Finite Element Methods for Flow Problems

Finite Element Methods for Flow Problems. Jean Donea and Antonio Huerta
Copyright © 2003 John Wiley & Sons, Ltd.
ISBN: 0-471-49666-9

Finite Element Methods for Flow Problems

Jean Donea and Antonio Huerta

WILEY

Copyright ©2003 John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester,
West Sussex PO19 8SQ, England

Telephone (+44) 1243 779777

Email (for orders and customer service enquiries): cs-books@wiley.co.uk

Visit our Home Page on www.wileyeurope.com or www.wiley.com

This publication is designed to provide accurate and authoritative information in regard to the subject matter covered. It is sold on the understanding that the Publisher is not engaged in rendering professional services. If professional advice or other expert assistance is required, the services of a competent professional should be sought.

Other Wiley Editorial Offices

John Wiley & Sons Inc., 111 River Street, Hoboken, NJ 07030, USA

Jossey-Bass, 989 Market Street, San Francisco, CA 94103-1741, USA

Wiley-VCH Verlag GmbH, Boschstr. 12, D-69469 Weinheim, Germany

John Wiley & Sons Australia Ltd, 33 Park Road, Milton, Queensland 4064, Australia

John Wiley & Sons (Asia) Pte Ltd, 2 Clementi Loop #02-01, Jin Xing Distripark, Singapore 129809

John Wiley & Sons Canada Ltd, 22 Worcester Road, Etobicoke, Ontario, Canada M9W 1L1

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

ISBN 0-471-49666-9

Typeset from pdf files supplied by the author
Printed and bound in Great Britain by Antony Rowe Ltd, Chippenham, Wiltshire
This book is printed on acid-free paper responsibly manufactured from sustainable forestry in which at least two trees are planted for each one used for paper production.

Contents

Preface

This book is based on lectures regularly taught in the fifth year of engineering diplomas or basic graduate courses at the University of Liège and at the Universitat Politècnica de Catalunya. The goal of the book is not to provide an exhaustive account of finite elements in fluids, which is an extremely active area of research. The objective is to present the fundamentals of stabilized finite element methods for the analysis of steady and time-dependent convection–diffusion and fluid dynamics problems with an engineering rather than a mathematical bias. Organized into six chapters, it combines theoretical aspects and practical applications and covers some of the recent research in several areas of computational fluid dynamics.

The authors wish to thank their colleagues and friends at the Joint Research Centre of the European Commission, Ispra, Alenia, Torino, ENEL Research Hydraulics and Structures, Milano, Politecnico di Milano, Université de Liège and Universitat Politècnica de Catalunya, who contributed to the development of several of the concepts presented in this book and provided worked examples. We are especially grateful to Folco Casadei, Luigi Quartapelle and Vittorio Selmin who reviewed parts of this book, provided numerous suggestions and corrections, and contributed with several illustrative examples. We are particularly in debt to our colleagues at the Laboratori de Càlcul Numèric: Sonia Férnandez-Méndez, Esther Sala-Lardies, Antonio Rodríguez-Ferran, Pedro Díez, Josep Sarrate, Agustí Pérez-Foguet and Xevi Roca. Their reviews, suggestions and contributions have made this book a reality. The second author must also thank Gilles Pijaudier-Cabot and the École Centrale de Nantes for giving him sanctuary during the last stages of this project.

Above all, however, our thanks go to our wives, Marie-Paule and Sara, for their continuous and unfailing support and patience for many years and particularly during the writing of this book.

J. DONEA AND A. HUERTA

1

Introduction and preliminaries

1.1 FINITE ELEMENTS IN FLUID DYNAMICS

Introduced in the late 1950s in the aircraft industry, see, for instance, the historical outline by Felippa (2001), the finite element method (FEM) has emerged as one of the most powerful numerical methods so far devised. Among the basic attributes of the method which have led to its widespread acceptance and use are the ease in modeling complex geometries, the consistent treatment of differential-type boundary conditions and the possibility to be programmed in a flexible and general purpose format.

Standard finite element approximations are based upon the Galerkin formulation of the method of weighted residuals. This formulation has proven eminently successful in application to problems in solid/structural mechanics and in other situations, such as heat conduction, governed by diffusion-type equations. The reason for this success is that, when applied to problems governed by self-adjoint elliptic or parabolic partial differential equations, the Galerkin finite element method leads to symmetric stiffness matrices. In this case the difference between the finite element solution and the exact solution is minimized with respect to the energy norm, see, for instance, Strang and Fix (1973). In practice, the Galerkin formulation is optimal in problems governed by self-adjoint equations. In such cases, there exists a quadratic functional the minimum of which corresponds to satisfying the partial differential equation governing the problem at hand. For instance, in linear elasticity the equilibrium position of a structure corresponds to the minimum of the quadratic functional expressing the total potential energy of the system. Similarly, in steady heat conduction problems the thermal equilibrium resulting from satisfying the Laplace or Poisson equation

Finite Element Methods for Flow Problems. Jean Donea and Antonio Huerta
Copyright © 2003 John Wiley & Sons, Ltd.
ISBN: 0-471-49666-9

corresponds to the minimum of a quadratic functional expressed in terms of the thermal flux, which physically represents the total thermal energy of the system.

The success of the Galerkin finite element method in structural mechanics, heat conduction and other problems of the potential type provided, in the early 1970s, a strong impetus for the utilization of the method in the simulation of problems in fluid dynamics. It was thought that the significant advantages gained in structural mechanics and diffusion-type problems would again be open to exploitation in the area of fluid flow simulation. Actually, this proved to be an optimistic point of view, especially with regard to modeling convection-dominated transport phenomena.

The main difficulty was due to the presence of convection operators in the formulation of flow problems based on kinematical descriptions other than Lagrangian. Convection operators are, in fact, non-symmetric and thus the best approximation property in the energy norm of the Galerkin method, which is the basis for success in symmetric cases, is lost when convection dominates the transport process.

In practice, solutions to convection-dominated transport problems by the Galerkin method are often corrupted by spurious node-to-node oscillations. These can only be removed by severe mesh (and time-step) refinement which clearly undermine the practical utility of the method. This has motivated the development of alternatives to the standard Galerkin formulation which preclude oscillations without requiring mesh or time-step refinement. Such alternatives are called stabilization techniques and have provided a major breakthrough in finite element modeling of problems in fluid dynamics.

In truly transient situations, another important issue is to ensure the proper coupling between spatial and temporal approximations. In fact, a stable and accurate spatial representation will be quickly spoiled if the algorithm used for transporting the solution in time is not of comparable accuracy. Space–time coupling is indeed particularly crucial when convection dominates the transport process, due to the directional character of propagation of information in hyperbolic problems. Significant progress has also been achieved in recent years in the development of algorithms for accurately tracing the transient solution of highly convective transport problems.

1.2 SUBJECTS COVERED

The purpose of this book is to describe methods of finite element analysis for steady and time-dependent convection–diffusion and fluid dynamics problems. The intent is to provide an introduction to a variety of modern methods, while preserving a pedagogical character through the presentation of simple worked examples.

The present chapter starts with a review of the basic kinematical descriptions used in fluid mechanics and recalls the conservation laws for mass, momentum and energy in differential and integral forms. It then provides an introduction to the basic ingredients of the finite element analysis of flow problems.

Chapter 2 introduces stabilized finite element methods for steady convection-dominated transport problems (elliptic equations). The difficulties of Galerkin finite elements are first recognized. This allows the design of possible cures for the node-to-node oscillations. The first alternative formulations proposed in the early 1970s to improve the standard Galerkin method tried to reproduce in the finite element context the effect of upwind differencing used in the finite difference context to stabilize the oscillatory results obtained by central difference approximations. These early, but not fully satisfactory, developments were quickly followed by more convincing finite element procedures, such as the Streamline-Upwind Petrov–Galerkin (SUPG) and the Galerkin/Least-squares (GLS) methods. Such methods do enjoy interesting stability and consistency properties and are nowadays widely used by the finite element community for solving convection-dominated transport problems.

After presenting the difficulties and remedies of Galerkin finite elements in steady convection-dominated problems, transient problems are introduced. In fact, Chapter 3 is devoted to pure convection. The scalar linear first-order hyperbolic equation allows discussion of time-dependent situations. In these problems, the objective has been to develop spatially stable and time-accurate finite element methods that take into account the role of the characteristics in the wave-like solution of hyperbolic equations. This has favored the development of solution algorithms in which attention is focused on achieving a proper coupling between the spatial discretization provided by the finite element method and the time discretization. Methods in this class include various characteristic Galerkin techniques, some classical time-stepping schemes and Taylor–Galerkin methods. The concept of accuracy and numerical stability is introduced. Moreover, spatial formulations especially suited for hyperbolic problems (least-squares and discontinuous Galerkin) are also introduced. A simple model problem also motivates a brief introduction to more recent techniques such as space–time formulations.

Engineering practice goes beyond linear scalar equations. Chapter 4 extends the concepts of the previous chapter to systems of nonlinear equations. In fact, it is concerned with a particular problem: the finite element modeling of inviscid compressible flows governed by the Euler equations of gas dynamics. Moreover, this extension allows discussion of the specificities of numerical methods to capture shocks. First, a brief review of the basic mathematical properties of nonlinear hyperbolic equations is presented. Second, a simple two-step procedure is introduced for the explicit integration of the governing conservation equations of mass, momentum and energy. It ensures second-order accuracy in the smooth part of the flow and, at the same time, allows easy incorporation of a modulated dissipation to avoid oscillatory results in the vicinity of shocks and other discontinuities in the flow. Then, various high-resolution shock-capturing techniques are described and their implementation in the finite element context is illustrated by several worked examples. The chapter closes with a discussion on the use of the Arbitrary Lagrangian–Eulerian (ALE) description for the finite element simulation of problems involving fluid–structure interaction. Both academic and industrial examples are proposed to illustrate the flexibility of the ALE technique in the modeling of coupled transient dynamic problems.

Once the basis of time integration (hyperbolic equations), Chapters 3 and 4, and spatial stabilization in steady problems (elliptic equations), Chapter 2, have been discussed, both methodologies converge in transient convection–diffusion problems (parabolic equations). Chapter 5 is still concerned with accurate time integration but has to deal with the second-order spatial operator introduced by the diffusion. This allows the incorporation in the transient schemes of the spatial stabilization introduced in Chapter 2, and, moreover, discussion of specific time integration techniques for convection–diffusion problems in order to obtain high-order accurate schemes.

The generalization to nonlinear systems of equations is done in Chapter 6. It provides an introduction to the finite element modeling of incompressible viscous flows governed by the Navier–Stokes equations. And consequently, apart from the numerical difficulties due to the presence of the nonlinear convective term, the incompressibility condition is also a major issue in this chapter. The problem is formulated in the primitive variables, namely velocity and pressure. Mixed and penalty methods in the framework of Stokes and Navier–Stokes equations are introduced. And a brief account is given of stabilization procedures capable of rendering convergent mixed finite element formulations which are unstable in the traditional Galerkin approach. To treat unsteady incompressible flows, a basic fractional-step projection method is introduced and some variants of it are discussed. Emphasis is placed on the treatment of the boundary conditions in each step of the time integration procedure and on the stable treatment of the pressure/incompressibility phase. The chapter closes with applications of the fractional-step method to forced and natural convection problems.

1.3 KINEMATICAL DESCRIPTIONS OF THE FLOW FIELD

In this section and the next one we summarize the continuum mechanics concepts that are needed for the mathematical description of flow problems. Classical references for the basic theory of fluid mechanics are Batchelor (1999), Landau and Lifshitz (1959) and Lamb (1993).

An important consideration when simulating fluid flow problems by any numerical method is the choice of an appropriate *kinematical description* of the flow field. The algorithms of continuum mechanics make use of three distinct types of description of motion: the Lagrangian description, the Eulerian description and the ALE description.

Lagrangian algorithms, in which each individual node of the computational mesh follows the associated material particle during motion, are mainly used in structural mechanics. Classical applications of the Lagrangian description in large deformation problems are the simulation of vehicle crash tests and the modeling of metal forming operations. In these applications, the Lagrangian algorithms are used in combination with both solid and structural (beam, plate, shell) elements. Numerical solutions are often characterized by large displacements and deformations and history-dependent constitutive relations are employed to describe elasto-plastic and visco-plastic material behavior. The Lagrangian description allows easy tracking of free surfaces and interfaces between different materials. 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 mechanics. Here, the computational mesh is fixed and the fluid moves with respect to the grid. The Eulerian formulation facilitates the treatment of large distortions in the fluid motion and is indispensable for

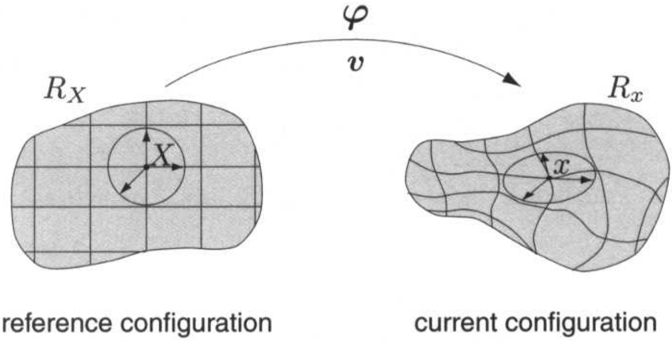


Fig. 1.1 Lagrangian description of motion.

the simulation of turbulent flows. Its handicap is the difficulty to follow free surfaces and interfaces between different materials or different media (e.g., fluid–fluid and fluid–solid interfaces).

ALE algorithms are particularly useful in flow problems involving large distortions in the presence of mobile and deforming boundaries. Typical examples are problems describing the interaction between a fluid and a flexible structure and the simulation of metal forming processes. The key idea in the ALE formulation is the introduction of a computational mesh which can move with a velocity independent of the velocity of the material particles. With this additional freedom with respect to the Eulerian and Lagrangian descriptions, the ALE method succeeds to a certain extend in minimizing the problems encountered in the classical kinematical descriptions, while combining at best their respective advantages.

1.3.1 Lagrangian and Eulerian descriptions

Two domains are commonly used in continuum mechanics: the material domain $R_X \subset \mathbb{R}^{\mathrm{n_{sd}}}$, with $\mathrm{n_{sd}}$ 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 1.1, 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, $R_{\boldsymbol{X}}$.

In the *total Lagrangian formulation*, $R_{\boldsymbol{X}}$ is considered fixed and it corresponds usually to the configuration of the continuum at the initial time. In the *updated Lagrangian formulation*, the reference configuration changes during the calculation and generally corresponds to the configuration relative to the previous time (or load) step.

The motion of the material points relates the material coordinates, $\boldsymbol{X}$, to the spatial ones, $\boldsymbol{x}$. It is defined by an application φ such that

$$\begin{aligned} \varphi \,:\, R_{\boldsymbol{X}} \times [t_0, t_{\text{final}}[&\longrightarrow R_{\boldsymbol{x}} \times [t_0, t_{\text{final}}[\\ (\boldsymbol{X}, t) &\longmapsto \varphi(\boldsymbol{X}, t) = (\boldsymbol{x}, t), \end{aligned} \tag{1.1}$$

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

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

which explicitly states the particular nature of φ: first, the spatial coordinates $\boldsymbol{x}$ depend on both 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 φ defines a configuration in the spatial domain. It is convenient to employ a matrix representation for the gradient of φ,

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

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

$$\boldsymbol{v}(\boldsymbol{X}, t) = \left.\frac{\partial \boldsymbol{x}}{\partial t}\right|_{\boldsymbol{X}}, \tag{1.2}$$

with $\Big|_{\boldsymbol{X}}$ meaning "holding $\boldsymbol{X}$ fixed".

Obviously, the one-to-one mapping φ must verify $\det(\partial\boldsymbol{x}/\partial\boldsymbol{X}) > 0$ (non-zero to impose a one-to-one correspondence and positive to avoid change of orientation in 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 $(\boldsymbol{X}, t) = \varphi^{-1}(\boldsymbol{x}, t)$, 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. These concepts are discussed in detail by Bonet and Wood (1997) in their excellent textbook on nonlinear continuum mechanics for finite element analysis. 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 due to the relative motion between the deforming material and the computational grid, see Remark 1.1. As will be seen in Chapter 2, this presents numerical difficulties, but permits 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.

Remark 1.1 (Material and spatial time derivatives). In order to relate the time derivative in the material and spatial domain let a scalar physical quantity be described by $f(\boldsymbol{x}, t)$ and $f^{**}(\boldsymbol{X}, t)$ in the spatial and material domains, respectively. Asterisks are employed to emphasize that the functional forms are, in general, different. Since the particle motion φ is a mapping, $f(\boldsymbol{x}, t)$ and $f^{**}(\boldsymbol{X}, t)$ can be related as

$$f^{**}(\boldsymbol{X}, t) = f\big(\varphi(\boldsymbol{X}, t), t\big) \quad \text{or} \quad f^{**} = f \circ \varphi.$$

The gradient of this expression can be easily computed as

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

which is amenable to the matrix form

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

which renders, after block multiplication, an obvious first expression, 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:

$$\frac{\partial f^{**}}{\partial t} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial \boldsymbol{x}} \boldsymbol{v}.$$

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

$$\left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{X}} = \left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{x}} + \boldsymbol{v} \cdot \boldsymbol{\nabla} f, \qquad \text{or} \qquad \frac{df}{dt} = \frac{\partial f}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} f, \tag{1.3}$$

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 the spatial (laboratory) system. Moreover, in order not to overload the rest of the text with notation, except for specific sections, the material time derivative is denoted as

$$\frac{d\cdot}{dt} := \left.\frac{\partial\cdot}{\partial t}\right|_{\boldsymbol{X}},$$

and the spatial time derivative as

$$\frac{\partial\cdot}{\partial t} := \left.\frac{\partial\cdot}{\partial t}\right|_{\boldsymbol{x}}.$$

1.3.2 ALE description of motion

The above review of the classical Lagrangian and Eulerian descriptions has highlighted the advantages and drawbacks of each individual formulation. It has also shown the potential interest of a generalized description capable of combining at best the interesting aspects of the classical mesh descriptions, while minimizing as far as possible their drawbacks. Such a generalized description is termed an ALE description. ALE methods were first proposed in the finite difference context where original developments were made, among others, by Noh (1964), Trulio (1966) and Hirt, Amsden and Cook (1974); this last contribution was 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, Fasoli-Stella and Giuliani (1977), Belytschko, Kennedy and Schoeberle (1978), Belytschko and Kennedy (1978) and Hughes, Liu and Zimmermann (1978).

In the ALE description of motion, neither the material $R_{\boldsymbol{X}}$ nor the spatial $R_{\boldsymbol{x}}$ configuration is taken as the reference. Thus, a third domain is needed: the referential configuration R_{χ} where reference coordinates χ are introduced to identify the grid points. Figure 1.2 shows these domains and the one-to-one transformations relating the configurations. The referential domain R_{χ} is mapped into the material and spatial domains by $\boldsymbol{\Psi}$ and $\boldsymbol{\Phi}$ respectively. The particle motion φ may then be expressed as $\varphi = \boldsymbol{\Phi} \circ \boldsymbol{\Psi}^{-1}$, clearly showing that, of course, the three mappings $\boldsymbol{\Psi}$, $\boldsymbol{\Phi}$ and φ are not independent.

The mapping $\boldsymbol{\Phi}$ 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

$$\begin{aligned} \boldsymbol{\Phi} : R_{\chi} \times [t_0, t_{\text{final}}[&\longrightarrow R_{\boldsymbol{x}} \times [t_0, t_{\text{final}}[\\ (\chi, t) &\longmapsto \boldsymbol{\Phi}(\chi, t) = (\boldsymbol{x}, t), \end{aligned} \tag{1.4}$$

and its gradient is

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

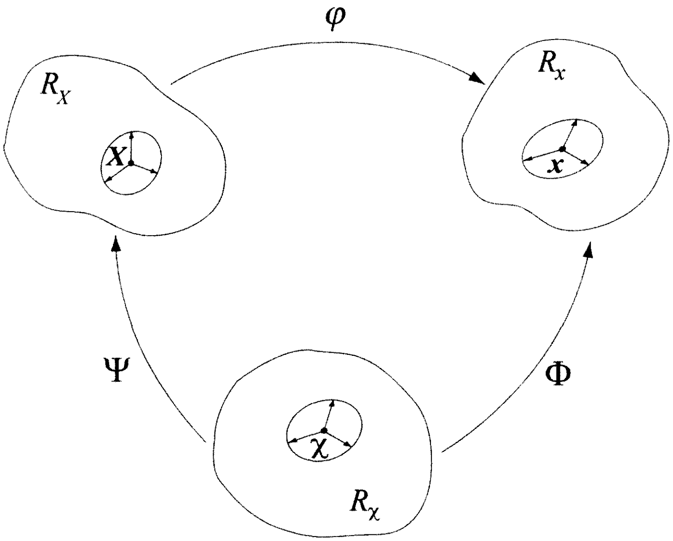


Fig. 1.2 The motion of the ALE computational mesh is independent of the material motion.

where now the mesh velocity $\hat{v}$ is involved,

$$\hat{v}(\chi, t) = \left.\frac{\partial x}{\partial t}\right|_{\chi}. \tag{1.5}$$

Note that both the material and the mesh move with respect to the laboratory. Thus, the corresponding material and mesh velocities have been defined, deriving with respect to time the equations of material motion and mesh motion respectively, see equations (1.2) and (1.5).

Finally, regarding Ψ, it is convenient to represent directly its inverse Ψ^{-1},

$$\begin{aligned} \Psi^{-1} : R_X \times [t_0, t_{\text{final}}[&\longrightarrow R_\chi \times [t_0, t_{\text{final}}[\\ (X, t) &\longmapsto \Psi^{-1}(X, t) = (\chi, t), \end{aligned} \tag{1.6}$$

and its gradient is

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

where velocity w is defined as

$$w = \left.\frac{\partial \chi}{\partial t}\right|_{X} \tag{1.7}$$

and can be interpreted as the particle velocity in the referential domain, since it measures the time variation of the referential coordinate χ holding the material particle

$\boldsymbol{X}$ fixed. The relation between velocities $\boldsymbol{v}$, $\hat{\boldsymbol{v}}$ and $\boldsymbol{w}$ can be obtained by differentiating $\boldsymbol{\varphi} = \boldsymbol{\Phi} \circ \boldsymbol{\Psi}^{-1}$,

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

or, in matrix format,

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

which yields, after block multiplication,

$$\boldsymbol{v} = \hat{\boldsymbol{v}} + \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\chi}}\boldsymbol{w}.$$

This equation may be rewritten as

$$\boldsymbol{c} := \boldsymbol{v} - \hat{\boldsymbol{v}} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\chi}}\boldsymbol{w}, \tag{1.8}$$

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

Remark 1.2. The convective velocity $\boldsymbol{c}$, see (1.8), should not be confused with $\boldsymbol{w}$, see (1.7). As stated before, $\boldsymbol{w}$ is the particle velocity as seen from the referential domain R_{χ}, 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 (1.8) 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.

Remark 1.3. After the fundamentals on ALE kinematics have been presented, it should be remarked that both Lagrangian and Eulerian formulations may be obtained as particular cases. With the choice $\boldsymbol{\Psi} = \boldsymbol{I}$, (1.6) reduces to $\boldsymbol{X} \equiv \boldsymbol{\chi}$ and a Lagrangian description results: the material and mesh velocities, equations (1.2) and (1.5), coincide, and the convective velocity $\boldsymbol{c}$, see (1.8), is null (there are no convective terms in the conservation laws).

If, on the other hand, $\boldsymbol{\Phi} = \boldsymbol{I}$, (1.4) simplifies into $\boldsymbol{x} \equiv \boldsymbol{\chi}$, thus implying an Eulerian description: a null mesh velocity is obtained from equation (1.5) and the convective velocity $\boldsymbol{c}$ is simply identical to the material velocity $\boldsymbol{v}$.

Remark 1.4 (Evaluation of the grid velocity). 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. In practice, such an algorithm selects the grid velocity to adapt the motion of the computational mesh to the peculiarities of the problem under investigation, trying to minimize both the squeeze and the distortion of the elements. See for instance the ALE mesh displacement algorithm developed by Giuliani (1982). When the ALE description is employed as a tool to adapt a finite element mesh to particular needs, for instance, to properly capture strong solution gradients, an error indicator or an error estimator is normally used to drive the algorithm in charge of computing the mesh velocities, see, for instance, Huerta et al. (1999).

1.3.3 The fundamental ALE equation

In order to express the conservation laws in an ALE framework, a relation between time derivatives is needed. In fact, the so-called *total* (or material) time derivatives, which are inherent to conservation laws, must be related to referential time derivatives. The relation presented in Remark 1.1 must be therefore extended in order to include the referential time derivative. Let a scalar physical quantity be described by $f(x, t)$, $f^*(\chi, t)$ and $f^{**}(\boldsymbol{X}, t)$ in the spatial, referential and material domains respectively. Asterisks are employed to emphasize that the functional forms are, in general, different.

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

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

and its gradient can be easily computed as

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

or, in matrix form,

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

which renders, after block multiplication,

$$\frac{\partial f^{**}}{\partial t} = \frac{\partial f^*}{\partial t} + \frac{\partial f^*}{\partial \chi} w.$$

Note that this equation relates the material and the referential time derivatives. However, it also requires the evaluation of the gradient 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 (1.8), the previous equation may be rearranged into

$$\frac{\partial f^{**}}{\partial t} = \frac{\partial f^{*}}{\partial t} + \frac{\partial f}{\partial \boldsymbol{x}}\boldsymbol{c}.$$

Dropping the asterisks to ease the notation, the fundamental ALE relation between material time derivatives, referential time derivatives and spatial gradient is finally cast as

$$\left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{X}} = \left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{\chi}} + \frac{\partial f}{\partial \boldsymbol{x}}\boldsymbol{c} = \left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{\chi}} + \boldsymbol{c}\cdot\boldsymbol{\nabla} f, \tag{1.9}$$

which can be interpreted in the usual way: the variation of the physical quantity f for a given particle $\boldsymbol{X}$ is the local variation (i.e., with respect to the reference $\boldsymbol{\chi}$) plus a convective term taking into account the relative motion between the material and the reference system. This equation is equivalent to (1.3) but in the ALE formulation; that is, when $(\boldsymbol{\chi}, t)$ is the reference.

1.3.4 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 therefore 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 the *Reynolds transport theorem*:

$$\frac{d}{dt}\int_{V_t} f(\boldsymbol{x}, t)\, dV = \int_{V_c \equiv V_t} \frac{\partial f(\boldsymbol{x}, t)}{\partial t}\, dV + \int_{S_c \equiv S_t} f(\boldsymbol{x}, t)\, \boldsymbol{v}\cdot\boldsymbol{n}\, dS, \tag{1.10}$$

which holds for smooth functions $f(\boldsymbol{x}, t)$. The volume integral on the r.h.s. 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 on the r.h.s. of (1.10) 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 $V_c \equiv V_t$. A similar expression holds for the volume integral of a vector or tensor quantity.

An analogous formula can be deduced in the ALE context, that is with a referential time derivative. In this case, however, $\boldsymbol{v}$ is no longer the material velocity, as will be seen in Section 1.4.5, see also Huerta and Liu (1988).

Equation (1.10) constitutes the starting point for the derivation of the ALE integral form of the conservation equations of mass, momentum and energy, which are discussed next.

1.4 THE BASIC CONSERVATION EQUATIONS

To serve as an introduction to the discussion of finite element models for both incompressible and compressible flow problems, we recall in this section the differential and integral forms of the conservation equations for mass, momentum and energy.

1.4.1 Mass equation

A fundamental law of Newtonian mechanics is the conservation of the mass contained in a material volume. The law of mass conservation for a varying material volume V_t occupied by fluid is given by

$$0 = \frac{dM}{dt} = \frac{d}{dt}\int_{V_t} \rho \, dV,$$

where ρ is the fluid density. Applying to this integral expression the formula (1.10) for the rate of change of integrals over a moving volume and the divergence theorem, one obtains

$$0 = \frac{dM}{dt} = \int_{V_t} \frac{\partial \rho}{\partial t}\, dV + \int_{S_t} \rho \boldsymbol{v} \cdot \boldsymbol{n} \, dS = \int_{V_t} \Big(\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{v}) \Big) \, dV.$$

Since this relation is valid for all choices of the volume V_t, the integrand must be identically zero. Hence

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{v}) = 0 \tag{1.11}$$

at all points in the fluid. Equation (1.11) is called the *mass-conservation equation*, or *continuity equation*. A different form of equation (1.11) is obtained by expanding the divergence term and noting that two of the terms together make up the material derivative of the density:

$$\frac{d\rho}{dt} + \rho \boldsymbol{\nabla} \cdot \boldsymbol{v} = 0. \tag{1.12}$$

1.4.2 Momentum equation

The *momentum equation*, also termed the *equation of motion*, is a relation equating the rate of change of momentum of a selected portion of fluid and the sum of all forces acting on that portion of fluid. For the portion of fluid of volume V_t enclosed by the *material surface* S_t, the momentum is $\int_{V_t} \rho \boldsymbol{v} \, dV$ and making use of the Reynolds transport theorem in vector form, that is (1.10), its rate of change is found to be

$$\begin{aligned} \frac{d}{dt}\int_{V_t} \rho \boldsymbol{v} \, dV &= \int_{V_t} \frac{\partial \rho \boldsymbol{v}}{\partial t} \, dV + \int_{S_t} (\rho \boldsymbol{v} \otimes \boldsymbol{v}) \cdot \boldsymbol{n} \, dS \\ &= \int_{V_t} \Big(\frac{\partial \rho \boldsymbol{v}}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{v} \otimes \boldsymbol{v}) \Big) \, dV, \end{aligned} \tag{1.13}$$

where the notation $\boldsymbol{v} \otimes \boldsymbol{v}$ denotes the tensor $[v_i v_j]$, $i, j = 1, \ldots, \mathrm{n_{sd}}$. Moreover, the i-th component of the divergence of a second-rank tensor $\mathbf{T}$ is

$$[\boldsymbol{\nabla} \cdot \mathbf{T}]_i := \sum_{j=1}^{\mathrm{n_{sd}}} \frac{\partial T_{ij}}{\partial x_j} \qquad \text{for } i = 1, \ldots, \mathrm{n_{sd}}.$$

Making use of the continuity equation (1.11) and expression (1.3) for the material time derivative, equation (1.13) can be transformed to

$$\frac{d}{dt}\int_{V_t} \rho \boldsymbol{v} \, dV = \int_{V_t} \rho \frac{d\boldsymbol{v}}{dt} \, dV, \tag{1.14}$$

which is simply the sum of the product of mass and acceleration for all the elements dV of the material volume V_t.

In general, a portion of fluid is acted upon by both volume and surface forces. We denote by $\boldsymbol{b}$ the volume force per unit mass of fluid, so that the total volume force on the selected portion of fluid is $\int_{V_t} \rho \boldsymbol{b} \, dV$. On the other hand, the i-component of the surface force exerted across a surface element of area dS and normal $\boldsymbol{n}$ is given by $\sigma_{ij} \, n_j \, dS$, so that the total force exerted on the selected portion of fluid by the surrounding matter can be expressed in terms of the Cauchy stress $\boldsymbol{\sigma}$ as

$$\int_{S_t} \sigma_{ij} \, n_j \, dS = \int_{V_t} \frac{\partial \sigma_{ij}}{\partial x_j} dV \qquad \text{or} \qquad \int_{S_t} \boldsymbol{\sigma} \cdot \boldsymbol{n} \, dS = \int_{V_t} \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} dV.$$

In the previous expression use has been made of the summation convention on repeated indices. It will be used when the limits of the summation are clear as in this case (from 1 to $\mathrm{n_{sd}}$). However, in this introductory chapter the summation will be used more than strictly necessary in order to preclude misinterpretation.

The momentum balance for the selected material volume of fluid, which accounts for both previous actions, is expressed by

$$\int_{V_t} \rho \frac{d\boldsymbol{v}}{dt} dV = \int_{V_t} \rho \boldsymbol{b} \, dV + \int_{V_t} \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} dV.$$

This integral relation holds for all choices of the material volume V_t. Thus,

$$\rho \frac{d\boldsymbol{v}}{dt} = \rho \boldsymbol{b} + \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} \tag{1.15}$$

at all points of the fluid. This equation is termed the *equation of motion*; making use of (1.3) it becomes

$$\rho\frac{\partial \boldsymbol{v}}{\partial t} + \rho(\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v} = \rho\boldsymbol{b} + \boldsymbol{\nabla}\cdot\boldsymbol{\sigma}, \quad \text{or} \quad \frac{\partial \rho\boldsymbol{v}}{\partial t} = \rho\boldsymbol{b} + \boldsymbol{\nabla}\cdot(\boldsymbol{\sigma} - \rho\,\boldsymbol{v}\otimes\boldsymbol{v}). \quad (1.16)$$

This last expression is the conservation form of the momentum equation.

Substituting equation (1.15) into (1.14) yields the following form of the rate of change of momentum for a material volume:

$$\frac{d}{dt}\int_{V_t}\rho\boldsymbol{v}\,dV = \int_{V_t}\rho\frac{d\boldsymbol{v}}{dt}\,dV = \int_{V_t}(\rho\boldsymbol{b} + \boldsymbol{\nabla}\cdot\boldsymbol{\sigma})\,dV.$$

For a fluid volume V (usually called control volume) whose position is *fixed* relative to the coordinate axes (i.e., fixed in the spatial domain), the rate of change of momentum is instead given from (1.16) by

$$\frac{\partial}{\partial t}\int_{V}\rho\boldsymbol{v}\,dV = \int_{V}\frac{\partial\rho\boldsymbol{v}}{\partial t}dV = \int_{V}(\rho\boldsymbol{b} + \boldsymbol{\nabla}\cdot\boldsymbol{\sigma})\,dV - \int_{S}\rho\boldsymbol{v}\,(\boldsymbol{v}\cdot\boldsymbol{n})dS.$$

1.4.3 Internal energy equation

In incompressible flow problems, the energy equation is not required to solve the momentum and continuity equations. Its form is given here in view of its use in Chapter 4, where the finite element modeling of compressible flow problems is addressed.

Consider therefore the energy balance for the fluid of volume V_t contained within a *material surface* S_t. Work is being done on this mass of fluid by both volume and surface forces, and it may also be gaining energy by heat conduction across the boundary. Some of this total gain of energy is manifested as an increase in the kinetic energy of the fluid, and the remainder, according to the first law of thermodynamics, appears as an increase in the internal energy of the fluid.

The first law of thermodynamics states that the increase of the internal energy per unit mass of the fluid, Δe, is the sum

$$\Delta e = q + w,$$

where q denotes the gain of heat per unit mass and w represents the work performed on the fluid per unit mass. Herein, we shall assume that the fluid is thermally isolated, so that no exchange of heat can occur, $q = 0$.

The rate at which work is being done on the fluid in the material volume V_t is the sum of a contribution

$$\int_{V_t}\boldsymbol{v}\cdot\rho\boldsymbol{b}\,dV = \int_{V_t}v_i\,\rho b_i\,dV$$

from the applied body force, and a contribution

$$\int_{S_t}\boldsymbol{v}\cdot(\boldsymbol{\sigma}\cdot\boldsymbol{n})dS = \int_{S_t}v_i\,\sigma_{ij}\,n_j\,dS = \int_{V_t}\frac{\partial(v_i\,\sigma_{ij})}{\partial x_j}dV = \int_{V_t}\boldsymbol{\nabla}\cdot(\boldsymbol{\sigma}\cdot\boldsymbol{v})dV$$

from the surface forces exerted by the surrounding fluid. Thus from these two expressions the total rate of work on an arbitrary material element is

$$\boldsymbol{v}\cdot\boldsymbol{b}+\frac{\boldsymbol{v}}{\rho}\cdot(\nabla\cdot\boldsymbol{\sigma})+\frac{\boldsymbol{\sigma}}{\rho}:\nabla\boldsymbol{v}=v_i\,b_i+\frac{v_i}{\rho}\frac{\partial\sigma_{ij}}{\partial x_j}+\frac{\sigma_{ij}}{\rho}\frac{\partial v_i}{\partial x_j}$$

per unit mass of fluid. Using the equation of motion (1.15), we may rewrite the rate of work per unit mass of fluid as

$$\boldsymbol{v}\cdot\frac{d\boldsymbol{v}}{dt}+\frac{\boldsymbol{\sigma}}{\rho}:\nabla\boldsymbol{v}=v_i\,\frac{dv_i}{dt}+\frac{\sigma_{ij}}{\rho}\frac{\partial v_i}{\partial x_j}.$$

The first term is clearly related to the gain in kinetic energy, while the second term represents the rate of work w done in deforming the fluid element. In the absence of heat transfer effects, the rate of change of internal energy per unit mass of a material element of fluid is thus given by

$$\frac{de}{dt}=\frac{\sigma_{ij}}{\rho}\frac{\partial v_i}{\partial x_j}=\frac{\boldsymbol{\sigma}}{\rho}:\nabla\boldsymbol{v}. \tag{1.17}$$

For an *inviscid fluid* the stress reduces to the static pressure (constitutive law),

$$\sigma_{ij}=-p\,\delta_{ij}\,,\quad\text{or }\boldsymbol{\sigma}=-p\boldsymbol{I},$$

where $\boldsymbol{I}$ is the identity tensor (same dimensions as $\boldsymbol{\sigma}$ in this case), and the *internal energy equation* then takes the form

$$\rho\frac{de}{dt}+p\,\nabla\cdot\boldsymbol{v}=0, \tag{1.18}$$

or, making use of (1.12),

$$\frac{d\rho e}{dt}+(p+\rho e)\nabla\cdot\boldsymbol{v}=0.$$

In terms of the partial time derivative, using (1.3), the internal energy equation reads

$$\frac{\partial\rho e}{\partial t}+\nabla\cdot(\rho e\,\boldsymbol{v})+p\nabla\cdot\boldsymbol{v}=0. \tag{1.19}$$

In a spatial representation, the rate of change of the internal energy for a material volume of an inviscid fluid is then given from the Reynolds transport theorem, (1.10),

$$\frac{d}{dt}\int_{V_t}\rho e\,dV=\int_{V_t}\frac{\partial\rho e}{\partial t}dV+\int_{S_t}\rho e(\boldsymbol{v}\cdot\boldsymbol{n})dS=\int_{V_t}\frac{\partial\rho e}{\partial t}dV+\int_{V_t}\nabla\cdot(\rho e\boldsymbol{v})dV.$$

Using (1.19) in the previous equation yields

$$\frac{d}{dt}\int_{V_t}\rho e\,dV=-\int_{V_t}p\,\nabla\cdot\boldsymbol{v}\,dV.$$

For a control fluid volume whose position is *fixed* relative to the coordinate axes (in the spatial domain), the rate of change of internal energy is instead given in the form

$$\frac{\partial}{\partial t}\int_{V}\rho e\,dV=\int_{V}\frac{\partial\rho e}{\partial t}dV=-\int_{V}p\,\nabla\cdot\boldsymbol{v}\,dV-\int_{S}\rho e(\boldsymbol{v}\cdot\boldsymbol{n})dS.$$

1.4.4 Total energy equation

As shown by the structure of equation (1.19), the internal energy equation is not a conservation equation of the type

$$\frac{\partial q}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{f} = 0.$$

In fact, the conserved quantity is the total energy. The total energy per unit mass of the fluid is the sum

$$E = e + \frac{1}{2}||\boldsymbol{v}||^2$$

of the internal energy, e, and the kinetic energy. The rate of change of the total energy per unit mass of a material element is thus

$$\frac{dE}{dt} = \frac{de}{dt} + \boldsymbol{v} \cdot \frac{d\boldsymbol{v}}{dt}. \tag{1.20}$$

For the particular case of an inviscid fluid, making use of the momentum equation (1.15) and of the internal energy equation (1.18), it becomes

$$\rho \frac{dE}{dt} = -\boldsymbol{\nabla} \cdot (p\,\boldsymbol{v}) + \boldsymbol{v} \cdot \rho \boldsymbol{b}$$

or

$$\frac{d}{dt}(\rho E) = \rho \frac{dE}{dt} + E\frac{d\rho}{dt} = -(p + \rho E)\boldsymbol{\nabla} \cdot \boldsymbol{v} + \boldsymbol{v} \cdot (\rho \boldsymbol{b} - \boldsymbol{\nabla} p).$$

In terms of the partial time derivative, the total energy equation reads

$$\frac{\partial}{\partial t}(\rho E) = -\boldsymbol{\nabla} \cdot \big((\rho E + p)\boldsymbol{v}\big) + \boldsymbol{v} \cdot \rho \boldsymbol{b}. \tag{1.21}$$

The rate of change of the total energy for a material volume is then given from (1.10) and (1.21) by

$$\begin{aligned}\frac{d}{dt}\int_{V_t} \rho E\, dV &= \int_{V_t} \frac{\partial}{\partial t}(\rho E)\, dV + \int_{S_t} \rho E\, \boldsymbol{v} \cdot \boldsymbol{n}\, dS \\ &= \int_{V_t} \boldsymbol{v} \cdot \rho \boldsymbol{b}\, dV - \int_{S_t} p\,\boldsymbol{v} \cdot \boldsymbol{n}\, dS.\end{aligned}$$

For a control fluid volume whose position is *fixed* relative to the coordinate axes, the rate of change of the total energy is also obtained using (1.21)

$$\frac{\partial}{\partial t}\int_V \rho E dV = \int_V \boldsymbol{v} \cdot \rho \boldsymbol{b}\, dV - \int_S (\rho E + p)\boldsymbol{v} \cdot \boldsymbol{n} dS = \int_V \boldsymbol{v} \cdot \rho \boldsymbol{b}\, dV - \int_S \rho H \boldsymbol{v} \cdot \boldsymbol{n} dS,$$

where

$$H = E + \frac{p}{\rho}$$

is the total specific enthalpy of the fluid.

1.4.5 ALE form of the conservation equations

The ALE differential form of the conservation equations is readily obtained from the corresponding material forms (1.12), (1.15), (1.17) and (1.20). The total derivatives are rewritten using (1.9) and the result is

$$\begin{aligned}
&\text{Mass:} && \left.\frac{\partial \rho}{\partial t}\right|_{\chi} + \boldsymbol{c}\cdot\boldsymbol{\nabla}\rho = -\rho\,\boldsymbol{\nabla}\cdot\boldsymbol{v},\\
&\text{Momentum:} && \rho\Big(\left.\frac{\partial \boldsymbol{v}}{\partial t}\right|_{\chi} + (\boldsymbol{c}\cdot\boldsymbol{\nabla})\boldsymbol{v}\Big) = \boldsymbol{\nabla}\cdot\boldsymbol{\sigma} + \rho\boldsymbol{b},\\
&\text{Internal energy:} && \rho\Big(\left.\frac{\partial e}{\partial t}\right|_{\chi} + \boldsymbol{c}\cdot\boldsymbol{\nabla}e\Big) = \boldsymbol{\sigma}:\boldsymbol{\nabla}\cdot\boldsymbol{v},\\
&\text{Total energy:} && \rho\Big(\left.\frac{\partial E}{\partial t}\right|_{\chi} + \boldsymbol{c}\cdot\boldsymbol{\nabla}E\Big) = \boldsymbol{\nabla}\cdot(\boldsymbol{\sigma}\cdot\boldsymbol{v}) + \boldsymbol{v}\cdot\rho\boldsymbol{b}.
\end{aligned} \tag{1.22}$$

The starting point for deriving the ALE integral form of the conservation equations is expression (1.10) applied to an *arbitrary* volume V_t whose boundary $S_t = \partial V_t$ moves with the mesh velocity $\hat{\boldsymbol{v}}$:

$$\left.\frac{\partial}{\partial t}\right|_{\chi}\int_{V_t} f(\boldsymbol{x},t)\,dV = \int_{V_t}\left.\frac{\partial f(\boldsymbol{x},t)}{\partial t}\right|_{\boldsymbol{x}} dV + \int_{S_t} f(\boldsymbol{x},t)\,\hat{\boldsymbol{v}}\cdot\boldsymbol{n}\,dS,$$

where, in this case, we have explicitly indicated that the time derivative in the first term of the r.h.s. is a spatial time derivative, as in (1.10). We then successively replace the scalar $f(\boldsymbol{x},t)$ by the fluid density ρ, momentum $\rho\boldsymbol{v}$, internal energy ρe and total energy ρE. Similarly, the spatial time derivative $\partial f/\partial t$ is substituted by the correponding expressions for the mass equation, see (1.11), for the momentum equation, see (1.16), for the internal energy and for the total energy. The end result is the following set of ALE integral forms:

$$\begin{aligned}
&\left.\frac{\partial}{\partial t}\right|_{\chi}\int_{V_t}\rho\,dV + \int_{S_t}\rho\,\boldsymbol{c}\cdot\boldsymbol{n}\,dS = 0,\\
&\left.\frac{\partial}{\partial t}\right|_{\chi}\int_{V_t}\rho\boldsymbol{v}\,dV + \int_{S_t}\rho\boldsymbol{v}\,\boldsymbol{c}\cdot\boldsymbol{n}\,dS = \int_{V_t}(\boldsymbol{\nabla}\cdot\boldsymbol{\sigma} + \rho\boldsymbol{b})\,dV,\\
&\left.\frac{\partial}{\partial t}\right|_{\chi}\int_{V_t}\rho e\,dV + \int_{S_t}\rho e\,\boldsymbol{c}\cdot\boldsymbol{n}\,dS = \int_{V_t}\boldsymbol{\sigma}:\boldsymbol{\nabla}\cdot\boldsymbol{v}\,dV,\\
&\left.\frac{\partial}{\partial t}\right|_{\chi}\int_{V_t}\rho E\,dV + \int_{S_t}\rho E\,\boldsymbol{c}\cdot\boldsymbol{n}\,dS = \int_{V_t}\big(\boldsymbol{\nabla}\cdot(\boldsymbol{\sigma}\cdot\boldsymbol{v}) + \boldsymbol{v}\cdot\rho\boldsymbol{b}\big)dV.
\end{aligned} \tag{1.23}$$

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 $\hat{\boldsymbol{v}} = \boldsymbol{v}$ (i.e., $\boldsymbol{c} = \boldsymbol{0}$), while the Eulerian description corresponds to selecting $\hat{\boldsymbol{v}} = \boldsymbol{0}$ (i.e., $\boldsymbol{c} = \boldsymbol{v}$).

The ALE differential and integral forms of the conservation equations derived in the present section will be used in Chapter 4 in connection with the finite element modeling of coupled fluid–structure interaction problems.

1.4.6 Closure of the initial boundary value problem

Once the governing equations are defined, boundary and initial conditions (the latter only in transient problems) must be adequately prescribed in order to close the problem. Three types of boundary conditions are used in this text: Dirichlet, Neumann and Robin. The last are also called mixed boundary conditions.

Dirichlet boundary conditions prescribe the value of the unknown function. Neumann conditions impose the normal gradient of the unknown function along the boundary. Robin boundary conditions prescribe a combination of the unknown function and its gradient.

Not all boundary conditions can be applied arbitrarily everywhere along the boundary of the domain. For instance, in hyperbolic problems Dirichlet boundary conditions cannot be imposed on the whole boundary, see Chapters 3 and 4. In every chapter we shall present the model problem to be studied, and in each case boundary conditions shall be presented and discussed in more detail.

1.5 BASIC INGREDIENTS OF THE FINITE ELEMENT METHOD

The application of the finite element method for solving a boundary value problem requires a certain number of basic ingredients that we shall briefly recall in this section on the basis of a simple model problem. Before that, we wish to introduce some notations and mathematical terminology used throughout the book. Standard finite element texts, such as, for instance, Strang and Fix (1973), Oden and Reddy (1976), Carey and Oden (1983), Temam (2001), Girault and Raviart (1986), Hughes (2000), Gunzburger (1989), Pironneau (1989) and Quarteroni and Valli (1994) provide a detailed exposition of the mathematical concepts, which are the basis of the finite element method.

1.5.1 Mathematical preliminaries

The process of spatial discretization by the finite element method rests upon the discrete representation of a weak integral form of the partial differential equation to be solved. The formulation and subsequent discretization of such an integral form requires the definition of some function spaces and associated norms, as well as the introduction of compact forms involving the functions pertaining to those spaces.

Consider a spatial region (or domain) $\Omega \subset \mathbb{R}^{\mathrm{n_{sd}}}$ with piecewise smooth boundary Γ. Here again, $\mathrm{n_{sd}} = 1, 2$ or 3 denotes the number of space dimensions. We shall use the notation

$$f : \bar{\Omega} \to \mathbb{R}$$

to state that for each spatial point $\boldsymbol{x} \in \bar{\Omega}$, $f(\boldsymbol{x}) \in \mathbb{R}$. $\bar{\Omega}$ denotes the closure of Ω, that is the union of the domain Ω with its boundary Γ: $\bar{\Omega} = \Omega \cup \Gamma$.

A function $f : \bar{\Omega} \to \mathbb{R}$ is said to be of class $C^m(\Omega)$ if all its derivatives up to order m exist and are continuous functions. For instance, the notation $f(x) \in C^m(]a, b[)$ indicates that $f(x)$ possesses m continuous derivatives for $x \in]a, b[$.

In the finite element analysis we work with integral equations and, thus, we are interested in functions belonging to larger spaces than C^m. As we will see, instead of requiring the m-th derivative to be continuous, we will require that its square is integrable. In fact, finite element functions should possess generalized derivatives (i.e., derivatives in the sense of distributions) and some integrability properties. Such classes of functions are particular examples of Sobolev function spaces. A detailed account concerning Sobolev spaces may be found in the book by Adams (1975).

1.5.1.1 Some useful Sobolev spaces We shall denote by $\mathcal{L}_2(\Omega)$ the space of functions that are square integrable over the domain Ω. This space is equipped with the standard inner product

$$(u,v) = \int_\Omega uv\, d\Omega \quad \text{and norm} \quad \|v\|_0 = (v,v)^{1/2}.$$

Next we describe a particular class of Sobolev spaces, those of square integrable functions and derivatives. This restriction suffices for our purposes throughout the book. For any non-negative integer k, we define the Sobolev space $\mathcal{H}^k(\Omega)$ using multi-index notation: given the n-tuple $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_{\mathrm{n_{sd}}}) \in \mathbb{N}^{\mathrm{n_{sd}}}$ and the non-negative integer $|\alpha| := \alpha_1 + \alpha_2 + \cdots + \alpha_{\mathrm{n_{sd}}}$,

$$\mathcal{H}^k(\Omega) = \left\{ u \in \mathcal{L}_2(\Omega) \,\middle|\, \frac{\partial^{|\alpha|} u}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \cdots \partial x_{\mathrm{n_{sd}}}^{\alpha_{\mathrm{n_{sd}}}}} \in \mathcal{L}_2(\Omega)\, \forall |\alpha| \leq k \right\}.$$

Therefore, $\mathcal{H}^k(\Omega)$ consists of square integrable functions all of whose derivatives of order up to k are also square integrable. $\mathcal{H}^k(\Omega)$ is equipped with the norm

$$\|u\|_k = \left(\sum_{s=0}^{k} \sum_{|\alpha|=s} \left\| \frac{\partial^{|\alpha|} u}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \cdots \partial x_{\mathrm{n_{sd}}}^{\alpha_{\mathrm{n_{sd}}}}} \right\|_0^2 \right)^{1/2}.$$

Note that $\mathcal{L}_2$ is, in fact, a Sobolev space, $\mathcal{H}^0(\Omega) = \mathcal{L}_2(\Omega)$, while the Sobolev space for $k = 1$ is defined by

$$\mathcal{H}^1(\Omega) = \left\{ v \in \mathcal{L}_2(\Omega) \mid \frac{\partial v}{\partial x_i} \in \mathcal{L}_2(\Omega)\ i = 1, \ldots, \mathrm{n_{sd}} \right\}.$$

This space is equipped with the inner product

$$(u,v)_1 = \int_\Omega \left(uv + \sum_{i=1}^{\mathrm{n_{sd}}} \frac{\partial u}{\partial x_i} \frac{\partial v}{\partial x_i} \right) d\Omega$$

and its induced norm

$$\|u\|_1 = \sqrt{(u,u)_1}.$$

We shall also frequently use the subspace

$$\mathcal{H}_0^1(\Omega) = \left\{ v \in \mathcal{H}^1(\Omega) \mid v = 0 \ \text{ on}\, \Gamma \right\},$$

the elements of which possess a square integrable first derivative over the domain Ω and vanish on its boundary Γ. Moreover, its inner product and norm coincide with those of $\mathcal{H}^1(\Omega)$.

Remark 1.5. Note that the Sobolev spaces used in what follows, namely $\mathcal{H}^0 = \mathcal{L}_2, \mathcal{H}^1$ and $\mathcal{H}^1_0$, are Hilbert spaces with their corresponding inner product (recall that a Hilbert space is a linear space with an inner product in which all Cauchy sequences are convergent sequences).

Remark 1.6. $\mathcal{H}^1_0$ is usually defined as the closure of $C^\infty_0(\Omega)$ (the set of all continuous functions with continuous derivatives whose support is a bounded subset of Ω) with respect to the norm of $||\cdot||_1$. That is, $\mathcal{H}^1_0(\Omega)$ is the set of all functions u in $\mathcal{H}^1(\Omega)$ such that u is the limit in $\mathcal{H}^1(\Omega)$ of a sequence $\{u_s\}^\infty_{s=1}$ whose u_s are in $C^\infty_0(\Omega)$.

1.5.1.2 Extension to vector-valued functions In the finite element analysis of flow problems consideration will be given not only to scalar functions (such as temperature or pressure) but also to vector-valued functions (such as fluid velocity). For vector-valued functions with m components, that is $\boldsymbol{u}, \boldsymbol{v} : \Omega \rightarrow \mathbb{R}^m$, the procedure is in fact essentially the same as for scalar functions.

Consider again a domain $\Omega \subset \mathbb{R}^{n_{sd}}$, $n_{sd} \geq 1$, and denote by $\boldsymbol{\mathcal{H}}^k(\Omega)$ or $[\mathcal{H}^k(\Omega)]^m$ the space of vector functions with m components

$$\boldsymbol{u} = (u_1, u_2, \ldots, u_m)$$

for which each component $u_i \in \mathcal{H}^k(\Omega)$, $1 \leq i \leq m$. The space $\boldsymbol{\mathcal{H}}^k(\Omega)$ is equipped with an inner product inducing the following norm

$$||\boldsymbol{u}||_k = \left(\sum_{i=1}^{m} ||u_i||^2_k\right)^{1/2} .$$

For the particular case of functions belonging to $\boldsymbol{\mathcal{L}}_2(\Omega) = \boldsymbol{\mathcal{H}}^0(\Omega)$, the inner product is given by

$$(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega \boldsymbol{u} \cdot \boldsymbol{v} \, d\Omega,$$

where there should be no ambiguity in using the same notation to represent the inner product of both scalar and vector-valued functions.

1.5.2 Trial solutions and weighting functions

To define the weak, or variational, form of the boundary value problems discussed in the present text, we need to define two classes, or collections, of functions: the *test* or weighting functions and the *trial* or admissible solutions. Here these spaces are defined in the context of the standard Galerkin formulation.

The first collection of functions, denoted by $\mathcal{V}$, is composed of *test* functions and consists of all functions which are square integrable, have square integrable first derivatives over the computational domain Ω, and vanish on the Dirichlet portion, Γ_D, of the boundary. It is defined as follows:

$$\mathcal{V} = \{w \in \mathcal{H}^1(\Omega) \mid w = 0 \quad \text{on}\,\Gamma_D\} \equiv \mathcal{H}^1_{\Gamma_D}(\Omega). \tag{1.24}$$

This is as previously noted a Sobolev space and its inner product and norm coincide with those of $\mathcal{H}^1(\Omega)$.

The second collection of functions is called the *trial solutions*. This collection is similar to the test functions except that these admissible functions are required to satisfy the Dirichlet conditions on Γ_D. This second collection is denoted by $\mathcal{S}$ and is defined by

$$\mathcal{S} = \{u \in \mathcal{H}^1(\Omega) \mid u = u_D \quad \text{on}\,\Gamma_D\} \equiv \mathcal{V} + \{\bar{u}_D\} \tag{1.25}$$

where $\bar{u}_D$ is any function in $\mathcal{H}^1(\Omega)$ such that $\bar{u}_D = u_D$ on Γ_D. Thus, $\mathcal{S}$ can be viewed as a translation of $\mathcal{V}$ and, consequently, it is an affine space. Note, for instance, that, for $u_D \neq 0$, the sum of two elements of $\mathcal{S}$ is not an element of $\mathcal{S}$. However, for homogeneous boundary conditions, $u_D = 0$, trial and test spaces coincide, $\mathcal{V} = \mathcal{S} = \mathcal{H}^1_0(\Omega)$.

The sets $\mathcal{S}$ and $\mathcal{V}$ clearly contain infinitely many functions. In the finite element method, $\mathcal{S}$ and $\mathcal{V}$ are approximated by convenient finite dimensional subsets of these collections which will be denoted by $\mathcal{S}^h$ and $\mathcal{V}^h$, respectively. These finite element spaces are characterized, among other things, by a partition of the domain.

At this point, we have to view the domain Ω as discretized into element domains. Let $\mathcal{T}^h(\Omega)$ be a regular partition, also called *triangulation*, of Ω into $\mathrm{n_{el}}$ convex subdomains $\Omega^e \neq \emptyset$, such that

$$\bar{\Omega} = \bigcup_{e=1}^{\mathrm{n_{el}}} \bar{\Omega}^e \qquad \text{and} \qquad \Omega^e \cap \Omega^f = \emptyset \qquad \text{for } e \neq f.$$

Each subdomain Ω^e has a piecewise smooth boundary $\Gamma_e = \partial\Omega^e$, and h is a characteristic mesh size ($\text{diam}(\Omega^e) \leq h$ for all elements).

The weighting functions $w^h \in \mathcal{V}^h$ vanish on Γ_D. The approximation u^h lies in $\mathcal{S}^h$ and satisfies, with the precision given by the characteristic mesh size h, the boundary condition u_D on Γ_D. In fact, along Γ_D we should have $u^h = u^h_D$; however, in order not to overload the notation u_D is also used instead of u^h_D unless it is necessary to explicitly show the precision of the boundary data. The interpolation spaces are defined as

$$\begin{aligned} \mathcal{V}^h &:= \{w \in \mathcal{H}^1(\Omega) \mid w|_{\Omega^e} \in \mathcal{P}_m(\Omega^e)\ \forall e \text{ and } w = 0 \text{ on } \Gamma_D\} \\ \mathcal{S}^h &:= \{u \in \mathcal{H}^1(\Omega) \mid u|_{\Omega^e} \in \mathcal{P}_m(\Omega^e)\ \forall e \text{ and } u = u_D \text{ on } \Gamma_D\}, \end{aligned} \tag{1.26}$$

where $\mathcal{P}_m$ is the ***finite element interpolating space.*** Note that $\mathcal{V}^h$ and $\mathcal{S}^h$ are finite dimensional subspaces of the spaces of test, $\mathcal{V}$, and trial, $\mathcal{S}$, functions. In several spatial dimensions three types of polynomial spaces are usually chosen: the set of polynomials, $\mathcal{P}_m$, of total degree $\leq m$ (usually defined over the reference triangle).

the set of polynomials, $\mathcal{Q}_m$, of degree $\leq m$ in each variable (defined over the reference square), or the set of polynomials $\mathcal{S}_m := \mathcal{P}_m \oplus \operatorname{span}\{x^m y, xy^m\}$ (known as serendipity or trunk spaces over the reference square). They all include a complete basis of the subspace of polynomials of degree m which characterizes the a priori convergence rate of the finite element approximation.

1.5.3 Compact integral forms

In establishing the weak, or variational, form of boundary value problems we shall make frequent use of the bilinear forms

$$a(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega \boldsymbol{\nabla u} : \boldsymbol{\nabla v} \, d\Omega \qquad \forall \boldsymbol{u}, \boldsymbol{v} \in \boldsymbol{\mathcal{H}}^1(\Omega), \tag{1.27a}$$

$$b(\boldsymbol{v}, q) = -\int_\Omega q \, \boldsymbol{\nabla} \cdot \boldsymbol{v} \, d\Omega \qquad \forall \boldsymbol{v} \in \boldsymbol{\mathcal{H}}^1(\Omega) \text{ and } q \in \mathcal{L}_2(\Omega), \tag{1.27b}$$

as well as of the trilinear form

$$c(\boldsymbol{v}; \boldsymbol{w}, \boldsymbol{u}) = \int_\Omega \boldsymbol{w} \cdot (\boldsymbol{v} \cdot \boldsymbol{\nabla}) \boldsymbol{u} \, d\Omega \qquad \forall \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w} \in \boldsymbol{\mathcal{H}}^1(\Omega), \tag{1.27c}$$

where the following notation has been introduced:

$$[\boldsymbol{\nabla u}]_{ij} = \frac{\partial u_i}{\partial x_j}, \qquad \text{for } i = 1, \ldots, m \text{ and } j = 1, \ldots, \mathrm{n_{sd}}$$

$$\boldsymbol{\nabla u} : \boldsymbol{\nabla v} = \sum_{i=1}^{m} \sum_{j=1}^{\mathrm{n_{sd}}} \frac{\partial u_i}{\partial x_j} \frac{\partial v_i}{\partial x_j} \quad \text{and} \quad \boldsymbol{w} \cdot (\boldsymbol{v} \cdot \boldsymbol{\nabla}) \boldsymbol{u} = \sum_{i=1}^{m} \sum_{j=1}^{\mathrm{n_{sd}}} w_i \, v_j \frac{\partial u_i}{\partial x_j}.$$

1.5.4 Strong and weak forms of a boundary value problem

The process of spatial discretization by the finite element method rests upon an integral form of the considered partial differential problem. This was the reason for introducing larger spaces than C^m based on integrability properties. Thus, the first task in a finite element analysis consists of formulating a (continuous) variational problem associated with the given partial differential equation and its boundary conditions.

Let us introduce a model boundary value problem that will illustrate the various steps in the practical implementation of the finite element method. Consider solving the Poisson equation

$$-\boldsymbol{\nabla}^2 u = s \quad \text{in } \Omega, \tag{1.28a}$$

where Ω is enclosed by a piecewise smooth boundary Γ, and $s \in C^0(\bar{\Omega})$ is a specified source term, which may depend on $\boldsymbol{x}$. The following notation has been introduced:

$$\boldsymbol{\nabla}^2 u := (\boldsymbol{\nabla} \cdot \boldsymbol{\nabla}) u = \sum_{i=1}^{\mathrm{n_{sd}}} \frac{\partial^2 u}{\partial x_i^2}.$$

We further assume that the value of the unknown u is prescribed on the Dirichlet portion Γ_D of the boundary,

$$u = u_D \quad \text{on } \Gamma_D; \tag{1.28b}$$

while the normal derivative of u is prescribed on the remaining Neumann portion Γ_N of the boundary Γ,

$$\frac{\partial u}{\partial n} := \boldsymbol{n} \cdot \boldsymbol{\nabla} u = h \quad \text{on } \Gamma_N. \tag{1.28c}$$

A function $u \in C^2(\Omega) \cap C^0(\bar{\Omega})$ that satisfies (1.28) is called a *classical solution* of the boundary value problem.

The first step in a ***weighted residual formulation*** leading to the finite element discretization of our model problem consists of formulating a ***weak (or variational) form*** of the boundary value problem. This is achieved by multiplying the governing equation (1.28a) by the weighting function w and integrating over the computational domain Ω:

$$-\int_\Omega w\, \boldsymbol{\nabla}^2 u \, d\Omega = \int_\Omega w\, s \, d\Omega.$$

Note that the continuity requirements on u still impose that it must be twice differentiable. Now, however, the second derivatives of u are not required to be continuous, they only need to be square integrable. Thus, it is sufficient that $u \in \mathcal{H}^2(\Omega)$. In any case, it is obvious that classical solutions of (1.28) will also verify this integral equation for all admissible functions w.

In order to derive the weak form and produce a natural Neumann boundary condition on Γ_N, we apply to the l.h.s. of the previous equation the Green–Gauss divergence theorem:

$$\begin{aligned} -\int_\Omega w\, \boldsymbol{\nabla}^2 u \, d\Omega &= -\int_\Omega \Big(\boldsymbol{\nabla} \cdot (w \boldsymbol{\nabla} u) - \boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} u \Big) d\Omega \\ &= \int_\Omega \boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} u \, d\Omega - \int_\Gamma w (\boldsymbol{n} \cdot \boldsymbol{\nabla} u) d\Gamma. \end{aligned}$$

Now the regularity requirements on w and u are modified; u is only differentiated once and w must be differentiable. In fact, its derivatives must be square integrable; thus, we should have $w \in \mathcal{H}^1(\Omega)$. If $w \in \mathcal{V}$, recall from (1.24) that $w = 0$ on Γ_D, and if we take into account the prescribed Neumann boundary condition (1.28c), we obtain the following weak form of our model problem:

$$\int_\Omega \boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} u \, d\Omega = \int_\Omega w\, s \, d\Omega + \int_{\Gamma_N} w\, h \, d\Gamma.$$

Note that the application of the divergence theorem has allowed us to naturally introduce the Neumann boundary condition on Γ_N. At the same time, it has removed the second-derivative terms of the Laplacian operator from the volume integral. This reduces the continuity requirements on u and allows us to select this function in $\mathcal{H}^1(\Omega)$, since only first derivatives appear under the integral sign.

Classical solutions of (1.28) will also satisfy this integral equation. But it only accounts for (1.28a) and (1.28c), not Dirichlet boundary conditions (1.28b). This is done by means of the proper choice of the space where u belongs, namely $\mathcal{S}$. Recall from (1.25) that every member of $\mathcal{S}$ satisfies the Dirichlet boundary condition $u = u_D$ on Γ_D. Noting that provided $u \in \mathcal{S}$ the Dirichlet boundary condition on Γ_D is satisfied, the weak form of our model problem can then be formally stated as follows:

$$\text{find } u \in \mathcal{S} \text{ such that } \quad a(w,u) = (w,s) + (w,h)_{\Gamma_N} \quad \forall w \in \mathcal{V}. \tag{1.29}$$

Here use has been made of the scalar version of (1.27a),

$$a(w,u) = \int_\Omega \nabla w \cdot \nabla u \, d\Omega, \tag{1.30}$$

and the linear functionals

$$(w,s) = \int_\Omega w\, s\, d\Omega, \quad \text{and} \quad (w,h)_{\Gamma_N} = \int_{\Gamma_N} w\, h\, d\Gamma.$$

By construction a classical solution u of (1.28) is a solution of the weak form (1.29). Let us show that a weak solution $u \in \mathcal{S}$ of (1.29) is unique; this is done by means of the Lax–Milgram lemma. Therefore, if we assume that a classical solution of (1.28) exists the same function u is a solution of both problems.

> **Remark 1.7.** Weak solutions can also be found in cases that do not have a classical solution. This is the case of many applications in which data do not have the required smoothness, for instance a discontinuous source term $s(\boldsymbol{x})$. In these circumstances the weak problem (1.29) can be formulated and, if the Lax–Milgram lemma can be applied, it admits a unique solution, but this solution is not a classical solution.

Theorem 1.1 (Lax–Milgram lemma). *Let $a(\cdot,\cdot)$ be a bilinear form on a Hilbert space $\mathcal{H}$ equipped with norm $||\cdot||_\mathcal{H}$. If $a(\cdot,\cdot)$ is continuous, that is*

$$\exists \gamma_1 > 0 \text{ such that } |a(w,v)| \leq \gamma_1 ||w||_\mathcal{H}\, ||v||_\mathcal{H} \qquad \forall w,v \in \mathcal{H},$$

and coercive (or $\mathcal{H}$-elliptic), that is

$$\exists \alpha > 0 \text{ such that } a(v,v) \geq \alpha ||v||^2_\mathcal{H} \qquad \forall v \in \mathcal{H},$$

then for all $l(\cdot)$ bounded linear mappings on $\mathcal{H}$ (thus continuous), that is

$$\exists \gamma_2 > 0 \text{ such that } |l(w)| \leq \gamma_2 ||w||_\mathcal{H} \qquad \forall w \in \mathcal{H},$$

there exists a unique $u \in \mathcal{H}$ such that

$$a(w,u) = l(w) \qquad \forall w \in \mathcal{H}$$

is satisfied.

A proof can be found elsewhere, see for instance Ciarlet (1978), Yosida (1995) or Quarteroni and Valli (1994).

In order to show that this theorem can be applied to (1.29) some further considerations are needed. First, note that both arguments of the bilinear form in Theorem 1.1 must belong to the the same Hilbert space. This is not the case in (1.29) unless, recalling the definition of $\mathcal{S}$ in (1.25), u_0 is defined as $u = u_0 + \bar{u}_D$ with $u_0 \in \mathcal{V}$. Replacing this decomposition in (1.29) the weak form becomes

$$\text{find } u_0 \in \mathcal{V} \text{ such that } a(w, u_0) = (w, s) + (w, h)_{\Gamma_N} - a(w, \bar{u}_D) \ \forall w \in \mathcal{V}. \quad (1.31)$$

Thus, in this case $\mathcal{H} = \mathcal{V} = \mathcal{H}^1_{\Gamma_D}(\Omega)$ and the associated norm is the $\mathcal{H}^1$ norm, $\|\cdot\|_{\mathcal{H}} = \|\cdot\|_1$.

Second, since s, h and $\bar{u}_D$ are data, we can define the linear operator

$$l(w) := (w, s) + (w, h)_{\Gamma_N} - a(w, \bar{u}_D).$$

And third, we need to check that the bilinear form defined in (1.30) is coercive and continuous, and that the linear form is bounded. The continuity of $a(\cdot, \cdot)$ and $l(\cdot)$ is easily verified assuming some regularity in the data and applying systematically the Cauchy–Schwartz inequality. However, coercivity of $a(\cdot, \cdot)$ is more involved in particular when Neumann or Robin boundary conditions are present. Quarteroni and Valli (1994, Sec. 6.1.2) give an excellent presentation of the conditions for existence and uniqueness of a solution, which is beyond the scope of this text.

Remark 1.8 (Symmetric bilinear form). If, under the assumptions of Theorem 1.1, the bilinear form $a(\cdot, \cdot)$ is symmetric, then it can define a scalar product on $\mathcal{H}$. Note that this is the case of (1.30) for functions in $\mathcal{V} = \mathcal{H}^1_{\Gamma_D}(\Omega)$. This inner product induces in $\mathcal{H}$ the so-called energy norm

$$\|u\| = \sqrt{a(u, u)},$$

which is a natural norm in the engineering finite element context. Moreover, in this case the weak problem can be viewed as a minimization problem as noted in the next remark.

Remark 1.9 (Variational principle). Under the assumptions of the previous remark, if we pose $w = \delta u$, where δu is a variation of the function u, then it can be shown that the weak form (1.29) emanates from the minimization of the quadratic functional

$$I(u) = \frac{1}{2} \int_\Omega (\boldsymbol{\nabla} u)^2 \, d\Omega - \int_\Omega u \, s \, d\Omega - \int_{\Gamma_N} u \, h \, d\Gamma,$$

or, equivalently,

$$I(u) = \frac{1}{2} a(u, u) - (s, u) - (h, u)_{\Gamma_N}.$$

Indeed, $\delta I = 0$ implies that

$$\int_\Omega (\boldsymbol{\nabla}\delta u)\cdot\boldsymbol{\nabla}(u)\,d\Omega - \int_\Omega \delta u\, s\, d\Omega - \int_{\Gamma_N} \delta u\, h\, d\Gamma = 0.$$

It follows that the solution of the boundary value problem (1.28) coincides with the value of u which minimizes the potential $I(u) \in \mathcal{H}^1_{\Gamma_D}(\Omega)$.

This is the case here for the Poisson equation, but also, for instance, in linear elasticity where the displacement field satisfying the differential equations of equilibrium of a body corresponds to the displacement field which minimizes the quadratic functional expressing the total potential energy of the body. It must, however, be emphasized that the finite element method is in no way dependent upon the existence of such a potential or variational principle.

1.5.5 Finite element spatial discretization

We now have all the necessary ingredients to discretize the weak form (1.29) by means of the Galerkin finite element method. Assuming that the reader has some familiarity with this method, we shall limit ourselves to a brief review of the main steps in the process of finite element spatial discretization.

For a partition $\mathcal{T}^h$, see Section 1.5.2, the Galerkin formulation of our model problem is obtained by restricting the weak form (1.29) to the finite dimensional spaces $\mathcal{S}^h \subset \mathcal{S}$ and $\mathcal{V}^h \subset \mathcal{V}$ defined in (1.26),

$$\text{find } u^h \in \mathcal{S}^h \text{ such that } \quad a(w^h, u^h) = (w^h, s) + (w^h, h)_{\Gamma_N} \quad \forall w^h \in \mathcal{V}^h. \quad (1.32)$$

1.5.5.1 Convergence of Galerkin approximations Convergence in finite elements rests upon a priori error bounds (this terminology is used because these bounds are stated before the approximation u^h is actually computed). Moreover, as previously done, the solution of (1.32) is decomposed as $u^h = u^h_0 + \bar{u}^h_D$ with $u^h_0 \in \mathcal{V}^h$ and $\bar{u}^h_D = u_D$ along Γ_D; then, (1.32) can be written as: find $u^h_0 \in \mathcal{V}^h$ such that

$$a(w^h, u^h_0) = (w^h, s) + (w^h, h)_{\Gamma_N} - a(w, \bar{u}^h_D) \quad \forall w^h \in \mathcal{V}^h. \quad (1.33)$$

Theorem 1.2 (Céa's lemma). *Under the assumptions of Theorem 1.1 there exists a unique solution u^h_0 to (1.33) and u_0 to (1.31); moreover, u^h_0 is the near-best fit to u_0 in the $||\cdot||_1$ norm and the error is bounded as*

$$||u_0 - u^h_0||_1 \le \frac{\alpha}{\gamma_1} \min_{v^h \in \mathcal{V}^h} ||u_0 - v^h||_1 .$$

The proof, which can be found elsewhere, see for instance Quarteroni and Valli (1994) or Wait and Mitchell (1985), is characterized by the *Galerkin orthogonality* property. This property is obtained by subtracting (1.32) from (1.29), or subtracting (1.33) from (1.31), where w is particularized as $w^h \in \mathcal{V}^h \subset \mathcal{V}$, namely

$$a(w^h, u - u^h) = 0 \text{ or } a(w^h, u_0 - u^h_0) = 0 \qquad \forall w^h \in \mathcal{V}^h.$$

Remark 1.10 (Symmetric bilinear form). Note that if the bilinear form is also symmetric, and thus $a(\cdot, \cdot)$ induces the energy norm (see Remark 1.8), the Galerkin orthogonality states in fact that the approximation error $u - u^h$ is orthogonal to $\mathcal{V}^h$ under the energy norm. Thus u^h can be viewed as an orthogonal projection of u with respect to the natural (energy) scalar product. Moreover, Céa's lemma can be particularized in terms of the energy norm and the approximation u^h is said to be the best fit in terms of energy

$$\|u_0 - u_0^h\| = \min_{v^h \in \mathcal{V}^h} \|u_0 - v^h\|.$$

Note that this best fit, which is standard in self-adjoint problems, is rarely used in flow problems where convection induces non-symmetric bilinear forms.

These bounds are of interest because, if we intuitively accept that the interpolation of any function in $\mathcal{V}$ is improved as we refine the finite element mesh (consistency), that is a family of partitions $\mathcal{T}^h(\Omega)$ of Ω, see Section 1.5.2, can be defined such that

$$\forall v \in \mathcal{V}, \qquad \min_{v^h \in \mathcal{V}^h} \|v - v^h\|_1 \to 0 \quad \text{as} \quad h \to 0, \tag{1.34}$$

then Theorem 1.2 ensures that convergence is enforced as we decrease h.

In fact, piecewise polynomial interpolation analysis allows us to ensure that

$$\forall v \in \mathcal{V}, \qquad \min_{v^h \in \mathcal{V}^h} \|v - v^h\|_1 \leq C(v)\, h^p$$

where $C(v)$ is a positive constant, which depends on the smoothness of v and the distortion of the finite element mesh, and p is a positive number, which also depends on the smoothness of v and the degree m of the complete basis of polynomials used (see the end of Section 1.5.2).

This bound combined with Céa's lemma produces the standard a priori error bound

$$\|u_0 - u_0^h\|_1 \leq C(u_0) \frac{\alpha}{\gamma_1}\, h^p. \tag{1.35}$$

Note, however, that such a bound is only interesting from a theoretical point of view. Based on interpolation theory and the Lax–Milgram and Céa lemmas, this bound states that a sequence of finite element solutions converges to the exact solution of the weak problem. In engineering practice it is also important to quantify the error associated with an approximation u^h. This is done by means of a posteriori error bounds, see for instance Ainsworth and Oden (2000) or Ladevèze and Pelle (2001).

1.5.5.2 Computational aspects Due to the presence of Dirichlet boundary conditions, a distinction must be made between the number of nodal points, n_{np}, of the discretized domain and the number of nodal unknowns, that is the number of equations n_{eq}, of the system which will result from the Galerkin approximation. Following the terminology introduced by Hughes (2000), we denote by $\eta = \{1, 2, \ldots, \mathrm{n}_{\mathrm{np}}\}$ the set of global node numbers in the finite element mesh. Furthermore, we denote by $\eta_D \subset \eta$ the subset of nodes belonging to the Dirichlet portion of the boundary. It follows, in scalar problems, that the cardinal of $\eta \setminus \eta_D$ is equal to $\mathrm{n_{eq}}$, the number of equations.

With this notation, the approximation u^h can be written as

$$u^h(\boldsymbol{x}) = \sum_{A \in \eta \setminus \eta_D} N_A(\boldsymbol{x})\, \mathrm{u}_A + \sum_{A \in \eta_D} N_A(\boldsymbol{x})\, u_D(\boldsymbol{x}_A) \tag{1.36}$$

where N_A is the shape function (see details below) associated with node number A in the finite element mesh and u_A is the nodal unknown. Moreover, in the Galerkin formulation the arbitrary test functions, w^h, are defined such that

$$w^h \in \mathcal{V}^h := \operatorname*{span}_{A \in \eta \setminus \eta_D} \{N_A\}. \tag{1.37}$$

Thus, using the definitions of (1.36) and (1.37) in (1.32), we obtain the following discrete weak form of our model problem:

$$\sum_{B \in \eta \setminus \eta_D} a(N_A, N_B)\mathrm{u}_B = (N_A, s) + (N_A, h)_{\Gamma_N} - \sum_{B \in \eta_D} a(N_A, N_B) u_D(\boldsymbol{x}_B), \quad \forall A \in \eta \setminus \eta_D. \tag{1.38}$$

Remark 1.11. Upper-case letters, such as A and B, are used to represent global node numbers in the finite element mesh: $1 \leq A, B \leq \mathrm{n_{np}}$. On the other hand, lower-case letters, such as a and b, will be used to represent local node numbers in an element: $1 \leq a, b \leq \mathrm{n_{en}}$ ($\mathrm{n_{en}}$ is the number of element nodes).

Remark 1.12. A suitable mesh generator is used to subdivide the computational domain Ω into element domains Ω^e. In two dimensions, meshes generally consist of triangular and/or quadrilateral elements. An interesting feature of the finite element method is that it handles naturally unstructured meshes, which can concentrate the elements in regions such as internal or boundary layers where sharp solution gradients are expected. The term "unstructured mesh" means that the number of elements meeting at a node (the element vertices) may vary from node to node. A representative unstructured mesh in two dimensions is shown in Figure 1.3. In the present text, we shall assume most of the time that the computational domain Ω is two-dimensional and that its boundary Γ is polygonal.

In the practical implementation of the finite element method, attention is focused on the computations in an individual element. For every element $\Omega^e \in \mathcal{T}^h$, the shape functions N_a, $a = 1, \ldots, \mathrm{n_{en}}$, are defined on a master element in terms of normalized coordinates. They define the finite element interpolation space, which includes a complete basis of the subspace of polynomials of degree m. These subspaces are denoted as $\mathcal{P}_m$, $\mathcal{Q}_m$ or $\mathcal{S}_m$, see Section 1.5.2.

As representative of the meshes used in two dimensions, let us consider, as suggested in Figure 1.4, a subdivision of the computational domain Ω into four-node

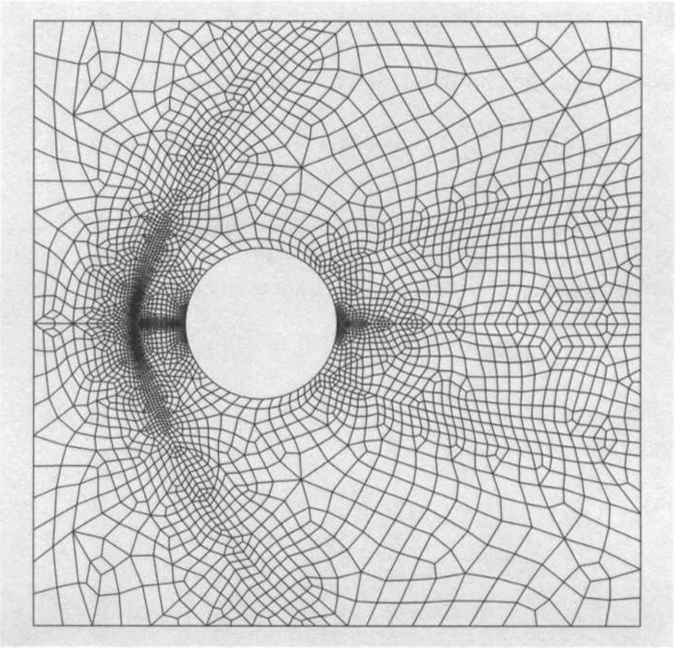


Fig. 1.3 Representative unstructured finite element mesh in two dimensions.

quadrilaterals. Each quadrilateral is mapped onto a canonical square with normalized local coordinates $(\xi, \eta) \in [-1, 1] \times [-1, 1]$ and the element shape functions are tensor products of those used in one dimension. This leads to a so-called isoparametric bilinear approximation on each element where both the global coordinates (x, y) and the unknown u are expressed by the same bilinear expansion parameterized by the nodal values:

$$\begin{Bmatrix} x \\ y \end{Bmatrix} = \sum_{a=1}^{4} N_a(\xi, \eta) \begin{Bmatrix} x_a \\ y_a \end{Bmatrix}, \quad \text{and} \quad u^h(x, y) \equiv u^h(\xi, \eta) = \sum_{a=1}^{4} N_a(\xi, \eta)\, \mathrm{u}_a,$$

where (x_a, y_a) are the coordinates of node a of the element.

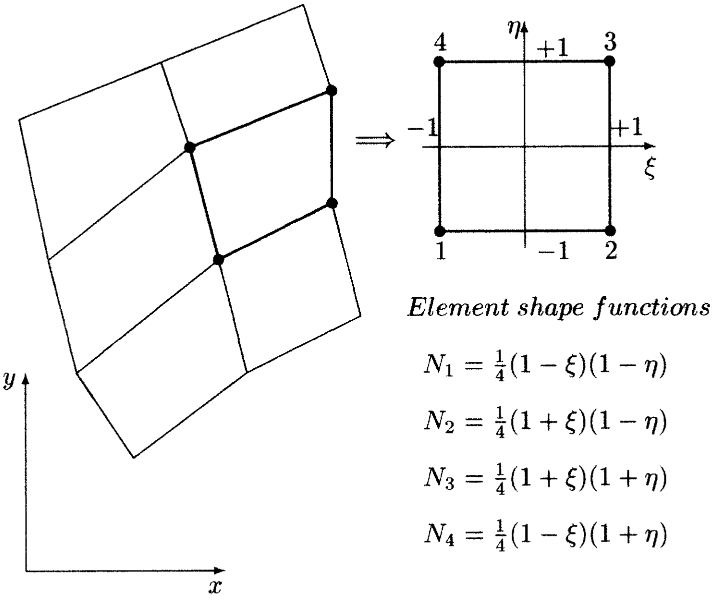
All the element contributions to the discrete weak form (1.38) are computed in the local coordinates (ξ, η) using numerical quadrature, see Hughes (2000) or Zienkiewicz and Taylor (2000a) for details.

The assembly of the element contributions to the discrete weak form into the complete system results in a matrix equation of the form

$$\mathbf{K}\,\mathbf{u} = \mathbf{f}, \tag{1.39}$$

where $\mathbf{u}$ is the vector of the unknown nodal values. Its dimension is n_{eq}, in fact, $\mathbf{u}^T = [\ldots, \mathrm{u}_A, \ldots]$ with $A \in \eta \setminus \eta_D$.

In practice, the global matrix, $\mathbf{K}$, and nodal vector, $\mathbf{f}$, result from the topological assembly of element contributions. The addition of the element contributions to the



Element shape functions

$$N_1 = \tfrac{1}{4}(1-\xi)(1-\eta)$$

$$N_2 = \tfrac{1}{4}(1+\xi)(1-\eta)$$

$$N_3 = \tfrac{1}{4}(1+\xi)(1+\eta)$$

$$N_4 = \tfrac{1}{4}(1-\xi)(1+\eta)$$

Fig. 1.4 Four-node quadrilateral elements and normalized reference element.

appropriate locations in the global matrix, **K**, and nodal vector, **f**, can be represented through the action of an assembly operator $\mathop{\mathbf{A}}^{e}$ acting on the local element matrix and nodal vector as follows:

$$\mathbf{K} = \mathop{\mathbf{A}}^{e} \mathbf{K}^e, \quad K^e_{ab} = \int_{\Omega^e} \boldsymbol{\nabla} N_a \cdot \boldsymbol{\nabla} N_b \, d\Omega \; = \; a(N_a, N_b)_{\Omega^e},$$

$$\mathbf{f} = \mathop{\mathbf{A}}^{e} \mathbf{f}^e, \qquad \mathrm{f}^e_a = (N_a, s)_{\Omega^e} + (N_a, h)_{\partial\Omega^e \cap \Gamma_N} - \sum_{b=1}^{\mathrm{n_{en}}} a(N_a, N_b)_{\Omega^e} u^e_{Db}.$$

Here, $u^e_{Db} = u_D(\boldsymbol{x}^e_b)$ if u_D is prescribed at node number b and equals zero otherwise. Hughes (2000) provides a detailed exposition on the topological assembly of the matrices and nodal vectors arising from the Galerkin finite element discretization.

Remark 1.13 (Alternative implementation of essential boundary conditions). The treatment of Dirichlet boundary conditions as indicated previously is not the only possible one. In fact, equation (1.39) is the matrix form of the weak form (1.33), but the blunt matrix form of (1.32) induces, in general, a singular system

$$\mathbf{K}^\star \mathbf{u}^\star = \mathbf{f}^\star$$

of $\mathrm{n_{np}}$ equations where some of the components (the cardinal of η_D in scalar problems) of $\mathbf{u}^\star = [\ldots, \mathrm{u}_A, \ldots]^T$, $A \in \eta$, are given by the Dirichlet boundary conditions.

Another popular method to implement essential boundary conditions is by means of Lagrange multipliers. If the Lagrange multiplier technique is chosen, the prescribed values on the Dirichlet portion of the boundary are introduced by means of linear constraints and the original system (singular in general) is enlarged by adding n_λ equations (the linear constraints) and n_λ unknowns (n_λ Lagrange multipliers λ, one per constraint, for instance the cardinal of η_D, i.e., $\mathrm{n_{np}} - \mathrm{n_{eq}}$, in scalar problems). The resulting system with the linear constraints is written as

$$\mathbf{r}(\mathbf{u}, \boldsymbol{\lambda}) = \begin{Bmatrix} \mathbf{K}^\star \mathbf{u}^\star + \mathbf{A}^T \boldsymbol{\lambda} - \mathbf{f}^\star \\ \mathbf{A}\mathbf{u}^\star - \mathbf{b} \end{Bmatrix} = \mathbf{0} \tag{1.40}$$

where $\mathbf{A}^T \boldsymbol{\lambda}$ is the vector of reaction forces that enforce the constraints, $\boldsymbol{\lambda}$ is a vector of n_λ Lagrange multipliers, $\mathbf{A}$ is an $n_\lambda \times \mathrm{n_{np}}$ rectangular matrix, and $\mathbf{b}$ a vector listing the prescribed values of the constraints.

Standard linear and nonlinear equation solvers have been adapted to handle linear constraints via the Lagrange-multiplier technique, see for instance Rodríguez-Ferran and Huerta (1999), and this procedure is very popular in object-oriented codes but it is not standard in other commercial programs. Note that apart from increasing the number of constraints the new matrix of the enlarged system,

$$\begin{pmatrix} \mathbf{K}^\star & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{pmatrix},$$

is no longer positive definite as it is usually the case in (1.39). A practical example of how to incorporate the constraints represented by Dirichlet boundary conditions by use of Lagrange multipliers is given in Section 4.7.2.3. Ainsworth (2001) discusses some further possibilities for implementing linear constraints.

2

Steady transport problems

This chapter recalls first the deficiencies of the standard Galerkin formulation in convection-dominated problems. Then, it provides an introduction to generalized methods of the Petrov–Galerkin type designed to produce stable and accurate results in the presence of highly convective effects. This includes the Streamline-Upwind Petrov–Galerkin (SUPG) method, the Galerkin/Least-squares (GLS) method, as well as an introduction to other stabilization methods such as bubble function and wavelet-based methods, and, finally, introduces the variational multiscale method.

2.1 PROBLEM STATEMENT

As an introduction to the study of steady convection–diffusion problems by means of generalized Galerkin methods, we start by recalling the basic steps in the formulation of the standard Galerkin finite element method. This will serve as a basis to point out the deficiencies of the classical Galerkin approach in the solution of convection-dominated transport problems and thus introduce more adequate formulations.

2.1.1 Strong form

Let us consider the transport by convection and diffusion of a scalar quantity $u = u(\boldsymbol{x})$ in a domain $\Omega \subset \mathbb{R}^{\mathrm{n_{sd}}}$, $\mathrm{n_{sd}} = 2$ or 3, with smooth boundary Γ. The boundary is assumed to consist of a portion Γ_D on which the value of u is prescribed and of a complementary portion Γ_N on which the diffusive flux is prescribed. Conditions on Γ_D are Dirichlet (or essential) conditions, while conditions on Γ_N are known

Finite Element Methods for Flow Problems. Jean Donea and Antonio Huerta
Copyright © 2003 John Wiley & Sons, Ltd.
ISBN: 0-471-49666-9

as Neumann (or natural) conditions. The boundary value problem associated with steady convective–diffusive transport is defined by the following equations:

$$\boldsymbol{a}\cdot\boldsymbol{\nabla}u - \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla}u) = s \qquad \text{in } \Omega, \tag{2.1a}$$

$$u = u_D \qquad \text{on } \Gamma_D, \tag{2.1b}$$

$$\boldsymbol{n}\cdot\nu\boldsymbol{\nabla}u = \nu\frac{\partial u}{\partial \boldsymbol{n}} = h \qquad \text{on } \Gamma_N, \tag{2.1c}$$

where u is the scalar unknown, $\boldsymbol{a}(\boldsymbol{x})$ is the convection velocity (also known as advection), $\nu > 0$ is the coefficient of diffusivity and $s(\boldsymbol{x})$ is a volumetric source term. The function u_D denotes prescribed values of u on the Dirichlet portion Γ_D of the boundary, while function h denotes prescribed values of the normal diffusive flux on the Neumann portion Γ_N. The unit outward normal vector to Γ is denoted by $\boldsymbol{n}$.

Remark 2.1 (Pure convection). In the absence of diffusion, $\nu = 0$, problem (2.1) is hyperbolic and boundary conditions can only be prescribed on the inflow portion of the boundary. In this case it is convenient to consider a partition of the boundary Γ such that

$$\begin{aligned} \Gamma &= \overline{\Gamma^{in} \cup \Gamma^{out}} \\ \Gamma^{in} &= \{\boldsymbol{x} \in \Gamma \mid \boldsymbol{a}\cdot\boldsymbol{n} < 0\} && \text{(inflow boundary)} \\ \Gamma^{out} &= \{\boldsymbol{x} \in \Gamma \mid \boldsymbol{a}\cdot\boldsymbol{n} > 0\} && \text{(outflow boundary).} \end{aligned}$$

In the case of Dirichlet inlet conditions, the equations governing purely convective transport are

$$\begin{aligned} \boldsymbol{a}\cdot\boldsymbol{\nabla}u &= s && \text{in } \Omega, \\ u &= u_D && \text{on } \Gamma^{in}. \end{aligned}$$

Note that the solution of linear pure convection problems can be discontinuous in the cross-flow direction (a jump may exist orthogonal to a characteristic curve; Chapter 3 presents in detail the concept of characteristic curves). This occurs when the data u_D, that are prescribed on the inflow boundary, are discontinuous. By contrast, in the case of a convection–diffusion problem (2.1), the solution is continuous over the whole computational domain, any possible jump being spread over a layer of width proportional to the square root of the diffusivity, $\sqrt{\nu}$, around the near characteristic.

