Developments of the Arbitrary Lagrangian–Eulerian Method in non-linear Solid Mechanics

Applications to Forming Processes

Christiaan Stoker

De promotiecommissie is als volgt samengesteld:

Voorzitter en secretaris:	
Prof.dr.ir. H.J. Grootenboer	(Universiteit Twente)
Promotor:	
Prof.dr.ir. J. Huétink	(Universiteit Twente)
Leden:	
Prof. J.-L. Chenot	(Ecole des Mines de Paris)
Prof.dr.ir. F.P.T. Baaijens	(Technische Universiteit Eindhoven)
Prof.dr.ir. H.W.M. Hoeijmakers	(Universiteit Twente)
Prof.dr.ir. C.B. Vreugdenhil	(Universiteit Twente)

Developments of the Arbitrary Lagrangian–Eulerian Method in non-linear Solid Mechanics. Applications to Forming Processes. /
Stoker, Hans Christiaan
Thesis Enschede.- With ref.
ISBN 90-36512646

Subject headings: Arbitrary Lagrangian–Eulerian method,
finite element method, meshing, remap of state variables

Printed by FEBO druk, Enschede

DEVELOPMENTS OF THE ARBITRARY LAGRANGIAN–EULERIAN METHOD IN NON-LINEAR SOLID MECHANICS

PROEFSCHRIFT

ter verkrijging van
de graad van doctor aan de Universiteit Twente,
op gezag van de rector magnificus,
prof.dr. F.A. van Vught,
volgens besluit van het College voor Promoties
in het openbaar te verdedigen
op vrijdag 26 februari 1999 te 15:00 uur

door

Hans Christiaan Stoker

geboren op 27 oktober 1969
te Appelscha

Dit proefschrift is goedgekeurd door de promotor:

prof.dr.ir. J. Huétink

Summary

The numerical simulation of forming processes can be performed using the updated Lagrangian method. However, in general in the case of large deformations distortion of the mesh occurs. As a result the calculation becomes inaccurate or it may even crash. Hence a complete remeshing for an element mesh with a new mesh topology is necessary. The Arbitrary Lagrangian–Eulerian (ALE) method can be used to reduce mesh distortion in order to prevent complete remeshings. In this thesis we investigate the applicability of the ALE method and try to improve several of its aspects.

The uncoupled ALE method consists of three consecutive steps: an updated Lagrangian step, a meshing step and a remap of the state variables. In the updated Lagrangian step we employ a finite element discretisation. In the meshing step of the ALE method the position of an new element mesh with the same mesh topology is determined. We use two types of meshing methods, *i.e.* the transfinite mapping method and methods based on solving the Laplace equation. Due to this meshing step a remap of state variables is necessary, which is described by the convection equation. Several methods are discussed for the discretisation of the convection equation in the context of the ALE method.

To examine the ALE method two academic test cases and simulations of forming processes were performed. In these simulations one can see that the ALE method is usually more suitable than the updated Lagrangian method. In general the remap of state variables is carried out sufficiently accurately. It is concluded that the meshing step is crucial for the effectiveness of the ALE method. In a number of cases it is not possible to calculate the whole simulation with the same element mesh topology. The ALE method in combination with remeshings should be used, where the ALE method tends to reduce the number of remeshings required.

Contents

Chapter 1

Introduction

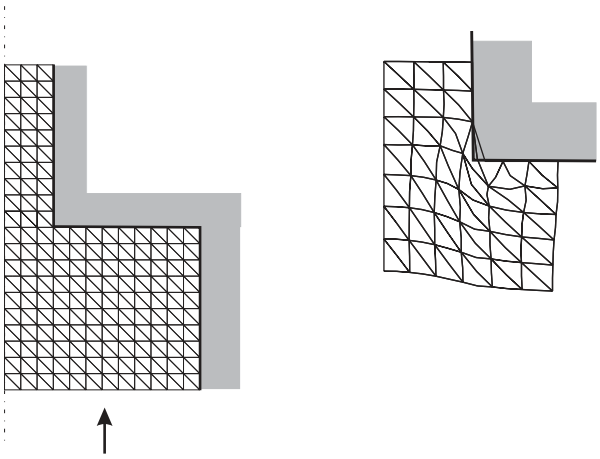
In this thesis the Arbitrary Lagrangian–Eulerian (ALE) method is investigated with the intention of making improvements. The ALE method can be useful in the simulation of forming processes. Numerical simulations of forming processes are applied to analyse or optimise processes such as extrusion, forging and rolling. For example, in the extrusion process hot material is forced through a hole to form a product with the desired shape. In general, high demands are made on the product shape by the manufacturer. The final shape of the product depends on a large number of parameters, for instance the temperature of the material, the design of the die, *etc.* When problems occur, the extrusion process must be analysed and corrections made to the process. Simulations can be very useful in the analysis of the extrusion process. Another example is injection moulding of a polymer. Due to the process, residual stresses can be introduced into the product, which results in an undesirable shape. These problems must be solved and simulations can also here be very practical in understanding the phenomena that occur during the process. The problems can also be investigated in an experimental setup by changing the process parameters. However, these experiments are generally expensive and time consuming. A third example of a forming process is forging. Often a product is fabricated in a number of stages with a different tool being used at each stage. It is desirable to reduce the number of stages. Within a simulation of the various stages the shape of the tools can be changed relatively easily. As a result a forging process can be optimised with the help of simulation techniques. Optimisation of the process with experiments is less efficient, as the fabrication of a new tool is much more expensive and time consuming.
Hence, one can employ two methods: experiments and numerical simulations. In general, experiments are more time consuming and more expensive. Nowadays numerical simulations have become more popular, because progress in computational capacity in recent years had enabled more extensive and

faster simulations. These simulations can give insight within a relatively short computer time. This enables parameter studies with numerical simulations that are cheaper than the trial-and-error methods of experiments, although of course experiments are still required in order to provide input data (*e.g.* material properties) and for validation of the numerical results.
A numerical simulation of a forming process usually consists of three steps. The first step consists of describing the effects that occur during the process (for example, the material behaviour and the frictional behaviour) in a mathematical model. One has to know the effects that occur in the process in order to construct a mathematical model that describes the real process accurately. The question might be for example "Is the friction behaviour well described by a viscous friction model or should the Coulomb friction model be used ?". Another example is that one should know whether temperature and elastic effects are important for the description of the process.
The second step is to solve the mathematical model with a numerical method on the computer, which requires a good understanding of numerical mathematics. One should know the limitations of the numerical method.
The last step is the interpretation of the results. Here it is important to have a good understanding of both numerical mathematics *and* the mechanics of the process. For example, effects that are due to the numerical algorithm should not be interpreted as physical effects of the process and *vice versa*.
In this thesis we focus on the second step, *i.e.* the solution of a mathematical model using a numerical method.

1.1 Numerical simulation

Several methods can be used to solve a mathematical problem. Many of these methods are based on the finite element method: the domain is subdivided into a number of elements on which the solution is approximated with simple basic functions. With the finite element method problems with complex geometries can be handled, which makes it a popular method. Forming processes are often characterised by path dependent material behaviour, *i.e.* the history of the material has to be taken into account. For this reason, the updated Lagrangian method is often applied in the simulation of forming processes. In this method the deformation path is approximated by increments in time. After each increment the reference situation is updated with the solution. In this way the history of the material is easily taken into account. This updated situation is used as an initial condition for the next increment. So the finite element mesh is connected with the material throughout the calculation. However, the updated Lagrangian method can suffer from mesh distortion

and mesh entanglement, because of large deformations in these calculations. A possible solution is to perform a remeshing each time the mesh becomes 'too distorted'. In order to take into account the history of the material, a remap of state variables from the old element mesh to the new element mesh with a different mesh topology must be carried out. This remeshing is computationally expensive [42] and the remap of state variables related to the remeshing will usually introduce additional inaccuracy. Therefore the updated Lagrangian method can become impractical in calculations in which a large number of remeshings is necessary.

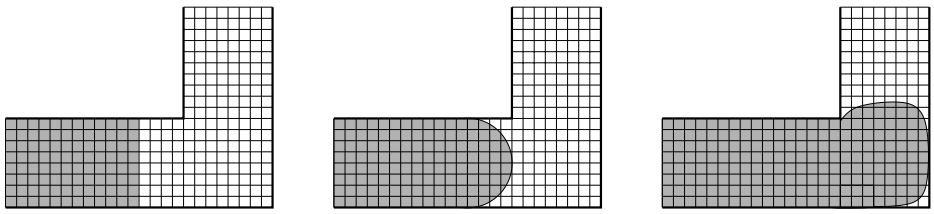


(a) Initial setup

(b) Zoom on distorted mesh

Figure 1.1: Updated Lagrangian calculation of the extrusion process

In Figure 1.1 for instance the simulation of the extrusion process with a sharp angle at the die exit is shown. The elements at the die exit are locally distorted. So a remeshing has to be made in order to be able to continue the calculation. Another possibility is the use of a Eulerian formulation. In contrast with the updated Lagrangian method it does not suffer from mesh distortion, because the mesh is spatially fixed during the simulation. However, it is more difficult to take into account the history of the material, as the element mesh is not connected to it. A second problem is that it is difficult to obtain an accurate description of the free surface, which is an important result of the simulation of forming processes. The free surface must be described in the fixed element mesh (see Figure 1.2).



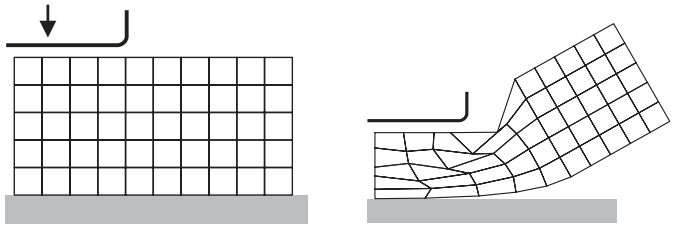
(a) (b) (c)

Figure 1.2: Eulerian simulation of mould filling where the shaded area indicates the material

Arbitrary Lagrangian–Eulerian method

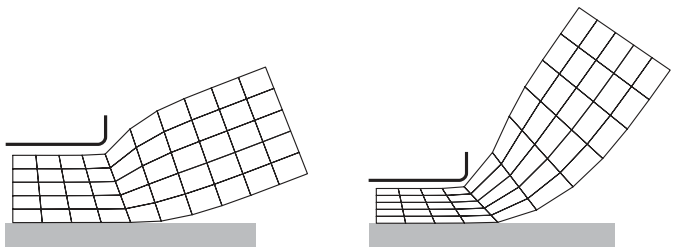
The Arbitrary Lagrangian–Eulerian (ALE) method is a combination of the two above-mentioned formulations. Originally, the method was introduced in the context of the finite difference method [22]. The ALE method is employed with the finite element method in fluid mechanics and fluid dynamics [13, 2, 48] and in fluid–structure interaction [44]. It is nowadays also used in the simulation of forming processes, such as for example extrusion [27].
In the ALE method the mesh is neither connected to the material nor fixed to the spatial coordinate system, but it can be prescribed in an arbitrary manner. As a result a mesh velocity has to be computed in order to obtain the mesh. Grid points on the surface also move with the mesh velocity, but these points must remain on the free surface. Since the mesh is not connected to the material during the calculation, a remap of state variables has to be performed. This remap of state variables is much easier than the one related to the remeshing in the updated Lagrangian method. The remeshing in the ALE method results in an element mesh with the same mesh topology. The calculation of the mesh velocity and the remap of state variables are two important aspects of the ALE method, which is the subject of this thesis.
The freedom of the choice of a mesh velocity in the ALE method can be employed for several purposes. Firstly, it can be used to reduce the error by computing the mesh velocity with the help of an error estimator. As a result a refined element mesh can be obtained in the region where the error estimation is large. In [50] for example the ALE method is employed to adapt the mesh to the plastic zones. Accurate results can be obtained while using a fine element mesh only locally. A second important purpose is to prevent mesh distortion with the computation of the mesh velocity (see *e.g.* the mesh distortion in Figure 1.1).
In Figure 1.3 is illustrated how the ALE method can prevent mesh distortion

in the simulation of the upsetting process. In Figure 1.3(a) the initial setup is presented. The updated Lagrangian calculation, where the element mesh is connected to the material, results in a distorted element mesh (see Figure 1.3(b)). In Figures 1.3(c) and (d) one can see that the ALE method results in less distorted meshes. So the calculation of the mesh velocity in this example was effective in preventing mesh distortion. From these figures it can also be seen that in the ALE method boundary conditions can often be imposed more accurately. Indeed, a nodal point can be prescribed to stay at the end of the contact area with the punch in this ALE calculation. For example, in [4] it was shown that the punch force fluctuated in time in the updated Lagrangian calculation because of an inaccurate description of the boundary condition, whereas the ALE method gave a regular development of the punch force in time.



(a) Initial setup

(b) Updated Lagrangian calculation



(c) ALE calculation

(d) ALE calculation

Figure 1.3: Simulation of upsetting process

However, there are also limitations. Often the ALE method cannot prevent the need for a complete remeshing. The remap of state variables is also a drawback compared to the updated Lagrangian method. When the remap is performed inaccurately the history of the material is not taken into account

properly.
The objective of this thesis is to investigate and improve the numerical performance of the ALE method in the simulation of forming processes. The uncoupled ALE method, which is explained at the end of Chapter 3, consists of first performing an updated Lagrangian step, then a remeshing step and finally a remap of state variables (Figure 1.4). The three steps are described in detail in this thesis. Several techniques are applied to improve the ALE method and they are compared with existing techniques.

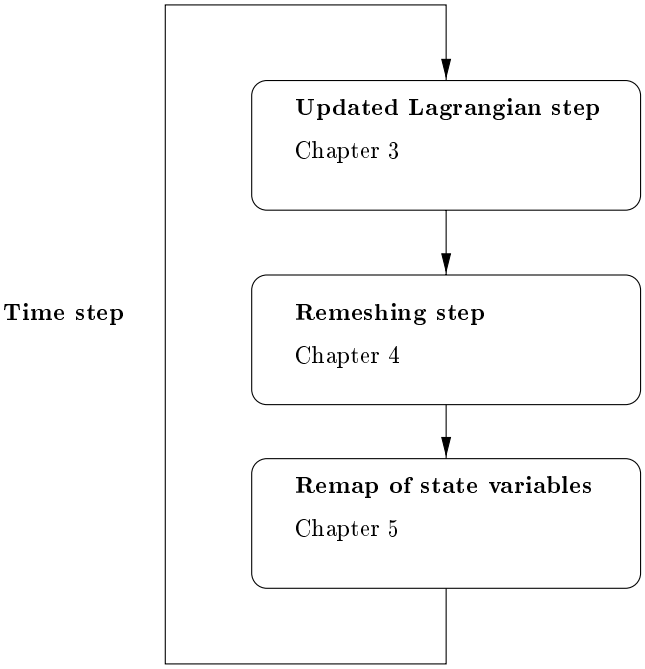


Figure 1.4: A global flow chart of the uncoupled ALE method

1.2 Outline of the thesis

The outline of this thesis is as follows. First in Chapter 2 a concise review of the basic theory of forming processes is given. The kinematics and definitions of stresses and strains are presented. This thesis is restricted to two relatively simple isotropic and isothermal material models: the viscoplastic and the elastoplastic model, which are briefly presented. We chose relatively simple models as we wish to focus on the numerical method. The chosen models represent two classes of material models and they already show the basic

difficulties of a numerical simulation, as will be seen in Chapter 3. At the end of Chapter 2 the equations are assembled, resulting in a definition of an overall mechanical problem.
The mechanical problem cannot be solved analytically, as it is usually too complex. It requires a discretisation of the mechanical problem in time and space. This is the topic of Chapter 3. The weak formulation of the equilibrium equation is the basis for the finite element method. Then the updated Lagrangian method is explained in detail for both the viscoplastic and the elastoplastic material model. Subsequently, the ALE method is discussed.
In Chapter 4 the topic of mesh management in the ALE method is discussed. Two types of meshing methods are described: the transfinite mapping method [46] and schemes based on the Laplace equation, of which the centering method [39] is one possibility. This method is a discretisation of the Laplace equation only in the case of meshes with a homogeneous distribution of elements. We introduce a second Laplacian based scheme, in which the Laplace equation for the mesh velocity is discretised with the finite element method.
The remap of state variables can be described as a convection problem or as an interpolation problem. The relation between the two is discussed in Chapter 5. Often simple interpolation schemes are applied for the remap of state variables, which can result in an inaccurate distribution of the state variables. In this chapter various techniques are applied to improve the solution to the remap problem.
After a detailed description of the ALE method, we apply the ALE method to two academic test cases (in Chapter 6). The remap of state variables is investigated using the Molenkamp test [61]. The second academic test case is introduced for the evaluation of the performance of the ALE method. The analytical solution can be obtained for this test case. A comparison is made between the updated Lagrangian method and the ALE method.
The applicability of the ALE method is demonstrated in the simulation of the following forming processes: extrusion, upsetting and forging. This is also discussed in Chapter 6.
Finally, conclusions are presented which are based on the experiences of the ALE method in the applications and the discussions on the several aspects of the ALE method.

Chapter 2

Continuum mechanics

In this chapter the continuum mechanics and the theory of plasticity used in this work are briefly reviewed, based on [40, 32]. Large displacements and deformations can occur in the simulations of forming processes. Various coordinate systems can be used to describe the state of the material under consideration. In Section 2.1 three coordinate systems are presented: the Lagrangian, the Eulerian and the referential description. The referential description is used in the Arbitrary Lagrangian–Eulerian formulation of Chapter 3. We will give kinematic relations in these coordinates. In Sections 2.2 and 2.3 the terms deformation, strain and stress are defined.
The relation between the stress and the strain for a number of simple isotropic constitutive models is described in Section 2.4. Relatively simple constitutive models are employed in this work to demonstrate the basics of simulation of forming processes. Two essentially different models are presented: the viscoplastic model and the elastoplastic model. The viscoplastic model gives an expression for the stress as a constitutive equation, while the elastoplastic model results in an expression for the *rate* of the stress. This results in two different approaches in the simulation of the forming processes.
Next, the conservation laws that have to be fulfilled are presented. We restrict the discussion to quasi-static processes, which means that inertia terms are neglected. Finally, the conservation equations with the boundary conditions and the constitutive model are summarised.

2.1 Coordinate systems

In the analysis of processes with large displacements and deformations the motion and the deformation can be specified with respect to several frames. Three descriptions are considered [40]:

- the Lagrangian description, where state variables are a function of the material coordinate $\mathbf{X}$,
- the Eulerian description, where state variables are a function of the current coordinate $\mathbf{x}$,
- the referential description, where state variables are a function of the referential coordinate $\boldsymbol{\chi}$.

In the conservation laws the material time derivative and the spatial derivative to $\mathbf{x}$ are required. We will give these properties in the various descriptions.
In the Lagrangian description the material particles $\mathbf{X}$ are marked with the initial position, which will also be referred to as $\mathbf{X} = \mathbf{x}(\mathbf{X}, 0)$. The current position is expressed as a function of the initial position $\mathbf{X}$ and time t, $\mathbf{x}(\mathbf{X}, t)$. The material displacement is defined as:

$$\mathbf{u}(\mathbf{X}, t) = \mathbf{x}(\mathbf{X}, t) - \mathbf{x}(\mathbf{X}, 0). \tag{2.1}$$

The material time derivative is obtained as follows, when a state variable ζ is expressed in terms of the initial coordinates:

$$\frac{\mathrm{d}\zeta(\mathbf{X}, t)}{\mathrm{d}t} = \dot{\zeta}(\mathbf{X}, t) = \left.\frac{\partial \zeta(\mathbf{X}, t)}{\partial t}\right|_{\mathbf{X}}. \tag{2.2}$$

The Lagrangian description is especially attractive for describing physical variables that are associated to material points. The material velocity is:

$$\mathbf{v}(\mathbf{X}, t) = \dot{\mathbf{x}} = \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \left.\frac{\partial \mathbf{x}}{\partial t}\right|_{\mathbf{X}}. \tag{2.3}$$

The spatial derivative with respect to $\mathbf{x}$, which is needed in conservation laws, can be determined using the chain rule,

$$\frac{\partial \zeta(\mathbf{X}, t)}{\partial \mathbf{x}} = \left.\frac{\partial \zeta}{\partial \mathbf{X}}\right|_{t} \cdot \frac{\partial \mathbf{X}}{\partial \mathbf{x}}. \tag{2.4}$$

In the Eulerian description the current position is used to describe the state variables. When the material time derivative is determined a convective term appears:

$$\frac{\mathrm{d}\zeta(\mathbf{x}, t)}{\mathrm{d}t} = \left.\frac{\partial \zeta}{\partial t}\right|_{\mathbf{x}} + \mathbf{v} \cdot \frac{\partial \zeta}{\partial \mathbf{x}}. \tag{2.5}$$

The spatial derivative of a function written in terms of the current coordinates is determined without applying the chain rule as in (2.4).

In the referential description a variable is connected neither to the current position nor to the material. The current position is expressed as a function of the referential coordinate and time, $\mathbf{x}(\boldsymbol{\chi}, t)$. In Chapter 3 it is shown that this referential description is convenient for the discretisation of the equations in the ALE method. The referential displacement becomes

$$\mathbf{u}_{\mathrm{g}}(\boldsymbol{\chi}, t) = \mathbf{x}(\boldsymbol{\chi}, t) - \mathbf{x}(\boldsymbol{\chi}, 0), \tag{2.6}$$

and the referential velocity $\mathbf{v}_{\mathrm{g}}$ is defined as

$$\mathbf{v}_{\mathrm{g}} = \left.\frac{\partial \mathbf{x}(\boldsymbol{\chi}, t)}{\partial t}\right|_{\boldsymbol{\chi}}. \tag{2.7}$$

The subscript g is applied to indicate the referential description, because the referential coordinate will be assigned to the grid points in Chapter 3. The material time derivative of a state variable ζ can also be written according to the referential frame, which again results in a convective term [30]:

$$\frac{\mathrm{d}\zeta(\boldsymbol{\chi}, t)}{\mathrm{d}t} = \left.\frac{\partial \zeta}{\partial t}\right|_{\boldsymbol{\chi}} + \mathbf{v}_c \cdot \frac{\partial \zeta}{\partial \mathbf{x}}, \tag{2.8}$$

where $\mathbf{v}_c = \mathbf{v} - \mathbf{v}_{\mathrm{g}}$ is called the convective velocity. The spatial derivative to the current coordinates is again determined by applying the chain rule:

$$\frac{\partial \zeta(\boldsymbol{\chi}, t)}{\partial \mathbf{x}} = \left.\frac{\partial \zeta}{\partial \boldsymbol{\chi}}\right|_{t} \cdot \frac{\partial \boldsymbol{\chi}}{\partial \mathbf{x}}. \tag{2.9}$$

In Figure 2.1 the relations between the three domains in the referential description are sketched. Since it represents the referential description, the referential coordinate $\boldsymbol{\chi}$ is the independent variable. As a result the mappings onto the spatial domain and the material domain are functions of time, which means that a referential point corresponds at each time to a different spatial point and a different material point. The mappings $\boldsymbol{\Phi}$, $\boldsymbol{\Psi}$ and $(\boldsymbol{\Phi} \circ \boldsymbol{\Psi}^{\Leftrightarrow 1})$ (see Figure 2.1) are one-to-one mappings between the three domains. This means that every point in one of the domains corresponds to one point in the other two domains. Note that the Lagrangian and Eulerian descriptions can be seen as special cases of the referential description. The Lagrangian and the Eulerian description are obtained when $\boldsymbol{\chi}$ equals $\mathbf{X}$ and $\mathbf{x}$ respectively. In the Lagrangian description the mapping $\boldsymbol{\Psi}$ is the identity function, in the Eulerian description the mapping $\boldsymbol{\Phi}$ equals the identity function. For more details the reader is referred to [40, 25, 38]

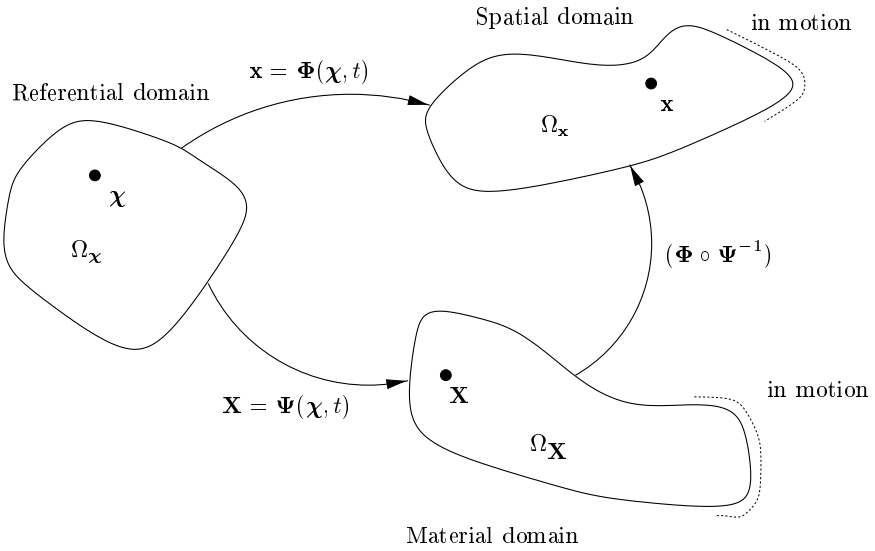


Figure 2.1: Relations between the domains in the referential description.

2.2 Strain definitions

The state of any material particle is described by stresses and strains, which can be defined in one of the coordinate systems mentioned in the previous section.
The *deformation gradient* $\mathbf{F}$ (see Figure 2.2) is defined as the transformation of the initial differential line element $\mathrm{d}\mathbf{X}$ to the deformed differential line element $\mathrm{d}\mathbf{x}$,

$$\mathrm{d}\mathbf{x} = \mathbf{F} \cdot \mathrm{d}\mathbf{X}, \tag{2.10}$$

with

$$\mathbf{F}(\mathbf{X}, t) = \left.\frac{\partial \mathbf{x}}{\partial \mathbf{X}}\right|_t = \mathbf{x} \overleftarrow{\nabla}_0 = \mathbf{1} + \mathbf{u} \cdot \overleftarrow{\nabla}_0, \tag{2.11}$$

where the subscript 0 is the derivative to the initial coordinates $\mathbf{X}$.
The relative change in volume between the undeformed and the deformed state is equal to J:

$$J = \det(\mathbf{F}). \tag{2.12}$$

Polar decomposition of the deformation gradient results in

$$\mathbf{F} = \mathbf{R} \cdot \mathbf{U} = \mathbf{V} \cdot \mathbf{R}, \tag{2.13}$$

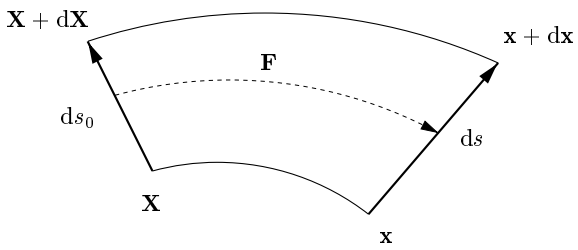


Figure 2.2: Deformation gradient **F**.

where **R** is called the orthogonal *rotation tensor*, **U** the symmetric *right stretch tensor* and **V** the symmetric *left stretch tensor*. The polar decomposition can be considered as first performing a stretch **U** and subsequently a rotation **R** or first a rotation **R** and then a stretch **V**.
The *right* and *left Cauchy–Green tensors* are defined respectively as:

$$\mathbf{C} = \mathbf{U}^2 = \mathbf{F}^{\mathrm{T}} \cdot \mathbf{F}, \tag{2.14}$$

$$\mathbf{B} = \mathbf{V}^2 = \mathbf{F} \cdot \mathbf{F}^{\mathrm{T}}. \tag{2.15}$$

Using (2.14) the length ds of d$\mathbf{x}$ can be expressed in terms of the tensor **C** (see also Figure 2.2),

$$\begin{aligned} \mathrm{d}s^2 &= \mathrm{d}\mathbf{x} \cdot \mathrm{d}\mathbf{x} = (\mathbf{F} \cdot \mathrm{d}\mathbf{X}) \cdot (\mathbf{F} \cdot \mathrm{d}\mathbf{X}) \\ &= \mathrm{d}\mathbf{X} \cdot (\mathbf{F}^{\mathrm{T}} \cdot \mathbf{F}) \cdot \mathrm{d}\mathbf{X} = \mathbf{C} : \mathrm{d}\mathbf{X}\mathrm{d}\mathbf{X}. \end{aligned} \tag{2.16}$$

So **C** can be interpreted as a measure of the stretch of the differential element length d**X**. The initial length ds_0 of d**X** can be expressed in terms of **B** using (2.15):

$$\begin{aligned} \mathrm{d}s_0^2 &= \mathrm{d}\mathbf{X} \cdot \mathrm{d}\mathbf{X} = (\mathbf{F}^{\Leftarrow 1} \cdot \mathrm{d}\mathbf{x}) \cdot (\mathbf{F}^{\Leftarrow 1} \cdot \mathrm{d}\mathbf{x}) \\ &= (\mathrm{d}\mathbf{x} \cdot (\mathbf{F} \cdot \mathbf{F}^{\mathrm{T}})^{\Leftarrow 1}) \cdot \mathrm{d}\mathbf{x} = \mathbf{B}^{\Leftarrow 1} : \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{x}. \end{aligned} \tag{2.17}$$

Generally strain is defined as the length difference of a line element in the deformed and the undeformed state. Various strain definitions exist depending on the description that is used. Using the deformed state d**x**,

$$\mathrm{d}s^2 - \mathrm{d}s_0^2 = \mathrm{d}\mathbf{x} \cdot \mathrm{d}\mathbf{x} - \mathrm{d}\mathbf{X} \cdot \mathrm{d}\mathbf{X} = (\mathbf{1} - \mathbf{B}^{\Leftarrow 1}) : \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{x} = 2\mathbf{e} : \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{x}, \tag{2.18}$$

the *Euler–Almansi* strain tensor **e** is obtained:

$$\mathbf{e} = \frac{1}{2}(\mathbf{1} - \mathbf{B}^{\Leftarrow 1}). \tag{2.19}$$

The Euler–Almansi strain can be expressed in terms of displacements using (2.15) and (2.11):

$$\mathbf{e} = \frac{1}{2}\big(\mathbf{u}\overleftarrow{\nabla} + \overrightarrow{\nabla}\mathbf{u} - (\overrightarrow{\nabla}\mathbf{u})\cdot(\mathbf{u}\overleftarrow{\nabla})\big). \tag{2.20}$$

In case of small deformations the Euler–Almansi strain $\mathbf{e}$ is identical to the classical linear strain tensor $\boldsymbol{\varepsilon}$:

$$\boldsymbol{\varepsilon} = \frac{1}{2}(\mathbf{u}\overleftarrow{\nabla} + \overrightarrow{\nabla}\mathbf{u}). \tag{2.21}$$

Some constitutive models describing plastic deformation require the deformation rate. Therefore the *velocity gradient* $\mathbf{L}$ is defined as:

$$\mathbf{L} = \frac{\partial \mathbf{v}}{\partial \mathbf{x}} = \mathbf{v}\overleftarrow{\nabla}. \tag{2.22}$$

The velocity gradient is split up into two parts, a symmetric part $\mathbf{D}$ and the skew-symmetric part $\mathbf{W}$:

$$\mathbf{L} = \mathbf{D} + \mathbf{W}, \tag{2.23}$$

with

$$\mathbf{D} = \frac{1}{2}(\mathbf{v}\overleftarrow{\nabla} + \overrightarrow{\nabla}\mathbf{v}) \qquad \text{and} \qquad \mathbf{W} = \frac{1}{2}(\mathbf{v}\overleftarrow{\nabla} - \overrightarrow{\nabla}\mathbf{v}). \tag{2.24}$$

$\mathbf{D}$ is called the *rate of deformation tensor* and $\mathbf{W}$ the *spin tensor*.
The velocity gradient can also be expressed in the incremental tensors $\mathbf{R}$ and $\mathbf{U}$, that are expressed in terms of the displacement. Therefore we take the material time derivative of $\mathbf{F}$,

$$\dot{\mathbf{F}} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{x}}{\partial \mathbf{X}}\right) = \frac{\partial \mathbf{v}}{\partial \mathbf{X}} = \frac{\partial \mathbf{v}}{\partial \mathbf{x}} \cdot \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \mathbf{L}\cdot\mathbf{F}. \tag{2.25}$$

With the substitution of (2.13) in (2.25) we obtain:

$$\mathbf{L} = \mathbf{R}\cdot\dot{\mathbf{U}}\cdot\mathbf{U}^{\Leftrightarrow 1}\cdot\mathbf{R}^{\Leftrightarrow 1} + \dot{\mathbf{R}}\cdot\mathbf{R}^{\Leftrightarrow 1}. \tag{2.26}$$

If $\mathbf{R} \approx \mathbf{1}$ and $\mathbf{U} \approx \mathbf{1}$, we can approximate the first term of the right hand side by $\mathbf{D}$ and the second term by $\mathbf{W}$.

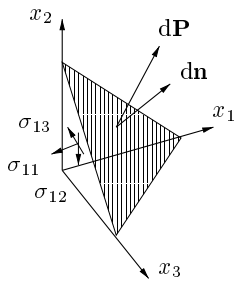


Figure 2.3: Stresses on tetrahedron.

2.3 Stress definitions

In this section the stress will also be defined using the deformed state. The stress vector $\mathbf{p}$ is defined by an infinitesimal force vector $\mathrm{d}\mathbf{P}$ acting on an infinitesimal surface dS:

$$\mathbf{p} = \lim_{\Delta S \to 0} \frac{\Delta \mathbf{P}}{\Delta S} = \frac{\mathrm{d}\mathbf{P}}{\mathrm{d}S}. \tag{2.27}$$

The relation between the stress tensor and the stress vector can be obtained by demanding equilibrium for an infinitesimal tetrahedron (see Figure 2.3). The *Cauchy stress* $\boldsymbol{\sigma}$ is defined in the current coordinates:

$$\mathrm{d}\mathbf{P} = \boldsymbol{\sigma} \cdot \mathbf{n}\, \mathrm{d}S, \tag{2.28}$$

where $\mathbf{n}$ is the normal vector on the surface pointing outwards.
From moment equilibrium follows the symmetry of $\boldsymbol{\sigma}$:

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^{\mathrm{T}}. \tag{2.29}$$

The stress tensor is split up into a deviatoric part $\mathbf{s}$ and a hydrostatic part p for use in constitutive models,

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^{\mathrm{d}} + \frac{1}{3}\mathrm{tr}(\boldsymbol{\sigma})\,\mathbf{1} = \mathbf{s} - p\,\mathbf{1}, \tag{2.30}$$

with $p = -\frac{1}{3}\mathrm{tr}(\boldsymbol{\sigma})$. The time derivative of the stress is often needed to describe the constitutive behaviour. These constitutive models employ an objective time derivative of the stress, which is a time derivative as observed in a frame corotating with the material. In this work the Jaumann time derivative is applied as the objective time derivative of $\boldsymbol{\sigma}$:

$$\overset{\circ}{\boldsymbol{\sigma}} = \dot{\boldsymbol{\sigma}} - \mathbf{W} \cdot \boldsymbol{\sigma} + \boldsymbol{\sigma} \cdot \mathbf{W}. \tag{2.31}$$

For more details on objective time derivatives the reader is referred to [29]. In the next section the elastoplastic model yields an expression for $\overset{\circ}{\boldsymbol{\sigma}}$.

2.4 Constitutive models

The constitutive equation defines the relation between the stresses and strains. It is generally based on experimental observations. The type of constitutive model employed depends on the material under investigation and on the applied loads.

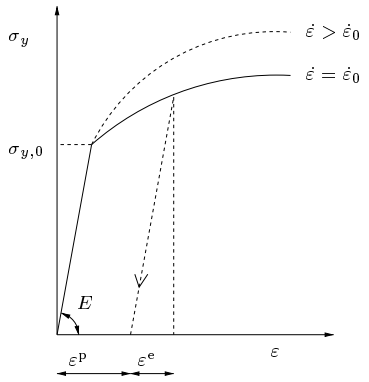


Figure 2.4: Example of stress–strain curves for the one-dimensional tensile test.

In Figure 2.4 the stress–strain relation obtained from a tensile test is illustrated. The material behaves in a linear elastic way up to the initial yield stress $\sigma_{y,0}$ with a slope E, which is called the *Young's modulus*. When the material is unloaded the elastic deformation ε^{e} is totally recovered.
Above the initial yield stress $\sigma_{y,0}$ the material is plastically deformed. The total deformation can be split up into an elastic part and a plastic part:

$$\varepsilon = \varepsilon^{\mathrm{e}} + \varepsilon^{\mathrm{p}}. \tag{2.32}$$

The yield stress σ_y increases when the material is plastically deformed and this phenomenon is called hardening. When the material is unloaded, the stress decreases again linearly according to E. The plastic deformation ε^{p} is not recovered (see Figure 2.4). Upon further plastic deformation the load has to come above the increased yield stress σ_y.
It is also possible that the material behaviour depends on the strain rate $\dot{\varepsilon}$, for example for many polymers or for metals at high temperatures. The stress–strain curve at a higher strain rate will lie above the original curve (see the dashed line of Figure 2.4).

In forming processes we have a multi-dimensional problem with a stress tensor and a strain tensor. In this section we give some constitutive models which give a relation between the Cauchy stress $\boldsymbol{\sigma}$ and the strain $\boldsymbol{\varepsilon}$ or the rate of

deformation $\mathbf{D}$. We will briefly present some relatively simple isotropic and isothermal constitutive models. For a more elaborate overview of constitutive models the reader is referred to [41].
Two types of models are considered in this work: the *viscoplastic* model and the *elastoplastic* model. In Section 2.4.1 the viscoplastic model is given. This is a rate dependent material model, where the elastic deformation is neglected. In Section 2.4.2 the rate independent elastoplastic model is given.

2.4.1 Viscoplastic behaviour

In the case residual stresses or springback phenomena are negligibly small, the elastic deformation can be omitted. In the viscoplastic model the deformation is considered to be completely plastic and the stress is rate dependent. The Cauchy stress tensor is split up into the deviatoric stress tensor $\mathbf{s}$ and the hydrostatic pressure p (see (2.30)). Here the deviatoric stress tensor is described with the isothermal and isotropic *Norton–Hoff* model,

$$\mathbf{s} = 2K(\sqrt{3}\dot{\bar{\varepsilon}})^{m \Leftrightarrow 1}\mathbf{D}, \tag{2.33}$$

where the equivalent strain rate $\dot{\bar{\varepsilon}}$ is defined as:

$$\dot{\bar{\varepsilon}} = \sqrt{\frac{2}{3}\mathbf{D} : \mathbf{D}} \tag{2.34}$$

and m is the *rate sensitivity index*. K is the material consistency parameter and is written as,

$$K = K_0(\bar{\varepsilon}_0 + \bar{\varepsilon})^n, \tag{2.35}$$

where n is the index for the amount of hardening. The equivalent strain $\bar{\varepsilon}$ is defined by the integration of the equivalent strain rate $\dot{\bar{\varepsilon}}$:

$$\dot{\bar{\varepsilon}} = \frac{\mathrm{d}\bar{\varepsilon}}{\mathrm{d}t} \tag{2.36}$$

The material is considered to behave incompressibly, so with (2.24) and (2.23),

$$\overrightarrow{\nabla} \cdot \mathbf{v} = \mathrm{tr}(\mathbf{L}) = \mathrm{tr}(\mathbf{D}) = 0. \tag{2.37}$$

The equivalent stress $\bar{\sigma}$ is needed in order to be able to use the results of one-dimensional tests,

$$\bar{\sigma} = \sqrt{\frac{3}{2}\mathbf{s} : \mathbf{s}} = K\sqrt{3}(\sqrt{3}\dot{\bar{\varepsilon}})^m. \tag{2.38}$$

Several types of model can be derived from the Norton–Hoff model. Two types of model are considered here: the first is the *generalised Newton model* and the second is the *rigid plastic model*.

Generalised Newtonian behaviour

Generalised Newtonian behaviour is obtained when the Norton–Hoff model (2.33) is written in the following form,

$$\mathbf{s} = \eta(\dot{\bar{\varepsilon}})\,\mathbf{D}, \tag{2.39}$$

and the shear coefficient $\eta(\dot{\bar{\varepsilon}})$ does not take into account the hardening of the material ($n = 0$). When the *power law* is applied, we take for $\eta(\dot{\bar{\varepsilon}}) = \eta_0\,\dot{\bar{\varepsilon}}^{m \Leftrightarrow 1}$. For $m = 1$ we obtain the Newtonian fluid:

$$\mathbf{s} = \eta_0\,\mathbf{D}, \tag{2.40}$$

where η_0 is a constant coefficient.
The constitutive model is independent of the total deformation $\bar{\varepsilon}$. Hence, it is not required to integrate (2.36) to obtain the constitutive model. As a result the stress can be determined from the instantaneous velocity field. The model is said to be *path independent*.

Rigid plastic behaviour

The rigid plastic model is a special case of the Norton–Hoff model, *i.e.* when $m = 0$. In that case the equivalent stress (2.38) becomes

$$\bar{\sigma} = K\sqrt{3}. \tag{2.41}$$

From the tensile test we obtain the stress–strain relation $\bar{\sigma}(\bar{\varepsilon}^{\mathrm{p}})$. Here we use this result in an isotropic manner, that is to say the hardening is the same in all directions. Eqs. (2.41) and (2.33), together with the Norton–Hoff model, result in the Von Mises model:

$$\mathbf{s} = \frac{2\bar{\sigma}_{\mathrm{y}}(\bar{\varepsilon}^{\mathrm{p}})}{3\dot{\bar{\varepsilon}}}\mathbf{D}. \tag{2.42}$$

In the case of hardening effects the rigid plastic model is *path dependent*, because $\bar{\varepsilon}^{\mathrm{p}}$ is needed to describe the physical state. This means that $\dot{\bar{\varepsilon}}$ has to be integrated in time in order to be able to describe the material behaviour. The stress cannot be determined directly from the instantaneous velocity field.

2.4.2 Elastoplastic behaviour

In the case of springback phenomena or when the elastic deformation is large, the elastic part of the deformation cannot be neglected. The elastoplastic model is described in this section. For more details the reader is referred to

[32]. In general the total deformation rate $\mathbf{D}$ is split up additively into an elastic deformation rate $\mathbf{D}^{\mathrm{e}}$ and a plastic deformation rate $\mathbf{D}^{\mathrm{p}}$:

$$\mathbf{D} = \mathbf{D}^{\mathrm{e}} + \mathbf{D}^{\mathrm{p}}. \tag{2.43}$$

Hooke's law is taken as a basis:

$$\boldsymbol{\sigma} = \mathbf{E} : \boldsymbol{\varepsilon}. \tag{2.44}$$

The following time derivative of Hooke's law is generally used:

$$\overset{\circ}{\boldsymbol{\sigma}} = \mathbf{E} : (\mathbf{D} - \mathbf{D}^{\mathrm{p}}), \tag{2.45}$$

where $\mathbf{E}$ is the fourth-order elasticity tensor. The plastic rate of deformation is assumed to be perpendicular to the yield surface ϕ (see Figure 2.6):

$$\mathbf{D}^{\mathrm{p}} = \dot{\lambda}\frac{\partial \phi}{\partial \boldsymbol{\sigma}}, \tag{2.46}$$

where $\dot{\lambda}$ is the scale factor for the amount of plastic deformation. Here the isotropic *Von Mises* yield criterion in quadratic form is applied,

$$\phi = \frac{3}{2}\mathbf{s} : \mathbf{s} - \bar{\sigma}_{\mathrm{y}}^2(\bar{\varepsilon}^{\mathrm{p}}) = 0. \tag{2.47}$$

In Figure 2.5 the yield surface is shown. The Von Mises yield surface is a cylinder around the hydrostatic axis.

