

DISS. ETH No 11107

**FE Simulation of Bulk Forming Processes with a
Mixed Eulerian-Lagrangian Formulation**

A dissertation submitted to the
SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of
Doctor of Technical Sciences

presented by

LONGCHANG TONG

M. Sc. Tsinghua University, Peking, China

born on 6th Feb. 1951

citizen of P. R. China

accepted on the recommendation of

Professor Dr. J. Reissner, examiner

Professor Dr. E. Anderheggen, co-examiner

1995

Acknowledgments

I would like to express my sincere thanks to Prof. Dr. J. Reissner for his constant support and guidance during this study. I am also very grateful to Prof. Dr. E. Anderheggen for many valuable comments and hints and for accepting the work of co-examiner. Special thanks are due to Dr. P. Hora for his guidance and many fruitful ideas when I took part in the work of numerical simulation and implementation of the FE program PressForm. Mr. W. Kubli provided the code for the equation solver, Dr. R. Maisch implemented the contact algorithm and Mr. R. Linder developed the excellent post-processor for PressForm. Without their cooperation the examples presented could not have been calculated and useful conclusions could not have been made. Their cooperation is gratefully appreciated. Thanks are also due to Dr. E.G. Prater for correcting the English text.

Financial support for the project was provided by the "Commission for Advancement of Scientific Research (KWF)". To the companies Metallwerk Swiss Dornach and Aluminium Menziken I owe my thanks for supporting this research work. My special thanks also go to Dr. F. Vanini of Metallwerk Swiss Dornach for many useful discussions and valuable suggestions.

10. April, 1995 Zurich

Zusammenfassung	iii
Abstract	v
1. Introduction	1
1.1 Some examples of failure during extrusion processes.....	1
1.2 Main difficulties in the simulation of 3-D bulk forming processes.....	3
1.3 State-of-the-Art of FE simulation of forming processes	4
1.4 Aim of the dissertation	5
2. Basic equations.....	8
2.1 Basic equations of continuum mechanics.....	8
2.2 Different strain and stress tensors.....	8
2.2.1 Strains, strain rates and strain increments	9
2.2.2 Cauchy stress and 2nd Piola-Kirchhof stress	11
2.3 Elastoplastic and rigid-plastic descriptions	13
2.4 Thermal analysis.....	16
3. Arbitrary Lagrangian-Eulerian Formulation	18
3.1 Two different methods of observation.....	18
3.1.1 Lagrangian method (material description)	18
3.1.2 Eulerian method (spatial description).....	19
3.2 Comparison of the two methods.....	21
3.3 Arbitrary Lagrangian-Eulerian description	25
3.4 Computational steps in weakly coupled analysis	28
3.5 Computation of the convective term	29
4. Mesh Discretization and Remeshing.....	33
4.1 Different updating and remeshing methods.....	33
4.1.1 Introduction	33
4.1.2 Combination of Eulerian and Lagrangian meshes.....	34
4.1.3 Surface and internal nodes.....	36
4.1.4 Updated Lagrangian method	38
4.2 Mesh generation	39
4.2.1 Mesh generator for profile extrusion	40
4.2.2 Transition between elements with different sizes.....	43
4.2.3 Example	45
4.3 On meshing and remeshing	45
4.3.1 Two basic ways of automatic mesh generation.....	45
4.3.2 An effective method to reduce the size of the system	46
4.3.3 Some future prospects	47
5. Rigid-plastic Element Formulation	49
5.1 Two different methods to implement incompressibility for the rigid-plastic element formulation	49
5.1.1 The Lagrange multiplier method	49
5.1.2 The Penalty method.....	54
5.1.3 Comparison of the two methods.....	56
5.2 Element formulation for stress-strain computations.....	58

5.3	Element formulation for thermal analysis	62
6.	Discussion of some aspects of the FE program	68
6.1	Comparison of different algorithms to solve the equations.....	68
6.1.1	Direct solvers.....	68
6.1.2	Iterative solver using preconditioned conjugate gradient method	70
6.1.3	Assessment of different algorithms using a computational example.....	72
6.2	Description of friction	75
6.2.1	Friction in bulk forming	75
6.2.2	Modelling of friction	76
6.2.3	Measures to stabilize the computation of friction	78
6.3	Estimate of the initial values for the iteration	81
6.3.1	Stationary and quasi-stationary cases	81
6.3.2	Non-stationary material flow.....	82
6.4	Criterion to determine the increment size	83
7.	Numerical Tests of the FE Program PressForm.....	84
7.1	The structure of the program	84
7.2	Forward extrusion of cross profiles	85
7.3	Simulation of extrusion with two openings.....	89
7.4	Benchmark test for a hot forging process.....	91
7.5	Heat conduction.....	95
8.	Examples of Application.....	99
8.1	Forward extrusion process of a T-profile material.....	99
8.1.1	Flow condition determined by the geometry	101
8.1.2	Influence of friction	102
8.1.3	Effect of recrystallization	104
8.2	Weakly coupled thermal analysis of extrusion processes.....	106
8.3	Some further examples	108
8.3.1	Forward-backward extrusion.....	108
8.3.2	Optimization of the position of the die.....	111
8.4	Simulation of a cold forging part using the updated Lagrangian method	112
9.	Conclusions	115
	References	116
	Notation	121
	Resume	123

Zusammenfassung

Die Methode der finiten Elemente wird seit mehr als zwanzig Jahren zur Simulation von Massivumformprozessen eingesetzt. Einem echten industriellen Einsatz standen bisher jedoch einerseits die Schwierigkeiten mit dem extrem hohen Rechenaufwand und andererseits die erheblichen Netzverzerrungen, verursacht durch grosse Deformationen, im Wege. Durch Anwendung von z.T. manuellen Remeshing-Verfahren konnte das letztere Problem zwar teilweise entschärft werden, keines der bisher existierenden Programme vermochte jedoch allgemeine 3D-Probleme effizient zu lösen.

Dank der stark gestiegenen Computerleistung ist es neulich möglich, Simulationen an realen Teilen durchzuführen. Keine zufriedenstellenden Lösungen liegen jedoch nach wie vor für das Remeshingproblem vor.

In der vorliegenden Arbeit wird ein neu entwickeltes FEM-Programm vorgestellt, welches auf der gemischten Lagrange-Euler-Formulierung (ALE) beruht. Dieses verwendet in Zonen mit starken Deformationen Euler-Netze. In den übrigen Zonen werden Lagrange-Elemente eingeführt, welche der Form der deformierten Bereiche folgen.

Die konvektiven Terme in den Euler-Zonen werden bei der Aufdatierung des Netzes mitberücksichtigt. Um die beiden Netzarten zu verbinden, werden am Zonenübergang neue Elemente eingeführt.

Die ALE-Methode eignet sich besonders gut zur Simulation von Strangpress-, Drahtzieh- und Walzprozessen. Bei diesen Verfahren kann das Rechengebiet klar in Euler- und Lagrange-Zonen unterteilt werden. Auch die Kriterien für das Generieren oder Löschen von Elementen sind einfach zu definieren.

Die ALE-Formulierung ist darüber hinaus auch bei allgemeinen Massivumformprozessen anwendbar. Die Knoten werden dazu in zwei Gruppen, nämlich Oberflächenknoten und interne Knoten, unterteilt. Die Oberflächenknoten werden gemäss der Lagrange-Formulierung aufdatiert, wohingegen die internen Knoten stationär bleiben. Da dadurch das Euler-Netz regelmässig bleibt und die Oberflächenknoten relativ einfach angepasst werden können, wird eine Netzverzerrung vermieden. Dieser Algorithmus hat sich bei der Simulation von 2D-Massivumformprozessen als äusserst vorteilhaft erwiesen.

Aufbauend auf diesem Konzept wurde das Special-Purpose-Programm 'PressForm' entwickelt. Der dazugehörige Preprozessor vermag sowohl das Rechnernetz als auch die Diskretisierung der Presswerkzeuge automatisch zu generieren, wobei einfach zu spezifizierende Eingabedaten verwendet werden. Zur Lösung der grossen linearen Gleichungssysteme wird ein iterativer Konjugierte-Gradienten-Löser mit Element-by-Element-Speicherung (EBE) eingesetzt. Für die Simulation von Strangpressvorgängen kann das Netz automatisch kontrolliert und falls notwendig, neu generiert werden.

Da bei Massivumformprozessen der Einfluss der Temperatur oft entscheidend ist, verfügt das Programm ebenfalls über eine schwach gekoppelte thermische Analyse.

Die Anwendbarkeit der Methode wird an zahlreichen Beispielen aus dem industriellen Umfeld getestet. Diese zeigen Möglichkeiten zur Optimierung der Prozessparameter und zur Voraussage des Versagens.

Das Ziel zukünftiger Entwicklungen ist die Bereitstellung vollautomatischer adaptiver Netzgeneratoren für den allgemeinen 3D-Fall. Eine zusätzliche Beschleunigung des linearen Gleichungslösers könnte die Leistung des Gesamtsystems zudem weiter verbessern. Für die nahe Zukunft kann mit einem stark zunehmenden Einsatz der FE-Simulation in der Massivumformung gerechnet werden.

Abstract

Efforts have been made for a long time to establish the finite element method (FEM) in industrial applications to simulate bulk forming processes. The main difficulties arise, on the one hand, from the enormous computational costs, and, on the other, from the resulting large deformations during bulk forming processes, which cause the computational mesh to deteriorate. Remeshing algorithms can solve partially the latter problem but unfortunately none of the existing packages has been able to solve generally 3-D complex forms.

However, the rapid development of computer technology has made it possible to compute real 3-D forming parts. Therefore, solving the problem caused by the distortion of the mesh becomes a very urgent task.

A method is proposed in the present work based on the concept of an arbitrary Lagrangian-Eulerian formulation (ALE). This method uses Eulerian meshes to avoid strong mesh distortion in the zones of strong deformation, and defines Lagrangian nodes in other zones to follow the shape of the deformed domain. The convective terms in the Eulerian zones are taken into account when the mesh is updated. New elements are generated or some elements are deleted to couple these two kinds of meshes. This is especially suitable for the simulation of extrusion, wire drawing and rolling processes. In these cases the domain can be clearly divided into two kinds of zones, namely Eulerian and Lagrangian zones. The cases when the elements should be generated or deleted are also easy to prescribe.

For the FE simulation of general bulk forming processes the ALE formulation can also be applied. The nodes are divided up into two groups, namely surface nodes and internal nodes. The surface nodes are updated as for a Lagrangian mesh and the internal nodes act as an Eulerian mesh. The distortion of the mesh is eliminated since the Eulerian mesh remains regular and it is relatively easy to adjust the surface nodes. This algorithm has shown its superiority when it is used in the simulation of 2-D bulk forming processes.

An FE program PressForm has been developed according to this concept as an easy to use special purpose package. The preprocessor is able to generate the computational

mesh as well as the discretization of the extrusion die and punch using very simple input data. A preconditioned conjugate gradient iterative solver together with the element-by-element (EBE) method is used to solve large systems of linear equations. For the simulation of extrusion processes the mesh is also automatically controlled and regenerated where necessary.

A weakly coupled thermal analysis with the ALE formulation has also been implemented in PressForm. This is an important aspect, especially for the simulation of bulk forming processes of non-ferrous metals.

The program has been well tested using different examples. These computational examples show the validity of this formulation and the power of the method to simulate forming processes as well as to predict possible failures.

Further development should include an adaptive mesh generator and regenerator for general 3-D cases. Acceleration of the solution procedure for the set of linear equations is also a very important factor in order to achieve better performance. More and more applications of FE simulation in the bulk forming industry can be expected in the near future.

1. Introduction

The finite element (FE) method has been recognized as a powerful tool to simulate forming processes for more than twenty years. Systematic information, which is almost impossible or very difficult to obtain by other methods, can be provided by FE simulation. Based on the computational results, the engineer is able to predict the failures which may occur during processes and to easily change the parameters to improve the processes. Due to the rapid advances in computer technology, the computation time of FE simulation has been shortened significantly and the accuracy of computation is continually increasing.

However, the application of FE simulation in the forming industry, especially in the bulk forming industry, is still limited. Difficulties arise, on the one hand, from the enormous computational costs, and, on the other, from the large deformations taking place inside the workpiece during the bulk forming processes. When the mesh is updated according to the material flow, it quickly becomes very strongly distorted. This phenomenon causes the computation to break down or leads to unreliable results.

An efficient and reliable FE package is indispensable for promoting the application of the FE method in the bulk forming industry. In order to implement such a package, investigations of different aspects have been carried out.

1.1 Some examples of failure during extrusion processes

Extrusion is one of the most important forming processes. In comparison with other methods, the workpiece can be deformed in just one step to get the final shape. Because of its efficiency forward extrusion is widely used in the metallurgical industries to produce shaped material. The machine industry also uses extrusion process to produce high precision machine parts. In order to control the material flow or to produce a complex form, the extrusion die might also have a preforming cavity. The material flows into this cavity, gets the desired distribution of velocity or is even welded there, and flows out of the die afterwards (Fig. 1. 1).

Problems might appear during extrusion processes. The shaped material might flow out of the die with an undesirable velocity distribution. Such a velocity distribution causes the product to bend. In more serious cases the desired shape cannot be produced at all.

The precision of the product is also a determinant factor in extrusion processes. If tensile stress is induced in some zones inside the workpiece, the dimension of the product may deviate from the dimension of the opening of the extrusion die.

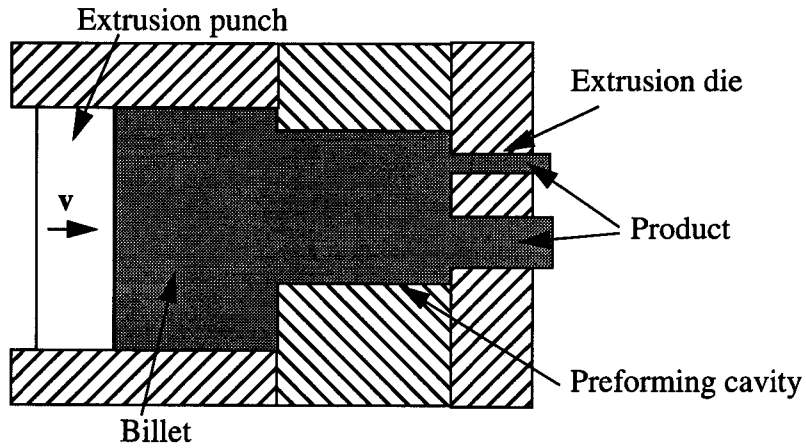


Fig. 1. 1 Forward extrusion with a preforming cavity

Surface fracture can also appear if the lubrication is poor or the temperature in the workpiece is not homogeneous. Fig. 1. 2 shows several cases of failure in forward extrusion.

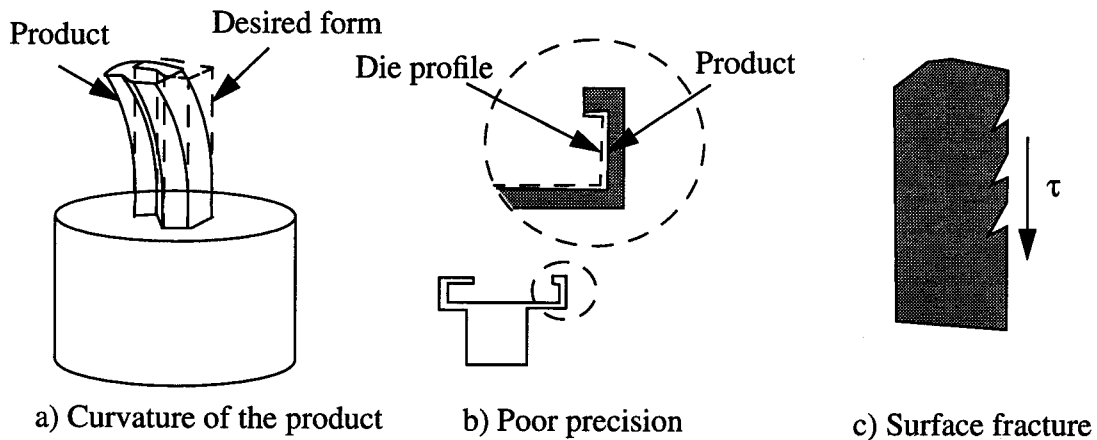


Fig. 1. 2 Predominant failure types in the forward extrusion process

Sometimes failures of the metallurgical properties of the materials also occur. For example, the size of crystal grains of the material after extrusion might exceed the allowed value. The large grains impair the mechanical properties of the material. This is

a particularly sensitive behaviour for many non-ferrous metals.

In addition to these problems the forming forces needed in extrusion processes are usually very large. The capacity of the press and the strength of the tools are the primary factors in determining whether the extrusion process can be used. Excessive forming forces can cause failure of the extrusion tools.

1.2 Main difficulties in the simulation of 3-D bulk forming processes

Usually the deformations inside the workpiece are very large in bulk forming processes. For example, in the forward extrusion process, the ratio of the sectional areas of billet and product is generally in the range 20-100. This means that the mean deformation of the workpiece is more than 2000%. Locally, deformations can be far greater. In the wire industry the same degree of deformation is achieved through many drawings. Annealing processes are needed in certain stages to keep the wire from failure in fracture. By contrast, in forward extrusion processes the deformations are completed in just one step, the deformation rate being as high as 300/s.

When such processes are simulated using the updated Lagrangian method, because the deformations are so large the mesh becomes seriously distorted in just a couple of small increments when the location of its nodes is updated to follow the material flow. The elements deteriorate rapidly. It is no longer possible to deliver reliable results with such meshes as the computation breaks down due to numerical errors.

In order to overcome this difficulty two procedures are proposed. One is the remeshing technique, which re-generates the computational mesh and transfers the data set from the old mesh to the new one. The other is the so-called arbitrary Lagrangian-Eulerian (ALE) method, which avoids excessive distortions of the mesh by separating the updating of the nodes from the material flow.

At present, existing remeshing algorithms for 3-D complex domains cannot ensure satisfactory operation. A breakthrough in this field is also not to be expected in the near future.

Another problem in the 3-D simulation of bulk forming are the very high computational

costs. Taking a discretization to divide up the domain with N nodes in each dimension, the number of degrees of freedom for a 3-D system is $3 N^3$ while for the 2-D case it is $2 N^2$. On the other hand, the size of the element stiffness matrix of the quadratic 4-node element used in the 2-D case is $8 \times 8 = 64$ with only 36 components of the matrix to be saved because of the symmetry. While in the 3-D case using hexahedral 8-node elements, the element stiffness matrix has $24 \times 24 = 576$ components with 300 of them to be saved. Moreover, the larger the system, the greater the number of iterations that are needed. Efficiency, therefore, is a critical factor for a 3-D simulation program.

1.3 State-of-the-Art of FE simulation of forming processes

The FE method was originally developed as a numerical procedure for mechanical analysis. Most attention was paid in the 1960's to the analysis of elastic structures. Yamada [YA68] derived the explicit form of the elastic-plastic constitutive relation for von Mises materials and used the FE method to compute plastic deformation processes. Just a couple of years later many successful computational examples were reported using different constitutive descriptions.

Lee, Kobayashi et al. suggested the rigid-plastic material description and demonstrated that the rigid-plastic FE method can be applied to simulate forming processes efficiently [LE73] [KO73] [LI82] [KO86]. Meanwhile, Argyris [AR78] [AR79] [AR84] and Zienkiewicz [ZI69] [ZI86] also presented successful examples with different formulations.

The constitutive relation has also been discussed in many works. Lee proposed a new method for decomposing the deformation gradient tensor [LE69]. McMeeking [MC75] and Nagtegaal et al. [NA74] [NA81] presented FE formulations based on the elastoplastic description and the updated Lagrangian method.

In recent years simulation of sheet forming processes has been emphasized. This is mainly due to two reasons: Firstly, sheet products are produced in great quantity and with high quality. The dies needed for this, which are often quite complicated, are very expensive. The forming processes have to be well controlled to improve quality and to accelerate production as well as to extend the life of the dies. Therefore, the demand for numerical

simulation is very strong in the sheet forming industry. Secondly, the simulation of sheet forming processes requires a strong scientific background, e.g. in modelling the constitutive law, anisotropy and in the theory of plates and shells and has attracted the attention of a number of research workers. The number of degrees of freedom of the system is usually smaller in the simulation of sheet forming processes and the computation can be easily performed on a workstation or even on a personal computer.

At the same time, extensive literature has appeared presenting the results of the FE simulation of bulk forming processes such as extrusion, wire drawing, plate rolling, forging, heading etc. [PI87] [MO84] [PA82] [NA86] [TO92]. Most calculations, however, are either for plane strain or axisymmetric problems, i.e. for 2-D simulation. They were obviously too simple to handle real applications in the bulk forming industry.

Most existing commercial FE packages are general purpose FE programs. They do not specifically implement the simulation of forming processes. For example, the well-known commercial FE package MARC has a rezoning option, but this only works properly in limited cases. In another commercial FE package, ABAQUS, the remeshing algorithm has not yet been implemented. Therefore, they are not able to simulate bulk forming processes like extrusion.

Because the FE method is still relatively new and closely related to modern computer science, most engineers in the forming industry are not yet familiar with it. It is not an easy task for them to use the general purpose FE packages without a fundamental knowledge of the FE method. This is another reason why its application is limited. A user-friendly special purpose FE package will certainly be welcomed by industry.

1.4 Aim of the dissertation

Accompanying the development of computer science, the FE method has been increasingly used as a powerful numerical tool. The computational costs have been reduced greatly. In spite of the great number of degrees of freedom, it is now feasible to calculate real 3-D forming processes.

Today, the forming industry demands not only computation results which show the feasibility of the simulation, but also exact information and the prediction of failures

occurring during the processes. For example, engineers want to know the shape of the products as well as the distributions of the stress and strain. They want to know the forming forces as well as whether failure occurs inside the workpiece, and so on. On the other hand, the FE package must be able to provide all results in a reasonable computing time, usually limited to a couple of hours. Interactive monitoring of the solution process and extensive graphics output are also required.

It is well known that the FE method is a numerical procedure to solve boundary value or initial value problems. Useful and reliable information is available only when the parameters of the program are properly determined, which is the task of the users. In comparison with a general purpose FE package, a special purpose FE program concentrates on a particular kind of problem. Thus the necessary input data is reduced. This in turn enables users to get started quickly and provides convenient operation. A special purpose package, therefore, is more suitable for industrial application.

The aim of this dissertation is to investigate the arbitrary Lagrangian-Eulerian (ALE) formulation in order to overcome the difficulty caused by strong distortion of the mesh. Based on this formulation, an efficient and reliable special purpose FE package for the simulation of bulk forming processes, especially suitable for the simulation of extrusion processes, was implemented in order to meet industrial needs.

The ALE method has been discussed in its different aspects [HU86] [LI86] [BE89] [PO91] [AK93]. Most examples that have been presented involved 2-D cases. Some formulations lead to very complex formulation, even to an unsymmetrical stiffness matrix.

Based on the discussion of both the Lagrangian and Eulerian methods, an alternative ALE method is proposed. The nodes are defined either as Lagrangian or as Eulerian nodes according to their positions. Iterations are performed as with the normal Lagrangian method and the gradients of the variables are taken into account in the updating of the variables. In order to couple these two kinds of meshes, new elements can be generated or old elements can be deleted.

Forming materials obeying the von Mises material law are usually considered to be

incompressible. Two general methods to deal with this property, namely the Lagrangian multiplier and penalty methods, are compared and discussed.

Because the 3-D simulation of forming processes involves the solution of a very large set of equations, an iterative technique, namely the conjugative gradient method is compared with the direct Gaussian elimination algorithm.

Friction is an important factor in forming processes. It may cause problems if it is not modelled correctly. Measures to stabilize the computation are proposed.

For many non-ferrous alloys, it is very significant to be able to predict the metallurgical properties after forming processes. Because temperature is an important factor, weakly coupled thermo-mechanical analysis is also implemented using the ALE formulation.

The program PressForm, based on the ideas set out in this thesis, has been developed and tested with different forming processes. The results show the validity of the formulation. The results from some test computations are compared either with experiments or with the commercial FE package MARC.

Several examples in the application show, on the one hand, the possibility of simulating forming processes and predicting failures in the processes. On the other hand, they indicate the directions of further development and the prospects of the FE simulation of forming processes.

2. Basic equations

At first in this chapter the basic equations and concepts of continuum mechanics and thermal analysis are briefly discussed.

2.1 Basic equations of continuum mechanics

The essential equations in continuum mechanics are the conservation of mass

$$\dot{\rho} = \frac{d\rho}{dt} = -\rho v_{i,i} ; \quad (2.1)$$

the conservation of momentum

$$\rho \frac{dv_i}{dt} = \sigma_{ij,j} + \rho b_i ; \quad (2.2)$$

and the conservation of energy (first thermodynamical law)

$$\rho \frac{du}{dt} = \sigma_{ij} \varepsilon_{ij} + q_{i,i} . \quad (2.3)$$

In the forming processes considered here the acceleration $\frac{dv_i}{dt}$ is negligible. (2.2) then becomes

$$\sigma_{ij,j} + \rho b_i = 0 . \quad (2.4)$$

Furthermore, the body forces b_i can also be assumed to be negligibly small. Therefore, the equilibrium equation

$$\sigma_{ij,j} = 0 \quad (2.5)$$

is obtained. All quantities introduced in the above equations are defined in the Notation, page 121.

2.2 Different strain and stress tensors

Because forming processes involve large deformations, differences between various definitions of the strain and stress tensors are not negligible. A brief discussion of several definitions of strain and stress is therefore necessary.

2.2.1 Strains, strain rates and strain increments

When a body moves from a configuration R to another configuration r a material point goes from X to x . Monotropic transformation $\frac{\partial x}{\partial X}$ must exist in the continuum. The deformation gradient is defined as

$$F = \frac{\partial x}{\partial X} \quad \text{or using the index notation:} \quad F_{ij} = \frac{\partial x_i}{\partial X_j} \quad (2.6)$$

and different strain tensors can be derived from it.

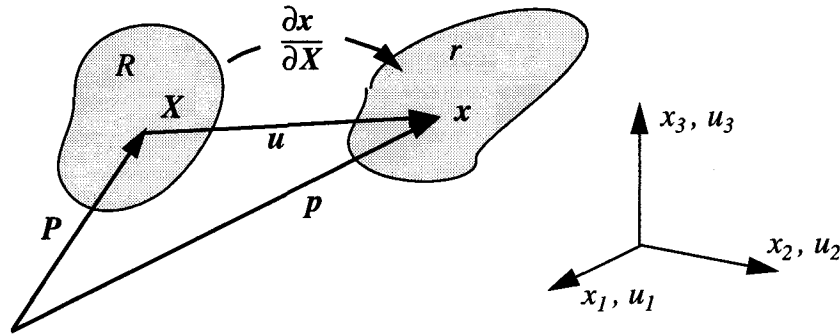


Fig. 2. 1 Transformation of the configurations

Because the so-called engineering strains

$$\epsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} \right) \quad (2.7)$$

are no longer suitable for the description of finite deformations, especially when the rotations are large, different strain tensors are defined to describe deformation. The Lagrange strain tensor

$$E_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} + \frac{\partial u_k}{\partial X_i} \frac{\partial u_k}{\partial X_j} \right) \quad (2.8)$$

is widely used in FE computations where the updated or total Lagrangian method is used. Lagrange strain eliminates the influence of rigid rotations. It can, therefore, be used in cases of finite deformation. It is related to the configuration before deformation, which is known in advance. But it involves non-linear operations. This makes the formulation complex and slows down the convergence process. Besides, it is coupled with the 2nd Piola-Kirchhof stress which cannot be substituted directly into (2.2).

Unlike Lagrangian strain tensor, Almansi strain tensor e is defined as

$$e_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{\partial u_k}{\partial x_i} \frac{\partial u_k}{\partial x_j} \right). \quad (2.9)$$

Almansi strain tensor takes the current configuration as a reference. The strains represent the deformation in the deformed state. With the simple shear deformation the difference in the two strains can be clearly explained.

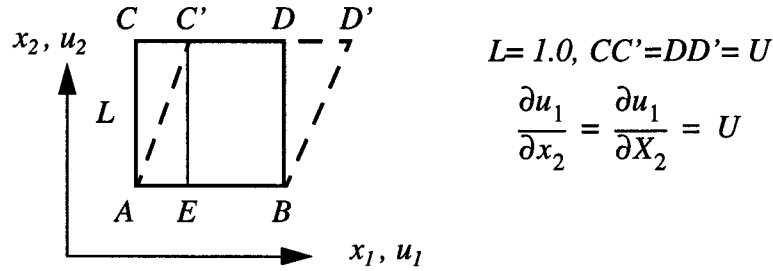


Fig. 2. 2 Simple shear deformation

Fig. 2. 2 shows a simple shear deformation. According to (2.8) and (2.9), the component of Lagrange strain E_{22} is

$$E_{22} = \frac{1}{2} \left(\frac{\partial u_1}{\partial X_2} \right)^2 = \frac{1}{2} U^2 \quad (2.10)$$

while the corresponding component of Almansi strain is

$$e_{22} = -\frac{1}{2} \left(\frac{\partial u_1}{\partial x_2} \right)^2 = -\frac{1}{2} U^2. \quad (2.11)$$

Lagrange strain tensor has a positive E_{22} because line AC is stretched to $A'C'$. No matter whether the deformation is large or small, it is the deformation of line AC . In contrast, Almansi strain uses the line element EC' as reference. This element was EC in the initial state. It is compressed by the deformation. Therefore, the component e_{22} of Almansi strain tensor is negative. At different deformation stages the position EC' is occupied by different material lines. Therefore, the element which is under consideration refers to different lines in the initial configuration.

Both Lagrange and Almansi strain tensors are mainly used to describe finite deformation

in the elastic range. In the subject of plasticity as well as in fluid dynamics, constitutive relations are based on the incremental or rate form. The strain rate

$$\dot{\epsilon}_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \quad (2.12)$$

is an essential quantity.

During a very small time interval the strain increment can be calculated as

$$\Delta \epsilon_{ij} = \dot{\epsilon}_{ij} \Delta t . \quad (2.13)$$

The strain increment can also be expressed as a function of displacement in the time steps

$$\Delta \epsilon_{ij} = \frac{1}{2} (\Delta u_{i,j} + \Delta u_{j,i}) . \quad (2.14)$$

Because the strain increment is not exactly rotationally neutral, some modifications have been suggested.

Hughes [HU84] proposed the rotationally neutral strain increment, which is calculated using the middle configuration as reference

$$\Delta \epsilon_{ij}^{RN} = \frac{1}{2} \left(\frac{\partial \Delta u_i}{\partial \bar{x}_j} + \frac{\partial \Delta u_j}{\partial \bar{x}_i} \right) \quad (2.15)$$

where

$$\bar{x}_i = \frac{1}{2} (X_i + x_i) = X_i + \frac{1}{2} u_i . \quad (2.16)$$

The advantages of this strain increment are summarised by Nagtegaal [NA86]. It is not only rotationally neutral but also a very good approximation of the logarithmic strain increment. It is also easy to evaluate and implement in FE programs.

2.2.2 Cauchy stress and 2nd Piola-Kirchhof stress

Different stress tensors are also defined in continuum mechanics. The most useful are the Cauchy and 2nd Piola-Kirchhof stress tensors.

The Cauchy stress tensor σ_{ij} is defined in terms of forces applied on the current area.

Because the equilibrium is reached at deformed states, the Cauchy stress is the proper measure to be used in the momentum equation (2.5).

The rate of deformation work is obviously

$$\dot{w} = \sigma_{ij} \dot{\epsilon}_{ij} \quad (2.17)$$

If the body undergoes rigid rotations the Cauchy stress should vary accordingly.

Because the Lagrangian strains are defined with respect to the undeformed configuration and measures the strain in the sense of square length, they do not change if an additional rigid rotation takes place. But the Cauchy stress varies due to the change of the orientation. Therefore, the Lagrangian strain cannot be coupled with the Cauchy stress.

The 2nd Piola-Kirchhof stress

$$S_{ij} = \frac{\rho_0}{\rho} F_{im}^{-1} \sigma_{mn} F_{jn}^{-1} \quad (2.18)$$

serves here as the conjugate stress measure. The 2nd Piola-Kirchhof stress tensor S_{ij} is an objective tensor because it does not change when additional rigid rotations are applied to the body. But it cannot be directly used in the equilibrium equation. Substitutions make the form very complex. Moreover, material laws are generally based on Cauchy stresses. An example is the yield locus Y . It is generally accepted that the yield state is determined by the Cauchy stress as

$$Y = Y(\sigma_{ij}, \sigma_Y) = 0 \quad (2.19)$$

where σ_Y is the yield stress. and the plastic strain increments are proportional to the gradient of the yield locus like

$$\Delta \epsilon_{ij}^p = d\lambda \frac{\partial F}{\partial \sigma_{ij}} \quad (2.20)$$

If the 2nd Piola-Kirchhof stress is substituted, the expression becomes

$$\hat{Y} = \hat{Y}(S_{ij}, F_{s^p}, \sigma_Y) = 0 \quad (2.21)$$

Not only the 2nd Piola-Kirchhof stress but also the deformations are introduced into the

yield criterion, which is clearly not favourable.

2.3 Elastoplastic and rigid-plastic descriptions

In the plastic range there are no monotropic relations between stress and strain. The relation is not only nonlinear but also history dependent. When the elasto-plastic description is used, only the relation between stress increment and strain increment is available.

In the theory of plasticity the equivalent plastic strain increment $\Delta \epsilon^p_v$ and equivalent stress σ_v are defined as

$$\Delta \epsilon^p_v = \sqrt{\frac{2}{3} \Delta \epsilon^p_{ij} \Delta \epsilon^p_{ij}} \quad (2.22)$$

and

$$\sigma_v = \sqrt{\frac{3}{2} \sigma'_{ij} \sigma'_{ij}} \quad (2.23)$$

respectively.

According to the von Mises yield criterion plastic deformations appear when the condition

$$\sigma_v = \sqrt{\frac{3}{2} \sigma'_{ij} \sigma'_{ij}} = \sigma_Y \quad (2.24)$$

is fulfilled. Here σ_Y is the yield stress of the material and can be determined by the hardening curve.

In order to derive the elasto-plastic description conveniently, the strain increments and stresses are written vectorially, for example

$$\Delta \boldsymbol{\epsilon} = \left[\Delta \epsilon_{xx} \quad \Delta \epsilon_{yy} \quad \Delta \epsilon_{zz} \quad \sqrt{2} \Delta \epsilon_{xy} \quad \sqrt{2} \Delta \epsilon_{yz} \quad \sqrt{2} \Delta \epsilon_{zx} \right]^T, \quad (2.25)$$

where the $\sqrt{2}$ is used to make the equivalence

$$\Delta \boldsymbol{\epsilon}^T \Delta \boldsymbol{\epsilon} = \Delta \epsilon_{ij} \Delta \epsilon_{ij}. \quad (2.26)$$

The strain increment can be decomposed as the sum of elastic and plastic parts

$$\Delta \boldsymbol{\varepsilon} = \Delta \boldsymbol{\varepsilon}^p + \Delta \boldsymbol{\varepsilon}^e . \quad (2.27)$$

It is generally accepted that Hooke's law holds for the elastic strain and stress increments, so it can be written as

$$\Delta \boldsymbol{\sigma} = \mathbf{D} \Delta \boldsymbol{\varepsilon}^e = \mathbf{D} (\Delta \boldsymbol{\varepsilon} - \Delta \boldsymbol{\varepsilon}^p) . \quad (2.28)$$

where \mathbf{D} represents the elasticity matrix. According to the associated flow rule for von Mises materials

$$\Delta \boldsymbol{\varepsilon}^p = d\lambda \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} , \quad (2.29)$$

the plastic equivalent strain increment is then

$$\Delta \varepsilon_v = \sqrt{\frac{2}{3} (\Delta \boldsymbol{\varepsilon}^p)^T \Delta \boldsymbol{\varepsilon}^p} = d\lambda \sqrt{\frac{2}{3} \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}}} . \quad (2.30)$$

If the hardening curve is given by $\sigma_Y = H(\varepsilon_v)$, the material exhibits plastic deformation when the stresses σ_{ij} fulfil the yield condition $\sigma_v = H(\varepsilon_v)$. Considering work hardening, the following equation is obtained

$$\Delta \sigma_v = H \Delta \varepsilon_v = \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T \Delta \boldsymbol{\sigma} = \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T \mathbf{D} \left(\Delta \boldsymbol{\varepsilon} - d\lambda \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right) . \quad (2.31)$$

Substituting (2.30) into the above equation, it becomes

$$H \left(d\lambda \sqrt{\frac{2}{3} \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}}} \right) = \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T \mathbf{D} \left(\Delta \boldsymbol{\varepsilon} - d\lambda \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right) . \quad (2.32)$$

Solving for the scalar factor $d\lambda$

$$d\lambda = \frac{\left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T \mathbf{D} \Delta \boldsymbol{\varepsilon}}{H \sqrt{\frac{2}{3} \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}}} + \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T \mathbf{D} \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}}} \quad (2.33)$$

and substituting (2.33) and (2.29) into (2.28), the relation is finally expressed as

$$\Delta \boldsymbol{\sigma} = \left(\begin{array}{c} D \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T D \\ D - \frac{D \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T D}{H \sqrt{\frac{2}{3}} \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} + \left(\frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}} \right)^T D \frac{\partial \sigma_v}{\partial \boldsymbol{\sigma}}} \end{array} \right) \Delta \boldsymbol{\varepsilon} = D^{ep} \Delta \boldsymbol{\varepsilon} \quad (2.34)$$

According to (2.23), it is easy to calculate the term

$$\frac{\partial \sigma_v}{\partial \sigma_{ij}} = \frac{3 \sigma'_{ij}}{2 \sigma_v} \quad (2.35)$$

(2.34) can be consequently implemented in FE program.