Remark 2.2. Instead of prescribing the normal diffusive flux on Γ_N, one may consider prescribing the total normal flux, h_T, consisting of the sum of a diffusive component and a convective component:

$$(\nu\boldsymbol{\nabla}u - u\boldsymbol{a})\cdot\boldsymbol{n} = h_T. \tag{2.2}$$

This is standard in conservation equations. Two options are available to deal with a total flux boundary condition. The first one consists in rewriting (2.2) as a standard diffusive flux boundary condition, but with a value of the flux depending on the solution u itself:

$$\nu \frac{\partial u}{\partial n} = h_T + u(\boldsymbol{a} \cdot \boldsymbol{n}). \tag{2.3}$$

In most applications the previous equation appears as

$$\nu \frac{\partial u}{\partial n} = (\boldsymbol{a} \cdot \boldsymbol{n})(u - u_{\text{ext}}),$$

when the total flux, h_T, is replaced by $-(\boldsymbol{a} \cdot \boldsymbol{n})u_{\text{ext}}$, where u_{ext} is the value of u outside the domain. Note that, in the numerical implementation, such a boundary condition will contribute to the construction of the l.h.s. matrix. This, as will be seen next, does not occur with the standard Neumann boundary condition, equation (2.1c).

The second option consists in expressing the convective term in the form $\boldsymbol{a} \cdot \boldsymbol{\nabla} u = \boldsymbol{\nabla} \cdot (u\,\boldsymbol{a}) - u\,\boldsymbol{\nabla} \cdot \boldsymbol{a}$. This allows us to rewrite the convection–diffusion equation, see (2.1a), in the form

$$\boldsymbol{\nabla} \cdot (u\,\boldsymbol{a} - \nu \boldsymbol{\nabla} u) - u\,\boldsymbol{\nabla} \cdot \boldsymbol{a} = s. \tag{2.4}$$

This option incorporates the divergence of the total flux appearing in the boundary term (2.2), and, as is shown next, will simplify the treatment of the boundary integral in the weak form.

2.1.2 Weak form

The first step towards the finite element spatial discretization of the convection–diffusion problem is to associate an equivalent weak (or variational) form to the strong form (2.1) of the boundary value problem. As seen in Section 1.5.2, the trial solution space $\mathcal{S}$ consists of real-valued functions, u, defined on Ω such that all members of $\mathcal{S}$ satisfy the Dirichlet condition in (2.1b). Similarly, the space $\mathcal{V}$ of the weighting functions, w, is chosen such that $w = 0$ on Γ_D. Namely, $\mathcal{S} := \{u \in \mathcal{H}^1(\Omega) \mid u = u_D \text{ on } \Gamma_D\}$ and $\mathcal{V} := \mathcal{H}^1_{\Gamma_D}(\Omega) = \{w \in \mathcal{H}^1(\Omega) \mid w = 0 \text{ on } \Gamma_D\}$.

The weak formulation of the convection–diffusion problem (2.1) then takes the following form: find $u \in \mathcal{S}$ such that

$$\int_\Omega w(\boldsymbol{a} \cdot \boldsymbol{\nabla} u)\,d\Omega - \int_\Omega w\boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} u)\,d\Omega = \int_\Omega w\,s\,d\Omega \quad \text{for all } w \in \mathcal{V},$$

or, using the divergence theorem on the diffusion term and noting that $w = 0$ on Γ_D,

$$\int_\Omega w(\boldsymbol{a} \cdot \boldsymbol{\nabla} u)\,d\Omega + \int_\Omega \boldsymbol{\nabla} w \cdot (\nu \boldsymbol{\nabla} u)\,d\Omega = \int_\Omega w\,s\,d\Omega + \int_{\Gamma_N} w\,h\,d\Gamma \quad \text{for all } w \in \mathcal{V}. \tag{2.5}$$

This weak form is at the basis of the finite element spatial discretization of the convection–diffusion problem. Note that the integration by parts of the diffusion term allows us to naturally introduce the prescribed flux condition on portion Γ_N of the boundary.

At this point, it is convenient to introduce a compact version of (2.5) based upon the integral forms defined in (1.27), namely

$$\begin{aligned} a(w,u) &= \int_\Omega \boldsymbol{\nabla} w \cdot (\nu \boldsymbol{\nabla} u)\,d\Omega, & (w,s) &= \int_\Omega w\, s\, d\Omega, \\ c(\boldsymbol{a}; w, u) &= \int_\Omega w(\boldsymbol{a}\cdot\boldsymbol{\nabla} u)\,d\Omega, & (w,h)_{\Gamma_N} &= \int_{\Gamma_N} w\, h\, d\Gamma. \end{aligned} \tag{2.6}$$

This allows us to rewrite the weak form (2.5) in the following compact form:

$$a(w,u) + c(\boldsymbol{a}; w, u) = (w,s) + (w,h)_{\Gamma_N}. \tag{2.7}$$

Remark 2.3. When the total flux boundary condition (2.2) is prescribed, see Remark 2.2, an alternative weak form of the convection–diffusion problem may be constructed from the differential equation in conservation form (2.4). After integration by parts the weak form in this case becomes

$$\begin{aligned} \int_\Omega \boldsymbol{\nabla} w \cdot (\nu \boldsymbol{\nabla} u - u\boldsymbol{a})\,d\Omega - \int_\Omega w\, u(\boldsymbol{\nabla}\cdot\boldsymbol{a})\,d\Omega \\ = \int_\Omega w\, s\, d\Omega + \int_{\Gamma_N} w\, h_T\, d\Gamma \quad \text{for all } w \in \mathcal{V}. \end{aligned} \tag{2.8}$$

Note that an extra term (due to the integration by parts of the convection term) appears on the l.h.s. This term cancels if the convection velocity is divergence free, which is often the case in engineering applications.

2.2 GALERKIN APPROXIMATION

We now have all the necessary ingredients to perform the spatial discretization of the convection–diffusion problem (2.1) by means of the Galerkin finite element method. Let $\mathcal{S}^h$ and $\mathcal{V}^h$ be finite dimensional spaces' subsets of $\mathcal{S}$ and $\mathcal{V}$, respectively. These interpolation spaces were defined in Section 1.5.2. The weighting functions $w^h \in \mathcal{V}^h$ vanish on Γ_D. The approximation u^h lies in $\mathcal{S}^h$ and satisfies, with the precision given by the characteristic mesh size h, the boundary condition u_D on Γ_D. The Galerkin formulation is obtained by restricting the weak form (2.7) to the finite dimensional spaces, namely, find $u^h \in \mathcal{S}^h$ such that

$$a(w^h, u^h) + c(\boldsymbol{a}; w^h, u^h) = (w^h, s) + (w^h, h)_{\Gamma_N} \qquad \text{for all } w^h \in \mathcal{V}^h. \tag{2.9}$$

At this point, we have to view Ω as discretized into elements Ω^e, $1 \le e \le \mathrm{n_{el}}$. As shown by equation (1.36), the approximation, u^h, can be written as

$$u^h(\boldsymbol{x}) = \sum_{A\in\eta\setminus\eta_D} N_A(\boldsymbol{x})\,\mathrm{u}_A + \sum_{A\in\eta_D} N_A(\boldsymbol{x})\, u_D(\boldsymbol{x}_A) \tag{2.10}$$

where N_A is the shape function associated with node number A and u_A is the nodal unknown. Moreover, the test functions, w^h, are defined such that

$$w^h \in \mathcal{V}^h := \underset{A\in\eta\setminus\eta_D}{\text{span}} \{N_A\}. \tag{2.11}$$

Thus, after substitution of (2.10) into (2.9), and using (2.11), we obtain the discrete weak form

$$\begin{aligned}\sum_{B\in\eta\setminus\eta_D} \big[a(N_A, N_B) + c(\boldsymbol{a}; N_A, N_B)\big]\, \mathrm{u}_B &= (N_A, s) + (N_A, h)_{\Gamma_N} \\ &- \sum_{B\in\eta_D} \big[a(N_A, N_B) + c(\boldsymbol{a}; N_A, N_B)\big]\, u_D(\boldsymbol{x}_B), \text{ for all } A \in \eta\setminus\eta_D.\end{aligned} \tag{2.12}$$

Assembling the element contributions to this weak form, we obtain the algebraic system governing the nodal values of the discrete solution of the convection–diffusion problem. After inclusion of the Dirichlet boundary conditions as explained in Chapter 1, this system takes the matrix form

$$(\mathbf{C} + \mathbf{K})\mathbf{u} = \mathbf{f}, \tag{2.13}$$

where $\mathbf{u}$ is the vector of the unknown nodal values with dimension n_{eq}, while $\mathbf{C}$ and $\mathbf{K}$ are, respectively, the convection matrix and the diffusion matrix. Both matrices are obtained by topological assembly of the element contributions evaluated, as explained in Section 1.5.5. In terms of the local node numbers $1 \leq a, b \leq \mathrm{n}_{\mathrm{en}}$, the assembly process takes the following form:

$$\begin{aligned} \mathbf{C} &= \mathop{\mathbf{A}}^{e} \mathbf{C}^e \quad C^e_{ab} = \int_{\Omega^e} N_a(\boldsymbol{a}\cdot\boldsymbol{\nabla} N_b)\,d\Omega \quad \text{(convection matrix)} \\ \mathbf{K} &= \mathop{\mathbf{A}}^{e} \mathbf{K}^e \quad K^e_{ab} = \int_{\Omega^e} \boldsymbol{\nabla} N_a\cdot\nu\boldsymbol{\nabla} N_b\,d\Omega \quad \text{(diffusion matrix).}\end{aligned} \tag{2.14}$$

The r.h.s. vector in (2.13) considers the contribution of the source term, s, the prescribed flux, h, and the Dirichlet data u_D. It results from the assembly of elemental contributions of the form $\mathbf{f} = \mathop{\mathbf{A}}^{e} \mathbf{f}^e$ with

$$\begin{aligned} f^e_a = (N_a, s)_{\Omega^e} &+ (N_a, h)_{\partial\Omega^e\cap\Gamma_N} \\ &- \sum_{b=1}^{\mathrm{n}_{\mathrm{en}}} \big[a(N_a, N_b)_{\Omega^e} + c(\boldsymbol{a}; N_a, N_b)_{\Omega^e}\big]\, u^e_{Db}, \end{aligned} \tag{2.15}$$

where $u^e_{Db} = u_D(\boldsymbol{x}^e_b)$ if u_D is prescribed at node number b and equals zero otherwise.

2.2.1 Piecewise linear approximation in 1D

As a model problem and for illustration purposes we consider a 1D scalar equation. Moreover, in order to determine the element matrices the convection and diffusion coefficients, a and ν, are assumed constants. However, the source term s is variable with x. The treatment of the boundary conditions is simplified by imposing homogeneous Dirichlet conditions on each side. Note, however, that other boundary conditions could easily be implemented. Thus the model problem is written as

$$au_x - \nu u_{xx} = s(x) \qquad \text{in }]0, L[\tag{2.16a}$$

$$u = 0 \qquad \text{at } x = 0 \text{ and } x = L. \tag{2.16b}$$

The weak form associated with this model problem is, after integration by parts of the diffusion term, given by

$$\int_0^L (w\, a\, u_x + w_x\, \nu\, u_x)\, dx = \int_0^L w\, s\, dx, \tag{2.17}$$

or in the compact form, see (2.6),

$$a(w, u) + c(a; w, u) = (w, s).$$

The weak form will now be discretized using a uniform mesh of linear elements of size h. Without loss of generality the node numbering is assumed to be consecutive; $\mathrm{n_{eq}}$ is, in this case, the number of interior nodes of the spatial discretization, that is $\eta_D = \{1, \mathrm{n_{np}}\}$ and $\eta = \{1, 2, \ldots, \mathrm{n_{eq}}, \mathrm{n_{eq}} + 1, \mathrm{n_{np}}\}$. The trial and weighting functions are defined as before, see (2.10) and (2.11). Then they are introduced in (2.17) to yield the following discrete equation at an interior node A, $A = 2, \ldots, \mathrm{n}_{\mathrm{eq}} + 1$:

$$\int_0^L \sum_{B=2}^{\mathrm{n_{eq}}+1} \left(aN_A \frac{\partial N_B}{\partial x} + \nu \frac{\partial N_A}{\partial x} \frac{\partial N_B}{\partial x} \right) \mathrm{u}_B\, dx = \int_0^L N_A\, s\, dx. \tag{2.18}$$

A linear element in 1D is defined by two nodes, $\mathrm{n_{en}} = 2$, locally denoted as 1 and 2. The shape functions of a linear element are given by

$$N_1(\xi) = \frac{1}{2}(1 - \xi) \qquad N_2(\xi) = \frac{1}{2}(1 + \xi),$$

where ξ is the normalized coordinate, $-1 \leq \xi \leq +1$. As usual, at any interior point of the element one has $u(\xi) = N_1(\xi)\, \mathrm{u}_1 + N_2(\xi)\, \mathrm{u}_2$, and $x(\xi) = N_1(\xi)\, x_1 + N_2(\xi)\, x_2$. Note that for a uniform mesh of size h

$$dx = \frac{1}{2}(x_2 - x_1) d\xi = \frac{h}{2} d\xi,$$

and thus

$$\frac{\partial N_b}{\partial x} = \frac{\partial N_b}{\partial \xi} \frac{\partial \xi}{\partial x} = \frac{2}{h} \frac{\partial N_b}{\partial \xi} \qquad \text{for } b = 1, 2.$$

With these preliminaries, the integrals in expression (2.18) are readily evaluated in closed form to obtain the element convection matrix, $\mathbf{C}^e$, and the element diffusion matrix, $\mathbf{K}^e$. In terms of the local node numbers 1 and 2 the result is

$$\mathbf{C}^e = a \int_{\Omega^e} \begin{pmatrix} N_1 \dfrac{\partial N_1}{\partial x} & N_1 \dfrac{\partial N_2}{\partial x} \\ N_2 \dfrac{\partial N_1}{\partial x} & N_2 \dfrac{\partial N_2}{\partial x} \end{pmatrix} dx = \frac{a}{2} \begin{pmatrix} -1 & +1 \\ -1 & +1 \end{pmatrix} \tag{2.19a}$$

$$\mathbf{K}^e = \nu \int_{\Omega^e} \begin{pmatrix} \dfrac{\partial N_1}{\partial x} \dfrac{\partial N_1}{\partial x} & \dfrac{\partial N_1}{\partial x} \dfrac{\partial N_2}{\partial x} \\ \dfrac{\partial N_2}{\partial x} \dfrac{\partial N_1}{\partial x} & \dfrac{\partial N_2}{\partial x} \dfrac{\partial N_2}{\partial x} \end{pmatrix} dx = \frac{\nu}{h} \begin{pmatrix} +1 & -1 \\ -1 & +1 \end{pmatrix}. \tag{2.19b}$$

Here $\Omega^e = [x_e, x_{e+1}]$ with $e = 1, \ldots, \mathrm{n_{el}}$ ($\mathrm{n_{el}} = \mathrm{n_{eq}} + 1 = \mathrm{n_{np}} - 1$ in this case). Furthermore, interpolating the source term s by means of the element shape functions, namely $s(\xi) = N_1(\xi)\, s_1 + N_2(\xi)\, s_2$, one finds that the components of the load vector $\mathbf{f}$ in (2.13) are obtained from element contributions of the form

$$\mathbf{f}^e = \int_{\Omega^e} \{N_1 \,(N_1 s_1 + N_2 s_2),\ N_2 \,(N_1 s_1 + N_2 s_2)\}^T \, dx.$$

With these results, and assembling in the usual finite element manner the contributions emanating from both elements to which a given node belongs, one finds that the Galerkin method delivers the following discrete equation at an interior node j:

$$a \left(\frac{u_{j+1} - u_{j-1}}{2h} \right) - \nu \left(\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} \right) = \frac{1}{6}(s_{j-1} + 4s_j + s_{j+1}). \tag{2.20}$$

Remark 2.4. Notice that the l.h.s. of the discrete equation produced with linear elements coincides with that of second-order central differences. In this respect, the Galerkin method based on linear elements and the central difference method appear to be closely related. Nevertheless, a significant difference between finite element and finite difference approximations is observed in the treatment of the source term. On one hand, the Galerkin method generates a weighted average on the r.h.s. of (2.20). That is, the finite element method uses the so-called consistent mass matrix to weigh the nodal values of the source term s_j. This consistent mass matrix is defined by

$$\mathbf{M} = \overset{e}{\mathbf{A}} \mathbf{M}^e$$

where the element mass matrix can be written in terms of the element shape functions as

$$M^e_{ab} = \int_{\Omega^e} \begin{pmatrix} N_1 N_1 & N_1 N_2 \\ N_2 N_1 & N_2 N_2 \end{pmatrix} dx = \frac{h}{6} \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}.$$

On the other hand, the difference method simply uses a local value of the source, namely s_j at node j. That is, the difference method employs a so-called lumped (or diagonal) mass matrix, $\mathbf{M}^L$, defined at the element level by

$$[M^L]^e_{ab} = \int_{\Omega^e} \begin{pmatrix} N_1 & 0 \\ 0 & N_2 \end{pmatrix} dx = \frac{h}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

2.2.2 Analysis of the discrete equation

In this section we show that the Galerkin finite element method as described in the previous section is not ideally suited to solve convection-dominated problems. To characterize the relative importance of convective and diffusive effects in a given flow problem, it is useful to introduce the mesh Péclet number

$$\mathit{Pe} = \frac{ah}{2\nu} \tag{2.21}$$

which expresses the ratio of convective to diffusive transport. This allows us to rewrite the discrete equation (2.20) in the form

$$\frac{a}{2h}\left(\frac{\mathit{Pe}-1}{\mathit{Pe}}u_{j+1} + \frac{2}{\mathit{Pe}}u_j - \frac{\mathit{Pe}+1}{\mathit{Pe}}u_{j-1}\right) = \frac{1}{6}(s_{j-1} + 4s_j + s_{j+1}). \tag{2.22}$$

The importance of the Péclet number will become more and more clear as we move along. However, the previous equation is already a good example. Note that the l.h.s. of (2.22), which corresponds (for linear elements and a Galerkin formulation) to the discrete counterpart of the l.h.s. of the weak form (2.9) or the differential operator in (2.1a), is characterized by the Péclet number.

To illustrate the deficiencies of the Galerkin finite element method in the solution of convection-dominated problems, we shall first present a simple numerical example based on linear elements and then use this example to identify a possible remedy for improving the finite element solution.

This problem simply consists of solving the 1D boundary value problem (2.16) with a constant source term, $s = 1$, in a dimensionless domain, $L = 1$. A uniform source has been chosen on purpose in order to avoid truncation errors due to the spatial discretization of the source term. In this manner, the truncation error, which will arise from the Galerkin discretization of our model problem, must be attributed fully to the discrete representation of the convection and diffusion operators.

The exact solution to the above model problem is given by

$$u(x) = \frac{1}{a}\left(x - \frac{1-\exp(\gamma x)}{1-\exp\gamma}\right), \tag{2.23}$$

where $\gamma = a/\nu$. The numerical approximation to (2.23) has been computed with a mesh of 10 uniform elements using the Galerkin scheme (2.20) or (2.22) for several values of the mesh Péclet number, see (2.21), namely, $\mathit{Pe} = 0.25$, 0.9 and 5 (the convection velocity a is taken as one). The results are displayed in Figure 2.1 in comparison with the exact solution. One notes that the Galerkin solution is corrupted by non-physical oscillations when the Péclet number is larger than one. The Galerkin method loses its best approximation property when the non-symmetric convection operator dominates the diffusion operator in the transport equation, and consequently spurious node-to-node oscillations appear.

The Galerkin equation (2.20) has a truncation error. This is standard in discrete equations. We shall now analyze it and, as a result, discover why the Galerkin method

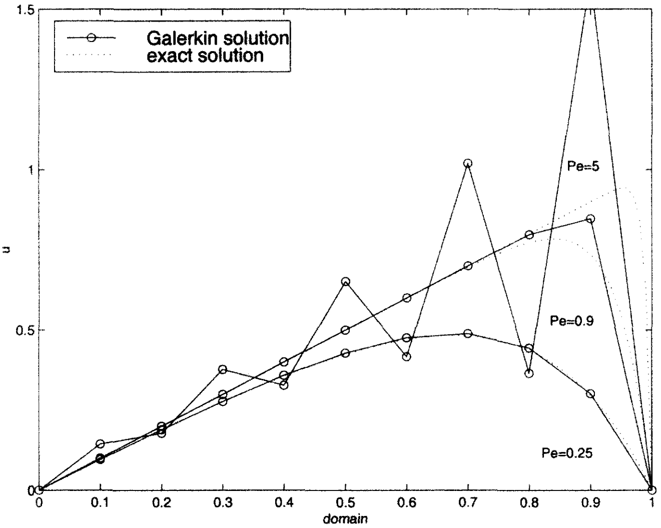


Fig. 2.1 Galerkin solution (solid lines) of the convection–diffusion problem (2.16) with $L = 1, s = 1$ using a uniform mesh of 10 linear elements. Dotted lines show the exact solution.

(and the intimately related central difference method) are not optimal methods for solving convection-dominated problems.

To reach our objective, we need to find a discrete scheme similar in structure to (2.22), but giving the exact solution at each node of a uniform mesh of linear elements for any mesh size h and all values of the Péclet number. The comparison between this exact scheme and the Galerkin scheme will then allow us to identify which modifications are necessary, in the Galerkin method, in order to improve its response in highly convective situations.

To obtain an exact scheme, we must identify the value of three coefficients, say α_1, α_2 and α_3, such that

$$\alpha_1\, u_{j-1} + \alpha_2\, u_j + \alpha_3\, u_{j+1} = 1 \tag{2.24}$$

for all nodal coordinates x_j, mesh dimensions h and Péclet numbers Pe. From the exact solution (2.23) we have

$$\begin{cases} u_{j-1} = \dfrac{1}{a}\left(x_j - h - \dfrac{1-\exp(\gamma x_j)\exp(-2Pe)}{1-\exp\gamma}\right), \\ u_j = \dfrac{1}{a}\left(x_j - \dfrac{1-\exp(\gamma x_j)}{1-\exp\gamma}\right), \\ u_{j+1} = \dfrac{1}{a}\left(x_j + h - \dfrac{1-\exp(\gamma x_j)\exp(2Pe)}{1-\exp\gamma}\right). \end{cases}$$

When these expressions are introduced in (2.24), the following three conditions are obtained:

$$\begin{cases} \alpha_1 + \alpha_2 + \alpha_3 = 0 \\ -\alpha_1 + \alpha_3 = a/h \\ \alpha_1 \exp(-2Pe) + \alpha_2 + \alpha_3 \exp(2Pe) = 0. \end{cases}$$

Solving for α_1, α_2 and α_3, we obtain

$$\begin{cases} \alpha_1 = -a(1 + \coth Pe)/(2h) \\ \alpha_2 = a(\coth Pe)/h \\ \alpha_3 = a(1 - \coth Pe)/(2h). \end{cases}$$

When these expressions are substituted in (2.24), the desired exact scheme becomes

$$\frac{a}{2h}\left[(1 - \coth Pe)u_{j+1} + (2\coth Pe)u_j - (1 + \coth Pe)u_{j-1}\right] = 1. \qquad (2.25)$$

As expected, this equation presents similarities but it is not identical to the one obtained with the Galerkin formulation (2.22). To highlight the similarities and the differences between both schemes, equation (2.25) may be rearranged in two interesting ways.

First, in a form similar to the original Galerkin scheme (2.20)

$$a\frac{u_{j+1} - u_{j-1}}{2h} - (\nu + \bar{\nu})\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} = 1, \qquad (2.26)$$

where $\bar{\nu}$ is an added numerical diffusion defined as

$$\bar{\nu} = \beta\frac{ah}{2} = \beta\nu Pe \qquad \text{with } \beta = \coth Pe - \frac{1}{Pe}. \qquad (2.27)$$

Note that this added numerical diffusion only depends on the parameters of the governing differential equation and the element size h.

Second, the same scheme (2.25) can also be rewritten as

$$\frac{1-\beta}{2}\left(a\frac{u_{j+1} - u_j}{h}\right) + \frac{1+\beta}{2}\left(a\frac{u_j - u_{j-1}}{h}\right) - \nu\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} = 1, \quad (2.28)$$

where the discretization of the convective term appears as a weighted average of the fluxes (convection) of the solution to the left and to the right of node j. That is, the convection term is not discretized using a centered scheme.

This analysis is at the origin of two families of early techniques developed for improving the standard Galerkin method: those based on adding an artificial diffusion and those concerned with non-centered discretizations of the convection operator, also called upwind schemes. These early methods are presented in Section 2.3 and the current stabilization techniques are discussed in Section 2.4.

Remark 2.5. To reduce the computational costs (repeated evaluations of hyperbolic cotangents), it is common practice to replace the optimal formula (2.27) by its doubly asymptotic approximation, see Figure 2.2, given by

$$\beta \approx \beta_{\text{approx}} = \begin{cases} Pe/3 & \text{if } -3 \leq Pe \leq 3, \\ \text{sign}(Pe) & \text{if } |Pe| > 3. \end{cases} \qquad (2.29)$$

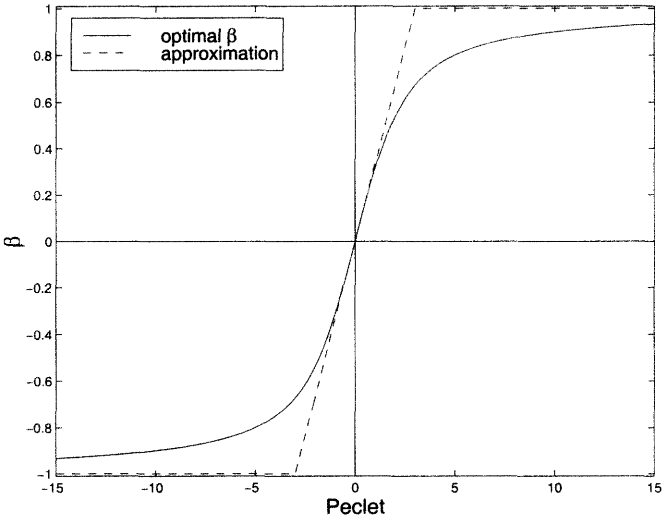


Fig. 2.2 Optimal value of the parameter β, see equation (2.27), and its doubly asymptotic approximation, see equation (2.29).

Remark 2.6 (The negative diffusion of the Galerkin method). A simple comparison between the Galerkin discrete scheme (2.20) and the exact difference scheme (2.26) indicates that the Galerkin method introduces a truncation error in the form of a diffusion operator. That is, the Galerkin scheme lacks $-\bar{\nu}(u_{j+1} - 2u_j + u_{j-1})/h^2$ to represent the exact solution. The diffusion coefficient $\bar{\nu}$, see (2.27), gives a measure of the magnitude of the truncation error as a function of the Péclet number. This truncation error is systematically negative for all values of Pe. Due to this negative truncation error, a *modified equation* is actually solved by the Galerkin method, which possesses a reduced diffusion coefficient. Such a modified equation is the differential equation that is solved exactly (exact nodal values) by the Galerkin method, namely

$$a\,u_x - \left[\nu - \bar{\nu}\frac{\sinh^2(Pe)}{Pe^2}\right] u_{xx} = 1. \tag{2.30}$$

Notice that as the Péclet number increases the diffusion coefficient in the modified equation (2.30) may become negative. In this case, no stable solution is guaranteed. Figure 2.3 shows the actual diffusion coefficient of the partial differential equation associated with the Galerkin scheme, namely the modified equation (2.30). Note that it becomes negative for values of the grid Péclet number larger than one.

The negative numerical diffusion inherent in the Galerkin finite element method (and the second-order central difference method) is indeed the cause of the numerical difficulties encountered in the simulation of highly convective transport problems.

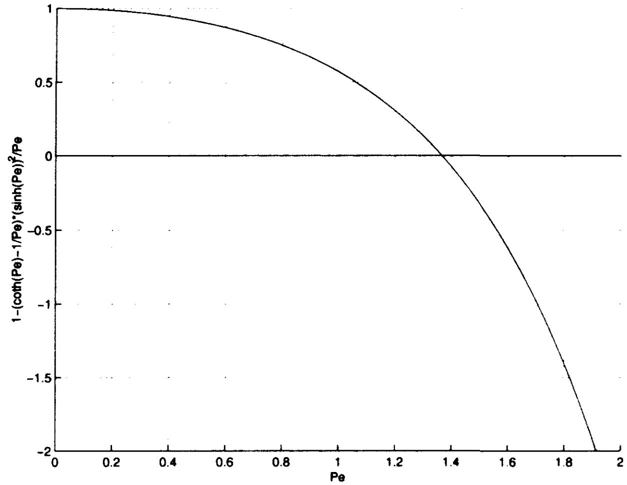


Fig. 2.3 Normalized diffusivity of the modified equation associated with a Galerkin formulation, see equation (2.30), namely $1 - (\bar{\nu}/\nu)(\sinh^2(Pe)/Pe^2)$, as a function of $Pe = ah/(2\nu)$.

Remark 2.7 (Linear difference equation). To further highlight the appearance of non-physical oscillations when the grid Péclet number exceeds unity, we note that the homogeneous form of the discrete equation (2.22) can be rewritten as

$$(1 - Pe)(u_{j+1} - u_j) = (1 + Pe)(u_j - u_{j-1}),$$

which shows that the slopes of the solution to the left and to the right of node j are of opposite sign when the Péclet number is larger than one.

Moreover, this linear difference equation can be solved exactly, see the details in Isaacson and Keller (1994, Chap. 8, Sec. 4). The characteristic equation is

$$(1 - Pe)\lambda^2 - 2\lambda + (1 + Pe) = 0.$$

Its roots are $\lambda_1 = 1$ and $\lambda_2 = (1 + Pe)/(1 - Pe)$, and thus the solution of the homogeneous form of equation (2.22) is given by

$$u_j = C_1 + C_2 \left(\frac{1 + Pe}{1 - Pe} \right)^j ,$$

where C_1 and C_2 are constants fixed by the boundary conditions. This expression clearly shows that the approximate solution delivered by the Galerkin method is oscillatory when $Pe > 1$.

By contrast, the homogeneous form of the discrete equation corresponding to the exact scheme (2.26) or (2.28) reads

$$[1 + (\beta - 1)Pe]\,(u_{j+1} - u_j) = [1 + (\beta + 1)Pe]\,(u_j - u_{j-1}), \tag{2.31}$$

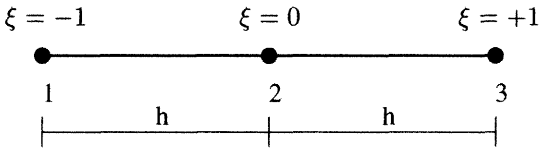


Fig. 2.4 Quadratic element in 1D.

where β is given in (2.27). The analytical solution of this linear difference equation can also be determined, and is given by

$$u_j = C_1 + C_2 \left(\frac{1+(1+\beta)Pe}{1-(1-\beta)Pe}\right)^j .$$

To avoid spurious oscillations of the Galerkin method, the absolute value of β should be larger than the critical value

$$|\beta| \geq \beta_{\text{crit}} = 1 - \frac{1}{|Pe|}. \tag{2.32}$$

Note that this condition is verified by both the optimal value of beta, (2.27), and its doubly asymptotic approximation defined in (2.29).

2.2.3 Piecewise quadratic approximation in 1D

To further illustrate the deficiencies of the Galerkin finite element formulation in convection-dominated problems, let us now examine the numerical solution of a 1D convection–diffusion problem discretized with quadratic shape function elements. First, we shall establish the form of the convection and diffusion matrices. Then the discrete equations for the model problem (2.16) are analyzed.

One of the objectives of this section is to show that quadratic elements and, in general, high-order finite elements present serious difficulties in convection-dominated problems. This is mainly due to the fact that the behavior of interior and corner nodes is different. This will be illustrated in detail in the next section.

As shown in Figure 2.4, we consider a generic element with end nodes 1 and 3, and a mid-side node 2. With reference to the normalized coordinate $-1 \leq \xi \leq +1$, the shape functions of the element are

$$N_1(\xi) = \frac{1}{2}\,\xi(\xi-1), \quad N_2(\xi) = 1-\xi^2, \quad N_3(\xi) = \frac{1}{2}\,\xi(\xi+1). \tag{2.33}$$

It follows that at any interior point of an element, one has $u(\xi) = N_1(\xi)\,\mathrm{u}_1 + N_2(\xi)\,\mathrm{u}_2 + N_3(\xi)\,\mathrm{u}_3$ and $x(\xi) = N_1(\xi)\,x_1 + N_2(\xi)\,x_2 + N_3(\xi)\,x_3$. If a uniform mesh is used, the middle node is located at $x_2 = (x_1 + x_3)/2$. Then, if the characteristic size is h (h is the distance between nodes *not the element size*), the following relations hold between the normalized and physical coordinates: $dx = h\,d\xi$ and

$$\frac{\partial N_b}{\partial x} = \frac{\partial N_b}{\partial \xi}\frac{\partial \xi}{\partial x} = \frac{1}{h}\frac{\partial N_b}{\partial \xi}, \quad b = 1,2,3.$$

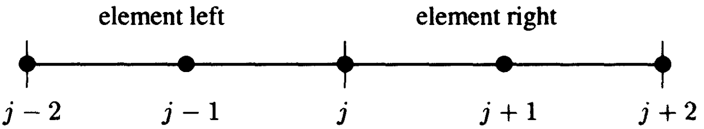


Fig. 2.5 Node numbering for a uniform mesh of quadratic elements.

With these interpolation functions the integrals in expression (2.18) are readily evaluated in closed form to obtain the convection, $\mathbf{C}^e$, and diffusion, $\mathbf{K}^e$, matrices of the quadratic element:

$$\mathbf{C}^e = a \int_{\Omega^e} \begin{pmatrix} N_1 \frac{\partial N_1}{\partial x} & N_1 \frac{\partial N_2}{\partial x} & N_1 \frac{\partial N_3}{\partial x} \\ N_2 \frac{\partial N_1}{\partial x} & N_2 \frac{\partial N_2}{\partial x} & N_2 \frac{\partial N_3}{\partial x} \\ N_3 \frac{\partial N_1}{\partial x} & N_3 \frac{\partial N_2}{\partial x} & N_3 \frac{\partial N_3}{\partial x} \end{pmatrix} dx$$

$$= \frac{a}{2} \begin{pmatrix} -1 & 4/3 & -1/3 \\ -4/3 & 0 & 4/3 \\ 1/3 & -4/3 & 1 \end{pmatrix} \tag{2.34a}$$

$$\mathbf{K}^e = \nu \int_{\Omega^e} \begin{pmatrix} \frac{\partial N_1}{\partial x}\frac{\partial N_1}{\partial x} & \frac{\partial N_1}{\partial x}\frac{\partial N_2}{\partial x} & \frac{\partial N_1}{\partial x}\frac{\partial N_3}{\partial x} \\ \frac{\partial N_2}{\partial x}\frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial x}\frac{\partial N_2}{\partial x} & \frac{\partial N_2}{\partial x}\frac{\partial N_3}{\partial x} \\ \frac{\partial N_3}{\partial x}\frac{\partial N_1}{\partial x} & \frac{\partial N_3}{\partial x}\frac{\partial N_2}{\partial x} & \frac{\partial N_3}{\partial x}\frac{\partial N_3}{\partial x} \end{pmatrix} dx$$

$$= \frac{\nu}{6h} \begin{pmatrix} 7 & -8 & 1 \\ -8 & 16 & -8 \\ 1 & -8 & 7 \end{pmatrix}. \tag{2.34b}$$

Here $\Omega^e = [x_{2e-1}, x_{2e+1}]$ with $e = 1, \ldots, \mathrm{n_{el}}$ ($\mathrm{n_{el}} = (\mathrm{n_{eq}} + 1)/2 = (\mathrm{n_{np}} - 1)/2$ in this case).

As with the linear element, we interpolate the source term $s(x)$ in (2.16) by means of the element shape functions,

$$s(\xi) = N_1(\xi)s_1 + N_2(\xi)s_2 + N_3(\xi)s_3, \tag{2.35}$$

and obtain the components of the load vector $\mathbf{f}$ in (2.13) from element contributions of the form

$$\mathrm{f}_b^e = \int_{\Omega^e} N_b \, (N_1 \, s_1 + N_2 \, s_2 + N_3 \, s_3) \, dx, \qquad b = 1, 2, 3.$$

In order to obtain in finite difference format the discrete equation at each node, the numbering sequence shown in Figure 2.5 is employed. With the element matrices presented in (2.34a) and (2.34b), and assembling in the usual finite element manner, one finds that the Galerkin method delivers two types of nodal equations representing the discrete counterpart of the convection–diffusion equation (2.16a):

1. *At a mid-side node i* (in Figure 2.5, $i = j+1$ or $i = j-1$) the discrete equation is obtained in the form

$$a\left(\frac{u_{i+1}-u_{i-1}}{2h}\right) - \nu\left(\frac{u_{i-1}-2u_i+u_{i+1}}{h^2}\right) = \frac{1}{10}\left(s_{i-1}+8s_i+s_{i+1}\right) \quad (2.36)$$

   its l.h.s. is identical to equation (2.20) obtained using piecewise linear approximations.

2. *At the corner (inter-element) node j* (see Figure 2.5) the discrete equation involves a stencil of five nodes and reads

$$\begin{aligned} a&\left[2\left(\frac{u_{j+1}-u_{j-1}}{2h}\right) - \left(\frac{u_{j+2}-u_{j-2}}{4h}\right)\right] \\ &- \nu\left[2\left(\frac{u_{j-1}-2u_j+u_{j+1}}{h^2}\right) - \left(\frac{u_{j-2}-2u_j+u_{j+2}}{4h^2}\right)\right] \\ &= \frac{1}{10}\left(-s_{j-2}+2s_{j-1}+8s_j+2s_{j+1}-s_{j+2}\right). \end{aligned} \quad (2.37)$$

As already done for the linear element, a first test for quadratic elements is performed by solving the linear convection–diffusion problem (2.16) with $L = 1$ and $s = 1$. The results obtained with the Galerkin formulation on a uniform mesh of five quadratic elements are displayed in Figure 2.6 and compared with the exact solution given, as previously, by (2.23). Again the Galerkin solution is characterized by spurious node-to-node oscillations when convective effects become important.

Remark 2.8. Similar conclusions can be drawn with another well-known academic problem, see Section 2.6.2. The same linear convection–diffusion equation (2.16a) is solved over a unit domain, $L = 1$, with no source term, $s = 0$, and the Dirichlet boundary conditions $u(0) = 0$ and $u(1) = 1$. The exact solution is, in this case,

$$u(x) = \frac{1-\exp(\gamma x)}{1-\exp(\gamma)},$$

with $\gamma = a/\nu$.

2.2.4 Analysis of the discrete equations

We shall now analyze the discrete equations (2.36) and (2.37) delivered by the Galerkin method on a uniform mesh of quadratic elements. As previously done for the piecewise linear approximations, see Section 2.2.2, an exact scheme is identified for both the mid-side and corner nodes.

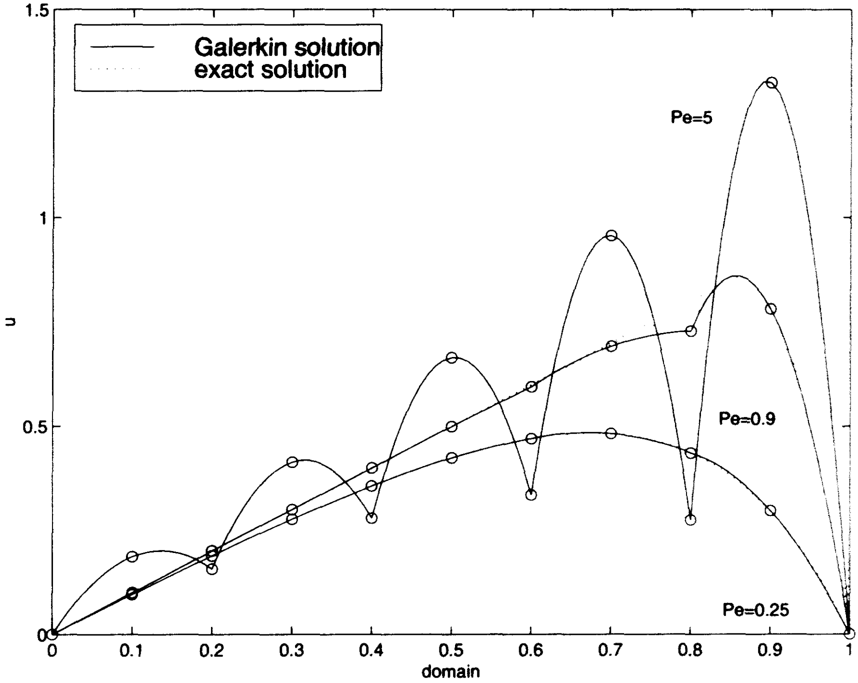


Fig. 2.6 Galerkin solution (solid lines) of the convection–diffusion problem (2.16) using a uniform mesh of five quadratic elements.

1. *At a mid-side node i*, see equation (2.36), a three-point exact difference formula is sought. Following the same steps as in Section 2.2.2, the same scheme will be obtained, that is equation (2.26) or (2.28) with the corresponding parameters given by (2.27). Thus as previously discussed, two interpretations (upwinding or artificial diffusion) are again possible. Moreover, it follows that the mid-side node equation delivered by the Galerkin method corresponds to a modified equation with a reduced diffusion coefficient.

2. *At the corner (inter-element) node j* (see Figure 2.5) a new exact discrete equation is needed. Each difference equation now involves five nodes, see (2.37). The same procedure as in Section 2.2.2 is used. Notice, however, that now five coefficients must be determined, which need to satisfy the condition

$$\alpha_1\, u_{j-2} + \alpha_2\, u_{j-1} + \alpha_3\, u_j + \alpha_4\, u_{j+1} + \alpha_5\, u_{j+2} = 1, \qquad (2.38)$$

but only three equations are available. This is reasonable because an exact scheme can be obtained with three nodes. If more nodes are employed, families of schemes are possible. For instance, the following two-parameter (c_1 and c_2)

family will induce exact nodal schemes:

$$\begin{cases} \alpha_1 = \dfrac{a}{4h}(1+\coth \mathit{Pe})\, c_1, \\ \alpha_2 = \dfrac{a}{4h}\left[-2(1+\coth \mathit{Pe}) - (1+3\coth \mathit{Pe})\, c_1 - (1+\coth \mathit{Pe})\, c_2\right], \\ \alpha_3 = \dfrac{a}{4h}\left[4\coth \mathit{Pe} - (1-3\coth \mathit{Pe})\, c_1 + (1+3\coth \mathit{Pe})\, c_2\right], \\ \alpha_4 = \dfrac{a}{4h}\left[2(1-\coth \mathit{Pe}) + (1-\coth \mathit{Pe})\, c_1 + 2(1-3\coth \mathit{Pe})\, c_2\right], \\ \alpha_5 = \dfrac{-a}{4h}(1-\coth \mathit{Pe})\, c_2. \end{cases} \tag{2.39}$$

However, as previously done, we prefer to rewrite equation (2.38) as a discrete equation delivered by the Galerkin formulation with an added numerical diffusion. To this end, we rewrite equation (2.37) in the form

$$\begin{aligned} & a\left[2\left(\frac{u_{j+1}-u_{j-1}}{2h}\right) - \left(\frac{u_{j+2}-u_{j-2}}{4h}\right)\right] \\ & \quad - (\nu + \bar{\nu}_{\text{corner}})\left[2\left(\frac{u_{j-1}-2u_j+u_{j+1}}{h^2}\right) - \left(\frac{u_{j-2}-2u_j+u_{j+2}}{4h^2}\right)\right] \\ & \qquad = 1, \end{aligned} \tag{2.40}$$

where a new artificial diffusion coefficient $\bar{\nu}_{\text{corner}}$ must be defined. In fact, after substitution of (2.39) into (2.38) and imposing its equivalence to (2.40), one obtains

$$\begin{cases} c_1 = 1 - \dfrac{1}{2}\coth \mathit{Pe}\, \dfrac{1-\coth \mathit{Pe}}{2-\coth \mathit{Pe}} \\ c_2 = 1 + \dfrac{1}{2}\coth \mathit{Pe}\, \dfrac{1+\coth \mathit{Pe}}{2-\coth \mathit{Pe}} \end{cases}$$

and

$$\bar{\nu}_{\text{corner}} = \frac{ah}{2}\, \frac{(\coth \mathit{Pe} - 1/\mathit{Pe}) - (\cosh \mathit{Pe})^2(\coth 2\mathit{Pe} - 1/(2\mathit{Pe}))}{1-(\cosh \mathit{Pe})^2/2}. \tag{2.41}$$

Comparing the discrete equation (2.37) delivered by the Galerkin method and the exact scheme (2.40), one notes that the truncation error introduced at corner nodes by the Galerkin method is again in the form of a spurious diffusion.

This analysis confirms that any standard discrete formulation will induce different responses for interior and corner nodes. Moreover, an exact nodal formulation must treat, as we have seen, differently interior and corner nodes.

We shall now look at ways to remedy the lack of stability of the Galerkin finite element method and thereby obtain stable and accurate approximations to convection-dominated problems.

2.3 EARLY PETROV–GALERKIN METHODS

The Galerkin formulation presents serious deficiencies in convection-dominated problems. These deficiencies can be interpreted and cured in two ways, see Section 2.2.2. First, diffusion was added in order to counterbalance the negative numerical diffusion introduced by the Galerkin approximation based on linear elements. Second, an upwind approximation of the convective term is used because the centered scheme employed is not ideal in convection-dominated problems. Precisely, the early remedies were based on these two philosophies. In fact, as will be seen later, both methodologies are equivalent. That is, an upwind approximation induces numerical diffusion and vice versa.

2.3.1 Upwind approximation of the convective term

In the framework of the *finite difference method*, numerical diffusion can be introduced by replacing the second-order accurate central approximation to the convective term by a first-order upwind approximation defined, for $a > 0$, by

$$u_x(x_j) \approx \frac{u_j - u_{j-1}}{h}. \tag{2.42}$$

If upwind differencing is used on the convective term instead of central differencing, the discrete convection–diffusion equation at node j becomes, in the absence of source term,

$$a\,\frac{u_j - u_{j-1}}{h} - \nu\,\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} = 0. \tag{2.43}$$

The upwind derivative of the convective term introduces a numerical dissipation which comes in addition to the physical diffusion ν, as can be seen from the Taylor series development of the convective term around x_j:

$$a\frac{u_j - u_{j-1}}{h} = a\,u_x(x_j) - \frac{ah}{2}\,u_{xx}(x_j) + \mathcal{O}(h^2).$$

An added diffusion of magnitude $ah/2$ has thus been introduced by the upwind approximation of the convective term. As a matter of fact, the discrete equation (2.43) also results from the central difference approximation of the equation

$$a\,u_x - \left(\nu + \frac{ah}{2}\right) u_{xx} = 0,$$

which includes an added numerical diffusion of magnitude $ah/2$. It has, however, been noted that in most cases the upwind treatment of the convective term as given in (2.43) leads to excessively dissipative results and this has given rise to many criticisms against the upwind technique, see for instance Davis and Mallinson (1976), Gresho and Lee (1979) and Leonard (1979). That a full upwind treatment of the convective term leads to stable, but overly diffusive, results can be appreciated from Figure 2.7 which shows the upwind solution of the model problem (2.16) in comparison with the

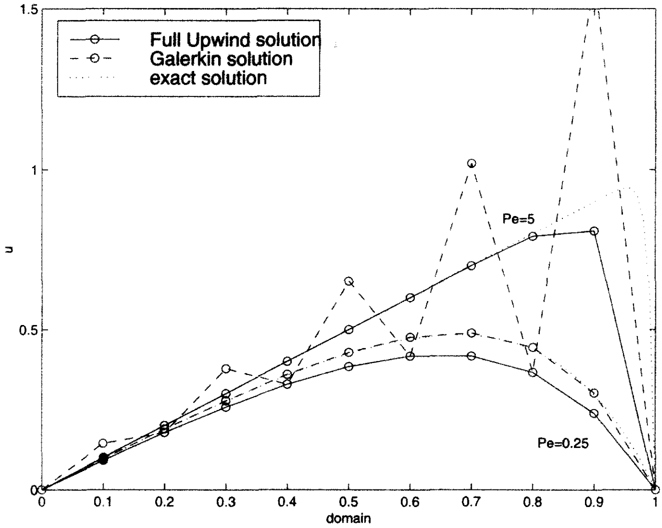


Fig. 2.7 Full upwind solution (solid lines) of the model convection–diffusion problem (2.16) with $L = 1$ and $s = 1$. Dashed lines show the Galerkin approximation and dotted lines the exact solution (2.23).

Galerkin approximation and the exact solution. Note that for large Péclet numbers the solution is stable and close to the exact one. However, for low values of the Péclet number, when the Galerkin approximation is accurate, the full upwind solution is clearly over diffusive.

2.3.2 First finite elements of upwind type

In a *finite element* framework, several different techniques can be utilized to achieve the upwind effect. The key idea in practically all the proposed finite element formulations of upwind type has been to replace the standard Galerkin formulation with a so-called *Petrov–Galerkin* weighted residual formulation in which the weighting function may be selected from a different class of functions than the approximate solution. The basic idea behind such an approach has been that the optimal approximation property, the basis of the success of the standard Galerkin finite element method in self-adjoint problems, could be carried over to convection–diffusion problems by means of the Petrov–Galerkin formulation.

The first upwind finite element formulations were presented in the 1970s by the Dundee and Swansea research groups (see, e.g., Christie et al., 1976; Heinrich et al., 1977; Heinrich and Zienkiewicz, 1979; Griffiths and Mitchell, 1979). These were based on *modified weighting functions* such that the element upstream of a node is weighted more heavily than the element downstream of a node. An example of such a weighting function is shown in Figure 2.8. In this way, the finite element equivalent

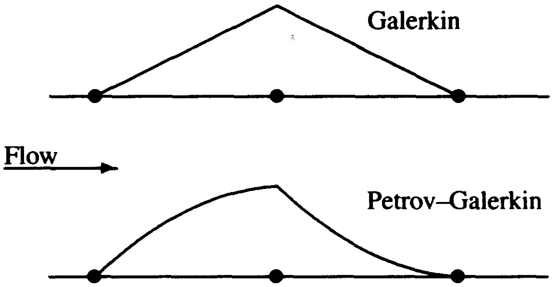


Fig. 2.8 Upwind-type weighting function.

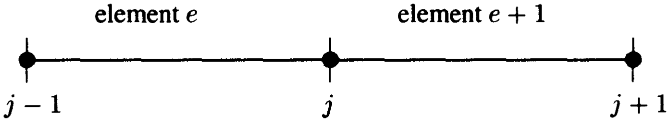


Fig. 2.9 Node numbering for two consecutive linear elements.

of upwind differencing was invented in the form of a Petrov–Galerkin method based upon an upwind distortion of the weighting function.

To give an example of upwind test functions, we consider again the solution of the 1D model problem (2.16) with piecewise linear approximations. Let j be an internal node belonging to elements e and $e+1$, see Figure 2.9. The shape functions of node j are given by

$$N_j = \begin{cases} N_2 = \frac{1}{2}(1+\xi) & \text{in element } e, \\ N_1 = \frac{1}{2}(1-\xi) & \text{in element } e+1, \end{cases} \quad \text{for } \xi \in [-1,1].$$

Upwind test functions, $\tilde{w}_j$, giving more weight to the upwind element e than to the downwind element $e+1$, can be constructed by adding and subtracting from the Galerkin weights a bubble function of amplitude proportional to a free parameter β as follows:

$$\tilde{w}_j = \begin{cases} \tilde{w}_2 = \frac{1}{2}(1+\xi) + \frac{3}{4}\beta(1-\xi^2) & \text{in element } e, \\ \tilde{w}_1 = \frac{1}{2}(1-\xi) - \frac{3}{4}\beta(1-\xi^2) & \text{in element } e+1, \end{cases} \quad \text{for } \xi \in [-1,1].$$

If we replace the Galerkin weighting function w_j with the above Petrov–Galerkin test functions in the weak form (2.17) of the convection–diffusion problem (2.16) with $s=0$, the resulting modified weak form reads

$$\int_0^L (\tilde{w}\, a\, u_x + \tilde{w}_x\, \nu\, u_x)\, dx = 0.$$

Discretization of this weak form yields the following equation at the internal node j:

$$\big[1 + (1+\beta)\mathit{Pe}\big](u_j - u_{j-1}) - \big[1 - (1-\beta)\mathit{Pe}\big](u_{j+1} - u_j) = 0.$$

The parameter β controls the amount of numerical diffusion (or upwinding). Note that the previous equation is in fact equation (2.31). Thus, as noted previously, choosing the parameter $|\beta| \geq \beta_{\text{crit}} = 1 - 1/|Pe|$, see equation (2.32), will avoid an oscillatory solution. Moreover, the exact solution at the nodes is obtained with β defined by (2.27), that is $\beta = \coth Pe - 1/Pe$.

Though giving exact solutions for the 1D steady convection–diffusion equation, this initial upwind finite element formulation suffered from severe shortcomings in application to more complicated situations. These early Petrov–Galerkin methods were able to deliver stable numerical results; however, they produce in general an excessive numerical dissipation and were thus subject to the same criticism as upwind differences. Moreover, the formulation requires the use of higher-order weighting functions, which make the computer implementation of this Petrov–Galerkin method more difficult and more costly than the classical Galerkin method as regards numerical integration.

As suggested by Hughes (1978), the upwind effect can also be achieved through a modification of the numerical quadrature rule for the convective term. In the 1D case, in which piecewise linear elements are employed, a single quadrature point, $\bar{\xi}$, is positioned within the element according to

$$\bar{\xi} = \coth Pe - \frac{1}{Pe}.$$

In the case of a bilinear quadrilateral, the location of the quadrature point is defined by

$$\bar{\boldsymbol{\xi}} = \left\{ \begin{array}{c} \bar{\xi} \\ \bar{\eta} \end{array} \right\}, \tag{2.44a}$$

$$\bar{\xi} = \coth Pe_{\xi} - 1/Pe_{\xi}, \qquad \bar{\eta} = \coth Pe_{\eta} - 1/Pe_{\eta}, \tag{2.44b}$$

$$Pe_{\xi} = a_{\xi} h_{\xi}/(2\nu), \qquad Pe_{\eta} = a_{\eta} h_{\eta}/(2\nu), \tag{2.44c}$$

$$a_{\xi} = \boldsymbol{e}_{\xi} \cdot \boldsymbol{a}, \qquad a_{\eta} = \boldsymbol{e}_{\eta} \cdot \boldsymbol{a}, \tag{2.44d}$$

where the unit vectors, $\boldsymbol{e}_{\xi}$ and $\boldsymbol{e}_{\eta}$, and the element lengths, h_{ξ} and h_{η}, are defined as shown in Figure 2.10, and $\boldsymbol{a}$ and ν are evaluated at the origin of the $\{\xi, \eta\}$ coordinate system in the element. Unfortunately, this early two-dimensional scheme was found to exhibit the shortcoming of excessive crosswind diffusion.

Another early approach to the development of finite element schemes of the upwind type was suggested by Belytschko and Eldib (1979). Their method is based on amplifying certain terms in the convection matrix and for linear elements the behavior of the amplification scheme was shown to be similar to other upwind finite element schemes.

2.3.3 The concept of balancing diffusion

In the previous section non-centered discretizations of the convection term (upwind schemes) were presented. However, as noted previously, see Section 2.2.2, another

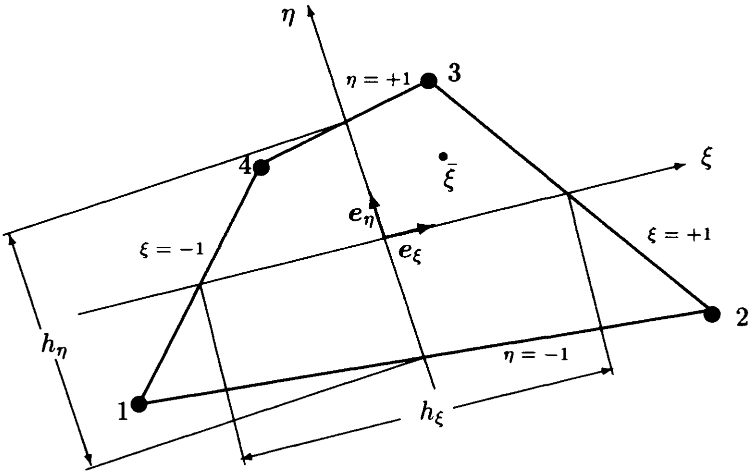


Fig. 2.10 Geometry of four-node quadrilateral element.

alternative to improve the behavior of the Galerkin method (with linear elements) in convection-dominated problems is to add artificial diffusion to counteract the negative dissipation introduced by the Galerkin formulation. This alternative is used next and will be shown to coincide with an upwind scheme.

2.3.3.1 Linear elements in 1D For the simple case of the linear convection–diffusion problem (2.16) with a constant source term, it is possible to formulate an optimal upwind technique producing the exact solution at the nodes of a uniform mesh of linear elements for all values of the Péclet number. Exploiting the results in Section 2.2.2, an optimal method can be constructed by considering a Galerkin approximation of the modified equation

$$a\, u_x - (\nu + \bar{\nu})\, u_{xx} = 0, \text{ with } \bar{\nu} = \beta \frac{ah}{2}$$

containing a free parameter β which governs the amplitude of the added numerical diffusion. Note that the source is taken as zero, $s = 0$, in order to simplify the developments.

For the convection–diffusion problem (2.16) with Dirichlet boundary conditions at $x = 0$ and $x = L$, Hughes and Brooks (1979) replace the usual weak formulation

$$\int_0^L (w\, a\, u_x + w_x\, \nu\, u_x)\; dx = 0,$$

where w denotes the weighting function, by the following weak statement

$$\int_0^L (w\, a\, u_x + w_x\, (\nu + \bar{\nu})\, u_x)\; dx = 0, \tag{2.45}$$

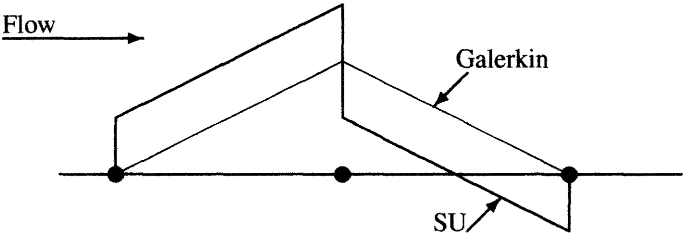


Fig. 2.11 Weighting function of the streamline-upwind (SU) method for linear elements.

where the magnitude of the added diffusion, $\bar{\nu}$, is governed by the free parameter β $(0 \leq \beta \leq 1)$. The optimal value of parameter β is given by (2.27), that is $\beta = \coth Pe - 1/Pe$, and the value $\beta = 1$ corresponds to full upwind differencing as in (2.42).

Since the added numerical diffusion counterbalances the negative diffusion introduced by the Galerkin method, it is also termed *balancing diffusion* (Kelly et al., 1980). In view of the definition of the added diffusion, $\bar{\nu} = \beta a h/2$, we note that the weak statement (2.45) can be rewritten in the form

$$\int_0^L \left[\left(w + \beta\frac{h}{2}w_x\right) a u_x + w_x \nu\, u_x\right] dx = 0, \tag{2.46}$$

which shows that the balancing diffusion method uses a *modified weighting function*, given by $\tilde{w} = w + \beta(h/2)w_x$ for the convective term only. Note that equation (2.46) does not correspond to a consistent Petrov–Galerkin formulation because the modified weighting function is only applied to the convective term. Moreover, the modified function is discontinuous at the inter-element boundaries, as shown in Figure 2.11 for the case of a linear element. Since it gives more weight to the element upstream of a node, the modified function is clearly an upwind-type weighting function. We shall refer to this scheme as the *streamline-upwind* (SU) method in view of its generalization to multiple dimensions. In fact, in several spatial dimensions diffusion is added in the flow direction only and not transversely.

To perform the spatial discretization of the weak form (2.46) using linear finite elements, we proceed as in Section 2.2.1 for the Galerkin formulation. The integrals in (2.46) are easily evaluated in closed form. They lead to the following SU form of the convection matrix:

$$\mathbf{C}^e_{SU} = \frac{a}{2}\begin{pmatrix} -(1-\beta) & 1-\beta \\ -(1+\beta) & 1+\beta \end{pmatrix}.$$

As shown by the weak form (2.46), the element diffusion matrix, $\mathbf{K}^e_{SU}$, possesses the same structure as the Galerkin diffusion matrix, $\mathbf{K}^e$, defined in (2.19b).

2.3.3.2 Quadratic elements in 1D Our objective is now to develop modified weighting functions giving exact results for all values of the Péclet number when approximating the 1D linear convection–diffusion equation by means of a uniform mesh of quadratic finite elements.

As shown in Section 2.2.3, the piecewise quadratic approximation generates two types of nodal equations which possess distinct truncation errors. This somehow complicates the formulation of an added diffusion method with respect to linear finite elements, since different values of the balancing diffusion should be incorporated into the corner node and the mid-side node equations given by the Galerkin formulation.

At mid-side nodes, the optimal value of the added diffusivity is given by the same formula obtained for linear 1D elements, that is equation (2.27). At corner nodes, the Galerkin method introduces another numerical diffusion and the optimal value of the diffusivity is given by equation (2.41). Thus, the artificial viscosity that needs to be added at the mid-side and corner nodes is, respectively, $\bar{\nu} = \beta\, ah/2$ with

$$\beta = \coth \mathit{Pe} - 1/\mathit{Pe} \tag{2.47a}$$

and $\bar{\nu}_{\mathtt{corner}} = \beta_{\mathtt{corner}}\, ah/2$, where

$$\beta_{\mathtt{corner}} = \frac{(\coth \mathit{Pe} - 1/\mathit{Pe}) - (\cosh \mathit{Pe})^2(\coth 2\mathit{Pe} - 1/(2\mathit{Pe}))}{1 - (\cosh \mathit{Pe})^2/2}. \tag{2.47b}$$

We now have all the ingredients to formulate the balancing diffusion method for 1D quadratic elements. As with linear elements the weak form reads

$$\int_0^L \left(\tilde{w}\, a\, u_x + w_x\, \nu\, u_x\right) dx = 0,$$

but now the modified weighting function $\tilde{w}$ depends on the type of nodal equation to be weighted:

$$\tilde{w}_j = \begin{cases} N_j + \beta\,(h/2)\,(dN_j/dx) & \text{with } j \text{ mid-side node,} \\ N_j + \beta_{\mathtt{corner}}\,(h/2)\,(dN_j/dx) & \text{with } j \text{ corner node,} \end{cases}$$

where β and $\beta_{\mathtt{corner}}$ are defined in (2.47).

On this basis, we can perform the spatial discretization of the 1D linear convection–diffusion equation using the quadratic shape functions defined in (2.33). Then, the SU form of the element convection matrix is found to be given by

$$\mathbf{C}^e_{SU} = \frac{a}{2}\begin{pmatrix} -1 + 7/6\,\beta_{\mathtt{corner}} & 4/3\,(1 - \beta_{\mathtt{corner}}) & -1/6\,(2 - \beta_{\mathtt{corner}}) \\ -4/3\,(1 + \beta) & 8/3\,\beta & 4/3\,(1 - \beta) \\ 1/6\,(2 + \beta_{\mathtt{corner}}) & -4/3\,(1 + \beta_{\mathtt{corner}}) & 1 + 7/6\,\beta_{\mathtt{corner}} \end{pmatrix},$$

while the element diffusion matrix, $\mathbf{K}^e_{SU}$, possesses the same structure as the Galerkin diffusion matrix, $\mathbf{K}^e$, defined in (2.34b).

The use of the above balancing diffusion methods for linear and quadratic elements in the solution of convection-dominated problems is illustrated in Figure 2.12 where exact solutions are obtained at the nodes. Compare these results with the corresponding Galerkin approximations shown in Figures 2.1 and 2.6.

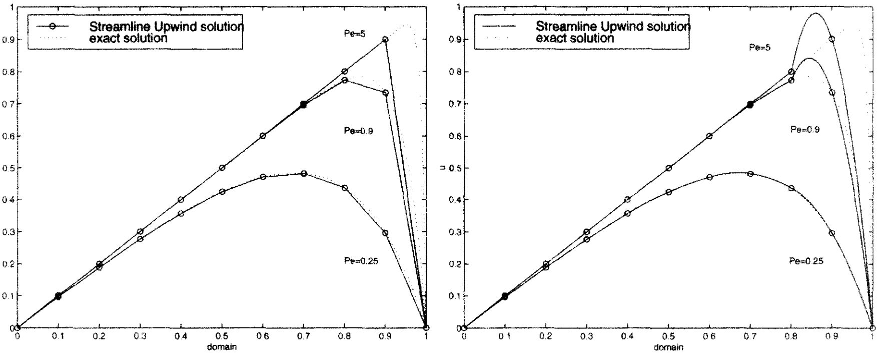


Fig. 2.12 Streamline-upwind (artificial diffusion) solution (solid lines) of the convection–diffusion problem (2.16) with $L = 1$, $s = 1$ using a uniform mesh of 10 linear elements (left) and 5 quadratic elements (right). Dotted lines show the exact solution.

Moreover, it is important to note that the exact nodal solution for quadratic elements presents, in convection-dominated cases, an important overshoot (recall that a uniform mesh is employed). This behavior, which is inherent to the quadratic approximation, will be an important drawback in nonlinear problems such as the ones discussed in Chapters 4 and 6.

2.3.3.3 *Multidimensional case: the concept of SU schemes* The extension to multidimensional domains of the concept of modified weighting functions discussed above is not trivial. The crucial issue is that the balancing diffusion should be added in the flow direction only, and not transversely. The reason is that convective transport takes place along the streamlines and adding diffusion transversely to the flow leads to overly diffusive results, due to an excess of crosswind diffusion.

In order to extend the added diffusion method to deal with multidimensional problems, Hughes and Brooks proposed in a series of papers (Hughes and Brooks, 1979; Brooks and Hughes, 1982; Hughes and Brooks, 1982) to construct the artificial diffusion operator in tensorial form to act only in the flow direction and not transversely. This leads to the concept of SU schemes, which account for the directional character of the convective term. The idea of adding diffusion along the streamlines was also exploited by Kelly et al. (1980) and described as *anisotropic balancing dissipation*. This is achieved by replacing the scalar artificial diffusion coefficient $\bar{\nu}$ in (2.45) by the tensor diffusivity

$$\tilde{\nu}_{ij} = \bar{\nu}\, a_i a_j / ||\boldsymbol{a}||^2, \qquad (2.48)$$

where a_i is the component of flow velocity, $\boldsymbol{a}$, along the coordinate direction x_i.

The development of a general theory to optimally select $\bar{\nu}$ is still an area of current research. For the basic isoparametric elements a simple generalization of the 1D definition is usually adopted. For example, in the case of the bilinear quadrilateral one takes

$$\bar{\nu} = \left(\bar{\xi}\, a_\xi h_\xi + \bar{\eta}\, a_\eta h_\eta\right)/2, \qquad (2.49)$$

where $\bar{\xi}$ and $\bar{\eta}$ are defined in (2.44), and the standard notation for the normalized coordinates defined in Figure 2.10 is employed.

> **Remark 2.9.** The tensor diffusivity, $\tilde{\boldsymbol{\nu}}$, represents a diffusion acting only in the direction of the flow and not transversely. In two dimensions this can be verified assuming that locally the x_1 direction is aligned with the streamlines and the x_2 direction is perpendicular. That is, the components of the convection velocity are $\boldsymbol{a} = (1, 0)^T$. In this coordinate system the artificial diffusivity matrix defined by (2.48) is
>
> $$\tilde{\boldsymbol{\nu}} = \bar{\nu} \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$
>
> which clearly manifests the absence of crosswind diffusion.

In order to determine the ***streamline-upwind test function***, that is the multidimensional modified weighting function, the weak form (2.45) is rewritten for n_{sd} spatial dimensions, namely: find $u \in \mathcal{S}$ such that for all $w \in \mathcal{V}$

$$\int_\Omega \Big[w(\boldsymbol{a} \cdot \boldsymbol{\nabla} u) + \boldsymbol{\nabla} w \cdot (\nu \mathbf{I}_{\mathrm{n_{sd}}} + \tilde{\boldsymbol{\nu}}) \cdot \boldsymbol{\nabla} u \Big] \, d\Omega = 0, \tag{2.50}$$

where $\mathbf{I}_{\mathrm{n_{sd}}}$ is an identity matrix of dimension n_{sd} and the added diffusion $\tilde{\boldsymbol{\nu}}$ has been incorporated. In fact, given the definition of this added tensor diffusivity, see (2.48), the added term can be modified as follows:

$$\int_\Omega \boldsymbol{\nabla} w \cdot \tilde{\boldsymbol{\nu}} \cdot \boldsymbol{\nabla} u \, d\Omega = \int_\Omega \frac{\bar{\nu}}{\|\boldsymbol{a}\|^2} (\boldsymbol{a} \cdot \boldsymbol{\nabla} w)(\boldsymbol{a} \cdot \boldsymbol{\nabla} u) \, d\Omega.$$

Consequently, equation (2.50) can be rewritten as

$$\int_\Omega \left\{ \left[w + \frac{\bar{\nu}}{\|\boldsymbol{a}\|^2} (\boldsymbol{a} \cdot \boldsymbol{\nabla} w) \right] (\boldsymbol{a} \cdot \boldsymbol{\nabla} u) + \nu \boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} u \right\} d\Omega = 0.$$

Therefore, the SU test function, which only affects the convective term, is defined as

$$\tilde{w} = w + \frac{\bar{\nu}}{\|\boldsymbol{a}\|^2} (\boldsymbol{a} \cdot \boldsymbol{\nabla} w). \tag{2.51}$$

Finally, the SU method can be interpreted as the Galerkin method plus an extra term introducing the SU added numerical diffusivity:

$$\underbrace{\int_\Omega \Big[w(\boldsymbol{a} \cdot \boldsymbol{\nabla} u) + \nu \boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} u \Big] \, d\Omega}_{\text{Standard Galerkin}} + \underbrace{\int_\Omega \frac{\bar{\nu}}{\|\boldsymbol{a}\|^2} (\boldsymbol{a} \cdot \boldsymbol{\nabla} w)(\boldsymbol{a} \cdot \boldsymbol{\nabla} u) \, d\Omega}_{\text{Added SU term}} = 0. \tag{2.52}$$

As noted previously, the perturbation added to the test function is discontinuous along the element edges or interfaces. Thus, the extra term in (2.52) is only computed in the element interiors. When this methodology is generalized for the original steady convection–diffusion problem, equations (2.1), the weak form, (2.7), associated with the SU stabilized technique becomes

$$a(w,u) + c(\boldsymbol{a}; w, u) \\ \underbrace{+ \sum_e \int_{\Omega^e} \frac{\bar{\nu}}{\|\boldsymbol{a}\|^2} (\boldsymbol{a}\cdot\boldsymbol{\nabla} w)(\boldsymbol{a}\cdot\boldsymbol{\nabla} u)\, d\Omega}_{\text{SU stabilization term}} = (w,s) + (w,h)_{\Gamma_N} \qquad (2.53)$$

where $\bar{\nu}$ is given by (2.49) and the constants therein by equations (2.44).

This approach, called the SU formulation, makes use of upwind test functions only for the convective term. As was shown, it is equivalent to the use of an artificial diffusion coefficient acting only along the streamline direction. This method produces smooth solutions for high Péclet numbers. Moreover, it is able to deliver exact results on a uniform 1D grid for the linear, constant coefficient, convection–diffusion equation. However, accuracy problems still remain in more complicated cases, such as spatially variable source term or convection velocity, as well as time-dependent problems. In fact, modifying the test function only for the convective term produces a non-residual formulation. Note that the true solution of the differential equation is no longer a solution to the weak problem (2.53). A discussion of these accuracy problems (in a finite difference context) and numerical remedies to overcome them can be found in Christie (1985), Leonard (1979) and Morton (1996). Herein, we shall limit ourselves to indicate that when the governing equation includes a source term, this term must, for consistency, be discretized using the same SU weighting function as the convective term. Failing to do this would lead to erroneous results, as will be seen in the next section, see also the numerical results in Figure 2.13.

Hughes and Brooks (1982) subsequently proposed to apply the modified weighting function to all terms in the equation in order to obtain a consistent formulation. Moreover, they noted that for linear elements the perturbation to the standard test function can be neglected in the diffusion term. The concept of adding diffusion along the streamlines in a consistent manner has been successfully exploited in the Streamline-Upwind Petrov–Galerkin (SUPG) method that will be described next.

2.4 STABILIZATION TECHNIQUES

In order to stabilize the convective term in a consistent manner (*consistent stabilization*), ensuring that the solution of the differential equation is also a solution of the weak form, Hughes and co-workers have proposed several techniques. They follow a structure similar to (2.53). That is, an extra term over the element interiors is added to the Galerkin weak form. This term is a function of the residual of the differential equation to ensure consistency. Note that this is not the case in (2.53). These methods are especially designed for the steady convection–diffusion equation, see (2.1a), or more generally for the steady convection–diffusion–reaction equation, namely

$$\boldsymbol{a}\cdot\boldsymbol{\nabla} u - \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla} u) + \sigma u = s \quad \text{in } \Omega, \qquad (2.54)$$

with the usual Dirichlet and Neumann boundary conditions shown in (2.1). The residual of the differential equation (2.54) is

$$\mathcal{R}(u) = \underbrace{\boldsymbol{a}\cdot\boldsymbol{\nabla} u - \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla} u) + \sigma u}_{\mathcal{L}(u)} - s = \mathcal{L}(u) - s, \tag{2.55}$$

where $\mathcal{L}$ is the differential operator associated with the differential equation. Note that as soon as we restrict ourselves to the finite dimensional spaces standard in finite elements (and u is replaced by u^h) $\mathcal{R}(u)$ is computed only for each element interior Ω^e. The general form of these (consistent) *stabilization techniques* is

$$a(w,u) + c(\boldsymbol{a};w,u) + (w,\sigma\,u) \\ + \underbrace{\sum_e \int_{\Omega^e} \mathcal{P}(w)\,\tau\,\mathcal{R}(u)\,d\Omega}_{\text{stabilization term}} = (w,s) + (w,h)_{\Gamma_N} \tag{2.56}$$

where $\mathcal{P}(w)$ is a certain operator applied to the test function, τ is the stabilization parameter (also called intrinsic time), and $\mathcal{R}(u)$ is the residual of the differential equation, see (2.55). The stabilization techniques are characterized by the definition of $\mathcal{P}(w)$. Here the Streamline-Upwind Petrov–Galerkin (SUPG) and the Galerkin/Least-squares (GLS) methods will be presented, see for instance Codina (1998) for a general presentation.

> **Remark 2.10.** Note that in this section we have introduced the reaction, or production, term in the original convection–diffusion problem. This is done because now stabilization techniques usually treat both equations, (2.1a) and (2.54), in a uniform manner. Specific analyses of stabilization techniques for the convection–diffusion–reaction equation can be found in Tezduyar and Park (1986), Harari and Hughes (1994), or Idelsohn et al. (1996) among others.

2.4.1 The SUPG method

This stabilization technique is defined by taking

$$\mathcal{P}(w) = \boldsymbol{a}\cdot\boldsymbol{\nabla} w, \tag{2.57}$$

which, in fact, corresponds to the perturbation of the test function introduced in the SU method, see (2.53). Note that this method corresponds to the standard weak forms (2.5) or (2.7) with the SU test function (2.51) consistently applied to all terms of the equation. Thus, since the space of the test functions does not coincide with the space of the interpolation functions, this is in fact a Petrov–Galerkin formulation.

The restriction of the weak form (2.56) for the SUPG method, that is with the perturbation defined in (2.57), to the usual finite dimensional subspaces leads to the discrete problem that must be solved: find $u^h \in \mathcal{S}^h$ such that

$$\begin{aligned} a(w^h, u^h) &+ c(\boldsymbol{a}; w^h, u^h) + (w^h, \sigma u^h) \\ &+ \sum_e \int_{\Omega^e} (\boldsymbol{a} \cdot \boldsymbol{\nabla} w^h)\tau\big[\boldsymbol{a} \cdot \boldsymbol{\nabla} u^h - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} u^h) + \sigma u^h - s\big]\, d\Omega \\ &= (w^h, s) + (w^h, h)_{\Gamma_N} \qquad \text{for all } w^h \in \mathcal{V}^h, \end{aligned} \tag{2.58}$$

where the stabilization parameter τ can be defined as

$$\tau = \bar{\nu}/\|\boldsymbol{a}\|^2 \tag{2.59}$$

with $\bar{\nu}$ given in (2.49) for 2D and $\bar{\nu} = \beta a h/2$ in 1D. Section 2.4.3 discusses in more detail the definition of τ.

Remark 2.11 (Exact nodal solutions in 1D for convection–diffusion). For the convection–diffusion equation discretized with linear elements, the stabilization term reduces to

$$\sum_e \int_{\Omega^e} (\boldsymbol{a} \cdot \boldsymbol{\nabla} w^h)\tau(\boldsymbol{a} \cdot \boldsymbol{\nabla} u^h - s)\, d\Omega$$

because second derivatives of u^h cancel out. The difference operator acting on u^h is identical to the one obtained with the non-residual formulation presented in Section 2.3.3.1. Thus, $\beta = \coth \mathit{Pe} - 1/\mathit{Pe}$ or $\tau = (h/2a)(\coth \mathit{Pe} - 1/\mathit{Pe})$ produce the exact nodal solution on a uniform mesh in 1D.

For quadratic elements, however, the stabilization term becomes

$$\sum_e \int_{\Omega^e} (\boldsymbol{a} \cdot \boldsymbol{\nabla} w^h)\tau\big[\boldsymbol{a} \cdot \boldsymbol{\nabla} u^h - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} u^h) - s\big]\, d\Omega,$$

and the second derivatives must be accounted for in the element interiors. Moreover, as previously observed in Section 2.3.3.2, the intrinsic time τ must be different for mid-side, τ_{m}, and corner, τ_{c}, nodes. The parameters computed previously, see (2.47), will not produce the nodally exact solution on a uniform mesh when used in (2.58) because the difference operator has changed. In fact, taking into account these two parameters, one for the mid-side nodes and the other for the corner nodes, the difference equations associated with the weak form (2.58) become:

1. At a *mid-side node*,

$$a\Big(\frac{u_{j+1} - u_{j-1}}{2h}\Big) - (\nu + \tau_{\mathrm{m}} a^2)\Big(\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2}\Big) = 0.$$

2. At a *corner node*,

$$\begin{aligned} 2\Big(a - \tau_{\mathrm{c}}\frac{3a\nu}{h^2}\Big)\Big(\frac{u_{j+1} - u_{j-1}}{2h}\Big) - \Big(a - \tau_{\mathrm{c}}\frac{6a\nu}{h^2}\Big)\Big(\frac{u_{j+2} - u_{j-2}}{4h}\Big) \\ -2(\nu + \tau_{\mathrm{c}}\, a^2)\Big(\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2}\Big) + (\nu + \tau_{\mathrm{c}}\, a^2)\Big(\frac{u_{j+2} - 2u_j + u_{j-2}}{4h^2}\Big) = 0. \end{aligned}$$

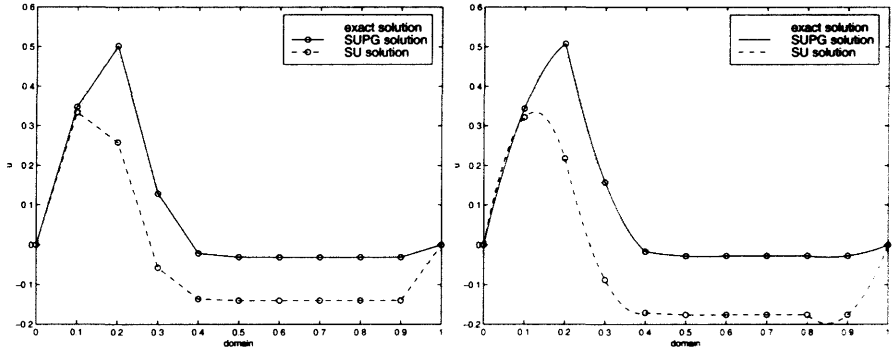


Fig. 2.13 SUPG (solid line) and SU (dashed line) solutions of the convection–diffusion problem (2.60) at $Pe = 5$ using a uniform mesh of 10 linear elements (left) and five quadratic elements (right). The dotted line shows the exact solution.

Note that the source term is taken equal to zero in order to simplify the expressions. However, this analysis can be extended to constant source terms as previously done in Section 2.3.3.2. Imposing now that the coefficients multiplying the nodal values are equal to the coefficients in the exact difference scheme, see Section 2.2.4, the optimal parameters are obtained: $\tau_{\rm m} = \beta\, h/(2a)$ with, as always, $\beta = \coth Pe - 1/Pe$; and $\tau_{\rm c} = \beta_{\rm c}\, h/(2a)$ with

$$\beta_{\rm c} = \frac{(2Pe - 1) + (-6Pe + 7)e^{-2Pe} + (-6Pe - 7)e^{-4Pe} + (2Pe + 1)e^{-6Pe}}{(Pe + 3) + (-7Pe - 3)e^{-2Pe} + (7Pe - 3)e^{-4Pe} - (Pe + 3)e^{-6Pe}}.$$

With these parameters, exact nodal solutions are obtained. That is, Figure 2.12 is reproduced exactly with the SUPG formulation. The above parameters were originally proposed by Codina (1993b) who used the concept of linear difference equation associated with the homogeneous linear convection–diffusion equation, see Remark 2.7, in order to determine them.

As noted earlier, the SUPG formulation performs better than the original SU technique. The influence of the consistent formulation can be noticed with a simple modification of the source term. Numerical solutions of the convection–diffusion problem,

$$\begin{cases} au_x - \nu u_{xx} = 5\,e^{-100(x-\frac{1}{8})^2} - 5\,e^{-100(x-\frac{1}{4})^2} & \text{in }]0,1[\\ u = 0 & \text{at } x = 0 \text{ and } x = 1, \end{cases} \tag{2.60}$$

are compared with the exact one in Figure 2.13.

One important issue in the SUPG method is the definition of the stabilization parameter τ. The stability and convergence analysis of this method (see Johnson, Nävert and Pitkäranta, 1984) allows us to determine the behavior of τ. The fact that the added stabilization term, see equation (2.58), is not symmetric introduces technical difficulties in establishing the stability of SUPG. This is avoided in the Galerkin/Least-squares stabilization technique because it introduces a symmetric stabilization term in a consistent manner.

2.4.2 The Galerkin/Least-squares method

The GLS technique is defined by imposing that the stabilization term in (2.56) is an element-by-element weighted least-squares formulation of the original differential equation. This corresponds to the following choice for the operator applied to the test function:

$$\mathcal{P}(w) = \mathcal{L}(w) = \boldsymbol{a} \cdot \boldsymbol{\nabla} w - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} w) + \sigma w. \tag{2.61}$$

With such a definition of operator $\mathcal{P}$, the weak form that must be solved is: find $u^h \in \mathcal{S}^h$ such that

$$\begin{aligned} a(w^h, u^h) + c(\boldsymbol{a}; w^h, u^h) + (w^h, \sigma\, u^h) + \sum_e \int_{\Omega^e} \mathcal{L}(w^h)\, \tau \left[\mathcal{L}(u^h) - s\right] d\Omega \\ = (w^h, s) + (w^h, h)_{\Gamma_N} \qquad \text{for all } w^h \in \mathcal{V}^h. \end{aligned} \tag{2.62}$$

This equation can also be written, for the steady convection–diffusion–reaction equation (2.54), as

$$\begin{aligned} & a(w^h, u^h) + c(\boldsymbol{a}; w^h, u^h) + (w^h, \sigma\, u^h) \\ & + \sum_e \int_{\Omega^e} \left[\boldsymbol{a} \cdot \boldsymbol{\nabla} w^h - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} w^h) + \sigma w^h\right] \tau \left[\boldsymbol{a} \cdot \boldsymbol{\nabla} u^h - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} u^h) + \sigma u^h\right] d\Omega \\ & = (w^h, s) + (w^h, h)_{\Gamma_N} + \sum_e \int_{\Omega^e} \left[\boldsymbol{a} \cdot \boldsymbol{\nabla} w^h - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} w^h) + \sigma w^h\right] \tau s, \end{aligned} \tag{2.63}$$

where it is important to notice that the stabilization term that affects the l.h.s. is symmetric. This symmetry is a major advantage in establishing stability.

From a practical point of view there is no major difference between SUPG and GLS methods. In fact, both methods are identical for convection–diffusion (no reaction) and with linear elements (the second-order derivatives are zero in the element interiors). Moreover, the qualitative influence of each term in the definition of $\mathcal{P}$, equation (2.61), may be interpreted as follows:

$$\mathcal{P}(w) = \mathcal{L}(w) = \underbrace{\boldsymbol{a} \cdot \boldsymbol{\nabla} w}_{\text{SUPG}} - \underbrace{\boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} w)}_{0} + \underbrace{\sigma w}_{\text{Galerkin}} .$$

The first term corresponds to the SUPG stabilization, the second term is zero for linear elements, and the third term is a Galerkin weighting. Thus, for linear elements and a constant positive reaction, GLS is SUPG with the Galerkin term weighted $1 + \sigma\,\tau$ times more. This implies that the instabilities introduced by Galerkin are a little more amplified in GLS compared with SUPG. See Figure 2.14 for a comparison of these methods.

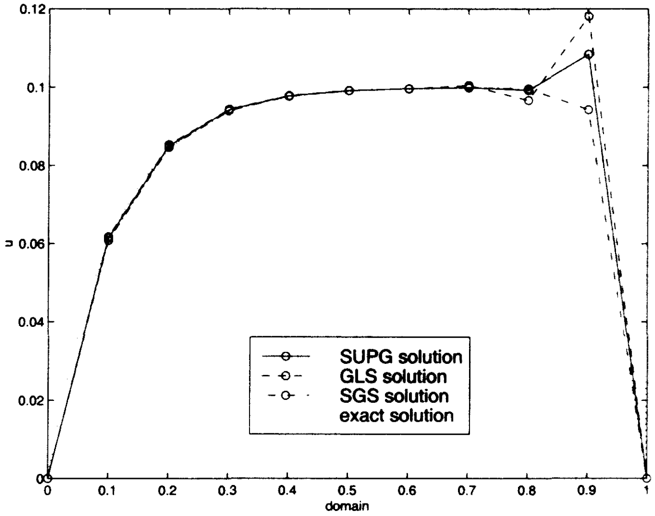


Fig. 2.14 Comparison between SUPG (solid line), GLS (dashed line) and SGS (dash–dot line) solutions of the convection–diffusion–reaction problem (2.54) for $a = 1$, $\nu = 10^{-2}$, $\sigma = 10$ and a uniform mesh of 10 linear elements. The dotted line shows the exact solution.

Remark 2.12. This minor problem of the GLS stabilization technique is overcome in the simplest version of the sub-grid scale (SGS) method, see Section 2.5.3, because in this case the stabilization operator involves the adjoint operator, $\mathcal{L}^*$, of $\mathcal{L}$, namely $\mathcal{P}(w) = -\mathcal{L}^*(w) = \boldsymbol{a}\cdot\boldsymbol{\nabla} w + \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla} w) - \sigma w$. Consequently, following the same rationale as before, in this case the Galerkin term is weighted by $1 - \sigma\,\tau$ and thus has less influence than in SUPG.

2.4.3 The stabilization parameter

The parameter τ plays a major role in stabilization techniques. Note first that τ is in fact a stabilization coefficient matrix, as noted earlier for quadratic elements or more typically for systems of differential equations, see for instance the original paper of Hughes and Mallet (1986a) or more recently Codina (2000). Here the scalar 1D convection–diffusion–reaction equation (2.54) is used for illustration purposes.

For scalar equations a number of definitions of parameter τ have been proposed and tested (see for instance Tezduyar and Ganjoo, 1986; Tezduyar and Osawa, 2000). Here, however, only three are recalled.

First, *superconvergence* (in the form of nodally exact results) is obtained for linear elements in 1D for convection–diffusion, see Remark 2.11 and Section 2.2.2, if

$$\tau = \frac{h}{2a}\Big(\coth(Pe) - 1/Pe\Big).$$

This property can be generalized for quadratic elements (Remark 2.11) but not for higher dimensions or general convection–diffusion systems.

It is obvious that τ must vanish when the mesh is refined (no stabilization is necessary for a fine enough mesh). In fact, convergence is affected by the asymptotic behavior of τ. An error analysis allows us to determine the structure of the stabilization coefficient as a function of the discretization (mesh size h) and the parameters in the differential equation (convection velocity a, diffusivity ν and reaction σ).

Second, an algebraic analysis is the basis of Codina's (2000) definition of τ,

$$\tau = \left(\frac{2a}{h} + \frac{4\nu}{h^2} + \sigma\right)^{-1} = \frac{h}{2a}\left(1 + \frac{1}{\mathit{Pe}} + \frac{h}{2a}\sigma\right)^{-1}, \tag{2.64}$$

which corresponds with the second-order accurate formula presented by Shakib, Hughes and Johan (1991) based on a local truncation error analysis.

Third, Shakib and co-workers propose the fourth-order accurate formula

$$\tau = \left(\left(\frac{2a}{h}\right)^2 + 9\left(\frac{4\nu}{h^2}\right)^2 + \sigma^2\right)^{-1/2} = \frac{h}{2a}\left(1 + \frac{9}{\mathit{Pe}^2} + \left(\frac{h}{2a}\sigma\right)^2\right)^{-1/2}. \tag{2.65}$$

An asymptotic analysis near the different limits (i.e., Taylor series expansions for $\mathit{Pe} = 0$, $\mathit{Pe} = \infty$, $\sigma = 0$ and $\sigma = \infty$) shows the superior convergence of (2.65) to the theoretical values compared with (2.64). However, the fourth-order accuracy does not extend to higher dimensions or general convection–diffusion systems. Nevertheless, numerical experiments indicate that (2.65) presents slightly lower errors (in the $\mathcal{L}_2$ norm) compared with (2.64).

It is important to note that for higher-order finite elements, apart from the results discussed in Section 2.2.4 and Remark 2.11 (see also Franca, Frey and Hughes, 1992; Codina, 1993b), no optimal definition of τ exits. Numerical experiments seem to indicate that for finite elements of order p the value of the stabilization parameter should be approximately τ/p.

Finally, note that this chapter has focused on convection–diffusion. Reaction has been added in some cases but diffusion–reaction problems are not discussed in detail, see Harari, Frey and Franca (2002) for the definition of stability parameters in such problems. The proper definition of the stabilization parameter is important in practical computations. Thus it must be adapted to the equation under consideration and to non-uniform meshes (element-by-element evaluation).

2.5 OTHER STABILIZATION TECHNIQUES AND NEW TRENDS

The subject of stabilization methods for convection-dominated transport problem is still an area of active research. Alternative approaches to the stabilization techniques discussed so far have been proposed in recent years and new trends are emerging. In this section we shall present a brief overview of three alternative stabilization techniques to SUPG and GLS and introduce a new promising technique based upon the so-called variational multiscale method.

2.5.1 Finite increment calculus

An alternative approach for deriving stabilized formulations for convection–diffusion and fluid flow problems has been suggested by Oñate (1998) under the name *finite increment calculus*. The basic idea in this method is that most stabilized numerical schemes can be derived by applying the standard Galerkin formulation to a so-called stabilized form of the governing differential equations of the problem. These equations are obtained by invoking higher-order balance (equilibrium) statements on a finite (as opposed to infinitesimal) domain.