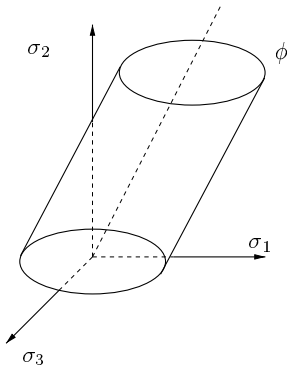


Figure 2.5: Von Mises yield surface.

In Figure 2.6 the projection in the direction of the hydrostatic axis is shown. The function $\bar{\sigma}_{\mathrm{y}}(\bar{\varepsilon}^{\mathrm{p}})$ is determined by the one-dimensional tensile test. After some manipulation one finds

$$\overset{\circ}{\boldsymbol{\sigma}} = [\mathbf{E} - \mathbf{Y}] : \mathbf{D} \tag{2.48}$$

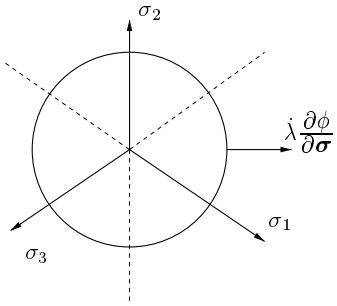


Figure 2.6: Deformation rate perpendicular to the yield surface.

with $\mathbf{Y}$ the yield tensor given by

$$\mathbf{Y} = \frac{\mathbf{E} : \dfrac{\partial\phi}{\partial\boldsymbol{\sigma}}\dfrac{\partial\phi}{\partial\boldsymbol{\sigma}} : \mathbf{E}}{\dfrac{\partial\phi}{\partial\boldsymbol{\sigma}} : \mathbf{E} : \dfrac{\partial\phi}{\partial\boldsymbol{\sigma}} + 4\bar{\sigma}_{\mathrm{y}}^2 \dfrac{\mathrm{d}\bar{\sigma}_{\mathrm{y}}{}^{\mathrm{p}}}{\mathrm{d}\bar{\varepsilon}}}. \tag{2.49}$$

In the next chapter it is shown that it is convenient to split (2.48) into

$$\overset{\circ}{\boldsymbol{\sigma}} = \mathbf{L}^S : \mathbf{D} + C_b \,\mathrm{tr}(\mathbf{D})\mathbf{1}, \tag{2.50}$$

where

$$\mathbf{L}^S = 2G\,\mathbf{I} - \frac{2}{3}G\,\mathbf{11} - \mathbf{Y}, \tag{2.51}$$

and G is the shear modulus and C_b is the bulk modulus:

$$G = \frac{E}{2(1+\nu)} \quad ; \quad C_b = \frac{E}{3(1-2\nu)}. \tag{2.52}$$

NOTE . For an elastoplastic model the complete tensor $\boldsymbol{\sigma}$ is described by the constitutive model in contrast to viscoplastic models. For viscoplastic constitutive models only the deviatoric part of $\boldsymbol{\sigma}$ is given by the constitutive law (see (2.33)). In case of an elastoplastic model the pressure p is described by the constitutive law. The hydrostatic deformation is completely elastic.

2.5 Conservation laws

Conservation laws have to be fulfilled and they are described in this section. In conservative form the conservation laws become:
Continuity equation:

$$\dot{\rho} + \rho\,(\mathbf{v} \cdot \overleftarrow{\nabla}) = 0, \tag{2.53}$$

Momentum equation:

$$\rho\,\dot{\mathbf{v}} = \boldsymbol{\sigma} \cdot \overleftarrow{\nabla} + \rho\,\mathbf{f}. \tag{2.54}$$

where ρ is the density and $\rho\,\mathbf{f}$ is the body force.
In the non-conservative form the material time derivative is given by (2.8) in terms of the referential coordinates $\boldsymbol{\chi}$. In non-conservative form we obtain:
Continuity equation:

$$\left.\frac{\partial \rho}{\partial t}\right|_{\boldsymbol{\chi}} + \mathbf{v}_c \cdot (\overrightarrow{\nabla}\rho) + \rho\,(\mathbf{v} \cdot \overleftarrow{\nabla}) = 0. \tag{2.55}$$

Momentum equation:

$$\rho \left.\frac{\partial \mathbf{v}}{\partial t}\right|_{\boldsymbol{\chi}} + \rho\,\mathbf{v}_c \cdot (\overrightarrow{\nabla}\mathbf{v}) = \boldsymbol{\sigma} \cdot \overleftarrow{\nabla} + \rho\,\mathbf{f}. \tag{2.56}$$

In this work *quasi-static* processes are investigated, which means that inertia terms are neglected. Especially in the Eulerian and the referential description this simplifies the problem, because the convection terms of the left hand side of (2.54) can be neglected. Moreover, body forces $\rho\,\mathbf{f}$ are not taken into account in the problems under investigation for simplicity. As a result we obtain:

$$\boldsymbol{\sigma} \cdot \overleftarrow{\nabla} = \mathbf{0}. \tag{2.57}$$

The equilibrium equation (2.57) has to be fulfilled in the quasi-static case.

2.6 Mechanical problem

In this section the *strong form* of the quasi-static problem is presented. We have to fulfill the equilibrium equation for all the points in the domain $\Omega(t)$,

$$\begin{cases} \text{Find } \mathbf{v}(\mathbf{x},t) \text{ and } \boldsymbol{\sigma}(\mathbf{x},t) \quad \forall\, \mathbf{x}(t) \in \Omega(t) \text{ such that} \\ \boldsymbol{\sigma} \cdot \overleftarrow{\nabla} = \mathbf{0} \end{cases} \tag{2.58}$$

At the same time the appropriate constitutive relations (2.50), (2.42) or (2.39) must be satisfied. These relations give an expression for the stress or the stress rate as a function of the deformation rate $\mathbf{D}$. The deformation rate $\mathbf{D}$ is expressed in terms of the velocity $\mathbf{v}$ by (2.24).
The boundary $,(t)$ is split up into three parts (see Figure 2.7):

$$,(t) = ,_v(t) \cup ,_T(t) \cup ,_C(t) \tag{2.59}$$

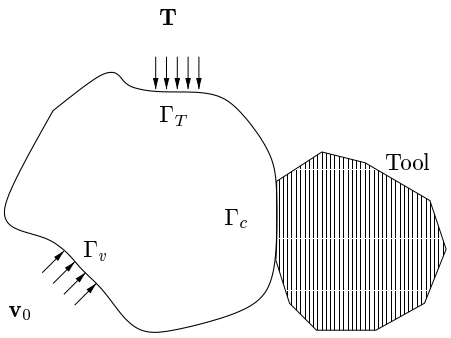


Figure 2.7: Boundary conditions.

where

$$\begin{aligned} &, _v(t) \cap , _T(t) = \emptyset \quad ; \quad , _C(t) \cap , _T(t) = \emptyset \quad ; \\ &, _v(t) \cap , _C(t) = \emptyset \ . \end{aligned} \tag{2.60}$$

On these boundaries the following conditions exist:

$$\begin{cases} \mathbf{v} = \mathbf{v}_0 & \text{on } , _v(t) \\ \boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{T} & \text{on } , _T(t) \\ h(\mathbf{v}) \leq 0 & \text{on } , _c(t) \\ \boldsymbol{\tau} = \boldsymbol{\tau}_{\text{fr}} & \text{on } , _c(t) \end{cases} \tag{2.61}$$

For $\boldsymbol{\tau}_{\text{fr}}$ viscous frictional behaviour is assumed:

$$\boldsymbol{\tau}_{\text{fr}} = -\mu \ (\ (\mathbf{v} - \mathbf{v}_{\text{tool}}) \cdot \mathbf{t} \ ) \ \mathbf{t}, \tag{2.62}$$

where $\mathbf{t}$ is the tangential vector at the boundary $, _C(t)$ and μ is a constant frictional coefficient. In this work we assume unilateral contact, which means that the tool is considered to be rigid. As a result the tool has a velocity $\mathbf{v}_{\text{tool}}$. The function $h(\mathbf{v})$ represents the non-penetration condition of the material into the tool [15] on $, _C(t)$. In a numerical simulation this condition also depends on the time discretisation that is used. The exact description of the non-penetration condition $h(\mathbf{v})$ is given in the next chapter and Appendix A. In words, on $, _v(t)$ essential boundary conditions and on $, _T(t)$ natural boundary conditions are imposed. On $, _c(t)$ we impose a mixed boundary condition: an essential boundary condition in the normal direction and a natural boundary condition in the direction tangential to the boundary.
From now on the mechanical problem of this section is referred to as the problem $\mathcal{P}(\mathbf{v}(\mathbf{x},t), \boldsymbol{\sigma}(\mathbf{x},t))$. In the next chapter the discretisation method of this problem $\mathcal{P}(\mathbf{v}(\mathbf{x},t), \boldsymbol{\sigma}(\mathbf{x},t))$ is discussed.

Chapter 3

Discretisation method

In Section 2.6 the quasi-static problem $\mathcal{P}$ is given. In order to approximate the solution of $\mathcal{P}$, we have to discretise $\mathcal{P}$ in space and in time. First of all, we approximate $\mathcal{P}$ by solving it at discrete time instants t^k. For the spatial discretisation it is important to decide which coordinates $\mathbf{x}$, $\mathbf{X}$ or $\boldsymbol{\chi}$ of Section 2.1 we use to describe the state variables. Each description has its advantages and disadvantages depending on the equations to be solved.
A Eulerian description, in which the state variables are described in terms of $\mathbf{x}$, in general is not good for describing free surfaces. As an example, the *pseudo-concentration* method [53] can be used to describe the movement of the free surface (see Figure 1.2). The pseudo-concentration function, indicating the material, is transported through the fixed element mesh, but often this leads to an inaccurate description of the free surface. In any case, a large number of elements is needed to describe the free surface. Moreover, it is often difficult to describe the boundary conditions accurately. A second difficulty with the Eulerian description is the evaluation of material derivatives. In the evaluation of the material derivative of a state variable ζ a convective term appears (see (2.5)):

$$\frac{\mathrm{d}\zeta}{\mathrm{d}t} = \left.\frac{\partial \zeta}{\partial t}\right|_{\mathbf{X}} + \mathbf{v} \cdot \frac{\partial \zeta}{\partial \mathbf{x}}. \tag{3.1}$$

In a total Lagrangian formulation the state variables are functions of the material coordinate $\mathbf{X}$. The main advantage is that the material derivative can easily be obtained (see (2.2)), because the state variables are connected to material points. However, in the equilibrium equation (2.57) we have the spatial derivative of the Cauchy stress $\boldsymbol{\sigma}$. In a Lagrangian formulation we have *e.g.* the 2$^{\text{nd}}$ Piola–Kirchhoff stress tensor defined in the material coordinate $\mathbf{X}$. As a result we need the mapping $\mathbf{F}$ (2.11) for the solution to the equilibrium equation.

The third possibility is describe the state variables in the referential coordinate $\boldsymbol{\chi}$. This referential coordinate can be chosen arbitrarily. However, the mappings onto the material domain and the spatial domain have to be one-to-one mappings (see Figure 2.1). In the updated Lagrangian method, the referential coordinate is called $\mathbf{X}_{\text{ref}}$ and is labeled with the spatial coordinate $\mathbf{x}_{\text{ref}}$ at the beginning of a time step. The state variables are described in terms of the referential coordinate $\mathbf{X}_{\text{ref}}$, which is connected to a material point during a time step. At the end of each time step the domain is updated with the material velocity field. The material associated quantities are updated easily, and the free surfaces are also accurately described. However, in problems with large deformations the updating of the domain often results in mesh distortion. For this reason the Arbitrary Lagrangian–Eulerian (ALE) method is used in this work. In the ALE method the state variables are described in terms of the referential coordinate $\boldsymbol{\chi}_{\text{ref}}$, which is also labeled with the spatial coordinate $\mathbf{x}_{\text{ref}}$ at the beginning of a time step. However, in contrast to the updated Lagrangian method the referential coordinate $\boldsymbol{\chi}_{\text{ref}}$ is not connected to material points during a time step, but is connected to the grid points that move with the mesh velocity. In Chapter 4 the definition of the mesh velocity, that is defined in a certain way in order to prevent mesh distortion in the ALE method, is discussed. At the end of each time step the domain is updated with the mesh velocity field. In this case we have to take into account convective terms when we evaluate material derivatives, as the state variables are not expressed in terms of the material coordinate $\mathbf{X}_{\text{ref}}$ of the time step (see (2.8)). The free surfaces can still be accurately described by the ALE method. This is shown in Chapter 4.

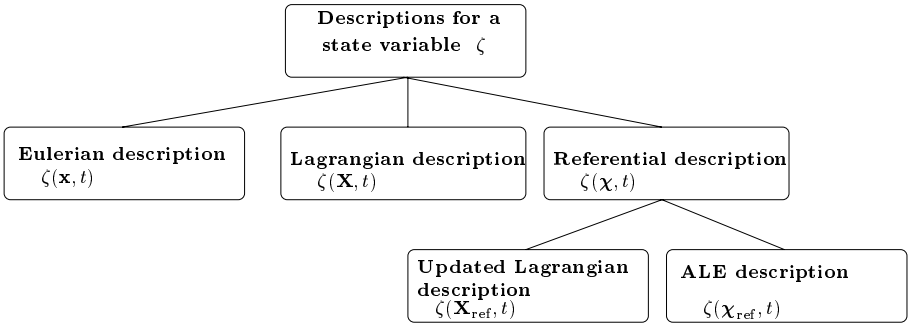


Figure 3.1: Various descriptions for a state variable.

So in both the updated Lagrangian method and the ALE method the reference configuration is the actual configuration at the beginning of the time step. In both methods the referential coordinates are labeled with the spatial

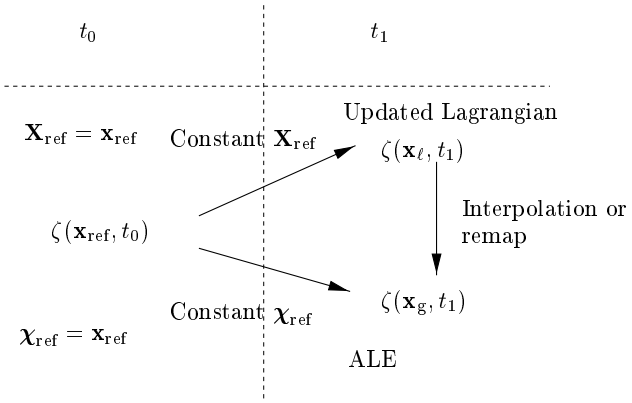


Figure 3.2: Relationships between the updated Lagrangian, the ALE and the uncoupled ALE methods.

coordinates of the beginning of a time step. However, the state variables are connected to different points during the time step.
Due to the spatial discretisation the state variables are known in a discrete set of points $\mathbf{x}_{\text{ref}}$ at the beginning of a time step. This is the initial condition for a next updated Lagrangian step or a next ALE step.
In the updated Lagrangian method the state variables are calculated in the points $\mathbf{x}_\ell$ at time t_1 (see Figure 3.2). The points $\mathbf{x}_{\text{ref}}$ at t_0 and the points $\mathbf{x}_\ell$ at t_1 have the same material coordinate $\mathbf{X}_{\text{ref}}$ of this time step. In other words the points correspond to the same material particles.
In the ALE method the state variables are calculated in the points $\mathbf{x}_{\text{g}}$ at t_1. The points $\mathbf{x}_{\text{ref}}$ at t_0 and the points $\mathbf{x}_{\text{g}}$ at t_1 have the same referential coordinate $\boldsymbol{\chi}_{\text{ref}}$ of this time step. So in the ALE method the solution is calculated in a different set of points in the spatial domain at t_1.
In this work the uncoupled ALE method is applied. In this method the state variables are calculated in the points $\mathbf{x}_{\text{g}}$, but it is performed in two steps. First an updated Lagrangian step is performed, which results in state variables known in the points $\mathbf{x}_\ell$ or the material increments of the state variable. Subsequently the state variables are calculated in the points $\mathbf{x}_{\text{g}}$. This second step can be interpreted as an interpolation problem between two sets of discrete points at t_1.
In this chapter first we give the weak formulation of the mechanical problem $\mathcal{P}$, since the finite element method is used for the spatial discretisation. Subsequently, the updated Lagrangian method is explained in Section 3.2. The spatial discretisation with the finite element method is worked out in detail in Appendix A. The resulting spatially non-linear problem is solved using a predictor–corrector mechanism, which is discussed in Section 3.2.

In Section 3.3 first the ALE method is globally described. Subsequently, the uncoupled ALE method, which is used in this work, is discussed.

3.1 Weak formulation of equilibrium

The *weak form* of the equations is the starting point for the finite element method. In this section this weak form of the mechanical problem $\mathcal{P}$ (see Section 2.6) is derived. In the weak form the equilibrium equation and the natural boundary conditions are weighted with the function $\mathbf{v}^*$, which is a kinematically admissible virtual velocity field, and integrated over the domain:

$$\begin{aligned}\int_{\Omega} \mathbf{v}^* \cdot (\overrightarrow{\nabla} \cdot \boldsymbol{\sigma})\, d\Omega - \int_{,_C} \mathbf{v}^* \cdot (\boldsymbol{\sigma} \cdot \mathbf{n} - \boldsymbol{\tau}_{\mathrm{fr}})\, d, \\ - \int_{,_T} \mathbf{v}^* \cdot (\boldsymbol{\sigma} \cdot \mathbf{n} - \boldsymbol{T})\, d, \ = 0\,, \ \forall \mathbf{v}^* \in \mathcal{V}_C^*(t)\,,\end{aligned} \tag{3.2}$$

where $\mathcal{V}_C^*(t)$ is characterized as the collection of functions satisfying:

$$\begin{cases} \mathbf{v}^* = \mathbf{0} & \text{on } ,_v(t) \\ \mathbf{v}^* \cdot \mathbf{n} = 0 & \text{on } ,_C(t). \end{cases} \tag{3.3}$$

The first term of (3.2) is rewritten using the partial integration rule into

$$\int_{\Omega} \mathbf{v}^* \cdot (\overrightarrow{\nabla} \cdot \boldsymbol{\sigma})\, d\Omega = \int_{\Omega} \overrightarrow{\nabla} \cdot (\mathbf{v}^* \cdot \boldsymbol{\sigma})\, d\Omega - \int_{\Omega} (\mathbf{v}^* \overleftarrow{\nabla}) : \boldsymbol{\sigma}\, d\Omega. \tag{3.4}$$

The first term of the right hand side is changed into a boundary integral with the Gauss theorem into

$$\int_{\Omega} \overrightarrow{\nabla} \cdot (\mathbf{v}^* \cdot \boldsymbol{\sigma})\, d\Omega = \int_{,} (\mathbf{v}^* \cdot \boldsymbol{\sigma}) \cdot \mathbf{n}\, d, \,. \tag{3.5}$$

Since $\mathbf{v}^*$ is in $\mathcal{V}_C^*$ (3.3), the boundary integral over $,$ reduces to

$$\int_{,} (\mathbf{v}^* \cdot \boldsymbol{\sigma}) \cdot \mathbf{n}\, d, \ = \int_{,_C} (\mathbf{v}^* \cdot \boldsymbol{\sigma}) \cdot \mathbf{n}\, d, \ + \int_{,_T} (\mathbf{v}^* \cdot \boldsymbol{\sigma}) \cdot \mathbf{n}\, d, \,, \tag{3.6}$$

where the normal component of $\mathbf{v}^*$ on $,_C$ is equal to zero. Substituting (3.4), (3.5) and (3.6) into (3.2) we obtain the following expression:

$$\int_{\Omega} (\mathbf{v}^* \overleftarrow{\nabla}) : \boldsymbol{\sigma}\, d\Omega = \int_{,_C} \mathbf{v}^* \cdot \boldsymbol{\tau}_{\mathrm{fr}}\, d, \ + \int_{,_T} \mathbf{v}^* \cdot \mathbf{T}\, d, \ , \ \forall \mathbf{v}^* \in \mathcal{V}_C^*(t). \tag{3.7}$$

This is called the weak form, because the equilibrium equations are in the integral form. In this form the order of differentiability for $\boldsymbol{\sigma}$ can be lower than in the strong formulation. However, the differentiability of the weight function is increased by one order.
The weak formulation can also be interpreted as the *virtual work equation*, where the weight functions $\mathbf{v}^*$ are kinematically admissible virtual velocities. Eq. (3.7) can be interpreted as

$$\delta W_{\text{int}} = \delta W_{\text{ext}}, \tag{3.8}$$

where δW_{int} is the internal virtual work and δW_{ext} the external virtual work.

It is not practical to construct *a priori* the velocity fields $\mathbf{v}^*$ and $\mathbf{v}$ that fulfill the boundary condition in the normal direction on $, _C(t)$, because the boundary $, _C(t)$ in contact with the tool is not known *a priori*. So the collection of functions $\mathcal{V}_C^*(t)$ is enlarged to $\mathcal{V}^*(t)$:

$$\mathbf{v}^* = \mathbf{0} \qquad \text{on } , _v(t). \tag{3.9}$$

As a result a constraint for the non-penetration condition $h(\mathbf{v})$ (2.61) has to be added to (3.7) and weighted with λ^*, and also the Lagrange multiplier σ_n appears in the final set of equations:

$$\int_\Omega (\mathbf{v}^* \overleftarrow{\nabla}) : \boldsymbol{\sigma} \; d\Omega = \int_{, C} \mathbf{v}^* \cdot \boldsymbol{\tau}_{\text{fr}} \; d, \; + \int_{, C} \mathbf{v}^* \cdot \mathbf{n} \; \sigma_n \; d, \\ + \int_{, T} \mathbf{v}^* \cdot \mathbf{T} \; d, \; , \; \forall \mathbf{v}^* \in \mathcal{V}^*(t) \tag{3.10}$$

and

$$\int_{, C} \lambda^* \, h(\mathbf{v}) \; d, \; \leq 0 \, , \; \forall \lambda^*.$$

The function $h(\mathbf{v})$ is the non-penetration condition for the normal component of $\mathbf{v}$ on $, _C$. This function is specified in Appendix A. The Lagrange multiplier σ_n is the normal stress component on $, _C$. The normal stress σ_n is an independent variable, which has to be solved. Eq.(3.10) is the weak form of $\mathcal{P}$ and it is also denoted as problem $\mathcal{P}$.

3.2 Updated Lagrangian method

The mechanical problem $\mathcal{P}(\mathbf{x}, \mathbf{v}, \boldsymbol{\sigma})$ is numerically approximated at discrete time instants. In the updated Lagrangian method the state variables are expressed in terms of the referential coordinate $\mathbf{X}_{\text{ref}}$. At the end of each time step the referential situation is updated with the current situation. The referential situation is denoted as Ω_{ref}, $\bar{\varepsilon}_{\text{ref}}$ and $\boldsymbol{\sigma}_{\text{ref}}$ and the current situation at the end of the increment indicated with the superscript $'$. The weak form of equilibrium (3.10) has to be fulfilled for Ω', $\mathbf{v}'$ and $\boldsymbol{\sigma}'$ while knowing the situation at the beginning of the time step:

$$\int_{\Omega'} (\mathbf{v}^* \overleftarrow{\nabla}') : \boldsymbol{\sigma}' \, d\Omega = \int_{,'_C} \mathbf{v}^* \cdot \boldsymbol{\tau}'_{\text{fr}} \, d, \; + \int_{,'_C} \mathbf{v}^* \cdot \mathbf{n}' \sigma'_n \, d, \\ + \int_{,'_T} \mathbf{v}^* \cdot \mathbf{T}' \, d, \; , \; \forall \mathbf{v}^* \in \mathcal{V}^* \tag{3.11}$$

and

$$\int_{,'_C} \lambda^* \, h'(\mathbf{v}) \, d, \; \leq 0 \, , \; \forall \lambda^*.$$

At each time step the velocity $\mathbf{v}_0$ on $,_v$, the velocity of the tool $\mathbf{v}_{\text{tool}}$ acting on $,_C$ and the stresses $\mathbf{T}$ on $,_T$ are prescribed.
In the updated Lagrangian method the situation at the end of a time step is obtained by time integration with constant $\mathbf{X}_{\text{ref}}$:

$$\mathbf{x}' \approx \mathbf{x}_{\text{ref}} + \left(\frac{\partial \mathbf{x}}{\partial t} \right)_{\mathbf{X}_{\text{ref}}} \Delta t = \mathbf{x}_{\text{ref}} + \dot{\mathbf{x}} \, \Delta t = \mathbf{x}_{\text{ref}} + \Delta \mathbf{u}, \tag{3.12}$$

$$\bar{\varepsilon}' \approx \bar{\varepsilon}_{\text{ref}} + \dot{\bar{\varepsilon}} \, \Delta t = \bar{\varepsilon}_{\text{ref}} + \Delta \bar{\varepsilon}, \tag{3.13}$$

$$\boldsymbol{\sigma}' \approx \boldsymbol{\sigma}_{\text{ref}} + \dot{\boldsymbol{\sigma}} \, \Delta t = \boldsymbol{\sigma}_{\text{ref}} + \Delta \boldsymbol{\sigma}. \tag{3.14}$$

Here $\Delta \mathbf{u}$, $\Delta \bar{\varepsilon}$ and $\Delta \boldsymbol{\sigma}$ have to be approximated for a time step. Since the domain is updated with the material displacement $\Delta \mathbf{u}$, the free surface is easily calculated. In the updated Lagrangian method material time derivatives are easily integrated, because the state variables are integrated in time with constant coordinate $\mathbf{X}_{\text{ref}}$.

The updated Lagrangian method becomes schematically:

1. At $t = 0$ the initial situation with $\Omega, \boldsymbol{\sigma}, \bar{\varepsilon}$ is taken as the referential situation $\Omega_{\text{ref}}, \boldsymbol{\sigma}_{\text{ref}}, \bar{\varepsilon}_{\text{ref}}$.
2. At each time step the problem $\mathcal{P}(\mathbf{x}', \mathbf{v}', \boldsymbol{\sigma}')$ is solved with constant $\mathbf{X}_{\text{ref}}$ while knowing the referential domain Ω_{ref} and the state variables $\boldsymbol{\sigma}_{\text{ref}}$ and $\bar{\varepsilon}_{\text{ref}}$.
3. Update the domain and the state variables, so $\Omega', \boldsymbol{\sigma}'$ and $\bar{\varepsilon}' \Longrightarrow \Omega_{\text{ref}}, \boldsymbol{\sigma}_{\text{ref}}$ and $\bar{\varepsilon}_{\text{ref}}$. Go to step 2 for the next time step.

In Sections 3.2.1 and 3.2.2 the following procedure is applied in order to solve $\mathcal{P}(\mathbf{x}', \mathbf{v}', \boldsymbol{\sigma}')$ at each time step. First of all, the stress $\boldsymbol{\sigma}'$ is expressed in terms of $\mathbf{v}'$ by one of the constitutive models of section 2.4. As a result (3.11) becomes a non-linear spatial problem in terms of $\mathbf{v}'$. In the remainder of this section we focus on solving this non-linear problem.
The second step is the spatial discretisation of (3.11) with the finite element method, which results in a non-linear problem $\mathcal{P}_h$ in terms of the nodal point velocities $\mathbf{V}$. In this work we use the 4-node quadrilateral plane strain element and the 6-node triangular plane strain element (see Figure 3.3). For both the elements a constant pressure field is assumed. The pressure variable p is eliminated from the system of equations using the penalty method, which is worked out in Appendix A.

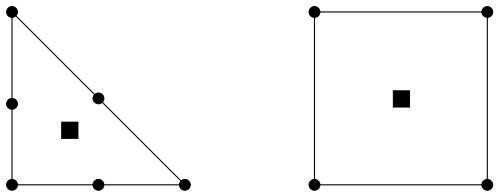


Figure 3.3: Quadrilateral and triangular plane strain element.

After the discretisation with the finite element method we obtain the problem $\boldsymbol{\mathcal{P}}_h$ in terms of the nodal point velocities $\mathbf{V}$ that is expressed in the following form:

$$\boldsymbol{\mathcal{P}}_h(\mathbf{V}) = \mathbf{K} \cdot \mathbf{V} - \mathbf{F} = \mathbf{0}, \tag{3.15}$$

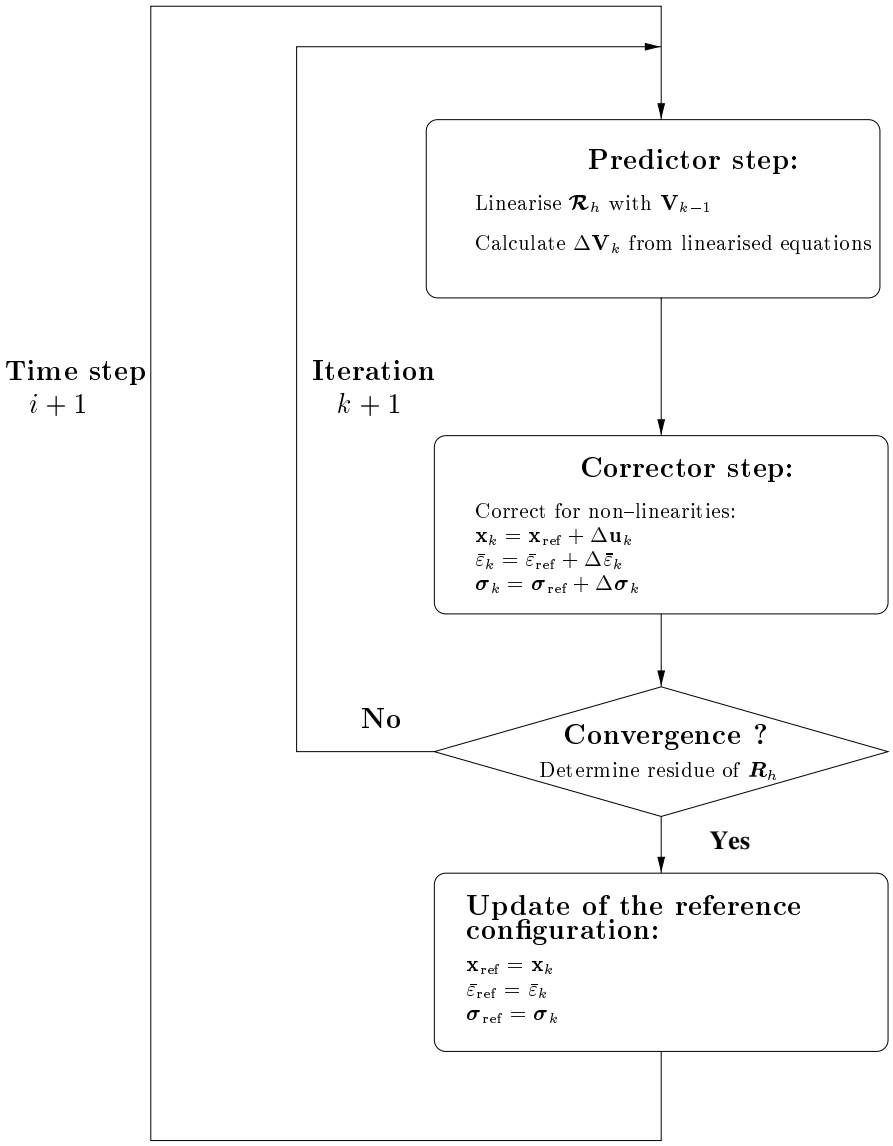


Figure 3.4: Predictor–corrector method.

where the terms $\mathbf{K}$ and $\mathbf{F}$ also depend on the nodal velocities $\mathbf{V}$. This non-linear problem $\boldsymbol{\mathcal{P}}_h(\mathbf{V})$ is numerically solved with an iteration method. During the iterations the set of equations $\boldsymbol{\mathcal{P}}_h(\mathbf{V})$ are approximated by the set of equations $\boldsymbol{\mathcal{R}}_h(\mathbf{V})$ This iteration method consists of a predictor and a corrector step, which we discuss now.

Predictor step

In the predictor step $\boldsymbol{\mathcal{R}}_h$ is linearised with respect to $\mathbf{V}$ and subsequently $\Delta\mathbf{V}$ is calculated from this linearised system of equations (see Figure 3.5):

$$\boldsymbol{\mathcal{R}}_h(\mathbf{V}+\Delta\mathbf{V}) \approx \boldsymbol{\mathcal{R}}_h(\mathbf{V}) + \frac{\partial \boldsymbol{\mathcal{R}}_h(\mathbf{V})}{\partial \mathbf{V}} \cdot \Delta\mathbf{V}, \tag{3.16}$$

where $\frac{\partial \boldsymbol{\mathcal{R}}_h}{\partial \mathbf{V}}$ is called the tangential stiffness matrix.

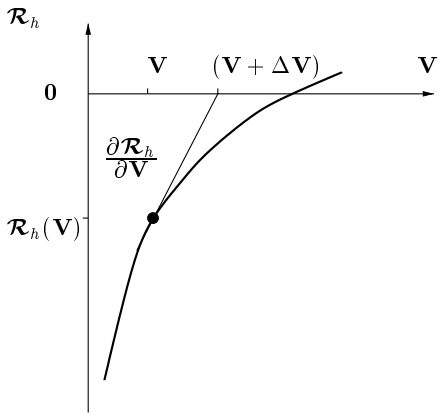


Figure 3.5: Schematic representation of the predictor step.

In the k^{th} iteration the following linear system of equations is solved:

$$\boldsymbol{\mathcal{R}}_h(\mathbf{V}_{k\Leftarrow 1}+\Delta\mathbf{V}_k) \approx \boldsymbol{\mathcal{R}}_h(\mathbf{V}_{k\Leftarrow 1}) + \frac{\partial \boldsymbol{\mathcal{R}}_h(\mathbf{V}_{k\Leftarrow 1})}{\partial \mathbf{V}} \cdot \Delta\mathbf{V}_k = \mathbf{0}. \tag{3.17}$$

Using (3.15), the following approximation is performed:

$$\left(\frac{\partial \boldsymbol{\mathcal{R}}_h}{\partial \mathbf{V}}\right)_{k\Leftarrow 1} \approx \mathbf{K}_{k\Leftarrow 1}. \tag{3.18}$$

As already mentioned the terms $\mathbf{K}$ and $\mathbf{F}$ depend on the solution $\mathbf{V}$. Together with (3.15) the following linear system of equations is obtained:

$$\mathbf{K}_{k\Leftarrow 1} \cdot \Delta\mathbf{V}_k = \mathbf{F}_{k\Leftarrow 1} - \mathbf{K}_{k\Leftarrow 1} \cdot \mathbf{V}_{k\Leftarrow 1}, \tag{3.19}$$

from which $\Delta\mathbf{V}_k$ can be calculated.

Corrector step

In the corrector step we correct for the non-linearity of the problem. First the velocity is updated:

$$\mathbf{V}_k = \mathbf{V}_{k \Leftrightarrow 1} + \Delta \mathbf{V}_k. \tag{3.20}$$

Then the domain and the stress are updated using the velocity $\mathbf{V}_k$ and the set of equations becomes:

$$\boldsymbol{\mathcal{R}}_h(\mathbf{V}_k) = \mathbf{F}_k - \mathbf{K}_k \cdot \mathbf{V}_k. \tag{3.21}$$

The iteration process has converged when it satisfies:

$$\frac{\| \boldsymbol{\mathcal{R}}_h(\mathbf{V}_k) \|}{\| \mathbf{K}_k \cdot \mathbf{V}_k \|} \leq C_R, \tag{3.22}$$

for some (small) C_R (the unbalance criterion).
If (3.22) is not satisfied, a next iteration is performed with the updated situation until convergence is reached (see Figure 3.4).

In the following two sections the discretisation method is discussed in more detail in the case of a viscoplastic and an elastoplastic model. Since the approach differs essentially in both cases, they are treated separately in Sections 3.2.1 and 3.2.2.

3.2.1 Viscoplastic model

The objective of this section is to describe the spatial discretisation of (3.11) in the case of a viscoplastic model and a material velocity field that is assumed to be incompressible. For clarity the indication $'$ is omitted in the remainder of this section.
The kinematically admissible velocity field $\mathbf{v}$ has to be in the set of functions $\mathcal{V}_{\text{div}}$:

$$\begin{cases} \mathbf{v} = \mathbf{v}_0 & \text{on } , _v, \\ (\overrightarrow{\nabla} \cdot \mathbf{v}) = 0 & \text{on } \Omega. \end{cases} \tag{3.23}$$

Consequently, the kinematically virtual velocity field $\mathbf{v}^*$ has to be in $\mathcal{V}^*_{\text{div}}$:

$$\begin{cases} \mathbf{v}^* = \mathbf{0} & \text{on } , _v, \\ (\overrightarrow{\nabla} \cdot \mathbf{v}^*) = 0 & \text{on } \Omega. \end{cases} \tag{3.24}$$

Using the symmetry of $\boldsymbol{\sigma}$ and $(\overrightarrow{\nabla} \cdot \mathbf{v}^*) = 0$, the left hand side of (3.11) is rewritten as follows:

$$\int_{\Omega} (\mathbf{v}^* \overleftarrow{\nabla}) : \boldsymbol{\sigma} \; d\Omega = \int_{\Omega} \mathbf{L}^* : \boldsymbol{\sigma} \; d\Omega = \int_{\Omega} \mathbf{D}^* : \mathbf{s} \; d\Omega. \tag{3.25}$$

So when (3.25) is substituted into (3.11) the hydrostatic pressure p does not appear into virtual work equation:

$$\begin{aligned} \int_{\Omega} \mathbf{D}^* : \mathbf{s} \; d\Omega = & \int_{,\, C} \mathbf{v}^* \cdot \boldsymbol{\tau}_{\mathrm{fr}} \; d, \; + \int_{,\, C} \mathbf{v}^* \cdot \mathbf{n} \, \sigma_n \; d, \\ & + \int_{,\, T} \mathbf{v}^* \cdot \boldsymbol{T} \; d, \; , \quad \forall \mathbf{v}^* \in \mathcal{V}^*_{\mathrm{div}} \end{aligned} \tag{3.26}$$

and

$$\int_{,\, C} \lambda^* \, h(\mathbf{v}) \; d, \; \leq 0 \, , \quad \forall \lambda^*.$$

The deviatoric stress $\mathbf{s}$ is determined by the instantaneous velocity field, so the Norton–Hoff model (2.33) is substituted directly into (3.26). The functions $\boldsymbol{\tau}_{\mathrm{fr}}$ (2.62) and $h(\mathbf{v})$ (A.15) are also described in terms of the velocity field, but these functions are not substituted here for the sake of clarity:

$$\begin{aligned} \int_{\Omega} 2K(\sqrt{3}\dot{\bar{\varepsilon}})^{m \Leftrightarrow 1} \mathbf{D}^* : \mathbf{D} \; d\Omega = & \int_{,\, C} \mathbf{v}^* \cdot \boldsymbol{\tau}_{\mathrm{fr}} \; d, \\ & + \int_{,} \mathbf{v}^* \cdot \mathbf{n} \, \sigma_n \; d, \; + \int_{,\, T} \mathbf{v}^* \cdot \boldsymbol{T} \; d, \; , \quad \forall \mathbf{v}^* \in \mathcal{V}^*_{\mathrm{div}} \end{aligned} \tag{3.27}$$

and

$$\int_{,\, C} \lambda^* \, h(\mathbf{v}) \; d, \; \leq 0 \, , \quad \forall \lambda^*.$$

Eq. (3.27) is completely described in the velocity field $\mathbf{v}$, which has to fulfill the boundary conditions on $,_v$ and the incompressibility condition (see (3.23)). However, it is inconvenient to construct velocity fields $\mathbf{v}$ and $\mathbf{v}^*$ that fulfill *a priori* the incompressibility condition. Therefore the incompressibility

condition is added as a constraint in the same way as is done for the non-penetration condition $h(\mathbf{v})$. As a result the following Lagrange multiplier problem is obtained:

$$\int_{\Omega} 2K(\sqrt{3}\dot{\bar{\varepsilon}})^{m \Leftrightarrow 1}\mathbf{D}^* : \mathbf{D}\; d\Omega - \int_{\Omega} (\overrightarrow{\nabla} \cdot \mathbf{v}^*)\; p\; d\Omega =$$

$$\int_{,\,C} \mathbf{v}^* \cdot \boldsymbol{\tau}_{\mathrm{fr}}\; d, \; + \int_{,\,C} \mathbf{v}^* \cdot \mathbf{n}\, \sigma_n\; d, \; + \int_{,\,T} \mathbf{v}^* \cdot \boldsymbol{T}\; d, \qquad \forall \mathbf{v}^* \in \mathcal{V}^* \tag{3.28}$$

and

$$\int_{,\,C} \lambda^*\, h(\mathbf{v})\; d, \; \leq 0\,, \;\; \forall \lambda^*,$$

$$\int_{\Omega} p^*(\overrightarrow{\nabla} \cdot \mathbf{v})\; d\Omega = 0\,, \;\; \forall p^*.$$

where the Lagrange multiplier p is the hydrostatic pressure, which is an independent variable. This mixed formulation can also be interpreted as a minimisation problem with constraints [64].

In Appendix A the problem (3.28) is spatially discretised with the finite element method and a penalty method is used for both Lagrange multipliers p and σ_n. They are locally eliminated, which results in a system of equations that can globally be written as:

$$\boldsymbol{\mathcal{R}}_h(\mathbf{V}) = \mathbf{K} \cdot \mathbf{V} - \mathbf{F} = \mathbf{0}. \tag{3.29}$$

This discretised problem $\mathbf{R}_h(\mathbf{V})$ is non–linear in $\mathbf{V}$. The predictor–corrector method described by (3.16) to (3.22) is applied to solve this system of equations.