Two problems arise when the elastoplastic description is used in FE programs. Firstly, because the components of the elastoplastic matrix D^{ep} are very large, the strain increments have to be very small to keep the calculation stable. Therefore, many increments are needed to compute forming processes.

Secondly, as mentioned before, there should be no strain increment if the body undergoes a rigid body rotation. But in this case the Cauchy stresses change as the orientation of the body changes. In other words, the Cauchy stress increments are not independent of rotation. In order to solve the problem the Jaumann rate of the stress

$$\dot{\sigma}_{ij}^J = \dot{\sigma}_{ij} + W_{ik} \sigma_{kj} - \sigma_{ik} W_{kj} \quad (2.36)$$

is introduced into the constitutive relation, where W_{ij} denotes the spin tensor. The Jaumann rate is independent of rotation. Although the Cauchy stress rate can then be evaluated and updated, more computation is necessary and numerical errors are inevitable.

Unlike the elastoplastic method, the rigid-plastic description ignores elastic strains ε_{ij}^e and connects directly the strain increments with the deviatoric stresses

$$\sigma'_{ij} = \frac{2 \sigma_v}{3 \Delta \varepsilon_v} \Delta \varepsilon_{ij} \quad (2.37)$$

The strain-stress curve which describes the hardening history is obtained from simple

experiments such as tensile, compression or bulge tests.

It is clear that the rigid-plastic description is much simpler than the elastoplastic one. It is also more stable for large increments. Practice shows that large strain increments of up to 1% do not present any problem. In contrast, the strain increments have to be kept under 0.1% in the elastoplastic computation. Furthermore, stress updating is no longer necessary, so that computational errors are reduced.

There are also disadvantages in the rigid-plastic description. First of all, according to von Mises plasticity conditions, the hydrostatic pressure is assumed to have no influence on plastic deformations. Therefore, only deviatoric stress instead of total stress can be calculated from the strain increment. In order to get the total stress special measures must be taken. Secondly, it cannot handle elastic loading or unloading. Moreover, in the zones where the body moves without deforming, i.e. as a rigid body, the rigid-plastic description fails as $\Delta\epsilon_v = 0$.

However, elastic loading and unloading are not important for most cases of bulk forming processes. The hydrostatic pressure can also be introduced into the computation by considering the incompressibility condition. Furthermore, rigid zones are not of practical interest. Therefore, it was decided to use the rigid-plastic description. Besides the reasons mentioned above, the only variable whose history has to be followed is the equivalent strain ϵ_v . It is specially suitable for the ALE formulation because only the convective term of equivalent strain has to be evaluated.

The rotation-neutral strain increment and Cauchy stress are adopted in the present work for the rigid-plastic description.

2.4 Thermal analysis

The rate of dissipation Φ is defined in thermodynamics as

$$\Phi = \rho T \dot{s} + q_{i,i} - \frac{T_{,i} q_i}{T} \quad (2.38)$$

where s is the entropy, q denotes the heat flux and T represents the temperature.

Combining with the first law of thermodynamics, it is found that

$$\Phi = \rho (T\dot{s} - \dot{u}) + \sigma_{ij}\dot{\epsilon}_{ij} - \frac{T_{,i}q_i}{T} \quad (2.39)$$

where u denotes the internal energy.

The second law of thermodynamics states that

$$\Phi \geq 0 \quad (2.40)$$

which is equivalent to

$$\rho (T\dot{s} - \dot{u}) + \sigma_{ij}\dot{\epsilon}_{ij} - \frac{T_{,i}q_i}{T} \geq 0 \quad (2.41)$$

Experiments show that more than 90% of plastic work is transformed into heat energy while less than 10% is stored in the dislocations produced as well as on the surfaces of the grains. It is also generally a reasonable assumption that the density ρ of the material does not change in forming processes. Besides, in the range of forming temperature the internal energy of the forming materials can be expressed as

$$\dot{u} = C_\rho \dot{T} \quad (2.42)$$

where C_ρ represents the specific heat of the material per unit volume.

In order to simplify the computation and not to involve the state variable entropy, the first law of thermodynamics is used. The plastic work is treated as a source of heat flow and the equation for heat conduction

$$q_i = \frac{\partial}{\partial x_i} (\lambda T) \quad (2.43)$$

is introduced, where λ is the coefficient of heat conductivity.

Accordingly another form of (2.3)

$$C_\rho \dot{T} = a \sigma_{ij} \dot{\epsilon}_{ij}^P + q_{i,i} \quad (2.44)$$

is solved. In above equation a is a dimensionless coefficient to determine the portion of plastic work which is transformed to heat energy. It should be determined by experiments.

3. Arbitrary Lagrangian-Eulerian Formulation

The kinematic relations of the arbitrary Lagrangian-Eulerian (ALE) method are established based on the discussion of two traditional methods in continuum mechanics.

3.1 Two different methods of observation

When the deformations are finite the difference between the deformed configuration and the initial configuration is not negligible. In order to describe the deformation as well as various state variables two methods have been introduced in continuum mechanics, namely the Lagrangian and Eulerian methods.

3.1.1 Lagrangian method (material description)

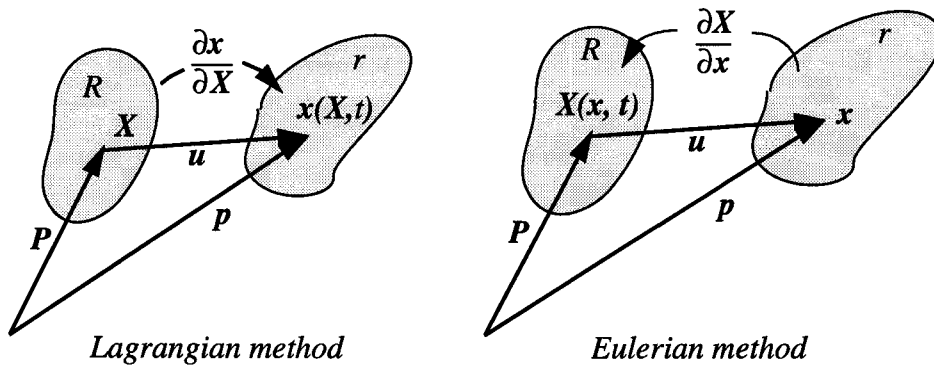


Fig. 3.1 Transformation of the configurations

The Lagrangian method uses the undeformed state as a reference configuration. The momentary configuration is expressed as

$$\mathbf{x} = \mathbf{x}(\mathbf{X}, t) \quad . \quad (3.1)$$

The velocity of any material particle which occupied position \mathbf{X} in the initial configuration R is

$$\mathbf{v} = \dot{\mathbf{x}} = \frac{\partial \mathbf{x}}{\partial t} \quad . \quad (3.2)$$

The acceleration is then derived as

$$\mathbf{a} = \dot{\mathbf{v}} = \ddot{\mathbf{x}} \quad . \quad (3.3)$$

Because \mathbf{X} always refers the same material particle, it is called the material description. The derivative with respect to time for the material description is defined as a material derivative.

An infinitesimal element $d\mathbf{x}$ in the current configuration is expressed as

$$dx_i = \frac{\partial x_i}{\partial X_j} dX_j = F_{ij} dX_j \quad (3.4)$$

The difference of the square length of an element before and after deformation is

$$ds^2 - dS^2 = dx_k dx_k - dX_j dX_j = (F_{ki} F_{kj} - \delta_{ij}) dX_i dX_j \quad (3.5)$$

Since $x_i = X_i + u_i$, the deformation gradient F_{ij} is expressed as

$$F_{ij} = \frac{\partial x_i}{\partial X_j} = \delta_{ij} + \frac{\partial u_i}{\partial X_j} \quad (3.6)$$

and therefore

$$ds^2 - dS^2 = 2E_{ij} dX_i dX_j \quad (3.7)$$

The physical explanation of the Lagrange strain is then seen.

If any variable ϕ , whether scalar, vector or tensor, is expressed using the Lagrangian method as

$$\phi = \phi(X, Y, Z, t) \quad (3.8)$$

the material derivative is simply

$$\dot{\phi} = \frac{d\phi}{dt}(X, Y, Z, t) \quad (3.9)$$

3.1.2 Eulerian method (spatial description)

Another way to describe continua is to express the initial configuration as a function of the current position

$$\mathbf{X} = \mathbf{X}(\mathbf{x}, t) \quad (3.10)$$

The infinitesimal element in the initial state is then

$$dX_i = \frac{\partial X_i}{\partial x_j} dx_j = F_{ij}^{-1} dx_j \quad (3.11)$$

In the same manner as in the Lagrangian method, but using the current configuration as reference,

$$ds^2 - dS^2 = 2e_{ij} dx_i dx_j \quad (3.12)$$

the Almansi strain e_{ij} is then introduced.

In the Eulerian formulation, a position \mathbf{x} is occupied by different material particles as the material particles move continually. \mathbf{x} works as a reference frame in space. Therefore, the Eulerian method is also called the spatial description.

When a state variable ϕ is expressed using the Eulerian method as

$$\phi = \phi(x, y, z, t) \quad (3.13)$$

The material derivative of ϕ with respect to time is

$$\dot{\phi} = \frac{\partial \phi}{\partial t} + \frac{\partial \phi}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial \phi}{\partial y} \frac{\partial y}{\partial t} + \frac{\partial \phi}{\partial z} \frac{\partial z}{\partial t} = \frac{\partial \phi}{\partial t} + v_i \phi_{,i} \quad (3.14)$$

The second term in the above equation is called the convective term. It can be explained using Fig. 3. 2.

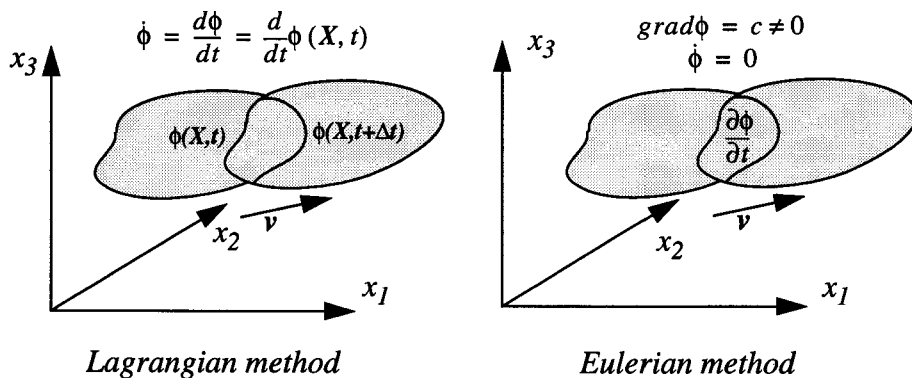


Fig. 3. 2 Schematic explanation of the convective effect

Suppose a constant gradient of variable ϕ exists in the body and ϕ is kept unchanged as the body moves. In the position \mathbf{x} a change of the variable $\frac{\partial\phi}{\partial t} = -v_i\phi_{,i}$ is observed. This change is caused by the flow of the material and the gradient inside the medium. It is clear that the derivative $\frac{\partial\phi}{\partial t}$ alone makes no sense because it is not connected to certain material particles. Only the material derivative is related to the deformation history.

3.2 Comparison of the two methods

From the discussions in the previous section, a comparison of these two methods is made in Table 3. 1, where two points are of special interest. One is how to evaluate the deformation, while the other is how to describe and update the state variables.

Table 3. 1 : The comparison of Lagrangian and Eulerian methods

	Lagrangian method	Eulerian method
Kinematic description	$\mathbf{x} = \mathbf{x}(\mathbf{X}, t)$ $\mathbf{v} = \dot{\mathbf{x}}(\mathbf{X}, t)$ $\mathbf{a} = \frac{d}{dt}\mathbf{v}(\mathbf{X}, t)$	$\mathbf{X} = \mathbf{X}(\mathbf{x}, t)$ $\mathbf{v} = \dot{\mathbf{X}}(\mathbf{x}, t)$ $\mathbf{a} = \frac{d\mathbf{v}}{dt} = \frac{\partial\mathbf{v}}{\partial t} + \frac{\partial\mathbf{v}}{\partial\mathbf{x}}\mathbf{v}$
Deformation	$F_{ij} = \frac{\partial x_i}{\partial X_j}$	$F_{ij}^{-1} = \frac{\partial X_i}{\partial x_j}$
Strain tensor	<p>Lagrange strain</p> $E_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} + \frac{\partial u_k}{\partial X_i} \frac{\partial u_k}{\partial X_j} \right)$	<p>Almansi strain</p> $e_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{\partial u_k}{\partial x_i} \frac{\partial u_k}{\partial x_j} \right)$
Conjugate stress tensor	<p>Second Piola-Kirchhof stress tensor</p> S_{ij}	<p>Cauchy stress tensor σ_{ij}</p>
Updating algorithm	$\mathbf{x}(\mathbf{X}, t + \Delta t) = \mathbf{x}(\mathbf{X}, t) + \mathbf{v}\Delta t$	$\mathbf{x}^{t+\Delta t} = \mathbf{x}^t$
Material derivative	$\dot{\phi} = \frac{d}{dt}\phi(\mathbf{X}, t)$	$\dot{\phi} = \frac{\partial\phi}{\partial t} + v_i\phi_{,i}$
Conservation of the mass	$\frac{d\rho}{dt} = -\rho v_{i,i}$	$\frac{\partial\rho}{\partial t} = -\rho v_{i,i} - \rho_{,i}v_i$
Conservation of momentum	$\rho \frac{dv_i}{dt} = \sigma_{ij,j} + \rho b_i$	$\rho \frac{\partial v}{\partial t} = \sigma_{ij,j} + \rho b_i - \rho v_{i,j}v_j$

Table 3. 1 : The comparison of Lagrangian and Eulerian methods

	Lagrangian method	Eulerian method
Conservation of energy	$\rho \frac{du}{dt} = \sigma_{ij} \dot{\epsilon}_{ij} + q_{i,i}$	$\rho \frac{\partial u}{\partial t} = \sigma_{ij} \dot{\epsilon}_{ij} + q_{i,i} - u_{,i} v_i$

Note: All variables introduced in this table are defined in the Notation given on page 121.

Firstly, the question of deformations is investigated. Suppose a square element of length L is stretched ΔL in the X direction and rotated at an angle α in the plane as shown in Fig.

3. 3. The configuration after deformation x can be expressed by the initial X

$$\begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \begin{bmatrix} 1 + \epsilon & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} X_1 \\ X_2 \end{Bmatrix} = \begin{bmatrix} (\epsilon + 1) \cos \alpha & -\sin \alpha \\ (\epsilon + 1) \sin \alpha & \cos \alpha \end{bmatrix} \begin{Bmatrix} X_1 \\ X_2 \end{Bmatrix} \quad (3.15)$$

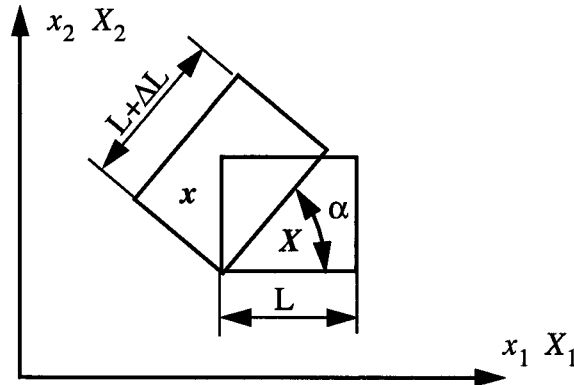


Fig. 3. 3 Deformation of a square element

where $\epsilon = \Delta L / L$ denotes the deformation. The deformation gradient is

$$F = \begin{bmatrix} (\epsilon + 1) \cos \alpha & -\sin \alpha \\ (\epsilon + 1) \sin \alpha & \cos \alpha \end{bmatrix} \quad (3.16)$$

The Lagrangian strains are evaluated as

$$E = \frac{1}{2} \begin{bmatrix} 2\epsilon + \epsilon^2 & 0 \\ 0 & 0 \end{bmatrix} \quad (3.17)$$

It is seen that the Lagrangian strain is independent of the rotation of the element.

When the same deformation is measured using the Almansi strains, these are given by

$$e = \frac{(1 + \varepsilon)^2 - 1}{2(1 + \varepsilon)^2} \begin{bmatrix} \cos \alpha \cos \alpha & -\sin \alpha \cos \alpha \\ -\sin \alpha \cos \alpha & \sin \alpha \sin \alpha \end{bmatrix}. \quad (3.18)$$

It means that the components of the Almansi strain tensor depend on the rotation. The principal directions of the Almansi strain are identical with the main deformation directions in the current configuration, so that the Cauchy stress can be coupled with it.

In addition to the different strain tensors, another difference between the Lagrangian and Eulerian methods is how to describe the change of the state variables of the system. As shown previously in Fig. 3. 2, if the Lagrangian method is used to describe ϕ , we obtain $\dot{\phi} = 0$. But in the Eulerian description $\frac{\partial \phi}{\partial t} = -\phi_{,i} v_i$ is obtained. Generally speaking, if the history of the variables should be followed, the updating must be in accordance with the material derivatives. Because the Lagrangian method uses the material description, the deformation history of the material is followed implicitly. In this sense, the Lagrangian method is superior.

However, because of the strongly nonlinear properties of problems in plasticity, computations are performed incrementally. Instead of the classical Total Lagrangian method, the Updated Lagrangian method is widely used in the computational plasticity. It uses the updated configuration X^t as reference and searches for the new equilibrium state $x^{t+\Delta t}$. Because the mesh is updated after each increment, the history of deformation has been followed. When the displacement in an increment is small enough, the difference between the updated configuration X^t and the new equilibrium state $x^{t+\Delta t}$ vanishes. In this sense, the rotationally neutral strain increment is a mixed Euler-Lagrangian method to evaluate the strain increments.

When these two methods are adopted to describe the kinematic condition and constitutive relation, they refer mainly to different ways of defining the strain and stress tensors. When they are applied in the algorithms of the FE method, more attention is paid to the aspect of how to describe the variation of the state variables and how to update the FE mesh.

The Lagrangian method, whether total or updated, uses a material related mesh. Every node in the FE mesh has a corresponding material point. When the material point moves,

the node must follow it and moves to the same position. Therefore, the movements of the domain are well simulated by the FE mesh and a deformation history-dependent constitutive relation is conveniently established.

But the updating causes serious problems when it is used to simulate forming processes like extrusion because of the very large deformations. Rezoning algorithms can partly solve this problem. But existing algorithms are still unable to handle general 3-D cases.

The Eulerian method in FEM involves using a stationary mesh which is independent of material movement. The computational results are the velocities at the nodes and the deformation rates in the elements. The velocities at the nodes are also the velocities of the material particles which currently occupy the position of the nodes. The computational mesh keeps its original state since it is not updated. But because the elements are occupied by different parts of material, the deformation history of the material is not directly included. The convective terms have to be taken into account to evaluate the material derivatives.

Beside the difficulty of following the deformation history, Eulerian elements also have disadvantages when they are used to handle problems with moving boundary conditions. For example, in forming processes the tools move against the workpiece to force it into the desired shape. The boundary conditions on the contact surfaces must be updated to give a proper description of the effective domain. Fig. 3. 4 shows this case schematically. The nodes which are in contact with the tool must be updated at least in the normal direction to avoid penetration into the tool. The purely Eulerian method cannot handle these situations.

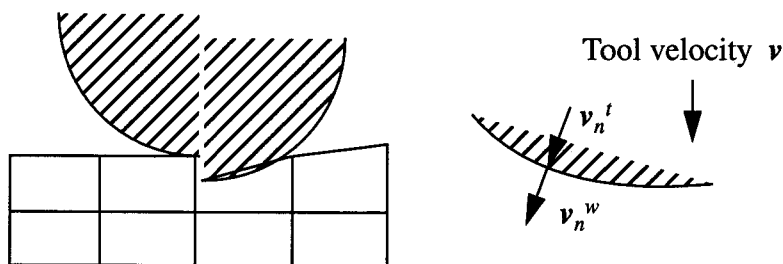


Fig. 3. 4 The moving boundary conditions

Generally speaking, the Eulerian method is suitable for two cases. One is the stationary process. In this case, the history-dependent variables may be integrated along the

streamlines, although this represents a complex task. The other case is where the materials are history-independent, as in fluid mechanics.

3.3 Arbitrary Lagrangian-Eulerian description

The arbitrary Lagrangian-Eulerian (ALE) method has been developed to utilise the advantages of both methods and to remedy their deficiencies. The basic idea of the ALE method is to uncouple the updating of the mesh nodes from the material flow in order to avoid strong distortion. Compared with the Eulerian description ALE provides more flexibility.

For a description of the kinematic state various measures of velocities are introduced as shown in Fig. 3.5

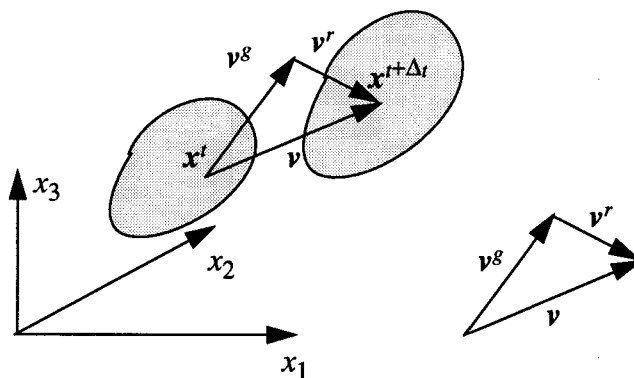


Fig. 3.5 Different definitions of velocity

where

- v : velocity of the material points in global system,
- v^g : arbitrary velocity of the mesh nodes,
- v^r : relative velocity of the material points and mesh nodes.

When the velocities of the mesh nodes are set as

$$v^g \equiv 0, \tag{3.19}$$

the relative velocities

$$v^r = v, \tag{3.20}$$

and the Eulerian method is used.

When the velocities of the mesh nodes are set as

$$v^g = v, \tag{3.21}$$

the relative velocities

$$v^r = 0, \tag{3.22}$$

and the Lagrangian method is used. Therefore, the Lagrangian and Eulerian methods are the two special cases of the ALE formulation.

The ALE method is explained schematically in Fig. 3. 6.

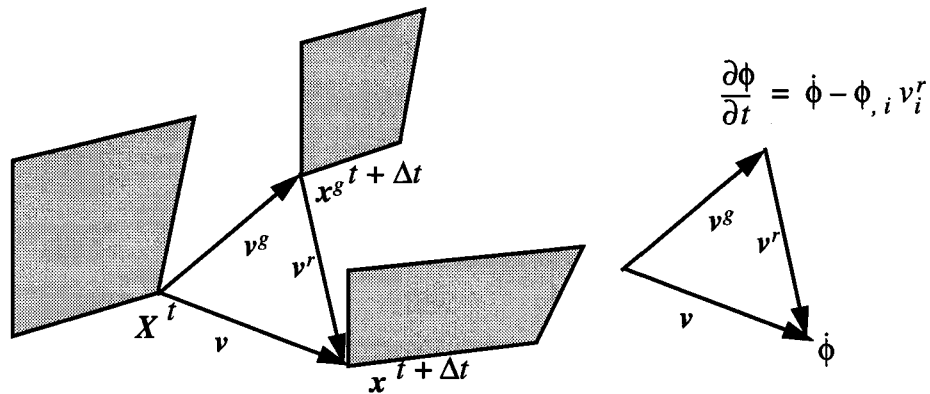


Fig. 3. 6 Scheme of the ALE formulation

Suppose a material point X at time t moves to the new position x in the time interval Δt . But the mesh node which coincides the material point at time t is moved to the position x^g . The rate of a state variable ϕ which is related to the material particle is $\dot{\phi}$. The question is how to evaluate the change of the variable at x^g .

There are two ways of implementing the ALE equations [BE89]. One is to use v^g as an independent variable and integrate the equations in the configuration x^g . This method leads to a very tedious derivation. Liu et al. discussed the implicit formulation, but their implementation was explicit [LI88] [BE89].

Another method, however, was found to be quite simple to understand and easy to implement. The main idea is to split up the analysis into two steps. In the first step the normal Lagrangian computation is performed incrementally. After the new equilibrium state is reached, the mesh nodes are updated in a certain way but not necessarily to follow the material flow. All state variables are then transformed from the results obtained in the first step to the new mesh by considering the convective effect.

The procedure can also be explained in another way. The mesh node follows the material point at first to the position \mathbf{x} and then is adjusted to \mathbf{x}^g using the velocity $-\mathbf{v}^r$. In the first step the rate of ϕ at the node is identically $\dot{\phi}$, while during the second step the rate is $-\phi_{,i} v_i^r$. Therefore, we obtain the expression

$$\frac{\partial \phi}{\partial t} = \dot{\phi} - \phi_{,i} v_i^r \quad (3.23)$$

The equation is valid for general cases, no matter if ϕ is a scalar, vector or tensor.

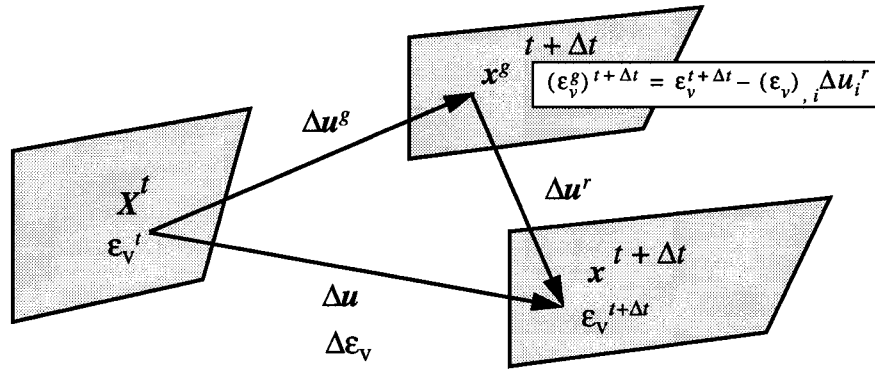


Fig. 3. 7 The computational steps of the ALE method

The second method is adopted in this work. This is not only due to its simplicity but also to the constitutive relation used, since the rigid-plastic relation uses only the equivalent strain ϵ_v to describe the deformation history. Therefore, only little computation is needed to evaluate the convection, which leads to significant savings in computational costs.

The procedure is explained in Fig. 3. 7. After iterating according to the Lagrangian method, a material point X at time t moves to the new equilibrium position \mathbf{x} at time $t + \Delta t$. A strain increment is also obtained at this point. But the mesh has to be updated in a way independent of the material flow. Therefore, the corresponding point in the updated mesh is \mathbf{x}^g instead of \mathbf{x} . The strain ϵ_v in this position is then evaluated as

$$(\epsilon_v^g)^{t+\Delta t} = \epsilon_v^{t+\Delta t} - (\epsilon_v)_{,i} \Delta u_i^r \quad (3.24)$$

It should be noticed that not only strain but also other variables like temperature can be calculated in this way.

3.4 Computational steps in weakly coupled analysis

The computations needed to determine the equilibrium state and the distribution of temperature at each time step are performed separately in the weakly coupled thermo-mechanical analysis. The new state of the domain is found with the temperature at time $t = t_0$. Then the field of temperature is calculated using the plastic dissipation work as the heat sources inside the domain.

The computational steps are illustrated in Fig. 3. 8. Both mechanical analysis and thermal computation are subdivided into two steps. Steps 1 and 3 involve calculating of the change of variables caused by convection. They should be carried out after the convergent solution has been obtained from the last increment. The results are saved when the mesh is updated according to the ALE description. When the convective effects are taken into account the iterations for the next increment can be performed using the general FE formulation.

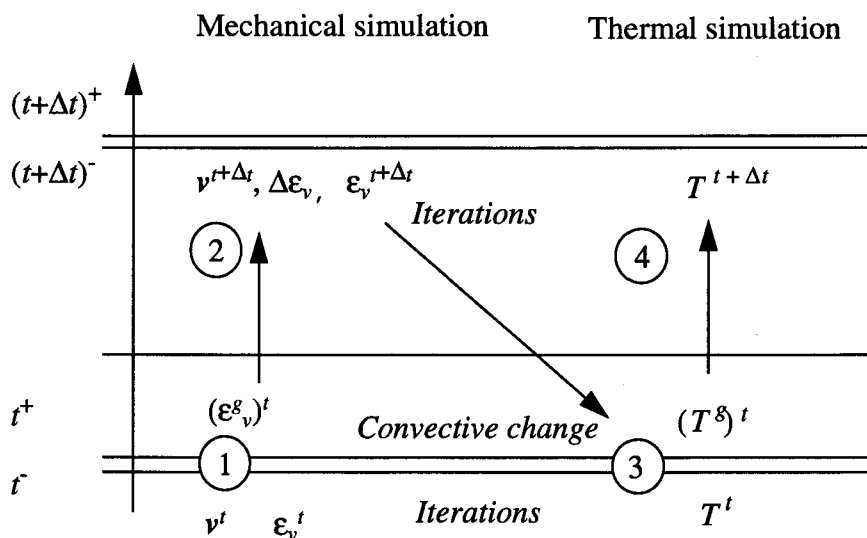


Fig. 3. 8 The computational steps of weakly coupled analysis using the ALE method

If an elastoplastic constitutive relation were used, all the stress components would have to be saved. The convective terms of the stress tensor would have to be calculated instead of only the equivalent strain. The formulation as well as the computation becomes complex. Therefore, the rigid-plastic method was found to be superior, especially in the ALE formulation.

3.5 Computation of the convective term

In contrast to the general FE formulation the ALE method introduces a convective term. The procedure used to calculate the convective terms have been discussed in detail [HU82] [PO91] [AK93]. The main problem is to keep the procedure stable.

Generally, the FE method uses the displacements or velocities at the nodes as system variables. The integration is performed numerically in the Gaussian points of the elements. Therefore, the results are related either to the nodes or to the integration points.

The equivalent strain, which is essential for the rigid-plastic description, is calculated in the integration points. If the values are directly used to calculate the gradient, the procedure is not stable. Huetink suggested using the nodal values, i.e. the mean values of the variables, (see Fig. 3. 9), which involve information from more than one element. The numerical instability is reduced and better results are obtained.

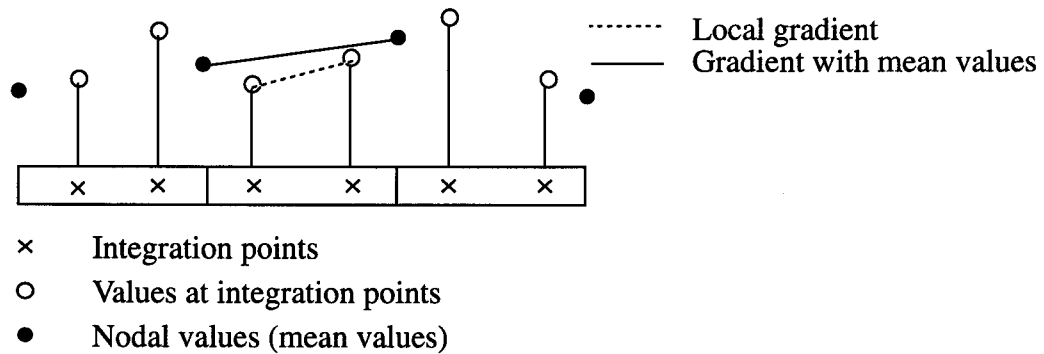


Fig. 3. 9 Different methods of calculating the gradient

The nodal values are calculated as mean values of the neighbouring elements

$$(\epsilon_v^n)^g = \frac{1}{N} \sum_{e=1}^N (\epsilon_v^n)^e, \quad (3.25)$$

where N is the number of elements connected with this node and $(\epsilon_v^n)^e$ is the nodal value at the element level.

Different methods of evaluating the nodal values, such as the FTC, upwind and Lax-Wendroff schemes have been discussed and compared in the one dimensional case by Akkerman [AK93]. However, there are other factors which are important in the

extrapolation. For example, the sizes and the orientations of the elements influence the nodal values. It seems neither possible nor necessary to include all the factors and give a universal comparison of different methods.

When standard interpolating functions N are used, the equivalent strain in the element can be expressed by nodal values through the interpolating function

$$(\boldsymbol{\epsilon}_v^i)^e = N (\boldsymbol{\epsilon}_v^n)^e \quad (3.26)$$

When the number of nodes is the same as the number of integration points, for example, as in the case of 8-node hexahedral elements, the nodal values at the element level can be calculated from the values at the integration points by the inverse operation as

$$(\boldsymbol{\epsilon}_v^n)^e = N^{-1} (\boldsymbol{\epsilon}_v^i)^e \quad (3.27)$$

But this procedure, due to the property of the shape function, is not very stable. The bilinear function in 2-D has the form shown in Fig. 3. 10. It sometimes produces wrong results. It is more serious for the 3-D case because the trilinear function increases even more quickly in the diagonal direction. Therefore, another method was investigated and is used in this work. The mean value of the element is first evaluated, and the value at the node is directly calculated as a linear function of the mean value and the value at the nearest integration point, as shown in Fig. 3. 10.

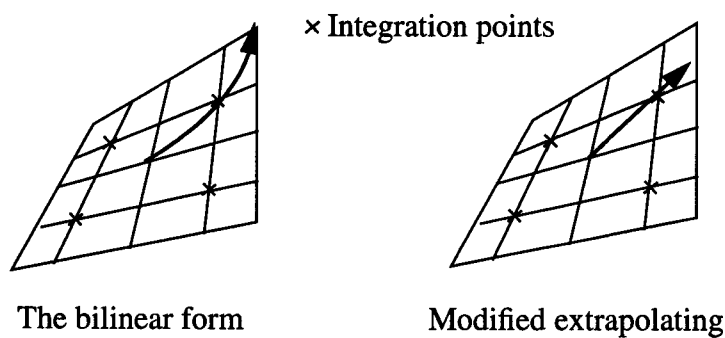


Fig. 3. 10 Different methods of computing the nodal values

The mean value of the element $(\bar{\boldsymbol{\epsilon}}_v)^e$ is evaluated by all values at the integration points as

$$(\bar{\epsilon}_v)^e = \frac{1}{N} \sum_{i=1}^N (\epsilon_v^i)^e \quad (3.28)$$

where N denotes the number of integration points. The n -th nodal value at the element level is calculated as

$$(\epsilon_v^n)^e = (\bar{\epsilon}_v)^e + \frac{1}{|\xi^i|} ((\epsilon_v^i)^e - (\bar{\epsilon}_v)^e) \quad (3.29)$$

where the superscript i is the i -th integration point nearest to the node and ξ^i are the nature coordinates of the integration points.