We shall illustrate the process for a 1D convection–diffusion problem. The classical balance equation for this problem is given in (2.16) and reads

$$a u_x - \nu u_{xx} = s(x).$$

For simplicity, we shall assume that the convection velocity a is constant. The idea is to replace the above balance equation by a higher-order one derived on a space slab $(x-\delta, x)$ of finite dimension δ. Over such a finite subdomain, the unknown u, the convective and diffusive fluxes and the source term may experience finite variations. As a consequence, the higher-order balance equation takes the following form in terms of the fluxes entering and leaving the subdomain:

$$\left(F_{conv}^{out} - F_{conv}^{in}\right) + \left(F_{diff}^{out} - F_{diff}^{in}\right) = \int_{x-\delta}^{x} s(x)\, dx,$$

where

$$F_{conv}^{out} = a\, u, \qquad F_{conv}^{in} = a\left(u - \delta\frac{\partial u}{\partial x} + \frac{1}{2}\delta^2\frac{\partial^2 u}{\partial x^2}\right) + \mathcal{O}(\delta^3),$$

$$F_{diff}^{out} = -\nu\,\frac{\partial u}{\partial x}, \qquad F_{diff}^{in} = -\nu\,\frac{\partial}{\partial x}\left(u - \delta\frac{\partial u}{\partial x} + \frac{1}{2}\delta^2\frac{\partial^2 u}{\partial x^2}\right) + \mathcal{O}(\delta^3),$$

$$\int_{x-\delta}^{x} s(x)\, dx = \frac{\delta}{2}\left(s(x-\delta) + s(x)\right) + \mathcal{O}(\delta^3) = \delta\, s(x) - \frac{\delta^2}{2}\frac{\partial s(x)}{\partial x} + \mathcal{O}(\delta^3).$$

This statement of equilibrium is easily seen to produce the higher-order balance equation, which is approximated to first order by

$$\left(1 - \frac{\delta}{2}\frac{\partial}{\partial x}\right)\left(a\, u_x - \nu\, u_{xx} - s(x)\right) = 0.$$

Note that the modified balance equation reduces to the standard form by assuming that the dimension of the balance space slab is infinitesimal. In the presence of Dirichlet conditions at both ends of the domain $[0, L]$, application of the standard Galerkin method to the higher-order balance equation produces the following consistent variational form:

$$\int_0^L \left(w + \frac{\delta}{2}\frac{\partial w}{\partial x}\right)\left(a\, u_x - \nu\, u_{xx} - s(x)\right) dx = 0,$$

in which the weighting function is identical in structure to an SU weighting function. Therefore, for suitable values of parameter δ, application of the standard Galerkin method to the higher-order balance equation produces stabilized finite element schemes for convection–diffusion similar to those obtained with the Petrov–Galerkin approaches discussed in the previous sections of this chapter. A similar approach had been invoked earlier by Donea, Belytschko and Smolinski (1985) with application to quadratic shape functions.

As shown by Oñate and co-workers, the modified differential equation can be used to derive a numerical scheme for iteratively computing the stabilization parameters in a sort of model adaptivity procedure. Further details on the finite increment calculus approach to stabilization and its application to convection–diffusion and fluid flow problems can be found in Oñate (1998) and Oñate and Manzán (1999) and references therein. A similar approach may also be found in the work of Ilinca, H étu and Peletier (2000).

2.5.2 Bubble functions and wavelet approximations

As amply discussed in the present chapter, the Galerkin method employing standard finite elements is not a robust approach for the numerical modeling of transport problems exhibiting multiscale behavior in the form of localized phenomena such as interior and boundary layers. This has motivated the development of new methods capable of dealing with phenomena involving both coarse-scale and fine-scale aspects. In this context, p-type finite elements typically used in adaptive refinement procedures, as well as other non-classical finite element procedures, such as elements enriched with bubble or wavelet functions, have proven to provide an interesting alternative to the previously described stabilized methods. In this section, we shall illustrate the construction of finite element models enriched with bubbles or wavelet functions to improve the resolution of localized phenomena and thereby enhance the stability of solutions to convection-dominated problems.

2.5.2.1 Bubble functions Bubbles are functions defined on the interiors of finite elements which vanish on the element boundaries. Baiocchi, Brezzi and Franca (1993) were the first to point out that the enrichment of the finite element space by summation of polynomial bubble functions results in a stabilization procedure for convection–diffusion problems that is formally similar to SUPG and GLS. Unfortunately, standard (low-order) polynomial bubbles are not able to adequately resolve thin internal layers with steep gradients or other small-scale phenomena. Then, the concept of enriching the Galerkin finite element method with so-called residual-free bubbles, see for instance Brezzi, Franca and Russo (1998) or Franca, Nesliturk and Stynes (1998), or with element Green's functions, see Hughes (1995) or Hughes et al. (1998), was introduced to provide a more general framework for the discretization of problems involving multiscale phenomena.

The basic idea in bubble function methods is to decompose the solution of a given boundary value problem into the sum of a coarse-scale solution and a fine-scale one. The classical Galerkin finite element method is used to represent the coarse-scale response, that is the resolvable part of the solution for the given finite element mesh. Bubble functions then take care of the fine-scale aspects of the solution which cannot be resolved by the finite element mesh. Bubbles have a stabilizing effect for convection–diffusion problems similar to that of the added diffusion method. A simple application of the bubble function method as a stabilizing formulation for convection–diffusion problems is described in Section 2.6.1.

2.5.2.2 Wavelet functions Recently, the finite element application of wavelets has become an active research area, in particular, to model localization phenomena. Like bubbles, wavelets are functions with local support. Wavelet elements have the potential of capturing localized phenomena and computing multiscale solutions to partial differential equations with higher convergence rates than conventional finite element methods. Dahmen et al. (1997) provide an overview of recent progress in the development and use of wavelet methods in the fluid mechanics area.

In the finite element solution of convection-dominated problems exhibiting internal or boundary layers, advantage can be taken of the properties of wavelet-based approximations to achieve an accurate localization of such layers. When wavelet approximations are combined with the Galerkin finite element method, the enhanced accuracy is obtained at the cost of some computational complexity. In particular, the computation of the numerical values of the wavelet functions and of their derivatives at the quadrature points is not an easy task. Another delicate aspect of the method is the prescription of the boundary conditions which is more complicated than for the standard finite element method. Several methods have been suggested to implement them, including Lagrange multipliers and penalty methods.

2.5.3 The variational multiscale method

The variational multiscale method introduced by Hughes (1995) provides the necessary mathematical framework for the construction of so-called sub-grid scale models. In its simplest form it is very similar to the *sub-grid viscosity* methods presented in a series of papers by Guermond (1999a; 1999b). The idea here is again that standard finite element approximations can only resolve the coarse-scale aspects of problems involving multiscale behavior. The variational multiscale method is based on the additive decomposition of the solution, u, on a coarse-scale component, $\bar{u}$, which can be resolved by the considered finite element mesh, and a fine-scale component, u', which one attempts to determine analytically. Notice that the fine-scale solution, u', actually represents the error, $u - \bar{u}$, of the coarse-scale component, while $\bar{u}$ is the resolvable scale approximated by the finite element method.

In order to simplify the exposition, the variational multiscale method is illustrated in the context of the convection–diffusion–reaction equation, see (2.54), with homogeneous Dirichlet boundary conditions (and with the usual assumption of a divergence-free convection velocity, $\boldsymbol{a}$), see Hughes et al. (1998) for a complete exposition. The standard variational form of this boundary value problem is, as usual, find $u \in \mathcal{V}$ such that

$$a(w, u) + c(\boldsymbol{a}; w, u) + \sigma\ (w, u) = (w, s) \qquad \text{for all } w \in \mathcal{V}.$$

The additive decomposition of the solution, $u = \bar{u} + u'$, and test function, $w = \bar{w} + w'$, is accompanied by the corresponding splitting of the functional space $\mathcal{V} = \bar{\mathcal{V}} \oplus \mathcal{V}'$ into a finite dimensional coarse-scale subspace and a necessarily infinite dimensional fine-scale subspace. Note that all the previously presented functions verify the homogeneous Dirichlet boundary conditions. Under these conditions the weak form becomes

$$a(\bar{w} + w', \bar{u} + u') + c(\boldsymbol{a}; \bar{w} + w', \bar{u} + u') + \sigma\ (\bar{w} + w', \bar{u} + u') = (\bar{w} + w', s),$$

which, by virtue of the linear independence of $\bar{w}$ and w', splits into two problems:

$$a(\bar{w}, \bar{u}) + c(\boldsymbol{a}; \bar{w}, \bar{u}) + \sigma\ (\bar{w}, \bar{u}) \\ + a(\bar{w}, u') + c(\boldsymbol{a}; \bar{w}, u') + \sigma\ (\bar{w}, u') = (\bar{w}, s) \quad \forall \bar{w} \in \bar{\mathcal{V}}, \tag{2.66a}$$

$$a(w', u') + c(\boldsymbol{a}; w', u') + \sigma\ (w', u') \\ + a(w', \bar{u}) + c(\boldsymbol{a}; w', \bar{u}) + \sigma\ (w', \bar{u}) = (w', s) \quad \forall w' \in \mathcal{V}'. \tag{2.66b}$$

The first problem governs the resolved scales and the second one the unresolvable scales.

The objective now is to solve analytically the fine-scale problem (2.66b) as a function of the coarse-scale solution $\bar{u}$. Then, this u', as a function of $\bar{u}$, is substituted in (2.66a) to obtain an equation for $\bar{u}$. A Green's function technique would be appropriate in some problems to obtain the fine-scale solution. However, the fine-scale Green's function belongs to an infinite dimensional subspace and is non-local. It is then standard to impose $u' = 0$ along the finite element edges in order to localize the fine-scale problem in the interior of each finite element.

Then the Euler–Lagrange equations associated with the fine-scale problem (2.66b) become

$$\begin{cases} \Pi\, \mathcal{L}(u') = -\Pi\big[\mathcal{L}(\bar{u}) - s\big] & \text{in } \Omega^e \\ u' = 0 & \text{on } \Gamma^e \end{cases} \tag{2.67}$$

where $\mathcal{L}(u) = \boldsymbol{a} \cdot \boldsymbol{\nabla} u - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} u) + \sigma u$ is the differential operator associated with the original convection–diffusion–reaction problem, and Π denotes the $\mathcal{L}_2$ projection onto $\mathcal{V}'$. Note that this problem for u' is driven by the residual of the coarse (resolvable) finite element solution. The simplest way to approximate the solution of problem (2.67) is to assume that

$$u' = -\tau\big[\mathcal{L}(\bar{u}) - s\big], \tag{2.68}$$

which obviously does not even verify the boundary conditions.

At this point we note, in view of the above assumption, that

$$a(\bar{w}, u') + c(\boldsymbol{a}; \bar{w}, u') + \sigma\ (\bar{w}, u') \\ = \sum_e \int_{\Omega^e} \big[-\boldsymbol{a} \cdot \boldsymbol{\nabla} \bar{w} - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} \bar{w}) + \sigma \bar{w}\big]\, u'\, d\Omega = \sum_e \big(\mathcal{L}^*(\bar{w}), u'\big)_{\Omega^e}, \tag{2.69}$$

where the adjoint operator $\mathcal{L}^*(u) = -\boldsymbol{a} \cdot \boldsymbol{\nabla} u - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} u) + \sigma u$ has been introduced. Now, the approximation (2.68) is substituted in (2.69), which is used to write the coarse-scale problem (2.66a) in a standard stabilization form

$$a(\bar{w}, \bar{u}) + c(\boldsymbol{a}; \bar{w}, \bar{u}) + \sigma\ (\bar{w}, \bar{u}) + \sum_e \tau\big(-\mathcal{L}^*(\bar{w}), \mathcal{L}(\bar{u}) - s\big)_{\Omega^e} = (\bar{w}, s).$$

Note that this equation corresponds to the standard stabilized form (2.56) with $\mathcal{P}(w) = -\mathcal{L}^*(w)$. This method is called *sub-grid scale* (SGS); it delivers similar results to the SUPG and GLS techniques unless the reaction term is dominant, see Remark 2.12. Figure 2.14 presents a comparison of these methods. The results for quadratic elements present a similar qualitative behavior and thus are not shown.

This method was proposed by Hughes (1995). A detailed presentation can be found in Hughes et al. (1998) where better approximations to the solution of the fine-scale problem (2.67) than the simplest algebraic approximation (2.68) are also discussed. A rigorous development of this technique for systems of differential equations without the restriction that $u' = 0$ along the element edges can be found in Codina (2000). Note, however, that the potentiality of this techniques extends far beyond stabilization techniques for convection–diffusion–reaction equations. See, for instance, Hughes, Mazzei and Jansen (2000) or Hughes et al. (2001) where this technique is used in turbulent flows to introduce modeling (Reynolds stresses) at small scales.

2.5.4 Complements

The idea of locally enriching the approximation basis in order to better reflect the local character of the solution has also been exploited with success in other classes of methods. For instance, the so-called generalized finite element method, see Strouboulis, Copps and Babuška (2000), which results from a combination of the standard finite element method and the partition of unity method introduced by Melenk and Babuška (1996), also provides an efficient way to model localization phenomena (internal and boundary layers) with higher convergence rates than conventional finite element methods. Similarly, modern mesh-free (or particle) methods pioneered by Nayroles, Touzot and Villon (1992) may be used to enrich a finite element mesh where needed without any need for remeshing, see Huerta and Fern ández-Méndez (2000). Methods in this class include the Element-Free Galerkin (EFG) method developed by Belytschko et al. (1994; 1996) and the Reproducing Kernel Particle Method (RKPM) introduced by Liu et al. (1995). Also noteworthy is the h-p clouds method introduced by Duarte and Oden (1996).

2.6 APPLICATIONS AND SOLVED EXERCISES

2.6.1 Construction of a bubble function method

In order to illustrate the use of the bubble function method introduced in Section 2.5.2.1, we consider the convection–diffusion equation with homogeneous Dirichlet boundary conditions. The standard Galerkin formulation of this problem consists of the following: find $u^h \in \mathcal{V}^h$ such that $\forall w^h \in \mathcal{V}^h$

$$a(w^h, u^h) + c(a; w^h, u^h) = (w^h, s).$$

In the bubble function method, each interpolation and test function, u^h and w^h, is decomposed in the standard piecewise continuous polynomials of Galerkin finite elements plus bubble functions:

$$u^h = u_G^h + u_B^h, \quad \text{and } w^h = w_G^h + w_B^h,$$

where subscript G refers to the standard polynomial approximating function of the Galerkin method and subscript B to the bubble part of the approximation, which is required to take on zero value at element boundaries.

This last property of the bubble functions allows us to use the classical static condensation procedure to extract the bubble part u_B^h of the solution in each individual element as a function of the piecewise polynomial part u_G.

We shall illustrate this idea on the homogeneous version of the 1D boundary value problem (2.16). In 1D the bubble function for a linear element is given by $(1-\xi^2)$, so that the decomposition becomes

$$u^h = \overbrace{\frac{1}{2}(1-\xi)\,u_1 + \frac{1}{2}(1+\xi)\,u_2}^{u_G^h} + \overbrace{(1-\xi^2)\,u_3}^{u_B^h} \tag{2.70a}$$

$$w^h = \underbrace{\frac{1}{2}(1-\xi)\,w_1 + \frac{1}{2}(1+\xi)\,w_2}_{w_G^h} + \underbrace{(1-\xi^2)\,w_3}_{w_B^h} \tag{2.70b}$$

where subscripts 1 and 2 characterize the end nodes of the linear element and subscript 3 refers to the bubble function. The weak form of the homogeneous ($s = 0$) problem is given by

$$\int_0^L \left(w^h\, a\, u_x^h + w_x^h\, \nu\, u_x^h\right) dx = 0. \tag{2.71}$$

By setting $w^h = w_B^h$, see (2.70b), one finds

$$\int_{-1}^{+1} \left[a(1-\xi^2) - 4\frac{\nu}{h}\xi\right] \left[\frac{u_2 - u_1}{2} - 2\xi u_3\right] d\xi = 0,$$

from which we extract the value of parameter u_3 associated with the bubble function in terms of the element nodal parameters:

$$u_3 = -\frac{ah}{8\nu}(u_2 - u_1) = -\frac{Pe}{4}(u_2 - u_1).$$

This allows us to rewrite (2.70a) in terms of the nodal values as

$$\begin{aligned} u^h &= \frac{1}{2}(1-\xi)\Big(1 + \frac{Pe}{2}(1+\xi)\Big)\,u_1 + \frac{1}{2}(1+\xi)\Big(1 - \frac{Pe}{2}(1-\xi)\Big)\,u_2 \\ &= P_1(\xi)\,u_1 + P_2(\xi)\,u_2. \end{aligned}$$

Introducing this expression into the weak form (2.71) and now imposing $w^h = w_G^h$, one finds that the bubble function method delivers the following element matrices:

$$\mathbf{C}^e = a \int_0^h \begin{pmatrix} N_1 \dfrac{\partial P_1}{\partial x} & N_1 \dfrac{\partial P_2}{\partial x} \\ N_2 \dfrac{\partial P_1}{\partial x} & N_2 \dfrac{\partial P_2}{\partial x} \end{pmatrix} dx = \frac{a}{2} \begin{pmatrix} -1 + \dfrac{Pe}{3} & 1 - \dfrac{Pe}{3} \\ -1 - \dfrac{Pe}{3} & 1 + \dfrac{Pe}{3} \end{pmatrix}$$

$$\mathbf{K}^e = \nu \int_0^h \begin{pmatrix} \dfrac{\partial N_1}{\partial x} \dfrac{\partial P_1}{\partial x} & \dfrac{\partial N_1}{\partial x} \dfrac{\partial P_2}{\partial x} \\ \dfrac{\partial N_2}{\partial x} \dfrac{\partial P_1}{\partial x} & \dfrac{\partial N_2}{\partial x} \dfrac{\partial P_2}{\partial x} \end{pmatrix} dx = \frac{\nu}{h} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}.$$

Comparing these element matrices with those derived in Section 2.2 for the Galerkin formulation, one notes that the bubble function method only modifies the convection matrix $\mathbf{C}^e$ of the linear element without affecting the diffusion matrix $\mathbf{K}^e$. With these results, and assembling in the usual finite element manner the contributions emanating from both elements to which a given node belongs, one finds that the bubble function method delivers the following discrete equation at an interior node j of a mesh of uniform linear elements:

$$a \frac{u_{j+1} - u_{j-1}}{2h} - \left(\nu + \frac{Pe}{3} \frac{ah}{2} \right) \frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} = 0.$$

This shows that bubbles have a stabilizing effect for convection–diffusion problems similar to that of the added diffusion method. In fact, the added diffusion in the previous equation corresponds to that given in (2.26) with the coefficient β given by the doubly asymptotic approximation in (2.29).

2.6.2 One-dimensional transport

This example is concerned with the solution of the 1D homogeneous convection–diffusion equation with prescribed Dirichlet conditions at both ends of the domain $[0, 1]$, namely $u(0) = 0$ and $u(1) = 1$. The problem is first solved using a mesh of 10 uniform linear elements and then repeated with five uniform quadratic elements. A unit convection velocity is assumed and three values of the mesh Péclet number are considered: $Pe = 0.25, 0.9$ and 5.0. Figure 2.15 reports the results obtained with linear elements using the Galerkin method, a full upwind approach, and the SUPG method. The results obtained with quadratic elements are reported in Figure 2.16. For small to moderate Pe, the exact solution varies rather smoothly over the entire domain. However, as Pe is increased beyond unity, the solution has a boundary layer, which cannot be resolved by the standard Galerkin method. This is in contrast with the stable and accurate results obtained with SUPG. As anticipated, the full upwind approximation delivers stable, but inaccurate, results due to an excess of numerically added dissipation.

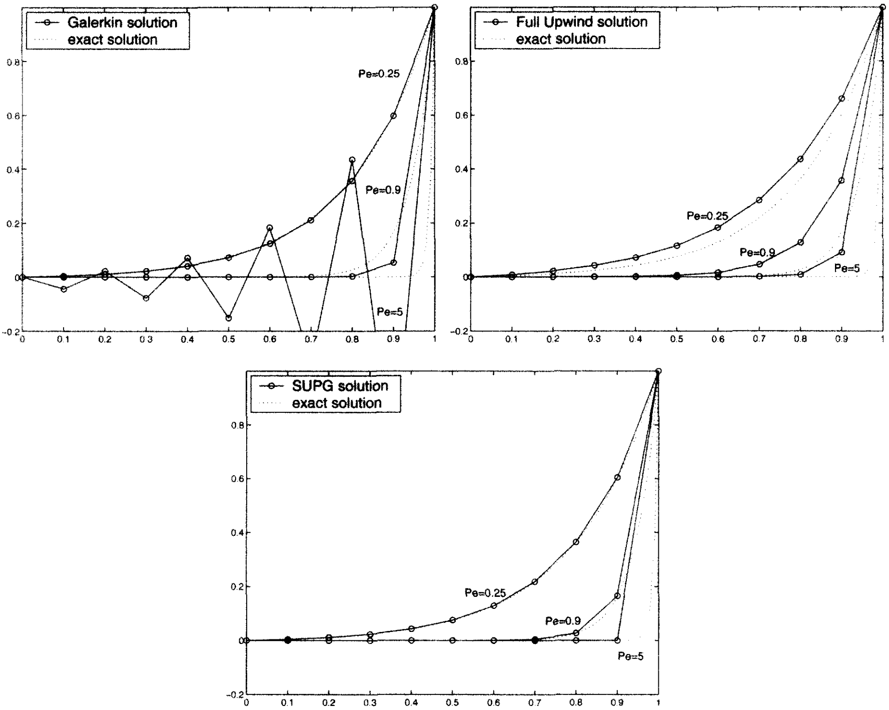


Fig. 2.15 Galerkin (top-left), full upwind (top-right) and SUPG (bottom) solutions (solid lines) for linear elements. Dotted lines show the exact solution for $Pe = 0.25$, 0.9 and 5.

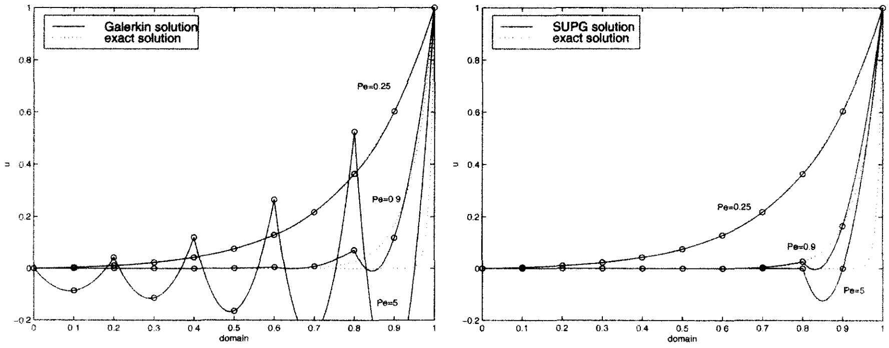


Fig. 2.16 Galerkin (left) and SUPG (right) solutions (solid lines) for quadratic elements. Dotted lines show the exact solution for $Pe = 0.25$, 0.9 and 5.

2.6.3 Convection–diffusion across a source term

The previous example is further studied introducing a source term:

$$\begin{cases} a\, u_x - \nu\, u_{xx} = 10e^{-5x} - 4e^{-x} & \text{for } x \in \,]0,1[\\ u(0) = 0 \text{ and } u(1) = 1. \end{cases}$$

Again, a unit convection velocity is assumed and the Péclet number is chosen as 0.25 or 5. Galerkin, SU (the non-consistent formulation) and SUPG are compared in Figure 2.17 for linear and quadratic elements. GLS and SGS results are not shown because they are almost identical to those of SUPG.

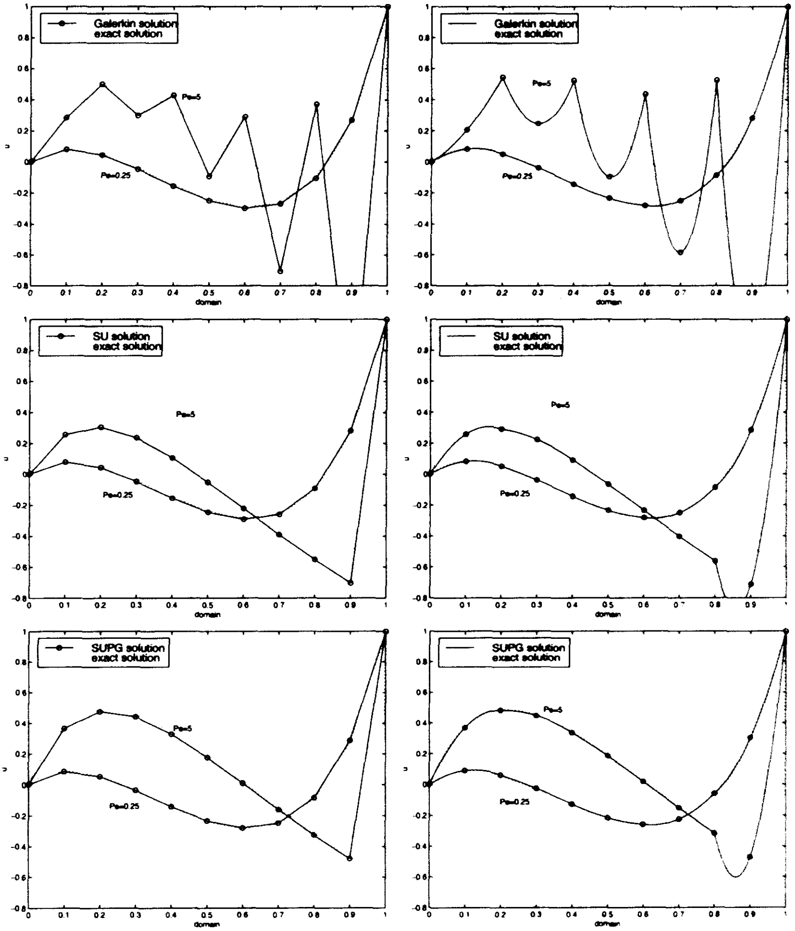


Fig. 2.17 Galerkin (top), SU (center) and SUPG (bottom) solutions (solid lines) for linear (left) and quadratic (right) elements. Dotted lines show the exact solution for *Pe* = 0.25 and 5.

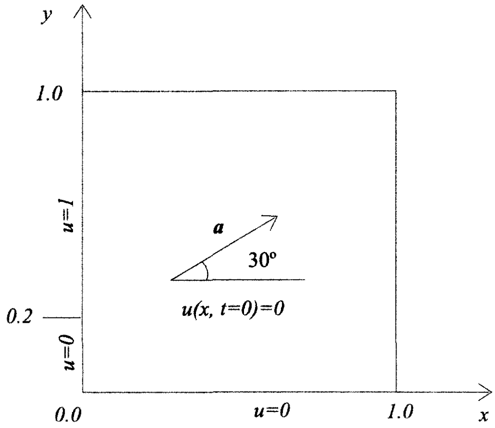


Fig. 2.18 Convection of discontinuous inlet data skew to the mesh: problem statement.

2.6.4 Convection–diffusion skew to the mesh

One of the major contributions in the stabilization of steady convection–diffusion problems is the concept of *anisotropic balancing diffusion* or *streamline upwind test functions* discussed in Section 2.3.3.3. The equivalence between added diffusion and upwinding is simple in 1D but its blunt extension to multiple dimensions leads to excessive crosswind diffusion. This is observed in the present example. Galerkin and SUPG formulations are compared with an artificial diffusion method; that is, adding a scalar artificial diffusivity defined by equation (2.59), namely $\bar{\nu} = \tau \, ||\boldsymbol{a}||^2$, with τ determined by any of the definitions in Section 2.4.3. Note that the present artificial diffusion method is based on a scalar diffusivity. Thus, it should not be confused with the SU method which possesses the tensorial structure indicated in (2.48) and is further illustrated in Remark 2.9.

The problem statement is depicted in Figure 2.18 where the unit square is taken as the computational domain, $\bar{\Omega} = [0, 1] \times [0, 1]$. This 2D test case has been widely used to illustrate the effectiveness of stabilized finite element methods in the modeling of convection-dominated flows. A mesh of 10 by 10 equal-sized bilinear elements is considered.

The flow is unidirectional and constant, $||\boldsymbol{a}|| = 1$, but the convective velocity is skew to the mesh with an angle of 30°. The diffusivity coefficient is taken to be 10^{-4}, corresponding to a mesh Péclet number of 10^4. The inlet boundary data are discontinuous and two types of boundary conditions are considered at the outlet:

- *Downwind homogeneous natural boundary conditions.* The results for this case are displayed in Figure 2.19. Given the elevated value of the Péclet number, the solution is practically one of pure convection. The Galerkin method is not able to satisfactorily resolve the discontinuity and produces spurious oscillations. The artificial diffusion method and the SUPG method yield better results, but SUPG introduces less crosswind diffusion.

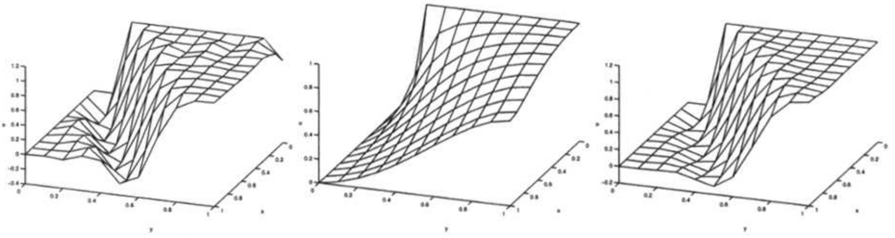


Fig. 2.19 Galerkin (left), artificial diffusion (center) and SUPG (right) solutions for the 2D convection–diffusion problem with downwind natural conditions.

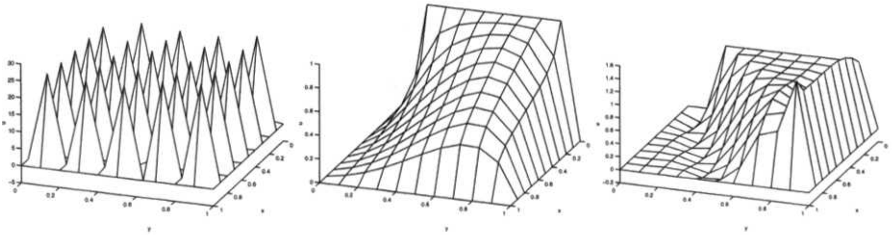


Fig. 2.20 Galerkin (left), artificial diffusion (center) and SUPG (right) solutions for the 2D convection–diffusion problem with downwind essential boundary conditions.

- *Downwind homogeneous essential boundary conditions.* Here, we impose $u = 0$ on the outlet portion of the boundary. The solution now involves a thin boundary layer at the outlet. As shown in Figure 2.20, the crude 10 by 10 mesh has difficulty capturing the details of the solution. The Galerkin results are wildly oscillatory and bear no resemblance to the exact results. Better results are obtained with the stabilized formulations. The artificial diffusion technique introduces excessive numerical diffusion.

2.6.5 Convection–diffusion–reaction in 2D

On the same unit square domain as in the previous example, $\bar{\Omega} = [0, 1] \times [0, 1]$, the influence of the convection and reaction terms are compared for the Galerkin, SUPG, GLS and SGS formulations. As in the previous example, convection is taken skew to the mesh with an angle of 30° and the diffusion is small, $\nu = 10^{-4}$. Two cases are studied: first the convection–reaction-dominated case, $||\boldsymbol{a}|| = 1/2$ and $\sigma = 1$, and then the reaction-dominated case, $||\boldsymbol{a}|| = 10^{-3}$ and $\sigma = 1$. The results are shown in Figures 2.21 and 2.22. The stabilized methods produce similar results. Nevertheless, as noted in Remark 2.12 and Section 2.5.3 in the presence of reaction (in particular for the reaction-dominated example, see Figure 2.22) SGS reduces the oscillations near the boundary layers.

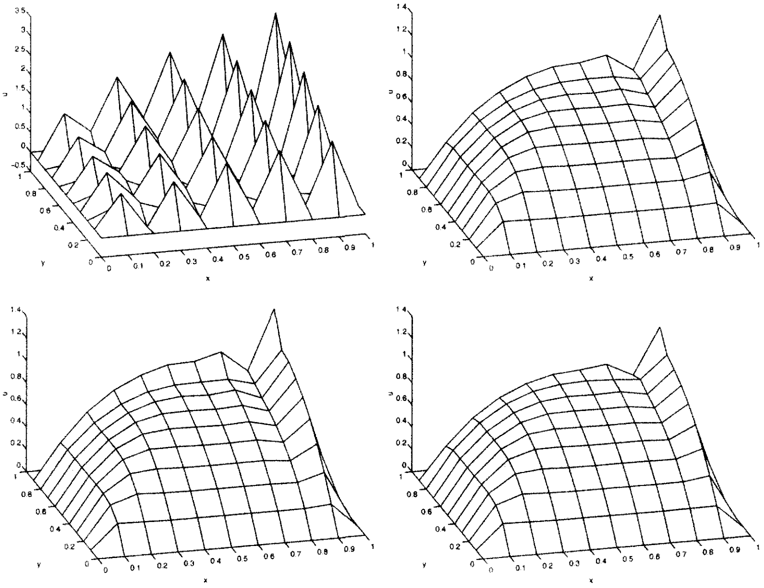


Fig. 2.21 Galerkin, SUPG, GLS and SGS (from left to right and top to bottom) solutions for the 2D convection–reaction-dominated problem.

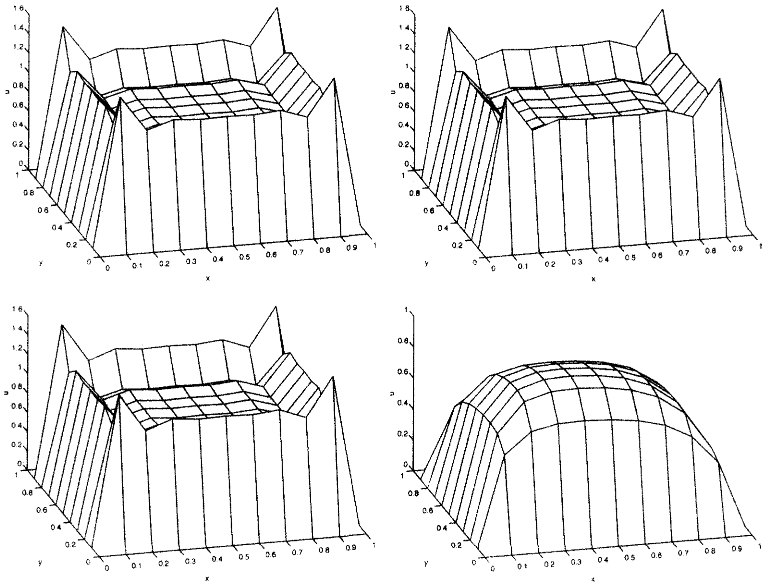


Fig. 2.22 Galerkin, SUPG, GLS and SGS (from left to right and top to bottom) solutions for the 2D reaction-dominated problem.

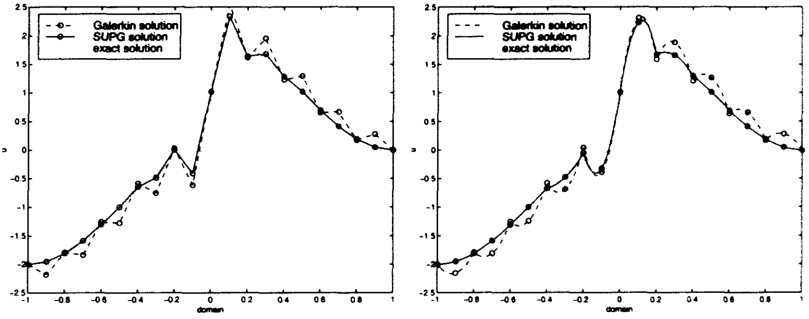


Fig. 2.23 Galerkin and SUPG solutions of Hemker problem for linear (left) and quadratic (right) elements.

2.6.6 The Hemker problem

The so-called Hemker problem is studied, which consists of solving the convection–diffusion equation

$$x\frac{\partial u}{\partial x} + \nu\frac{\partial^2 u}{\partial x^2} = -\nu\pi^2\cos(\pi x) - \pi x\sin(\pi x) \qquad \text{on } [-1, 1],$$

with space-dependent convection velocity and source term. The boundary conditions are specified as

$$u(-1) = -2; \qquad u(1) = 0.$$

The exact solution is

$$u(x) = \cos(\pi x) + \text{erf}(x/\sqrt{(}2\nu))/\text{erf}(1/\sqrt{(}2\nu))$$

and has a turning point in the middle of the domain.

We take $\nu = 10^{-10}$ and use a mesh of 20 uniform linear elements ($h = 0.1$). The problem is first solved in the Galerkin formulation. The numerical solution is depicted in Figure 2.23 in comparison with the exact solution. Node-to-node oscillations extending over the whole computational domain characterize the Galerkin approach. The solution is then repeated using the SUPG formulation. Since the convection velocity $a(x) = x$ is negative on $]-1, 0[$ and positive on $]0, 1[$, care must be taken to add a positive diffusion everywhere in the mesh. The SUPG results are displayed in Figure 2.23. Though the SUPG method succeeds in removing the oscillations in the smooth part of the solution, residual oscillations remain near the turning point where a near-discontinuity is present. Additional nonlinear viscosity should be added locally to suppress these residual oscillations. This aspect will be treated in detail in Chapter 4.

3

Unsteady convective transport

This chapter addresses the development of time-accurate methods for solving transient convection problems using finite elements. First, a brief review is presented of the method of characteristics and of its use in the finite element context. The chapter then proceeds with the presentation and analysis of classical, first- and second-order accurate, time-stepping algorithms. This is followed by the description of higher-order methods which indirectly account for the role of the characteristics in convection problems. The last part of the chapter considers the coupling of a finite-element-based spatial discretization and high-order accurate time-stepping algorithms. Emphasis is placed on the use of least-squares-type finite elements to ensure stability when non-dissipative schemes are used for marching in time.

3.1 INTRODUCTION

Time-dependent problems describing convective transport are governed by hyperbolic equations and the characteristics play a dominant role in their solution. This has important consequences as regards the development of accurate numerical methods for solving evolutionary problems in this class. So, space and time being linked through the characteristics, the discretization of one certainly has an influence on the discretization of the other. In fact, an accurate spatial representation can be quickly eroded when it is transported in time if the time integration algorithm is not able to propagate the information along the directions prescribed by the convection problem.

In principle, one could circumvent the issue of convecting the information along the streamlines by resorting to a Lagrangian formulation. In such a moving coordinate

Finite Element Methods for Flow Problems. Jean Donea and Antonio Huerta
Copyright © 2003 John Wiley & Sons, Ltd.
ISBN: 0-471-49666-9

system, convective terms disappear from the governing equations. Unfortunately, a purely Lagrangian description is in general not practicable in fluid flow problems because of excessive distortions of the computational mesh. However, a number of numerical methods based on the concept of convecting the information along the streamlines have been developed and applied in the finite element modeling of convective transport using a fixed (Eulerian) element mesh. The chapter is organized as follows.

In Section 3.2, we define the strong form of the initial boundary value problem for convective transport. Then, Section 3.3 illustrates the role of the characteristics in the solution of these problems. Several methods exploiting the characteristics are described. This includes semi-Lagrangian, Lagrange–Galerkin and characteristic-based methods. Their common feature is to exploit the fact that the solution is constant along a particle path or a characteristic. Unfortunately, finite element methods exclusively based on the characteristics are rather difficult to implement and expensive to use. This is the reason why direct time integration of the convection equations is often preferred. However, it must be based on suitable time-stepping schemes. In Section 3.4, we review the properties of second-order time-stepping algorithms classically used to produce the transient response of unsteady problems. Their accuracy analysis in Section 3.5 reveals that such algorithms are not optimal for convection problems. This is because second-order methods are unable to take into account the directional character of propagation of information in convective transport. In particular, a rapid fall-off in accuracy occurs as the time step is increased. Moreover, in explicit finite element schemes the stability range is reduced with respect to the corresponding finite difference schemes. Attention is therefore focused on higher-order time-stepping schemes capable of indirectly taking into account the propagation of information along the characteristics.

One of the indirect methods which exploit the characteristics in the numerical solution of convective transport problems, the Taylor–Galerkin method, is described in Section 3.6. It represents an attempt to simulate by a Taylor series in time the fact that the solution of convective transport problems is constant along the characteristics. This Taylor series is usually extended to third or fourth order. The discussion of time-stepping algorithms closes with an introductory presentation in Section 3.7 of monotonicity-preserving schemes. Such schemes are used near sharp solution gradients to suppress the residual oscillations that linear stabilization techniques cannot remove.

We then address the problem of coupling highly accurate time-stepping algorithms and finite element spatial discretization techniques based on the classical C^0-continuous spatial representation. In Chapter 2 we have shown that the Galerkin finite element method is not optimal to model steady transport problems in which convective effects are dominant. Stabilization techniques capable of introducing sufficient numerical dissipation are needed to remedy the lack of numerical stability of the Galerkin formulation. In the present context of transient problems, the required amount of numerical damping can be provided either by the use of dissipative time-stepping methods or, as in Chapter 2, by the finite element spatial discretization. In Section 3.8 least-squares-based finite elements are introduced to provide the numerical dissipation needed to achieve stable results with non-dissipative time-stepping algorithms. To conclude the discussion of discretization procedures for convection problems, we briefly introduce in Section 3.9 the discontinuous Galerkin method and then discuss space–time formulations in Section 3.10. Finally, representative test problems in 1D and 2D are solved in Section 3.11 to illustrate the use of time-accurate finite element methods in the numerical solution of convective transport problems.

3.2 PROBLEM STATEMENT

Before discussing numerical algorithms, we shall introduce the strong form of the linear convection problems treated in this chapter. In the so-called *conservation law form*, the equations governing unsteady convective transport are

$$u_t + \boldsymbol{\nabla} \cdot \boldsymbol{f}(u) = s(\boldsymbol{x}, t) \qquad \text{in } \Omega \times]0, T[, \tag{3.1a}$$

$$u(\boldsymbol{x}, 0) = u_0(\boldsymbol{x}) \qquad \text{on } \Omega \text{ at } t = 0, \tag{3.1b}$$

$$u = u_D \qquad \text{on } \Gamma_D^{in} \times]0, T[, \tag{3.1c}$$

$$-\boldsymbol{f} \cdot \boldsymbol{n} = h \qquad \text{on } \Gamma_N^{in} \times]0, T[, \tag{3.1d}$$

where now the solution u and the source term s are a function of time. Note that in the hyperbolic case we cannot specify boundary conditions on the outflow portion of the boundary $\Gamma = \partial\Omega$ of the domain Ω. As in Chapter 2, Γ^{in} denotes the inflow portion of the boundary (see Remark 2.1). We employ the partition $\Gamma^{in} = \Gamma_D^{in} \cup \Gamma_N^{in}$ to identify the Dirichlet and the Neumann portions of the inflow boundary.

The value of u is specified on the Dirichlet portion of the inflow boundary, while the inlet normal flux is prescribed on the Neumann portion. The vector-valued flux function $\boldsymbol{f}(u)$ can be defined in terms of the convection velocity $\boldsymbol{a}$ by

$$\boldsymbol{f}(u) = \boldsymbol{a}\, u. \tag{3.2}$$

If the convection velocity $\boldsymbol{a}$ is independent of u, the problem is linear. Nonlinear problems are studied in Chapter 4.

When the convection velocity $\boldsymbol{a}$ is divergence free, use can be made of the identity

$$\boldsymbol{\nabla} \cdot (\boldsymbol{a}\, u) = u\, \boldsymbol{\nabla} \cdot \boldsymbol{a} + \boldsymbol{a} \cdot \boldsymbol{\nabla} u \tag{3.3}$$

to recast problem (3.1) in the *convective* (also termed *advective*) form:

$$u_t + (\boldsymbol{a} \cdot \boldsymbol{\nabla}) u = s \qquad \text{in } \Omega \times]0, T[, \tag{3.4a}$$

$$u(\boldsymbol{x}, 0) = u_0(\boldsymbol{x}) \qquad \text{on } \Omega \text{ at } t = 0, \tag{3.4b}$$

$$u = u_D \qquad \text{on } \Gamma_D^{in} \times]0, T[\tag{3.4c}$$

$$-\boldsymbol{a} u \cdot \boldsymbol{n} = h \qquad \text{on } \Gamma_N^{in} \times]0, T[,. \tag{3.4d}$$

Equations (3.1a) and (3.4a) describe the convective transport of a quantity u whose speed of propagation is given by the vector function $\boldsymbol{a}(\boldsymbol{x}, t)$.

The numerical solution of the above initial boundary value problems clearly involves a double discretization process, namely the spatial and temporal discretizations. While the former will be performed here by the finite element method, two broad classes of methods will be employed to trace the temporal evolution of the solution of convection problems. The first class is based on the characteristics and exploits them to transport the information in time. The second class of methods is based on a standard coordinate system and makes use of time-stepping algorithms to advance the solution from one time level to the next until the final time for the problem is reached.

3.3 THE METHOD OF CHARACTERISTICS

In the present section, we recall the basic properties of convection equations with an emphasis on the role of the characteristics in their solution. We then review finite-element-based algorithms capable of producing time-accurate solutions of convective transport problems through a direct use of the characteristics. See also the introductory paper by Donea and Quartapelle (1992).

3.3.1 The concept of characteristic lines

For first-order equations, such as the convection equations studied in this chapter, the prototype linear partial differential equation (PDE) is of the form

$$u_t + a\,u_x = s, \tag{3.5}$$

where a and s may depend on x and t. We may interpret this equation as follows: for a given solution $u(x,t)$ and at a given space–time point (x,t), the total derivative of u with respect to time in the direction defined by the slope

$$dx/dt = a$$

is equal to s. In this way, we have introduced the concept of propagation of the information u, with a speed a, as a function of the considered space–time point (x,t). For linear PDEs with constant coefficients the speed of propagation is constant.

When the PDE involves the physical concept of propagation it is said to be *hyperbolic* and the direction $dx/dt = a$ is termed the *characteristic direction*. We thus have the following result: *any first-order PDE is hyperbolic*. If the PDE is linear, the characteristic curves are fixed in the (x,t) plane, independent of the solution $u(x,t)$. Furthermore, if the linear PDE has constant coefficients, the characteristics are straight lines. Nonlinear first-order PDEs in which the speed a depends on the solution u are studied in Chapter 4. To further illustrate the concept of transport typical of first-order PDEs, let us consider the homogeneous form of equation (3.5) with a constant. The change of variables

$$\xi = x - at, \qquad \eta = x + at$$

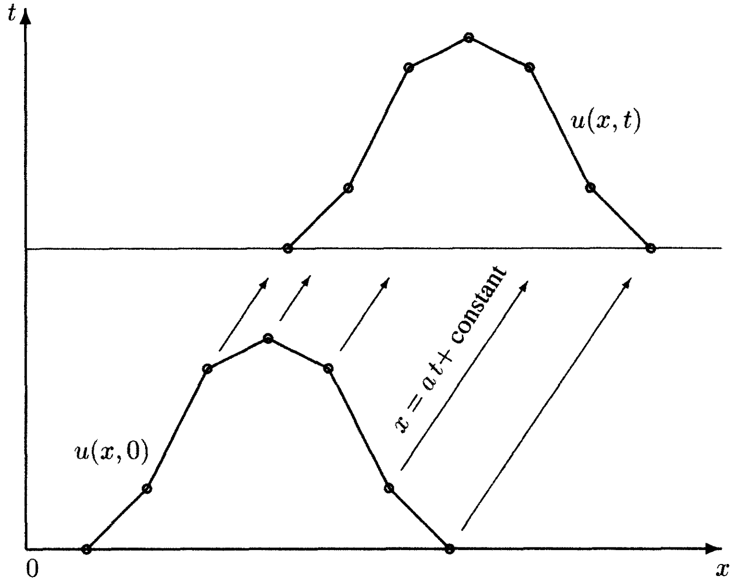


Fig. 3.1 The concept of characteristics: $du/dt = 0$ and consequently u is constant along the lines $x = a\,t +$ constant.

induces the transformation

$$\begin{Bmatrix} u_t \\ u_x \end{Bmatrix} = \begin{pmatrix} \dfrac{\partial\xi}{\partial t} & \dfrac{\partial\eta}{\partial t} \\ \dfrac{\partial\xi}{\partial x} & \dfrac{\partial\eta}{\partial x} \end{pmatrix} \begin{Bmatrix} u_\xi \\ u_\eta \end{Bmatrix} = \begin{pmatrix} -a & a \\ 1 & 1 \end{pmatrix} \begin{Bmatrix} u_\xi \\ u_\eta \end{Bmatrix}.$$

Thus, the PDE $u_t + au_x = 0$ becomes

$$2a\,u_\eta = 0,$$

whose general solution is

$$u = f(\xi) = f(x - at),$$

where f is an arbitrary function. It follows that the solution $u(x,t)$ at point x and time t is equal to the solution at time $t - \Delta t$ at the point $x - a\Delta t$:

$$u(x,t) = u(x - a\Delta t, t - \Delta t). \tag{3.6}$$

In other words, the solution is a rigid transport in time of the spatial profile of u. For this reason, the name *transport equation* is given to equation (3.5). The concept of transport along the characteristics is illustrated in Figure 3.1 which shows how an initial distribution $u(x,0) = u_0(x)$ of the unknown undergoes a uniform translation along the x coordinate axis as time goes on.

3.3.2 Properties of the linear convection equation

To provide a basis for the discussion of characteristic-based finite element algorithms, we recall in this section the basic mathematical properties of the unsteady convection equation. For this purpose, we consider first the case of a scalar quantity u transported by a prescribed convection velocity field $\boldsymbol{a}(\boldsymbol{x},t)$ in the presence of a known source term $s(\boldsymbol{x},t)$. The initial boundary value problem for the quantity u is assumed linear and defined as in (3.4).

As seen previously, the analytical solution of this linear initial boundary value problem at a given point in space and time can be determined with the aid of the concept of characteristic lines associated with the velocity field $\boldsymbol{a}(\boldsymbol{x},t)$. The idea is to replace the operator $\partial/\partial t + \boldsymbol{a}\cdot\boldsymbol{\nabla}$ in the l.h.s. of equation (3.4a), which represents the material derivative in the Eulerian description, by a simple time derivative using the Lagrangian viewpoint.

Accordingly, for a given space–time point $(\boldsymbol{x},\tau)$, where $\boldsymbol{x}\in\Omega$ and $\tau\in\,]0,T[$, one determines the characteristic line $\boldsymbol{X}=\boldsymbol{X}(\boldsymbol{x},\tau;t)$ passing through the considered space–time point by solving the ordinary (vector) differential equation

$$\frac{d\boldsymbol{X}}{dt}(\boldsymbol{x},\tau;t)=\boldsymbol{a}(\boldsymbol{X}(\boldsymbol{x},\tau;t)), \qquad t\in\,]0,T[, \tag{3.7a}$$

subject to the condition

$$\boldsymbol{X}(\boldsymbol{x},\tau;\tau)=\boldsymbol{x}. \tag{3.7b}$$

With reference to the Lagrangian point of view, $\boldsymbol{X}(\boldsymbol{x},\tau;t)$ can be interpreted as providing the position at time t of a fluid particle transported by the convection velocity field $\boldsymbol{a}$, which occupies the spatial position $\boldsymbol{x}$ at time τ. In other words, (3.7a) defines the particle trajectory. Along this trajectory, the *material (or total) derivative*

$$\frac{du}{dt}=\frac{\partial u}{\partial t}+\boldsymbol{a}\cdot\boldsymbol{\nabla}u,$$

which is the l.h.s. of the convection equation (3.4a), reduces to a simple time derivative. In fact, by definition, the material derivative is the time rate of change felt by an observer moving with the material particles.

It is worth noting that equation (3.7a) is generally nonlinear, except when the convection field has a specific spatial dependence of the form $\boldsymbol{a}(\boldsymbol{x},t)=\boldsymbol{\alpha}(t)+\beta(t)\,\boldsymbol{x}$, where $\boldsymbol{\alpha}(t)$ and $\beta(t)$ are arbitrary functions. Excluding such a situation, the presence of the nonlinearity is the price to pay for the reduction of the linear unsteady convection equation (3.4a) to an ordinary differential equation.

The integration of equation (3.7a) proceeds from $t=\tau$ backwards, until the characteristic curve either intersects the boundary Γ_{in} or reaches the initial time $t=0$, see Pironneau (1989) and Douglas and Russell (1982) for details:

- The intersection of the characteristic with Γ_{in} occurs at a time $t_\Gamma(\boldsymbol{x},\tau)$ that depends on the considered characteristic curve. If the intersection point is denoted by $\boldsymbol{X}_\Gamma=\boldsymbol{X}_\Gamma(\boldsymbol{x},\tau)$, one has

$$\boldsymbol{X}_\Gamma(\boldsymbol{x},\tau)=\boldsymbol{X}(\boldsymbol{x},\tau;t_\Gamma).$$

It follows that the solution at the considered space–time position $(\boldsymbol{x}, \tau)$ depends on the value $u_D(\boldsymbol{X}_\Gamma, t_\Gamma)$ prescribed at the boundary point $\boldsymbol{X}_\Gamma$. To determine $u(\boldsymbol{x}, \tau)$, the governing equation (3.4a) is written in the so-called characteristic form. We shall denote the value of the unknown along the characteristic curve $\boldsymbol{X}(\boldsymbol{x}, \tau; t)$ by $U(t) := u(\boldsymbol{X}(\boldsymbol{x}, \tau; t), t)$, or, omitting the explicit indication of the dependence on $(\boldsymbol{x}, \tau)$ to simplify the notation, by $U(t) = u(\boldsymbol{X}(t), t)$. In characteristic form equation (3.4a) along $\boldsymbol{X}(t)$ reduces to the following linear ordinary differential equation:

$$\frac{dU}{dt} = S(t), \tag{3.8}$$

where $S(t) = s(\boldsymbol{X}(t), t)$. Equation (3.8) must then be integrated up to $t = \tau$ subject to the *initial condition*

$$U(t_\Gamma) = u_D(\boldsymbol{X}_\Gamma, t_\Gamma)$$

at the intersection time with Γ_{in}. Thus, for characteristic lines intersecting Γ^{in} the solution reads

$$u(\boldsymbol{x}, \tau) = U(\tau) = u_D(\boldsymbol{X}_\Gamma, t_\Gamma) + \int_{t_\Gamma}^{\tau} S(t) dt. \tag{3.9}$$

- On the contrary, when the characteristic curve reaches the plane $t = 0$, one has $\boldsymbol{X}(\boldsymbol{x}, \tau; 0) \in \Omega$. Then, the solution $u(\boldsymbol{x}, t)$ is determined uniquely by integrating the characteristic equation (3.8) with the initial condition $U(0) = u_0(\boldsymbol{X}(0))$ stemming from the original initial condition (3.1b). This gives

$$u(\boldsymbol{x}, \tau) = U(\tau) = u_0(\boldsymbol{X}(0)) + \int_0^{\tau} S(t)\, dt. \tag{3.10}$$

Remark 3.1. In this chapter we shall assume differentiability of $u(x, t)$. However, if the characteristics transport the initial condition, one could easily construct a "solution" transporting initial non-smooth data. In fact, any singularity in the initial data is transported along the characteristics. This is a fundamental property of linear hyperbolic equations. The concept of non-smooth solutions of hyperbolic equations, which is intimately related to the concept of *weak solutions*, will be discussed in the next chapter. Nonlinear hyperbolic equations may have non-smooth solutions even if initial data are smooth.

Remark 3.2 (Equation in conservation form). The analysis just described also applies when the linear hyperbolic equation is formulated in the conservation form (3.1a), that is

$$u_t + \boldsymbol{\nabla} \cdot (\boldsymbol{a}\, u) = s, \tag{3.11}$$

where the velocity field $\boldsymbol{a}(\boldsymbol{x}, t)$ is still assumed to be a known quantity. In view of the identity (3.3), and introducing the quantities

$$b(\boldsymbol{x}, t) = \boldsymbol{\nabla} \cdot \boldsymbol{a}(\boldsymbol{x}, t) \quad \text{and} \quad B(t) = b(\boldsymbol{X}(t), t),$$

equation (3.11) becomes along the characteristic $\boldsymbol{X}(t)$

$$\frac{dU}{dt} + B(t)\,U = S(t), \tag{3.12}$$

where again $S(t) = s(\boldsymbol{X}(t), t)$. Due to the additional variable coefficient term, $B(t)$, the solution for the case where the characteristic line crosses the domain Ω at $t = 0$ is

$$u(\boldsymbol{x}, \tau) = e^{-\int_0^\tau B(t)\,dt}\left[u_0(\boldsymbol{X}(0)) + \int_0^\tau S(t')\,e^{\int_0^{t'} B(t)\,dt}\,dt'\right]. \tag{3.13}$$

A similar expression is obtained when the characteristic intersects the boundary Γ_{in}, see Pironneau (1989, pp. 75–76).

Remark 3.3 (System of hyperbolic equations). In the case of a system of coupled hyperbolic equations, such as the Euler equations of gas dynamics to be discussed in Chapter 4, the problem cannot be reduced to a standard ordinary differential problem along the characteristics, unless the coefficients of the equations are constants. It is actually still possible to introduce characteristic curves passing through a given space–time point $(\boldsymbol{x}, \tau)$, the number of such characteristics being in general equal to the number of components in the governing system. However, the original system of coupled equations written along these characteristic curves takes the form of an ordinary differential system of a rather peculiar nature. Its solution is in fact defined by a system of integral equations in which each component of the vector unknown involves integration along a different curve, see John (1991, pp. 46–52) for a detailed description of the 1D case.

Remark 3.4 (Nonlinear convection equation). Let us briefly extend the above concepts to the case of a nonlinear convection equation of the form

$$u_t + \boldsymbol{a}(u) \cdot \boldsymbol{\nabla} u = s, \tag{3.14a}$$

where $\boldsymbol{a}(u) = \partial \boldsymbol{f}(u)/\partial u$. This is the quasi-linear form of a nonlinear scalar conservation law. Here, the source term s may also depend on the unknown. If the solution remains smooth and the equation holds in the classical sense, the characteristic curve $\boldsymbol{X}(t)$ reaching the space–time point $(\boldsymbol{x}, \tau)$ satisfies the following ordinary differential equation with final condition:

$$\frac{d\boldsymbol{X}}{dt} = \boldsymbol{a}(u(\boldsymbol{X}(t), t)), \qquad \boldsymbol{X}(\tau) = \boldsymbol{x}. \tag{3.14b}$$

Due to the dependence of the convection field on the unknown u, both equations (3.14) must be solved in a coupled manner. It is not possible to solve equation (3.14b) first in order to obtain the characteristic curves separately from the solution of the convection equation (3.14a) itself.

Assume that the foot of the considered characteristic lies inside Ω at $t = 0$, so that the initial condition for u has to be imposed. If one introduces the unknown $U(t) := u(\boldsymbol{X}(t), t)$, with $0 \leq t \leq \tau$, the equation and condition (3.14b) become

$$\frac{d\boldsymbol{X}}{dt} = \boldsymbol{a}(U(t)), \qquad \boldsymbol{X}(\tau) = \boldsymbol{x}, \tag{3.15a}$$

whereas the nonlinear convection equation (3.14a) expressed in characteristic form and the associated initial condition take the form

$$\frac{dU}{dt} = s(U(\boldsymbol{X}(t), t); t), \qquad U(0) = u_0(\boldsymbol{X}(0)). \tag{3.15b}$$

Equations (3.15) represent a coupled system of first-order ordinary differential equations to be solved over the interval $0 \leq t \leq \tau$ and supplemented by conditions at both ends of the integration interval. System (3.15) therefore represents a two-point boundary value problem for a first-order system with separated end-conditions. For solving such a problem, use can be made of the numerical technique proposed by Quartapelle and Rebay (1990) in combination with Newton iterations. The initial condition consists of a homogeneous relationship between the initial values of U and $\boldsymbol{X}$ in the form

$$U(0) \equiv u_0(\boldsymbol{X}(0)),$$

which is a nonlinear equation with respect to the unknown $\boldsymbol{X}(0)$ whenever $u_0(\boldsymbol{x})$ is a nonlinear function of its argument. Summarizing, the nonlinearity of the convection equation does not preclude a complete reduction of the partial differential problem to an ordinary differential one. However, the nonlinear convection equation introduces two extra difficulties:

1. The equation defining the characteristic curve and the governing equation in characteristic form, which are in general both nonlinear, must be solved as a coupled set of equations;

2. The initial condition is a coupled and nonlinear condition for the unknowns $(\boldsymbol{X}, U)$ of the system.

3.3.3 Methods based on the characteristics

In this section we briefly present finite element methods exploiting the characteristics for solving the convection equation (3.4a) with a space–time variable convection field $\boldsymbol{a}(\boldsymbol{x}, t)$. A more complete review of characteristic-based finite element methods can be found in a paper by Donea and Quartapelle (1992). The reader interested in a detailed exposition of such methods should consult the books by Pironneau (1989) and Morton (1996).

3.3.3.1 Semi-Lagrangian method A popular method of characteristic type specifically adapted to treat unsteady convection problems is the semi-Lagrangian method introduced by Purnell (1976), Robert (1981; 1982), Pudykiewicz and Staniforth (1984) and Staniforth and Pudykiewicz (1985).

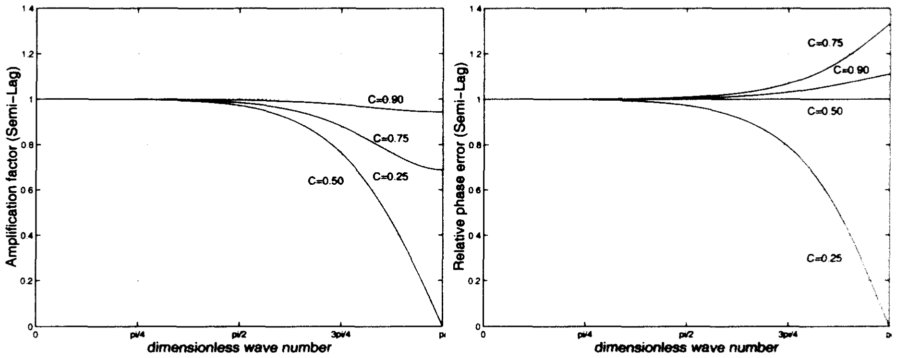


Fig. 3.2 Accuracy properties in pure convection of the semi-Lagrangian scheme with cubic interpolation: amplification factor modulus $|G|$ (left) and relative phase error $\phi_{numer}/\phi_{exact}$ (right). $C = \|\boldsymbol{a}\|\Delta t/h$ is the Courant number, see Section 3.5.1

With reference to the linear convection equation (3.4a), let $\boldsymbol{x}$ be a mesh point and $u^n(\boldsymbol{x})$ the finite element solution obtained at time $t^n = n\Delta t$, Δt being the time increment. The solution $u^{n+1}(\boldsymbol{x})$ at the next time level $t^{n+1} = t^n + \Delta t$ is obtained by first approximating equation (3.7a) defining the characteristic over the interval $]t^n, t^{n+1}[$ with second-order accuracy using the mid-point rule. This gives

$$\Delta\boldsymbol{X}(\boldsymbol{x}) = \boldsymbol{a}\Big(\boldsymbol{x} - \frac{1}{2}\Delta\boldsymbol{X},\ t^{n+1/2}\Big)\,\Delta t.$$

This nonlinear equation for the displacement vector $\Delta\boldsymbol{X}$ can be solved iteratively using the recursive scheme

$$\Delta\boldsymbol{X}_{(k+1)}(\boldsymbol{x}) = \boldsymbol{a}\Big(\boldsymbol{x} - \frac{1}{2}\Delta\boldsymbol{X}_{(k)},\ t^{n+1/2}\Big)\,\Delta t \tag{3.16}$$

and an interpolation formula to evaluate $\boldsymbol{a}$ between mesh points. Then, the characteristic equation (3.8) is approximated with second-order accuracy in time, according to the relationship

$$\frac{u^{n+1}(\boldsymbol{x}) - u^n(\boldsymbol{x} - \Delta\boldsymbol{X})}{\Delta t} = \frac{1}{2}\left[s^{n+1}(\boldsymbol{x}) + s^n(\boldsymbol{x} - \Delta\boldsymbol{X})\right].$$

Here $s^n(\boldsymbol{x}) = s(\boldsymbol{x}, t^n)$ and the quantities $u^n(\boldsymbol{x} - \Delta\boldsymbol{X})$ and $s^n(\boldsymbol{x} - \Delta\boldsymbol{X})$ are evaluated from $u^n(\boldsymbol{x})$ and $s^n(\boldsymbol{x})$ by interpolation. In practice, bi-cubic splines have been considered for two- and three-dimensional calculations. The value of $\Delta\boldsymbol{X}(\boldsymbol{x})$ at the previous time step can be used as a first guess to solve the nonlinear system (3.16) at each node. The convergence of the iterative procedure (3.16) is guaranteed by the fixed-point theorem provided that $\boldsymbol{a}(\boldsymbol{x}, t^{n+1/2})$ and its first partial derivatives are continuous, the time step Δt is sufficiently small and the first guess sufficiently close to the actual solution.

The accuracy properties of the semi-Lagrangian method with cubic interpolation are reported in Figure 3.2. They were derived, as explained in detail in Section 3.5, using Fourier mode analysis. The accuracy of a time integration method is usually characterized by its amplification factor and its relative phase error. These quantities are expressed in terms of the dimensionless wave number $\xi = k\,h$, where k is the wave number of the considered Fourier mode and h the mesh size.

In the present case of pure convection, the amplification factor for the exact solution is $G_{\text{exact}} = 1$ (no damping). As can be seen from Figure 3.2, the semi-Lagrangian scheme exhibits excellent amplitude response for dimensionless wave numbers in the range $0 \leq \xi \leq \pi/4$, which can be accurately resolved by the numerical method. Also the phase response of the semi-Lagrangian method appears to be excellent at small and intermediate wave numbers.

The same properties apply to the characteristic Galerkin method of Morton mentioned below. For its accuracy and efficiency the semi-Lagrangian method is widely used in meteorological forecasting as well as in the modeling of environmental flows using non-uniform Cartesian meshes, see for instance Staniforth and C ôté (1991) and Tanguay, Simard and Staniforth (1989) for practical applications of the method.

3.3.3.2 Lagrange–Galerkin methods Methods in this class also exploit the property of constant solution along a fluid particle trajectory to approximate the total derivative in the l.h.s. of (3.4a). The particle paths coincide with the associated characteristic curves and are defined by equations (3.7). Lagrange–Galerkin methods bear similarity with semi-Lagrangian methods, but are specifically adapted to the use of a finite element spatial discretization based on the Galerkin projection. Two variants of the basic Lagrange–Galerkin method are described next.

Direct integration along the characteristics: If the convection velocity $\boldsymbol{a}(\boldsymbol{x}, t)$ is known in advance over the entire time interval $]0, T[$, the complete solution of the linear convection equation (3.4a) can be determined as follows.

For a given triangulation of the domain Ω, one constructs the characteristic curves passing through all the nodes in the finite element mesh by approximating them as continuous lines consisting of straight segments. Recall that the characteristics are defined by equations (3.7). The nodal values of the unknown u at time t are then obtained by evaluating integrals in equations (3.9) or (3.10) and using numerical quadrature. A detailed description of a method of this kind is provided by Pironneau (1981/82).

In nonlinear situations, the field $\boldsymbol{a}(u, \boldsymbol{x}, t)$ depends on the solution itself, so that it is necessary to advance the solution by increments from a given time t^n to time $t^{n+1} = t^n + \Delta t$, where Δt is a convenient time step. As suggested by Pironneau (1981/82), equation (3.4a) is integrated along trajectories drawn backwards from $(\boldsymbol{x}, t^{n+1})$ to $\boldsymbol{X}(\boldsymbol{x}, t^{n+1}; t^n)$ using a suitable approximation for the characteristic velocity $\boldsymbol{a}$. Then, the standard Galerkin projection is employed to obtain the nodal values of the solution at time t^{n+1}. Both first-order methods, the conditionally stable explicit Euler scheme and the unconditionally stable backward Euler scheme, can be used in the Galerkin formulation which reads

$$\int_\Omega w\, u^{n+1}\, d\Omega = \int_\Omega w\, u^n \left(\boldsymbol{X}(\cdot, t^{n+1}; t^n)\right)\, d\Omega + \Delta t \int_\Omega w\, s^{n+\theta}\, d\Omega.$$

The choice $\theta = 0$ corresponds to the explicit method, and $\theta = 1$ corresponds to the implicit one.

Two critical issues arise in the implementation of the Lagrange–Galerkin method. First, the numerical approximation of the first term on the r.h.s. of the previous equation can give rise to instability phenomena due to errors introduced by the numerical quadrature. Second, at each integration point, $\boldsymbol{\xi}$, the interpolation of $u^n(\boldsymbol{X}(\boldsymbol{\xi}, t^{n+1}; t^n))$ must be evaluated and this may not be a trivial task. Note that, when the mapping $\boldsymbol{\xi} \rightarrow \boldsymbol{X}(\boldsymbol{\xi}, t^{n+1}; t^n)$ is used, the new position $\boldsymbol{X}(\boldsymbol{\xi}, t^{n+1}; t^n)$ may fall in a different element. See also the interesting discussion by Bermejo (1995) and Allievi and Bermejo (2000)

Characteristic variational formulation: The fact that the solution of convection problems remains constant along fluid particle paths can also be exploited in a closer connection with a weak variational formulation, as exemplified by the characteristic-based finite element method proposed by Benqu é et al. (1980; 1982).

In this method, the convection equation (3.4a) is first recast into weak form by multiplying it by a suitable test function $\psi(\boldsymbol{x}, t)$ in space–time and then integrating over the space–time domain $Q_n := \Omega \times]t^n, t^{n+1}[$. This produces the weak form

$$\iint_{Q_n} \psi\, (u_t + \boldsymbol{a} \cdot \boldsymbol{\nabla} u)\, d\Omega\, dt = \iint_{Q_n} \psi\, s\, d\Omega\, dt.$$

After integration by parts in both space and time, and assuming that the function $\psi(\boldsymbol{x}, t)$ vanishes on the spatial boundary, the previous integral equation becomes

$$\int_{\Omega} (\psi^{n+1}\, u^{n+1} - \psi^n\, u^n) d\Omega = \iint_{Q_n} \big((\psi_t + \boldsymbol{a} \cdot \boldsymbol{\nabla} \psi)\, u + \psi\, s\big) d\Omega\, dt, \tag{3.17}$$

where we have assumed that the convection velocity $\boldsymbol{a}$ is divergence free.

In order to simplify the integral on the r.h.s. of (3.17) the arbitrary test function $\psi(\boldsymbol{x}, t)$ is required to satisfy the homogeneous initial boundary value problem

$$\psi_t + \boldsymbol{a} \cdot \boldsymbol{\nabla} \psi = 0 \qquad \text{on } Q_n, \tag{3.18a}$$

$$\psi(\boldsymbol{x}, t) = 0 \qquad \text{on } \Gamma, \tag{3.18b}$$

$$\psi(\boldsymbol{x}, t^{n+1}) = w(\boldsymbol{x}), \tag{3.18c}$$

where $w(\boldsymbol{x})$ is the standard weighting function associated with the considered finite element approximation over the domain Ω.

Under these circumstances, the weak form (3.17) can be rewritten as

$$\int_{\Omega} w\, u^{n+1}\, d\Omega = \int_{\Omega} \psi^n\, u^n\, d\Omega + \iint_{Q_n} \psi\, s\, d\Omega\, dt,$$

where the unknown u^{n+1} appears explicitly on the l.h.s. Note that the finite element discretization of this equation yields a linear system of algebraic equations governed by a symmetric positive definite matrix in the form of a consistent mass matrix.

However, this simplicity is only apparent. The function $\psi(\boldsymbol{x}, t)$ must be known at t^n and where needed for the evaluation of the last integral on the r.h.s. of the previous equation. The difficulty now lies in the resolution of the problem (3.18). The homogeneous pure convection equation (3.18a) can be solved using the characteristics as shown in Section 3.3.2. In fact, the function $\psi(\boldsymbol{x}, t)$ remains constant along the characteristics because the source term is zero, see (3.9) and (3.10). Thus,

$$\psi(\boldsymbol{x}, t) = w(\boldsymbol{X}(\boldsymbol{x}, t^{n+1}; t)) \quad \text{with } t \in [t^n, t^{n+1}].$$

Two issues are relevant for this method: the evaluation of the characteristics and the computational burden due to the interpolation of $w(\boldsymbol{X}(\boldsymbol{x}, t^{n+1}; t))$. Note that this was also the case in the previous method based on a direct integration along the characteristics.

The method is nonetheless very accurate as confirmed by Morton (1996, p. 329) who demonstrates that "*on a uniform mesh in one dimension and for linear constant coefficient advection, the Lagrange–Galerkin method based on continuous piecewise linear basis functions is equivalent to a semi-Lagrangian method using cubic spline interpolation*".

In addition to the original works previously mentioned, the idea of using a Lagrangian (or characteristic-based) approach to integrate the convective terms in transport equations has been exploited by many researchers. In particular, mention should be made of the characteristic Galerkin method introduced by Morton (1982; 1983; 1985), which, with linear elements, possesses accuracy properties similar to the semi-Lagrangian method with cubic interpolation. Also noteworthy is the least-squares characteristic method due to Li (1990). The starting consideration for this method is that the solution obtained by convection of the initial data along the characteristics does not belong to the interpolation space spanned by the shape functions of the base finite element mesh. To cure this situation, the idea of Li (1990) was to employ the least-squares approach to minimize the difference between the rigidly translated solution and the desired one expressed in terms of shape functions referred to the base element mesh.

As we have seen, finite element methods directly exploiting the characteristics present technical difficulties in their practical implementation. For this reason, instead of discretizing convection problems along the particle trajectories in a Lagrangian fashion, we prefer to discretize them along the Cartesian coordinates, a choice which in our view simplifies the numerical developments. The fact nevertheless remains that time discretization schemes for convection problems should be able to mimic the role of the characteristics, or, in other words, to take into account the directional character of propagation of the information.

3.4 CLASSICAL TIME AND SPACE DISCRETIZATION TECHNIQUES

As seen in Chapter 2, achieving a stable and accurate spatial representation is the main objective in the finite element solution of steady transport problems. This is not the only issue in time-dependent problems. In fact, the spatial representation provided by finite elements needs to be accurately transported in time to trace the transient response. This implies that one must select an appropriate algorithm for numerical time integration. A proper balance between the spatial and the temporal approximations must be considered. In other words, phase accuracy now becomes an important consideration in addition to spatial stability.

A usual practice in the finite analysis of time-dependent problems consists of discretizing first with respect to the spatial variables, thus obtaining a system of coupled first-order ordinary differential equations (with respect to time), a procedure called *semi-discrete method*. Then, to complete the discretization of the PDE, it remains to integrate the first-order differential system forward in time to trace the temporal evolution of the solution starting from the initial data $u_0(\boldsymbol{x})$. The complete apparatus of numerical methods for ordinary differential equations can be used; this procedure is called in the numerical analysis literature the *method of lines*. Which discretization is performed first is not an issue for linear spatial operators with constant coefficients and a Galerkin formulation. However, for future cases (nonlinear problems, stabilization techniques for transient analysis, etc.) it is preferable that the time discretization be performed before the spatial discretization.

In this section we recall some of the most usual algorithms for solving first-order differential equations. Their accuracy and stability properties when used in conjunction with linear finite elements are presented in Section 3.5.

3.4.1 Time discretization

3.4.1.1 The θ family of methods This family is widely used for integrating first-order differential equations. It is a single step method, that is the value u^{n+1} of the problem unknown at time $t^{n+1} = t^n + \Delta t$ is determined from the value u^n at time t^n. In this case, this is done by a weighted average of u_t^n and u_t^{n+1} at the end points of the integration step:

$$\frac{u(t^{n+1}) - u(t^n)}{\Delta t} = \theta\, u_t(t^{n+1}) + (1-\theta)\, u_t(t^n) + \mathcal{O}\big((1/2-\theta)\Delta t, \Delta t^2\big).$$

The time step Δt is assumed constant for the time being and θ is a parameter taken to be in the interval $[0, 1]$. Note that u_t will be replaced using equation (3.4a).

Several well-known methods are obtained for different values of the θ parameter. For values of $\theta < 1/2$ the schemes are conditionally stable. The *Euler method*, that is $\theta = 0$, is the best known. For values of $\theta \geq 1/2$ the methods are unconditionally stable. *Backward Euler*, $\theta = 1$, *Galerkin*, $\theta = 2/3$, and *Crank–Nicolson*, $\theta = 1/2$, are the most usual ones. As the truncation error in the previous equation indicates, the only method with second-order accuracy is Crank–Nicolson (see the details in Ames, 1992; Mitchell and Griffiths, 1980; Wait and Mitchell, 1985; Johnson, 1987; Lambert, 1991). The stability properties of these methods are studied in Section 3.5.

In practice, it is usual and sometimes preferable to solve for the incremental unknown $\Delta u = u^{n+1} - u^n$ rather than for u^{n+1}:

$$\frac{\Delta u}{\Delta t} - \theta\, \Delta u_t = u_t^n, \tag{3.19}$$

where $\Delta u_t = u_t^{n+1} - u_t^n$, u^n is the approximation of $u(t^n)$, $t^n = t^0 + n\Delta t$ (t^0 is the initial time), and u_t is again replaced using the original PDE (3.4a). This results in the following semi-discrete equation:

$$\frac{\Delta u}{\Delta t} + \theta(\boldsymbol{a} \cdot \boldsymbol{\nabla})\Delta u = \theta s^{n+1} + (1-\theta)s^n - \boldsymbol{a} \cdot \boldsymbol{\nabla} u^n. \tag{3.20}$$

3.4.1.2 The Lax–Wendroff method In addition to Crank–Nicolson, second-order accurate, explicit, time-stepping algorithms are also widely used. The Lax–Wendroff method, which is based upon a truncated Taylor series expansion, is one of the most popular. In the Taylor series expansion

$$u(t^{n+1}) = u(t^n) + \Delta t\, u_t(t^n) + \frac{1}{2}\Delta t^2 u_{tt}(t^n) + \mathcal{O}(\Delta t^3),$$

the first and second time derivatives are substituted by spatial derivatives using the governing equation (3.4a). Thus, the following relationships are used:

$$\begin{aligned} u_t^n &= s^n - \boldsymbol{a} \cdot \boldsymbol{\nabla} u^n, \\ u_{tt}^n &= s_t^n - \boldsymbol{a} \cdot \boldsymbol{\nabla} u_t^n = s_t^n - \boldsymbol{a} \cdot \boldsymbol{\nabla} s^n + (\boldsymbol{a} \cdot \boldsymbol{\nabla})^2 u^n, \end{aligned}$$

and the second-order explicit method is obtained from the truncation of the previous Taylor series expansion, that is $(u^{n+1} - u^n)/\Delta t = u_t^n + \Delta t\, u_{tt}^n/2$. The resulting time-stepping algorithm reads

$$\frac{\Delta u}{\Delta t} = -\boldsymbol{a} \cdot \boldsymbol{\nabla} u^n + \frac{\Delta t}{2}\big(\boldsymbol{a} \cdot \boldsymbol{\nabla}\big)^2 u^n + s^n + \frac{\Delta t}{2}\big(s_t^n - \boldsymbol{a} \cdot \boldsymbol{\nabla} s^n\big). \tag{3.21}$$

One of the main distinguishing features of the Lax–Wendroff method is that, despite its appearance, the second term on the r.h.s. of (3.21) is not to be thought of as an added artificial diffusion term. In fact, as far as time-dependent solutions are concerned, the second-order spatial derivatives are simply a consequence of the second-order accurate temporal approximation. The tensorial structure of these terms indicates that the correction introduced by the second time derivative acts only in the direction of the streamlines, in complete analogy with the SUPG method discussed in Chapter 2.

3.4.1.3 The leap-frog method Another widely used explicit time-stepping method is the three-level leap-frog method

$$u(t^{n+1}) = u(t^{n-1}) + 2\Delta t\, u_t(t^n) + \mathcal{O}(\Delta t^3),$$

which is second-order accurate in the time step Δt because the rate of change u_t^n is evaluated at the mid-point between the time stations t^{n-1} and t^{n+1}. When applied to the convection equation (3.4a) the leap-frog method reads

$$\frac{u^{n+1} - u^{n-1}}{2\Delta t} = u_t^n = s^n - \boldsymbol{a} \cdot \boldsymbol{\nabla} u^n. \tag{3.22}$$

3.4.2 Galerkin spatial discretization

3.4.2.1 The integral equation First, the standard Galerkin method of weighted residuals will be employed for the spatial discretization of the convection problem (3.4). This can be done directly on the differential equation (3.4a), or as previously indicated, on the time discretized equations (3.19), (3.21) or (3.22). For completeness, we present the *semi-discrete* scheme first. By means of the weighted residuals method applied directly to (3.4a) the following integral equation is obtained:

$$\int_\Omega w\, u_t \, d\Omega + \int_\Omega w(\boldsymbol{a} \cdot \boldsymbol{\nabla} u) d\Omega = \int_\Omega w\, s\, d\Omega, \tag{3.23}$$

where, as in previous chapters, the test functions w belong to the space $\mathcal{V}$ and satisfy the homogeneous boundary conditions on Γ_D^{in}:

$$\mathcal{V} = \left\{ w(\boldsymbol{x}) \in \mathcal{H}^1(\Omega) \mid w(\boldsymbol{x}) = 0 \text{ for } \boldsymbol{x} \in \Gamma_D^{in} \right\}.$$

Note that the functions in $\mathcal{V}$ do not depend on time. By contrast, the solution u of (3.4) lies in $\mathcal{L}_2(0, T; \mathcal{H}^1(\Omega))$. In fact, the time dependence of u can be translated to the trial space $\mathcal{S}_t$, which varies as a function of time,

$$\mathcal{S}_t := \left\{ u \mid u(\cdot, t) \in \mathcal{H}^1(\Omega), t \in [0, T] \text{ and } u(\boldsymbol{x}, t) = u_D \text{ for } \boldsymbol{x} \in \Gamma_D^{in} \right\}.$$

Dirichlet boundary conditions are taken into account by the definition of trial space. But, to account for the Neumann boundary conditions, the convective term, the second term on the l.h.s. of (3.23), is integrated by parts. Then, the variational problem associated with the initial boundary value problem defined in (3.4) becomes: for any $t \in [0, T]$ find $u \in \mathcal{S}_t$ such that for all weighting functions $w \in \mathcal{V}$

$$\begin{cases} (w, u_t) - (\boldsymbol{\nabla} w, \boldsymbol{a}\, u) + (w, u(\boldsymbol{a} \cdot \boldsymbol{n}))_{\Gamma^{out}} = (w, h)_{\Gamma_N^{in}} + (w, s), \\ (w, u(\boldsymbol{x}, 0)) = (w, u_0(\boldsymbol{x})), \end{cases} \tag{3.24}$$

where the standard notation presented in Section 1.5 is used for the $\mathcal{L}_2$ scalar product, and h is the prescribed normal flux on Γ_N^{in}. Note that the surface integral on the l.h.s. is limited to Γ^{out} because $w = 0$ on Γ^{in}.

> **Remark 3.5 (Only Dirichlet boundary conditions).** If no Neumann conditions are prescribed on the inflow portion of the boundary the variational form presented in (3.23) can be used directly. Then, given the source term s, the Dirichlet boundary condition u_D and the initial condition u_0, one has to find $u(\boldsymbol{x}, t) \in \mathcal{S}_t$ and $t \in]0, T[$ such that for all $w \in \mathcal{V}$,
>
> $$\begin{cases} (w, u_t) + c(\boldsymbol{a}; w, u) = (w, s), \\ (w, u(\boldsymbol{x}, 0)) = (w, u_0(\boldsymbol{x})), \end{cases}$$
>
> where the trilinear form, already introduced in (2.6), is employed, namely
>
> $$c(\boldsymbol{a}; w, u) = \int_\Omega w(\boldsymbol{a} \cdot \boldsymbol{\nabla} u) d\Omega.$$

3.4.2.2 Galerkin formulation of the semi-discrete scheme The spatial discretization of the unsteady transport equation by means of the Galerkin formulation consists of defining two finite dimensional spaces $\mathcal{S}^h$ and $\mathcal{V}^h$ as subsets of $\mathcal{S}$ and $\mathcal{V}$,

$$\mathcal{V}^h := \left\{w \in \mathcal{H}^1(\Omega) \mid w|_{\Omega^e} \in \mathcal{P}_p(\Omega^e)\ \forall e \text{ and } w = 0 \text{ on } \Gamma_D\right\}$$
$$\mathcal{S}^h_t := \left\{u \mid u(\cdot,t) \in \mathcal{H}^1(\Omega), u(\cdot,t)|_{\Omega^e} \in \mathcal{P}_p(\Omega^e)\ t \in [0,T]\ \forall e \text{ and } u = u_D \text{ on } \Gamma_D\right\}$$

where $\mathcal{P}_p$ is the finite element interpolating space consisting of polynomials of order p. The *semi-discrete* Galerkin formulation is obtained by restricting forms (3.24) to the above finite dimensional spaces, namely, for any $t \in [0,T]$ find $u^h \in \mathcal{S}^h_t$ such that for all $w^h \in \mathcal{V}^h$,

$$\begin{cases} \left(w^h, u^h_t\right) - \left(\boldsymbol{\nabla} w^h, \boldsymbol{a}\, u^h\right) + \left(w^h, u^h(\boldsymbol{a}\cdot\boldsymbol{n})\right)_{\Gamma^{out}} = \left(w^h, h\right)_{\Gamma^{in}_N} + \left(w^h, s\right), \\ \left(w^h, u^h(\boldsymbol{x},0)\right) = \left(w^h, u_0(\boldsymbol{x})\right). \end{cases}$$

Once the Galerkin forms are defined, we follow the same rationale as already discussed in Section 2.2. Note, however, that now the time dependence of the solution, $u^h(\boldsymbol{x},t)$, is taken into account in the following manner: the shape functions $N_A(\boldsymbol{x})$ do not depend on time and the time dependence is accounted for by the nodal values of the unknown. Thus, (2.10) becomes

$$u^h(\boldsymbol{x},t) = \sum_{A\in\eta\setminus\eta_D} N_A(\boldsymbol{x})\,\mathrm{u}_A(t) + \sum_{A\in\eta_D} N_A(\boldsymbol{x})\,u_D(\boldsymbol{x}_A,t),$$

where, as before, η is the set of global node numbers in the finite element mesh and $\eta_D \subset \eta$ the subset of nodes belonging to the Dirichlet portion of the boundary, Γ^{in}_D. The test functions are defined as before, see (2.11), $w^h \in \mathcal{V}^h = \operatorname{span}_{B\in\eta\setminus\eta_D}\{N_B\}$.

Finally, the usual assembly process delivers the *semi-discrete* system

$$\mathbf{M}\dot{\mathbf{u}} - (\mathbf{C}^T - \mathbf{B}^{out})\mathbf{u} = \mathbf{f}, \tag{3.25}$$

which governs the transient response of the convection problem. Note that vectors $\mathbf{u}$ and $\dot{\mathbf{u}}$ contain, respectively, the nodal values of the unknown u and of its time derivative, while $\mathbf{M}$ and $\mathbf{C}$ are, respectively, the consistent mass matrix and the convection matrix. Matrix $\mathbf{B}^{out}$ is related to the outflow boundary, Γ^{out}. These matrices are obtained by topological assembly of element contributions as follows:

$$\begin{aligned} \mathbf{M} &= \mathop{\mathsf{A}}^{e}\mathbf{M}^e & M^e_{ab} &= \int_{\Omega^e} N_a N_b \, d\Omega \\ \mathbf{C} &= \mathop{\mathsf{A}}^{e}\mathbf{C}^e & C^e_{ab} &= \int_{\Omega^e} N_a(\boldsymbol{a}\cdot\boldsymbol{\nabla} N_b)d\Omega \\ \mathbf{B}^{out} &= \mathop{\mathsf{A}}^{e}[\mathbf{B}^{out}]^e & [B^{out}]^e_{ab} &= \int_{\partial\Omega^e\cap\Gamma^{out}} N_a N_b(\boldsymbol{a}\cdot\boldsymbol{n})\, d\Omega \end{aligned} \tag{3.26}$$

where A denotes the assembly operator, and $1 \leq a, b \leq \mathrm{n_{en}}$. The r.h.s. vector, $\mathbf{f}$, considers the contribution of the source term, s, the prescribed flux, h, and the

Dirichlet data u_D. It results from the assembly of nodal contributions of the form $\mathbf{f} = \overset{e}{\mathbf{A}} \mathbf{f}^e$ and

$$\begin{aligned}\mathrm{f}_a^e = \left(N_a, s\right)_{\Omega^e} + \left(N_a, h\right)_{\partial\Omega^e \cap \Gamma_N^{in}} \\ - \sum_{b=1}^{\mathrm{n_{en}}} \Bigg[\left(N_a, N_b\right)_{\Omega^e} \frac{\partial u_{Db}^e(t)}{\partial t} - \left(\boldsymbol{\nabla} N_a, \boldsymbol{a} N_b\right)_{\Omega^e} u_{Db}^e(t) \\ + \left(N_a, N_b(\boldsymbol{a} \cdot \boldsymbol{n})\right)_{\partial\Omega^e \cap \Gamma^{out}} u_{Db}^e(t) \Bigg] ,\end{aligned}$$

where $\mathrm{n_{en}}$ is the number of element nodes, and $u_{Db}^e(t) = u_D(\boldsymbol{x}_b, t)$ if u_D is prescribed at node number b and equals zero otherwise.

Remark 3.6. It is worth noting that a particularly high spatial accuracy is obtained by the Galerkin approach when a uniform mesh of linear finite elements is employed to discretize a pure convection problem. In this simple case,

$$u_t + a u_x = 0 \tag{3.27}$$

produces the following semi-discrete equation at an interior node j:

$$\dot{\mathrm{u}}_j + \frac{1}{6} \left(\dot{\mathrm{u}}_{j-1} - 2\dot{\mathrm{u}}_j + \dot{\mathrm{u}}_{j+1}\right) + a \left(\frac{u_{j+1} - u_{j-1}}{2h}\right) = 0,$$

where h is the size of the elements. It is easy to show, by means of Taylor series developments, that the previous scheme is fourth-order accurate. This is indeed a significant gain with respect to the second-order accuracy obtained with the central difference method. Such a gain is due to the presence of the consistent mass matrix; note the second term on the l.h.s. in addition to the term $\dot{\mathrm{u}}_j$ present in the central finite difference method. However, this high accuracy is lost on a non-uniform mesh or when data are not smooth enough (recall that the proof requires Taylor series developments).

Note also that the differential system (3.25) must be integrated in time to produce the transient response of the convection problem. It is easily conceived that the fourth-order spatial accuracy would be quickly eroded if the numerical time integration procedure were not of comparable accuracy. The issue of an adequate coupling between space and time discretizations is particularly important in convection problems due to the role of the characteristics in their solution.

3.4.2.3 *Galerkin formulation of the θ family methods*

As previously noted, which discretization is performed first is not an issue for linear spatial operators with constant coefficients and a Galerkin formulation. However, for future cases (nonlinear problems, stabilization techniques for transient analysis, etc.) it is preferable that time discretization precedes the spatial one. The time discretized equations, (3.19), (3.21) and (3.22), do have a truncation error. However, if the temporal truncation error is neglected, these equations can be interpreted as a spatial differential operator applied to u, or in some cases to the unknown increment Δu. In fact, they represent a strong form that must be solved at each time step. Under this rationale it is easy to determine from (3.19) the variational form associated with the θ family methods:

$$\left(w, \frac{\Delta u}{\Delta t}\right) - \theta\left(w, \Delta u_t\right) = \left(w, u_t^n\right).$$

When u_t is replaced using the original PDE, (3.4a), or if a weighted residual formulation is applied to (3.20), we obtain an equation for the unknown Δu at each time-step:

$$\begin{aligned}\left(w, \frac{\Delta u}{\Delta t}\right) &- \theta\big(\boldsymbol{\nabla} w, \boldsymbol{a}\, \Delta u\big) + \theta\big((\boldsymbol{a} \cdot \boldsymbol{n}) w, \Delta u\big)_{\Gamma^{out}} \\ &= \big(\boldsymbol{\nabla} w, \boldsymbol{a}\, u^n\big) - \big((\boldsymbol{a} \cdot \boldsymbol{n}) w, u^n\big)_{\Gamma^{out}} \\ &\quad + \big(w, \theta h^{n+1} + (1-\theta) h^n\big)_{\Gamma_N^{in}} + \big(w, \theta s^{n+1} + (1-\theta) s^n\big). \qquad (3.28)\end{aligned}$$

Note that in the previous equation the unknown Δu appears in the same three terms as in the l.h.s. of (3.24). Thus, after spatial discretization, the unknown Δu will be governed by the same mass matrix, convection matrix and outflow boundary matrix as presented in (3.25) and (3.26), but scaled by $1/\Delta t$ and θ, respectively. The r.h.s. terms include also the influence of the inflow Neumann boundary condition, h, and the source term, s. These terms are linearly interpolated, using θ, between time t^n and t^{n+1}. Moreover, since the unknown is the increment, $\Delta u = u^{n+1} - u^n$, the r.h.s. also includes the convection part at t^n.