3.2.2 Elastoplastic model

Again the aim is to solve (3.11), but now using an elastoplastic constitutive model:

$$\int_{\Omega'} (\mathbf{v}^* \overleftarrow{\nabla}') : \boldsymbol{\sigma}'\; d\Omega = \int_{,\,C'} \mathbf{v}^* \cdot \boldsymbol{\tau}'_{\mathrm{fr}}\; d, \; + \int_{,\,C'} \mathbf{v}^* \cdot \mathbf{n}'\, \sigma'_n\; d,$$

$$+ \int_{,\,T'} \mathbf{v}^* \cdot \mathbf{T}'\; d, \;\; , \;\; \forall \mathbf{v}^* \in \mathcal{V}^* \tag{3.30}$$

and

$$\int_{,'_C} \lambda^* \, h'(\mathbf{v}) \; d, \; \leq 0 \,, \quad \forall \lambda^*.$$

In the elastoplastic case the Jaumann rate of the Cauchy stress $\overset{\circ}{\boldsymbol{\sigma}}$ follows from the velocity $\mathbf{v}$ (see (2.50)). This in contrast with the viscoplastic model where $\boldsymbol{\sigma}'$ is calculated using the velocity $\mathbf{v}$ by (2.33). This results in a different approach in the case of an elastoplastic model. The Cauchy stress $\boldsymbol{\sigma}'$ is now also depending on the history of the Cauchy stress $\boldsymbol{\sigma}_{\text{ref}}$. In order to obtain $\boldsymbol{\sigma}'$ in an updated Lagrangian method, the material time derivative $\dot{\boldsymbol{\sigma}}$ has to be integrated in time (see (3.14)).
In most cases the elastic deformation is negligibly small compared to the plastic deformation. Since the plastic deformation is assumed to be incompressible, the problem is said to be *quasi-incompressible*. As a result volume locking can occur, which means that too much incompressibility constraints are applied compared to the number of unknowns of the velocity $\mathbf{V}$. In order to prevent volume locking, the hydrostatic pressure is reduced integrated [64]. This means that the hydrostatic pressure p is integrated with a smaller number of integration points than the deviatoric stress $\mathbf{s}$. For this reason the stress tensor is split into the deviatoric stress and the hydrostatic pressure:

$$\boldsymbol{\sigma}' = \mathbf{s}' - p'\mathbf{1}. \tag{3.31}$$

Using the elastoplastic model (2.50), the constitutive equation for the rate of the pressure, that is purely elastic, is written as

$$\dot{p} = -C_b \; \text{tr}(\mathbf{D}). \tag{3.32}$$

The weak form, in which the incremental constitutive equation for p' is added as a constraint, becomes

$$\begin{aligned} \int_{\Omega'} (\mathbf{v}^* \overleftarrow{\nabla}) : (\mathbf{s}' - p'\mathbf{1}) \; d\Omega = & \int_{,'_C} \mathbf{v}^* \cdot \boldsymbol{\tau}'_{\text{fr}} \; d, \\ & + \int_{,'_C} \mathbf{v}^* \cdot \mathbf{n}' \sigma'_n \; d, \; + \int_{,'_T} \mathbf{v}^* \cdot \mathbf{T}' \; d, \; , \quad \forall \mathbf{v}^* \in \mathcal{V}^* \end{aligned} \tag{3.33}$$

and

$$\int_{\Omega'} p^* (p' - p_{\text{ref}} + C_b \; \text{tr}(\mathbf{D}) \; \Delta t) \; d\Omega = 0 \,, \quad \forall p^*,$$

$$\int_{,'_C} \lambda^* \, h'(\mathbf{v}) \; d, \; \leq 0 \,, \quad \forall \lambda^*.$$

In Appendix A the finite element discretisation is written down in detail, which results in the following system of equations:

$$\boldsymbol{\mathcal{R}}_h(\mathbf{V}) = \mathbf{Q}_M + \mathbf{K}_C \cdot (\mathbf{V} - \mathbf{V}_{\text{tool}}) - \mathbf{F}_T + \mathbf{F}_C = \mathbf{0}, \tag{3.34}$$

where $\mathbf{Q}_M$ stands for the discretisation of δW_{int} (the left hand side of (3.33)), which includes the time integration of the stress rate. $\boldsymbol{\mathcal{R}}_h$ is solved with the predictor–corrector method, which is discussed next.

Predictor step

In this section we focus on the linearisation of the term $\mathbf{Q}_M$, since the other terms of (3.34) can be treated in the same way as in the case with a viscoplastic model (see Section 3.2.1 and Appendix A). In order to obtain a stiffness matrix of δW_{int}, we have to determine the part of (3.33) that depends on the material velocity $\mathbf{v}$. Therefore the left hand side of (3.33) is rewritten in the following form:

$$\delta W'_{\text{int}} \approx \delta W_{\text{int,ref}} + \delta \dot{W}_{\text{int}} \ \Delta t, \tag{3.35}$$

where $\delta W_{\text{int,ref}}$ is the internal virtual work at the beginning of the time step. The second term on the right hand side depends on $\mathbf{v}$ and this term is used for the linearisation of the predictor step. The second term on the right hand side is the so-called *weak rate form* of internal virtual work. The weak rate form is considered in order to obtain the stiffness matrix needed for the linearisation of the predictor step. The same method is applied as in [38], where the linearised expression of virtual work in the ALE method is derived. In order to determine the time derivative at a constant $\mathbf{X}_{\text{ref}}$, the integral is first transformed to the updated Lagrangian coordinates (using (2.11) and (2.12)):

$$\delta W_{\text{int}} = \int_{\Omega} (\mathbf{v}^* \overleftarrow{\nabla}) : \boldsymbol{\sigma} d\Omega = \int_{\Omega} \left(\frac{\partial \mathbf{v}^*}{\partial \mathbf{X}_{\text{ref}}} \cdot \frac{\partial \mathbf{X}_{\text{ref}}}{\partial \mathbf{x}} \right) : \boldsymbol{\sigma} \ d\Omega \tag{3.36}$$

$$= \int_{\Omega_{\mathbf{X}_{\text{ref}}}} (\mathbf{v}^* \overleftarrow{\nabla}_{\mathbf{X}_{\text{ref}}}) : (\boldsymbol{\sigma} \cdot \mathbf{F}^{\Leftrightarrow T}) J \ d\Omega. \tag{3.37}$$

The weak rate formulation becomes, neglecting the time derivative of the virtual velocity $\mathbf{v}^*$:

$$\delta \dot{W}_{\text{int}} = \int_{\Omega_{\mathbf{X}_{\text{ref}}}} (\mathbf{v}^* \overleftarrow{\nabla}_{\mathbf{X}_{\text{ref}}}) : \left(\dot{\boldsymbol{\sigma}} \ \mathbf{F}^{\Leftrightarrow T} J + \boldsymbol{\sigma} \cdot \dot{\mathbf{F}}^{\Leftrightarrow T} J + \boldsymbol{\sigma} \cdot \mathbf{F}^{\Leftrightarrow T} \dot{J} \right) \ d\Omega. \tag{3.38}$$

Taking the time derivative of $\mathbf{v}^*$ would result in weighting the equilibrium equations of the beginning of the time step [26]. Therefore $\mathbf{v}^*$ is taken independent of time. In the case the linearisation is less accurate the convergence of the predictor–corrector mechanism is slower. In the corrector step the original weak form (3.33) is considered.
Now we determine the time derivatives of (3.38). First, $\dot{\boldsymbol{\sigma}}$ is obtained by the elastoplastic model (2.50), which gives an expression for the Jaumann derivative of the Cauchy tensor $\overset{\circ}{\boldsymbol{\sigma}}$ (2.31):

$$\overset{\circ}{\boldsymbol{\sigma}} = \dot{\boldsymbol{\sigma}} - \mathbf{W}\cdot\boldsymbol{\sigma} + \boldsymbol{\sigma}\cdot\mathbf{W} = \mathbf{L}^S : \mathbf{D} - \dot{p}\,\mathbf{1}. \tag{3.39}$$

Second, we use the following relations:

$$\dot{J} = \operatorname{tr}(\mathbf{D})J \qquad ; \qquad \dot{\mathbf{F}}^{\Leftrightarrow T} = -\mathbf{L}^T\cdot\mathbf{F}^{\Leftrightarrow T}. \tag{3.40}$$

Substituting (3.39) and (3.40) into (3.38), we obtain:

$$\begin{aligned}\delta\dot{W}_{\text{int}} = \int_{\Omega_{\mathbf{X}_{\text{ref}}}} (\mathbf{v}^* \overleftarrow{\nabla}_{\mathbf{X}_{\text{ref}}}) : \big((\mathbf{W}\cdot\boldsymbol{\sigma} - \boldsymbol{\sigma}\cdot\mathbf{W} + \mathbf{L}^S : \mathbf{D} - \dot{p}\,\mathbf{1} \\ - \boldsymbol{\sigma}\cdot\mathbf{L}^T + \boldsymbol{\sigma}\operatorname{tr}(\mathbf{D})\big)\cdot\mathbf{F}^{\Leftrightarrow T} J\; d\Omega.\end{aligned} \tag{3.41}$$

Using (2.24) and some manipulations results in:

$$\begin{aligned}\delta\dot{W}_{\text{int}} = \int_{\Omega_{\mathbf{X}_{\text{ref}}}} (\mathbf{v}^* \overleftarrow{\nabla}_{\mathbf{X}_{\text{ref}}}) : (\mathbf{L}\cdot\boldsymbol{\sigma} - \mathbf{D}\cdot\boldsymbol{\sigma} - \boldsymbol{\sigma}\cdot\mathbf{D} + \mathbf{L}^S : \mathbf{D} \\ - \dot{p}\,\mathbf{1} + \boldsymbol{\sigma}\operatorname{tr}(\mathbf{D}))\cdot\mathbf{F}^{\Leftrightarrow T} J\; d\Omega.\end{aligned} \tag{3.42}$$

The last step in the linearisation is the transformation back to the current coordinates:

$$\begin{aligned}\delta\dot{W}_{\text{int}} = &\int_\Omega \mathbf{L}^* : (\mathbf{L}\cdot\boldsymbol{\sigma})\; d\Omega - \int_\Omega (\operatorname{tr}(\mathbf{D}^*)\,\dot{p})\; d\Omega \\ &+ \int_\Omega \mathbf{D}^* : (-\mathbf{I}\cdot\boldsymbol{\sigma} - \boldsymbol{\sigma}\cdot\mathbf{I} + \mathbf{L}^S + \boldsymbol{\sigma}\mathbf{1}) : \mathbf{D}\; d\Omega.\end{aligned} \tag{3.43}$$

The constraint for the constitutive equation of the pressure (3.33) gives the expression for $\dot{p}$:

$$\int_\Omega p^*(\dot{p} + C_b\,\operatorname{tr}(\mathbf{D}))\; d\Omega = 0, \;\; \forall p^* \tag{3.44}$$

In Appendix A the finite element discretisation of this linearised expression for internal virtual work (3.43) and the constraint for the pressure (3.44) is written down in detail. The pressure variable is locally eliminated which results in a reduced integration scheme with the nodal velocity $\mathbf{V}$ as the only variable to solve. Using this linearisation for $\mathbf{Q}_M$, we use the stiffness matrix $\mathbf{K}_M$ in the linearisation of the predictor step. The linearised expression becomes:

$$\delta\dot{W}_{\text{int}} = \mathbf{K}_M \cdot \mathbf{V} \tag{3.45}$$

The other terms of $\boldsymbol{\mathcal{R}}_h(\mathbf{V})$ in (3.34) are linearised as in the case with a viscoplastic model (see Section 3.2.1 and Appendix A). As a result we have a system of equations of the form (3.15). From this linearised system of equations the nodal velocities are solved in the predictor step. Next we correct for the non-linearities of $\boldsymbol{\mathcal{R}}_h(\mathbf{V})$ in the corrector step, which is discussed in the following section.

Corrector step

In the corrector step, first the velocity is updated with the solution of the predictor step. The weak formulation of equilibrium is calculated with this updated velocity

$$\boldsymbol{\mathcal{R}}_h(\mathbf{V}) = \mathbf{Q}_M - \mathbf{K}_C \cdot (\mathbf{V} - \mathbf{V}_{\text{tool}}) - \mathbf{F} + \mathbf{F}_C = \mathbf{0}. \tag{3.46}$$

In this section the updating of the internal work $\mathbf{Q}_M$ is discussed. The other terms of (3.46) are updated as in the corrector step in the case of a viscoplastic model.
$\mathbf{Q}_M$ is written in the following form (see Appendix A):

$$\mathbf{Q}_M = \int_{\Omega',e} \left((\mathbf{B}_{\text{d}})^T : \mathbf{s}' - (\mathbf{B}_{\text{i}})^T \; p' \right) d\Omega. \tag{3.47}$$

where $\mathbf{B}_{\text{i}}$ represents the isotropic part of $\mathbf{B}$ and $\mathbf{B}_{\text{d}}$ the deviatoric part of $\mathbf{B}$. The term with the pressure p' is reduced integrated.
The coordinates are updated with the material displacement:

$$\mathbf{x}' = \mathbf{x}_{\text{ref}} + \Delta\mathbf{u}. \tag{3.48}$$

where $\Delta\mathbf{u} = (\mathbf{V}_{k \Leftrightarrow 1} + \Delta\mathbf{V}_k)\Delta t$ for the k^{th} iteration. In order to determine spatial derivatives we use the domain half-way along the time step, $\mathbf{x}_{\frac{1}{2}}$:

$$\mathbf{x}_{\frac{1}{2}} = \mathbf{x}_{\text{ref}} + \frac{1}{2}\Delta\mathbf{u}, \tag{3.49}$$

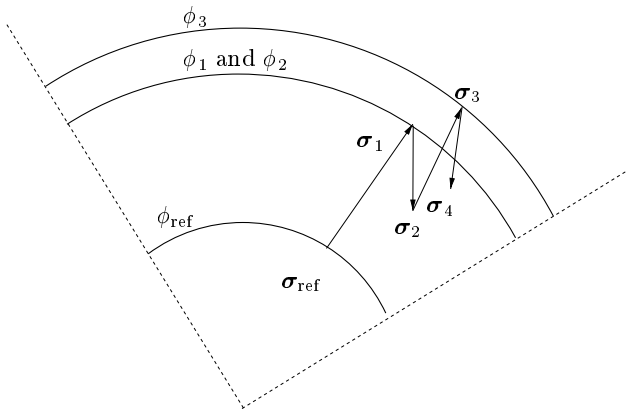


Figure 3.6: Schematic presentation of the integration of the stress with path dependent behaviour due to the iteration method.

where the subscript $\frac{1}{2}$ indicates this domain. The total strain increment $\Delta\boldsymbol{\varepsilon}$ is approximated as:

$$\Delta\boldsymbol{\varepsilon} = \frac{1}{2}(\Delta\mathbf{u}\overleftarrow{\nabla}_{\frac{1}{2}} + \overrightarrow{\nabla}_{\frac{1}{2}}\Delta\mathbf{u}). \tag{3.50}$$

The stress $\boldsymbol{\sigma}'$ is obtained by integration of the constitutive model (2.50):

$$\overset{\circ}{\boldsymbol{\sigma}} = \mathbf{L}^S : \mathbf{D} + C_b \operatorname{tr}(\mathbf{D})\,\mathbf{1} = \overset{\circ}{\mathbf{s}} - \dot{p}\,\mathbf{1}. \tag{3.51}$$

Using (2.50) the elastic pressure p' is calculated with:

$$p' = p_{\text{ref}} - C_b \operatorname{tr}(\mathbf{D})\,\Delta t = p_{\text{ref}} - C_b \operatorname{tr}(\Delta\boldsymbol{\varepsilon}). \tag{3.52}$$

The term with p' in (3.47) is reduced integrated. The hydrostatic pressure p' is constant per element.
The deviatoric stress $\mathbf{s}'$ in (3.47) is obtained by integrating the stress rate $\dot{\mathbf{s}}$. At each corrector step the integration is carried out from the initial stress of the time step $\mathbf{s}_{\text{ref}}$, because the iteration method should not result in different path dependent behaviour. In Figure 3.6 it is shown that the final yield surface $\phi' = \phi_3$ is a result of the path performed during the iterations, as the integration is performed with the stress of the previous iteration.
The calculation of $\mathbf{s}'$ is performed in two steps. First, the $\overset{\circ}{\mathbf{s}}$ described by the constitutive model (3.51) is integrated for which we apply the mean normal method. During this integration the pressure p' remains unchanged when a Von Mises yield criterion is used. For other methods for the integration of the constitutive model, the reader is referred to [45].

The obtained stress $\boldsymbol{\sigma}'^R = \mathbf{s}'^R - p'$ is corotated with the rotation during the time step. The second step consists of rotating back this corotated stress $\boldsymbol{\sigma}'^R$ to $\boldsymbol{\sigma}'$, which is used in (3.47). The spin tensor $\mathbf{W}$ is employed to obtain the rotation of the time step. The spin tensor $\mathbf{W}$ is approximated by

$$\widetilde{\mathbf{W}} = \frac{1}{2}(\mathbf{v}\overleftarrow{\nabla}_{\frac{1}{2}} - \overrightarrow{\nabla}_{\frac{1}{2}}\mathbf{v}). \tag{3.53}$$

So we can write

$$\widetilde{\mathbf{W}} = \begin{bmatrix} 0 & \omega \\ -\omega & 0 \end{bmatrix}, \tag{3.54}$$

where $\omega = \frac{1}{2}(\frac{\partial v_x}{\partial y} - \frac{\partial v_y}{\partial x})_{\frac{1}{2}}$. In case of a rigid rotation with a constant spin tensor $\widetilde{\mathbf{W}}$, the rotation $\mathbf{R}$ is described by the following equation (using (2.26)):

$$\widetilde{\mathbf{W}} = \dot{\mathbf{R}} \cdot \mathbf{R}^{\Leftrightarrow 1}. \tag{3.55}$$

Integration of (3.55) gives:

$$\int_{t=0}^{t=\Delta t} \widetilde{\mathbf{W}}\, dt = \ln\left(\frac{\mathbf{R}(\Delta t)}{\mathbf{R}(0)}\right). \tag{3.56}$$

The solution for $\mathbf{R}(\Delta t)$ of this differential equation is:

$$\mathbf{R}(\Delta t) = \exp(\widetilde{\mathbf{W}}\ \Delta t)\,\mathbf{R}(0) = \sum_{n=0}^{\infty} \frac{1}{n!}(\widetilde{\mathbf{W}}\ \Delta t)^n\,\mathbf{R}(0). \tag{3.57}$$

In the updated Lagrangian method the initial rotation of the beginning of a time step $\mathbf{R}(0)$ is equivalent to $\mathbf{1}$. Then the expansion (3.57) equals

$$\mathbf{R}(\Delta t) = \begin{bmatrix} \cos(\Delta\omega) & -\sin(\Delta\omega) \\ \sin(\Delta\omega) & \cos(\Delta\omega) \end{bmatrix}, \tag{3.58}$$

where $\Delta\omega = \frac{1}{2}\Delta t\,(\frac{\partial v_x}{\partial y} - \frac{\partial v_y}{\partial x})_{\frac{1}{2}}$. The corotated Cauchy stress $\boldsymbol{\sigma}'^R$ is then back-rotated to the current coordinate system by:

$$\boldsymbol{\sigma}' = \mathbf{R} \cdot \boldsymbol{\sigma}'^R \cdot \mathbf{R}^T. \tag{3.59}$$

This can be interpreted as a transformation according to the Mohr circles. Since the Von Mises yield criterion is used in this work, the back-rotated stress also satisfies the yield criterion. For more elaborated schemes to obtain the rotation tensor by polar decomposition or the application of incrementally objective schemes, the reader can consult [29]: these schemes carry out a rigid rotation exactly.
Knowing $\boldsymbol{\sigma}'$ we can update the complete expression (3.47) for internal virtual work $\mathbf{Q}_M$.

3.3 Arbitrary Lagrangian–Eulerian method

The ALE method used in this work is also a discretisation method in time similar to the updated Lagrangian method. However, in contrast to the updated Lagrangian method the state variables are written in terms of the referential coordinate $\boldsymbol{\chi}_{\text{ref}}$. At the end of each time step the referential situation is updated with the current situation. The coordinate $\boldsymbol{\chi}_{\text{ref}}$ is labeled with the spatial coordinates at the beginning of the time step. In Figure 2.1 it is shown that the referential coordinates in the updated Lagrangian method and in the ALE method are both labeled with the spatial coordinates at the beginning of a time step. At the end of a time step the spatial positions $\mathbf{x}'$ differ in both cases. However, the same spatial domain Ω' is described by both methods.

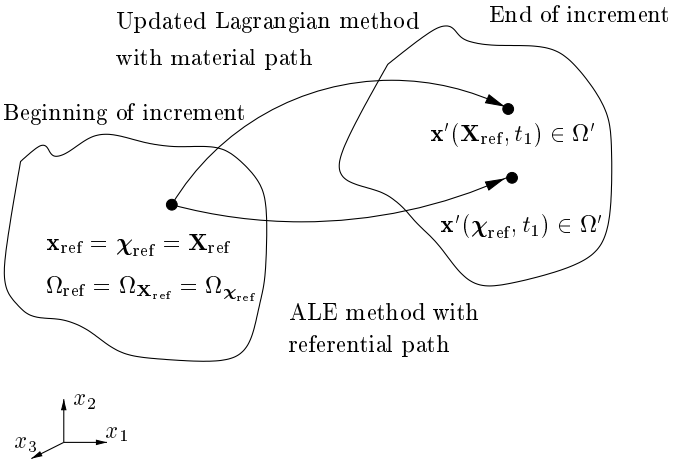


Figure 3.7: Spatial domains with the coordinates $\mathbf{X}_{\text{ref}}$ and $\boldsymbol{\chi}_{\text{ref}}$.

It is sometimes also called the *Updated Arbitrary Lagrangian Eulerian method* [2]. The initial situation at the beginning of a time step is denoted as $\Omega_{\chi_{\text{ref}}}, \bar{\varepsilon}_{\chi_{\text{ref}}}$ and $\boldsymbol{\sigma}_{\chi_{\text{ref}}}$ and the current situation at the end of the increment indicated with the superscript $'$. The weak form of equilibrium (3.10) has to be fulfilled for $\Omega', \mathbf{v}'$ and $\boldsymbol{\sigma}'$ while knowing the situation at the beginning of the time step:

$$\int_{\Omega'} (\mathbf{v}^* \overleftarrow{\nabla}') : \boldsymbol{\sigma}' \, d\Omega = \int_{,'_C} \mathbf{v}^* \cdot \boldsymbol{\tau}'_{\text{fr}} \, d, \; + \; \int_{,'_C} \mathbf{v}^* \cdot \mathbf{n}' \sigma'_n \, d, \\ + \int_{,'_T} \mathbf{v}^* \cdot \mathbf{T}' \, d, \; , \; \forall \mathbf{v}^* \in \mathcal{V}^* \tag{3.60}$$

and

$$\int_{,\,{}'_C} \lambda^* \, h'(\mathbf{v}) \; d, \;\; \leq 0 \,, \;\; \forall \lambda^* .$$

The same methodology of updating of the coordinates and the state variables as with the updated Lagrangian method (3.12), (3.13) and (3.14) is applied. However, in the ALE method this updating is performed with constant referential coordinate $\boldsymbol{\chi}_{\text{ref}}$. The updating of the coordinates becomes:

$$\mathbf{x}' \approx \mathbf{x}_{\text{ref}} + \left.\frac{\partial \mathbf{x}}{\partial t}\right|_{\boldsymbol{\chi}_{\text{ref}}} \Delta t = \mathbf{x}_{\text{ref}} + \mathbf{v}_{\text{g}} \, \Delta t = \mathbf{x}_{\text{ref}} + \Delta \mathbf{u}_{\text{g}}, \tag{3.61}$$

where the mesh velocity $\mathbf{v}_{\text{g}}$ has to be determined in some way, which will be discussed in Chapter 4. The calculation of the mesh velocity has to result in a spatial domain Ω' that is approximately the same as in the updated Lagrangian method (see Figure 2.1).
The state variables $\bar{\varepsilon}$ and $\boldsymbol{\sigma}$ are described by the material time derivative. The relation between the material time derivative and the grid time derivative with a constant referential coordinate $\boldsymbol{\chi}_{\text{ref}}$ of a state variable ζ is given by (2.8):

$$\dot{\zeta} = \left.\frac{\mathrm{d}\zeta}{\mathrm{d}t}\right|_{\mathbf{x}_{\text{ref}}} = \left.\frac{\partial \zeta}{\partial t}\right|_{\boldsymbol{\chi}_{\text{ref}}} + \mathbf{v}_c \cdot \frac{\partial \zeta}{\partial \mathbf{x}}, \tag{3.62}$$

where $\mathbf{v}_{\text{c}} = \mathbf{v} - \mathbf{v}_{\text{g}}$ is called the convective velocity. Using (3.62), a state variable ζ is updated according to [7]:

$$\begin{aligned} \zeta' &\approx \zeta_{\text{ref}} + \left.\frac{\partial \zeta}{\partial t}\right|_{\boldsymbol{\chi}_{\text{ref}}} \Delta t \\ &\approx \zeta_{\text{ref}} + \left(\dot{\zeta} - \mathbf{v}_c \cdot \frac{\partial \zeta}{\partial \mathbf{x}}\right) \Delta t \\ &\approx \zeta_{\text{ref}} + \frac{\partial_{\text{g}} \zeta}{\partial t} \Delta t = \zeta_{\text{ref}} + \Delta \zeta_{\text{g}}. \end{aligned} \tag{3.63}$$

The domain Ω' is obtained with the mesh velocity $\mathbf{v}_{\text{g}}$. So (3.60) is coupled with the calculation of the mesh velocity. The mesh velocity is an independent variable in the problem (3.60). As a result the number of degrees of freedom to be solved for the problem (3.60) is doubled.
In comparison with the updated Lagrangian method it is more difficult to update state variables, as the mesh is no longer connected to the same material points throughout the calculation. In the case of the ALE method a convective term appears when a material associated parameter has to be updated. The effect of this problem is demonstrated below for the case of a viscoplastic model and for the case of an elastoplastic model.

Viscoplastic model

In the case of a viscoplastic model $\boldsymbol{\sigma}$ follows directly from the instantaneous material velocity field $\mathbf{v}$ (see (2.33)). So we do not need to integrate a stress rate as in the elastoplastic case. The equivalent plastic strain is a state variable that must be determined by (3.63):

$$\frac{\partial_{\mathrm{g}}\bar{\varepsilon}}{\partial t} = \dot{\bar{\varepsilon}} - \mathbf{v}_{\mathrm{c}} \cdot \nabla(\bar{\varepsilon}), \tag{3.64}$$

where $\dot{\bar{\varepsilon}}$ is the material time derivative of $\bar{\varepsilon}$, which is used in the updated Lagrangian method. The weak form of equilibrium in the ALE method (3.60) is coupled with the calculation of the mesh velocity by the calculation of $\mathbf{x}'$ and $\bar{\varepsilon}'$. This non-linear system of equations can be solved by the predictor–corrector method of page 31, but that is not considered here.
In that case the material velocity $\mathbf{v}$ and the mesh velocity $\mathbf{v}_{\mathrm{g}}$ are solved in the predictor step. The correction of non-linearities is performed in the corrector step and the weak form of equilibrium is checked on the domain Ω'. In [25] a viscoplastic material model is applied in dynamic problems without path dependence.

Elastoplastic model

In the case of an elastoplastic model the stress has also to be obtained by integration of the time derivative with constant referential coordinates (3.63):

$$\frac{\partial_{\mathrm{g}}\boldsymbol{\sigma}}{\partial t} = \dot{\boldsymbol{\sigma}} - \mathbf{v}_{\mathrm{c}} \cdot \boldsymbol{\nabla}(\boldsymbol{\sigma}). \tag{3.65}$$

As a result a convective term for the Cauchy stress must be taken into account when (3.60) is solved for an elastoplastic model. So the mesh velocity influences (3.60) by the calculation of $\mathbf{x}'$, $\bar{\varepsilon}'$ and $\boldsymbol{\sigma}'$. This problem can be solved again with a predictor–corrector method. The tangential stiffness matrix has to be determined in a similar way as in the updated Lagrangian method for an elastoplastic model (see Section 3.2.2). In [38] the tangential stiffness matrix is derived by linearisation of the weak form of equilibrium (3.60).
The correction for the non-linearities are performed in the corrector step and subsequently (3.60) is checked on the domain Ω' obtained with the mesh velocity. In [4],[38], [37] and [17] path dependent material behaviour is assumed and a coupled ALE method is used to solve the equations.

3.3.1 Uncoupled ALE method

In this thesis we prefer to solve the set of equations (3.60) in an uncoupled way. In the uncoupled ALE method first an updated Lagrangian step is performed. In the updated Lagrangian step the weak form of equilibrium is solved as in the updated Lagrangian method (see Section 3.2). As a result we fulfill the equilibrium equations on the domain Ω_ℓ which is obtained with the material displacements $\Delta\mathbf{u}$. Secondly, the mesh velocity $\mathbf{v}_\mathrm{g}$ is calculated. Finally, a remap of state variables is carried out in order to obtain the values of the state variables on the domain Ω' which is obtained with the grid displacement $\Delta\mathbf{u}_\mathrm{g}$ (3.61). This remap of state variables consists of integrating the grid time derivative:

$$\frac{\partial_\mathrm{g}\zeta}{\partial t} = \dot{\zeta} - \mathbf{v}_c \cdot \frac{\partial\zeta}{\partial\mathbf{x}}. \tag{3.66}$$

where $\dot{\zeta}$ is determined in the updated Lagrangian step and $\mathbf{v}_c = \mathbf{v} - \mathbf{v}_\mathrm{g}$ is determined with the updated Lagrangian step and the calculation of the mesh velocity $\mathbf{v}_\mathrm{g}$ (see Figure 3.2). So the remap of state variables consists of calculating the convective term in (3.66).
When the uncoupled approach is applied, the equilibrium equations are disturbed by the remap step. In [2] it is shown that an uncoupled approach can be applied when the remap step is performed accurately. The uncoupled approach has some advantages with respect to the coupled approach. It simplifies the problem, since the stiffness matrix does not contain convective terms and we can use the standard updated Lagrangian method. The number of degrees to be solved in the system of equilibrium equations is halved, as only the material velocity $\mathbf{v}$ has to be calculated. It is also not necessary to describe the mesh velocity in a set of equations. The uncoupled ALE method allows a greater freedom to determine a new element mesh.
In Chapter 4 several methods of the mesh management are shown and discussed. Subsequently in Chapter 5 the remap step is explained in more detail.

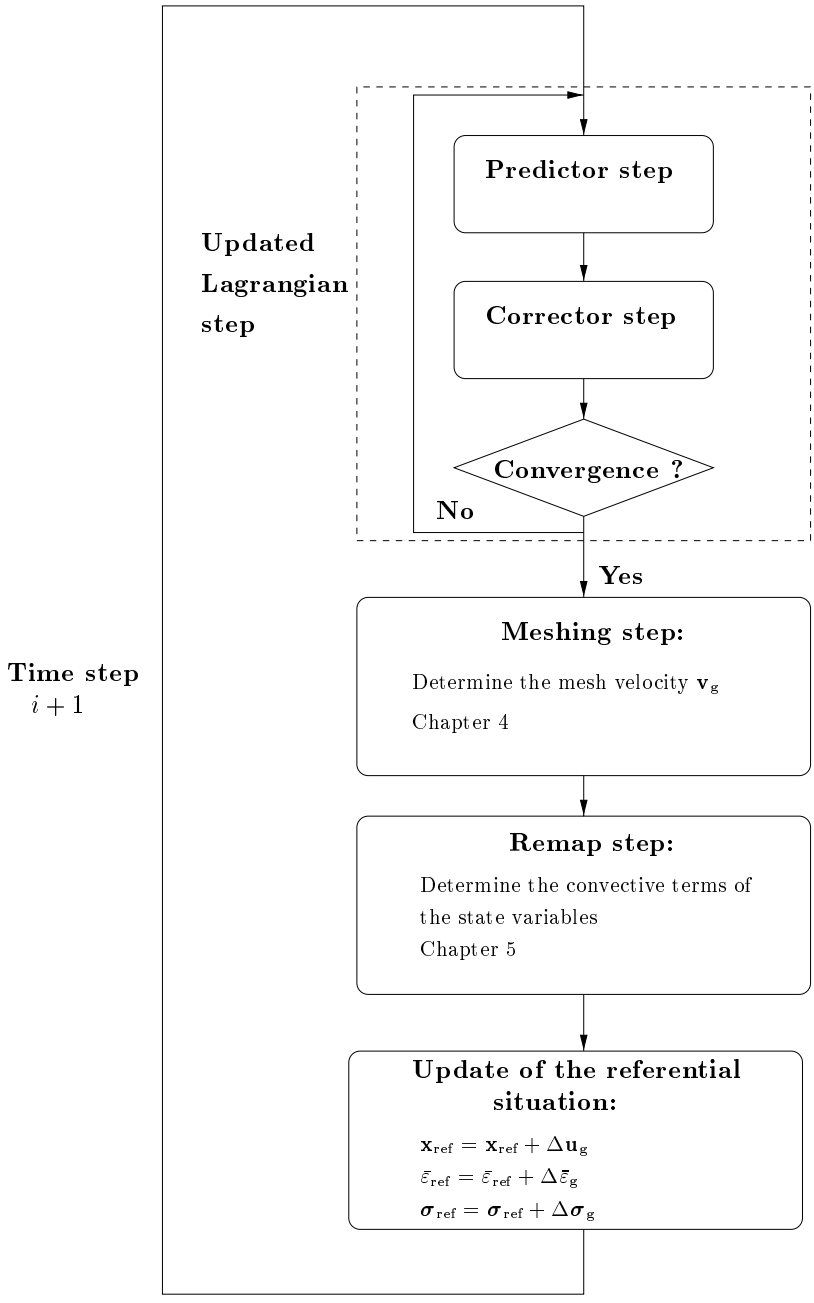


Figure 3.8: Flow chart of the uncoupled ALE method.

Chapter 4

Mesh management

In updated Lagrangian calculations the element mesh often becomes highly distorted. In order to overcome this problem we have two possibilities: perform a complete remeshing or apply the ALE method. In the ALE method a remeshing is performed at each time step and the topology of the mesh is kept intact. In the previous chapter the uncoupled ALE method used in this work was globally described. In this method first an updated Lagrangian step is performed, subsequently the mesh velocity is determined and finally a remap of state variables is carried out. The updated Lagrangian part of the ALE method was already described in detail in the previous chapter. The mesh management is the subject of the current chapter and the remap of state variables is discussed in Chapter 5.
Since in the uncoupled ALE method an updated Lagrangian step is carried out before the calculation of the mesh velocity $\mathbf{v}_g$, the material velocity $\mathbf{v}$ is assumed known in this chapter. The new mesh calculated with the mesh velocity can be defined according to various criteria. The mesh can be defined with the help of error estimators or error indicators. In such cases we obtain a refined mesh in areas *e.g.* with steep gradients of the solution. In this work the remeshing is mainly applied in order to prevent mesh distortion. The following requirements are considered for the remeshing method:

- It must accurately model the domain: no material should be generated or lost within the accuracy of the method.
- It must give optimum element shapes for 'optimum' numerical performance of the finite element discretisation.
- It must preserve the refinements of the initial mesh.
- It must be computationally efficient, as the remeshing has to be performed at each time step.

- It must be stable and robust.

The following relation exists between the domains $\mathbf{x}_{\text{ref}}$ and $\mathbf{x}'$:

$$\mathbf{x}' = \mathbf{x}_{\text{ref}} + \Delta\mathbf{u}_{\text{g}}, \tag{4.1}$$

where the mesh displacement $\Delta\mathbf{u}_{\text{g}} = \mathbf{v}_{\text{g}}\,\Delta t$ and the mesh velocity $\mathbf{v}_{\text{g}}$ is taken constant during a time step. For the updating of the mesh and the remap of state variables both the new domain $\mathbf{x}'$ and the mesh displacement $\Delta\mathbf{u}_{\text{g}}$ are required. Two types of methods are discussed: the transfinite mapping method of [46] (Section 4.1) and the Laplace method (Section 4.2). In the first method the new domain $\mathbf{x}'$ is determined and subsequently the mesh displacement $\Delta\mathbf{u}_{\text{g}}$ is calculated by (4.1). With the Laplace method the mesh velocity $\mathbf{v}_{\text{g}}$ is described by the Laplace equation and subsequently the new domain $\mathbf{x}'$ is determined by (4.1). Two discretisations of the Laplace equation are applied. First the centering method of [39] is described. This method becomes less effective in the case of distorted element meshes. For this reason a new method is introduced in Section 4.2.2, where the Laplace equation is discretised with the finite element method. This method is an improvement of the centering method in the case of distorted element meshes. Finally in Section 4.3 the remeshing methods are compared.

4.1 Transfinite mapping

In [46] the transfinite mapping method was used for the mesh management in the ALE method. This remeshing method requires an initial mesh with the same structure as described by [19], which is discussed in Section 4.1.1. In this work the transfinite mapping method is only applied to construct meshes with quadrilateral elements. It can also be applied to construct triangular meshes, but that is not considered here. The initial mesh is obtained by first subdividing the material domain into a number of regions. Subsequently, the boundaries of these regions are discretised. The discretisation of the interior of the regions follows from the discretisation of the boundaries. In Section 4.1.2 the remeshing technique with the transfinite mapping method in the ALE method is given.

4.1.1 Construction of the initial mesh

The construction of the initial mesh consists of two steps. First, the domain is subdivided into a number of quadrilateral regions, where each region is bounded by four masterlines (see Figure 4.1). This is called the hierarchical

partitioning of the domain. The total number of regions that is needed mainly depends on the geometrical complexity of the domain. The masterlines are described *e.g.* as a polynomial function or an elliptical arc. The ends of the masterlines are called the poles.

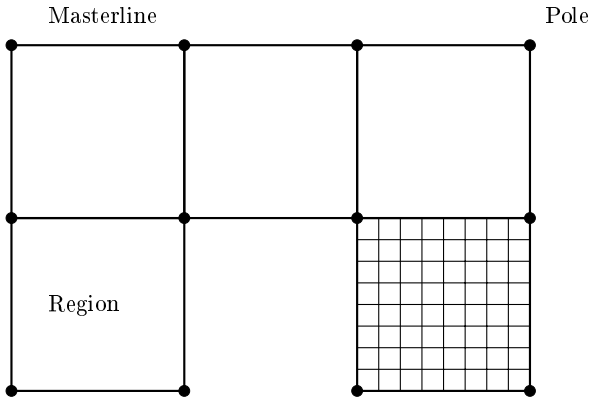


Figure 4.1: Partitioning of the domain into regions, masterlines and poles.

After the partitioning of the domain, each subregion is discretised. Hence each masterline of the region is discretised: $\mathbf{\Psi}_1(\xi_1^i)$ and $\mathbf{\Psi}_2(\xi_1^i)$ with $1 \leq i \leq n$, $\xi_1^1 = 0$ and $\xi_1^n = 1$, $\mathbf{\Psi}_3(\xi_2^j)$ and $\mathbf{\Psi}_4(\xi_2^j)$ with $1 \leq j \leq m$, $\xi_2^1 = 0$ and $\xi_2^m = 1$ (see Figure 4.2). The functions $\mathbf{\Psi}_i$ represent the coordinates of the nodal points of the boundary of a region. Note that two opposite boundaries are necessarily discretised with the same number of nodal points. After the discretisation of the boundaries the nodal points in the interior of the region are calculated using the bilinear operator [19]:

$$\begin{aligned}\mathbf{x}(\xi_1^i, \xi_2^j) \; = & (1 - \xi_2^j)\, \mathbf{\Psi}_1(\xi_1^i) \; + \; \xi_2^j\, \mathbf{\Psi}_2(\xi_1^i) \; + \; (1 - \xi_1^i)\, \mathbf{\Psi}_3(\xi_2^j) \\ & + \; \xi_1^i\, \mathbf{\Psi}_4(\xi_2^j) \; - \; \xi_1^i\, \xi_2^j\, \mathbf{x}(1,1) \; - \; (1 - \xi_1^i)\, (1 - \xi_2^j)\, \mathbf{x}(0,0) \\ & - \; (1 - \xi_1^i)\, \xi_2^j\, \mathbf{x}(0,1) \; - \; \xi_1^i\, (1 - \xi_2^j)\, \mathbf{x}(1,0).\end{aligned} \tag{4.2}$$

As a result the nodal points in the interior of the region are determined by the discretisation of the boundaries and (4.2).

4.1.2 Remeshing method

The use of the transfinite mapping method for mesh management in the ALE method [46] is described in this section. Here we focus on the discretisation of the boundaries of a region according to a criterion. The nodal points in the

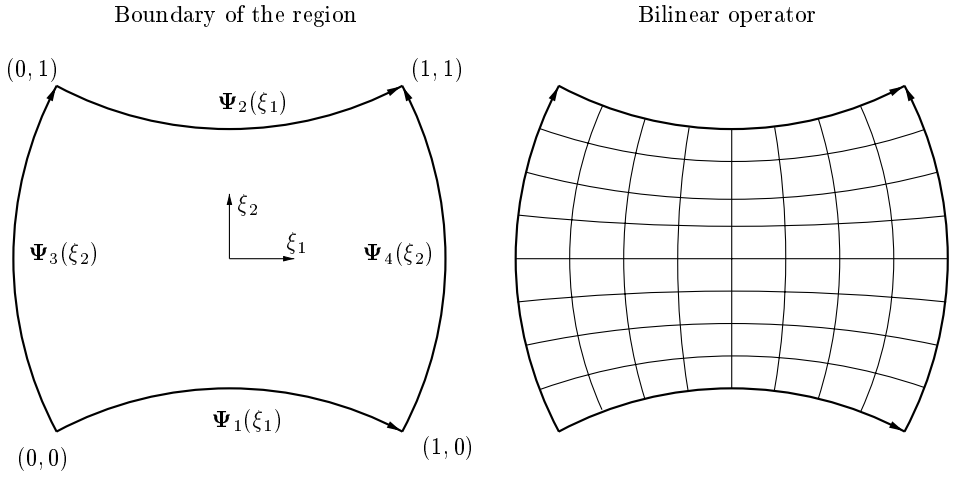


Figure 4.2: Transfinite mapping.

interior of a region again follow from this discretisation of the boundary by the bilinear operator (4.2). We can distinguish three steps:

1. Updating of the boundaries with the material displacement;
2. Redistribution of the nodal points on the boundaries;
3. Calculation of the new nodal point coordinates.

The three steps are described below.

Updating of the boundaries

First, the nodal points on the curves are updated with the material displacement $\Delta\mathbf{u}$ of the updated Lagrangian step:

$$\mathbf{x}_\ell = \mathbf{x}_{\text{ref}} + \Delta\mathbf{u}. \tag{4.3}$$

The coordinates of a nodal point on the 'updated Lagrangian' boundary is indicated as $\mathbf{x}_\ell$.