Stability is obtained by using this method, whose implementation is quite straightforward.

Therefore, the computation of the convective term is performed in two steps. In the first step equation (3.25) is used to calculate the nodal values by gathering the values from the neighbouring elements and the convective term is then calculated using (3.24) at the element level. Here the mean values at the nodes are used. The steps are shown in Fig. 3.11.

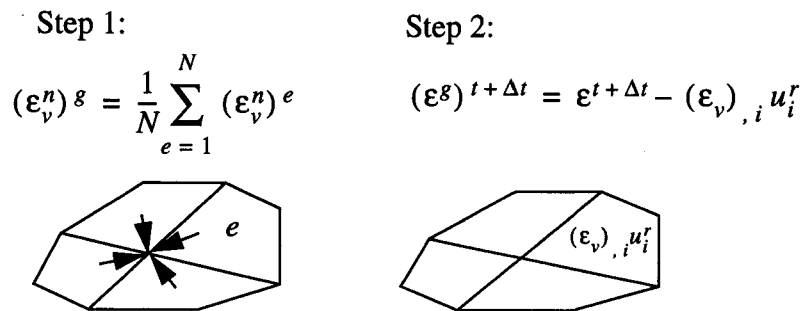


Fig. 3.11 The computation of convection

The Eulerian forward method of numerical integration is used in the computation of convection, i.e. the values at the beginning of the increment are used to evaluate the gradient.

Although the Eulerian forward method is known to be only conditionally stable, it is quite simple and delivers in most cases good results. Two measures have been taken developed to reduce the instability. Firstly, the size of the time increment is controlled such that the maximum displacement in each increment does not exceed the minimum side length of the smallest element. This prevents the local gradients being applied too far. Secondly,

limiting values have been given to the convective term. For example, the equivalent strain ϵ_v may not be less than zero. If the gradient is very high resulting in a negative value, as shown in Fig. 3. 12, ϵ_v is set to a prescribed minimum value.

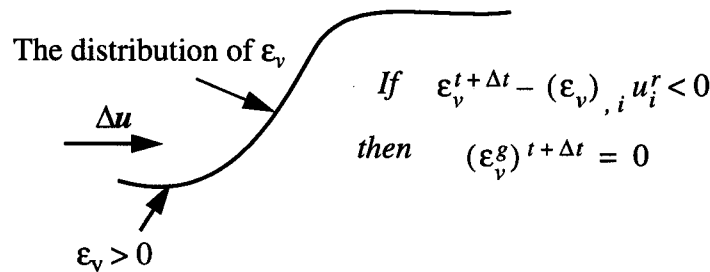


Fig. 3. 12 The correction of the convective results

Actually, little difficulty has been met in the computation of the magnitude of the convective term in the simulation of forming processes. This can be explained as follows: In processes like extrusion the deformations are so large that they have to be performed at high temperatures. In such cases the deformations are accompanied by recrystallisation. The materials exhibit a rate-dependent behaviour like a viscous fluid. The equivalent strain is no longer the important variable. In other cases such as cold forging the deformations as well as the gradients of the deformations are not high enough to cause numerical problems.

As mentioned above, the aim of this dissertation is to develop a user-friendly special-purpose FE package for use in industry to simulate bulk forming processes. Therefore, the optimization of the computation of convection is obviously beyond its scope.

4. Mesh Discretization and Remeshing

As discussed in chapter 3, the basic idea of the arbitrary Lagrangian-Eulerian method is to separate the updating of the mesh nodes from the movement of the material particles in order to reduce the mesh distortion. The updating method is rather flexible. Different updating methods as well as the adjustment of the mesh are discussed in this chapter.

A simple but effective mesh generator has been implemented for the simulation of forward extrusion processes. The method is also discussed in this chapter.

4.1 Different updating and remeshing methods

Generally speaking, as long as the prescribed boundary conditions are not violated, the updating of the nodes is quite flexible. However, a suitable updating method avoids strong mesh distortions, thus giving better results and saving computation time. It should also be easy to implement it into the FE code. According to the characteristics of different forming processes several updating methods have been specified.

Although the arbitrary Lagrangian-Eulerian formulation keeps the greater part of the mesh unchanged during the step-by-step computation, local distortion of elements cannot be entirely eliminated. Due to different updating methods the method of remeshing or regeneration of the mesh has to be handled.

4.1.1 Introduction

The kinematic relation of the ALE method is described in chapter 3. The convective effect due to the relative velocity is

$$\frac{\partial \phi}{\partial t} = \dot{\phi} - (v_i - v_i^g) \phi_{,i} \quad (4.1)$$

Suppose the mesh velocity v^g can be expressed by the material velocity through a prescribed linear transformation as

$$v^g = A v \quad (4.2)$$

then v^g is no longer an independent variable. If we prescribe the relation in this way, nine coefficients for each node have to be calculated and saved.

As mentioned before, the necessary condition for the velocity of mesh nodes on the

surface is

$$v_i^g n_i = v_i n_i \quad , \quad (4.3)$$

where n_i denotes the normal vector of the surface at the position of the node. The velocities of the nodes which do not lie on the surface are theoretically arbitrary.

Although it is not sufficient to determine v^g using (4.3) as shown in Fig. 4. 1, we can let v^g have the same direction as the normal vector, i. e.

$$v^g = a n \quad , \quad (4.4)$$

the coefficient $a = \frac{v_i n_i}{n_i n_i}$ as well as the mesh velocity is then determined.

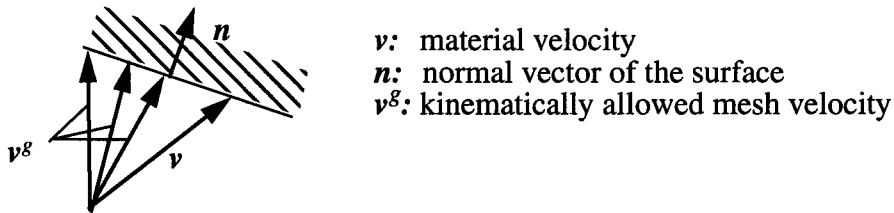


Fig. 4. 1 Allowed mesh velocities

The simplest way is to specify the velocities of mesh nodes v^g to be proportional to the material velocities. The minimum value is zero and the maximum value is one. The former is the Eulerian method and the latter is the Lagrangian description. Only one coefficient has to be saved for each node. With this definition, if the projection of the velocity of a node on the normal vector

$$v^n = v_i n_i \neq 0 \quad (4.5)$$

the node must be totally updated. It is then a Lagrangian node. The other nodes may be arbitrarily defined, for example, as Eulerian nodes.

In practice this method is flexible and simple. The relative velocities as well as the convective terms are also convenient to evaluate.

4.1.2 Combination of Eulerian and Lagrangian meshes

In forming processes such as extrusion, wire drawing and rolling, it is easy to divide the

mesh into two kinds of zones. In the zones where the deformations take place the Eulerian mesh is used. The nodes are not updated but the convective effect is evaluated. No matter how large the deformations are, the mesh remains regular. The difficulty caused by the distortion of the mesh is then eliminated. In the zones where the material flows in or out of the deformation zones, the material undergoes almost rigid body movements or very small deformations. The nodes in these zones are then set to be material related and follow the material flow. Therefore, the deformed shape of the workpiece is well described as by the updated Lagrangian formulation.

Using this method the coordinates of the nodes are either updated or kept unchanged. At the point of transition there are elements with both kinds of nodes.

In these processes the material flow in the Lagrangian zones is mainly in one direction, i.e. the extrusion, drawing or rolling direction. As the updating proceeds the elements with both kinds of node are either stretched or pressed.

At the point where the material flows out the element sides are lengthened. If the length of a side exceeds a given criterion, it is subdivided by adding a node at the middle of the side. The related elements are subdivided to generate new elements.

In contrast, if the sides of the elements which are compressed become shorter than a certain criterion, the elements are combined with the elements in front of them. This procedure works well in the simulation of 3-D simulation of extrusion processes.

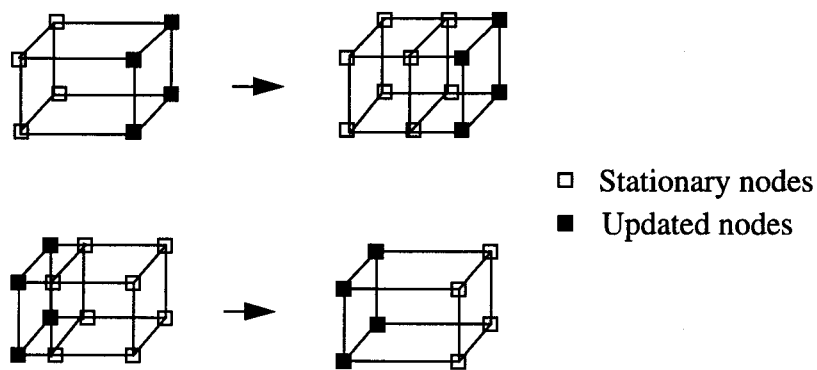


Fig. 4. 2 Different cases for the remeshing operation

Fig. 4. 3 shows the definition by means of an example of the extrusion of a square bar. In wire drawing and rolling processes the mesh can be defined in a similar way.

Because the field magnitudes like velocity and temperature change only slightly between two increments, the equilibrium states can be quickly found if the results from the last increment are taken as initial values for the iterations of the next increment.

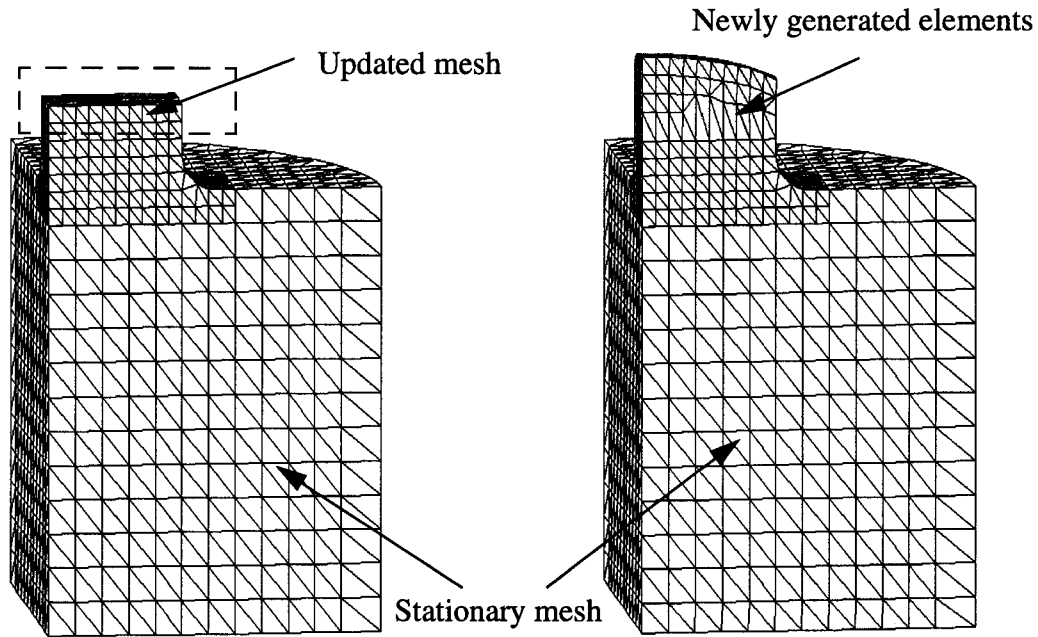


Fig. 4. 3 The updating and remeshing technique for the simulation of forward extrusion processes

4.1.3 Surface and internal nodes

In general cases of forming processes the material flow as well as the form of the workpiece is rather complex. Therefore, it is not possible to divide the zones directly into Lagrangian and Eulerian ones. Other methods of classifying the mesh are needed.

One direct method is to divide the nodes into two groups. One consists of all the surface nodes, the other of the internal nodes.

As long as the surface nodes are updated, the boundary conditions are satisfied and the form of the workpiece is well described. At the same time, the internal nodes can be kept as Eulerian nodes. When this method is used, the greater part of the mesh remains in the initial state.

This method allows the surface elements to change their form as the updating goes on. There are different cases to be handled in order to keep the mesh regular as shown in Fig. 4. 4.

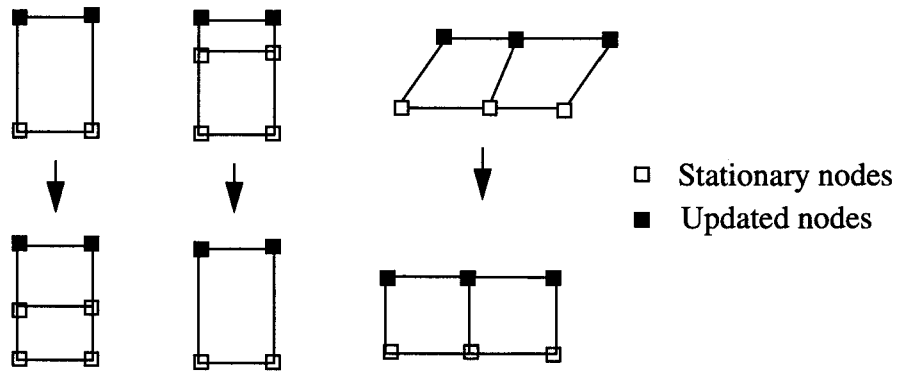


Fig. 4. 4 Three possible cases of the surface element

In the case where the element sides are stretched too much in one direction, the elements are subdivided and their material data is distributed to the new elements.

When the element is compressed so that its sides become too short, the nodes of the sides are combined and the element is removed.

The third case is when the element is distorted. The mesh is adjusted to reduce the distortion.

It is also possible that the sides of an element possess different cases. Degenerated elements are then needed, as shown in Fig. 4. 5.

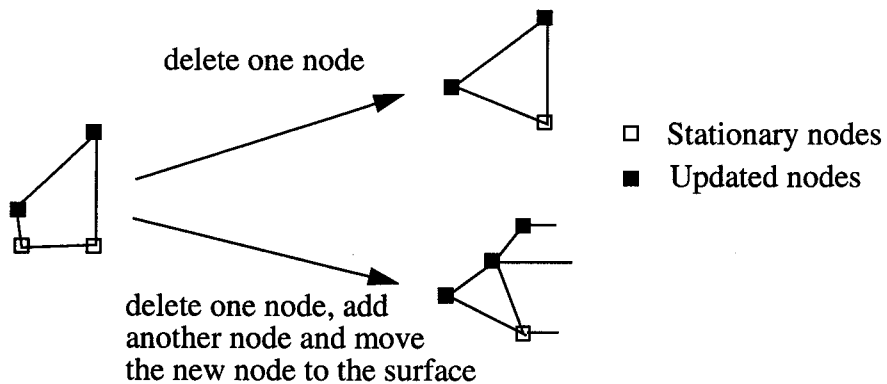


Fig. 4. 5 Degenerated elements

In the actual computer program the internal elements are kept and new elements are generated to fit the surface of the workpiece. The procedure is repeated after every increment. Therefore, the mesh is guaranteed to remain regular.

This method is easily implemented for the 2-D case. Serious difficulties, however,

prevent its extension to 3-D cases.

One problem comes from the filling of edges and corners. The edges and corners have to be filled in order to keep a reasonable form of the workpiece. Because the internal mesh is made up of basic regular elements, if the surface nodes are moved too far to fit the edges and corners, an undesirably distorted mesh is obtained again.

Another difficulty is that many more cases have to be treated in 3-D situations than those shown in Fig. 4. 4. A hexahedral element has 12 sides. Each may get too long or too short or does not exceed the criterion. To obtain a reasonable mesh, the neighbouring elements have to be considered. The number of the cases increases so quickly that it becomes questionable whether it is the right way for handling 3-D situations.

Fig. 4. 6 is an example of 2-D axisymmetric simulation using this updating method.

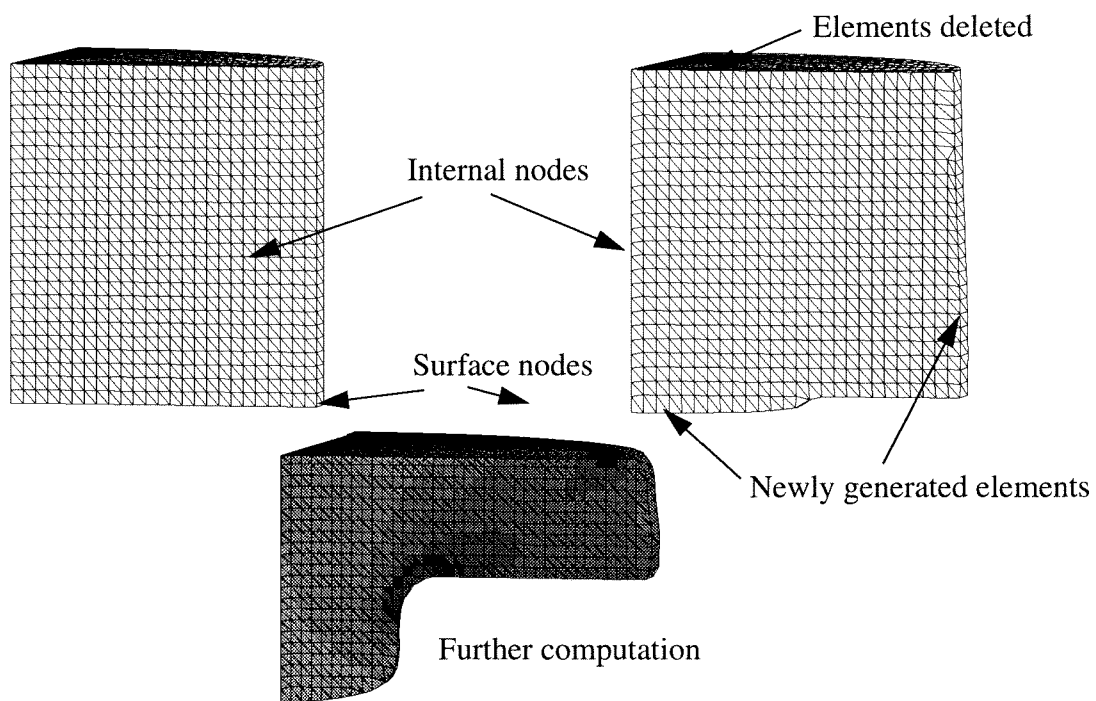


Fig. 4. 6 The simulation of a forging process using the concept of internal and surface nodes

4.1.4 Updated Lagrangian method

Actually, the updated Lagrangian method can be considered as a special case of ALE, in which all the nodes are completely updated. As long as the distortion of the mesh does not

cause serious problems, the updated Lagrangian method is a valuable formulation. The error caused by the computation of gradient is eliminated. The form of the workpiece is well visualized.

The most difficult task for 3-D simulation using the updated Lagrangian method seems to be the optimum adaptive remeshing algorithm, a topic which is being intensively investigated.

Fig. 4. 7 shows the simulation of the ring compression test using the updated Lagrangian method.

The standard scheme of ring test is to measure the inner and outer radii of the ring after compression. The values are compared with a diagram, which is computed according to the upper bound method, to get the friction coefficient. It assumed that the radii are constant over the height. The FE simulation, however, shows that both the inner and outer radii are not constant, although constant friction is applied. Even a double-drum form is obtained. Such difficulty was actually encountered in the experiments.

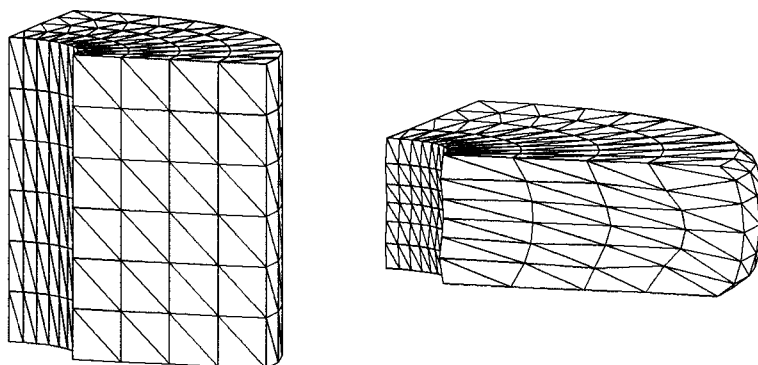


Fig. 4. 7 Simulation of the ring compression test using the updated Lagrangian method

4.2 Mesh generation

The basic idea of the FE method is to subdivide the continuum into discrete elements using simple interpolating functions at the element level to approach the exact solution. Successful simulation depends on a reasonable discretization. Therefore, the first thing to do in an FE program is to generate the computational mesh.

In a real simulation with several thousand elements, the mesh obviously cannot be generated manually. An automatic mesh generator is then necessary. As the computation

proceeds, the mesh may get distorted even though the ALE method is used. The remeshing technique is also an integral part of an FE package.

There are many commercial software packages which can generate the computational mesh automatically. Examples are IDEAS, ProEngineer, MENTAT and so on. Many FE packages have their own pre-processor and can generate the mesh semi-automatically.

Unfortunately, none of the these packages can generate hexahedral elements automatically in arbitrary 3-D domains. They use tetrahedral elements instead. The quality of the mesh is also not ensured.

As will be discussed later, the reduced integration algorithms cannot be applied to the 4-node tetrahedral element. The 10-node tetrahedral element will increase the amount of computation as well as the memory requirements. Therefore, the 8-node hexahedral element is preferable in cases of 3-D bulk forming simulation.

4.2.1 Mesh generator for profile extrusion

The following idea is adopted to implement a simple mesh generator. A basic lattice is located in the domain and it is cut with the given description of the surface of the domain.

For program users, the least possible amount of input data the better. In the simulation of profile extrusion processes the simplest description of the profile is a polygon, and the billet is a cylinder.

It is easy to determine whether a node lies inside the polygon by taking the sum of the angles (Fig. 4. 8). The node is outside the polygon if

$$\sum_n \alpha_n < 0 \quad . \quad (4.6)$$

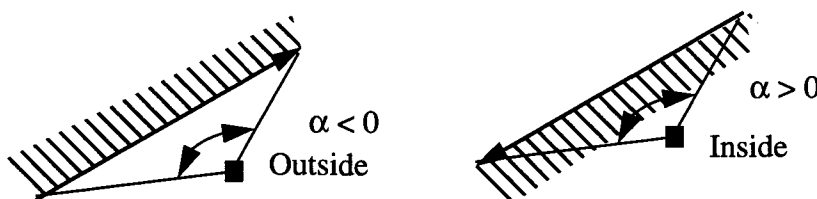


Fig. 4. 8 Angles between a node and a line

To keep the mesh as regular as possible, not all the nodes outside the polygon are

removed. Only those which satisfy the condition

$$d > \max\left(\frac{\Delta x}{2} \cos \alpha, \frac{\Delta y}{2} \sin \alpha\right) \quad (4.7)$$

are deleted from the lattice. d , Δx , Δy and α are found as in Fig. 4. 9.

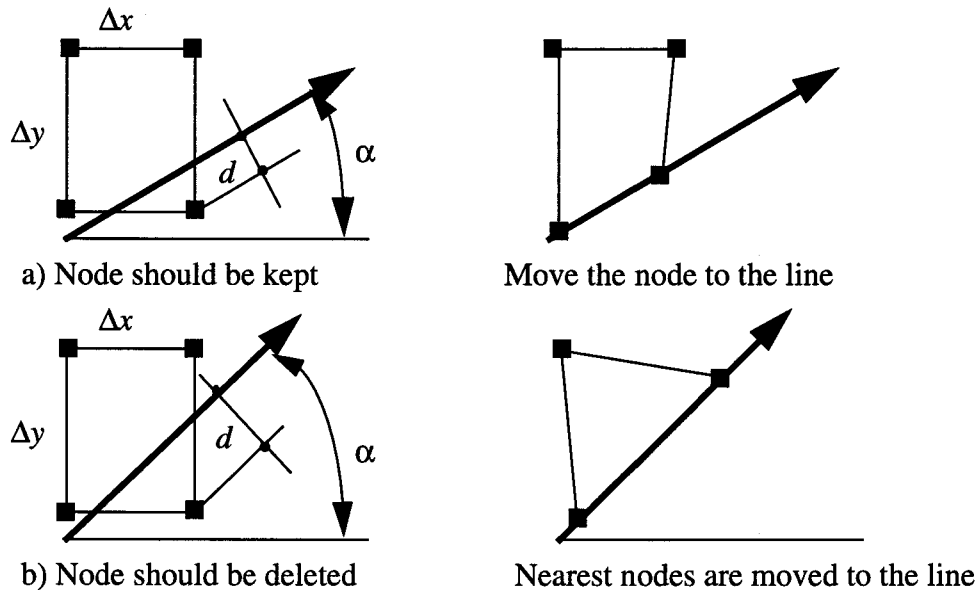


Fig. 4. 9 The condition for removing the node

If a node outside the polygon is not removed, it should be moved to the corresponding side of the polygon. If any node is deleted, the nearest nodes are moved to the side to keep the domain filled.

In the case b) shown in Fig. 4. 9 the 4-node element has a collapsed side or it can degenerate a 3-node triangular element. If more nodes have to be deleted the element itself should be deleted.

In general 3-D cases it becomes more complicated. According to the orientation, only the elements which have 4, 5 or more deleted nodes should be removed from the mesh. Therefore, the elements in the mesh may have 4, 5, 6, 7 or 8 nodes. In other words, 1, 2, 3 or 4 nodes of the hexahedral element might be cut by the boundary of the domain.

Although it is possible to use the degenerated hexahedral elements, the properties of the element deteriorate when the shape function is unable to describe the form of the element. It is very difficult, therefore, in some cases even impossible, to use the hexahedral elements to fit an arbitrary geometry.

An alternative is to fill the form of a cut hexagon with tetrahedral elements. The tetrahedral element may be used with flexibility to fill complex 3-D domains. It is the only option of the 3-D mesh generator of some CAD systems such as IDEAS and ProEngineer. Different cases are listed in Fig. 4. 10.

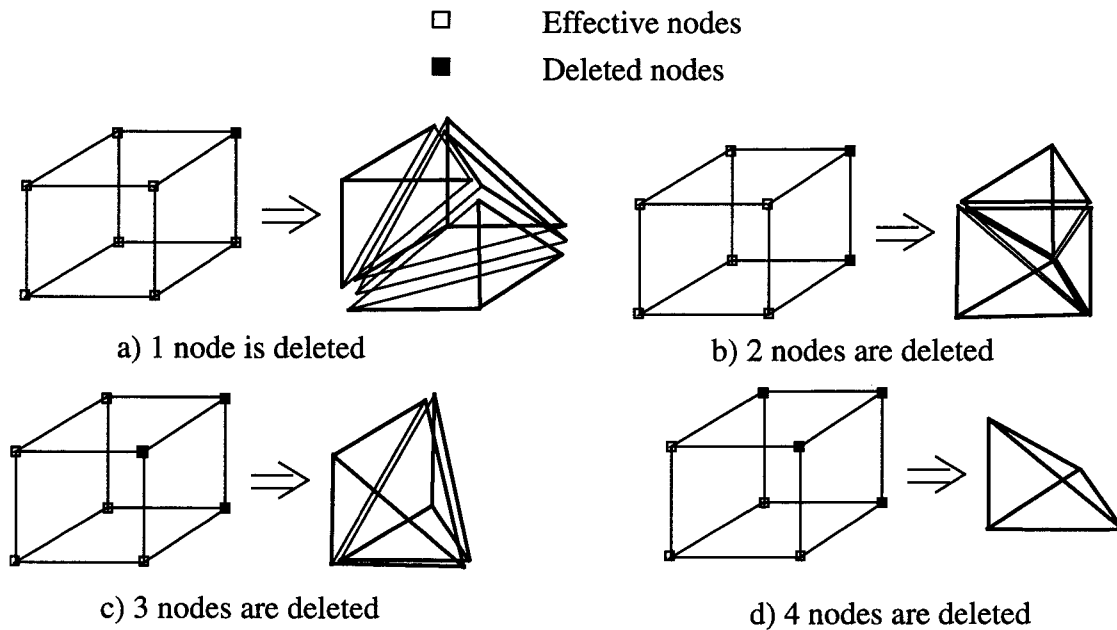


Fig. 4. 10 The transformation from degenerated hexahedral element to tetrahedral elements.

If more than 4 nodes of an element are cut by the surface of the domain, the element should be removed from the mesh. It can be seen from Fig. 4. 10 that the tetrahedral elements can fit the surface of the domain without serious distortion.

Although tetrahedral elements have some disadvantages and are not kinematically compatible with hexahedral elements, the flexibility of this method makes it attractive for implementation in the mesh generator. As the tetrahedral elements are only used to fill the boundary of the domain, no problem appears from the coupling of tetrahedral and hexahedral elements, especially for the computation of displacements and strains. For the simulation of extrusion processes the tetrahedral elements are mainly used to fill the domain where the deformations are very small, and the incompatibility of these two kinds of element is hardly noticed.

Such a combination is also used in the commercial package ABAQUS. In spite of the violation of the kinematic continuity condition, it is allowed in ABAQUS to use both

elements together so as to take advantage of the flexibility of the combination.

4.2.2 Transition between elements with different sizes

One of the main difficulties of 3-D bulk forming simulation is the large size of the problem. The number of degrees of freedom can be as large as 100,000 or even more. The computational costs can be enormous. In fact, there are two different cases in forming processes. In some processes such as upsetting and wire-drawing, the deformations are not very large and the gradient of deformations is also not high. It is reasonable to use a relatively coarse mesh to reduce the computational time. In other forming processes such as extrusion, the deformations are localized in a small zone. The deformation gradients are also extremely high there, and a very fine mesh has to be used. But for the other zone, such a mesh should be avoided to reduce the amount of computation. The transition from the fine mesh to the coarse one should be handled in the mesh generator.

There are basically three methods for the transition. The first one is to extend the dimension of element in certain directions so that the transition is carried out smoothly. The second method is to use special element structures to realise the transition. The third method is to use the constrained nodes and perform the transition by different element classes. Fig. 4. 11 shows different transition methods.

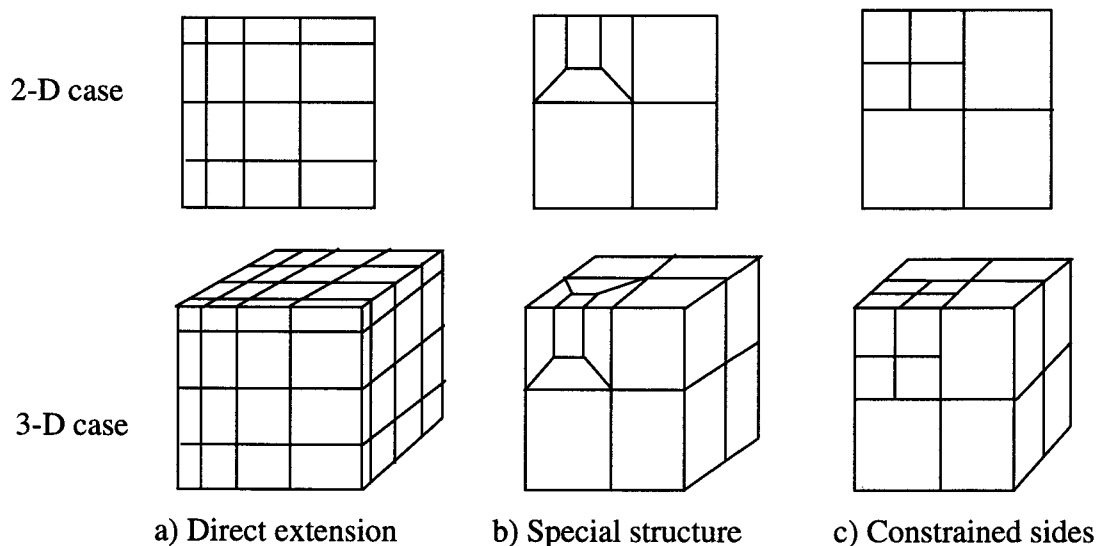


Fig. 4. 11 Different methods to handle the transition between the fine and coarse mesh

It is clear from Fig. 4. 11 that the first method may produce elements which have very different dimensional ratios. Such elements are not suitable in the FE method. The second method uses elements with quite large distortion in the initial state. It is also rather complicated to generate such a structure automatically. By contrast, the third method can keep the best form of the elements and it is easier to implement.

When constrained elements are used, it is favourable to let the constrained node lie in the middle of the related nodes. The stiffness of a constrained element as well as the loads on the nodes of the element are equally distributed to the related nodes. Less information has to be saved.

The concept is shown in Fig. 4. 12. If we use at first the finest elements to fill the domain and combine them afterwards to get the constraints, it is not an easy task to ensure that the number of elements along the common sides are always an even number in the finer zone. An alternative method, therefore, is used to overcome this difficulty. The size of the largest element is determined first and the domain is filled with the largest element. The elements which lie inside the zones where the finest mesh is necessary are found. The elements connected to the finest element are defined as 2nd finest elements. The 3rd finest elements are defined in the same way. Each element detected by the search is then divided into 8 elements. The process goes on until the desired rank is reached. The constraints are searched out over all the common surfaces. Because the elements are generated from the same model, the constraints are easily found. Fig. 4. 12 shows this method schematically.

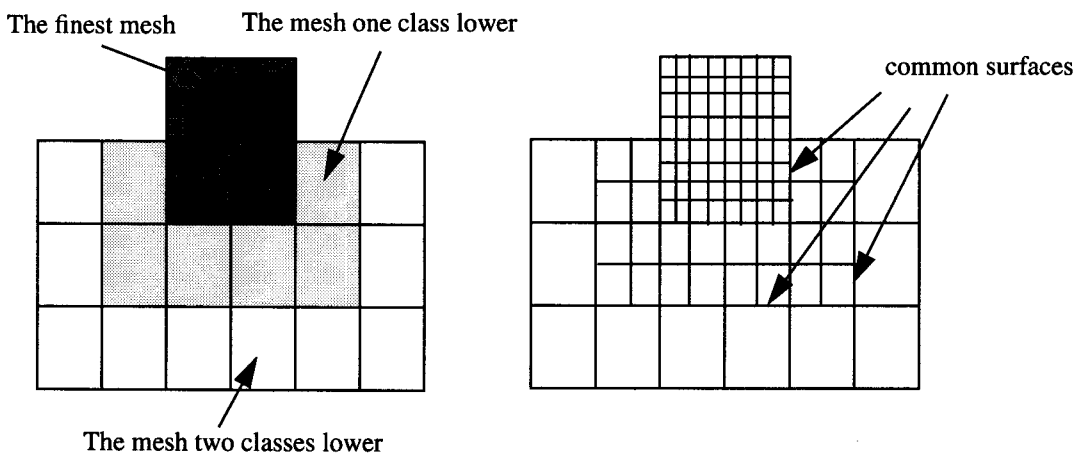


Fig. 4. 12 The method of subdividing the elements into groups.

4.2.3 Example

A simple but effective mesh generator has been developed as a pre-processor to simulate the profile forward extrusion process. Fig. 4. 13 shows an example to generate the computational mesh for the simulation of a cross profile extrusion. With a quite simple input file and a short interactive input on the screen, it takes less than one minute on a SUN Sparc-10 workstation to generate the whole computational mesh as well as the discretization of the extrusion die, although 5000 nodes, 4000 elements and more than 1000 constraints have to be determined. Three classes of elements are used. Without using constrained nodes the mesh would need more than 44,000 nodes and 40,000 elements.

