Wedge-shaped Dies", Int. J. Mech. Sci., Vol.4, pp151-164, 1974.

Appendix A

Material and Referential Time Rate

A physical quantity ${}^tf = f({}^t\mathbf{x})$ at Material Reference System (MRS) point $P({}^tx_1, {}^tx_2, {}^tx_3)$ at time t becomes ${}^{t+\Delta t}f = f({}^{t+\Delta t}\mathbf{x})$ at time $t+\Delta t$ as the point P moves to $Q({}^{t+\Delta t}x_1, {}^{t+\Delta t}x_2, {}^{t+\Delta t}x_3)$. The MRS change of ${}^tf$ may be expressed by:

$$\Delta_{\mathbf{x}}f \quad = \quad f({}^{t+\Delta t}\mathbf{x}, t+\Delta t) - f({}^t\mathbf{x}, t) \tag{A.1}$$

Similarly, the change of ${}^tf$ at the corresponding Computational Reference System (CRS) point $P({}^t\chi_1, {}^t\chi_2, {}^t\chi_3)$ when the point moves to $R({}^{t+\Delta t}\chi_1, {}^{t+\Delta t}\chi_2, {}^{t+\Delta t}\chi_3)$ is denoted by:

$$\Delta_{\boldsymbol{\chi}}f \quad = \quad f({}^{t+\Delta t}\boldsymbol{\chi}, t+\Delta t) - f({}^t\boldsymbol{\chi}, t) \tag{A.2}$$

To derive the relation between $\Delta_{\mathbf{x}}f$ and $\Delta_{\boldsymbol{\chi}}f$, we assume that the MRS and CRS points coincide at point P at time t, as shown in Figure (A.1), so that:

$${}^t\mathbf{x} = {}^t\boldsymbol{\chi}$$

and,

$$f({}^t\boldsymbol{\chi}, t) = f({}^t\mathbf{x}, t) = {}^tf \tag{A.3}$$

at time $t + \Delta t$,

$${}^{t+\Delta t}\mathbf{x} = {}^t\mathbf{x} + \Delta\mathbf{x} \tag{A.4}$$

$${}^{t+\Delta t}\boldsymbol{\chi} = {}^t\boldsymbol{\chi} + \Delta\boldsymbol{\chi} \tag{A.5}$$