Redistribution of the nodal points

The redistribution of the nodal points on these updated curves is the crucial step of this remeshing method. It determines the quality of the new element mesh. First the two poles of the boundary are redistributed. Several possibilities exist for the position of the poles: the poles can be prescribed with the

material displacement or the poles can be kept at the corner of a tool. Then the internal nodal points of the boundary are redistributed over the boundary with the help of a positive weight function w. We want the nodal points to be closer to each other where w is large and conversely, further away from each other where w is small. For this reason the weight function w is evenly distributed across the segments of the boundary:

$$w^k \Delta s^k = \text{constant}, \tag{4.4}$$

where w^k is the value of the weight function and $\Delta s^k = s^{k+1} - s^k$ is the length of the k^{th} segment. With (4.4) the new position of the nodal points on the boundary is determined. We use the function $E(s^i)$ to obtain an even distribution of w across the segments:

$$E(s^i) = \sum_{k=1}^{i \Leftrightarrow 1} w^k \Delta s^k, \tag{4.5}$$

where E can be interpreted as a length weighted with w. Eq. (4.4) becomes, in terms of the function E, as follows:

$$\Delta E_k^{k+1} = E(s^{k+1}) - E(s^k) = \text{constant}. \tag{4.6}$$

First we calculate $E(s_\ell^n)$ at the end of the updated Lagrangian boundary with (4.5), where the boundary is assumed to consist of n nodal points. For the redistribution of the nodal points we demand (4.6), which gives the new value of E for nodal point i:

$$E(s^i) = \frac{(i-1)}{(n-1)} E(s_\ell^n), \quad 1 = 1, \ldots n. \tag{4.7}$$

Knowing $E(s^i)$ we can calculate the new coordinate s^i by (4.5). The new coordinate s^i is indicated without a subscript. In Figure 4.3 the calculation of the new coordinate s^i is demonstrated. One can see that the segments are small where the weight function is large.
For example, the weight function w can be taken equal to the absolute value of the gradient of stresses or strains. Then the weight function is used as an error indicator function. For a mesh with a global initial refinement $\Delta s^{n \Leftrightarrow 1} = \alpha \, \Delta s^1$ and we keep this initial refinement intact by applying the weight function:

$$w^k = \frac{1}{\alpha^{\frac{(k-1)}{(n-1)}}}. \tag{4.8}$$

Eq. (4.8) is derived using $\Delta s^{n \Leftrightarrow 1} = \alpha \, \Delta s^1$ and (4.4).

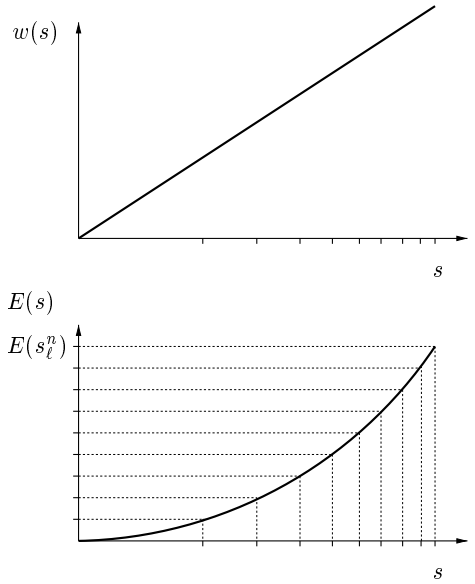


Figure 4.3: Example of a weight function w and the weighted length E.

Calculation of the new nodal point coordinates

Knowing the new positions s^i along the boundary, the new coordinates $\mathbf{x}'^i$ of these nodal points are now calculated. The new coordinates of the nodal points are obtained by an interpolation of updated Lagrangian coordinates $\mathbf{x}_\ell^i$ (see Figure 4.4).

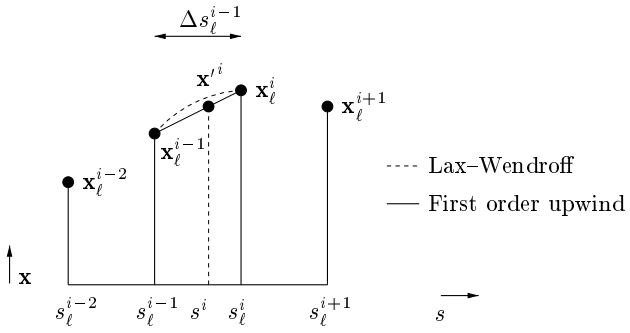


Figure 4.4: Discrete representation of the curve.

We suppose that the new position $\mathbf{x}'^i$ of a nodal point i is close to the updated Lagrangian position $\mathbf{x}_\ell^i$ and that it lies between $\mathbf{x}_\ell^i$ and $\mathbf{x}_\ell^{i\Leftrightarrow 1}$. We can use a first order Taylor expansion for nodal point i, which is the basis for the interpolation

scheme:

$$\mathbf{x}'^i = \mathbf{x}_\ell^i + \big(ds\frac{\partial \mathbf{x}_\ell}{\partial s}\big)_i + O((ds)^2)\,. \tag{4.9}$$

Assuming locally an approximate equidistantial distribution of nodal points on the curve and the fact that the gradient can be approximated in an upwind manner, we obtain:

$$\mathbf{x}'^i = \mathbf{x}_\ell^i - C(\mathbf{x}_\ell^i - \mathbf{x}_\ell^{i\Leftrightarrow 1})\,, \tag{4.10}$$

where the local Courant number C is defined by

$$C = \frac{s_\ell^i - s^i}{\Delta s_\ell^{i\Leftrightarrow 1}}\,. \tag{4.11}$$

This can be interpreted as the linear interpolation between the $\mathbf{x}_\ell^i$ and $\mathbf{x}_\ell^{i\Leftrightarrow 1}$. The interpolation problem has the same characteristics as a convection problem. A first order interpolation scheme can result in an inaccurate description of the boundary. Therefore a higher order accurate interpolation method is used, namely the limited Lax–Wendroff scheme:

$$\begin{aligned}\mathbf{x}'^i =& \mathbf{x}_\ell^i - C(\mathbf{x}_\ell^i - \mathbf{x}_\ell^{i\Leftrightarrow 1}) \\ &- \frac{1}{2}C(1-C)\big(\phi(r^{i+\frac{1}{2}})(\mathbf{x}_\ell^{i+1} - \mathbf{x}_\ell^i) - \phi(r^{i\Leftrightarrow\frac{1}{2}})(\mathbf{x}_\ell^i - \mathbf{x}_\ell^{i\Leftrightarrow 1})\big)\,,\end{aligned} \tag{4.12}$$

where the Van Leer limiter ϕ is:

$$\phi(r) = \frac{r + |r|}{1 + |r|} \quad \text{and} \quad r^{i+\frac{1}{2}} = \frac{\mathbf{x}_\ell^i - \mathbf{x}_\ell^{i\Leftrightarrow 1}}{\mathbf{x}_\ell^{i+1} - \mathbf{x}_\ell^i}\,. \tag{4.13}$$

The limiter ϕ is employed to suppress spurious oscillations, which is necessary to calculate the new nodal coordinates in a stable way. When the limiters are equal to one, the Lax–Wendroff scheme is third order accurate.
Now that all the boundaries of the regions are discretised, the coordinates $\mathbf{x}'$ of the nodal points in the interior of the region are determined by (4.2). For the remap of state variables we need the mesh displacement $\Delta\mathbf{u}_\mathrm{g}$, which can be calculated using (4.1).

4.2 Laplace methods

In this section the mesh velocity $\mathbf{v}_\mathrm{g}$ is calculated instead of the new coordinates $\mathbf{x}'$. The mesh velocity $\mathbf{v}_\mathrm{g}$ is calculated such that mesh distortion is reduced. Here the Laplace operator is used to describe the mesh velocity:

$$\nabla^2 \mathbf{v}_\mathrm{g} = \mathbf{0}\,. \tag{4.14}$$

Eq. (4.14) tends to smooth steep gradients of the mesh velocity over the domain. Then the Laplace equation results in an updating of the new domain with a smoothed mesh velocity field. In this way we try to reduce mesh distortion. The mesh velocity $\mathbf{v}_g$ has to result in the same evolution of the boundary as with the material velocity $\mathbf{v}$. So the following boundary condition is applied:

$$\mathbf{v}_g \cdot \mathbf{n} = \mathbf{v} \cdot \mathbf{n} \,. \tag{4.15}$$

The mesh velocity in the tangential direction at the boundary is described by the natural boundary condition of (4.14), which is used to improve the quality of the element mesh. For several nodal points on the boundary the mesh velocity is completely prescribed, which will be explained in the following sections.
The centering method of [39] is a discretisation of the Laplace equation, which is explained in Section 4.2.1. However, the centering method assumes a homogeneous distribution of nodal points. So this method is less effective in reducing mesh distortion in cases with a non-homogeneous distribution of nodal points. For this reason an alternative method in Section 4.2.2 is introduced. With this method the Laplace equation is discretised with the finite element method.

4.2.1 Centering method

The centering method for the six-node triangular element introduced in [39] is applied here. The mesh velocity is smoothed over the domain by taking the nodal mesh velocity $\mathbf{V}_g^i$ equal to the average mesh velocity $\overline{\mathbf{V}}_g^i$ of the neighbouring nodal points. This is equivalent to minimising the potential function

$$I = \sum_{i \in , \, CN} (\mathbf{V}_g^i - \overline{\mathbf{V}}_g^i)^2, \tag{4.16}$$

which is calculated using the corner nodal points , $_{CN}$. The centering method represents a discretisation of the Laplace equation in the case of a homogeneous distribution of nodal points. The total set of nodal points consists of corner nodal points, midside nodal points (see Figure 4.5) and nodal points on the boundary of the domain. The calculation of the mesh velocity is specified for each subset of nodal points separately. For nodal points on the boundary the integral form of the boundary condition (4.15) must also be fulfilled.

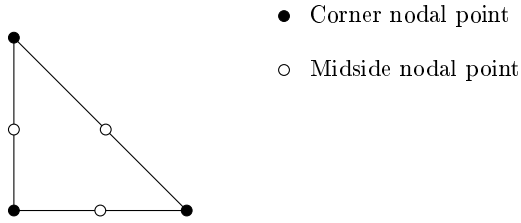


Figure 4.5: Triangular element with corner and midside nodal points.

Corner nodal points

For nodal points in the interior of the mesh the average velocity $\overline{\mathbf{V}}_{\mathrm{g}}^{i}$ is specified by

$$\overline{\mathbf{V}}_{\mathrm{g}}^{i} = \frac{1}{N(\Theta)} \sum_{j \in \Theta} \mathbf{V}_{\mathrm{g}}^{j}, \tag{4.17}$$

where Θ contains the neighbouring corner nodal points of nodal point i and $N(\Theta)$ is the number of nodal points in Θ. If the nodal points are situated according to a rectangular grid, we have the finite difference discretisation of the Laplace equation (see Figure 4.6).

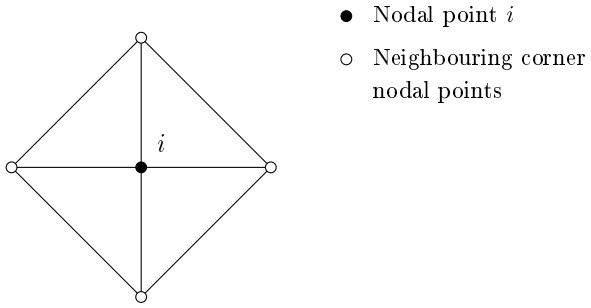


Figure 4.6: Finite difference scheme of the Laplace equation.

Midside nodal points

For a midside nodal point i the average mesh velocity becomes:

$$\overline{\mathbf{V}}_{\mathrm{g}}^{i} = \frac{1}{2}(\mathbf{V}_{\mathrm{g}}^{i \Leftrightarrow 1} + \mathbf{V}_{\mathrm{g}}^{i+1}), \tag{4.18}$$

where $(i-1)$ and $(i+1)$ denote the two corresponding corner nodal points on the same element side. If the midside nodal points are initially in the middle of an element side, they also remain in the middle according to this definition.

Nodal points on boundary

At the boundary the boundary condition (4.15) must also be fulfilled. For the discretisation of the boundary condition at the free surface we want the total material flux to be equal to:

$$\Phi = \int_{,} \mathbf{v} \cdot \mathbf{n} \; d, \; = \int_{,} \mathbf{v}_{\mathrm{g}} \cdot \mathbf{n} \; d, \; . \tag{4.19}$$

Now the derivation of consistent normal $\mathbf{n}_c$ [14] is given, which makes it possible to impose (4.19) at a nodal point. The velocity component in the direction $\mathbf{t}_c$ at a nodal point can be chosen freely, as long as (4.19) is fulfilled.
Substituting the finite element approximation for the material velocity, the material flux can be written as:

$$\Phi = \int_{,} \mathbf{v} \cdot \mathbf{n} \; d, \; = \sum_{i} \mathbf{V}^i \int_{,} N^i \cdot \mathbf{n} \; d, \; , \tag{4.20}$$

where i denotes the nodal points on the boundary. $\mathbf{V}^i$ is decomposed in the normal and tangential consistent unity vectors:

$$\mathbf{V}^i = V_n^i \; \mathbf{n}_c^i + V_t^i \; \mathbf{t}_c^i. \tag{4.21}$$

Using

$$t_{c,x} = n_{c,y} \quad \text{and} \quad t_{c,y} = -n_{c,x}, \tag{4.22}$$

and Figure 4.7, the components of $\mathbf{V}^i$ become

$$\begin{cases} V_x^i = V_n^i \; n_{c,x}^i + V_t^i \; n_{c,y}^i, \\ V_y^i = V_n^i \; n_{c,y}^i - V_t^i \; n_{c,x}^i. \end{cases} \tag{4.23}$$

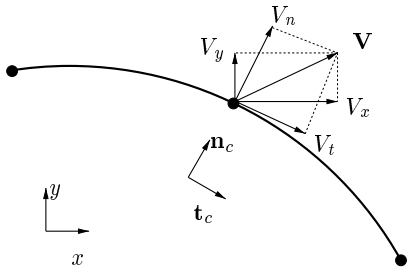


Figure 4.7: Velocity at the boundary.

Substituting this for Φ, we obtain

$$\Phi = \sum_i \Big(V_n^i \ (\ n_{c,x}^i \int_{,} N^i \ n_x \ d, \ + n_{c,y}^i \int_{,} N^i \ n_y \ d, \ ) \\ + V_t^i \ (\ n_{c,y}^i \int_{,} N^i \ n_x \ d, \ - n_{c,x}^i \int_{,} N^i \ n_y \ d, \ ) \Big). \tag{4.24}$$

The material flux is required to be independent of the nodal tangential velocity V_t^i. Then the normalised consistent normal vector becomes

$$n_{c,x}^i = \frac{1}{|\mathbf{n}_c^i|} \int_{,} N^i \ n_x \ d, \quad ; \quad n_{c,y}^i = \frac{1}{|\mathbf{n}_c^i|} \int_{,} N^i \ n_y \ d, \ , \tag{4.25}$$

where

$$|\mathbf{n}_c^i| = \sqrt{(\int_{,} N^i \ n_x \ d, \ )^2 + (\int_{,} N^i \ n_y \ d, \ )^2}. \tag{4.26}$$

Eq. (4.25) can be interpreted as a weighting of the normal vectors of the two adjacent elements (see Figure 4.8).

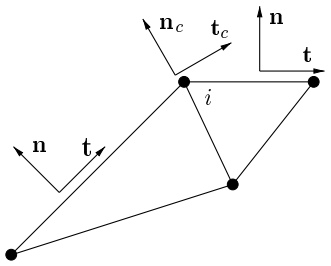


Figure 4.8: Consistent vectors $\mathbf{n}_c^i$ and $\mathbf{t}_c^i$.

As a result, the total flux Φ depends only on the normal components of $\mathbf{V}^i$. Similarly the mesh velocity is decomposed into the normal and tangential directions:

$$\mathbf{V}_{\mathrm{g}}^i = V_{\mathrm{g},n}^i \ \mathbf{n}_c^i + V_{\mathrm{g},t}^i \ \mathbf{t}_c^i. \tag{4.27}$$

We prescribe the normal component of the mesh velocity:

$$V_{\mathrm{g},n}^i = V_n^i \tag{4.28}$$

and (4.19) is automatically fulfilled.
The mesh velocity in the tangential direction $\mathbf{t}_c$ is used to improve the element

mesh. For a corner nodal point at the boundary, (4.17) is projected in the tangential direction:

$$\overline{\mathbf{V}}_{\mathrm{g}}^{i} \cdot \mathbf{t}_{c}^{i} = \frac{1}{N(\Theta)} \sum_{j \in \Theta} \mathbf{V}_{\mathrm{g}}^{j} \cdot \mathbf{t}_{c}^{j}. \tag{4.29}$$

Using (4.18), we obtain for a midside nodal point at the boundary:

$$\overline{\mathbf{V}}_{\mathrm{g}}^{i} \cdot \mathbf{t}_{c}^{i} = \frac{1}{2}(\mathbf{V}_{\mathrm{g}}^{i \Leftrightarrow 1} + \mathbf{V}_{\mathrm{g}}^{i+1}) \cdot \mathbf{t}_{c}^{i}. \tag{4.30}$$

For nodal points at the end of the contact with a tool or nodal points at a large curvature of the boundary we impose:

$$\mathbf{V} = \mathbf{V}_{\mathrm{g}}. \tag{4.31}$$

Solving the system of equations

The solution of this coupled system of equations is the mesh velocity $\mathbf{v}_{\mathrm{g}}$. It is not necessary to obtain the exact solution for $\mathbf{v}_{\mathrm{g}}$, as it gives the element mesh on which we want to obtain the solution for the mechanical problem. For this reason an iterative solver is preferred for the solution of this coupled system of equations for $\mathbf{v}_{\mathrm{g}}$. The average velocity $\overline{\mathbf{V}}_{\mathrm{g}}^{i}$ is calculated from the neighbouring nodal point values of the previous iteration. This can be interpreted as the classical Jacobi iteration method. The matrix that determines the corner nodal points is symmetric and diagonally dominant. Since we prescribe the mesh velocity at certain nodal points (see (4.31)), these rows of the matrix are strictly diagonally dominant. This is a sufficient condition for convergence of the Jacobi iteration method. The midside nodal points are determined from the corresponding corner nodal points. The iteration process converges when the following condition is satisfied:

$$\frac{\sqrt{\max(\Delta_k \mathbf{V}_{\mathrm{g}} \cdot \Delta_k \mathbf{V}_{\mathrm{g}})}}{\sqrt{\max(\mathbf{V} \cdot \mathbf{V})}} < C_m, \tag{4.32}$$

where $\Delta_k \mathbf{V}_{\mathrm{g}}$ is the difference in $\mathbf{V}_{\mathrm{g}}$ between the iterations $k-1$ and k and C_m a small parameter. The value of C_m is discussed in Chapter 6. In general $C_m = 1 \cdot 10^{\Leftrightarrow 3}$ is sufficient to obtain a satisfactory mesh. For a given time step the solution for the mesh velocity of the previous time step is used as the initial value, which turns out to be a good initial guess in general. This makes the iteration method more efficient than a direct method.

4.2.2 A finite element based method

The centering method is a discretisation of the Laplace equation on a homogeneous mesh. However, if the mesh is distorted, the centering method is not an accurate discretisation of the Laplace equation. As a result the centering method is less effective in reducing mesh distortion. Hence an alternative method, which discretises the Laplace equation on the basis of the finite element method, is introduced here. The total set of nodal points is again as with the centering method split up into a number of subsets of nodal points. The calculation of the mesh velocity is then given for each subset.

Corner nodal points

For the corner nodal points the following potential function $\Phi(\mathbf{v}_g, \lambda)$ is minimised:

$$\Phi(\mathbf{v}_g, \lambda) = \int_\Omega \frac{1}{2}(\nabla \mathbf{v}_g) : (\nabla \mathbf{v}_g)\ d\Omega + \int_{,} \lambda\ (\mathbf{v} - \mathbf{v}_g) \cdot \mathbf{n}\ d, \ . \tag{4.33}$$

The boundary condition is imposed with the help of the Lagrange multiplier λ, that is zero in the interior of the domain. Weighting the derivatives of Φ to $\mathbf{v}_g$ and λ results in the following equations:

$$\int_\Omega (\nabla \mathbf{v}_g) : (\nabla \mathbf{v}^*) d\Omega - \int_{,} \lambda\ \mathbf{n} \cdot \mathbf{v}^* d, \ = 0, \qquad \forall \mathbf{v}^*, \tag{4.34}$$

$$\int_{,} (\mathbf{v} - \mathbf{v}_g) \cdot \mathbf{n}\ \lambda^*\ d, \ = 0, \qquad \forall \lambda^*. \tag{4.35}$$

The first term of (4.34) represents the Laplace equation for the components of $\mathbf{v}_g$. The equations for the velocity components are coupled by the second term of (4.34) and by (4.35). The discretisation of the Laplace equation is performed with the standard Galerkin finite element method using linear triangular elements [64]. So with this discretisation the corner nodal points of the mesh are taken into account.
In the previous section it is shown by ((4.19) – (4.26)) that the integral form of the boundary condition (4.19) can be imposed for the velocity component in the direction of the consistent normal vector $\mathbf{n}_c$ for each nodal point separately. Therefore the constraint equation (4.35) is imposed at the nodal point:

$$(\mathbf{V}^i - \mathbf{V}^i_g) \cdot \mathbf{n}^i_c = 0, \tag{4.36}$$

where the nodal material velocity $\mathbf{V}$ is known. So only the components of $\mathbf{V}^i_g$ in the direction $\mathbf{n}^i_c$ are prescribed. Writing down this system of equations in

matrix form, we obtain:

$$\begin{bmatrix} K_L & 0 & K_x^T \\ 0 & K_L & K_y^T \\ K_x & K_y & 0 \end{bmatrix} \begin{Bmatrix} V_{g,x} \\ V_{g,y} \\ \lambda \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ \mathbf{V} \cdot \mathbf{n}_c \end{Bmatrix} \tag{4.37}$$

where K_L represents the Laplace equation, K_x and K_y are due to the boundary condition.

Midside nodal points

For midside nodal points we use (4.18) of the centering method, as we prefer midside nodal points to be located in the middle of the element sides:

$$\overline{\mathbf{V}}_{\mathrm{g}}^{i} = \frac{1}{2}(\mathbf{V}_{\mathrm{g}}^{i \Leftrightarrow 1} + \mathbf{V}_{\mathrm{g}}^{i+1}). \tag{4.38}$$

For a midside nodal point on the boundary, we apply (4.30) of the centering method:

$$\overline{\mathbf{V}}_{\mathrm{g}}^{i} \cdot \mathbf{t}_c^i = \frac{1}{2}(\mathbf{V}_{\mathrm{g}}^{i \Leftrightarrow 1} + \mathbf{V}_{\mathrm{g}}^{i+1}) \cdot \mathbf{t}_c^i. \tag{4.39}$$

Solving the system of equations

The total system of equations is solved with a Jacobi iteration method (see (4.32). However, the Lagrange multiplier λ is calculated in a different way. If λ had been calculated with the Jacobi iteration method, the boundary condition would be approximately fulfilled. For this reason λ is calculated from (4.37) for each nodal point by the Gaussian elimination method. The elimination is performed on the equation for the Lagrange multiplier λ and λ is calculated from this equation. Subsequently λ is substituted into the two corresponding Laplace equations for the x- and y-directions. As a result (4.19) is fulfilled at each Jacobi iteration step. This calculation method for λ results in two equations for each nodal point, that are equivalent to prescribing the velocity in the direction $\mathbf{n}_c$ and to solving the natural boundary condition of the Laplace equation in the direction $\mathbf{t}_c$ (see also (4.29)). This can be shown by expressing the mesh velocity in components in the directions $\mathbf{n}_c$ and $\mathbf{t}_c$ and working out the obtained equations in these directions.

4.3 Discussion

In this chapter we have seen two types of methods: the transfinite method and the Laplace methods. In this section a short comparison of the basic

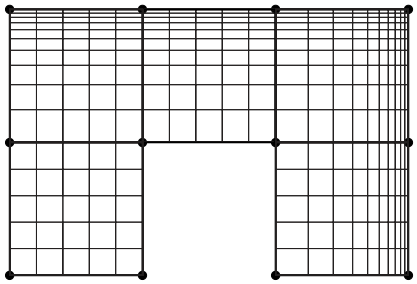


Figure 4.9: Refinement in the upper right hand corner with the transfinite mapping method.

characteristics of these methods is presented.
The transfinite mapping method is computationally efficient, since the nodal points coordinates are simply calculated by (4.2), unlike the Laplace methods. There, a coupled system of equations has to be solved to obtain the new element mesh. The system of equations is solved iteratively and a good initial guess is known. It should also be remarked that an exact solution is not necessary, as it is the element mesh that is calculated and not the solution of the simulation. A disadvantage of the transfinite mapping method is that it imposes restrictions on the mesh topology, as two opposite curves have to be discretised with the same number of elements. This makes the method less favourable for complex geometries, where more than one region is needed (see e.g. a blanking simulation [62]). If the domain consists of more than one region, the curves in the interior of the domain must also be discretised in the way described in section 4.1.2. Then it is difficult to control the mesh displacements in order to obtain satisfactory results. In the papers in which this mesh adaptation method is applied, relatively simple geometries are used (see e.g. [1, 49, 47, 4]). Also, local refinements result in an unnecessarily large number of nodal points. As an example, in Figure 4.9 a local refinement is desired in the upper right hand corner of the domain. However, this refinement leads to an increase of the number of elements in the other regions. The Laplace method does not require limitations on the element topology: it can be applied to structured and unstructured meshes.
With the transfinite mapping method, a good correlation exists between the discretisation of the masterlines and the interior of the region. As a result the mesh is structured and the shape of the elements is well controlled. With the Laplace methods the process of preventing mesh distortion is less controllable by the user. For more details and applications of the centering method, the reader is referred to [16] and [39], where the centering method is applied to the simulation of the injection moulding process.

Chapter 5

Remap of state variables

The uncoupled ALE method has been sketched in Chapter 3. In the uncoupled ALE method three steps can be distinguished: first an updated Lagrangian step, then a remeshing step and finally a remap step. In the updated Lagrangian part the material velocity $\mathbf{v}$ and the material time derivatives of the state variables are calculated. The mesh velocity $\mathbf{v}_\mathrm{g}$ is obtained in the remeshing step with a method from Chapter 4, where the mesh topology remains the same. Then the coordinates of the nodal points are updated using the mesh velocity (4.1):

$$\mathbf{x}' = \mathbf{x}_\mathrm{ref} + \mathbf{v}_\mathrm{g}\,\Delta t, \tag{5.1}$$

where the mesh velocity $\mathbf{v}_\mathrm{g}$ is assumed to be constant in time during a time step. Because of this updating of the nodal coordinates, the updating of a state variable ζ has to be performed with the grid time derivative:

$$\zeta' = \zeta_\mathrm{ref} + \int_{t_0}^{t_1} \frac{\partial_\mathrm{g}\zeta}{\partial t}\,dt, \tag{5.2}$$

where the grid time derivative is

$$\frac{\partial_\mathrm{g}\zeta}{\partial t} = \dot{\zeta} - \mathbf{v}_\mathrm{c}\cdot\nabla(\zeta), \tag{5.3}$$

with the convective velocity $\mathbf{v}_\mathrm{c} = \mathbf{v} - \mathbf{v}_\mathrm{g}$. This updating of state variables is called the remap of state variables and this is considered in this chapter. In the case of a viscoplastic model the equivalent plastic strain $\bar{\varepsilon}$ has to be updated and in the case of an elastoplastic model also the components of the Cauchy stress tensor have to be updated with the grid time derivative (see (3.65)). From now on the remap of state variables is described for an arbitrary state

variable ζ, which may represent *e.g.* any of the stress or strain components. The material time derivative $\dot{\zeta}$ is calculated in the updated Lagrangian step. In the updated Lagrangian method where $\mathbf{v}_g = \mathbf{v}$ the convective term vanishes. The convective velocity $\mathbf{v}_c$ is generally not equal to zero in the ALE method. As the updated Lagrangian step and the remeshing step are performed before the remap step, the material velocity $\mathbf{v}$ and the mesh velocity $\mathbf{v}_g$ are known. Therefore in this chapter the convective velocity $\mathbf{v}_c$ is also assumed to be known. The main topic of this chapter is the discretisation of the convective term in (5.3) within the framework of the ALE method.
First we discuss various discretisation methods for the convection term, which are evaluated according to several criteria in Section 5.1. Then two different approaches are described in Sections 5.2 and 5.3. Finally, the two approaches are briefly compared in Section 5.4.

5.1 Convection problem

The topic of this chapter consists of numerically solving the remap problem for a state variable ζ:

Find ζ':

$$\zeta' = \zeta_{\text{ref}} + \Delta\zeta_g,$$

where

$$\Delta\zeta_g = \int_{t_0}^{t_1} \frac{\partial_g \zeta}{\partial t}\, dt = \int_{t_0}^{t_1} (\dot{\zeta} - \mathbf{v}_c \cdot \nabla\zeta)\, dt, \tag{5.4}$$

$$\zeta(x, t_0) = \zeta_{\text{ref}},$$

$$\mathbf{v}_c \cdot \mathbf{n} = \mathbf{0} \qquad \text{on} \quad ,$$

and ζ_{ref}, $\dot{\zeta}$ and $\mathbf{v}_c$ are given.

In Chapter 4 it is shown that the boundary after the remeshing should approximately coincide with the boundary obtained with the updated Lagrangian calculation. This is imposed through the condition $\mathbf{v}_c \cdot \mathbf{n} = \mathbf{0}$ on the boundary. Hence we do not require a further boundary condition for the state variable ζ for the remap problem. Another characteristic of this problem is that in general the state variable ζ has to be calculated for the integration points of the elements. Using the element structure of the finite element calculation

we obtain discontinuous distributions of ζ between elements that have to be remapped. The discretisation method for (5.4) must account for this fact in order to obtain an efficient and accurate method.
The remeshing step of the ALE method is characterised by the fact that the mesh topology remains unchanged. Therefore the remap problem reduces to the discretisation of a convection problem. It is a linear convection problem, as the convective velocity $\mathbf{v}_c$ is assumed to be known here.
In this thesis two approaches are applied to solve problem (5.4) (see Figure 5.1).

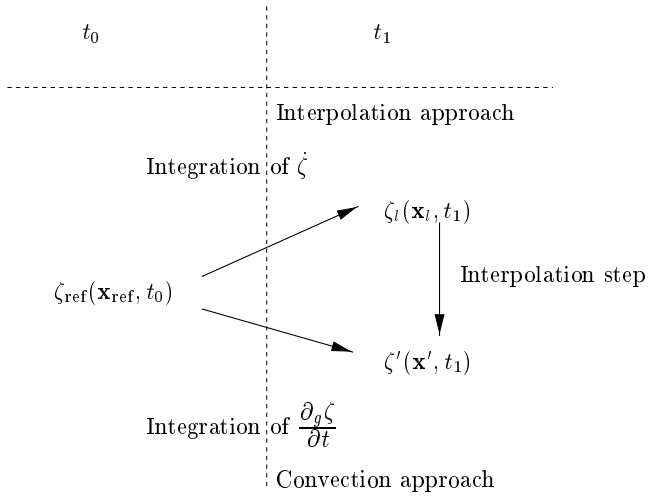


Figure 5.1: The convection approach and the interpolation approach.

In Section 5.2 the *convection approach* is discussed. In this approach the grid time derivative is integrated in time. The second approach, the *interpolation approach* consists of two phases: in the Lagrangian phase the material time derivative is integrated in time and subsequently an interpolation step is performed in the so-called Eulerian phase. This approach is explained in detail in Section 5.3.
Several methods can be applied to discretise the convective term. The choice of which discretisation method is used is based on several criteria [5], *e.g.* accuracy and stability.
First of all the method should be sufficiently *accurate* in relation to other parts of the calculation. This means that the error must become small when the mesh is refined.
Secondly, the remap algorithm should also be numerically *stable*, *i.e.* a small perturbance of the grid should not result in large changes in the results.
A third criterion is *conservativity*: the integral of a state variable over the

material domain should be approximately the same before and after the remap. Finally, the remap algorithm should be *consistent*: if the relative velocity $\mathbf{v}_c$ is equal to zero, the change due to the convective term should vanish. So the state variables are updated with the material time derivative as in the updated Lagrangian method.
During an increment, an updated Lagrangian step and one remap step for each state variable are performed. As the remap is performed many times, *efficiency* is also an important aspect.
The method should also be *monotonicity preserving*, which means that a monotone distribution should remain monotone after the remap. In other words it should not introduce spurious oscillations. The remap must be monotonicity preserving as state variables can otherwise be assigned unphysical values. For instance, a negative equivalent plastic strain could result from a remap and this has no physical meaning.

5.2 Convection approach

In the so-called convection approach the grid time derivative of the remap problem (5.4) is integrated in time:

$$\zeta' = \zeta_{\text{ref}} + \int_{t_0}^{t_1} \frac{\partial_g \zeta}{\partial t}\, dt = \zeta_{\text{ref}} + \int_{t_0}^{t_1} (\dot{\zeta} - \mathbf{v}_c \cdot \nabla(\zeta))\, dt. \tag{5.5}$$

In the uncoupled ALE method the material time derivative $\dot{\zeta}$ is calculated in the updated Lagrangian step. Subsequently, the convective term must be integrated in time. Eq. (5.5) is rewritten into

$$\zeta' = \zeta_{\text{ref}} + \Delta\zeta_m + \Delta\zeta_c, \tag{5.6}$$

where

$$\Delta\zeta_m = \int_{t_0}^{t_1} \dot{\zeta}\, dt \quad \text{and} \quad \Delta\zeta_c = -\int_{t_0}^{t_1} \mathbf{v}_c \cdot \nabla(\zeta))\, dt. \tag{5.7}$$

Therefore, we consider the discretisation of the linear convection equation:

$$\Delta\zeta_c = \int_{t_0}^{t_1} \frac{\partial \zeta}{\partial t}\, dt = -\int_{t_0}^{t_1} \mathbf{v}_c \cdot \nabla(\zeta)\, dt. \tag{5.8}$$

In the literature a large number of methods are known to solve this convection equation. The methods are roughly divided into three classes: finite difference methods, finite element methods and finite volume methods, and are discussed with respect to solving the remap problem in the ALE method.
The *finite difference method* is based on Taylor expansions. In these methods often a uniform, structured mesh is assumed, and because of this these methods are computationally efficient. For example, a first order derivative at a point i can be obtained with a forward difference:

$$\left(\frac{\partial \zeta}{\partial x}\right)_i = \frac{\zeta^i - \zeta^{i \Leftrightarrow 1}}{\Delta x} + O(\Delta x), \tag{5.9}$$

where Δx is the space between the points. Finite difference methods for the discretisation of the convection equation can be found for example in [21].
The *finite element method* can also be employed to discretise convection equations (see for example [12]). These finite element methods have to take into account the upwinding aspect needed to obtain a stable solution. Finite element methods that are suitable for solving the pure convection equation in the uncoupled ALE method are *e.g.* the Taylor–Galerkin method [11] and the Discontinuous Galerkin method [35]. The main advantage of the finite element method for the convection term in the ALE method is that the updated Lagrangian step is also performed with the finite element method, which facilitates the implementation of the method. The finite element method is well suited for unstructured meshes. However, when integration point values are convected, the finite element method is usually less convenient. One possibility is to first calculate the nodal values from the integration point values. A drawback is that discontinuities exist at the element boundaries, whereas in most of the finite element methods a continuous distribution is assumed across element boundaries. When average integration point values are used to calculate nodal point values, this tends to smooth the gradients between the elements [26]. As a second step, the convection problem is solved with these nodal values. Finally, the integration point values are calculated from the nodal values.
The Discontinuous Galerkin method allows discontinuous distributions over the element boundaries. This method can be seen as a method 'in between' the finite element methods and finite volume methods. In the Discontinuous Galerkin method boundary conditions are weighted for each element. At the element boundaries we also obtain boundary flux expressions. This method is applied to convection problems in the context of the ALE method in [3].
In the *finite volume method* the integral form is the basis for the discretisation:

$$\int_\Omega \frac{\partial \zeta}{\partial t}\, d\Omega = -\int_\Omega \mathbf{v}_c \cdot \nabla \zeta \; d\Omega. \tag{5.10}$$

The volume integral on the right hand side is transformed into a boundary integral with the theorem of Gauss. Assuming a divergence-free convective velocity field this results in the following expression:

$$\int_{\Omega} \frac{\partial \zeta}{\partial t}\, d\Omega = - \int_{,} \mathbf{n} \cdot (\mathbf{v}_c\, \zeta)\, d, \ . \tag{5.11}$$

This results in a balance of fluxes between the elements, which makes the method conservative. A method is said to be conservative if the integral of a state variable ζ over the whole domain remains the same. The boundary integral is determined independently from the element from which it is considered. That results in an ingoing flux for one element that equals the outgoing flux for another element. In the framework of the ALE method the finite volume method was applied for example in [23].
A large number of finite volume methods in the computational fluid dynamics have been introduced that are able to suppress oscillations [21], which makes it favourable for us to use a finite volume method. For this reason and the fact that a computationally efficient method can be obtained we chose to apply a finite volume method. Although other methods could also have been applied successfully. In the following section we apply the finite volume method for the convection equation in the ALE method.

5.2.1 Finite volume scheme

In this section the finite volume method is applied to quadrilateral meshes. It can also be applied to triangular meshes, but that is not considered here. The element mesh is supposed to be locally homogeneous and orthogonal. Eq. (5.8) is the convection equation for integration point values of the finite element mesh. In this section we consider control volumes that coincide with the elements. As a result, in general, discontinuous distributions will exist across element boundaries. A finite volume scheme using a Godunov-type technique [18] is then employed, which is adequate for handling discontinuities between control volumes.
We apply the finite volume method on a structured quadrilateral element mesh obtained with the transfinite mapping method of Chapter 4. The finite difference notation is therefore used to indicate an element (see Figure 5.2).
The convective increment $\Delta\zeta_c$ of (5.8) is approximated with a constant value in each element. A cell-centered finite volume scheme is employed, where the cell-centered value is the averaged value of the four integration points of the quadrilateral element.
In order to apply the finite volume method for the spatial discretisation of (5.10) the integrand of (5.10) is rewritten (see also [23]) and is integrated over

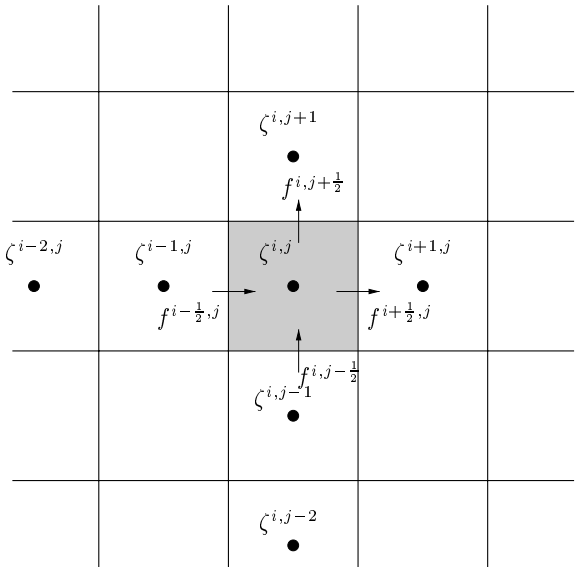


Figure 5.2: Finite volume method for quadrilateral elements with fluxes f.

the volume $\Omega^{i,j}$ of the element (i,j):

$$\int\limits_{\Omega^{i,j}} \frac{\partial \zeta^{i,j}}{\partial t}\, d\Omega = \int\limits_{\Omega^{i,j}} (-\nabla \cdot (\mathbf{v}_c\, \zeta^{i,j}) + \zeta^{i,j}(\nabla \cdot \mathbf{v}_c))\, d\Omega\,. \tag{5.12}$$

The time derivative for the cell-centered value $\zeta^{i,j}$ becomes:

$$\frac{\partial \zeta^{i,j}}{\partial t} = \frac{1}{|\Omega^{i,j}|}\left(- \int\limits_{\Omega^{i,j}} (\nabla \cdot (\mathbf{v}_c\, \zeta^{i,j})\, d\Omega + \int\limits_{\Omega^{i,j}} \zeta^{i,j}(\nabla \cdot \mathbf{v}_c))\, d\Omega \right), \tag{5.13}$$

where $|\Omega^{i,j}|$ is the surface of element (i,j). The convective velocity $\mathbf{v}_c$ is arbitrary, so in general the divergence of $\mathbf{v}_c$ is not zero. However, the value of the second integral on the right hand side is usually smaller than the value of the first integral, as the divergence of the convective velocity is often small. For example, in problems with a fixed grid in space and an incompressible material, the term $\nabla \cdot \mathbf{v}_c$ equals zero. The second integral is approximated by taking $\zeta^{i,j}$ to be constant for each element. As a result $\zeta^{i,j}$ can be taken out of the second integral of the right hand side. The volume integrals are transformed into boundary integrals using the theorem of Gauss:

$$\frac{\partial \zeta^{i,j}}{\partial t} = \frac{1}{|\Omega^{i,j}|}\left(- \int\limits_{,_{i,j}} \mathbf{n} \cdot (\mathbf{v}_c\, \zeta)\, d, \; + \zeta^{i,j} \int\limits_{,_{i,j}} \mathbf{n} \cdot \mathbf{v}_c\, d, \; \right), \tag{5.14}$$

where $\mathbf{n}$ is the normal vector pointing outwards on the boundary , ^{i,j} of the element. At the element boundary, ζ is assumed to be discontinuous. Therefore, ζ cannot be indicated with a superscript (i,j) in the first integral, which is an integral over the boundary , ^{i,j}.
The second integral is approximated for each element side using the nodal convective velocities and the normal vector. This results in

$$\frac{\zeta^{i,j}}{|\Omega^{i,j}|} \int_{,\,i,j} \mathbf{n} \cdot \mathbf{v}_c \, d, \;\; = \frac{\zeta^{i,j}}{|\Omega^{i,j}|} \sum_{k=1}^{4} h^k v_{\mathrm{n}}^k, \tag{5.15}$$

where h^k is the length of side k and v_{n}^k the average normal convective velocity of side k. The normal velocity v_{n}^k is calculated with:

$$v_{\mathrm{n}}^k = \frac{1}{2}(\mathbf{V}_{\mathrm{c}}^m + \mathbf{V}_{\mathrm{c}}^{m+1}) \cdot \mathbf{n}^k, \tag{5.16}$$

where $\mathbf{V}_{\mathrm{c}}^m$ and $\mathbf{V}_{\mathrm{c}}^{m+1}$ are the convective velocities of the nodal points at side k (see Figure 5.3).

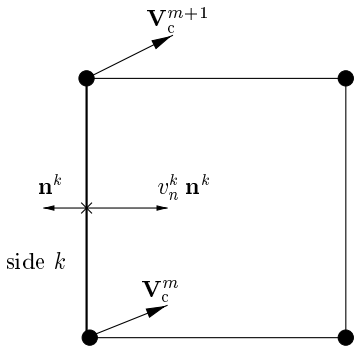


Figure 5.3: Eq. (5.15) for element side k.

The first integral of (5.14) is approximated more accurately than the second integral by taking into account the discontinuity of ζ on the element boundaries. This integral is considered as the total flux $F^{i,j}$ for element (i,j):

$$F^{i,j} = - \int_{,\,i,j} \mathbf{n} \cdot (\mathbf{v}_c \, \zeta) \, d, \;\; . \tag{5.17}$$

Integral (5.17) is split up into four fluxes as shown schematically in Figure 5.2:

$$F_{i,j} = \sum_{k=1}^{4} f^k \, . \tag{5.18}$$

The flux is a quantity associated with the side of the element. So the fluxes are calculated independently of the element. Hence, the outgoing flux for an element is equal to the ingoing flux for the other element. For example in Figure 5.2, the element (i, j) has an ingoing flux $f^{i \Leftrightarrow \frac{1}{2}, j}$ and the element $(i-1, j)$ has the same outgoing flux $f^{i \Leftrightarrow \frac{1}{2}, j}$. This makes the finite volume method a conservative method.
The flux f^k is written in the normal convective velocity v_n^k at the corresponding element side k:

$$f^k = -h^k \left(v_\mathrm{n}^k \, \zeta^k\right), \tag{5.19}$$

where h^k is the element side length and v_n^k is calculated from (5.16). The problem is now to calculate the value of ζ^k at the element boundary, as ζ is discontinuous here: obviously, the values of neighbouring elements are also needed to obtain an appropriate approximation of this value at the element boundary.