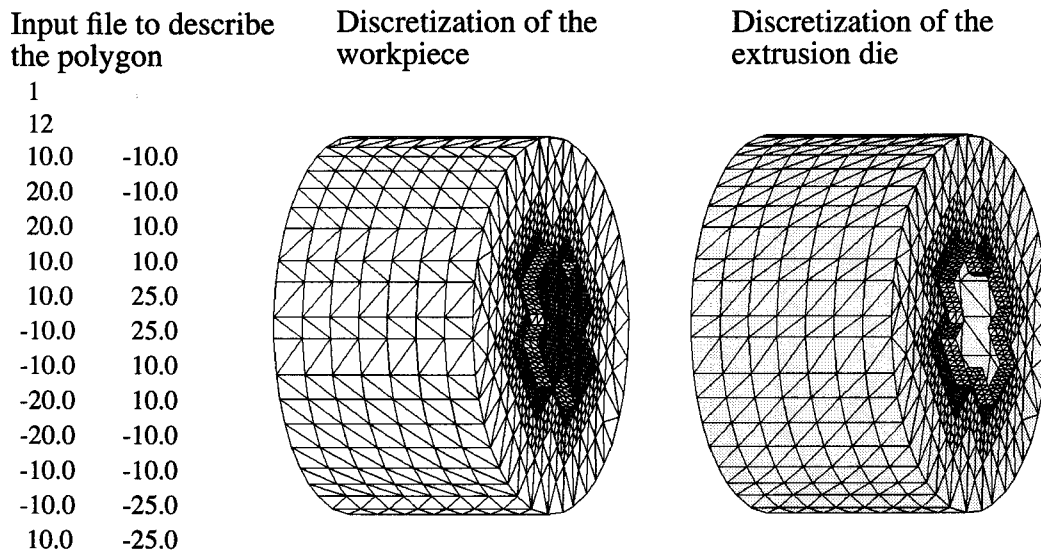


Fig. 4. 13 An example of the mesh generator of PressForm

4.3 On meshing and remeshing

Although the fundamental theory of the FE method has been established for some thirty years, many aspects still need further investigation and progress is possible only when certain problems have been solved. From our viewpoint, the most urgent task for the FE simulation of bulk forming processes is to solve the automatic meshing and adaptive remeshing problems for 3-D cases.

4.3.1 Two basic ways of automatic mesh generation

In 2-D mesh generation the closed boundary of the domain must be specified with one or more curves. For 3-D operations one or more surfaces are necessary to cover the surface

of the domain.

From the known conditions there may be two different ways to perform the discretization.

One method is to fit the boundary first and expand the mesh into the domain. The elements are connected inside the domain to give a continuous and not overlapping mesh.

This method is used in CAD packages like IDEAS and ProEngineer. The advantage is that boundary conditions are satisfied to a given precision. But it is sometimes very difficult to connect the elements from different sides to form a reasonable mesh. Besides, such operations for 3-D cases using hexahedral elements are not yet available.

Another method is to generate regular elements inside the domain and fill the domain last. The method can generate a regular mesh, but it is difficult to fit the boundary of the domain. This method has been used in our FE program PressForm. It was also published in the proceedings of a recent conference [KO94] [TE94]. Fig. 4. 14 shows different methods in the 2-D case.

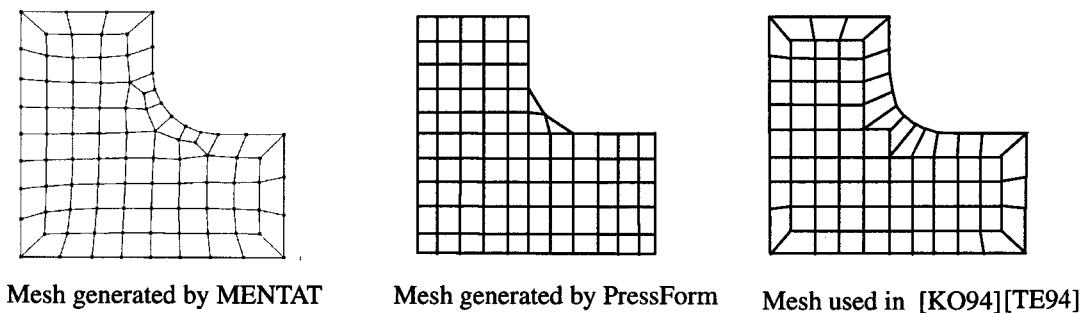


Fig. 4. 14 Different methods of mesh generation

The advantages and disadvantages of each mesh generator can be directly seen from the picture. Generally speaking, if the elements are generated from the boundary, the surface is well described but there may be distorted elements inside the domain. If the procedure is done inversely, the elements inside the domain are properly generated but they may exhibit undesirable distortions on the surface.

4.3.2 An effective method to reduce the size of the system

It is clear that computational time is still a great obstacle for the extension of the

applications even if computers are getting faster and faster. If the FE program cannot provide reliable results in at most a few hours, it is hardly suitable for industrial application.

Obviously, the size of the system is the decisive factor in the computational time. In actual forming processes, the deformations are often concentrated in local zones. For example, the deformation rate in the zones near the opening of the die in an extrusion process is usually a couple of orders of magnitude larger than in the other zones. The computational time can be greatly reduced if we can use fine elements in the deformation zones and have a coarse mesh in other zones.

The constraint elements provide benefit in this way. In 3-D cases one element equals 8 elements of a finer class. The number increases as 8, 64, 512, 4096 etc.

4.3.3 Some future prospects

The mesh generator and the remeshing algorithm in our FE program PressForm are considered to represent a preliminary stage. Although many successful examples of simulation have been performed, shortcomings have been noticed during the application of the program. In industry there are many products produced by profile extrusion which have complex and slender forms, as shown in Fig. 4. 15. It becomes more complicated if fine structures such as radii and frictional bands are considered. Both meshes generated by our method and the method introduced by [KO94] [TE94] do not give a satisfactory mesh because the orientation of the elements should follow the profile. The other method to produce a mesh from the surface may provide better results in zones A and B.

Another shortcoming of the program is that the constraints are not automatically adjusted. In extrusion processes the mesh should become coarse again when it is out of the deformation zones. This will be implemented in the near future.

It seems impossible to implement an algorithm which works for all complex cases and satisfies all conditions. Some suggestions may be useful for further development.

The forward extrusion process possesses a unique characteristic which is worthy of special mention. In this process the form of the profile may be complex but it is actually a 2-D figure. The billet is basically a cylinder which can be fit with a reasonable mesh using a standard method. On one side the 2-D mesh for the profile is expanded to get the

3-D description, and the mesh for the cylinder is refined step by step until it can be connected with the mesh of the profile through proper constraints.

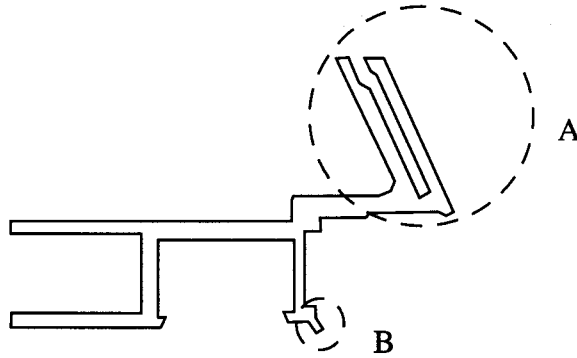


Fig. 4. 15 Complex profile for extrusion process

In general cases it is necessary to take all the information about the surface of the domain, including curvature, edges and corners, into account. There should be enough nodes to adequately describe the surface. The internal nodes should lead to as regular a form of the elements as possible. Constraints should be adopted to reduce the number of degrees of freedom when the gradient is not very high.

5. Rigid-plastic Element Formulation

Following the discussion in chapter 2, the rigid-plastic description has been adopted in our FE program. It works well in most cases of the simulation of bulk forming processes. The basic equations are listed in chapter 2. We concentrate here on another aspect, namely the implementation of the incompressibility condition.

5.1 Two different methods to implement incompressibility for the rigid-plastic element formulation

It is generally accepted that the materials involved are incompressible during forming processes. Although small changes in density are inevitable, the changes in volumetric strain are negligible in comparison with the plastic deformations.

It is clear that the rigid-plastic constitutive relation does not involve elastic deformations and the volumetric strain. It gives only the relation between the plastic strain increments and the deviatoric stresses. The hydrostatic pressure is actually indefinite. Incompressibility serves as the additional condition to determine the hydrostatic pressure. A corresponding term must be added to the formulation of the FE method.

5.1.1 The Lagrange multiplier method

The FE formulation can be derived using different methods. One of these is the variational principle. In this method, solving the equilibrium state is equivalent to searching for the minimum value of a potential function. This means, if we have a potential function

$$\Pi = \Pi (\mathbf{u}^n) , \quad (5.1)$$

the solution of the variational equation

$$\delta \Pi (\mathbf{u}^n) = 0 \quad (5.2)$$

gives the \mathbf{u}^n for the equilibrium position.

If we have a function $\phi = \phi (\mathbf{x})$ and search for the extreme value under the constraint condition $\psi (\mathbf{x}) = 0$, the standard method is the Lagrangian multiplier method. In this method a new function

$$G (\mathbf{x}, \lambda) = \phi (\mathbf{x}) + \lambda \psi (\mathbf{x}) \quad (5.3)$$

is constructed first and the solution of the equation set

$$\left\{ \begin{array}{l} \frac{\partial G}{\partial x} \\ \frac{\partial G}{\partial \lambda} \end{array} \right\} = 0 \quad (5.4)$$

x gives the extreme value and satisfies at the same time

$$\psi(x) = 0 . \quad (5.5)$$

The incompressibility condition in forming processes

$$\varepsilon_{ii} = 0 \quad (5.6)$$

may be treated as an additional condition. Therefore, the variational procedure becomes a constrained extreme value problem.

The potential function of the rigid-plastic description is

$$\Pi = \int_v \Delta \boldsymbol{\varepsilon}^T \boldsymbol{\sigma}' dv - \int_v \Delta \boldsymbol{u}^T \boldsymbol{f} dv - \int_s \Delta \boldsymbol{u}^T \boldsymbol{f}^s ds . \quad (5.7)$$

Considering the incompressibility condition using the Lagrangian multiplier method, equation (5.7) becomes

$$\Pi = \int_v \Delta \boldsymbol{\varepsilon}^T \boldsymbol{\sigma}' dv + \int_v \lambda \Delta \varepsilon_{ii} dv - \int_v \Delta \boldsymbol{u}^T \boldsymbol{f} dv - \int_s \Delta \boldsymbol{u}^T \boldsymbol{f}^s ds . \quad (5.8)$$

The state $[\Delta \boldsymbol{u}^n, \boldsymbol{\lambda}]^T$ which satisfies the equation

$$\left\{ \begin{array}{l} \frac{\partial \Pi}{\partial \Delta \boldsymbol{u}^n} \\ \frac{\partial \Pi}{\partial \boldsymbol{\lambda}} \end{array} \right\} = 0 \quad (5.9)$$

represents the solution for the equilibrium state. λ designates the hydrostatic pressure.

Following the standard FE procedure the variables are expressed by the nodal values using suitable interpolating functions

$$\Delta u = N \Delta u^n \tag{5.10}$$

$$\lambda = N^* \lambda , \tag{5.11}$$

and a set of equations is obtained

$$\begin{bmatrix} K^{uu} & K^{u\lambda} \\ K^{\lambda u} & 0 \end{bmatrix} \begin{Bmatrix} \Delta u^n \\ \lambda \end{Bmatrix} = \begin{Bmatrix} P \\ 0 \end{Bmatrix} . \tag{5.12}$$

This can be solved to get the solution of the problem if the system is linear. For nonlinear systems like in the forming processes iteration is necessary until the given tolerance is achieved.

It should be noticed that we used N and N^* in (5.10) and (5.11) respectively. Actually the form of the interpolating function as well as the number of nodes for interpolation will generally be different.

Most work involving Lagrangian multipliers used normal isoparametric functions for the displacement and added a node in the middle of each element to give the values of the Lagrangian multiplier which are therefore assumed to be constant inside each element, as shown in Fig. 5. 1.

This method was adopted by Lee and Kobayashi [LE73], but they did not explain their reasoning in detail.

An explanation is that since the displacement is described with bi- and tri-linear functions inside the element, the strains and the stresses are nearly constant being derivatives of the displacement. Therefore, one node is enough to satisfy the incompressibility condition.

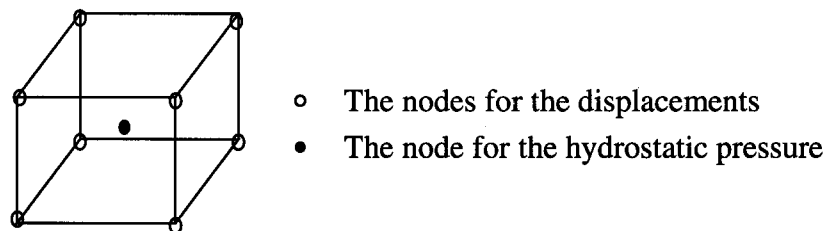


Fig. 5. 1 The structure of an element for the Lagrangian multiplier method

This method causes difficulties when the size of the system increases. Suppose we arrange

the nodes in the order that the nodes for the displacement come first followed by the nodes for the Lagrangian multipliers inside the element, then the stiffness matrix loses the band property leading to high computational costs for large systems.

If the nodes are arranged such that the bandwidth of the matrix is kept as small as possible, the nodes for the displacements must be mixed with the nodes inside the elements. As the number of degrees of freedom varies from node to node, this makes the programming complex.

First we investigate the possibility of using the same function for both displacement and Lagrangian multipliers, i.e. each node has 4 degrees of freedom, three for the displacement and one for the hydrostatic pressure. The same interpolating function is used for both variables.

With this formulation we get the stiffness matrix for an 8-node hexahedral element like

$$[K^{uu}]_{24 \times 24} = \int_v \frac{2\sigma_v}{3\Delta\epsilon_v} \mathbf{B}^T \mathbf{B} dv \quad (5.13)$$

and

$$[K^{u\lambda}]_{24 \times 8} = \int_v \mathbf{B}^T \mathbf{C}^T \mathbf{N} dv \quad (5.14)$$

where \mathbf{B} is the matrix defined by $\Delta\boldsymbol{\epsilon} = \mathbf{B}\Delta\mathbf{u}^n$ and $\mathbf{C} = [1 \ 1 \ 1 \ 0 \ 0 \ 0]$ is the operator to get the volumetric strain.

The matrices are then assembled in the order that the variables are arranged in the global system as

$$\begin{Bmatrix} \Delta\mathbf{u}^n \\ \lambda \end{Bmatrix} = [u^1 \ v^1 \ w^1 \ \lambda^1 \ \dots]^T \quad (5.15)$$

Although the formulation keeps a relatively small bandwidth and is easy to implement, it failed to reach a convergent solution. In order to discover the reason the eigenvalues of the element stiffness were investigated.

The stiffness was computed according to (5.13) and (5.14) and the eigenvalues are

evaluated. The results are shown in Fig. 5. 2. It can be seen from the results that this element has 7 vanishing eigenvalues. 6 of them are for the 6 degrees of freedom of rigid body movements which are prescribed as boundary conditions. The additional one is for the Lagrangian multiplier or the hydrostatic pressure. If such an element is used, the boundary conditions for the hydrostatic pressure must be given to get a convergent solution. Unfortunately, this is impossible in most cases of forming processes. This investigation shows, therefore, why the same interpolating function cannot be used for the Lagrangian multipliers.

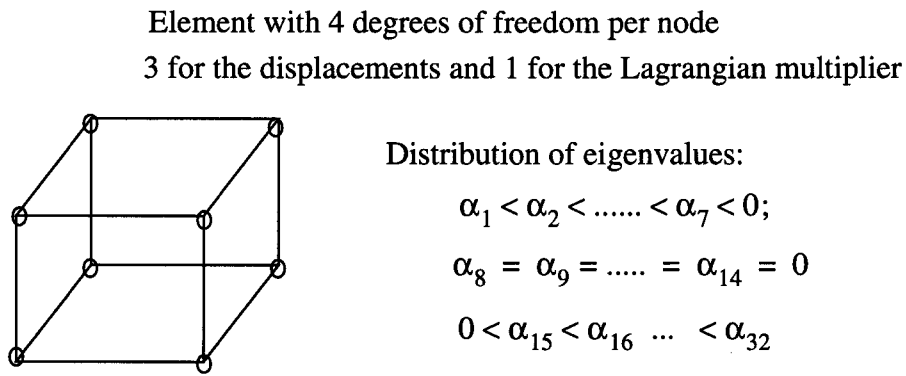


Fig. 5. 2 Distribution of eigenvalues of a cubic element with 32 degrees of freedom

Therefore, a structure like Fig. 5. 1 has to be used. The only difference is the interpolation of the Lagrangian multiplier. Because there is only one node for the Lagrangian multiplier, it is expressed directly as

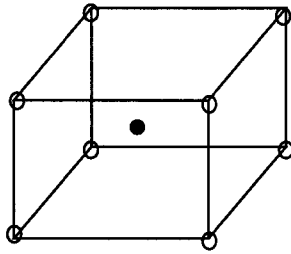
$$\lambda = \lambda^j , \tag{5.16}$$

which means that the interpolating function N^* has only one component

$$N_j^* = 1.0 = \text{constant} . \tag{5.17}$$

The eigenvalues are also evaluated for this element and the results are shown in Fig. 5. 3. It can be seen that the eigenvalue for the hydrostatic pressure is negative. The normal boundary conditions can be applied to get the solution.

Element with 8 nodes for the displacements and
1 node in the middle for the Lagrangian multipliers



Distribution of eigenvalues:

$$\alpha_1 < 0;$$

$$\alpha_2 = \alpha_3 = \dots = \alpha_7 = 0$$

$$0 < \alpha_8 < \alpha_9 \dots < \alpha_{25}$$

Fig. 5.3 Distribution of eigenvalues of a cubic element with 25 degrees of freedom

However, the difficulty caused by a large bandwidth remains, and further we see that the element matrix is not positive definite. Both these properties are not favourable for the solution of a large set of equations.

5.1.2 The Penalty method

An alternative method to handle the incompressibility condition is the penalty method.

Suppose the task is to find the extreme value of

$$\phi = \phi(x, y, z) \quad , \quad (5.18)$$

under the condition

$$\psi(x, y, z) = 0 \quad , \quad (5.19)$$

a new function can be constructed as

$$\Pi = \phi(x, y, z) + \frac{1}{2}\kappa\psi^2(x, y, z) \quad (5.20)$$

where κ is a large number to ensure the condition (5.19).

The extreme value of the new function Π is computed with the standard method and the solution goes to the exact solution of (5.18) and (5.19) if the penalty factor κ is large enough.

The penalty method does not use additional variables so that the bandwidth of the stiffness matrix can be optimized easily. The stiffness matrix is positive semi-definite. But if the penalty factor is not large enough, the additional condition is not well satisfied. If the

penalty factor is chosen too large, the condition number of the global equation system deteriorates so that the accuracy of solution is lost.

Another problem in the formulation of the penalty method is the so-called 'locked deformation' and a reduced integration technique must be used. This aspect was intensively discussed in the 80's [TH79] [RE88]. In the formulation of the penalty method the stiffness matrix is made up of two terms, one coming from the original problem and the other from the penalty formulation. The equation can be written as

$$[\mathbf{K}^s + \mathbf{K}^k] \mathbf{u}^n = \mathbf{P} \quad (5.21)$$

where \mathbf{K}^s is the stiffness for the deviatoric term and \mathbf{K}^k denotes the penalty term.

Because the penalty term is a couple of orders of magnitude larger than the deviatoric term in the equation, the numerical solution procedure may be converge to the wrong solution. In other words, the penalty method may also prevent some real deformation models when it prevents volume variation.

The comparison of simulation results obtained from different integration methods is shown in Fig. 5. 4. It can be seen directly that only reduced integration can give a reasonable description of the deformation.

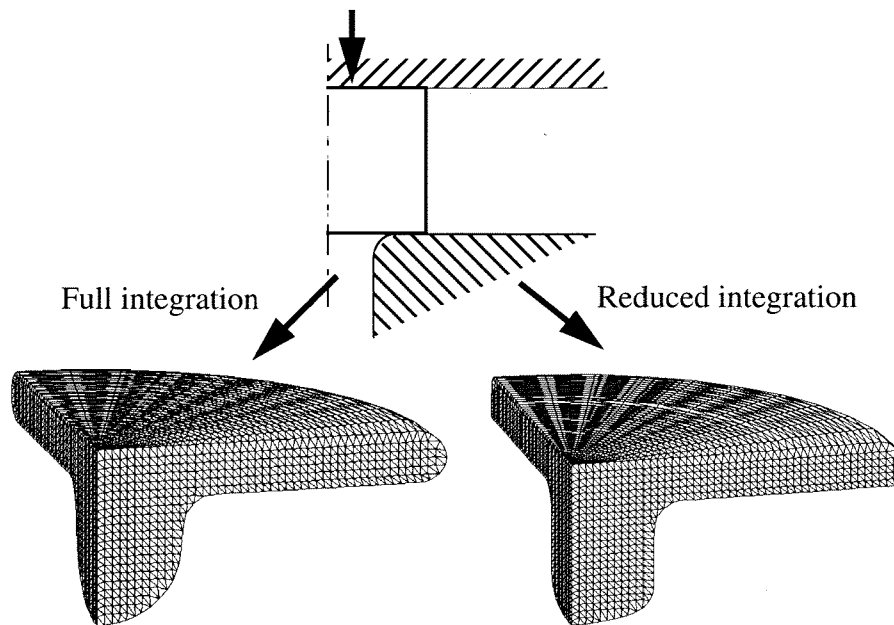


Fig. 5. 4 Different deformation models by different integration methods

The reason can be explained using a 4-node plane strain element as shown in Fig. 5. 5. The element is deformed in the x-y plane and the incompressibility condition is then

$$\dot{\epsilon}_{xx} + \dot{\epsilon}_{yy} = 0 \quad (5.22)$$

When the element is fully integrated using 4 Gaussian integration points for both terms, the deformation model shown in Fig. 5. 5 a) is 'locked' through the penalty term because the component of strain rate $\dot{\epsilon}_{xx}$ is not zero at the integration points but the component $\dot{\epsilon}_{yy}$ vanished. Actually this deformation does not violate the incompressibility condition at the element level, and therefore it is permissible.

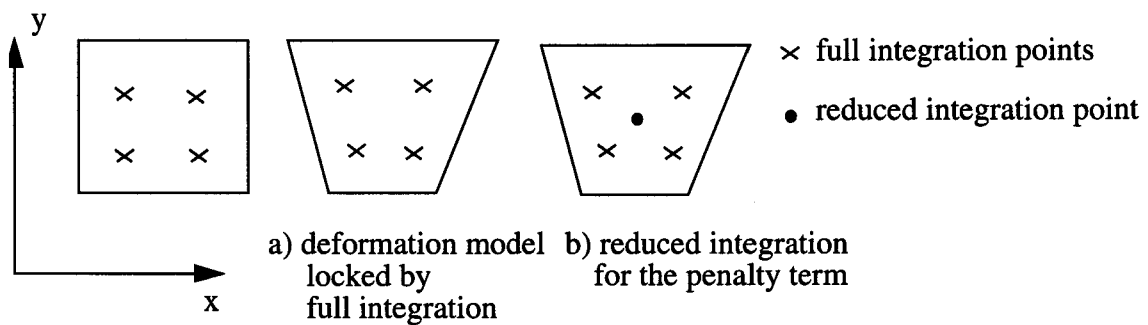


Fig. 5. 5 Reduced integration method

Reduced integration is then used to release the locked deformation, i.e. full integration is performed for the deviatoric term and a lower order integration is used for the penalty term. As seen from b) of Fig. 5. 5, the strain component $\dot{\epsilon}_{xx}$ is also zero at the reduced integration point. The incompressibility condition is then satisfied in the mean value. The difficulty arising from the locking model is then overcome.

5.1.3 Comparison of the two methods

From the discussions in previous sections a comparison is given in Table 5. 1.

Table 5. 1

	Lagrangian multiplier method	Penalty method
Mathematical formulation	Exact solution	Approximate solution
Degrees of freedom	3 x No. of Nodes + No. of Elements	3 x No. of Nodes

Table 5. 1

	Lagrangian multiplier method	Penalty method
Band of global stiffness matrix	Large and difficult to optimize	Normal and easy to optimize
Property of positive definiteness	Non-positive definitive stiffness matrix	Positive-definitive stiffness matrix
Integration of the element	Normal integration	Reduced integration for the penalty term

Apart from the reasons mentioned above, the Lagrangian multiplier method has another disadvantage. From (5.12) it can be seen that all the diagonal terms for the Lagrangian multipliers are zero. This is not favourable for most solution algorithms. Although exchanging rows and columns can avoid the problem, it leads to complex operations.

The penalty method requires the actual choice of the penalty factor. A relative penalty factor like

$$\kappa = \alpha \frac{\sigma_v}{\Delta \epsilon_v} \tag{5.23}$$

is adopted. Generally speaking, the larger the factor is, the better the incompressibility condition is satisfied but more CPU time is needed for the iterative algorithm to solve the equations. Fig. 5. 6 shows the influence of the penalty factor. Computational examples show that when $\alpha < 100$, there are significant volume changes. When $\alpha > 10^5$, the increase of CPU time is noticeable. Therefore, a reasonable penalty factor should lie between these values. In PressForm we used $10^3 < \alpha < 10^4$ for 3-D cases and $10^4 < \alpha < 10^5$ for 2-D cases. The incompressibility condition is satisfied and no difficulty arises with the iterative solver.

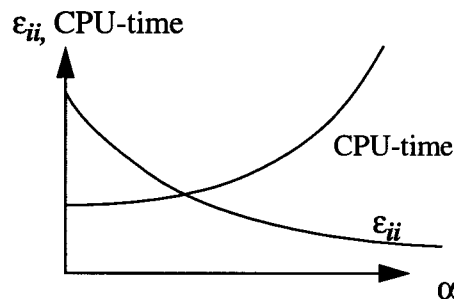


Fig. 5. 6 Influence of the penalty factor on CPU time and incompressibility

A simple example, namely the upsetting of a cylinder under moderate friction, was chosen as a computational example to compare both methods.

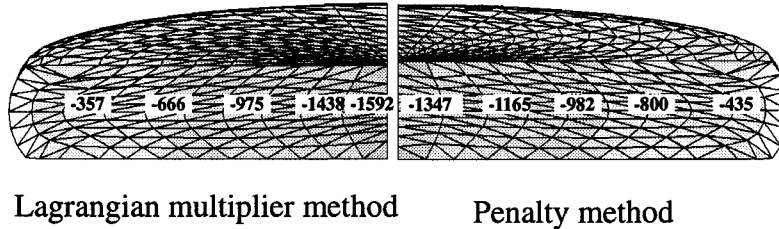


Fig. 5.7 Comparison of Lagrangian multiplier method and Penalty method

It was found that both methods give reliable results. The difference of the form of the workpiece obtained from the simulations is very small. The hydrostatic pressure, which shows the maximum difference, is labelled in the picture. The two distributions show the same tendency, although the values are not identical.

From the above comparison we arrive at the following conclusion: Although both methods can provide reliable results, it is better to use the penalty method because it uses less degrees of freedom and the stiffness matrix remains positive-definite. Only in the simulation of such processes where the hydrostatic pressure and the incompressibility condition are determinant, should the Lagrangian multiplier method be used. In other words, the penalty method is the superior method, while the Lagrangian multiplier method should only be used as a complementary one.

5.2 Element formulation for stress-strain computations

The FE formulation can be derived from the variational principle. Using the penalty method the potential energy of the domain is defined as

$$\Pi = \int_v \Delta \boldsymbol{\varepsilon}^T \boldsymbol{\sigma}' dv + \int_v \frac{K}{2} (\Delta \varepsilon_{ii})^2 dv - \int_v \Delta \mathbf{u}^T \mathbf{f}^b dv - \int_s \Delta \mathbf{u}^T \mathbf{f}^s ds \quad (5.24)$$

The coordinates and the displacements inside the element are expressed by the values at the nodes

$$\mathbf{x} = \mathbf{N} \mathbf{x}^n \quad (5.25)$$

and

$$\Delta u = N \Delta u^n \quad , \quad (5.26)$$

where N is the interpolating function defined in the local coordinate system as shown in Fig. 5. 8. The interpolating functions must satisfy certain conditions. Firstly,

$$\sum_{i=1}^n N_i(\xi, \eta, \zeta) = 1.0 \quad (5.27)$$

and secondly

$$N_i(\xi_j, \eta_j, \zeta_j) = \delta_{ij} \quad (5.28)$$

where i and j are the numbers of the nodes, ξ_j, η_j, ζ_j are the values of the local coordinates of the nodes and δ_{ij} is the Kronecker delta.

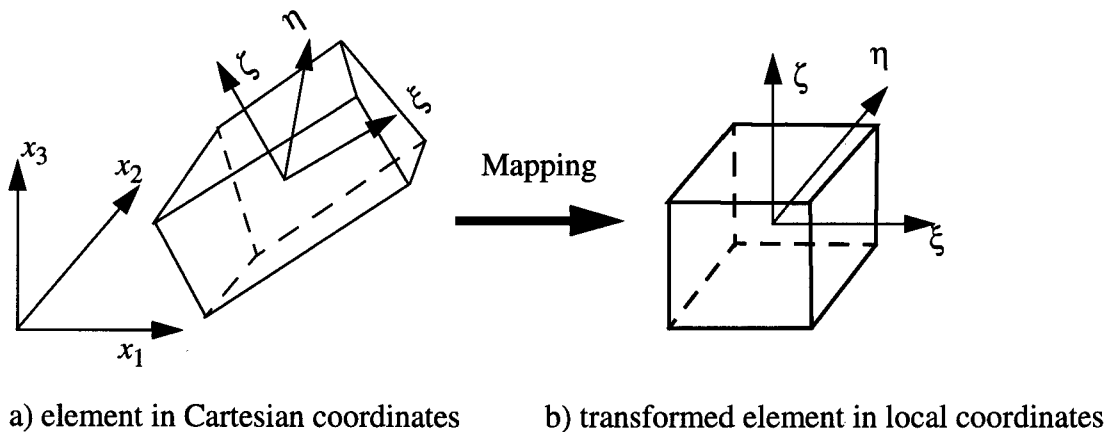


Fig. 5. 8 Local coordinates for the hexahedral element

Typically, the 8-node isoparametric hexahedral element uses the interpolating functions

$$N_i(\xi, \eta, \zeta) = \frac{1}{8} (1 + \xi_i \xi) (1 + \eta_i \eta) (1 + \zeta_i \zeta) \quad . \quad (5.29)$$

The derivatives of the interpolating functions can be calculated as

$$\begin{Bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \\ \frac{\partial N_i}{\partial z} \end{Bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{bmatrix}^{-1} \begin{Bmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \\ \frac{\partial N_i}{\partial \zeta} \end{Bmatrix} = [J]^{-1} \begin{Bmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \\ \frac{\partial N_i}{\partial \zeta} \end{Bmatrix} \quad (5.30)$$

where the Jacobian Matrix $[J]$ is computed from (5.25).

Then the strain increment can be expressed as

$$\Delta \boldsymbol{\varepsilon} = \mathbf{B} \Delta u^n \quad (5.31)$$

with

$$\mathbf{B} = [B_1 \ B_2 \ \dots \ B_n] \quad (5.32)$$

and

$$\mathbf{B}_i = \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 & 0 & \frac{1}{\sqrt{2}} \frac{\partial N_i}{\partial y} & 0 & \frac{1}{\sqrt{2}} \frac{\partial N_i}{\partial z} \\ 0 & \frac{\partial N_i}{\partial y} & 0 & \frac{1}{\sqrt{2}} \frac{\partial N_i}{\partial x} & \frac{1}{\sqrt{2}} \frac{\partial N_i}{\partial z} & 0 \\ 0 & 0 & \frac{\partial N_i}{\partial z} & 0 & \sqrt{2} \frac{\partial N_i}{\partial y} & \frac{1}{\sqrt{2}} \frac{\partial N_i}{\partial x} \end{bmatrix}^T \quad (5.33)$$

It is easy to get the expression for the incremental volumetric strain

$$\Delta \varepsilon_{ii} = \mathbf{C} \mathbf{B} \Delta u^n \quad (5.34)$$

where

$$\mathbf{C} = [1 \ 1 \ 1 \ 0 \ 0 \ 0] \quad (5.35)$$

Substituting equations (5.26), (5.31), (5.34) and (2.37) into (5.24), the variation of potential energy is expressed as

$$\begin{aligned} \delta\Pi = & (\delta\Delta\mathbf{u}^n)^T \left(\int_v \frac{2\sigma_v}{3\Delta\varepsilon_v} \mathbf{B}^T \mathbf{B} dv \right) \Delta\mathbf{u}^n + (\delta\Delta\mathbf{u}^n)^T \left(\int_v \kappa \mathbf{B}^T \mathbf{C}^T \mathbf{C} \mathbf{B} dv \right) \Delta\mathbf{u}^n \\ & - (\delta\Delta\mathbf{u}^n)^T \int_v \mathbf{N}^T \mathbf{f}^b dv - (\delta\Delta\mathbf{u}^n)^T \int_s \mathbf{N}^T \mathbf{f}^s ds = 0 \quad . \end{aligned} \quad (5.36)$$

Because the variation $\delta\Delta\mathbf{u}^n$ is arbitrary, the sum of other terms must be zero. It leads to the finite element formulation

$$\begin{aligned} & \left(\int_v \frac{2\sigma_v}{3\Delta\varepsilon_v} \mathbf{B}^T \mathbf{B} dv \right) \Delta\mathbf{u}^n + \left(\int_v \kappa \mathbf{B}^T \mathbf{C}^T \mathbf{C} \mathbf{B} dv \right) \Delta\mathbf{u}^n \\ & = \int_v \mathbf{N}^T \mathbf{f}^b dv + \int_s \mathbf{N}^T \mathbf{f}^s ds \end{aligned} \quad (5.37)$$

which can be written in matrix form as

$$(\mathbf{K}^s + \mathbf{K}^\kappa) \Delta\mathbf{u}^n = \mathbf{P} \quad . \quad (5.38)$$

The Gaussian integration method is usually used to integrate (5.37) to get (5.38). As mentioned before, full integration are chosen for the \mathbf{K}^s and \mathbf{P} , while reduced integration is performed for \mathbf{K}^κ to avoid the ‘locking’ problem.

Then the matrices of an element are obtained as

$$\mathbf{K}^s = \sum_{k=1}^n \frac{2\sigma_v}{3\Delta\varepsilon_v} \mathbf{B}^T \mathbf{B} \left| \frac{\partial X}{\partial \xi} \right| w_k \Big|_{\xi = \xi_k, \eta = \eta_k, \zeta = \zeta_k} \quad (5.39)$$

and

$$\mathbf{P} = \sum_{k=1}^n \mathbf{N}^T \mathbf{f} \left| \frac{\partial X}{\partial \xi} \right| w_k \Big|_{\xi = \xi_k, \eta = \eta_k, \zeta = \zeta_k} \quad (5.40)$$

where

$$\xi_k, \eta_k, \zeta_k = \pm \frac{1}{\sqrt{3}} \quad \text{and} \quad w_k = 1.0 \quad (5.41)$$

are respectively the local coordinates and the weighting factor for the hexahedral element.

For the penalty term it becomes

$$K^k = \kappa B^T C^T C B \left| \frac{\partial X}{\partial \xi} \right| w_0 \Big|_{\xi = \xi_0, \eta = \eta_0, \zeta = \zeta_0} \quad (5.42)$$

where

$$\xi_0 = \eta_0 = \zeta_0 = 0.0 \text{ and } w_0 = 8.0 \quad (5.43)$$

are the local coordinates and the weighting factor of the reduced integration.

The matrices and force vectors of the elements are then assembled as usual and the equations are solved to obtain the solution for each iteration.

Other kinds of elements, for example, the tetrahedral element and the 4-node quadrilateral element for the 2-D simulation are obtained in a similar way.

5.3 Element formulation for thermal analysis

The basic equation of thermal analysis may be derived from the first law of thermodynamics and the plastic work is treated just as a source of heat flux inside the domain.