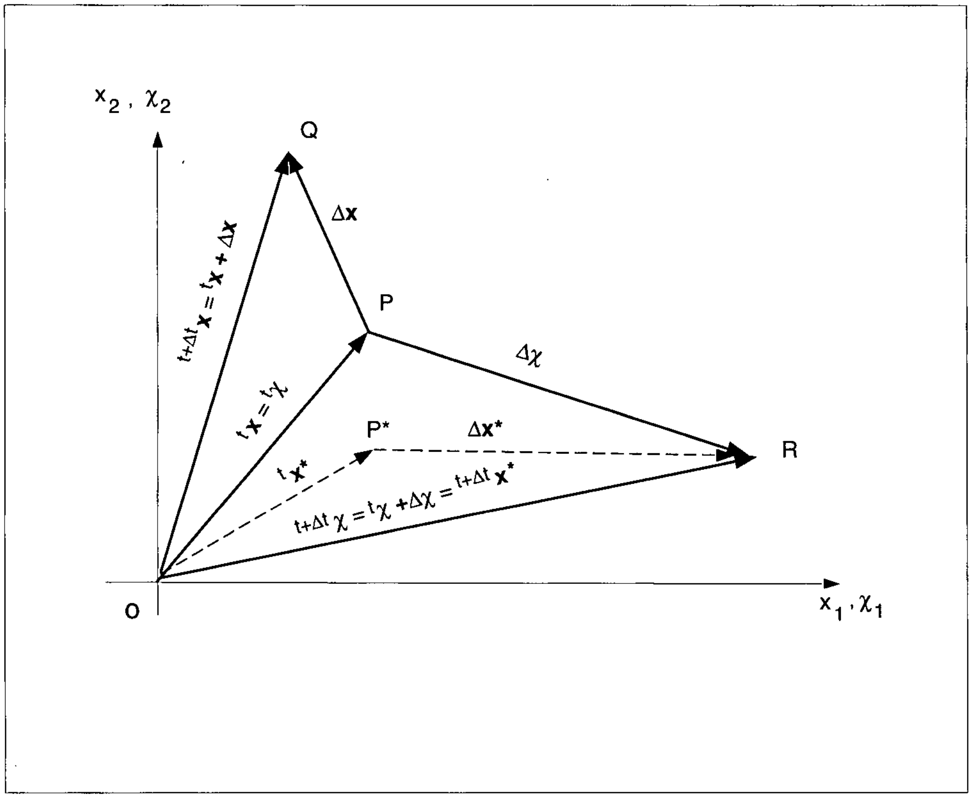

x_2 , χ_2
Q
$\Delta \mathbf{x}$
$^{t+\Delta t}\mathbf{x} = {}^{t}\mathbf{x} + \Delta \mathbf{x}$
P
$\Delta \chi$
$^{t}\mathbf{x} = {}^{t}\chi$
P*
$\Delta \mathbf{x}^*$
R
$^{t}\mathbf{x}^*$
$^{t+\Delta t}\chi = {}^{t}\chi + \Delta\chi = {}^{t+\Delta t}\mathbf{x}^*$
O
x_1 , χ_1

Figure A.1: The description of relation between mesh motion and material motion

Assuming that the MRS point located at R at time $t + \Delta t$ comes from a material point $P^*({}^t x_1^*, {}^t x_2^*, {}^t x_3^*)$ with displacement $\Delta\mathbf{x}^*$, we have:

$$ {}^{t+\Delta t}\chi = {}^t\mathbf{x}^* + \Delta\mathbf{x}^* = {}^{t+\Delta t}\mathbf{x}^* \tag{A.6}$$

$$ f({}^{t+\Delta t}\chi, t + \Delta t) = f({}^{t+\Delta t}\mathbf{x}^*, t + \Delta t) \tag{A.7}$$

Substitute Equation (A.3) and (A.7) into Equation (A.2):

$$\begin{aligned}
\Delta_{\chi} f &= f({}^{t+\Delta t}\mathbf{x}^*, t + \Delta t) - f({}^{t+\Delta t}\mathbf{x}, t + \Delta t) + f({}^{t+\Delta t}\mathbf{x}, t + \Delta t) - f({}^t\mathbf{x}, t) \\
&= \Delta_{\mathbf{x}} f + \left(f({}^{t+\Delta t}\mathbf{x}^*, t + \Delta t) - f({}^{t+\Delta t}\mathbf{x}, t + \Delta t) \right) \\
&= \Delta_{\mathbf{x}} f + \nabla f({}^{t+\Delta t}\mathbf{x}, t + \Delta t) \bullet ({}^{t+\Delta t}\mathbf{x}^* -{}^{t+\Delta t} \mathbf{x}) + h.o.t.
\end{aligned}$$

where ∇ is gradient symbol and $h.o.t.$ is higher order term of $({}^{t+\Delta t}\mathbf{x}^* -{}^{t+\Delta t} \mathbf{x})$. But,

$$\begin{aligned}
{}^{t+\Delta t}\mathbf{x}^* -{}^{t+\Delta t} \mathbf{x} &= {}^{t+\Delta t}\chi -{}^{t+\Delta t} \mathbf{x} \\
&= {}^t\chi + \Delta\chi -{}^t \mathbf{x} - \Delta\mathbf{x} \\
&= \Delta\chi - \Delta\mathbf{x}
\end{aligned}$$

Therefore,

$$\Delta_{\chi} f = \Delta_{\mathbf{x}} f + \nabla f({}^{t+\Delta t}\mathbf{x}, t + \Delta t) \bullet (\Delta\chi - \Delta\mathbf{x}) + h.o.t.$$