Determination of the flux

The calculation of the flux consists of determining the value ζ^k at the element side k. One-dimensional schemes are described in this section. The flux for the element side k that lies between the element i and $(i+1)$. This side is indicated with the superscript $(i + \frac{1}{2})$ (see Figure 5.4).

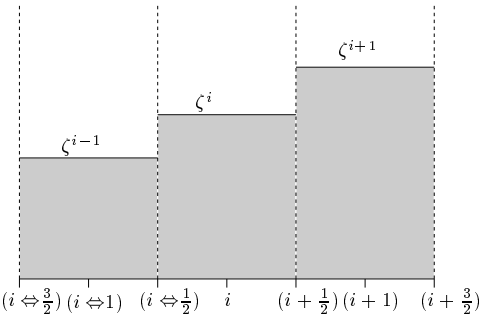


Figure 5.4: Distribution of cell-centered values.

The velocity v_c is taken positive in the direction of increasing i. The superscript j for the other local direction is omitted for the sake of clarity. With the Godunov scheme [18] the value $\zeta^{i+\frac{1}{2}}$ is approximated as follows:

$$\zeta^{i+\frac{1}{2}} = \zeta^i \,. \tag{5.20}$$

The value on the boundary is taken equal to the cell-centered value of the element upwind according to v_c. However, the Godunov scheme is only first

order accurate. Steep gradients are smoothed by this first order upwind scheme.
The order of accuracy can be improved by using the κ-schemes [59]:

$$\zeta^{i+\frac{1}{2}} = \zeta^i + \frac{1+\kappa}{4}(\zeta^{i+1} - \zeta^i) + \frac{1-\kappa}{4}(\zeta^i - \zeta^{i\Leftrightarrow 1}), \text{ for } \kappa \in [-1, 1]. \quad (5.21)$$

It can be shown that for all values of κ, the scheme is at least second-order accurate. When $\kappa = -1$ the upwind flux definition is obtained, the central flux definition is obtained when $\kappa = 1$. In this thesis we employ the $\kappa = \frac{1}{3}$ according to [34], where it is shown that this scheme is even third-order accurate for steady state situations.
Linear higher order accurate schemes cannot be monotonicity preserving according to Godunov's theorem [18]. A method that is not monotonicity preserving can result in oscillations in the distributions, especially in the neighbourhood of extrema and steep gradients. In order to prevent these oscillations a limiter function is applied. A limited scheme is written as follows:

$$\zeta^{i+\frac{1}{2}} = \zeta^i + \frac{1}{2}\phi(r_{i+\frac{1}{2}})(\zeta^i - \zeta^{i\Leftrightarrow 1}), \quad (5.22)$$

where $\phi(r)$ is the limiter function in terms of the ratio of consecutive gradients:

$$r_{i+\frac{1}{2}} = \frac{\zeta^{i+1} - \zeta^i}{\zeta^i - \zeta^{i\Leftrightarrow 1}}. \quad (5.23)$$

With the use of this limiter function, a non-linear scheme is obtained. With (5.22) the schemes $\kappa = -1$ and $\kappa = 1$ can be obtained with the specific choices of the limiter function, $\phi = 1$ and $\phi = r$ respectively (see Figure 5.5). The first order upwind scheme (5.20) is obtained when $\phi = 0$. The limiter function has to lie inside the monotonicity domain according to Sweby's monotonicity theory [51] (see Figure 5.5). A κ-scheme that lies in the monotonicity domain is a second-order accurate monotonicity preserving method.

For negative r, the limiter function $\phi = 0$, which is the first order upwind scheme (see (5.20)). In regions with smooth gradients ($r \approx 1$) the limited schemes are second-order accurate, as $\phi = 1$ for these limited schemes. However, with large variations of the gradient the limited schemes reduce to first order accuracy.
Several limiters can be applied, see *e.g.* [21]. The minmod-limiter lies on the lower boundary of the second-order accurate part of the monotonicity domain:

$$\phi(r) = \max(0, \min(r, 1)). \quad (5.24)$$

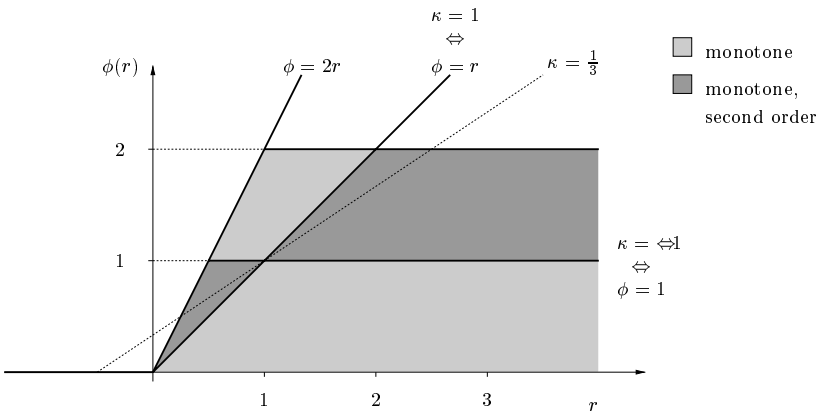


Figure 5.5: Monotonicity domain for second-order schemes.

When an initial block distribution is convected with a scheme using the minmod-limiter, the block changes into a Gaussian distribution [21]. The minmod-limiter leads to more diffusive schemes. Conversely, the Superbee-limiter is the upper bound of the second-order accurate part of the monotonicity domain:

$$\phi(r) = \max\big(0, \min(2r, 1), \min(r, 2)\big) \tag{5.25}$$

When a Gaussian distribution is convected, the Superbee-limiter tends to change it into a block distribution [21].
The limited κ=1/3-scheme [34] is employed in this thesis:

$$\zeta^{i+\frac{1}{2}} = \zeta^i + \frac{1}{2}\phi(r_{i+\frac{1}{2}})(\zeta^i - \zeta^{i \Leftrightarrow 1})\,, \tag{5.26}$$

where the limiter function is defined as:

$$\phi(r) = \max(0, \min(2r, \min(\frac{1}{3} + \frac{2}{3}r, 2)))\,. \tag{5.27}$$

The three limiter functions are illustrated in Figure 5.6.

These one-dimensional limited schemes are monotonicity preserving when they also fulfill the *CFL*-condition for the Courant number C:

$$C = \frac{v_c \Delta t}{\Delta x} < 1\,, \tag{5.28}$$

where Δx is the element length in the x-direction. In the two-dimensional case using a one-dimensional scheme in each direction the method is not monotonicity preserving. Multi-dimensional limiting can be performed with the multi-dimensional flux corrected transport (MFCT) method [63]. The application

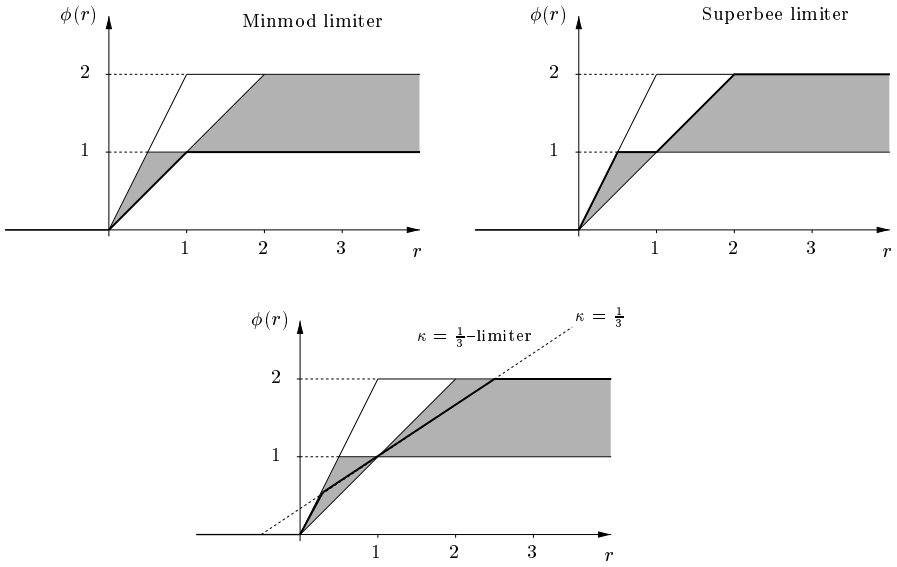


Figure 5.6: Limiter functions in the monotonicity domain.

of this multi-dimensional flux limiter is computationally expensive, as extra loops over the elements have to be taken to determine the limiter values. Multi-dimensional limiters are not considered in this thesis. Furthermore, experience shows that the application of limited one-dimensional schemes in two-dimensional cases usually give oscillation-free results (see Chapter 6). For stability and monotonicity the time step should be chosen small enough.

The calculation of the value $\zeta^{i+\frac{1}{2}}$ is now completed. For each element side a flux f is calculated using (5.19). With the forward Euler method the increment $\Delta\zeta_c$ is determined from (5.8) and (5.14), where the two integrals of (5.14) are computed using (5.18) and (5.15). Finally, the material increment $\Delta\zeta_m$ and $\Delta\zeta_c$ are added to ζ_{ref} which gives the updated value ζ' (5.6).

5.2.2 Scheme for nodal values

In this section, in contrast with the previous section, it is assumed that the state variable is known at the nodal points. So we have a continuous distribution of the state variable across the element boundaries. The scheme of this section uses the finite element interpolation to discretise the convective increment $\Delta\zeta_c$ in (5.8).
The gradient of the state variable is determined in an upwind manner with respect to the convective velocity $\mathbf{v}_c$ to obtain a stable solution of the convec-

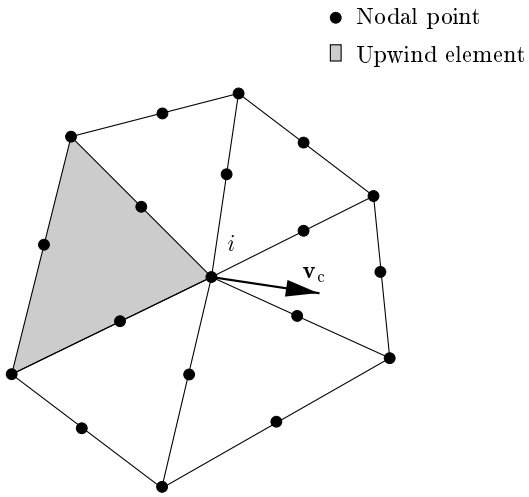


Figure 5.7: Element upwind with respect to the convective velocity $\mathbf{v}_c$ for a nodal point i.

tion equation. The distribution of the state variable in the element upwind according to the convective velocity is known by interpolation of the nodal points values ζ^n:

$$\zeta(x,y)_{\text{upw}} = \sum_n \zeta^n N^n(\boldsymbol{\xi}). \tag{5.29}$$

This is illustrated for quadratic triangular elements in Figure 5.7. The gradient at a nodal point i is determined from the upwind element only:

$$\nabla\zeta^i_{\text{upw}} = \sum_n \zeta^n \nabla(N^n(\boldsymbol{\xi}^i)). \tag{5.30}$$

With this spatial gradient and a forward Euler time integration the increment $\Delta\zeta_c$ of (5.8) is calculated. The updated value ζ' is obtained from (5.6) at the nodal point.
The same finite element data structure as in the updated Lagrangian step is applied to the calculation of the gradient, which makes the method easy to implement. Only local information from one element connected to a nodal point is needed, which makes it an efficient method. The upwind element for the nodal points has to be found from the surrounding elements. In the next increment the only verification required is whether the upwind element of the previous increment is still valid. When this is no longer the case, the upwind element has to be sought again among the neighbouring elements. Since the convective velocity in general does not vary much between two increments, the

upwind element normally remains unchanged.

Now the convection approaches have been discussed, in the following section the interpolation approach is presented.

5.3 Interpolation approach

In contrast to the convection approaches of Section 5.2 we use here the so-called interpolation approach. In the convection approach an increment due to the convective term is calculated. In the interpolation approach we perform an interpolation step after the material time derivative has been integrated in time. The state variable ζ is first updated with respect to the material time derivative:

$$\zeta_\ell = \zeta_{\text{ref}} + \int_{t_0}^{t_1} \dot{\zeta}\ dt. \tag{5.31}$$

The nodal coordinates are updated with respect to the material displacement $\Delta\mathbf{u}$:

$$\mathbf{x}_\ell = \mathbf{x}_{\text{ref}} + \Delta\mathbf{u}. \tag{5.32}$$

Hence we know ζ_ℓ at the updated Lagrangian coordinates $\mathbf{x}_\ell$. In the ALE method the domain at end of the time step is obtained from (4.1) and (5.32):

$$\mathbf{x}' = \mathbf{x}_\ell - \Delta\mathbf{u}_{\text{c}}, \tag{5.33}$$

where

$$\Delta\mathbf{u}_{\text{c}} = \int_{t_0}^{t_1} \mathbf{v}_{\text{c}}\ dt. \tag{5.34}$$

The state variable on this domain $\mathbf{x}'$ becomes:

$$\zeta'(\mathbf{x}') = \zeta_\ell(\mathbf{x}_\ell - \Delta\mathbf{u}_{\text{c}}). \tag{5.35}$$

The calculation of ζ' at $\mathbf{x}'$ can be considered as an interpolation problem where the value ζ_ℓ is known at the coordinates $\mathbf{x}_\ell$.
Several methods exist for solving an interpolation problem within the framework of the ALE method. Often they are based on a reconstruction of the state variable field by calculating average nodal point values from the integration

point values [28, 20, 55]. The second step consists of interpolating between the nodal points to the position of the new integration points. These methods fit well into the context of a finite element method. However, the step from integration point values to average nodal values tends to smooth steep gradients between elements, when element averages are used for the calculation of nodal values [26]. As a result these methods can become inaccurate.

In this thesis we apply the method of Van Leer [58]. This method is capable of handling the discontinuities between elements. In [5] the Van Leer scheme was also applied to the remap step of the ALE method. In the following two sections the method of Van Leer is described for quadrilateral and triangular elements. In the case of a quadrilateral element, we split the remap problem into two one-dimensional problems. For triangular elements a two-dimensional approach is worked out in detail.

5.3.1 Van Leer scheme for quadrilateral elements

The starting point for this method is a linear distribution of ζ_ℓ for each element that is constructed using a least–square fit of the integration point values:

$$\zeta_\ell(x,y) = \zeta_\ell^i + (\zeta_\ell^i)_{,x}\,(x - x_\ell^i) + (\zeta_\ell^i)_{,y}\,(y - y_\ell^i), \tag{5.36}$$

where the mean value ζ_ℓ^i and the spatial derivatives $(\zeta_\ell^i)_{,x}$ and $(\zeta_\ell^i)_{,y}$ are defined at the centroid of the element (x_ℓ^i, y_ℓ^i). The piecewise linear distribution is used for the interpolation step. There is no averaging of integration point values, which can result in a loss of accuracy. We use the method of Van Leer [58] to discretise the interpolation problem (5.35) with the piecewise linear distribution constructed with ζ_ℓ^i, $(\zeta_\ell^i)_{,x}$ and $(\zeta_\ell^i)_{,y}$. The two local directions x and y of an element are treated independently and in the same way. Hence we describe the method in one of the two directions. The main differences compared to the original scheme of Van Leer [58] are: the convective displacements are known at the nodal points instead of at the centroid of the element and the grid is not fixed in space. The mesh displacements are more or less arbitrary, so an element can change in size after the convective displacements have been added. This results in a more complicated scheme than the original Van Leer method.

In Figure 5.8, the remap of a piecewise linear distribution for element i is demonstrated in one dimension. First of all, a discontinuous distribution within the new elements is obtained. A new piecewise linear distribution is obtained in the new element mesh by a least square approximation of the

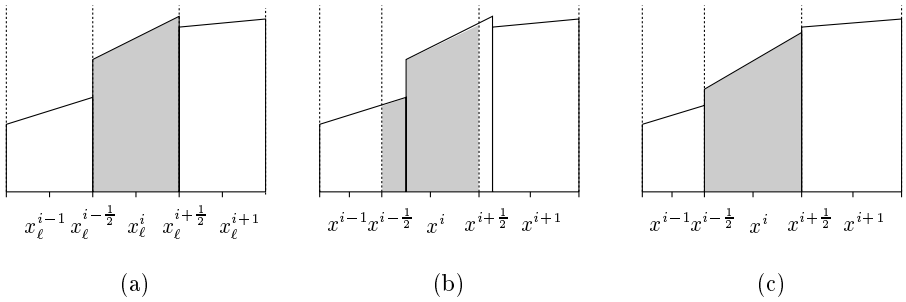


Figure 5.8: Remap of a piecewise linear distribution in one dimension. (a) A piecewise linear distribution with a mean value ζ_l^i and a mean gradient $(\zeta_l^i)_{,x}$ for element i. (b) In the shifted element mesh indicated with the dashed lines a discontinuous distribution exists in the element with centroid x^i. (c) A new piecewise linear distribution (ζ^i and $(\zeta^i)_{,x}$) is constructed with a least square approximation ((5.39) and (5.40)).

discontinuous distribution. The least square error Φ in direction x is:

$$\Phi = \int_{x^{i-\frac{1}{2}}}^{x^{i+\frac{1}{2}}} (\zeta_\ell - \zeta')^2 dx. \tag{5.37}$$

The new distribution ζ' in the x-direction is described by the mean value ζ^i and the gradient $(\zeta^i)_{,x}$ around the centroid x^i:

$$\zeta'(x,y) = \zeta^i + (\zeta^i)_{,x}\,(x - x^i)\,. \tag{5.38}$$

Minimisation to the unknowns ζ^i and $(\zeta^i)_{,x}$ results in the following integrals over the discontinuous distribution in the new element:

$$\zeta^i = \frac{1}{\Delta x^i} \int_{x^{i-\frac{1}{2}}}^{x^{i+\frac{1}{2}}} \zeta_\ell \, dx, \tag{5.39}$$

$$(\zeta^i)_{,x} = \frac{12}{(\Delta x^i)^3} \int_{x^{i-\frac{1}{2}}}^{x^{i+\frac{1}{2}}} \zeta_\ell \,(x - x^i)\, dx, \tag{5.40}$$

where $\Delta x^i = \mid x^{i+\frac{1}{2}} - x^{i \Leftrightarrow \frac{1}{2}} \mid$. In these integrals the superscript $'$ is omitted for clarity. The variables ζ^i and $(\zeta^i)_{,x}$ are calculated using the integrals over

the discontinuous distributions of the right hand side of (5.39) and (5.40). In Appendix B the integrals of (5.39) and (5.40) are worked out in detail.
In order to prevent possible oscillations the gradients of the updated Lagrangian distribution $(\zeta_\ell^i)_{,x}$, $(\zeta_\ell^i)_{,y}$ are limited for one dimension in the same manner as in [58]:

$$(\zeta_\ell^i)_{,x}^{\text{mono}} = \begin{cases} \min\big(2 \mid \zeta_\ell^i - \zeta_\ell^{i \Leftrightarrow 1} \mid, \mid (\zeta_\ell^i)_{,x}\, \Delta x_\ell^i \mid, 2 \mid \zeta_\ell^{i+1} - \zeta_\ell^i \mid\big)\, \text{sgn}((\zeta_\ell^i)_{,x}) \\ \qquad \text{if } \ \text{sgn}(\zeta_\ell^i - \zeta_\ell^{i \Leftrightarrow 1}) = \text{sgn}((\zeta_\ell^i)_{,x}) = \text{sgn}(\zeta_\ell^{i+1} - \zeta_\ell^i) \\ 0 \qquad \text{otherwise.} \end{cases} \tag{5.41}$$

In Figure 5.9 the limiting of expression (5.41) is explained.

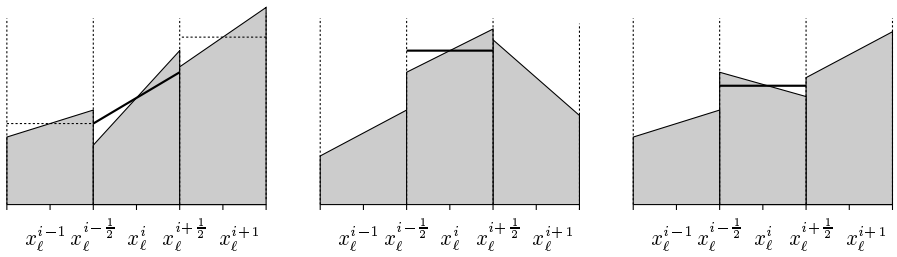


Figure 5.9: Illustration of the limiter (5.41); the bold line indicates the limited distribution and the dashed line indicates mesh averages. (a) The values should not go beyond the average values of the neighbouring elements. (b) Near an extremum the gradient is limited. (c) The gradient has to correspond to the global gradient of the mesh averages.

Normally, in the two-dimensional case, the method is monotonicity preserving when the time step is small enough. This will be shown experimentally in Chapter 6. Finally, the integration point values in the new element mesh are calculated from the piecewise linear distribution:

$$\zeta'(x,y) = \ \zeta^i \ + \ (\zeta^i)_{,x}\,(x - x^i) \ + \ (\zeta^i)_{,y}\,(y - y^i)\,. \tag{5.42}$$

This new state variable ζ' is then used as reference state for the next time step of the ALE method.

5.3.2 Van Leer scheme applied to triangular elements

In contrast to the previous section the Van Leer scheme is now applied to triangular elements [56]. However, in case of triangular elements the equations

cannot be split up into two one-dimensional problems. So the approach in this section differs in this aspect from the scheme for quadrilateral elements of the previous section. Again we start with a linear distribution in the element i:

$$\zeta_\ell(x,y) = \zeta_\ell^i + (\zeta_\ell^i)_{,x}\,(x - x_\ell^i) + (\zeta_\ell^i)_{,y}\,(y - y_\ell^i)\,, \tag{5.43}$$

where the mean value ζ_ℓ^i and the gradients $(\zeta_\ell^i)_{,x}$ and $(\zeta_\ell^i)_{,y}$ are again defined in the centroid of the element with the coordinates (x_ℓ^i, y_ℓ^i). The new distribution is again obtained by minimising the least square error. In contrast with the quadrilateral elements we take the least square error across the element:

$$\Phi = \int_{\Omega^i} (\zeta_\ell - \zeta')^2 d\Omega. \tag{5.44}$$

Minimisation of Φ to the three unknowns ζ^i, $(\zeta^i)_{,x}$ and $(\zeta^i)_{,y}$ of ζ' leads to the three following equations:

$$\int_{\Omega^i} \zeta^i \, d\Omega = \int_{\Omega^i} \zeta_\ell \, d\Omega, \tag{5.45}$$

$$\int_{\Omega^i} (\zeta^i)_{,x}\,(x - x^i)^2 + (\zeta^i)_{,y}\,(y - y^i)(x - x^i) d\Omega = \int_{\Omega^i} \zeta_\ell\,(x - x^i)\,d\Omega, \tag{5.46}$$

$$\int_{\Omega^i} (\zeta^i)_{,x}\,(x - x^i)(y - y^i) + (\zeta^i)_{,y}\,(y - y^i)^2\,d\Omega = \int_{\Omega^i} \zeta_\ell\,(y - y^i)\,d\Omega. \tag{5.47}$$

The integrals are taken over the *new* subdomain Ω^i of the element i. Hence, the right hand sides consist of integrals over the discontinuous distribution ζ_ℓ.

The integrals on the right hand side are approximated with the help of the Figures 5.10 and 5.11:

$$\int_{\Omega^i} \zeta^i \, d\Omega = \int_{\Omega_\ell^i} \zeta_\ell^i \, d\Omega - \sum_{k=1}^{3} \int_{\Omega_\ell^i \cap \Omega^k} \zeta_\ell^i \, d\Omega + \sum_{k=1}^{3} \int_{\Omega^i \cap \Omega_\ell^k} \zeta_\ell^k \, d\Omega, \tag{5.48}$$

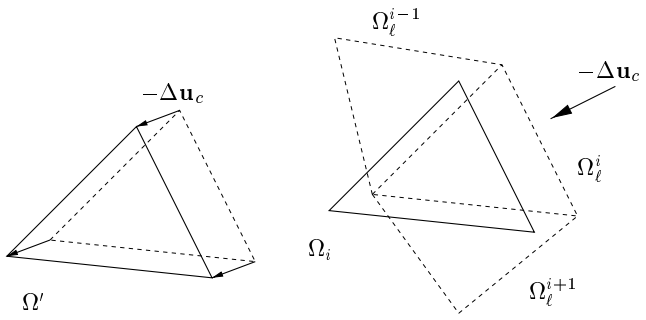


Figure 5.10: Relation between updated Lagrangian elements (dashed lines) and elements at the end of the time step (solid lines).

$$\begin{aligned}
&\int_{\Omega^i} \left((\zeta^i)_{,x} (x - x^i)^2 + (\zeta^i)_{,y} (y - y^i)(x - x^i) \right) d\Omega = \\
&\quad \int_{\Omega_\ell^i} \zeta_\ell^i (x - x^i)\, d\Omega - \sum_{k=1}^{3} \int_{\Omega_\ell^i \cap \Omega^k} \zeta_\ell^i (x - x^i)\, d\Omega \\
&\quad + \sum_{k=1}^{3} \int_{\Omega^i \cap \Omega_\ell^k} \zeta_\ell^k (x - x^i)\, d\Omega,
\end{aligned} \tag{5.49}$$

$$\begin{aligned}
&\int_{\Omega^i} \left((\zeta^i)_{,x} (x - x^i)(y - y^i) + (\zeta^i)_{,y} (y - y^i)^2 \right) d\Omega = \\
&\quad \int_{\Omega_\ell^i} \zeta_\ell^i (y - y^i)\, d\Omega - \sum_{k=1}^{3} \int_{\Omega_\ell^i \cap \Omega^k} \zeta_\ell^i (y - y^i)\, d\Omega \\
&\quad + \sum_{k=1}^{3} \int_{\Omega^i \cap \Omega_\ell^k} \zeta_\ell^k (y - y^i)\, d\Omega,
\end{aligned} \tag{5.50}$$

where k is a neigbouring element having an element side in common with element i. From these equations the mean value ζ^i and the gradients $(\zeta^i)_{,x}$ and $(\zeta^i)_{,y}$ are obtained.

When calculating these integrals we assume that the element sides remain straight. This is ensured by the meshing method (see Chapter 4). Only the sides of boundary elements can become curved. However, on the boundary

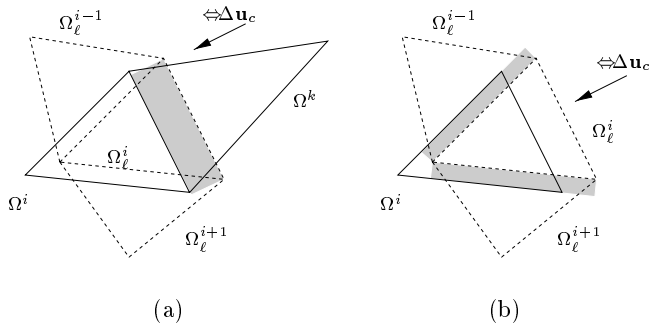


Figure 5.11: Illustration of (5.48), (5.49) and (5.50) by the shaded areas. (a) Outgoing fluxes from the integrals over $\Omega_\ell^i \cap \Omega^k$; (b) Ingoing fluxes from the integrals over $\Omega^i \cap \Omega_\ell^k$.

itself, nodal points can only move along the boundary:

$$(\mathbf{v} - \mathbf{v}_g) \cdot \mathbf{n} = 0 \qquad \text{on} \quad , . \tag{5.51}$$

So again no flux is transported across these curved boundary element sides.

5.4 Conclusion

In this chapter a number of schemes for the remap problem in the ALE method were discussed. Two different approaches were applied for the convective term of the remap problem (5.4): the convection approach and the interpolation approach. The convection approach is based on the integration of the grid time derivative (5.5). Two finite volume schemes were described: the first order upwind scheme and the $\kappa = \frac{1}{3}$-scheme of Section 5.2.1. These two schemes are based on the remap of integration point values and they calculate a constant convective increment for each element. A third scheme was presented in Section 5.2.2, based on the remap of nodal values.
In the interpolation approach, first of all the material time derivative is integrated (5.31). Then the remap problem (5.4) turns into an interpolation problem. Two schemes for quadrilateral and triangular elements were given. Both schemes are based on the method of Van Leer, in which piecewise linear distributions are assumed.
In the next chapter the ALE method is used in the simulation of several applications. In these applications the various schemes introduced in this chapter are investigated.

Chapter 6

Applications

In the previous chapters the ALE method was described in detail. In the current chapter we apply the ALE method to various problems. First, two academic test cases are used to investigate the method. One case is the Molenkamp test (Section 6.1) and the other is a case for which the exact solution is also known (Section 6.2). After the academic test cases we discuss simulations of several forming processes (Section 6.3), such as forging, upsetting and extrusion. In these simulations the different aspects of the ALE method are shown and its applicability is discussed.

6.1 Molenkamp test

The Molenkamp test case [61] is a well-known test case used in computational fluid dynamics to check convection algorithms. In this test a Gaussian profile is rotated on a square domain. After one full rotation the initial and the calculated distributions are compared. We use this test case to investigate the remap algorithm of the ALE method (see Chapter 5). The material velocity field, describing a rigid body rotation, is prescribed as follows:

$$v_x = -2\pi y \; ; \quad v_y = 2\pi x, \tag{6.1}$$

in the domain:

$$-1 \leq x \leq 1 \; ; \quad -1 \leq y \leq 1. \tag{6.2}$$

The element mesh is kept fixed in space ($\mathbf{v}_{\mathrm{g}} = \mathbf{0}$), so the grid time derivative for a state variable ζ (5.3) becomes:

$$\frac{\partial_{\mathrm{g}} \zeta}{\partial t} = -\mathbf{v} \cdot \nabla(\zeta). \tag{6.3}$$

The exact solution for ζ for $t \in [0,1]$ is

$$\zeta_{\text{exact}}(x,y,t) = 0.01^{4r^2}, \tag{6.4}$$

where

$$r = \sqrt{(x + \frac{1}{2}\cos 2\pi t)^2 + (y + \frac{1}{2}\sin 2\pi t)^2}\,. \tag{6.5}$$

The initial distribution for ζ is prescribed at $t = 0$ (see Figure 6.1) and then one full revolution of the Gaussian profile is performed ($t = 1$).

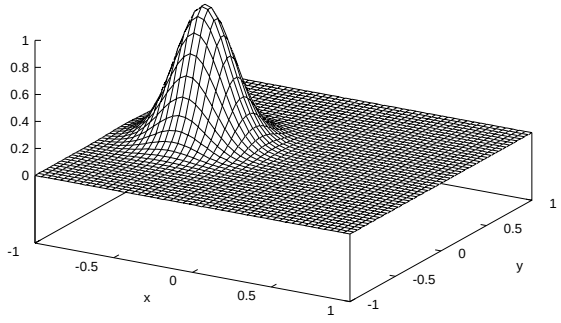


Figure 6.1: Initial distribution of ζ.

The following errors are determined in order to investigate the accuracy of the calculations:

$$E_1 = \frac{\sum_{i=1}^{N} \mid (\zeta - \zeta_{\text{exact}})_i \mid}{N}, \qquad E_2 = \sqrt{\frac{\sum_{i=1}^{N} (\zeta - \zeta_{\text{exact}})_i^2}{N}}, \qquad E_\infty = \max \mid (\zeta - \zeta_{\text{exact}})_i \mid, \tag{6.6}$$

where the exact solution ζ_{exact} is known by (6.4) and N is the number of nodal points.

6.1.1 Quadrilateral meshes

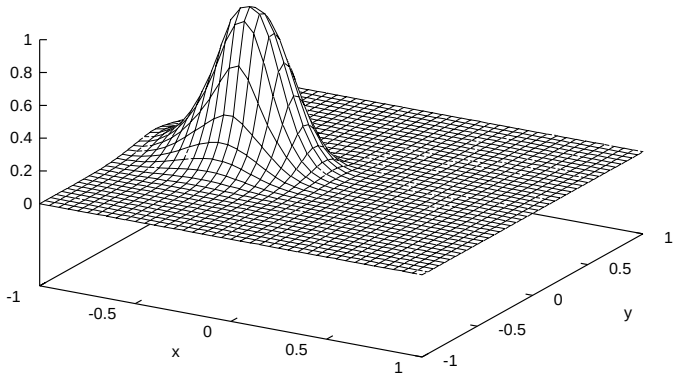
First the Molenkamp test is carried out with quadrilateral elements. Three meshes are used with 400 (mesh I), 1600 (mesh II) and 6400 (mesh III) elements

with corresponding time step refinements $\Delta t = 0.005, 0.0025$ and 0.00125 respectively. The Molenkamp test is thus carried out with the same Courant numbers (see (5.28)) on the different meshes. At time $t = 1$, the cone has performed exactly one revolution. Four different numerical schemes were used: the first order upwind scheme, the $\kappa = \frac{1}{3}$-scheme of Section 5.2.1 with a forward Euler time integration scheme, the $\kappa = \frac{1}{3}$-scheme with a fourth order Runge–Kutta time integration scheme [31] and the Van Leer scheme of Section 5.3.1. The $\kappa = \frac{1}{3}$-scheme is performed using Runge–Kutta time stepping in order to investigate the influence of the time integration.

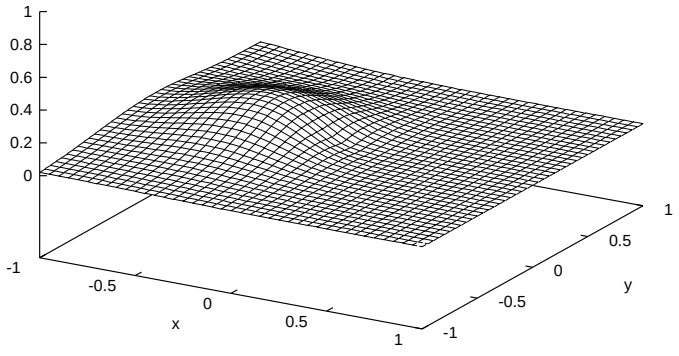
Scheme	mesh	$\zeta_{\max}$	$\zeta_{\min}$	E_1	E_2	E_∞
First order	I	0.14	0.	$3.9 \cdot 10^{\Leftrightarrow 2}$	$1.1 \cdot 10^{\Leftrightarrow 1}$	0.88
upwind	II	0.24	0.	$3.4 \cdot 10^{\Leftrightarrow 2}$	$9.8 \cdot 10^{\Leftrightarrow 2}$	0.77
	III	0.39	0.	$2.6 \cdot 10^{\Leftrightarrow 2}$	$7.6 \cdot 10^{\Leftrightarrow 2}$	0.62
Van Leer	I	0.69	$-2.4 \cdot 10^{\Leftrightarrow 2}$	$1.6 \cdot 10^{\Leftrightarrow 2}$	$4.9 \cdot 10^{\Leftrightarrow 2}$	0.33
	II	0.96	$-9.2 \cdot 10^{\Leftrightarrow 3}$	$8.6 \cdot 10^{\Leftrightarrow 3}$	$2.5 \cdot 10^{\Leftrightarrow 2}$	0.17
	III	1.03	$-4.8 \cdot 10^{\Leftrightarrow 4}$	$4.8 \cdot 10^{\Leftrightarrow 3}$	$1.5 \cdot 10^{\Leftrightarrow 2}$	0.16
$\kappa = \frac{1}{3}$	I	0.60	$-3.1 \cdot 10^{\Leftrightarrow 11}$	$1.4 \cdot 10^{\Leftrightarrow 2}$	$4.6 \cdot 10^{\Leftrightarrow 2}$	0.40
Forward Euler	II	0.91	$-2.8 \cdot 10^{\Leftrightarrow 10}$	$1.5 \cdot 10^{\Leftrightarrow 2}$	$4.6 \cdot 10^{\Leftrightarrow 2}$	0.35
	III	0.98	$-3.9 \cdot 10^{\Leftrightarrow 10}$	$1.1 \cdot 10^{\Leftrightarrow 2}$	$3.5 \cdot 10^{\Leftrightarrow 2}$	0.28
$\kappa = \frac{1}{3}$	I	0.48	$-1.3 \cdot 10^{\Leftrightarrow 10}$	$2.0 \cdot 10^{\Leftrightarrow 2}$	$6.3 \cdot 10^{\Leftrightarrow 2}$	0.53
Runge–Kutta	II	0.76	$-1.5 \cdot 10^{\Leftrightarrow 10}$	$5.3 \cdot 10^{\Leftrightarrow 3}$	$2.0 \cdot 10^{\Leftrightarrow 2}$	0.25
	III	0.92	$-2.2 \cdot 10^{\Leftrightarrow 10}$	$9.3 \cdot 10^{\Leftrightarrow 4}$	$4.3 \cdot 10^{\Leftrightarrow 3}$	0.09

Table 6.1: Results of the Molenkamp test for quadrilateral meshes.

The results of the calculations are shown in Table 6.1. When we look at the errors, it is clear that the first order upwind scheme is the most inaccurate scheme. The $\kappa = \frac{1}{3}$-scheme with the Runge–Kutta time stepping is the most accurate scheme on mesh III. When it is compared to the results of the $\kappa = \frac{1}{3}$-scheme with a forward Euler time stepping, it turns out that the latter is worse due to a less accurate time integration. The Van Leer scheme is more accurate than the $\kappa = \frac{1}{3}$-scheme with forward Euler time stepping, but not as good as the $\kappa = \frac{1}{3}$-scheme with Runge–Kutta time stepping.
The initial distribution has a minimum of 0 and a maximum of 1, so extrema that are beyond these initial limits indicate the presence of oscillations. The first order upwind scheme does not introduce any oscillations. The other three schemes show hardly any oscillations. The Van Leer scheme shows some oscillations, which can be suppressed using a more diffusive limiter. Although the schemes hardly show oscillations in these cases, it is known that oscillations will grow if the time step is increased.
In Figure 6.2 the distributions are shown of the Van Leer scheme and the



(a) Van Leer scheme



(b) First order upwind scheme

Figure 6.2: Results of the Molenkamp test on mesh II

first order upwind scheme on mesh II. Comparing the results with the initial distribution (Figure 6.1) one can see that the maximum value is reduced, but also that the value at the boundary is increased. Clearly, the result of the Van Leer scheme is better than the first order upwind scheme.
The Molenkamp test is a good test case for numerical convection schemes, as it gives a good impression of their behaviour. However, it must be noted that the schemes behave well in the case of homogeneous and structured meshes. In general, in ALE calculations this is not always the case. However, the transfinite mapping method of Chapter 4, as we suggest, can be used to obtain structured meshes.

For quadrilateral meshes the $\kappa = \frac{1}{3}$-scheme with Runge–Kutta time stepping is more accurate, but computationally more expensive. Note also that the accuracy of the remap of state variables should be compared to the accuracy of the updated Lagrangian step. When the state variables are calculated with a first order time integration in the updated Lagrangian step, it is not necessary to calculate the convective terms with a fourth order time integration. Hence Runge–Kutta time stepping is not required for the remap of state variables in the ALE method.

6.1.2 Triangular meshes

In this section we carry out the Molenkamp test on quadratic triangular element meshes. The schemes of Sections 5.2.2 and 5.3.2 are used.
First we discuss the scheme of Section 5.3.2, based on the Van Leer method. The numerical scheme is applied to the integration point values. The element meshes are chosen such that we have nearly the same number of elements as in the quadrilateral case. In this way we are able to compare the results with the results obtained on the quadrilateral meshes of the previous section. Hence, we use triangular element meshes with 15×15, 29×29 and 57×57 vertices and 392, 1568 and 6272 elements respectively. The same time steps are applied as in the previous section, so $\Delta t = 0.005, 0.0025$ and 0.00125 respectively. Then the Molenkamp test is performed with more or less the same Courant number. In Table 6.2 the results of the calculations are given. The error norms are smaller than in the results of the Van Leer method for quadrilateral elements. In the Van Leer scheme for quadrilateral elements limiters are applied and consists of two one-dimensional schemes. For the triangular elements no limiters are applied and the scheme takes into account the two dimensions in a coupled manner. This could be an explanation for the more accurate results of the Van Leer based scheme for triangular elements.
In the Figures 6.3 and 6.4 the results are shown for this scheme on a structured

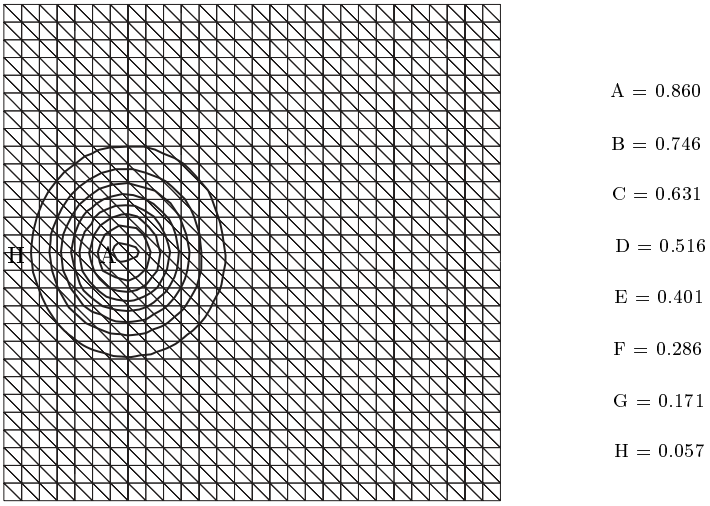


Figure 6.3: Result on the mesh with 29×29 vertices and the scheme of Section 5.3.2.

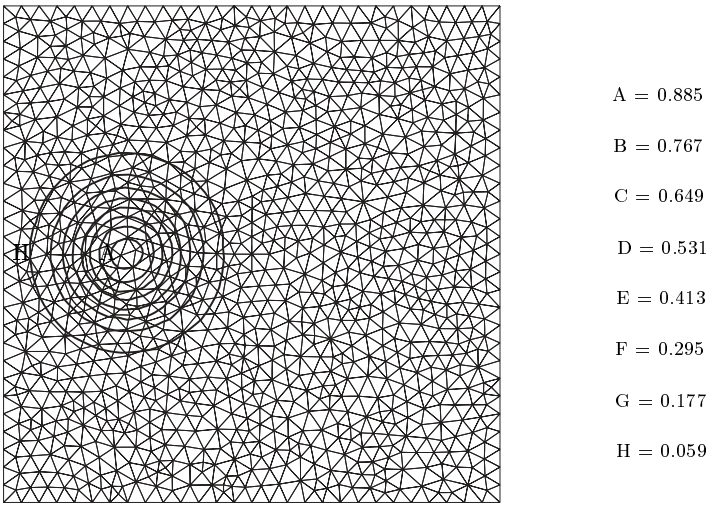


Figure 6.4: Result on an unstructured element mesh and the scheme of Section 5.3.2.

mesh (mesh II) and also on an unstructured mesh.