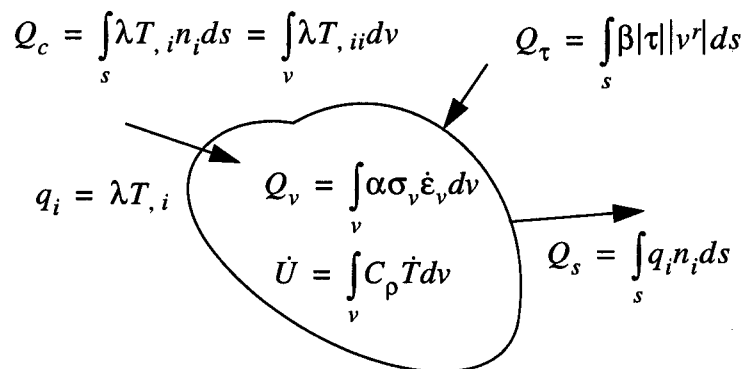


Fig. 5.9 Thermal system of an element

We consider the element shown in Fig. 5.9. The change of the internal energy of the element, which may be interpreted as the change of the temperature, is caused by three terms. The first term is the heat conduction. It follows from the conduction law that

$$q_i = \lambda T_{,i} \quad (5.44)$$

where the coefficient of conductivity λ is assumed to be constant.

The total contribution of this term to the volume is

$$Q_c = \int_s \lambda T_{,i} n_i ds \quad . \quad (5.45)$$

According to the divergence theorem, it becomes

$$Q_c = \int_v \lambda T_{,ii} dv \quad . \quad (5.46)$$

The heat transformed from the plastic work equals

$$Q_v = \int_v a \sigma_v \dot{\epsilon}_v dv \quad . \quad (5.47)$$

The dimensionless coefficient a is determined by experiment to be 0.9 - 0.95.

On the surface of the domain the frictional forces can also induce heat, which is expressed as

$$Q_\tau = \oint_s \beta |\tau| |v^r| ds \quad . \quad (5.48)$$

The sum of all the terms gives the change of the internal energy as

$$\dot{U} = \int_v C_\rho \dot{T} dv \quad (5.49)$$

where C_ρ is the specific heat per unit volume.

The conservation of energy then gives the equation for the thermal analysis as

$$\int_v C_\rho \dot{T} dv = \int_v \lambda T_{,ii} dv + \int_v a \sigma_v \dot{\epsilon}_v dv + \int_s \beta |\tau| |v^r| ds \quad . \quad (5.50)$$

Galerkin's method is used to obtain the FE formulation. Integrating equation (5.50) under a weighting function w leads to

$$\int_v w C_\rho \dot{T} dv = \int_v w \lambda T_{,ii} dv + \int_v w a \sigma_v \dot{\epsilon}_v dv + \int_s w \beta |\tau| |v^r| ds \quad . \quad (5.51)$$

Only the first term of the right hand side has to be transformed as

$$\begin{aligned} \int_{\nu} w \lambda T_{,ii} dv &= \int_{\nu} (w \lambda T_{,i})_{,i} dv - \int_{\nu} (w_{,i}) \lambda T_{,i} dv \\ &= \int_s w \lambda T_{,i} n_i ds - \int_{\nu} (w_{,i}) \lambda T_{,i} dv \end{aligned} \quad (5.52)$$

The term $\lambda T_{,i} n_i ds$ is simply the heat flux on the surface q_s , and (5.51) can then be written as

$$\begin{aligned} &\int_{\nu} w C_{\rho} \dot{T} dv + \int_{\nu} w_{,i} \lambda T_{,i} dv \\ &= \int_s w q_s ds + \int_{\nu} w \alpha \sigma_{\nu} \dot{\epsilon}_{\nu} dv + \int_s w \beta |\tau| |v^r| ds \end{aligned} \quad (5.53)$$

Using the normal interpolating function for temperature T as well as for the weighting function w

$$T = NT \text{ and } w = Nw \quad , \quad (5.54)$$

equation (5.53) becomes

$$\begin{aligned} &w^T \left(\int_{\nu} C_{\rho} N^T N dv \right) \dot{T} + w^T \left(\int_{\nu} \lambda L^T L dv \right) T \\ &= w^T \int_s N^T q_s ds + w^T \int_{\nu} N^T \alpha \sigma_{\nu} \dot{\epsilon}_{\nu} dv + w^T \int_s N^T \beta |\tau| |v^r| ds \end{aligned} \quad (5.55)$$

where L is the operator matrix for the gradient.

Since the weighting function w is arbitrary

$$\begin{aligned} &\left(\int_{\nu} C_{\rho} N^T N dv \right) \dot{T} + \left(\int_{\nu} \lambda L^T L dv \right) T \\ &= \int_s N^T q_s ds + \int_{\nu} N^T \alpha \sigma_{\nu} \dot{\epsilon}_{\nu} dv + \int_s N^T \beta |\tau| |v^r| ds \end{aligned} \quad (5.56)$$

or in matrix form

$$CT + KT = Q_s + Q_{\nu} + Q_{\tau} \quad (5.57)$$

is obtained. In above equation

$$C = \int_{\nu} C_{\rho} N^T N d\nu \quad (5.58)$$

is the heat capacity matrix,

$$K = \int_{\nu} \lambda L^T L d\nu \quad (5.59)$$

is the heat conduction matrix, and Q_s, Q_v, Q_{τ} are the equivalent heat fluxes at the nodes caused by the heat flux on the surface, the heat generated from plastic work and the heat generated by the frictional forces on the surface respectively.

In most cases of forming processes the temperature is non-stationary. (5.57) should be integrated over time. Thus the equation

$$C \Delta T + K T \Delta t = (Q_s + Q_v + Q_{\tau}) \Delta t \quad (5.60)$$

must be solved. Although it is generally acceptable that K and C are constant during a small time interval, the temperature as well as the heat flux is not constant. Numerical instability may occur if the time increment is too large.

Two measures to prevent this are taken in PressForm. Firstly, we used the average value of the increment instead of the initial value used in the Euler forward method. Using the mean temperature

$$\bar{T} = T^0 + \theta \Delta T \quad (5.61)$$

and the equation is transformed to

$$C^* \Delta T + K T^0 \Delta t = (Q_s + Q_v + Q_{\tau}) \Delta t \quad (5.62)$$

where

$$C^* = C + \theta K \quad (5.63)$$

By simply setting $\theta = 0.0, 0.5$ and 1.0 respectively, the forward, median and backward integration methods can be used. In practice quite good numerical stability is achieved by using $\theta = 0.5$, i. e. the average value.

Another measure is to divide the time interval into subincrements and to control the maximum temperature increment of each time subincrement and not to let it exceed a given error tolerance ΔT^{tol} . Because the matrices C and K remain constant during the whole increment, they have to be assembled only once at the beginning of the increment. In every subincrement only the vectors Q_s and KT are reevaluated and equation (5.62) solved.

If the maximum temperature increment ΔT^{max} is larger than ΔT^{tol} , the updating coefficient

$$\alpha = \begin{cases} \frac{\Delta T^{tol}}{\Delta T^{max}} & (\Delta T^{max} > \Delta T^{tol}) \\ 1.0 & (\Delta T^{max} \leq \Delta T^{tol}) \end{cases} \quad (5.64)$$

is calculated and the corresponding subincrement $d\Delta t$

$$(d\Delta t)_i = \alpha (\Delta t)_i^{rest} \quad (5.65)$$

is subtracted from the total time interval Δt and the rest of the time increment $(\Delta t)_i^{rest}$ is calculated accordingly. The temperature is updated as

$$T_i = T_{i-1} + \alpha (\Delta T)_i \quad (5.66)$$

This procedure is repeated until the total time is reached. Fig. 5. 10 shows the comparison of the different integration methods.

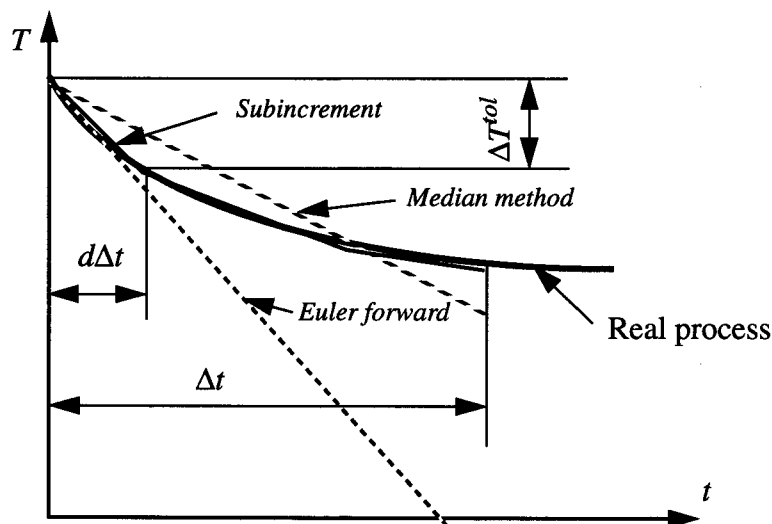


Fig. 5. 10 Different methods to integrate the temperature increment

Fig. 5. 11 shows the diagram used to perform the thermal analysis.

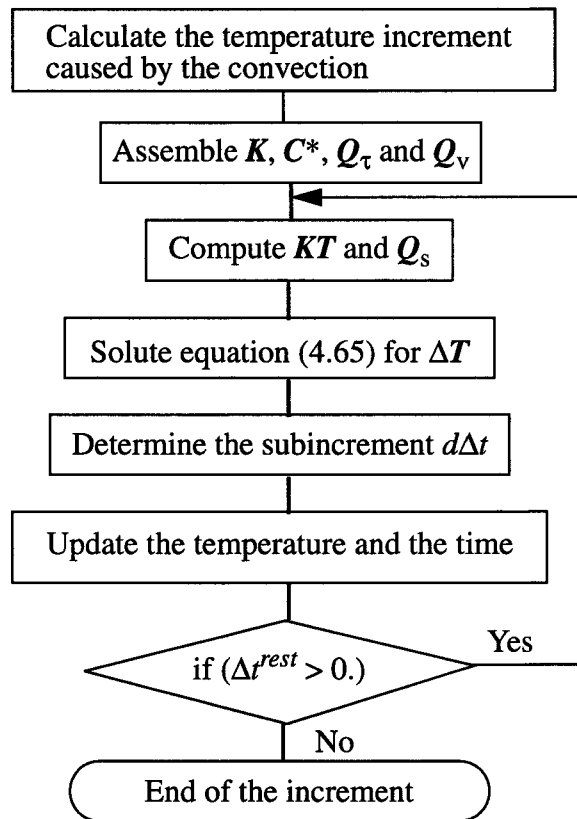


Fig. 5. 11 Diagram for the computation of temperature

The method has been shown in practice to be very stable. Because the conduction matrix and the heat capacity matrix are assembled only once at the beginning of the increment and the number of degrees of freedom is only one per node, the computational time for thermal analysis is very small in comparison with the stress-strain calculation.

6. Discussion of some aspects of the FE program

Besides the discretization of the computational domain and the formulation of the element properties, there are still several important aspects to be discussed in detail in order to compute forming processes efficiently. These aspects concern: the methods used for solving the linear equations, the modelling of friction and estimating the initial values needed to start the iteration for the solution of the system.

6.1 Comparison of different algorithms to solve the equations

Assembling the element contributions the set of global equilibrium equations

$$Ku = P \tag{6.1}$$

is obtained. This set of equations has to be solved using a suitable method. If the problem is nonlinear the linearised equations have to be solved repeatedly until the convergence tolerance is reached.

Investigations have shown that if the number of degrees of freedom increases from 600 in a 2-D case to 10,000 in a 3-D case, the CPU time used by the equation solver increases from 36% to 73% of the total CPU time [DO92]. In practice it is not seldom that the number of degrees of freedom is larger than 100,000 for a 3-D simulation of a real-life forming process. In this case, the proportion of time used by the equation solver is still higher. Since the total CPU time increases quadratically to cubically when the size of the system increases, the CPU time used by the solver is huge. Therefore, it is very important to optimize solution procedures in order to make savings in the total computing time.

There are basically two approaches to solve large sets of linear equations: direct solution algorithms and iterative methods.

It is difficult to give a general evaluation of different solution methods because the results vary from case to case. But to compare some methods with computational examples will help choosing adequate solution algorithms, thereby saving large amounts of computing time.

6.1.1 Direct solvers

Using direct solving procedures is the classical way of solving of a set of linear equations. An example is the Gaussian elimination procedure which transforms the matrix to an

upper triangular form and obtains the solution by substituting the unknowns backwards.

The matrix has the form

$$K = \begin{bmatrix} K_{11} & K_{12} & \dots & K_{1n} \\ K_{21} & K_{22} & \dots & K_{2n} \\ \dots & \dots & \dots & \dots \\ K_{n1} & K_{n2} & \dots & K_{nn} \end{bmatrix} . \quad (6.2)$$

When the diagonal terms K_{rr} are not zero, they can be used to eliminate the r -th column in the i -th row ($r < i \leq n$)

$$K'_{ij} = K_{ij} - \frac{K_{ir}}{K_{rr}} K_{rj} \quad \text{and} \quad P'_i = P_i - \frac{K_{ir}}{K_{rr}} P_r . \quad (6.3)$$

After three loops ($r = 1$ to $n-1$; $i = r+1$ to n and $j = r$ to n), the matrix becomes an upper triangular matrix with the components

$$K' = \begin{bmatrix} K'_{11} & K'_{12} & \dots & K'_{1n} \\ 0 & K'_{22} & \dots & K'_{2n} \\ \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & K'_{nn} \end{bmatrix} . \quad (6.4)$$

It is easy to substitute backwards to get the desired solution as

$$u_i = \left(P'_i - \sum_{k=i+1}^n K'_{ik} u_k \right) / K'_{ii} \quad (i=n, n-1, \dots, 1) . \quad (6.5)$$

An alternative but equivalent algorithm is Crout decomposition which factorises the matrix K to the product of two matrices

$$K = LU \quad (6.6)$$

where L and U are, respectively, the lower and upper triangular matrices. The matrix L has the form

$$L = \begin{bmatrix} 1.0 & 0 & \dots & 0 \\ L_{21} & 1.0 & \dots & 0 \\ \dots & & & \\ L_{n1} & L_{n2} & \dots & 1.0 \end{bmatrix} . \quad (6.7)$$

and the upper triangular matrix U is well known to be the same as K in (6.4).

After performing the decomposition the solution is obtained by forward and backward substitution procedures, i. e. a vector P^* is obtained by solving the equation

$$LP^* = P \quad (6.8)$$

and the solution of the original equations is the solution of

$$Uu = P^* . \quad (6.9)$$

The number of operations needed for the decomposition as well as for the substitutions depends on the structure of the sparse matrix K . The accuracy of the solution depends on the condition number of the matrix

$$cond(K) = \frac{\lambda_{max}}{\lambda_{min}} \quad (6.10)$$

where λ_{max} and λ_{min} represent the maximal and minimal eigenvalues of the matrix K .

The stiffness matrices for FEM are sparse, i.e. most of their coefficients are zero. In most cases they are also symmetric. If the width of the band can be reduced by properly numbering the nodes, computational costs can be considerably reduced. The size of memory needed can be reduced if only the non-zero coefficients are saved. But obviously the programming work becomes more complicated.

6.1.2 Iterative solver using preconditioned conjugate gradient method

In comparison with the direct solution methods, there are many iterative methods of solution.

The simplest one is the Jacobi method which rewrites equation (6.1) as

$$u_k^{i+1} = \left(P_k - \sum_{\substack{j=1 \\ j \neq k}}^n K_{kj} u_j^i \right) / K_{kk} \quad (6.11)$$

and iterates until the given tolerance is reached.

The computing costs for an iterative solving procedure depend not only on the size of the set of equations and the structure of the matrix but also on some other factors. These factors are: the formulation of the algorithm, the condition number of the matrix and the convergence tolerance for the iteration. There are many different methods to accelerate the convergence.

When the penalty method is used, the stiffness matrix remains positive definite, i.e.

$$v^T K v > 0 \quad \text{for } v \neq 0 .$$

This property makes it suitable to use the conjugate gradient method. This method is efficient because it takes the conjugate gradient of the last iteration into account for the estimate of the searching direction. For a set with N equations the procedure must reach the exact solution in a maximum number of N iterations as long as the matrix is not singular.

There are many variations based on this method, e. g. Preconditioned Conjugate Gradient Method(PCG), Idealized Generalized Conjugate Gradient Method(IGCG), Orthogonal Error Method (OEM), Hermitian Krylov Projection Method (HKPM) and so on [KI90] [BA89]. Some of them are also able to solve unsymmetrical systems. For a symmetrical positive definite system the PCG method is a suitable one.

The procedure for iterating (6.1) consists of the following steps:

- 1: $\hat{r}^k = M^{-1} r^k$,
- 2: $\gamma^k = (r^k)^T \hat{r}^k$ and $\beta^k = \frac{\gamma^k}{\gamma^{k-1}}$ ($\beta^0 = 0$) ,
- 3: $p^k = \hat{r}^k - \beta^k p^{k-1}$,
- 4: $\alpha^k = \frac{\gamma^k}{(p^k)^T K p^k}$,

$$5: \quad \mathbf{r}^{k+1} = \mathbf{r}^k - \alpha^k \mathbf{K} \mathbf{p}^k ,$$

$$6: \quad \mathbf{u}^{k+1} = \mathbf{u}^k + \alpha^k \mathbf{p}^k .$$

Here \mathbf{M}^{-1} is the matrix used for the preconditioning and $\mathbf{r}^k = \mathbf{P} - \mathbf{K}\mathbf{u}^k$ is the residual at the k-th step. An arbitrary initial guess \mathbf{u}^0 is assumed to be given, e.g. $\mathbf{u}^0 = \mathbf{0}$.

There are many methods available for choosing the preconditioning matrix \mathbf{M} . The method used in [BA89] is adopted in the present work. The diagonal matrix

$$\mathbf{M} = \mathit{diag}(\mathbf{K}) \tag{6.12}$$

is used for \mathbf{M} . In this way the preconditioning reduces to a simple vector multiplication.

Besides efficiency of performance, another advantage provided by the conjugate gradient method is the simplicity of writing code for vectorization and parallelization. In the conjugate gradient method most operations are multiplications of matrices and vectors. The dimensions of the matrices as well as of the vectors are known in advance. The vectorization is easily achieved. Furthermore, the Element-By-Element (EBE) method may be used, meaning that the total stiffness matrix is not assembled. The matrix-vector multiplications $\mathbf{K}\mathbf{p}^k$ are calculated at the element level and added together to give the result.

Investigations have shown that the number of operations and the size of memory needed for the EBE method are proportional to the number of elements. In contrast, the number of operations and the size of memory needed for the Gaussian direct solver are much higher, usually about the square of the number of degree of freedom.

6.1.3 Assessment of different algorithms using a computational example

The behaviour of different solution methods can be best seen with the help of a computational example.

It should be noticed that not all examples give the same result. They can vary as the boundary conditions and the initial values change. The example given here provides only a general idea showing how the different methods work.

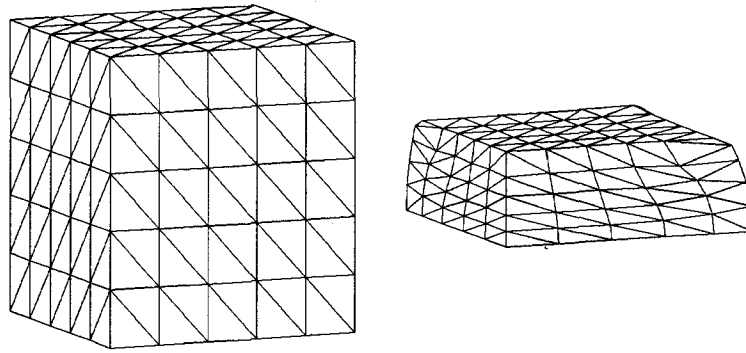


Fig. 6. 1 Example to test different solution methods.

A cubic block which is compressed to 50% of its original height is taken as the example. Friction is assumed between the workpiece and the die. By subdividing the elements it is possible to change the number of degrees of freedom. The total simulation is divided into 100 increments. The computations are carried out on an SGI 4400.

The results are given in Table 6. 1.

Table 6. 1 The CPU time of simulation using different solution methods (min./Inc.)

Degrees of freedom	Direct solver using LU decomposition method	Iterative solver of conjugate gradient method
192	0.0079	0.0061
648	0.065	0.037
3000	1.05	0.47
6591	7.5	1.75
12,288	51	4.70

The data listed in the table give the total CPU times used in each increment, including assembling, solution and recovery. The differences are caused solely by the solution procedures. The table shows that when the system is relatively small, the difference between these two methods is not very significant. As the number of degrees of freedom increases the direct solving procedure performs badly in comparison to the conjugate gradient iterative method.

The bad performance of the direct solver is due to the so-called fill-in, meaning that many zero coefficients of the original matrix K become non-zero in the decomposed matrix K'

Optimization of the node numbering can increase efficiency. But the improvement is limited and the program becomes complicated.

The computing time for an iterative solver depends on the condition number of the equations. If the equations are well-conditioned, the solution procedure can reach the convergent solution with few iterations. If the penalty factor for the incompressibility condition is too large, it may cause trouble in the solution procedure, because it increases the condition number. The choice of a reasonable penalty factor was discussed in previous chapter.

As mentioned before, the CPU time used by the iterative solver depends also on the convergence tolerance

$$e = \frac{(\mathbf{r}^i)^T \mathbf{r}^i}{\mathbf{P}^T \mathbf{P}} \quad (6.13)$$

prescribed by the user. Taking the previous example with 3000 degrees of freedom and using different tolerances for the solution procedure we obtain the comparison shown in Fig. 6. 2. No differences in the results but different CPU times are observed.

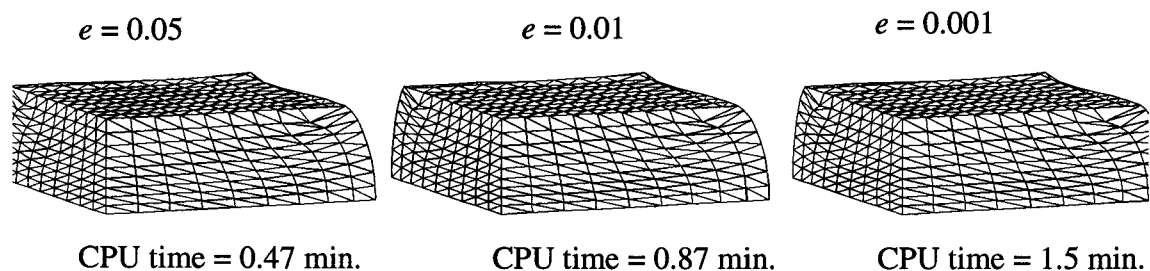


Fig. 6. 2 Simulation with different tolerances for the equation solver

For nonlinear problems the iteration loop to determine the nodal displacement increments is anyway necessary. No matter how accurate the solution of the equations is, it may be still far from the true solution of the problem. It is therefore advantageous to specify a relatively large tolerance e . In PressForm a default value of e is set to 0.05 and satisfactory performance is obtained. Further increasing of the factor does not make much sense, because although the solver might need less time, more iterations for the increments are necessary.

6.2 Description of friction

Friction is an important factor in forming processes. It influences not only the geometric form of the products but also the distribution of deformations. Although some processes like rolling cannot be carried out without friction forces, efforts are often made to reduce frictional forces as much as possible in order to obtain a homogeneous deformation state.

From the physical point of view friction is a very complex phenomenon. Various models have been developed to describe friction and evaluate frictional forces. Friction depends on many factors, the most important being the pressure between the contact surfaces, lubrication condition, smoothness of the surfaces, temperature, material properties etc.

Although establishing a friction model is beyond our scope, frictional forces have to be reasonably evaluated in the simulation of forming processes. We do not intend to involve all the factors to describe friction, neither is it possible for users from industry to determine many parameters by complicated experiments. We seek a simple but effective description to calculate the frictional forces.

6.2.1 Friction in bulk forming

According to Coulomb's law the friction force is proportional to the pressure between two bodies,

$$\tau = \mu p \quad . \quad (6.14)$$

The coefficient may vary according to different environments. The directions of the friction forces are always opposite to the direction of the relative movement of the two contact surfaces. This means that friction always tends to prevent relative movement.

In the forming industry the workpiece is deformed under external forces. The pressure applied on the workpiece is very high especially in bulk forming processes. The surface of the workpiece changes continually, the temperature rises and the lubrication deteriorates at high temperature or high pressure. Therefore, friction in bulk forming processes is a very complex phenomenon.

It is clear that the friction force on the surface of the workpiece is one component of the stress tensor which determines the local yield state. If the von Mises flow rule is used, the

yield condition is

$$2\sigma_Y^2 = (\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2 + 6(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2) \quad (6.15)$$

where σ_Y denotes the yield stress.

Assuming that the shear stress τ_{xy} is the friction force per unit area in the x-direction, then in the extreme case when all other deviatoric stress components are zero, i. e. $\sigma_x = \sigma_y = \sigma_z$; $\tau_{yz} = \tau_{zx} = 0$ we obtain

$$\tau_{xy} = \frac{\sigma_Y}{\sqrt{3}} \quad (6.16)$$

If other components are not zero, the friction force must be smaller. A factor m is often employed to describe the friction force as

$$\tau_{xy} = \frac{m\sigma_Y}{\sqrt{3}} \quad (6.17)$$

The pressure applied on the surface of the workpiece can be much higher than the yield stress in bulk forging processes. For example, the extrusion pressure may be as much as 10 times higher than the yield stress in an extrusion. Coulomb's law may overestimate the friction forces although the frictional coefficient may seem to be reasonable. Coulomb's law is then no longer suitable under these circumstances.

The factor m may vary between 0 and 1 according to different stress states. Because the deviatoric stress components different from τ_{xy} are seldom zero, $m < 0.5$ seems to be a reasonable assumption. Numerical experiences also show that when $m > 0.5$, instabilities may appear during the computation and special measures have to be taken to prevent them.

6.2.2 Modelling of friction

As mentioned before, we are not going to take into account all the factors to describe the friction. The assumptions made are

- Coulomb's law is valid when the pressure is not high otherwise the shear model is applied;
- Both the frictional coefficient μ and the shear factor m are constant during forming processes. They are to be determined by experiments.

Considering the surface of the workpiece which is in contact with the tools, there are three different cases. The first case is when the material particles are stuck to the surface of the tool. In this case the friction may be quite different. If the particles have a tendency to move on the surface of the tool the friction force must be large enough to prevent the relative movement. In contrast, if the particles have no tendency to relative movement, no frictional forces should be induced.

The second case is when the material particles exhibit relative movement on the surface of the tools but the pressure is low. The friction forces evaluated according to Coulomb's law are less than the forces predicted by the shear model. In this case Coulomb's law can be used and the directions of the frictional forces can also be determined according to the relative movement.

As the pressure on the surface increases, the value calculated according to Coulomb's friction exceeds the value from the shear model. In this case the shear model should be used. The magnitudes as well as the directions of the friction forces are applied as external forces. Fig. 6. 3 shows the three cases of friction.

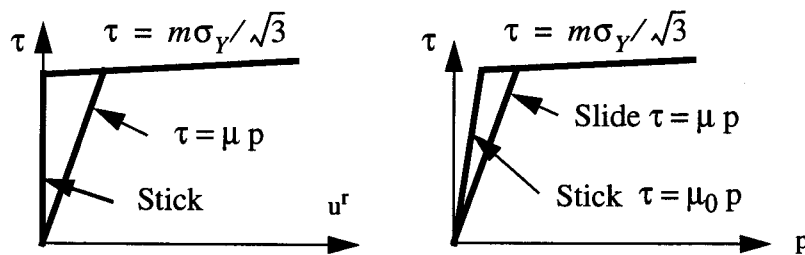


Fig. 6. 3 Different cases of friction

If there is no relative movement a static frictional coefficient μ_0 is used when Coulomb's law applies. The kinematic frictional coefficient μ is substituted as the relative movement appears. Both have the same limit set by the shear model.

The direction of the friction forces is opposite to the relative movement. It is clear that if there is no relative movement, both the direction and the magnitude of frictional forces are indefinite.

An expression has been introduced to evaluate the frictional forces [KO89]

$$f_{\tau} = -|\tau| \left(\frac{2}{\pi} \operatorname{atan} \frac{|v_r|}{A} \right) \frac{v_r}{|v_r|} \quad (6.18)$$

Similarly, we used the exponential function as

$$f_{\tau} = -|\tau| \left(1 - \exp \left(-\frac{|v_r|}{A} \right) \right) \frac{v_r}{|v_r|} \quad (6.19)$$

Both expressions vanish as the relative velocity goes to zero. A comparison is given in Fig. 6. 4. It can be seen that (6.19) approaches the actual frictional forces more quickly when the relative velocity increases. The value A signifies the relative velocity where 63.2% of the frictional forces should be applied.

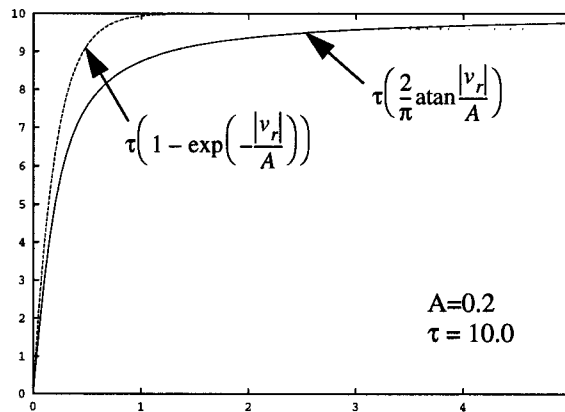


Fig. 6. 4 Two expressions of the frictional force

6.2.3 Measures to stabilize the computation of friction

As mentioned in the previous sections friction is difficult to evaluate because of its complex physical properties as well as the kinematic conditions. If the frictional forces are not properly applied, the results may be unreliable and numerical instabilities might arise.

Fig. 6. 5 shows an example of the simulation of a forging process. The bottom part has already been filled so that the material should flow up to fill the upper flange. If the

frictional coefficient is made too large, numerical instability occurs. The surface nodes change the flow direction every increment causing local whirling, which can never happen in reality.

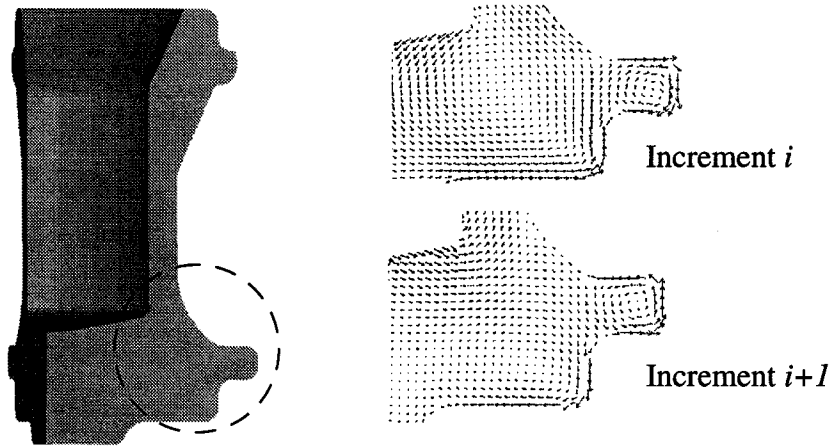


Fig. 6.5 Problem of instabilities caused by too high a frictional coefficient.

This problem is caused by an overestimation of the frictional forces. It is obviously not accurate enough to calculate friction using the simple model. However frictional forces are never active forces, since friction is induced as the relative movement begins and decreases rapidly as the relative movement vanishes. It can only slow down the movement but never cause the material to move in the opposite direction. In other words, the frictional forces exhibit a strong non-linear property if the movements of the nodes are obstructed by friction.

Experience shows that it is not good practice to iterate over the frictional forces. If the friction forces are checked by means of a convergence criterion, the process might take much more computing time but never reach the convergent solution because friction is not involved in the matrix for the iteration. If the convergence criterion is set for the global reaction forces and residual forces, the convergence of the process may be better but the results are not greatly improved because the frictional forces are always small in comparison with the forces applied to the workpiece. The local instabilities caused by friction may exist in spite of the fact that global convergence is reached.

Nevertheless, special measures must be taken to prevent instabilities. Investigations show

that the problem is caused by applying the frictional forces as external forces on the surface of the workpiece. This method is valid as long as the movements of the material particles are not reversed. If the material flow changes to the opposite direction, the friction should pass through a zero value no matter how small the increment is.

Along these lines an improvement was made. The velocities of the surface nodes in the last increment are saved. If the scalar product

$$\alpha = (v^r)_i^t (v^r)_i^{t-\Delta t} \quad (6.20)$$

is larger than zero, the friction forces are applied completely at the node. A mean relative velocity

$$v_i^r = 0.5 [(v^r)_i^t + (v^r)_i^{t-\Delta t}] \quad (6.21)$$

is also used to take the previous state into account. If

$$\alpha = (v^r)_i^t (v^r)_i^{t-\Delta t} < 0 \quad , \quad (6.22)$$

the frictional forces applied in last increment are clearly too large. The friction force is then reduced to half its value.

Another way to stabilise friction is to set a reasonable value for the parameter A in (6.19). The smaller A is, the more quickly the frictional forces increase as relative movement starts. Because the most serious problem arises when the relative velocity is small, a relatively large A is favourable. The default value in PressForm is

$$A = 0.2 |v^r|_{max} \quad . \quad (6.23)$$

With these measures the friction is well taken care of. The upsetting of a truncated cone with a small conicity is taken as a test example. Because of the symmetrical condition only a quarter is computed. Fig. 6. 6 shows the results of the simulation. The form as well as the distribution of the equivalent strain obtained from the frictionless condition is quite different from that with high friction. Although rather high friction ($m = 0.7$) is applied, which corresponds to forming at high temperature with poor lubrication, no numerical instability occurs.

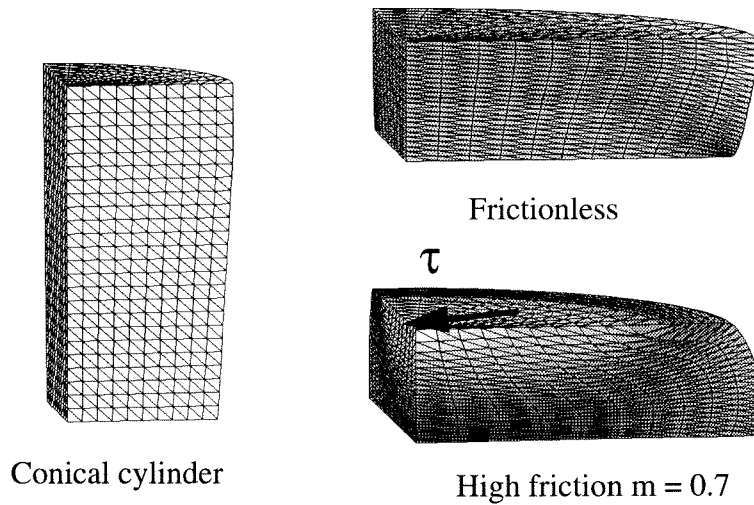


Fig. 6. 6 Upsetting of a truncated cone

6.3 Estimate of the initial values for the iteration

In FEM for nonlinear problems a set of equations

$$\mathbf{K}_T \delta \mathbf{u} = \mathbf{R} \tag{6.24}$$

has to be solved, where \mathbf{K}_T is the tangential stiffness matrix and

$$\mathbf{R} = \mathbf{P} - \mathbf{K}\mathbf{u} \tag{6.25}$$

are the residual nodal forces at the current state. In (6.25) \mathbf{P} is the vector of applied forces and $\mathbf{K}\mathbf{u}$ denotes the vector of internal nodal forces. The solution of equation (6.24) $\delta \mathbf{u}$ is updated to the total solution of the problem \mathbf{u} until the residual forces are small enough.

It is certainly advantageous to give \mathbf{u} initial values so as to make the residual as small as possible. Fewer iterations and quick convergence can be expected.