Where $h.o.t.$ is higher order term of $(\Delta\chi - \Delta\mathbf{x})$. If $h.o.t.$ is ignored, we have:

$$\Delta_{\chi} f = \Delta_{\mathbf{x}} f + \frac{\partial^{t+\Delta t} f}{\partial^{t+\Delta t} x_k}(u_k^c - u_k) \tag{A.8}$$

Dividing the above equation by Δt and taking the limit when $\Delta t \to 0$,

$$ {}^t f^{\wedge} ={}^t f^{\cdot} + \frac{\partial^t f}{\partial^t x_k}({}^t v_k^c - {}^t v_k) \tag{A.9}$$

Appendix B

Updated Lagrangian Formulation

An updated-Lagrangian formulation given by Rice [3] is based on the following equation:

$$\begin{aligned} &\int_{{}^tV} \left({}^t\tau_{ij}^{\cdot J} \delta {}^tD_{ij} - \frac{1}{2} {}^t\sigma_{ij} \delta \left(2 {}^tD_{ik} {}^tD_{kj} - \frac{\partial^t v_k}{\partial^t x_i} \frac{\partial^t v_k}{\partial^t x_j} \right) \right) d^tV \\ = &\int_{{}^tV} {}^tf_i^{B\cdot} \delta v_i d^tV + \int_{{}^tS} {}^tf_i^{S\cdot} \delta v_i d^tS \end{aligned} \qquad \text{(B.10)}$$

where ${}^tD_{ij} = \frac{1}{2}\left(\frac{\partial^t v_i}{\partial^t x_j} + \frac{\partial^t v_j}{\partial^t x_i}\right)$ is the deformation rate at time t and ${}^t\tau_{ij}^{\cdot J}$ is Jaumann rate of Kirchoff stress which is defined by:

$${}^t\tau_{ij} = {}^tJ {}^t\sigma_{ij}$$

where ${}^t\sigma_{ij}$ is Cauchy stress and ${}^tJ$ is the ratio of volume in the current state to the volume in reference state. The time rate can be written as:

$${}^t\tau_{ij}^{\cdot} = {}^tJ^{\cdot} {}^t\sigma_{ij} + {}^tJ {}^t\sigma_{ij}^{\cdot}$$

using the Jaumann rate for both ${}^t\tau_{ij}^{\cdot}$ and ${}^t\sigma_{ij}^{\cdot}$, we get:

$${}^t\tau_{ij}^{\cdot J} = {}^tJ^{\cdot} {}^t\sigma_{ij} +^t J^t\sigma_{ij}^{\cdot J} \qquad \text{(B.11)}$$

Since ${}^t\tau_{ij}^{\cdot J}$ is symmetric, the right hand side of Equation (B.10) may be expressed by:

$$\begin{aligned} LHS &= \int_{{}^tV} \left({}^t\tau_{ij}^{\cdot J} \frac{\partial \delta v_i}{\partial^t x_j} - 2\delta {}^tD_{ki} {}^t\sigma_{ij} {}^tD_{kj} + \frac{\partial \delta v_k}{\partial^t x_i} {}^t\sigma_{ij} \frac{\partial^t v_k}{\partial^t x_j} \right) d^tV \\ &= \int_{{}^tV} \left({}^t\tau_{ij}^{\cdot J} \frac{\partial \delta v_i}{\partial^t x_j} - \frac{1}{2} \left(\frac{\partial \delta v_k}{\partial^t x_i} + \frac{\partial \delta v_i}{\partial^t x_k} \right) {}^t\sigma_{ij} \left(\frac{\partial^t v_k}{\partial^t x_j} + \frac{\partial^t v_j}{\partial^t x_k} \right) + \frac{\partial \delta v_k}{\partial^t x_i} {}^t\sigma_{ij} \frac{\partial^t v_k}{\partial^t x_j} \right) d^tV \\ &= \int_{{}^tV} \frac{\partial \delta v_i}{\partial^t x_j} \left({}^t\tau_{ij}^{\cdot J} - \frac{{}^t\sigma_{ik}}{2} \left(\frac{\partial^t v_j}{\partial^t x_k} + \frac{\partial^t v_k}{\partial^t x_j} \right) + \frac{{}^t\sigma_{jk}}{2} \left(\frac{\partial^t v_i}{\partial^t x_k} - \frac{\partial^t v_k}{\partial^t x_i} \right) \right) d^tV \end{aligned} \qquad \text{(B.12)}$$

