Chapter 14
Arbitrary Lagrangian–Eulerian Methods

J. Donea[1], Antonio Huerta[2], J.-Ph. Ponthot[1] and A. Rodríguez-Ferran[2]

[1] *Université de Liège, Liège, Belgium*
[2] *Universitat Politècnica de Catalunya, Barcelona, Spain*

1 INTRODUCTION

The numerical simulation of multidimensional problems in fluid dynamics and nonlinear solid mechanics often requires coping with strong distortions of the continuum under consideration while allowing for a clear delineation of free surfaces and fluid–fluid, solid–solid, or fluid–structure interfaces. A fundamentally important consideration when developing a computer code for simulating problems in this class is the choice of an appropriate *kinematical description* of the continuum. In fact, such a choice determines the relationship between the deforming continuum and the finite grid or mesh of computing zones, and thus conditions the ability of the numerical method to deal with large distortions and provide an accurate resolution of material interfaces and mobile boundaries.

Encyclopedia of Computational Mechanics, Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes. Volume 1: *Fundamentals*. ISBN: 0-470-84699-2.

The algorithms of continuum mechanics usually make use of two classical descriptions of motion: the Lagrangian description and the Eulerian description; see, for instance, (Malvern, 1969). The arbitrary Lagrangian–Eulerian (ALE, in short) description, which is the subject of the present chapter, was developed in an attempt to combine the advantages of the above classical kinematical descriptions, while minimizing their respective drawbacks as far as possible.

Lagrangian algorithms, in which each individual node of the computational mesh follows the associated material particle during motion (see Figure 1), are mainly used in structural mechanics. The Lagrangian description allows an easy tracking of free surfaces and interfaces between different materials. It also facilitates the treatment of materials with history-dependent constitutive relations. Its weakness is its inability to follow large distortions of the computational domain without recourse to frequent remeshing operations.

Eulerian algorithms are widely used in fluid dynamics. Here, as shown in Figure 1, the computational mesh is fixed and the continuum moves with respect to the grid. In the Eulerian description, large distortions in the continuum motion can be handled with relative ease, but generally at the expense of precise interface definition and the resolution of flow details.

Because of the shortcomings of purely Lagrangian and purely Eulerian descriptions, a technique has been developed that succeeds, to a certain extent, in combining the best features of both the Lagrangian and the Eulerian approaches. Such a technique is known as the *arbitrary Lagrangian–Eulerian (ALE) description*. In the ALE description, the nodes of the computational mesh may be moved with the continuum in normal Lagrangian fashion, or be held fixed in Eulerian manner, or, as suggested in

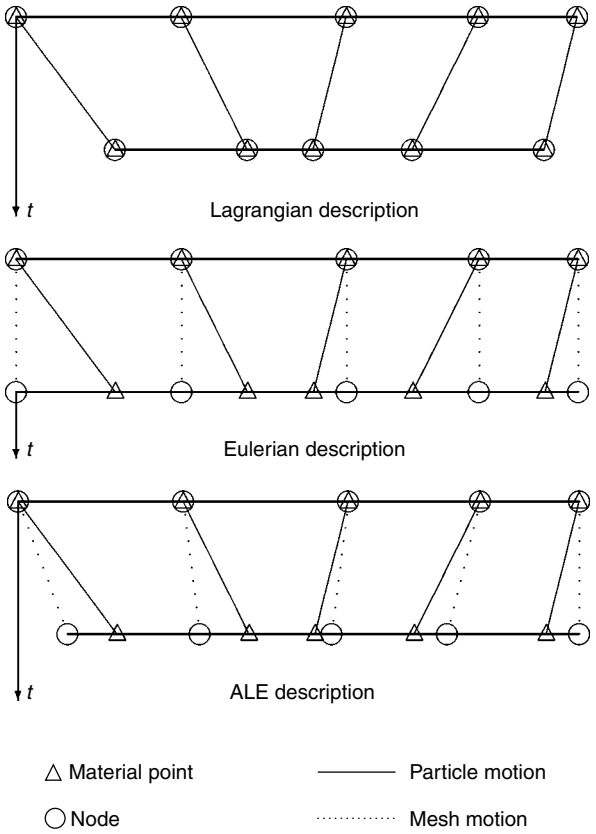

Figure 1. One-dimensional example of Lagrangian, Eulerian and ALE mesh and particle motion.

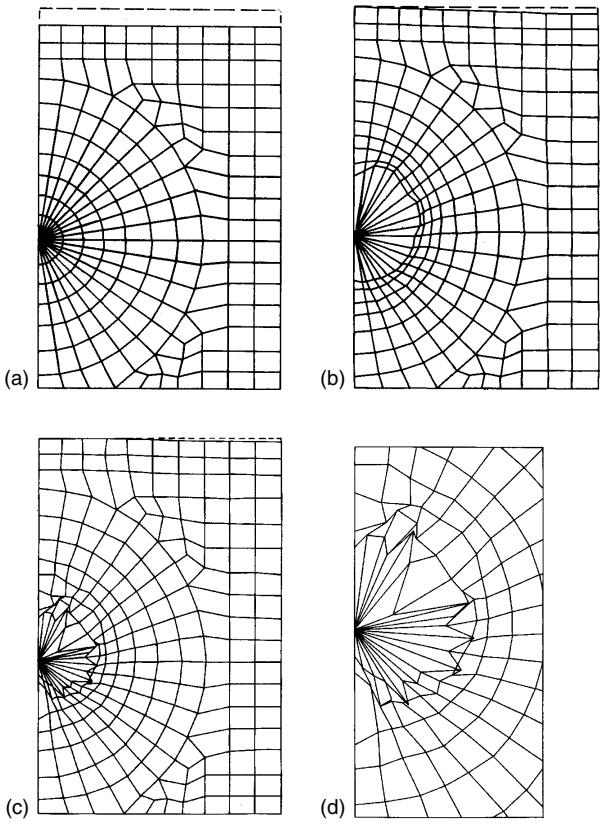

Figure 2. Lagrangian versus ALE descriptions: (a) initial FE mesh; (b) ALE mesh at $t = 1$ ms; (c) Lagrangian mesh at $t = 1$ ms; (d) details of interface in Lagrangian description.

Figure 1, be moved in some arbitrarily specified way to give a continuous rezoning capability. Because of this freedom in moving the computational mesh offered by the ALE description, greater distortions of the continuum can be handled than would be allowed by a purely Lagrangian method, with more resolution than that afforded by a purely Eulerian approach. The simple example in Figure 2 illustrates the ability of the ALE description to accommodate significant distortions of the computational mesh, while preserving the clear delineation of interfaces typical of a purely Lagrangian approach. A coarse finite element mesh is used to model the detonation of an explosive charge in an extremely strong cylindrical vessel partially filled with water. A comparison is made of the mesh configurations at time $t = 1.0$ ms obtained respectively, with the ALE description (with automatic continuous rezoning) and with a purely Lagrangian mesh description. As further evidenced by the details of the charge–water interface, the Lagrangian approach suffers from a severe degradation of the computational mesh, in contrast with the ability of the ALE approach to maintain quite a regular mesh configuration of the charge–water interface.

The aim of the present chapter is to provide an in-depth survey of ALE methods, including both conceptual aspects and numerical implementation details in view of the applications in large deformation material response, fluid dynamics, nonlinear solid mechanics, and coupled fluid–structure problems. The chapter is organized as follows. The next section introduces the ALE kinematical description as a generalization of the classical Lagrangian and Eulerian descriptions of motion. Such generalization rests upon the introduction of a so-called *referential domain* and on the mapping between the referential domain and the classical, material, and spatial domains. Then, the fundamental ALE equation is introduced, which provides a relationship between material time derivative and referential time derivative. On this basis, the ALE form of the basic conservation equations for mass, momentum, and energy is established. Computational aspects of the ALE algorithms are then addressed. This includes mesh-update procedures in finite element analysis, the combination of ALE and mesh-refinement procedures, as well as the use of ALE in connection with mesh-free methods. The chapter closes with a discussion of problems commonly encountered in the computer implementation of ALE algorithms in fluid dynamics, solid mechanics, and coupled problems describing fluid–structure interaction.

2 DESCRIPTIONS OF MOTION

Since the ALE description of motion is a generalization of the Lagrangian and Eulerian descriptions, we start with a brief reminder of these classical descriptions of motion. We closely follow the presentation by Donea and Huerta (2003).

2.1 Lagrangian and Eulerian viewpoints

Two domains are commonly used in continuum mechanics: the material domain $R_{\boldsymbol{X}} \subset \mathbb{R}^{n_{sd}}$, with n_{sd} spatial dimensions, made up of material particles $\boldsymbol{X}$, and the spatial domain $R_{\boldsymbol{x}}$, consisting of spatial points $\boldsymbol{x}$.

The Lagrangian viewpoint consists of following the material particles of the continuum in their motion. To this end, one introduces, as suggested in Figure 3, a computational grid, which follows the continuum in its motion, the grid nodes being permanently connected to the same material points. The material coordinates, $\boldsymbol{X}$, allow us to identify the reference configuration, $R_{\boldsymbol{X}}$. The motion of the material points relates the material coordinates, $\boldsymbol{X}$, to the spatial ones, $\boldsymbol{x}$. It is defined by an application $\boldsymbol{\varphi}$ such that

$$\begin{aligned} \boldsymbol{\varphi}\colon R_{\boldsymbol{X}} \times [t_0, t_{\text{final}}[&\longrightarrow R_{\boldsymbol{x}} \times [t_0, t_{\text{final}}[\\ (\boldsymbol{X}, t) &\longmapsto \boldsymbol{\varphi}(\boldsymbol{X}, t) = (\boldsymbol{x}, t) \end{aligned} \tag{1}$$

which allows us to link $\boldsymbol{X}$ and $\boldsymbol{x}$ in time by the law of motion, namely

$$\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{X}, t), \qquad t = t \tag{2}$$

which explicitly states the particular nature of $\boldsymbol{\varphi}$: first, the spatial coordinates $\boldsymbol{x}$ depend both on the material particle, $\boldsymbol{X}$, and time t, and, second, physical time is measured by the same variable t in both material and spatial domains. For every fixed instant t, the mapping $\boldsymbol{\varphi}$ defines a configuration in the spatial domain. It is convenient to employ a matrix representation for the gradient of $\boldsymbol{\varphi}$,

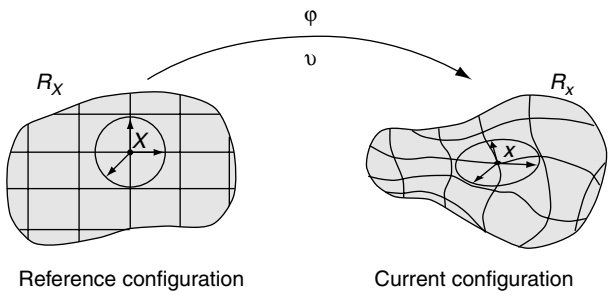

Figure 3. Lagrangian description of motion.

$$\frac{\partial \boldsymbol{\varphi}}{\partial(\boldsymbol{X}, t)} = \begin{pmatrix} \dfrac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}} & \boldsymbol{v} \\ \boldsymbol{0}^{\mathrm{T}} & 1 \end{pmatrix} \tag{3}$$

where $\boldsymbol{0}^{\mathrm{T}}$ is a null row-vector and the material velocity $\boldsymbol{v}$ is

$$\boldsymbol{v}(\boldsymbol{X}, t) = \left.\frac{\partial \boldsymbol{x}}{\partial t}\right|_{\boldsymbol{x}} \tag{4}$$

with $\left|_{\boldsymbol{x}}\right.$ meaning "holding the material coordinate $\boldsymbol{X}$ fixed".

Obviously, the one-to-one mapping $\boldsymbol{\varphi}$ must verify $\det(\partial \boldsymbol{x}/\partial \boldsymbol{X}) > 0$ (nonzero to impose a one-to-one correspondence and positive to avoid orientation change of the reference axes) at each point $\boldsymbol{X}$ and instant $t > t_0$. This allows us to keep track of the history of motion and, by the inverse transformation $(\boldsymbol{X}, t) = \boldsymbol{\varphi}^{-1}(\boldsymbol{x}, t)$, to identify, at any instant, the initial position of the material particle occupying position $\boldsymbol{x}$ at time t.

Since the material points coincide with the same grid points during the whole motion, there are no convective effects in Lagrangian calculations: the material derivative reduces to a simple time derivative. The fact that each finite element of a Lagrangian mesh always contains the same material particles represents a significant advantage from the computational viewpoint, especially in problems involving materials with history-dependent behavior. This aspect is discussed in detail by Bonet and Wood (1997). However, when large material deformations do occur, for instance vortices in fluids, Lagrangian algorithms undergo a loss in accuracy, and may even be unable to conclude a calculation, due to excessive distortions of the computational mesh linked to the material.

The difficulties caused by an excessive distortion of the finite element grid are overcome in the Eulerian formulation. The basic idea in the Eulerian formulation, which is very popular in fluid mechanics, consists in examining, as time evolves, the physical quantities associated with the fluid particles passing through a fixed region of space. In an Eulerian description, the finite element mesh is thus fixed and the continuum moves and deforms with respect to the computational grid. The conservation equations are formulated in terms of the spatial coordinates $\boldsymbol{x}$ and the time t. Therefore, the Eulerian description of motion only involves variables and functions having an instantaneous significance in a fixed region of space. The material velocity $\boldsymbol{v}$ at a given mesh node corresponds to the velocity of the material point coincident at the considered time t with the considered node. The velocity $\boldsymbol{v}$ is consequently expressed with respect to the fixed-element mesh without any reference to the initial configuration of the continuum and the material coordinates $\boldsymbol{X}$: $\boldsymbol{v} = \boldsymbol{v}(\boldsymbol{x}, t)$.

Since the Eulerian formulation dissociates the mesh nodes from the material particles, convective effects appear because of the relative motion between the deforming material and the computational grid. Eulerian algorithms present numerical difficulties due to the nonsymmetric character of convection operators, but permit an easy treatment of complex material motion. By contrast with the Lagrangian description, serious difficulties are now found in following deforming material interfaces and mobile boundaries.

2.2 ALE kinematical description

The above reminder of the classical Lagrangian and Eulerian descriptions has highlighted the advantages and drawbacks of each individual formulation. It has also shown the potential interest in a generalized description capable of combining at best the interesting aspects of the classical mesh descriptions while minimizing their drawbacks as far as possible. Such a generalized description is termed *arbitrary Lagrangian–Eulerian* (ALE) description. ALE methods were first proposed in the finite difference and finite volume context. Original developments were made, among others, by Noh (1964), Franck and Lazarus (1964), Trulio (1966), and Hirt *et al.* (1974); this last contribution has been reprinted in 1997. The method was subsequently adopted in the finite element context and early applications are to be found in the work of Donea *et al.* (1977), Belytschko *et al.* (1978), Belytschko and Kennedy (1978), and Hughes *et al.* (1981).

In the ALE description of motion, neither the material configuration R_X nor the spatial configuration R_x is taken as the reference. Thus, a third domain is needed: the referential configuration $R_{\boldsymbol{\chi}}$ where reference coordinates $\boldsymbol{\chi}$ are introduced to identify the grid points. Figure 4 shows these domains and the one-to-one transformations relating the configurations. The referential domain $R_{\boldsymbol{\chi}}$ is mapped into the material and spatial domains by $\boldsymbol{\Psi}$ and $\boldsymbol{\Phi}$ respectively. The particle motion $\boldsymbol{\varphi}$ may then be expressed as $\boldsymbol{\varphi} = \boldsymbol{\Phi} \circ \boldsymbol{\Psi}^{-1}$, clearly showing that, of course, the three mappings $\boldsymbol{\Psi}$, $\boldsymbol{\Phi}$, and $\boldsymbol{\varphi}$ are not independent.

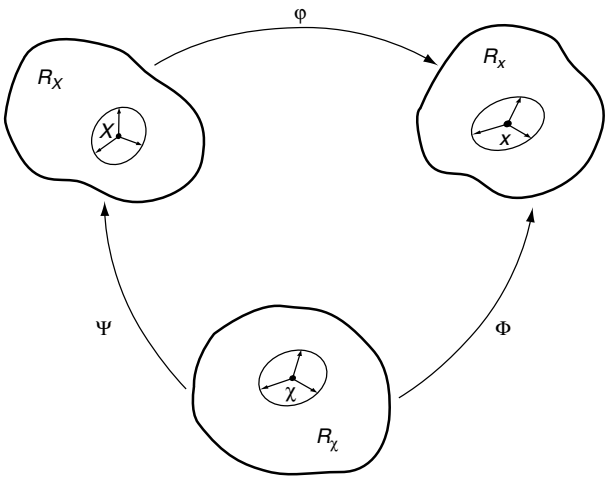

Figure 4. The motion of the ALE computational mesh is independent of the material motion.