6.3.1 Stationary and quasi-stationary cases

In a forming process like extrusion and rolling, the material flow is stationary or quasi-stationary. The velocities of the nodes change only slightly especially in the zones where the Eulerian mesh is used. It is clear that the solution of last increment should be taken as an estimate of the current increment.

6.3.2 Non-stationary material flow

The situation is more complicated if the material flow is non-stationary. Although most of the nodes change their velocities gradually, a sudden local change of nodal velocity is not excluded. If the previous velocities are taken as the estimate, trouble may arise from the contact searching.

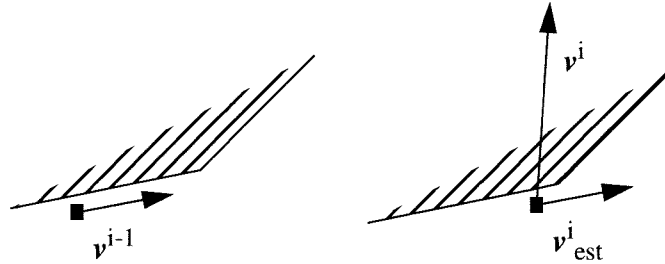


Fig. 6. 7 Problem caused by incorrect estimation

Fig. 6. 7 shows the problem schematically. A node flows along the surface of the tool during the (i-1)-th increment. If the old solution is taken as the initial value for the next increment, the node may no longer fulfil the contact condition and become a free node. Under the pressure from the internal medium, which is usually high in bulk forming processes, the solution for this node will be totally wrong. If the error cannot be corrected in the next increment, the simulation will produce meaningless results.

A useful way to get the initial values is first to calculate the system as linear elastic and take the elastic solution as the initial value for iteration purposes. The rigid-plastic description is equivalent to the linear elastic model if we set

$$\frac{2\sigma_v}{3\Delta\epsilon_v} = \frac{E}{(1+\nu)} = \text{const} \quad (6.26)$$

and the penalty factor

$$\kappa = \frac{3E}{(1-2\nu)} = \text{const} \quad (6.27)$$

Therefore, it is very convenient to obtain the elastic solution as the first estimate. This method works well. The reason is, despite the differences in the material properties, the velocity field may still have some similarities as long as the boundary conditions are the

same. Fig. 6. 8 shows the typical convergence curves by using different methods to estimate the initial values.

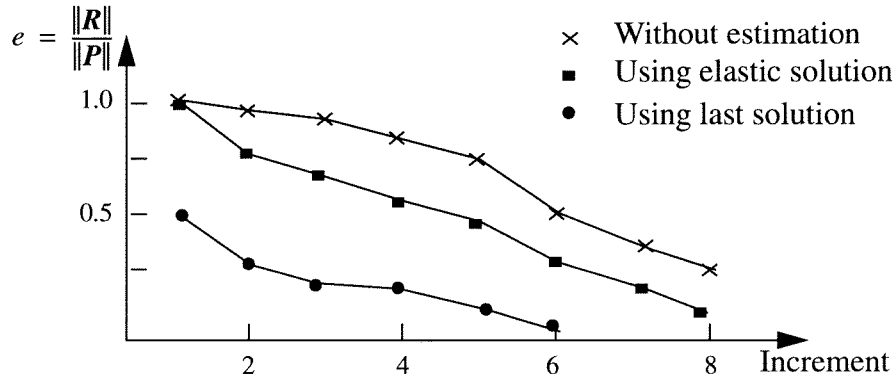


Fig. 6. 8 Convergence rate using different initial values

6.4 Criterion to determine the increment size

It is also very important to adjust the size of the increment to get a stable solution for nonlinear problems. Because we calculate the gradient at the element level, we use the minimum length of the element as the criterion to determine the size of increment.

Suppose the minimum length of the element sides is ΔL . A small time increment Δt^{tri} is used to start the calculation. After iterating, the time increment is adjusted according to the ratio between ΔL and the maximum displacement of the nodes $|\Delta \mathbf{u}|_{max}$

$$\Delta t = \alpha \Delta t^{tri} \frac{\Delta L}{|\Delta \mathbf{u}|_{max}} \quad (6.28)$$

By setting $\alpha = 0.5$, a good estimate is obtained and the simulation is carried out without the problem caused by unsuitable time increments.

7. Numerical Tests of the FE Program PressForm

Based on the principles discussed in previous chapters a special purpose FE program has been developed, whose main features are a flexible ALE formulation, efficient performance and user friendly interfaces. Even with only a little basic knowledge of FEM, it is possible to use the program after a short introduction.

The program is especially suitable for the simulation of the profile forward extrusion process. Not only the mesh for the workpiece and the discretization of the die and punch are automatically generated, but also adaptive remeshing is performed during the simulation. The program can also be extended to the simulation of other bulk forming processes.

In order to verify the validity of the formulation as well as the program itself, several computations have been carried out. The results are compared either with experiments or with the commercial FE program MARC. Two examples with known analytical solutions are chosen to check the results of thermal analysis.

7.1 The structure of the program

Like most FE programs, PressForm consists of different modules as shown in Fig. 7. 1.

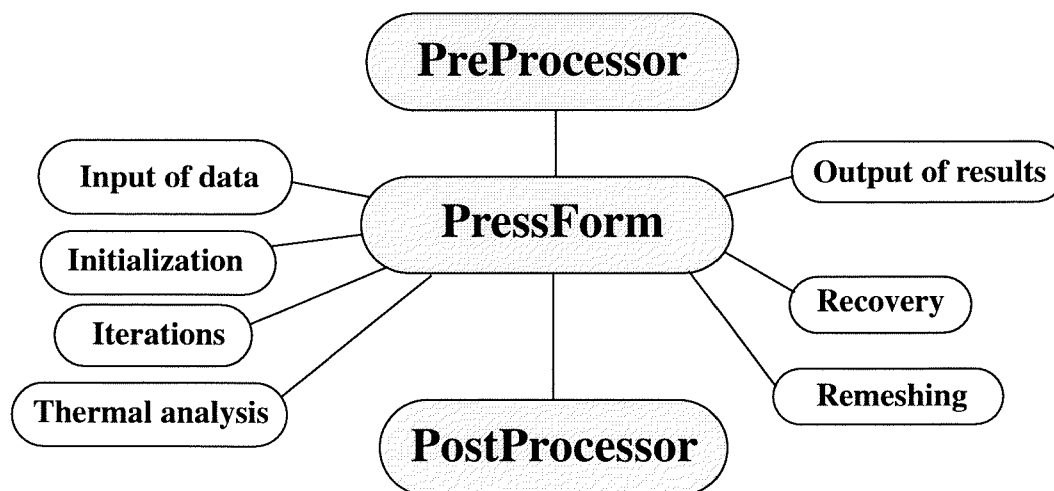


Fig. 7. 1 Structure of PressForm

The main part of the program deals with equilibrium iteration for nonlinear problems, which is performed according to the diagram shown in Fig. 7.2.

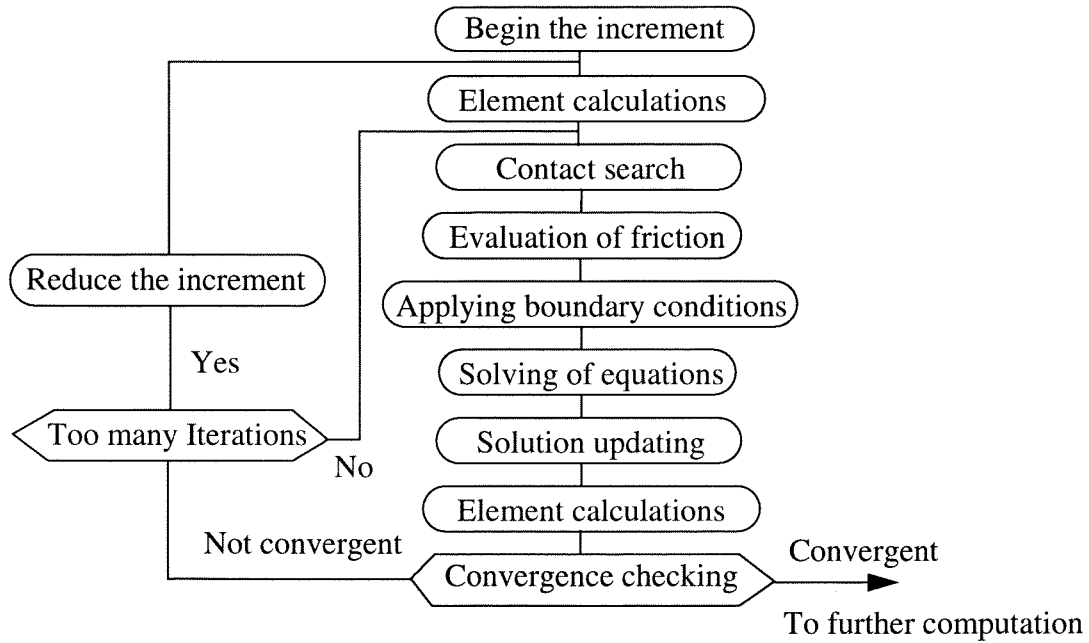


Fig. 7.2 Diagram for the iterative solution of nonlinear problems

The modules are called repeatedly according to the definitions of the problem.

7.2 Forward extrusion of cross profiles

The first example used to test the program is the forward extrusion of cross profiles. Two profiles of different size are chosen both for simulation and for the experiments. The dimensions are shown in Fig. 7.3.

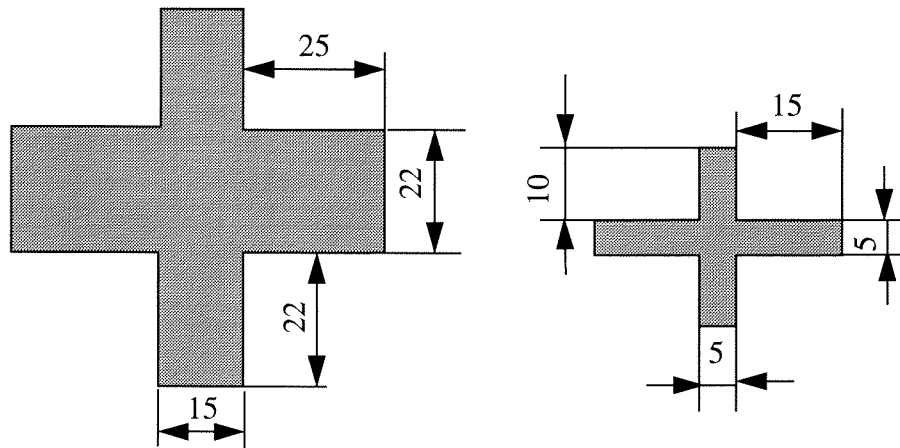


Fig. 7.3 The dimensions of the cross profiles

The material used in the experiment is CuZn39Pb3. The initial forming temperature is 700° C. At such high temperature the material exhibits strong rate dependent properties. The yield curve is assumed to be a function of strain rate and the temperature. It can be

approximated as

$$\sigma_Y = 32.2 (\dot{\epsilon}_v + 0.01)^{0.17} \frac{450}{T - 250} \quad (7.1)$$

The form is shown in Fig. 7.4.

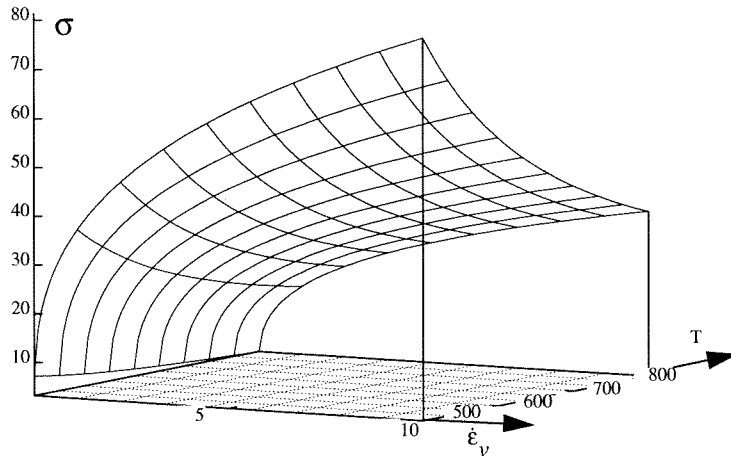


Fig. 7.4 Flow stress in function of strain rate and temperature

The initial mesh and the results obtained from the simulation are shown in Fig. 7.5. Good agreement is shown with the experiment. The description of material flow, therefore, is reliable.

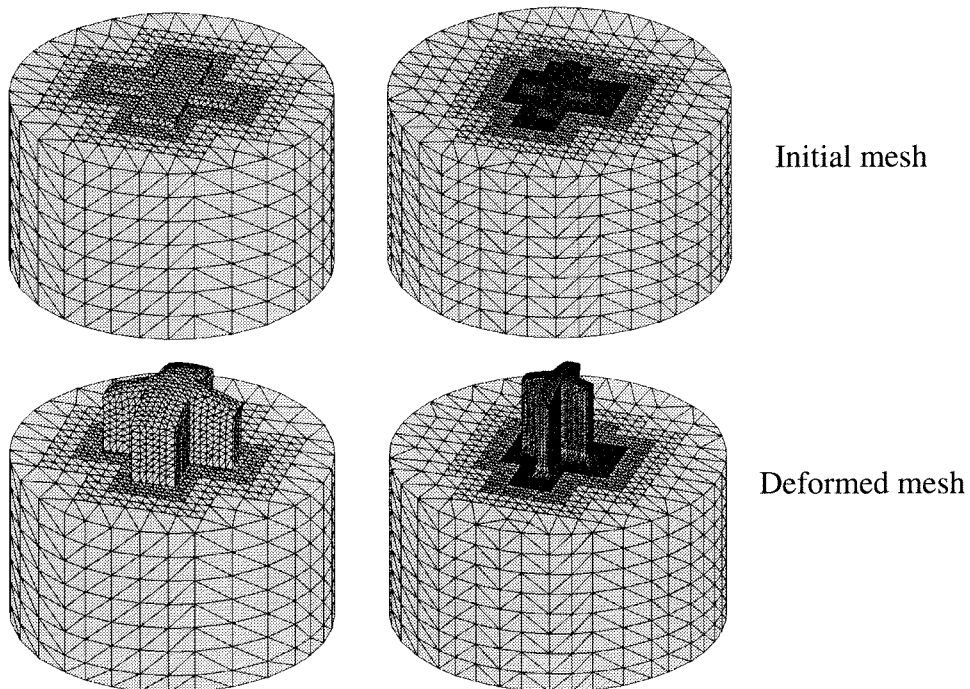


Fig. 7.5 Simulation of cross profile extrusion

Another aspect which is verified by this test simulation is the convection effect. In the deformation zones where the Eulerian mesh is used, the nodes are not updated. The physical variables such as equivalent strain and temperature are transported by the convection to the Lagrangian zones where the deformation has practically finished. With the numerical example, it is clear from Fig. 7.6 how the transportation is performed. The high strain rate zones are localized around the radii at the very beginning. Gradients are induced inside the domain. The equivalent strain is then transported by convection to the neighbouring elements. Although the strain rate here is still small, strain is accumulated. As only one half of the work piece is simulated, the effect is well seen on the plane of symmetry.

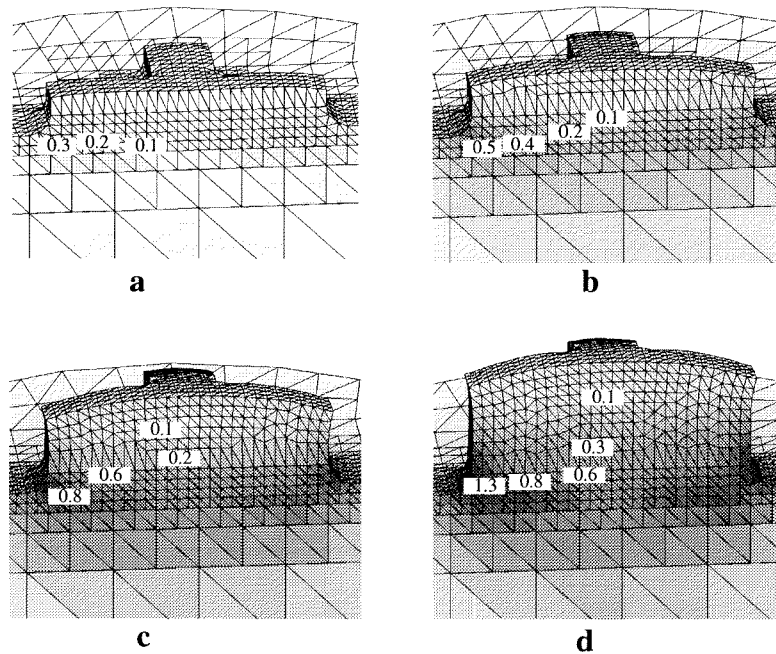
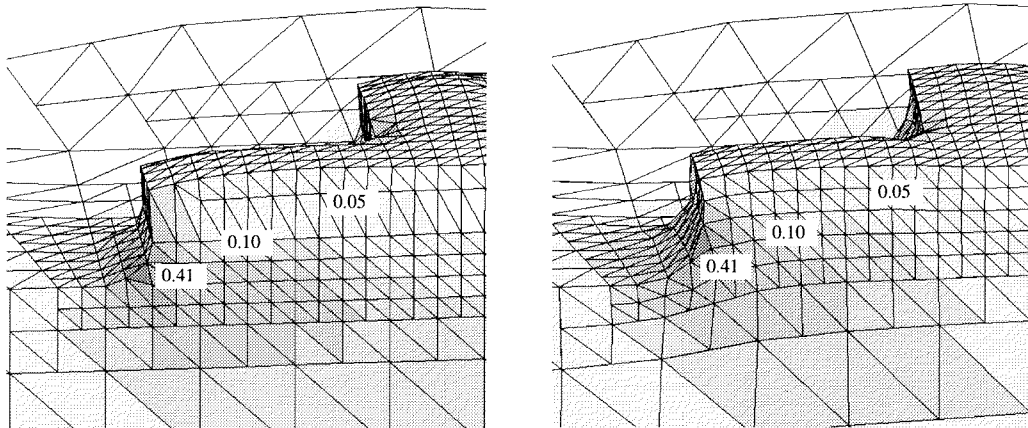


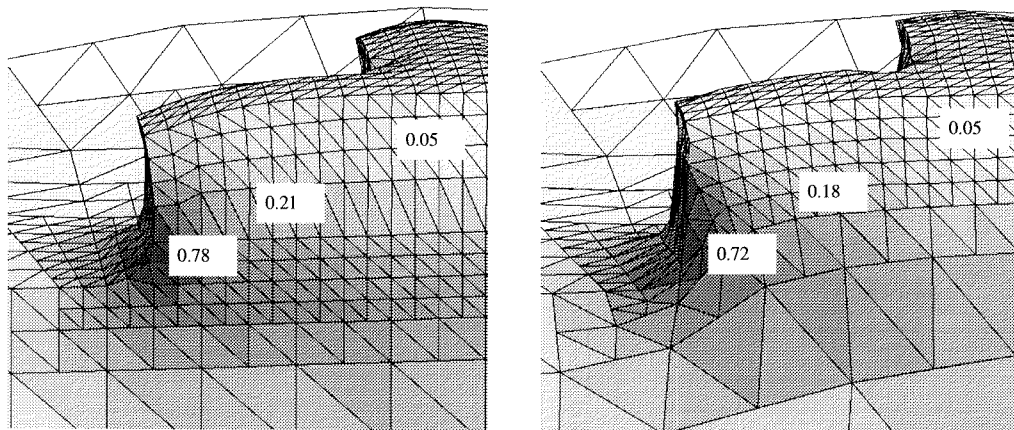
Fig. 7.6 Transportation effect of convection

If the process is calculated with the updated Lagrangian method (ULM), because the elements always contain the same material medium, the accumulation of plastic strain is performed naturally, and the results are more accurate. In contrast, the Eulerian method forces the nodes to be stationary and updates the time history variable ϵ_v according to the velocities and the gradients. Numerical error may exist in the evaluation. Fig. 7.7 shows the comparison of both methods. Good agreement is found in the first few increments.

The difference increases during the computation. This is partially due to the error introduced by the convective term, and partially to the distorted mesh used in the updated Lagrange description. The simulation using the updated Lagrangian method breaks down because of the distortion of the mesh. There is not enough evidence to make a general evaluation of both methods. However, the results obtained by both methods are reasonable and acceptable.



a) Increment 3



b) Increment 10

ALE formulation

ULM

Fig. 7.7 Comparison of ULM and ALE methods

Another significant difference between these two methods concerns the CPU time (Fig. 7.8). The ALE method is more efficient. The figure shows the CPU time at different increment. Actually, because smaller time steps have to be used by the updated Lagrangian method, the ALE method is almost twice as fast in this example.

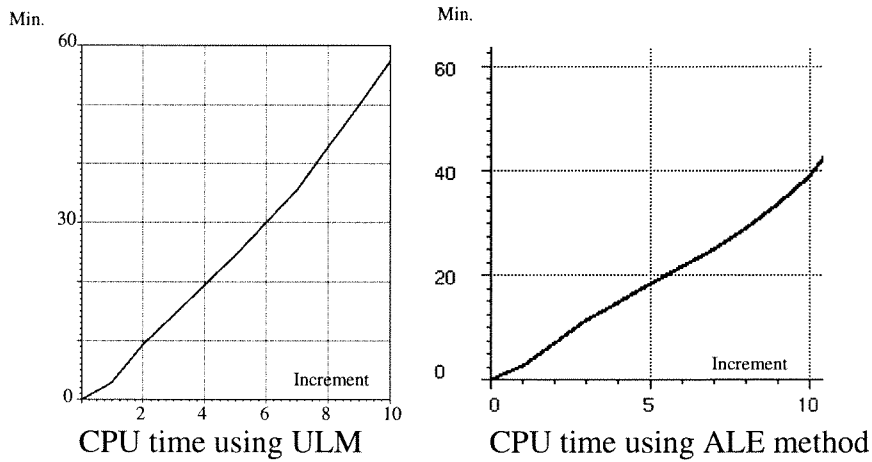


Fig. 7.8 CPU times of the two different methods

7.3 Simulation of extrusion with two openings

To test the capability of the program to predict different flow models the extrusion process with two openings was simulated. By having different dimensions and positioning, different material flow models are expected. The program MARC is used as a reference.

Generally speaking, the material tends to flow in the opposite direction, where the gradient of pressure is the largest. If there are two openings on the die, the material may flow out from the two openings with different velocities.

The dimensions and geometries of the die are shown in Fig. 7.9. Because the axisymmetric element is used, they are actually a thick wall tube with a rod in the centre. Two different arrangements are computed.

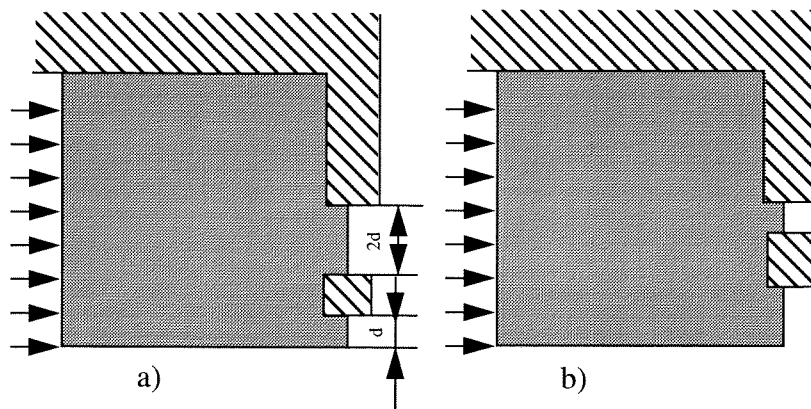


Fig. 7.9 Axisymmetric extrusion with two openings

The velocity fields in the extrusion direction from PressForm and from MARC are shown in Fig. 7.10. An axisymmetric element with the updated Lagrange formulation and rigid-plastic description is used in MARC. Despite a different element formulation and thus some deviation in boundary conditions, both programs deliver similar results. In the arrangement a) the material flows with almost the same velocity through both openings. If the openings are arranged like b) the material flows with higher velocity through the hole in the middle.

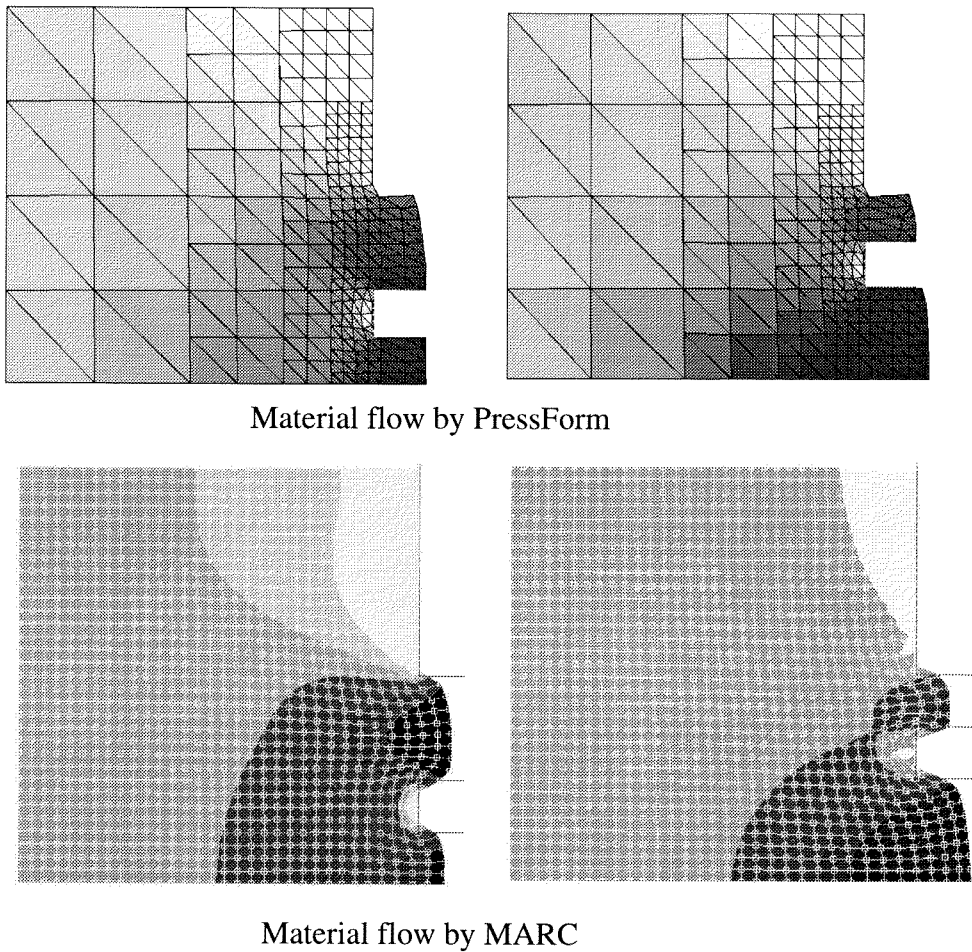


Fig. 7.10 Comparison with the FE program MARC

The computation with program MARC, however, broke down after a couple of increments. In contrast, PressForm is especially suitable for the simulation of this process and the computation proceeds smoothly until stationary or quasi-stationary states are reached. Fig. 7.11 shows the results from further computation of PressForm.

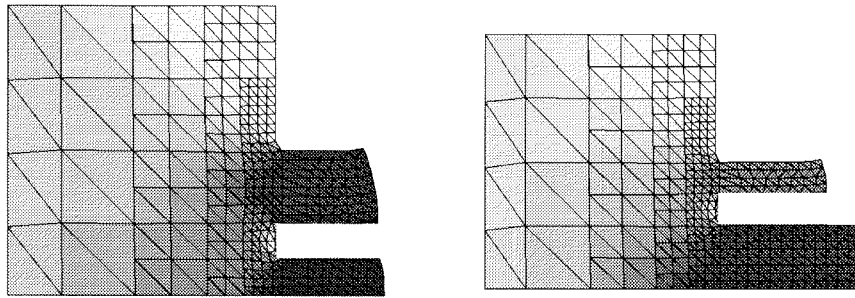


Fig. 7.11 Further simulation using PressForm

7.4 Benchmark test for a hot forging process

A benchmark test for the simulation of a hot forging process of an axisymmetric aluminium alloy part was defined for the conference *Metal Forming Process Simulation in Industry* in Baden-Baden 1994. It provided us with an excellent opportunity to test the program PressForm.

The geometry of the part as well as the description of the yield curve can be found in [HE94] of this conference. The forming procedure and the forming forces are compared with industrial measurements. The velocity fields, distributions of strain and temperature at certain stages were also demonstrated by different FE programs. Although it has not yet been possible to measure these quantities experimentally, a comparison among the participants of the benchmark test proved to be very useful.

The problem is shown schematically in Fig. 7.12. The characteristics of the form of the workpiece at different forming stages were presented at the conference.

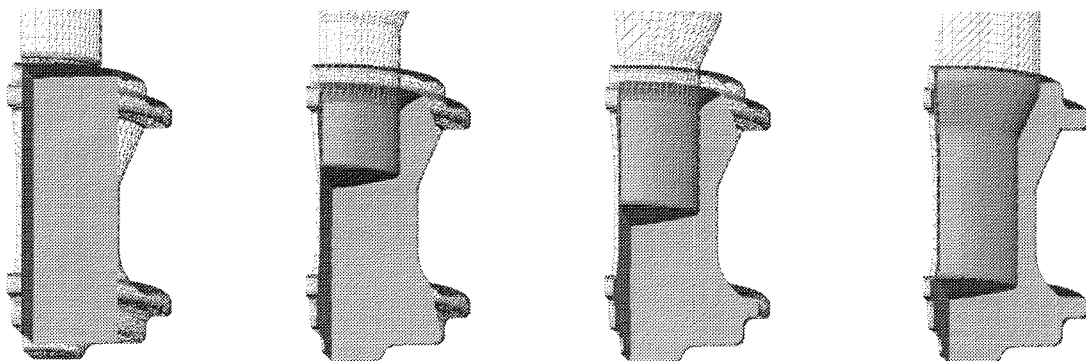


Fig. 7.12 Benchmark test for the FE simulation of an axisymmetric aluminium part

The first stage is when the punch stroke $H = 30$ mm. The indication from the experiment is that the radius at the bottom is still not completely filled.

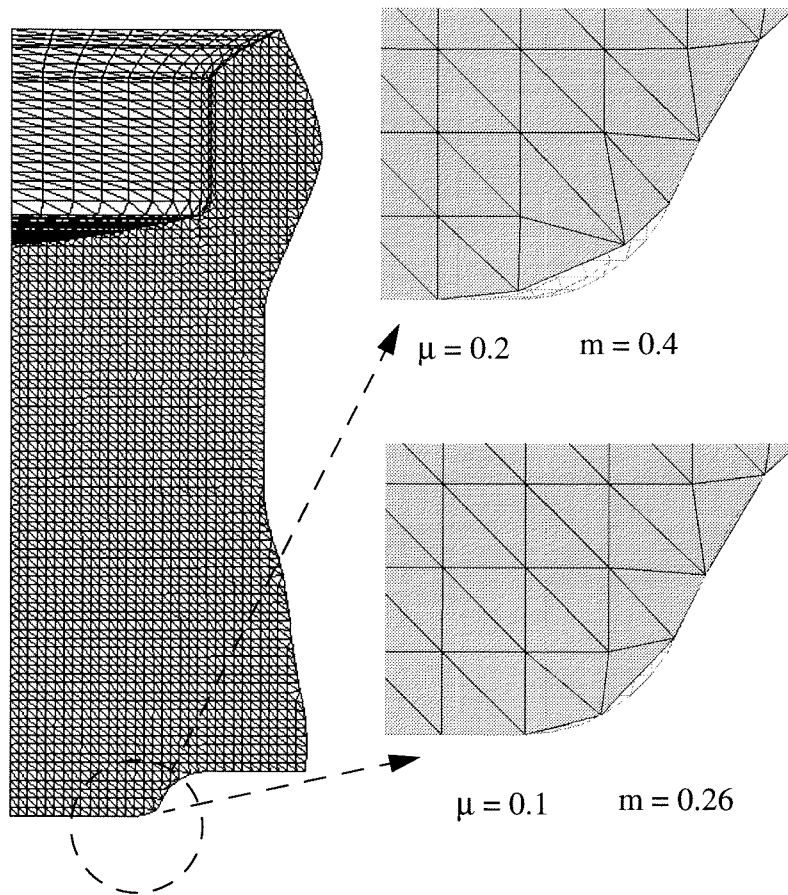


Fig. 7.13 First stage of the benchmark test of Baden-Baden

When we used the frictional coefficient suggested by the organizers of the conference, i.e. $m=0.26$, $\mu=0.1$, we obtained a filled radius at this place. During conference discussions it was pointed out that the frictional coefficient specified by the conference organizers is too small according to the measurements. Following this information, we increased the frictional coefficient to $m=0.4$, $\mu=0.2$. With these values we obtained the unfilled radius at $H=30$ mm as compared in Fig. 7.13. This example illustrates how the parameters of the simulation influence the details of the results.

The second stage is $H=60$ mm. At this stage the material should fill the lower flange and begin to flow upward. The simulation showed a very good agreement in this aspect. Fig. 7.14 shows the results at this stage

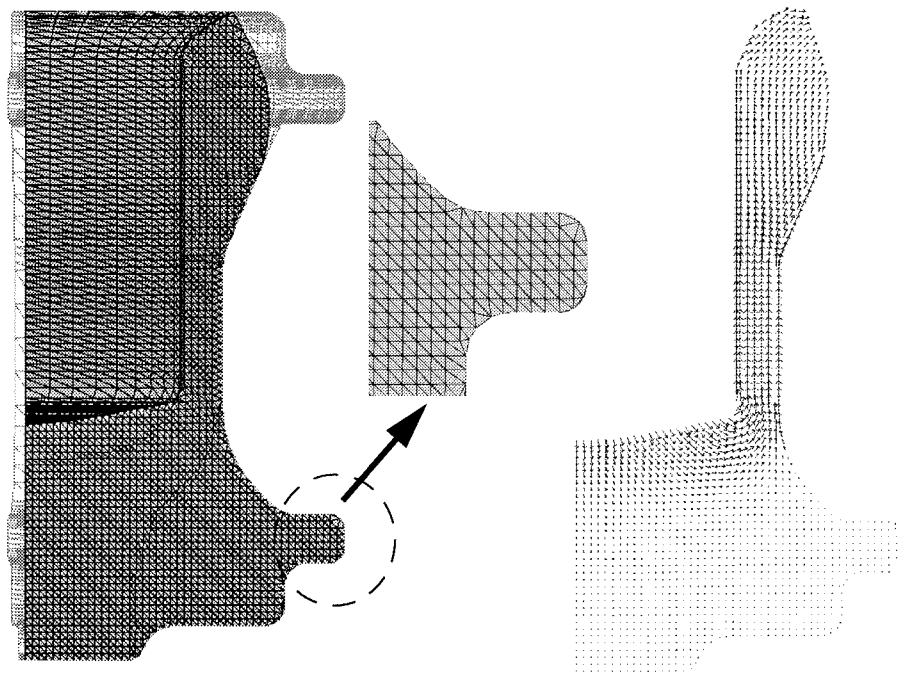


Fig. 7.14 The result at second test stage

The distributions of temperature and equivalent strain are also presented. Fig. 7.15 shows the distribution of temperature as well as the equivalent strain at the end of the forming process obtained from PressForm. Although the measured values were not available, we can compare our results with the results provided by other FE programs.