It is important to note that in equation (3.28) time is already discretized. Therefore, its solutions are approximations of order 1 or 2 depending on the value of θ. Moreover, as previously noted, the test functions belong to the space $\mathcal{V}$ presented earlier (no variation with time) and the unknown Δu belongs to a solution space $\mathcal{S}_t$ which varies as a function of time:

$$\mathcal{S}_t := \left\{\Delta u \mid \Delta u(\cdot, t) \in \mathcal{H}^1(\Omega),\ t \in [0, T] \text{ and } \Delta u(\boldsymbol{x}, t) \text{ is given for } \boldsymbol{x} \in \Gamma_D^{in}\right\}.$$

3.4.2.4 Galerkin formulation of the Lax–Wendroff method Similar variational forms can be derived for other time-stepping schemes. From (3.21), which characterizes the Lax–Wendroff method, the following variational form is obtained:

$$\left(w, \frac{\Delta u}{\Delta t}\right) = -\left(w, \boldsymbol{a} \cdot \boldsymbol{\nabla} u^n - \frac{\Delta t}{2} (\boldsymbol{a} \cdot \boldsymbol{\nabla})^2 u^n\right) + \left(w, s^n + \frac{\Delta t}{2}\left(s_t^n - \boldsymbol{a} \cdot \boldsymbol{\nabla} s^n\right)\right).$$

After integration by parts, this equation becomes

$$\begin{aligned}\left(w, \frac{\Delta u}{\Delta t}\right) = &\left(\boldsymbol{a} \cdot \boldsymbol{\nabla} w, u^n + \frac{\Delta t}{2}\left[s^n - (\boldsymbol{a} \cdot \boldsymbol{\nabla}) u^n\right]\right) \\ &- \left((\boldsymbol{a} \cdot \boldsymbol{n}) w, u^n + \frac{\Delta t}{2}\left[s^n - (\boldsymbol{a} \cdot \boldsymbol{\nabla}) u^n\right]\right)_{\Gamma^{out}} \\ &+ \left(w, h^{n+1/2}\right)_{\Gamma_N^{in}} + \left(w, s^n + \frac{\Delta t}{2} s_t^n\right), \qquad (3.29)\end{aligned}$$

where the following expression has been used for the prescribed normal flux:

$$h^{n+1/2} = -\Big(u^n + \frac{\Delta t}{2}[s^n - (\boldsymbol{a}\cdot\boldsymbol{\nabla})u^n]\Big)(\boldsymbol{a}\cdot\boldsymbol{n}) + \mathcal{O}(\Delta t^2),$$

because it introduces a truncation error of the same order as the time-stepping scheme. Note that, as expected, this one step Lax–Wendroff method requires, at each time step, the resolution of an algebraic system. This system has a constant matrix, the consistent mass matrix. Note also that some authors call this method *second-order Taylor–Galerkin*. Thus, we shall refer to this scheme as TG2.

3.4.2.5 Galerkin formulation of the leap-frog method This method induces a variational equation obtained from (3.22) in the form

$$\left(w, \frac{u^{n+1}}{2\Delta t}\right) = \left(w, \frac{u^{n-1}}{2\Delta t}\right) + \left(w, s^n - \boldsymbol{a}\cdot\boldsymbol{\nabla}u^n\right), \tag{3.30}$$

which, after integration by parts of the convective term, becomes

$$\left(w, \frac{u^{n+1}}{2\Delta t}\right) = \left(w, \frac{u^{n-1}}{2\Delta t} + s^n\right) + (\boldsymbol{a}\cdot\boldsymbol{\nabla}w, u^n) - \big(w, u^n(\boldsymbol{a}\cdot\boldsymbol{n})\big)_{\Gamma^{out}} + \big(w, h^n\big)_{\Gamma_N^{in}}.$$

In this case, as for the previous method, the matrix governing the system of discrete equations is the consistent mass matrix.

3.5 STABILITY AND ACCURACY ANALYSIS

A fundamental result, which is classical in the finite difference literature, is the *Lax equivalence theorem* (see Richtmyer and Morton, 1967). It relates the concepts of convergence and consistency to the concept of *stability* of a numerical scheme. Section 1.5.5.1 in Chapter 1 already introduces consistency, see (1.34), and convergence, see (1.35), in the context of elliptic problems. They can be extended to initial value problems, see Quarteroni and Valli (1994, Sec. 14.3) for a detailed presentation. More precisely, the Lax equivalence theorem states that for a *well-posed* linear initial value problem, a consistent scheme is convergent if and only if it is stable. In order to comprehend this theorem some basic concepts such as *well-posed* or *stable* problem shall be introduced. A problem is considered *well-posed* when its solution depends continuously on its initial value and is uniformly bounded in any compact interval. For linear problems such as the ones considered here this concept can be formalized as follows: for a homogeneous problem (source term equal to zero) with homogeneous boundary conditions for every $T > 0$ there exists $0 < C(T) < \infty$ such that $||u(\boldsymbol{x}, t)|| \leq C(T)$ uniformly for all $t \in [0, T]$ (the constant C depends on the initial condition $||u(\boldsymbol{x}, 0)||$). For instance, the convection problem described by (3.4) with constant coefficients is well-posed because the norm of the solution is constant in time. The transient convection–diffusion–reaction problem (i.e., the transient counterpart of (2.54) that will be studied in Chapter 5) is also well-posed because diffusion will induce a decay in the norm of the solution. Thus the Lax equivalence theorem ensures that, for the complete analysis of the numerical scheme, it suffices to study its stability. Consequently, we concentrate our efforts on the stability analysis.

Stability of numerical schemes for PDEs is not a trivial issue. Here we will not study stability in detail. There are excellent references that study stability in a more formal manner for pure convection or convection–diffusion–reaction problems. The classical one is by Richtmyer and Morton (1967). Mitchell and Griffiths (1980) present a more accessible textbook for finite differences and the monograph by Quarteroni and Valli (1994), which is more finite element oriented, is also worth reading; other textbooks can also be very helpful, see for instance Hughes (2000) or Morton (1996). Different techniques (energy methods, eigenvalue techniques) can be employed to perform a stability analysis, but here we will use the Fourier method. One of the advantages of using such a technique is that apart from stability another crucial issue is studied: *accuracy*. In fact, at the end of this section, the *modified equation method* of Warming and Hyett (1974) is also introduced to investigate the accuracy properties of the various time integration schemes.

3.5.1 Analysis of stability by Fourier techniques

The classical Fourier analysis, also called *Von Neumann stability analysis* (see for instance Ames, 1992; Mitchell and Griffiths, 1980; Wait and Mitchell, 1985), considers a homogeneous linear differential equation with constant coefficients, namely equation (3.4a) with $s = 0$. But it is important to note that such a technique is designed to analyze *Cauchy problems*. That is, in a 1D situation, problems defined over the whole real axis ($-\infty < x < \infty$), and thus with no boundary conditions (there is, however, the requirement that the solution is square integrable). Hence, it is applied automatically to problems with periodic boundary conditions and it can also be extended to problems with Dirichlet conditions over finite domains for the PDEs studied here. Perhaps more important is the necessity that for each point the fully discretized equation should be identical, thus restricting treatment to uniform discretizations. In fact, we are actually going to deal with linear elements and the fully discrete equation at one interior node; this is standard procedure in finite differences where the stability literature is vast and accessible to practitioners.

Instead of restricting ourselves to the pure convection problem, we present this technique for the homogeneous transient convection–diffusion–reaction equation with constant coefficients, namely

$$u_t + \boldsymbol{a} \cdot \boldsymbol{\nabla} u - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} u) + \sigma u = 0. \tag{3.31}$$

Furthermore, we assume that the Fourier series for the initial condition, $u(\boldsymbol{x}, 0)$, is absolutely convergent (see Richtmyer and Morton, 1967, Chap. 4). This equation is studied in detail in Chapter 5.

The linearity of the problem is crucial. The evolution of the initial condition is the sum of the evolutions for each single mode in which the initial condition is decomposed. The analytical solution of (3.31) for one Fourier mode in n_{sd} spatial

Table 3.1 Spatial, discrete and mode transformation operators for 1D.

Spatial operator	Discrete	Mode transf.	Expression
$(N_i, N_j)/h^2$	$1+\frac{1}{6}\delta^2$	$\mathcal{M}(\xi)$	$1-(2/3)\sin^2(\xi/2)$
$\Delta t\,(N_i, \boldsymbol{a}\cdot\boldsymbol{\nabla}N_j)/h^2$	$\frac{1}{2}C\,\delta$	$\mathcal{A}(\xi, C)$	$i\,C\,\sin\xi$
$-\Delta t\,\nu\,(\boldsymbol{\nabla}N_i, \boldsymbol{\nabla}N_j)/h^2$	$d\,\delta^2$	$\mathcal{K}(\xi; d)$	$-4\,d\,\sin^2(\xi/2)$
$\Delta t\,(\boldsymbol{a}\cdot\boldsymbol{\nabla}N_i, N_j)/h^2$	$-\frac{1}{2}C\,\delta$	$-\mathcal{A}(\xi, C)$	$-i\,C\,\sin\xi$
$-\Delta t^2\,(\boldsymbol{a}\cdot\boldsymbol{\nabla}N_i, \boldsymbol{a}\cdot\boldsymbol{\nabla}N_j)/h^2$	$C^2\,\delta^2$	$\mathcal{D}(\xi, C)$	$-4\,C^2\,\sin^2(\xi/2)$

dimensions reads

$$u(\boldsymbol{x}, t) = \prod_{j=1,\dots,\mathrm{n_{sd}}} e^{-(\nu k_j^2+\sigma)t}\, e^{ik_j(x_j-a_jt)}, \tag{3.32}$$

where a_j is the j component of the convective velocity $\boldsymbol{a} = (a_1, a_2, \dots, a_{\mathrm{n_{sd}}})^T$ and k_j is the j component of the wave vector $\mathbf{k} = (k_1, k_2, \dots, k_{\mathrm{n_{sd}}})^T$, which defines the generalized Fourier component

$$e^{i\mathbf{k}\cdot\boldsymbol{x}} = e^{ik_1x_1}e^{ik_2x_2}\dots e^{ik_{\mathrm{n_{sd}}}x_{\mathrm{n_{sd}}}}.$$

The *exact amplification factor*, G_{ex}, which relates the value of the unknown u at two consecutive instants t^n and t^{n+1} and at each point in Ω, is then determined:

$$u^{n+1} = G_{\mathrm{ex}}u^n, \qquad G_{\mathrm{ex}} = e^{-(\delta_{\mathrm{ex}}+i\omega_{\mathrm{ex}})\Delta t} \tag{3.33}$$

where $\delta_{\mathrm{ex}} = \nu\,\mathbf{k}\cdot\mathbf{k}+\sigma$ and $\omega_{\mathrm{ex}} = \boldsymbol{a}\cdot\mathbf{k}$.

To determine the numerical amplification factor for a given time-stepping scheme, taking into account the spatial discretization, it is necessary to determine the influence of each spatial operator on a Fourier mode. Thus discrete operators associated with any of the weak continuous spatial operators defined in Section 3.4.2 must be determined. In Table 3.1 the relations between spatial operators, finite difference discrete operators and the mode transformations are shown. This table is particularized for uniform linear 1D elements of size h and consequently it requires the well-known difference operators (see Wait and Mitchell, 1985):

- Second-order centered first difference

$$\delta u(x, t) = u(x+h, t) - u(x-h, t).$$

- Second-order centered second difference

$$\delta^2 u(x, t) = u(x-h, t) - 2u(x, t) + u(x+h, t).$$

To determine the influence of the numerical scheme on a Fourier component one introduces the dimensionless wave vector ("number" in the 1D case), $\boldsymbol{\xi} = h\,\mathbf{k}$. Moreover, using Δt and h the parameters in the differential equation, (3.31), are scaled. The

Courant number is $C = \|\boldsymbol{c}\|$ with $\boldsymbol{c} = \boldsymbol{a}\Delta t/h$, the diffusion number is $d = (\nu\Delta t)/h^2$, and the dimensionless reaction is $r = \sigma\Delta t$. The exact amplification factor, (3.33), is rewritten as

$$G_{\text{ex}}(\boldsymbol{\xi}, \boldsymbol{c}, d, r) = e^{-(\delta_{\text{ex}}+i\omega_{\text{ex}})\Delta t} = e^{-(d\,\boldsymbol{\xi}\cdot\boldsymbol{\xi}+r+i\boldsymbol{c}\cdot\boldsymbol{\xi})}. \tag{3.34}$$

As noted previously we will restrict ourselves to 1D. Thus, the exact amplification factor (3.34) will be compared with the numerical one obtained for a particular numerical scheme at an arbitrary interior node j,

$$u_j^{n+1} = G(\xi, C, d, r)\; u_j^n, \qquad G(\xi, C, d, r) = e^{-(\delta+i\omega)\Delta t}. \tag{3.35}$$

Here, $u^n(x)$ denotes the piecewise linear finite element solution obtained at time $t^n = n\Delta t$ on a uniform mesh and the corresponding nodal values are denoted $u_j^n = u^n(jh)$.

Then, if the method is stable, the amplification factor G must verify

$$|\,G(\xi, C, d, r)| \leq 1$$

for all values of the dimensionless wave number $\xi \in [0, \pi]$ (note that $\xi = \pi$ corresponds to two elements per wavelength) and the dimensionless numbers C, d and r. Moreover, the accuracy of the numerical scheme can be assessed by comparing numerical and exact damping and phase values. These relative errors are defined respectively as

$$\frac{\delta}{\delta_{\text{ex}}} = \frac{|\,G(\xi, C, d, r)|}{|\,G_{\text{ex}}(\xi, C, d, r)|} \quad \text{and} \quad \frac{\omega}{\omega_{\text{ex}}} = \frac{\arg(G(\xi, C, d, r))}{\arg(G_{\text{ex}}(\xi, C, d, r))}\,. \tag{3.36}$$

From the point of view of accuracy only the range $0 \leq \xi \leq \pi/4$ is of interest because $\xi = \pi/4$ corresponds to eight elements per wavelength.

3.5.2 Analysis of classical time-stepping schemes

We shall now apply the Fourier technique to investigate the stability and accuracy properties of the previously presented time-stepping techniques combined with the Galerkin finite element formulation, see Section 3.4.2, in pure convection. Unfortunately, disadvantages of the standard Galerkin finite element method will become apparent when it is combined with the classical time-stepping schemes described in Section 3.4 (see also Morton and Parrott, 1980; Donea, Quartapelle and Selmin, 1987; Donea and Quartapelle, 1992; Morton, 1996). In particular, we shall see that second-order time schemes do not properly combine with linear finite elements in convection problems, because large values of the time step imply large phase errors.

The θ family of methods, the Lax–Wendroff method and the leap-frog method are considered for time discretization. The discrete equation obtained at an interior node j with the **$\boldsymbol{\theta}$ family method** is determined from (3.28) (recall that there is no source term, $s = 0$, and we consider an interior node). The fully discrete equation can be written from Table 3.1 as

$$\left(1 + \frac{1}{6}\delta^2 + \frac{1}{2}\theta C \delta\right)(u_j^{n+1} - u_j^n) = -\frac{1}{2}\theta C \delta u_j^n,$$

which allows us to find the equation that modifies each Fourier component using also Table 3.1,

$$(\mathcal{M}(\xi) + \theta \mathcal{A}(\xi, C))(u_j^{n+1} - u_j^n) = -\mathcal{A}(\xi, C)\, u_j^n.$$

Thus the numerical amplification factor is

$$G_\theta = \frac{\mathcal{M}(\xi) - (1-\theta)\mathcal{A}(\xi, C)}{\mathcal{M}(\xi) + \theta \mathcal{A}(\xi, C)} = \frac{1 - \frac{2}{3}\sin^2\frac{\xi}{2} - i(1-\theta)C \sin\xi}{1 - \frac{2}{3}\sin^2\frac{\xi}{2} + i\,\theta\, C \sin\xi}$$

The stability can now be studied for each value of θ verifying the condition $|G_\theta| \leq 1$. In particular for $\theta = 1/2$, that is Crank–Nicolson, one can verify that the amplification factor is always (for $0 \leq \xi \leq \pi$ and $C \geq 0$) equal to one. Thus, Crank–Nicolson is *unconditionally stable* and non dissipative (with a Galerkin formulation). Moreover, the accuracy of this method can be evaluated by means of (3.36).

A similar analysis can be performed for the **Lax–Wendroff method** departing from (3.29) (with no source term and at an interior node). The equation that modifies each Fourier component is in this case

$$\mathcal{M}(\xi)\,(u_j^{n+1} - u_j^n) = \left(\frac{1}{2}\mathcal{D}(\xi, C) - \mathcal{A}(\xi, C)\right) u_j^n,$$

and the amplification factor is

$$G_{TG2} = \frac{\mathcal{M}(\xi) + \frac{1}{2}\mathcal{D}(\xi, C) - \mathcal{A}(\xi, C)}{\mathcal{M}(\xi)} = \frac{1 - (\frac{2}{3} + 2C^2)\sin^2\frac{\xi}{2} - i\,C \sin\xi}{1 - \frac{2}{3}\sin^2\frac{\xi}{2}}$$

which induces the condition of numerical stability: $C^2 < 1/3$.

Finally, from (3.30) the analysis of the **leap-frog method** can be performed, namely

$$\mathcal{M}(\xi)\, u_j^{n+1} = \mathcal{M}(\xi)\, u_j^{n-1} - \mathcal{A}(\xi, C)\, u_j^n,$$

where the amplification factor is determined by solving a quadratic equation (hint: use $u_j^{n+1} = G_{LF}^2 u_j^{n-1}$)

$$G_{LF} = \frac{-\mathcal{A}(\xi, C) \pm \sqrt{\mathcal{M}(\xi)\mathcal{M}(\xi) + \mathcal{A}(\xi, C)\mathcal{A}(\xi, C)}}{\mathcal{M}(\xi)}$$

$$= \frac{-i\,C \sin\xi \pm \sqrt{(1 - \frac{2}{3}\sin^2\frac{\xi}{2}) - C^2 \sin^2\frac{\xi}{2}}}{1 - \frac{2}{3}\sin^2\frac{\xi}{2}}.$$

Genuine finite element schemes are based on a so-called consistent mass representation in which the mass matrix is defined as in (3.26). Another option consists of

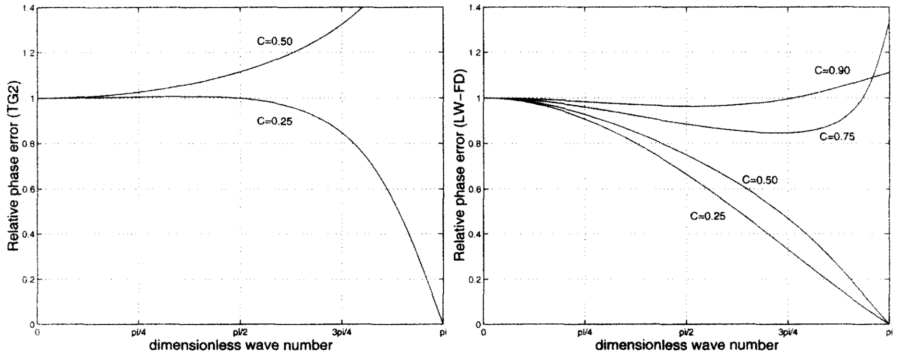


Fig. 3.3 Relative phase error in pure convection of the Lax–Wendroff method combined with linear finite elements using a consistent (left) and a diagonal (right) mass representation.

using a diagonal mass matrix. In this case, the discrete equations obtained in 1D on a uniform mesh of linear elements are identical to those obtained with second-order central differences. The finite element scheme corresponding to (3.29) with a diagonal mass representation instead of the consistent mass matrix yields the following scheme for an interior node:

$$u_j^{n+1} = u_j^n - \frac{1}{2}\,C\big(u_{j+1}^n - u_{j-1}^n\big) + \frac{1}{2}\,C^2\big(u_{j-1}^n - 2u_j^n + u_{j+1}^n\big).$$

This scheme corresponds to the central finite difference Lax–Wendroff discretization (LW-FD). Its amplification factor is

$$G_{\text{LW-FD}} = 1 - 2C^2 \sin^2\frac{\xi}{2} - i\,C\,\sin\xi.$$

It is stable up to $C^2 = 1$ and possesses the so-called *unit CFL property*; that is, the exact nodal solution is obtained on a uniform mesh when $C^2 = 1$. Recall that CFL stands for the initials of Courant, Friedrichs and Lewy authors of the celebrated paper originally published in 1928 and reprinted in english in 1967. Note that the finite difference method is far more economical from a computational point of view than its finite element analogue (a diagonal instead of a consistent matrix is employed) and it also has a larger domain of stability. In order to further compare the Lax–Wendroff finite element scheme (TG2) and its finite difference analogue the relative phase errors are depicted in Figure 3.3.

One notes that for small values of the time step Δt, that is small Courant numbers, the consistent finite element scheme has a higher-order phase accuracy than the finite element scheme using a diagonal mass representation. This is because semi-discrete (time continuous) consistent finite element schemes are fourth-order accurate on a uniform mesh, while methods based on a diagonal mass matrix only produce a second-order accurate spatial discretization. At intermediate and short wavelengths the phase error of the TG2 scheme becomes positive as the time step is increased and deteriorates

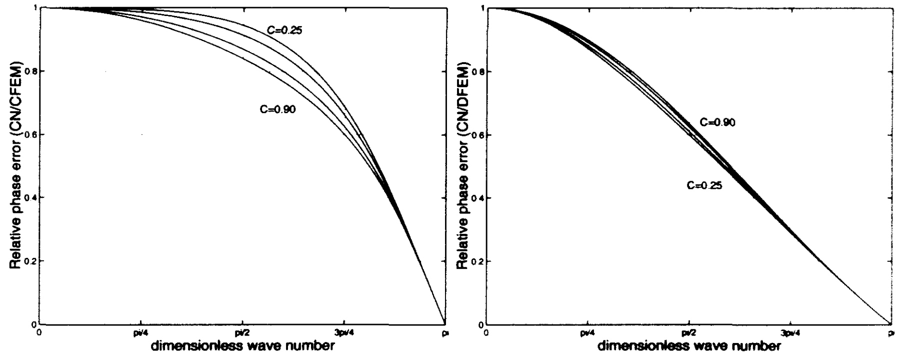


Fig. 3.4 Relative phase error in pure convection of the Crank–Nicolson method combined with linear finite elements using a consistent (left) and a diagonal (right) mass representation.

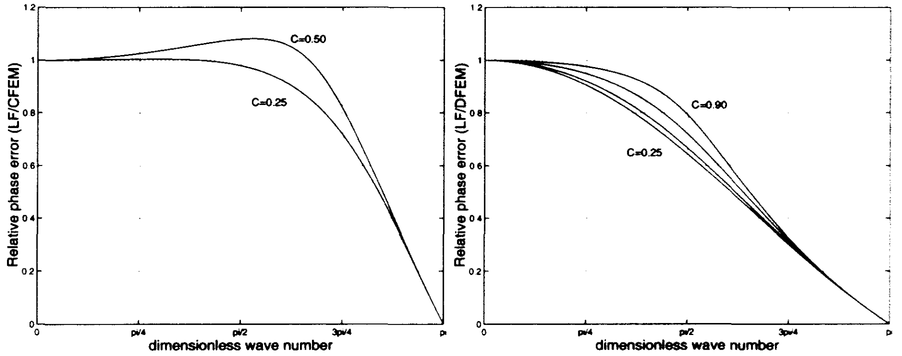


Fig. 3.5 Relative phase error in pure convection of the leap-frog method combined with linear finite elements using a consistent (left) and a diagonal (right) mass representation.

seriously as the stability limit $C^2 \rightarrow \frac{1}{3}$ is approached. By contrast, the scheme with diagonal mass matrix has a predominantly lagging phase error, except for large wave numbers when $C > \frac{1}{2}$. Here, the phase error is seen to decrease as the Courant number approaches the stability limit $C = 1$.

A similar analysis can be performed for the Crank–Nicolson and leap-frog methods. The relative phase errors of the Crank–Nicolson and leap-frog schemes combined with linear finite elements using either a consistent or a diagonal mass representation are reported in Figures 3.4 and 3.5, respectively. One notes that the phase response of the consistent finite element schemes deteriorates as the Courant number increases, while the phase accuracy in the case of a lumped-mass matrix improves as C approaches the stability limit of the explicit scheme. Furthermore, in the case of the leap-frog method, the combination with finite elements using a consistent mass matrix leads to a reduced stability range, namely $C^2 \leq 1/3$, while the scheme based on a diagonal mass representation is stable up to $C^2 \leq 1$.

In summary, the superior phase accuracy of the finite element schemes based on a consistent mass matrix, which is due to their fourth-order spatial accuracy on a uniform mesh, cannot be exploited as the time step is increased. Moreover, the stability interval in explicit methods is reduced when compared with the corresponding lumped-mass (or central difference) methods. These facts will be confirmed in the next section by an analysis based on the modified equation method.

Further evidence of the difficulties in coupling linear finite elements and second-order time-stepping algorithms is provided in Section 3.11 with the solved exercises. Here we have considered the pure convection equation, namely the first-order wave equation. Christon (1991) presents a complete analysis of the influence of the mass matrix on the dispersive nature of the semi-discrete second-order wave equation.

3.5.3 The modified equation method

Numerical methods for PDEs inevitably introduce truncation errors. Thus, numerical schemes do not solve the original PDE, but instead what Warming and Hyett (1974) call a *modified equation*. The modified equation is the PDE which is actually solved, apart from round-off errors, when a given numerical scheme is applied to solve an initial value problem. For example, the numerical solution of the linear convection equation in 1D, see (3.27), induces a modified equation of the generic form

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} + \sum_{p=1}^{\infty} \mu_{2p+1}\frac{\partial^{2p+1} u}{\partial x^{2p+1}} + \sum_{p=1}^{\infty} \mu_{2p}\frac{\partial^{2p} u}{\partial x^{2p}} = 0, \qquad (3.37)$$

where $\mu_p = \mu_p(h, \Delta t)$, and h is the element size.

The two summation terms that appear in the modified equation and do not belong to the original equation represent the truncation error of the numerical scheme. These terms provide immediate information on the dissipation and dispersion properties of the numerical scheme. In fact, even and odd derivatives are associated, respectively, with amplitude and phase errors. Moreover, the correct sign of the coefficient of the lowest-order even-derivative term in the modified equation is, in pure convection, a necessary, but not always sufficient, condition for numerical stability.

The procedure to determine the modified equation requires some tedious algebra and goes as follows:

1. The difference equation resulting from time and space discretization of the governing equation is first expanded in a double Taylor series in time and space around the space–time point $jh, n\Delta t$.

2. This produces a PDE including an infinite number of terms and also spatial, temporal and mixed derivatives of every order.

3. One then eliminates the temporal derivatives of order higher than one, as well as the mixed derivatives in favor of purely spatial derivatives using the PDE obtained in step 2. This produces the modified equation in the form indicated by equation (3.37).

Table 3.2 Modified equation in pure convection for second-order Lax–Wendroff, leap-frog and Crank–Nicolson schemes combined with linear finite elements.

TG2 + Linear elements

$$u_t + au_x =$$
$$\frac{ah^2}{6}C^2\frac{\partial^3 u}{\partial x^3} - \frac{ah^3}{24}C(1-3C^2)\frac{\partial^4 u}{\partial x^4} + \frac{ah^4}{180}(1 - \frac{15}{2}C^2 + 9C^4)\frac{\partial^5 u}{\partial x^5} + \cdots$$

Leap-frog + Linear elements

$$u_t + au_x =$$
$$\frac{ah^2}{6}C^2\frac{\partial^3 u}{\partial x^3} + \frac{ah^4}{360}(2 - 27C^4)\frac{\partial^5 u}{\partial x^5} + \cdots$$

Crank–Nicolson + Linear elements

$$u_t + au_x =$$
$$-\frac{ah^2}{12}C^2\frac{\partial^3 u}{\partial x^3} + \frac{ah^4}{720}(4 - 9C^4)\frac{\partial^5 u}{\partial x^5} + \cdots$$

The modified equations associated with the classical second-order schemes presented so far for the 1D convection equation will now be presented. They offer an alternative approach to the Fourier techniques to investigate the accuracy properties of the discrete convection schemes. The modified equations associated with the various second-order time schemes are obtained in the form indicated in Table 3.2. They are derived assuming uniform linear elements and a Galerkin spatial discretization.

Note that fourth-order spatial accuracy of finite element schemes is clearly apparent from the modified equations (the mesh size h does not affect the coefficient of the third-derivative terms, which only depends on the square of the time step: $h^2C^2 = a^2\Delta t^2$). These schemes exhibit a dominating phase error controlled by third-derivative terms which is due to the time discretization only. Unfortunately, the dominating phase error increases with the square of the time step. This implies a rapid accuracy fall-off as the time step is increased. Furthermore, the stability range of these explicit schemes (TG2 and LF) is drastically reduced compared with the corresponding lumped-mass finite element (or central difference) schemes, usually by a factor $\sqrt{3}$. See in Table 3.2 the coefficient of $\partial^4 u/\partial x^4$ in the modified equation for the Lax–Wendroff method (recall that this coefficient must be negative for stability).

By contrast, the modified equations in Table 3.3 show that when finite elements with a diagonal mass representation (or second-order central differences) are used for space discretization, the phase error of explicit schemes decreases as the Courant number increases. Note, however, that the implicit Crank–Nicolson method exhibits the same drawback as its consistent-mass counterpart. The phase error in both schemes increases with the square of the time-step size.

The behavior of second-order time schemes in the finite element solution of convection problems is further illustrated by the solved exercises in Section 3.11. There, they are compared with the higher-order time-stepping methods described in the next sections.

Table 3.3 Modified equation in pure convection for second-order Lax–Wendroff, leap-frog and Crank–Nicolson schemes combined with second-order central differences or lumped-mass linear finite elements.

Lax–Wendroff + Finite differences

$$u_t + au_x =$$
$$-\tfrac{ah^2}{6}(1-C^2)\tfrac{\partial^3 u}{\partial x^3} - \tfrac{ah^3}{8}C(1-C^2)\tfrac{\partial^4 u}{\partial x^4} - \tfrac{ah^4}{120}(1+5C^2-6C^4)\tfrac{\partial^5 u}{\partial x^5} + \cdots$$

Leap-frog + Finite differences

$$u_t + au_x =$$
$$-\tfrac{ah^2}{6}(1-C^2)\tfrac{\partial^3 u}{\partial x^3} + \tfrac{ah^4}{120}(1-10C^2+9C^4)\tfrac{\partial^5 u}{\partial x^5} + \cdots$$

Crank–Nicolson + Finite differences

$$u_t + au_x =$$
$$-\tfrac{ah^2}{6}(1+\tfrac{1}{2}C^2)\tfrac{\partial^3 u}{\partial x^3} - \tfrac{ah^4}{120}(1+5C^2+\tfrac{3}{2}C^4)\tfrac{\partial^5 u}{\partial x^5} + \cdots$$

3.6 TAYLOR–GALERKIN METHODS

3.6.1 The need for higher-order time schemes

We have underlined in Section 3.5 the difficulties in coupling second-order time schemes with linear finite elements in convection problems. These difficulties can be overcome by extending the time accuracy beyond second order.

To highlight the need for accurate time-stepping methods in the finite element solution of time-dependent convection problems we will restrict ourselves to the homogeneous, $s = 0$, 1D case, see equation (3.5).

Section 3.3 showed that the solution of this equation remains constant along the characteristic lines $dx/dt = a$, see equation (3.6). Therefore, provided the solution is regular enough, the value of the unknown u at two consecutive time levels $t^n = n\Delta t$ and $t^{n+1} = t^n + \Delta t$ satisfies

$$\begin{aligned} u(x,t^{n+1}) &= u(x - a\Delta t, t^n) \\ &= u(x,t^n) - a\Delta t\,\frac{\partial u}{\partial x}(x,t^n) + \frac{(a\Delta t)^2}{2}\frac{\partial^2 u}{\partial x^2}(x,t^n) + \cdots \\ &= \exp\left(-a\Delta t\frac{\partial}{\partial x}\right) u(x,t^n). \end{aligned} \tag{3.38}$$

Moreover, for an infinitely differentiable function a Taylor series of $u(x,t^{n+1})$ indicates that

$$u(x,t^{n+1}) = \exp\left(\Delta t\frac{\partial}{\partial t}\right) u(x,t^n). \tag{3.39}$$

Relations (3.38) and (3.39) highlight a key aspect of the numerical approximation of convection problems: space and time are linked by the characteristics and the discretization of one certainly influences the other. Good methods are thus needed for both numerical time integration and spatial representation; that is, the exponential function in (3.39) and the spatial operator in (3.38) must be properly approximated. The fact is that second-order time schemes do not allow a sufficiently accurate approximation of the exponential operator in (3.39), or, stated in other words, they do not properly account for the directional character of propagation of information in hyperbolic problems. Higher-order time-stepping schemes provide a better approximation to the exponential function in (3.39), and consequently allow a better account of the propagation of information along the characteristics. Higher-order implicit time-stepping methods are discussed in Chapter 5 where the finite element solution of transient convection–diffusion problems is addressed. In the present chapter devoted to pure convection problems, we shall present higher-order explicit methods known as Taylor–Galerkin (TG) methods (Donea, 1984; Donea et al., 1987; Donea and Quartapelle, 1992). Such methods represent an attempt to simulate, by a Taylor series in time extended to third or fourth order, the concept that in convective transport the solution remains constant along the characteristics.

3.6.2 Third-order explicit Taylor–Galerkin method

3.6.2.1 Time discretization Explicit Taylor–Galerkin methods represent a generalization of the Lax–Wendroff method discussed in Section 3.4.1.2. They are based on a Taylor series expansion up to the desired order. Thus, the solution u must be sufficiently smooth. In order to obtain a third-order method the Taylor series is taken as

$$\frac{u(t^{n+1}) - u(t^n)}{\Delta t} = u_t(t^n) + \frac{1}{2}\Delta t\, u_{tt}(t^n) + \frac{1}{6}\Delta t^2 u_{ttt}(t^n) + \mathcal{O}(\Delta t^3). \quad (3.40)$$

Then, time derivatives of u $(u_t, u_{tt}, \dots)$ are replaced using the original differential equation, (3.4a) in our case. For pure convection equations the order of the time derivatives does correspond with that of the space derivatives. In this case, in order to substitute the time derivatives in (3.40), they are expressed as

$$u_t = s - \boldsymbol{a}\cdot\boldsymbol{\nabla}u, \quad (3.41\text{a})$$

$$u_{tt} = s_t - \boldsymbol{a}\cdot\boldsymbol{\nabla}u_t = s_t - \boldsymbol{a}\cdot\boldsymbol{\nabla}s + (\boldsymbol{a}\cdot\boldsymbol{\nabla})^2 u, \quad (3.41\text{b})$$

$$u_{ttt} = s_{tt} - \boldsymbol{a}\cdot\boldsymbol{\nabla}s_t + (\boldsymbol{a}\cdot\boldsymbol{\nabla})^2 u_t. \quad (3.41\text{c})$$

Note, however, that in (3.41c) u_t has not been substituted by (3.41a). This is done on purpose to have at most second spatial derivatives and preserve the use of standard $\mathcal{C}^0$-continuous finite element approximations. In practice, u_t in (3.41c) is substituted by $(u^{n+1} - u^n)/\Delta t$, and the third-order approximation of the Taylor series (3.40) becomes

$$\left[1 - \frac{\Delta t^2}{6}(\boldsymbol{a}\cdot\boldsymbol{\nabla})^2\right]\frac{u^{n+1} - u^n}{\Delta t} = -(\boldsymbol{a}\cdot\boldsymbol{\nabla})u^n + \frac{\Delta t}{2}(\boldsymbol{a}\cdot\boldsymbol{\nabla})^2 u^n + s^n + \frac{\Delta t}{2}(s_t^n - \boldsymbol{a}\cdot\boldsymbol{\nabla}s^n) + \frac{\Delta t^2}{6}(s_{tt}^n - \boldsymbol{a}\cdot\boldsymbol{\nabla}s_t^n). \quad (3.42)$$

This time-stepping scheme only involves first and second time derivatives and is the basis of the one-step, third-order accurate, explicit Taylor–Galerkin method.

3.6.2.2 *Spatial discretization* Let us now illustrate the construction of an explicit third-order Taylor–Galerkin method for the convection problem (3.4a). Mixed Dirichlet and Neumann conditions are prescribed on the inlet portion of the boundary, as specified in equations (3.4c) and (3.4d). The convection velocity $\boldsymbol{a}$ is assumed to be divergence free, so that $\boldsymbol{a} \cdot \boldsymbol{\nabla} u = \boldsymbol{\nabla} \cdot (u\, \boldsymbol{a})$.

To produce the Galerkin variational form associated with scheme (3.42), we multiply it by the test function w and integrate over the spatial domain Ω. After integration by parts, the following variational problem is obtained where $\Delta u = u^{n+1} - u^n$ denotes the incremental unknown.

Given u^n, find $\Delta u \in \mathcal{S}$, such that for all $w \in \mathcal{V}$,

$$\begin{aligned}\Big(w, \frac{\Delta u}{\Delta t}\Big) &+ \frac{\Delta t^2}{6}\Big(\boldsymbol{a} \cdot \boldsymbol{\nabla} w, \boldsymbol{a} \cdot \boldsymbol{\nabla} \frac{\Delta u}{\Delta t}\Big) - \frac{\Delta t^2}{6}\Big((\boldsymbol{a} \cdot \boldsymbol{n}) w, \boldsymbol{a} \cdot \boldsymbol{\nabla} \frac{\Delta u}{\Delta t}\Big)_{\Gamma^{out}} \\ &= \Big(\boldsymbol{a} \cdot \boldsymbol{\nabla} w, u^n - \frac{\Delta t}{2}(\boldsymbol{a} \cdot \boldsymbol{\nabla} u^n)\Big) - \Big((\boldsymbol{a} \cdot \boldsymbol{n}) w, u^n - \frac{\Delta t}{2}(\boldsymbol{a} \cdot \boldsymbol{\nabla} u^n)\Big)_{\Gamma^{out}} \\ &\quad + \frac{\Delta t}{2}\big(\boldsymbol{a} \cdot \boldsymbol{\nabla} w, s^{n+1/3}\big) - \frac{\Delta t}{2}\big((\boldsymbol{a} \cdot \boldsymbol{n}) w, s^{n+1/3}\big) \\ &\quad + \Big(w, \frac{3}{4} s^{n+2/3} + \frac{1}{4} s^n\Big) + \Big(w, \frac{3}{4} h^{n+2/3} + \frac{1}{4} h^n\Big)_{\Gamma_N^{in}}. \end{aligned} \tag{3.43}$$

Note that the term responsible for the third-order accuracy of the explicit Taylor–Galerkin method, which we call TG3, introduces a modification to the standard mass matrix which nevertheless remains symmetric if only Dirichlet boundary conditions are present.

Notice also that the integration by parts produces boundary terms, which on a portion Γ_N^{in} of the inlet boundary generate a natural condition for the prescribed normal inlet flux h in (3.4d). As shown by equation (3.43), the boundary term on the outlet boundary contributes partly to the matrix governing the incremental unknown Δu, the remainder being part of the known independent term. The boundary terms on Γ^{out} are necessary to obtain the correct form of the discrete equation at nodes on the outlet boundary of the computational domain. Neglecting these terms generates spurious reflections at outflow boundaries (Donea and Quartapelle, 1992).

The linear system to be solved at each time step has characteristics similar to the one obtained by applying the second-order Lax–Wendroff scheme with finite elements. The advantage here is that, with similar computational cost, we obtain third-order accuracy. The generalized mass matrix of TG3 is tridiagonal in 1D and has the typical profile of a stiffness matrix in 2D and 3D. Generally, the matrix is characterized by a diagonal dominance that allows an approximate but accurate solution of the corresponding linear system with a few Jacobi iterations.

Remark 3.7. When the velocity field $\boldsymbol{a}$ and the source term s are functions of space and time, as originally stated for the initial boundary value problem (3.4), the third-order explicit method (3.42) becomes

$$\begin{aligned}\left[1-\frac{\Delta t^2}{6}(\boldsymbol{a}^{n+1}\cdot\boldsymbol{\nabla})^2+\frac{\Delta t^2}{6}(\boldsymbol{a}_t^{n+1}\cdot\boldsymbol{\nabla})\right]u^{n+1}\\ =\left[1+\frac{\Delta t^2}{3}(\boldsymbol{a}^{n}\cdot\boldsymbol{\nabla})^2-\frac{\Delta t^2}{3}(\boldsymbol{a}_t^{n}\cdot\boldsymbol{\nabla})-\Delta t(\boldsymbol{a}^{n}\cdot\boldsymbol{\nabla})\right]u^{n}+\Delta t\, s^n\\ +\frac{\Delta t^2}{6}\left[s_t^{n+1}-(\boldsymbol{a}^{n+1}\cdot\boldsymbol{\nabla})s^{n+1}\right]+\frac{\Delta t^2}{3}\left[s_t^{n}-(\boldsymbol{a}^{n}\cdot\boldsymbol{\nabla})s^{n}\right],\end{aligned}\qquad(3.44)$$

where $\boldsymbol{a}^{n+1}=\boldsymbol{a}(\boldsymbol{x},t^{n+1})$, etc. Here, the time dependence of the characteristic velocity causes the modified mass matrix to lose its symmetric character.

Remark 3.8. The Taylor–Galerkin method can be applied as well in the solution of nonlinear conservation law equations of the form

$$\frac{\partial u}{\partial t}+\boldsymbol{\nabla}\cdot\boldsymbol{f}(u)=s.$$

Examples will be given in Chapter 4 where the inviscid Burgers' equation and the Euler equations of gas dynamics are considered.

3.6.2.3 Stability analysis For the 1D convection equation (3.27), the TG3 scheme (3.43) produces the following discrete equation at an interior node j of a uniform mesh of linear elements of size h:

$$\left(1+\frac{1}{6}(1-C^2)\delta^2\right)(u_j^{n+1}-u_j^n)=-\frac{1}{2}C\delta u_j^n+\frac{1}{2}C^2\delta^2u_j^n,\qquad(3.45)$$

where use has been made of the central difference operators, see Table 3.1. The amplification factor of the TG3 scheme is given by

$$G_{\text{TG3}}(\xi,C)=\frac{1-\frac{2}{3}(1+2C^2)\sin^2\frac{\xi}{2}-iC\sin\xi}{1-\frac{2}{3}(1-C^2)\sin^2\frac{\xi}{2}},\qquad(3.46)$$

and the stability condition is $|C|\leq 1$. The TG3 scheme possesses the unit CFL property and its accuracy characteristics are illustrated in Figure 3.6 which shows the diagrams of its phase and amplitude errors. The comparison with the same data in Figure 3.2 for the semi-Lagrangian method with cubic interpolation and the characteristic Galerkin method of Morton reveals that the phase is exact in the three methods for $C=1/2$ and $C=1$, and that the phase error is almost identical, the TG3 scheme being slightly inferior in the range $1/2<C<1$.

Unfortunately, the TG3 scheme experiences a drastic reduction of its stability range in multidimensional situations. As shown in Figure 3.7, the limit curve for stability in 2D strongly depends on the orientation of the Courant vector $\boldsymbol{c}$ whose components are (c_x,c_y). It is given by the equation

$$c_x^{2/3}+c_y^{2/3}=1,$$

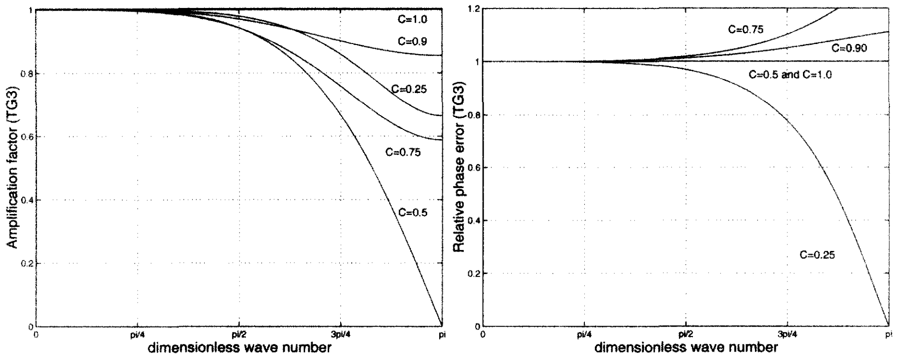


Fig. 3.6 Accuracy properties in pure convection of the third-order explicit Taylor–Galerkin scheme TG3: amplification factor modulus $|G|$ (left) and relative phase error $\phi_{numer}/\phi_{exact}$ (right).

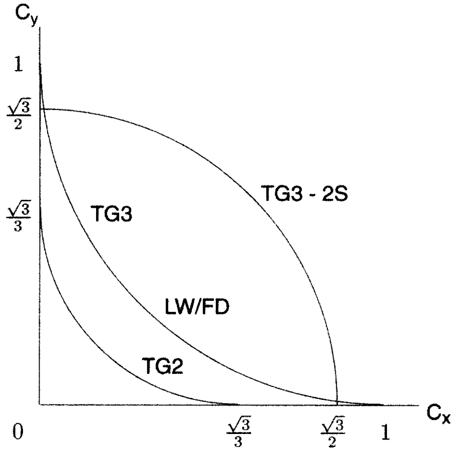


Fig. 3.7 Stability range in two-dimensional convection of one-step Taylor–Galerkin method (TG3), two-step Taylor–Galerkin method (TG3-2S), and Lax–Wendroff finite element (TG2) and finite difference (LW/FD) methods.

where the components of the Courant vector are defined as follows:

$$c_x = a_x \Delta t / h_x, \quad c_y = a_y \Delta t / h_y,$$

with a_x and a_y being the velocity components and h_x and h_y the mesh sizes along the Cartesian coordinate directions. The 1D stability condition $|C| \leq 1$ becomes in the most restrictive 2D case, that is $|c_x| = |c_y| := |C|$, $|C| < 1/(2\sqrt{2}) = 0.353$ and $|C| < 1/(3\sqrt{3}) = 0.192$ in 3D. As shown below, a two-step version of the method can be developed which possesses a more extended stability domain in multidimensional situations.

Table 3.4 Modified equation in pure convection for two second-order schemes (TG2 and leap-frog) and two higher-order methods (3rd-order TG3 and 4th-order LFTG).

<u>TG2 + Linear elements</u>

$$u_t + au_x =$$
$$\frac{ah^2}{6}C^2\frac{\partial^3 u}{\partial x^3} - \frac{ah^3}{24}C(1-3C^2)\frac{\partial^4 u}{\partial x^4} + \frac{ah^4}{180}(1-\tfrac{15}{2}C^2+9C^4)\frac{\partial^5 u}{\partial x^5} + \cdots$$

<u>TG3 + Linear elements</u>

$$u_t + au_x =$$
$$-\frac{ah^3}{24}C(1-C^2)\frac{\partial^4 u}{\partial x^4} + \frac{ah^4}{180}(1-5C^2+4C^4)\frac{\partial^5 u}{\partial x^5} + \cdots$$

<u>Leap-frog + Linear elements</u>

$$u_t + au_x =$$
$$\frac{ah^2}{6}C^2\frac{\partial^3 u}{\partial x^3} + \frac{ah^4}{360}(2-27C^4)\frac{\partial^5 u}{\partial x^5} + \cdots$$

<u>LFTG + Linear elements</u>

$$u_t + au_x =$$
$$\frac{ah^4}{360}(2+5C^2-7C^4)\frac{\partial^5 u}{\partial x^5} + \cdots$$

Figure 3.7 also shows that the situation is even worse for the Lax–Wendroff finite element scheme. The stability limit for Lax–Wendroff in 2D, called TG2, is in fact given by

$$c_x^{2/3} + c_y^{2/3} = (1/3)^{1/3}.$$

3.6.2.4 Properties of the explicit Taylor–Galerkin method The basic distinguishing features of the third-order explicit Taylor–Galerkin method TG3 can be underlined as follows:

1. The modified equation associated with TG3 is reported in Table 3.4 where it can be compared with that of the second-order Lax–Wendroff finite element method (TG2). Note that the leading dispersion error due to the time discretization has moved from the third- to the fifth-order derivative because of the increased time accuracy. The leading term of the dissipation error is a fourth-order derivative in both the second-order TG2 scheme and the third-order TG3 scheme. The necessary condition for numerical stability in 1D is $C^2 \leq 1$ for TG3 as compared with $C^2 \leq 1/3$ for the Lax–Wendroff finite element method (TG2). Moreover, the lowest-order terms of both dispersion and dissipation errors for TG3 are found to be zero for $C^2 = 1$. The optimal stability limit and the unit CFL property of TG3 are confirmed by the amplification factor (3.46) of the scheme.

2. As already pointed out for the Lax–Wendroff method, the terms in the TG3 scheme, see (3.43), involving $(\boldsymbol{a} \cdot \nabla)^2$ are not to be thought of as an artificial

numerical diffusion inherent to the scheme. In fact, as far as time-dependent solutions are concerned, the second-order spatial terms are only an element of the improved temporal approximation. The tensorial structure of these terms indicates that the correction introduced by the second time derivative acts only in the direction of the streamlines, in complete analogy with the SUPG method discussed in Chapter 2.

3.6.3 Fourth-order explicit leap-frog method

The concept in the Taylor–Galerkin method of improving the time accuracy of finite element schemes for unsteady convection can be extended to other time-stepping algorithms. As an illustrative example, we mention the fourth-order accurate extension of the standard second-order leap-frog method

$$\frac{u^{n+1}-u^{n-1}}{2\Delta t} = u_t^n .$$

The improved method is based upon the forward and backward Taylor series

$$\begin{aligned} u(t^{n+1}) &= u(t^n) + \Delta t\, u_t^n(t^n) + \frac{\Delta t^2}{2} u_{tt}^n(t^n) + \frac{\Delta t^3}{6} u_{ttt}^n(t^n) + O(\Delta t^4), \\ u(t^{n-1}) &= u(t^n) - \Delta t\, u_t^n(t^n) + \frac{\Delta t^2}{2} u_{tt}^n(t^n) - \frac{\Delta t^3}{6} u_{ttt}^n(t^n) + O(\Delta t^4), \end{aligned} \tag{3.47}$$

which by subtraction produce the fourth-order accurate leap-frog method

$$\frac{u^{n+1}-u^{n-1}}{2\Delta t} = u_t^n + \frac{1}{6}\Delta t^2 u_{ttt}^n . \tag{3.48}$$

The construction of the fourth-order leap-frog Taylor–Galerkin (LFTG) method then proceeds as for the TG3 method. The LFTG scheme is fourth-order accurate in time and also fourth-order accurate in space on a mesh of uniform linear elements in 1D. Its stability condition in 1D is $C^2 \leq 1$, while the second-order leap-frog finite element scheme has the reduced stability limit $C^2 \leq 1/3$.

Because the leap-frog time discretization is a centered discretization, the method is non-dissipative, that is $|G| = 1$, in pure convection. The relative phase error of the LFTG scheme for $C = 0.25, 0.50, 0.75$ and 0.90 is illustrated in Figure 3.8 in comparison with its second-order counterpart which cannot be operated at the last two values of the Courant number. To further appraise the gain in accuracy brought about by the fourth-order leap-frog Taylor–Galerkin scheme LFTG, we compare in Table 3.4 its modified equation for pure convection with that for the corresponding second-order leap-frog method. As can be seen, the LFTG scheme is characterized by a fifth-order dominating phase error, as compared with the third-order one for the classical second-order leap-frog method.

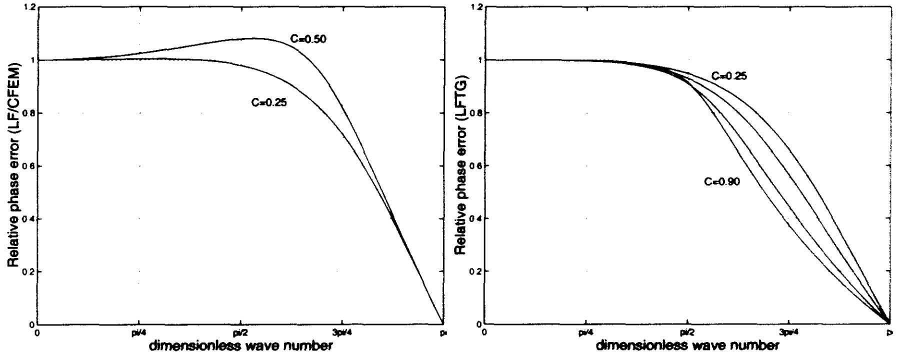


Fig. 3.8 Relative phase error $\phi_{numer}/\phi_{exact}$ in pure convection of the second-order (left) and fourth-order (right) leap-frog schemes combined with consistent linear finite elements.

3.6.4 Two-step explicit Taylor–Galerkin methods

We shall now describe two-step versions of the explicit Taylor–Galerkin method TG3 which include second time derivatives only and are thus easier to implement than the one-step method TG3, especially for solving nonlinear multidimensional hyperbolic problems, see Selmin, Donéa and Quartapelle (1985) for details. A further advantage of the two-step Taylor–Galerkin methods is their extended stability range in multidimensional situations as compared with the one-step TG3 method. A two-step third-order method is discussed first. Then, fourth-order ones are briefly presented.

3.6.4.1 Two-step third-order method A two-step version of the third-order TG3 scheme has been proposed by Selmin (1987). The method, called TG3-2S, is based on the idea of achieving a third-order temporal accuracy by means of the following two-step procedure:

$$\begin{aligned} \tilde{u}^n &= u^n + \frac{1}{3}\Delta t\, u_t^n + \alpha\, \Delta t^2\, u_{tt}^n, \\ u^{n+1} &= u^n + \Delta t\, u_t^n + \frac{1}{2}\Delta t^2\, \tilde{u}_{tt}^n, \end{aligned} \tag{3.49}$$

where the value of the parameter α is left unspecified for the time being. Third-order accuracy is achieved when combining both steps; the parameter α only influences the coefficient of the fourth-order term in the overall time series. As a consequence, its value will only affect the modulus of the amplification factor of the resulting scheme but not its phase. A convenient way to determine the available degree of freedom, α, consists in selecting it in order to have, in the 1D linear case, the same phase speed as that of the one-step TG3 scheme, equation (3.45). For the 1D linear convection equation, see (3.27), the fully discrete version of the two-step scheme (3.49) becomes

$$\begin{aligned}\left(1+\frac{1}{6}\delta^2\right)(\tilde{u}^n-u^n) &= -\frac{1}{6}\,C\,\delta u^n+\alpha\, C^2\,\delta^2 u^n,\\ \left(1+\frac{1}{6}\delta^2\right)(u^{n+1}-u^n) &= -\frac{1}{2}\,C\,\delta u^n+\frac{1}{2}\,C^2\,\delta^2\tilde{u}^n,\end{aligned} \tag{3.50}$$

and the amplification factor $G_{\text{TG3-2S}}$ of the parameterized two-step scheme is derived in the form

$$G_{\text{TG3-2S}}(\xi,C,\alpha)=\frac{1-\frac{2}{3}\sin^2\frac{\xi}{2}-2C^2\,\tilde{G}(\xi,C,\alpha)\,\sin^2\frac{\xi}{2}-iC\sin\xi}{1-\frac{2}{3}\sin^2\frac{\xi}{2}}$$

where

$$\tilde{G}(\xi,C,\alpha)=1+\frac{-\frac{1}{3}iC\sin\xi-4\alpha C^2\sin^2\frac{\xi}{2}}{1-\frac{2}{3}\sin^2\frac{\xi}{2}}.$$

It is immediately verified that

$$G_{\text{TG3-2S}}(\xi,C,1/9)=G_{\text{TG3}}(\xi,C)\,R(\xi,C), \tag{3.51}$$

where $R(\xi,C)$ is the real function

$$R(\xi,C)=\frac{\left(1-\frac{2}{3}(1-C^2)\sin^2\frac{\xi}{2}\right)\left(1-\frac{2}{3}(1+C^2)\sin^2\frac{\xi}{2}\right)}{\left(1-\frac{2}{3}\sin^2\frac{\xi}{2}\right)^2}. \tag{3.52}$$

Therefore, for the choice $\alpha=1/9$ the two-step procedure (3.49) reproduces exactly the phase–speed characteristics of the single-step TG3 scheme. The condition of numerical stability for the two-step method is $|C|\le\sqrt{3}/2\simeq 0.866$. Figure 3.9 illustrates the amplitude response of the two-step scheme which appears to be slightly more dissipative than its one-step counterpart, especially for Courant numbers close to the stability limit.

It is important to note that the excellent phase properties of the one-step third-order TG3 scheme can be reproduced exactly by the two-step procedure in 2D and 3D, see Selmin (1987) and Quartapelle (1993). Moreover, as shown in Figure 3.7, the stability range of the two-step scheme in 2D remains practically unaltered with respect to that in 1D. In fact, the stability limit in 2D is defined by

$$c_x^2+c_y^2=3/4.$$

This is in sharp contrast with the one-step Taylor–Galerkin scheme (TG3), which experiences a drastic reduction of its stability limit in multidimensional situations. Thus, the two-step formulation of the explicit Taylor–Galerkin method, besides making high-order accuracy accessible for truly nonlinear problems, offers the additional advantage of giving an isotropic stability domain in multidimensional problems.

Observe that the effect of the modified mass matrix in TG3 is achieved here, for the two-step scheme, through a double application of the standard consistent mass

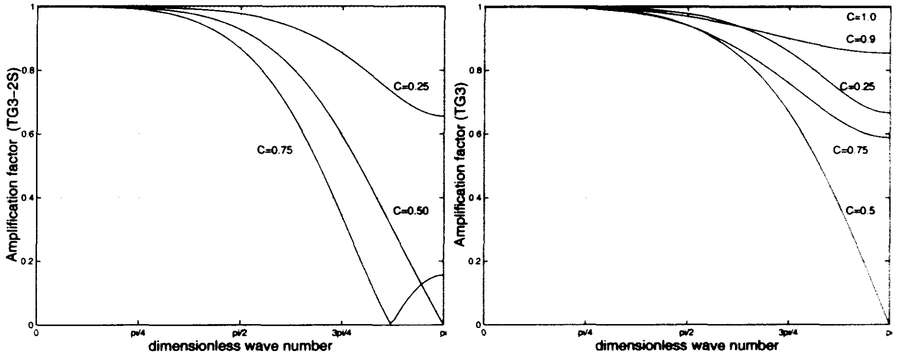


Fig. 3.9 Amplification factor modulus $|G|$ of Selmin's two-step third-order explicit scheme TG3-2S (left) compared with the one-step third-order method TG3 (right).

matrix. This alleviates the computational effort with respect to the one-step method (3.43) in which the modified mass matrix depends on the problem unknown.

> **Remark 3.9 (Vector convection equation).** When a system of nonlinear convection equations is considered, the application of the two-step method becomes much more involved than with a single equation. Examples will be given in Chapter 4 where the method is applied for solving the Euler equations. We refer to the articles by Laval (1988) and Laval and Quartapelle (1990), or to the book by Quartapelle (1993), for the implementation details of the two-step method in application to a vector convection equation.

3.6.4.2 Two-step fourth-order methods As a last example of high-order explicit schemes for convective transport, we shall illustrate the construction of two fourth-order methods suggested by Quartapelle (1993).

Consider the fourth-order explicit temporal approximation

$$u(t^{n+1}) = u(t^n) + \Delta t\, u_t^n(t^n) + \frac{\Delta t^2}{2} u_{tt}^n(t^n) \\ + \frac{\Delta t^3}{6} u_{ttt}^n(t^n) + \frac{\Delta t^4}{24} u_{tttt}^n(t^n) + \mathcal{O}(\Delta t^5).$$

It can be transformed into a two-stage method by means of the following factorization:

$$1 + \Delta t\frac{\partial}{\partial t} + \frac{\Delta t^2}{2}\frac{\partial^2}{\partial t^2} + \frac{\Delta t^3}{6}\frac{\partial^3}{\partial t^3} + \frac{\Delta t^4}{24}\frac{\partial t}{\partial t^4} \\ = 1 + \Delta t\frac{\partial}{\partial t} + \frac{\Delta t^2}{2}\frac{\partial^2}{\partial t^2}\left(1 + \frac{\Delta t}{3}\frac{\partial}{\partial t} + \frac{\Delta t^2}{12}\frac{\partial^2}{\partial t^2}\right),$$

which produces the two-stage explicit method

$$\begin{aligned}\tilde{u} &= u^n + \frac{1}{3}\Delta t\, u_t^n + \frac{1}{12}\Delta t^2\, u_{tt}^n,\\ u^{n+1} &= u^n + \Delta t\, u_t^n + \frac{1}{2}\Delta t^2\, \tilde{u}_{tt}.\end{aligned} \tag{3.53}$$

This two-step method has the same stability and accuracy properties as the classical fourth-order explicit Runge–Kutta method. For the linear convection equation (3.4a) the method is stable up to $C^2 = 1$.

As shown by Quartapelle (1993), there is another possibility of achieving a fourth-order temporal accuracy with a two-step strategy. It is based on two second-order expansions

$$\begin{aligned}\tilde{u} &= u^n + \alpha\Delta t\, u_t^n + \beta\Delta t^2 u_{tt}^n,\\ u^{n+1} &= u^n + \Delta t\, \tilde{u}_t + \gamma\Delta t^2 \tilde{u}_{tt},\end{aligned} \tag{3.54}$$

where the parameters are defined as

$$\alpha = 0.141, \quad \beta = 0.116, \quad \gamma = 0.359.$$

The stability limit of this alternative two-step fourth-order scheme is $|C| \leq 0.847$ and its phase response properties are found to be superior to those of the two-step method (3.53). The book by Quartapelle (1993) contains a detailed presentation of the above two-step fourth-order explicit methods.

3.7 AN INTRODUCTION TO MONOTONICITY-PRESERVING SCHEMES

Up to now, the emphasis has been placed on achieving time-accurate solutions to unsteady convection problems. Here, we wish to recall that discontinuities may appear in the solution of convection problems, even if they are governed by linear equations. This will indeed be the case when discontinuous data are prescribed on the inflow portion of the boundary.

In such a situation, the numerical solution delivered by stabilized finite element methods, see Chapter 2, may still present localized oscillations in the vicinity of sharp solution gradients. The consequence of such undesirable overshoots may be that some physical constraint, such as for instance positivity of the unknown, might be violated, thus leading to non-physical results.

There is therefore a need to ensure non-oscillatory, or *monotone*, numerical behavior near strong solution gradients to obtain physically correct results. This aspect will be discussed in a certain depth in Chapter 4 in connection with the solution of the Euler equations of gas dynamics. In this section, our aim is simply to introduce the concept of high-order *monotonicity-preserving schemes* along the lines introduced by Harten (1983) (or the recent reprint Harten, 1997). For simplicity, we shall make reference to the 1D convection equation.

A scheme is said to be *monotonicity preserving* if monotonicity of the solution u^{n+1} at time t^{n+1} follows from monotonicity of the solution u^n at time t^n. By monotonicity of a solution, one means that the following inequalities hold for all of its components:

$$\min(u_{j-1}, u_{j+1}) \leq u_j \leq \max(u_{j-1}, u_{j+1}).$$

An important theorem concerning monotonicity-preserving schemes is due to Godunov (1959) who showed that any linear monotonicity-preserving scheme is at most first-order accurate in space. Linear monotonicity-preserving schemes add the same artificial dissipation all over the computational domain and, consequently, are too dissipative in the smooth part of the solution.

In response to the deficiency of monotone schemes, so-called *high-resolution schemes* were developed in the 1980s. Such schemes are at least second-order accurate in the smooth part of the solution, but include a nonlinear (solution-dependent) damping mechanism that permits a sharp resolution of strong solution gradients.

Central to the development of high-resolution schemes is the concept of *total variation diminishing* (TVD) schemes introduced by Harten (1983). The total variation of a smooth function u is defined as

$$\mathrm{TV}(\mathbf{u}) = \sum_{\forall j} |u_j - u_{j-1}|,$$

and a scheme is said to be TVD if

$$\mathrm{TV}(\mathbf{u}^{n+1}) \leq \mathrm{TV}(\mathbf{u}^n).$$

The TVD requirement is less stringent than the monotonicity preservation requirement and allows a significant improvement in accuracy.

The guiding idea behind the design of TVD schemes is that physical solutions to scalar hyperbolic equations do not allow the appearance of any new extremum in the evolution of the unknown. Accordingly, in a TVD scheme the total variation of the numerical solution is controlled in a nonlinear way to prevent the appearance of any spurious new extremum.

Most high-order TVD schemes can be viewed as centered-difference schemes with an appropriate numerical dissipation calibrated so as to preserve monotonicity without compromising the accuracy. Modern methods use nonlinear numerical dissipation with diffusion coefficients dependent upon the local behavior of the solution, being larger near discontinuities than in smooth regions.

Basic references for the construction of TVD schemes in the finite difference context are, in addition to the pioneering work of Harten, Yee (1987), Hirsch (1990), LeVeque (1992) and Quarteroni and Valli (1994). Notice that the development of TVD schemes for use in connection with finite elements is still the object of active research.

To give a simple example of a TVD scheme, suppose we wish to solve the 1D convection equation (3.27) with $a > 0$ using the Crank–Nicolson method and linear finite elements. Using a diagonal mass representation, we obtain the following discrete equation at an interior node j:

$$u_j^{n+1} + \frac{\Delta t}{2h}\big(h_{j+1/2}^{n+1} - h_{j-1/2}^{n+1}\big) = u_j^n - \frac{\Delta t}{2h}\big(h_{j+1/2}^n - h_{j-1/2}^n\big),$$

where

$$h_{j+1/2} = \frac{1}{2}a(u_{j+1} + u_j) \qquad \text{and} \qquad h_{j-1/2} = \frac{1}{2}a(u_j + u_{j-1}) \tag{3.55}$$

are the numerical fluxes for the considered centered scheme. Note that a first-order upwind scheme would use

$$h_{j+1/2} = a\,u_j \qquad \text{and} \qquad h_{j-1/2} = a\,u_{j-1}.$$

As seen in Chapter 2, full upwinding generally introduces an excessive amount of numerical damping. Following Yee (1987), the above second-order scheme can be rendered into a TVD scheme by transforming the fluxes (3.55) as follows:

$$\begin{aligned} h_{j+1/2} &= \frac{1}{2}\Big(a\,(u_{j+1} + u_j) - |a|(1 - Q_{j+1/2})\Delta_{j+1/2}\,u\Big), \\ h_{j-1/2} &= \frac{1}{2}\Big(a\,(u_j + u_{j-1}) - |a|(1 - Q_{j-1/2})\Delta_{j-1/2}\,u\Big), \end{aligned} \tag{3.56}$$

where $Q_{j+1/2}$ and $Q_{j-1/2}$ are limiting functions depending on the solution gradient. The variations $\Delta_{j+1/2}\,u$ and $\Delta_{j-1/2}\,u$ are defined as follows:

$$\Delta_{j+1/2}\,u = u_{j+1} - u_j, \qquad \Delta_{j-1/2}\,u = u_j - u_{j-1}.$$

The limiting function $Q_{j+1/2}$ depends on three consecutive element gradients

$$\Delta_{j-1/2}u, \quad \Delta_{j+1/2}u, \quad \Delta_{j+3/2}u$$

and is of the form

$$Q_{j+1/2} = Q(r_{j+1/2}^-\,, r_{j+1/2}^+),$$

where

$$r_{j+1/2}^- = \frac{\Delta_{j-1/2}\,u}{\Delta_{j+1/2}\,u} \qquad \text{and} \qquad r_{j+1/2}^+ = \frac{\Delta_{j+3/2}\,u}{\Delta_{j+1/2}\,u}.$$

Two examples for the limiting function Q are

$$\begin{aligned} Q(r^-\,, r^+) &= \text{minmod}(1, r^-, r^+), \\ Q(r^-\,, r^+) &= \text{minmod}\big(2, 2r^-, 2r^+, 0.5(r^- + r^+)\big). \end{aligned}$$

The "minmod" function of a list of arguments is equal to the smallest number in absolute value if all arguments are of the same sign, or is equal to zero if any argument is of the opposite sign.

After substitution of the modified fluxes (3.56) in equation (3.55), one gets

$$
\begin{aligned}
u_j^{n+1} &+ \frac{\Delta t}{4h}\Big(a\,(u_{j+1}^{n+1} + u_j^{n+1}) - |a|\,(1 - Q_{j+1/2}^{n+1})\,\Delta_{j+1/2}\,u^{n+1}\Big) \\
&- \frac{\Delta t}{4h}\Big(a\,(u_j^{n+1} + u_{j-1}^{n+1}) - |a|\,(1 - Q_{j-1/2}^{n+1})\,\Delta_{j-1/2}\,u^{n+1}\Big) \\
&\qquad = u_j^n - \frac{\Delta t}{4h}\Big(a\,(u_{j+1}^n + u_j^n) - |a|\,(1 - Q_{j+1/2}^n)\,\Delta_{j+1/2}\,u^n\Big) \\
&\qquad\quad + \frac{\Delta t}{4h}\Big(a\,(u_j^n + u_{j-1}^n) - |a|\,(1 - Q_{j-1/2}^n)\,\Delta_{j-1/2}\,u^n\Big).
\end{aligned}
$$

Notice that the above TVD algorithm is nonlinear even when it is applied to the linear convection equation (3.27). To solve this set of nonlinear equations non-iteratively, a linearized version is usually considered. It consists in the replacement of Q^{n+1} by Q^n in the l.h.s., see Yee (1987). Another benefit of this linearization is that the scheme, though involving five points, now leads to a tridiagonal system of linear equations. This is because, at the $(n+1)$-th time level, only three points are involved, namely u_{j-1}^{n+1}, u_j^{n+1} and u_{j+1}^{n+1}, the other two points being at the n-th time level.

After linearization the implicit TVD scheme reads

$$
\begin{aligned}
u_j^{n+1} &+ \frac{a\Delta t}{4h}(u_{j+1}^{n+1} - u_{j-1}^{n+1}) \\
&- \frac{|a|\Delta t}{4h}\Big((1 - Q_{j+1/2}^n)(u_{j+1}^{n+1} - u_j^{n+1}) - (1 - Q_{j-1/2}^n)(u_j^{n+1} - u_{j-1}^{n+1})\Big) \\
&= u_j^n - \frac{a\Delta t}{4h}(u_{j+1}^n - u_{j-1}^n) \\
&\quad + \frac{|a|\Delta t}{4h}\Big((1 - Q_{j+1/2}^n)(u_{j+1}^n - u_j^n) - (1 - Q_{j-1/2}^n)(u_j^n - u_{j-1}^n)\Big).
\end{aligned}
$$

Note that the TVD method selectively adds a numerical dissipation, the maximum value of which is

$$\nu_{\max} = \frac{|a|h}{2}.$$

The above 1D strategy can be adapted to deal with 2D situations. In this case, as suggested by Donea, Selmin and Quartapelle (1988), one works with segments connecting adjacent nodes, for instance the element sides in a mesh of triangular elements. Other examples of high-order monotonicity-preserving schemes will be given in Chapter 4 where shock-capturing techniques are discussed in detail.

3.8 LEAST-SQUARES-BASED SPATIAL DISCRETIZATION

The implicit Crank–Nicolson scheme discussed in Section 3.4 is not dissipative (pure convection is considered here). It must therefore be combined with a finite element spatial representation capable of introducing the amount of numerical dissipation required to produce stable results in the presence of steep solution gradients.

In the case of purely convective transport, a stabilization technique can be developed on the basis of a pure least-squares minimization. Two variants of such an approach will be described in this section. They require that time discretization is performed before the spatial discretization and rely on a quadratic functional associated with the semi-discrete version of the governing equation. Further extensions of least squares to convection problems using space–time formulations are discussed in Section 3.10. The book by Jiang (1998) can be consulted for a detailed account of least-squares finite element methods.

3.8.1 Least-squares approach for the θ family of methods

An interesting least-squares method for pure convection was originally proposed by Carey and Jiang (1988) in connection with the θ family of methods presented in Section 3.4.1. Consider the linear convection equation (3.4a) discretized with respect to time by means of this family of methods, and recall (3.20):

$$\frac{\Delta u}{\Delta t} + \theta(\boldsymbol{a}\cdot\boldsymbol{\nabla})\Delta u = \theta s^{n+1} + (1-\theta)s^n - (\boldsymbol{a}\cdot\boldsymbol{\nabla})u^n.$$

This equation can be viewed as a spatial strong form that must be solved at each time step, namely

$$\mathcal{L}(\Delta u) - f = 0,$$

where $\mathcal{L} = 1/\Delta t + \theta\boldsymbol{a}\cdot\boldsymbol{\nabla}$ is the spatial differential operator, and the known source term is $f = \theta s^{n+1} + (1-\theta)s^n - \boldsymbol{a}\cdot\boldsymbol{\nabla}u^n$. Minimization of the least-squares functional, $\big(\mathcal{L}(\Delta u) - f, \mathcal{L}(\Delta u) - f\big)$, produces the least-squares equation (still in the spatial continuum)

$$\big(\mathcal{L}(w), \mathcal{L}(\Delta u) - f\big) = 0,$$

which takes the following explicit form:

$$\begin{aligned}\Big(\frac{w}{\Delta t} + \theta\boldsymbol{a}\cdot\boldsymbol{\nabla}w, \frac{\Delta u}{\Delta t} + \theta\boldsymbol{a}\cdot\boldsymbol{\nabla}\Delta u\Big)&\\ = \Big(\frac{w}{\Delta t} + \theta\boldsymbol{a}\cdot\boldsymbol{\nabla}w, \theta s^{n+1} + (1-\theta)s^n - \boldsymbol{a}\cdot\boldsymbol{\nabla}u^n\Big).&\qquad (3.57)\end{aligned}$$

This equation highlights the symmetric character of the implicit operator of the least-squares method. On a mesh of uniform linear elements in 1D, this scheme provides the fully discrete equation

$$\left(1 + \left(\frac{1}{6} - \theta^2C^2\right)\delta^2\right)\Delta u = -\frac{1}{2}C\,\delta u_j^n + \theta\,C^2\delta^2u_j^n,$$

and consequently the following amplification factor:

$$G_{\mathrm{CJ}} = \frac{1 - 2(\frac{1}{3} + 2(1-\theta)\theta C^2)\sin^2\frac{\xi}{2} - ic\sin\xi}{1 - 2(\frac{1}{3} - 2\theta^2C^2)\sin^2\frac{\xi}{2}}.$$

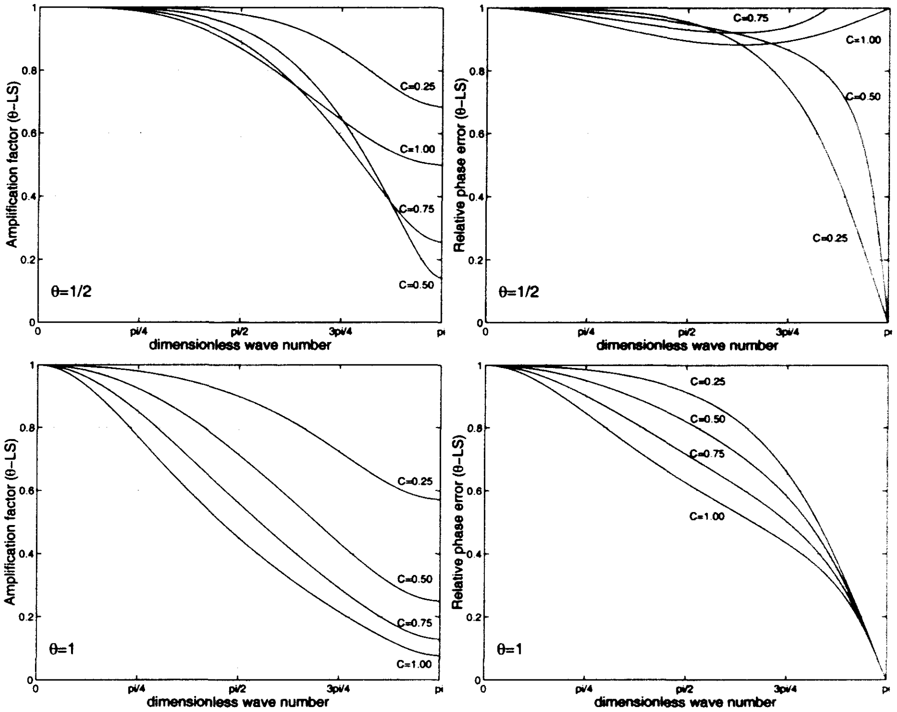


Fig. 3.10 Numerical properties of the least-squares method of Carey and Jiang for Crank–Nicolson $\theta = 1/2$ (top) and backward Euler $\theta = 1$ (bottom) for several values of the Courant number.

From the previous equations one can see that the least-squares scheme is unconditionally stable for $\frac{1}{2} \leq \theta \leq 1$ (Carey and Jiang, 1988). Figure 3.10 displays its numerical properties for $\theta = 1/2$ and $\theta = 1$. For $\theta = 1/2$ the scheme has a rather accurate phase response provided it is operated with $|C| \leq 1$. But, the fully implicit scheme corresponding to $\theta = 1$ has a poor phase accuracy, except for small values of the Courant number. This scheme can nevertheless be of interest for computing steady-state solutions by means of a false transient.

3.8.2 Taylor least-squares method

An extremely accurate least-squares method for the spatial discretization of transient convection problems was introduced by Park and Liggett (1990). Their method combines Taylor–Galerkin and least-squares concepts. It starts from the time discretized version of the linear convection equation (3.4a) provided by the following fourth-order accurate implicit scheme originally used by Harten and Tal-Ezer (1981):

$$\frac{\Delta u}{\Delta t} = \frac{1}{2}\left(u_t^{n+1} + u_t^n\right) - \frac{\Delta t}{12}\left(u_{tt}^{n+1} - u_{tt}^n\right). \tag{3.58}$$

Assuming a time-independent convection velocity, the scheme reads

$$\mathcal{L}\Big(\frac{\Delta u}{\Delta t}\Big) = -\boldsymbol{a}\cdot\boldsymbol{\nabla}u^n + f^*, \tag{3.59}$$

where

$$\mathcal{L} = 1 + \frac{\Delta t}{2}\boldsymbol{a}\cdot\boldsymbol{\nabla} + \frac{\Delta t^2}{12}(\boldsymbol{a}\cdot\boldsymbol{\nabla})^2 \qquad \text{and}$$

$$f^* = \frac{1}{2}(s^{n+1} + s^n) - \frac{\Delta t}{12}(s_t^{n+1} - s_t^n) + \frac{\Delta t}{12}(\boldsymbol{a}\cdot\boldsymbol{\nabla})(s^{n+1} - s^n).$$

The quadratic functional associated with this semi-discrete equation is

$$\Big(\mathcal{L}\Big(\frac{\Delta u}{\Delta t}\Big) + \boldsymbol{a}\cdot\boldsymbol{\nabla}u^n - f^*, \mathcal{L}\Big(\frac{\Delta u}{\Delta t}\Big) + \boldsymbol{a}\cdot\boldsymbol{\nabla}u^n - f^*\Big),$$

and the minimization procedure leads to the following least-squares weighted residual formulation:

$$\Big(\mathcal{L}(w), \mathcal{L}\Big(\frac{\Delta u}{\Delta t}\Big)\Big) = \big(\mathcal{L}(w), -\boldsymbol{a}\cdot\boldsymbol{\nabla}u^n + f^*\big).$$

This equation can also be derived if one recalls the strong form (Euler–Lagrange equation) associated with a least-squares functional induced by $\mathcal{L}(u) = s$. It corresponds to the higher-order problem $\mathcal{L}^*\mathcal{L}(u) = \mathcal{L}^*(s)$, where $\mathcal{L}^*$ is the formal adjoint of operator $\mathcal{L}$. In the present case the differential equation is defined in (3.59), and thus the adjoint operator is

$$\mathcal{L}^* = 1 - \frac{\Delta t}{2}(\boldsymbol{a}\cdot\boldsymbol{\nabla}) + \frac{\Delta t^2}{12}(\boldsymbol{a}\cdot\boldsymbol{\nabla})^2.$$

It follows that the Taylor least-squares weak formulation for the semi-discrete equation (3.59) could be obtained by applying the standard Galerkin projection to the time integration scheme:

$$\begin{aligned}\Big(1 - \frac{\Delta t^2}{12}(\boldsymbol{a}\cdot\boldsymbol{\nabla})^2 + \Big(\frac{\Delta t^2}{12}\Big)^2(\boldsymbol{a}\cdot\boldsymbol{\nabla})^4\Big)\frac{\Delta u}{\Delta t}\\ = \Big(1 - \frac{\Delta t}{2}(\boldsymbol{a}\cdot\boldsymbol{\nabla}) + \frac{\Delta t^2}{12}(\boldsymbol{a}\cdot\boldsymbol{\nabla})^2\Big)\big(-\boldsymbol{a}\cdot\boldsymbol{\nabla}u^n + f^*\big).\end{aligned}$$

Due to the presence of third- and fourth-order derivatives in this least-squares equation, Park and Liggett (1990) employed $\mathcal{C}^1$ finite elements (cubic Hermitian polynomials) instead of the standard $\mathcal{C}^0$ finite elements.

$\mathcal{C}^1$ finite elements require, as nodal unknowns, the value of the interpolating function and also its derivatives, namely u and u_x in 1D. This implies an important increase in nodal unknowns. For instance, a 2D four-noded element constructed by the tensor product of the 1D Hermite cubics has 16 unknowns, four per node: u, u_x, u_y and u_{xy}, see for instance Carey and Oden (1983). This increase is more dramatic in 3D.

The study of the amplification factor and accuracy properties of the resulting scheme is given in the original paper. It also compares the Taylor/Least-squares scheme with the third-order explicit Taylor–Galerkin method and other least-squares methods. Numerical results for 2D calculations indicate that the Taylor/Least-squares scheme is extremely accurate. Park and Liggett (1991) also implemented this method in 3D using serendipity-type Hermitian elements, which have less nodal unknowns compared with the complete Hermite interpolation and thus less accuracy.

3.9 THE DISCONTINUOUS GALERKIN METHOD

Due to the compactness of its formulation, the discontinuous Galerkin method initially introduced by Lasaint and Raviart (1974) is being increasingly used in the solution of convection and convection–diffusion problems, see for instance the work of Cockburn (1998) and Baumann and Oden (1999). We briefly introduce the discontinuous Galerkin method in application to the linear first-order hyperbolic problem

$$
\begin{aligned}
&u_t + \boldsymbol{\nabla}\cdot(\boldsymbol{a}\,u) = s(\boldsymbol{x},t) && \text{in } \Omega\times]0,T[, && (3.60a)\\
&u(\boldsymbol{x},0) = u_0(\boldsymbol{x}) && \text{on } \Omega \text{ at } t=0, && (3.60b)\\
&u = u_D && \text{on } \Gamma_D^{in}\times]0,T[. && (3.60c)
\end{aligned}
$$

The discontinuous approximations belong to a so-called *broken space*. Given a regular partition, $\mathcal{T}^h$, of the computational domain Ω into subdomains Ω^e, the test functions belonging to the broken space $\mathcal{V}(\mathcal{T}^h)$ are continuous and smooth in every element $\Omega^e \in \mathcal{T}^h(\Omega)$, but discontinuous across inter-element boundaries. Moreover, as previously done in Section 3.4.2, the trial space varies with time, namely

$$
\begin{aligned}
\mathcal{V}(\mathcal{T}^h) &:= \left\{w \in \mathcal{L}_2(\Omega) \mid w|_{\Omega^e} \in \mathcal{H}^2(\Omega^e)\,\forall\,\Omega^e \in \mathcal{T}^h(\Omega)\right\}\\
\mathcal{S}_t(\mathcal{T}^h) &= \left\{u \mid u(\cdot,t) \in \mathcal{L}_2(\Omega),\ u(\cdot,t)|_{\Omega^e} \in \mathcal{H}^2(\Omega^e)\, t \in [0,T]\,\forall\,\Omega^e \in \mathcal{T}^h(\Omega)\right\}.
\end{aligned}
$$

Typical elements of $\mathcal{S}_t(\mathcal{T}^h)$ and $\mathcal{V}(\mathcal{T}^h)$ are non-zero functions on one element Ω^e and zero everywhere else on the mesh.

The discontinuous Galerkin formulation of problem (3.60) can then be stated as follows: given $u_0(\boldsymbol{x})$, for any $t \in]0,T[$ find $u \in \mathcal{S}_t(\mathcal{T}^h)$, such that $u(\boldsymbol{x},0) = u_0(\boldsymbol{x})$ and

$$
\begin{aligned}
\big(w, u_t + \boldsymbol{\nabla}\cdot(\boldsymbol{a}\,u) - s\big)_{\Omega^e} - \big(w, (u^+ - u^-)(\boldsymbol{a}\cdot\boldsymbol{n}_e)\big)_{\partial\Omega^{e,in}\setminus\Gamma^{in}}&\\
- \big(w, (u^+ - u_D)(\boldsymbol{a}\cdot\boldsymbol{n}_e)\big)_{\partial\Omega^{e,in}\cap\Gamma^{in}} &= 0
\end{aligned}
$$

for all $w \in \mathcal{V}(\mathcal{T}^h)$ and $\Omega^e \in \mathcal{T}^h(\Omega)$. Here $\boldsymbol{n}_e$ is the outward normal to element e. Note that this weak form is local, that is it is defined over one element, not global, over the complete spatial domain Ω. This is possible because the test functions are discontinuous along inter-element boundaries. Note that this does not imply that each element can be solved independently. In fact, the boundary integral introduces

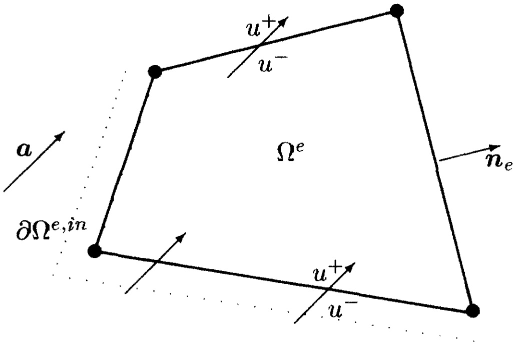


Fig. 3.11 Notation for discontinuous Galerkin method.

a weakly enforced continuity condition across the boundaries, which couples the unknowns of adjacent elements. As shown in Figure 3.11, to cope with the discontinuity of the field variable across inter-element boundaries, one defines

$$u^{\pm} = \lim_{\epsilon \to 0^+} u(\boldsymbol{x} \pm \epsilon \boldsymbol{a}) \qquad \text{for } \boldsymbol{x} \in \partial\Omega^e .$$

Moreover, the inlet portion of the element boundary, $\partial\Omega^{\,e}$, is defined by

$$\partial\Omega^{e,in} = \{\boldsymbol{x} \in \partial\Omega^e \,|\, \boldsymbol{a} \cdot \boldsymbol{n}_e(\boldsymbol{x}) < 0\} .$$

An alternative discontinuous Galerkin formulation is obtained by integration by parts of the divergence term. It provides a natural boundary condition for the convective flux and reads as follows:

$$\begin{aligned}&\big(w, u_t - s\big)_{\Omega^e} - \big(\boldsymbol{\nabla} w \cdot \boldsymbol{a}, u\big)_{\Omega^e}\\ &\qquad + \big(w, u^-(\boldsymbol{a} \cdot \boldsymbol{n}_e)\big)_{\partial\Omega^e \setminus \Gamma^{in}} + \big(w, u_D(\boldsymbol{a} \cdot \boldsymbol{n}_e)\big)_{\partial\Omega^e \cap \Gamma^{in}} = 0. \qquad (3.61)\end{aligned}$$

Despite their conditional stability, the use of explicit time-stepping schemes is very convenient here, because the global mass matrix is block diagonal with uncoupled blocks. The problem can then be solved element by element in one sweep by inverting at very low cost the element mass matrix. In Section 3.11.4 we shall illustrate the application of the discontinuous Galerkin method in the solution of a simple 1D problem of propagation of a steep front. Another attractive aspect of this method is its easy combination with adaptive refinement procedures. This is because in each element the polynomial order can be adapted to the local smoothness of the solution. A typical example is the adaptive-order discontinuous Galerkin method developed by Baumann and Oden (2000). In Chapter 4 this method is discussed in more detail and used to solve the Euler equations of gas dynamics.

3.10 SPACE–TIME FORMULATIONS

Up to here, time discretization of the convection equation has been performed using finite difference formula while finite elements are employed for spatial discretization only. As a matter of fact, finite element interpolations can also be used in the time domain. Methods in this class include the time-discontinuous Galerkin method (Jamet, 1978; Johnson et al., 1984), the space–time least-squares method proposed by Nguyen and Reynen (1984), as well as the space–time Galerkin/Least-squares algorithms developed by Shakib and Hughes (1991) and the space–time integrated least-squares approach proposed by Perrochet and Azérad (1995). In fact, the weighted residual formulation is now extended over the space–time domain. Usually, low-order approximations, such as piecewise constant or piecewise linear ones, are employed to describe time dependency. If a linear interpolation over a time slab $]t^n, t^{n+1}[$ is assumed, the local space–time interpolation is written in product form

$$u^h(\boldsymbol{x}, t) = \sum_{A=1}^{\mathrm{n_{np}}} N_A(\boldsymbol{x})\big((1-\theta)\mathrm{u}_A^n + \theta\,\mathrm{u}_A^{n+1}\big),$$

where $\theta = (t - t^n)/(t^{n+1} - t^n)$.

Let us now illustrate the extension to the space–time domain of finite element methods for transient convection problems. Note that the method has also been applied to the Navier–Stokes equations to be discussed in Chapter 6, see for instance Masud and Hughes (1997).

We consider piecewise continuous approximations in space and discontinuous approximations in time. Discontinuous approximations in time allow us to solve independently for each time slab instead of solving a global problem over the whole time domain. Recall that such a procedure was already used in space when we discussed the discontinuous Galerkin method in Section 3.9.

The time domain is partitioned in $\mathrm{n_{st}}$ sub-intervals, where each sub-interval is defined as $I^n =]t^n, t^{n+1}[$, $n = 0, 1, \dots, \mathrm{n_{st}} - 1$. Space–time slabs are then obtained in the form

$$Q^n = \Omega \times I^n.$$

For the considered space–time slab Q^n, the spatial domain Ω is subdivided into $\mathrm{n_{el}}$ elements, Ω^e, $e = 1, \dots, \mathrm{n_{el}}$, giving space–time element domains

$$Q_e^n = \Omega^e \times I^n, \quad e = 1, \dots, \mathrm{n_{el}}.$$

3.10.1 Time-discontinuous Galerkin formulation

To introduce time-discontinuous approximations in a simple context, let us first consider the time-discontinuous Galerkin formulation (Johnson et al., 1984; Shakib, 1989; Shakib and Hughes, 1991). Since the finite element interpolation is discontinuous at the space–time slab interfaces, it is useful to employ the notation

$$u^h(t_\pm^n) = \lim_{\epsilon \to 0^+} u^h(t^n \pm \epsilon).$$

The finite element spaces for the trial and weighting functions are defined as follows:

$$
\begin{aligned}
\mathcal{S}^h &= \bigcup_{n=0}^{\mathrm{n_{st}}-1} \mathcal{S}^h_n,\ \mathcal{S}^h_n = \left\{u^h \mid u^h \in \mathcal{C}^0(Q^n),\ u^h|_{Q^n_e} \in \mathcal{P}_k(Q^n_e),\ u^h|_{\Gamma^{in}} = u_D\right\} \\
\mathcal{V}^h &= \bigcup_{n=0}^{\mathrm{n_{st}}-1} \mathcal{V}^h_n,\ \mathcal{V}^h_n = \left\{w^h \mid w^h \in \mathcal{C}^0(Q^n),\ w^h|_{Q^n_e} \in \mathcal{P}_k(Q^n_e),\ w^h|_{\Gamma^{in}} = 0\right\}.
\end{aligned}
\tag{3.62}
$$

Note that continuity over Q^n implies continuity in space but does not require a continuous interpolation between time slabs. Although $\mathcal{P}_k$ indicates the space of polynomials of total degree $\leq k$ (a complete basis of degree k, see Section 1.5.2) the degrees of the polynomials in space and time can be chosen independently.