Substituting ${}^{t}J^{\cdot} = {}^{t}J\frac{\partial^t v_k}{\partial^t x_k}$ [48], and choosing the current state as a reference so that ${}^{t}J = 1$. The Equation (B.11) will take the form:

$$ {}^{t}\tau_{ij}^{\cdot J} = {}^{t}\sigma_{ij}\frac{\partial^t v_k}{\partial^t x_k} + \sigma_{ij}^{\cdot J} $$

Substituting the above equation in Equation (B.12), we get:

$$ LHS = \int_{{}^{t}V} \frac{\partial \delta v_i}{\partial^t x_j}\left({}^{t}\sigma_{ij}^{\cdot J} +^{t} \sigma_{ij}\frac{\partial^t v_k}{\partial^t x_k} - \frac{{}^{t}\sigma_{ik}}{2}\left(\frac{\partial^t v_j}{\partial^t x_k} + \frac{\partial^t v_k}{\partial^t x_j}\right) + \frac{{}^{t}\sigma_{jk}}{2}\left(\frac{\partial^t v_i}{\partial^t x_k} - \frac{\partial^t v_k}{\partial^t x_i}\right)\right) d^tV $$

So the updated Lagrangian formulation Equation (B.10) takes the following form:

$$ \begin{aligned} & \int_{{}^{t}V} \frac{\partial \delta v_i}{\partial^t x_j}\left({}^{t}\sigma_{ij}^{\cdot J} - \frac{{}^{t}\sigma_{ik}}{2}\left(\frac{\partial^t v_j}{\partial^t x_k} + \frac{\partial^t v_k}{\partial^t x_j}\right) + \frac{{}^{t}\sigma_{jk}}{2}\left(\frac{\partial^t v_i}{\partial^t x_k} - \frac{\partial^t v_k}{\partial^t x_i}\right) +^{t} \sigma_{ij}\frac{\partial^t v_k}{\partial^t x_k}\right) d^tV \\ = & \int_{{}^{t}V} \delta v_i^t f_i^{B\cdot} d^tV + \int_{{}^{t}S} \delta v_i^t f_i^{S\cdot} d^tS \end{aligned} \tag{B.13} $$

Equation (B.13) is same as Equation (2.25) in Chapter 2 except the load correction terms introduced in Equation (2.25).

Appendix C

Mapping Between Material and Mesh Domain

It is important to ensure that the mesh motion scheme proposed Equation (3.3) guarantees one-to-one mapping between material points and mesh points, i.e., ${}^{t}J^{c} = \left|\frac{\partial^{t}x_i}{\partial^{t}\chi_j}\right| \neq 0$. This may be verified as follow:

At time $t - \Delta t$, assume ${}^{t-\Delta t}x_i =^{t-\Delta t} \chi_i$, i.e., material points and mesh points coincide and from $t - \Delta t$ to t the material velocity is ${}^{t-\Delta t}v_i$, then,

$$\begin{aligned} {}^{t}x_i &= {}^{t-\Delta t}x_i + \Delta t^{t-\Delta t}v_i = {}^{t-\Delta t}\chi_i + \Delta t^{t-\Delta t}v_i \\ {}^{t}\chi_i &= {}^{t-\Delta t}\chi_i + \Delta t^{t-\Delta t}v_i^c = {}^{t-\Delta t}\chi_i + \Delta t\left(a_i + b_{(i)}{}^{t-\Delta t}v_{(i)}\right) \end{aligned} \tag{C.14}$$