The mapping of $\boldsymbol{\Phi}$ from the referential domain to the spatial domain, which can be understood as the motion of the grid points in the spatial domain, is represented by

$$\begin{aligned} \boldsymbol{\Phi}\colon R_{\boldsymbol{\chi}} \times [t_0, t_{\text{final}}[&\longrightarrow R_x \times [t_0, t_{\text{final}}[\\ (\boldsymbol{\chi}, t) &\longmapsto \boldsymbol{\Phi}(\boldsymbol{\chi}, t) = (\boldsymbol{x}, t) \end{aligned} \tag{5}$$

and its gradient is

$$\frac{\partial \boldsymbol{\Phi}}{\partial(\boldsymbol{\chi}, t)} = \begin{pmatrix} \dfrac{\partial \boldsymbol{x}}{\partial \boldsymbol{\chi}} & \hat{\boldsymbol{v}} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{pmatrix} \tag{6}$$

where now, the mesh velocity

$$\hat{\boldsymbol{v}}(\boldsymbol{\chi}, t) = \left.\frac{\partial \boldsymbol{x}}{\partial t}\right|_{\boldsymbol{\chi}} \tag{7}$$

is involved. Note that both the material and the mesh move with respect to the laboratory. Thus, the corresponding material and mesh velocities have been defined by deriving the equations of material motion and mesh motion respectively with respect to time (see equations 4 and 7).

Finally, regarding $\boldsymbol{\Psi}$, it is convenient to represent directly its inverse $\boldsymbol{\Psi}^{-1}$,

$$\begin{aligned} \boldsymbol{\Psi}^{-1}\colon R_X \times [t_0, t_{\text{final}}[&\longrightarrow R_{\boldsymbol{\chi}} \times [t_0, t_{\text{final}}[\\ (\boldsymbol{X}, t) &\longmapsto \boldsymbol{\Psi}^{-1}(\boldsymbol{X}, t) = (\boldsymbol{\chi}, t) \end{aligned} \tag{8}$$

and its gradient is

$$\frac{\partial \boldsymbol{\Psi}^{-1}}{\partial(\boldsymbol{X}, t)} = \begin{pmatrix} \dfrac{\partial \boldsymbol{\chi}}{\partial \boldsymbol{X}} & \boldsymbol{w} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{pmatrix} \tag{9}$$

where the velocity $\boldsymbol{w}$ is defined as

$$\boldsymbol{w} = \left.\frac{\partial \boldsymbol{\chi}}{\partial t}\right|_{X} \tag{10}$$

and can be interpreted as the particle velocity in the referential domain, since it measures the time variation of the referential coordinate $\boldsymbol{\chi}$ holding the material particle $\boldsymbol{X}$ fixed. The relation between velocities $\boldsymbol{v}$, $\hat{\boldsymbol{v}}$, and $\boldsymbol{w}$ can be obtained by differentiating $\boldsymbol{\varphi} = \boldsymbol{\Phi} \circ \boldsymbol{\Psi}^{-1}$,

$$\begin{aligned} \frac{\partial \boldsymbol{\varphi}}{\partial(\boldsymbol{X}, t)}(\boldsymbol{X}, t) &= \frac{\partial \boldsymbol{\Phi}}{\partial(\boldsymbol{\chi}, t)}\left(\boldsymbol{\Psi}^{-1}(\boldsymbol{X}, t)\right) \quad \frac{\partial \boldsymbol{\Psi}^{-1}}{\partial(\boldsymbol{X}, t)}(\boldsymbol{X}, t) \\ &= \frac{\partial \boldsymbol{\Phi}}{\partial(\boldsymbol{\chi}, t)}(\boldsymbol{\chi}, t) \quad \frac{\partial \boldsymbol{\Psi}^{-1}}{\partial(\boldsymbol{X}, t)}(\boldsymbol{X}, t) \end{aligned} \tag{11}$$

or, in matrix format:

$$\begin{pmatrix} \dfrac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}} & \boldsymbol{v} \\ \boldsymbol{0}^{\mathrm{T}} & 1 \end{pmatrix} \begin{pmatrix} \dfrac{\partial \boldsymbol{x}}{\partial \boldsymbol{\chi}} & \hat{\boldsymbol{v}} \\ \boldsymbol{0}^{\mathrm{T}} & 1 \end{pmatrix} \begin{pmatrix} \dfrac{\partial \boldsymbol{\chi}}{\partial \boldsymbol{X}} & \boldsymbol{w} \\ \boldsymbol{0}^{\mathrm{T}} & 1 \end{pmatrix} \tag{12}$$

which yields, after block multiplication,

$$\boldsymbol{v} = \hat{\boldsymbol{v}} + \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\chi}} \cdot \boldsymbol{w} \tag{13}$$

This equation may be rewritten as

$$\boldsymbol{c} := \boldsymbol{v} - \hat{\boldsymbol{v}} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\chi}} \cdot \boldsymbol{w} \tag{14}$$

thus defining the convective velocity $\boldsymbol{c}$, that is, the relative velocity between the material and the mesh.

The convective velocity $\boldsymbol{c}$ (see equation 14), should not be confused with $\boldsymbol{w}$ (see equation 10). As stated before, $\boldsymbol{w}$ is the particle velocity as seen from the referential domain $R_{\boldsymbol{\chi}}$, whereas $\boldsymbol{c}$ is the particle velocity relative to the mesh as seen from the spatial domain $R_{\boldsymbol{x}}$ (both $\boldsymbol{v}$ and $\hat{\boldsymbol{v}}$ are variations of coordinate $\boldsymbol{x}$). In fact, equation (14) implies that $\boldsymbol{c} = \boldsymbol{w}$ if and only if $\partial \boldsymbol{x}/\partial \boldsymbol{\chi} = \boldsymbol{I}$ (where $\boldsymbol{I}$ is the identity tensor), that is, when the mesh motion is purely translational, without rotations or deformations of any kind.

After the fundamentals on ALE kinematics have been presented, it should be remarked that both Lagrangian or Eulerian formulations may be obtained as particular cases. With the choice $\boldsymbol{\Psi} = \boldsymbol{I}$, equation (3) reduces to $\boldsymbol{X} \equiv \boldsymbol{\chi}$ and a Lagrangian description results: the material and mesh velocities, equations (4) and (7), coincide, and the convective velocity $\boldsymbol{c}$ (see equation 14), is null (there are no convective terms in the conservation laws). If, on the other hand, $\boldsymbol{\Phi} = \boldsymbol{I}$, equation (2) simplifies into $\boldsymbol{x} \equiv \boldsymbol{\chi}$, thus implying a Eulerian description: a null mesh velocity is obtained from equation (7) and the convective velocity $\boldsymbol{c}$ is simply identical to the material velocity $\boldsymbol{v}$.

In the ALE formulation, the freedom of moving the mesh is very attractive. It helps to combine the respective advantages of the Lagrangian and Eulerian formulations. This could, however, be overshadowed by the burden of specifying grid velocities well suited to the particular problem under consideration. As a consequence, the practical implementation of the ALE description requires that an automatic mesh-displacement prescription algorithm be supplied.

3 THE FUNDAMENTAL ALE EQUATION

In order to express the conservation laws for mass, momentum, and energy in an ALE framework, a relation between material (or total) time derivative, which is inherent in conservation laws, and referential time derivative is needed.

3.1 Material, spatial, and referential time derivatives

In order to relate the time derivative in the material, spatial, and referential domains, let a scalar physical quantity be described by $f(\boldsymbol{x}, t)$, $f^*(\boldsymbol{\chi}, t)$, and $f^{**}(\boldsymbol{X}, t)$ in the spatial, referential, and material domains respectively. Stars are employed to emphasize that the functional forms are, in general, different.

Since the particle motion $\boldsymbol{\varphi}$ is a mapping, the spatial description $f(\boldsymbol{x}, t)$, and the material description $f^{**}(\boldsymbol{X}, t)$ of the physical quantity can be related as

$$f^{**}(\boldsymbol{X}, t) = f\left(\boldsymbol{\varphi}(\boldsymbol{X}, t), t\right) \quad \text{or} \quad f^{**} = f \circ \boldsymbol{\varphi} \tag{15}$$

The gradient of this expression can be easily computed as

$$\frac{\partial f^{**}}{\partial (\boldsymbol{X}, t)}(\boldsymbol{X}, t) = \frac{\partial f}{\partial (\boldsymbol{x}, t)}(\boldsymbol{x}, t) \quad \frac{\partial \boldsymbol{\varphi}}{\partial (\boldsymbol{X}, t)}(\boldsymbol{X}, t) \tag{16}$$

which is amenable to the matrix form

$$\begin{pmatrix} \dfrac{\partial f^{**}}{\partial \boldsymbol{X}} & \dfrac{\partial f^{**}}{\partial t} \end{pmatrix} = \begin{pmatrix} \dfrac{\partial f}{\partial \boldsymbol{x}} & \dfrac{\partial f}{\partial t} \end{pmatrix} \begin{pmatrix} \dfrac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}} & \boldsymbol{v} \\ \boldsymbol{0}^{\mathrm{T}} & 1 \end{pmatrix} \tag{17}$$

which renders, after block multiplication, a first expression, which is obvious, that is, $(\partial f^{**}/\partial \boldsymbol{X}) = (\partial f/\partial \boldsymbol{x})(\partial \boldsymbol{x}/\partial \boldsymbol{X})$; however, the second one is more interesting:

$$\frac{\partial f^{**}}{\partial t} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial \boldsymbol{x}} \cdot \boldsymbol{v} \tag{18}$$

Note that this is the well-known equation that relates the material and the spatial time derivatives. Dropping the stars to ease the notation, this relation is finally cast as

$$\left.\frac{\partial f}{\partial t}\right|_X = \left.\frac{\partial f}{\partial t}\right|_x + \boldsymbol{v} \cdot \nabla f \qquad \text{or} \qquad \frac{\mathrm{d} f}{\mathrm{d} t} = \frac{\partial f}{\partial t} + \boldsymbol{v} \cdot \nabla f \tag{19}$$

which can be interpreted in the usual way: the variation of a physical quantity for a given particle $\boldsymbol{X}$ is the local variation plus a convective term taking into account the relative motion between the material and spatial (laboratory) systems. Moreover, in order not to overload the rest of the text with notation, except for the specific sections, the material time derivative is denoted as

$$\frac{\mathrm{d} \cdot}{\mathrm{d} t} := \left.\frac{\partial \cdot}{\partial t}\right|_X \tag{20}$$

and the spatial time derivative as

$$\frac{\partial \cdot}{\partial t} := \left.\frac{\partial \cdot}{\partial t}\right|_x \tag{21}$$

The relation between material and spatial time derivatives is now extended to include the referential time derivative.

With the help of mapping $\boldsymbol{\Psi}$, the transformation from the referential description $f^*(\boldsymbol{\chi}, t)$ of the scalar physical quantity to the material description $f^{**}(\boldsymbol{X}, t)$ can be written as

$$f^{**} = f^* \circ \boldsymbol{\Psi}^{-1} \tag{22}$$

and its gradient can be easily computed as

$$\frac{\partial f^{**}}{\partial(\boldsymbol{X}, t)}(\boldsymbol{X}, t) = \frac{\partial f^*}{\partial(\boldsymbol{\chi}, t)}(\boldsymbol{\chi}, t) \quad \frac{\partial \boldsymbol{\Psi}^{-1}}{\partial(\boldsymbol{X}, t)}(\boldsymbol{X}, t) \tag{23}$$

or, in matrix form

$$\begin{pmatrix} \dfrac{\partial f^{**}}{\partial \boldsymbol{X}} & \dfrac{\partial f^{**}}{\partial t} \end{pmatrix} = \begin{pmatrix} \dfrac{\partial f^*}{\partial \boldsymbol{\chi}} & \dfrac{\partial f^*}{\partial t} \end{pmatrix} \begin{pmatrix} \dfrac{\partial \boldsymbol{\chi}}{\partial \boldsymbol{X}} & \boldsymbol{w} \\ \boldsymbol{0}^{\mathrm{T}} & 1 \end{pmatrix} \tag{24}$$

which renders, after block multiplication,

$$\frac{\partial f^{**}}{\partial t} = \frac{\partial f^*}{\partial t} + \frac{\partial f^*}{\partial \boldsymbol{\chi}} \cdot \boldsymbol{w} \tag{25}$$

Note that this equation relates the material and the referential time derivatives. However, it also requires the evaluation of the gradient of the considered quantity in the referential domain. This can be done, but in computational mechanics it is usually easier to work in the spatial (or material) domain. Moreover, in fluids, constitutive relations are naturally expressed in the spatial configuration and the Cauchy stress tensor, which will be introduced next, is the natural measure for stresses. Thus, using the definition of $\boldsymbol{w}$ given in equation (14), the previous equation may be rearranged into

$$\frac{\partial f^{**}}{\partial t} = \frac{\partial f^*}{\partial t} + \frac{\partial f}{\partial \boldsymbol{x}} \cdot \boldsymbol{c} \tag{26}$$

The fundamental ALE relation between material time derivatives, referential time derivatives, and spatial gradient is finally cast as (stars dropped)

$$\left.\frac{\partial f}{\partial t}\right|_X = \left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{\chi}} + \frac{\partial f}{\partial \boldsymbol{x}} \cdot \boldsymbol{c} = \left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{\chi}} + \boldsymbol{c} \cdot \nabla f \tag{27}$$

and shows that the time derivative of the physical quantity f for a given particle $\boldsymbol{X}$, that is, its material derivative, is its local derivative (with the reference coordinate $\boldsymbol{\chi}$ held fixed) plus a convective term taking into account the relative velocity $\boldsymbol{c}$ between the material and the reference system. This equation is equivalent to equation (19) but in the ALE formulation, that is, when $(\boldsymbol{\chi}, t)$ is the reference.

3.2 Time derivative of integrals over moving volumes

To establish the integral form of the basic conservation laws for mass, momentum, and energy, we also need to consider the rate of change of integrals of scalar and vector functions over a moving volume occupied by fluid.

Consider thus a material volume V_t bounded by a smooth closed surface S_t whose points at time t move with the material velocity $\boldsymbol{v} = \boldsymbol{v}(\boldsymbol{x}, t)$ where $\boldsymbol{x} \in S_t$. A material volume is a volume that permanently contains the same particles of the continuum under consideration. The material time derivative of the integral of a scalar function $f(\boldsymbol{x}, t)$ (note that f is defined in the spatial domain) over the time-varying material volume V_t is given by the following well-known expression, often referred to as Reynolds transport theorem (see, for instance, (Belytschko *et al.* 2000) for a detailed proof):

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{V_t} f(\boldsymbol{x}, t)\,\mathrm{d}V = \int_{V_c \equiv V_t} \frac{\partial f(\boldsymbol{x}, t)}{\partial t}\,\mathrm{d}V + \int_{S_c \equiv S_t} f(\boldsymbol{x}, t)\,\boldsymbol{v} \cdot \boldsymbol{n}\,\mathrm{d}S \tag{28}$$

which holds for smooth functions $f(\boldsymbol{x}, t)$. The volume integral in the right-hand side is defined over a control volume V_c (fixed in space), which coincides with the moving material volume V_t at the considered instant, t, in time. Similarly, the fixed control surface S_c coincides at time t with the closed surface S_t bounding the material volume V_t. In the surface integral, $\boldsymbol{n}$ denotes the unit outward normal to the surface S_t at time t, and $\boldsymbol{v}$ is the material velocity of points of the boundary S_t. The first term in the right-hand side of expression (28) is the *local time derivative* of the volume integral. The boundary integral represents the flux of the scalar quantity f across the fixed boundary of the control volume $V_c \equiv V_t$.

Noting that

$$\int_{S_c} f(\boldsymbol{x}, t)\,\boldsymbol{v} \cdot \boldsymbol{n}\,\mathrm{d}S = \int_{V_c} \nabla \cdot (f\boldsymbol{v})\,\mathrm{d}V \tag{29}$$

one obtains the alternative form of Reynolds transport theorem:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{V_t} f(\boldsymbol{x}, t)\,\mathrm{d}V = \int_{V_c \equiv V_t} \left(\frac{\partial f(\boldsymbol{x}, t)}{\partial t} + \nabla \cdot (f\boldsymbol{v})\right)\mathrm{d}V \tag{30}$$

Similar forms hold for the material derivative of the volume integral of a vector quantity. Analogous formulae can be developed in the ALE context, that is, with a referential time derivative. In this case, however, the characterizing

velocity is no longer the material velocity $\boldsymbol{v}$, but the grid velocity $\hat{\boldsymbol{v}}$.