The weighted residual formulation of the homogeneous linear convection equation (3.4a) with Dirichlet inlet conditions is: for $n = 0, 1, \ldots, \mathrm{n_{st}} - 1$, find $u^h \in \mathcal{S}^h_n$, such that for all $w^h \in \mathcal{V}^h_n$,

$$
\iint_{Q^n} w^h \left(u^h_t + \boldsymbol{a} \cdot \boldsymbol{\nabla} u^h\right) d\Omega\, dt + \int_{\Omega} w^h(t^n_+) \left(u^h(t^n_+) - u^h(t^n_-)\right) d\Omega = 0, \tag{3.63}
$$

with the initial condition $u^h(\boldsymbol{x}, t^0_-) = u_0(\boldsymbol{x})$. The last integral in (3.63) is a jump condition which imposes a weakly enforced continuity condition across the slab interfaces and is the mechanism by which information is propagated from one slab to another.

We shall use finite element approximations over a space–time slab which are piecewise polynomials in space and linear in time; that is, for $(\boldsymbol{x}, t) \in Q^n = \Omega \times I^n$,

$$
u^h(\boldsymbol{x}, t) = \sum_{A=1}^{\mathrm{n_{np}}} N_A(\boldsymbol{x}) \left(\Theta_1(t)\, \tilde{\mathrm{u}}^n_A + \Theta_2(t)\, \mathrm{u}^{n+1}_A\right).
$$

$N_A(\boldsymbol{x})$ is the spatial shape function at node A; $\Theta_1(t)$ and $\Theta_2(t)$ are the time interpolation functions defined for the linear case as

$$
\begin{aligned}
\Theta_1(t) &= \frac{t^{n+1} - t}{t^{n+1} - t^n} = \frac{t^{n+1} - t}{\Delta t}, \quad \text{and} \\
\Theta_2(t) &= \frac{t - t^n}{t^{n+1} - t^n} = \frac{t - t^n}{\Delta t};
\end{aligned}
$$

and the nodal values of u^h for node A at t^n_+ and t^{n+1}_- are, respectively, $\tilde{\mathrm{u}}^n_A$ and u^{n+1}_A. The test functions w^h for each time slab (recall: piecewise polynomials in space and linear in time) are similarly defined, $N_A(\boldsymbol{x})\, \Theta_1(t)$ and $N_A(\boldsymbol{x})\, \Theta_2(t)$ for $A = 1, \ldots, \mathrm{n_{np}}$. With these definitions the weighted residual equation (3.63) yields the following couple of equations for each node A:

$$
\sum_{B=1}^{\mathrm{n_{np}}} \left\{ \iint_{Q^n} N_A\, \Theta_1 \left[N_B \frac{\mathrm{u}^{n+1}_B - \tilde{\mathrm{u}}^n_B}{\Delta t} + \left(\Theta_1\, \tilde{\mathrm{u}}^n_B + \Theta_2\, \mathrm{u}^{n+1}_B\right)(\boldsymbol{a} \cdot \boldsymbol{\nabla}) N_B \right] d\Omega\, dt \right\} = 0
$$

$$\sum_{B=1}^{\mathrm{n_{np}}} \left\{ \iint_{Q^n} N_A\, \Theta_2 \left[N_B \frac{\mathrm{u}_B^{n+1} - \tilde{\mathrm{u}}_B^n}{\Delta t} + (\Theta_1\, \tilde{\mathrm{u}}_B^n + \Theta_2\, \mathrm{u}_B^{n+1})(\boldsymbol{a}\cdot\boldsymbol{\nabla}) N_B \right] d\Omega\, dt \right\}$$
$$+ \int_\Omega N_A \sum_{B=1}^{\mathrm{n_{np}}} N_B(\tilde{\mathrm{u}}_B^n - \mathrm{u}_B^n)\, d\Omega = 0.$$

The time-discontinuous formulation employing linear approximations in time is third-order accurate with respect to Δt and unconditionally stable, see Shakib and Hughes (1991). Like the standard Galerkin method, the time-discontinuous Galerkin method must be stabilized in space. For instance, a least-squares approach can be employed to improve stability without compromising the accuracy. Two variants of a least-squares space–time methodology are described in the next sections.

3.10.2 Time-discontinuous least-squares formulation

Consider again the homogeneous form of the linear convection equation (3.4a). As before, the finite element approximations are assumed to be discontinuous in time and continuous in space. However, instead of the Galerkin formulation, a least-squares approach is considered, based upon the following space–time quadratic functional:

$$\iint_{Q^n} (u_t + \boldsymbol{a}\cdot\boldsymbol{\nabla} u)^2\, d\Omega\, dt.$$

This functional is minimized with respect to a variation of the time-dependent unknown $u(\boldsymbol{x}, t)$. The jump condition enforcing continuity across the slab interfaces must also be added. Finally, using the trial and test spaces defined by (3.62), we obtain the following least-squares weighted residual formulation: for $n = 0, 1, \ldots, \mathrm{n_{st}} - 1$, find $u^h \in \mathcal{S}_n^h$, such that for all $w^h \in \mathcal{V}_n^h$,

$$\iint_{Q^n} (w_t^h + \boldsymbol{a}\cdot\boldsymbol{\nabla} w^h)(u_t^h + \boldsymbol{a}\cdot\boldsymbol{\nabla} u^h)\, d\Omega\, dt$$
$$+ \frac{1}{\Delta t} \int_\Omega w^h(t_+^n)\big(u^h(t_+^n) - u^h(t_-^n)\big)\, d\Omega = 0. \quad (3.64)$$

This time-discontinuous least-squares formulation is also third-order accurate with respect to Δt when a linear-in-time approximation is employed.

3.10.3 Space–time Galerkin/Least-squares formulation

To conclude the presentation of space–time finite element methods for convection problems, we shall describe the space–time Galerkin/Least-squares method proposed by Shakib and Hughes (1991). For the homogeneous convection equation (3.4a), the space–time Galerkin/Least-squares weighted residual formulation becomes: for

$n = 0, 1, \dots, n_{\mathrm{st}} - 1$, find $u^h \in \mathcal{S}_n^h$, such that for all $w^h \in \mathcal{V}_n^h$,

$$\iint_{Q^n} w^h \left(u_t^h + \boldsymbol{a} \cdot \boldsymbol{\nabla} u^h\right) d\Omega\, dt + \iint_{Q^n} \left(w_t^h + \boldsymbol{a} \cdot \boldsymbol{\nabla} w^h\right) \tau \left(u_t^h + \boldsymbol{a} \cdot \boldsymbol{\nabla} u^h\right) d\Omega\, dt + \int_\Omega w^h(t_+^n) \left(u^h(t_+^n) - u^h(t_-^n)\right) d\Omega = 0, \quad (3.65)$$

where $u^h(\boldsymbol{x}, t_-^0) = u(\boldsymbol{x}, 0)$. The first and the last integrals are the same as in the time-discontinuous Galerkin formulation (3.63). As before, the last integral is a jump condition. The second integral is the least-squares operator already encountered in (3.64). Parameter τ was already discussed in Section 2.4.3, but here it is extended to transient problems and particularized for pure convection following Shakib (1989)

$$\tau = \left(\left(\frac{2}{\Delta t}\right)^2 + \left(\frac{2a}{h}\right)^2 \right)^{-1/2}. \quad (3.66)$$

The stability and accuracy analysis performed by Shakib and Hughes (1991) indicates that for linear-in-time approximations the method is third-order accurate with respect to Δt and unconditionally stable.

3.11 APPLICATIONS AND SOLVED EXERCISES

Several test problems describing purely convective transport are presented in this section to confirm the accuracy characteristics of the finite element schemes discussed in the present chapter.

Tables 3.5 and 3.6 summarize the amplification factors and the stability properties of the most relevant classical schemes discussed here for a uniform linear/bilinear mesh.

3.11.1 Propagation of a cosine profile

A simple 1D problem is proposed to illustrate and compare the performance of explicit and implicit schemes. The convection equation (3.27) is solved over the spatial interval $]0, 1[$ considering the following initial:

$$u(x, 0) = \begin{cases} \frac{1}{2}\left(1 + \cos(\pi(x - x_0)/\sigma)\right) & \text{if } |x - x_0| \leq \sigma, \\ 0 & \text{otherwise,} \end{cases}$$

and boundary condition: $u(0, t) = 0$ for $t \geq 0$, where $x_0 = 0.2$ and $\sigma = 0.12$.

Explicit methods. The exact solution of equation (3.27) with $a = 1$ corresponds to the translation to the right of the initial profile at unit speed. Figure 3.12 compares

Table 3.5 Amplification factor for different schemes.

Scheme	Amplification factor
CN	$G_{\mathrm{CN}}(\xi, C) = \dfrac{1 - \frac{2}{3}\sin^2\frac{\xi}{2} - i\frac{1}{2}C\sin\xi}{1 - \frac{2}{3}\sin^2\frac{\xi}{2} + i\frac{1}{2}C\sin\xi}$
TG2	$G_{\mathrm{TG2}}(\xi, C) = \dfrac{1 - (\frac{2}{3} + 2C^2)\sin^2\frac{\xi}{2} - iC\sin\xi}{1 - \frac{2}{3}\sin^2\frac{\xi}{2}}$
TG3	$G_{\mathrm{TG3}}(\xi,\, C) = \dfrac{1 - \frac{2}{3}(1 + 2C^2)\sin^2\frac{\xi}{2} - iC\sin\xi}{1 - \frac{2}{3}(1 - C^2)\sin^2\frac{\xi}{2}}$
TG3-2S	$G_{\mathrm{TG3\text{-}2S}}(\xi, C, \alpha) = \dfrac{1 - \frac{2}{3}\sin^2\frac{\xi}{2} - 2C^2\,\tilde{G}(\xi, C, \alpha)\,\sin^2\frac{\xi}{2} - iC\sin\xi}{1 - \frac{2}{3}\sin^2\frac{\xi}{2}}$
CJ	$G_{\mathrm{CJ}}(\xi, C, \theta) = \dfrac{1 - 2\big(\frac{1}{3} + 2(1-\theta)\theta C^2\big)\sin^2\frac{\xi}{2} - iC\sin\xi}{1 - 2(\frac{1}{3} - 2\theta^2 C^2)\sin^2\frac{\xi}{2}}$

Table 3.6 Stability limits for different schemes.

Scheme	Stability limits	
	1D	2D
CN	unconditional stability	
TG2	$C^2 \leq 1/3$	$c_x^{2/3} + c_y^{2/3} \leq (1/3)^{1/3}$
TG3	$C^2 \leq 1$	$c_x^{2/3} + c_y^{2/3} \leq 1$
TG3-2S	$C^2 \leq 3/4$	$c_x^2 + c_y^2 \leq 3/4$
CJ	unconditional stability for $1/2 \leq \theta \leq 1$	

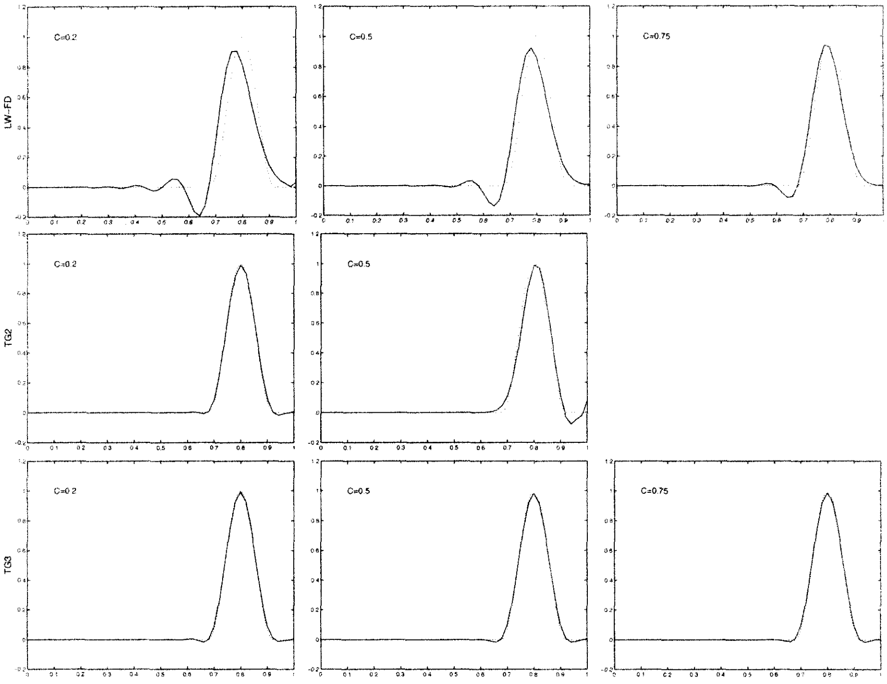


Fig. 3.12 Propagation of a cosine profile (explicit methods): comparison between the exact solution (dotted line) and the Lax–Wendroff finite element solution with diagonal mass (LW-FD), consistent mass (TG2) and the third-order Taylor–Galerkin solution (TG3).

the numerical solutions obtained at time $t = 0.6$ using a mesh of 50 uniform linear elements and different values of the Courant number C.

The problem is solved using in succession with:

- the second-order Lax–Wendroff finite element method is combined with a diagonal mass representation (LW-FD);
- the Lax–Wendroff finite element method is combined with a consistent mass representation (TG2);
- the third-order explicit Taylor–Galerkin scheme TG3.

We refer to Section 3.4.2.4 for the Galerkin formulation of the Lax–Wendroff method and to Section 3.6.2 for the Taylor–Galerkin method.

Figure 3.12 shows that both schemes using a consistent mass matrix exhibit a better phase accuracy than the Lax–Wendroff scheme combined with a diagonal mass representation (LW-FD). Note that the Lax–Wendroff scheme with consistent mass representation (TG2) cannot be operated with $C^2 > 1/3$. Moreover, it shows a phase lead at $C = 1/2$.

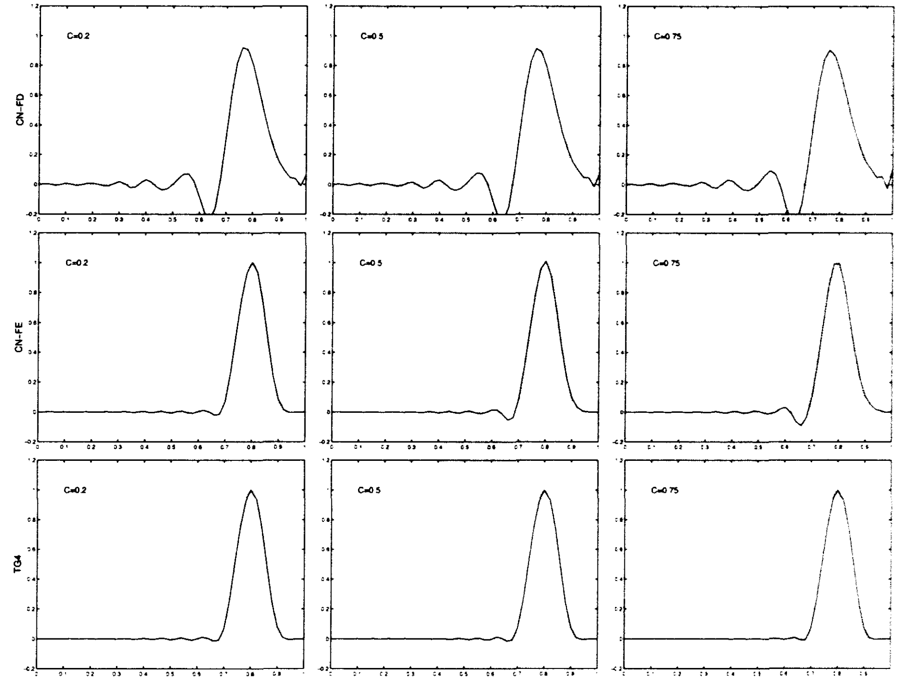


Fig. 3.13 Propagation of a cosine profile (implicit methods): comparison between the exact solution (dotted line) and the Crank–Nicolson solution with diagonal mass (CN-FD), consistent mass (CN-FE) and the fourth-order Taylor–Galerkin solution (TG4).

These findings are fully consistent with the phase error diagrams in Figure 3.3. As regards the third-order accurate TG3 scheme, we note that it has a rather uniform phase accuracy over the entire stable interval $0 < C < 1$ of the Courant number. This is in agreement with the phase–speed characteristics reported in Figure 3.6.

Implicit methods. Figure 3.13 shows the results of the same convection problem obtained with Galerkin and three implicit schemes, namely:

- the second-order Crank–Nicolson method combined with linear elements and a diagonal mass (CN-FD);
- the same method but with a consistent mass representation (CN-FE);
- the fourth-order Taylor–Galerkin method (TG4) resulting from the combination of the Harten and Tal-Ezer time scheme in equation (3.58) with linear elements and a consistent mass representation.

The Galerkin formulation of the Crank–Nicolson method is described in Section 3.4.2.3. For the TG4 method, the following variational equation is obtained from equations (3.27) and (3.58):

$$\int_0^L \left(w\left(\Delta u + \frac{a\Delta t}{2}\frac{\partial \Delta u}{\partial x} \right) - \frac{a^2\Delta t^2}{12}\frac{\partial w}{\partial x}\frac{\partial \Delta u}{\partial x} \right) dx = -\int_0^L wa\Delta t \frac{\partial u^n}{\partial x} dx,$$

where $\Delta u = u^{n+1} - u^n$. Similarly to the explicit methods, the schemes using finite elements with a consistent mass matrix show a superior phase accuracy. This is in agreement with the phase–speed characteristics reported in Figure 3.4. Observe that the phase error for the Crank–Nicolson/consistent finite element scheme increases with the time-step size, while the fourth-order accurate implicit method shows an excellent phase response at all values of the Courant number.

3.11.2 Travelling wave package

This example tries to visualize the influence of numerical damping and phase lag for different explicit and implicit methods. The conclusions drawn from the graphical representations of equations (3.36) (accuracy analysis) are recovered in this practical example. An initial condition is convected at a unitary speed over a uniform mesh of linear elements. It is defined as the product of a square wave an a sinusoidal wave,

$$u(x,0) = \exp(-(\alpha_1(x+1-\beta_1)^{n_1}))\exp(-(\alpha_2(1-x-\beta_2)^{n_2}))\sin(\kappa x),$$

with $\alpha_1 = \alpha_2 = 7$, $n_1 = n_2 = 30$, $\beta_1 = 1/\sqrt{2}$, $\beta_2 = 0$, and $\kappa = 20\pi$. The element size is chosen such that the sinusoidal wave induces a dimensionless wave number $\xi = kh = \pi/4$. This value corresponds to the minimum number of elements (eight) per wave length to accurately represent the sinus. On one hand, this high frequency at $\xi = \pi/4$ will be affected by the phase errors. On the other, the amplitude of this wave (controlled by the square function induced by the exponentials) will be affected by the amplitude error.

The time increment for the explicit methods has been chosen such that the Courant number is 90% of the stability limit, Section 3.5.2 and Table 3.6. This criterion is usually employed in practice. Implicit methods are stable for every time increment. Their precision degrades for large values of the Courant number, C, and, usually, at $C = 1$ they present their maximum accuracy. Thus, for comparison purposes, the time increment is chosen such that $C = 0.9$.

Figure 3.14 shows the results for different methods. Phase errors clearly affect the second-order, explicit, methods: Lax–Wendroff with a diagonal mass matrix representation (LW-FD), that is finite differences, Lax–Wendroff with the consistent mass matrix (TG2) and leap-frog. As noted earlier, the diagonal matrix representation increases the stability range (LW-FD uses a larger time step) but accuracy is compromised: important amplitude errors. The third-order explicit Taylor–Galerkin scheme (TG3) shows its superior behavior, in particular with respect to phase errors, due to its higher-order time accuracy.

There is not a clear improvement with second-order implicit methods: Crank–Nicolson with diagonal matrix representation (CN-FD), Crank–Nicolson with a consistent mass matrix (CN-FE) and the least-squares Crank–Nicolson (CJ), see Section

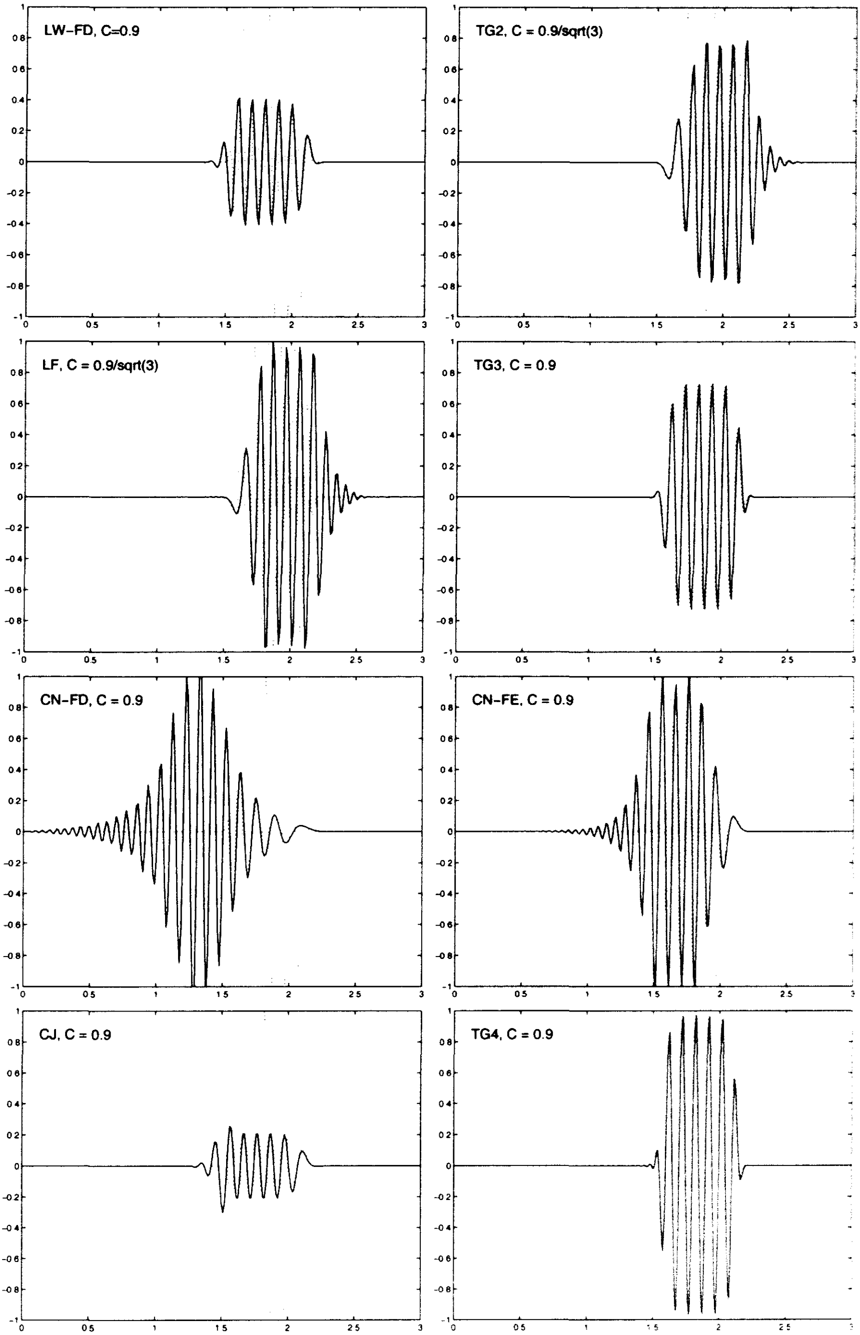


Fig. 3.14 Comparison between different explicit and implicit methods for $\xi = \pi/4$.

3.8.1. They all show large phase errors and the least-squares formulation, as expected, is over-diffusive. The one-step fourth-order Taylor–Galerkin method (TG4) introduced in the previous example, shows very accurate results because of its high-order time accuracy.

This example, however, represents a limit case: $\xi = kh = \pi/4$. Figure 3.15 shows results of the same problem (in particular, same mesh) with a different initial condition: the parameter κ is modified, $\kappa = 10\pi$, in order to have $\xi = kh = \pi/8$. All methods show a clear improvement, except for Crank–Nicolson with diagonal matrix representation (CN-FD), which still presents large phase errors, all the other results seem acceptable. The accuracy of high-order schemes, TG3 and TG4, is still remarkable.

3.11.3 The rotating cone problem

This classical test problem for 2D convection schemes considers the convection of a product-cosine hill in a pure rotation velocity field. The initial data is

$$u(\boldsymbol{x},0) = \begin{cases} \frac{1}{4}(1+\cos\pi X_1)(1+\cos\pi X_2) & \text{if} X_1^2 + X_2^2 \leq 1, \\ 0 & \text{otherwise,} \end{cases}$$

where $\boldsymbol{X} = (\boldsymbol{x} - \boldsymbol{x}_0)/\sigma$, and the boundary condition is $u = 0$ on Γ^{in}. The initial position of the center and the radius of the cosine hill are $\boldsymbol{x}_0$ and σ, respectively. In the examples they are chosen as $\boldsymbol{x}_0 = (\frac{1}{6}, \frac{1}{6})$ and $\sigma = 0.2$. The convection field is a pure rotation one with unit angular velocity, namely $\boldsymbol{a}(\boldsymbol{x}) = (-x_2, x_1)$. A uniform mesh of 30 × 30 four-node elements over the unit square $[-\frac{1}{2}, \frac{1}{2}] \times [-\frac{1}{2}, \frac{1}{2}]$ is employed in the calculations.

Explicit methods. The numerical solutions are shown in Figure 3.16 after a full revolution completed in 200 time steps ($\Delta t = 2\pi/200$). They have been computed using three explicit finite element schemes in the Galerkin formulation, namely:

- the second-order Lax–Wendroff method combined with bilinear elements and a diagonal mass representation;
- the Lax–Wendroff method combined with bilinear elements and a consistent mass representation (TG2);
- the third-order explicit Taylor–Galerkin scheme (TG3).

The weak form for the Lax–Wendroff method is the particularization of equation (3.29) for the case of $s = 0$ and $h = 0$, namely

$$\left(w, \frac{\Delta u}{\Delta t}\right) = \left(\boldsymbol{a}\cdot\boldsymbol{\nabla}w, u^n - \frac{\Delta t}{2}(\boldsymbol{a}\cdot\boldsymbol{\nabla})u^n\right) - \left((\boldsymbol{a}\cdot\boldsymbol{n})w, u^n - \frac{\Delta t}{2}(\boldsymbol{a}\cdot\boldsymbol{\nabla})u^n\right)_{\Gamma^{out}}.$$

Similarly, from (3.43) we determine the weak form for the TG3 method

$$\left(w, \frac{\Delta u}{\Delta t}\right) + \frac{\Delta t^2}{6}\left(\boldsymbol{a}\cdot\boldsymbol{\nabla}w, \boldsymbol{a}\cdot\boldsymbol{\nabla}\frac{\Delta u}{\Delta t}\right) - \frac{\Delta t^2}{6}\left((\boldsymbol{a}\cdot\boldsymbol{n})w, \boldsymbol{a}\cdot\boldsymbol{\nabla}\frac{\Delta u}{\Delta t}\right)_{\Gamma^{out}}$$
$$= \left(\boldsymbol{a}\cdot\boldsymbol{\nabla}w, u^n - \frac{\Delta t}{2}(\boldsymbol{a}\cdot\boldsymbol{\nabla}u^n)\right) - \left((\boldsymbol{a}\cdot\boldsymbol{n})w, u^n - \frac{\Delta t}{2}(\boldsymbol{a}\cdot\boldsymbol{\nabla}u^n)\right)_{\Gamma^{out}}.$$

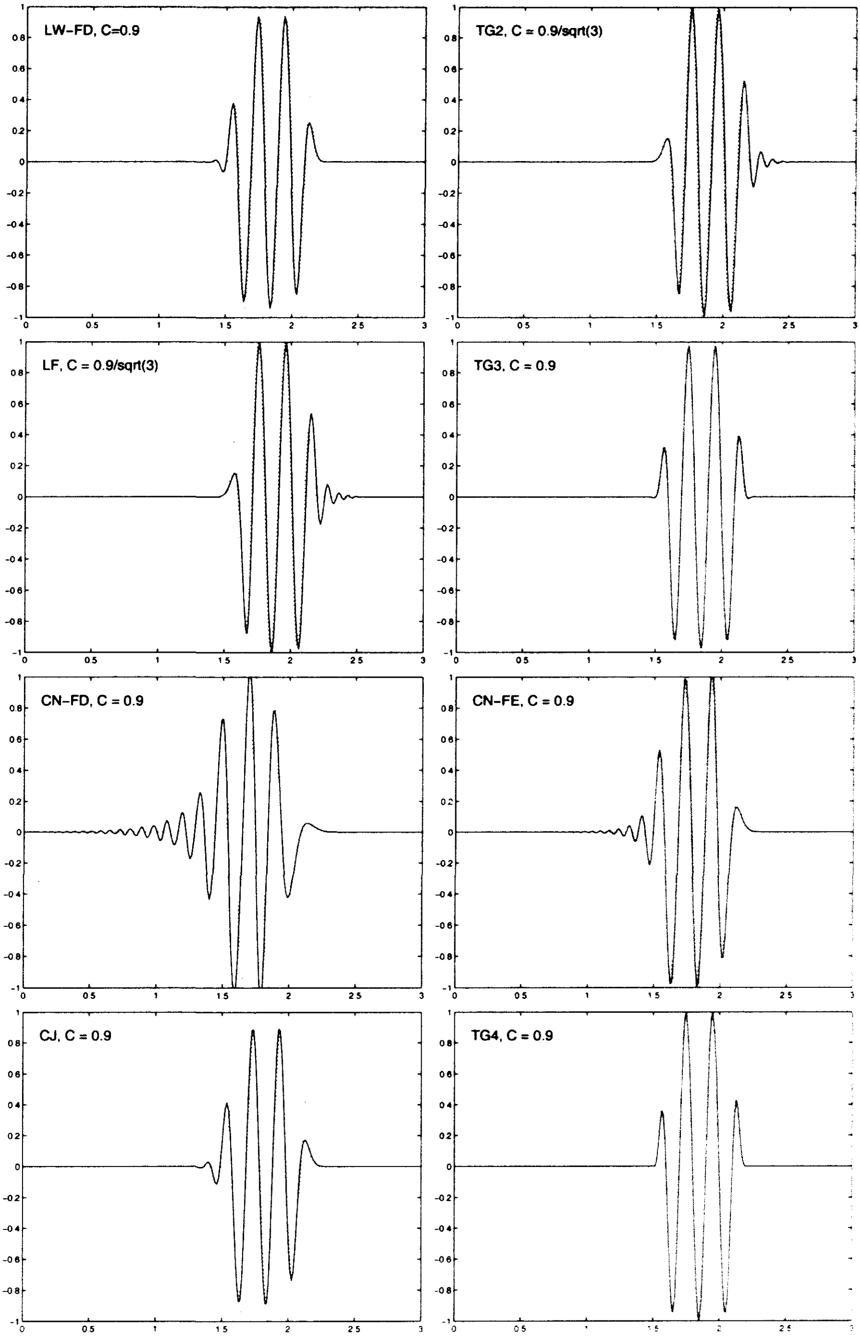


Fig. 3.15 Comparison between different explicit and implicit methods for $\xi = \pi/8$.

To compare the accuracy of the various explicit methods, the maximum and minimum values of the computed solutions are provided in Figure 3.16. The greater accuracy of the finite element schemes employing a consistent (TG2) or generalized (TG3) mass matrix is clearly apparent. Admittedly, the consistent finite element schemes are computationally more expensive than the Lax–Wendroff scheme using a diagonal mass matrix, because the solution of a banded (symmetric) linear system is required at each time step.

Implicit methods. We repeat the same problem using now the implicit Crank–Nicolson method. The first tests are performed using the Galerkin finite element formulation. The associated weak form is given from (3.28) for s and h equal to zero, namely

$$\Big(w, \frac{\Delta u}{\Delta t}\Big) - \frac{1}{2}\big(\boldsymbol{\nabla} w, \boldsymbol{a}\,\Delta u\big) + \frac{1}{2}\big((\boldsymbol{a}\cdot\boldsymbol{n})w, \Delta u\big)_{\Gamma^{out}} \\ = \big(\boldsymbol{\nabla} w, \boldsymbol{a}\,u^n\big) - \big((\boldsymbol{a}\cdot\boldsymbol{n})w, u^n\big)_{\Gamma^{out}}.$$

Three increasing values of the time step are employed, which correspond to a complete revolution of the cone in, respectively, 120, 60 and 30 time steps. In this way, we shall appraise the behavior of the Crank–Nicolson/Galerkin method beyond the stability limit of the explicit TG3 scheme. The numerical results after one complete revolution of the cone are displayed in Figure 3.17. Note that the Crank–Nicolson scheme with the Galerkin formulation represents a non-dissipative method in pure convection. Moreover, the phase accuracy of the method decreases when the time step is increased. As a result, significant non-physical oscillations develop as soon as the time-step size exceeds the stability limit of the explicit schemes.

These oscillations can be attenuated using a dissipative spatial formulation, such as the least-squares FEM of Carey and Jiang described in Section 3.8.1. From (3.57) we obtain the weak form for this method

$$\Big(\frac{w}{\Delta t} + \frac{1}{2}\boldsymbol{a}\cdot\boldsymbol{\nabla} w, \frac{\Delta u}{\Delta t} + \frac{1}{2}\boldsymbol{a}\cdot\boldsymbol{\nabla}\Delta u\Big) = -\Big(\frac{w}{\Delta t} + \frac{1}{2}\boldsymbol{a}\cdot\boldsymbol{\nabla} w, \boldsymbol{a}\cdot\boldsymbol{\nabla} u^n\Big).$$

Note that the dissipative effect of the least-squares approach increases with the square of the time step. Again, the rotating cone problem is solved using 120, 60 and 30 time increments for a full rotation. Figure 3.18 reports the results which can be compared with those in Figure 3.17 for the Galerkin approach. The dissipative nature of the least-squares formulation results in lower values of the cone height with respect to the Galerkin results in Figure 3.17. Also, reduced minimum values of the computed solutions are obtained with the least-squares approach.

3.11.4 Propagation of a steep front

Crank–Nicolson methods. This 1D problem considers the convection at unit speed of discontinuous initial data. The discontinuity occurs over one element and is initially located at position $x = 0.2$ of the computational domain $]0, 1[$. The inlet condition $u(0, t) = 1$ is imposed. A mesh of uniform linear elements of size $h = 1/50$

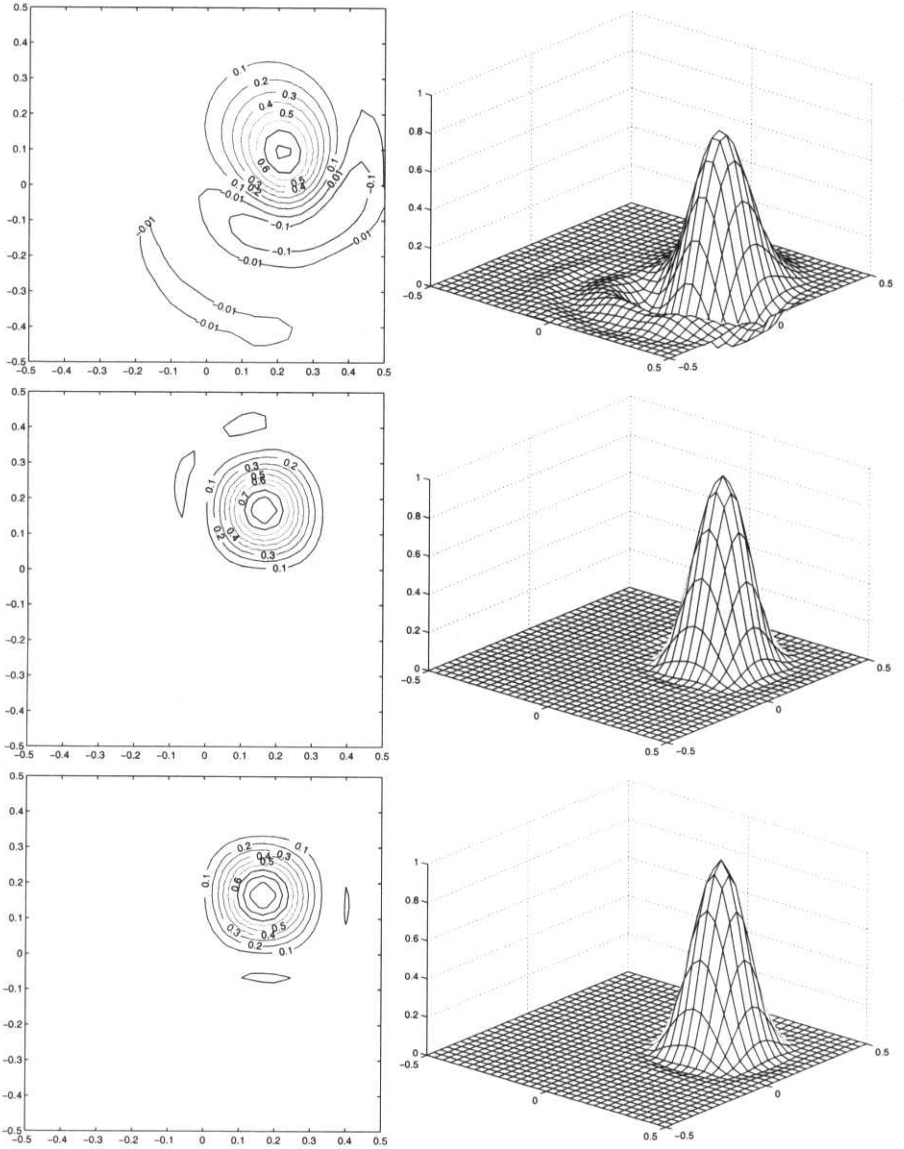


Fig. 3.16 Convection of a cosine hill in a pure rotation velocity field: comparison of the numerical solutions after a complete revolution calculated with $\Delta t = 2\pi/200$ by means of (Top): Lax–Wendroff/diagonal mass scheme ($u_{\max} = 0.8186, u_{\min} = -0.1774$); (Middle): TG2 ($u_{\max} = 0.9830, u_{\min} = -0.0186$); and (Bottom): TG3 ($u_{\max} = 0.9835, u_{\min} = -0.0148$).

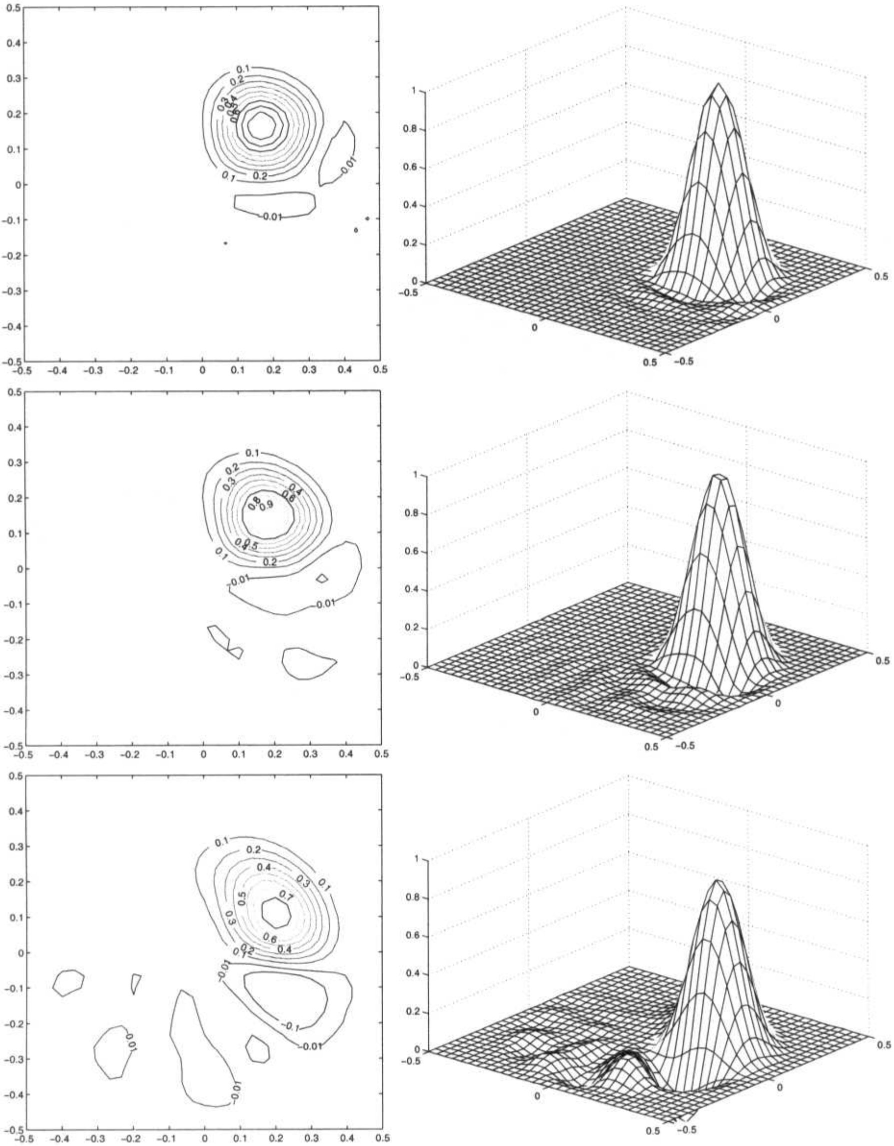


Fig. 3.17 Convection of a cosine hill in a pure rotation velocity field using the Crank–Nicolson/Galerkin method: comparison of the numerical solutions after a complete revolution computed with (Top): $\Delta t = 2\pi/120$, $u_{\max} = 0.9969$, $u_{\min} = -0.0454$; (Middle): $\Delta t = 2\pi/60$, $u_{\max} = 0.9691$, $u_{\min} = -0.1096$; and (Bottom): $\Delta t = 2\pi/30$, $u_{\max} = 0.8931$, $u_{\min} = -0.2694$.

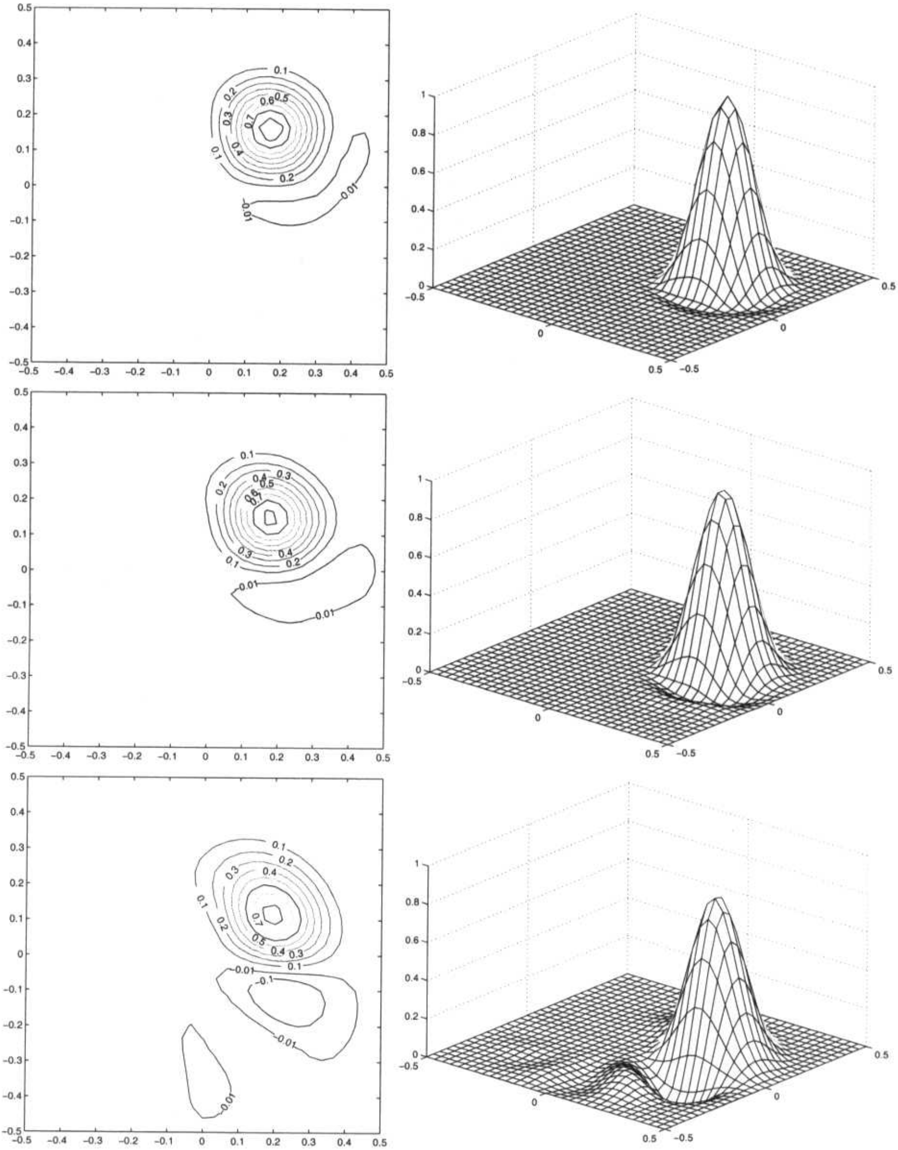


Fig. 3.18 Convection of a cosine hill in a pure rotation velocity field using the Crank–Nicolson/least-squares method: comparison of the numerical solutions after a complete revolution computed with (Top): $\Delta t = 2\pi/120$, $u_{\max} = 0.9691$, $u_{\min} = -0.0266$; (Middle): $\Delta t = 2\pi/60$, $u_{\max} = 0.9165$, $u_{\min} = -0.0616$; and (Bottom): $\Delta t = 2\pi/30$, $u_{\max} = 0.8370$, $u_{\min} = -0.2009$.

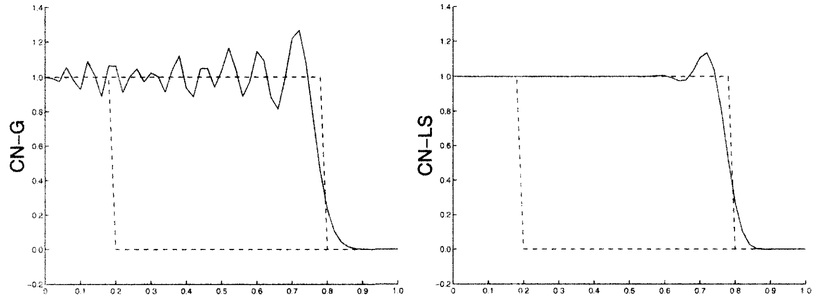


Fig. 3.19 Propagation of a steep front using the Crank–Nicolson scheme with the Galerkin method (left) and with the least-squares method (right). The Courant number is $C = 0.75$ in both cases. The graphs show the computed solutions at time $t = 0.60$, together with the initial profile and the exact solution.

is employed. The results at time $t = 0.6$ are displayed in Figure 3.19 together with the initial data. They were obtained (for a Courant number $C = 0.75$) by combining the Crank–Nicolson scheme (with linear elements) and (1) the Galerkin formulation, and (2) the least-squares formulation of Carey and Jiang (1988), see Section 3.8.1.

Note that Crank–Nicolson with least-squares succeeds in removing the spurious oscillations induced by the Galerkin formulation over the whole computational domain. Since Crank–Nicolson is not a monotone scheme, residual oscillations remain at the front. These could be removed using nonlinear viscosity, which is added at the front to render the scheme locally first-order accurate, see Section 3.7.

Discontinuous Galerkin in space and the second-order two-step Lax–Wendroff method in time. The steep-front problem is again solved with a uniform mesh of 50 linear elements. The two-step TG2 method integrates in time the semi-discrete equations resulting from the discontinuous Galerkin method. The method follows the two-step rationale presented in Section 3.6.4, see also Section 4.2.3.2:

$$\begin{aligned} u^{n+1/2} &= u^n + \frac{\Delta t}{2}\, u_t^n &&= u^n - \frac{\Delta t}{2}\, a\, u_x^n, \\ u^{n+1} &= u^n + \Delta t\, u_t^{n+1/2} &&= u^n - \Delta t\, a\, u_x^{n+1/2}. \end{aligned}$$

Mid-interval values of the unknown are computed within each element using a diagonal mass representation. Denoting by indices i and j the nodal values of the unknown u in a typical linear element, we have

$$\left(u_i^+\right)^{n+1/2} = \left(u_i^+\right)^n - \frac{a\Delta t}{2h}\left(u_j^- - u_i^+\right)^n, \; \left(u_j^-\right)^{n+1/2} = \left(u_j^-\right)^n - \frac{a\Delta t}{2h}\left(u_j^- - u_i^+\right)^n.$$

To compute the end-of-step values, we use a consistent mass representation and obtain the following equations from expression (3.61):

$$
\begin{aligned}
\frac{h}{6}\left(2\dot{u}_i^+ + \dot{u}_j^-\right) &= -\frac{a}{2}\left(u_i^+ + u_j^-\right)^{n+1/2} + a\left(u_i^-\right)^{n+1/2},\\
\frac{h}{6}\left(\dot{u}_i^+ + 2\dot{u}_j^-\right) &= \frac{a}{2}\left(u_i^+ + u_j^-\right)^{n+1/2} - a\left(u_j^-\right)^{n+1/2},
\end{aligned}
$$

from which we obtain upon inversion of the element mass matrix:

$$
\begin{aligned}
\left(u_i^+\right)^{n+1} &= \left(u_i^+\right)^n + \frac{a\Delta t}{h}\left(4u_i^- - 3u_i^+ - u_j^-\right)^{n+1/2},\\
\left(u_j^-\right)^{n+1} &= \left(u_j^-\right)^n + \frac{a\Delta t}{h}\left(3u_i^+ - 2u_i^- - u_j^-\right)^{n+1/2}.
\end{aligned}
$$

As explained in detail in Section 4.4.2, artificial diffusion is needed close to zones of sharp gradients to avoid spurious oscillations in the numerical solution. Here, artificial diffusion consists in adding the following terms to the r.h.s. of the second-step equations:

$$
\begin{aligned}
&\frac{\nu_e\Delta t}{h^2}\Big((u_j^- - u_i^+)^n - (u_i^- - u_{i-1}^+)^n\Big) \quad \text{at node } i,\\
&\frac{\nu_e\Delta t}{h^2}\Big((u_{j+1}^- - u_j^+)^n - (u_j^- - u_i^+)^n\Big) \quad \text{at node } j.
\end{aligned}
$$

Note that these terms emanate from the usual Galerkin term for the diffusion operator:

$$
\sum_{\Omega^e}\int_{\Omega^e} \nu_e \frac{\partial w}{\partial x}\frac{\partial u}{\partial x}dx.
$$

Zero diffusive flux is imposed at both extremes of the computational domain. The coefficient of numerical diffusion ν_e is computed in each element Ω^e as a function of the local value of the second derivative of the unknown: $\nu_e = \dfrac{ah}{4}\max(d_i, d_j)$, where

$$
d_i = \frac{u_j^- - u_i^+ - u_i^- + u_{i-1}^+}{u_j^- + u_i^+ + u_i^- + u_{i-1}^+}, \qquad d_j = \frac{u_{j+1}^- - u_j^+ - u_j^- + u_i^+}{u_{j+1}^- + u_j^+ + u_j^- + u_i^+}.
$$

Note that we are actually solving a convection–diffusion problem, instead of a pure convection one. The stability limit of the two-step Lax–Wendroff method applied to the steep-front problem was found to be $C = 0.30$. This reduced stability is compensated by the extremely low cost per time step of the discontinuous Galerkin method. The results obtained with $C = 0.30$ are displayed in Figure 3.20. There are no spurious oscillations, but the discontinuity is spread over a few elements through the action of the added diffusivity.

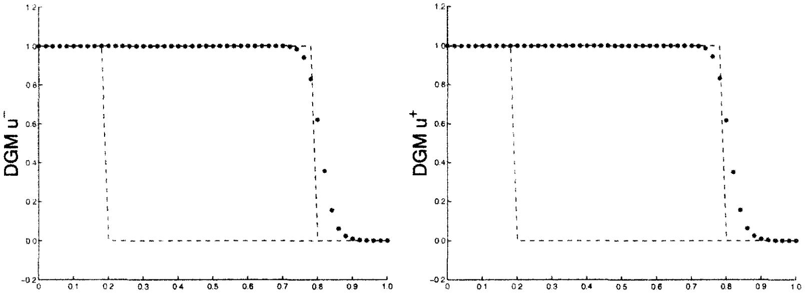


Fig. 3.20 Propagation of a steep front using the discontinuous Galerkin method and the two-step Lax–Wendroff method in time. The Courant number is $C = 0.30$. The graphs show the computed solutions u^- (left) and u^+ (right) at time $t = 0.60$, together with the initial profile and the exact solution.

Space–time methods. We repeat the solution of the problem using three space–time methods, namely:

- the time-discontinuous Galerkin formulation described in Section 3.10.1;
- the time-discontinuous least-squares formulation described in Section 3.10.2;
- the space–time Galerkin/Least-squares formulation described in Section 3.10.3.

Linear finite element approximations are employed in both space and time. Two values of the Courant number are used with these unconditionally stable methods, namely, $C = 1$, and $C = 2$.

Time-discontinuous Galerkin: the developments of Section 3.10.1 induce the following partitioned matrix system for the nodal unknowns $\mathbf{u}^{n+1}$ and $\mathbf{u}^{n^+}$:

$$\begin{aligned}\left(\mathbf{M} + \frac{2}{3}\Delta t\mathbf{C}\right)\mathbf{u}^{n+1} - \left(\mathbf{M} - \frac{1}{3}\Delta t\mathbf{C}\right)\mathbf{u}^{n^+} &= 0\\ \left(\mathbf{M} + \frac{1}{3}\Delta t\mathbf{C}\right)\mathbf{u}^{n+1} + \left(\mathbf{M} + \frac{2}{3}\Delta t\mathbf{C}\right)\mathbf{u}^{n^+} &= 2\,\mathbf{M}\,\mathbf{u}^{n^-},\end{aligned}$$

where $\mathbf{M}$ denotes the mass matrix and $\mathbf{C}$ the convection matrix. The condition $u^{n+1}(1) = u^{n^+}(1) = 1$ is then enforced to satisfy the inlet condition. Note that this third-order accurate and unconditionally stable method requires the solution of an algebraic system double the size of usual time-stepping algorithms. Results are displayed in Figure 3.21. Once again, oscillations extending over the whole computational domain characterize the Galerkin formulation. Note that the level of the spurious oscillations appears to decrease when the Courant number is increased.

Time-discontinuous least-squares: here, we use the pure least-squares approach described in Section 3.10.2. The partitioned matrix system for the nodal unknowns $\mathbf{u}^{n+1}$ and $\mathbf{u}^{n^+}$ is now obtained in the form

$$\begin{aligned}\left(\mathbf{M} + \frac{1}{3}a^2\Delta t^2\mathbf{K}\right)\mathbf{u}^{n+1} - \left(\mathbf{M} - \Delta t\mathbf{C} + \frac{1}{6}a^2\Delta t^2\mathbf{K}\right)\mathbf{u}^{n^+} &= 0\\ \left(-\mathbf{M} - \Delta t\mathbf{C} + \frac{1}{6}a^2\Delta t^2\mathbf{K}\right)\mathbf{u}^{n+1} + \left(2\mathbf{M} + \frac{1}{3}a^2\Delta t^2\mathbf{K}\right)\mathbf{u}^{n^+} &= \mathbf{M}\,\mathbf{u}^{n^-}.\end{aligned}$$

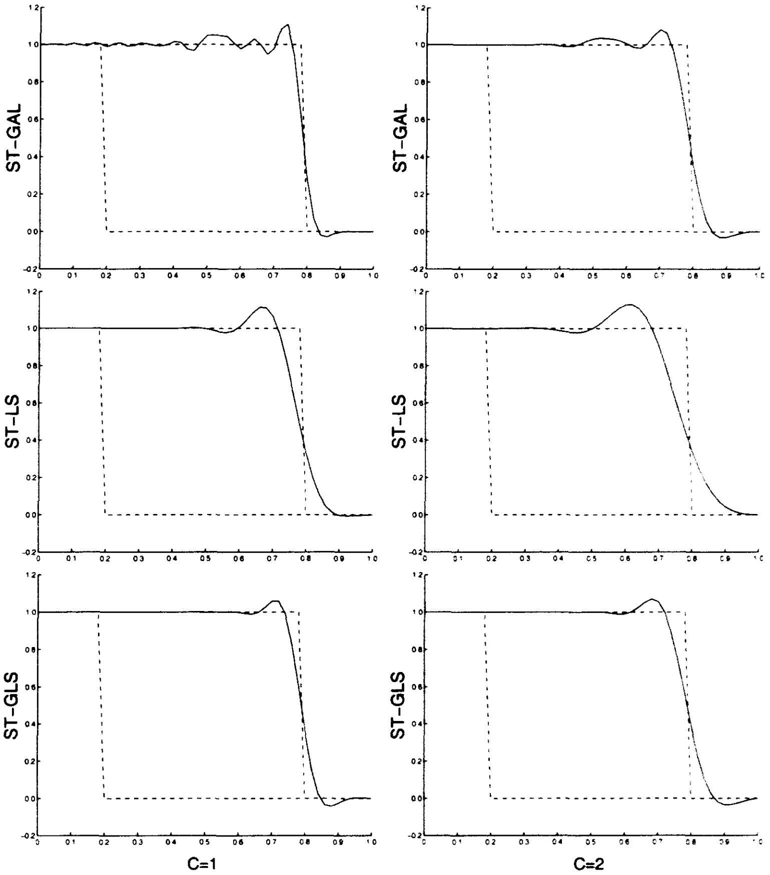


Fig. 3.21 Propagation of a steep front using space–time methods with Courant numbers $C = 1$ (left) and $C = 2$ (right). Different methods are employed: the space–time Galerkin (top), the space–time least-squares (middle), and the space–time Galerkin/Least-squares (bottom). The curves show the computed solutions at time $t = 0.60$, together with the initial profile and the exact solution.

Here, matrix $\mathbf{K}$ is a diffusivity matrix accounting for the numerical dissipation introduced by the least-squares method. Results displayed in Figure 3.21 show that pure least-squares is over diffusive and smears out excessively the steep front. In fact, from the previous equations, we see that the added diffusivity is proportional to the square of the Courant number.

Time-discontinuous Galerkin/Least-squares: we now follow the GLS approach for bilinear space–time elements described in Section 3.10.3. The partitioned matrix system for the nodal unknowns $\mathbf{u}^{n+1}$ and $\mathbf{u}^{n^+}$ now depends on the stabilization parameter τ. We have selected the value of τ as given in expression (3.66). In terms of the Courant number, this gives

$$\tau = \frac{\Delta t}{2\sqrt{1+C^2}}.$$

The partitioned matrix system for the nodal unknowns $\mathbf{u}^{n+1}$ and $\mathbf{u}^{n^+}$ is now obtained in the form:

$$\left(\left(1+\frac{2\tau}{\Delta t}\right)\mathbf{M} + \frac{2}{3}\Delta t\mathbf{C} + \frac{2}{3}\tau a^2 \Delta t\mathbf{K}\right)\mathbf{u}^{n+1} \\ - \left(\left(1+\frac{2\tau}{\Delta t}\right)\mathbf{M} - \left(\frac{1}{3}\Delta t + 2\tau\right)\mathbf{C} - \frac{1}{3}\tau a^2 \Delta t\mathbf{K}\right)\mathbf{u}^{n^+} = 0$$

$$\left(\left(1-\frac{2\tau}{\Delta t}\right)\mathbf{M} + \left(\frac{1}{3}\Delta t - 2\tau\right)\mathbf{C} + \frac{1}{3}\tau a^2 \Delta t\mathbf{K}\right)\mathbf{u}^{n+1} \\ + \left(\left(1+\frac{2\tau}{\Delta t}\right)\mathbf{M} + \frac{2}{3}\Delta t\mathbf{C} + \frac{2}{3}\tau a^2 \Delta t\mathbf{K}\right)\mathbf{u}^{n^+} = 2\,\mathbf{M}\,\mathbf{u}^{n^-}.$$

Figure 3.21 depicts the results. Time-discontinuous GLS clearly delivers the best solution for this steep-front problem. Note that the dissipation introduced by the GLS operator also increases with the value of the Courant number. In fact, from the previous equations and the definition of the stabilization parameter τ, we see that the added diffusivity of the GLS method is proportional to $C^2/\sqrt{1+C^2}$. The GLS method is nevertheless significantly less diffusive than pure least-squares.

4

Compressible Flow Problems

Engineering practice goes beyond scalar linear hyperbolic equations. This chapter is concerned with nonlinear systems of hyperbolic equations. An important and well-known problem of this class is the modeling of inviscid compressible flows governed by the Euler equations of gas dynamics. The nonlinear nature of the Euler equations typically leads to non-smooth solutions characterized by the presence of shocks. Various finite-element-based strategies are discussed for tracing transient solutions in the presence of flow discontinuities. The Eulerian description is used throughout the chapter, except for the finite element treatment of coupled fluid–structure problems where the Arbitrary Lagrangian–Eulerian (ALE) description is employed because it is more convenient.

4.1 INTRODUCTION

The development of numerical methods for simulating complex flow problems is of major importance in view of the numerous applications of fluid dynamics in many different areas of applied science and engineering. Numerical modeling of fluid dynamics problems is indeed of particular relevance to the aerospace and automotive industries, to the oil industry, in meteorology, hydrology, oceanography, protection of the environment, as well as in the safety assessment of industrial plants. In recent years, numerical simulation of flow problems has been extended to several new application areas and disciplines. These include coupled problems which combine fluid mechanics and electromagnetic theory (Maxwell equations), or fluid mechanics and

Finite Element Methods for Flow Problems. Jean Donea and Antonio Huerta
Copyright © 2003 John Wiley & Sons, Ltd.
ISBN: 0-471-49666-9

biomechanics, such as the modeling of arterial blood flow with the aim of providing guidance for surgery planning.

The equations governing fluid motion are generally quite complex and of a form strongly dependent upon the particular problem under investigation. Consider, for instance, the motion of a fluid under the effect of an explosion: the transient phenomenon is of very short duration and viscosity effects can be neglected. Moreover, a realistic modeling of such fast-transient dynamics problems requires the compressibility of the fluid to be taken into account. By contrast, if we are to analyze the flow of a visco-plastic fluid, such as, for instance, in the simulation of forming processes for fiber-reinforced thermo-plastics, viscosity will be the important factor, while the fluid can generally be considered as incompressible and its flow as steady. Given the great variety of fluid mechanics problems, it is advisable to focus attention on a particular class of problems at a time, rather than attempting to consider the problem of fluid motion under a general form.

The present chapter is devoted to the finite element treatment of compressible flow problems governed by the Euler equations of gas dynamics. It starts with a review in Section 4.2 of the properties of nonlinear hyperbolic equations. We underline the numerical difficulties introduced by the directional character of propagation of information in hyperbolic problems and by the possible generation of discontinuities in the solution, even starting from continuous initial data. Emphasis is then placed on the numerical treatment of non-smooth solutions by a weak formulation in which spatial derivatives are no longer acting on the problem variables but only on smooth weighting functions.

Systems of hyperbolic equations are then treated in Section 4.3 where the Euler equations governing compressible high-speed gas flow are introduced. We first discuss the basic properties of the Euler equations and the possible form of the associated boundary conditions according to the flow regime under consideration. Then, starting from the strong form of the conservation equations for mass, momentum and energy, we construct the associated weak variational forms which are the basis of the spatial discretization by the finite element method. Various options for the spatial discretization are considered. This includes continuous or discontinuous interpolation of all the conservation variables, as well as mixed representations where some variables are continuous and other are discontinuous across inter-element boundaries.

As already seen in Chapter 3, all second- and higher-order time discretization schemes are not monotone methods and generate oscillations in the vicinity of sharp solution gradients. These have to be damped by the addition of artificial dissipation terms. We discuss in Section 4.4 various spatial discretization techniques of the upwind type able to cope with the directional character of propagation of information in hyperbolic problems. Then, the finite element implementation of specific artificial viscosity techniques for the accurate representation of shocks and other flow discontinuities is discussed in Section 4.5. This includes the construction of so-called high-resolution schemes.

Most numerical methods designed for compressible flow problems present difficulties when applied to low-speed (low Mach number) flow situations. This is due to the fast propagation of pressure waves as flow conditions approach the incompressible limit. Finite element algorithms can be developed that work for both the compressible and the nearly incompressible regime. A summarized account is given in Section 4.6.

We then describe in Section 4.7 finite element models for coupled fluid–structure problems. The response of a linear elastic structure interacting with an acoustic fluid is treated first. Then, ALE finite element models are discussed for fluid–structure interaction in the nonlinear regime. Illustrative examples indicate the effectiveness of the ALE method for problems with moving boundaries and deforming fluid–structure interfaces. The chapter closes with the presentation in Section 4.8 of solved exercises illustrating in the simple context of one spatial dimension the finite element solution of Burgers' and Euler equations.

4.2 NONLINEAR HYPERBOLIC EQUATIONS

4.2.1 Scalar equations

In nonlinear hyperbolic problems discontinuities can be generated from continuous initial conditions. Such solutions cannot verify the partial differential equation in the classical sense, but they may satisfy a weak form of this equation. This leads us to introduce the concept of *generalized* or *weak solutions* which admit the presence of discontinuities provided a condition, called the *jump condition*, is verified. Weak solutions are not necessarily unique; an extra condition, the *entropy condition*, allows us to determine the physically correct solution. We shall briefly recall these concepts which are the basis of the numerical approximation of nonlinear hyperbolic equations. Specialized texts, such as for instance Hirsch (1990), LeVeque (1992), Quarteroni and Valli (1994) or Godlewski and Raviart (1996), should be consulted for a more complete exposition of the properties of nonlinear hyperbolic equations.

Let us start the discussion of nonlinear scalar equations by considering the 1D Cauchy problem

$$\begin{cases} u_t + f_x(u) = 0, \\ u(x,0) = u_0(x), \end{cases} \tag{4.1}$$

where $f(u)$ is a nonlinear function of the unknown u. As in previous chapters u_t represents the partial time derivative of the unknown u, while f_x indicates the spatial derivative of $f(u)$. A classical example of this class of equations is the inviscid *Burgers' equation*, where $f(u) = u^2/2$, so that equation (4.1) becomes

$$\begin{cases} u_t + \left(\dfrac{u^2}{2}\right)_x = 0, \\ u(x,0) = u_0(x), \end{cases} \quad \text{or in convective form:} \quad \begin{cases} u_t + u\,u_x = 0, \\ u(x,0) = u_0(x). \end{cases} \tag{4.2}$$

This is a nonlinear transport equation where the convection velocity is the solution u itself. Thus, the characteristics, see Section 3.3.1, satisfy the equation

$$\frac{dx}{dt} = u(x,t). \tag{4.3}$$

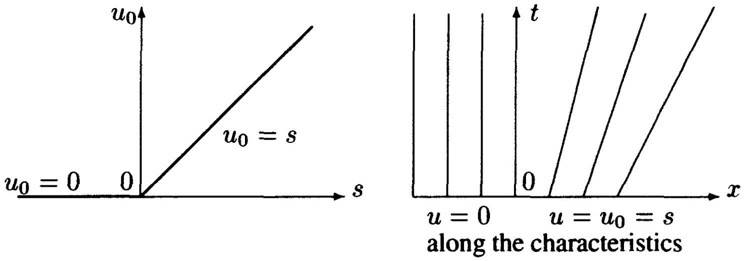


Fig. 4.1 Increasing continuous initial data produce a continuous solution to the inviscid Burgers' equation.

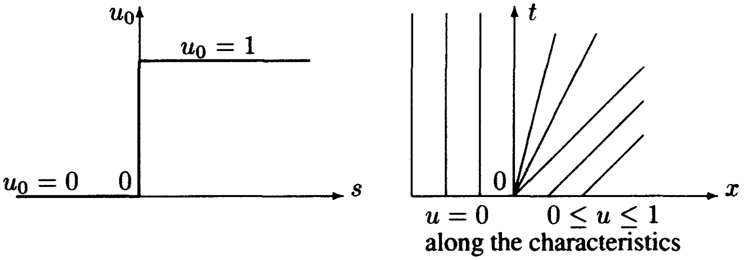


Fig. 4.2 Increasing discontinuous initial data produce a continuous solution to Burgers' equation.

The material derivative of u along a characteristic can be evaluated from Burgers' equation (4.2) and the characteristic (4.3)

$$\frac{du}{dt} = \frac{\partial u}{\partial t} + \frac{\partial u}{\partial x}\frac{dx}{dt} = u_t + u\, u_x = 0.$$

This shows that the solution u is constant along each characteristic. Thus, equation (4.3) implies that the slope dx/dt of the characteristics is constant. Therefore, the characteristics are straight lines. When the initial data are continuous this property implies that the characteristic equation is

$$x = s + u_0(s)\, t, \tag{4.4a}$$

for any typical space–time point $(s, 0)$ along the x-axis and with a slope $u_0(s)$ defined by the initial data. Moreover, along this characteristic

$$u = \text{constant} = u_0(s). \tag{4.4b}$$

Equations (4.4) provide a parametric representation of the solution $u(x,t)$ of Burgers' equation. This equation possesses a unique solution as long as the characteristics do not intersect. As shown in Figures 4.1 and 4.2, non-intersecting characteristics (and thus a unique solution) are obtained for increasing continuous initial data, as well as for increasing discontinuous initial data. By contrast, see Figure 4.3, if the

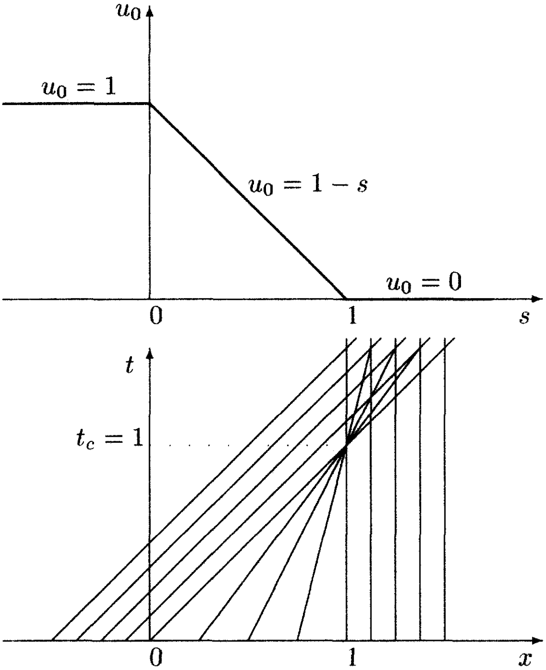


Fig. 4.3 With decreasing initial data the characteristics $x = s + u_0(s)\,t$ intersect causing the solution to become discontinuous.

initial data are such that the derivative $du_0(s)/ds$ is negative at some point of the x-axis, then there is a critical time t_c at which characteristics first cross, causing the solution to become discontinuous. Let us illustrate the possible situations by means of three examples.

First, for continuous initial data, reproduced in Figure 4.1,

$$u_0(s) = \begin{cases} 0 & \text{for } s \leq 0, \\ s & \text{for } s \geq 0, \end{cases}$$

relations (4.4) indicate that the solution of Burgers' problem (4.2) is given by

$$\begin{aligned} x &= s, & u &= 0 & &\text{for } s \leq 0,\ t > 0, \\ x &= s\,(1+t), & u &= s & &\text{for } s \geq 0,\ t > 0. \end{aligned}$$

Elimination of parameter s between these equations yields the solution u:

$$u(x,t) = \begin{cases} 0 & \text{for } x \leq 0,\ t > 0, \\ x/(1+t) & \text{for } x \geq 0,\ t > 0. \end{cases}$$

The corresponding characteristics are drawn in Figure 4.1.

Second, for the discontinuous initial data shown in Figure 4.2,

$$u_0(s) = \begin{cases} 0 & \text{for } s < 0, \\ 1 & \text{for } s > 0, \end{cases}$$

the solution is also determined from (4.4) and consists of three branches:

$$u(x,t) = \begin{cases} 0 & \text{for } x \leq 0,\ t > 0, \\ x/t & \text{for } 0 \leq x \leq t,\ t > 0, \\ 1 & \text{for } x \geq t,\ t > 0. \end{cases}$$

These expressions represent a continuous function $u(x,t)$ in the semi-plane $t > 0$. This continuous solution is called an *expansion fan* and the corresponding characteristics are drawn in Figure 4.2. This situation is representative of a supersonic expansion ramp. The important point to note here is that, due to the nonlinearity of Burgers' equation, discontinuous initial data may generate a continuous solution. This is, however, not always the case.

And third, in the case of decreasing initial data, even if continuously distributed, the situation is in fact quite different. Characteristics eventually cross and a discontinuity of the solution forms. Consider the following decreasing, and continuous, initial data:

$$u_0(s) = \begin{cases} 1 & \text{for } s \leq 0, \\ 1-s & \text{for } 0 \leq s \leq 1, \\ 0 & \text{for } s \geq 1. \end{cases}$$

As can be seen from the characteristics in Figure 4.3, the solution is unique for $t < t_c = 1$, but not beyond t_c. For $t \geq 1$ the solution is discontinuous because signals from the left portion of the considered domain travel faster than those from the right. This causes signals to pile up, thereby creating a discontinuity. This situation is typical for a compression profile and the analogy with a supersonic compression ramp should be noted.

In the more general case of the conservation law (4.1), it can be shown that if $u_0(s_1) > u_0(s_2)$ with $s_1 < s_2$, the two characteristics $x = s_1 + u_0(s_1)t$ and $x = s_2 + u_0(s_2)t$ intersect at time $t_c = [s_2 - s_1]/[u_0(s_1) - u_0(s_2)]$. Beyond time t_c a classical solution of the conservation law (4.1) no longer exists; however, a generalized (or weak) solution (which is discontinuous) can be defined.

4.2.2 Weak solutions and entropy condition

Weak solutions were already introduced in Chapter 1, see Remark 1.7, for steady problems. Here the same concept is extended to transient problems, in particular, for equation (4.1). The objective is, as in Section 1.5.4, to weaken the continuity requirements on the solution $u(x,t)$. In this case, since we are confronted with a first-order hyperbolic equation, the objective is to avoid differentiating $u(x,t)$. Thus, discontinuous solutions will be admissible. Denote by $C_0^1(\mathbb{R} \times [0,\infty[)$ the space of test functions $w(x,t)$ that are continuously differentiable and with compact support in the space–time domain $\mathbb{R} \times [0,\infty[$. That is, $w(x,t) = 0$ outside of some bounded set. Multiplying the conservation law (4.1) by $w(x,t)$, integrating over space and time, and integrating by parts (using Green's formula), yields the integral relationship

$$\int_0^\infty \int_{-\infty}^\infty \Big[w_t(x,t)\,u(x,t) \;+ w_x(x,t)\,f\big(u(x,t)\big)\Big]\,dx\,dt \\ = -\int_{-\infty}^\infty w(x,0)\,u_0(x)\,dx. \qquad (4.5)$$

Note that boundary terms disappear because $w(x,t)$ has compact support and thus vanishes at infinity (only the initial condition remains). A function $u(x,t)$ is called a *weak solution* of the conservation law (4.1) provided it is measurable and the integral relation (4.5) holds for all functions $w \in C_0^1(\mathbb{R} \times [0,\infty[)$.

Note that equation (4.5) does not include derivatives of the unknown u, nor of the flux function $f(u)$. Weak solutions are not necessarily unique in the sense that different solutions can be obtained with the same initial condition. The physically correct solution is the one that satisfies the so-called *entropy condition*, see for instance Oleĭnik (1957) for an in-depth analysis.

The correct solution can be determined by the *vanishing viscosity* approach. For instance, the physically correct weak solution of the inviscid Burgers' equation (4.2) corresponds to the solution of Burgers' equation when viscosity goes to zero. That is, the inviscid case (4.2) is seen as a model of

$$u_t^\epsilon + u^\epsilon u_x^\epsilon = \epsilon\, u_{xx}^\epsilon$$

with $u^\epsilon(x,0) = u_0(x)$, valid only for very small ϵ and smooth u^ϵ. In order to avoid working with the viscous equation there are other conditions easier to check and that will also allow us to determine the correct solution. They are called *entropy conditions*. This is also the reason for calling the viscous solution the entropy solution.

In order to establish the generic form of the entropy condition, we consider the conservation equation (4.1) with piecewise constant initial data with a single discontinuity. This is known as the *Riemann problem*. As an example, let us take Burgers' equation (4.2) with the following piecewise constant initial data:

$$u_0(s) = \begin{cases} u_l & \text{for } s < 0, \\ u_r & \text{for } s > 0. \end{cases}$$

When $u_l > u_r$, the unique weak solution is given by

$$u(x,t) = \begin{cases} u_l & \text{for } x < \sigma t \\ u_r & \text{for } x > \sigma t \end{cases} \qquad (4.6)$$

where

$$\sigma = \frac{u_l + u_r}{2} \qquad (4.7)$$

is the speed at which the discontinuity propagates. At each instant $t \geq 0$ the solution has two constant states and, as illustrated in Figure 4.4, the characteristics penetrate into the discontinuity from both regions. This kind of discontinuity is called a *jump*, or a *shock* in application to gas dynamics, and is physically admissible. Note that the smooth solution u^ϵ, namely

$$u^\epsilon(x,t) = u_r + \frac{1}{2}(u_l - u_r)\left[1 - \tanh\left(\frac{(u_l - u_r)(x - \sigma t)}{2\epsilon}\right)\right],$$

of Burgers' equation converges to (4.6) as ϵ goes to zero. Therefore, this unique solution is the desired vanishing viscosity solution.

If $u_l < u_r$ there are infinitely many weak solutions. One of them is again (4.6), but now, as indicated in Figure 4.5, the characteristics emanate from the discontinuity. It can be shown that this solution is unstable in the sense that small perturbations of the data produce large changes of the solution. Moreover, if a small amount of viscosity is introduced in the equation the solution changes completely. Thus solution (4.6) is not physically desirable. Among the other weak solutions, one that is stable to perturbations is the *rarefaction wave* given by

$$u(x,t) = \begin{cases} u_l, & x < u_l\, t \\ x/t, & u_l\, t \leq x \leq u_r\, t \\ u_r, & x > u_r\, t. \end{cases}$$

This is in fact the vanishing viscosity solution to Burgers' equation, see Figure 4.6.

In order to generalize these concepts for a generic flux function f we need first to define the propagation speed of the discontinuity, that is the *Rankine–Hugoniot jump condition*, and second to establish if this discontinuity is physically acceptable using the previous observation that shock should have characteristic lines going *into* and *not out of* the discontinuity, as time advances.

Still in the context of the Riemann problem, that is piecewise constant data with a single discontinuity, we want to obtain for a generic flux function f the propagation speed of the discontinuity, namely the generalization of (4.7) which is also denoted as σ. We integrate equation (4.1) with respect to the spatial variable x from left, $\sigma t - 1$, to right, $\sigma t + 1$, of the discontinuity. This yields

$$\int_{\sigma t-1}^{\sigma t+1} \big(u_t(x,t) + f_x(u)\big)dx = 0 \Rightarrow \int_{\sigma t-1}^{\sigma t+1} u_t(x,t)\, dx = -f(u_r) + f(u_l) \quad (4.8)$$

for each $t > 0$. On the other hand, since data are piecewise constant we may write

$$\begin{aligned} 0 = \frac{d}{dt}(u_l + u_r) &= \frac{d}{dt}\left[\int_{\sigma t-1}^{\sigma t} u(x,t)\, dx + \int_{\sigma t}^{\sigma t+1} u(x,t)\, dx\right] \\ &= \int_{\sigma t-1}^{\sigma t+1} u_t(x,t)\, dx - \sigma\,(u_l - u_r), \end{aligned} \quad (4.9)$$

where the left, $u_l = u(\sigma t - 1, t)$, and right, $u_l = u(\sigma t + 1, t)$, values of the solution are used on each side of the discontinuity, $x = \sigma t$. Then, from (4.8) and (4.9) we

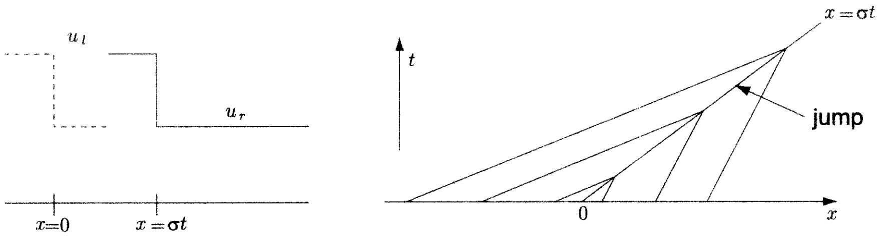


Fig. 4.4 Physically acceptable (entropy-compliant) weak solution: shock wave.

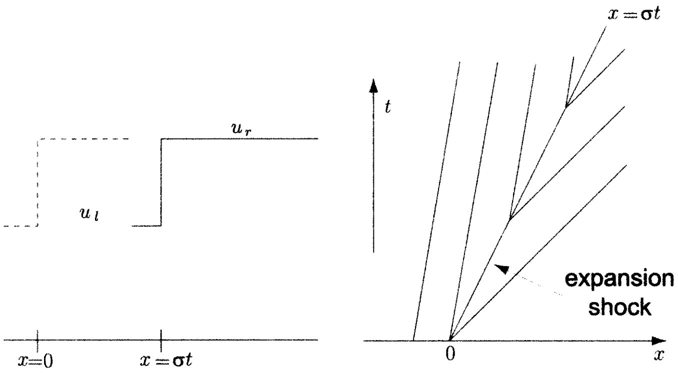


Fig. 4.5 Unstable (entropy-violating) weak solution.

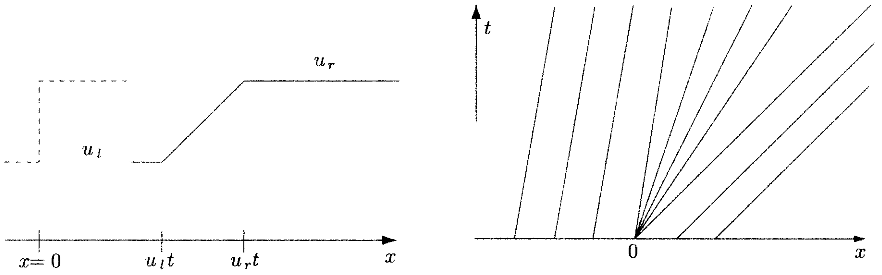


Fig. 4.6 Entropy-compliant weak solution: rarefaction wave.

obtain the propagation speed of the discontinuity for a scalar problem and for any flux function f,

$$\sigma = \frac{f(u_l) - f(u_r)}{u_l - u_r}. \tag{4.10}$$

As expected, the particularization to Burgers' equation recovers expression (4.7).

The previous analysis was carried out for the Riemann problem, that is for piecewise constant data. It can be generalized to more typical problems where the solution to the left and right of the discontinuity is varying smoothly. In general relation (4.10) across any propagating shock where u_l and u_r denote the values immediately to the left and right of the discontinuity is called the *Rankine–Hugoniot jump condition*.

Now that the speed of the discontinuity is defined we need to determine if the discontinuity corresponding to the weak solution is physically acceptable (i.e., it corresponds to the vanishing viscosity solution). Previous observations showed that characteristics must penetrate the discontinuity as time advances in order to have a physical solution. When the characteristics emanate from a discontinuity, small perturbations of the initial data, or of the equation itself (for instance, adding a small amount of viscosity), may result in a strong change of the solution (and typically this will cause a rarefaction of characteristics). From this analysis, the first entropy condition is derived:

- A discontinuity propagating with speed σ given by (4.10) satisfies the entropy condition if

$$a(u_l) = \frac{df}{du}(u_l) > \sigma > \frac{df}{du}(u_r) = a(u_r),$$

where the advection velocity is used, see Remark 3.4. For a convex flux function f, the speed of the discontinuity σ, namely the Rankine–Hugoniot condition, see (4.10), must lie between $a(u_l)$ and $a(u_r)$. Thus, the previous condition simply requires that $a(u_l) > a(u_r)$, which, in view of the convexity of the flux function f, amounts to imposing $u_l > u_r$.

A more general statement for the entropy condition, which applies to non-convex flux functions f, is:

- A discontinuity propagating with speed σ given by (4.10) satisfies the entropy condition if

$$\frac{f(u) - f(u_l)}{u - u_l} \geq \sigma \geq \frac{f(u) - f(u_r)}{u - u_r}$$

for all u between u_l and u_r.

4.2.3 Time and space discretization

Classical time integration methods, such as the θ family of methods and the Taylor–Galerkin schemes discussed in Chapter 3, can be applied in the solution of nonlinear hyperbolic problems. Note, however, that implicit methods generate systems of nonlinear equations and thus require iterative solution techniques. When the relevant time scales of the problem are close enough to justify the use of explicit time-marching schemes, this type of formulation should be favored. In fact, most explicit schemes keep their simple algorithmic structure in application to nonlinear problems. Multi-step explicit algorithms only involving first time derivatives are particularly suited, as they do not require evaluation of the spatial derivatives of the flux function. To illustrate the algorithmic simplicity of explicit methods, we shall consider the one-step and two-step Taylor–Galerkin methods, but note that other explicit methods can be implemented just as easily.

4.2.3.1 *One-step Taylor–Galerkin method* Consider the multidimensional scalar conservation law

$$u_t + \boldsymbol{\nabla} \cdot \boldsymbol{f}(u) = 0. \tag{4.11}$$

Proceeding as in Section 3.6.2, we replace the first and second time derivatives in the Taylor expansion with spatial derivatives using equation (4.11). This gives

$$\begin{aligned} u_t &= -\boldsymbol{\nabla} \cdot \boldsymbol{f}, \\ u_{tt} &= -\boldsymbol{\nabla} \cdot \boldsymbol{f}_t = -\boldsymbol{\nabla} \cdot \left(\frac{\partial \boldsymbol{f}}{\partial u} u_t \right) = \boldsymbol{\nabla} \cdot \big(\boldsymbol{a}(u) \boldsymbol{\nabla} \cdot \boldsymbol{f} \big), \end{aligned}$$

where the convection/advection velocity $\boldsymbol{a}(u) = \partial \boldsymbol{f}(u)/\partial u$ is again used, see Remark 3.4 or equation (3.2). The resulting second-order accurate time-stepping method reads

$$\frac{u^{n+1} - u^n}{\Delta t} = -\boldsymbol{\nabla} \cdot \boldsymbol{f}^n + \frac{\Delta t}{2} \boldsymbol{\nabla} \cdot \big(\boldsymbol{a}(u^n) \boldsymbol{\nabla} \cdot \boldsymbol{f}^n \big),$$

where as usual $\boldsymbol{f}^n = \boldsymbol{f}(u^n)$. A weighted residual formulation is obtained with the usual procedure: multiply by a test function belonging to

$$\mathcal{V} = \left\{ w(\boldsymbol{x}) \in \mathcal{H}^1(\Omega) \mid w(\boldsymbol{x}) = 0 \text{ for } \boldsymbol{x} \in \Gamma_D^{in} \right\},$$

integrate over space, and use Green's formula to weaken the smoothness requirement on the trial solution. The problem is then: find $u \in \mathcal{S}_t$, such that for all $w \in \mathcal{V}$,

$$\begin{aligned} \left(w, \frac{u^{n+1} - u^n}{\Delta t} \right) = (\boldsymbol{\nabla} w, \boldsymbol{f}^n) - \frac{\Delta t}{2} (\boldsymbol{\nabla} w, \boldsymbol{a}(u^n)\, \boldsymbol{\nabla} \cdot \boldsymbol{f}^n) \\ - \left(w\, \boldsymbol{n}, \boldsymbol{f}^n - \frac{\Delta t}{2} \boldsymbol{a}(u^n)\, \boldsymbol{\nabla} \cdot \boldsymbol{f}^n \right)_{\Gamma^{out}}, \end{aligned}$$

where a Dirichlet inlet condition is assumed and the trial space is now

$$\mathcal{S}_t := \left\{ u \mid u(\cdot, t) \in \mathcal{L}_2(\Omega), t \in [0, T] \text{ and } u(\boldsymbol{x}, t) = u_D \text{ for } \boldsymbol{x} \in \Gamma^{in} \right\}.$$

Note that discontinuous solutions are admissible. Nevertheless, in finite element practice continuous spatial approximations are usually chosen and shocks are modeled via adaptive mesh refinement and smearing the discontinuity through the introduction of artificial viscosity.

From a computational point of view at each step a system of algebraic equations must be solved. The matrix is the mass matrix, which in an Eulerian formulation does not vary with time and possesses nice properties for iterative solvers. The flux $\boldsymbol{f}(u)$ within an element is usually linearly interpolated in terms of its nodal values and the advection velocity $\boldsymbol{a}(u)$ is evaluated at the element integration points.

A possible disadvantage of the one-step TG method in application to nonlinear equations is the need to evaluate the second time derivative of the unknown in terms of the flux and its derivatives. Though perfectly feasible for scalar equations (Burgers), such evaluation becomes much more involved for systems of hyperbolic equations (Euler). Therefore, a two-step version of the method involving only first time derivatives is generally preferred in the modeling of nonlinear systems. Moreover, the divergence of the flux in the boundary integral of the previous weak form (last term on the r.h.s.) is also avoided in the two-step TG method.

4.2.3.2 Two-step Taylor–Galerkin method Consider now solving the multidimensional scalar conservation law (4.11) by means of the two-step time scheme

$$
\begin{aligned}
u^{n+1/2} &= u^n + \frac{\Delta t}{2}\, u_t^n &&= u^n - \frac{\Delta t}{2}\, \nabla \cdot \boldsymbol{f}^n, \\
u^{n+1} &= u^n + \Delta t\, u_t^{n+1/2} &&= u^n - \Delta t\, \nabla \cdot \boldsymbol{f}^{n+1/2},
\end{aligned}
$$

where $\boldsymbol{f}^n = \boldsymbol{f}(u^n)$ and $\boldsymbol{f}^{n+1/2} = \boldsymbol{f}(u^{n+1/2})$.

In weak form, the problem in the second integration step is defined as

$$
\left(w, \frac{u^{n+1} - u^n}{\Delta t}\right) = \left(\nabla w, \boldsymbol{f}^{n+1/2}\right) - \left(w\boldsymbol{n}, \boldsymbol{f}^{n+1/2}\right)_{\Gamma_{out}}. \qquad (4.12)
$$

Note that the integrals on the r.h.s. require the evaluation of the intermediate flux, $\boldsymbol{f}^{n+1/2}$, at the Gauss points only. As a result, the two-step TG method is implemented as follows:

Step 1: Compute at the integration points of the elements the intermediate value $u^{n+1/2}$ of the unknown using

$$
u^{n+1/2} = u^n - \frac{1}{2}\Delta t \nabla \cdot \boldsymbol{f}(u^n).
$$

The divergence of the flux within an element is evaluated using the so-called group representation of the flux components (see Remark 4.1 below):

$$
f_i = \sum_{a=1}^{\mathrm{n_{en}}} N_a(\boldsymbol{x})\, f_{ia}, \quad i = 1, \ldots, \mathrm{n_{sd}}.
$$

Once the intermediate value $u^{n+1/2}$ is obtained, the intermediate flux $\boldsymbol{f}^{n+1/2}$ at the considered element integration point is readily evaluated. Notice that Step 1 only involves element-based computations and thus avoids completely the need for the assembly of element contributions.