Scheme	mesh	$\zeta_{\max}$	$\zeta_{\min}$	E_1	E_2	E_∞
Van Leer	I	0.74	$-7.6 \cdot 10^{\Leftrightarrow 3}$	$1.1 \cdot 10^{\Leftrightarrow 2}$	$3.4 \cdot 10^{\Leftrightarrow 2}$	0.27
	II	0.92	$-7.6 \cdot 10^{\Leftrightarrow 4}$	$4.3 \cdot 10^{\Leftrightarrow 3}$	$1.4 \cdot 10^{\Leftrightarrow 2}$	0.13
	III	0.99	$-2.7 \cdot 10^{\Leftrightarrow 7}$	$2.1 \cdot 10^{\Leftrightarrow 3}$	$6.6 \cdot 10^{\Leftrightarrow 3}$	0.07

Table 6.2: Results of the Molenkamp test for triangular meshes.

Now we discuss the scheme of Section 5.2.2. In this scheme the state variable is assumed to be located at the nodal points. The Molenkamp test was performed with the quadratic element meshes I, II and III with 11×11, 21×21 and 41×41 vertices with time steps $\Delta t = 0.005$, 0.0025 and 0.00125 respectively. With these meshes the number of nodal points equals the number of elements of the quadrilateral meshes of the previous section. As a result the number of unknowns is also equal.

scheme	mesh	$\zeta_{\max}$	$\zeta_{\min}$	E_1	E_2	E_∞
Scheme for	I	0.61	$-7.6 \cdot 10^{\Leftrightarrow 2}$	$3.0 \cdot 10^{\Leftrightarrow 2}$	$7.4 \cdot 10^{\Leftrightarrow 2}$	0.47
nodal values	II	1.02	$-1.4 \cdot 10^{\Leftrightarrow 1}$	$2.2 \cdot 10^{\Leftrightarrow 2}$	$5.8 \cdot 10^{\Leftrightarrow 2}$	0.35
	III	1.17	$-6.5 \cdot 10^{\Leftrightarrow 2}$	$1.2 \cdot 10^{\Leftrightarrow 2}$	$3.6 \cdot 10^{\Leftrightarrow 2}$	0.27

Table 6.3: Results of the Molenkamp test the scheme for nodal values.

In Table 6.3 the results of the three calculations are shown. The error norms are compared with the results of Tables 6.2 and 6.1. Clearly, this scheme is more accurate than the first order upwind scheme. The results of the scheme of Section 5.2.2 are comparable to the $\kappa = \frac{1}{3}$-scheme with forward Euler time integration. However, the maximum and the minimum values indicate that oscillations are introduced by the numerical scheme. The maximum value is increased and the minimum value is decreased, so the scheme is not monotonicity preserving. It is possible to use limiting techniques from computational fluid dynamics to preserve monotonicity, *e.g.* [63] or [52]. However, this is difficult for triangular elements. Although the method is not monotonicity preserving it seems to be stable in this test case. In ALE calculations of metal forming processes usually oscillation-free results are seen (see for example Section 6.2).

6.2 Verification with an analytical solution

In this section we carry out another academic test. A case is constructed for which the exact solution is known; in contrast with the Molenkamp test, in

this test case the material velocity and the mesh velocity have to be calculated. In the first section the exact solution is given. Subsequently, the calculations are performed using the FORGE2® software package and the ALE method is compared to the updated Lagrangian method. The combination of errors arising from mesh distortion and those arising from the remap of state variables is investigated. Finally, the two Laplacian based schemes for the calculation of the mesh velocity (see Section 4.2) are compared with respect to mesh distortion. The evolution of the boundary, which is an important result of the simulations, is also investigated.

6.2.1 Description of the case

In Figure 6.5 the applied domain with the symmetry axes (x and y) is shown.

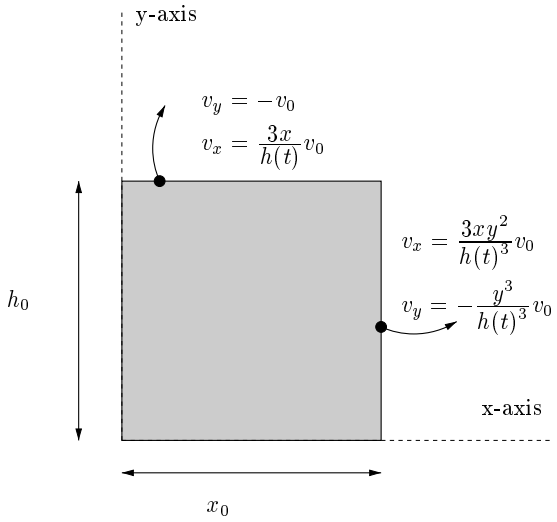


Figure 6.5: Description of the analytical case at time $t = 0$.

In this test case we have the following constants: a velocity $v_0 = 1$, initial lengths $h_0 = 50$ and $x_0 = 50$. The function $h(t) = h_0 - v_0\, t$ is the y-coordinate of the upper boundary of the domain moving downwards with a constant velocity v_0:

$$\frac{\partial h(t)}{\partial t} = -v_0. \tag{6.7}$$

For this test case we assume the incompressible material velocity field to be

$$v_x = \frac{3xy^2}{h(t)^3} v_0 \qquad ; \qquad v_y = -\frac{y^3}{h(t)^3} v_0. \tag{6.8}$$

The Newtonian model (2.40) gives an expression for the deviatoric stress tensor:

$$\mathbf{s} = \eta_0 \, \mathbf{D}. \tag{6.9}$$

This simplification does not affect the specific characteristics of an ALE calculation, but this enables us to calculate an exact solution. The velocity field (6.8) gives a deformation rate tensor, using (2.24), as follows:

$$\mathbf{D}(x,y) \;=\; \frac{3\, v_0}{h(t)^3} \begin{bmatrix} y^2 & xy \\ xy & -y^2 \end{bmatrix}. \tag{6.10}$$

We substitute (6.10) into (6.9) and then the pressure can be obtained using the quasi–static equilibrium equation (2.57):

$$p = \frac{3\eta_o v_0}{2h(t)^3}(x^2 - y^2) + C_1, \tag{6.11}$$

where C_1 is a constant. So the velocity field (6.8) together with the pressure field (6.11) fulfill the equilibrium equation (2.57), the constitutive model (6.9) and the incompressibility condition:

$$(\overrightarrow{\nabla} \cdot \mathbf{v}) = 0. \tag{6.12}$$

At the upper boundary and the right hand side of the domain the exact solution for the velocity field is used as a boundary condition (see Figure 6.5). This ensures that the velocity field fulfills the boundary conditions.
The equivalent strain rate $\dot{\bar{\varepsilon}}(x,y)$ (2.34) at time t becomes:

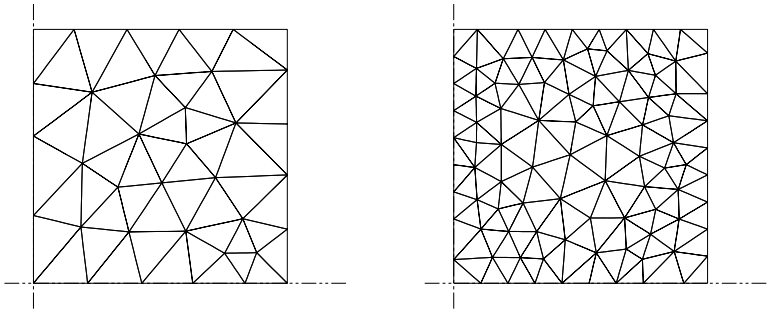
$$\dot{\bar{\varepsilon}}(x,y) = \frac{\sqrt{12}\; v_0}{h(t)^3} \sqrt{y^4 + x^2 y^2}. \tag{6.13}$$

The equivalent strain $\bar{\varepsilon}(x,y)$ at time t is obtained by numerical integration of $\dot{\bar{\varepsilon}}$ along the trajectories. The trajectory for the particle at position (x,y) at time t is known from the velocity field. First, the initial position is calculated, and then the numerical integration of (6.13) along the trajectory is performed.

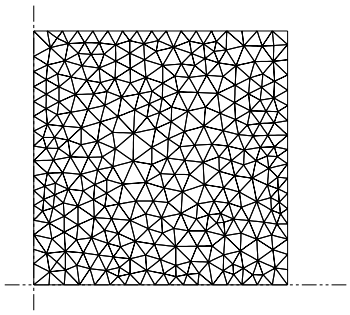
The calculations are carried out on triangular meshes using the updated Lagrangian method and the ALE method. Before we can discuss the results a number of definitions is required for the evaluation of the results. Moreover, a description of the simulations is given.

Description of the error norms and the meshes

The calculations were carried out on quadratic triangular meshes with increasing mesh refinement with 51, 154 and 599 elements (see Figure 6.6) and 40, 80 and 170 increments respectively from $t_{\text{begin}} = 0\,\text{s}$ to $t_{\text{end}} = 15\,\text{s}$.



(a) Mesh I (b) Mesh II



(c) Mesh III

Figure 6.6: Element meshes for this test case.

The ALE method was applied with the two mesh velocity methods based on the Laplace equation of Section 4.2. In the updated Lagrangian step the strain rate $\dot{\bar{\varepsilon}}$ is determined in the nodal points from the material velocity field [36]. Since the equivalent strain is calculated at the nodal points the remap algorithm of Section 5.2.2 for nodal points is applied.
In order to quantify the amount of mesh distortion, the quality of an element is defined here. For this we need the transformation $\mathbf{T}^i$ of an element i to the referential equilateral element (Figure 6.7):

$$\mathbf{T}^i = \begin{bmatrix} \frac{\partial x}{\partial \xi_{eq}} & \frac{\partial x}{\partial \eta_{eq}} \\ \frac{\partial y}{\partial \xi_{eq}} & \frac{\partial y}{\partial \eta_{eq}} \end{bmatrix}, \tag{6.14}$$

where ξ_{eq} and η_{eq} are the local coordinates of referential equilateral element.

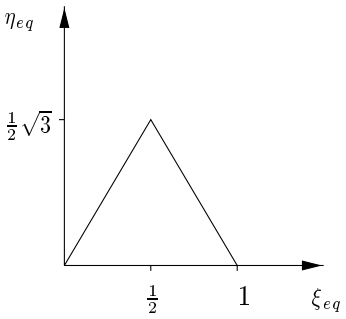


Figure 6.7: Referential equilateral element.

The following quality definition for an element i is used (see [10] and [6]):

$$Q^i = \frac{2}{|\Omega_i|} \int_{\Omega_i} \frac{\det(\mathbf{T}^i)}{\mathbf{T}^i : \mathbf{T}^i} d\Omega, \tag{6.15}$$

where $|\Omega_i|$ is the area of element i. This quality definition results in an optimum value if the element is conform to the referential element. The value for the quality is 1 for an equilateral element.
The quality of the element mesh is indicated using two values: the minimum quality and the average quality. The minimum quality of the mesh is defined as the quality of the element with the lowest quality. The minimum quality is often applied as a criterion for remeshing. In order to obtain a global assessment of the quality of the element mesh, the average quality is used. The average quality of the element mesh is defined as

$$\overline{Q} = \frac{1}{N} \sum_{i=1}^{N} Q^i, \tag{6.16}$$

where N is the total number of elements.

For the error of the evolution of the boundary we consider the area between the calculated boundary and the exact boundary on the right hand side of the domain. We take the right hand side of the domain, because at this curved part of the boundary an error will be made in the calculation of the position of the boundary here. When a boundary is curved, the movement of nodal points along the boundary is not trivial, because the direction of the tangential vector changes (see Section 4.2). The total area between the calculated and

the exact boundary is approximated by (see Figure 6.8):

$$\| A \|_{\mathrm{b}} = \frac{1}{N_{\mathrm{b}}} \sum_{i=1}^{N_{\mathrm{b}}} |dx^i \, dy^i|, \tag{6.17}$$

where N_{b} is the number of elements on the right hand side of the domain.

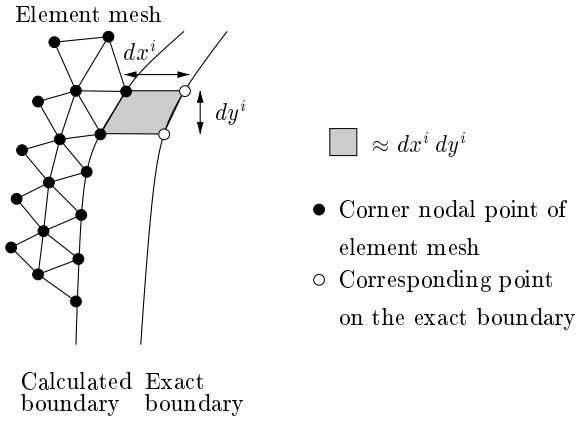


Figure 6.8: Error in the evolution of the right hand side of the domain at time t.

The equivalent strain $\bar{\varepsilon}$ is influenced by the accuracy of the calculation of the material velocity field $\mathbf{v}$ and the convective term of (3.64). For this reason the error in the equivalent strain is taken. The following error norm is applied:

$$\| \Delta\bar{\varepsilon} \|_2 \, (t) = \sqrt{\frac{\int\limits_{\Omega} (\bar{\varepsilon}(t) - \bar{\varepsilon}_{\mathrm{exact}}(t))^2 d\Omega}{\int\limits_{\Omega} (\bar{\varepsilon}_{\mathrm{exact}}(t_{\mathrm{end}})^2 d\Omega}}. \tag{6.18}$$

with $t_{\mathrm{end}} = 15\,\mathrm{s}$. The integrals of (6.18) are integrated using the finite element mesh. The exact solution $\bar{\varepsilon}_{\mathrm{exact}}$ is approximated very accurately by numerically integrating (6.13) along the trajectories. This numerical integration will be performed accurately using a standard routine from the IMSL library, using very small time steps.

6.2.2 Results

In this section we discuss the results of the updated Lagrangian and ALE calculations. For the evaluation of the results we use the definitions of the previous section.

Comparison between ALE and updated Lagrangian calculations

In Figure 6.9 the equivalent strain distribution is shown for an ALE calculation using the mesh velocity algorithm of Section 4.2.2 (called ALE(fem)) on mesh II. The gradients are steep, which means that the calculation of the remap of the equivalent strain is not trivial, since for the remap of the equivalent strain the spatial derivative is needed (see (5.3)).
The boundary condition for the mesh velocity consists of prescribing the mesh velocity $\mathbf{v}_g$ in the normal direction with the material velocity $\mathbf{v}$ (4.15). The mesh velocity is taken to be free in the tangential direction. In order to be able to investigate the accuracy of the position of the boundary it is important that the boundary is curved. It can be seen in Figure 6.9 that the right hand side of the domain is strongly curved.

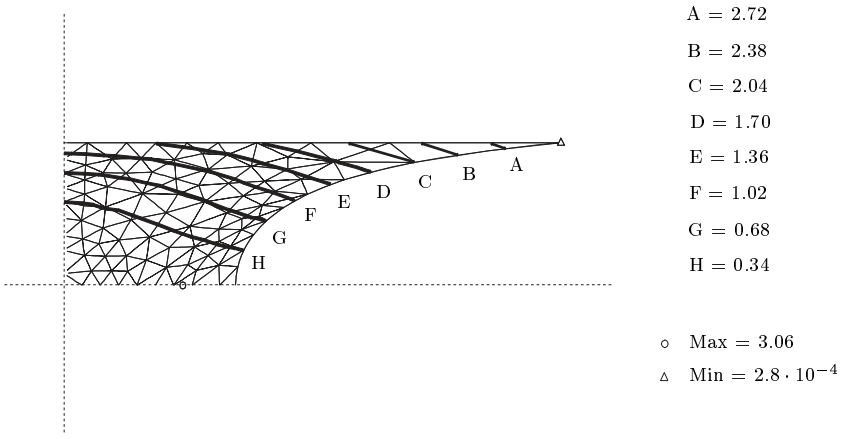
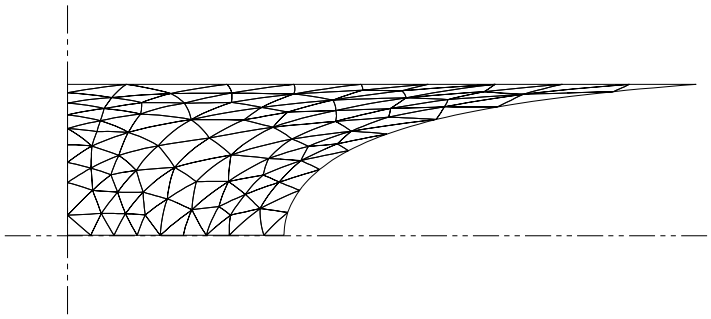
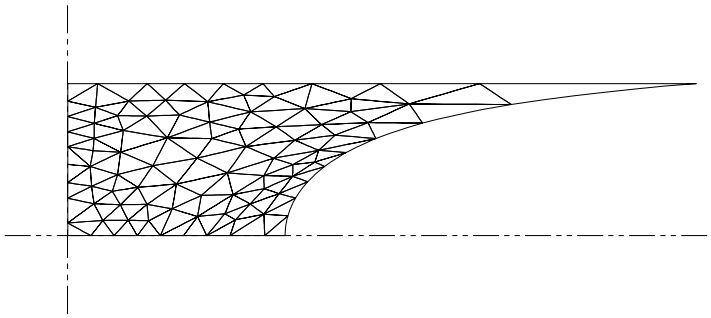


Figure 6.9: Equivalent strain distribution after an ALE calculation on mesh II.

In Figure 6.10 the element meshes at $t = t_{\text{end}}$ are shown. The ALE method is effective in reducing mesh distortion in this academic case, which was the objective of the mesh velocity algorithm. The element sides remain straight in the interior of the domain, because the midside nodal points always lie in the middle of the element sides. However, in the upper right hand corner, the region with the largest strains, large elements are present. A more refined element mesh would be desirable in that region to increase the accuracy of the calculation.
In Figure 6.11(a), it is also shown that the average quality $\overline{Q}$ of the element mesh is indeed improved by the ALE method. The initial mesh consists of nearly equilateral elements (see Figure 6.6(b)), which results in an initial quality $\overline{Q}$ of nearly 1. The ALE method gives a higher minimum quality (see Figure 6.11(b)), which is observed also in other test examples, which was



(a) Updated Lagrangian calculation



(b) ALE calculation

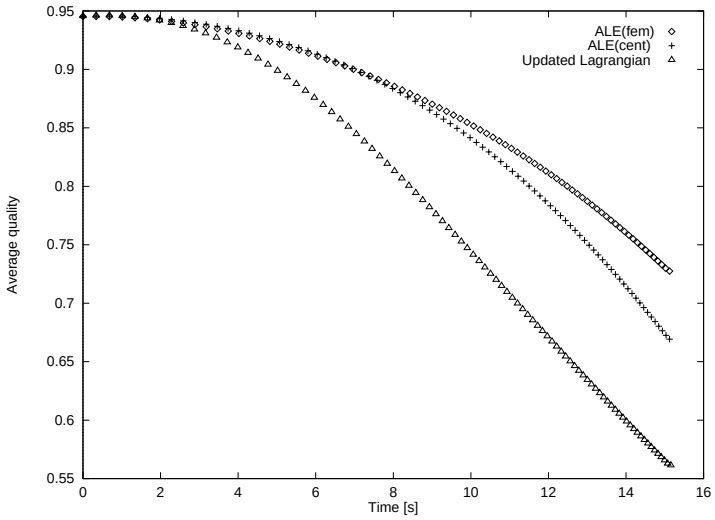
Figure 6.10: Updated Lagrangian and ALE calculation on mesh II.

to be expected as it tries to smooth gradients in the mesh velocity. The kinks in the figures indicate a change of the element with lowest quality.

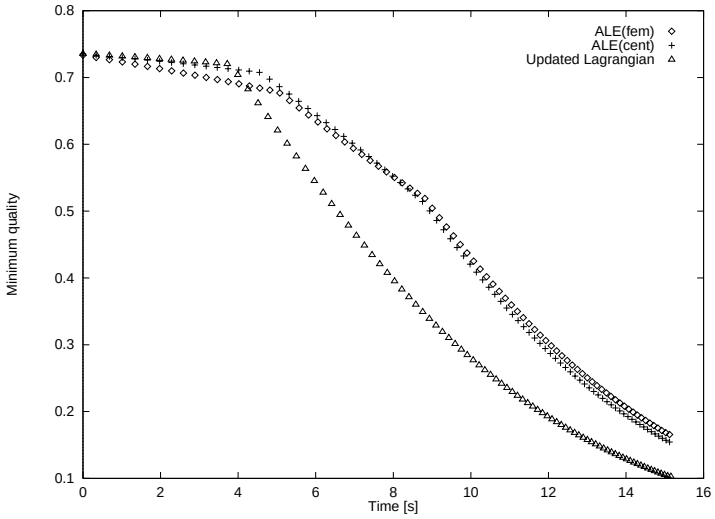
Method	mesh	$\| \Delta\bar{\varepsilon} \|_2$	$\| A \|_b$
Updated	I	$3.4 \cdot 10^{\Leftrightarrow 4}$	15.1
Lagrangian	II	$1.7 \cdot 10^{\Leftrightarrow 4}$	7.52
	III	$8.4 \cdot 10^{\Leftrightarrow 5}$	3.85
ALE(fem)	I	$2.5 \cdot 10^{\Leftrightarrow 4}$	6.96
	II	$9.3 \cdot 10^{\Leftrightarrow 5}$	3.21
	III	$5.2 \cdot 10^{\Leftrightarrow 5}$	1.49

Table 6.4: Results of the updated Lagrangian and the ALE method (with the mesh velocity method of Section 4.2.2).

In Table 6.4 the errors (6.17) and (6.18) of the ALE method (with the mesh velocity algorithm of Section 4.2.2) and the updated Lagrangian method on the three meshes are given. The convergence rates of the errors are almost the same for both methods, but the ALE method gives smaller errors than updated Lagrangian method for this case (see also Figure 6.12).
The error in the resulting boundary is small compared to the total displacements at the right side of the domain. It is important to notice that this error decreases with mesh and time step refinement for both methods. The error made in the evolution of the boundary is even smaller for the ALE method than for the updated Lagrangian method. In the updated Lagrangian calculation at each time step the exact material velocity is prescribed at the boundary, but it is taken to be constant during the time step. This results in an error in the position of the boundary. With the ALE method the mesh velocity at the boundary has to fulfill the boundary condition (4.15) and the nodal points are free to move in along the boundary. The fact that this error is even smaller with the ALE method might be explained by the fact that the ALE method has a larger number of elements where the boundary is strongly curved. It can therefore be concluded that the description of the boundary is performed accurately with the ALE method.



(a) Average quality



(b) Minimum quality

Figure 6.11: Results with the updated Lagrangian method and the ALE method.

Investigation of the mesh velocity algorithms

In Section 4.2 two algorithms based on solving the Laplace equation for the mesh velocity have been given. The ALE method using the centering method (ALE(cent)) and the ALE method using the finite element method (ALE(fem)) for the Laplace equation for the mesh velocity have been used. In this section the numerical results of these two mesh velocity algorithms are discussed. The results are obtained from calculations carried out on mesh II.
In Figure 6.11(a) the average quality $\overline{Q}$ is displayed. Clearly, both methods reduce mesh distortion compared to the updated Lagrangian method. Up to about $t = 8\,\mathrm{s}$ both ALE methods (ALE(cent) and ALE(fem)) result in almost the same average quality. From $t = 8\,\mathrm{s}$ the average quality decreases more rapidly using ALE(cent) than using ALE(fem). The centering method assumes a homogeneous distribution of nodal points, which is the case for the initial mesh with equilateral elements. However, in the case of a distorted element mesh the nodal points are not homogeneously distributed. So the centering method becomes less effective in reducing mesh distortion. It can be concluded that the mesh velocity method using the finite element method for the Laplace equation is more effective for distorted element meshes.
In our example, the two ALE calculations show an error for the equivalent strain distribution (6.18) that is smaller than in the updated Lagrangian method (see Figure 6.12). Only a small difference can be observed between the two ALE calculations. The results of the two mesh velocity algorithms do not differ greatly.

An important aspect is the efficiency of the mesh velocity calculation. Both mesh velocity schemes consist of solving the Laplace equation using an iteration method. These equations for the mesh velocity do not need to be solved exactly, as it is just the calculation of the position of the new mesh, where the solution of the simulation is obtained. The iteration process is stopped when the condition (4.32) is satisfied:

$$\frac{\sqrt{\max(\Delta_k \mathbf{V}_g \cdot \Delta_k \mathbf{V}_g)}}{\sqrt{\max(\mathbf{V} \cdot \mathbf{V})}} < C_m, \tag{6.19}$$

where C_m is a chosen threshold.
In the Figures 6.13(a) and (b) the influence of the convergence threshold C_m on the minimum and average qualities is shown. One can see that the convergence threshold C_m has a small influence on the quality of the element mesh. Also, negligible differences in the error of the equivalent strain distribution are obtained in the calculations for various C_m (see Figure 6.14(a)). However, the number of iterations differs considerably for various C_m (see Figure 6.14(b)).

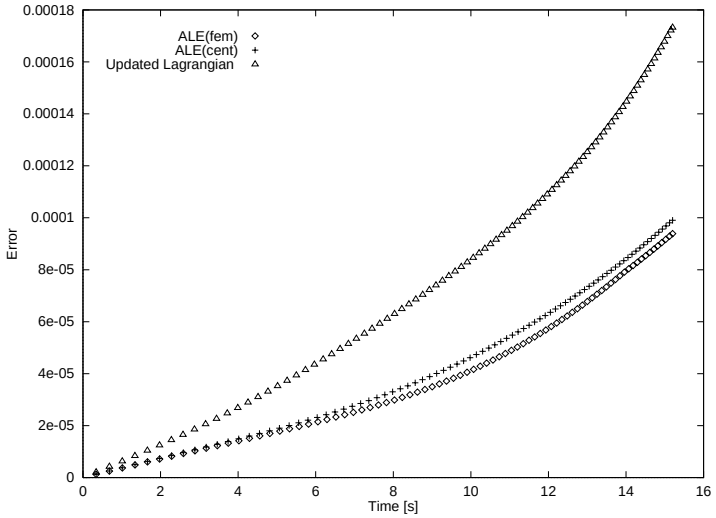
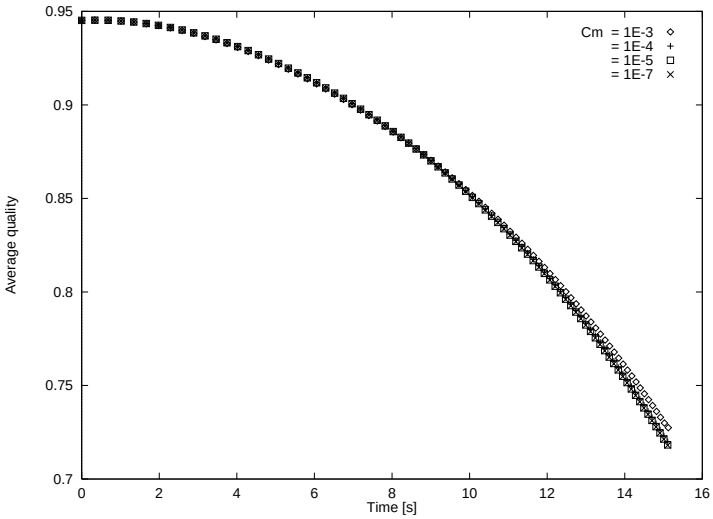
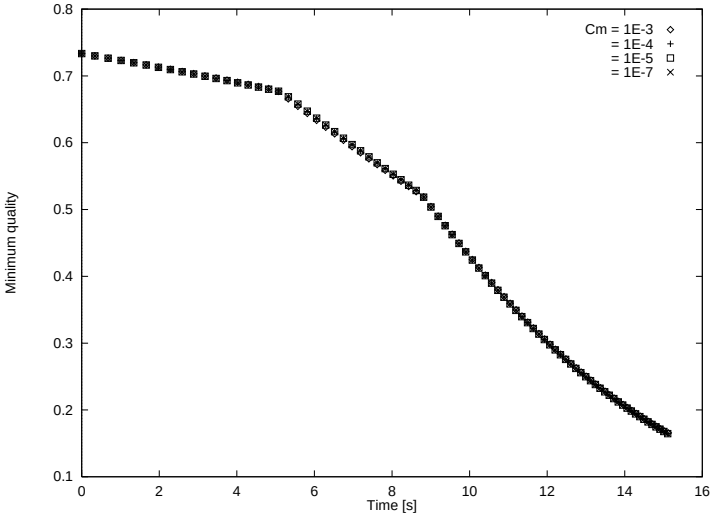


Figure 6.12: The error $\| \Delta\bar{\varepsilon} \|_2$ (6.18) with the two ALE methods and the updated Lagrangian method.

For $C_m = 1 \cdot 10^{\Leftrightarrow 3}$ less than 10 iterations are necessary whereas for $C_m = 1 \cdot 10^{\Leftrightarrow 7}$ more than 150 iteration are needed. As the results of the calculations are almost the same for various C_m, one can state that the first iterations are the most effective ones for the calculation of the mesh velocity. It can be concluded that relatively large values of C_m ($\sim 10^{\Leftrightarrow 3}$) are sufficient in order to reduce mesh distortion and that therefore the number of iterations can be kept small. One can also see that in the first increment of a calculation more iterations are necessary to fulfill the convergence condition. For the subsequent increments the solution for the mesh velocity of the last increment is used as the initial value. Since this is often a good initial guess for the iteration method, the number of iterations required to fulfill the convergence condition is much smaller in consecutive increments. The same tendency was observed with the centering method, but the results are omitted here.

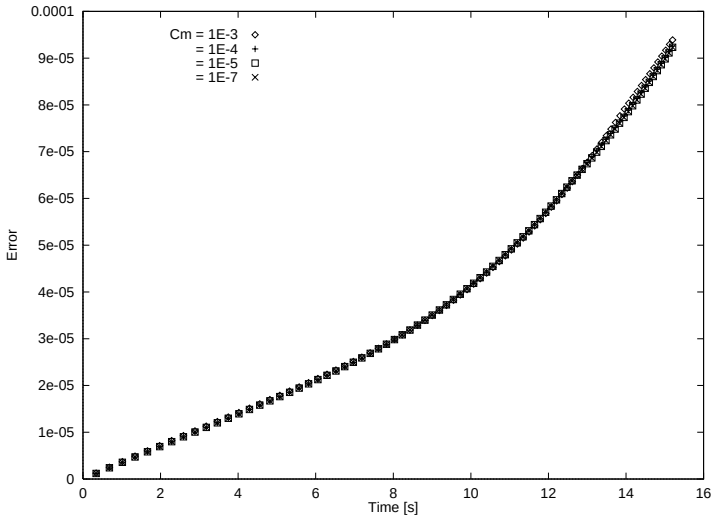


(a) Average quality

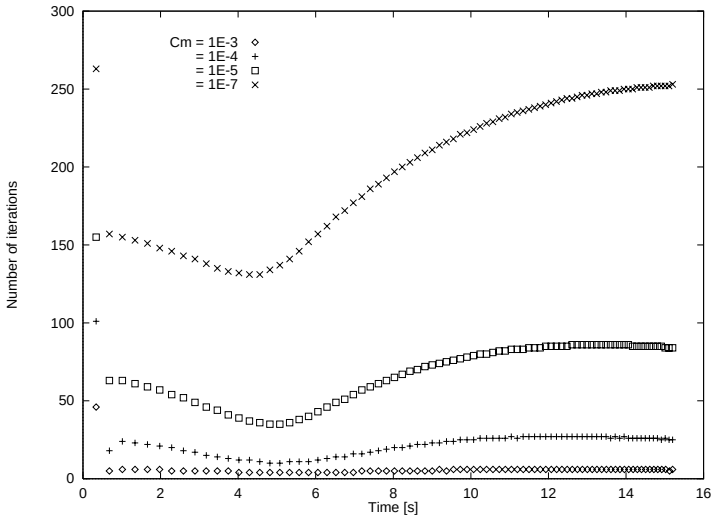


(b) Minimum quality

Figure 6.13: Influence of the convergence limit for ALE(fem).



(a) Error



(b) Number of iterations

Figure 6.14: Influence of the convergence limit C_m for ALE(fem).

6.3 Applications to forming processes

In the previous two sections the ALE method was investigated in two academic test cases. In the present section the applicability of the ALE method in simulations of forming processes is discussed.
In Section 6.3.1 a punch indentation problem is simulated with the updated Lagrangian method and the ALE method. Next, an upsetting process is simulated in Section 6.3.2. Subsequently, in Section 6.3.3 both methods are applied in the simulation of a forging process. Finally, the extrusion process is simulated, which is only performed using the ALE method (Section 6.3.4).

6.3.1 Punch indentation problem

In Figure 6.15 the plane strain punch indentation problem simulated, which is discussed in this section, is shown. A rigid cylindrical punch is indented into a block of material. At the left and right hand side of the domain the material is free to move.

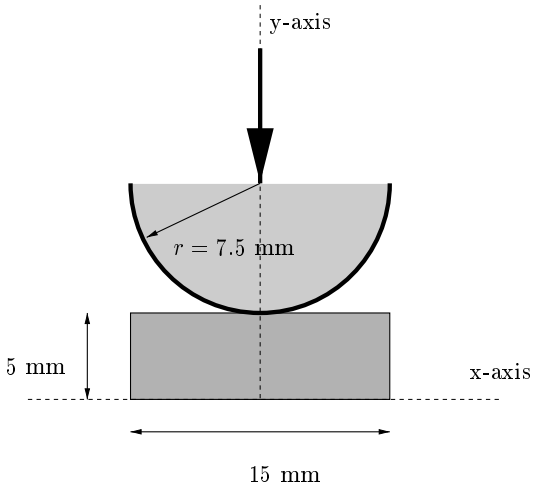


Figure 6.15: Punch indentation problem.

A material is assumed to be described by the elastoplastic model (2.50) with Young's modulus $E = 2 \cdot 10^5\,\mathrm{N/mm^2}$ and Poisson ratio $\nu = 0.3$. The hardening effect is described by the yield stress $\bar{\sigma}_y$ with the following Voce model:

$$\bar{\sigma}_y(\bar{\varepsilon}^p) = 250 + 100\left(1 - \exp(\frac{\bar{\varepsilon}^p}{0.1})\right). \tag{6.20}$$

The contact behaviour between the material and the cylinder is described by the Coulomb friction model with a friction coefficient of $\mu = 0.4$. In the

simulations we make use of the symmetry of the x-axis and the y-axis.
The simulations are carried out using both the updated Lagrangian method and the ALE method with quadrilateral elements, in which the punch is moved 4 mm downwards (80% reduction) in 400 increments.
The mesh calculation in the ALE simulations is performed using the transfinite mapping method of Section 4.1. The element mesh consists of one region. At each time step the nodal points on the four boundaries are moved such that the nodal points are equidistantially distributed over the boundaries.

In Figure 6.16 the element mesh of the updated Lagrangian calculation is shown. The element mesh becomes strongly distorted in this case, which results in inaccurate results or even a crash of the calculation.
In Figure 6.17 the element mesh of an ALE calculation is shown. Clearly, the element mesh remains more regular using the transfinite mapping method.

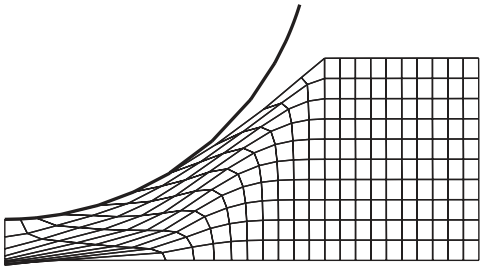


Figure 6.16: Element mesh after the updated Lagrangian calculation.

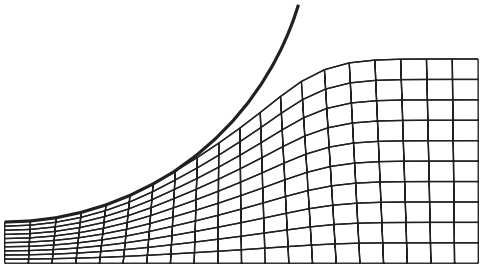


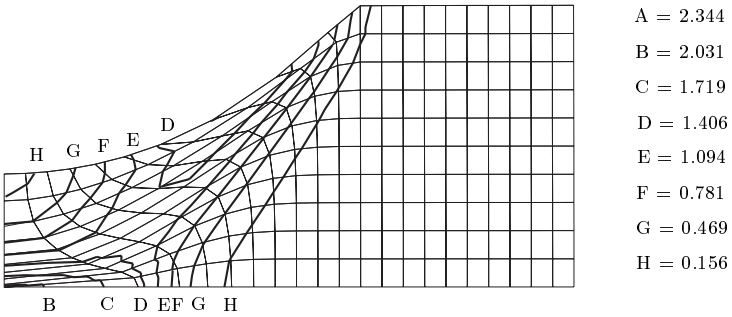
Figure 6.17: Element mesh after an ALE calculation.

It can be concluded that the transfinite mapping method was effective in reducing mesh distortion for this process. However, one can see that in the updated Lagrangian calculation a sharp edge exists on the upper surface of the material (see Figure 6.16). In the ALE calculation the edge is more rounded (see Figure 6.17) due to the movement of nodal points along the

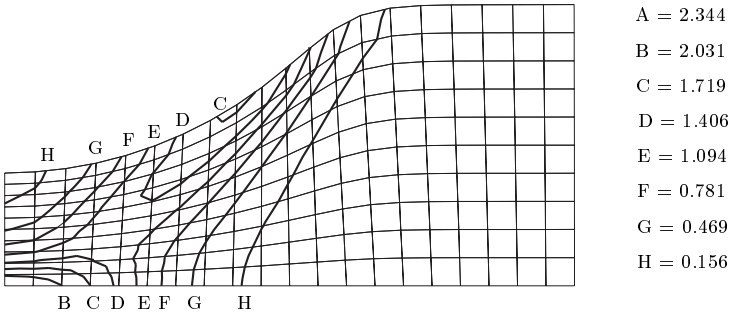
boundary. The updated Lagrangian method describes the boundary more accurately than the ALE method. The description of sharp curvatures of the boundary can be improved by changing the weight function w (4.4) in the transfinite mapping method for the redistribution of the nodal points on the boundary. The curvature of the boundary can be taken into account in the weight function w, which results then in smaller elements close to strong curvatures of the boundary (see [46]).

In the case of ALE calculations with an elastoplastic material model, the components of the Cauchy stress $\boldsymbol{\sigma}$ and the equivalent plastic strain $\bar{\varepsilon}^p$ are state variables, which have to be updated with the grid time derivative (3.63). For the convective term of the grid time derivative we have used several methods: the first order upwind scheme, the $\kappa = \frac{1}{3}$-scheme and the Van Leer scheme of Chapter 5. In order to demonstrate the effect of these methods the equivalent plastic strain distributions of the different schemes are given in the Figures 6.18 and 6.19. The results are shown after a punch displacement of 3 mm, as reasonable results of the updated Lagrangian calculation cannot be shown after a punch displacement of 4 mm.

Clearly, the result of the updated Lagrangian calculation differs strongly from the results of the ALE calculations. In the updated Lagrangian method the elements become strongly stretched in the region with large strains. This is a general characteristic of the updated Lagrangian method, *i.e.* elements become distorted or stretched in regions with large strains. Especially in these regions one would like to describe the solution more accurately. In the ALE calculations the elements are kept smaller in regions with large strains. As a result localisation phenomena can be described more accurately with an ALE method, as the elements can then be refined in these areas.
The differences between ALE calculations that use different schemes for the convective terms for the remap of the state variables, are small. This can be explained by the fact that the convective term due to the convective velocity $\mathbf{v}_c$ is small in these calculations: the mesh velocity $\mathbf{v}_g$ almost equals the material velocity $\mathbf{v}$. Under the punch, where the element lengths in the updated Lagrangian calculations differ from those in the ALE calculations, differences occur between the mesh velocity $\mathbf{v}_g$ and the material velocity $\mathbf{v}$ and gradients in the equivalent strain exist there. As a consequence the differences in the equivalent strain distribution are largest there. The convective velocity is relatively small in this calculation. So the choice of the remap algorithm is not so important in these 'almost' updated Lagrangian calculations ($\mathbf{v}_g \approx \mathbf{v}$). Nevertheless, the ALE calculations improve results as a refined element mesh can be applied in regions with large strains.

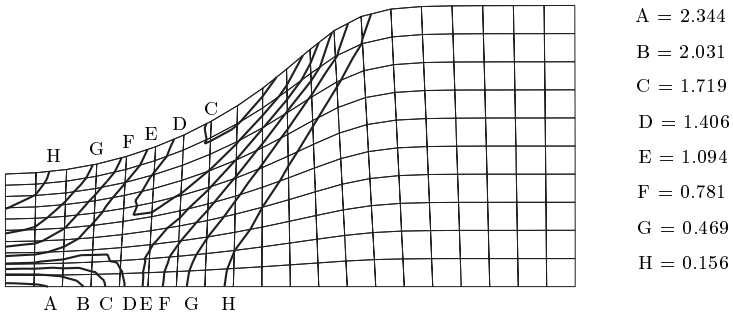


(a) Updated Lagrangian

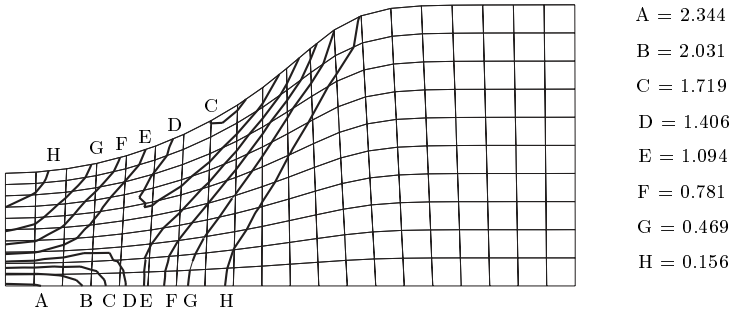


(b) ALE with first order upwind scheme

Figure 6.18: Equivalent plastic strain distribution after a punch displacement of 3 mm.



(a) ALE with $\kappa = \frac{1}{3}$-scheme



(b) ALE with Van Leer scheme

Figure 6.19: Equivalent plastic strain distribution after a punch displacement of 3 mm.

6.3.2 Upsetting problem

In the simulation of the punch indentation problem the transfinite mapping method is effective in preventing mesh distortion. However, the transfinite mapping method is not always effective in reducing mesh distortion. In order to demonstrate this, we performed the upsetting simulation of Figure 6.20. In this case a flat rigid tool is pushed onto a block of material. The material is free to move at the right hand side. The contact behaviour between the punch and the material is assumed to be described by the Coulomb friction model, where the friction coefficient $\mu = 0.1$.