4 ALE FORM OF CONSERVATION EQUATIONS

To serve as an introduction to the discussion of ALE finite element and finite volume models, we establish in this section the differential and integral forms of the conservation equations for mass, momentum, and energy.

4.1 Differential forms

The ALE differential form of the conservation equations for mass, momentum, and energy are readily obtained from the corresponding well-known Eulerian forms

$$\begin{aligned}
&\text{Mass:} && \frac{\mathrm{d}\rho}{\mathrm{d}t} = \frac{\partial \rho}{\partial t}\Big|_{\boldsymbol{x}} + \boldsymbol{v}\cdot\nabla\rho = -\rho\nabla\cdot\boldsymbol{v}\\
&\text{Momentum:} && \rho\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = \rho\left(\frac{\partial \boldsymbol{v}}{\partial t}\Big|_{\boldsymbol{x}} + (\boldsymbol{v}\cdot\nabla)\boldsymbol{v}\right) = \nabla\cdot\boldsymbol{\sigma} + \rho\boldsymbol{b}\\
&\text{Energy:} && \rho\frac{\mathrm{d}E}{\mathrm{d}t} = \rho\left(\frac{\partial E}{\partial t}\Big|_{\boldsymbol{x}} + \boldsymbol{v}\cdot\nabla E\right)\\
& && \quad = \nabla\cdot(\boldsymbol{\sigma}\cdot\boldsymbol{v}) + \boldsymbol{v}\cdot\rho\boldsymbol{b} \qquad (31)
\end{aligned}$$

where ρ is the mass density, $\boldsymbol{v}$ is the material velocity vector, $\boldsymbol{\sigma}$ denotes the Cauchy stress tensor, $\boldsymbol{b}$ is the specific body force vector, and E is the specific total energy. Only mechanical energies are considered in the above form of the energy equation. Note that the stress term in the same equation can be rewritten in the form

$$\begin{aligned}
\nabla\cdot(\boldsymbol{\sigma}\cdot\boldsymbol{v}) &= \frac{\partial}{\partial x_i}(\sigma_{ij}\,v_j) = \frac{\partial \sigma_{ij}}{\partial x_i}\,v_j + \sigma_{ij}\,\frac{\partial v_j}{\partial x_i}\\
&= (\nabla\cdot\boldsymbol{\sigma})\cdot\boldsymbol{v} + \boldsymbol{\sigma}{:}\nabla\boldsymbol{v} \qquad (32)
\end{aligned}$$

where $\nabla\boldsymbol{v}$ is the spatial velocity gradient.

Also frequently used is the balance equation for the internal energy

$$\rho\,\frac{\mathrm{d}e}{\mathrm{d}t} = \rho\left(\frac{\partial e}{\partial t}\Big|_{\boldsymbol{x}} + \boldsymbol{v}\cdot\nabla e\right) = \boldsymbol{\sigma}{:}\nabla^{\mathrm{S}}\boldsymbol{v} \qquad (33)$$

where e is the specific internal energy and $\nabla^{\mathrm{S}}\boldsymbol{v}$ denotes the stretching (or strain rate) tensor, the symmetric part of the velocity gradient $\nabla\boldsymbol{v}$; that is, $\nabla^{\mathrm{S}}\boldsymbol{v} = (1/2)(\nabla\boldsymbol{v} + \nabla^{\mathrm{T}}\boldsymbol{v})$.

All one has to do to obtain the ALE form of the above conservation equations is to replace in the various convective terms, the material velocity $\boldsymbol{v}$ with the convective velocity $\boldsymbol{c} = \boldsymbol{v} - \hat{\boldsymbol{v}}$. The result is

$$\begin{aligned}
&\text{Mass:} && \frac{\partial \rho}{\partial t}\Big|_{\boldsymbol{\chi}} + \boldsymbol{c}\cdot\nabla\rho = -\rho\,\nabla\cdot\boldsymbol{v}\\
&\text{Momentum:} && \rho\left(\frac{\partial \boldsymbol{v}}{\partial t}\Big|_{\boldsymbol{\chi}} + (\boldsymbol{c}\cdot\nabla)\boldsymbol{v}\right) = \nabla\cdot\boldsymbol{\sigma} + \rho\boldsymbol{b}\\
&\text{Total energy:} && \rho\left(\frac{\partial E}{\partial t}\Big|_{\boldsymbol{\chi}} + \boldsymbol{c}\cdot\nabla E\right) = \nabla\cdot(\boldsymbol{\sigma}\cdot\boldsymbol{v}) + \boldsymbol{v}\cdot\rho\boldsymbol{b}\\
&\text{Internal energy:} && \rho\left(\frac{\partial e}{\partial t}\Big|_{\boldsymbol{\chi}} + \boldsymbol{c}\cdot\nabla e\right) = \boldsymbol{\sigma}{:}\nabla^{\mathrm{S}}\boldsymbol{v}. \qquad (34)
\end{aligned}$$

It is important to note that the right-hand side of equation (34) is written in classical Eulerian (spatial) form, while the arbitrary motion of the computational mesh is only reflected in the left-hand side. The origin of equations (34) and their similarity with the Eulerian equations (31) have induced some authors to name this method the *quasi-Eulerian* description; see, for instance, (Belytschko *et al.*, 1980).

Remark *(Material acceleration)* Mesh acceleration plays no role in the ALE formulation, so, only the material acceleration $\boldsymbol{a}$, the material derivative of velocity $\boldsymbol{v}$, is needed, which is expressed in the Lagrangian, Eulerian, and ALE formulation respectively as

$$\boldsymbol{a} = \frac{\partial \boldsymbol{v}}{\partial t}\Big|_{X} \qquad (35a)$$

$$\boldsymbol{a} = \frac{\partial \boldsymbol{v}}{\partial t}\Big|_{\boldsymbol{x}} + \boldsymbol{v}\frac{\partial \boldsymbol{v}}{\partial \boldsymbol{x}} \qquad (35b)$$

$$\boldsymbol{a} = \frac{\partial \boldsymbol{v}}{\partial t}\Big|_{\boldsymbol{\chi}} + \boldsymbol{c}\frac{\partial \boldsymbol{v}}{\partial \boldsymbol{x}} \qquad (35c)$$

Note that the ALE expression of acceleration (35c) is simply a particularization of the fundamental relation (27), taking the material velocity $\boldsymbol{v}$ as the physical quantity f. The first term in the right-hand side of relationships (35b) and (35c) represents the local acceleration, the second term being the convective acceleration.

4.2 Integral forms

The starting point for deriving the ALE integral form of the conservation equations is Reynolds transport theorem (28) applied to an *arbitrary* volume V_t whose boundary $S_t = \partial V_t$ moves with the mesh velocity $\hat{\boldsymbol{v}}$:

$$\begin{aligned}
\frac{\partial}{\partial t}\Big|_{\boldsymbol{\chi}}\int_{V_t} f(\boldsymbol{x},t)\,\mathrm{d}V &= \int_{V_t}\frac{\partial f(\boldsymbol{x},t)}{\partial t}\Big|_{\boldsymbol{x}}\,\mathrm{d}V\\
&\quad + \int_{S_t} f(\boldsymbol{x},t)\,\hat{\boldsymbol{v}}\cdot\boldsymbol{n}\,\mathrm{d}S \qquad (36)
\end{aligned}$$

where, in this case, we have explicitly indicated that the time derivative in the first term of the right-hand side is a spatial time derivative, as in expression (28). We then successively replace the scalar $f(\boldsymbol{x}, t)$ by the fluid density ρ, momentum $\rho\boldsymbol{v}$, and specific total energy ρE. Similarly, the spatial time derivative $\partial f/\partial t$ is substituted with expressions (31) for the mass, momentum, and energy equation. The end result is the following set of ALE integral forms:

$$\begin{aligned}
&\frac{\partial}{\partial t}\Big|_{\boldsymbol{\chi}} \int_{V_t} \rho \, \mathrm{d}V + \int_{S_t} \rho \, \boldsymbol{c} \cdot \boldsymbol{n} \, \mathrm{d}S = 0 \\
&\frac{\partial}{\partial t}\Big|_{\boldsymbol{\chi}} \int_{V_t} \rho \boldsymbol{v} \, \mathrm{d}V + \int_{S_t} \rho \boldsymbol{v} \, \boldsymbol{c} \cdot \boldsymbol{n} \, \mathrm{d}S = \int_{V_t} (\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} + \rho \boldsymbol{b}) \, \mathrm{d}V \\
&\frac{\partial}{\partial t}\Big|_{\boldsymbol{\chi}} \int_{V_t} \rho E \, \mathrm{d}V + \int_{S_t} \rho E \, \boldsymbol{c} \cdot \boldsymbol{n} \, \mathrm{d}S \\
&\qquad = \int_{V_t} (\boldsymbol{v} \cdot \rho \boldsymbol{b} + \boldsymbol{\nabla} \cdot (\boldsymbol{\sigma} \cdot \boldsymbol{v})) \, \mathrm{d}V
\end{aligned} \tag{37}$$

Note that the integral forms for the Lagrangian and Eulerian mesh descriptions are contained in the above ALE forms. The Lagrangian description corresponds to selecting $\hat{\boldsymbol{v}} = \boldsymbol{v}$ $(\boldsymbol{c} = \boldsymbol{0})$, while the Eulerian description corresponds to selecting $\hat{\boldsymbol{v}} = \boldsymbol{0}$ $(\boldsymbol{c} = \boldsymbol{v})$.

The ALE differential and integral forms of the conservation equations derived in the present section will be used as a basis for the spatial discretization of problems in fluid dynamics and solid mechanics.

5 MESH-UPDATE PROCEDURES

The majority of modern ALE computer codes are based on either finite volume or finite element spatial discretizations, the former being popular in the fluid mechanics area, the latter being generally preferred in solid and structural mechanics. Note, however, that the ALE methodology is also used in connection with so-called mesh-free methods (see, for instance, (Ponthot and Belytschko, 1998) for an application of the element-free Galerkin method to dynamic fracture problems). In the remainder of this chapter, reference will mainly be made to spatial discretizations produced by the finite element method.

As already seen, one of the main advantages of the ALE formulation is that it represents a very versatile combination of the classical Lagrangian and Eulerian descriptions. However, the computer implementation of the ALE technique requires the formulation of a mesh-update procedure that assigns mesh-node velocities or displacements at each station (time step, or load step) of a calculation. The mesh-update strategy can, in principle, be chosen by the user. However, the remesh algorithm strongly influences the success of the ALE technique and may represent a big burden on the user if it is not rendered automatic.

Two basic mesh-update strategies may be identified. On one hand, the geometrical concept of *mesh regularization* can be exploited to keep the computational mesh as regular as possible and to avoid mesh entanglement during the calculation. On the other hand, if the ALE approach is used as a *mesh-adaptation* technique, for instance, to concentrate elements in zones of steep solution gradient, a suitable indication of the error is required as a basic input to the remesh algorithm.

5.1 Mesh regularization

The objective of mesh regularization is of a geometrical nature. It consists in keeping the computational mesh as regular as possible during the whole calculation, thereby avoiding excessive distortions and squeezing of the computing zones and preventing mesh entanglement. Of course, this procedure decreases the numerical errors due to mesh distortion.

Mesh regularization requires that updated nodal coordinates be specified at each station of a calculation, either through step displacements, or from current mesh velocities $\hat{\boldsymbol{v}}$. Alternatively, when it is preferable to prescribe the relative motion between the mesh and the material particles, the *referential velocity* $\boldsymbol{w}$ is specified. In this case, $\hat{\boldsymbol{v}}$ is deduced from equation (13). Usually, in fluid flows, the mesh velocity is interpolated, and in solid problems, the mesh displacement is directly interpolated.

First of all, these mesh-updating procedures are classified depending on whether the *boundary motion is prescribed a priori* or its motion is unknown.

When the motion of the material surfaces (usually the boundaries) is known *a priori*, the mesh motion is also prescribed *a priori*. This is done by defining an adequate mesh velocity in the domain, usually by simple interpolation. In general, this implies a Lagrangian description at the moving boundaries (the mesh motion coincides with the prescribed boundary motion), while a Eulerian formulation (fixed mesh velocity $\hat{\boldsymbol{v}} = \boldsymbol{0}$) is employed far away from the moving boundaries. A transition zone is defined in between. The interaction problem between a rigid body and a viscous fluid studied by Huerta and Liu (1988a) falls in this category. Similarly, the crack propagation problems discussed by Koh and Haber (1986) and Koh *et al.* (1988), where the crack path is known *a priori*, also allow the use of this kind of mesh-update procedure. Other examples of prescribed mesh motion in nonlinear solid mechanics can be found in the works by Liu *et al.* (1986), Huétink *et al.* (1990), van

Haaren *et al.* (2000), and Rodríguez-Ferran *et al.* (2002), among others.

In all other cases, at least a part of the boundary is a material surface whose position must be tracked at each time step. Thus, a Lagrangian description is prescribed along this surface (or at least along its normal). In the first applications to fluid dynamics (usually free surface flows), ALE degrees of freedom were simply divided into purely Lagrangian ($\hat{\boldsymbol{v}} = \boldsymbol{v}$) or purely Eulerian ($\hat{\boldsymbol{v}} = 0$). Of course, the distortion was thus concentrated in a layer of elements. This is, for instance, the case for numerical simulations reported by Noh (1964), Franck and Lazarus (1964), Hirt *et al.* (1974), and Pracht (1975). Nodes located on moving boundaries were Lagrangian, while internal nodes were Eulerian. This approach was used later for fluid–structure interaction problems by Liu and Chang (1984) and in solid mechanics by Haber (1984) and Haber and Hariandja (1985). This procedure was generalized by Hughes *et al.* (1981) using the so-called Lagrange–Euler matrix method. The referential velocity, $\boldsymbol{w}$, is defined relative to the particle velocity, $\boldsymbol{v}$, and the mesh velocity is determined from equation (13). Huerta and Liu (1988b) improved this method avoiding the need to solve any equation for the mesh velocity inside the domain and ensuring an accurate tracking of the material surfaces by solving $\boldsymbol{w} \cdot \boldsymbol{n} = 0$, where $\boldsymbol{n}$ is the unit outward normal, only along the material surfaces. Once the boundaries are known, mesh displacements or velocities inside the computational domain can in fact be prescribed through potential-type equations or interpolations as is discussed next.

In fluid–structure interaction problems, solid nodes are usually treated as Lagrangian, while fluid nodes are treated as described above (fixed or updated according to some simple interpolation scheme). Interface nodes between the solid and the fluid must generally be treated as described in Section 6.1.2. Occasionally they can be treated as Lagrangian (see, for instance, (Belytschko and Kennedy, 1978; Belytschko *et al.*, 1980, 1982; Belytschko and Liu, 1985); Argyris *et al.*, 1985; Huerta and Liu, 1988b).

Once the boundary motion is known, several interpolation techniques are available to determine the mesh rezoning in the interior of the domain.

5.1.1 Transfinite mapping method

This method was originally designed for creating a mesh on a geometric region with specified boundaries; see e.g. (Gordon and Hall, 1973; Haber and Abel, 1982; and Eriksson, 1985). The general transfinite method describes an approximate surface or volume at a nondenumerable number of points. It is this property that gives rise to the term *transfinite mapping*. In the 2-D case, the transfinite mapping can be made to exactly model all domain boundaries, and, thus, no geometric error is introduced by the mapping. It induces a very low-cost procedure, since new nodal coordinates can be obtained explicitly once the boundaries of the computational domain have been discretized. The main disadvantage of this methodology is that it imposes restrictions on the mesh topology, as two opposite curves have to be discretized with the same number of elements. It has been widely used by the ALE community to update nodal coordinates; see e.g. (Ponthot and Hogge, 1991; Yamada and Kikuchi, 1993; Gadala and Wang, 1998, 1999; and Gadala *et al.*, 2002).

5.1.2 Laplacian smoothing and variational methods

As in mesh generation or smoothing techniques, the rezoning of the mesh nodes consists in solving a Laplace (or Poisson) equation for each component of the node velocity or position, so that on a logically regular region the mesh forms lines of equal potential. This method is also sometimes called *elliptic mesh generation* and was originally proposed by Winslow (1963). This technique has an important drawback: in a nonconvex domain, nodes may run outside it. Techniques to preclude this pitfall either increase the computational cost enormously or introduce new terms in the formulation, which are particular to each geometry. Examples based on this type of mesh-update algorithms are presented, among others, by Benson (Benson, 1989; Benson, 1992a; Benson, 1992b), (Liu *et al.*, 1988, 1991), Ghosh and Kikuchi (1991), Chenot and Bellet (1995), and Löhner and Yang (1996). An equivalent approach based on a mechanical interpretation: (non)linear elasticity problem is used by Schreurs *et al.* (1986), Le Tallec and Martin (1996), Belytschko *et al.* (2000), and Armero and Love (2003), while Cescutti *et al.* (1988) minimize a functional quantifying the mesh distortion.