Step 2: It remains to compute the end-of-step values u^{n+1} by spatial discretization of the weak form (4.12). This results in an algebraic system with a mass matrix.

Remark 4.1 (Flux representation). An issue in the finite element discretization of nonlinear hyperbolic problems is the choice of a local approximation for the nonlinear flux function:

- A first possible option consists of using an elementwise constant representation of the flux:

$$
\boldsymbol{f}(u) = \boldsymbol{f}(\bar{u}),
$$

where $\bar{u}$ is a mean value of the unknown within the considered element.

- A second option is to interpolate first u at the desired point in the element and then evaluate the flux:

$$
\boldsymbol{f}(u) = \boldsymbol{f}\left(\sum_{a=1}^{\mathrm{n_{en}}} N_a(\boldsymbol{x})\, u_a\right).
$$

This is used in the one-step Lax–Wendroff method: the flux is evaluated at the Gauss points once the value of the unknown u is interpolated.

- Finally, another possible choice is to interpolate the flux directly by means of the same shape functions used for u:

$$\boldsymbol{f}(u) = \sum_{a=1}^{\mathrm{n_{en}}} N_a(\boldsymbol{x})\boldsymbol{f}(u_a).$$

This option is called *group representation* and is often used in Step 1 of the two-step Lax–Wendroff method to compute the divergence of the flux within an element. In this case, the group representation of the nonlinear flux function presents computational advantages over the other interpolation methods. It simplifies the evaluation of the flux divergence because the spatial dependency of the flux function is described directly by the element shape functions. Note also that on a uniform mesh of linear elements, the group representation provides a fourth-order accurate representation of the spatial derivative of the flux function (Donea and Giuliani, 1981).

To appraise the performance of the various flux representations, the reader is referred to the solved exercises in Section 4.8.

4.3 THE EULER EQUATIONS

4.3.1 Strong form of the conservation equations

The Euler equations of gas dynamics considered here express conservation of mass, momentum and energy in a compressible, inviscid and non-conducting fluid. They were developed in Section 1.4 both in differential and integral form. Recall the strong form of mass (1.11), momentum (1.16) and energy (1.21) conservation

$$\begin{aligned}\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla}\cdot(\rho\boldsymbol{v}) &= 0\\ \frac{\partial \rho\boldsymbol{v}}{\partial t} + \boldsymbol{\nabla}\cdot(\rho\,\boldsymbol{v}\otimes\boldsymbol{v} + p\,\mathbf{I}) &= \rho\boldsymbol{b}\\ \frac{\partial \rho E}{\partial t} + \boldsymbol{\nabla}\cdot\big((\rho E + p)\boldsymbol{v}\big) &= \boldsymbol{v}\cdot\rho\boldsymbol{b}\end{aligned} \tag{4.13}$$

where the Cauchy stress $\boldsymbol{\sigma}$ has been replaced by $-p\,\mathbf{I}$ because we assume an inviscid fluid (constitutive law), and the rest of the variables were already defined in Chapter 1: the density ρ, the momentum $\rho\boldsymbol{v}$, the total energy ρE per unit volume of the fluid, the fluid pressure p and the external volume force per unit volume $\rho\boldsymbol{b}$. In vector form and in 3D ($\mathrm{n_{sd}} = 3$), these equations can be rewritten as

$$\mathbf{U}_t + \frac{\partial \mathbf{F}_1}{\partial x_1} + \frac{\partial \mathbf{F}_2}{\partial x_2} + \frac{\partial \mathbf{F}_3}{\partial x_3} = \mathbf{B}, \tag{4.14}$$

where $\mathbf{U}$ is the vector of conservation variables, $\mathbf{F}_i$ are the associated flux vectors for each spatial dimension and $\mathbf{B}$ is a source term. They are defined as follows:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho \boldsymbol{v} \\ \rho E \end{pmatrix}, \ \mathbf{F}_i = \begin{pmatrix} \rho v_i \\ \rho \boldsymbol{v}\, v_i + p \\ (\rho E + p)\, v_i \end{pmatrix}, \ i = 1, \ldots, \mathrm{n_{sd}}, \text{ and } \mathbf{B} = \begin{pmatrix} 0 \\ \rho \boldsymbol{b} \\ \boldsymbol{v} \cdot \rho \boldsymbol{b} \end{pmatrix}. \quad (4.15)$$

The previous differential equation can be compacted further using the divergence of the flux vector

$$\mathbf{F} = \begin{pmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \\ \mathbf{F}_3 \end{pmatrix},$$

and finally, the Euler conservation equations, boundary conditions and initial condition induce the strong form of the problem:

$$\begin{aligned} &\mathbf{U}_t + \boldsymbol{\nabla} \cdot \mathbf{F} = \mathbf{B} && \text{in } \Omega \times]0, T[, && (4.16\text{a}) \\ &\mathbf{F}^{in} \cdot \boldsymbol{n} = \mathbf{G} && \text{on } \Gamma \times]0, T[, && (4.16\text{b}) \\ &\mathbf{U}(\boldsymbol{x}, 0) = \mathbf{U}_0(\boldsymbol{x}) && \text{on } \Omega \text{ at } t = 0. && (4.16\text{c}) \end{aligned}$$

In the previous chapter we have already seen that boundary conditions for scalar hyperbolic problems are only imposed on the inflow portion of the boundary. For systems of hyperbolic equations only inflow *components* of the flux vector can be prescribed. This is discussed in detail in Section 4.3.4.

Boundary conditions for this problem will be discussed in detail in Section 4.3.4. Before then, we need to define the equation of state to complete the previous system of conservation equations. The equation of state relates the internal energy, $e = E - (1/2)\|\boldsymbol{v}\|^2$, see Section 1.4.4, to pressure, p, and density, ρ. For an ideal gas, internal energy is only a function of temperature. In fact, for a perfect polytropic gas the internal energy per unit mass is proportional to temperature: $e = c_v T$ where c_v is constant and known as the specific heat at constant volume. Thus the state equation relating e to p and ρ is obtained using the ideal gas law

$$p = R\rho T = \frac{R}{c_v}\rho e = \frac{R}{c_v}\rho\Big(E - \frac{1}{2}\|\boldsymbol{v}\|^2\Big)$$

where R is the gas constant per unit mass, which is equal to the universal gas constant $\mathcal{R}$ divided by the molecular mass of the fluid.

Other expressions of the equation of state are also common. For instance, for a polytropic gas the total enthalpy of the fluid $H = E + p/\rho$ is also proportional to temperature: $H = c_p T$ where c_p is constant and known as the specific heat at constant pressure. The ratio $\gamma = c_p/c_v$ of the specific heat coefficients can relate the specific heat constants and the gas constant per unit mass, R,

$$c_p = \frac{\gamma}{\gamma - 1} R \qquad \text{and} \qquad c_v = \frac{R}{\gamma - 1}.$$

With the help of this new constant γ the usual form of the equation of state is obtained

$$E = \frac{p}{\rho(\gamma - 1)} + \frac{1}{2}\|\boldsymbol{v}\|^2 \tag{4.17a}$$

or in terms of the total enthalpy of the fluid

$$H = E + \frac{p}{\rho} = \frac{\gamma p}{\rho(\gamma - 1)} + \frac{1}{2}\|\boldsymbol{v}\|^2. \tag{4.17b}$$

The speed of sound, c, is also common in this formulation. It enters in the definition of the Mach number

$$M = \frac{\|\boldsymbol{v}\|}{c},$$

and it is given by

$$c = \sqrt{\frac{\gamma p}{\rho}}.$$

Thus using the state equations (4.17) the speed of sound can also be written as

$$c^2 = \gamma(\gamma - 1)\Big(E - \frac{1}{2}\|\boldsymbol{v}\|^2\Big) \quad \text{or} \quad c^2 = (\gamma - 1)\Big(H - \frac{1}{2}\|\boldsymbol{v}\|^2\Big).$$

Remark 4.2 (Non-conservative form). The Euler equations can be written in various equivalent forms, depending on the choice of the dependent flow variables. For instance, use can be made of the primitive variables, ρ, $\boldsymbol{v}$, p. In this case, equations (4.13) and (4.16) transform to the following non-conservative form, see for example Hirsch (1990, Chap. 16) for details:

$$\begin{aligned}
\rho_t + \boldsymbol{v} \cdot \boldsymbol{\nabla}\rho + \rho\, \boldsymbol{\nabla} \cdot \boldsymbol{v} &= 0,\\
\boldsymbol{v}_t + (\boldsymbol{v} \cdot \boldsymbol{\nabla})\boldsymbol{v} + \frac{1}{\rho}\,\boldsymbol{\nabla} p &= \boldsymbol{b},\\
p_t + \boldsymbol{v} \cdot \boldsymbol{\nabla} p + \rho\, c^2\, \boldsymbol{\nabla} \cdot \boldsymbol{v} &= 0,
\end{aligned}$$

where c is the speed of sound. It is important to note that the conservation form of the Euler equations is preferred in order to correctly compute the propagation speed and the intensity of flow discontinuities.

4.3.2 The quasi-linear form of the Euler equations

In order to investigate the basic properties of the Euler equations (see for instance Hirsch, 1990) it is necessary to write these equations in a quasi-linear form. This is similar in structure to the convection equations studied in Chapter 3. In the case of a perfect polytropic gas or, more generally, for fluids satisfying the relation $p = \rho\, f(e)$, the components $\mathbf{F}_i$ of the inviscid flux vector $\mathbf{F}$, see (4.15), are *homogeneous functions of degree 1*, see Remark 4.3, in the conservation variables $\mathbf{U}$. Therefore, the flux components can be written as

$$\mathbf{F}_i(\mathbf{U}) = \mathbf{A}_i(\mathbf{U})\,\mathbf{U}, \tag{4.18}$$

where $\mathbf{A}_i(\mathbf{U}) = \partial \mathbf{F}_i / \partial \mathbf{U}$, with $i = 1, \ldots, \mathrm{n_{sd}}$, are the Jacobian matrices. Thus, from (4.14), the quasi-linear form of the Euler equations is given by

$$\mathbf{U}_t + \mathbf{A}_1 \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{A}_2 \frac{\partial \mathbf{U}}{\partial x_2} + \mathbf{A}_3 \frac{\partial \mathbf{U}}{\partial x_3} = \mathbf{B},$$

or in compact form

$$\mathbf{U}_t + (\mathbf{A} \cdot \boldsymbol{\nabla})\, \mathbf{U} = \mathbf{B}. \tag{4.19}$$

Remark 4.3 (Homogeneous functions). Let f be a scalar function differentiable in an open set $S \subset \mathbb{R}^n$. f is a homogeneous function of degree p in S if

$$f(t\boldsymbol{x}) = t^p f(\boldsymbol{x}) \qquad \forall t > 0 \text{ and } \forall \boldsymbol{x} \in S.$$

Moreover, the so-called Euler theorem for homogeneous functions claims that a homogeneous function of degree p verifies

$$p\, f(\boldsymbol{x}) = \boldsymbol{x} \cdot \boldsymbol{\nabla} f(\boldsymbol{x}).$$

Remark 4.4 (Euler equations in 2D). In 2D, the vector $\mathbf{U}$ of conservative variables and the associated flux vector $\mathbf{F}$ reduce to

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho v_1 \\ \rho v_2 \\ \rho E \end{pmatrix}, \quad \mathbf{F}_1 = \begin{pmatrix} \rho v_1 \\ \rho v_1^2 + p \\ \rho v_1 v_2 + p \\ H \rho v_1 \end{pmatrix} \quad \text{and} \quad \mathbf{F}_2 = \begin{pmatrix} \rho v_2 \\ \rho v_1 v_2 + p \\ \rho v_2^2 + p \\ H \rho v_2 \end{pmatrix}.$$

In 2D the Jacobian matrices for a polytropic ideal gas are

$$\mathbf{A}_1 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \frac{\gamma-1}{2}\|\boldsymbol{v}\|^2 - v_1^2 & (3-\gamma)v_1 & (1-\gamma)v_2 & \gamma - 1 \\ -v_1 v_2 & v_2 & v_1 & 0 \\ \frac{\gamma-1}{2} v_1 \|\boldsymbol{v}\|^2 - v_1 H & -v_1^2(\gamma-1) + H & (1-\gamma)v_1 v_2 & \gamma v_1 \end{pmatrix},$$

$$\mathbf{A}_2 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ -v_1 v_2 & v_2 & v_1 & 0 \\ \frac{\gamma-1}{2}\|\boldsymbol{v}\|^2 - v_2^2 & (1-\gamma)v_1 & (3-\gamma)v_2 & \gamma - 1 \\ \frac{\gamma-1}{2} v_2 \|\boldsymbol{v}\|^2 - v_2 H & (1-\gamma)v_1 v_2 & -v_2^2(\gamma-1) + H & \gamma v_1 \end{pmatrix}.$$

Remark 4.5 (Euler equations in 1D). In 1D, the vector $\mathbf{U}$ of conservative variables and the associated flux vector $\mathbf{F}$ reduce to

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho v \\ \rho E \end{pmatrix} \quad \text{and} \quad \mathbf{F} = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ \rho v H \end{pmatrix}.$$

The corresponding Jacobian matrix $\mathbf{A} = \partial \mathbf{F} / \partial \mathbf{U}$ takes the form

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 \\ -\frac{1}{2}(3-\gamma)\, v^2 & (3-\gamma)\, v & \gamma - 1 \\ (\gamma-1)\, v^3 - \gamma\, v\, E & \gamma\, E - \frac{3}{2}(\gamma-1)\, v^2 & \gamma\, v \end{pmatrix}.$$

4.3.3 Basic properties of the Euler equations

The Euler equations form a hyperbolic system in time and the solution of the homogeneous quasi-linear system (4.19),

$$\mathbf{U}_t + (\mathbf{A} \cdot \boldsymbol{\nabla})\mathbf{U} = 0,$$

can be expressed in Fourier series. A typical term (mode) of the series has the wave-like form

$$\mathbf{U} = \hat{\mathbf{U}} e^{i(\mathbf{k} \cdot \boldsymbol{x} - \omega t)}.$$

Here, $\mathbf{k} = (k_1, \ldots, k_{\mathrm{n_{sd}}})^T$ is the wave-number vector and ω represents (up to a factor 2π) the frequency of the propagating wave. Note that $(\mathbf{k} \cdot \boldsymbol{x} - \omega t)$ represents the phase of the wave propagating in the direction $\mathbf{k}$ with frequency ω. Introducing the previous single mode solution into the homogeneous equation leads to the following condition for the existence of a wave-like solution:

$$\det\big(\omega \mathbf{I} - \mathbf{A} \cdot \mathbf{k}\big) = 0,$$

where $\mathbf{I}$ is the identity matrix of order $\mathrm{n_{sd}}$. Thus, wave-like solutions will exist if the eigenvalues λ_j, $j = 1, \ldots, \mathrm{n_{sd}} + 2$, of the projection matrix $\mathbf{A_k} = \mathbf{A} \cdot \mathbf{k}$ are real with linear independence of the corresponding right eigenvectors. These eigenvalues represent the frequency ω. Let $\mathbf{R}$ be the matrix whose columns are the right eigenvectors of $\mathbf{A_k}$, that is

$$\mathbf{A_k R} = \mathbf{R \Lambda}, \tag{4.20}$$

where $\mathbf{\Lambda} := \mathrm{diag}(\lambda_j)$, $j = 1, \ldots, \mathrm{n_{sd}}+ 2$, is a diagonal matrix, such that the diagonal elements are the eigenvalues λ_j of $\mathbf{A_k}$. These eigenvalues are associated with an arbitrary direction of propagation $\mathbf{k}$, $\lambda_j(\mathbf{k})$, and can be evaluated in terms of the fluid velocity $\boldsymbol{v}$ and the sound speed c as

$$\begin{aligned} \lambda_1 = \cdots = \lambda_{\mathrm{n_{sd}}} &= \boldsymbol{v} \cdot \mathbf{k}, \\ \lambda_{\mathrm{n_{sd}}+1} &= \boldsymbol{v} \cdot \mathbf{k} + c, \\ \lambda_{\mathrm{n_{sd}}+2} &= \boldsymbol{v} \cdot \mathbf{k} - c. \end{aligned}$$

Note that the characteristics associated with the eigenvalue $\boldsymbol{v} \cdot \mathbf{k} = \sum_{s=1}^{\mathrm{n_{sd}}} v_s k_s$ of multiplicity $\mathrm{n_{sd}}$ represent the trajectories of the particles. Finally, we can consider the decomposition of the projection Jacobian matrices along the direction of the propagation $\mathbf{k}$

$$\mathbf{A_k} = \sum_{s=1}^{\mathrm{n_{sd}}} k_s \mathbf{A}_s = \mathbf{R \Lambda R}^{-1}. \tag{4.21}$$

It is important to note that this equation shows that a projection of the Jacobian matrix $\mathbf{A}$ along an arbitrary direction is diagonalizable, not the Jacobian matrix itself. This has a major consequence in the diagonalization of the Euler equations. In fact, the Euler equations can no longer be diagonalized in 2D and 3D. This is because

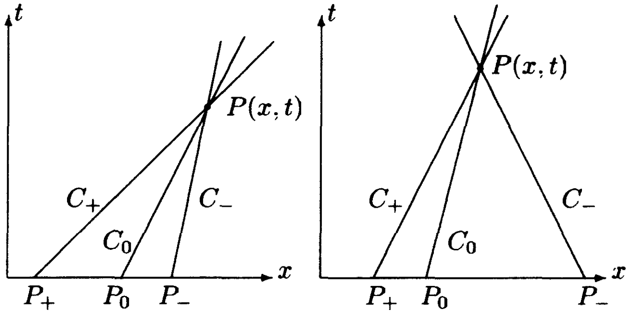


Fig. 4.7 Illustration of characteristics at point $P(x,t)$ for supersonic (left) and subsonic (right) flow conditions.

waves can travel in an infinite number of directions and the decomposition into scalar waves is no longer unique. In *one dimension*, see Remark 4.5, we may multiply the system (4.19) in quasi-linear form by matrix $\mathbf{R}^{-1}$ to obtain

$$\mathbf{R}^{-1}\left(\mathbf{U}_t + \mathbf{A} \cdot \mathbf{U}_x\right) = \mathbf{R}^{-1}\mathbf{B}.$$

Then, introducing the *characteristic variables* $\mathbf{W} = \mathbf{R}^{-1}\mathbf{U}$ and under the assumption that the coefficients of matrix $\mathbf{R}$ are constant, the previous equation becomes:

$$\mathbf{W}_t + \Lambda \mathbf{W}_x = \mathbf{R}^{-1}\mathbf{B}. \tag{4.22}$$

Unfortunately, the characteristic variables can only be defined for the 1D Euler equations and not for the more general case of multidimensional flows. The characteristic variables, also called *Riemann variables*, are (for isentropic flows of a polytropic ideal gas)

$$\mathbf{W} = \left(s\,;\, v + \frac{2}{\gamma - 1}c\,;\, v - \frac{2}{\gamma - 1}c\right)^T, \tag{4.23}$$

where s is the entropy per unit mass of fluid, see Remark 4.6. The associated eigenvalues are

$$\Lambda = \text{diag}(v, v + c, v - c), \tag{4.24}$$

and they define the three characteristics, see Figure 4.7:

$$\begin{aligned} \frac{dx}{dt} &= v && \text{on } C_0, \\ \frac{dx}{dt} &= v + c && \text{on } C_+, \\ \frac{dx}{dt} &= v - c && \text{on } C_-. \end{aligned} \tag{4.25}$$

From (4.22) the 1D homogeneous Euler equations are

$$\begin{aligned}\left(\frac{\partial}{\partial t}+v\frac{\partial}{\partial x}\right)s&=0,\\ \left(\frac{\partial}{\partial t}+(v+c)\frac{\partial}{\partial x}\right)\left(v+\frac{2c}{\gamma-1}\right)&=0,\\ \left(\frac{\partial}{\partial t}+(v-c)\frac{\partial}{\partial x}\right)\left(v-\frac{2c}{\gamma-1}\right)&=0.\end{aligned}$$

This allows us to observe that the material derivative of s is zero, that is the specific entropy is constant along the particle trajectories in the smooth part of the flow (away from discontinuities). In fact, the entropy s is transported at speed v along the straight characteristic C_0 of equation $dx/dt = v$, which coincides with the particle path. The quantities $v \pm 2c/(\gamma - 1)$ are transported at speed $v \pm c$ along the characteristic lines $C_\pm$ illustrated in Figure 4.7.

It is important to note that, in contrast to the scalar case, the characteristics associated with the Euler equations are not in general straight lines and do not transport constant values of the unknowns. The above 1D system is an exception, because it is homogeneous (the r.h.s. is zero), and the diagonalization of the system induces that each l.h.s. expresses the vanishing of the material derivative of the transported quantity.

These properties, which are restricted to the 1D problem, are useful for the specification of suitable boundary conditions in multiple dimensions. Moreover, information on the sign of the eigenvalues $\mathbf{\Lambda}$, which represent the propagation speeds, will be exploited in Section 4.4.2 to derive directional (upwind-type) spatial discretization models for the Euler equations.

Remark 4.6 (Entropy per unit mass). This is another fundamental thermodynamic variable, which can be interpreted as a measure of the disorder in the system. The entropy per unit mass, s, is defined from a reference state (up to a constant) as

$$s - s_{\text{ref}} = c_v \ln \frac{p/p_{\text{ref}}}{(\rho/\rho_{\text{ref}})^\gamma} = c_p \ln \frac{T}{T_{\text{ref}}} - R \ln \frac{p}{p_{\text{ref}}}.$$

4.3.4 Boundary conditions

The issue of deciding which boundary conditions to impose on the Euler equations is not a trivial one; an inadequate choice can affect the existence and uniqueness of solutions.

For instance, at a *solid boundary*, only the normal component of the velocity must be specified equal to the solid velocity normal to the fluid–solid interface. Consequently, in the case of a fixed solid boundary, the normal component v_n of the fluid velocity is set to zero. Moreover, in the absence of thermal conduction, the energy flux across the boundary vanishes. The same applies to the flux of mass. Thus, neither the density ρ nor the total energy ρE must be specified at a solid boundary. The conditions of zero normal flux for both mass and energy are automatically satisfied in the finite element context, because of the natural boundary conditions on prescribed normal fluxes.

In general, the computational domain is delimited by an *external boundary*. A linearized Riemann analysis in the direction of the outward normal to the contour is required to determine the speed of wave propagation. As indicated in the previous section, the Euler equations can be diagonalized in 1D; the outward normal to the boundary is this dimension. Then, three distinct values of the propagation speed are obtained from the linearization of the Euler equations, namely

$$\begin{aligned} \lambda_1 = \cdots = \lambda_{\mathrm{n_{sd}}} &= v_n, \\ \lambda_{\mathrm{n_{sd}}+1} &= v_n + c, \\ \lambda_{\mathrm{n_{sd}}+2} &= v_n - c, \end{aligned} \tag{4.26}$$

where $v_n = \boldsymbol{v} \cdot \boldsymbol{n}$ is the velocity along the outward normal. Then, at each point of the boundary of the computational domain one should prescribe as many conditions as there are negative eigenvalues (incoming information) in (4.26).

Two distinct possibilities must be considered in the case of *supersonic flow*, that is when the local Mach number is such that $M = (|v_n|/c) \geq 1$.

1. *Supersonic inflow boundaries*, where $v_n < 0$ and $|v_n| > c$: for such boundaries all components of vector $\mathbf{U}$ defined in (4.15) must be specified because all eigenvalues (4.26) are negative.

2. *Supersonic outflow boundaries* where $v_n > 0$ and $v_n > c$: here, all eigenvalues are positive and no component of vector $\mathbf{U}$ can be prescribed.

For *subsonic boundaries*, $M = (|v_n|/c) < 1$, the situation is more complex and the components of $\mathbf{U}$ that can be specified are those of the incoming Riemann variables. Here we also distinguish two cases:

1. *Subsonic inflow boundaries*, where $v_n < 0$ and $|v_n| < c$: only the eigenvalue $\lambda_{\mathrm{n_{sd}}+1}$ is not negative, and $\mathrm{n_{sd}}+1$ appropriate conditions can be imposed.

2. *Subsonic outflow boundaries*, where $v_n > 0$ and $v_n < c$: only one condition (for instance, the pressure) can be prescribed in this case, because only the eigenvalue $\lambda_{\mathrm{n_{sd}}+2}$ is negative.

4.4 SPATIAL DISCRETIZATION TECHNIQUES

Various options are available for the finite element spatial discretization of the Euler equations. This comprises several possible choices for the weighted residual formulation, as well as various options for the interpolation of the field variables. As an alternative to the standard Galerkin formulation, formulations of upwind-type are available which possess good stability properties and produce a directional discretization in accordance with the physical behavior of inviscid flows. Finite element models of upwind type include stabilized formulations, such as SUPG and GLS, as well as discontinuous Galerkin formulations combined with flux vector splitting or other devices such as approximate Riemann solvers. We start by introducing the standard (continuous) Galerkin formulation of the Euler equations, and then discuss ways of producing more stable upwind-type discretizations.

4.4.1 Galerkin formulation

To perform the classical Galerkin spatial discretization of the Euler equations (4.16) we need to generalize the functional spaces defined in Section 3.4.2 for the scalar case. For clarity of exposition we assume that no Dirichlet boundary conditions are prescribed at the inflow boundaries, and as indicated by equation (4.16b), the boundary conditions are specified in terms of prescribed normal flux. The test functions, $\mathbf{W}$, belong to $\boldsymbol{\mathcal{V}} := \boldsymbol{\mathcal{H}}^1(\Omega)$ (space of vector functions, see Section 1.5.1.2) and do not depend on time, and the space of trial functions, $\boldsymbol{\mathcal{S}}_t$, varies as a function of t

$$\boldsymbol{\mathcal{S}}_t := \left\{\mathbf{U} \mid \mathbf{U}(\cdot,t) \in \boldsymbol{\mathcal{H}}^1(\Omega), t \in [0,T]\right\}.$$

The Galerkin spatial discretization of the Euler equations (4.16) can then be stated as follows: for $t \in]0,T[$, find $\mathbf{U}(\boldsymbol{x},t) \in \boldsymbol{\mathcal{S}}_t$ such that $\mathbf{U}(\boldsymbol{x},0) = \mathbf{U}_0(\boldsymbol{x})$ and

$$\int_\Omega \mathbf{W}^T\,\mathbf{U}_t\,d\Omega - \int_\Omega \boldsymbol{\nabla}\mathbf{W}^T\,\mathbf{F}(\mathbf{U})\,d\Omega = \int_\Omega \mathbf{W}^T\,\mathbf{B}\,d\Omega - \int_\Gamma \mathbf{W}^T\,\mathbf{G}\,d\Gamma$$

for all test functions $\mathbf{W} \in \boldsymbol{\mathcal{V}}$. In compact form, this equation reads

$$(\mathbf{W},\mathbf{U}_t) - (\boldsymbol{\nabla}\mathbf{W},\mathbf{F}) = (\mathbf{W},\mathbf{B}) - (\mathbf{W},\mathbf{G})_\Gamma. \tag{4.27}$$

The conservation variables in vector $\mathbf{U}$ are approximated spatially within an element using standard finite element shape functions:

$$\rho_e = \mathbf{N}_\rho^T\,\widetilde{\boldsymbol{\rho}}, \qquad (\rho\boldsymbol{v})_e = \mathbf{N}_{\rho v}^T\,\widetilde{\boldsymbol{\rho v}}, \qquad (\rho E)_e = \mathbf{N}_{\rho E}^T\,\widetilde{\boldsymbol{\rho E}},$$

where vectors $\widetilde{\boldsymbol{\rho}}$, $\widetilde{\boldsymbol{\rho v}}$ and $\widetilde{\boldsymbol{\rho E}}$ list the values of the conservation variables at their respective nodal points and $\mathbf{N}$ denotes the associated shape functions. Note that each conservation variable can be interpolated differently.

A common practice in compressible flow problems is to use low-order approximations. In fact, high-order polynomial approximations are not optimal in zones where the solution presents steep gradients. Moreover, the polynomial approximation for the density and the energy is often taken at one order less than for the momentum components. Note, however, that, in contrast to the incompressible case discussed in Chapter 6, this is not indispensable for compressible flows. For instance, when using primitive variables a common choice is constant element-by-element approximations for the density and the specific internal energy (and hence the pressure), and piecewise linear/multilinear representations for the velocity.

As seen in Chapters 2 and 3, the Galerkin approach to the spatial discretization of highly convective problems is often characterized by a lack of sufficient stability.

This is also the case for the Euler system of hyperbolic equations. Finite element formulations of the upwind type are therefore better adapted for the spatial discretization of the Euler equations. Such formulations are presented in the next section.

4.4.2 Upwind-type discretizations

Centered spatial discretization techniques, such as the Galerkin finite element method, do not account for the physical propagation properties of the solutions of the Euler equations. That is, the propagation of perturbations along the characteristics is not taken into account. For this reason, finite element formulations of upwind type are generally preferred for the Euler equations. As already seen in Chapters 2 and 3 for scalar convection problems, spatial discretization schemes of streamline-upwind type are able to produce directional discretizations in accordance with the physical behavior of the solution of convective transport problems. In the case of the Euler equations, the situation is more complex than in the scalar case, because there are generally mixed sign eigenvalues in the coupled system of conservation equations.

Three basic classes of upwind-type methods have been developed to produce a directional spatial discretization of the Euler equations in accordance with the physical behavior of inviscid compressible flows. Note, however, that standard upwind schemes are only first-order accurate in space and therefore generally too dissipative. They have to be corrected in the form of high-resolution methods to achieve, as indicated in Section 4.5.3, higher-order accuracy away from discontinuities.

A first family of upwind schemes is inspired by Godunov's (1959) method which solves the locally 1D Euler equations for discontinuous neighboring states (the Riemann problem in Figure 4.4). This approach introduces in the discretization scheme information from the exact local solution of the nonlinear Euler equations. It has generated many variants which are based on *approximate Riemann solvers*. A popular method in this class is Roe's (1981) approximate Riemann solver in which a constant coefficient linear system of conservation laws is considered instead of the original nonlinear system. The reader interested in an in-depth discussion of approximate Riemann solvers is urged to consult specialized textbooks, such as, for instance, Hirsch (1990) and LeVeque (1992).

A second family of upwind-type methods are the so-called *flux vector splitting* methods introduced in the finite difference context by Steger and Warming (1981) and van Leer (1982). These methods use information on the sign of the eigenvalues of the Jacobian matrices to split the flux terms in the Euler equations and discretize them directionally according to the sign of the associated propagation speeds. We shall describe the flux vector splitting technique in the next section and then illustrate its use in conjunction with the discontinuous Galerkin method.

A third family of upwind-type finite element methods for the Euler equations includes the SUPG and GLS methods introduced in Chapter 2. Sections 4.4.2.3 and 4.4.2.4 provide an introduction to such stabilization methods in the context of the Euler equations.

4.4.2.1 Flux vector splitting technique The rationale of flux vector splitting techniques is presented for a 1D problem, and then extended to multidimensional situations. The Euler equations have in general positive and negative eigenvalues, which correspond to the inflow and outflow components. To produce an upwind-type discretization, the idea is to decompose the flux vector $\mathbf{F}$ into the sum

$$\mathbf{F} = \mathbf{F}^+ + \mathbf{F}^-,$$

such that $\partial\mathbf{F}^+/\partial\mathbf{U}$ has only positive eigenvalues and $\partial\mathbf{F}^-/\partial\mathbf{U}$ has only negative eigenvalues. Hence, the spatial terms in a 1D problem

$$\mathbf{U}_t + \mathbf{F}_x^+ + \mathbf{F}_x^- = 0$$

can be discretized directionally according to the sign of the eigenvalues by expressions such as

$$\frac{\partial\mathbf{F}^+}{\partial x}(x_i) \approx \frac{\mathbf{F}_i^+ - \mathbf{F}_{i-1}^+}{h} \quad \text{and} \quad \frac{\partial\mathbf{F}^-}{\partial x}(x_i) \approx \frac{\mathbf{F}_{i+1}^- - \mathbf{F}_i^-}{h}.$$

The splitting of the flux vector is defined using (4.18), that is $\mathbf{F} = \mathbf{A}\,\mathbf{U}$, as

$$\mathbf{F} = \mathbf{F}^+ + \mathbf{F}^-, \qquad \mathbf{F}^+ = \mathbf{A}^+\,\mathbf{U}, \qquad \mathbf{F}^- = \mathbf{A}^-\,\mathbf{U}.$$

Therefore we need to split the Jacobian matrix $\mathbf{A}$ into $\mathbf{A} = \mathbf{A}^+ + \mathbf{A}^-$. To define these matrices we recall the diagonalization of the 1D projection of the Jacobian matrix $\mathbf{A}$, see Section 4.3.3 and equation (4.21), that is $\mathbf{A} = \mathbf{R}\boldsymbol{\Lambda}\mathbf{R}^{-1}$, where, for the 1D Euler equations, $\boldsymbol{\Lambda}$ is defined in (4.24) and $\mathbf{R}$ is the matrix whose columns are the right eigenvectors of $\mathbf{A}$, see (4.20). The splitting of the Jacobian matrix becomes obvious from the splitting of the diagonal matrix of eigenvalues $\boldsymbol{\Lambda}$, that is

$$\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^+ + \boldsymbol{\Lambda}^- \text{ induces: } \mathbf{A}^+ = \mathbf{R}\boldsymbol{\Lambda}^+\mathbf{R}^{-1}, \text{and } \mathbf{A}^- = \mathbf{R}\boldsymbol{\Lambda}^-\mathbf{R}^{-1},$$

where $\boldsymbol{\Lambda}^+$ and $\boldsymbol{\Lambda}^-$ have the positive and negative eigenvalues, respectively,

$$\begin{aligned} \boldsymbol{\Lambda}^+ &= \operatorname{diag}\left(\frac{(v-c)+|v-c|}{2}, \frac{v+|v|}{2}, \frac{(v+c)+|v+c|}{2}\right), \\ \boldsymbol{\Lambda}^- &= \operatorname{diag}\left(\frac{(v-c)-|v-c|}{2}, \frac{v-|v|}{2}, \frac{(v+c)-|v+c|}{2}\right). \end{aligned} \tag{4.28}$$

Remark 4.7. In this 1D case, each of the three characteristic variables (4.23) can be integrated separately, and a scalar upwind scheme can then be used for each one:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = -\frac{1}{h}\left(a^+(u_i^n - u_{i-1}^n) + a^-(u_{i+1}^n - u_i^n)\right),$$

with $a^+ = \max(0, a) = (a + |a|)/2$, $a^- = \min(a, 0) = (a - |a|)/2$, and where it suffices to replace the convection speeds a^+ and a^- with the positive and negative projections of the eigenvalues defined in (4.28). Note that u in the previous equation must be replaced by the Riemann variables.

This final decomposition allows us to compute $\mathbf{A}^+$ and $\mathbf{A}^-$ and consequently $\mathbf{F}^+$ and $\mathbf{F}^-$. Then, as noted previously, the numerical scheme can account for inflow or outflow fluxes. This, however, is only valid for a 1D problem. The generalization to higher spatial dimension is not trivial. As seen in Section 4.3.3 the Euler equations cannot be diagonalized in 2D or 3D. Consequently, in higher dimensions the 1D concept is used restricting the splitting of the flux to the normal direction. The normal flux to an element side (face in 3D) with outward normal $\boldsymbol{n}$ is

$$\mathbf{F}_{\boldsymbol{n}}(\mathbf{U}) = \sum_{i=1}^{n_{\mathrm{sd}}} \mathbf{F}_i(\mathbf{U})\, n_i.$$

Consequently, the upwind characterization leading to the splitting of the normal flux is based on the signs of the eigenvalues of the projected Jacobian matrix

$$\mathbf{A}_{\boldsymbol{n}}(\mathbf{U}) = \frac{\partial \mathbf{F}_{\boldsymbol{n}}(\mathbf{U})}{\partial \mathbf{U}} = \sum_{i=1}^{n_{\mathrm{sd}}} \mathbf{A}_i(\mathbf{U})\, n_i = (\mathbf{A} \cdot \boldsymbol{n}).$$

The normal flux $\mathbf{F}_{\boldsymbol{n}}(\mathbf{U})$ can then be split, see for instance Figure 4.8, into inflow and outflow components $\mathbf{F}_{\boldsymbol{n}}^-$ and $\mathbf{F}_{\boldsymbol{n}}^+$, respectively, which are defined by

$$\mathbf{F}_{\boldsymbol{n}}^{\pm}(\mathbf{U}) = \mathbf{A}_{\boldsymbol{n}}^{\pm}\,\mathbf{U} = \left(\mathbf{R}\,\boldsymbol{\Lambda}^{\pm}\,\mathbf{R}^{-1}\right)\mathbf{U}, \text{ with } \boldsymbol{\Lambda}^{\pm} = \frac{1}{2}\left(\boldsymbol{\Lambda} \pm |\boldsymbol{\Lambda}|\right).$$

The splittings $\mathbf{F}_{\boldsymbol{n}}^{\pm}$ are known in closed form for several flux vector splittings, see Hirsch (1990, Chap. 20) for specific cases. In the next section, we show how flux vector splitting techniques ideally combine with the discontinuous Galerkin method for the upwind discretization of the Euler equations. Examples of applications of the flux vector splitting technique in the finite element context are described in Section 4.8.2.

4.4.2.2 Discontinuous Galerkin method with flux vector splitting The solution of the Euler equations is often characterized by discontinuities. An alternative to the classical Galerkin method (continuous in space) is a finite element method where the approximation $\mathbf{U}^h$ and the fluxes $\mathbf{F}(\mathbf{U}^h)$ are allowed to be discontinuous across element boundaries. The *discontinuous Galerkin method* was introduced in Section 3.9. Here we follow the seminal work of Baumann and Oden (2000).

The discontinuous Galerkin method is globally conservative and also elementwise conservative (i.e., the conservation equations are approximately satisfied at the element level) if a conservative formulation is employed, see (4.16). Moreover, the method involves a weak imposition of the Rankine–Hugoniot jump conditions across inter-element and domain boundaries. Thus, in principle, one could easily capture shocks (discontinuities) with such a method. This requires, however, adaptive remeshing in order to align element boundaries and discontinuities.

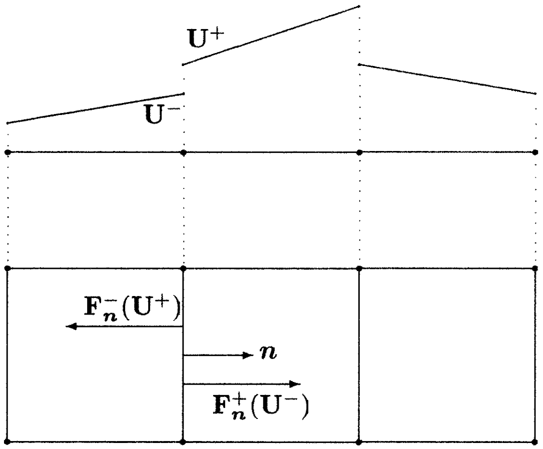


Fig. 4.8 Schematic representation of discontinuous Galerkin with flux vector splitting.

For clarity of exposition we drop the superscript h that denotes the approximation to the solution. Thus, in this section, $\mathbf{U}$ represents the finite element approximation to the conservative variables. The functional spaces needed for the resolution of the Euler equations with the discontinuous Galerkin method are the vector analogues, $\mathcal{V}(\mathcal{T}^h)$ and $\mathcal{S}_t(\mathcal{T}^h)$, of the *broken spaces* introduced in Section 3.9, namely

$$\mathcal{V}(\mathcal{T}^h) = \left\{\mathbf{W} \in \mathcal{L}_2(\Omega) \mid \mathbf{W}|_{\Omega^e} \in \mathcal{H}^2(\Omega^e)\ \forall\,\Omega^e \in \mathcal{T}^h(\Omega)\right\},$$
$$\mathcal{S}_t(\mathcal{T}^h) = \left\{\mathbf{U} \mid \mathbf{U}(\cdot,t) \in \mathcal{L}_2(\Omega),\ \mathbf{U}(\cdot,t)|_{\Omega^e} \in \mathcal{H}^2(\Omega^e)\, t \in [0,T]\,\forall\,\Omega^e \in \mathcal{T}^h(\Omega)\right\}.$$

Note that a partition $\mathcal{T}^h$ of the domain Ω is also required.

As shown in Figure 4.8, the approximation of the conservation variables in the state vector $\mathbf{U}$ is discontinuous across inter-element boundaries. To deal with discontinuous approximations, it is useful to employ the following notation:

$$\mathbf{U}^{\pm} = \lim_{\epsilon \to 0^+} \mathbf{U}(\boldsymbol{x} \pm \epsilon \boldsymbol{n})$$

where $\boldsymbol{x}$ is a point on the boundary and $\boldsymbol{n}$ the outward normal. Likewise, as indicated in Figure 4.9, the normal flux $\mathbf{F}_{\boldsymbol{n}}(\mathbf{U})$ at any point on a boundary with outward normal $\boldsymbol{n}$ is split into an inflow component $\mathbf{F}_{\boldsymbol{n}}^-$ and an outflow component $\mathbf{F}_{\boldsymbol{n}}^+$. With this notation, $\mathbf{F}_{\boldsymbol{n}}^+(\mathbf{U}^-)$ represents the flux of mass, momentum and energy in the direction $\boldsymbol{n}$, while $\mathbf{F}_{\boldsymbol{n}}^-(\mathbf{U}^+)$ is the flux in the opposite direction. Using this notation, the influx boundary condition of the Euler problem, (4.16b), can be rewritten as

$$\mathbf{F}_{\boldsymbol{n}}^-(\mathbf{U}^-) = \mathbf{G} \qquad \text{on } \Gamma\times]0,T[.$$

The space discretization of Euler equations is based on the following weak formulation: given $\mathbf{U}_0(\boldsymbol{x})$ and the normal flux $\mathbf{G}(\boldsymbol{x},t)$ on Γ, find $\mathbf{U}(\boldsymbol{x},t) \in \mathcal{S}_t(\mathcal{T}^h)$, such

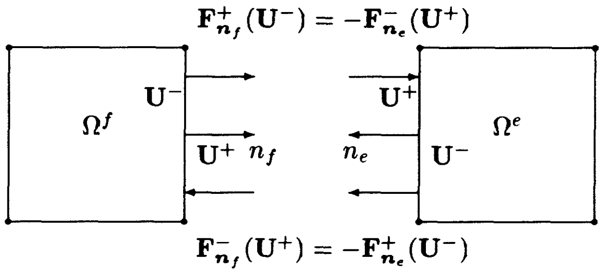

$F^+_{n_f}(U^-) = -F^-_{n_e}(U^+)$
Ω^f
U^-
U^+
n_f
n_e
U^+
U^-
Ω^e
$F^-_{n_f}(U^+) = -F^+_{n_e}(U^-)$

Fig. 4.9 Fluxes at interface between contiguous elements.

that for all weighting functions $\mathbf{W} \in \mathcal{V}(\mathcal{T}^h)$ and every element Ω^e

$$(\mathbf{W}, \mathbf{U}_t)_{\Omega^e} - (\nabla\mathbf{W}, \mathbf{F}(\mathbf{U}))_{\Omega^e} + (\mathbf{W}, \mathbf{F}^+_{\boldsymbol{n}_e}(\mathbf{U}^-))_{\partial\Omega^e}$$
$$+ (\mathbf{W}, \mathbf{F}^-_{\boldsymbol{n}_e}(\mathbf{U}^+))_{\partial\Omega^e/\Gamma} = (\mathbf{W}, \mathbf{B})_{\Omega^e} - (\mathbf{W}, \mathbf{G})_{\partial\Omega^e\cap\Gamma};$$

where

$$\mathbf{F}_{\boldsymbol{n}_e}(\mathbf{U}) = \sum_{i=1}^{n_{sd}} \mathbf{F}_i(\mathbf{U})\,[\boldsymbol{n}_e]_i.$$

For each flux vector splitting a closed form of $\mathbf{F}^{\pm}_{\boldsymbol{n}}$ is known, see Hirsch (1990, Chap. 20) for specific cases. The first two terms on the l.h.s. of the previous equation and those on the r.h.s. have the same structure as in a standard Galerkin formulation. The last two terms on the l.h.s. represent a jump condition. The jump condition enforces a weak continuity of the normal flux across the inter-element boundaries.

Explicit time-stepping algorithms are very convenient for the time discretization of the semi-discrete equations resulting from the discontinuous Galerkin method. Baumann and Oden (2000) suggest using a discontinuous space–time discretization of the Euler equations with piecewise constant approximation in time. Space–time methods of this type were discussed in Section 3.10 of Chapter 3.

Remark 4.8. The discontinuous Galerkin method makes it very easy to use different spatial approximations in adjacent elements to maximize the accuracy. Note, however, that only low-order approximations (piecewise linear or piecewise constant) should be used near sharp solution gradients. A good resolution of discontinuities can only be obtained through the use of a locally fine enough mesh of low-order elements and, as shown in the worked examples of Section 4.8, through the injection of a suitable amount of artificial viscosity. Higher-order polynomial approximations can be used in the smooth part of the flow to maximize the accuracy.

4.4.2.3 Extension of the SUPG stabilization technique Hughes and coworkers (see Hughes, Tezduyar and Brooks, 1982; Hughes and Tezduyar, 1984) present the first attempts to extend SUPG methods, see Section 2.4.1, to systems of nonlinear hyperbolic conservation laws. A breakthrough in the formulation of stabilized finite element methods for compressible flow problems came with the idea of working with symmetrized conservation laws (Hughes, Franca and Mallet, 1987). The symmetrization is accomplished by a change of variables leading to the formulation of the Euler equations in the entropy variables described in Section 4.3.2.

For practical reasons, however, it is desirable to develop stabilized finite element methods for the Euler equations which instead of entropy variables employ conservation variables. With this choice, the state equations for general gas laws are simpler to establish, the numerical implementation is easier, and a closer similarity exists with traditional finite element formulations. Hansbo (1993) was the first to introduce a formulation of the SUPG method for the Euler equations expressed in conservation variables. Let us briefly illustrate the derivation of the SUPG formulation of the Euler equations and underline the difficulty of constructing the stabilization matrix which plays the role of the stabilization parameter in the scalar convection–diffusion case discussed in Chapter 2.

Consider the initial boundary value problem associated with the Euler equations (4.16). Generalizing the formulation given in Section 2.4.1 of Chapter 2, the SUPG method for the Euler equations consists of adding to the Galerkin variational formulation, see (4.27), an element-by-element contribution depending on the local residual of equation (4.16a). This results in the following problem: for $t \in]0,T[$, find $\mathbf{U}(\boldsymbol{x},t) \in \boldsymbol{\mathcal{S}}_t$ for all $\mathbf{W} \in \boldsymbol{\mathcal{V}}$, such that $\mathbf{U}(\boldsymbol{x},0) = \mathbf{U}_0(\boldsymbol{x})$ and

$$(\mathbf{W},\mathbf{U}_t) - (\boldsymbol{\nabla}\mathbf{W},\mathbf{F}) + \sum_e \big((\mathbf{A}\cdot\boldsymbol{\nabla})\mathbf{W}, \boldsymbol{\tau}\mathcal{R}(\mathbf{U})\big)_{\Omega^e} = (\mathbf{W},\mathbf{B}) - (\mathbf{W},\mathbf{G})_\Gamma$$

where $\mathcal{R}(\mathbf{U}) = \mathbf{U}_t + \boldsymbol{\nabla}\cdot\mathbf{F} - \mathbf{B}$ is the local residual of the governing equation. Note that in complete analogy with the scalar convection–diffusion case discussed in Chapter 2, the present stabilized formulation results from the introduction of a stabilizing term governed by the matrix $\boldsymbol{\tau}$. Since

$$\boldsymbol{\nabla}\cdot\mathbf{F} = \sum_{j=1}^{n_{\mathrm{sd}}} \frac{\partial \mathbf{F}_j}{\partial x_j} = \sum_{j=1}^{n_{\mathrm{sd}}} \frac{\partial \mathbf{F}_j}{\partial \mathbf{U}}\frac{\partial \mathbf{U}}{\partial x_j} = \sum_{j=1}^{n_{\mathrm{sd}}} \mathbf{A}_j \frac{\partial \mathbf{U}}{\partial x_j} = (\mathbf{A}\cdot\boldsymbol{\nabla})\mathbf{U},$$

we note that the stabilization term includes the integral

$$\big((\mathbf{A}\cdot\boldsymbol{\nabla})\mathbf{W}, \boldsymbol{\tau}(\mathbf{A}\cdot\boldsymbol{\nabla})\mathbf{U}\big)_{\Omega^e},$$

which is responsible for adding diffusion along the characteristic directions, provided the matrix $\boldsymbol{\tau}$ is properly designed.

The structure of the stabilization matrix $\boldsymbol{\tau}$ is therefore essential in SUPG methods for compressible flow problems. Only in simple cases, such as 1D systems and/or systems amenable to diagonal form, can one find optimal definitions of the coefficients of matrix $\boldsymbol{\tau}$. An appropriate stabilization matrix $\boldsymbol{\tau}$ should be symmetric, positive definite, have dimensions of time, and scale linearly with the element size (no stabilization is needed for a fine enough mesh). In multiple dimensions, matrix $\boldsymbol{\tau}$ should be designed to introduce numerical dissipation along the characteristic directions only and not transversely. Its construction is, in general, not a trivial task, because the Jacobian matrices $\mathbf{A}_i$, $i = 1, \ldots, \mathrm{n_{sd}}$, are not simultaneously diagonalizable in multidimensional Euler equations.

In 1D, the system of Euler equations can be diagonalized as explained in Section 4.3.3. Then, the scalar stabilization parameter τ, see Section 2.4.3, can be used for each individual equation. In pure convection, $\tau = h/(2a)$, see both (2.64) and (2.65). For Euler equations in 1D this definition can be generalized, producing a stabilization matrix $\boldsymbol{\tau}$, if we recall the diagonalization of $\mathbf{A} = \mathbf{R}\boldsymbol{\Lambda}\mathbf{R}^{-1}$, see Section 4.3.3, and basic algebra definitions, such as $\mathbf{A}^p = \mathbf{R}\boldsymbol{\Lambda}^p\mathbf{R}^{-1}$ and $|\mathbf{A}| = \mathbf{R}\,|\boldsymbol{\Lambda}|\,\mathbf{R}^{-1}$,

$$\boldsymbol{\tau} = \frac{h}{2}\left(\mathbf{A}^2\right)^{-\frac{1}{2}} = \frac{h}{2}|\mathbf{A}|^{-1}.$$

We have assumed that all the eigenvalues, see (4.24), are non-zero. If there is any zero eigenvalue it is suppressed, as well as its corresponding eigenvector. This produces a formula similar to the previous one, namely

$$\boldsymbol{\tau} = \frac{h}{2}\,\tilde{\mathbf{R}}\,|\tilde{\boldsymbol{\Lambda}}|^{-1}\tilde{\mathbf{R}}^{-1},$$

where $\tilde{\boldsymbol{\Lambda}}$ only includes the non-zero eigenvalues and $\tilde{\mathbf{R}}$ their corresponding right eigenvectors.

Such a definition of $\boldsymbol{\tau}$ induces, in general, a non-symmetric positive definite matrix ($\mathbf{A}$ is non-symmetric). Hughes and Mallet (1986a) generalize the previous expression to multidimensional situations in the framework of a symmetric system of equations,

$$\boldsymbol{\tau} = \frac{h}{2}\left(\sum_{j=1}^{\mathrm{n_{sd}}}\mathbf{A}_j^2\right)^{-\frac{1}{2}}.$$

Furthermore, they extend it to non-regular meshes using

$$\boldsymbol{\tau} = \left(\sum_{j=1}^{\mathrm{n_{sd}}}\mathbf{B}_j^2\right)^{-1/2}, \text{ where } \quad \mathbf{B}_j = \sum_{j=1}^{\mathrm{n_{sd}}}\frac{\partial \xi_j}{\partial x_i}\,\mathbf{A}_i$$

takes into account the parametric mapping $\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{\xi})$ between the actual coordinates, $\boldsymbol{x}$, and the normalized local coordinates, $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_{\mathrm{n_{sd}}})^T$. In practice, the stabilization matrix, $\boldsymbol{\tau}$, is assumed constant in each individual element. If there is any zero eigenvalue the same procedure as before is applied.

Finally, Soulaïmani and Fortin (1994) show that the previous expression is also valid for non-symmetric Euler formulations such as the conservation form used in this chapter and employ the equivalence between norms to produce a simpler formula

$$\boldsymbol{\tau} = \left(\sum_{j=1}^{\mathrm{n_{sd}}}|\mathbf{B}_j|\right)^{-1}.$$

In practice, it is possible to obtain an algebraic expression for $\boldsymbol{\tau}^{-1}$ to avoid the solution of an eigenvalue problem, see the cited references.

4.4.2.4 Stabilized space–time formulation Here we illustrate the use of stabilized formulations in the context of a space–time formulation. As in Section 3.10, to which we refer for the notation, we consider piecewise continuous approximations in space and discontinuous approximations in time. Accordingly, we introduce space–time slabs $Q^n = \Omega \times I^n$, where $I^n =]t^n, t^{n+1}[$, and space–time element domains $Q^n_e = \Omega^e \times I^n$, $e = 1, \ldots, \mathrm{n_{el}}$. The finite element space for the trial and weighting functions is then defined as follows:

$$\mathcal{S}^h = \bigcup_{n=0}^{\mathrm{n_{st}}-1} \mathcal{S}^h_n, \quad \mathcal{S}^h_n = \{\mathbf{U}^h \mid \mathbf{U}^h \in [\mathcal{C}(Q^n)]^{\mathrm{n_{sd}}+2}, \, \mathbf{U}^h|_{Q^n_e} \in [\mathcal{P}_k(Q^n_e)]^{\mathrm{n_{sd}}+2}, \, \forall e\}.$$

It is the vector analogue of (3.62) with no Dirichlet boundary conditions.

With reference to the Euler equations in conservative variables, system (4.16), the GLS method can now be formulated as follows: for $n = 0, 1, \ldots, \mathrm{n_{st}} - 1$, find $\mathbf{U}^h \in \mathcal{S}^h_n$ such that for all $\mathbf{W} \in \mathcal{S}^h_n$

$$\iint_{Q^n} (\mathbf{W}^h)^T \left(\mathbf{U}^h_t + \boldsymbol{\nabla}\cdot\mathbf{F}^h - \mathbf{B}^h\right) d\Omega\, dt$$
$$+ \sum_e \iint_{Q^n_e} (\mathbf{W}^h_t + (\mathbf{A}\cdot\boldsymbol{\nabla})\mathbf{W}^h)^T \boldsymbol{\tau} (\mathbf{U}^h_t + \boldsymbol{\nabla}\cdot\mathbf{F}^h - \mathbf{B}^h)\, d\Omega\, dt$$
$$+ \int_\Omega (\mathbf{W}^h(t^n_+))^T (\mathbf{U}^h(t^n_+) - \mathbf{U}^h(t^n_-))\, d\Omega = 0,$$

where $\mathbf{U}^h(t^0_-) = \mathbf{U}_0$. The last integral in this equation imposes a weakly enforced continuity condition across the slab interfaces at t^n, see Section 3.10, and it is the mechanism by which information is propagated from one slab to another.

For instance, in the case of a (discontinuous) piecewise constant approximation in time, all time derivatives in the previous equation vanish, the GLS formulation coincides with SUPG, and an implicit approximation over the time interval $\Delta t = t^{n+1} - t^n$ is obtained

$$\Delta t \int_\Omega (\mathbf{W}^h)^T ((\mathbf{A}\cdot\boldsymbol{\nabla})\mathbf{U}^h_{n+1} - \mathbf{B}^h_{n+1})\, d\Omega\, dt$$
$$+ \Delta t \sum_e \int_{\Omega^e} ((\mathbf{A}\cdot\boldsymbol{\nabla})\mathbf{W}^h)^T \boldsymbol{\tau} ((\mathbf{A}\cdot\boldsymbol{\nabla})\mathbf{U}^h_{n+1} - \mathbf{B}^h_{n+1})\, d\Omega\, dt$$
$$+ \int_\Omega (\mathbf{W}^h)^T (\mathbf{U}^h_{n+1} - \mathbf{U}^h_n)\, d\Omega = 0.$$

In deriving this expression we have used the quasi-linear form (4.19) of the Euler equations to replace the divergence of the flux, $\boldsymbol{\nabla}\cdot\mathbf{F}^h = (\mathbf{A}\cdot\boldsymbol{\nabla})\mathbf{U}^h$. The following convention has been used: $U^h_{n+1} = U^h(t^{n+1}) := U^h(t)$ for $t \in]t^n, t^{n+1}[$. If the values at t^n are known, the only unknown in the previous equation is $\mathbf{U}^h_{n+1}$. An implicit scheme is thus obtained that is reminiscent of the first-order backward Euler method.

4.4.2.5 Residual decomposition technique Alternative ways of extending the SUPG formulation to the Euler equations have been suggested using the so-called multidimensional residual decomposition technique. This technique was originated by Roe and Deconinck and their co-workers and a detailed account of it may, for instance, be found in the thesis of Carette (1997) and the references therein. The main idea consists of decomposing the elementwise residual of the system of Euler equations into a set of scalar components which can be treated separately with stabilized convection schemes. This formulation allows a natural equation-by-equation definition of the stabilization parameter.

In 1D, the Euler equations can be diagonalized and, as seen in Section 4.3.3, they can be decomposed into three scalar convection equations associated with the characteristics defined in (4.25). In 2D and 3D, the Euler equations can no longer be diagonalized. Waves can travel in an infinite number of directions and the decomposition of the flux divergence into scalar waves is no longer unique. However, in the framework of the multidimensional residual decomposition approach, several techniques have been developed to decompose the system of Euler equations into scalar contributions just as in the 1D case. The basic ingredient of the method is an *approximate diagonalization* along the lines suggested by Roe (1986) and Deconinck, Hirsch and Peuteman (1986) in their characteristic-based approach. Various techniques are actually available to decompose the multidimensional Euler equations into a set of scalar convection equations. They are described in detail in the theses of Paill ère (1995) and Carette (1997) and the references therein to the original works of Roe, Deconinck, Hirsch, and their collaborators.

4.5 NUMERICAL TREATMENT OF SHOCKS

4.5.1 Introduction

One of the most striking features of compressible fluid flow is the presence of shock waves. A shock wave is produced in a fluid when a succession of compression waves, each propagating faster than its predecessor, pile up, determining an abrupt transition of the field variables.

In the solution of inviscid compressible equations, shocks are seen as discontinuities of the solution, that is surfaces across which the fluid variables are discontinuous. In nature, shocks have a finite thickness, usually very small, associated with the mean free path of the particles of the fluid. Shock waves produce entropy; thus over the thickness of the shock the kinetic energy of the short wavelengths is transferred into internal energy.

From a mathematical point of view, the Rankine–Hugoniot jump conditions, which relate fluid variables across the shock, guarantee the solution of the inviscid equations in the case of a shock discontinuity. However, the numerical computation of shock flows via a direct application of the Rankine–Hugoniot jump conditions would simply be prohibitive. Fortunately, it can be shown that, provided appropriate *conservative* *discretization schemes* are employed, the Rankine–Hugoniot jump conditions are satisfied by numerical solutions.

The key idea for the numerical treatment of shocks and other flow discontinuities is the introduction into the considered conservation law equation of an artificial (numerical) viscosity term. This term has the effect of broadening the thickness of a shock over a few elements in the mesh, thereby smearing the discontinuity and removing spurious oscillations at its front. Discontinuity surfaces are thus replaced by thin transition layers over which the flow variables (density, pressure, velocity) vary rapidly, but continuously. Provided the structure of the artificial dissipation terms is correctly chosen, the Rankine–Hugoniot conditions are satisfied without degrading the results over the remainder of the flow field. Recall the concept of vanishing viscosity solution introduced in Section 4.2.2.

There are several ways of introducing terms representing an added dissipation in the numerical schemes resulting from the finite element method. In Chapter 2 added numerical diffusion was already introduced in the context of stabilization of convection-dominated problems. Among early *shock-capturing techniques* for the Euler equations, mention must be made of the method of pseudo-viscous pressure due to von Neumann and Richtmyer (1950). This method is discussed in Section 4.5.2, together with another early method for treating shocks due to Lapidus (1967). These early methods introduce free parameters whose calibration is not trivial, because they must introduce just enough numerical viscosity to preserve monotonicity without spreading shocks over too many grid points.

Higher-order shock-capturing schemes have been developed for the approximation of nonlinear hyperbolic conservation laws. They produce non-oscillatory discontinuities by a different approach than explicitly adding an artificial viscosity term to the considered hyperbolic conservation law. The damping mechanism is implicitly introduced by choosing an appropriate form of the discrete equations. Here, mention must be made, among others, of the works of van Leer (1974), Harten (1983), Roe (1984), Sweby (1984) and Woodward and Colella (1984). These modern schemes, called *high-resolution schemes*, were briefly introduced in Section 3.7 and are further discussed in Section 4.5.3. They are based on conditions less severe than monotonicity and have the following properties:

- They are at least of second-order accuracy in smooth parts of the flow.

- They sharply resolve discontinuities without generating spurious oscillations and can be modified to produce solutions satisfying the entropy condition.

- They do not need explicit artificial viscosity such as in the pseudo-viscous pressure method; damping is implicitly included in the form of the discrete equations through the definition of appropriate flux functions incorporating an upwind effect.

We close the discussion of shock-capturing techniques by mentioning in Section 4.5.3.3 that locally monotone solutions can also be obtained by a suitable choice of the stabilization matrix in the SUPG and GLS methods.

4.5.2 Early artificial diffusion methods

Early methods for the numerical treatment of shocks and other flow discontinuities consisted of introducing a suitable amount of artificial diffusion near sharp solution gradients. Among the various artificial diffusion methods proposed to produce a local smoothing of the solution to the Euler equations, we have selected the widely used method of pseudo-viscous pressure and the Lapidus viscosity.

4.5.2.1 Method of pseudo-viscous pressure The treatment of shocks by the method of pseudo-viscous pressure due to von Neumann and Richtmyer (1950) consists of rewriting the equations in Section 4.3.1 of conservation of momentum and total energy in the form

$$\begin{aligned}\frac{\partial \rho \boldsymbol{v}}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{v} \otimes \boldsymbol{v} + (p+q)\mathbf{I}) &= \rho \boldsymbol{b}\\ \frac{\partial \rho E}{\partial t} + \boldsymbol{\nabla} \cdot \big((\rho E + p + q)\boldsymbol{v}\big) &= \boldsymbol{v} \cdot \rho \boldsymbol{b},\end{aligned}$$

where p is the fluid static pressure and the scalar q added to the fluid pressure is called *pseudo-viscous pressure.* This additional term is defined below and represents the dissipative mechanism. Its value should be negligible everywhere in the flow domain, except in the vicinity of a shock.

Since dissipation is being introduced for purely numerical reasons, the pseudo-viscous pressure q can be defined through any appropriate function of p, ρ, $\boldsymbol{v}$, etc., and their derivatives. Nevertheless, the following conditions must be satisfied by any artificial viscosity technique:

1. The conservation equations must have solutions free of discontinuities (i.e., the conservation variables must be continuous across the shock).
2. The shock thickness must be of the order of magnitude of the mesh size h.
3. The effect of the terms containing q must be negligible outside the shocks.
4. The Rankine–Hugoniot jump conditions must be verified when the dimensions characterizing the flow are large compared with the shock thickness.

von Neumann and Richtmyer (1950) show that in 2D the choice

$$q = \begin{cases} c_Q\, \rho\, A_e\, (\boldsymbol{\nabla} \cdot \boldsymbol{v})^2 & \text{if } \boldsymbol{\nabla} \cdot \boldsymbol{v} < 0,\\ 0 & \text{if } \boldsymbol{\nabla} \cdot \boldsymbol{v} \geq 0,\end{cases}$$

allows all the above conditions to be satisfied. In this expression A_e is the element area and c_Q a dimensionless constant (close to unity). The constant c_Q controls the spread of the shock and it is determined empirically. Note from the previous definition that the dissipation mechanism introduced by q is a nonlinear viscosity.

Frequently, and in particular for modeling liquids, a so-called linear pseudo-viscosity is required as suggested by Wilkins (1969). It is added to the previous definition of q,

$$q_L = \begin{cases} c_L\, c\, \rho\, A_e^{1/2}\, |\boldsymbol{\nabla} \cdot \boldsymbol{v}| & \text{if } \boldsymbol{\nabla} \cdot \boldsymbol{v} < 0,\\ 0 & \text{if } \boldsymbol{\nabla} \cdot \boldsymbol{v} \geq 0.\end{cases}$$

This linear pseudo-viscosity is added to the previously defined nonlinear pseudo-viscosity to reduce spurious oscillations; c_L must be small to avoid excessive diffusion of the shock wave front. A value of c_L of the order of 0.05 is standard.

The implementation of the pseudo-viscous pressure in the framework of the finite element method is simple. The pseudo-viscous pressure q is added to the static fluid pressure p in the discrete form of the momentum and energy equations.

4.5.2.2 Lapidus viscosity Another widely used artificial viscosity method to capture shocks was proposed by Lapidus (1967) for finite difference context. In 1D, the added viscosity normally acts as a diffusion operator

$$\nu_{\text{Lap}} = c_{\text{Lap}}\, h^2 |\partial v/\partial x|,$$

where c_{Lap} is a user-specified coefficient, which normally varies between 0.0 and 2.0. In order to extend this concept to multiple dimensions Löhner, Morgan and Peraire (1985) and Peraire (1986) constructed a local coordinate system oriented in the direction of $\boldsymbol{\ell}$, the unit vector in the direction of maximum change in the absolute value of the velocity $\|\boldsymbol{v}\|$, that is

$$\ell_i = \left[\frac{\partial\|\boldsymbol{v}\|}{\partial x_i}\right] \Big/ \left[\left(\sum_{j=1}^{\mathrm{n_{sd}}} \frac{\partial\|\boldsymbol{v}\|}{\partial x_j}\frac{\partial\|\boldsymbol{v}\|}{\partial x_j}\right)^{1/2}\right], \quad i = 1,\ldots,\mathrm{n_{sd}},$$

or

$$\boldsymbol{\ell} = \frac{\boldsymbol{\nabla}\|\boldsymbol{v}\|}{\|(\boldsymbol{\nabla}\|\boldsymbol{v}\|)\|}.$$

Then, with the added viscosity

$$\nu_{\text{Lap}} = c_{\text{Lap}}\, h^2 \left|\frac{\partial(\boldsymbol{v}\cdot\boldsymbol{\ell})}{\partial\boldsymbol{\ell}}\right| = c_{\text{Lap}}\, h^2 |\boldsymbol{\ell}\cdot\boldsymbol{\nabla}(\boldsymbol{v}\cdot\boldsymbol{\ell})|,$$

one can finally define the added diffusion term used to smooth the velocities

$$\frac{\partial}{\partial\boldsymbol{\ell}}\left(\nu_{\text{Lap}}\frac{\partial\boldsymbol{v}}{\partial\boldsymbol{\ell}}\right) = (\boldsymbol{\ell}\cdot\boldsymbol{\nabla})\big(\nu_{\text{Lap}}\,(\boldsymbol{\ell}\cdot\boldsymbol{\nabla})\boldsymbol{u}\big).$$

This artificial diffusion method has the essential features required in these techniques. The sensing function, which here is $|\boldsymbol{\ell}\cdot\boldsymbol{\nabla}(\boldsymbol{v}\cdot\boldsymbol{\ell})|$, induces a zero added viscosity when velocity is smooth. It is invariant under coordinate rotation; thus it is independent of the mesh orientation. It does not smear shear layers (contact discontinuities) because velocity is continuous across them and neither boundary layers because $\boldsymbol{v}$ and $\boldsymbol{\ell}$ are orthogonal in these regions.

To limit the addition of artificial viscosity close to zones of sharp gradients, other sensors can be employed which are activated, for instance, by the variation of the pressure gradient (see the exercises in Section 4.8). Moreover, instead of explicitly adding artificial viscosity terms, a damping mechanism may be introduced implicitly by choosing an appropriate form of the discrete equations. This leads to so-called *high-resolution schemes*, some of which are illustrated in the next section.

4.5.3 High-resolution methods

As already mentioned, monotone schemes are generally too dissipative and cannot produce accurate solutions for complex flow problems. Higher-order shock-capturing schemes are introduced for the purpose of adding minimal numerical dissipation and to give non-oscillatory solutions in the presence of steep solution gradients.

As explained in Section 3.7, the basic idea underlying high-order shock-capturing methods is to produce a high-order method in the smooth part of the flow and to modify it by adding numerical dissipation only in the neighborhood of a discontinuity. Modern methods use high-order TVD schemes which introduce a controlled amount of nonlinear dissipation to give non-oscillatory shocks and ensure convergence to the entropy solution. The term nonlinear dissipation means that the diffusion coefficient depends on the local behavior of the solution. Thus, diffusion should be larger near discontinuities than in smooth regions of the flow.

The guiding idea behind the design of TVD schemes for inviscid flow problems is that physical solutions to the scalar hyperbolic equations do not allow the appearance of any new extremum in the evolution of the flow variables. In a TVD scheme, the total variation of the numerical solution is controlled in a nonlinear way, such as to prevent the appearance of any new extremum. High-resolution TVD schemes in the finite difference context have been developed by van Leer (1974), Harten (1983), Roe (1984), Sweby (1984) and Yee (1987). A detailed account of the theory underlying modern shock-capturing schemes can be found in the textbooks by Hirsch (1990), LeVeque (1992), Quarteroni and Valli (1994) and Morton (1996). Note that the adaptation of TVD schemes for use in connection with finite elements is still an area of active development, see Donea et al. (1988) for an early application.

The critical issue in the design of high-order TVD schemes is to introduce enough dissipation to preserve monotonicity without affecting the level of accuracy away from the flow discontinuities. To this end, most schemes incorporate a mechanism which automatically controls the amount of added numerical dissipation. Such mechanisms are often in the form of *limiters* which impose constraints on the gradient of the considered dependent variable (*slope limiters*) or on the flux function (*flux limiters*). Some of the approaches to obtain a better resolution of flow discontinuities are briefly presented in the next paragraphs.

4.5.3.1 Flux-limiter method Boris and Book (1997) and van Leer (1974) introduced high-resolution schemes using nonlinear flux limiters. They generally require a first-order upwind scheme and an anti-diffusion mechanism. The latter compensates the large amount of diffusion introduced by the first-order scheme and thus induces sharper resolution of discontinuities. Moreover, high accuracy in the smooth parts of the flow and desirable properties, in particular satisfaction of the entropy condition, are preserved. Such schemes satisfy the basic requirements of a good convection scheme: they are TVD, conservative and less diffusive than the simple upwind schemes.

We illustrate this concept of flux limiter on a 1D scalar convection equation:

$$u_t + f_x = 0, \qquad \text{with} \quad f = a\,u.$$

The forward Euler time discretization of this equation can be expressed at node i in terms of the convective flux as

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{h}\left(f_{i+1/2}^n - f_{i-1/2}^n\right).$$

In the case of a centered (second-order) approximation of the flux term, one has

$$\left[f_{i+1/2}^n\right]_{\text{cent}} = \frac{1}{2}\left(f_i^n + f_{i+1}^n\right), \qquad \left[f_{i-1/2}^n\right]_{\text{cent}} = \frac{1}{2}\left(f_i^n + f_{i-1}^n\right).$$

These approximations give rise to instabilities. But, a first-order upwind approximation based upon

$$\left[f_{i+1/2}^n\right]_{\text{up}} = \frac{1}{2}\Big(\left(a_{i+1/2}^n + |a_{i+1/2}^n|\right) u_i^n + \left(a_{i+1/2}^n - |a_{i+1/2}^n|\right) u_{i+1}^n\Big)$$

generally introduces an excessive numerical diffusion. Then, an optimal strategy consists in using the second-order numerical flux in smooth regions of the flow, and the monotonic upwind method in the vicinity of discontinuities. This procedure is best implemented by introducing *limiter schemes* based on the local gradient of the solution u (slope limiters), or of the flux function f (flux limiters). In practice, a flux-limiter method uses a flux function of the form

$$\left[f_{i+1/2}^n\right]_{\text{lim}} = \left[f_{i+1/2}^n\right]_{\text{up}} + \Phi(r_{i+1/2})\Big(\left[f_{i+1/2}^n\right]_{\text{cent}} - \left[f_{i+1/2}^n\right]_{\text{up}}\Big), \tag{4.29}$$

where

$$r_{i+1/2} = \begin{cases} (u_i^n - u_{i-1}^n)/(u_{i+1}^n - u_i^n) & \text{if } a_{i+1/2} \geq 0, \\ (u_{i+2}^n - u_{i+1}^n)/(u_{i+1}^n - u_i^n) & \text{if } a_{i+1/2} < 0. \end{cases}$$

The term in equation (4.29) depending on the function $\Phi(r)$ is called anti-diffusive flux. It compensates the high numerical diffusion introduced by the upwind approximation. For stability, flux-limiter schemes must be TVD. This imposes the following restrictions on the function $\Phi(r)$:

$$0 \leq \frac{\Phi(r)}{r} \leq 2 \qquad \text{and} \qquad 0 \leq \Phi(r) \leq 2.$$

Note, however, that the function $\Phi(r)$ is normally selected so that $0 \leq \Phi(r) \leq 1$. Two classical examples are the superbee limiter of Roe (1985) for which

$$\Phi(r) = \max\big(0, \min(1, 2r), \min(2, r)\big),$$

and the limiter suggested by van Leer (1974):

$$\Phi(r) = \frac{r + |r|}{1 + |r|}.$$

4.5.3.2 Practical implementation of high-resolution schemes As an example of the previous discussion, we describe a method (see Donea et al., 1988), which introduces a locally controlled dissipation, in the two-step Taylor–Galerkin method (TG2). This locally controlled dissipation induces a non-oscillatory scheme for the representation of shocks. In complete analogy with the idea behind the flux-limiter method discussed in the previous section, the proposed modification combines the good resolution offered by a second-order time scheme in the regular part of the flow and the capability to damp out the non-physical oscillations in the vicinity of flow discontinuities.

Suppose we want to apply the two-step TG2 method, see Section 4.2.3.2, to the Euler equations, (4.16a). After Galerkin spatial discretization of the second step, the following algebraic system is obtained:

$$\mathbf{M}\left(\mathbf{U}^{n+1} - \mathbf{U}^{n}\right) = \Delta t\, \mathbf{r}^{n+1/2} \tag{4.30}$$

where the mass matrix $\mathbf{M}$ has dimensions $\mathrm{n_{eq}} \times (\mathrm{n_{sd}}+2)$ and vector $\mathbf{r}^{n+1/2}$ accounts for the mid-step value of the non-inertia terms, see (4.12). Note that, in (4.30), $\mathbf{M}$ could, in principle, be replaced by the diagonal matrix $\mathbf{M}^{L}$ obtained by row sum; a second-order scheme would also be obtained but, as seen in 3.5.2, with poor phase properties.

To reduce the second-order scheme (TG2) to first-order near shocks, we replace the consistent mass matrix $\mathbf{M}$ acting on $\mathbf{U}^{n+1}$ with the diagonal matrix $\mathbf{M}^{L}$. This substitution produces the first-order accurate scheme

$$\mathbf{M}^{L}\mathbf{U}^{n+1} = \mathbf{M}\mathbf{U}^{n} + \Delta t\, \mathbf{r}^{n+1/2},$$

which can be rewritten as

$$\mathbf{M}^{L}\mathbf{U}^{n+1} = \mathbf{M}^{L}\mathbf{U}^{n} + \left(\mathbf{M} - \mathbf{M}^{L}\right)\mathbf{U}^{n} + \Delta t\, \mathbf{r}^{n+1/2}. \tag{4.31}$$

Remark 4.9. The second term on the r.h.s. of (4.31), which contains the difference between the consistent mass matrix and its diagonal counterpart, represents an added dissipation. For instance, the scalar 1D linear convection equation scheme associated with (4.31) delivers the following discrete equation at an interior node j:

$$u_j^{n+1} = u_j^n - \frac{1}{2}C\left(u_{j+1}^n - u_{j-1}^n\right) + \left(\frac{1}{2}C^2 + \frac{1}{6}\right)\left(u_{j-1}^n - 2u_j^n + u_{j+1}^n\right),$$

where C is the Courant number. This discrete equation is clearly of the Lax–Wendroff type with an added dissipative term. It is stable for $|C| \leq \sqrt{2/3}$ and monotone over this interval, as shown by Harten, Hyman and Lax (1976).

In order to modulate the added dissipation and thereby better mimic the flux-limiter method, in the sense of equation (4.29), we introduce a parameter d in the diffusion term of the first-order scheme (4.31). That is,

$$\mathbf{M}^{L}\mathbf{U}^{n+1} = \mathbf{M}^{L}\mathbf{U}^{n} + d\left(\mathbf{M} - \mathbf{M}^{L}\right)\mathbf{U}^{n} + \Delta t\, \mathbf{r}^{n+1/2}. \tag{4.32}$$

Note the conceptual relation between d and the anti-diffusive flux: $d = 1 - \Phi$. A second-order scheme is obtained for $d = 0$ and a first-order method with maximum dissipative effect is obtained when $d = 1$.

Furthermore, in order to separate the effect of convective transport from that associated with the added dissipation (and to recover the benefits of the consistent mass matrix), it is appropriate to implement the first-order scheme (4.32) according to the following two-stage procedure:

$$\mathbf{M}(\mathbf{U}^* - \mathbf{U}^n) = \Delta t\, \mathbf{r}^{n+1/2}, \tag{4.33a}$$

$$\mathbf{M}^L(\mathbf{U}^{n+1} - \mathbf{U}^*) = d(\mathbf{M} - \mathbf{M}^L)\mathbf{U}^*. \tag{4.33b}$$

The first stage, (4.33a), corresponds to the second-order Taylor–Galerkin scheme (4.30) and is characterized by the same (complex) amplification factor. The second stage, (4.33b), only introduces a multiplicative coefficient into the amplification factor of the first stage. The advantage of the two-stage procedure is to preserve the second-order phase accuracy of the TG2 method in the smooth part of the flow, while facilitating the introduction of a modulated dissipation around shocks.

The dissipative term on the r.h.s. of (4.33b) must be further modified in order to produce a *local modulation* of the added numerical dissipation. This is achieved by replacing the global parameter d with a local parameter defined in terms of a sensor which recognizes the discontinuities in the flow. At a given finite element node A the dissipative term of the first-order scheme (4.33b) takes the form

$$d \sum_B M_{AB} U_B^* - M_{AA}^L U_A^*, \tag{4.34a}$$

where the summation extends to all nodes B topologically connected to node A. Since the lumped-mass matrix is obtained by row sum $M_{AA}^L = \sum_B M_{AB}$, we may rewrite the dissipative term (4.34a) as

$$\sum_B d_{AB}\, M_{AB}(U_B^* - U_A^*), \tag{4.34b}$$

where $0 \leq d_{AB} \leq 1$ is the required local modulation coefficient. Observe that the dissipation operator consists of segment contributions (the sides of the elements for a triangular mesh); moreover, the 1D character of segments allows an easy adaptation to a finite element context of the procedures developed in 1D to limit the dissipative effect.

To illustrate the proposed limitation procedure, we consider its application in connection with the artificial viscosity method. As already mentioned, the local parameter d_{AB} must be expressed in terms of a sensor which recognizes the discontinuities in the flow. As suggested by Jameson (1985), an effective sensor to detect the presence of shocks can be constructed by considering the second derivative of the pressure. To this end, for a given segment A–B one introduces the quantities

$$d_A = \left| \frac{p_B - 2p_A + p_{A_-}}{p_B + 2p_A + p_{A_-}} \right| \quad \text{and} \quad d_B = \left| \frac{p_{B_+} - 2p_B + p_A}{p_{B_+} + 2p_B + p_A} \right|, \tag{4.35a}$$

where

$$p_{A_-} = p_B - 2(\boldsymbol{x}_B - \boldsymbol{x}_A) \cdot [\boldsymbol{\nabla} p]_A \quad \text{and} \quad p_{B_+} = p_A + 2(\boldsymbol{x}_B - \boldsymbol{x}_A) \cdot [\boldsymbol{\nabla} p]_B. \tag{4.35b}$$

Then, the artificial viscosity coefficient for segment A–B is evaluated from the relation

$$d_{AB} = \min(\chi \max(d_A, d_B), 1), \tag{4.35c}$$

where χ is an adjustable parameter. In this way d_{AB} is maximum on both sides of a shock and zero inside it. Note the presence of a free parameter in this last expression. This is avoided in more elaborate TVD schemes, such as, for instance, those developed by Harten (1983) and Yee (1987).

4.5.3.3 Nonlinear artificial dissipation for stabilized methods Stabilized methods such as SUPG, see Chapter 2, guarantee that spurious oscillations created in the neighborhood of sharp solution gradients do not propagate all over the computational domain. Recall that this is not the case with the standard Galerkin method. Nevertheless, stabilization techniques such as SUPG are linear high-order methods and consequently some oscillations will remain close to near-discontinuities (recall Godunov's theorem). This is due to the absence of control of the solution gradient in directions other than along the streamlines. This has motivated the development of nonlinear discontinuity-capturing versions of stabilized methods, with the objective of obtaining monotone solutions in the presence of sharp layers. For simplicity, we shall illustrate the discontinuity-capturing version of the SUPG method with reference to a scalar convection–diffusion problem.

To suppress the residual oscillations near strong solution gradients an additional nonlinear diffusion term is added. That is, an extra consistent (residual-based) term is added apart from the standard stabilization term presented in Section 4.4.2.3. Recall that in scalar form this stabilization term is $\sum_e \big(\mathcal{P}(w), \tau\, \mathcal{R}(u)\big)_{\Omega^e}$ where $\mathcal{R}(u)$ is the residual of the differential equation. The added diffusion can be isotropic or anisotropic. The former has the same structure as standard SUPG, see equation (2.57), namely $\mathcal{P}(w) = \boldsymbol{a} \cdot \boldsymbol{\nabla} w$. Thus, the shock-capturing term takes the form

$$\sum_e \big(\hat{\boldsymbol{a}} \cdot \boldsymbol{\nabla} w, \hat{\tau}\, \mathcal{R}(u)\big)_{\Omega^e}, \tag{4.36}$$

where $\hat{\boldsymbol{a}}$ is a newly defined vector field. For instance, Hughes and Mallet (1986b) propose using the projection of $\boldsymbol{a}$ onto $\boldsymbol{\nabla} u$. That is, in the case $\|\boldsymbol{\nabla} u\| \neq 0$, the projected velocity is given by

$$\hat{\boldsymbol{a}} = \frac{\boldsymbol{a} \cdot \boldsymbol{\nabla} u}{\|\boldsymbol{\nabla} u\|^2} \boldsymbol{\nabla} u.$$

The new stabilization parameter $\hat{\tau}$ depends on the standard τ, see Section 2.4.3. Recall that for each element

$$\tau = \beta \frac{h}{2\|\boldsymbol{a}\|},$$

where h is a measure of the element size, whereas β is a function of the mesh Péclet number defined by

$$\mathit{Pe} = \frac{h\|\boldsymbol{a}\|}{2\nu};$$

where ν is the physical diffusivity. As seen in Chapter 2, in 1D convection–diffusion the optimal choice for β is given by

$$\beta = \coth\,\mathit{Pe} - \frac{1}{\mathit{Pe}}$$

and the choice $\beta = 1$ corresponds to the full upwind case, which maximizes the stabilization parameter τ.

The stability parameter $\hat{\tau}$ controlling the discontinuity-capturing term can be defined in a number of ways. Hughes and Mallet (1986b) initially suggest using $\hat{\tau} = \tau$, but this choice is overly diffusive when the convection velocity and the gradient of u are parallel vectors, that is when $\hat{\boldsymbol{a}} = \boldsymbol{a}$. It amounts to double the SUPG stabilization term. They subsequently propose using

$$\hat{\tau} = \max(0, \tau_{\hat{\mathrm{a}}} - \tau), \qquad \text{with} \quad \tau_{\hat{\mathrm{a}}} = \beta\,\frac{h}{2\|\hat{\boldsymbol{a}}\|}. \tag{4.37}$$

This method produces an added diffusion of form, see equation (4.36),

$$\sum_e \left(\hat{\boldsymbol{a}}\cdot\boldsymbol{\nabla}w, \hat{\tau}\,\mathcal{R}(u)\right)_{\Omega^e} = \sum_e \int_{\Omega^e} \left[\hat{\tau}\frac{\boldsymbol{a}\cdot\boldsymbol{\nabla}u}{\|\boldsymbol{\nabla}u\|^2}\mathcal{R}(u)\right]\boldsymbol{\nabla}u\cdot\boldsymbol{\nabla}w\,d\Omega, \tag{4.38}$$

where the bracketed term, which acts as nonlinear diffusion, may be negative.

An alternative approach is to change the definition of $\hat{\boldsymbol{a}}$ and still use a residual-based approach. Johnson, Szepessy and Hansbo (1990), Hansbo and Johnson (1991) and Galeão and do Carmo (1988) suggest using the following definition of $\hat{\boldsymbol{a}}$:

$$\hat{\boldsymbol{a}} = \frac{\mathcal{R}(u)}{\|\boldsymbol{\nabla}u\|^2}\,\boldsymbol{\nabla}u,$$

where $\mathcal{R}(u)$ is the residual of the governing equation. The nonlinear artificial viscosity introduced by this method is given by

$$\nu_{\mathtt{shock}} = \begin{cases} \hat{\tau}\,\mathcal{R}(u)^2/\,\|\boldsymbol{\nabla}u\|^2 & \text{if } \|\boldsymbol{\nabla}u\| \neq 0, \\ 0 & \text{otherwise.} \end{cases}$$

Here, parameter $\hat{\tau}$ is computed as in equation (4.37) but making use of the new definition of $\hat{\boldsymbol{a}}$. Finally, note that the nonlinearity of the discontinuity-capturing term always originates from the definition of $\hat{\boldsymbol{a}}$, which depends on $\boldsymbol{\nabla}u$.

Codina (1993a) suggests, based on the discrete maximum principle, an anisotropic nonlinear viscosity that should act in the crosswind direction only. The objective is to add diffusion without affecting the streamline dissipation introduced by the linear form of the SUPG method. This requires, first, a nonlinear added diffusion,

$$\nu_{\mathtt{shock}} = \begin{cases} \frac{1}{2}\alpha h|\mathcal{R}(u)|/\|\boldsymbol{\nabla}u\| & \text{if } \|\boldsymbol{\nabla}u\| \neq 0, \\ 0 & \text{otherwise,} \end{cases}$$

where $\alpha = \max\left(0, \chi - \frac{2\nu}{h\|\boldsymbol{a}\|}\right)$. The recommended value of the constant χ for linear elements is $\chi = 0.7$. Second, it also requires a tensorial structure in the added term

$$\sum_{e=1}^{n_{\mathrm{el}}} \int_{\Omega^e} \nu_{\mathrm{shock}} \boldsymbol{\nabla} w \cdot \left(\mathbf{I} - \frac{\boldsymbol{a} \otimes \boldsymbol{a}}{\|\boldsymbol{a}\|^2}\right) \cdot \boldsymbol{\nabla} u \, d\Omega,$$

where $\mathbf{I}$ is the unit tensor.

Finally, it is important to note that both isotropic and anisotropic added dissipations are only active when $\|\boldsymbol{\nabla} u\| \neq 0$ and that numerical results are sensitive to the numerical tolerance ($\|\boldsymbol{\nabla} u\| \leq$ tolerance $\Leftrightarrow \|\boldsymbol{\nabla} u\| = 0$) implemented in the code.

4.6 NEARLY INCOMPRESSIBLE FLOWS

Most numerical methods for the analysis of compressible flow problems present numerical difficulties when applied to low-speed, nearly incompressible flows. This is clearly due to the fast propagation of pressure signals as flow conditions approach the incompressible limit. In the case of low-speed flow, the accurate representation of pressure transients with explicit finite element schemes designed for the compressible regime would require the use of extremely small time steps, clearly undermining the practical utility of such schemes. In more mathematical terms, pressure behavior changes from hyperbolic (wave propagation) to elliptic character when passing from the compressible to the incompressible regime: the mass-conservation equation and the equation of state are replaced by the incompressibility constraint requiring that the velocity field be divergence-free.

Different approaches have been pursued to develop computational strategies for approximating flows at all speeds, from highly compressible to nearly incompressible. Methods for all flow speeds require an implicit treatment of the mass-conservation equation and of the pressure terms in the momentum and energy equations, in order to introduce the required elliptic behavior into the governing equations. A fractional-step approach to the time integration of the conservation equations is usually adopted, as it allows the separation of the necessarily implicit pressure terms from the other terms in the conservation equations.

An interesting splitting-up scheme for 2D flows consisting of three distinct phases was originally developed in finite difference format by Hirt et al. (1974) and subsequently extended to 3D problems by Stein, Gentry and Hirt (1977). The key idea in deriving such a fractional-step approach has been to note that the convective terms in the conservation equations vanish in the Lagrangian formulation; that is, when the computational mesh moves with the fluid. It is this reduction that suggests to first solve the Lagrangian form of the equations in the first phase of the calculation cycle and then to add the convective contributions as a separate step in the last phase. The middle phase adds an implicit pressure calculation that permits solutions to be obtained at all flow speeds.

This strategy has subsequently been adapted to a finite element context by Donea, Giuliani and Halleux (1982), see also Donea (1983), and has been widely used in the simulation of transient dynamic problems involving fluid–structure interaction. An application of this splitting-up technique is described in Section 4.7.2.

More recently, the splitting-up approach to flows at all speeds has been further exploited in the finite element context. Particularly noteworthy is the characteristic-based split (CBS) algorithm introduced by Zienkiewicz and Codina (1995), see also Zienkiewicz et al. (1995), Codina, Vázquez and Zienkiewicz (1998) and Zienkiewicz and Taylor (2000b). A related modeling approach of nearly incompressible flows has been proposed by Sampaio and Moreira (2000).

The central feature of the above methods is the derivation of an evolution equation for the pressure field through the elimination of the density from the mass balance equation. The pressure, momentum and energy equations are then discretized in time by a splitting-up procedure which ensures an implicit treatment of the equation for the pressure field and of the pressure terms in the momentum and energy equations. In this way, nearly incompressible flow situations can be treated without numerical stability problems. A further advantage of the splitting-up procedure is the possibility to isolate the convective terms and treat them by means of algorithms, such as the Taylor–Galerkin (or characteristic Galerkin) methods, designed for pure convection problems.

Stabilized finite element algorithms of the GLS or SUPG type can also be adapted to deal with nearly incompressible flows. In the low Mach number limit, the classical choice for the stabilization matrix $\boldsymbol{\tau}$ given in Section 4.4.2.3 fails to provide adequate stabilization. This is essentially due to a mismatch which occurs for low Mach numbers between the magnitude of the fluxes in the original equations and the corresponding terms in the numerically added viscosity. An alternative definition of $\boldsymbol{\tau}$ has been proposed by Wong, Darmofal and Peraire (2001) which leads to a formulation which can handle very low Mach number flows accurately.

4.7 FLUID–STRUCTURE INTERACTION

Fluid–structure interaction phenomena are an important consideration in several engineering fields. This is manifestly the case in the design of automotive and aerospace structures, as well as in modeling the response of offshore structures, long-span bridges and high-rise buildings. Fluid–structure interaction also plays an important role in the safety assessment of power generation plants and many other industrial installations.

The general topic of fluid–structure interaction is indeed a particularly broad subject in that it simultaneously brings together all the aspects associated with both structural mechanics and fluid mechanics. Each of these two areas are complex by themselves; however, when considered together, the situation becomes even more complex. In fact, the interaction (or coupling) between the fluid and solid response can be viewed as a feedback loop of the type illustrated in Figure 4.10: the structure surface loading is not known a priori but depends on the interface pressures in the fluid; the fluid response is in turn a function of the structure's surface motion.