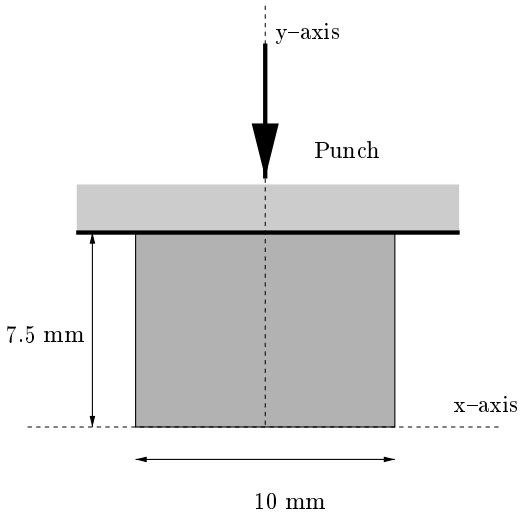
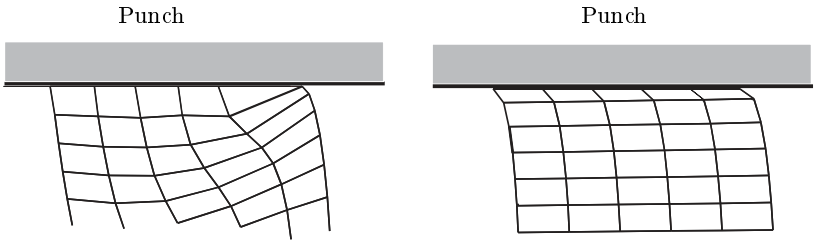


Figure 6.20: Initial situation.

The element mesh again consists of one region. At each time step the nodal points are placed equidistantially over the boundaries of the domain with the transfinite element method in the same manner as in the punch indentation problem.
In Figure 6.21 a zoom-in view of the upper right corner of the updated Lagrangian element mesh and the ALE element mesh. In both calculations the nodal points on the free surface move upwards due to the friction with the punch. When this nodal point comes in contact with the tool the corresponding element becomes too distorted to continue the calculation with this element mesh. As a result a remeshing step with a new mesh topology is necessary in order to be able to continue the calculation in both cases. However, the mesh



(a) Updated Lagrangian method (b) ALE method

Figure 6.21: Zoom-in view of the upper right corner of the element mesh in the upsetting simulation.

is kept more regular in the ALE calculation than in the updated Lagrangian calculation. So a remeshing step is delayed with the ALE method, but in the end the nodal points of the free surfaces come in contact with the tool and the quadrilateral elements in the corner also become 'too distorted'. So in this case the transfinite mapping method is not capable of effectively preventing mesh distortion.

6.3.3 Forging process: ALE method with remeshings

In this section a forging process is simulated of which the initial situation is shown in Figure 6.22.
The upper die moves 25 mm downwards with a velocity $\mathbf{v}_{\text{tool}} = 100\,\text{mm/s}$ and deforms a block of material with a height of 70 mm and a width of 50 mm. The material is assumed to be described by the Norton–Hoff model (2.33) with the rate sensitivity index $m = 0.15$, $K_0 = 9.7 \cdot 10^7\,\text{N/mm}^2$ and a strain hardening index $n = 0$ (see (2.35)). Between the material and the tool we assume a sticking contact condition, which means that the material velocity $\mathbf{v}$ of nodal points in contact equals the velocity of the tool $\mathbf{v}_{\text{tool}}$.
The updated Lagrangian method and the ALE method of the numerical software package FORGE2® are applied in the simulation of this forging process. The domain is discretised with quadratic triangular elements. The simulations were carried out with the use of complete remeshings with a new mesh topology in case the element mesh becomes 'too distorted'. An element mesh is said to be 'too distorted', when the curvature of element side is too large, or the midside node is too far from the middle of the element side or when the quality

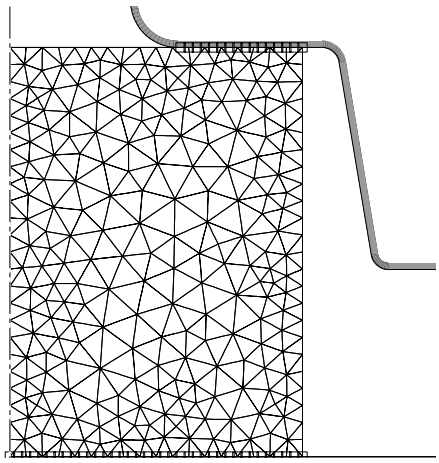


Figure 6.22: Initial situation of the forging simulation.

of an element (6.15) is too low. The complete remeshing is based on a Delaunay triangulation. For more details on the remeshing and the remeshing criteria the reader is referred to [9] and [54]. In the ALE simulations, the centering method of Section 4.2.1 is applied for the mesh velocity calculation.

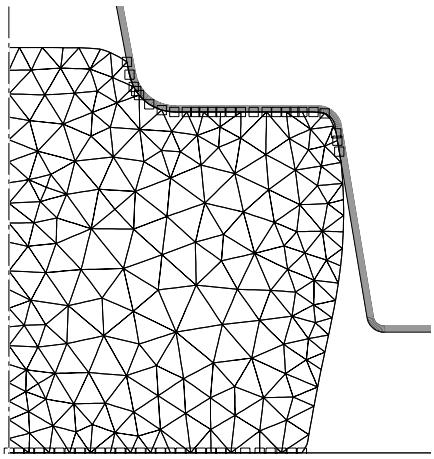


Figure 6.23: Element mesh after a tool displacement of 11 mm.

In general it is seen that the number of required complete remeshings for a new element mesh topology is reduced with the ALE method when compared to the calculations with the updated Lagrangian method. However, even in

the ALE method, remeshings are necessary in this simulation. This forging process cannot be calculated with one element mesh topology for the complete simulation, as the shape of the domain changes considerably. In the ALE calculation nine complete remeshings were necessary. In Figure 6.23 the element mesh is shown after a die displacement of 11 mm.

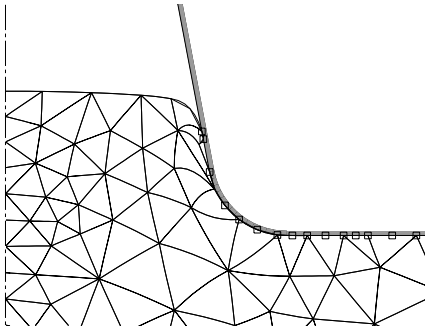


Figure 6.24: Zoom-in view of the element mesh with the updated Lagrangian simulation.

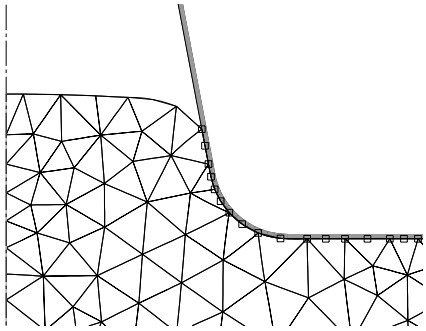


Figure 6.25: Zoom-in view of the element mesh with the ALE simulation.

From Figures 6.24 and 6.25, can be inferred that the mesh becomes more distorted in the case of the updated Lagrangian calculation than with the ALE calculation. In the updated Lagrangian calculation the distorted quadratic elements lead to less accurate results, and also a wrinkle in the surface of the material can be seen (see Figure 6.24). In the ALE method the number of complete remeshings is reduced, which leads to more accurate results, because in general the remap of state variables due to a complete remeshing is performed with a significant loss of accuracy [7]. Moreover, in the ALE calculation the internal element sides are kept straight (see Section 4.2.1). Quadratic elements

become less accurate in the case of strongly curved element sides.

To conclude, one can state that in the ALE method the mesh distortion is reduced in this forging case. This results in a smaller number of complete remeshings, which is computationally more efficient and without the extra loss in accuracy due to the remap of state variables related to a complete remeshing. So a reduction of the number of complete remeshings with the ALE method leads to more accurate results. Moreover, the quadratic triangular elements have a better shape in the ALE calculation (see also Section 6.2). However, complete remeshings remain necessary in the ALE calculation of this forging case.

6.3.4 Extrusion

In the extrusion process a bulk of material is forced through a narrow hole with the shape of the cross section of the profile. In this section the relatively simple plane strain extrusion process of Figure 6.26 is simulated. Our aim is to see whether the ALE method is capable of dealing with this type of simulation. The material is assumed to be described by the rigid plastic model (2.42). The hardening is described by the following Nadai model:

$$\bar{\sigma}_y(\bar{\varepsilon}^p) = C(\bar{\varepsilon}^p)^n \tag{6.21}$$

where $C = 565\,\mathrm{N/mm^2}$ and $n = 0.259$.
Frictionless contact behaviour between the material and the die is assumed.
The simulation is carried out with the ALE method. The updated Lagrangian method cannot be employed, as it turns out that the elements become strongly distorted in the outflow region (see also Figure 1.1).
In the ALE calculations of extrusion the mesh is kept fixed in space, so the mesh velocity $\mathbf{v}_\mathrm{g} = 0$ (see (4.1)). So the convective velocity $\mathbf{v}_\mathrm{c}$ equals the material velocity $\mathbf{v}$ in this 'Eulerian' simulation. In the description of the remap problem of state variables (5.4) in Chapter 5, it is stated that $\mathbf{v}_\mathrm{c} \cdot \mathbf{n} = 0$ on the boundary. This means that the mesh velocity $\mathbf{v}_\mathrm{g}$ and the material velocity $\mathbf{v}$ are equal in the normal direction of the boundary. However, in this extrusion case this condition is no longer valid on the inflow boundary and the outflow boundary of the domain. On the lower boundary we need an inflow condition. In the simulation we applied the condition $(\nabla\zeta) \cdot \mathbf{n} = 0$, which can be employed when $\dot{\zeta}$ is negligibly small at the inflow region. The results will validate the assumption of this boundary condition.

In this extrusion case the equivalent strain is the state variable which has to be updated with the grid time derivative, as we apply the rigid plastic model. In

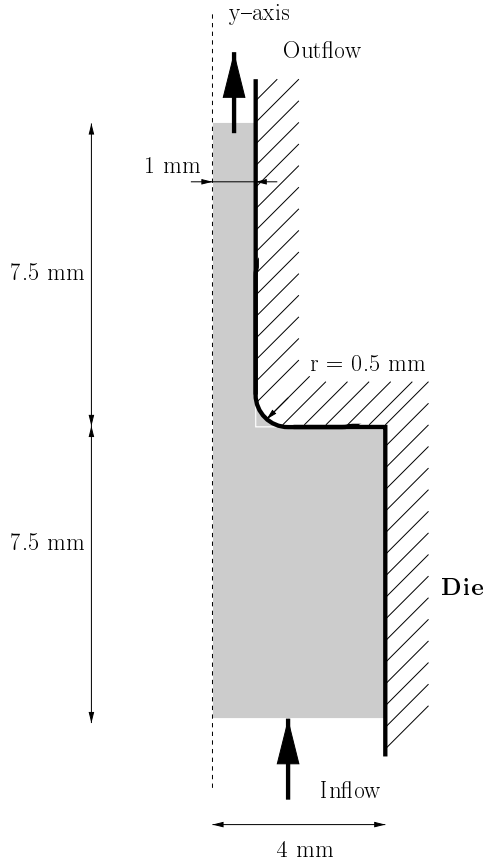
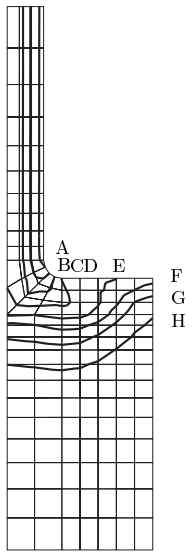


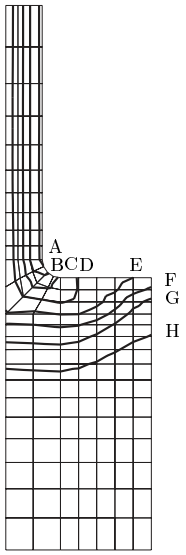
Figure 6.26: Extrusion process.

Figure 6.27 the equivalent strain distributions of ALE calculations with various remap schemes of Chapter 5 are shown. The inflow condition $(\nabla\bar{\varepsilon})\cdot\mathbf{n}=0$ can be applied when the inflow boundary is 'far' enough from the deformation zone of the outflow region. One can see that the first order upwind scheme tends to smooth the gradient of the equivalent strain over the thickness in the outflow region and the maximum equivalent strain is reduced. The differences in the equivalent strain distributions are due to the choice of the scheme applied for the convective term for the updating of the equivalent strain.
In the outflow region, the isolines should be straight in the stationary case, as no deformation takes place there. The equivalent strain distributions remain unchanged in the outflow region in all the ALE calculations. This is a result of the fact that the applied schemes reduce to one-dimensional schemes when the convection velocity is aligned along the mesh. In Figure 6.27 it can be seen

that the streamlines of the material velocity are aligned along the mesh in the outflow region. As the convective velocity $\mathbf{v}_c$ equals the material velocity in this extrusion case ($\mathbf{v}_g = \mathbf{0}$), the convective velocity is also mesh aligned.
In the deformation zone the streamlines are also approximately grid aligned. As a result the schemes of Chapter 5 behave almost optimally in these cases. They show hardly any crosswind diffusion across the width. In the simulation of the extrusion case with various schemes for the remap of the equivalent strain, the differences are not as large as in the Molenkamp test case of Section 6.1. Several reasons can be suggested to support this fact. The Molenkamp test is a pure convection test case with steep gradients. In the extrusion the gradients are usually less steep, as the material behaviour tends to smooth out gradients of the equivalent strain. Another reason is that the convective velocity in the extrusion case was more or less along the mesh direction. The above schemes behave optimally in that case.
The extrusion process was also carried out with quadratic triangular elements. The Van Leer scheme of Section 5.3.2 and the first order upwind scheme were applied for the remap of the equivalent strain. In Figure 6.28 the equivalent strain distribution is shown of both calculations.
The convective velocity is not grid aligned in these simulations. Nevertheless, the isolines of the equivalent strain distribution remain almost straight in the outflow region in the case of the Van Leer based scheme. Again one can see that the gradients are smoothed across the width with the first order upwind scheme.

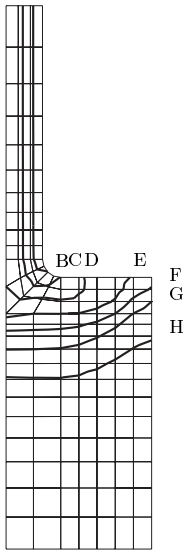


(a) Van Leer scheme

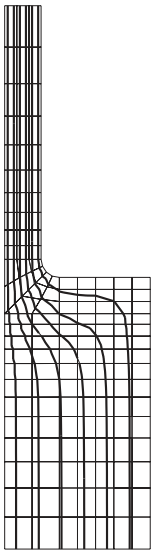


(b) $\kappa = \frac{1}{3}$-scheme

A = 3.047
B = 2.641
C = 2.234
D = 1.828
E = 1.422
F = 1.016
G = 0.609
H = 0.203

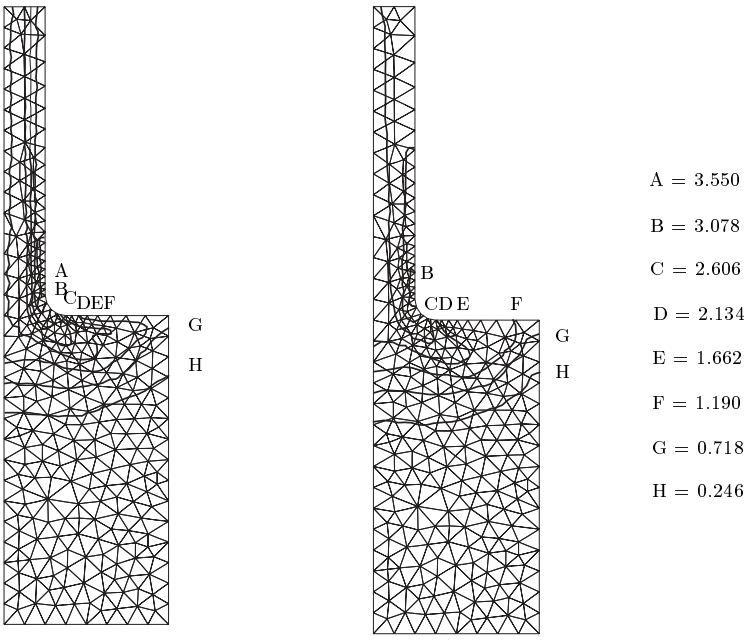


(c) First order upwind scheme



(d) Streamlines

Figure 6.27: Equivalent plastic strain distribution for various schemes and the streamline distribution.



(a) Van Leer scheme

(b) First order upwind scheme

Figure 6.28: Equivalent plastic strain distribution using quadratic triangular elements.

6.4 Conclusion

In the simulation of forming processes with various schemes for the remap of the equivalent strain, the differences in the results are not as large as in the Molenkamp test case of Section 6.1. Several reasons can be suggested for this fact. The Molenkamp test is a pure convection test case with steep gradients. In the simulations of forming processes gradients are generally less steep, as gradients in the equivalent strain are smoothed due to material behaviour. In forming processes the convective terms can also be small in so-called 'almost' updated Lagrangian calculations ($\mathbf{v}_g \approx \mathbf{v}$). The calculation of the convective term for the updating of state variables is not crucial in these 'almost' updated Lagrangian calculations. The ALE method is more suitable than the updated Lagrangian method in the punch indentation problem. The most important aspect of the ALE method was that, with the ALE method, we can obtain refined meshes in regions with steep gradients (see the punch indentation problem of Section 6.3.1). Moreover, the ALE method could be applied with the help of an error estimator to control the mesh velocities (see *e.g.* [24]).
In calculations with large time steps, oscillations can occur due to the convective terms. However, in ALE calculations the time step is usually restricted by other parts of the calculation, for example, by the contact behaviour or by the material behaviour.
When the meshes are strongly distorted the schemes for quadrilateral elements become less accurate as they assume two local orthogonal directions. In that case these schemes become less accurate. The Van Leer based scheme extended for triangular elements of Section 5.3.2 can also be worked out for quadrilateral elements, which leads to better results in the case of unstructured or distorted meshes.
It can be concluded from the simulation of the punch indentation problem and the upsetting problem that the transfinite mapping method is effective in reducing the mesh distortion in specific cases.
The Laplace schemes for the mesh velocity calculation tends to prevent mesh distortion in the second academic test case and the forging example. However, even these schemes cannot prevent the need for a complete remeshing with a new mesh topology. An important aspect of the ALE method is that the number of complete remeshings can be reduced, which usually results in more accurate results, but they cannot be prevented entirely.

Chapter 7

Conclusions & recommendations

The general conclusions are:

- The Molenkamp test is a good test case for the numerical behaviour of convection schemes. In ALE calculations the influence of the convection schemes for the remap of state variables is generally less important.
- The ALE method can be useful in reducing the mesh distortion and as a result in reducing the number of necessary complete remeshings. The application of the ALE method in combination with complete remeshings is preferred.
- The application of the ALE method is interesting in the case of quadratic elements, as the elements preserve a better shape, which is important for the accuracy of these quadratic elements.
- For the meshing in the ALE method, the Laplacian based methods are preferred to the transfinite mapping method, as the transfinite mapping method puts restrictions on the mesh topology.

The general recommendations are:

- The meshing methods in the ALE method should be improved. These methods should be more effective in reducing the mesh distortion and they should also employ error estimators.
- The Van Leer based scheme for triangular elements can be worked out for quadrilateral element in order to improve the accuracy in the case of unstructured or distorted quadrilateral element meshes.
- As complete remeshings cannot be prevented, these remeshings and the corresponding remap of state variables should be the subject of further research in order to improve the results.

List of symbols

Scalars

C	Courant number
C_b	bulk modulus
C_C	penalty coefficient for contact
C_m	convergence limit for the Laplace schemes for the mesh velocity definition
f	flux
G	shear modulus
g	gap function in contact
$h(\mathbf{v})$	non-penetration contact condition
I	potential function
J	Jacobian
K	consistency parameter in the Norton–Hoff model
N	finite element interpolation function
M	finite element interpolation function
m	strain rate sensitivity index
n	hardening parameter
p	hydrostatic pressure
Q	quality of an element
r	ratio of consecutive gradients
t	time
s	length
w	weight function in the transfinite mapping method
ζ	an arbitrary state variable
η	shear coefficient
κ	κ finite volume scheme
λ	Lagrange multiplier
μ	friction coefficient
ν	Poisson ratio
ρ	density

ρ_p	penalty coefficient for incompressibility
ϕ	yield function or limiter function
$\bar{\sigma}$	equivalent stress
σ_y	yield stress
σ_n	contact stress in normal direction
ε	logarithmic strain
$\bar{\varepsilon}$	equivalent strain

Vectors

$\mathbf{f}$	body force
$\mathbf{F}$	nodal force
$\mathbf{n}$	normal vector
$\mathbf{p}$	stress vector
$\mathbf{Q}_M$	discretisation of ∂W_{int}
$\mathbf{t}$	tangential vector
$\mathbf{T}$	prescribed stress
$\mathbf{u}$	material displacement
$\mathbf{u}_{\text{c}}$	convective displacement
$\mathbf{u}_{\text{g}}$	mesh displacement
$\mathbf{v}$	material velocity
$\mathbf{v}_{\text{c}}$	convective velocity
$\mathbf{v}_{\text{g}}$	mesh velocity
$\mathbf{v}_{\text{tool}}$	tool velocity
$\mathbf{v}^*$	virtual velocity
$\mathbf{V}$	nodal velocity vector
$\mathbf{x}$	spatial coordinate
$\mathbf{X}$	material coordinate (also the initial coordinate)
$\boldsymbol{\tau}$	tangential stress vector on boundary , $_C$
$\boldsymbol{\xi}$	local coordinates
$\boldsymbol{\chi}$	referential coordinates
$\boldsymbol{\psi}$	coordinates of a discretised curve

Second order tensors

$\mathbf{B}$	left Cauchy–Green
$\mathbf{C}$	right Cauchy–Green
$\mathbf{D}$	rate of deformation

$\mathbf{e}$	Euler–Almansi strain
$\mathbf{E}$	Green–Lagrange strain
$\mathbf{F}$	deformation gradient
$\mathbf{K}$	stiffness matrix
$\mathbf{L}$	velocity gradient
$\mathbf{R}$	rotation tensor
$\mathbf{U}$	right stretch tensor
$\mathbf{V}$	left stretch tensor
$\mathbf{W}$	spin tensor
$\boldsymbol{\varepsilon}$	linear strain tensor
$\boldsymbol{\sigma}$	Cauchy stress
$\mathbf{s}$	deviatoric stress
$\mathbf{1}$	second order identity tensor

Fourth order tensors

$\mathbb{E}$	elasticity tensor
$\mathbb{I}$	fourth order identity tensor
$\mathbb{Y}$	yield tensor
$\mathbb{L}^S$	constitutive tensor for the deviatoric stress

Operators

$\mathbf{A}^{\Leftrightarrow 1}$	inverse of tensor $\mathbf{A}$
$\mathbf{A}^T$	transpose of tensor $\mathbf{A}$
$\det(\mathbf{A})$	determinant of tensor $\mathbf{A}$
$\mathrm{tr}(\mathbf{A})$	trace of tensor $\mathbf{A}$
$\overset{\circ}{\mathbf{A}}$	Jaumann or corotational time derivative
$\dot{\zeta}$	material time derivative of a state variable ζ
$\frac{\partial_{\mathrm{g}}\zeta}{\partial t}$	grid time derivative with constant referential coordinates
$\overleftarrow{\nabla}$	post-gradient to the spatial coordinates
$\overrightarrow{\nabla}$	pre-gradient to the spatial coordinates
$(\zeta)_{,x}$	gradient of ζ to x

Subscripts

0	with respect to the initial coordinates
c	contact or convective
g	with respect to the referential (or grid) coordinates
ref	the reference situation of the beginning of the time step
e	per element
l	updated Lagrangian situation
m	with respect to the material coordinates

Superscripts

d	deviatoric part
e	elastic part
p	plastic part
$'$	situation at the end of the time step

Others

Ω	domain
$,$	boundary of domain
$, _C$	part of boundary in contact with tool
$, _v$	part of boundary with prescribed velocity
$, _T$	part of boundary with prescribed stresses
$\mathcal{P}$	mechanical problem
$\mathcal{P}_h$	discretised mechanical problem
$\mathcal{R}_h$	solution of $\mathcal{P}_h$
$\mathcal{V}^*$	collection of kinematically admissible velocities
$\mathcal{V}^*_C$	collection of kinematically admissible velocities fulfilling the non-penetration contact condition
$\mathcal{V}^*_{\text{div}}$	collection of kinematically admissible velocities fulfilling the incompressibility condition

Acknowledgements

First of all, I would like to thank Han Huétink for giving me the opportunity to carry out this interesting research. His enthusiasm and knowledge of applied mechanics are admirable. I highly appreciate that I was allowed to stay one year at CEMEF in France. I would also like to thank Jean-Loup Chenot for giving me this opportunity. I am grateful for the fruitful cooperation and the valuable discussions. I would also like to thank my supervisor at CEMEF, Francois Bay, for his continuous support and help, even after I had already left the 'Cote d'Azur'.

In the Dieka group I have experienced a very pleasant atmosphere. Among them, I thank my roommates in these years Gerrit Rekers, Eisso Atzema, Antoine Labrosse. I wish to thank my roommates, Ruth van de Moesdijk and Dejana Djokovic, and Bert Rietman and Bert Geijselaers for carefully reading the manuscript. Ruud van Damme, who helped me a lot with mathematics and the writing of my thesis, is especially acknowledged. Remko Akkerman introduced the ALE method to me in the beginning of my Ph.D. period. I thank Marcel van Haaren for his contribution to this work. Thanks to Debbie Vrieze and Annemarie Teunissen for the administrative support and Katrina Emmett for the corrections of the English language.

I would like to thank my 'colocataires' Mike van Iseghem, Roland Logé, Etienne Perchat, Silvano Dal Zilio and Massimo Merro for making my stay in France very comfortable. I thank Christiane Fourment for the cooperation and the hard, but pleasant mountain bike journeys.

I would also like to thank all my friends for reminding me of the life besides my work. My paranymfs, Menno Oppedijk and Kees Buikema, were a great support to me.
Last but not least I would to thank my family for their support, understanding and encouragement.

Christiaan Stoker.

Bibliography

[1] Akkerman, R. *Euler–Lagrange simulations of nonisothermal viscoelastic flows.* PhD thesis, University of Twente, 1993.

[2] Baaijens, F. An U–ALE formulation of 3–D unsteady viscoelastic flow. *International Journal of Numerical Methods in Engineering 36(7)* (1993), 1115–1143.

[3] Baaijens, F. Numerical experiments with a discontinuous Galerkin method including monotonicity enforcement on the stick-slip problem. *Journal of Non-Newtonian Fluid Mechanics 51* (1994), 141–159.

[4] Bayoumi, H., Gadala, M., and Wang, J. Numerical simulation of metal forming processes. In *Simulation of Materials Processing : Theory, Methods and Applications* (Enschede, The Netherlands, 22–25 June 1998), J. Huétink and F. Baaijens, Eds., Numiform '98, A.A. Balkema, Rotterdam, pp. 103–108.

[5] Benson, D. An efficient, accurate, simple ALE method for nonlinear finite element programs. *Computer Methods in Applied Mechanics and Engineering 72* (1989), 305–350.

[6] Brackbill, J., and Saltzman, J. Adaptive zoning for singular problems in two dimensions. *Journal of Computational Physics 46* (1982), 342–368.

[7] Chenot, J.-L., and Bellet, M. The ALE–method for the numerical simulation of material forming processes. In *Simulation of Materials Processing: Theory, Methods and Applications* (Ithaca, USA, 18–21 june 1995), S.-F. Shen and P. Dawson, Eds., Numiform '95, A.A. Balkema, Rotterdam, pp. 39–48.

[8] Chenot, J.-L., and Fourment, L. Numerical formulations and algorithms for solving contact problems in metal forming simulation. In *Computational Mechanics: New Trends and Applications* (Buenos Aires,

Argentina, 29 June–2 July 1998), S. Idelsohn, E. Oñate, and E. Dvorkin, Eds., CIMNE, Barcelona, Spain.

[9] Coupez, T. *Grandes transformations et remaillage automatique.* PhD thesis, Ecole Nationale Supérieure des Mines de Paris, 1991.

[10] Demkowicz, L., and Oden, J. On a mesh optimization method based on a minimization of interpolation error. *International Journal of Engineering Science 24*, 1 (1986), 55–68.

[11] Donea, J. A Taylor–Galerkin method for convective transport problems. *International Journal of Numerical Methods in Engineering 20* (1984), 101–119.

[12] Donea, J. Generalized Galerkin methods for convection dominated transport phenomena. *Applied Mechanics Reviews 44*, 5 (1991), 205–214.

[13] Donea, J., Fasoli-Stella, P., and Giuliani, S. Lagrangian and Eulerian finite element techniques for transient fluid–structure interaction problems. In *Transaction of the 4^{th} Conference* (San Francisco, 1977), p. Paper B1/2.

[14] Engelman, M., Sani, R., and Gresho, P. The implementation of normal and/or tangential boundary conditions in finite element code for incompressible fluid flow. *International Journal of Numerical Methods in Fluids 2* (1982), 225–238.

[15] Fourment, L., Mocellin, K., and Chenot, J.-L. An implicit contact algorithm for the 3D simulation of the forging process. In *Computational Plasticity: Fundamentals and Applications* (Barcelona,Spain, 17–20 March 1997), D. R. J. Owen, E. Oñate, and E. Hinton, Eds., CIMNE, pp. 873–877.

[16] Gaston, L. *Simulation numérique par éléments finis bidimensionnels du remplissage de moules de fonderie et étude expérimentale sur maquette hydraulique.* PhD thesis, Ecole Nationale Supérieure des Mines de Paris, 1997.

[17] Ghosh, S., and Kikuchi, N. Finite element formulation for the simulation of hot sheet metal forming processes. *International Journal of Engineering Science 26* (1988), 143–161.

[18] Godunov, S. Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics, (Cornell

Aeronautical Lab. Transl. from the Russian). *Math. Sbornik 47* (1959), 272–306.

[19] Haber, R., Shephard, M., Abel, J., Gallagher, R., and Greenberg, D. A general two-dimensional, graphical finite element preprocessor utilizing discrete transfinite mappings. *International Journal of Numerical Methods in Engineering 17* (1981), 1015–1044.

[20] Hinton, E., and Campbell, J. Local and global smoothing of discontinuous finite element functions using least square method. *International Journal of Numerical Methods in Engineering 8* (1974), 461–480.

[21] Hirsch, C. *Numerical computation of internal and external flows. Computational methods for inviscid and viscous flows.*, vol. 2. John Wiley and Sons, 1990.

[22] Hirt, C., Amsden, A., and Cook, J. An Arbitrary Lagrangian–Eulerian finite element method for all flow speeds. *Journal of Computational Physics 14* (1974), 227–253.

[23] Huerta, A., Casadei, F., and Donea, J. ALE stress update in transient plasticity problems. In *Computational Plasticity: Fundamentals and Applications* (Barcelona, Spain, 3–6 April 1995), D. R. J. Owen and E. Oñate, Eds., Pineridge Press, pp. 1865–1876.

[24] Huerta, A., Díez, P., and Rodríquez-Ferran, A. Adaptivity and error estimation. In *Simulation of Materials Processing : Theory, Methods and Applications* (Enschede, The Netherlands, 22–25 June 1998), J. Huétink and F. Baaijens, Eds., Numiform '98, A.A. Balkema, Rotterdam, pp. 63–74.

[25] Huerta, A., and Liu, W. Viscous flow with large free surface motion. *Computer Methods in Applied Mechanics and Engineering 69* (1988), 277–324.

[26] Huétink, J. *On the simulation of thermo-mechanical forming processes.* PhD thesis, University of Twente, 1986.

[27] Huétink, J., and Mooi, H. Application of numerical simulation in design and control of forming processes. *Journal de Physique IV 3* (1993), 1129–1134.

[28] Huétink, J., Vreede, P., and van der Lugt, J. Progress in mixed Eulerian–Lagrangian finite element simulation of forming processes.

International Journal of Numerical Methods in Engineering 30 (1990), 1441–1457.

[29] Hughes, T., and Belytschko, T. *Nonlinear finite element analysis.* 1998. A short course.

[30] Hughes, T., Liu, W., and Zimmermann, T. Lagrangian–Eulerian finite element formulation for incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering 29* (1981), 329–349.

[31] Jameson, A., Schmidt, W., and Turkel, E. Numerical solution of the Euler equations by finite volume methods using Runge–Kutta time stepping schemes. *AIAA paper* (1981), 81–1259.

[32] Kaliszky, S. *Plasticity: Theory and Engineering Applications.* Elsevier, 1989.

[33] Kobayashi, S., Oh, S.-I., and Altan, T. *Metal forming and the finite element method.* Oxford University Press, 1989.

[34] Koren, B. *A robust upwind discretization method for advection, diffusion and source terms*, vol. 45 of *Notes on Numerical Fluid Mechanics.* Vieweg, Braunschweig, 1993, ch. 5, pp. 117–138.

[35] Lesaint, P., and Raviart, P. *On a finite element method for solving the neutron transport equation.* Academic Press, 1974.

[36] Liszka, T., and Orkiz, J. The finite difference method at arbitrary irregular grids and its application in applied mechanics. *Computer and Structures 11* (1980), 83–95.

[37] Liu, W., Belytschko, T., and Chang, H. An Arbitrary Lagrangian–Eulerian finite element method for path-dependent materials. *Computer Methods in Applied Mechanics and Engineering 58* (1986), 227–245.

[38] Liu, W., H.Chang, Chen, J., and Belytschko, T. Arbitrary Lagrangian–Eulerian Petrov–Galerkin finite elements for nonlinear continua. *Computer Methods in Applied Mechanics and Engineering 68* (1988), 259–310.

[39] Magnin, B. *Modélisation du remplissage des moules d'injection pour polymères thermoplastiques par une méthode Eulérienne–Lagrangienne Arbitraire.* PhD thesis, Ecole Nationale Supérieure des Mines de Paris, 1994.

[40] Malvern, L. *Introduction to the mechanics of a continuous medium.* Prentice–Hall Inc., Englewood Cliffs, New Jersey, 1969.

[41] Mooi, H. *Finite element simulations of aluminium extrusion.* PhD thesis, University of Twente, 1996.

[42] Muttin, F., Coupez, T., Bellet, M., and Chenot, J.-L. Lagrangian finite-element analysis of time-dependent viscous free-surface flow using an automatic remeshing technique : application to metal casting flow. *International Journal of Numerical Methods in Engineering 36* (1993), 2001–2015.

[43] Nagtegaal, J., Parks, D., and Rice, J. On numerically accurate finite element solutions in the fully plastic range. *Computer Methods in Applied Mechanics and Engineering 4* (1974), 153–177.

[44] Nomura, T., and Hughes, T. An Arbitrary Lagrangian–Eulerian finite element method for interaction of fluid and a rigid body. *Computer Methods in Applied Mechanics and Engineering 95* (1992), 115–138.

[45] Ortiz, M., and Popov, E. Accuracy and stability of integration algorithms for elastoplastic constitutive relations. *International Journal of Numerical Methods in Engineering 21* (1985), 1561–1576.

[46] Ponthot, J. Efficient mesh management in Eulerian–Lagrangian method for large deformation analysis. *Proceedings of the 3rd international conference on numerical methods in industrial forming processes* (1989), 203–210.

[47] Ponthot, J. *Traitement unifie de la mecanique des milieux continus solides en grandes transformations par la methode des elements finis.* PhD thesis, University of Liège, 1995.

[48] Ramaswamy, B., and Kawahara, M. Arbitrary Lagrangian–Eulerian finite element method for unsteady, convective, incompressible viscous free surface fluid flow. *International Journal of Numerical Methods in Engineering 7* (1987), 1053–1075.

[49] Rekers, G. *Numerical simulation of unsteady viscoelastic flow of polymers.* PhD thesis, University of Twente, 1995.

[50] Rodríguez-Ferran, A., Askes, H., and Huerta, A. Arbitrary Lagrangian–Eulerian analyses of plastic failure in plates. In *Computational Mechanics: New Trends and Applications* (Buenos Aires,

Argentina, 29 June–2 July 1998), S. Idelsohn, E. Oñate, and E. Dvorkin, Eds., CIMNE,Barcelona,Spain.

[51] Sweby, P. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM Journal Numerical Analysis 21* (1984), 995–1011.

[52] Tamamidis, P. A new upwind scheme on triangular meshes using the finite volume method. *Computer Methods in Applied Mechanics and Engineering 124* (1995), 15–31.

[53] Thompson, E. Use of pseudo-concentrations to follow creeping viscous flows during transient analysis. *International Journal of Numerical Methods in Engineering 6* (1986), 749–761.

[54] Transvalor, CEMEF. *Forge2, Solving tomorrow's forging challenge today (manual).* Sophia Antipolis, France.

[55] van der Helm, P., Huétink, J., and Akkerman, R. Comparison of artificial dissipation and limited flux schemes in Arbitrary Lagrangian–Eulerian finite element formulations. *International Journal of Numerical Methods in Engineering 41* (1998), 1057–1076.

[56] van Haaren, M. ALE simulations with quadratic triangular elements. Graduate report Computational Mechanics, University of Twente, 1998.

[57] van Haaren, M., Stoker, H., van den Boogaard, A., and Huétink, J. The ALE–method with triangular elements: direct convection of integration point values. *International Journal of Numerical Methods in Engineering* (1998). to be published.

[58] van Leer, B. Towards the ultimate conservative difference scheme. IV. A new approach to numerical convection. *Journal of Computational Physics 23* (1977), 276–299.

[59] van Leer, B. Upwind-difference methods for aerodynamic problems governed by the Euler equations. *Lectures in Applied Mathematics 22* (1985), 327–336.

[60] Vreede, P. *A finite element method for simulations of 3-dimensional sheet metal forming.* PhD thesis, University of Twente, 1992.

[61] Vreugdenhil, C., and Koren, B., Eds. *Numerical methods for advection–diffusion problems* (Braunschweig, 1993), Vieweg. Notes on numerical fluid mechanics, volume 45.

[62] Wisselink, H., and Huétink, J. Simulation of stationary sheet metal cutting process. In *Simulation of Materials Processing : Theory, Methods and Applications* (Enschede, The Netherlands, 22–25 June 1998), J. Huétink and F. Baaijens, Eds., Numiform '98, A.A. Balkema, Rotterdam, pp. 985–990.

[63] Zalesak, S. Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of Computational Physics 31* (1979), 335–362.

[64] Zienkiewicz, O., and Taylor, R. *The finite element method*, fourth ed. McGraw–Hill book Company, 1991.

Appendix A

Finite element method

The finite element method is applied to discretise the weak formulations (3.28), (3.33) and (3.43). The domain Ω is split up into a finite number of subdomains Ω_e. As a result the integrals of the weak formulations are subdivided into a finite number of subintegrals. Subsequently we solve the weighted equilibrium equation of an element. For each element the velocity is described by the interpolation functions N and the nodal velocities $\mathbf{V}$ [64, 33]. In the same manner the coordinates $\mathbf{x}$ are interpolated. The pressure is interpolated using the interpolation functions M. We obtain:

$$\begin{aligned} &\mathbf{v} = \mathbf{V}^\alpha N^\alpha(\boldsymbol{\xi}) \quad ; \quad \mathbf{x} = \mathbf{X}^\alpha N^\alpha(\boldsymbol{\xi}) \quad ; \\ &p = P^\gamma M^\gamma(\boldsymbol{\xi}) \quad ; \quad \lambda = \lambda^l N^l(\boldsymbol{\xi}) \quad . \end{aligned} \tag{A.1}$$

It should be remarked that $\mathbf{V}$ contains the velocity components of the two dimensions of the problem. The Galerkin method is applied here, which means that the same interpolation functions are employed for the weight functions $\mathbf{v}^*$, p^* and λ^*.

$$\begin{aligned} &\mathbf{v}^* = \delta\mathbf{V}^\alpha N^\alpha(\boldsymbol{\xi}) \quad ; \quad p^* = \delta P^\gamma M^\gamma(\boldsymbol{\xi}) \quad ; \\ &\lambda^* = \delta\lambda^l N^l(\boldsymbol{\xi}) \quad . \end{aligned} \tag{A.2}$$

In the following sections these interpolations are applied to describe the integrals for each element. As a consequence we obtain a discrete set of equations per element. The total system of equations is derived by demanding continuity for the velocity and equilibrium of forces between the elements.

A.1 Viscoplastic model

In the case of a viscoplastic model the weak formulation for an element is (3.28):

$$\int_{\Omega_e} 2K(\sqrt{3}\dot{\bar{\varepsilon}})^{m \Leftrightarrow 1}\mathbf{D}^* : \mathbf{D}\ d\Omega - \int_{\Omega_e} (\overrightarrow{\nabla} \cdot \mathbf{v}^*)\ p\ d\Omega =$$
$$\int_{,\,C,e} \mathbf{v}^* \cdot \boldsymbol{\tau}_{\mathrm{fr}}\ d,\ + \int_{,\,T,e} \mathbf{v}^* \cdot \boldsymbol{T}\ d,\ + \int_{,\,C,e} \mathbf{v}^* \cdot \sigma_n\, \mathbf{n}\ d,\ ,\ \forall \mathbf{v}^* \in \mathcal{V}^* \tag{A.3}$$

and

$$\int_{,\,C,e} \lambda^*\, h(\mathbf{v})\ d,\ \leq 0\,,\ \forall \lambda^*\,,$$
$$\int_{\Omega_e} p^*(\overrightarrow{\nabla} \cdot \mathbf{v})\ d\Omega = 0\,,\ \forall p^*\,.$$

The interpolations (A.1) and (A.2) are substituted into these integrals resulting finally in a discrete system of equations. Now the various terms in the integrals are discretised consecutively.