5.1.3 Mesh-smoothing and simple interpolations

In fact, in ALE, it is possible to use any mesh-smoothing algorithm designed to improve the shape of the elements once the topology is fixed. Simple iterative averaging procedures can be implemented where possible; see, for instance, (Donea *et al.*, 1982; Batina, 1991; Trépanier *et al.*, 1993; Ghosh and Raju, 1996; and Aymone *et al.*, 2001). A more robust algorithm (especially in the neighborhood of boundaries with large curvature) was proposed by Giuliani (1982) on the basis of geometric considerations. The goal of this method is to minimize both the squeeze and distortion of each element in the mesh. Donea (1983) and Huerta and Casadei (1994) show examples using this algorithm; Sarrate and Huerta (2001) and Hermansson and Hansbo (2003)

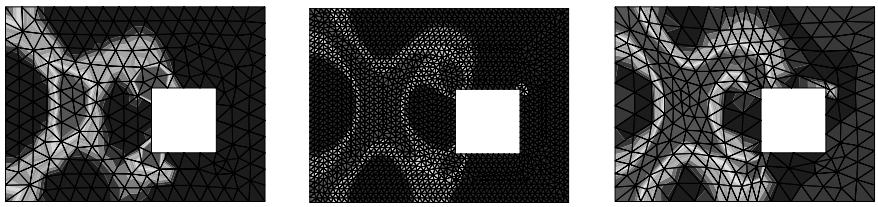

Figure 5. Use of the ALE formulation as an r-adaptive technique. The yield-line pattern is not properly captured with (a) a coarse fixed mesh. Either (b) a fine fixed mesh or (c) a coarse ALE mesh is required. A color version of this image is available at http://www.mrw.interscience.wiley.com/ecm

made improvements to the original procedure. The main advantage of these mesh-regularization methods is that they are both simple and rather general. They can in fact be applied to unstructured meshes consisting of triangular and quadrilateral elements in 2-D, and to tetrahedral, hexahedral, prism, and pyramidal elements in 3-D.

5.2 Mesh adaptation

When the ALE description is used as an adaptive technique, the objective is to optimize the computational mesh to achieve an improved accuracy, possibly at low computing cost (the total number of elements in a mesh remains unchanged throughout the computation, as well as the element connectivity). Mesh refinement is typically carried out by moving the nodes towards zones of strong solution gradient, such as localization zones in large deformation problems involving softening materials. The ALE algorithm then includes an indicator of the error, and the mesh is modified to obtain an equi-distribution of the error over the entire computational domain. The remesh indicator can, for instance, be made a function of the average or the jump of a certain state variable. Equi-distribution can be carried out using an elliptic or a parabolic differential equation. The ALE technique can nevertheless be coupled with traditional mesh-refinement procedures, such as h-adaptivity, to further enhance accuracy through the selective addition of new degrees of freedom (see (Askes and Rodríguez-Ferran, 2001)).

Consider, for instance, the use of the ALE formulation for the prediction of yield-line patterns in plates, see (Askes *et al.*, 1999). With a coarse fixed mesh, (Figure 5(a)), the spatial discretization is too poor and the yield-line pattern cannot be properly captured. One possible solution is, of course, to use a finer mesh (see Figure 5(b)). Another very attractive possibility from a computational viewpoint is to stay with the coarse mesh and use the ALE formulation to relocate the nodes (see Figure 5(c)). The level of plastification is used as the remesh indicator. Note that, in this problem, element distortion is not a concern (contrary to the typical situation illustrated in Figure 2); nodes are relocated to concentrate them along the yield lines.

Studies on the use of ALE as a mesh-adaptation technique in solid mechanics are reported, among others, by Pijaudier-Cabot *et al.* (1995), Huerta *et al.* (1999), Askes and Sluys (2000), Askes and Rodríguez-Ferran (2001), Askes *et al.* (2001), and Rodríguez-Ferran *et al.* (2002).

Mesh adaptation has also found widespread use in fluid dynamics. Often, account must be taken of the directional character of the flow, so anisotropic adaptation procedures are to be preferred. For example, an efficient adaptation method for viscous flows with strong shear layers has to be able to refine directionally to adapt the mesh to the anisotropy of the flow. Anisotropic adaptation criteria again have an error estimate as a basic criterion (see, for instance, (Fortin *et al.*, 1996; Castro-Diaz *et al.*, 1996; Ait-Ali-Yahia *et al.*, 2002; Habashi *et al.*, 2000; and Müller, 2002) for the practical implementation of such procedures).

6 ALE METHODS IN FLUID DYNAMICS

Owing to its superior ability with respect to the Eulerian description to deal with interfaces between materials and mobile boundaries, the ALE description is being widely used for the spatial discretization of problems in fluid and structural dynamics. In particular, the method is frequently employed in the so-called hydrocodes, which are used to simulate the large distortion/deformation response of materials, structures, fluids, and fluid–structure systems. They typically apply to problems in impact and penetration mechanics, fracture mechanics, and detonation/blast analysis. We shall briefly illustrate the specificities of ALE techniques in the modeling of viscous incompressible flows and in the simulation of inviscid, compressible flows, including interaction with deforming structures.

The most obvious influence of an ALE formulation in flow problems is that the convective term must account for the mesh motion. Thus, as already discussed in Section 4.1,

the convective velocity $\boldsymbol{c}$ replaces the material velocity $\boldsymbol{v}$, which appears in the convective term of Eulerian formulations (see equations 31 and 34). Note that the mesh motion may increase or decrease the convection effects. Obviously, in pure convection (for instance, if a fractional-step algorithm is employed) or when convection is dominant, stabilization techniques must be implemented. The interested reader is urged to consult **Chapter 2 of Volume 3** for a thorough exposition of stabilization techniques available to remedy the lack of stability of the standard Galerkin formulation in convection-dominated situations, or the textbook by Donea and Huerta (2003).

It is important to note that in standard fluid dynamics, the stress tensor only depends on the pressure and (for viscous flows) on the velocity field at the point and instant under consideration. This is not the case in solid mechanics, as discussed below in Section 7. Thus, stress update is not a major concern in ALE fluid dynamics.

6.1 Boundary conditions

The rest of the discussion of the specificities of the ALE formulation in fluid dynamics concerns boundary conditions. In fact, boundary conditions are related to the problem, not to the description employed. Thus, the same boundary conditions employed in Eulerian or Lagrangian descriptions are implemented in the ALE formulation, that is, along the boundary of the domain, kinematical and dynamical conditions must be defined. Usually, this is formalized as

$$\begin{cases} \boldsymbol{v} = \boldsymbol{v}_{\mathrm{D}} & \text{on } \Gamma_{\mathrm{D}} \\ \boldsymbol{n} \cdot \boldsymbol{\sigma} = \boldsymbol{t} & \text{on } \Gamma_{\mathrm{N}} \end{cases}$$

where $\boldsymbol{v}_{\mathrm{D}}$ and $\boldsymbol{t}$ are the prescribed boundary velocities and tractions respectively; $\boldsymbol{n}$ is the outward unit normal to Γ_{N}, and Γ_{D} and Γ_{N} are the two distinct subsets (Dirichlet and Neumann respectively), which define the piecewise smooth boundary of the computational domain. As usual, stress conditions on the boundaries represent the 'natural boundary conditions', and thus, they are automatically included in the weak form of the momentum conservation equation (see 34).

If part of the boundary is composed of a material surface whose position is unknown, then a mixture of both conditions is required. The ALE formulation allows an accurate treatment of material surfaces. The conditions required on a material surface are: (a) no particles can cross it, and (b) stresses must be continuous across the surface (if a net force is applied to a surface of zero mass, the acceleration is infinite). Two types of material surfaces are discussed here: free surfaces and fluid–structure interfaces, which may or may not be frictionless (whether or not the fluid is inviscid).

6.1.1 Free surfaces

The unknown position of free surfaces can be computed using two different approaches. First, for the simple case of a single-valued function $z = z(x, y, t)$, a hyperbolic equation must be solved,

$$\frac{\partial z}{\partial t} + (\boldsymbol{v} \cdot \boldsymbol{\nabla}) z = 0$$

This is the kinematic equation of the surface and has been used, for instance, by Ramaswamy and Kawahara (1987), Huerta and Liu, 1988b, 1990; Souli and Zolesio (2001). Second, a more general approach can be obtained by simply imposing the obvious condition that no particle can cross the free surface (because it is a material surface). This can be imposed in a straightforward manner by using a Lagrangian description (i.e. $\boldsymbol{w} = \boldsymbol{0}$ or $\boldsymbol{v} = \hat{\boldsymbol{v}}$) along this surface. However, this condition may be relaxed by imposing only the necessary condition: $\boldsymbol{w}$ equal to zero along the normal to the boundary (i.e. $\boldsymbol{n} \cdot \boldsymbol{w} = 0$, where $\boldsymbol{n}$ is the outward unit normal to the fluid domain, or $\boldsymbol{n} \cdot \boldsymbol{v} = \boldsymbol{n} \cdot \hat{\boldsymbol{v}}$). The mesh position, normal to the free surface, is determined from the normal component of the particle velocity and remeshing can be performed along the tangent; see, for instance (Huerta and Liu, 1989) or (Braess and Wriggers, 2000). In any case, these two alternatives correspond to the kinematical condition; the dynamic condition expresses the stress-free situation, $\boldsymbol{n} \cdot \boldsymbol{\sigma} = \boldsymbol{0}$, and since it is a homogeneous natural boundary condition, as mentioned earlier, it is directly taken into account by the weak formulation.

6.1.2 Fluid–structure interaction

Along solid-wall boundaries, the particle velocity is coupled to the rigid or flexible structure. The enforcement of the kinematic requirement that no particles can cross the interface is similar to the free-surface case. Thus, conditions $\boldsymbol{n} \cdot \boldsymbol{w} = 0$ or $\boldsymbol{n} \cdot \boldsymbol{v} = \boldsymbol{n} \cdot \hat{\boldsymbol{v}}$ are also used. However, due to the coupling between fluid and structure, extra conditions are needed to ensure that the fluid and structural domains will not detach or overlap during the motion. These coupling conditions depend on the fluid.

For an inviscid fluid (no shear effects), only normal components are coupled because an inviscid fluid is free to slip along the structural interface; that is,

$$\begin{cases} \boldsymbol{n} \cdot \boldsymbol{u} = \boldsymbol{n} \cdot \boldsymbol{u}_{\mathrm{S}} & \text{continuity of normal displacements} \\ \boldsymbol{n} \cdot \boldsymbol{v} = \boldsymbol{n} \cdot \boldsymbol{v}_{\mathrm{S}} & \text{continuity of normal velocities} \end{cases}$$

where the displacement/velocity of the fluid ($\boldsymbol{u}/\boldsymbol{v}$) along the normal to the interface must be equal to the displacement/velocity of the structure ($\boldsymbol{u}_S/\boldsymbol{v}_S$) along the same direction. Both equations are equivalent and one or the other is used, depending on the formulation employed (displacements or velocities).

For a viscous fluid, the coupling between fluid and structure requires that velocities (or displacements) coincide along the interface; that is,

$$\begin{cases} \boldsymbol{u} = \boldsymbol{u}_S & \text{continuity of displacements} \\ \boldsymbol{v} = \boldsymbol{v}_S & \text{continuity of velocities} \end{cases}$$

In practice, two nodes are placed at each point of the interface: one fluid node and one structural node. Since the fluid is treated in the ALE formulation, the movement of the fluid mesh may be chosen completely independent of the movement of the fluid itself. In particular, we may constrain the fluid nodes to remain contiguous to the structural nodes, so that all nodes on the sliding interface remain permanently aligned. This is achieved by prescribing the grid velocity $\hat{\boldsymbol{v}}$ of the fluid nodes at the interface to be equal to the material velocity $\boldsymbol{v}_S$ of the adjacent structural nodes. The permanent alignment of nodes at ALE interfaces greatly facilitates the flow of information between the fluid and structural domains and permits fluid–structure coupling to be effected in the simplest and the most elegant manner; that is, the imposition of the previous kinematic conditions is simple because of the node alignment.

The dynamic condition is automatically verified along fixed rigid boundaries, but it presents the classical difficulties in fluid–structure interaction problems when compatibility at nodal level in velocities and stresses is required (both for flexible or rigid structures whose motion is coupled to the fluid flow). This condition requires that the stresses in the fluid be equal to the stresses in the structure. When the behavior of the fluid is governed by the linear Stokes law ($\boldsymbol{\sigma} = -p\,\mathbf{I} + 2\nu\nabla^S\boldsymbol{v}$) or for inviscid fluids this condition is

$$-p\,\boldsymbol{n} + 2\nu(\boldsymbol{n}\cdot\nabla^S)\boldsymbol{v} = \boldsymbol{n}\cdot\boldsymbol{\sigma}_S \qquad \text{or} \qquad -p\,\boldsymbol{n} = \boldsymbol{n}\cdot\boldsymbol{\sigma}_S$$

respectively, where $\boldsymbol{\sigma}_S$ is the stress tensor acting on the structure. In the finite element representation, the continuous interface is replaced with a discrete approximation and instead of a distributed interaction pressure, consideration is given to its resultant at each interface node.

There is a large amount of literature on ALE fluid–structure interaction, both for flexible structures and for rigid solids; see, among others, (Liu and Chang, 1985; Liu and Gvildys, 1986; Nomura and Hughes, 1992; Le Tallec and Mouro, 2001; Casadei *et al.*, 2001; Sarrate *et al.*, 2001; and Zhang and Hisada, 2001).

Remark *(Fluid–rigid-body interaction)* In some circumstances, especially when the structure is embedded in a fluid and its deformations are small compared with the displacements and rotations of its center of gravity, it is justified to idealize the structure as a rigid body resting on a system consisting of springs and dashpots. Typical situations in which such an idealization is legitimate include the simulation of wind-induced vibrations in high-rise buildings or large bridge girders, the cyclic response of offshore structures exposed to sea currents, as well as the behavior of structures in aeronautical and naval engineering where structural loading and response are dominated by fluid-induced vibrations. An illustrative example of ALE fluid–rigid-body interaction is shown is Section 6.2.

Remark *(Normal to a discrete interface)* In practice, especially in complex 3-D configurations, one major difficulty is to determine the normal vector at each node of the fluid–structure interface. Various algorithms have been developed to deal with this issue; Casadei and Halleux (1995) and Casadei and Sala (1999) present detailed solutions. In 2-D, the tangent to the interface at a given node is usually defined as parallel to the line connecting the nodes at the ends of the interface segments meeting at that node.

Remark *(Free surface and structure interaction)* The above discussion of the coupling problem only applies to those portions of the structure that are always submerged during the calculation. As a matter of fact, there may exist portions of the structure, which only come into contact with the fluid some time after the calculation begins. This is, for instance, the case for structural parts above a fluid-free surface. For such portions of the structural domain, some sort of sliding treatment is necessary, as for Lagrangian methods.

6.1.3 Geometric conservation laws

In a series of papers, see (Lesoinne and Farhat, 1996; Koobus and Farhat, 1999; Guillard and Farhat, 2000; and Farhat *et al.*, 2001), Farhat and coworkers have discussed the notion of *geometric conservation laws* for unsteady flow computations on moving and deforming finite element or finite volume grids.

The basic requirement is that any ALE computational method should be able to predict exactly the trivial solution of a uniform flow. The ALE equation of mass balance $(37)_1$ is usually taken as the starting point to derive the geometric conservation law. Assuming uniform fields of density ρ and material velocity $\boldsymbol{v}$, it reduces to the *continuous geometric conservation law*

$$\frac{\partial}{\partial t}\Big|_{\boldsymbol{\chi}} \int_{V_t} \mathrm{d}V = \int_{S_t} \hat{\boldsymbol{v}}\cdot\boldsymbol{n}\,\mathrm{d}S \tag{38}$$

As remarked by Smith (1999), equation (38) can also be derived from the other two ALE integral conservation laws (37) with appropriate restrictions on the flow fields.