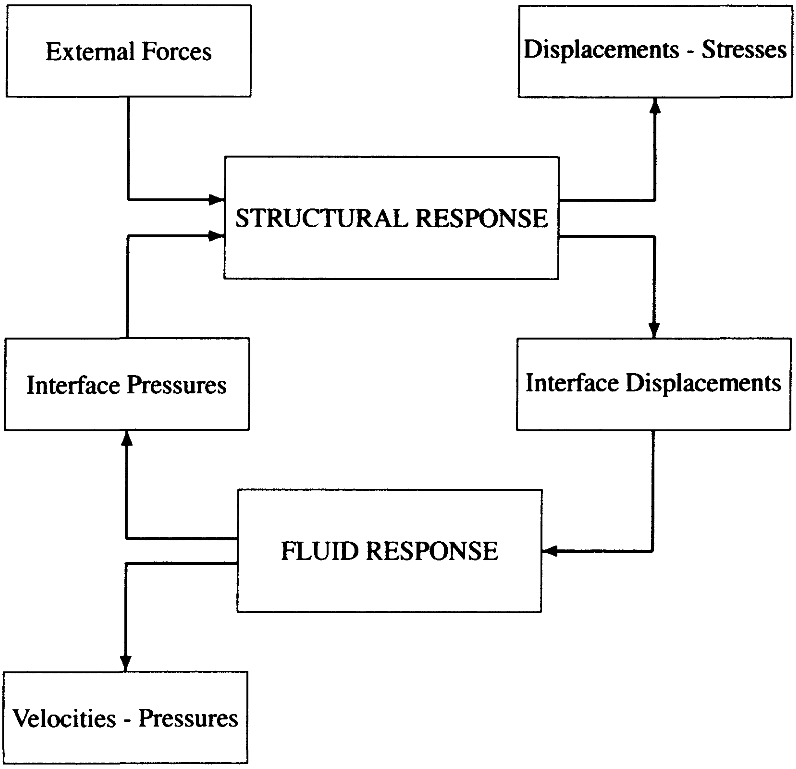


Fig. 4.10 Feedback loop in fluid–structure interaction.

In this section we restrict the fluid under consideration to be compressible and inviscid. There are indeed well-developed fluid–structure formulations for viscous fluids, in both the compressible and the incompressible regime. The reader interested in fluid–structure algorithms for viscous fluids should consult the relevant literature, such as, for instance, the original works of Liu and co-workers (Liu and Chang, 1985; Liu and Chang, 1986; Liu and Gvildys, 1986)

Here, we shall consider two classes of fluid–structure problems. The first is concerned with situations in which the displacements of the fluid are small, so that the so-called *acoustic approximation* for the fluid is valid. Modeling the response of a linear elastic structure interacting with an acoustic fluid is of particular interest in the framework of vibration studies of aeronautical structures containing an inviscid fluid. The second class of fluid–structure problems is concerned with situations involving a large-displacement response of the fluid–structure system. In such cases, the use of the Arbitrary Lagrangian–Eulerian (ALE) formulation introduced in Section 1.4 is recommended to model the nonlinear fluid domain and ease its coupling with the nonlinear structural domain usually treated in the Lagrangian description.

4.7.1 Acoustic approximation

When displacements are small, the acoustic approximation leads to simplified formulations of the fluid response, either in terms of fluid displacements or in terms of pressure. The displacement formulation of the fluid is attractive in that it is directly compatible with the standard formulation used for structures, thus facilitating the coupling between fluid and structural domains. Nevertheless, a formulation based upon the discretization of the fluid pressure is generally preferred. The reason is that only one dependent variable is involved in the pressure formulation, while the displacement formulation includes as many unknowns as the number of space dimensions. In this section subscript F will be used to refer to the fluid domain and subscript S to the structure.

The derivation of the pressure equation governing the fluid response in the acoustic approximation starts from the linearized form (convective terms neglected) of the conservation equation for momentum introduced in Section 1.4. When specialized to an inviscid compressible fluid, the linearized form of the momentum equation reads

$$\rho \frac{\partial^2 \boldsymbol{u}_F}{\partial t^2} = -\boldsymbol{\nabla} p, \tag{4.39}$$

where $\boldsymbol{u}_F$ is the fluid displacement vector and the pressure p is assumed to be given by the barotropic (independent of temperature) equation of state

$$p = -\kappa \boldsymbol{\nabla} \cdot \boldsymbol{u}_F, \tag{4.40}$$

where κ is the fluid bulk modulus. Note that pressure is positive in compression. Differentiating this equation twice with respect to time and using the linearized form (4.39) of the momentum equation yields the wave equation

$$\frac{1}{c^2} \frac{\partial^2 p}{\partial t^2} = \boldsymbol{\nabla}^2 p \tag{4.41}$$

for the pressure field, where $c = \sqrt{\kappa/\rho}$ is the speed of wave propagation.

The boundary conditions associated with the wave equation (4.41) usually consist of both Dirichlet and Neumann conditions. On portion Γ_D^F of the boundary of the fluid domain Ω_F the value of pressure is prescribed as $p = p_D$, where p_D is a given function. The remaining part of the boundary consist of a portion Γ_N^F where the Neumann condition $\partial p/\partial n = h_F$ is prescribed (h_F is given), and a portion Γ_I, the interface between the fluid and the structure, on which the fluid–structure interaction relations must be enforced.

The key point in coupling the fluid and structural domains lies in the imposition of the interface conditions. The relations on the fluid–solid interface Γ_I are:

- *Displacement/velocity compatibility.* An inviscid fluid is free to slip along the structural interface, but the displacement/velocity of the fluid along the normal to the interface must be equal to the displacement/velocity of the structure along the same direction, namely

$$\boldsymbol{n}_F \cdot \boldsymbol{u}_F = \boldsymbol{n}_F \cdot \boldsymbol{u}_S \quad \text{(continuity of normal displacements),} \tag{4.42a}$$

$$\boldsymbol{n}_F \cdot \boldsymbol{v}_F = \boldsymbol{n}_F \cdot \boldsymbol{v}_S \quad \text{(continuity of normal velocities).} \tag{4.42b}$$

This condition ensures that the fluid and structural domains will not detach or overlap during the motion. Both equations are equivalent and we use one or the other depending on the formulation employed (displacements or velocities).

- *Traction equilibrium.* That is, the stresses in the fluid must be equal to the stresses in the structure. For inviscid fluids this condition reduces to

$$\boldsymbol{n}_F\, p = \boldsymbol{n}_S \cdot \boldsymbol{\sigma} \qquad \text{(continuity of interface traction vector),} \qquad (4.42c)$$

where $\boldsymbol{\sigma}$ is the stress tensor on the structure.

Then, denoting by q the weighting function for the fluid, the weak form of the pressure formulation is given by

$$\left(q, \frac{1}{c^2}\ddot{p}\right)_{\Omega_F} + \left(\boldsymbol{\nabla} q, \boldsymbol{\nabla} p\right)_{\Omega_F} - \left(q, h_F\right)_{\Gamma_N^F} + \left(q, \rho_F(\boldsymbol{n}_F \cdot \ddot{\boldsymbol{u}}_S)\right)_{\Gamma_I} = 0. \qquad (4.43)$$

The last integral represents the coupling term with the structure and is obtained using

$$\frac{\partial p}{\partial n} = -\rho_F\,(\ddot{u}_n)_F = -\rho_F\,(\ddot{u}_n)_S$$

from the linearized momentum equation (4.39) and the continuity condition (4.42a).

The governing equation for the structural domain is the linearized momentum equation

$$\rho_S \frac{\partial^2 \boldsymbol{u}_S}{\partial t^2} = \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} + \boldsymbol{b}, \qquad (4.44)$$

where $\boldsymbol{u}_S$ is the structural displacement vector, $\boldsymbol{\sigma}$ the Cauchy stress and $\boldsymbol{b}$ the prescribed body force per unit volume. The boundary conditions consist of the Dirichlet condition $\boldsymbol{u}_S = \boldsymbol{u}_D$ on Γ_D^S, the Neumann condition $\boldsymbol{n} \cdot \boldsymbol{\sigma} = \mathbf{h}_S$ on Γ_N^S and relations (4.42) on the interface Γ_I with the fluid.

Denoting by $\boldsymbol{w}_S$ the vector of weighting functions for the momentum equation (4.44) and noting that the normal traction applied by the fluid on Γ_I is $p\,\boldsymbol{n}_F$, the weak form of the structural problem is

$$\begin{aligned}\left(\boldsymbol{w}_S, \rho_S\, \ddot{\boldsymbol{u}}_S\right)_{\Omega_S} + \left(\boldsymbol{\nabla}\boldsymbol{w}_S, \boldsymbol{\sigma}\right)_{\Omega_S} - \left(\boldsymbol{w}_S, \boldsymbol{b}\right)_{\Omega_S}& \\ - \left(\boldsymbol{w}_S, \mathbf{h}_S\right)_{\Gamma_N} - \left(\boldsymbol{w}_S, p\boldsymbol{n}_F\right)_{\Gamma_I} &= 0. \qquad (4.45)\end{aligned}$$

For a linear elastic structure interacting with an acoustic fluid, the system of semi-discrete equations resulting from the finite element spatial discretization of the weak forms (4.43) and (4.45) takes the following partitioned matrix form in the absence of damping, see Belytschko (1983) for details:

$$\begin{pmatrix} \mathbf{M}_S & 0 \\ -\rho_F \mathbf{R} & \mathbf{M}_F \end{pmatrix} \begin{pmatrix} \ddot{\mathbf{u}}_S \\ \ddot{\mathbf{p}} \end{pmatrix} + \begin{pmatrix} \mathbf{K}_S & \mathbf{R}^T \\ 0 & \mathbf{K}_F \end{pmatrix} \begin{pmatrix} \mathbf{u}_S \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_S^{\text{ext}} \\ \mathbf{f}_F^{\text{ext}} \end{pmatrix}. \qquad (4.46)$$

Unfortunately, the partitioned matrices in this system are not symmetric. This lack of symmetry represents a drawback in the case of implicit time integration of the equations and requires non-standard algorithms to extract eigenvalues for vibration analysis. This has motivated the development of symmetric formulations, such as the ones proposed by Moran and Ohayon (1979) and Kehr-Candille and Ohayon (1992), see also the references therein. In the framework of vibro-acoustic studies of aeronautical structures, such as liquid-propelled launch vehicles, they consider the response of an elastic structure containing an inviscid compressible fluid. The structure is again described by the displacement field $\boldsymbol{u}_S$, but the fluid is now described by two scalar fields, the pressure p and the displacement potential ϕ, such that $\boldsymbol{u}_F = \boldsymbol{\nabla}\phi$ and $\int_{\Omega_F} \phi = 0$. This choice of flow variables leads to symmetric formulations and allows the incorporation of a damping mechanism on the fluid–structure interface.

Remark 4.10 (Added mass concept). If the fluid can be considered incompressible, the wave equation (4.41) for the pressure field becomes the Laplace equation $\boldsymbol{\nabla}^2 p = 0$ and thus, there is no pressure-mass term in the second equation of system (4.46). The pressure is then obtained from the second equation as

$$\mathbf{p} = \mathbf{K}_F^{-1}\left(\mathbf{f}_F^{\text{ext}} + \rho_F\,\mathbf{R}\,\ddot{\mathbf{u}}_S\right)$$

and can be substituted in the first equation of system (4.46) to obtain the modified structural equation

$$\left(\mathbf{M}_S + \rho_F\,\mathbf{R}^T\,\mathbf{K}_F^{-1}\,\mathbf{R}\right)\ddot{\mathbf{u}}_S + \mathbf{K}_S\,\mathbf{u}_S = \mathbf{f}_S^{\text{ext}} - \mathbf{R}^T\,\mathbf{K}_F^{-1}\,\mathbf{f}_F^{\text{ext}}.$$

The need for a coupled solution has thus been eliminated and the effect of the incompressible fluid on the structure is described by a so-called *added mass* term that modifies the standard mass matrix $\mathbf{M}_S$. Note that, in contrast to the usual sparse and banded finite element matrices, the added mass matrix is full for the structural nodes in contact with the fluid.

4.7.2 Nonlinear transient dynamic problems

The second class of fluid–structure problems we wish to consider is concerned with situations involving a large-displacement response of the fluid, possibly accompanied by a nonlinear response of the structure. In such cases, the use of the ALE formulation introduced in Section 1.4 is recommended to model the fluid domain and ease its coupling with the structural domain usually treated in the Lagrangian description. We shall illustrate a particular problem in this class, namely the nonlinear transient dynamic response of fluid–structure systems. Fluid–structure interaction phenomena of this type are often encountered in the safety studies of industrial plants or components, where transient events such as explosions, impacts or crashes must be simulated. The particular algorithms discussed below are implemented in PLEXIS-3C, a finite element computer code for fast transient analyses (Bung et al., 1989).

4.7.2.1 Structural modeling A Lagrangian mesh in which material points and nodal points remain coincident throughout the calculation is usually adopted for the structural domain. If equilibrium is expressed in the current configuration in terms of the Cauchy stress, σ, the weak form of the momentum equation is identical to that in equation (4.45) for the linear theory. Then, the following set of differential equations in time results from its finite element spatial discretization:

$$\mathbf{M}_S \, \mathbf{a}_S = \mathbf{f}_{ext} - \mathbf{f}_{int}, \tag{4.47}$$

where $\mathbf{M}_S$ is the mass matrix of the structure, $\mathbf{a}_S$ the vector of nodal accelerations, $\mathbf{f}_{ext}$ is a nodal vector of externally applied loads and $\mathbf{f}_{int}$ the nodal vector of internal forces arising from the discretization of the stress-divergence term in the momentum equation. See for instance Belytschko (1983) or Belytschko, Liu and Moran (2000) for a more detailed account of the Lagrangian mesh/Cauchy stress formulation. In fast transient problems, the time integration of the differential system (4.47) is usually performed using the explicit central difference scheme

$$\mathbf{v}_S^{n+1} = \mathbf{v}_S^n + \frac{\Delta t}{2}\left(\mathbf{a}_S^n + \mathbf{a}_S^{n+1}\right) \quad \text{and} \quad \mathbf{u}_S^{n+1} = \mathbf{u}_S^n + \Delta t\left(\mathbf{v}_S^n + \frac{\Delta t}{2}\mathbf{a}_S^n\right),$$

where $\mathbf{v}_S$ is the vector of nodal velocities and $\mathbf{u}_S$ the vector of nodal displacements. The last equation gives the new mesh node displacements and thus the updated configuration of the structural domain. In this new configuration, the stresses $\sigma^{\,n+1}$ are then evaluated by application of constitutive models relating an objective rate of Cauchy stress to the strain rate (or stretching). The latter is evaluated from the half-step velocity $\mathbf{v}_S^{n+1/2} = \mathbf{v}_S^n + \frac{\Delta t}{2}\mathbf{a}_S^n$.

Note that the compatibility conditions (4.42) must be enforced along the interface between the structure and the fluid. Their implementation in the context of an ALE description of the fluid domain is discussed in Section 4.7.2.4.

4.7.2.2 *ALE algorithms for compressible fluids* The fluid is again assumed inviscid and compressible and its flow governed by the Euler equations. To model the hydrodynamic domain in the ALE description, a common practice is to use simple finite elements with linear or multilinear local approximations for the fluid velocity, $\boldsymbol{v}$, and the mesh velocity, $\hat{\boldsymbol{v}}$. Furthermore, the density, ρ, the specific internal energy, e, and hence the pressure, p, will be assumed elementwise constant.

According to the developments in Section 1.4, where the relative velocity $\boldsymbol{c} := \boldsymbol{v} - \hat{\boldsymbol{v}}$ between the material and the mesh has been introduced, the following integral form of the mass and internal energy equations is used for updating the element mass and internal energy:

$$\left.\frac{\partial M^e}{\partial t}\right|_{\chi} = \left.\frac{\partial}{\partial t}\right|_{\chi} \int_{\Omega^e} \rho \, d\Omega = -\int_{\Gamma^e} \rho\, \boldsymbol{c}\cdot\boldsymbol{n}\, d\Gamma,$$

$$\left.\frac{\partial I^e}{\partial t}\right|_{\chi} = \left.\frac{\partial}{\partial t}\right|_{\chi} \int_{\Omega^e} \rho\, e\, d\Omega = -\int_{\Omega^e} p\, \boldsymbol{\nabla}\cdot\boldsymbol{v}\, d\Omega - \int_{\Gamma^e} \rho\, e\, \boldsymbol{c}\cdot\boldsymbol{n}\, d\Gamma.$$

In addition, use is made of the rate of change of the element volume V^e given by

$$\left.\frac{\partial V^e}{\partial t}\right|_{\chi} = \int_{\Gamma^e} \hat{\boldsymbol{v}}\cdot\boldsymbol{n}\, d\Gamma$$

to update the element density and specific internal energy via

$$\rho^e = \frac{M^e}{V^e} \qquad \text{and} \qquad e^e = \frac{I^e}{M^e}.$$

On the other hand, the weak form of the ALE momentum equation (1.22) is given by

$$\big(\boldsymbol{w}_F, \rho \boldsymbol{v}_t\big) + \big(\boldsymbol{w}_F, \rho(\boldsymbol{c}\cdot\boldsymbol{\nabla})\boldsymbol{v}\big) = \big(\boldsymbol{\nabla}\boldsymbol{w}_F, p\mathbf{I}\big) + \big(\boldsymbol{w}_F, \rho\boldsymbol{b}\big) + \big(\boldsymbol{w}_F, \mathbf{t}_F\big)_{\Gamma_N},$$

where $\boldsymbol{w}_F$ represents the vector of weighting functions, $\mathbf{t}_F = p\boldsymbol{n}$ is the normal traction vector prescribed on portion Γ_N of the fluid domain boundary. The finite element discretization of the previous momentum equation results in the matrix system

$$\mathbf{M}_F \left.\frac{\partial \boldsymbol{v}}{\partial t}\right|_{\chi} = \mathbf{f}_t + \mathbf{f}_b + \mathbf{f}_p + \bar{\mathbf{f}}, \tag{4.48}$$

where $\mathbf{M}_F$ is the mass matrix, $\mathbf{f}_t$ represents the nodal loads induced by the transport of momentum components, $\mathbf{f}_b$ denotes the nodal loads due to the applied body forces $\rho\boldsymbol{b}$, $\mathbf{f}_p$ represents the nodal loads due to the fluid pressure p, and $\bar{\mathbf{f}}$ accounts for the prescribed boundary pressure. This last vector includes the reaction induced by prescribed displacements and velocities, as well as the compatibility conditions at the fluid–structure interface

Because of the presence of transport terms in the above ALE equations for mass, momentum and internal energy, special treatment is suggested for time integration of the equations in the fluid domain. As already mentioned in Section 4.6, a time integration scheme valid for flows at all speeds can be derived by adapting the fractional-step method introduced in the finite difference context by Hirt et al. (1974) and Stein et al. (1977). The method is organized in three steps. Since the transport terms in the semi-discrete conservation equations vanish in the Lagrangian formulation, when $\boldsymbol{c} = \boldsymbol{v} - \hat{\boldsymbol{v}} = 0$, the Lagrangian form of the equations is solved in the first step using an explicit time scheme. This provides intermediate values of the density, velocity and internal energy. Note that the element mass does not change in the Lagrangian phase. In the second step, which is also assumed Lagrangian, an implicit pressure calculation is introduced. This step eliminates the usual stability condition of explicit schemes that limits sound waves to travel no further than one element per time step. This allows solutions to be obtained for flows at all speeds, from highly compressible to nearly incompressible. The transport (convection) contributions are then added in the last step of the time integration procedure. The evaluation of the convective terms requires an updating of the mesh velocities $\hat{\boldsymbol{v}}$. An automatic algorithm, such as the one proposed by Giuliani (1982), should be employed for this purpose. Note that the transport phase is hyperbolic and must be stabilized. Donea, Giuliani and Halleux (1982) discuss additional details on the above fractional-step integration of the conservation equations.

4.7.2.3 Boundary conditions Having established the general computational framework for the treatment of the structural and fluid domains, we now consider how boundary conditions are enforced. As usual, they are subdivided into natural and essential conditions. Natural conditions are prescribed external loads, such as concentrated forces or distributed pressures. Their treatment does not pose any problem, as they fit as natural boundary conditions in the relevant variational formulation. The essential boundary conditions include prescribed displacements and velocities, as well as the compatibility conditions at the fluid–structure interface. Essential boundary conditions on the fluid domain are best implemented by means of the Lagrange multiplier method, see Remark 1.13.

Consider for instance that the essential conditions are imposed on the half-step value $\mathbf{v}^{n+1/2}$ of the velocity at the fluid–structure interface (thus, $\mathbf{f}_t$ is zero in this step). Such conditions can be expressed as a linear set of constraints of the form

$$\mathbf{A}\mathbf{v}^{n+1/2} = \mathbf{b}^{n+1/2}, \tag{4.49}$$

where $\mathbf{A}$ is a sparse (most of its coefficients are zero, in particular those related to degrees of freedom with no essential boundary condition) rectangular matrix (number of degrees of freedom for which essential boundary conditions are imposed times total number of degrees of freedom), $\mathbf{v}$ is the vector of nodal velocities and $\mathbf{b}$ a vector of (possibly time-dependent) prescribed values. The equilibrium equations for the half-step are written in the form

$$\mathbf{M}_F\left(\frac{\mathbf{v}^{n+1/2} - \mathbf{v}^{n-1/2}}{\Delta t}\right) = \mathbf{f}_b^n + \mathbf{f}_p^n - \mathbf{f}_{\text{reac}}^n, \tag{4.50}$$

where $\mathbf{M}_F$ is the mass matrix, $\mathbf{f}_b$ and $\mathbf{f}_p$ were already defined in (4.48), and finally, $\mathbf{f}_{\text{reac}}$ stands for the unknown reaction forces produced by the essential boundary conditions. We assume, for simplicity, that all the prescribed boundary pressure is due to Dirichlet boundary conditions, that is $\bar{\mathbf{f}}^n \equiv -\mathbf{f}_{\text{reac}}$. In order to introduce the constraints defined by (4.49) in the equilibrium equations (4.50), Lagrange multipliers are introduced. The unknown reactions are expressed as

$$\Delta t\, \mathbf{f}_{\text{reac}}^n = \mathbf{A}^T \boldsymbol{\lambda}^n, \tag{4.51}$$

where $\boldsymbol{\lambda}$ is the vector of Lagrange multipliers. Substituting this expression into the equilibrium equations (4.50) and rewriting (4.49) yields

$$\begin{pmatrix} \mathbf{M}_F & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{v}^{n+1/2} \\ \boldsymbol{\lambda}^n \end{pmatrix} = \begin{pmatrix} \mathbf{M}_F\, \mathbf{v}^{n-1/2} + \Delta t(\mathbf{f}_b^n + \mathbf{f}_p^n) \\ \mathbf{b}^{n+1/2} \end{pmatrix}.$$

Since the mass matrix is regular (in fact, it is symmetric and positive definite), we can compute its inverse and the previous system can be solved in two steps. The first equation can be written as

$$\mathbf{v}^{n+1/2} = \mathbf{v}^{n-1/2} + \mathbf{M}_F^{-1}\left(\Delta t(\mathbf{f}_b^n + \mathbf{f}_p^n) - \mathbf{A}^T\boldsymbol{\lambda}^n\right), \tag{4.52a}$$

and after substitution in the second one, we solve for $\boldsymbol{\lambda}^n$

$$\mathbf{A}\mathbf{M}_F^{-1}\mathbf{A}^T\boldsymbol{\lambda}^n = \mathbf{A}\mathbf{v}^{n-1/2} + \Delta t\mathbf{A}\mathbf{M}_F^{-1}(\mathbf{f}_b^n + \mathbf{f}_p^n) - \mathbf{b}^{n+1/2}. \tag{4.52b}$$

Note that $\mathbf{A}\mathbf{M}_F^{-1}\mathbf{A}^T$ is regular if and only if the rank of $\mathbf{A}$ is maximum, namely rank $\mathbf{A}$ is equal to the number of degrees of freedom for which essential boundary conditions are imposed. Once the Lagrange multipliers are known, $\mathbf{v}^{n+1/2}$ is determined from the first equation. Finally, if required the reaction forces $\mathbf{f}^n_{\text{reac}}$ are then obtained from expression (4.51).

This approach as stated has an important drawback (specially in the context of an explicit code): the inverse of the mass matrix is full! Although the system of equations in (4.52b) is not large (its size is only the number of degrees of freedom for which essential boundary conditions are imposed), the evaluation of the inverse matrix precludes the use of the algorithm as presented. Note however, that in transient dynamics it is standard to use the lumped (or diagonal) mass matrix, $\mathbf{M}^L_F$. If this matrix is used in (4.52) things change drastically: its inverse is diagonal (and trivial) and thus $\mathbf{A}[\mathbf{M}^L_F]^{-1}\mathbf{A}^T$ can be constructed easily. Moreover, the lumped-mass matrix uncouples each degree of freedom of $\mathbf{v}^{n+1/2}$ and consequently equations (4.52) can be particularized for the subset of degrees of freedom for which essential boundary conditions are imposed, that is each matrix in the product $\mathbf{A}[\mathbf{M}^L_F]^{-1}\mathbf{A}^T$ can be square. Under these circumstances the previous algorithm is efficient and can be used in the context of fast transient dynamics computations.

4.7.2.4 Fluid–structure coupling As already noticed for the linear acoustic approximation, Section 4.7.1, a key point in the fluid–structure coupling lies in the prescription of the interface conditions. These conditions are already described by equations (4.42).

To illustrate the ALE coupling procedure along fluid–solid interfaces, consider a structural member in permanent contact with an inviscid fluid which is allowed to slide along the structure. 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 imposing that the grid velocity $\hat{v}_F$ of the fluid nodes at the interface be equal to the material velocity 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 most elegant manner; that is, the imposition of equations (4.42) is simple because of the node alignment.

The equilibrium condition, see equation (4.42c), as in the linear acoustic regime, states that at each point of the interface between structure and inviscid fluid a contact pressure is transmitted. Moreover, the interaction pressure is directed along the normal to the interface. 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, that is to the interaction force $\mathbf{r}$.

The compatibility condition, stated in equation (4.42b), imposes continuity of the velocity components normal to the interface. Recall that the tangential velocities at the fluid and solid nodes on the interface are unconstrained. This condition can be expressed in the form (4.49). For instance, in 2D and for one point we have

$$\mathbf{A} = \left(n_x, n_y, -n_x, -n_y\right); \quad \mathbf{v} = \left(v_{Fx}, v_{Fy}, v_{Sx}, v_{Sy}\right)^T; \quad \mathbf{b} = (0).$$

The interaction force $\mathbf{r}$, see (4.51), at this point is expressed in terms of one Lagrange multiplier (only one condition is presented):

$$\Delta t \left(r_{Fx}, r_{Fy}, r_{Sx}, r_{Sy}\right) = \left(n_x, n_y, -n_x, -n_y\right) \lambda.$$

This is repeated for each interaction node to construct the vectors and matrices in (4.49) and (4.51). This allows us to solve (4.52b) and from (4.51) obtain the interaction force at each node on the ALE interface. In this manner, the fluid and structural responses can be solved separately, though achieving a tight coupling between the two subdomains.

Remark 4.11 (Normal to a discrete interface). In practice, especially in complex 3D 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 2D 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 4.12. The above treatment of the coupling problem only applies to those portions of the structure which 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 standard Lagrangian methods.

4.7.3 Illustrative examples

Two numerical simulations are presented that illustrate ALE finite elements in transient dynamic fluid–structure interaction as outlined in the previous sections. Calculations were performed with PLEXIS-3C finite element code (Bung et al., 1989).

4.7.3.1 Flexible vessel experiment A schematic of the configuration is depicted in Figure 4.11. A thin cylindrical vessel with a hemispherical bottom is nearly completely filled with liquid and impulsively loaded by the detonation of an explosive charge located on the vessel axis. The vessel contains a flexible inner shield hinged at its base. The top of the vessel is clamped to a rigid cover. Due to symmetry, only half of the configuration is actually modeled. First, a purely Lagrangian solution was attempted using the finite element mesh shown in Figure 4.11. Conical shell elements are used to model the structural parts, while triangular and quadrilateral fluid elements

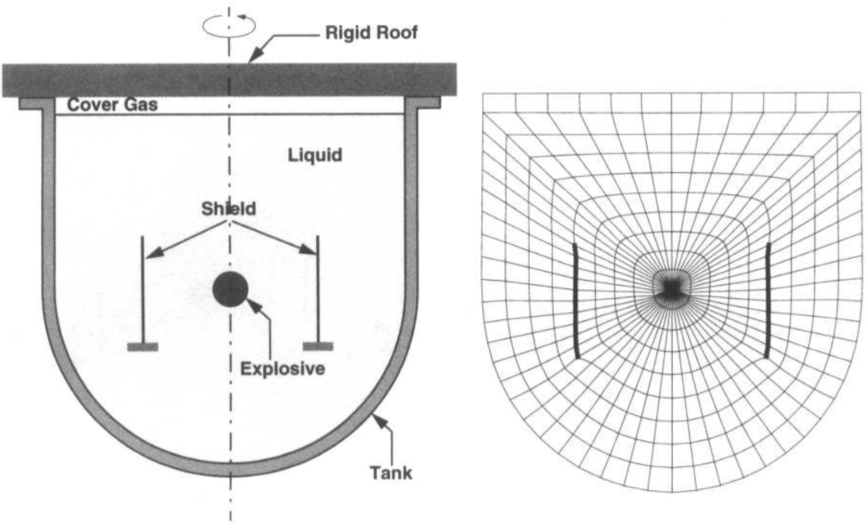


Fig. 4.11 Model of containment vessel and element mesh for Lagrangian and multiphase, multicomponent ALE solutions.

are employed to model the explosive charge, the liquid and the cover gas at the top of the vessel. The fluid–structure interfaces are not properly treated in this Lagrangian example because fluid and structural nodes coincide, causing the fluid to stick to the structural walls. As can be observed from Figure 4.12, large distortions of the fluid mesh do occur and due to mesh entanglement the calculations failed after 95 ms.

The experiment was then repeated using the ALE description in the hydrodynamic domain and treating fluid–structure interaction at the ALE interfaces as explained in the previous sections. The fluid is now able to slide along the structural walls and, as shown in Figure 4.12, the automatic mesh rezoning algorithm succeeds in keeping the hydrodynamic mesh reasonably regular.

Another experiment was then performed in which the ALE algorithm was used together with a heterogeneous (multiphase, multicomponent) fluid formulation. Hence, there are no Lagrangian fluid–fluid interfaces in this calculation. The finite element mesh is depicted in Figure 4.13. As can be appreciated from this same figure, the mesh distortions are now minimal. The position of physical fluid–fluid interfaces may be estimated from Figure 4.13 where the mass fractions of the various components are plotted. The present results are found to be in substantial agreement with those reported in Figure 4.12 obtained with the ALE multi-fluid formulation.

4.7.3.2 An industrial application As an example of an industrial application of the ALE methodology, we consider the simulation of an explosion in a power transformer cell which is part of an underground power plant, see Figure 4.14a. This problem is discussed in detail by Casadei et al. (2001) and the results are reproduced

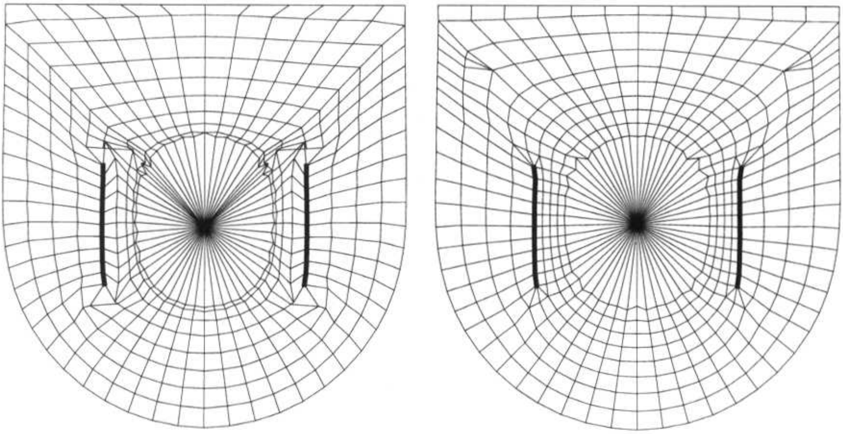


Fig. 4.12 Purely Lagrangian solution of containment vessel (left) and ALE solution using Lagrangian fluid–fluid interfaces (right).

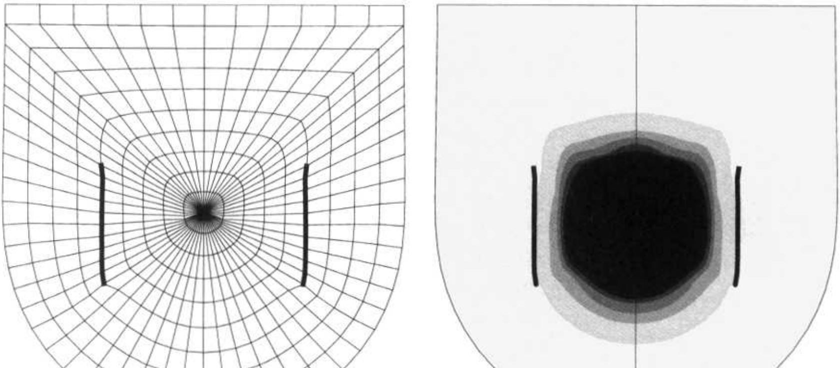


Fig. 4.13 ALE solution of containment vessel using multicomponent fluid formulation: final mesh configuration (left) and mass fractions of the various components (right).

here with their kind permission. If an electrical fault occurs inside the oil-insulated equipment, an arc is likely to occur causing pyrolysis of the mineral oil. As a result, a large quantity of gaseous hydrocarbons is produced containing over 70% hydrogen which may ultimately break the transformer tank and propagate inside the cell.

Under critical circumstances, the air–hydrocarbon mixture may ignite giving rise to a strong blast. The aim of the study is to assess the structural response in case of explosion of the aluminum tubes (1 m diameter) that contain and insulate by means of SF_6 gas the high-voltage bars running out from the power transformer. Suitable assumptions allow the determination of the initial conditions of a high-pressure and temperature bubble that simulates the energy associated with the explosion. The explosive bubble is surrounded by air at normal conditions, see Figure 4.14b. An elasto-plastic material describes the behavior of the aluminum tubes represented in Figure 4.14c.

Figure 4.14d shows the structural portion of the computational domain. The aluminum tubes are discretized by means of shell elements, while the transformer and cell walls are assumed rigid and are thus not discretized. Figure 4.14e shows a portion of the fluid domain discretized by means of tetrahedral elements and the embedded shell structural elements for the aluminum tubes.

Figures 4.14f and 4.14g show the computed mass fractions of air at half the time and at the end of the transient. A pressure plot at half the time of the transient is shown in Figure 4.14h, while Figure 4.14i illustrates the deformed shape of the aluminum tubes at 50 ms. Deformation is amplified 10 times.

As shown by the above example, ALE finite element computer codes, such as PLEXIS-3C, represent very useful modeling tools for the simulation of postulated accident situations in industrial plants.

4.8 SOLVED EXERCISES

4.8.1 One-step Taylor–Galerkin solution of Burgers' equation

The inviscid Burgers' equation (4.2) is solved over the 1D domain $]0, L[$. The inlet boundary condition $u(0,t) = 1$ is prescribed and the following initial data are used:

$$u(x,0) = \begin{cases} 1 & 0 \leq x \leq 0.64, \\ 1 - (x - 0.64)/(0.20) & 0.64 \leq x \leq 0.84, \\ 0 & 0.84 \leq x \leq 1.0. \end{cases}$$

This skew initial profile straightens as time goes on until a discontinuity is formed which propagates at the dimensionless speed of 0.5. Time integration is performed by means of the one-step second-order scheme

$$\frac{u^{n+1} - u^n}{\Delta t} = u_t^n + \frac{\Delta t}{2} u_{tt}^n,$$

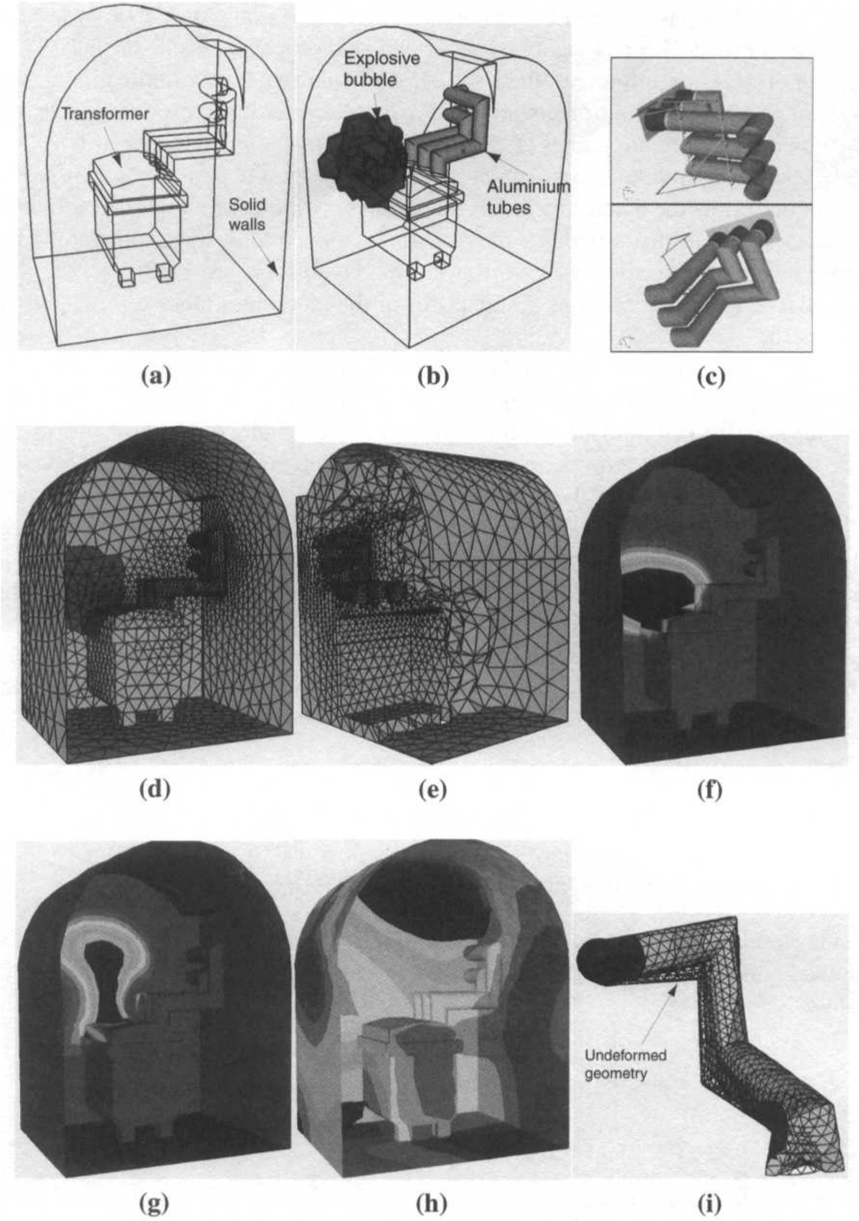


Fig. 4.14 Simulation of an explosion in a power transformer cell (Courtesy of ENEL Research Hydraulics and Structures, Milano).

where in view of the differential equation, see (4.2),

$$u_t = -f_x, \quad \text{and}\, u_{tt} = -f_{xt} = -f_{tx} = -\left(f_u\, u_t\right)_x = \left(u^2\, u_x\right)_x.$$

The time discretized version of Burgers' equation therefore reads

$$\frac{u^{n+1} - u^n}{\Delta t} = -f_x^n + \frac{\Delta t}{2}\left((u^n)^2\, u_x^n\right)_x.$$

After integration by parts of both spatial terms, the second-order Taylor–Galerkin formulation is given as follows:

$$\int_0^L w\, \frac{u^{n+1} - u^n}{\Delta t}\, dx = \int_0^L w_x\, f^n\, dx - \frac{\Delta t}{2} \int_0^L w_x\, (u^n)^2\, u_x^n\, dx$$
$$- \left[w\left(f^n - \frac{\Delta t}{2} (u^n)^2\, u_x^n\right)\right]_{x=0}^{x=L}.$$

The boundary term can be rewritten using $f_t = f_u\, u_t = -u^2\, u_x$, as

$$-\left[w\left(f^n - \frac{\Delta t}{2} (u^n)^2\, u_x^n\right)\right]_{x=0}^{x=L} = -\left[w\left(f^n + \frac{\Delta t}{2}\, f_t^n\right)\right]_{x=0}^{x=L},$$

which provides a natural condition for prescribed boundary flux. We therefore replace the inlet boundary condition $u(0,t) = 1$ by the condition $f(0,t) = \left(\frac{1}{2}u^2\right)_{x=0} = 0.5$.

The problem unknown u is linearly interpolated over an element and three different representations of the flux f in the convective term are tested:

1. f is elementwise constant and determined from the mean value of u in the considered element (*constant representation*);

2. f is determined from the value of u at the two element Gauss points and a two-point Gaussian quadrature is used to evaluate the convective term (*classical representation*);

3. f is linearly interpolated using its evaluation at the element nodes (*group representation*).

The two-point quadrature rule is also used on the diffusion term.

The scope of the exercise is essentially to compare the above three representations of the flux. The results obtained at dimensionless times $t = 0.16$ and $t = 0.30$ using a mesh of 50 uniform linear elements and a Courant number $C = u_{max}\Delta t/h = 0.5$ are shown in Figure 4.15. As long as the initial profile straightens, the solutions obtained with the three flux representations are quite similar. Nevertheless, one may already note a more pronounced dissipative effect when the linear interpolation of the flux is employed. The differences are definitely more marked when the discontinuity propagates, the group representation of the flux giving the best solution.

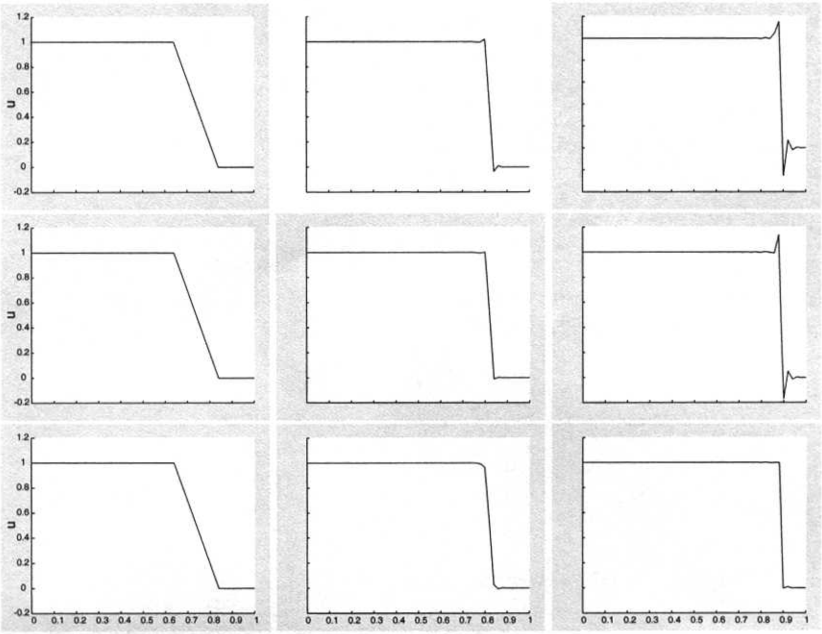


Fig. 4.15 Simulation by the one-step second-order Taylor–Galerkin method of the formation and propagation of a discontinuity. Comparison of three local representations of the flux in Burgers' equation: constant flux interpolation (top); classical interpolation (middle); linear flux interpolation (bottom). Solution times are from left to right: $t = 0$, $t = 0.16$ and $t = 0.30$.

4.8.2 The shock tube problem

We now solve the classical shock tube problem proposed by Sod (1978) for which there exists an exact solution to the 1D Euler equations. The problem involves a shock wave, a contact discontinuity and an expansion fan. A contact discontinuity is an interface between two fluid regions of different densities, but equal pressure. The shock tube problem is thus representative of the numerical difficulties encountered in the solution of the Euler equations of gas dynamics.

On the spatial domain $]0, 1[$, we solve the 1D system of Euler equations, see (4.16a)

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0,$$

where $\mathbf{U}$, $\mathbf{F}$ and the Jacobian matrix $\mathbf{A}$ are described in Remark 4.5, the eigenvalues of $\mathbf{A}$ are explicitly determined in (4.24), and they define the three characteristics described by (4.25) and plotted in Figure 4.7. Under the assumption of a perfect gas, the pressure is given by (4.17a), namely

$$p = (\gamma - 1)\rho(E - \frac{v^2}{2}).$$

and we take $\gamma = 1.4$. The following initial data are used:

$0 \leq x \leq 1/2$	$1/2 < x \leq 1$
$\rho = 1.0$	$\rho = 0.125$
$\rho v = 0.0$	$\rho v = 0.0$
$\rho E = 2.5$	$\rho E = 0.25$

The initial density and pressure difference is maintained by a diaphragm which is ruptured at $t = 0$. The objective is to compute the evolution of the conservation variables as a shock and a contact discontinuity propagate along the tube.

One-step Taylor–Galerkin. Time integration of the Euler equations is performed by means of the one-step second-order Taylor–Galerkin scheme

$$\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} = \mathbf{U}_t^n + \frac{\Delta t}{2} \mathbf{U}_{tt}^n.$$

The associated variational form is given as follows after integration by parts of the spatial terms:

$$\int_0^L \mathbf{W} \cdot \frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t}\, dx = \int_0^L \mathbf{W}_x \cdot \mathbf{F}^n \, dx$$
$$- \frac{\Delta t}{2} \int_0^L \mathbf{W}_x \cdot (\mathbf{A}^n)^2 \, \mathbf{U}_x^n \, dx - \left[\mathbf{W} \cdot \left(\mathbf{F}^n + \frac{\Delta t}{2} \mathbf{F}_t^n \right) \right]_{x=0}^{x=L},$$

where, as in the previous example, the differential equation, $\mathbf{U}_t = -\mathbf{F}_x$, is used to determine

$$\mathbf{U}_{tt} = -\mathbf{F}_{xt} = -\mathbf{F}_{tx} = -\big(\mathbf{A}\,\mathbf{U}_t\big)_x = \big(\mathbf{A}\,\mathbf{F}_x\big)_x = \big(\mathbf{A}^2\,\mathbf{U}_x\big)_x.$$

Note that the boundary term allows us to introduce prescribed flux components. In the present example, the definitions of $\mathbf{U}$, $\mathbf{F}$, see Remark 4.5, indicate that the boundary value of the density and energy flux is zero, while the boundary value of the momentum flux is given by the left and right pressures associated with the initial data.

The problem was solved using a mesh of 100 uniform linear elements. Time integration has been performed until $t = 0.2$ using a fixed time step $\Delta t = 1.5 \times 10^{-3}$. This corresponds to a Courant number

$$C = (v_{\max} + c)\frac{\Delta t}{h} = 0.45.$$

The convective and diffusive terms were integrated using a two-point Gaussian quadrature. A linear approximation of the flux was used in compression regions ($\partial v / \partial x < 0$), and the classical flux representation otherwise. No artificial viscosity was introduced into the Galerkin formulation.

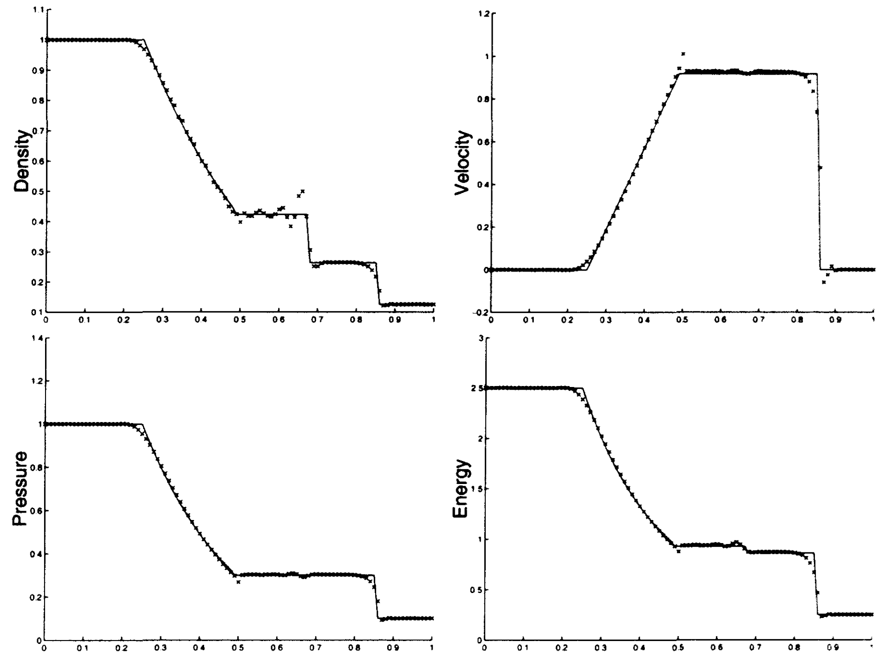


Fig. 4.16 Sod's shock tube: one-step Taylor–Galerkin without artificial viscosity.

Figure 4.16 shows a comparison with the exact solution. One notes that the position of the flow discontinuities is well predicted. However, in the absence of shock-capturing terms, oscillations are generated around sharp solution gradients. They are reduced in the next example by the selective addition of artificial viscosity.

Stabilization of the one-step Taylor–Galerkin solution. We repeat the solution of Sod's shock tube problem with the selective addition of artificial viscosity, as explained in Section 4.5.3.2. The idea is to first compute the solution to the Euler equations using the one-step second-order method as in the previous test case. Then, the second-order scheme is locally reduced to first order as indicated in equation (4.32).

The local modulation of the added viscosity is performed according to expressions (4.34) using the artificial viscosity method in equations (4.35) with parameter $\chi = 3.0$. The results obtained at time $t = 0.2$ are displayed in Figure 4.17 in comparison with the exact solution. The effect of the shock-capturing terms is to smooth out the flow discontinuities and thereby suppress the oscillations near sharp gradients typical of the second-order method.

Two-step Taylor–Galerkin. This method is described in Section 4.2.3.2. In the first step of the time integration procedure, we compute the half-step value $\mathbf{U}^{n+1/2}$

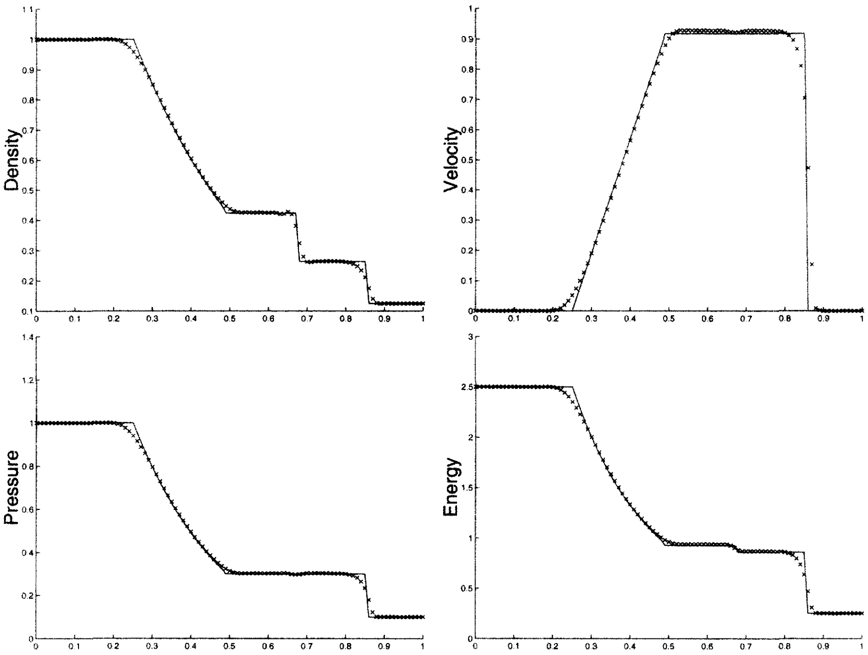


Fig. 4.17 Sod's shock tube: one-step Taylor–Galerkin with artificial viscosity.

of the flow variables and associated flux components at the center of the elements. Once the flux vector $\mathbf{F}^{n+1/2}$ is obtained, the end-of-step values are computed solving the variational equation

$$\int_0^L \mathbf{W} \cdot \frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t}\, dx = \int_0^L \mathbf{W}_x \cdot \mathbf{F}^{n+1/2}\, dx - \left[\mathbf{W} \cdot \mathbf{F}^{n+1/2}\right]_{x=0}^{x=L},$$

where the flux term has again been integrated by parts to produce a natural condition of prescribed boundary flux.

Once a solution is obtained with the second-order method in time, it is smoothed, locally, as in the previous test to reduce the non-physical oscillations near the discontinuities. The results obtained at time $t = 0.2$ with a time-step size $\Delta t = 0.002$ are displayed in Figure 4.18 in comparison with the exact solution. One notes that the simple two-step Taylor–Galerkin method gives results of the same quality as the one-step method.

Flux vector splitting. In order to produce an upwind-type spatial discretization of the 1D Euler equations we split the flux vector in the conservation equation

$$\mathbf{U}_t + \mathbf{F}_x = 0$$

into the form

$$\mathbf{U}_t + \mathbf{F}_x^+ + \mathbf{F}_x^- = 0$$

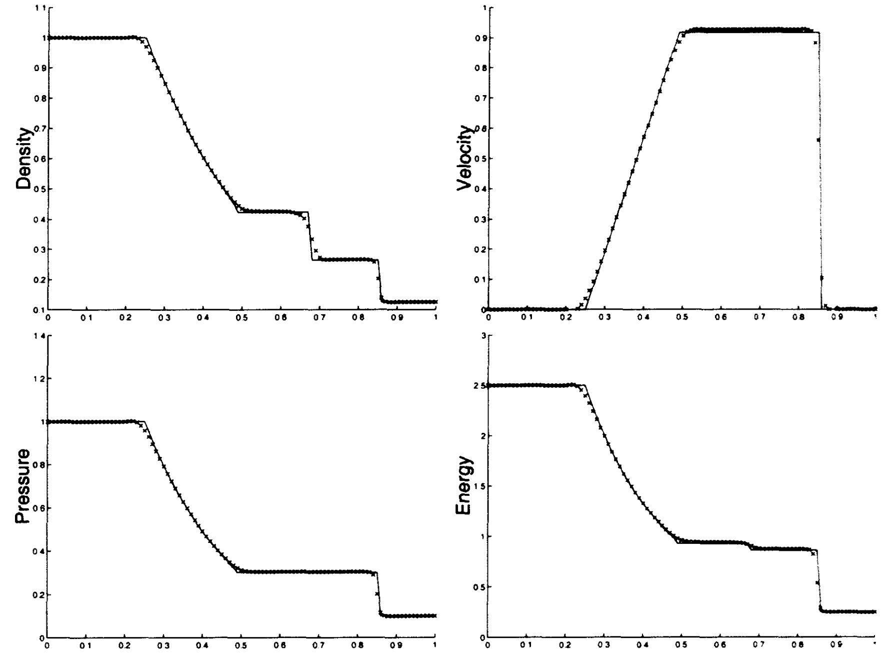


Fig. 4.18 Sod's shock tube: two-step Taylor–Galerkin with artificial viscosity.

as explained in Section 4.4.2.1. Moreover, we generalize the standard first-order upwind scheme as suggested by Steger and Warming (1981):

$$\frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} = -\frac{(\mathbf{F}_i^+ - \mathbf{F}_{i-1}^+)^n}{h} - \frac{(\mathbf{F}_{i+1}^- - \mathbf{F}_i^-)^n}{h}.$$

Recall that $\mathbf{F}^+$ is always positive (or zero), while $\mathbf{F}^-$ is always negative (or zero).

To reproduce the flux vector splitting technique in the finite element context, we make use of the construction of upwind finite element schemes discussed in Chapter 2 and consider the weak form

$$\int_0^L \mathbf{W} \cdot \mathbf{U}_t \, dx = -\int_0^L \left(\mathbf{W} + \frac{h}{2}\mathbf{W}_x\right) \cdot \mathbf{F}_x^+ \, dx - \int_0^L \left(\mathbf{W} - \frac{h}{2}\mathbf{W}_x\right) \cdot \mathbf{F}_x^- \, dx,$$

or, after integration by parts of the convective terms

$$\int_0^L \mathbf{W} \cdot \mathbf{U}_t \, dx = \int_0^L \mathbf{W}_x \cdot \left(\mathbf{F}^+ - \frac{h}{2}\mathbf{F}_x^+\right) dx$$
$$+ \int_0^L \mathbf{W}_x \cdot \left(\mathbf{F}^- + \frac{h}{2}\mathbf{F}_x^-\right) dx - \Big[\mathbf{W} \cdot (\mathbf{F} \cdot \boldsymbol{n})\Big]_{x=0}^{x=L}.$$

We use a uniform mesh of 100 elements with piecewise linear interpolation of the conservation variables and associated flux components. Figure 4.19 shows the results.

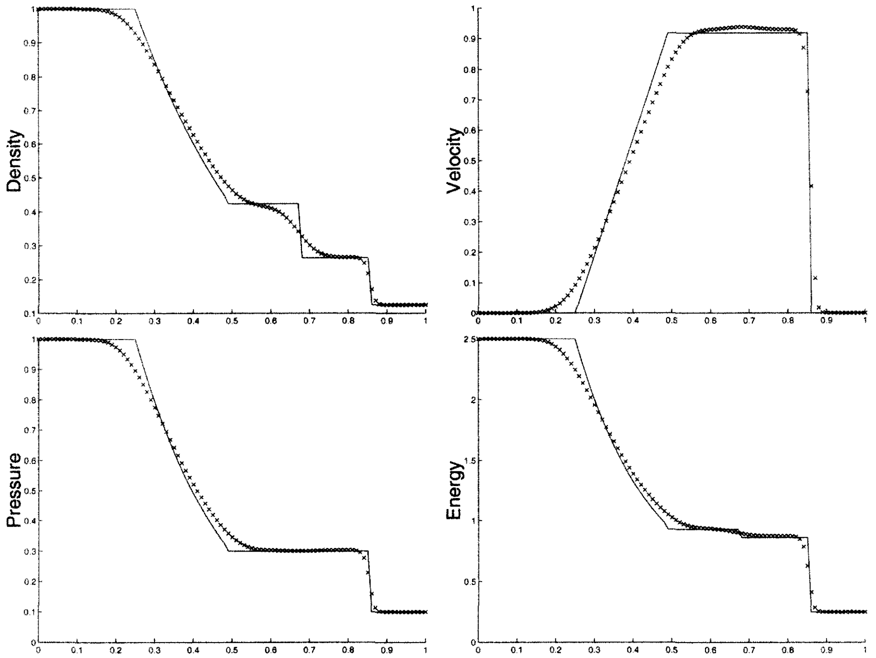


Fig. 4.19 Sod's shock tube: first-order in time and space flux vector splitting technique. Note the excessive diffusion introduced by standard monotone methods.

A time step $\Delta t = 0.004$, corresponding to a Courant number of 0.8, was employed with the Euler explicit method. The following flux vector splitting introduced by Steger and Warming (1981) was used (see also Hirsch, 1990, Chap. 20):

$$\mathbf{F}^{+} = \frac{1}{2\gamma}\begin{pmatrix} (2\gamma-1)\rho v + \rho c \\ 2(\gamma-1)\rho v^2 + \rho(v+c)^2 \\ (\gamma-1)\rho v^3 + \frac{1}{2}\rho(v+c)^3 + \frac{1}{2}\frac{3-\gamma}{\gamma-1}\rho c^2(v+c) \end{pmatrix}, \tag{4.53a}$$

and

$$\mathbf{F}^{-} = \frac{1}{2\gamma}\begin{pmatrix} \rho v - \rho c \\ \rho(v-\rho c)^2 \\ \frac{1}{2}\rho(v-c)^3 + \frac{1}{2}\frac{3-\gamma}{\gamma-1}\rho c^2(v-c) \end{pmatrix}. \tag{4.53b}$$

Here, $c = \sqrt{\gamma p/\rho}$ is the speed of sound, and p the pressure defined by the equation of state of a perfect gas.

At this point, it is important to recall the definitions of $\mathbf{F}_n^+$ and $\mathbf{F}_n^-$ adopted in Section 4.4.2.1, see Figure 4.9. It follows from such definitions that the splitting in (4.53) directly applies in the present example only at element ends where the unit outward normal has components $n = (1, 0)$. At element ends with outward normal

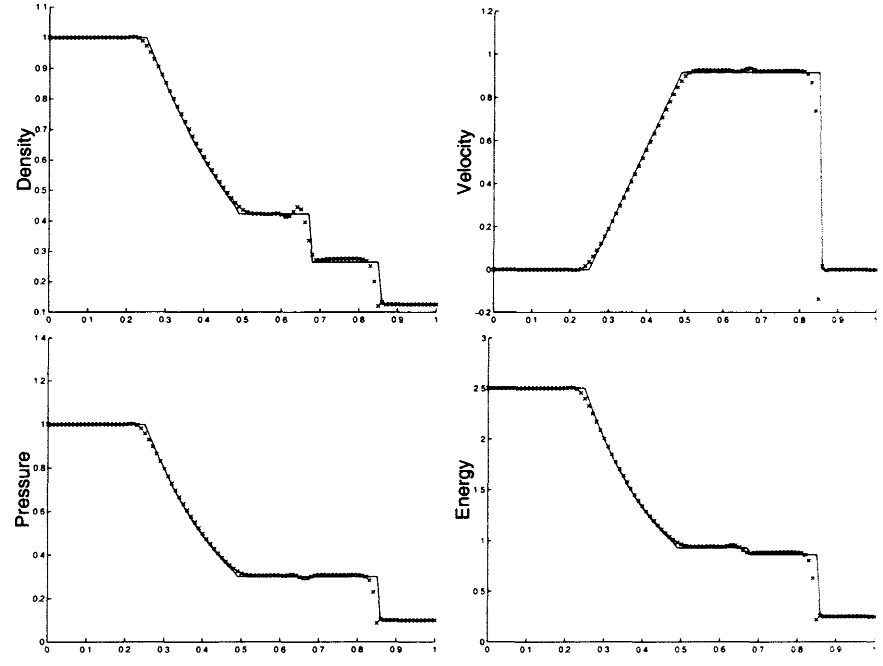


Fig. 4.20 Sod's shock tube: discontinuous Galerkin with artificial diffusion and forward Euler for time integration.

with components $n = (-1, 0)$, the positive flux component obtained from (4.53a) is actually $\mathbf{F}_n^-$ and the negative component is $\mathbf{F}_n^+$.

As shown by the results in Figure 4.19, the use of a first-order flux vector splitting technique gives stable, but overly diffusive, results, especially as regards the representation of the contact discontinuity. Flux-limiter techniques in Section 4.5.3.1 should be applied to concentrate the added numerical diffusion around sharp gradients only.

Discontinuous Galerkin. Here, the shock tube problem is solved using the discontinuous Galerkin method described in Section 4.4.2.2. The flux vector $\mathbf{F}_n(\mathbf{U})$ is split into inflow and outflow components using Steger and Warming's (1981) flux splitting, see equations (4.53). Figure 4.20 shows the results. Conservative variables and the associated flux components are linearly interpolated over an element. The Euler first-order method was used for time integration. Artificial viscosity was added to the momentum and energy equations, but only on elements close to sharp gradients. The coefficient of artificial diffusion was taken as $\nu^e = h^e\lambda^e/16$, where h^e is the size of the element and $\lambda^e = \lambda^e(\mathbf{U}^e)$ is the maximum characteristic speed, taken as $c + v$, where $c = \sqrt{\gamma p/\rho}$ is the speed of sound and v the modulus of the velocity vector.

5

Unsteady convection–diffusion problems

After studying elliptic problems in Chapter 2 and hyperbolic problems in Chapters 3 and 4, this chapter is concerned with parabolic problems. It combines the spatial discretization techniques for steady convection–diffusion problems discussed in Chapter 2 and the numerical schemes for time integration presented in Chapter 3. The objective is to produce time-accurate finite element methods for unsteady problems describing transport by convection and diffusion. Only linear and scalar problems are considered. Emphasis is placed on the accuracy properties of the Galerkin and spatially stabilized formulations.

5.1 INTRODUCTION

Time-accurate numerical methods for solving unsteady convection–diffusion problems using finite elements are studied here. Problems in this class are parabolic. In contrast to the pure convection problems discussed in Chapters 3 and 4, there are no discontinuous solutions in the presence of physical diffusivity and boundary conditions must be imposed everywhere on the boundary of the domain. However, Dirichlet boundary conditions may produce internal and boundary layers with steep solution gradients. Thus, on one hand, the stability problems of the Galerkin formulation studied in Chapter 2 are also present here. And, on the other hand, in convection-dominated problems, the need stressed in Chapter 3 to use time-stepping algorithms capable of simulating the role of characteristics is also present here. However, because of the presence in convection–diffusion of a second-order spatial operator, the

Finite Element Methods for Flow Problems. Jean Donea and Antonio Huerta
Copyright © 2003 John Wiley & Sons, Ltd.
ISBN: 0-471-49666-9

finite element techniques discussed so far for achieving spatial stabilization and time accuracy need to be suitably adapted to deal with parabolic transport problems.

The chapter begins with the presentation in Section 5.2 of the initial boundary value problem for convection–diffusion–reaction. This is followed in Section 5.3 by the description of time-stepping algorithms specifically adapted to trace the transient solution of mixed transport problems. Various classes of methods are considered. After a brief review of classical algorithms, such as the θ family of methods, we discuss operator-splitting methods separating convection and diffusion operators, as well as high-order accurate multistep methods. This includes time-stepping techniques based on Runge–Kutta methods and multistage schemes based on the factorization of Pad é approximations of the exponential function.

Spatial discretization procedures for unsteady convection–diffusion problems are then introduced in Section 5.4. The classical Galerkin formulation is presented first and its lack of sufficient stability in convection-dominated cases is underlined. The stabilization procedures for steady problems discussed in Chapter 2, such as SUPG, GLS and SGS, are then extended to the transient case. Moreover, pure least-squares formulations are introduced along the lines discussed in Chapter 3 for convective transport problems. The accuracy properties of fully discrete schemes are also illustrated. To conclude the discussion of spatial discretization procedures, we briefly consider in Section 5.5 the extension of stabilized formulations to the space–time domain. Finally, Section 5.6 presents solved numerical examples.

5.2 PROBLEM STATEMENT

The strong form of the convection–diffusion–reaction initial/boundary value problem is stated as follows: given the velocity field $\boldsymbol{a}(\boldsymbol{x}, t)$, the diffusion coefficient $\nu(\boldsymbol{x}, t)$, the reaction $\sigma(\boldsymbol{x}, t)$, the source term $s(\boldsymbol{x}, t)$, and the necessary initial and boundary conditions, find $u(\boldsymbol{x}, t)$ such that

$$\begin{aligned}
&u_t + \boldsymbol{a} \cdot \boldsymbol{\nabla} u - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} u) + \sigma u = s && \text{in } \Omega \times]0, T[, && (5.1a)\\
&u(\boldsymbol{x}, 0) = u_0(\boldsymbol{x}) && \text{on } \Omega, && (5.1b)\\
&u = u_D && \text{on } \Gamma_D, && (5.1c)\\
&\nu(\boldsymbol{n} \cdot \boldsymbol{\nabla}) u = h && \text{on } \Gamma_N. && (5.1d)
\end{aligned}$$

The numerical solution of convection–diffusion–reaction problems clearly involves a double discretization process, that is space discretization and time discretization. The former will be performed by the finite element method and will be discussed in Section 5.4. Various classes of methods can be employed to trace the temporal evolution of the solution of convection–diffusion–reaction problems. They are presented in the necessary detail in the next section.

Remark 5.1 (Boundary conditions). Equation (5.1a) is parabolic. Thus boundary conditions are imposed on Γ which is the smooth boundary of $\Omega \subset \mathbb{R}^{n_{sd}}$. Moreover, the boundary is assumed to consist of a portion Γ_D on which the value of u is prescribed, see (5.1c), and of a complementary portion Γ_N on which the diffusive flux is prescribed, see (5.1d), where h is given. Conditions on Γ_D are Dirichlet (or essential) conditions, while conditions on Γ_N are known as Neumann (or natural) conditions. Note that this last condition could be generalized in the form

$$\nu(\boldsymbol{n}\cdot\boldsymbol{\nabla})u = \varrho(\boldsymbol{n}\cdot\boldsymbol{a})(u-u_e)+\eta \qquad \text{on } \Gamma_N$$

where u_e is also given. This condition is either Neumann or Robin depending on the given parameters ϱ and η (they may vary with $\boldsymbol{x}$ and time). For simplicity, we shall assume that the Neumann condition (5.1d) is prescribed on Γ_N.

Remark 5.2 (Existence and uniqueness). Quarteroni and Valli (1994, Chap. 12) and Morton (1996, Thm 2.5.1) present the exact conditions for existence and uniqueness of the solution. In convection-dominated problems, it is sufficient for existence and uniqueness to impose that $\nu \geq \nu_0 > 0$ and $0 < \mu_0 \leq \sigma - \frac{1}{2}\boldsymbol{\nabla}\cdot\boldsymbol{a} \leq \mu_1$ in Ω. With these conditions, the bilinear form associated with the spatial operator (in a Galerkin formulation) is continuous and coercive, with a coercivity constant $\alpha = \min\{\nu_0, (\nu_0 + C_\Omega\mu_0)/(1+C_\Omega)\}$, where C_Ω is the Poincaré inequality constant ($\int_\Omega v^2 d\Omega \leq C_\Omega \int_\Omega |\boldsymbol{\nabla} v|^2 d\Omega$). That is, the Lax–Milgram lemma, see Section 1.5.4, can be applied.

5.3 TIME DISCRETIZATION PROCEDURES

5.3.1 Classical methods

Finite differences are usually employed for time discretization. The time-stepping schemes presented in Section 3.4.1 for purely convective transport can be directly employed for convection–diffusion–reaction problems. A noticeable exception is the one-step Lax–Wendroff method introduced in Section 3.4.1.2, which cannot be used in connection with $\mathcal{C}^0$-continuous finite elements due to the presence of the second-order diffusion operator. Thus, if standard finite elements must be used, the time-marching schemes should only involve first time derivatives of the unknown.

The most popular methods for parabolic problems are the *θ family of methods* already discussed in Section 3.4.1.1. The θ scheme, see (3.19), applied to the convection–diffusion–reaction equation (5.1a) for constant coefficients yields

$$\frac{\Delta u}{\Delta t} + \theta\big[\boldsymbol{a}\cdot\boldsymbol{\nabla} - \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla}) + \sigma\big]\Delta u$$
$$= \theta s^{n+1} + (1-\theta)s^n - \big[\boldsymbol{a}\cdot\boldsymbol{\nabla} - \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla}) + \sigma\big]u^n. \quad (5.2)$$

Note that now second derivatives of u are present due to the diffusion operator. As previously observed in Section 3.4.1.1, Crank–Nicolson, $\theta = 1/2$, is the only second-order accurate method. $\theta = 1$ corresponds to the implicit backward Euler method, and $\theta = 0$ is the explicit forward Euler method. For $\theta \geq 1/2$, the scheme is A-stable. Usually, the Crank–Nicolson scheme is used for true transient problems where time accuracy is important. The backward Euler or the so-called Galerkin scheme, $\theta = 2/3$, may be used to obtain steady-state solutions by means of a time-marching approach with large time steps (usually called relaxation methods).

Another classical method for transient problems is the *Adams–Bashforth method.* This is a second-order accurate explicit method which requires evaluation of the first time derivative at two consecutive time levels. Its derivation rests upon the forward Taylor series

$$u(t^{n+1}) = u(t^n) + \Delta t\, u_t(t^n) + \frac{\Delta t^2}{2} u_{tt}(t^n) + \mathcal{O}(\Delta t^3),$$

in which the second time derivative is approximated by

$$u_{tt}(t^n) = \frac{u_t(t^n) - u_t(t^{n-1})}{\Delta t} + \mathcal{O}(\Delta t),$$

thus giving the two-level second-order explicit method

$$u^{n+1} = u^n + \frac{\Delta t}{2}\left(3u_t^n - u_t^{n-1}\right), \tag{5.3}$$

where the time derivative of the unknown u_t at a given instant must be replaced using equation (5.1a). Note that the Adams–Bashforth method is not self-starting. The first step is usually performed using the forward Euler method, preferably with a small time step.

Note that all of these methods only involve first time derivatives. Thus, they are easily implemented in conjunction with standard C^0-continuous finite elements. The reader interested in an in-depth discussion of classical time-stepping methods for convection–diffusion problems may consult the books by Mitchell and Griffiths (1980) and Gresho and Sani (2000).

Remark 5.3 (Linear multistep methods). Linear multistep methods are popular because they make use of previously computed values of u and u_t at previous steps. A k-step linear multistep (LMS) needs u^{n+1-i} and u_t^{n+1-i}, $i = 1, \dots, k$, to compute u^{n+1}. They are defined by the following expression, see for instance Gear (1971) or Hughes (2000),

$$\sum_{i=0}^{k} \alpha_i\, u^{n+1-i} = \Delta t \sum_{i=0}^{k} \beta_i\, u_t(t^{n+1-i}, u^{n+1-i}),$$

where α_i and β_i are parameters defining a specific method. The normalization $\alpha_0 = 1$ is standard and the method is explicit if $\beta_0 = 0$; otherwise, it is implicit. For instance, taking

$$\alpha_0 = -\alpha_1 = 1 \quad \text{and} \quad \beta_0 = -\theta,\ \beta_1 = \theta - 1,$$

gives the θ family of methods. Similarly, taking

$$\alpha_0 = 1,\ \alpha_1 = -1,\ \alpha_2 = 0 \quad \text{and} \quad \beta_0 = 0,\ \beta_1 = -3/2,\ \beta_2 = 1/2$$

reproduces the explicit Adams–Bashforth method, see equation (5.3). Usually, explicit methods are combined with implicit ones, of the Adams–Moulton family for instance, to produce predictor–corrector schemes.

Unfortunately, the second Dahlquist barrier (see, for instance, Hairer and Wanner, 1996) states that A-stable LMS methods are at most second-order accurate. Among the second-order methods, the Crank–Nicolson method has the smallest truncation error.

5.3.2 Fractional-step methods

Fractional-step (or operator-splitting) methods are widely used for time integration of unsteady problems. They can take different forms according to the way they split a complex problem into two or more simpler ones. For instance, in Chapter 4 a fractional-step approach was suggested, in the context of fluid–structure interaction, to separate material (Lagrangian) and convective terms. Another application of fractional-step methods will be discussed in Chapter 6 to separate the viscous terms and the pressure/incompressibility terms to properly account for the incompressibility constraint in the unsteady Navier–Stokes equations.

Here fractional-step techniques are applied to the solution of unsteady convection–diffusion–reaction problems. In this context, the original problem is transformed into the sum of a pure convection problem and a diffusion–reaction problem to be solved in sequence at each station of the time integration procedure. Methods in this class have been thoroughly studied by the Russian mathematicians Yanenko (1971) and Marchuk (1982; 1990). In view of the different characteristics of convection and diffusion operators, operator-splitting techniques allow us to select the most appropriate numerical algorithms to solve the convection and the diffusion phases.

Consider again equation (5.1a) and assume for simplicity that the convection velocity $\boldsymbol{a}$, the diffusivity ν and the reaction σ are constant. A class of fractional-step methods operate at the level of the semi-discrete form, that is, the system of ordinary differential equations obtained after spatial discretization of (5.1a), namely

$$\frac{d\mathbf{u}}{dt} + \mathbf{L}\mathbf{u} = \mathbf{f},$$

where matrix $\mathbf{L}$ arises from the spatial discretization of the convection, diffusion and reaction operators and vector $\mathbf{f}$ accounts for the source term. In this case, the splitting is of the form

$$\frac{d\mathbf{u}}{dt} + (\mathbf{L}_1 + \mathbf{L}_2)\mathbf{u} = \mathbf{f}$$

and is called *algebraic splitting*, see Quarteroni and Valli (1994).

For instance, Yanenko (1971) proposed a first-order accurate implicit method in the form of the following algebraic splitting:

$$\begin{aligned} \frac{\mathbf{u}^{n+1/2} - \mathbf{u}^n}{\Delta t} + \mathbf{L}_1\, \mathbf{u}^{n+1/2} &= 0, \\ \frac{\mathbf{u}^{n+1} - \mathbf{u}^{n+1/2}}{\Delta t} + \mathbf{L}_2\, \mathbf{u}^{n+1} &= \mathbf{f}^n. \end{aligned}$$

This method produces a couple of implicit problems, but unconditional stability can only be guaranteed for cases where both $\mathbf{L}_1$ and $\mathbf{L}_2$ are positive definite matrices.

Other widely used splitting methods are described in the technical literature (e.g., Ames, 1992; Mitchell and Griffiths, 1980; Quarteroni and Valli, 1994). These include the method of Peaceman and Rachford

$$\frac{\mathbf{u}^{n+1/2}-\mathbf{u}^n}{\Delta t/2}+\mathbf{L}_1\mathbf{u}^{n+1/2}=\mathbf{f}^{n+1/2}-\mathbf{L}_2\mathbf{u}^n,$$
$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^{n+1/2}}{\Delta t/2}+\mathbf{L}_2\mathbf{u}^{n+1}=\mathbf{f}^{n+1/2}-\mathbf{L}_1\mathbf{u}^{n+1/2},$$

and a similar one by Douglas and Rachford:

$$\frac{\mathbf{u}^{n+1/2}-\mathbf{u}^n}{\Delta t}+\mathbf{L}_1\mathbf{u}^{n+1/2}=\mathbf{f}^{n}-\mathbf{L}_2\mathbf{u}^n,$$
$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^{n+1/2}}{\Delta t}+\mathbf{L}_2\mathbf{u}^{n+1}=\mathbf{L}_2\mathbf{u}^{n}.$$

Other splitting methods operate at the level of the differential operators. They are best explained by rewriting the governing equation (5.1a) in the symbolic form

$$u_t+\mathcal{L}u=s, \qquad \text{with} \quad \mathcal{L}=\boldsymbol{a}\cdot\boldsymbol{\nabla}-\boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla})+\sigma,$$

and then the convection–diffusion–reaction operator $\mathcal{L}$ splits into a sum of two components as follows:

$$\mathcal{L}=\mathcal{L}_1+\mathcal{L}_2, \qquad \text{with} \quad \mathcal{L}_1=\boldsymbol{a}\cdot\boldsymbol{\nabla}, \quad \text{and} \quad \mathcal{L}_2=-\boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla})+\sigma.$$

An example of splitting at operator level is described by Donea, Giuliani, Laval and Quartapelle (1982) who follow Marchuk (1982). They consider a convection–diffusion problem ($\sigma = 0$) and split the operator as in the previous equation on a typical time interval Δt. Moreover, the source term is associated with the second step. This induces the following algorithm for each time step:

$$\text{First step:} \quad \begin{cases} v_t+\mathcal{L}_1 v=0 & \text{in } \Omega\times[t^n,t^{n+1}[\\ v(t^n)=u^n \end{cases}$$
$$\text{Second step:} \quad \begin{cases} w_t+\mathcal{L}_2 w=s & \text{in } \Omega\times[t^n,t^{n+1}[\\ w(t^n)=v^{n+1} \end{cases}$$
$$\text{Final update:} \quad u^{n+1}=w^{n+1}.$$

Since the first step represents a pure convection problem, methods for hyperbolic equations discussed in Chapter 3 can be employed. For instance, the third-order explicit Taylor–Galerkin scheme (TG3), see (3.42), would produce the scheme

$$\left(1-\frac{\Delta t^2}{6}\mathcal{L}_1^2\right)\frac{v^{n+1}-v^n}{\Delta t}=-\left(\mathcal{L}_1-\frac{\Delta t}{2}\mathcal{L}_1^2\right)v^n.$$

The spatial discretization of this semi-discrete equation can then be performed with the standard Galerkin finite element method.

The Crank–Nicolson method can be used for the second step, that is the diffusion equation. Thus,

$$\left(1+\frac{\Delta t}{2}\mathcal{L}_2\right)\frac{w^{n+1}-w^n}{\Delta t}=-\mathcal{L}_2\,w^n+\frac{1}{2}\left(s^n+s^{n+1}\right).$$

Here again, the spatial discretization would be performed by means of the standard Galerkin method. Donea, Giuliani, Laval and Quartapelle (1982) present a detailed discussion of the above fractional-step method which confirms its interest for approximating the solution of unsteady convection–diffusion problems.

Note that in the case of operator splitting, the sequential treatment of the convection and diffusion phases requires a splitting between the boundary conditions to endow each individual phase of a fractional-step algorithm with consistent boundary data. For instance, in the convection phase, it is clear that data can only be prescribed on the inflow portion of the boundary. Fractional-step methods are also sensitive to the treatment of the source term, see for instance Peraire (1986) and Peraire, Zienkiewicz and Morgan (1986). Another critical issue with operator-splitting methods is the overall accuracy of the procedure. Most methods are in fact only first-order accurate. The reader interested in a detailed exposition of fractional-step and operator-splitting methods is urged to consult the specialized textbooks by Marchuk (1982; 1990), Yanenko (1971), as well as Glowinski and Le Tallec (1989) and Quarteroni and Valli (1994).

5.3.3 High-order time-stepping schemes

Unsteady convection–diffusion–reaction problems are more difficult to solve using high-order time-stepping methods than the pure convection problems discussed in Chapter 3. The reason is the presence in the governing equation (5.1a) of the second-order diffusion operator. This operator limits one-step time integration algorithms to second-order temporal accuracy when they are combined with C^0-continuous finite element approximations. In order to use a standard implementation of C^0 finite elements, time-stepping schemes for convection–diffusion should only involve first time derivatives. This leads to the two-step schemes presented in Section 3.6.4, or more generally, to Runge–Kutta methods (Lambert, 1991; Hairer, Nørsett and Wanner, 1993) or to multistage schemes emanating from the factorization of Padé approximations to the exponential function (Donea, Roig and Huerta, 1998).

5.3.3.1 Runge–Kutta methods The technical literature contains many references to the use of Runge–Kutta time-stepping schemes in connection with finite element and finite volume algorithms. Only a few representative examples are mentioned here. For instance, they can be used to reach a steady solution (the concern is efficiency and not time accuracy), see Jameson, Schmidt and Turkel (1981) or Liu and Jameson (1993). Iannelli and Baker (1991) use a stiffly stable, second-order accurate, implicit Runge–Kutta method for Euler and Navier–Stokes aerodynamic applications. Tezduyar, Mittal and Shih (1991) and Jiang and Kawahara (1993) employ a third-order explicit Runge–Kutta method for the finite element analysis of unsteady incompressible flows. Giles (1997) presents an application of modern stability analysis to the Galerkin formulation of Runge–Kutta methods applied to the Navier–Stokes equations. In particular, examples of predictor–corrector and explicit Pad é approximations are shown. A fourth-order method is used by Pereira et al. (2001) for a finite volume solution of the incompressible Navier–Stokes equations. In the context of nonlinear convection-dominated problems, Cockburn and Shu (2001) review the development of the Runge–Kutta discontinuous Galerkin methods.

Runge–Kutta methods are multistage methods that only make use of the solution u^n at time t^n to compute the solution u^{n+1} at t^{n+1}. This is achieved by computing a number $\mathtt{n_{tg}}$ of intermediate values of the time derivative of the unknown u, within the interval $\Delta t = t^{n+1} - t^n$. The differential equation (5.1a) is rewritten as

$$u_t + \mathcal{L}(u) = s, \quad \text{or} \quad u_t = \mathcal{F}(t, u) \tag{5.4}$$

where the spatial differential operators are defined as

$$\mathcal{L} := \boldsymbol{a} \cdot \boldsymbol{\nabla} + \sigma - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla}), \quad \text{and} \quad \mathcal{F}(t, u) := s - \mathcal{L}(u). \tag{5.5}$$

Then, the standard form (see Lambert, 1991; Hairer et al., 1993; Hairer and Wanner,

[redacted]

$$u^{n+1} = u^n + \Delta t \sum_{i=1}^{\mathtt{n_{tg}}} b_i \, \mathcal{F}(t^n + c_i \Delta t\,,\, u^{n+\beta_i}), \tag{5.7a}$$

$$u^{n+\beta_i} = u^n + \Delta t \sum_{j=1}^{\mathtt{n_{tg}}} a_{i,j} \, \mathcal{F}(t^n + c_j \Delta t\,,\, u^{n+\beta_j}), \quad i = 1, 2, \ldots, \mathtt{n_{tg}} \tag{5.7b}$$

using the interpretation

and they are conveniently displayed as a *Butcher array*:

$$\begin{array}{c|cccc}
c_1 & a_{1,1} & a_{1,2} & \cdots & a_{1,\mathrm{n_{tg}}} \\
c_2 & a_{2,1} & a_{2,2} & \cdots & a_{2,\mathrm{n_{tg}}} \\
\vdots & \vdots & \vdots & & \vdots \\
c_{\mathrm{n_{tg}}} & a_{\mathrm{n_{tg}},1} & a_{\mathrm{n_{tg}},2} & \cdots & a_{\mathrm{n_{tg}},\mathrm{n_{tg}}} \\
\hline
 & b_1 & b_2 & \cdots & b_{\mathrm{n_{tg}}}
\end{array}$$

Implicit Runge–Kutta methods are those where at least one $a_{i,j} \neq 0$ for $j \geq i$. If this is not the case, the method is explicit.

5.3.3.2 *Explicit Runge–Kutta methods* It is well-known that $\mathrm{n_{tg}}$-stage explicit Runge–Kutta methods are of order $\mathrm{n_{tg}}$ for $\mathrm{n_{tg}} \leq 4$. For $\mathrm{n_{tg}} > 4$, $\mathrm{n_{tg}}$-stage methods systematically have order lower than $\mathrm{n_{tg}}$. For this reason, Runge–Kutta methods of fourth-order are very popular among the explicit methods. The classical fourth-order Runge–Kutta method is defined by the following Butcher array:

$$\begin{array}{c|cccc}
0 & & & & \\
1/2 & 1/2 & & & \\
1/2 & 0 & 1/2 & & \\
1 & 0 & 0 & 1 & \\
\hline
 & 1/6 & 1/3 & 1/3 & 1/6
\end{array}$$

The resulting time-stepping algorithm follows from (5.7) and reads

$$\begin{aligned}
u^{n+\beta_1} &= u^n, & u^{n+\beta_2} &= u^n + \tfrac{\Delta t}{2}\mathcal{F}(t^n + \tfrac{\Delta t}{2}, u^{n+\beta_1}), \\
u^{n+\beta_3} &= u^n + \tfrac{\Delta t}{2}\mathcal{F}(t^n + \tfrac{\Delta t}{2}, u^{n+\beta_2}), & u^{n+\beta_4} &= u^n + \Delta t\mathcal{F}(t^n + dt, u^{n+\beta_3}), \\
u^{n+1} &= u^n + \tfrac{\Delta t}{6}\big(\mathcal{F}(t^n, u^{n+\beta_1}) + 2\mathcal{F}(t^n + \tfrac{\Delta t}{2}, u^{n+\beta_2}) \\
&\qquad + 2\mathcal{F}(t^n + \tfrac{\Delta t}{2}, u^{n+\beta_3}) + \mathcal{F}(t^n + \Delta t, u^{n+\beta_4})\big).
\end{aligned}$$

Remark 5.4 (Stability properties). Explicit Runge–Kutta methods are only conditionally stable. In application to the modal equation

$$u_t + \lambda u = 0,$$

where λ denotes a typical eigenvalue, their amplification factor has the same structure as the corresponding Padé approximation (first row in Table 5.1)

$$G_{(RK,\mathrm{n_{tg}})}(\lambda \Delta t) = \mathsf{R}_{\mathrm{n_{tg}},0}(-\lambda \Delta t).$$

Their domain of numerical stability in the complex $\lambda\Delta t$ plane is depicted in Figure 5.1. The fourth-order method presented previously has interesting stability properties. The associated absolute stability curve, shown in Figure 5.1, is the same as that of Padé approximation $\mathsf{R}_{4,0}$. It cuts the real and imaginary axes at -2.78 and $\pm 2\sqrt{2}$, respectively. Since the absolute stability region

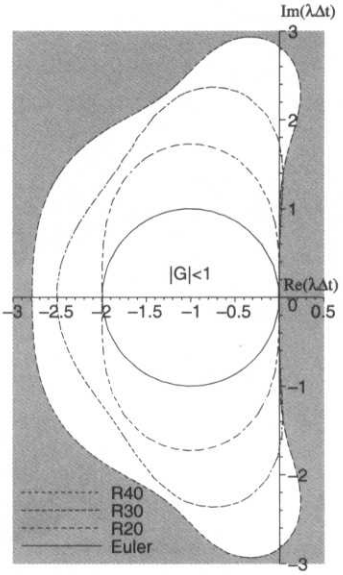


Fig. 5.1 Stability domain of explicit Runge–Kutta and Padé methods.

contains a finite portion of the imaginary axis, the fourth-order Runge–Kutta method can be used in convection-dominated situations where the problem eigenvalues are distributed close to the imaginary axis of the complex $\lambda\Delta t$ plane.

5.3.3.3 Explicit Padé approximations As an alternative to explicit Runge–Kutta methods, one may consider using a multistage factorization of the explicit Pad é approximations $\mathsf{R}_{n,0}$ to the exponential function, see the first row in Padé Table 5.1 and set $z = \Delta t \partial/\partial t$, see Donea, Roig and Huerta (1998; 2000). They are easier to implement in conjunction with finite elements due to their simple algorithmic structure and possess the same properties as their corresponding Runge–Kutta methods, for instance the same stability domain, if the coefficients in (5.1a) do not depend on time.

For the second-order Padé approximation $\mathsf{R}_{2,0}$, a two-stage method can be derived from the factorization

$$u(t^{n+1}) = u(t^n) + \Delta t \frac{\partial}{\partial t}\Big(u + \frac{\Delta t}{2}\frac{\partial u}{\partial t}\Big)\Big|_{t=t^n} + \mathcal{O}(\Delta t^3),$$

which yields the two-stage ($\mathrm{n_{tg}} = 2$) Lax–Wendroff method

$$\begin{aligned} u^{n+1/2} &= u^n + \frac{\Delta t}{2} u_t^n; \\ u^{n+1} &= u^n + \Delta t \, u_t^{n+1/2}. \end{aligned}$$

Table 5.1 Padé approximations of the exponential function e^z.

$\mathsf{R}_{n,m}(z)$	$n=0$	$n=1$	$n=2$	$n=3$
$m=0$	1	$1+z$	$1+z+\frac{z^2}{2}$	$1+z+\frac{z^2}{2}+\frac{z^3}{6}$
$m=1$	$\frac{1}{1-z}$	$\frac{1+\frac{z}{2}}{1-\frac{z}{2}}$	$\frac{1+\frac{2z}{3}+\frac{z^2}{6}}{1-\frac{z}{3}}$	$\frac{1+\frac{3z}{4}+\frac{z^2}{4}+\frac{z^3}{24}}{1-\frac{z}{4}}$
$m=2$	$\frac{1}{1-z+\frac{z^2}{2}}$	$\frac{1+\frac{z}{3}}{1-\frac{2z}{3}+\frac{z^2}{6}}$	$\frac{1+\frac{z}{2}+\frac{z^2}{12}}{1-\frac{z}{2}+\frac{z^2}{12}}$	$\frac{1+\frac{3z}{5}+\frac{3z^2}{20}+\frac{z^3}{60}}{1-\frac{2z}{5}+\frac{z^2}{20}}$
$m=3$	$\frac{1}{1-z+\frac{z^2}{2}-\frac{z^3}{6}}$	$\frac{1+\frac{z}{4}}{1-\frac{3z}{4}+\frac{z^2}{4}-\frac{z^3}{24}}$	$\frac{1+\frac{2z}{5}+\frac{z^2}{20}}{1-\frac{3z}{5}+\frac{3z^2}{20}-\frac{z^3}{60}}$	$\frac{1+\frac{z}{2}+\frac{z^2}{10}+\frac{z^3}{120}}{1-\frac{z}{2}+\frac{z^2}{10}-\frac{z^3}{120}}$

The third-order Padé approximation $\mathsf{R}_{3,0}$ can also be factorized in the form

$$u(t^{n+1}) = u(t^n) + \Delta t \frac{\partial}{\partial t}\left(u + \frac{\Delta t}{2}\frac{\partial}{\partial t}\left(u + \frac{\Delta t}{3}\frac{\partial u}{\partial t}\right)\right)\bigg|_{t=t^n} + \mathcal{O}(\Delta t^4),$$

which yields the three-stage ($\mathtt{n}_{\mathtt{tg}} = 3$) method

$$\begin{aligned} u^{n+1/3} &= u^n + \frac{\Delta t}{3} u_t^n, \\ u^{n+1/2} &= u^n + \frac{\Delta t}{2} u_t^{n+1/3}, \\ u^{n+1} &= u^n + \Delta t\, u_t^{n+1/2}. \end{aligned}$$

This third-order scheme has been used by Jiang and Kawahara (1993) and by Tezduyar et al. (1991) in the finite element analysis of unsteady incompressible flows.