From which we obtain a relation between ${}^{t}x_i$ and ${}^{t}\chi_i$ as:

$$\begin{aligned} {}^{t}x_i &= {}^{t}\chi_i - \Delta t\left(a_i + b_{(i)}{}^{t-\Delta t}v_{(i)}\right) + \Delta t^{t-\Delta t}v_i \\ &= {}^{t}\chi_i - \Delta t a_i + \left(1 - b_{(i)}\right)\Delta t^{t-\Delta t}v_{(i)} \end{aligned} \tag{C.15}$$

and the determinant of the Jacobian transformation becomes:

$$\begin{aligned} {}^{t}J^{c} &= \left|\frac{\partial^{t}x_i}{\partial^{t}\chi_j}\right| \\ &= \left|\delta_{ij} + \left(1 - b_{(i)}\right)\Delta t\frac{\partial^{t-\Delta t}v_{(i)}}{\partial^{t}\chi_j}\right| \end{aligned} \tag{C.16}$$

No summation on (i). Since ${}^{t-\Delta t}v_i$ is independent of ${}^{t}\chi_i$, we have $\frac{\partial^{t-\Delta t}v_{(i)}}{\partial^{t}\chi_j} = 0$ and:

$$\begin{aligned} {}^{t}J^{c} &= \left|\frac{\partial^{t}x_i}{\partial^{t}\chi_j}\right| \\ &= |\delta_{ij}| \neq 0 \end{aligned} \tag{C.17}$$

Appendix D

Stress Increment Procedure

We consider decomposing the general deformation into two steps; straining and rigid body motion. In the first step, due to straining, the Cauchy stress would have an increment of

$$\Delta\sigma_{ij} = C_{ijkl}{}^{t}D_{kl}\Delta t \tag{D.18}$$

where, C_{ijkl} is the element constitutive tensor, Δt is the time increment and ${}^{t}D_{kl}$ is the rate of deformation tensor. The stress after the first step would be:

$$^{t+\Delta t}\sigma_{ij}^{(I)} = {}^{t}\sigma_{ij} + \Delta\sigma_{ij} = {}_{t}^{t+\Delta t}S_{ij}^{(I)} \tag{D.19}$$

where, ${}_{t}^{t+\Delta t}S_{ij}^{(I)}$ is the 2nd Piola-Kirchhoff stress at an imaginary intermediate state with respect to the configuration at time t and ${}^{t+\Delta t}\sigma_{ij}^{(I)}$ is the corresponding Cauchy stress. The 2nd Piola-Kirchhoff stress is related to Cauchy stress by:

$$^{t+\Delta t}\sigma_{ij} = \frac{1}{\left|{}_{t}^{t+\Delta t}\mathbf{F}\right|}\frac{\partial^{t+\Delta t}x_i}{\partial^{t}x_m}{}_{t}^{t+\Delta t}S_{mn}\frac{\partial^{t+\Delta t}x_j}{\partial^{t}x_n} \tag{D.20}$$

and since the second step is only a rigid body motion, then,

$${}_{t}^{t+\Delta t}S_{mn} = {}_{t}^{t+\Delta t}S_{mn}^{(I)} \tag{D.21}$$

Substituting Equation (D.21) and (D.19) into Equation (D.20), we get:

$$^{t+\Delta t}\sigma_{ij} = \frac{1}{\left|{}_{t}^{t+\Delta t}\mathbf{F}\right|}\frac{\partial^{t+\Delta t}x_i}{\partial^{t}x_m}{}^{t}\sigma_{mn}\frac{\partial^{t+\Delta t}x_j}{\partial^{t}x_n} + \frac{1}{\left|{}_{t}^{t+\Delta t}\mathbf{F}\right|}\frac{\partial^{t+\Delta t}x_i}{\partial^{t}x_m}\Delta\sigma_{mn}\frac{\partial^{t+\Delta t}x_j}{\partial^{t}x_n} \tag{D.22}$$

Equation (D.22) is same as the one developed by Pinsky [58] and both are consistent with Truesdell stress rate constitutive equation.