Integrating equation (38) in time from t^n to t^{n+1} renders the *discrete geometric conservation law (DGCL)*

$$|\Omega_e^{n+1}| - |\Omega_e^n| = \int_{t^n}^{t^{n+1}} \left(\int_{S_t} \hat{\boldsymbol{v}} \cdot \boldsymbol{n} \, \mathrm{d}S \right) \mathrm{d}t \qquad (39)$$

which states that the change in volume (or area, in 2-D) of each element from t^n to t^{n+1} must be equal to the volume (or area) swept by the element boundary during the time interval. Assuming that the volumes Ω_e in the left-hand side of equation (39) can be computed exactly, this amounts to requiring the exact computation of the flux in the right-hand side also. This poses some restrictions on the update procedure for the grid position and velocity. For instance, Lesoinne and Farhat (1996) show that, for first-order time-integration schemes, the mesh velocity should be computed as $\hat{\boldsymbol{v}}^{n+1/2} = (\boldsymbol{x}^{n+1} - \boldsymbol{x}^n)/\Delta t$. They also point out that, although this intuitive formula was used by many time-integrators prior to DGCLs, it is violated in some instances, especially in fluid–structure interaction problems where mesh motion is coupled with structural deformation.

The practical significance of DGCLs is a debated issue in the literature. As admitted by Guillard and Farhat (2000), 'there are recurrent assertions in the literature stating that, in practice, enforcing the DGCL when computing on moving meshes is unnecessary'. Later, Farhat *et al.* (2001) and other authors have studied the properties of DGCL-enforcing ALE schemes from a theoretical viewpoint. The link between DGCLs and the stability (and accuracy) of ALE schemes is still a controversial topic of current research.

6.2 Applications in ALE fluid dynamics

The first example consists in the computation of cross-flow and rotational oscillations of a rectangular profile. The flow is modeled by the incompressible Navier–Stokes equations and the rectangle is regarded as rigid. The ALE formulation for fluid–rigid-body interaction proposed by Sarrate *et al.* (2001) is used.

Figure 6 depicts the pressure field at two different instants. The flow goes from left to right. Note the cross-flow translation and the rotation of the rectangle. The ALE kinematical description avoids excessive mesh distortion (see Figure 7). For this problem, a computationally efficient rezoning strategy is obtained by dividing the mesh into three zones: (1) the mesh inside the inner circle is prescribed to

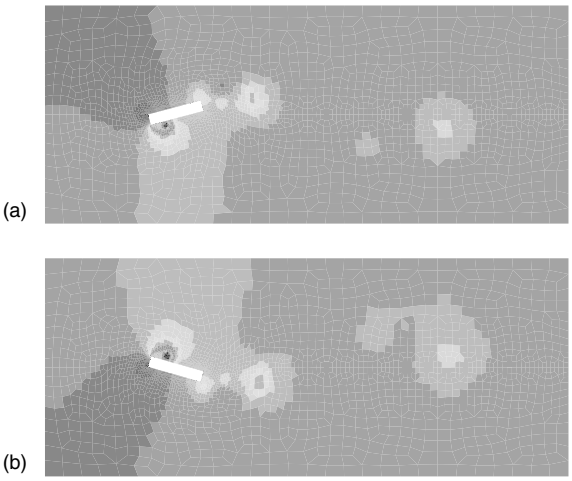

Figure 6. Flow around a rectangle. Pressure fields at two different instants. A color version of this image is available at http://www.mrw.interscience.wiley.com/ecm

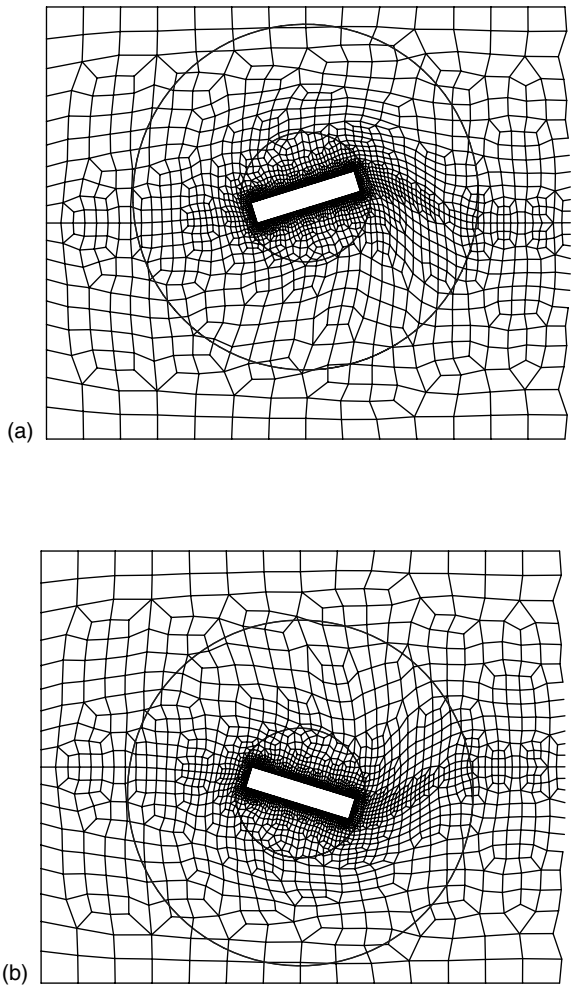

Figure 7. Details of finite element mesh around the rectangle. The ring allows a smooth transition between the rigidly moving mesh around the rectangle and the Eulerian mesh far from it.

move rigidly attached to the rectangle (no mesh distortion and simple treatment of interface conditions); (2) the mesh outside the outer circle is Eulerian (no mesh distortion and no need to select mesh velocity); (3) a smooth transition is prescribed in the ring between the circles (mesh distortion under control).

The second example highlights ALE capabilities for fluid–structure interaction problems. The results shown here, discussed in detail by Casadei and Potapov (2004), have been provided by Casadei and are reproduced here with the authors' kind permission. The example consists in a long 3-D metallic pipe with a square cross section, sealed at both ends, containing a gas at room pressure (see Figure 8). Initially, two 'explosions' take place at the ends of the pipe, simulated by the presence of the same gas, but at a much higher initial pressure.

The gas motion through the pipe is partly affected by internal structures within the pipe (diaphragms #1, #2 and #3) that create a sort of labyrinth. All the pipe walls, and the internal structures, are deformable and are characterized by elastoplastic behavior. The pressures and structural-material properties are so chosen that very large motions and relatively large deformations occur in the structure.

Figure 9 shows the real deformed shapes (not scaled up) of the pipe with superposed fluid-pressure maps. Note the strong wave-propagation effects, the partial wave reflections at obstacles, and the 'ballooning' effect of the thin pipe walls in regions at high pressure. This is a severe test, among other things, for the automatic ALE rezoning algorithms that must keep the fluid mesh reasonably uniform under large motions.

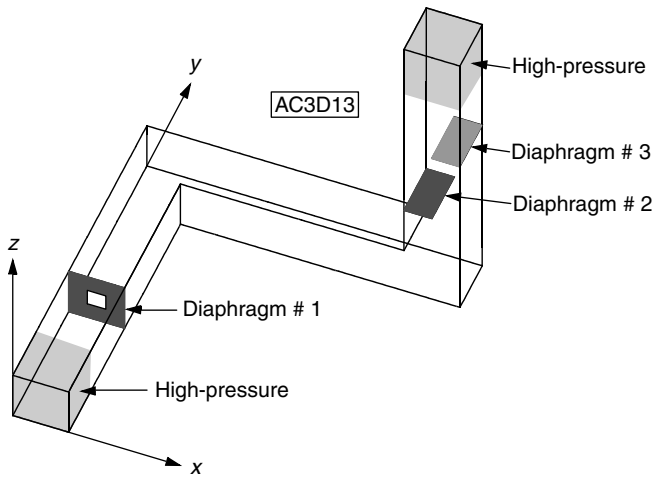

Figure 8. Explosions in a 3-D labyrinth. Problem statement. A color version of this image is available at http://www.mrw.interscience.wiley.com/ecm

7 ALE METHODS IN NONLINEAR SOLID MECHANICS

Starting in the late 1970s, the ALE formulation has been extended to nonlinear solid and structural mechanics. Particular efforts were made in response to the need to simulate problems describing crack propagation, impact, explosion, vehicle crashes, as well as forming processes of materials. The large distortions/deformations that characterize these problems clearly undermine the utility of the Lagrangian approach traditionally used in problems involving materials with path-dependent constitutive relations. Representative publications on the use of ALE in solid mechanics are, among many others, (Liu *et al.*, 1986), (Liu *et al.*, 1988), (Schreurs *et al.*, 1986), (Benson, 1989), (Huétink *et al.*, 1990), (Ghosh and Kikuchi, 1991), (Baaijens, 1993), (Huerta and Casadei, 1994), (Rodríguez-Ferran *et al.*, 1998), (Askes *et al.*, 1999), (Askes and Sluys, 2000), and (Rodríguez-Ferran *et al.*, 2002).

If mechanical effects are uncoupled from thermal effects, the mass and momentum equations can be solved independently from the energy equation. According to expressions (34), the ALE version of these equations is

$$\left.\frac{\partial \rho}{\partial t}\right|_{\boldsymbol{\chi}} + (\boldsymbol{c} \cdot \nabla)\rho = -\rho \nabla \cdot \boldsymbol{v} \tag{40a}$$

$$\rho \boldsymbol{a} = \rho \left.\frac{\partial \boldsymbol{v}}{\partial t}\right|_{\boldsymbol{\chi}} + \rho(\boldsymbol{c} \cdot \nabla)\boldsymbol{v} = \nabla \cdot \boldsymbol{\sigma} + \rho \boldsymbol{b} \tag{40b}$$

where $\boldsymbol{a}$ is the material acceleration defined in (35a, b and c), $\boldsymbol{\sigma}$ denotes the Cauchy stress tensor and $\boldsymbol{b}$ represents an applied body force per unit mass.

A standard simplification in nonlinear solid mechanics consists of dropping the mass equation (40a), which is not explicitly accounted for, thus solving only the momentum equation (40b). A common assumption consists of taking the density ρ as constant, so that the mass balance (40a) reduces to

$$\nabla \cdot \boldsymbol{v} = 0 \tag{41}$$

which is the well-known incompressibility condition. This simplified version of the mass balance is also commonly neglected in solid mechanics. This is acceptable because elastic deformations typically induce very small changes in volume, while plastic deformations are volume preserving (isochoric plasticity). This means that changes in density are negligible and that equation (41) automatically holds to sufficient approximation without the need to add it explicitly to the set of governing equations.

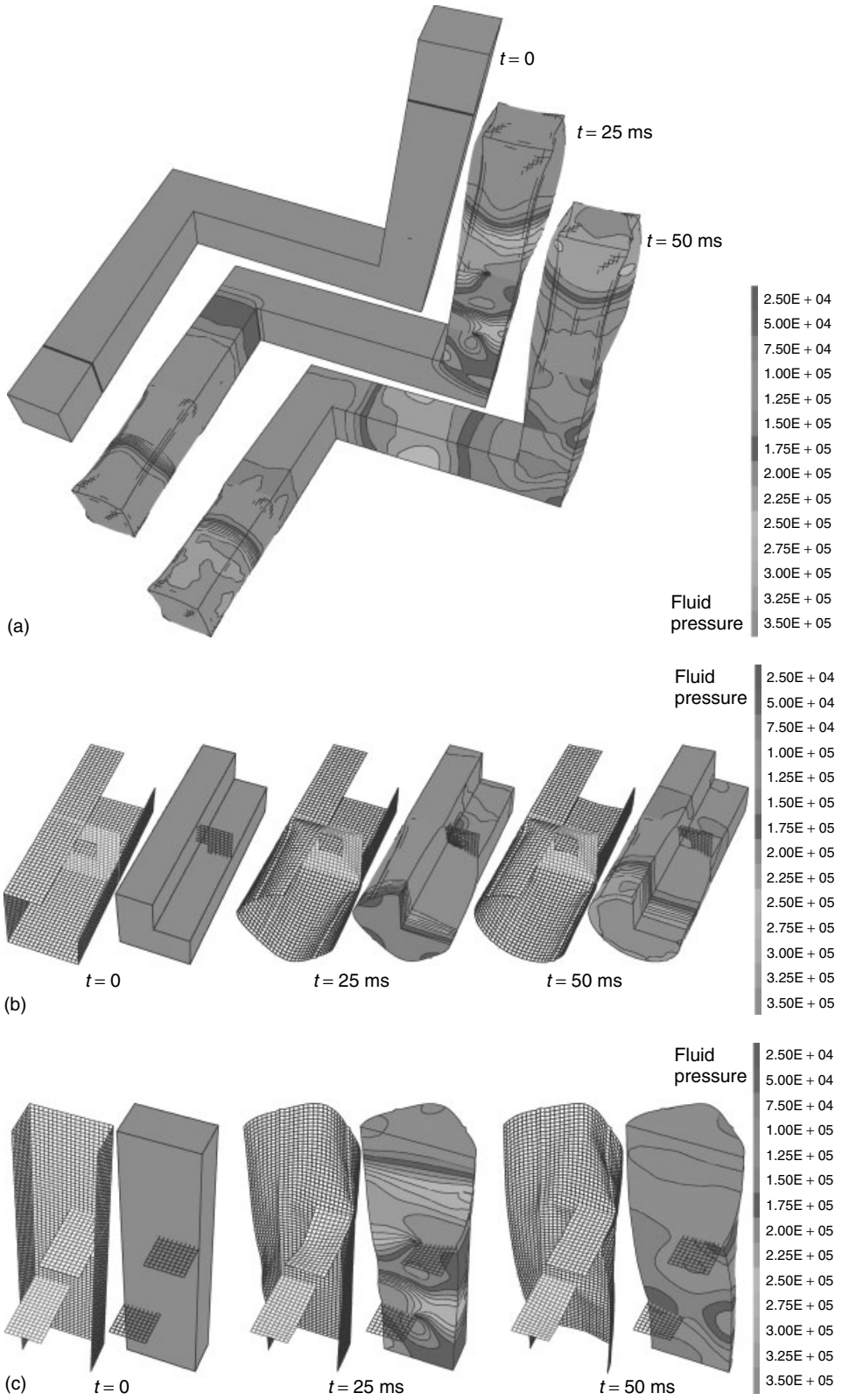

Figure 9. Explosions in a 3-D labyrinth. Deformation in structure and pressure in fluid are properly captured with ALE fluid–structure interaction: (a) whole model; (b) zoom of diaphragm #1; (c) zoom of diaphragms #2 and #3. A color version of this image is available at http://www.mrw.interscience.wiley.com/ecm

7.1 ALE treatment of steady, quasistatic and dynamic processes

In discussing the ALE form (40b) of the momentum equation, we shall distinguish between steady, quasistatic, and dynamic processes. In fact, the expression for the inertia forces $\rho\boldsymbol{a}$ critically depends on the particular type of process under consideration.

A process is called *steady* if the material velocity $\boldsymbol{v}$ in every spatial point $\boldsymbol{x}$ is constant in time. In the Eulerian description (35b), this results in zero local acceleration $\partial\boldsymbol{v}/\partial t|_{\boldsymbol{x}}$ and only the convective acceleration is present in the momentum balance, which reads

$$\rho\boldsymbol{a} = \rho(\boldsymbol{v}\cdot\nabla)\boldsymbol{v} = \nabla\cdot\boldsymbol{\sigma} + \rho\boldsymbol{b} \tag{42}$$

In the ALE context, it is also possible to assume that a process is steady with respect to a grid point $\boldsymbol{\chi}$ and neglect the local acceleration $\partial\boldsymbol{v}/\partial t|_{\boldsymbol{\chi}}$ in the expression (35c); see for instance, (Ghosh and Kikuchi, 1991). The momentum balance then becomes

$$\rho\boldsymbol{a} = \rho(\boldsymbol{c}\cdot\nabla)\boldsymbol{v} = \nabla\cdot\boldsymbol{\sigma} + \rho\boldsymbol{b} \tag{43}$$

However, the physical meaning of a null ALE local acceleration (that is, of an "ALE-steady" process) is not completely clear, due to the arbitrary nature of the mesh velocity and, hence, of the convective velocity $\boldsymbol{c}$.

A process is termed *quasistatic* if the inertia forces $\rho\boldsymbol{a}$ are *negligible* with respect to the other forces in the momentum balance. In this case, the momentum balance reduces to the static equilibrium equation

$$\nabla\cdot\boldsymbol{\sigma} + \rho\boldsymbol{b} = 0 \tag{44}$$

in which time and material velocity play no role. Since the inertia forces have been neglected, the different descriptions of acceleration in equations (35a, b and c) do not appear in equation (44), which is therefore valid in both Eulerian and ALE formulations. The important conclusion is that *there are no convective terms in the ALE momentum balance for quasistatic processes*. A process may be modeled as quasistatic if stress variations and/or body forces are much larger than inertia forces. This is a common situation in solid mechanics, encompassing, for instance, various metal-forming processes. As discussed in the next section, convective terms are nevertheless present in the ALE (and Eulerian) constitutive equation for quasistatic processes. They reflect the fact that grid points are occupied by different particles at different times.