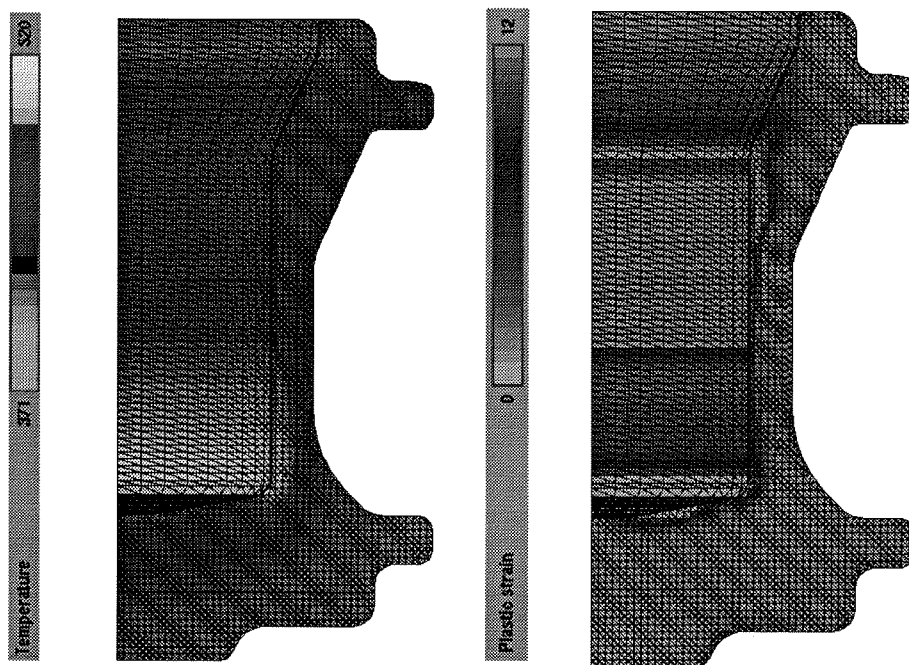


Fig. 7.15 Distribution of temperature at the end of the forming stage

Fig. 7.16 shows the comparisons of the results obtained by PressForm with those from other FE programs. Apart from the maximum value of the equivalent strain, PressForm delivers reasonable results. The high value of the equivalent strain may be due either to the friction on the surface or to the errors introduced by the convective computation. It is noticed that MARC gave almost the same high value.

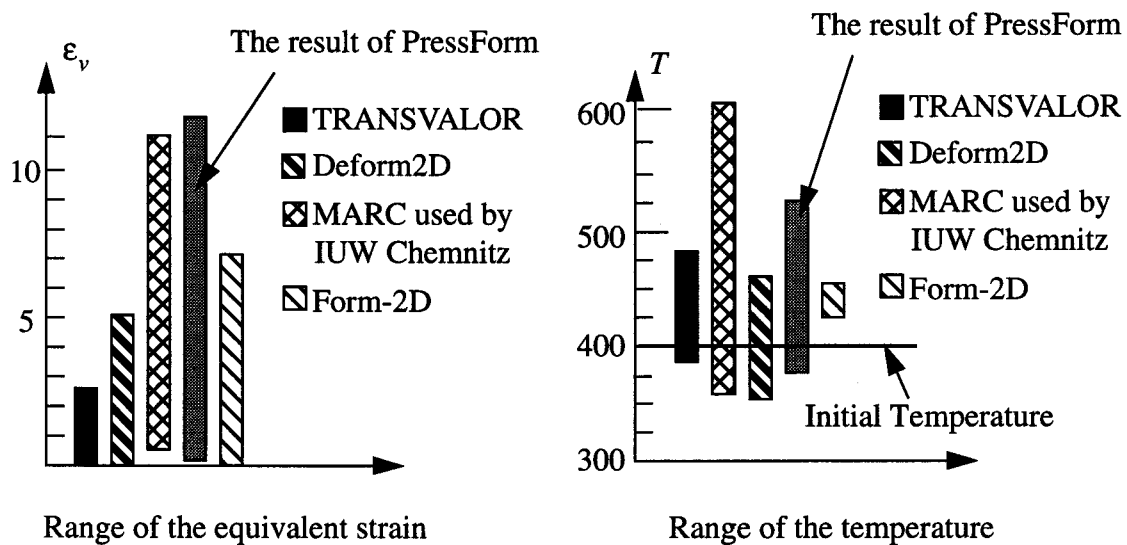


Fig. 7.16 Comparison with other FE programs

It should be pointed out that thermal analysis can be properly performed only when all the boundary values as well as the initial values are known. The results are sensitive to the computational parameters.

The forming forces obtained using different programs were plotted by the conference organizers as shown in Fig. 7.17.

The deviation of the results using MARC should be fairly interpreted. Because MARC is a general purpose package, there are a lot of options and parameters in the input data. If any of them are not properly set, the results may be totally different. This shows again the advantage of a special purpose packages.

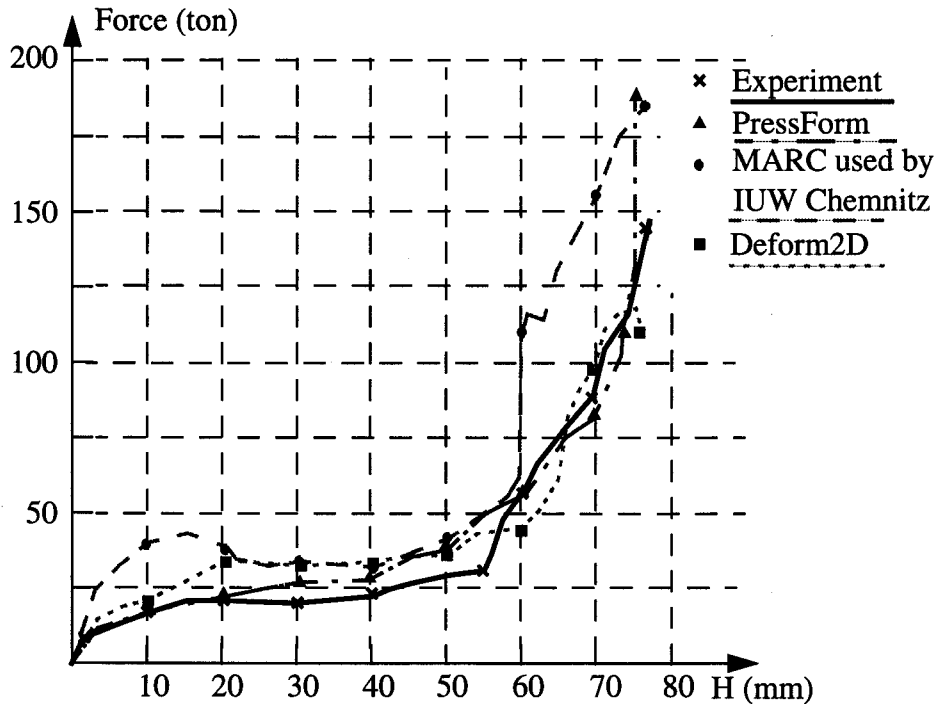


Fig. 7.17 Comparison of forming forces obtained by different FE programs

7.5 Heat conduction

Thermal analysis is also an important part of the simulation of forming processes. Especially for some non-ferrous alloys temperature is an important factor in relation to metallurgical qualities.

It is not an easy task to measure the distribution of temperature inside a workpiece. Fortunately, there are analytical solutions for several simple heat conduction problems. Here we use PressForm to calculate the thermal field of known problems and compare the results with the analytical solutions to verify the accuracy of the computation.

Suppose that there is an infinitely long rod with initial temperature of T_0 . The temperature at $x = x_0$ is suddenly increased. At time $t = 1.0$ the temperature at this place is measured $T_0 + T_1$. The analytical solution for this problem is known as

$$T = T_0 + \frac{T_1}{\sqrt{t}} \exp\left(-\frac{(x-x_0)^2}{4a^2t}\right), \quad (7.2)$$

where $a^2 = \frac{\lambda}{C\rho}$.

It is assumed that the rod is made of aluminium with the parameters $a^2 = \frac{\lambda}{C\rho} = \frac{230}{896 \times 0.0027} = 95.1$. With $T_1=200.0$, $T_0=20.0$ and $x_0=250.0$ the function is shown in Fig. 7.18.

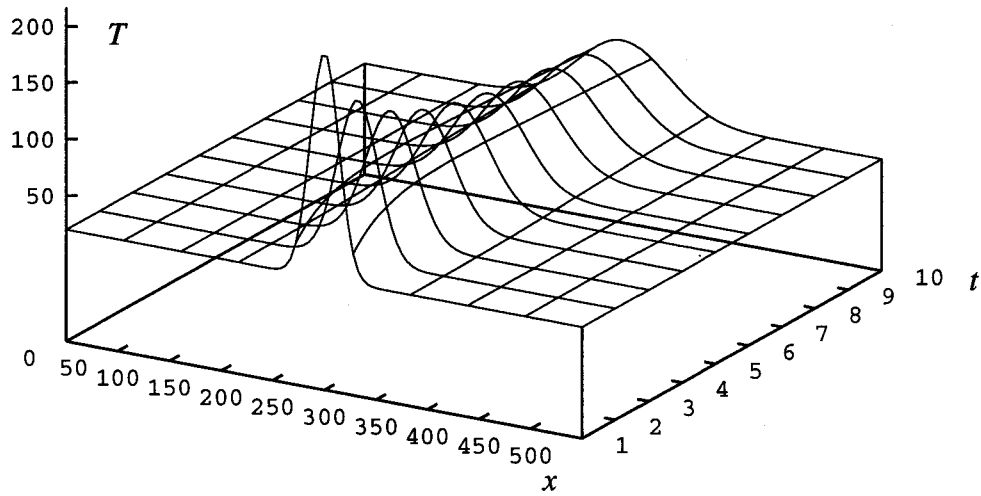


Fig. 7.18 Heat conduction due to a Dirac impulse function

This problem is simulated using PressForm. The results are compared with the theoretical solution in Fig. 7.19.

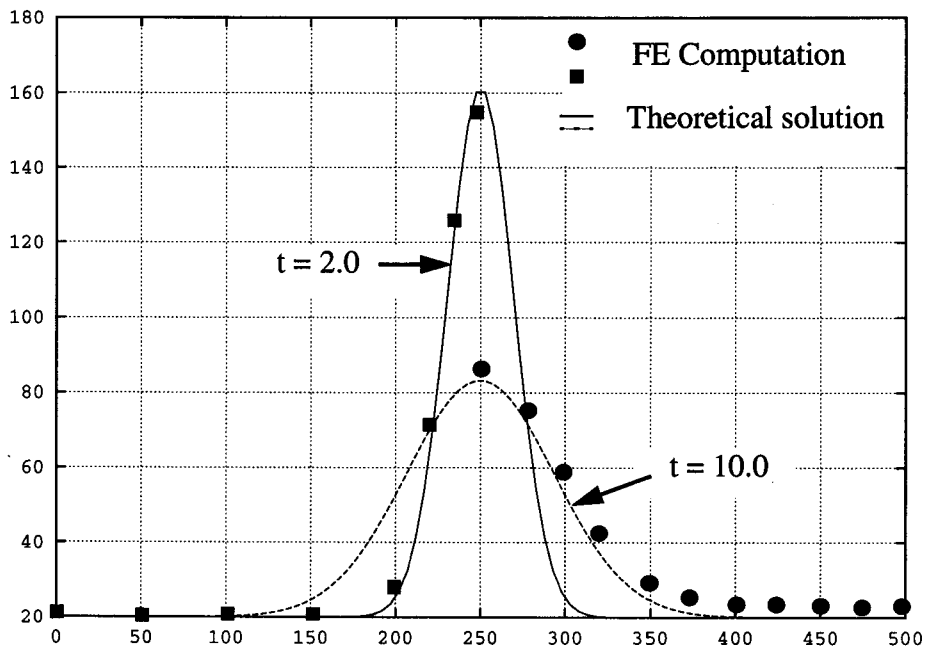


Fig. 7.19 Comparison of computation and theoretical solution

The results of the numerical computation show good agreement with the theoretical solution. The small deviation may be due to the discretization of the rod as well as the finite length of the rod used in the computation.

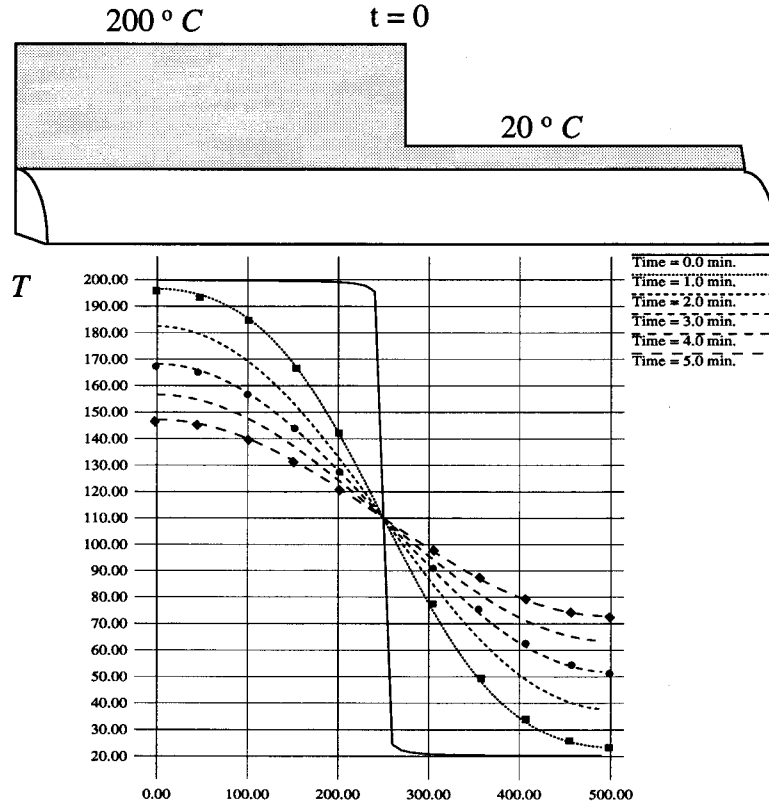


Fig. 7.20 Simulation of heat conduction in a rod of finite length

Another problem with a known solution was also calculated. Two rods with different initial temperatures are joined together at one end. An adiabatic boundary condition is applied to the other ends. The analytical solution for this problem is

$$T = b_0 + \sum_{n=1}^{\infty} b_n \exp\left(-\frac{a^2 n^2 \pi^2 t}{L^2}\right) \cos\left(\frac{xn\pi}{L}\right) \quad (7.3)$$

where

$$b_0 = T^l + 0.5T^h, \quad b_n = \frac{2T^h \sin\left(\frac{n\pi}{2}\right)}{n\pi}, \quad a^2 = \frac{\lambda}{C\rho} \quad (7.4)$$

and L denotes the length of the rod, T^l and T^h are the lower and higher temperatures,

respectively.

The results of the FE computation and the analytical solution are shown in Fig. 7.20. The results obtained from the computation agree very well with the theoretical solution. The errors are less than 1.0 degree in the whole domain and whole process.

In all the test computations we come to the conclusion that the ALE method is valid for the simulation of forming processes. It leads to reasonable results. Heat conduction is properly described and the program PressForm is able to simulate real industrial processes.

8. Examples of Application

Since our aim is to develop a special purpose FE program using the ALE formulation to meet the needs of industrial application, the program, after being thoroughly tested, was used to simulate real production processes. An FE program should be able to provide users with valuable information to improve the parameters of the process, to predict possible failures in production and to shorten the design-production cycles.

8.1 Forward extrusion process of a T-profile material

Material with different profiles is produced in the metallurgical industry in large quantities. Several forming processes such as rolling, extrusion and drawing are typical. The efficiency of extrusion processes is very high because the final form is obtained in a single step. But extrusion processes also have shortcomings. For example, the forces needed are usually very large in comparison with those of drawing processes, which deform the material in many drawing operations. The large forces needed by an extrusion process may prevent this forming method from being used. In addition, other problems might also appear if the process is not well designed.

Generally speaking, the most important factors in the extrusion process are

- not too large extrusion forces,
- well-distributed velocity in the extrusion direction,
- high precision of the product,
- good metallurgical quality.

The velocity of the punch, the initial temperature of the workpiece as well as the tools, the geometry of the die and the lubrication conditions are the main system variables controlling the processes.

Extrusion products can be divided into two classes according to the geometrical form of profiles. There are many products which can be readily produced by extrusion. Most of them have thick walls, no suddenly changing sections and are often symmetrical. In contrast, there are other products which are difficult to extrude. These products are often,

but not necessarily, complex in geometry. Profiles with abruptly changing sections or thin-walled profiles are especially difficult to produce by an extrusion process.

The thin T-profile is one such example. The geometry is quite simple but the thin wall and the different lengths of the legs make it difficult to extrude the specified product.

The section of the T-profile is shown in Fig. 8. 1. The material might flow out of the die with different velocities in different places. In the longer leg it might flow much more quickly than in the short legs. The velocity differences causes bending of the product, which might even produce wrinkles in the long leg of the profile.

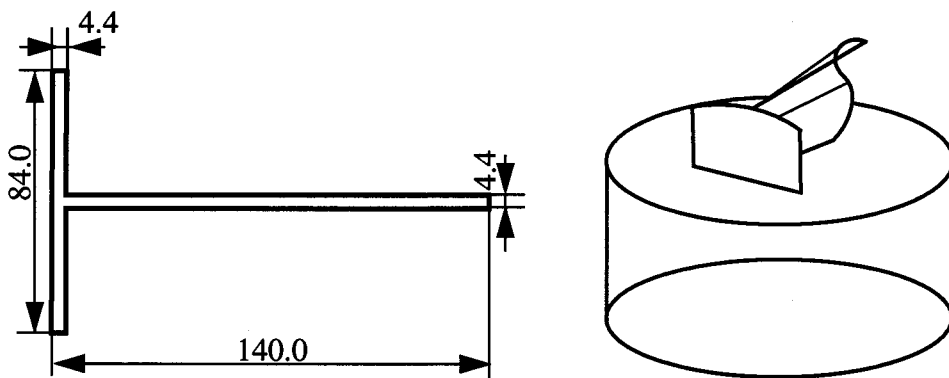


Fig. 8. 1 T-profile produced by the extrusion process

PressForm has been used to simulate and analyze the extrusion process of this profile. Different factors are investigated by changing the corresponding parameters. Although the results have not yet been compared with actual production, neither has the function used to describe the flow stress been verified by experiments because of the high strain rate of ca. 300 s^{-1} , the way to use the FE simulation to solve a real problem is shown.

Because of the symmetry condition only half of the workpiece is computed.

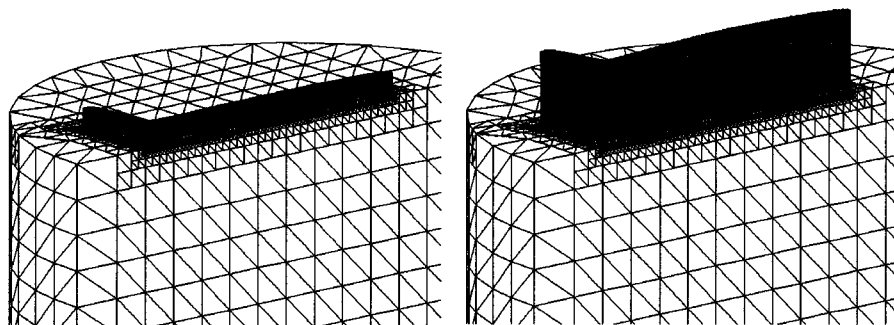


Fig. 8.2 Computation under normal conditions

The first computation was performed under normal conditions. Although the distribution of velocity in the extrusion direction exhibits a small difference across the section of the profile, it causes no serious problem because the velocity difference subsequently decreases, as shown in Fig. 8.2.

8.1.1 Flow condition determined by the geometry

The geometry of the profile is one of the factors which determines the material flow. It is known from fluid dynamics that for confined channels fluid flow is impeded.

If the thickness of the profile is not constant, a difference in the velocity might be induced. Indeed Fig. 8.3 shows this effect. If the long leg is twice as thick as the short one, the velocity in the long leg is twice as fast as in the short leg.

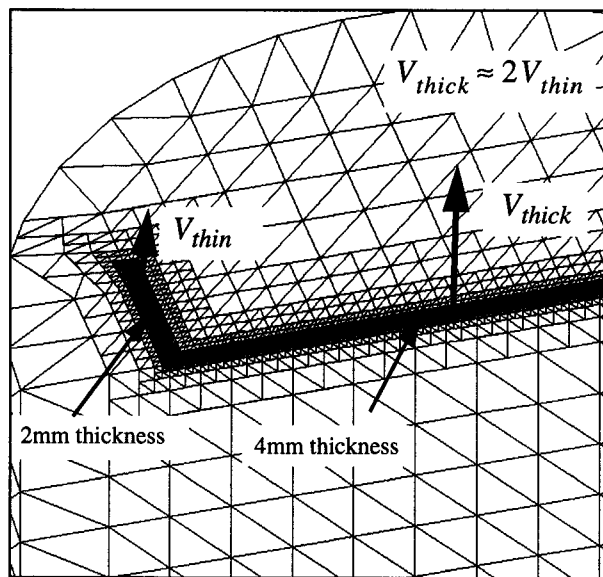


Fig. 8.3 Influence of the geometry factor

This simulation shows that there is a limit for the geometry of the profile. If the flow condition in the profile section is unevenly distributed, the material may flow out of the die with a different velocity. In this case the product no longer possesses the desired shape. The part exhibiting high velocity may pull the part with lower velocity forward. Tensile stress is induced in the latter. Due to such tensile stress the dimension of this part no longer takes on the form of the die, resulting poor precision of the product.

If the situation is more serious, the material may flow out only through a part of the die where flow is much easier, so that the desired form cannot be produced at all. In reality,

there are parts which can be produced only using preforming dies to control the material flow.

8.1.2 Influence of friction

It is well known that friction is an important factor in determining material flow. The extrusion die is generally designed with frictional bands. By changing the length of the bands at different places it is possible to vary the material flow.

Since the geometry should not cause serious problems for the T-profile shown in Fig. 8.1, the effects of friction are investigated.

Although a long frictional band is used for the long leg, the friction applied to the surface of the workpiece in this zone might still not be large enough for the following reasons:

- When the material flows around sharp edges it might lose contact with the die.
- The elastic deformation of the die makes the frictional band ineffective.

Both cases are shown schematically in Fig. 8.4. Their combination may make the situation more serious.

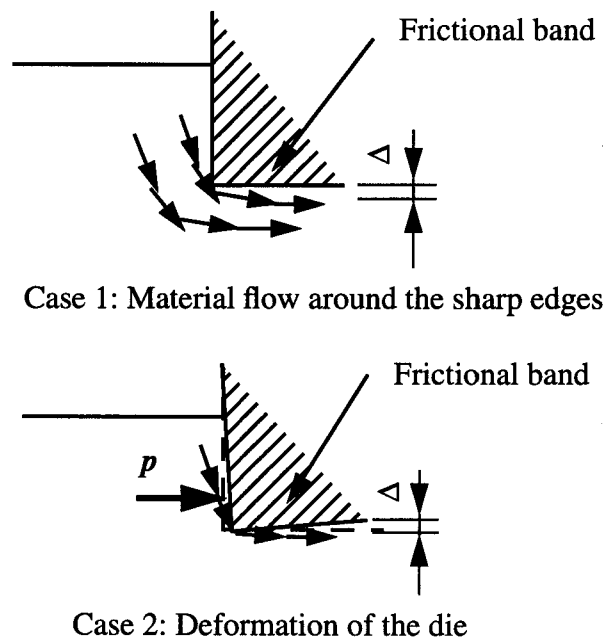


Fig. 8.4 Material loses contact with the frictional band

Because the deflection at the middle of the die under the extrusion pressure is the largest,

it is reasonable to suppose that the material might lose contact with the die so that here the friction forces lose their function.

With this assumption the example has been recalculated. Fig. 8.5 shows the resulting velocity distribution. Such a distribution obviously causes problems as the velocities are quite different.

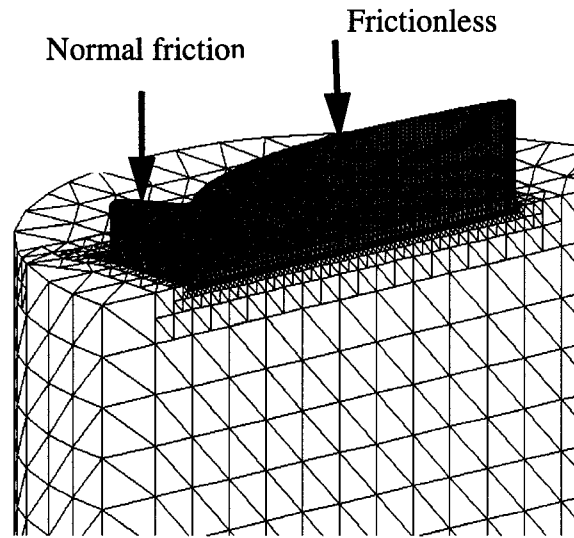


Fig. 8.5 Effect of the distribution of friction

The effect of friction can be seen more clearly if we assume that the frictional forces are linearly distributed along the long leg. Fig. 8.6 shows the results obtained with this assumption.

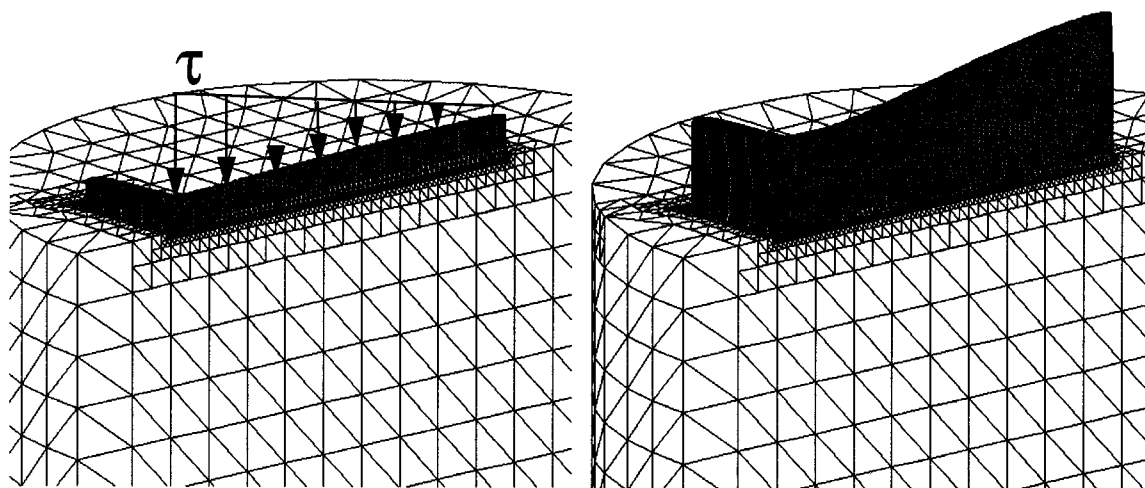


Fig. 8.6 Result obtained assuming linear distribution of friction

8.1.3 Effect of recrystallization

Generally, it is assumed that the materials are both strain rate- and temperature-dependent at high temperature. But with some alloys the experimental curves also exhibit strain softening, i. e. the flow stress decreases when the strain increases. This can be explained as the effect of recrystallization. Deformation energy helps the recrystallization process in alloys. As a result the flow stress gets smaller, but the effect of this phenomenon is limited. If the alloy becomes fully recrystallized, the influence of strain vanishes. A typical isothermal curve is shown in Fig. 8.7 a). The shape of the curve may be described mathematically as follows:

$$\sigma_Y = \sigma_0(\dot{\epsilon}_v, T) \left(A - \frac{B}{\pi} \operatorname{atan} \left(\frac{\epsilon_v - \epsilon_0}{\Delta\epsilon} \right) \right). \quad (8.1)$$

The parameters A , B , ϵ_0 and $\Delta\epsilon$ should be obtained from experimental data. If $A = 1.0$ and $B = 0.0$, the original description is obtained again. The form is shown in Fig. 8.7 b).

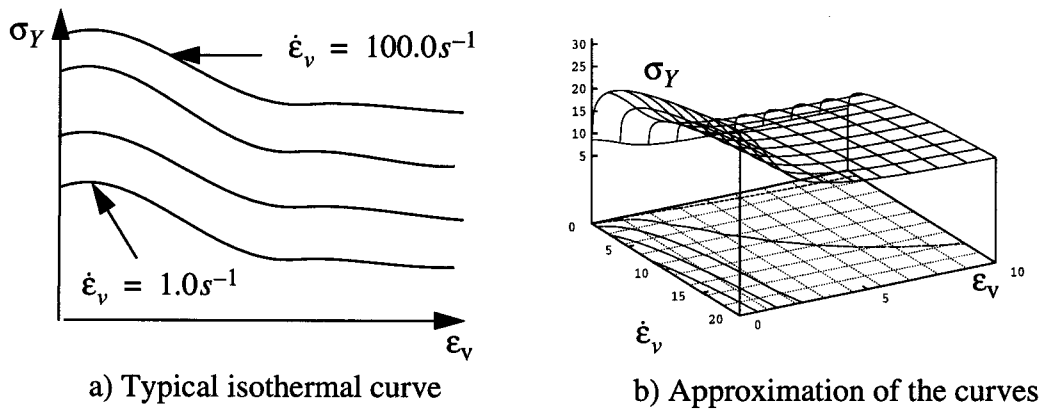


Fig. 8.7 Effect of recrystallization

The above description is used to calculate the profile. The results are compared with the initial computation in Fig. 8.8, in which no serious problem is observed from the results of simulation.

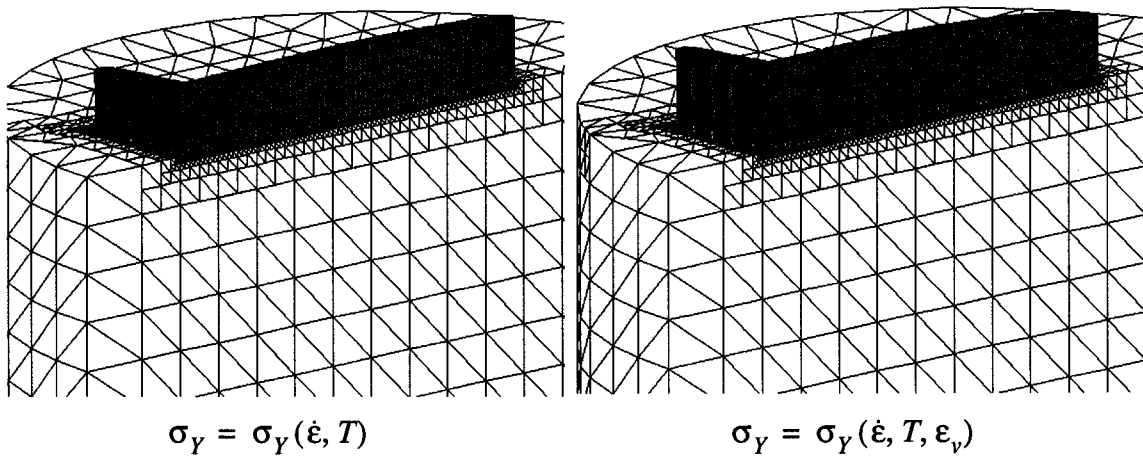


Fig. 8.8 Consideration of the recrystallization

From all these FE simulations the following conclusions can be made:

- Among the factors which influence material flow friction is the most sensitive and important one. Although friction bands are already widely used in the design of extrusion dies, they must be properly constructed. The radii of the edges and sometimes the deformation of the die are of importance.
- Some geometrical shapes are difficult to produce by extrusion. Such profiles often exhibit a sudden change in thickness or a complex form. FE simulation shows up obvious differences in the velocity field and therefore highlights the problem.
- The recovery and recrystallization of some alloys can also change the flow model because the flow stress varies when the plastic strain increases. In such a case, more experimental data is needed to get an accurate approximation of the flow curves used for the FE simulation.

Although the conclusions made above have to be confirmed in the forming industry, so that some revisions are certainly going to be needed, the computation shows clearly the way to use the results of the FE method in the design of the die and to modify the process parameters.

8.2 Weakly coupled thermal analysis of extrusion processes

Thermal analysis is an integral part of the simulation of forming processes. It is most important for the extrusion of some alloys whose metallurgical properties are very sensitive to the temperature distribution.

Some results are available from experiments. For example, the sizes of the crystal grains can be determined experimentally as a function of temperature and plastic strain. A typical example is the diagram to determine the size of the crystal grains after the rolling process [SC66], as shown in Fig. 8.9. Although it is not always possible to obtain a mathematical expression, the size of the grains can be determined if both the deformations and the rolling temperature are known.

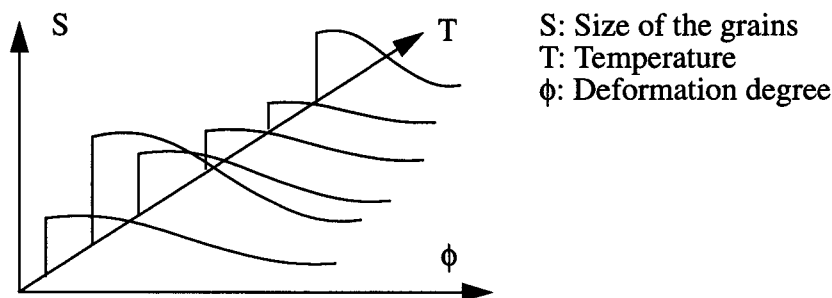


Fig. 8.9 Diagram to determine the size of crystal grains

Of course, there are still other factors such as annealing temperature and time which play important roles in determining the size of grains. However, all these factors can be determined experimentally.

Adequate tolerances for the size of the grains should be enforced to check if the material is qualified for certain applications.

When an engineer has to choose the parameters for the extrusion process an FE simulation can provide him with reliable information. By checking the distribution of temperature and deformation over the whole domain with the known diagram and criterion, it is possible to see in advance whether the resulting products will be acceptable. If not, the parameters should be changed until the desired quality is reached.

The procedure is shown schematically in Fig. 8.10.

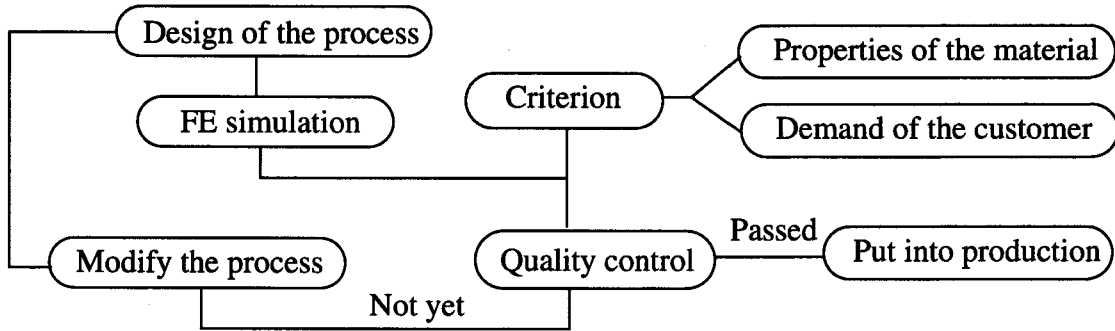


Fig. 8.10 Flow diagram for an FE simulation to check the metallurgical quality

The function of an FE simulation is best explained by means of a numerical example. A rod is produced using the extrusion process. From the viewpoint of economics, it is better to press with a high punch velocity. The increase of punch velocity is limited on the one hand by the capacity of the pressing machine and the strength of the extrusion die, while on the other it is limited by thermal conditions. Fig. 8.11 shows the results of such a simulation.