The rate of deformation tensor $\mathbf{D}$ (2.24) becomes:

$$\mathrm{D}_{ij} = \frac{1}{2}(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}) = \frac{1}{2}(\frac{\partial N^\alpha}{\partial x_i}\delta_{jk} + \frac{\partial N^\alpha}{\partial x_j}\delta_{ik})V_k^\alpha = B_{ijk}^\alpha V_k^\alpha \quad \Longrightarrow$$
$$\mathbf{D} = \mathbf{B}^\alpha \cdot \mathbf{V}^\alpha\,, \tag{A.4}$$

where the nodal velocity vector $\mathbf{V}^\alpha$ consists of the velocity components of nodal point α. The divergence of the velocity field becomes:

$$D_{ii} = B_{iik}^\alpha V_k^\alpha \qquad \Longrightarrow \qquad \mathrm{tr}(\mathbf{D}) = \mathbf{B}_{\mathrm{i}}^\alpha \cdot \mathbf{V}^\alpha\,. \tag{A.5}$$

$\mathbf{D}^*$ is discretised in the same manner as $\mathbf{D}$:

$$\mathbf{D}^* = \delta\mathbf{V}^\alpha \cdot (\mathbf{B}^\alpha)^T. \tag{A.6}$$

Subsequently, the boundary integrals are discretised. The velocity field is interpolated using the same interpolation functions N on $,_C$ as in Ω. The contact with the tool is assumed to be unilateral, which means that the whole tool has a velocity $\mathbf{V}_{\mathrm{tool}}$. For the discretisation contact elements are used, so corresponding nodes are projected onto the tool with a prescribed nodal

velocity $\mathbf{V}^{\alpha}_{\text{tool}}$. The frictional behaviour can then be discretised using the viscous friction law (2.62):

$$\boldsymbol{\tau}_{\text{fr}} = -\mu\ N^{\alpha}\big((\mathbf{V}^{\alpha} - \mathbf{V}^{\alpha}_{\text{tool}}) \cdot \mathbf{t}\ \mathbf{t}\big) \tag{A.7}$$

For each element the following expression is obtained:

$$\begin{aligned}
&\delta\mathbf{V}^{\alpha} \cdot \int_{\Omega,e} 2K(\sqrt{3}\dot{\bar{\varepsilon}})^{m\Leftrightarrow 1}(\mathbf{B}^{\alpha})^{T} : \mathbf{B}^{\beta}\ d\Omega \cdot \mathbf{V}^{\beta} \\
&- \delta\mathbf{V}^{\alpha} \cdot \int_{\Omega,e} (\mathbf{B}^{\alpha}_{\text{i}})^{T}\ M^{\gamma}\ d\Omega\, P^{\gamma} \\
&+ \delta\mathbf{V}^{\alpha} \cdot \int_{,\ C,e} \mu\ \mathbf{t}\ N^{\alpha}N^{\beta}\ \mathbf{t}\ d,\ \cdot(\mathbf{V}^{\beta} - \mathbf{V}^{\beta}_{\text{tool}}) \\
&- \delta\mathbf{V}^{\alpha} \cdot \int_{,\ C,e} \mathbf{n}\ N^{\alpha}\ N^{l}\ d,\ \sigma^{l}_{n} = \delta\mathbf{V}^{\alpha} \cdot \int_{,\ T,e} N^{\alpha}\ \mathbf{T}\ d,\ ,\ \forall\delta\mathbf{V}^{\alpha}\,.
\end{aligned} \tag{A.8}$$

The equation for the incompressibility constraint is worked out in detail as follows:

$$\delta P^{\delta} \int_{\Omega,e} M^{\delta}\ \mathbf{B}^{\beta}_{\text{i}}\ d\Omega \cdot \mathbf{V}^{\beta} = 0\,,\ \ \forall\delta P^{\delta} \tag{A.9}$$

In incompressible or nearly incompressible cases volume locking can occur [64]. The problem of volume locking can be prevented by using a lower order interpolation of the pressure P and the term $(\overrightarrow{\nabla} \cdot \mathbf{v})$ than the interpolation of the velocity field. In this work a constant P and a constant $(\overrightarrow{\nabla} \cdot \mathbf{v})$ for each element is used and they are discontinuous across the element boundaries. So a constant interpolation function for an element, $M = 1$, is used for the pressure. As a result the pressure P can be eliminated for each element. The penalty method is applied to solve the pressure variable. The penalised form of the incompressibility constraint is:

$$\int_{\Omega,e} (\mathbf{B}^{\beta}_{\text{i}} \cdot \mathbf{V}^{\beta} - \frac{1}{\rho_p} P)\ d\Omega = 0\,, \tag{A.10}$$

with a large penalty coefficient ρ_p. When $\rho_p \to \infty$, the penalised problem converges to the original Lagrange multiplier problem. The expression (A.10) is evaluated in the middle of the element, which yields an expression for P:

$$P = \rho_p\ \mathbf{B}^{\beta}_{\text{i}} \cdot \mathbf{V}^{\beta}\,, \tag{A.11}$$

where $\mathbf{B}_{\mathrm{i}}^{\beta}$ is evaluated in the middle of the element. This is substituted into the second integral of (A.8), resulting in:

$$\delta\mathbf{V}^{\alpha} \cdot \int_{\Omega,e} \rho_p \, (\mathbf{B}_{\mathrm{i}}^{\alpha})^T \, \mathbf{B}_{\mathrm{i}}^{\beta} \, d\Omega \cdot \mathbf{V}^{\beta} \, . \tag{A.12}$$

This integral is evaluated in the middle of the element.
Since some compressibility is allowed in the penalty method, the isotropic part of $\mathbf{D} = \mathbf{B}^{\beta} \cdot \mathbf{V}^{\beta}$ has to be eliminated in the first integral of (A.8). We then obtain:

$$\int_{\Omega,e} 2K(\sqrt{3}\tilde{\dot{\varepsilon}})^{m \Leftrightarrow 1} (\mathbf{B}^{\alpha})^T : \mathbf{B}_{\mathrm{d}}^{\beta} \, d\Omega \cdot \mathbf{V}^{\beta}, \tag{A.13}$$

where

$$\mathbf{B}_{\mathrm{d}}^{\beta} = (\mathbf{B}^{\beta} - \mathbf{B}_{\mathrm{i}}^{\beta}\mathbf{1}). \tag{A.14}$$

The boundary , $_c$ is discretised using contact elements (Figure A.1). A nodal point on , $_c$ has a corresponding nodal point, the projection onto the tool (see Figure A.2). The non-penetration condition described by $h(\mathbf{v})$ is also solved using a penalty technique [8]. In the tangential direction at the boundary viscous frictional behaviour is assumed.

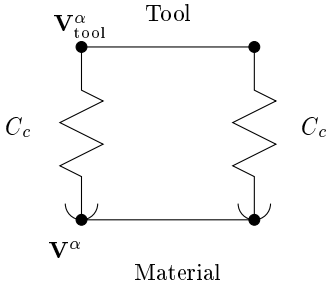


Figure A.1: Contact element.

Using the velocity non-penetration condition including the time integration according to [15]:

$$\begin{aligned} h(\mathbf{v})\, n_j &= N^{\alpha}\Big(\big((V_i^{\alpha} - V_{\mathrm{tool},i}^{\alpha})\; n_i \; \Delta t - g^{\alpha}(t)\big) \; n_j \Big) \qquad \Longrightarrow \\ h(\mathbf{v})\, \mathbf{n} &= N^{\alpha}\Big(\big((\mathbf{V}^{\alpha} - \mathbf{V}_{\mathrm{tool}}^{\alpha}) \cdot \mathbf{n} \; \Delta t - g^{\alpha}(t)\big) \; \mathbf{n} \Big) \end{aligned} \tag{A.15}$$

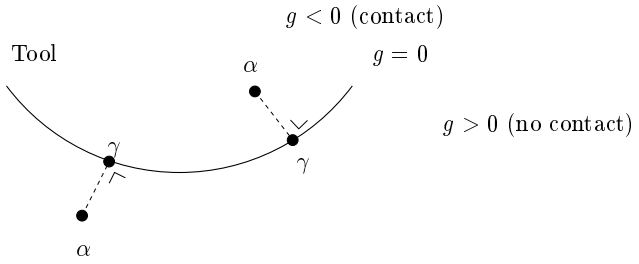


Figure A.2: Description of the gap function $g^\alpha(t)$.

where $g^\alpha(t)$ is the gap function of node α at time t. This is the distance between nodal point α and point γ, which is the projection of α onto the tool (see Figure A.2). The non-penetration constraint is written in the penalised form in a similar way as with the incompressibility condition (*cf.* (A.10)):

$$\delta\lambda^n \int_{,\ C,e} N^n N^\alpha \left(\left((\mathbf{V}^\alpha - \mathbf{V}^\alpha_{\text{tool}}) \cdot \mathbf{n} \Delta t - g^\alpha(t) \right) \mathbf{n} - \frac{1}{C_C} \sigma^\alpha_n \right) d,\ \leq 0\,, \forall \delta\lambda\,,$$

where the Lagrange multiplier σ^α_n must be less than or equal to zero. We interpolate σ_n in the same way as the velocity $\mathbf{v}$ and the integrals are evaluated between two corresponding contact nodal points. As a result the constraint equations for σ_n can be written for the two corresponding nodal points separately. So σ^α_n can be written in terms of the properties of nodal point α:

$$\sigma^\alpha_n = C_C \left((\mathbf{V}^\alpha - \mathbf{V}^\alpha_{\text{tool}}) \cdot \mathbf{n}\ \Delta t - g^\alpha(t) \right)\ \mathbf{n} \leq 0. \tag{A.16}$$

The contact stress σ^α_n should be smaller equal to 0. The penalty coefficient C_c can also be called a contact stiffness, as C_C can be interpreted as the stiffness of a spring between the tool and the material which allows some penetration of material into the tool. Substituting this into the last term of (A.3):

$$\int_{,\ C,e} C_C\ \mathbf{n}\ N^\alpha N^\beta \left\langle (\mathbf{V}^\beta - \mathbf{V}^\beta_{\text{tool}}) \cdot \mathbf{n} \Delta t - g^\beta(t) \right\rangle d,\ , \tag{A.17}$$

where $\langle . \rangle$ indicates that it is only taken into account when it is negative.
The final system of equations becomes:

$$[\mathbf{K}_M]^{\alpha\beta} \cdot \mathbf{V}^\beta + [\mathbf{K}_C]^{\alpha\beta} \cdot (\mathbf{V}^\beta - \mathbf{V}^\beta_{\text{tool}}) = \mathbf{F}^\alpha_T - \mathbf{F}^\alpha_C \tag{A.18}$$

where

$$[\mathbf{K}_M]^{\alpha\beta} = \int_{\Omega,e} \left(2K(\sqrt{3}\dot{\bar{\varepsilon}})^{m\Leftrightarrow 1}(\mathbf{B}^\alpha)^T : \mathbf{B}_{\mathrm{d}}^\beta + \rho_p(\mathbf{B}_{\mathrm{i}}^\alpha)^T\ \mathbf{B}_{\mathrm{i}}^\beta\right)\ d\Omega\,, \tag{A.19}$$

$$[\mathbf{K}_C]^{\alpha\beta} = \int_{,\ C,e} \left(\mu\ \mathbf{t}\ N^\alpha N^\beta\ \mathbf{t} - C_C\ \Delta t\ \mathbf{n}\ N^\alpha N^\beta\ \mathbf{n}\right)\ d,\ ,\ , \tag{A.20}$$

$$\mathbf{F}_T^\alpha = \int_{,\ T,e} N^\alpha \mathbf{T}\ d,\ ,\ , \tag{A.21}$$

$$\mathbf{F}_C^\alpha = \int_{,\ C,e} \left(C_C\ g(t)^\beta\ N^\alpha N^\beta\ \mathbf{n}\right)\ d,\ . \tag{A.22}$$

The first term of (A.18) is the discretisation of internal work $\partial W_{\mathrm{int}}$, where the hydrostatic part is obtained by reduced integration. The two integrands of the first integral can also be combined into one $\bar{\mathbf{B}}$-matrix. This is the so-called *B-bar method* [43]. The second term takes into account the discretisation of the contact with the tool. It is equal to zero when there is no contact. The last term of the right hand side $\mathbf{F}_C^\alpha$ takes into account the penalty on the gap at time t. $\mathbf{F}_T$ is the nodal force vector on $,\ _T$.

The system of equations can also be written in the form of the stiffness matrix $[\mathbf{K}]$, the nodal velocity $\mathbf{V}$ and the nodal force vector $\mathbf{F}$:

$$[\mathbf{K}]^{\alpha\beta} \cdot \mathbf{V}^\beta = \mathbf{F}^\alpha, \tag{A.23}$$

where $[\mathbf{K}] = [\mathbf{K}_M] + [\mathbf{K}_C]$ and $\mathbf{F} = \mathbf{F}_T - \mathbf{F}_C$. The vector $\mathbf{V}$ contains all the nodal velocities and it also contains the velocity of the nodal points projected onto the tool. For one single element the system of equations is known and hence the system of equations for the whole domain can be obtained.

A.2 Elastoplastic model

In the case of an elastoplastic model the weak form (3.33) must be discretised:

$$\begin{aligned}\int_{\Omega'} (\mathbf{v}^* \overleftarrow{\nabla}) : (\mathbf{s}' - p'\,\mathbf{1})\ d\Omega &= \int_{,\ C}' \mathbf{v}^* \cdot \boldsymbol{\tau}_{\mathrm{fr}}'\ d,\\ + \int_{,\ C}' \mathbf{v}^* \cdot \sigma_n\,\mathbf{n}'\ d,\ + \int_{,\ T}' \mathbf{v}^* \cdot \mathbf{T}'\ d,\ ,\ &\forall \mathbf{v}^* \in \mathcal{V}^*\end{aligned} \tag{A.24}$$

and

$$\int_{\Omega'} p^*(p' - p_{\text{ref}} + C_b \ \text{tr}(\mathbf{D}) \ \Delta t) \ d\Omega = 0 \,, \ \ \forall p^* \,,$$
$$\int_{,'_C} \lambda^* \, h'(\mathbf{v}) \ d, \ \leq 0 \,, \ \ \forall \lambda^* \,. \tag{A.25}$$

The same interpolation functions for $\mathbf{v}$, p and λ are used as in the previous section. The hydrostatic pressure p is interpolated and substituted into the first equation. This results again in a reduced integration of p. The non-penetration condition is discretised in the same way as in the previous section. The deviatoric stress $\mathbf{s}$ is obtained by the mean normal method in the corrector step instead of directly from the instantaneous material velocity field. Altogether we obtain the following system of equations:

$$\mathbf{Q}_M^\alpha + [\mathbf{K}_C]^{\alpha\beta} \cdot (\mathbf{V}^\beta - \mathbf{V}_{\text{tool}}^\beta) = \mathbf{F}_T^\alpha - \mathbf{F}_C^\alpha \tag{A.26}$$

where

$$\mathbf{Q}_M^\alpha = \int_{\Omega',e} \left((\mathbf{B}^\alpha)^T : \mathbf{s}' - (\mathbf{B}_{\text{i}}^\alpha)^T \ (p_{\text{ref}} + C_{\text{b}} \ \mathbf{B}_{\text{i}}^\beta \cdot \mathbf{V}^\beta \Delta t) \right) d\Omega \tag{A.27}$$

Comparing this result with (A.18), we now have $\mathbf{Q}_M^\alpha$ instead of $\left([\mathbf{K}_M]^{\alpha\beta} \cdot \mathbf{V}^\beta \right)$. The form $\mathbf{Q}_M^\alpha$ is not practical for the linearisation in the predictor step. Here the weak rate formulation of the internal virtual work δW_{int} will be used to provide the linearisation of $\mathbf{Q}_M^\alpha$ for the predictor step. The weak rate formulation results in an expression that is written in terms of a stiffness matrix and the vector $\mathbf{V}$.

A.2.1 Discretisation of the weak rate form

Now we concentrate on the discretisation of the weak rate formulation of ∂W_{int} (3.43):

$$\partial \dot{W}_{\text{int}} = \int_\Omega \mathbf{L}^* : (\mathbf{L} \cdot \boldsymbol{\sigma}) \ d\Omega - \int_\Omega (\text{tr}(\mathbf{D}^*) \ \dot{p}) \ d\Omega \tag{A.28}$$
$$+ \int_\Omega \mathbf{D}^* : (-\mathbf{I} \cdot \boldsymbol{\sigma} - \boldsymbol{\sigma} \cdot \mathbf{I} + \mathbf{L}^S + \boldsymbol{\sigma}\mathbf{1}) : \mathbf{D} \ d\Omega \tag{A.29}$$

and

$$\int_{\Omega} p^*(\dot{p} + C_b \operatorname{tr}(\mathbf{D}))\, d\Omega = 0\,, \quad \forall p^*\,. \tag{A.30}$$

The quantities are interpolated in the same way as in the previous section. Again using reduced integration for $\dot{p}$ and $(\operatorname{tr}(\mathbf{D})$ as for the incompressibility equation in the viscoplastic case, the following integral results:

$$\begin{aligned}\delta\mathbf{V}^\alpha \cdot \int_{\Omega,e} \big((\mathbf{B}^\alpha)^T : (-\mathbf{I}\cdot\boldsymbol{\sigma} - \boldsymbol{\sigma}\cdot\mathbf{I} + \mathbf{L}^S) : \mathbf{B}^\beta + C_b\, (\mathbf{B}^\alpha_{\mathrm{i}})^T\, \mathbf{B}^\beta_{\mathrm{i}} \\ + N^\alpha \overleftarrow{\nabla} \cdot \boldsymbol{\sigma} \cdot N^\beta \overleftarrow{\nabla} \big)\, d\Omega \cdot \mathbf{V}^\beta\,.\end{aligned} \tag{A.31}$$

This can be rewritten in the form of a stiffness matrix multiplied by the velocity:

$$\delta\mathbf{V}^\alpha \cdot [\mathbf{K}_M]^{\alpha\beta} \cdot \mathbf{V}^\beta\,. \tag{A.32}$$

In the predictor step this expression is employed for the linearisation of the term $\mathbf{Q}^\alpha_M$ of (A.26).
This method of obtaining the stiffness matrix from from the weak rate formulation can also be used for other history dependent models. The Coulomb friction model can be treated in a similar way as the elastoplastic model. In order to obtain a stiffness matrix, a weak rate formulation can be used [60].

Appendix B

Van Leer scheme for quadrilateral elements

For quadrilateral elements the new distribution is calculated from the following integrals:

$$\zeta^i = \frac{1}{\Delta x^i} \int\limits_{x^{i-\frac{1}{2}}}^{x^{i+\frac{1}{2}}} \zeta_\ell \, dx, \tag{B.1}$$

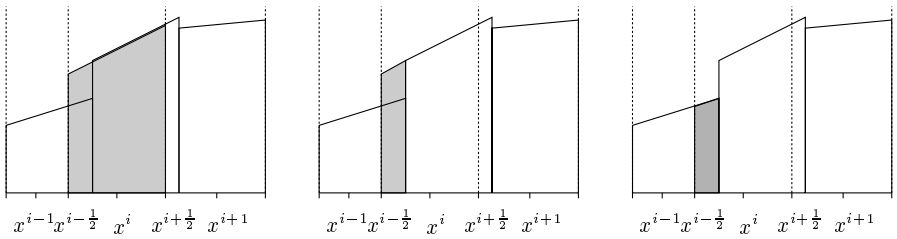
$$(\zeta^i)_{,x} = \frac{12}{(\Delta x^i)^3} \int\limits_{x^{i-\frac{1}{2}}}^{x^{i+\frac{1}{2}}} \zeta_\ell (x - x^i) \, dx \,, \tag{B.2}$$

where the integrands of the right hand sides are discontinuous distributions of ζ_ℓ in the new elements. We use for ζ_ℓ the distribution that is limited according to (5.41). First we work out in detail (B.1) with the help of Figure B.1:

$$\zeta^i = \frac{1}{\Delta x^i} \left(\int\limits_{\Omega^i} \zeta_\ell^i(x) \, dx - \sum_k \int\limits_{\Omega_\ell^k \cap \Omega^i} \zeta_\ell^i(x) \, dx + \sum_k \int\limits_{\Omega_\ell^k \cap \Omega^i} \zeta_\ell^k(x) \, dx \right) , \tag{B.3}$$

where k is a neighbour element of element i. The integrals are worked out over the new element with a length $\Delta x^i = \mid x^{i+\frac{1}{2}} - x^{i \Leftrightarrow \frac{1}{2}} \mid$. The first integral of (B.3) is worked out in detail as follows:

$$\frac{1}{\Delta x^i} \int\limits_{\Omega^i} \zeta_\ell^i(x) \, dx = \zeta_\ell^i - C_i \Delta x^i (\zeta_\ell^i)_{,x} \,, \tag{B.4}$$



(a) (b) (c)

Figure B.1: Calculation of the discontinuous integrals of (B.3) for positive $\Delta u_c^{i\Leftrightarrow\frac{1}{2}}$ and $\Delta u_c^{i+\frac{1}{2}}$. (a) Integrating the distribution $\zeta_\ell^i(x)$ over the total element Ω^i. (b) Subtracting the distribution $\zeta_\ell^i(x)$ for the indicated area. (c) Adding the distribution $\zeta_\ell^{i\Leftrightarrow 1}(x)$.

where

$$C_i = \frac{\Delta u_c^{i\Leftrightarrow\frac{1}{2}} + \Delta u_c^{i+\frac{1}{2}}}{2\Delta x^i}. \tag{B.5}$$

The convective displacements Δu_c are defined by (5.33). The second integral of (B.3) is the overlap of the new element Ω^i with an old neighbour element $\Omega_\ell^{i\Leftrightarrow 1}$ or Ω_ℓ^{i+1}. In Figure B.1(b) only the overlap with an updated Lagrangian neighbour element $\Omega_\ell^{i\Leftrightarrow 1}$ is considered, which means that $\Delta u_c^{i\Leftrightarrow\frac{1}{2}}$ is positive. This integral becomes:

$$\frac{1}{\Delta x^i} \int\limits_{\Omega_\ell^{i-1}\cap\Omega^i} \zeta_\ell^i(x)\,dx = C_{i\Leftrightarrow\frac{1}{2}}\Big(\zeta_\ell^i + \Big(C_i\Delta x^i + \frac{\Delta x^i}{2}(1 - C_{i\Leftrightarrow\frac{1}{2}})\Big)(\zeta_\ell^i)_{,x}\Big), \tag{B.6}$$

where

$$C_{i\Leftrightarrow\frac{1}{2}} = \begin{cases} \dfrac{\Delta u_c^{i\Leftrightarrow\frac{1}{2}}}{\Delta x^i} & \text{if } \Delta u_c^{i\Leftrightarrow\frac{1}{2}} > 0, \\ 0 & \text{if } \Delta u_c^i < 0. \end{cases} \tag{B.7}$$

One can interpret this as the area of the element multiplied by the mean value of this part. The third integral of (B.3) represents the contribution $\zeta_\ell^{i\Leftrightarrow 1}$ of the updated Lagrangian neighbour element $(i-1)$ (see Figure B.1(c)):

$$\frac{1}{\Delta x^i} \int\limits_{\Omega_\ell^{i-1}\cap\Omega^i} \zeta_\ell^{i\Leftrightarrow 1}(x)\,dx = C_{i\Leftrightarrow\frac{1}{2}}\Big(\zeta_\ell^{i\Leftrightarrow 1} + \frac{\Delta x^i}{2}(1 - C_{i\Leftrightarrow\frac{1}{2}})(\zeta_\ell^{i\Leftrightarrow 1})_{,x}\Big). \tag{B.8}$$

Here we have assumed that the element $(i-1)$ also has an element length of Δx^i, *i.e.* the same as element i. So we assume a locally homogeneous mesh. If there is an overlap between the new element Ω^i and the updated Lagrangian neighbour element $(i+1)$, so that $\Delta u_c^{i+\frac{1}{2}}$ is negative, we can derive similar equations. All these integrals are assembled and arranged into

$$\begin{aligned}\zeta^i = & C_{i\Leftrightarrow\frac{1}{2}}\zeta_\ell^{i\Leftrightarrow 1} \; + \; (1 - C_{i\Leftrightarrow\frac{1}{2}} + C_{i+\frac{1}{2}})\zeta_\ell^i \; - \; C_{i+\frac{1}{2}}\zeta_\ell^{i+1} \\ & + \; \frac{C_{i\Leftrightarrow\frac{1}{2}}\Delta x^i}{2}(1 - C_{i\Leftrightarrow\frac{1}{2}})(\zeta_\ell^{i\Leftrightarrow 1})_{,x} \; + \; \Bigg(C_i\Delta x^i(-1 + C_{i\Leftrightarrow\frac{1}{2}} - C_{i+\frac{1}{2}}) \\ & + \frac{C_{i\Leftrightarrow\frac{1}{2}}\Delta x^i}{2}(1 - C_{i\Leftrightarrow\frac{1}{2}}) + \frac{C_{i+\frac{1}{2}}\Delta x^i}{2}(1 + C_{i+\frac{1}{2}})\Bigg)(\zeta_\ell^i)_{,x} \\ & + \frac{C_{i+\frac{1}{2}}\Delta x^i}{2}(1 + C_{i+\frac{1}{2}})(\zeta_\ell^{i+1})_{,x} \; ,\end{aligned} \tag{B.9}$$

where

$$\begin{aligned} C_{i\Leftrightarrow\frac{1}{2}} &= \begin{cases} \dfrac{\Delta u_c^{i\Leftrightarrow\frac{1}{2}}}{\Delta x^i} & \text{if } \Delta u_c^{i\Leftrightarrow\frac{1}{2}} > 0, \\ 0 & \text{if } \Delta u_c^i < 0. \end{cases} \\ C_{i+\frac{1}{2}} &= \begin{cases} \dfrac{\Delta u_c^{i+\frac{1}{2}}}{\Delta x^i} & \text{if } \Delta u_c^{i+\frac{1}{2}} < 0, \\ 0 & \text{if } \Delta u_c^{i+\frac{1}{2}} > 0, \end{cases} \\ C_i &= \frac{\Delta u_c^{i\Leftrightarrow\frac{1}{2}} + \Delta u_c^{i+\frac{1}{2}}}{2\Delta x^i}. \end{aligned} \tag{B.10}$$

In the same way the integrals of (B.2) are calculated with the help of Figure B.1:

$$\begin{aligned}(\zeta_\ell^i)_{,x} = & \frac{12}{(\Delta x^i)^3}\Bigg(\int_{\Omega_\ell^i} \zeta_\ell^i(x)\; (x - x^i)\, dx - \sum_k \int_{\Omega_\ell^k \cap \Omega^i} \zeta_\ell^i(x)\; (x - x^i) dx \\ & + \sum_k \int_{\Omega_\ell^k \cap \Omega^i} \zeta_\ell^k\; (x - x^i)\, dx \Bigg) .\end{aligned} \tag{B.11}$$

The first integral becomes:

$$\frac{12}{(\Delta x^i)^3} \int_{\Omega_\ell^i} \zeta_\ell^i(x)\; (x - x^i)\, dx = (\zeta_\ell^i)_{,x} \, . \tag{B.12}$$

In case of overlap with neighbour element $(i-1)$ the second integral becomes:

$$\begin{aligned}\frac{12}{(\Delta x^i)^3}\int\limits_{\Omega_\ell^{i-1}\cap\,\Omega^i}\zeta_\ell^i(x)\;(x-x^i)\,dx &= \frac{6C_{i\Leftrightarrow\frac{1}{2}}}{\Delta x^i}(1-C_{i\Leftrightarrow\frac{1}{2}})\zeta_\ell^i \\ +\;6\Big(C_i(C_{i\Leftrightarrow\frac{1}{2}}(1-C_{i\Leftrightarrow\frac{1}{2}})&+C_{i\Leftrightarrow\frac{1}{2}}(-3+6C_{i\Leftrightarrow\frac{1}{2}}-4C^2_{i\Leftrightarrow\frac{1}{2}})\Big)(\zeta_\ell^i)_{,x}\,.\end{aligned} \tag{B.13}$$

The third integral of (B.11) yields the contribution of neighbour element $(i-1)$:

$$\begin{aligned}\frac{12}{(\Delta x^i)^3}\int\limits_{\Omega_\ell^{i-1}\cap\,\Omega^i}\zeta_\ell^{i\Leftrightarrow 1}\;(x-x^i)\,dx &= \frac{6C_{i\Leftrightarrow\frac{1}{2}}}{\Delta x^i}(-1+C_{i\Leftrightarrow\frac{1}{2}})\zeta_\ell^{i\Leftrightarrow 1} \\ -\;C_{i\Leftrightarrow\frac{1}{2}}(3-6C_{i\Leftrightarrow\frac{1}{2}}&+2C^2_{i\Leftrightarrow\frac{1}{2}})(\zeta_\ell^{i\Leftrightarrow 1})_{,x}\,.\end{aligned} \tag{B.14}$$

Summing the integrals and rearranging the terms results in:

$$\begin{aligned}(\zeta^i)_{,x} = \frac{6C_{i\Leftrightarrow\frac{1}{2}}}{\Delta x^i}(-1+C_{i\Leftrightarrow\frac{1}{2}})\zeta_\ell^{i\Leftrightarrow 1}\;&+\;\frac{6}{\Delta x^i}\big(C_{i+\frac{1}{2}}(1+C_{i+\frac{1}{2}}) \\ +\;C_{i\Leftrightarrow\frac{1}{2}}(1-C_{i\Leftrightarrow\frac{1}{2}})\big)\zeta_\ell^i\;&+\;\frac{6C_{i+\frac{1}{2}}}{\Delta x^i}(-1-C_{i+\frac{1}{2}})\zeta_\ell^{i+1} \\ -\;C_{i\Leftrightarrow\frac{1}{2}}(3-6C_{i\Leftrightarrow\frac{1}{2}}&+2C^2_{i\Leftrightarrow\frac{1}{2}})(\zeta_\ell^{i\Leftrightarrow 1})_{,x}\;+ \\ \Big(1+C_{i+\frac{1}{2}}(3+6C_{i+\frac{1}{2}}&+4C^2_{i+\frac{1}{2}}) \\ +\;C_{i\Leftrightarrow\frac{1}{2}}(-3+6C_{i\Leftrightarrow\frac{1}{2}}&-4C^2_{i\Leftrightarrow\frac{1}{2}})- \\ 6C_i(C_{i+\frac{1}{2}}(1+C_{i+\frac{1}{2}})&+C_{i\Leftrightarrow\frac{1}{2}}(1-C_{i\Leftrightarrow\frac{1}{2}}))\Big)(\zeta_\ell^i)_{,x}\;+ \\ C_{i+\frac{1}{2}}(3+6C_{i+\frac{1}{2}}&+2C^2_{i+\frac{1}{2}})(\zeta_\ell^{i+1})_{,x}.\end{aligned} \tag{B.15}$$

Appendix C

Van Leer based scheme for triangular elements

In this appendix the integrals of the Van Leer based scheme for triangular elements of section 5.3.2, *i.e.* (5.48) to (5.50) are worked out in detail. This work is taken from [56, 57].
These integrals can be split into integrals over triangles and integrals over rectangles. For convenience the integrals are worked out in sums over the corner nodes. In these sums the index j refers to the node number of the triangle or rectangle. However, the index $j-1$ can become 0. In that case $j-1$ represents node 3 for a triangle and node 4 for a rectangle. The index $j+1$ can exceed the maximum node number. In the case that $j+1$ has the value 4 for a triangle or the value 5 for a rectangle, $j+1$ represents node 1.

First, we work out the third integral of the left hand side of (5.48). The second integral on the left hand side can be worked out in a similar way. The integral is approximated with an integral over the rectangle as shown in Figure C.1. This integral becomes

$$\begin{aligned}
\int\limits_{\Omega_\ell^k \cap \Omega^i} & \left\{ \zeta_\ell^k + (\zeta_\ell^k)_{,x}(x-\bar{x}_\ell^k) + (\zeta_\ell^k)_{,y}(y-\bar{y}_\ell^k) \right\} d\Omega = \\
& A_\square \Big(\zeta_\ell^k - (\zeta_\ell^k)_{,x}\,\bar{x}_\ell^k - (\zeta_\ell^k)_{,y}\,\bar{y}_\ell^k \Big) \\
& + \frac{1}{6}(\zeta_\ell^k)_{,x} \sum_{j=1}^{4} x_j \Big(x_j(y_{j+1} - y_{j\Leftarrow 1}) - y_j(x_{j+1} - x_{j\Leftarrow 1}) \Big) \\
& + \frac{1}{6}(\zeta_\ell^k)_{,y} \sum_{j=1}^{4} y_j \Big(x_j(y_{j+1} - y_{j\Leftarrow 1}) - y_j(x_{j+1} - x_{j\Leftarrow 1}) \Big),
\end{aligned} \tag{C.1}$$

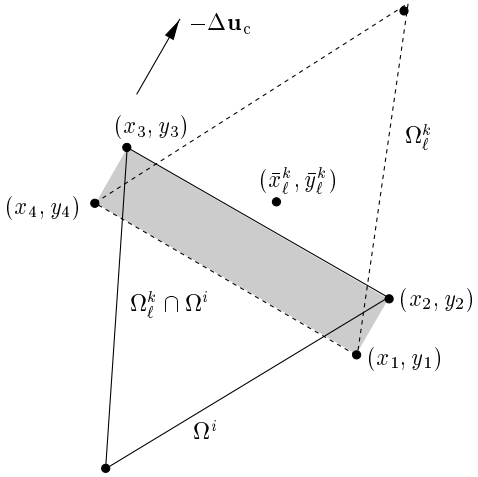


Figure C.1: Approximation of the overlap (shaded area) of the updated Lagrangian element k with the area Ω_ℓ^k (dashed lines) and the new element i with the area Ω^i (solid lines).

where

$$A_\square = \frac{1}{2} \sum_{j=1}^{4} x_j (y_{j+1} - y_{j\Leftarrow 1}), \tag{C.2}$$

which represents the area of the rectangle.

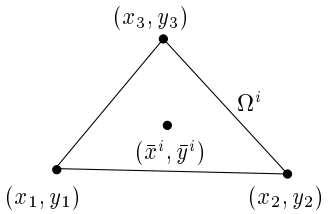


Figure C.2: New triangular element i with area Ω^i and the centroid $(\bar{x}^i, \bar{y}^i)$.

The left hand sides of (5.49) and (5.50) are integrals over the *new* triangular

element area Ω^i as shown in Figure C.2 and this yields:

$$\int_{\Omega^i} \Big\{ \zeta^i + (\zeta^i)_{,x}(x-\bar{x}^i) + (\zeta^i)_{,y}(y-\bar{y}^i) \Big\} (x-\bar{x}^i)\, d\Omega =$$
$$\frac{1}{36} A_\triangle \Big((\zeta^i)_{,x} \sum_{j=1}^{3} x_j (2x_j - x_{j+1} - x_{j\Leftrightarrow 1}) \tag{C.3}$$
$$+ (\zeta^i)_{,y} \sum_{j=1}^{3} x_j (2y_j - y_{j+1} - y_{j\Leftrightarrow 1}) \Big)$$

and

$$\int_{\Omega^i} \Big\{ \zeta^i + (\zeta^i)_{,x}(x-\bar{x}^i) + (\zeta^i)_{,y}(y-\bar{y}^i) \Big\} (y-\bar{y}^i)\, d\Omega =$$
$$\frac{1}{36} A_\triangle \Big((\zeta^i)_{,x} \sum_{j=1}^{3} y_j (2x_j - x_{j+1} - x_{j\Leftrightarrow 1}) \tag{C.4}$$
$$+ (\zeta^i)_{,y} \sum_{j=1}^{3} y_j (2y_j - y_{j+1} - y_{j\Leftrightarrow 1}) \Big),$$

where $(\bar{x}^i, \bar{y}^i)$ is the centroid of the *new* element i. The area of the triangle $A_\triangle$ is given by:

$$A_\triangle = \frac{1}{2} \sum_{j=1}^{3} x_j (y_{j+1} - y_{j\Leftrightarrow 1}). \tag{C.5}$$

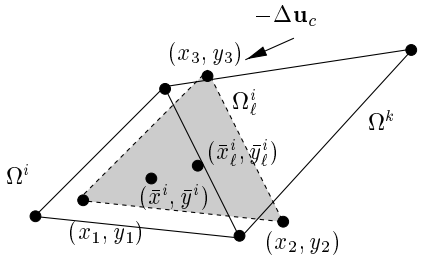


Figure C.3: The integral over the updated Lagrangian element Ω^i_ℓ (shaded area).

The integrals over the updated Lagrangian element Ω^i_ℓ in the right hand sides of (5.49) and (5.50) are worked out in a similar way as with the previous

two integrals (C.3) and (C.4). An important difference is that now the first moment of the updated Lagrangian distribution ζ_ℓ^i is taken with respect to $(\bar{x}_i, \bar{y}_i)$. So the centroid of the updated Lagrangian distribution (x_ℓ^i, y_ℓ^i) does not coincide with (x^i, y^i). This leads to the following two relations:

$$\int\limits_{\Omega_\ell^i} \Big({\zeta_\ell}^i + ({\zeta_\ell}^i)_{,x}(x - \bar{x}_\ell^i) + ({\zeta_\ell}^i)_{,y}(y - \bar{y}_\ell^i) \Big) (x - \bar{x}^i)\, d\Omega = A_\triangle \Big(C_1 {\zeta_\ell}^i + C_2({\zeta_\ell}^i)_{,x} + C_3({\zeta_\ell}^i)_{,y} \Big), \tag{C.6}$$

where

$$C_1 = \frac{1}{3}\Big(\sum_{j=1}^{3} x_j - 3\bar{x}^i\Big)\,, \tag{C.7}$$

$$C_2 = \frac{1}{12}\Big(\sum_{j=1}^{3} x_j(2x_j + x_{j+1} + x_{j\Leftarrow 1} - 4\bar{x}^i - 4\bar{x}_\ell^i) + 12\bar{x}^i\bar{x}_\ell^i \Big)\,, \tag{C.8}$$

$$C_3 = \frac{1}{12}\Big(\sum_{j=1}^{3} \big(y_j(2x_j + x_{j+1} + x_{j\Leftarrow 1} - 4\bar{x}^i) - 4x_j{\bar{y}_\ell}^i\big) + 12\bar{x}_i\bar{y}_\ell^i \Big) \tag{C.9}$$

and

$$\int\limits_{\Omega_\ell^i} \Big\{ {\zeta_\ell}^i + ({\zeta_\ell}^i)_{,x}(x - \bar{x}_\ell^i) + ({\zeta_\ell}^i)_{,y}(y - \bar{y}_\ell^i) \Big\} (y - \bar{y}^i)\, d\Omega = 2A_\triangle \Big(C_1 {\zeta_\ell}^i + C_2({\zeta_\ell}^i)_{,x} + C_3({\zeta_\ell}^i)_{,y} \Big)\,, \tag{C.10}$$

where

$$C_1 = \frac{1}{3}\Big(\sum_{j=1}^{3} y_j - 3\bar{y}^i\Big)\,, \tag{C.11}$$

$$C_2 = \frac{1}{12}\Big(\sum_{j=1}^{3} \big(x_j(2y_j + y_{j+1} + y_{j\Leftarrow 1} - 4\bar{y}^i) - 4y_j\bar{x}_\ell^i\big) + 12\bar{y}_i{\bar{x}_\ell}^i\Big)\,, \tag{C.12}$$

$$C_3 = \frac{1}{12}\Big(\sum_{j=1}^{3} y_j(2y_j + y_{j+1} + y_{j\Leftrightarrow 1} - 4\bar{y}^i - 4\bar{y}^i_\ell) + 12\bar{y}^i\bar{y}^i_\ell)\Big)\,. \tag{C.13}$$

Finally, expressions for the other integrals on the right hand side in (5.49) and (5.50) are given (see again Figure C.1):

$$\begin{aligned}\int\limits_{\Omega^i\cap\Omega^k_\ell} &\Big\{\zeta^k_\ell + (\zeta^k_\ell)_{,x}(x-\bar{x}^k_\ell) + (\zeta^k_\ell)_{,y}(y-\bar{y}^k_\ell)\Big\}(x-\bar{x}^i)\,d\Omega = \\ &A_\square\Big(C_1\zeta^k_\ell + C_2(\zeta^k_\ell)_{,x} + C_3(\zeta^k_\ell)_{,y}\Big),\end{aligned} \tag{C.14}$$

where

$$C_1 = -\bar{x}^i + \frac{1}{6}\sum_{j=1}^{4} x_j\Big(x_j(y_{j+1} - y_{j\Leftrightarrow 1}) - y_j(x_{j+1} - x_{j\Leftrightarrow 1})\Big)\,, \tag{C.15}$$

$$\begin{aligned}C_2 = \bar{x}^k_\ell\bar{x}^i + \frac{1}{12}\sum_{j=1}^{4} x_j\Big(&x_j(x_{j+1}y_{j+1} - x_{j\Leftrightarrow 1}y_{j\Leftrightarrow 1} \\ &+ x_j\Big(x_j - 2(\bar{x}^i + \bar{x}^k_\ell)\Big)(y_{j+1} - y_{j\Leftrightarrow 1}) \\ &- y_j\Big(x_j - 2(\bar{x}^i + \bar{x}^k_\ell)\Big)(x_{j+1} - x_{j\Leftrightarrow 1})\Big)\,,\end{aligned} \tag{C.16}$$

$$\begin{aligned}C_3 = \bar{y}^k_\ell\bar{x}^i + \frac{1}{24}\sum_{j=1}^{4} x_j\Big(&x_j(y^2_{j+1} - y^2_{j\Leftrightarrow 1}) \\ &+ 2y_j\Big(x_j - 2\bar{x}^i - 2(\bar{y}^k_\ell)\frac{x_j}{y_j}\Big)(y_{j+1} - y_{j\Leftrightarrow 1}) \\ &- 2y_j\Big(y_j - 2\bar{x}^i\frac{y_j}{x_j} - 2(\bar{y}^k_\ell)\Big)(x_{j+1} - x_{j\Leftrightarrow 1})\Big)\end{aligned} \tag{C.17}$$

and

$$\begin{aligned}\int\limits_{\Omega^i\cap\Omega^k_\ell} &\Big\{\zeta^k_\ell + (\zeta^k_\ell)_{,x}(x-\bar{x}^k_\ell) + (\zeta^k_\ell)_{,y}(y-\bar{y}^k_\ell)\Big\}(y-\bar{y}_i)\,d\Omega = \\ &A_\square\Big(C_1\zeta^k_\ell + C_2(\zeta^k_\ell)_{,x} + C_3(\zeta^k_\ell)_{,y}\Big),\end{aligned} \tag{C.18}$$

where

$$C_1 = -\bar{y}^i + \frac{1}{6}\sum_{j=1}^{4} y_j\Big(x_j(y_{j+1} - y_{j\Leftrightarrow 1}) - y_j(x_{j+1} - x_{j\Leftrightarrow 1})\Big)\,, \tag{C.19}$$

$$\begin{aligned} C_2 = \bar{x}^k_\ell \bar{y}^i + \frac{1}{24}\sum_{j=1}^{4} y_j\Bigg(&-y_j(x^2_{j+1} - x^2_{j\Leftrightarrow 1}) \\ &+ 2x_j\Big(x_j - 2\bar{y}^i\frac{x_j}{y_j} - 2(\bar{x}^k_\ell)\Big)(y_{j+1} - y_{j\Leftrightarrow 1}) \\ &- 2x_j\Big(y_j - 2\bar{y}^i - 2(\bar{x}^k_\ell)\frac{y_j}{x_j}\Big)(x_{j+1} - x_{j\Leftrightarrow 1})\Bigg)\,, \end{aligned} \tag{C.20}$$

$$\begin{aligned} C_3 = \bar{y}^k_\ell \bar{y}_i + \frac{1}{12}\sum_{j=1}^{4} y_j\Bigg(&-y_j(x_{j+1}y_{j+1} - x_{j\Leftrightarrow 1}y_{j\Leftrightarrow 1}) \\ &+ x_j\Big(y_j - 2(\bar{y}_i + \bar{y}^k_\ell)\Big)(y_{j+1} - y_{j\Leftrightarrow 1}) \\ &- y_j\Big(y_j - 2(\bar{y}_i + \bar{y}^k_\ell)\Big)(x_{j+1} - x_{j\Leftrightarrow 1})\Bigg)\,. \end{aligned} \tag{C.21}$$

Remark that the relations of this appendix can also be applied for quadrilateral elements.