Finally, in *transient dynamic processes*, all terms must be retained in expression (35c) for the material acceleration, and the momentum balance equation is given by expression (40b).

7.2 ALE constitutive equations

Compared to the use of the ALE description in fluid dynamics, the main additional difficulty in nonlinear solid mechanics is the design of an appropriate stress-update procedure to deal with history-dependent constitutive equations. As already mentioned, constitutive equations of ALE nonlinear solid mechanics contain convective terms that account for the relative motion between mesh and material. This is the case for both hypoelastoplastic and hyperelastoplastic models.

7.2.1 Constitutive equations for ALE hypoelastoplasticity

Hypoelastoplastic models are based on an additive decomposition of the stretching tensor $\nabla^{\mathsf{S}}\boldsymbol{v}$ (symmetric part of the velocity gradient) into elastic and plastic parts; see, for instance, (Belytschko *et al.*, 2000) or (Bonet and Wood, 1997). They were used in the first ALE formulations for solid mechanics and are still the standard choice. In these models, material behavior is described by a rate-form constitutive equation

$$\boldsymbol{\sigma}^{\star} = \boldsymbol{f}(\boldsymbol{\sigma}, \nabla^{\mathsf{S}}\boldsymbol{v}) \tag{45}$$

relating an objective rate of Cauchy stress $\boldsymbol{\sigma}^{\star}$ to stress and stretching. The material rate of stress

$$\dot{\boldsymbol{\sigma}} = \frac{\partial\boldsymbol{\sigma}}{\partial t}\Big|_{X} = \frac{\partial\boldsymbol{\sigma}}{\partial t}\Big|_{\boldsymbol{\chi}} + (\boldsymbol{c}\cdot\nabla)\boldsymbol{\sigma} \tag{46}$$

cannot be employed in relation (45) to measure the stress rate because it is not an objective tensor, so large rigid-body rotations are not properly treated. An objective rate of stress is obtained by adding to $\dot{\boldsymbol{\sigma}}$ some terms that ensure the objectivity of $\boldsymbol{\sigma}^{\star}$; see, for instance, (Malvern, 1969) or (Belytschko *et al.*, 2000). Two popular objective rates are the Truesdell rate and the Jaumann rate

$$\boldsymbol{\sigma}^{\star} = \dot{\boldsymbol{\sigma}} - \nabla^{\mathsf{W}}\boldsymbol{v}\cdot\boldsymbol{\sigma} - \boldsymbol{\sigma}\cdot(\nabla^{\mathsf{W}}\boldsymbol{v})^{\mathrm{T}} \tag{47}$$

where $\nabla^{\mathsf{W}}\boldsymbol{v} = \frac{1}{2}(\nabla\boldsymbol{v} - \nabla^{\mathrm{T}}\boldsymbol{v})$ is the spin tensor.

Substitution of equation (47) (or similar expressions for other objective stress rates) into equation (45) yields

$$\dot{\boldsymbol{\sigma}} = \boldsymbol{q}(\boldsymbol{\sigma}, \nabla^{\mathsf{S}}\boldsymbol{v}, \ldots) \tag{48}$$

where $\boldsymbol{q}$ contains both $\boldsymbol{f}$ and the terms in $\boldsymbol{\sigma}^{\star}$, which ensure its objectivity.

In the ALE context, referential time derivatives, not material time derivatives, are employed to represent evolution in time. Combining expression (46) of the material rate of stress and the constitutive relation (48) yields a rate-form constitutive equation for ALE nonlinear solid mechanics

$$\dot{\boldsymbol{\sigma}} = \frac{\partial \boldsymbol{\sigma}}{\partial t}\Big|_{\boldsymbol{\chi}} + (\boldsymbol{c} \cdot \nabla)\boldsymbol{\sigma} = \boldsymbol{q} \tag{49}$$

where, again, a convective term reflects the motion of material particles relative to the mesh. Note that this relative motion is inherent in ALE kinematics, so the convective term is present in all the situations described in Section 7.1, including quasistatic processes.

Because of this convective effect, the stress update cannot be performed as simply as in the Lagrangian formulation, in which the element Gauss points correspond to the same material particles during the whole calculation. In fact, the accurate treatment of the convective terms in ALE rate-type constitutive equations is a key issue in the accuracy of the formulation, as discussed in Section 7.3.

7.2.2 Constitutive equations for ALE hyperelastoplasticity

Hyperelastoplastic models are based on a multiplicative decomposition of the deformation gradient into elastic and plastic parts, $\mathbf{F} = \mathbf{F}^{\mathrm{e}}\mathbf{F}^{\mathrm{p}}$; see, for instance, (Belytschko *et al.*, 2000) or (Bonet and Wood, 1997). They have only very recently been combined with the ALE description (see (Rodríguez-Ferran *et al.*, 2002) and (Armero and Love, 2003)).

The evolution of stresses is not described by means of a rate-form equation, but in closed form as

$$\boldsymbol{\tau} = 2\frac{\mathrm{d}W}{\mathrm{d}\boldsymbol{b}^{\mathrm{e}}}\boldsymbol{b}^{\mathrm{e}} \tag{50}$$

where $\boldsymbol{b}^{\mathrm{e}} = \mathbf{F}^{\mathrm{e}} \cdot (\mathbf{F}^{\mathrm{e}})^{\mathrm{T}}$ is the elastic left Cauchy–Green tensor, W is the free energy function, and $\boldsymbol{\tau} = \det(\mathbf{F})\boldsymbol{\sigma}$ is the Kirchhoff stress tensor.

Plastic flow is described by means of the flow rule

$$\dot{\boldsymbol{b}}^{\mathrm{e}} - \nabla \boldsymbol{v} \cdot \boldsymbol{b}^{\mathrm{e}} - \boldsymbol{b}^{\mathrm{e}} \cdot (\nabla \boldsymbol{v})^{\mathrm{T}} = -2\dot{\gamma}\boldsymbol{m}(\boldsymbol{\tau}) \cdot \boldsymbol{b}^{\mathrm{e}} \tag{51}$$

The left-hand side of equation (51) is the Lie derivative of $\boldsymbol{b}^{\mathrm{e}}$ with respect to the material velocity $\boldsymbol{v}$. In the right-hand side, $\boldsymbol{m}$ is the flow direction and $\dot{\gamma}$ is the plastic multiplier.

Using the fundamental ALE relation (27) between material and referential time derivatives, the flow rule (51) can be recast as

$$\frac{\partial \boldsymbol{b}^{\mathrm{e}}}{\partial t}\Big|_{\boldsymbol{\chi}} + (\boldsymbol{c} \cdot \nabla)\boldsymbol{b}^{\mathrm{e}} = \nabla \boldsymbol{v} \cdot \boldsymbol{b}^{\mathrm{e}} + \boldsymbol{b}^{\mathrm{e}} \cdot (\nabla \boldsymbol{v})^{\mathrm{T}} - 2\dot{\gamma}\boldsymbol{m}(\boldsymbol{\tau}) \cdot \boldsymbol{b}^{\mathrm{e}} \tag{52}$$

Note that, like in equation (49), a convective term in this constitutive equation reflects the relative motion between mesh and material.

7.3 Stress-update procedures

In the context of hypoelastoplasticity, various strategies have been proposed for coping with the convective terms in equation (49). Following Benson (1992b), they can be classified into *split* and *unsplit* methods.

If an *unsplit* method is employed, the complete rate equation (49) is integrated forward in time, including both the convective term and the material term $\boldsymbol{q}$. This approach is followed, among others, by Liu *et al.* (1986), who employed an explicit time-stepping algorithm and by Ghosh and Kikuchi (1991), who used an implicit unsplit formulation.

On the other hand, *split*, or fractional-step methods treat the material and convective terms in (49) in two distinct phases: a material (or Lagrangian) phase is followed by a convection (or transport) phase. In exchange for a certain loss in accuracy due to splitting, split methods are simpler and especially suitable in upgrading a Lagrangian code to the ALE description. An implicit split formulation is employed by Huétink *et al.* (1990) to model metal-forming processes. An example of explicit split formulation may be found in (Huerta and Casadei, 1994), where ALE finite elements are used to model fast-transient phenomena.

The situation is completely analogous for hyperelastoplasticity, and similar comments apply to the split or unsplit treatment of material and convective effects. In fact, if a split approach is chosen, see (Rodríguez-Ferran *et al.*, 2002), the only differences with respect to the hypoelastoplastic models are (1) the constitutive model for the Lagrangian phase (hypo/hyper) and (2) the quantities to be transported in the convection phase.

7.3.1 Lagrangian phase

In the Lagrangian phase, convective effects are neglected. The constitutive equations recover their usual expressions (48) and (51) for hypo- and hyper-models respectively. The ALE kinematical description has (momentarily) disappeared from the formulation, so all the concepts, ideas, and algorithms of large strain solid mechanics with a Lagrangian description apply (see (Bonet and Wood, 1997); (Belytschko *et al.*, 2000) and **Chapter 7 of Volume 2**).

The issue of objectivity is one of the main differences between hypo- and hypermodels. When devising time-integration algorithms to update stresses from $\boldsymbol{\sigma}^n$ to $\boldsymbol{\sigma}^{n+1}$

in hypoelastoplastic models, a typical requirement is incremental objectivity (that is, the appropriate treatment of rigid-body rotations over the time interval $[t^n, t^{n+1}]$). In hyperelastoplastic models, on the contrary, objectivity is not an issue at all, because there is no rate equation for the stress tensor.

7.3.2 Convection phase

The convective effects neglected before have to be accounted for now. Since material effects have already been treated in the Lagrangian phase, the ALE constitutive equations read simply

$$\left.\frac{\partial \boldsymbol{\sigma}}{\partial t}\right|_{\boldsymbol{\chi}} + (\boldsymbol{c}\cdot\nabla)\boldsymbol{\sigma} = \mathbf{0}; \quad \left.\frac{\partial \boldsymbol{b}^{\mathrm{e}}}{\partial t}\right|_{\boldsymbol{\chi}} + (\boldsymbol{c}\cdot\nabla)\boldsymbol{b}^{\mathrm{e}} = \mathbf{0};$$
$$\left.\frac{\partial \boldsymbol{\alpha}}{\partial t}\right|_{\boldsymbol{\chi}} + (\boldsymbol{c}\cdot\nabla)\boldsymbol{\alpha} = \mathbf{0} \tag{53}$$

Equations $(53)_1$ and $(53)_2$ correspond to hypo- and hyperelastoplastic models respectively (cf. with equations 49 and 52). In equation $(53)_3$, valid for both hypo- and hyper-models, $\boldsymbol{\alpha}$ is the set of all the material-dependent variables, i.e. variables associated with the material particle $\boldsymbol{X}$: internal variables for hardening or softening plasticity, the volume change in nonisochoric plasticity, and so on, see (Rodríguez-Ferran *et al.*, 2002).

The three equations in (53) can be written more compactly as

$$\left.\frac{\partial \blacksquare}{\partial t}\right|_{\boldsymbol{\chi}} + (\boldsymbol{c}\cdot\nabla)\blacksquare = \mathbf{0} \tag{54}$$

where $\blacksquare$ represents the appropriate variable in each case. Note that equation (54) is simply a first-order linear hyperbolic PDE, which governs the transport of field $\blacksquare$ by the velocity field $\boldsymbol{c}$. However, two important aspects should be considered in the design of numerical algorithms for the solution of this equation:

1. $\blacksquare$ is a tensor (for $\boldsymbol{\sigma}$ and $\boldsymbol{b}^{\mathrm{e}}$) or vector-like (for $\boldsymbol{\alpha}$) field, so equation (54) should be solved for each component $\square$ of $\blacksquare$:

$$\left.\frac{\partial \square}{\partial t}\right|_{\boldsymbol{\chi}} + \boldsymbol{c}\cdot\nabla\square = 0 \tag{55}$$

   Since the number of scalar equations (55) may be relatively large (for instance: eight for a 3-D computation with a plastic model with two internal variables), the need for efficient convection algorithms is a key issue in ALE nonlinear solid mechanics.
2. $\square$ is a Gauss-point-based (i.e. not a nodal-based) quantity, so it is discontinuous across finite element edges. For this reason, its gradient $\nabla\square$ cannot be reliably computed at the element level. In fact, handling $\nabla\square$ is the main numerical challenge in ALE stress update.

Two different strategies may be used to tackle the difficulties associated with $\nabla\square$. One possible approach is to approximate $\square$ by a continuous field $\widetilde{\square}$, and replace $\nabla\square$ by $\nabla\widetilde{\square}$ in equation (55). The smoothed field $\widetilde{\square}$ can be obtained, for instance, by least-squares approximation, see (Huétink *et al.*, 1990).

Another possibility is to retain the discontinuous field $\square$ and devise appropriate algorithms that account for this fact. To achieve this aim, a fruitful observation is noting that, for a piecewise constant field $\square$, equation (55) is the well-known Riemann problem. Through this connection, the ALE community has exploited the expertise on approximate Riemann solvers of the CFD community, see (Le Veque, 1990).

Although $\square$ is, in general, not constant for each element (expect for one-point quadratures), it can be approximated by a piecewise constant field in a simple manner. Figure 10 depicts a four-noded quadrilateral with a 2×2 quadrature subdivided into four subelements. If the value of $\square$ for each Gauss point is taken as representative for the whole subelement, then a field constant within each subelement results.

In this context, equation (55) can be solved explicitly by looping all the subelements in the mesh by means of a Godunov-like technique based on Godunov's method for conservation laws, see (Rodríguez-Ferran *et al.*, 1998):

$$\square^{n+1} = \square^{\mathrm{L}} - \frac{\Delta t}{V}\sum_{\Gamma=1}^{N_\Gamma} f_\Gamma(\square_\Gamma^{\mathrm{c}} - \square^{\mathrm{L}})\left[1 - \mathrm{sign}(f_\Gamma)\right] \tag{56}$$

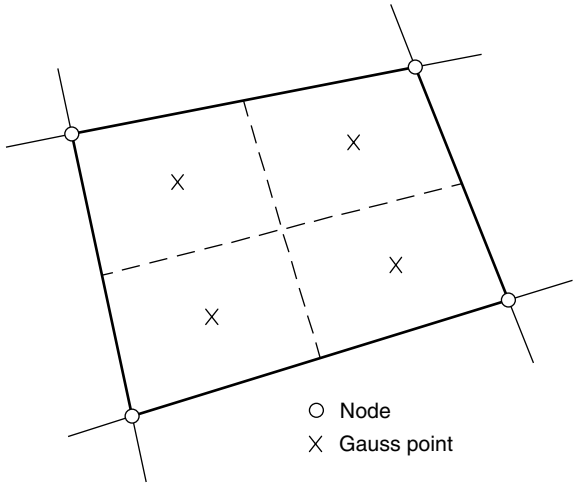

Figure 10. Finite element subdivided into subelements for the Godunov-like stress update.

According to equation (56), the Lagrangian (i.e. after the Lagrangian phase) value $\square^{L}$ is updated into the final value $\square^{n+1}$ by taking into account the flux of $\square$ across the subelement edges. In the second term on the right-hand side, Δt is the time step, V is the volume (or area, in 2-D) of the subelement, N_Γ is the number of edges per subelement, $\square^{c}_{\Gamma}$ is the value of $\square$ in the contiguous subelement across edge Γ, and f_Γ is the flux of convective velocity across edge Γ, $f_\Gamma = \int_\Gamma (\boldsymbol{c} \cdot \boldsymbol{n})\, \mathrm{d}\Gamma$. Note that a full-donor (i.e. full upwind) approach is obtained by means of $\mathrm{sign}(f_\Gamma)$.

Remark *(Split stress update and iterations)* In principle, the complete stress update must be performed at each iteration of the nonlinear equilibrium problem. However, a common simplification consists in leaving the convection phase outside the iteration loop, see (Baaijens, 1993); (Rodríguez-Ferran *et al.*, 1998); and (Rodríguez-Ferran *et al.*, 2002); that is, iterations are performed in a purely Lagrangian fashion up to equilibrium and the convection phase is performed after remeshing, just once per time step. Numerical experiments reveal that disruption of equilibrium caused by convection is not severe and can be handled as extra residual forces in the next load step (see references just cited).