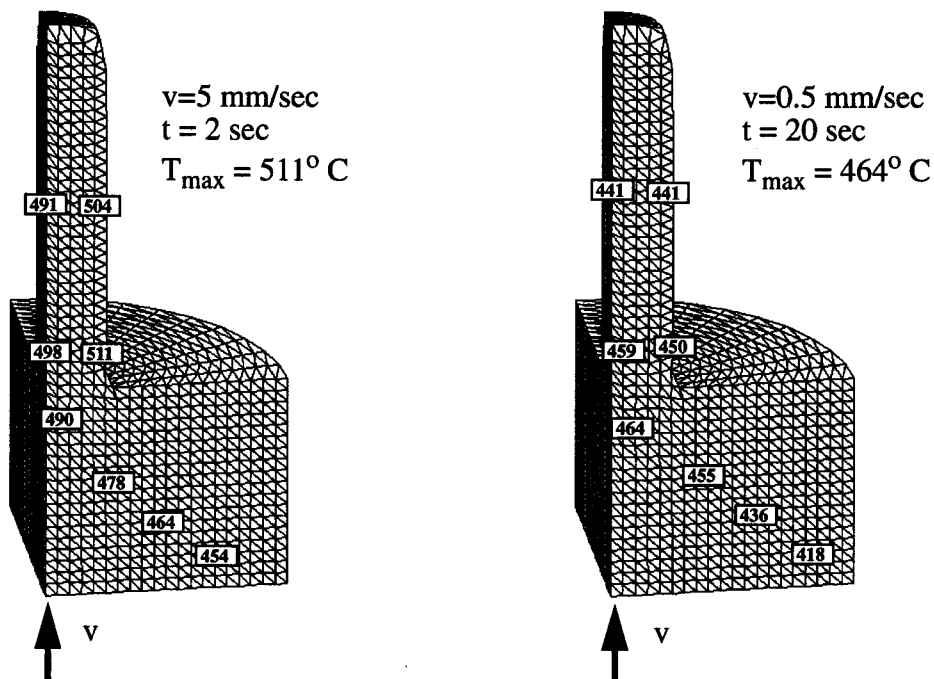


Fig. 8.11 Temperature distributions with different punch velocities

If the punch moves quickly, the maximum temperature inside the workpiece is higher. If the combination of the maximum temperature and the deformation produces excessively

large grains, the quality of the material will deteriorate. In this case the extrusion process must be modified. If the maximum temperature is too high according to the corresponding experiment, since the amount of deformation cannot be changed greatly for a given product, the punch velocity has to be reduced to keep the maximum temperature lower. It is also possible to use a cooling system, but the production costs will increase considerably.

8.3 Some further examples

The following examples show further possibilities of application of FE simulation. In many cases it is easy to obtain the desired profile with the desired metallurgical quality. The key to the processes may lie in other aspects, for example, in the forming forces. The FE simulation can quickly provide the engineer with useful information and help him to determine the proper parameters for the processes.

8.3.1 Forward-backward extrusion

For cold extrusion processes of steel parts one of the critical factors is the load applied to the punch. The cold extrusion of bearing rings is an example. The steel for bearing rings is usually a type of high carbon alloy, which is very hard. When the normal backward extrusion process is adopted to produce the rings, the punch would be broken under such very high pressures.

It is well known that combined forward-backward extrusion requires smaller forces. It is also possible to produce both inner and outer bearing rings in a single stroke. This leads to less material and smaller forming forces, which are certainly favourable in forming processes.

Furthermore, a new process of combined extrusion has been developed for the production of bearing rings. Ring billets instead of solid cylinder billets are used. With this variation the forming forces can be reduced even further. But it is obvious that a ring billet is more expensive than the solid cylindrical one. Different processes are shown in Fig. 8.12.

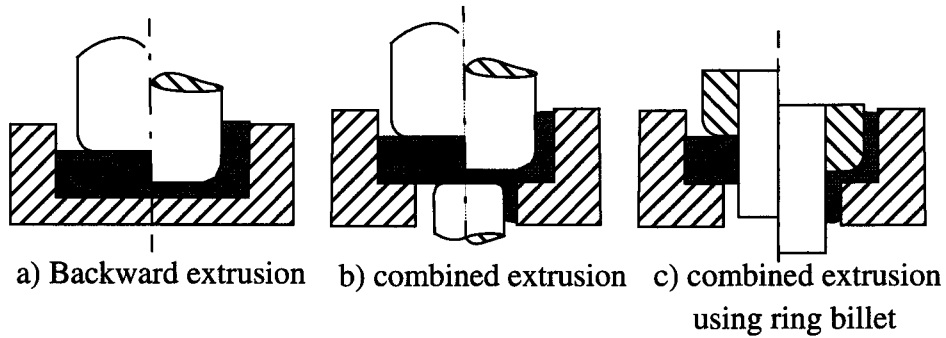


Fig. 8.12 Different processes for the extrusion of bearing rings

Both process b) and process c) are superior to process a) because they require smaller forming forces and less material. Therefore, only these two processes were analyzed using PressForm.

Systematic information is obtained from the simulation. Fig. 8.13 and Fig. 8.14 show the forming procedures obtained from simulation. Although both methods can produce the bearing rings with the desired geometry, the simulation reveals the differences. Fig. 8.15 shows the velocity field of the extrusion processes. When the solid cylinder is used, the material in the middle undergoes compression. The material is forced to flow outwards to fill the inner ring. Larger forming forces as well as more energy are needed in this deformation procedure.

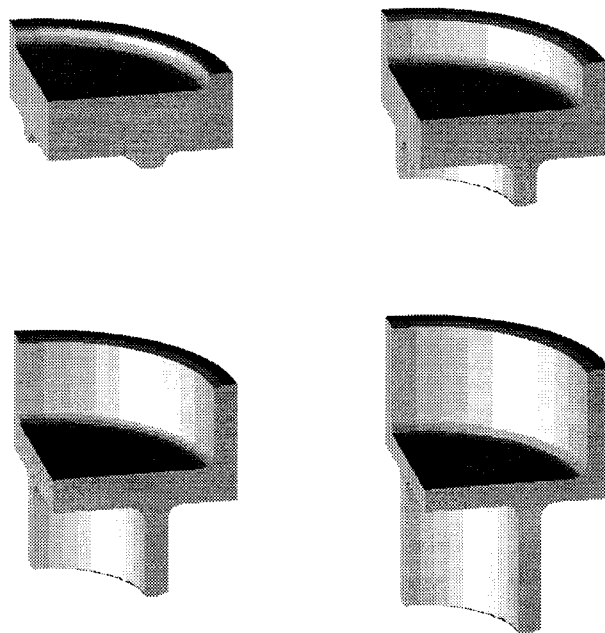


Fig. 8.13 Forming procedure using solid cylindrical billets

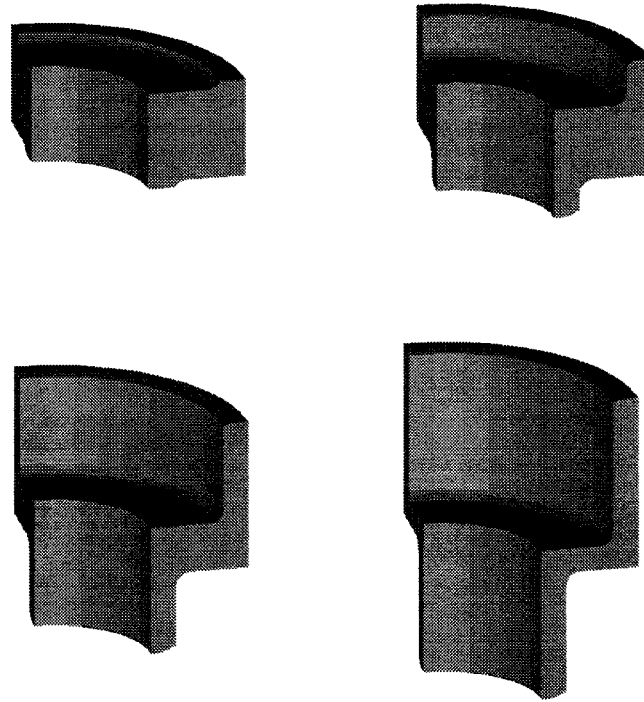


Fig. 8.14 Forming procedure using ring billets

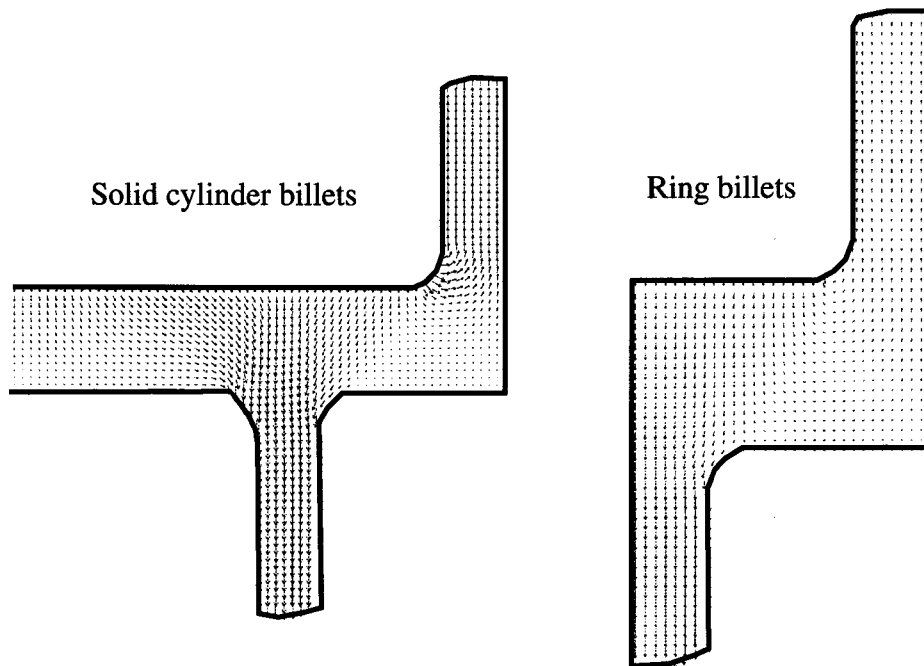


Fig. 8.15 Velocity fields for different extrusion processes

Assuming the flow stress of the steel can be expressed as

$$\sigma = 900 (\epsilon_v + 0.02)^{0.29} (N/mm^2) \quad (8.2)$$

the forming forces obtained from the computation are shown in Fig. 8.16. The force needed using solid billets is about 50% higher than that using ring billets.

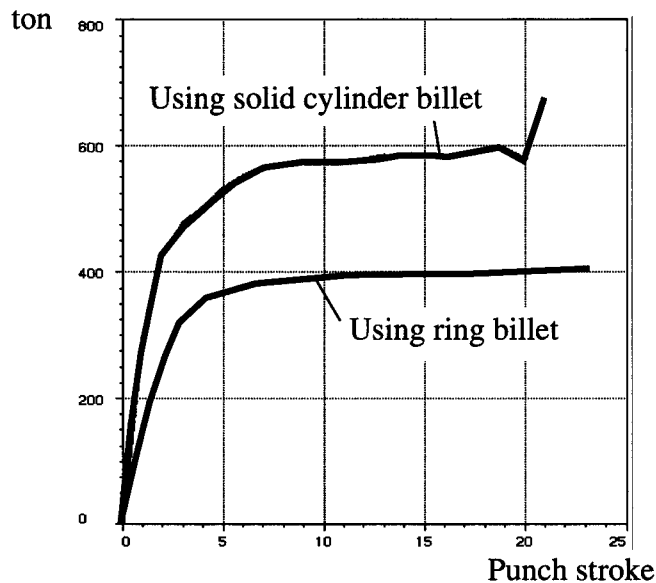


Fig. 8.16 Forming forces for different processes

With the results of simulation it is possible to control the capacity of the press and the strength of the tools. The best process, therefore, can be chosen and optimized.

8.3.2 Optimization of the position of the die

There are many profiles which are not difficult to produce by extrusion processes. Nevertheless, if the dies are not properly designed, problems may still arise during production.

Suppose a T-profile, not as thin and long as in the previous example, has to be produced by the extrusion process. Among other parameters, the engineer must determine the position of the opening in the die.

Experts can perform this job according to experience or using the trial and error method. FE simulation can also carry out this task, e.g. Fig. 8.17 shows the simulation of the extrusion process for a thick T-profile.

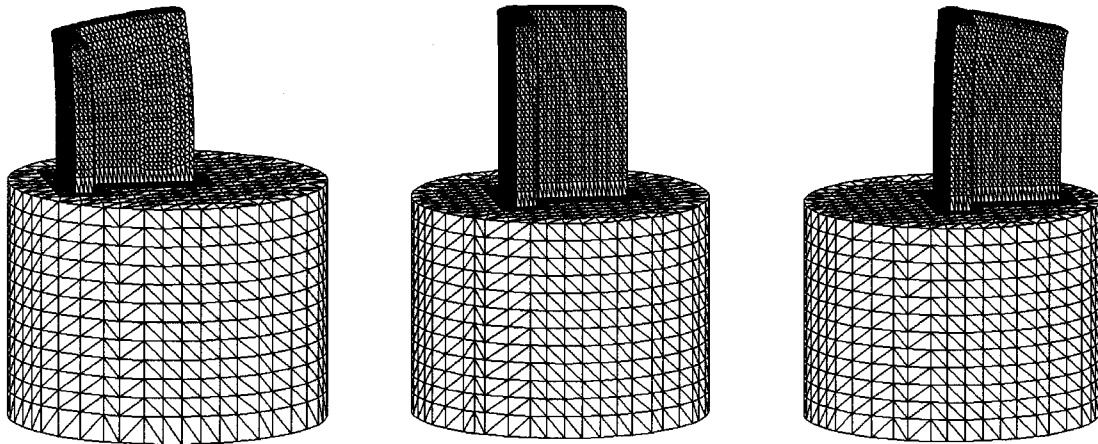


Fig. 8.17 Different position of the opening

If the opening in the die is not properly arranged, differences in the velocity will be induced. Such uneven distributions of velocity result in a curved form of the product. Calibration may be necessary which should be avoided.

If the opening in the die is located in such a way that the material flows out of the die with a constant velocity, the product has a straight form, as shown in the middle of Fig. 8.17.

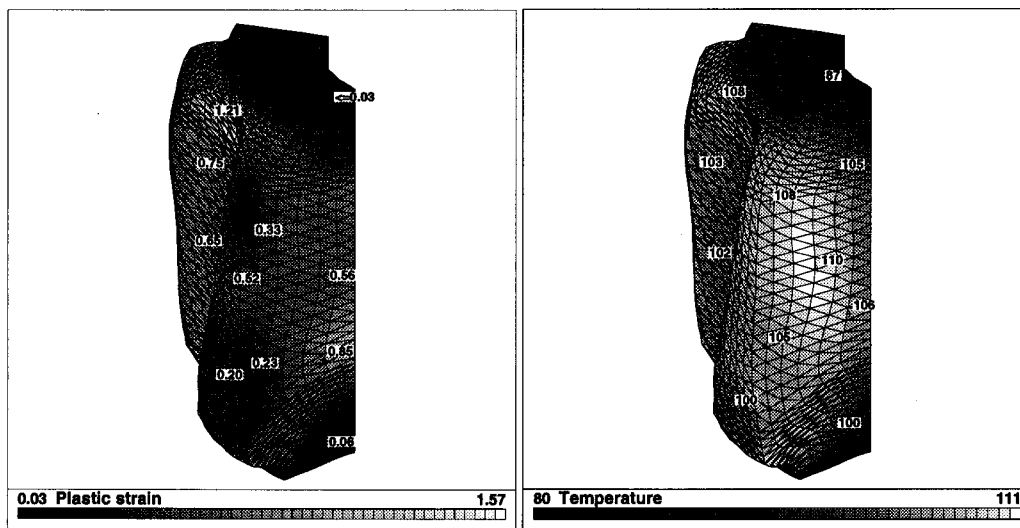
This example also shows the possibility of employing FE simulation repeating the calculations with different parameters. Valid rules for the design of the processes can then be obtained from the simulation results.

8.4 Simulation of a cold forging part using the updated Lagrangian method

Although PressForm is mainly used in connection with the simulation of extrusion processes, it is possible to use it to simulate some general 3-D bulk forming processes.

An example is the cold forging of a steel part (see Fig. 8.18). The part can be produced using different combinations of forming processes. The choice of material also involves several variations. The traditional method of determining the suitable process is to do many experiments, possibly using softer materials in a physical simulation. As the equipment for the experiments is often expensive and not very flexible, the cycle to design a process can be long and expensive.

The FE method offers a completely different way of performing this task. Only basic properties of the material are needed which are known or can be obtained from standard experiments. The geometrical description of the workpiece and the tools can be generated on the computer. After the calculation the post-processor shows the results of the simulation which will help the engineer in making decisions. The results from FE simulation are obtained in a much shorter time than they are from experiments. It is also very easy to modify the parameter set for the FE simulation. Besides, the numerical computation provides systematic information about the process. Some field variables such as plastic strain and temperature inside the workpiece are hardly available from experiments but are very easy to obtain from the FE simulation.



The distribution of plastic strain

The distribution of temperature

Fig. 8.18 Distribution of strain and temperature at the end stage of the simulation of a forming process

As an example, only the distributions of the equivalent strain and temperature at the end of the forming process are shown in Fig. 8.18.

Schematically, the way FE method supports the CAD method is shown in Fig. 8.19. Engineers should have the necessary material data, the description of workpiece and tools and the parameters of the process. Several criteria to determine whether the process is valid such as the maximum forming forces allowed by the tools and forming machines and the limit values for deformation or temperature should also be available. They then

run the FE program to perform the computation. The results of the simulation show the states of deformation and stress, the temperature distribution, the necessary forming forces and possibly can also predict failures. With this information engineers can optimize the parameters of the process until all the criteria fixed by customers or production are passed. There may be several different methods for the same desired product. According to the information from FE simulation, it is no longer difficult to choose the most suitable one.

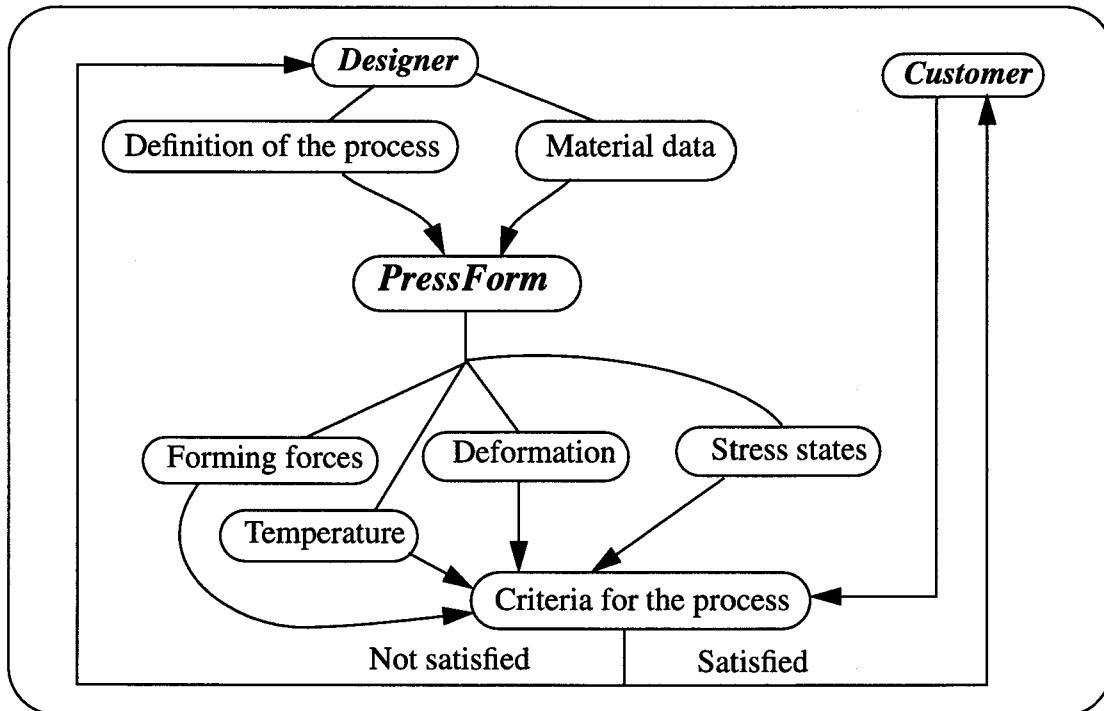


Fig. 8.19 Flow diagram of the FE-supported CAD procedure

9. Conclusions

From the above investigations the following conclusions can be made:

- The ALE method provides a way of overcoming the difficulty of strong mesh distortion in the FE simulation of bulk forming processes. The advantages of this formulation are fully realized in the simulation of forward extrusion processes. The mesh is automatically adjusted to couple the Lagrange and Euler zones.
- The formulation not only avoids strong distortions but also provides better convergence of the iterations because the velocities of the last increment can serve as a good estimate for the next increment. With the iterative solution algorithm it is possible to compute a realistic extrusion process in a couple of hours on a modern workstation. The rigid-plastic description with the penalty method for the incompressibility is valid for the simulation of bulk forming processes.
- The application of PressForm can be extended to the case of simulating general 3-D bulk forming processes without making big changes in the program. The necessary condition for such an extension is the support of a more powerful remeshing algorithm.

In the future development of PressForm the following aspects should be emphasized:

- A more powerful adaptive mesh regenerator is urgently needed if the application of the program is to be extended to general 3-D bulk forming processes.
- Because the size of the 3-D bulk forming problem is very large, the solution procedure takes more than 80% of the CPU time. It is important to improve the iterative solution method further to perform the simulation in a shorter time.
- Bulk forming processes involve many factors such as the behaviour of materials in a state of very large deformation and deformation rate, the lubrication of the dies, the thermal conditions and so on. Only when all these boundary conditions and initial values are correctly determined, can reliable solutions be expected. It is therefore very important to obtain the necessary experimental data. A data base for materials is also an important aspect.

References

- [AK93] Akkerman R., Euler-Lagrange simulations of nonisothermal viscoelastic flows, University of Twente, Enschede, (1993)
- [AR78] Argyris, J. H., Dunne P. C., Haase M., and Orkisz J., *Higher-order simpler elements for large strain analysis - natural approach*, Comp. Meth. Appl. Mech. Eng. 16, pp. 369-403, (1978)
- [AR79] Argyris, J. H. and Doltsinis J. St., *On the large strain inelastic analysis in natural formulation*, Comp. Mech. Appl. Eng., 20, pp. 213-251, (1979)
- [AR84] Argyris, J. H., Doltsinis J. St. and Wustenburg H., *Analysis of thermoplastic forming processes - natural formulation*, Numerical Analysis of Forming Processes, J. F. T. Pitterman et al. (eds.) pp. 89-115, (1984)
- [AV68] Avitzur B., *Metal Forming Processes and Analysis*, McGraw-Hill (1968)
- [BA89] Bartelt P., *Finite element procedures on vector/tightly coupled parallel computers*, PhD thesis Swiss Federal Institute of Technology, (1989)
- [BE82] Belytschko T., Flangan D. P. and Kennedy J. M., *Finite element method with user-controlled meshes for fluid-structure interaction*, Comp. Meth. Appl. Mech. Eng. 33 pp. 669-688, (1982)
- [BE89] Benson D. J., *An efficient, accurate, simple ALE method for nonlinear finite element programs*, Comp. Meth. Appl. Mech. Eng. 72, pp. 305-350, (1989)
- [CH88] Cheng J. H. Int. J. Num. Meth. Eng. 26, pp. 1-8, 1988
- [DO92] Doltsinis I. ST. & Nölting S. *Parallel algorithms for the modelling and simulation of industrial metal forming*, NUMIFORM 92, Chenot J.-L., Wood R. D. and Zienkiewicz O. C.(eds.) pp. 35-43, (1992)
- [GH91] Ghosh S. and Kicuchi N. *An Arbitrary Lagrangian-Eulerian Finite Element Method for Large Deformation Analysis of Elastic-Viscoplastic Solids*, Comp. Meth. App. Mech. Eng. 86, pp. 127-188, (1991)
- [HE94] Herrmann et. al., *Benchmark definitions and major results*, Supplemental publication of the Proceedings of Metal Forming Process Simulation in Industry, (1994)
- [HI50] Hill R. *The mathematical Theory of Plasticity*, Clarendon Press (1950)
- [HU82] Huetink J. *Numerical Method in Industrial Forming Processes*, J. F. T.

- Pitterman et al. (eds.) pp. 501-509, Swansea: Pineridge Press, 1982
- [HU84] Hughes T. J. R. *Numerical Implementation of Constitutive models, Rate-Independent Deviatoric Plasticity*, Theoretical Foundation for Large Scale computations for Nonlinear Material Behavior, Nemat-Nasser S et al. (eds.) (1984)
- [HU86] Huetink J., On the simulation of thermo-mechanical forming processes, a mixed Eulerian-Lagrangian finite element method, Diss. University of Twente, Enschede, 1986
- [HU88] Huang G. C., Liu Y. C. and Zienkiewicz O. C. *Modelling of Metal Forming Processes*, J. L. Chenot and E. Onate (eds.), pp. 75-83, Kluwer Academic Publishers, (1988)
- [HU90] Huetink J., Vreede P. T. and Van der Lugt J., *Progress in mixed Eulerian-Lagrangian finite element simulation of forming processes*, Int. J. Num. Meth. Eng. 30, pp. 1441-1457, (1990)
- [JO73] Johnson W. and Mellor F. B., *Engineering Plasticity*, Nostrand Reinhold (1973)
- [KI90] Kincaid D. R. and Hayes L. J., *Iterative Methods for Large Linear Systems*, Academic Press, (1990)
- [KO73] Kobayashi S. and Lee C. H., *Deformation mechanics and workability in upsetting solid circular cylinders*, Proc. 1st. North American Metalworking Res. Conf., SME, pp. 185-204, (1973)
- [KO86] Kobayashi S., *Advances in forging technology by the finite element method*, NUMIFORM 86, Mattiasson K., Samuelsson A. Wood R. D. and Zienkiewicz O. C.(eds.) pp. 19-28, (1986)
- [KO89] Kobayashi S., Oh S. I. and Altan T., *Metal Forming and the Finite-Element Method*, Oxford University Press, (1989)
- [KO94] Kopp R., *Dreidimensionale FEM-Simulation mit Netzgenerierung im industriellen Einsatz*, Metal Forming Process Simulation in Industry Baden-Baden, pp. 26-44, (1994)
- [LA87] Lange K., *Advanced Technology of Plasticity 1987*, Proceedings of the Second International Conference of Technology of Plasticity, Springer-Verlag (1987)
- [LE69] Lee E. H. *Elastic-plastic deformations at finite strains*, J. Appl. Mech. 36, pp. 1-6, (1969)
- [LE73] Lee C. H. and Kobayashi S., *New solution to rigid-plastic deformation problems using a matrix method*, J. Eng. Ind. 95, pp. 865-873, (1973)
- [LI82] Li G. J. and Kobayashi S., *Spread analysis in rolling by the rigid-plastic finite element method*, Proc. 1st Int. Conf. on Numerical

- Methods in Industrial Forming Processes, J. F. T. Pitterman et al. (eds.) pp. 777-786, (1982)
- [LI86] Liu W. K., Belytschko T., Chang H., *An arbitrary Lagrangian-Eulerian FE-method for path-dependent materials*, Comp. Meth. App. Mech. Eng. 58, pp. 227-245, (1986)
- [LI88] Liu W. K., Chang H., Belytschko T., Chen J. S., *Arbitrary Lagrangian-Eulerian Petrov-Galerkin Finite Elements for Nonlinear Continua*, Comp. Meth. App. Mech. Eng. 68, pp. 259-310, (1988)
- [MC75] McMeeking R. M. and Rice J. R., *Finite-element formulations for problems of large elastic-plastic deformation*, Int. J. Solid. Struct. 11, pp. 601-616, (1975)
- [MO84] Mori K., Osakada K., and Fukuda M. *Simulation of severe plastic deformation by finite element method with spatial fixed elements*, Int. J. Mech. Sci. 26, pp. 515-525, (1984)
- [NA74] Nagtegaal J. C. Parks D. M. and Rice J. R. *On numerically accurate finite element solutions in the fully plastic range*, Comp. Meth. Appl. Mech. Eng. 4, pp. 153-177, (1974)
- [NA81] Nagtegaal J. C. and de Jong J. E. *Some computational aspects of elastic-plastic large strain analysis*, Int. J. Num. Meth. Eng. 17, pp. 15-41, (1981)
- [NA86] Nagtegaal J. C. and Rebelo N. *On the development of a general purpose finite element program for analysis of forming processes*, NUMIFORM 86, Mattiasson K., Samuelsson A. Wood R. D. and Zienkiewicz O. C.(eds.) pp. 41-50, (1986)
- [NO82] Nonea J., Giuliani S. and Halleux J. P., *An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactor*, Comp. Meth. Appl. Mech. Eng. 33, pp. 689-723, (1982)
- [OW80] Owen D. R. J. and Hinton E., *Finite Elements in Plasticity, Theory and Practice*, Pineridge Press (1980)
- [PA82] Pacheco L. A. and Alexander J. M. *On the hydrostatic extrusion of copper-covered aluminium rods*, Numerical Method in Industrial Forming Processes, J. F. T. Pitterman et al. (eds.) pp. 205-216, (1982)
- [PI87] Pietrzyk M., Luksza J. and Sodak L., *Finite element analysis of the shear strain in the axisymmetrical drawing*, Proc. 2nd Int. Conf. on Technology of Plasticity, ed. K. Lange, Springer, pp. 835-840, (1987)
- [PO91] Ponthot J.-P. and Hogge M., *The use of the Eulerian-Lagrangian FEM in metal forming applications including contact and adaptive mesh*, Advances in Finite Deformation Problems in Material Processing and

- Structures, ASME vol. 125, pp. 49-63, (1991)
- [RE88] Reddy J. N., Krishnamoorthy C. S. and Seetharamu K. N. (eds.), *Finite Element Analysis for Engineering Design*, Springer-Verlage, (1988)
- [SC66] Schneider E., *Grundlagen des Walzens mit Kalibern, Grundlagen der bildsamen Formgebung*, pp. 321-347, (1966)
- [SH82] Schreurs P., Veldpaus F. and Brekelmans W., *An arbitrary Eulerian-Lagrangian finite element model for the simulation of geometrical non-linear hyperelastic and elastic-plastic deformation processes*, Numerical method in industrial forming processes. J. F. T. Pittman et al. (eds.), Swansea: Pineridge Press, (1982)
- [SH83] Schreurs P., *Numerical Simulation of Forming Processes*, Dissertation, TH Eindhoven, (1983)
- [TE94] Tekkaya A. E. et. al., *Rigid-plastic Finite Element Modelling of Three-Dimensional Bulk-Metal Forming Processes*, Metal Forming Process Simulation in Industry Baden-Baden, pp. 60-79, (1994)
- [TH79] Thomas J. R. et al, *Finite Element Analysis of Incompressible Viscous Flows by the Penalty Function Formulation*, J. Comp. Phys. 30. pp. 1-60, (1979)
- [TO92] Tong L., Hora P. and Reissner J. *Application of the arbitrary Lagrangian-Eulerian method in the FE-simulation of 3-D bulk forming processes*, NUMIFORM 92, Chenot J.-L., Wood R. D. and Zienkiewicz O. C.(eds.) pp. 669-674, (1992)
- [WA92] Wagoner R. H. and Zhou D. *Analyzing sheet forming operations - recent numerical and experimental advances*, NUMIFORM 92, Chenot J.-L., Wood R. D. and Zienkiewicz O. C.(eds.) pp. 123-132, (1992)
- [YA68] Yamada Y., Yoshimura N. and Sakurai T., *Plastic stress-strain matrix and its application for the solution of elastic-plastic problems by the finite element method*, Int. J. Mech. Sci. 10, pp. 343-354, (1968)
- [ZI69] Zienkiewicz O. C. Vallappan S. and King I. P. *Elasto-plastic solutions of engineering problems by the 'initial stress' finite element approach*, Int. J. Num. Meth. Eng. 1, pp. 75-100, (1969)
- [ZI77] Zienkiewicz O. C. *The finite Element Method*, McGraw-Hill (1977)
- [ZI84] Zienkiewicz, O. C., *Numerical analysis of forming processes*, J. F.T. Pittman et al. (eds.), p. 1-44, (1984)
- [ZI86] Zienkiewicz O. C. et al., *Flow formulation for numerical solution of forming processes*, NUMIFORM 86, Mattiasson K., Samuelsson A.

Wood R. D. and Zienkiewicz O. C.(eds.) pp. 3-10, (1986)

[ZI89]

Zienkiewicz, O. C. and Huang J. C., *Adaptive modelling of transient coupled metal forming processes*, Numiform '89, Thomson et al. (eds.), pp. 3-7, 1989

Notation

$A, B, C, \dots, a, b, c, \dots$	Scalar values, constants
$\mathbf{A}, \mathbf{B}, \mathbf{C}, \dots, \mathbf{a}, \mathbf{b}, \mathbf{c}, \dots$	Tensors, matrices or vectors
$[A], [B]$	Matrices
$\{A\}, \{B\}$	Column matrices
A_i	Components of a vector
A_{ij}	Components of a tensor or a matrix
v_i, \mathbf{v}	Velocity
a_i, \mathbf{a}	Acceleration
$\frac{d}{dt}, (\dot{\quad})$	Material derivative
u	Internal energy
\mathbf{u}	Displacement
$\Delta \mathbf{u}$	Incremental displacement
ρ	Density
C, C_ρ	Specific heat
λ	Coefficient of conductivity, Lagrangian multiplier
$d\lambda$	Scalar coefficient of associated flow rule
\mathbf{b}, \mathbf{f}^b	Body forces
R, r	Configuration
\mathbf{F}, F_{ij}	Deformation gradient
δ_{ij}	Kroneker delta
$\varepsilon_{ij}, \dot{\varepsilon}_{ij}, \Delta \varepsilon_{ij}$	Strain, strain rate and strain increment
$\Delta \boldsymbol{\varepsilon}$	Strain increment saved as a vector
\mathbf{E}, E_{ij}	Lagrange strain
\mathbf{e}, e_{ij}	Almansi strain
σ_{ij}	Cauchy stress
$\boldsymbol{\sigma}$	Cauchy stress saved as a vector
S_{ij}	The 2nd Piola-Kirchhof stress
σ'_{ij}	Deviatoric Cauchy stress
\mathbf{D}	Elasticity matrix
W_{ij}	Spin tensor
$\dot{\sigma}_{ij}$	Rate of Cauchy stress
$\dot{\sigma}^J_{ij}$	Jaumann's rate of Cauchy stress
σ_v	The equivalent stress

ϵ_v	The equivalent strain
σ_Y	The yield stress
\dot{w}	Rate of deformation work
q, q_i	Heat flux
Φ	Dissipation rate
w, w	Weighting function
α	Eigenvalue
s	Entropy
$T, \dot{T}, \Delta T$	Temperature, rate of temperature and temperature increment
$T, \dot{T}, \Delta \dot{T}$	Temperature, rate of temperature and temperature increment at the nodes
$t, \Delta t$	Time, small time increment
Y	Yield locus
p	Hydrostatic pressure
u^n	Displacements at nodes
$N, N_i(\xi, \eta, \zeta)$	Interpolating function
τ	Friction stress
m, μ	Frictional coefficient
$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$	Kronecker delta
$A_k B_k = \sum_{k=1}^3 A_k B_k$	Einstein's summation convention
$F_{i,j} = \frac{\partial F_i}{\partial x_j}$	Partial derivative with respect to current configuration
$v_{i,i} = \text{div} v$	Divergence of v
$T_{,ii} = \text{div}(\text{grad} T)$	Divergence of gradient of T (Laplace operator)

Resume

Personal particulars:	Name	TONG	First name	LONGCHANG
	Sex	Male	Date of Birth	6th Feb. 1951
	Civil state	Married		
Education:	1958 - 1964	Primary school in Peking		
	1964 - 1968	Secondary school in Peking		
	1978 - 1982	College education at Taiyuan University of Technology, Shanxi, China; majored in forming processes and machinery		
	1982 - 1984	Postgraduate student at Tsinghua University, Peking, China		
	10. 1984	Examination for Degree of Master of Science, with a topic in forming processes		
	1987 - 1988	German course at Peking Language Institute		
Experience:	1968 - 1972	Worked in the countryside during the "Cultural Revolution"		
	1972 - 1978	Worked at Datong gear factory, Shanxi, China		
	1984 - 1987	Assistant and lecturer at Tsinghua University, Peking, China		
	Since 1988	Assistant and doctoral candidate at Institute of Forming Technology, ETH-Zurich, specialized studies in the field of numerical simulation		