Remark *(ALE finite strain elasticity)* Since hyperelasticity can be seen as a particular case of hyperelastoplasticity, we have chosen here the more general situation for a more useful presentation. In the particular case of large *elastic* strains (hyperelasticity), where $\mathbf{F} = \mathbf{F}^{e}$, an obvious option is to particularize the general approach just described by solving only the relevant equation (i.e. equation 52 for $\boldsymbol{b}^{e}$; note that there are no internal variables $\boldsymbol{\alpha}$ in elasticity). In the literature, other approaches exist for this specific case. Yamada and Kikuchi (1993) and Armero and Love (2003) exploit the relationship $\mathbf{F} = \mathbf{F}_{\boldsymbol{\Phi}} \mathbf{F}_{\boldsymbol{\Psi}}^{-1}$, where $\mathbf{F}_{\boldsymbol{\Phi}}$ and $\mathbf{F}_{\boldsymbol{\Psi}}$ are the deformation gradients of mappings $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ respectively (see Figure 4), to obtain an ALE formulation for hyperelasticity with no convective terms. In exchange for the need to handle the convective term in the update of $\boldsymbol{b}^{e}$ (see equations 52 and 53_2), the former approach has the advantage that only the quality of mapping $\boldsymbol{\Phi}$ needs to be controlled. In the latter approaches, on the contrary, both the quality of $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ must be ensured; that is, two meshes (instead of only one) must be controlled.

7.4 Applications in ALE nonlinear solid mechanics

For illustrative purposes, two powder compaction ALE simulations are briefly discussed here. More details can be found in (Rodríguez-Ferran *et al.*, 2002) and (Pérez-Foguet *et al.*, 2003).

The first example involves the bottom punch compaction of an axisymmetric flanged component (see Figure 11). If a Lagrangian description is used (see Figure 11(a)), the large upward mass flow leads to severe element distortion in the reentrant corner, which in turn affects the accuracy in the relative density. With an ALE description (see Figure 11(b)), mesh distortion is completely precluded by means of a very simple ALE remeshing strategy: a uniform vertical mesh compression is prescribed in the bottom, narrow part of the piece, and no mesh motion (i.e. Eulerian mesh) in the upper, wide part.

The second example involves the compaction of a multilevel component. Owing to extreme mesh distortion, it is not possible to perform the simulation with a Lagrangian approach. Three ALE simulations are compared in Figure 12, corresponding to top, bottom, and simultaneous top–bottom compaction. Even with an unstructured mesh and a more complex geometry, the ALE description avoids mesh distortion, so the final relative density profiles can be computed (see Figure 13).

7.5 Contact algorithms

Contact treatment, especially when frictional effects are present, is a very important feature of mechanical modeling and certainly remains one of the more challenging problems in computational mechanics. In the case of the classical Lagrangian formulation, much attention has been devoted to contact algorithms and the interested reader is referred to references by Zhong, 1993, Wriggers, 2002, and Laursen, 2002 in order to get acquainted with the subject. By contrast, modeling of contact in connection with the ALE description has received much less attention. Paradoxically, one of the interesting features of ALE is that, in some situations, the formulation can avoid the burden of implementing cumbersome contact algorithms. The coining problem in Figure 14 is a good illustration of the versatility of ALE algorithms in the treatment of frictionless contact over a known (moving) surface. The material particle M located at the punch corner at time t has been represented. At time $t + \Delta t$, the punch has moved slightly downwards and, due to compression, the material particle M has moved slightly to the right. When a Lagrangian formulation is used in the simulation of such a process, due to the fact that the mesh sticks to the material, a contact algorithm has to be used to obtain a realistic simulation. On the contrary, using an ALE formalism, the implementation of a contact algorithm can be avoided. This is simply because the ALE formulation allows us to prevent the horizontal displacement of the mesh nodes located under the

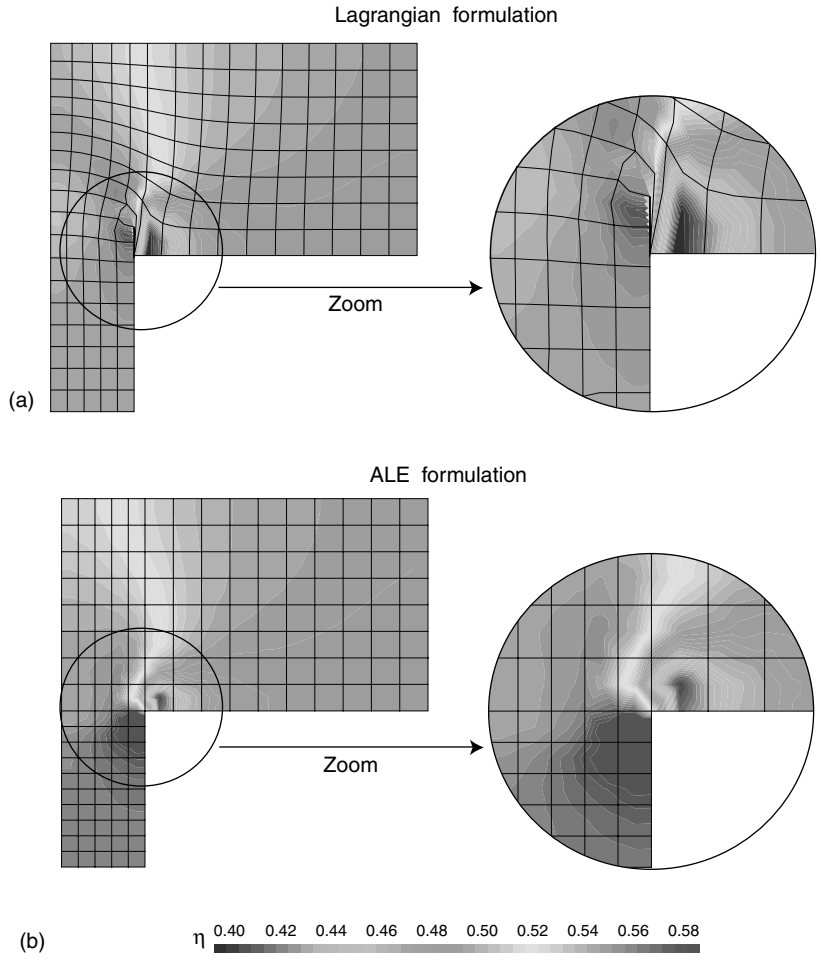

Figure 11. Final relative density after the bottom punch compaction of a flanged component: (a) Lagrangian approach leads to severe mesh distortion; (b) ALE approach avoids distortion. A color version of this image is available at http://www.mrw.interscience.wiley.com/ecm

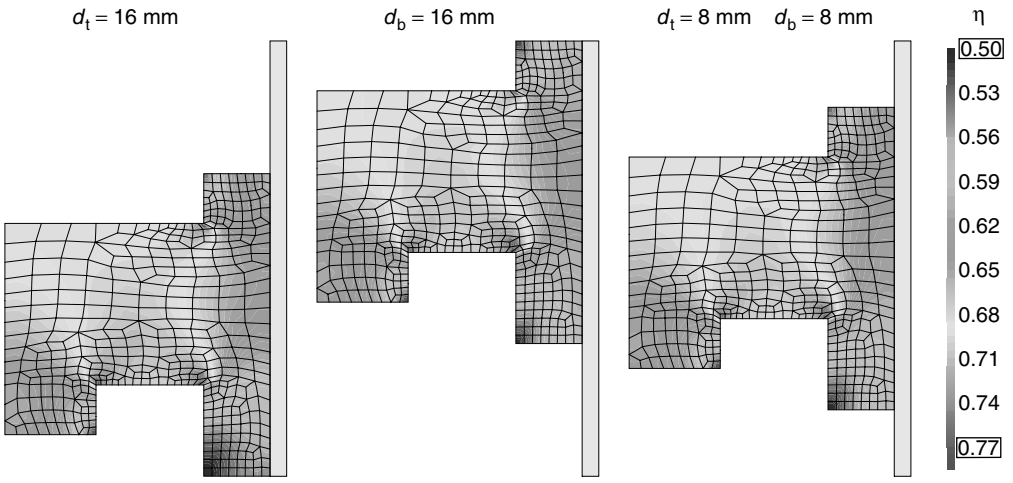

Figure 12. Final relative density of a multilevel component. From left to right: top, bottom and double compaction. A color version of this image is available at http://www.mrw.interscience.wiley.com/ecm

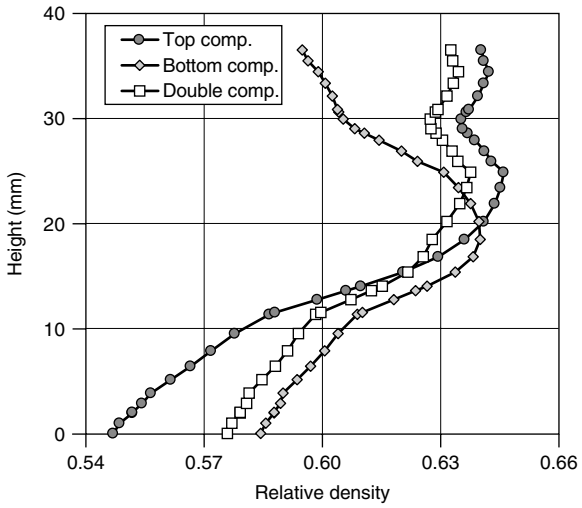

Figure 13. Relative density profiles in a multilevel component along a vertical line for the three-compaction processes.

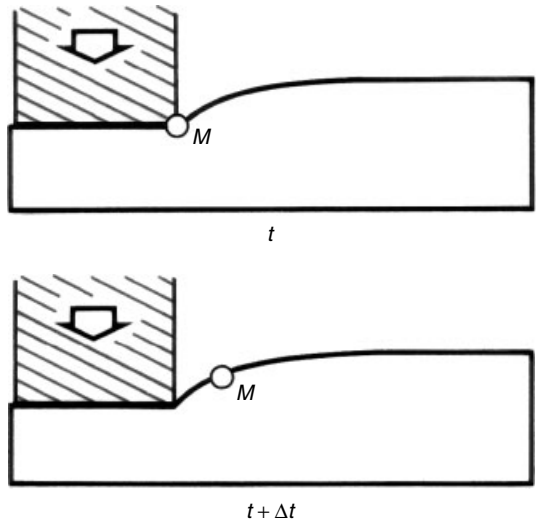

Figure 14. Schematic description of the coining process.

punch, irrespective of the material flow. Numerous illustrations of this particular case can be found, amongst others, in (Schreurs *et al.*, 1986), (Huétink *et al.*, 1990), (Hogge and Ponthot, 1991a), (Huerta and Casadei, 1994), (Rodríguez-Ferran *et al.*, 2002), (Gadala and Wang, 1998), (Gadala *et al.*, 2002), and (Martinet and Chabrand, 2000).

However, in more general situations, a contact algorithm cannot be avoided. In such a case, the nodes of contact elements have to be displaced convectively in accordance with the nodes on the surface of the bulk materials and tools. A direct consequence of this displacement is that convective effects have to be taken into account for history-dependent variables. In the simplest penalty case in which the normal pressure is proportional to the penetration, the normal contact stress depends only on the current geometry and not on the history of penetration. As a consequence, no convection algorithm needs to be activated for that quantity. On the contrary, for a Coulomb-friction model, the shear stresses in the contact elements are incrementally calculated. They therefore depend on the history and hence a convective increment of the shear stress should be evaluated. One simple way to avoid the activation of the convection algorithm for the contact/friction quantities is indeed to keep the boundaries purely Lagrangian. However, in general this will not prevent mesh distortions and nonconvexity in the global mesh.

Various applications of the ALE technology have been developed so far to treat the general case of moving boundaries with Coulomb or Tresca frictional contact. For example, in their pioneering work on ALE contact, Haber and Hariandja (1985) update the mesh, so that nodes and element edges on the surface of contacting bodies coincide exactly at all points along the contact interface in the deformed configuration. In such a case, the matching of node pairs and element edges ensures a precise satisfaction of geometric compatibility and allows a consistent transfer of contact stresses between the two bodies. A similar procedure was established by Ghosh (1992) but, in this case, the numerical model introduces ALE nodal points on one of the contacting (slave) surfaces that are constrained to follow Lagrangian nodes on the other (master) surface. Liu *et al.*, 1991 presented an algorithm, mostly dedicated to rolling applications, where the stick nodes are assumed Lagrangian, whereas the slip nodes are equally spaced between the two adjacent stick nodes. More general procedures have been introduced by Huétink *et al.* (1990).

More recently, sophisticated frictional models incorporating lubrication models have been used in ALE formulations. In such a case, the friction between the contacting lubricated bodies is expressed as a function of interface variables (mean lubrication film thickness, sheet and tooling roughness) in addition to more traditional variables (interface pressure, sliding speed, and strain rate). Examples of complex lubrication models integrated into an ALE framework have been presented by Hu and Liu (1992, 1993, 1994), Martinet and Chabrand (2000), and Boman and Ponthot (2002).

REFERENCES

Ait-Ali-Yahia D, Baruzzi G, Habashi WG, Fortin M, Dompierre J and Vallet M-G Anisotropic mesh adaptation: towards user-independent, mesh-independent and solver-independent CFD. Part II. Structured grids. *Int. J. Numer. Methods Fluids* 2002; **39**(8):657–673.

Argyris JH, Doltsinis JSt, Fisher H and Wüstenberg H. TA ΠΑΝΤΑ PEI. *Comput. Methods Appl. Mech. Eng.* 1985; **51**(1–3):289–362.

Armero F and Love E. An arbitrary Lagrangian-Eulerian finite element method for finite strain plasticity. *Int. J. Numer. Methods Eng.* 2003; **57**(4):471–508.

Askes H and Rodríguez-Ferran A. A combined rh-adaptive scheme based on domain subdivision. Formulation and linear examples. *Int. J. Numer. Methods Eng.* 2001; **51**(3):253–273.

Askes H, Rodríguez-Ferran A and Huerta A. Adaptive analysis of yield line patterns in plates with the arbitrary Lagrangian-Eulerian method. *Comput. Struct.* 1999; **70**(3):257–271.

Askes H and Sluys LJ. Remeshing strategies for adaptive ALE analysis of strain localisation. *Eur. J. Mech., A-Solids* 2000; **19**(3):447–467.

Askes H, Sluys LJ and de Jong BBC. Remeshing techniques for r-adaptive and combined h/r-adaptive analysis with application to 2D/3D crack propagation. *Struct. Eng. Mech.* 2001; **12**(5):475–490.

Aymone JLF, Bittencourt E and Creus GJ. Simulation of 3D metal forming using an arbitrary Lagrangian-Eulerian finite element method. *J. Mater. Process. Technol.* 2001; **110**(2): 218–232.

Baaijens FPT. An U-ALE formulation of 3-D unsteady viscoelastic flow. *Int. J. Numer. Methods Eng.* 1993; **36**(7): 1115–1143.

Batina JT. Implicit flux-split Euler schemes for unsteady aerodynamic analysis involving unstructured dynamic meshes. *AIAA J.* 1991; **29**(11):1836–1843.

Belytschko T and Kennedy JM. Computer methods for subassembly simulation. *Nucl. Eng. Des.* 1978; **49**:17–38.

Belytschko T and Liu WK. Computer methods for transient fluid-structure analysis of nuclear reactors. *Nucl. Saf.* 1985; **26**(1):14–31.

Belytschko T, Flanagan DP and Kennedy JM. Finite element methods with user-controlled meshes for fluid-structure interaction. *Comput. Methods Appl. Mech. Eng.* 1982; **33**(1–3): 669–688.

Belytschko T, Kennedy JM and Schoeberle DF. Quasi-Eulerian finite element formulation for fluid-structure interaction. *Proceedings of Joint ASME/CSME Pressure Vessels and Piping Conference*. ASME: New York, 1978; p. 13, ASME paper 78-PVP-60.

Belytschko T, Kennedy JM and Schoeberle DF. Quasi-Eulerian finite element formulation for fluid-structure interaction. *J. Press. Vessel Technol.-Trans. ASME* 1980; **102**:62–69.

Belytschko T, Liu WK and Moran B. *Nonlinear Finite Elements for Continua and Structures*. Wiley: Chichester, 2000.

Benson DJ. An efficient, accurate, simple ALE method for nonlinear finite element programs. *Comput. Methods Appl. Mech. Eng.* 1989; **72**(3):305–350.

Benson DJ. Vectorization techniques for explicit ALE calculations. An efficient, accurate, simple ALE method for nonlinear finite element programs. *Comput. Methods Appl. Mech. Eng.* 1992a; **96**(3):303–328.

Benson DJ. Computational methods in Lagrangian and Eulerian hydrocodes. *Comput. Meth. Appl. Mech. Eng.* 1992b; **99**(2–3):235–394.

Boman R and Ponthot JP. Finite elements for the lubricated contact between solids in metal forming processes. *Acta Metall. Sinica* 2000; **13**(1):319–327.

Boman R and Ponthot JP. Numerical simulation of lubricated contact in rolling processes. *J. Mater. Process. Technol.* 2002; **125–126**:405–411.

Bonet J and Wood RD. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press: Cambridge, 1997.

Braess H and Wriggers P. Arbitrary Lagrangian-Eulerian finite element analysis of free surface flow. *Comput. Methods Appl. Mech. Eng.* 2000; **190**(1–2):95–110.

Casadei F and Halleux JP. An algorithm for permanent fluid-structure interaction in explicit transient dynamics. *Comput. Methods Appl. Mech. Eng.* 1995; **128**(3–4):231–289.

Casadei F and Potapov S. Permanent fluid-structure interaction with nonconforming interfaces in fast transient dynamics. *Comput. Methods Appl. Mech. Eng.* 2004; **193**: to appear in the Special Issue on the ALE formulation.

Casadei F and Sala A. Finite element and finite volume simulation of industrial fast transient fluid-structure interactions. In *Proceedings European Conference on Computational Mechanics – Solids, Structures and Coupled Problems in Engineering*, Wunderlich W (ed.). Lehrstuhl für Statik: Technische Universität München, 1999.

Casadei F, Halleux JP, Sala A and Chillè F. Transient fluid-structure interaction algorithms for large industrial applications. *Comput. Methods Appl. Mech. Eng.* 2001; **190**(24–25): 3081–3110.

Castro-Diaz MJ, Bourouchaki H, George PL, Hecht F and Mohammadi B. Anisotropic adaptative mesh generation in two dimensions for CFD. In *Proceedings of the Third ECCOMAS Computational Fluid Dynamics Conference*, Paris, 9–13 September, 1996, 181–186.

Cescutti JP, Wey E and Chenot JL. Finite element calculation of hot forging with continuous remeshing. In *Modelling of Metal Forming Processes, EUROMECH-233*, Chenot JL and Oñate E (eds). Sophia-Antipolis, 1988; 207–216.

Chenot JL and Bellet M. The ALE method for the numerical simulation of material forming processes. In *Simulation of Materials Processing: Theory, Methods and Applications – NUMIFORM 95*, Shen SF and Dawson P (eds). Balkema: Ithaca, New York, 1995; 39–48.

Donea J. Arbitrary Lagrangian-Eulerian finite element methods. In *Computational Methods for Transient Analysis*, Belytschko T and Hughes TJR (eds). North-Holland: Amsterdam, 1983; 474–516.

Donea J and Huerta A. *Finite Element Methods for Flow Problems*. Wiley: Chichester, 2003.

Donea J, Fasoli-Stella P and Giuliani S. Lagrangian and Eulerian finite element techniques for transient fluid-structure interaction problems. In *Trans. 4th Int. Conf. on Structural Mechanics in Reactor Technology*, Paper B1/2, San Francisco, 1977.

Donea J, Giuliani S and Halleux JP. An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions. *Comput. Methods Appl. Mech. Eng.* 1982; **33**(1–3):689–723.

Eriksson LE. Practical three-dimensional mesh generation using transfinite interpolation. *SIAM J. Sci. Statist. Comput.* 1985; **6**(3):712–741.

Farhat C, Geuzaine Ph and Grandmont C. The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids. *J. Comput. Phys.* 2001; **174**(2):669–694.

Fortin M, Vallet M-G, Dompierre J, Bourgault Y and Habashi WG. Anisotropic mesh adaptation: theory, validation and applications. In *Proceedings of the Third ECCOMAS Computational Fluid Dynamics Conference*, Paris, 9–13 September, 1996, 174–180.

Franck RM and Lazarus RB. Mixed Eulerian-Lagrangian method. In *Methods in Computational Physics*, *Vol. 3: Fundamental methods in Hydrodynamics*, Alder B, Fernbach S and Rotenberg M (eds). Academic Press: New York, 1964; 47–67.

Gadala MS and Wang J. ALE formulation and its application in solid mechanics. *Comput. Methods Appl. Mech. Eng.* 1998; **167**(1–2):33–55.

Gadala MS and Wang J. Simulation of metal forming processes with finite element methods. *Int. J. Numer. Methods Eng.* 1999; **44**(10):1397–1428.

Gadala MS, Movahhedy MR and Wang J. On the mesh motion for ALE modeling of metal forming processes. *Finite Elem. Anal. Des.* 2002; **38**(5):435–459.

Ghosh S. Arbitrary Lagrangian-Eulerian finite element analysis of large deformation in contacting bodies. *Int. J. Numer. Methods Eng.* 1992; **33**(9):1891–1925.

Ghosh S and Kikuchi N. An arbitrary Lagrangian-Eulerian finite element method for large deformation analysis of elastic-viscoplastic solids. *Comput. Methods Appl. Mech. Eng.* 1991; **86**(2):127–188.

Ghosh S and Raju S. R-S adapted arbitrary Lagrangian Eulerian finite element method for metal-forming problems with strain localization. *Int. J. Numer. Methods Eng.* 1996; **39**(19): 3247–3272.

Giuliani S. An algorithm for continuous rezoning of the hydrodynamic grid in arbitrary Lagrangian-Eulerian computer codes. *Nucl. Eng. Des.* 1982; **72**:205–212.

Gordon WJ and Hall CH. Construction of curvilinear co-ordinate systems and applications to mesh generation. *Int. J. Numer. Methods Eng.* 1973; **7**(4):461–477.

Guillard H and Farhat C. On the significance of the geometric conservation law for flow computations on moving meshes. *Comput. Methods Appl. Mech. Eng.* 2000; **190**(11–12):1467–1482.

Habashi WG, Fortin M, Vallet M-G, Dompierre J, Bourgault Y and Ait-Ali-Yahia D. Anisotropic mesh adaptation: towards user-independent, mesh-independent and solver-independent CFD solutions. Part I: Theory. *Int. J. Numer. Methods Fluids* 2000; **32**(6):725–744.

Haber RB. A mixed Eulerian-Lagrangian displacement model for large-deformation analysis in solid mechanics. *Comput. Methods Appl. Mech. Eng.* 1984; **43**(3):277–292.

Haber R and Abel JF. Discrete transfinite mappings for the description and meshing of three-dimensional surfaces using interactive computer graphics. *Int. J. Numer. Methods Eng.* 1982; **18**(1):41–66.

Haber RB and Hariandja BH. Computational strategies for nonlinear and fracture mechanics problems. An Eulerian-Lagrangian finite element approach to large deformation frictional contact. *Comput. Struct.* 1985; **20**(1–3):193–201.

Hermansson J and Hansbo P. A variable diffusion method for mesh smoothing. *Commun. Numer. Methods Eng.* 2003; **19**(11):897–908.

Hirt CW, Amsden AA and Cook JL. An arbitrary Lagrangian-Eulerian computing method for all flow speeds. *J. Comput. Phys.* 1974; **14**:227–253. Reprinted in *J. Comput. Phys.* 1997; **135**(2):203–216.

Hogge M and Ponthot JP. Metal forming analysis via Eulerian-Lagrangian FEM with adaptive mesh. *Latin Am. Res.* 1991; **21**:217–224.

Hu YK and Liu WK. ALE finite element formulation for ring rolling analysis. *Int. J. Numer. Methods Eng.* 1992; **33**(6):1217–1236.

Hu YK and Liu WK. An ALE hydrodynamic lubrication finite element method with application to strip rolling. *Int. J. Numer. Methods Eng.* 1993; **36**(5):855–880.

Hu YK and Liu WK. Finite element hydrodynamic friction model for metal forming. *Int. J. Numer. Methods Eng.* 1994; **37**(23):4015–4037.

Huerta A and Casadei F. New ALE applications in nonlinear fast-transient solid dynamics. *Eng. Comput.* 1994; **11**(4):317–345.

Huerta A and Liu WK. Viscous flow structure interaction. *J. Press. Vessel Technol.-Trans. ASME* 1988a; **110**(1):15–21.

Huerta A and Liu WK. Viscous flow with large free-surface motion. *Comput. Methods Appl. Mech. Eng.* 1988b; **69**(3): 277–324.

Huerta A and Liu WK. ALE formulation for large boundary motion. In *Trans. 10th Int. Conf. Structural Mechanics in Reactor Technology*, Vol. B, Anaheim, 1989; 335–346.

Huerta A and Liu WK. Large amplitude sloshing with submerged blocks. *J. Press. Vessel Technol.-Trans. ASME* 1990; **112**:104–108.

Huerta A, Rodríguez-Ferran A, Díez P and Sarrate J. Adaptive finite element strategies based on error assessment. *Int. J. Numer. Methods Eng.* 1999; **46**(10):1803–1818.

Huétink J, Vreede PT and van der Lugt J. Progress in mixed Eulerian-Lagrangian finite element simulation of forming processes. *Int. J. Numer. Methods Eng.* 1990; **30**(8):1441–1457.

Hughes TJR, Liu WK and Zimmermann TK. Lagrangian-Eulerian finite element formulation for incompressible viscous flows. *Comput. Methods Appl. Mech. Eng.* 1981; **29**(3):329–349; Presented at the U.S.-Japan conference on Interdisciplinary Finite Element Analysis, Cornell University, August 7–11, 1978.

Koh HM and Haber RB. Elastodynamic formulation of the Eulerian-Lagrangian kinematic description. *J. Appl. Mech.-Trans. ASME* 1986; **53**(4):839–845.

Koh HM, Lee HS and Haber RB. Dynamic crack propagation analysis using Eulerian-Lagrangian kinematic descriptions. *Comput. Mech.* 1988; **3**:141–155.

Koobus B and Farhat C. Second-order accurate and geometrically conservative implicit schemes for flow computations on unstructured dynamic meshes. *Comput. Methods Appl. Mech. Eng.* 1999; **170**(1–2):103–129.

Laursen TA. *Computational Contact and Impact Mechanics*. Springer: Berlin, 2002.

Lesoinne M and Farhat C. Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations. *Comput. Methods Appl. Mech. Eng.* 1996; **134**(1–2):71–90.

Le Tallec P and Martin C. A nonlinear elasticity model for stuctured mesh adaptation. In *Proceedings of the Third ECCOMAS Computational Fluid Dynamics Conference*, Paris, 1996; 174–180.

Le Tallec P and Mouro J. Fluid-structure interaction with large structural displacements. *Comput. Methods Appl. Mech. Eng.* 2001; **190**(24–25):3039–3067.

Le Veque RJ. *Numerical Methods for Conservation Laws*. Lectures in Mathematics, ETH Zürich. Birkhäuser Verlag: Basel, 1990.

Liu WK and Chang HG. Efficient computational procedures for long-time duration fluid-structure interaction problems. *J. Press. Vessel Technol.-Trans. ASME* 1984; **106**:317–322.

Liu WK and Chang HG. A method of computation for fluid structure interaction. *Comput. Struct.* 1985; **20**(1–3):311–320.

Liu WK and Gvildys J. Fluid-structure interaction of tanks with an eccentric core barrel. *Comput. Methods Appl. Mech. Eng.* 1986; **58**(1):51–77.

Liu WK, Belytschko T and Chang H. An arbitrary Lagrangian-Eulerian finite element method for path-dependent materials. *Comput. Methods Appl. Mech. Eng.* 1986; **58**(2):227–245.

Liu WK, Chang H, Chen JS and Belytschko T. Arbitrary Lagrangian-Eulerian Petrov-Galerkin finite elements for nonlinear continua. *Comput. Methods Appl. Mech. Eng.* 1988; **68**(3):259–310.

Liu WK, Chen JS, Belytschko T and Zhang YF. Adaptive ALE finite elements with particular reference to external work rate on frictional interface. *Comput. Methods Appl. Mech. Eng.* 1991; **93**(2):189–216.

Löhner R and Yang C. Improved ALE mesh velocities for moving bodies. *Commun. Numer. Methods Eng.* 1996; **12**(10):599–608.

Malvern LW. *Introduction to the Mechanics of a Continuous Medium*. Prentice-Hall: Englewood Cliffs, 1969.

Martinet F and Chabrand P. Application of ALE finite element method to a lubricated friction model in sheet metal forming. *Int. J. Solids Struct.* 2000; **37**(29):4005–4031.

Müller J-D. Anisotropic adaptation and multigrid for hybrid grids. *Int. J. Numer. Methods Fluids* 2002; **40**(3–4):445–455.

Noh WF. CEL:A time-dependent two-space dimensional coupled Eulerian-Lagrangian code. In *Methods in Computational Physics*, Alder B, Fernbach S and Rotenberg M (eds), vol. 3. Academic Press: New York, 1964; 117–179.

Nomura T and Hughes TJR. An arbitrary Lagrangian-Eulerian finite element method for interaction of fluid and a rigid body. *Comput. Methods Appl. Mech. Eng.* 1992; **95**(1):115–138.

Pérez-Foguet A, Rodríguez-Ferran A and Huerta A. Efficient and accurate approach for powder compaction problems. *Comput. Mech.* 2003; **30**(3):220–234.

Pijaudier-Cabot G, Bodé L and Huerta A. Arbitrary Lagrangian-Eulerian finite element analysis of strain localization in transient problems. *Int. J. Numer. Methods Eng.* 1995; **38**(24): 4171–4191.

Ponthot JP and Belytschko T. Arbitrary Lagrangian-Eulerian formulation for element-free Galerkin method. *Comput. Methods Appl. Mech. Eng.* 1998; **152**(1–2):19–46.

Ponthot JP and Hogge M. The use of the Eulerian-Lagrangian FEM in metal forming applications including contact and adaptive mesh. In *Advances in Finite Deformation Problems in Material Processing*, Chandra N and Reddy JN (eds). ASME Winter Annual Meeting, ASME: AMD-125, ASME (American Society of Mechanical Engineers): Atlanta, 1991; 44–64.

Pracht WE. Calculating three-dimensional fluid flows at all flow speeds with an Eulerian-Lagrangian computing mesh. *J. Comput. Phys.* 1975; **17**:132–159.

Ramaswamy B and Kawahara M. Arbitrary Lagrangian-Eulerian finite element method for unsteady, convective, incompressible viscous free surface fluid flow. *Int. J. Numer. Methods Fluids* 1987; **7**(10):1053–1075.

Rodríguez-Ferran A, Casadei F and Huerta A. ALE stress update for transient and quasistatic processes. *Int. J. Numer. Methods Eng.* 1998; **43**(2):241–262.

Rodríguez-Ferran A, Pérez-Foguet A and Huerta A. Arbitrary Lagrangian-Eulerian (ALE) formulation for hyperelastoplasticity. *Int. J. Numer. Methods Eng.* 2002; **53**(8):1831–1851.

Sarrate J and Huerta A. An improved algorithm to smooth graded quadrilateral meshes preserving the prescribed element size. *Commun. Numer. Methods Eng.* 2001; **17**(2):89–99.

Sarrate J, Huerta A and Donea J. Arbitrary Lagrangian-Eulerian formulation for fluid rigid-body interaction. *Comput. Methods Appl. Mech. Eng.* 2001; **190**(24–25):3171–3188.

Schreurs PJG, Veldpaus FE and Brekelmans WAM. Simulation of forming processes using the Arbitrary Eulerian-Lagrangian formulation. *Comput. Methods Appl. Mech. Eng.* 1986; **58**(1):19–36.

Souli M and Zolesio JP. Arbitrary Lagrangian-Eulerian and free-surface methods in fluid mechanics. *Comput. Methods Appl. Mech. Eng.* 2001; **191**(3–5):451–466.

Smith RW. AUSM(ALE): a geometrically conservative Arbitrary Lagrangian-Eulerian flux splitting scheme. *J. Comput. Phys.* 1999; **150**(1):268–286.

Trépanier JY, Reggio M, Paraschivoiu M and Camarero R. Unsteady Euler solutions for arbitrarily moving bodies and boundaries. *AIAA J.* 1993; **31**(10):1869–1876.

Trulio JG. *Theory and Structure of the AFTON Codes*. Report AFWL-TR-66-19, Air Force Weapons Laboratory: Kirtland Air Force Base, 1966.

van Haaren MJ, Stoker HC, van den Boogaard AH and Huétink J. The ALE method with triangular elements: Direct convection of integration point values. *Int. J. Numer. Methods Eng.* 2000; **49**(5):697–720.

Winslow AM. *Equipotential Zoning of Two-dimensional Meshes*. Report UCRL-7312, University of California, Lawrence Livermore Laboratory: Livermore, California, 1963.

Wriggers P. *Computational Contact Mechanics*. Wiley: Chichester, 2002.

Yamada T and Kikuchi F. An arbitrary Lagrangian-Eulerian finite element method for incompressible hyperelasticity. *Comput. Methods Appl. Mech. Eng.* 1993; **102**(2):149–177.

Zhang Q and Hisada T. Analysis of fluid-structure interaction problems with structural buckling and large domain changes by ALE finite element method. *Comput. Methods Appl. Mech. Eng.* 2001; **190**(48):6341–6357.

Zhong ZH. *Finite Element Procedures for Contact-Impact Problems*. Oxford University Press: Oxford, 1993.