Similarly higher-order methods can be developed (for constant-in-time coefficients). In fact, multistage explicit methods of the Padé family are easily expressed in the incremental form

$$\begin{aligned} u^{n+\beta_1} &= u^n, \\ u^{n+\beta_i} &= u^n + \beta_i\, \Delta t\, u_t^{n+\beta_{i-1}}, \qquad i = 2, \ldots, \mathtt{n}_{\mathtt{tg}} + 1, \end{aligned} \tag{5.8}$$

where $\beta_1 = 0$, and $\beta_i = 1/(\mathtt{n}_{\mathtt{tg}}+2-i), i = 2, \ldots, \mathtt{n}_{\mathtt{tg}}+1$. Note that these methods can be expressed using a Butcher array:

$$\begin{array}{c|ccccc} 0 & 0 & & & & \\ 1/\mathtt{n}_{\mathtt{tg}} & 1/\mathtt{n}_{\mathtt{tg}} & 0 & & & \\ \frac{1}{\mathtt{n}_{\mathtt{tg}}-1} & 0 & \frac{1}{\mathtt{n}_{\mathtt{tg}}-1} & 0 & & \\ \vdots & \vdots & \ddots & \ddots & \ddots & \\ 1/2 & 0 & \cdots & 0 & 1/2 & 0 \\ \hline & 0 & \cdots & \cdots & 0 & 1 \end{array}$$

Then, the time derivative is replaced using the convection–diffusion–reaction equation (5.1a). Finally, as shown in Section 5.4, a weak form of the resulting semi-discrete system is introduced to serve as a basis for the finite element spatial discretization.

Remark 5.5. Note that although explicit multistage Padé schemes can be written using a Butcher array, they are not Runge–Kutta methods. For instance, one can easily check that the coefficients of the three-stage method do not verify the required conditions (see for instance Hairer et al., 1993, p. 144). The factorization of the Taylor series and the subsequent substitution of u_t can be performed if the operators in (5.5) are linear (not dependent on u) and constants ($\boldsymbol{a}$, σ, ν and s are independent of t).

5.3.3.4 Implicit Runge–Kutta methods Implicit Runge–Kutta (IRK) methods are popular in the ordinary differential equations community (Lambert, 1991; Hairer and Wanner, 1996). Their elevated cost, when implemented in the context of convection–diffusion equations, has deterred their use. For each time step the IRK method requires the simultaneous solution of (5.7b) to determine $u^{n+\beta_i}$ for $i = 1, 2, \ldots, \mathrm{n}_{\mathrm{tg}}$, or, equivalently, solution of (5.6b) to determine every $\boldsymbol{\ell}_i$. Then, equation (5.7a), or (5.6a), can be used to compute u^{n+1} explicitly.

The IRK methods based on Lobatto quadrature are of particular interest for the time integration of unsteady transport problems, because they include the end points of the integration interval, namely $u^{n+\beta_1} = u^n$ and $u^{n+\beta_{\mathrm{n}_{\mathrm{tg}}}} = u^{n+1}$. This reduces the computational cost: the system of equations has one equation less and equation (5.7a) is verified automatically. They have a maximum order of $2\mathrm{n}_{\mathrm{tg}} - 2$ where n_{tg} is the number of stages. The first family of such methods is called Lobatto IIIA and it is based on the following Butcher arrays:

$$\mathrm{n}_{\mathrm{tg}} = 2 \text{ (thus order 2)} \qquad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \end{array}$$

$$\mathrm{n}_{\mathrm{tg}} = 3 \text{ (thus order 4)} \qquad \begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ 1/2 & 5/24 & 1/3 & -1/24 \\ 1 & 1/6 & 2/3 & 1/6 \\ \hline & 1/6 & 2/3 & 1/6 \end{array}$$

Note that the first row of the matrix in the Butcher arrays is, for Lobatto IIIA methods, always zero as well as c_1, that is $a_{1,1} = a_{1_2} = \cdots = a_{1,\mathrm{n}_{\mathrm{tg}}} = 0$. Thus, as expected, the first stage that corresponds to u^n is not evaluated. Therefore, the system of simultaneous equations that must be solved at each time step has dimension $\mathrm{n}_{\mathrm{tg}} - 1$. Moreover, the last explicit computation in (5.7a) corresponding to the evaluation of u^{n+1} is also unnecessary because the last row of that matrix is identical to $[b_1, b_2, \ldots, b_{\mathrm{n}_{\mathrm{tg}}}]$.

5.3.3.5 Implicit Padé approximations Lobatto IIIA IRK methods in the previous section are closely related to the Padé approximations $\mathsf{R}_{n,n}$ on the diagonal of Table 5.1. As shown by Donea et al. (1998; 2000), multistage schemes can be derived directly from the diagonal Padé approximations. They are obtained in the following compact form:

$$\frac{\Delta \boldsymbol{u}}{\Delta t} - \mathbf{W}\,\Delta \boldsymbol{u}_t = \mathbf{w}\,u_t^n, \tag{5.9}$$

where the unknown $\Delta \boldsymbol{u} \in \mathbb{R}^{\mathrm{n_{tg}}-1}$ is a vector whose dimension varies with the number of stages, $\mathrm{n_{tg}}$, and the vector $\Delta \boldsymbol{u}_t$ is the partial derivative of $\Delta \boldsymbol{u}$ with respect to time, namely

$$\Delta \boldsymbol{u} = \begin{Bmatrix} u^{n+\beta_2} - u^n \\ u^{n+\beta_3} - u^{n+\beta_2} \\ \vdots \\ u^{n+1} - u^{n+\beta_{\mathrm{n_{tg}}-1}} \end{Bmatrix}; \quad \text{and } \Delta \boldsymbol{u}_t = \begin{Bmatrix} u_t^{n+\beta_2} - u_t^n \\ u_t^{n+\beta_3} - u_t^{n+\beta_2} \\ \vdots \\ u_t^{n+1} - u_t^{n+\beta_{\mathrm{n_{tg}}-1}} \end{Bmatrix}. \tag{5.10}$$

$\mathbf{W}$ is an $(\mathrm{n_{tg}}-1)\times(\mathrm{n_{tg}}-1)$ matrix and $\mathbf{w}$ an $(\mathrm{n_{tg}}-1)$ vector. They are defined below for two specific cases. As with IRK methods, the time derivatives in (5.9) are replaced by spatial derivatives using the original differential equation, see equations (5.4) and (5.5). For instance, in the case of a linear operator $\mathcal{L}$ with constant coefficients and a source term s, equation (5.9) becomes

$$\frac{\Delta \boldsymbol{u}}{\Delta t} + \mathbf{W}\mathcal{L}(\Delta \boldsymbol{u}) = \mathbf{w}\big[s^n - \mathcal{L}(u^n)\big] + \mathbf{W}\Delta \boldsymbol{s}. \tag{5.11}$$

The precise definition of $\Delta \boldsymbol{u}$ (recall that $\Delta \boldsymbol{u}_t$ is simply $\partial \Delta \boldsymbol{u}/\partial t$), $\Delta \boldsymbol{s}$, $\mathbf{w}$ and $\mathbf{W}$ depends on each particular method.

As noted earlier, these methods are closely related to Lobatto IIIA IRK methods. The noticeable difference is that, while the present scheme is based on non-overlapping step increments, the corresponding Runge–Kutta method employs total increments. This implies that a linear correspondence exits between both techniques, namely

$$\begin{Bmatrix} u^{n+\beta_2} - u^n \\ u^{n+\beta_3} - u^n \\ \vdots \\ u^{n+1} - u^n \end{Bmatrix} = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 1 & 1 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 & 0 \\ 1 & \cdots & \cdots & 1 & 1 \end{pmatrix} \begin{Bmatrix} u^{n+\beta_2} - u^n \\ u^{n+\beta_3} - u^{n+\beta_2} \\ \vdots \\ u^{n+1} - u^{n+\beta_{\mathrm{n_{tg}}-1}} \end{Bmatrix}.$$

This minor difference does not affect linear problems discretized with a Galerkin formulation. However, as shown in Section 5.4.6.5, it has major consequences as regards numerical stability when implemented in connection with stabilized spatial formulations.

We shall illustrate the construction of multistage Padé schemes with reference to the second-order approximation $\mathsf{R}_{1,1}$ and the fourth-order approximation $\mathsf{R}_{2,2}$, for which equation (5.9) takes the following particular forms:

***Second-order Padé approximation:* $\mathsf{R}_{1,1}$ (Crank–Nicolson)**

$$\Delta \boldsymbol{u} = u^{n+1} - u^{n}, \qquad \Delta \boldsymbol{s} = s^{n+1} - s^{n},$$

$$\mathbf{W} = \frac{1}{2}, \qquad \mathbf{w} = 1.$$

Note that in this case $\mathtt{n_{tg}} = 1$ and, consequently, the vectors and matrix in (5.9) become scalars.

***Fourth-order Padé approximation:* $\mathsf{R}_{2,2}$**

$$\Delta \boldsymbol{u} = \begin{Bmatrix} u^{n+1/2} - u^{n} \\ u^{n+1} - u^{n+1/2} \end{Bmatrix}; \qquad \Delta \boldsymbol{s} = \begin{Bmatrix} s^{n+1/2} - s^{n} \\ s^{n+1} - s^{n+1/2} \end{Bmatrix};$$

$$\mathbf{W} = \frac{1}{24}\begin{pmatrix} 7 & -1 \\ 13 & 5 \end{pmatrix}; \qquad \mathbf{w} = \frac{1}{2}\begin{Bmatrix} 1 \\ 1 \end{Bmatrix}.$$

Both equations (5.9) and (5.11) with the corresponding initial and boundary conditions define a problem in *strong form*, which must be solved at each time step. For simplicity, the truncation errors are not explicitly shown in (5.9) and (5.11). Note, however, that if these truncation errors are accounted for, equations (5.1a), (5.9) and (5.11) are equivalent, and the exact solution verifies the new strong form. The spatially continuous differential equations, (5.9) or (5.11), are the basis of the finite element discretization.

5.4 SPATIAL DISCRETIZATION PROCEDURES

5.4.1 Galerkin formulation of the semi-discrete scheme

Here, we present the Galerkin formulation associated with the strong form (5.1) of the unsteady convection–diffusion–reaction problem. As in Section 3.4.2, Galerkin can be applied directly on the differential equation (5.1a), or on the time discretized equations (5.2), (5.3), (5.8), (5.7) or (5.9). For completeness, we present the *semi-discrete* scheme first.

The space of weighting functions denoted by $\mathcal{V}$ satisfies the homogeneous boundary conditions on Γ_D. The functions, w, in $\mathcal{V}$ do not depend on time. As in Section 3.4.2.1, the time dependency of the approximate solution u can be translated to the trial space $\mathcal{S}_t$, which varies as a function of time,

$$\mathcal{S}_t := \left\{ u \mid u(\cdot, t) \in \mathcal{H}^1(\Omega), t \in [0, T] \text{ and } u(\boldsymbol{x}, t) = u_D \text{ for } \boldsymbol{x} \in \Gamma_D \right\}.$$

As usual, trial solutions a priori verify Dirichlet boundary conditions. The weak form of the initial boundary value problem (5.1) is defined as follows: given s, u_D, h, u_ϵ

and u_0, for any $t \in [0,T]$ find $u(\boldsymbol{x},t) \in \mathcal{S}_t$, such that for all $w \in \mathcal{V}$

$$\big(w, u_t\big) + c\big(\boldsymbol{a}; w, u\big) + a\big(w, u\big) + \big(w, \sigma u\big) = \big(w, s\big) + \big(w, h\big)_{\Gamma_N}. \tag{5.12}$$

We recall the definitions already presented in (2.6), namely

$$a(w,u) = \int_\Omega \boldsymbol{\nabla} w \cdot (\nu \boldsymbol{\nabla} u) d\Omega, \qquad (w,s) = \int_\Omega w\, s\, d\Omega,$$
$$c(\boldsymbol{a}; w,u) = \int_\Omega w(\boldsymbol{a} \cdot \boldsymbol{\nabla} u) d\Omega, \qquad (w,h)_{\Gamma_N} = \int_{\Gamma_N} w\, h\, d\Gamma.$$

The spatial discretization by means of the Galerkin formulation consists of defining two finite dimensional spaces $\mathcal{S}^h$ and $\mathcal{V}^h$, subsets of $\mathcal{S}$ and $\mathcal{V}$, see Section 3.4.2.2. Then, the semi-discrete Galerkin formulation is obtained by restricting the previous weak form to these finite dimensional spaces.

The system of ordinary differential equations is obtained following the same rationale as in Section 3.4.2.2. Recall that the time dependence of the solution, $u^h(\boldsymbol{x},t)$, is taken into account by the nodal values of the unknown. The shape functions $N_A(\boldsymbol{x})$ do not depend on time. Thus, (2.10) becomes

$$u^h(\boldsymbol{x},t) = \sum_{A \in \eta \setminus \eta_D} N_A(\boldsymbol{x})\, \mathrm{u}_A(t) + \sum_{A \in \eta_D} N_A(\boldsymbol{x})\, u_D(\boldsymbol{x}_A, t),$$

where, as before, η is the set of global node numbers in the finite element mesh and $\eta_D \subset \eta$ the subset of nodes belonging to the Dirichlet portion of the boundary, Γ_D. The test functions are defined as before, see (2.11), $w^h \in \mathcal{V}^h = \mathrm{span}_{B \in \eta \setminus \eta_D}\{N_B\}$.

Finally, the usual assembly process delivers the semi-discrete system of ordinary differential equations

$$\mathbf{M}\dot{\mathbf{u}} + (\mathbf{C} + \mathbf{K} + \sigma \mathbf{M})\mathbf{u} = \mathbf{f}, \tag{5.13}$$

where, for simplicity, we have assumed that the reaction, σ, is uniform and constant. Note that vectors $\mathbf{u}$ and $\dot{\mathbf{u}}$ contain, respectively, the nodal values of the unknown u and of its time derivative, while $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$ are, respectively, the consistent mass matrix, the convection matrix and the diffusion matrix. These matrices are obtained by topological assembly of element contributions as follows:

$$\begin{aligned} \mathbf{M} &= \mathop{\mathbf{A}}^{e} \mathbf{M}^e & M^e_{ab} &= \int_{\Omega^e} N_a N_b \, d\Omega \\ \mathbf{C} &= \mathop{\mathbf{A}}^{e} \mathbf{C}^e & C^e_{ab} &= \int_{\Omega^e} N_a (\boldsymbol{a} \cdot \boldsymbol{\nabla} N_b) d\Omega \\ \mathbf{K} &= \mathop{\mathbf{A}}^{e} \mathbf{K}^e & K^e_{ab} &= \int_{\Omega^e} \boldsymbol{\nabla} N_a \cdot (\nu \boldsymbol{\nabla} N_b)\, d\Omega \end{aligned} \tag{5.14}$$

where $\mathbf{A}$ denotes the assembly operator, $1 \le a, b \le \mathrm{n_{en}}$ and $\mathrm{n_{en}}$ is the number of element nodes. The r.h.s. vector, $\mathbf{f}$, considers the contribution of the source term,

s, the prescribed flux, h, and the Dirichlet data u_D. It results from the assembly of nodal contributions of the form $\mathbf{f} = \mathbf{A}^e \mathbf{f}^e$ and

$$\mathrm{f}_a^e = \left(N_a, s\right)_{\Omega^e} + \left(N_a, h\right)_{\partial\Omega^e \cap \Gamma_N} - \sum_{b=1}^{n_{en}} \left[\left(N_a, N_b\right)_{\Omega^e} \frac{\partial u_{Db}^e(t)}{\partial t} \right.$$
$$\left. + \left(c(\boldsymbol{a}; N_a, N_b)_{\Omega^e} + a(N_a, N_b)_{\Omega^e} + \left(N_a, \sigma N_b\right)_{\Omega^e} \right) u_{Db}^e(t) \right],$$

where $u_{Db}^e(t) = u_D(\boldsymbol{x}_b, t)$ if u_D is prescribed at node number b and zero otherwise.

5.4.2 Galerkin formulation of θ family methods

Which discretization is performed first is not an issue for linear spatial operators with constant coefficients and a Galerkin formulation. The time discretized equations, (5.2), (5.3), (5.8), (5.7) or (5.9), do have a truncation error. However, if the temporal truncation error is neglected, these equations can be interpreted as a spatial differential operator. In fact, they represent a strong form that must be solved at each time step.

Under this rationale it is easy to determine from (5.2) the variational form associated with the θ family methods:

$$\left(w, \frac{\Delta u}{\Delta t}\right) + \theta\left[c(\boldsymbol{a}; w, \Delta u) + a(w, \Delta u) + \left(w, \sigma \Delta u\right)\right]$$
$$= -\left[c(\boldsymbol{a}; w, u^n) + a(w, u^n) + \left(w, \sigma u^n\right)\right]$$
$$+ \left(w, \theta s^{n+1} + (1-\theta) s^n\right) + \left(w, \theta h^{n+1} + (1-\theta) h^n\right). \quad (5.15)$$

Note that in the previous equation the unknown Δu appears in the same three terms as in the l.h.s. of (5.12). Thus, after spatial discretization, we obtain the same mass, convection and diffusion matrices (scaled by $1/\Delta t$ and θ) as in (5.13) and (5.14). The r.h.s. terms also include the influence of the Neumann boundary condition, h, and the source term, s. These terms are linearly interpolated between time t^n and t^{n+1}. It is important to note that time is already discretized in equation (5.15). Therefore, the solutions of (5.15) are first or second-order approximations in time depending on the value of θ.

In order to analyze this family of methods we follow the methodology proposed in Section 3.5.1, which already included diffusion and reaction. Thus to determine the influence of the numerical scheme on a Fourier mode of wavelength $1/k$ one introduces the dimensionless wave number, $\xi = h\,k$. Recall the dimensionless scalars: Courant number $C = a\Delta t/h$, diffusion number $d = (\nu \Delta t)/h^2$, and dimensionless reaction $r = \sigma \Delta t$. Then, we follow the same steps of Section 3.5.2; the discrete equation obtained at an interior node j with the θ family method is determined from (5.15) (recall that there is no source term, $s = 0$, and we consider an interior node).

The fully discrete equation can be written from Table 3.1 as

$$\left(\left(1+\frac{\delta^2}{6}\right)+\theta\left(\frac{C\,\delta}{2}-d\,\delta^2+r\left(1+\frac{\delta^2}{6}\right)\right)\right)(u_j^{n+1}-u_j^n) = -\theta\left(\frac{C\,\delta}{2}-d\,\delta^2+r\left(1+\frac{\delta^2}{6}\right)\right)u_j^n,$$

which allows us to find the equation that modifies each Fourier component,

$$\Big(\mathcal{M}(\xi)-\theta\big(\mathcal{K}(\xi,d)-\mathcal{A}(\xi,C)-r\mathcal{M}(\xi)\big)\Big)(u_j^{n+1}-u_j^n) = \big(\mathcal{K}(\xi,d)-\mathcal{A}(\xi,C)-r\mathcal{M}(\xi)\big)u_j^n.$$

Thus the numerical amplification factor for the convection–diffusion–reaction equation and a Galerkin formulation is

$$G_\theta=\frac{\mathcal{M}(\xi)-(1-\theta)(\mathcal{K}(\xi,d)-\mathcal{A}(\xi,C)-r\mathcal{M}(\xi))}{\mathcal{M}(\xi)-\theta(\mathcal{K}(\xi,d)-\mathcal{A}(\xi,C)-r\mathcal{M}(\xi))}. \tag{5.16}$$

Using the last column in Table 3.1 one can explicitly determine the dependence of G_θ on C, d, r, θ and ξ. Then stability can be studied for each value of θ verifying the condition $|G_\theta|\leq 1$. In particular for $\theta\geq 1/2$ one can verify that the amplification factor is always (for $0\leq\xi\leq\pi$ and $C\geq 0$) less than one. Thus, Crank–Nicolson and backward Euler are *unconditionally stable*. Figure 5.2 plots $|G_\theta|$ as a function of ξ for different values of the Péclet, $\mathit{Pe}=C/(2d)$, and Courant, C, numbers. The curves are plotted in polar coordinates, the angle is ξ and the radius is $|G_\theta|$. The unit radius circle is also plotted. Thus when a method produces a radius larger than one it is unstable. Note that as expected the Euler method is *conditionally stable*, see Remark 5.6. The stability results for the pure diffusion case are well-known (see Wait and Mitchell, 1985) and can be determined from (5.16),

$$d\leq\frac{1}{6(1-2\theta)}\leq\frac{1}{6(1-2\theta)}\,\frac{2\big(2+\cos(\xi)\big)}{1-\cos(\xi)}.$$

Remark 5.6 (Stability range of explicit methods). In pure diffusion in 1D, the forward Euler method is stable for

$$\Delta t\leq\frac{h^2}{6\nu}\quad\text{or}\quad d\leq\frac{1}{6}.$$

The method is unstable in pure convection if a centered (Galerkin) approximation is used for the convective term. When an upwind approximation is employed, the method is stable provided $\Delta t\leq h/a$ (Courant condition). In convection–diffusion, the stability condition depends on the value of Pe:

$$\begin{cases}\text{if } \mathit{Pe}\leq\sqrt{3}, & C\leq \mathit{Pe}/3,\\ \text{if } \mathit{Pe}>\sqrt{3}, & C\leq 1/\mathit{Pe}.\end{cases}$$

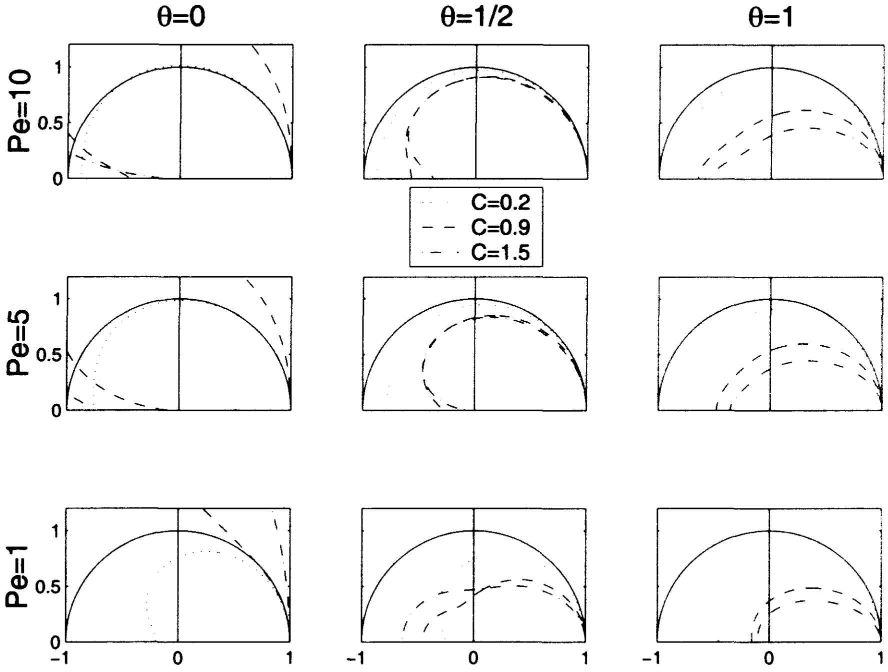


Fig. 5.2 Stability of the θ family methods for different values of Péclet and Courant numbers.

Stability of the second-order Adams–Bashforth method is governed by the same type of criteria, but its stability range is only half that for the Euler method. The leap-frog (LF) method is unstable in pure diffusion, and it is also unstable in pure convection if an upwind approximation is used for the convective term. When a centered approximation is employed, the LF method is stable under the Courant condition $\Delta t \leq h/a$. See, for instance, Gresho and Sani (2000) for additional information on the stability of explicit methods in convection–diffusion.

The accuracy of these methods can be evaluated by means of (3.36). Figure 5.3 shows relative errors at different Péclet numbers for numerical damping, $|G|/|G_{\text{ex}}|$, and phase error, $\arg(G)/\arg(G_{\text{ex}})$. From a precision point of view only the range $0 \leq \xi \leq \pi/4$ is of interest. Note that backward Euler is as expected overly diffusive, $|G|/|G_{\text{ex}}| \leq 1$. Moreover, reasonable phase errors require small Courant numbers.

Crank–Nicolson shows better accuracy due to its second-order accuracy, see Figure 5.4. However, its excellent behavior in damping for high values of Pe degrades as the diffusion coefficient and the Courant number increase. Moreover, its phase error is always important for moderately high values of the Courant number. For both methods the introduction of a reaction does not drastically affect these conclusions.

Apart from stability and accuracy (in time) we should recall the intrinsic deficiencies of the Galerkin formulation in the presence of boundary layers. As shown in the

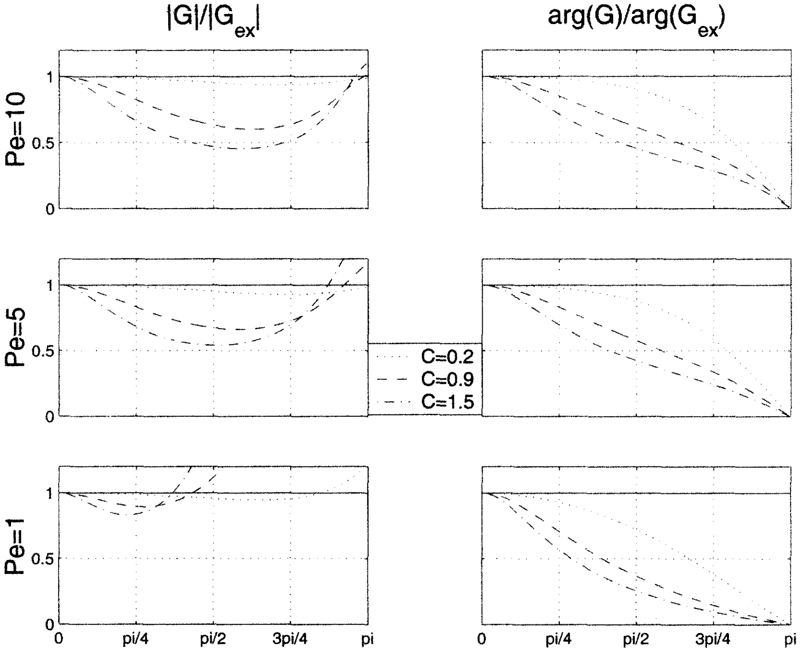


Fig. 5.3 Accuracy properties of backward Euler at different values of the Péclet number.

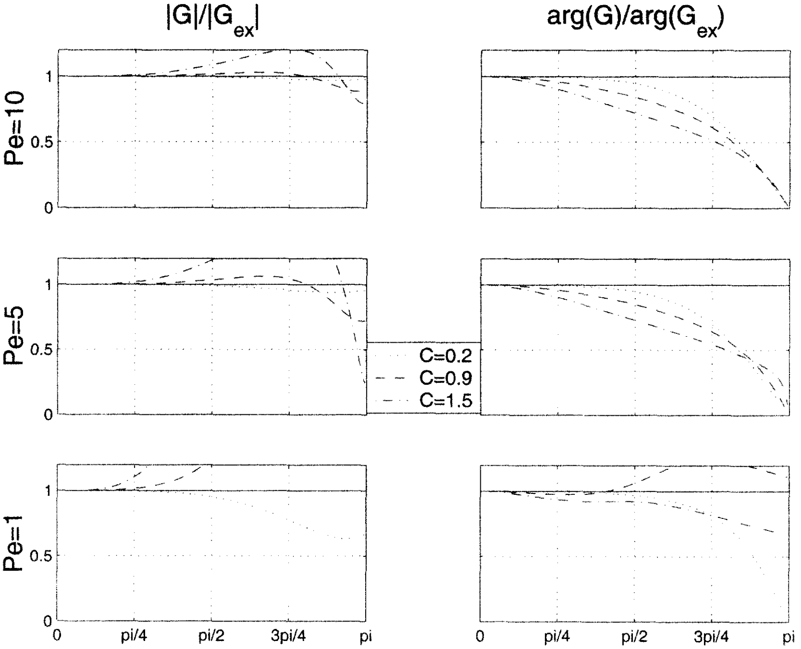


Fig. 5.4 Accuracy properties of Crank–Nicolson at different values of the Péclet number.

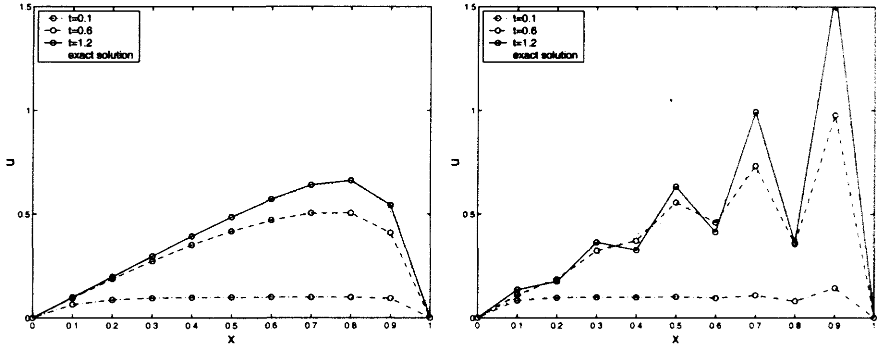


Fig. 5.5 Model problem for Crank–Nicolson at *Pe* = 0.5 (left) and *Pe* = 5 (right).

examples, if a smooth function is transported far from Dirichlet boundaries Galerkin will produce reasonable results. The previous accuracy analysis can be confirmed and, as seen in Chapter 3, Crank–Nicolson performs better (both in numerical damping and phase) than lower-order methods. Nevertheless, if the solution is confronted with Dirichlet boundary conditions developing a boundary layer, spatial instabilities (already seen in Chapter 2) will pollute the solution and spatial stabilization is required. A simple example using a uniform mesh of linear elements illustrates this point. It is the transient version of the example presented in Section 2.2.2, namely

$$\begin{cases} u_t + a\,u_x - \nu\,u_{xx} = 1 & \text{for } (x,t) \in \,]0,1[\times\mathbb{R}^+, \\ u(0,t) = 0 \quad \text{and} \quad u(1,t) = 0 & \text{for } t \in \mathbb{R}^+, \\ u(x,0) = 0 & \text{for } x \in \,]0,1[. \end{cases} \tag{5.17}$$

Figure 5.5 compares exact solution with results of the Galerkin/Crank–Nicolson formulation for different values of *Pe* and different instants.

5.4.3 Galerkin formulation of explicit Padé schemes

The Galerkin formulation of multistage explicit Padé methods, see (5.8), is

$$\begin{aligned}(w, u^{n+\beta_i}) = (w, u^n) + \beta_i\,\Delta t\Big[(w, s^{n+\beta_{i-1}}) + (w, h^{n+\beta_{i-1}})_{\Gamma_N} \\ - c(\boldsymbol{a}; w, u^{n+\beta_{i-1}}) - a(w, u^{n+\beta_{i-1}}) - (w, \sigma\,u^{n+\beta_{i-1}})\Big],\end{aligned}$$

for $i = 2, \dots, \mathrm{n_{tg}} + 1$. Recall that the first stage is trivial, $u^{n+\beta_1} = u^n$, thus $\beta_1 = 0$, and that $\beta_i = 1/(\mathrm{n_{tg}} + 2 - i)$, for $i = 2, \dots, \mathrm{n_{tg}} + 1$.

The amplification factor associated with these schemes is evaluated by means of the Padé approximations presented in Table 5.1. The first row in this table defines the amplification factor:

$$G(\xi, C, d, c, r) = \mathsf{R}_{\mathrm{n_{tg}},0}(z) \quad \text{with } z := \frac{\mathcal{K}(\xi, d) - \mathcal{A}(\xi, C) - r\mathcal{M}(\xi)}{\mathcal{M}(\xi)}.$$

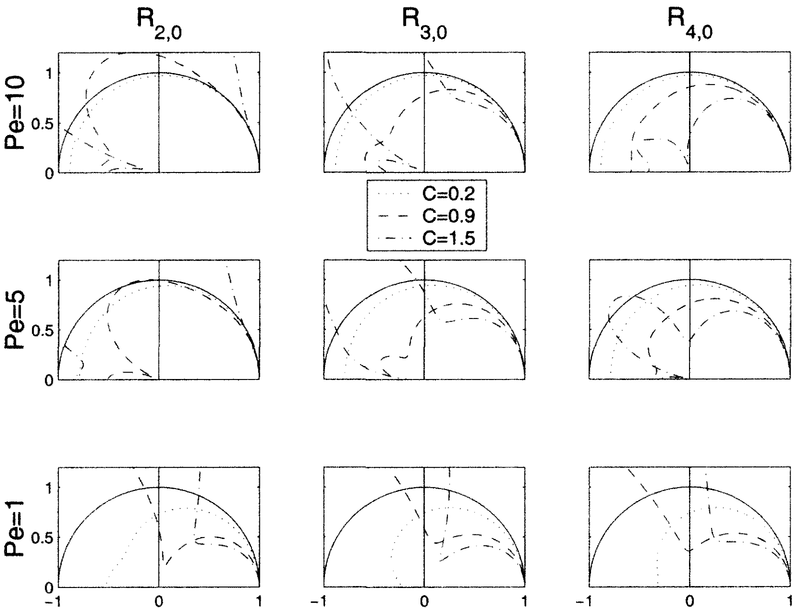


Fig. 5.6 Stability of multistage explicit Padé methods for different values of Péclet, *Pe*, and Courant, C, numbers.

These explicit methods are conditionally stable. Figure 5.1 shows the stability regions. Figure 5.6 presents the absolute value of the amplification factor in polar coordinates (ξ angle and $|G|$ radius) for different values of *Pe* and C. Note that diffusion strongly influences stability. For instance, the fourth-order method is stable for $C = 1.5$ at high Péclet numbers but becomes unstable as *Pe* decreases. This generates serious difficulties when explicit multistage methods are implemented in problems where convection is dominant in a region while diffusion dominates another, or simply when non-uniform meshes are needed due to geometrical requirements. From an accuracy point of view the same conclusions of Chapter 3 can be drawn here: higher-order methods are more accurate, in particular, with respect to phase errors.

5.4.4 Galerkin formulation of implicit multistage schemes

The time discretized equations (5.9) represent a strong form that must be solved at each time step. Note that any Runge–Kutta method, see (5.7), can also be expressed in the same form (with vectors and matrices of dimension $\mathrm{n}_{\mathrm{tg}} + 1$). The weighted residual form is

$$\left(\boldsymbol{w}, \frac{\Delta \boldsymbol{u}}{\Delta t}\right) - \left(\boldsymbol{w}, \mathbf{W}\, \Delta \boldsymbol{u}_t\right) = \left(\boldsymbol{w}, \mathbf{w}\, u_t^n\right),$$

where $\boldsymbol{w} \in [\mathcal{V}]^{\mathrm{n}}$, $\mathrm{n} = \mathrm{n}_{\mathrm{tg}} - 1$, for implicit multistage Padé schemes and $\mathrm{n} = \mathrm{n}_{\mathrm{tg}} + 1$, for standard implicit Runge–Kutta methods. Recall that $[\mathcal{V}]^{\mathrm{n}}$ imposes homogeneous conditions along the Dirichlet boundary. The solution $\Delta\boldsymbol{u} \in [\mathcal{S}]^{\mathrm{n}}$ and verifies the Dirichlet boundary conditions.

In the case of a linear operator $\mathcal{L}$, see (5.5), with constant coefficients and a source term s, see (5.11), the previous weighted residual equation can be rewritten as

$$\begin{aligned}\left(w, \frac{\Delta\boldsymbol{u}}{\Delta t}\right) + c(\boldsymbol{a}; w, \mathbf{W}\Delta\boldsymbol{u}) + a(w, \mathbf{W}\Delta\boldsymbol{u}) + (w, \sigma\mathbf{W}\Delta\boldsymbol{u})& \\ = \left(w, \mathbf{w}\, s^n + \mathbf{W}\Delta\boldsymbol{s}\right) + \left(w, \mathbf{w}\, h^n + \mathbf{W}\Delta\boldsymbol{h}\right)& \\ - \Big(c(\boldsymbol{a}; w, \mathbf{w}\, u^n) + a(w, \mathbf{w}\, u^n) + (w, \sigma\mathbf{w}\, u^n)\Big)&.\end{aligned}$$

From this weak form and Table 3.1 the equation which modifies each Fourier component is readily obtained

$$\begin{aligned}\Big(\mathcal{M}(\xi)\mathbf{I} - \theta\big(\mathcal{K}(\xi, d) - \mathcal{A}(\xi, C) - r\mathcal{M}(\xi)\big)\mathbf{W}\Big)\Delta\boldsymbol{u}& \\ = \big(\mathcal{K}(\xi, d) - \mathcal{A}(\xi, C) - r\mathcal{M}(\xi)\big)u^n\mathbf{w}&.\end{aligned}$$

Now the unknown is a vector with $\mathrm{n} = \mathrm{n_{tg}} - 1$ components for implicit multistage schemes. The amplification is the relation between the solution at t^{n+1} and t^n at a given interior point. To determine the numerical amplification factor, given the definition of $\Delta\boldsymbol{u}$, see (5.10), the following steps are necessary:

1. Solve the linear system of equations for $\boldsymbol{z} = (z_1, z_2, \ldots, z_{\mathrm{n_{tg}}-1})^T$:

$$\begin{aligned}\Big(\mathcal{M}(\xi)\,\mathbf{I} - \big(\mathcal{K}(\xi, d) - \mathcal{A}(\xi, C) - r\mathcal{M}(\xi)\big)\mathbf{W}\Big)\boldsymbol{z}& \\ = \big(\mathcal{K}(\xi, d) - \mathcal{A}(\xi, C) - r\mathcal{M}(\xi)\big)\mathbf{w}&.\end{aligned}$$

2. And finally determine the amplification factor as

$$G(\xi, C, d, r) = 1 + z_1 + z_2 + \cdots + z_{\mathrm{n_{tg}}-1}.$$

Remark 5.7. This procedure to determine G is necessary for the stabilized methods shown next. In the case of a Galerkin formulation, similarly to the previous section, the Padé Table 5.1 allows us to directly compute the amplification factor. In this case implicit multistage schemes correspond to the diagonal of Table 5.1 and the amplification factor can be readily obtained as

$$G(\xi, C, d, c, r) = \mathsf{R}_{\mathrm{n_{tg}}-1,\mathrm{n_{tg}}-1}(z) \quad \text{with } z := \frac{\mathcal{K}(\xi, d) - \mathcal{A}(\xi, C) - r\mathcal{M}(\xi)}{\mathcal{M}(\xi)}.$$

These methods are unconditionally stable. They allow us to use high-order time-stepping schemes. Thus, accuracy can be improved drastically (obviously, for the solvable frequencies $0 \leq \xi \leq \pi/4$). Figure 5.7 shows relative errors in amplitude and phase for the fourth-order method, $\mathsf{R}_{2,2}$. But as previously observed for the θ family of methods, see Section 5.4.2, a Galerkin formulation presents spatial instabilities in the presence of boundary layers. The problem stated in (5.17) is solved again with a uniform mesh of 10 linear elements, see Figure 5.8. Since the $\mathsf{R}_{2,2}$ method has twice as many nodal unknowns, the Courant number is increased to $C = 3$.

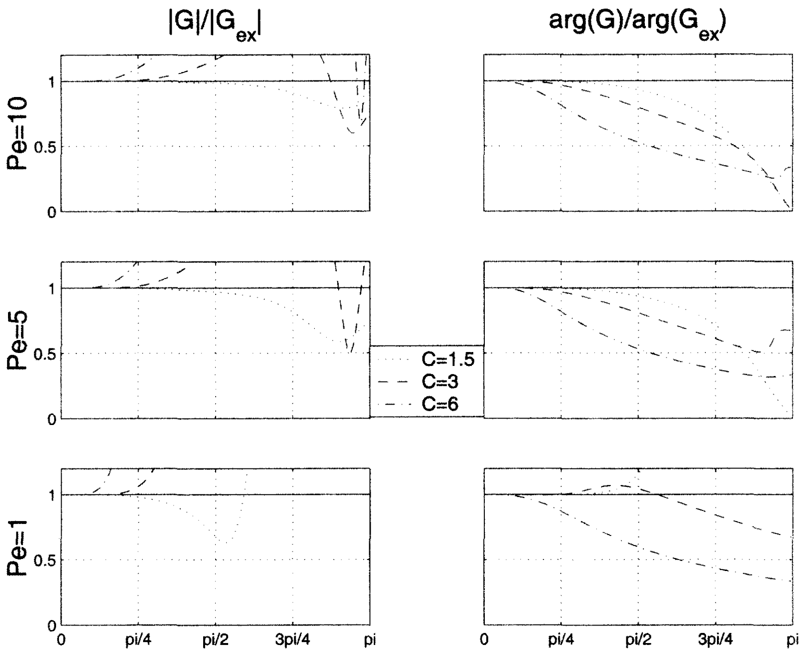


Fig. 5.7 Accuracy properties of implicit multistage $R_{2,2}$ at different values of Pe and C.

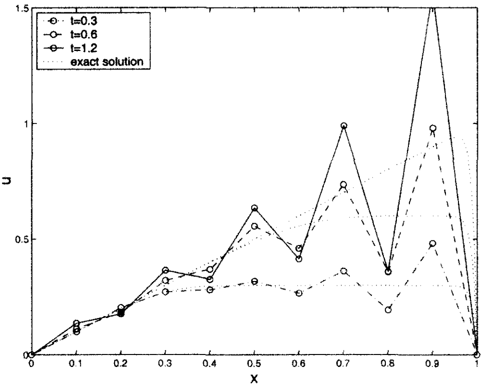


Fig. 5.8 Model problem for a Galerkin formulation of implicit multistage $R_{2,2}$.

5.4.5 Stabilization of the semi-discrete scheme

As amply discussed the Galerkin formulation lacks sufficient spatial stability when convective effects are important in the presence of boundary layers. In order to stabilize the convective term in a consistent manner, that is ensuring that the solution

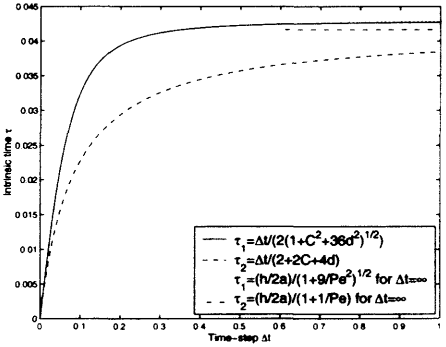


Fig. 5.9 Evolution of τ as Δt varies for the particular case: $a = 1, \nu = 10^{-2}$ and $h = 1/10$.

of the differential equation is also a solution of the weak form, Hughes and coworkers have proposed several techniques described in Chapter 2. These methods were subsequently extended to transient problems integrated with second-order time schemes or treated using space–time formulations.

A stabilized formulation of the convection–diffusion–reaction problem (5.1) can be stated as follows:

$$\big(w, u_t + \boldsymbol{a}\cdot\boldsymbol{\nabla}u + \sigma u\big) + a(w,u) + \sum_{e=1}^{n_{el}}\big(\mathcal{P}(w), \tau\mathcal{R}(u)\big)_{\Omega^e} = \big(w, s\big) + \big(w, h\big)_{\Gamma_N},$$

where the perturbation operator $\mathcal{P}$ characterizes the stabilization method, see Section 2.4. The stabilization term involves the residual $\mathcal{R}(u) = u_t + \boldsymbol{a}\cdot\boldsymbol{\nabla}u + \sigma u - \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla}u) - s$ of the governing equation, thus giving in principle a consistent formulation. Note that the residual includes the time derivative u_t of the unknown. This will result in a rather cumbersome implementation, except for the θ family of methods where u_t is replaced by $(u^{n+1} - u^n)/\Delta t$, or for space–time formulations.

Remark 5.8 (The transient stabilization parameter). When we are concerned by the steady solution it is usual to implement the parameters presented in Section 2.4.3. But if the transient solution is of interest we use the convection–diffusion–reaction extensions of the parameters described in Section 4.4.2.3. That is, for the θ family of methods, $\theta\ \in]0,1]$, following Shakib et al. (1991)

$$\tau = \left(\left(\frac{1}{\theta\Delta t}\right)^2 + \left(\frac{2a}{h}\right)^2 + 9\left(\frac{4\nu}{h^2}\right)^2 + \sigma^2\right)^{-1/2} = \frac{\Delta t}{2}\left(\left(\frac{1}{2\theta}\right)^2 + C^2 + 36d^2 + \left(\frac{r}{2}\right)^2\right)^{-1/2}$$

or from Soulaïmani and Fortin (1994), see also Codina (2000),

$$\tau = \left(\frac{1}{\theta\Delta t} + \frac{2a}{h} + \frac{4\nu}{h^2} + \sigma\right)^{-1} = \frac{\Delta t}{2}\left(\frac{1}{2\theta} + C + 2d + \frac{r}{2}\right)^{-1}.$$

Figure 5.9 shows, for $\theta = 1/2$, the variation of τ with Δt in a particular case. Note that in the limit $\Delta t = \infty$ both definitions are similar (the limits coincide in pure convection), but one definition reaches the limit much more rapidly.

5.4.6 Stabilization of multistage schemes

When accuracy requires us to go beyond second-order schemes and higher-order time integration schemes are employed, the standard stabilization of the semi-discrete equations is not trivial. In the context of transient problems, see Chapters 3 and 4, least-squares formulations can also be used to stabilize pure convection problems, see the appendix at the end of this chapter. However, their direct extension to convection–diffusion problems is also not trivial because of the presence of the second-order operator (the diffusion term). Huerta and Donea (2002) propose to combine spatial stabilization techniques with high-order implicit time-stepping schemes. Moreover, a least-squares (LS) formulation is also presented for these high-order time schemes combined with C^0 finite elements (in spite of the diffusion operator and without reducing the strong form to a system of first-order differential equations). Reference will be made to three stabilization techniques, namely, the *streamline-upwind Petrov–Galerkin* (SUPG), the *Galerkin/Least-squares* (GLS), and the *sub-grid scale* (SGS) methods, see Section 2.4.

In order to have a consistent stabilization of time discretized equations, such as those resulting from multistage time schemes, a residual must be defined. The residual, in the present case, is clearly chosen after time discretization. Thus, from the strong form (5.9) one gets

$$\mathcal{R}(\Delta \boldsymbol{u}) := \frac{\Delta \boldsymbol{u}}{\Delta t} - \mathbf{W}\,\Delta \boldsymbol{u}_t - \mathbf{w}\,u_t^n.$$

In order to better visualize the spatial differential structure of the residual $\mathcal{R}(\Delta \boldsymbol{u})$, this equation is rewritten for the particular case of linear spatial operator $\mathcal{L}$ with constant coefficients, that is from (5.11),

$$\mathcal{R}(\Delta \boldsymbol{u}) = \frac{\Delta \boldsymbol{u}}{\Delta t} + \mathbf{W}\mathcal{L}(\Delta \boldsymbol{u}) - \mathbf{w}\big[s^n - \mathcal{L}(u^n)\big] - \mathbf{W}\Delta \boldsymbol{s}, \tag{5.18}$$

where, from the second term on the r.h.s., one can observe that the operator $\mathcal{L} = \boldsymbol{a}\cdot\boldsymbol{\nabla} - \nu\boldsymbol{\nabla}^2 + \sigma$ acts on each component of $\Delta \boldsymbol{u}$.

The consistently stabilized weak form of the time discretized problem is given by

$$\Big(\boldsymbol{w}, \frac{\Delta \boldsymbol{u}}{\Delta t}\Big) - \big(\boldsymbol{w}, \mathbf{W}\,\Delta \boldsymbol{u}_t\big) + \underbrace{\sum_e \big(\boldsymbol{\tau}\mathcal{P}(\boldsymbol{w}), \mathcal{R}(\Delta \boldsymbol{u})\big)_{\Omega^e}}_{\text{Stabilization term}} = \big(\boldsymbol{w}, \mathbf{w}\,u_t^n\big). \tag{5.19}$$

Here again, the stabilization term is added to the Galerkin weak form. It contains the intrinsic time scale matrix $\boldsymbol{\tau}$ because (5.11), or (5.18), is a system of equations. The operator $\mathcal{P}$ characterizes the stabilization technique (i.e., SUPG, GLS, SGS or LS).

Remark 5.9 (On the stabilization matrix $\boldsymbol{\tau}$ for multistage schemes). When the convection–diffusion–reaction operator $\mathcal{L} = \boldsymbol{a}\cdot\boldsymbol{\nabla} - \nu\boldsymbol{\nabla}^2 + \sigma$ is replaced in the multistage form of the time-discretized equation, that is in (5.11), the system of equations that must be solved at each time step, namely

$$\begin{aligned}\frac{\Delta \boldsymbol{u}}{\Delta t} &+ \mathbf{W}(\boldsymbol{a}\cdot\boldsymbol{\nabla}\Delta\boldsymbol{u} - \nu\boldsymbol{\nabla}^2\Delta\boldsymbol{u} + \sigma\Delta\boldsymbol{u}) \\ &= \mathbf{w}\big[s^n - (\boldsymbol{a}\cdot\boldsymbol{\nabla}u^n - \nu\boldsymbol{\nabla}^2u^n + \sigma u^n)\big] + \mathbf{W}\Delta\boldsymbol{s},\end{aligned}$$

can be reinterpreted as a system of *steady* convection–diffusion–reaction equations.

In this case, convection is controlled by $a_i\mathbf{W}$, $i = 1, \ldots, \mathsf{n_{sd}}$ (recall the Jacobian matrices in Section 4.4.2.3), the diffusion matrix is $\nu\mathbf{W}$ and the reaction matrix, $\mathbf{I}/\Delta t + \sigma\mathbf{W}$, includes the contribution form the transient term. Thus, in 1D, the stabilization matrix can be computed as

$$\begin{aligned}\boldsymbol{\tau} &= \left[\frac{\mathbf{W}^{-1}}{\Delta t} + \left(\frac{2a}{h} + \frac{4\nu}{h^2} + \sigma\right)\mathbf{I}\right]^{-T}\mathbf{W}^{-1} \\ &= \Delta t\big[\mathbf{W}^{-1} + (2C + 4d + r)\mathbf{I}\big]^{-T}\mathbf{W}^{-1},\end{aligned}$$

when the analysis of Soulaïmani and Fortin (1994), see also Codina (2000), is particularized in this case. Note also that such a definition of the intrinsic matrix $\boldsymbol{\tau}$ is also valid for the scalar case and generalizes the previous one, see Remark 5.8.

In particular, for implicit Padé approximations, see the definitions of $\mathbf{W}$ in Section 5.3.3.5, $\mathbf{W}^{-1} = \frac{1}{2}\left(\begin{smallmatrix} 5 & 1 \\ -13 & 7\end{smallmatrix}\right)$ for $\mathsf{R}_{2,2}$.

Finally, note that other definitions for $\boldsymbol{\tau}$ can also be used. For instance, if the steady solution is our main interest, as noted in Remark 5.8, it is usual to implement the parameters presented in Section 2.4.3, that is

$$\boldsymbol{\tau} = \frac{h}{2a}\left[\coth(Pe) - \frac{1}{Pe}\right]\mathbf{W}^{-1},$$

see the final example in the appendix at the end of this chapter.

Remark 5.10 (Consistent stabilization). The stabilization term involves the residual, which includes the second-order term $\nabla^2 u$. When linear finite elements are used this term vanishes or is largely under-represented, with the corresponding degradation in the consistency of the stabilized formulation. The lack of consistency leads to errors of order $\mathcal{O}(\tau)$, apart from the errors inherent in the time integration scheme.

In order to keep the convergence rates in time, several possibilities can be useful. The stabilization parameter τ can be defined to be asymptotically of order $\mathcal{O}(\Delta t^{2\mathrm{n}})$. That is, a specific intrinsic time τ should be designed for each one of the time integration schemes. Another possibility is to include flux jump terms across the element boundaries in the stabilized formulation to take into account the neglected terms, see Tezduyar and Osawa (2000) for details. In fact, Jansen, Collis, Whiting and Shakib (1999) show that when linear finite elements are used the lack of consistency due to the neglected terms leads also

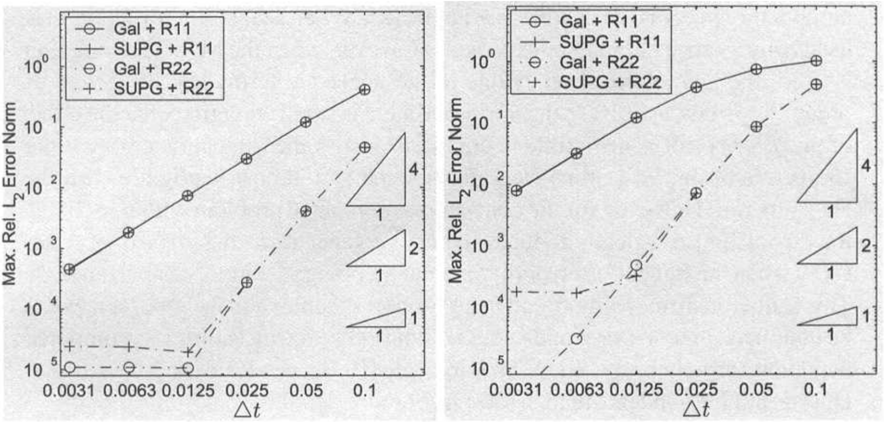


Fig. 5.10 Convergence results: $h = 0.001$, $a = 1$, $\nu = 10^{-2}$ (left) and $\nu = 10^{-4}$ (right).

to reduced convergence in space. For linear finite elements, they propose a global reconstruction of second derivatives. This method recovers the ability to approximate the residual in the stabilization term yielding a better consistency, through an iterative process. However, the increase in the computational cost is not negligible: a system of equations with global mass matrix must be solved at each iteration.

The use of high-order finite elements, such as quadratic elements, allows the inclusion of second derivatives of the approximation in the residual in the stabilization term, and thus consistent stabilized formulations can be defined. However, the computational cost and the implementation difficulties are highly increased due to the computation of second derivatives of the element mapping (see Jansen et al., 1999).

A numerical example is considered in order to experimentally assess the need for considering second derivatives in space. We solve the 1D convection–diffusion equation with constant coefficients,

$$u_t + au_x = \nu u_{xx}, \qquad (x,t) \in \,]0,2[\times]0,1[,$$

with homogeneous Dirichlet boundary conditions. The initial condition, at $t = 0$, is chosen such that the analytical solution is known

$$u(x,t) = \frac{\sigma_0}{\sigma} \exp\left\{ -\frac{\big(x - (x_0 + at)\big)^2}{2\sigma^2} \right\}, \qquad \sigma^2 = \sigma_0^2 + 2\nu t,$$

where $x_0 = 0.35$ and $\sigma_0 = 0.05$. Figure 5.10 shows the evolution of the error against the time step for linear finite elements with element size $h = 0.001$, and for two different values of the diffusion parameter ν. The error is evaluated in the $\mathcal{L}^2(\Omega)$ norm. Results are shown for Galerkin, `Gal`, and SUPG with Crank–Nicolson, $\mathsf{R}_{1,1}$, and a fourth-order Padé scheme, $\mathsf{R}_{2,2}$. In the stabilized computations second derivatives are neglected. When a Galerkin formulation is used convergence rates are as expected. However, when the SUPG formulation is used, the lack of consistency due to the neglected terms leads to errors of order $\mathcal{O}(\tau)$. For, $\nu = 10^{-2}$ the intrinsic time τ is small enough so that the effect of this $\mathcal{O}(\tau)$ error is negligible in comparison with the truncation errors of the Crank–Nicolson, $\mathsf{R}_{1,1}$, time-stepping scheme and almost negligible with the $\mathsf{R}_{2,2}$ scheme. However, for the convection-dominated problem with $\nu = 10^{-4}$, the error $\mathcal{O}(\tau)$ drastically reduces the convergence rate: the error is of order $\mathcal{O}(\tau)$ when the truncation errors of the time-stepping schemes are small enough. This is obviously more important in high-order schemes and its importance may be unnoticed in first- or second-order schemes. The intrinsic time τ is computed using the formula proposed by Shakib et al. (1991), see Remark 5.8. See also Huerta and Fernández-Méndez (2003) for more details.

5.4.6.1 Streamline-upwind Petrov–Galerkin stabilization The SUPG stabilization technique is defined by taking

$$\mathcal{P}(w) := \mathbf{W}(\boldsymbol{a} \cdot \boldsymbol{\nabla})w. \tag{5.20}$$

Note that matrix $\mathbf{W}$, which affects the convection term in (5.18), induces a *non-scalar* stabilization that is, each equation of the multistage time scheme is affected by different coefficients. The weak form for the SUPG method is obtained after substitution of the perturbation operator (5.20) in equation (5.19). The non-symmetric structure of the stabilization term induces some technical difficulties in the stability analysis of the SUPG method. This is avoided in the GLS stabilization technique, because it introduces a symmetric stabilization term in a consistent manner.

5.4.6.2 Galerkin/Least-squares stabilization The need for a previous time discretization is clear in this method. The GLS stabilization uses as perturbation operator $\mathcal{P}$ the spatial differential operator of the strong form, which in this case is affected by the time discretization. Note that the present approach differs from the standard space–time GLS formulation: no time derivatives appear in $\mathcal{P}$ and thus there is no need for multicorrector algorithms (see Shakib and Hughes, 1991; Codina, 1998). In the case of a linear convection–diffusion–reaction equation with constant coefficients, from (5.18) one deduces the operator $\mathcal{P}$, namely

$$\mathcal{P}(w) := \frac{w}{\Delta t} + \mathbf{W}\,\mathcal{L}(w). \tag{5.21}$$

From a practical point of view there is no major difference between SUPG and GLS methods. Note that both methods are not identical, as in steady problems, for convection–diffusion (no reaction) and with linear elements (the second-order derivatives are zero in the element interiors). Moreover, the qualitative influence of each term in the definition of $\mathcal{P}$, equation (5.21), may be interpreted as follows:

$$\mathcal{P}(w) = \frac{w}{\Delta t} + \mathbf{W}\,\mathcal{L}(w) = \underbrace{\frac{w}{\Delta t}}_{\text{Galerkin}} + \mathbf{W}\Big[\underbrace{(\boldsymbol{a} \cdot \boldsymbol{\nabla})w}_{\text{SUPG}} - \underbrace{\boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} w)}_{0} + \underbrace{\sigma w}_{\text{Galerkin}}\Big].$$

The first term is a Galerkin weighting, the second term corresponds to the SUPG stabilization, the third term is zero for linear elements, and the fourth term is also a Galerkin weighting. Thus, for linear elements and a constant positive reaction, GLS is SUPG with the Galerkin term weighted $1 + \tau(\mathrm{C}\sigma + 1/\Delta t)$ times more (with C a constant related to $\mathbf{W}$). This implies that the instabilities introduced by Galerkin are slightly amplified in GLS compared with SUPG. This minor problem of the GLS stabilization technique is overcome in the simplest version of the sub-grid scale (SGS) method, see the next section.

In the case of the nonlinear convection–diffusion–reaction equation it is helpful to define the quasi-linear operator related to (5.5), see for instance the discussion by Hauke and Hughes (1998). This quasi-linear operator is then used in (5.21). Note, finally, that the stabilization term is symmetric but the complete weak form, that is (5.19) with (5.21), is in general non-symmetric.

5.4.6.3 *Sub-grid scale stabilization* The simplest version of the SGS stabilization assumes that the perturbation operator $\mathcal{P}(\boldsymbol{w})$ is minus the adjoint of the operator used in the GLS stabilization, that is

$$\mathcal{P}(\boldsymbol{w}) := -\frac{\boldsymbol{w}}{\Delta t} - \mathbf{W}\,\mathcal{L}^*(\boldsymbol{w}) = -\frac{\boldsymbol{w}}{\Delta t} + \mathbf{W}\big[(\boldsymbol{a}\cdot\boldsymbol{\nabla} + \sigma)\boldsymbol{w} + \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla}\boldsymbol{w})\big], \quad (5.22)$$

where $\mathcal{L}^*$ is the adjoint operator of $\mathcal{L}$. Following the same rationale as in the previous section to qualitatively determine the influence of the Galerkin term in SGS compared with SUPG, one gets that in SGS the Galerkin term is weighted $1 - \tau(\mathrm{C}\sigma + 1/\Delta t)$ times more than in SUPG. Thus it has less influence than in SUPG and GLS.

5.4.6.4 *Least-squares stabilization* The classical implementation of a standard LS formulation for the convection–diffusion–reaction problem (5.1) requires us to work in $\mathcal{H}^2$, unless a mixed LS formulation is used (see Park and Liggett, 1990; Carey, Shen and McLay, 1998). In fact, the least-squares minimization of the square of the residual of the governing equation invariably includes second spatial derivatives, thus requiring continuity of the unknown itself and of its first derivative.

In this section, an alternative procedure is proposed, which allows the use of standard $\mathcal{C}^0$ finite element interpolation and test functions. Furthermore, the proposed method does not increase the number of nodal unknowns (this is the case for a mixed formulation that yields a system of first-order equations or for higher-order interpolation methods). A standard LS formulation directly uses the spatial strong form to construct the integral equation. Here, since time discretization is already performed, equations (5.9) or (5.11) are used. Consequently, one gets the integral form

$$\left(\frac{\boldsymbol{w}}{\Delta t} + \mathbf{W}\,\mathcal{L}(\boldsymbol{w}), \mathcal{R}(\Delta\boldsymbol{u})\right) = 0, \quad (5.23)$$

where $\boldsymbol{w}$ and $\Delta\boldsymbol{u}$ are in subspaces of $[\mathcal{H}^2]^{\mathrm{n_{tg}}-1}$. However, an "equivalent" form following the same rationale as for standard stabilized methods, see equation (5.19), can be devised. It is equivalent in the sense that its unique solution is also the unique solution of (5.1) and also the solution of (5.23). The first argument in (5.23) is split by linearity and the term containing $\mathcal{L}(\boldsymbol{w})$ is only evaluated on the element interiors, namely

$$(\boldsymbol{w}, \mathcal{R}(\Delta \boldsymbol{u})) + \sum_{e} (\Delta t\, \mathbf{W}\, \mathcal{L}(\boldsymbol{w}), \mathcal{R}(\Delta \boldsymbol{u}))_{\Omega^e} = 0. \tag{5.24}$$

Now the interpolation and test functions can be taken in a subspace of $[\mathcal{H}^{1+}]^{\mathrm{n_{tg}}-1}$,

$$\mathcal{H}^1 \subsetneq \mathcal{H}^{1+} := \{ w \in \mathcal{H}^1(\Omega) \mid w|_{\Omega^e} \in \mathcal{H}^2(\Omega^e) \text{ for all element } \Omega^e \} \subsetneq \mathcal{H}^2.$$

Thus, standard $\mathcal{C}^0$ finite elements can be used. This approach can also be viewed as a standard stabilization technique: the first term accounts for the Galerkin part of the weak form, and the second one introduces the desired stabilization with the following definition of the operator $\mathcal{P}(\boldsymbol{w})$ and the intrinsic time τ:

$$\tau := \Delta t \quad \text{and} \quad \mathcal{P}(\boldsymbol{w}) := \mathbf{W}\, \mathcal{L}(\boldsymbol{w}). \tag{5.25}$$

In fact, the GLS formulation so-called "neglecting the inertia term" (i.e., $\boldsymbol{w}/dt$ is neglected) is used in common practice. It corresponds to the same perturbation operator (previous equation) and uses the standard τ instead of Δt.

5.4.6.5 Accuracy properties The Fourier analysis described in Section 5.4.4 for the Galerkin formulation can be repeated for each stabilized technique. For brevity we only present the GLS case; Huerta, Roig and Donea (2002) show further details. The equation associated with the GLS stabilization technique for a typical Fourier mode corresponding to a dimensionless wave number ξ is

$$\begin{aligned}
&\Big\{ \mathcal{M}(\xi)\,\mathbf{I} - [\mathcal{K}(\xi,d) - \mathcal{A}(\xi,C) - r\mathcal{M}(\xi)]\,\mathbf{W} \\
&\quad - \frac{\tau}{\Delta t}\Big[-\mathcal{M}(\xi)\,\mathbf{I} + \mathcal{A}(\xi,C)(\mathbf{W}^T - \mathbf{W}) - r\mathcal{M}(\xi)(\mathbf{W}^T + \mathbf{W}) \\
&\qquad + (\mathcal{D}(\xi,C) - r^2\mathcal{M}(\xi))\,\mathbf{W}^T\mathbf{W} \Big] \Big\} \Delta \boldsymbol{u} \\
&\qquad = \Big\{ [\mathcal{K}(\xi,d) - \mathcal{A}(\xi,C) - r\mathcal{M}(\xi)] \\
&\qquad\qquad - \frac{\tau}{\Delta t}\Big[\mathcal{A}(\xi,C) + r\mathcal{M}(\xi) - (\mathcal{D}(\xi,C) - r^2\mathcal{M}(\xi))\,\mathbf{W}^T \Big] \Big\} u^n\, \mathbf{w}.
\end{aligned}$$

Note the nonlinear dependence on $\mathbf{W}$.

Recall that the intrinsic time scale can be defined in various ways, see Remark 5.8, although no major differences are observed for the usual definitions. Figures 5.11 and 5.12 show the relative errors for the numerical damping, $|G| \,/\, |G_{\text{ex}}|$, and the phase error, $\arg(G) \,/\, \arg(G_{\text{ex}})$, for different values of the Péclet number, $Pe = C/(2d)$. All the curves are plotted in polar coordinates, the angle is ξ and the radius is the relative error. From a precision point of view only the range $0 \le \xi \le \pi/4$ is of interest. Note that for Crank–Nicolson, Figure 5.11, the Courant numbers employed are 0.75, 1.5 and 3. For $\mathsf{R}_{2,2}$ larger values of the Courant number (1.5, 3 and 6) are used to

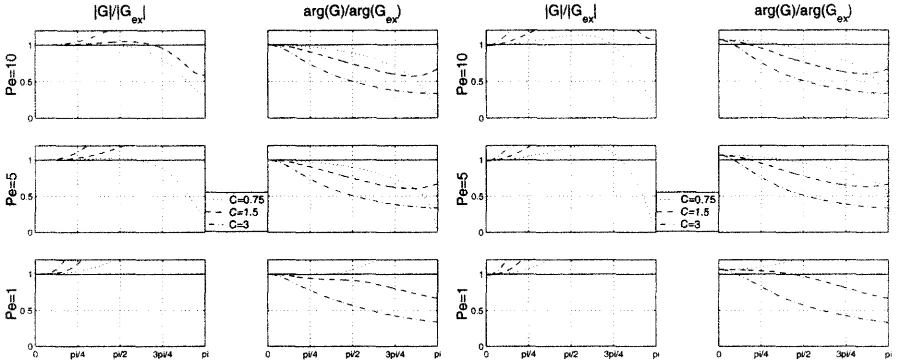


Fig. 5.11 Accuracy properties of Crank–Nicolson ($R_{1,1}$) with GLS stabilization: relative errors in amplitude and phase for $r = 0$ (left) and $r = 0.5$ (right) and different values of P_e.

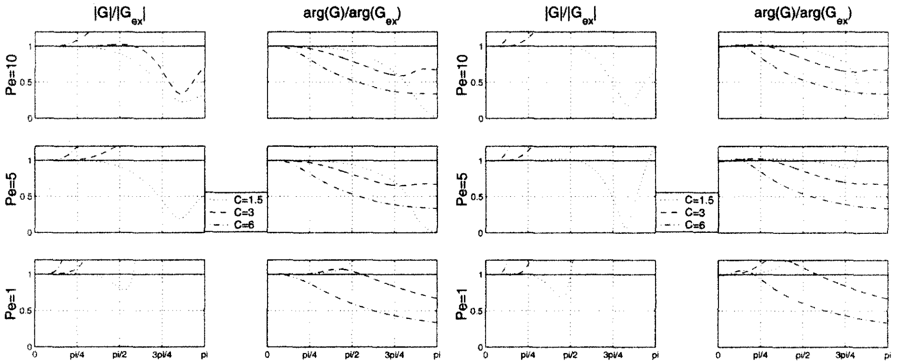


Fig. 5.12 Accuracy properties of $R_{2,2}$ with GLS stabilization: relative errors in amplitude and phase for $r = 0$ (left) and $r = 1$ (right) and different values of P_e.

highlight its range of precision which must compensate its extra cost (double number of unknowns per node).

Only the GLS stabilization results are shown, the other stabilization techniques present little differences. The LS formulation introduces, as expected, more numerical diffusion for $R_{2,2}$. Moreover, it is only noticeable in pure convection ($P_e = 0$) and this effect is beneficial in the presence of reaction. The phase accuracy is also affected by the LS stabilization but this only occurs for low Péclet numbers.

One can observe that, as expected, the stabilized methods introduce numerical damping for high frequencies (ξ close to π), which cannot be properly represented on the discrete mesh. Recall that Galerkin methods for pure convection do not introduce any numerical damping when combined with diagonal Padé approximations. Moreover, from an accuracy point of view (i.e., for dimensionless wave numbers such that $0 \leq \xi \leq \pi/4$), the response (for different Péclet, Courant and reaction values) is quantitatively the same as in the Galerkin formulation. Thus, the stabilization pro-

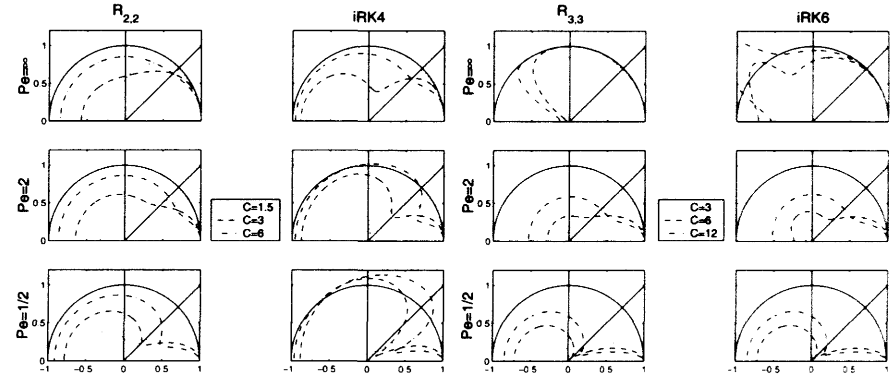


Fig. 5.13 Modulus of the amplification factor for stabilized Padé ($R_{n,n}$) and Lobatto IIIA implicit Runge–Kutta (iRKn) methods at different values of the Péclet number: fourth-order methods with LS (left) and sixth-order methods with GLS (right).

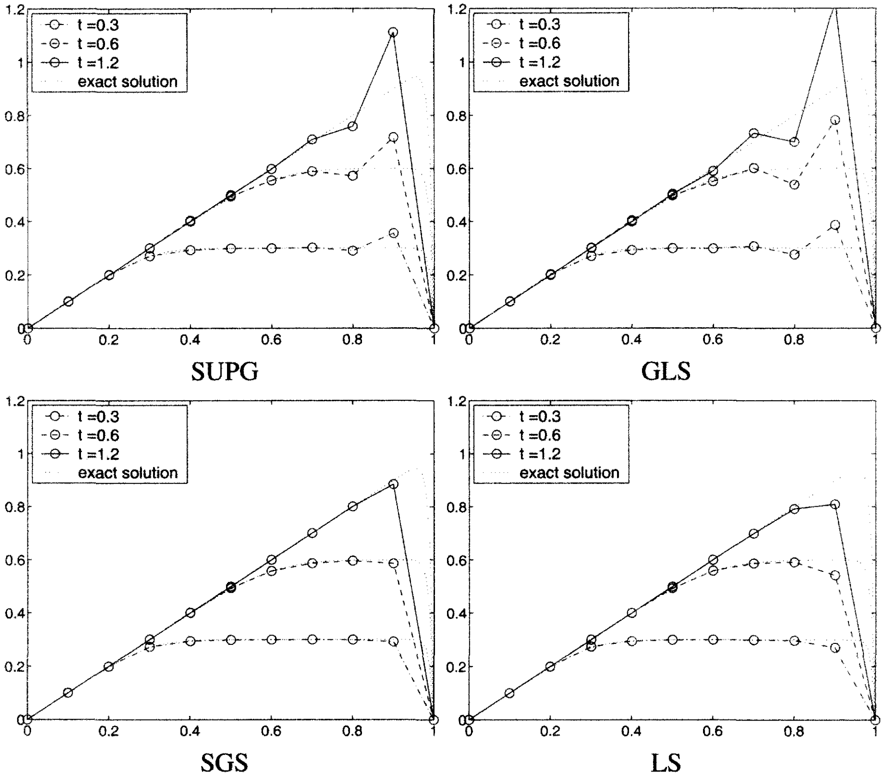
cedure introduces numerical diffusion where it is needed and does not compromise the accuracy. Moreover, from the point of view of the phase errors, the stabilized methods show a behavior identical to the Galerkin formulation. That is, phase errors are drastically reduced when the order of the temporal approximation is increased.

The reaction only influences the accuracy of the damping response; only for very large values of the reaction is the phase accuracy moderately degraded as r increases.

An important point to be made in concluding the accuracy analysis is that the use of non-overlapping increments in the multistage time scheme preserves unconditional stability when the method is combined with stabilized spatial formulations. By contrast, Figure 5.13 shows that the classical, fourth-order accurate, Lobatto IIIA implicit Runge–Kutta method combined with the LS stabilization loses stability for moderate to low Péclet numbers. It also shows that the sixth-order accurate Lobatto IIIA implicit Runge–Kutta method loses stability in pure convection when combined with the GLS stabilization. This is not the case for the corresponding multistage schemes derived from Padé approximations.

Now the stabilized formulation can be used in the model problem defined in (5.17) to preclude the usual global Galerkin instabilities. Figure 5.14 presents the results for Crank–Nicolson stabilized with SUPG, GLS, SGS and LS. In this case the Courant number is chosen equal to one because as seen in Figure 5.11 this value produces accurate results. Similarly, Figure 5.15 presents the results at the same instants for the stabilized implicit multistage Padé of fourth order, $R_{2,2}$. This scheme presents reasonable accuracy properties up to $C = 3$, see Figure 5.12. In this examples we use the definition of stabilization parameter for Crank–Nicolson proposed in Remark 5.8, and the definition of stabilization matrix for $R_{2,2}$ the given in Remark 5.9.

Note first that the solution is stabilized: oscillations, when they exist, are confined to the boundary layer and do not pollute the solution everywhere. Second, as already discussed in Section 2.4 (see Figure 2.14), the different stabilization (SUPG, GLS



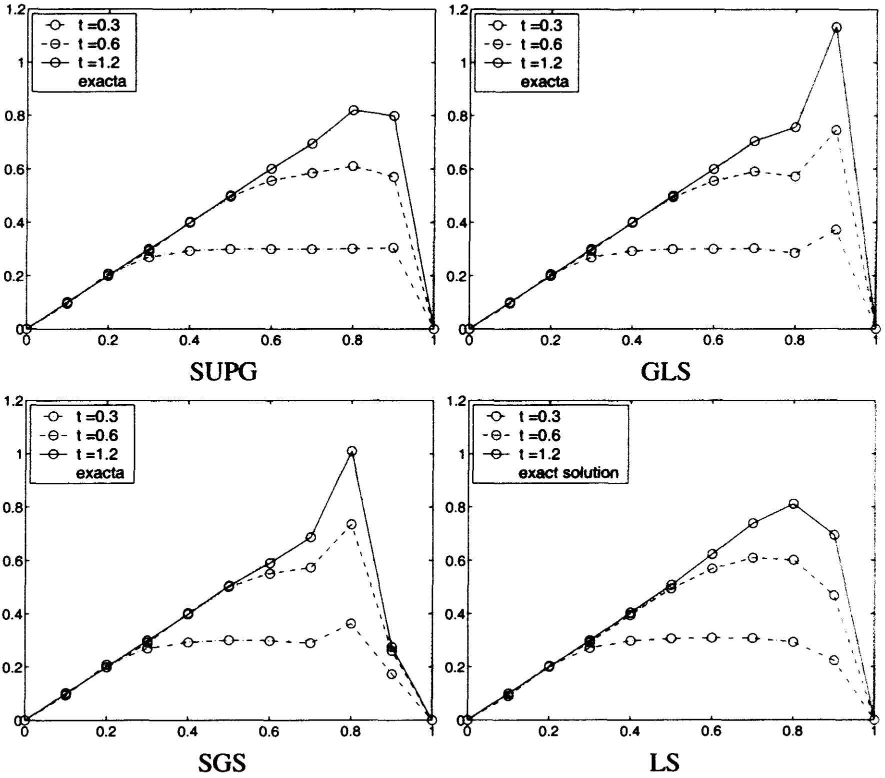
SUPG GLS SGS LS

Fig. 5.14 Model problem with stabilized Crank–Nicolson and $C = 1$.

and SGS) techniques weight more or less the Galerkin term and thus the oscillation at the boundary layer is more or less amplified. Finally, these examples also show that LS stabilization techniques are more diffusive than SUPG, GLS or SGS. Moreover, by definition, see (5.25), the added diffusion is controlled by the time step ($\tau = \Delta t$). Therefore, the LS solution introduces more or less diffusion depending on the imposed Courant number.

5.5 STABILIZED SPACE–TIME FORMULATIONS

We conclude the discussion of spatial discretization procedures mentioning that stabilized methods for transient convection–diffusion–reaction can easily be extended to the space–time domain. Methods in this class were described for pure convection in Section 3.10 and we follow the notation introduced there. To give an example in the present context we consider the space–time Galerkin/Least-squares formulation, see Section 3.10.3 and in particular equation (3.65). We assume homogeneous Dirichlet boundary conditions. Due to the presence of the diffusion operator, least-squares



SUPG

GLS

SGS

LS

Fig. 5.15 Model problem with stabilized $\mathsf{R}_{2,2}$ and $C = 3$.

terms are again acting in element interiors only and the weighted residual formulation becomes: for $n = 0, 1, \ldots, \mathrm{n_{st}} - 1$, find $u^h \in \mathcal{S}_n^h$ such that for all $w^h \in \mathcal{V}_n^h$

$$\begin{aligned}\iint_{Q^n} w^h(u_t^h + \boldsymbol{a}\cdot\boldsymbol{\nabla} u^h + \sigma u^h)\,d\Omega\,dt + \iint_{Q^n} \boldsymbol{\nabla} w^h \cdot \nu \boldsymbol{\nabla} u^h\,d\Omega dt \\ + \sum_{e=1}^{\mathrm{n_{el}}} \iint_{Q_e^n} \left(w_t^h + \boldsymbol{a}\cdot\boldsymbol{\nabla} w^h + \sigma w^h - \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla} w^h)\right) \\ \tau\left(u_t^h + \boldsymbol{a}\cdot\boldsymbol{\nabla} u^h + \sigma u^h - \boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla} u^h)\right) d\Omega\,dt \\ + \int_\Omega w^h(t_+^n)\left(u^h(t_+^n) - u^h(t_-^n)\right) d\Omega = 0, \quad (5.26)\end{aligned}$$

with $u^h(t_-^0) = u_0$. The last integral is the jump condition. The third integral is the least-squares operator and parameter τ is the least-squares metric. Shakib and Hughes (1991) perform a Fourier stability and accuracy analysis of the space–time GLS method for constant-in-time and linear-in-time approximations.

5.6 SOLVED EXERCISES

5.6.1 Convection–diffusion of a Gaussian hill

The problem consists of solving the homogeneous linear convection–diffusion equation on the 1D domain]0, 1[with the initial condition

$$u(x,0) = \frac{5}{7}\exp\left\{-\left(\frac{x-x_0}{\ell}\right)^2\right\},$$

where $x_0 = 2/15$ and $\ell = 7\sqrt{2}/300$, and with boundary conditions: $u(0,t) = 0$, and $u(1,t) = 0$. The grid Péclet number is first taken as $Pe = 1$ and then increased to $Pe = 5$ and finally to $Pe = 100$. The convection velocity is $a = 1$. The problem is solved using a uniform mesh of linear elements of size $h = 1/150$ until time $t = 0.6$. The exact solution is given by

$$u(x,t) = \frac{5}{7\sigma(t)}\exp\left\{-\left(\frac{x-x_0-a\,t}{\ell\sigma(t)}\right)^2\right\}, \quad \text{where } \sigma(t) = \sqrt{1+4\nu\, t/(\ell^2)}.$$

Galerkin and Crank–Nicolson. We first want to highlight the fact that linear finite elements in the standard Galerkin formulation do not ideally combine with the second-order Crank–Nicolson time-stepping method in highly convective situations. The results reported in Figure 5.16 for a Courant number $C = 1$ show that the second-order time scheme performs well at low and moderate values of the Péclet number, but exhibits significant phase errors when the Péclet number is further increased. Moreover, the situation becomes worse when the time-step size corresponds to a Courant number larger than one, see the results for $C = 1.5$ and $Pe = 100$. As shown by the next tests of the Gaussian hill, the situation improves very much when passing to third- and fourth-order accurate time-stepping algorithms.

Galerkin and $\mathbf{R}_{2,2}$. Higher-order methods in time provide a gain in accuracy. Figure 5.17 shows, for a Courant number $C = 3$, that the fourth-order time scheme performs well for all values of the Péclet number. Obviously, results degrade when the time step is too large, see the results for $C = 4$ and $Pe = 100$.

Time-discontinuous Galerkin. The time-discontinuous Galerkin formulation introduced, for pure convection, in Section 3.10.1, see also the examples in Section 3.11.4, is used here. Linear finite element approximations are employed in both space and time, giving a third-order accurate and unconditionally stable method. Adapting the developments in Section 3.10.1 (pure convection) to the present convection–diffusion case, the following partitioned matrix system is obtained for the nodal unknowns $\mathbf{u}^{n+1}$ and $\mathbf{u}^{n^+}$:

$$\begin{aligned}
\left(\mathbf{M} + \frac{2}{3}\Delta t\mathbf{C} + \frac{2}{3}\nu\Delta t\mathbf{K}\right)\mathbf{u}^{n+1} - \left(\mathbf{M} - \frac{1}{3}\Delta t\mathbf{C} - \frac{1}{3}\nu\Delta t\mathbf{K}\right)\mathbf{u}^{n^+} &= 0 \\
\left(\mathbf{M} + \frac{1}{3}\Delta t\mathbf{C} + \frac{1}{3}\nu\Delta t\mathbf{K}\right)\mathbf{u}^{n+1} + \left(\mathbf{M} + \frac{2}{3}\Delta t\mathbf{C} + \frac{2}{3}\nu\Delta t\mathbf{K}\right)\mathbf{u}^{n^+} &= 2\mathbf{M}\,\mathbf{u}^{n^-}.
\end{aligned}$$

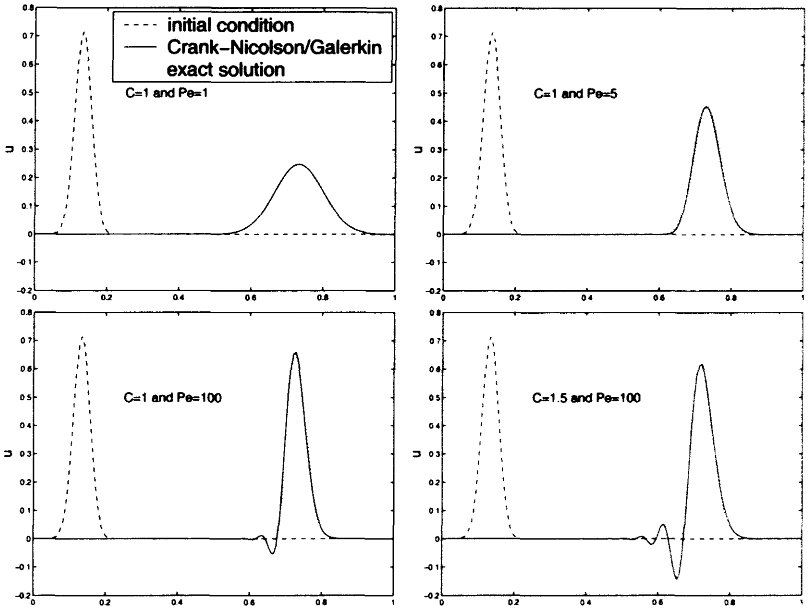


Fig. 5.16 Gaussian hill: standard Galerkin and Crank–Nicolson.

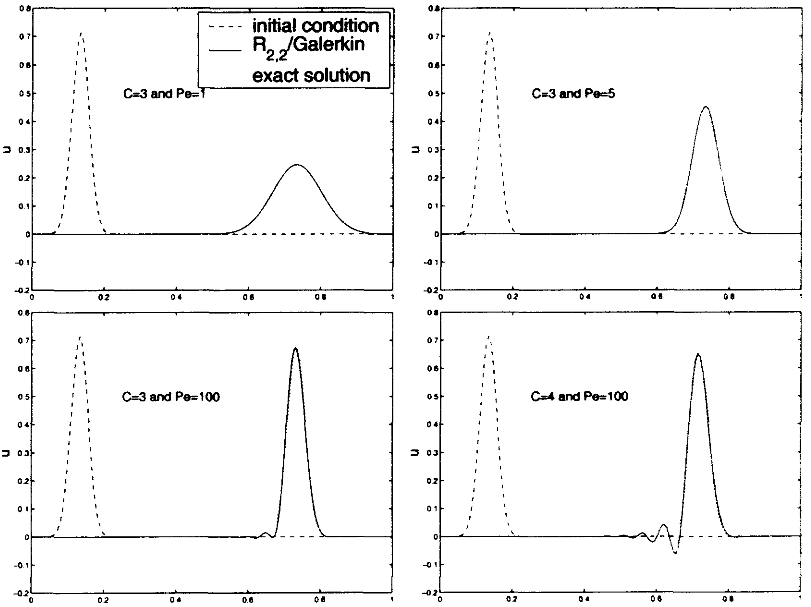


Fig. 5.17 Gaussian hill: standard Galerkin and $R_{2,2}$.

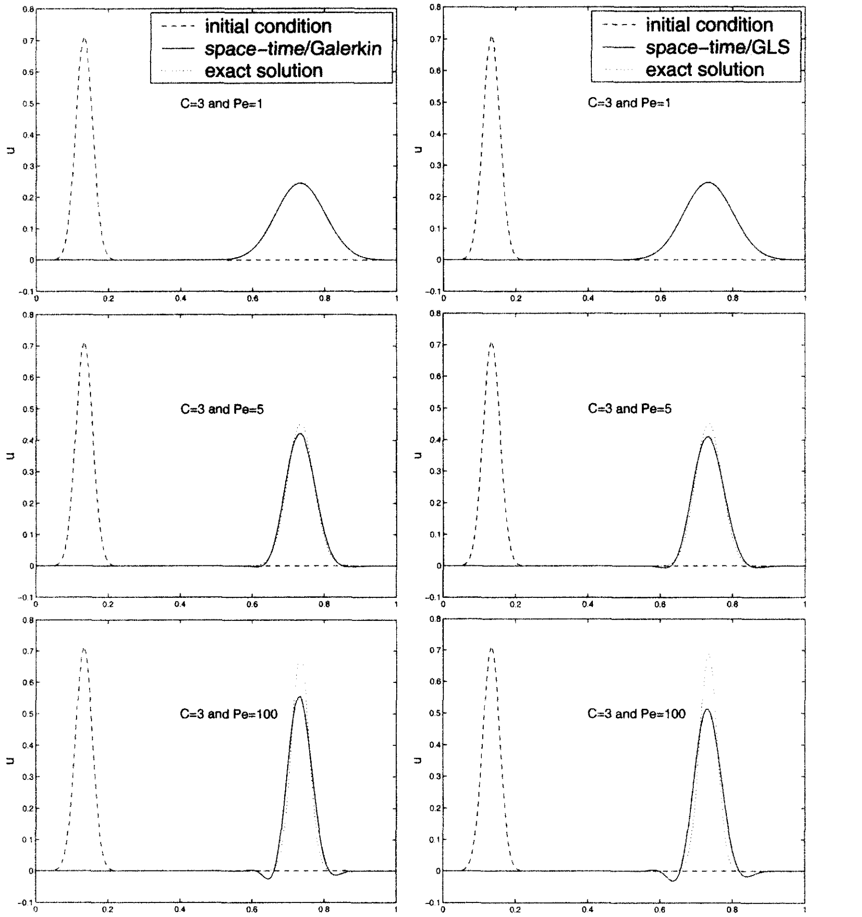


Fig. 5.18 Gaussian hill: space–time Galerkin (left) and Galerkin/Least-squares (right).

The matrices **M** (consistent mass), **C** (convection), and **K** (diffusion) are defined in (5.14). The conditions $u^{n+1}(0) = u^{n^+}(0) = 0$ and $u^{n+1}(1) = u^{n^+}(1) = 0$ are enforced to satisfy the boundary specifications.

Large time steps, corresponding to a Courant number $C = 3$, are used to test the capacity of this unconditionally stable scheme to deliver accurate results well beyond the stability limit of explicit methods for an equivalent computational cost (there are two unknowns per node). Figure 5.18 shows the results at $t = 0.6$ for the Galerkin formulation and three values of the Péclet number, $Pe = 1$, 5 and 100. Note that phase accuracy has improved with respect to the Crank–Nicolson results in Figure 5.16.

Time-discontinuous Galerkin/Least-squares. The formulation described in Section 5.5 is used. The partitioned matrix system for the nodal unknowns $\mathbf{u}^{n+1}$ and

$\mathbf{u}^{n^+}$ is now obtained in the form

$$\begin{aligned}
&\left(\left(1+\frac{2\tau}{\Delta t}\right)\mathbf{M}+\frac{2}{3}\Delta t\mathbf{C}+\frac{2}{3}(\tau a^2+\nu)\Delta t\mathbf{K}\right)\mathbf{u}^{n+1}\\
&\qquad-\left(\left(1+\frac{2\tau}{\Delta t}\right)\mathbf{M}-\left(\frac{1}{3}\Delta t+2\tau\right)\mathbf{C}-\frac{1}{3}(\tau a^2+\nu)\Delta t\mathbf{K}\right)\mathbf{u}^{n^+}=0\\
&\left(\left(1-\frac{2\tau}{\Delta t}\right)\mathbf{M}+\left(\frac{1}{3}\Delta t-2\tau\right)\mathbf{C}+\frac{1}{3}(\tau a^2+\nu)\Delta t\mathbf{K}\right)\mathbf{u}^{n+1}\\
&\qquad+\left(\left(1+\frac{2\tau}{\Delta t}\right)\mathbf{M}+\frac{2}{3}\Delta t\mathbf{C}+\frac{2}{3}(\tau a^2+\nu)\Delta t\mathbf{K}\right)\mathbf{u}^{n^+}=2\mathbf{M}\,\mathbf{u}^{n^-}.
\end{aligned}$$

Remark 5.8 presents two expressions for the stabilization parameter τ. The first one is used here but other definitions can be employed. Numerical results at $t = 0.6$ are displayed in Figure 5.18. They confirm that the Galerkin/Least-squares space–time method can be operated with time increments much larger than the stability limit of explicit schemes. Nevertheless, the numerical results indicate that space–time methods exhibit a rather elevated numerical damping in smooth convection-dominated problems when a large value of the Courant number is employed. This is in agreement with the fact that τ increases with the time-step size.

5.6.2 Transient rotating pulse

In this 2D example both problems of the transient convection–diffusion equation are present: (1) accurate transport of the unknown is needed and (2) boundary layers appear in the solution due to the Dirichlet boundary conditions. Therefore, high-order time-stepping schemes and stabilized formulations are needed in order to obtain an accurate solution. The 2D convection–diffusion problem

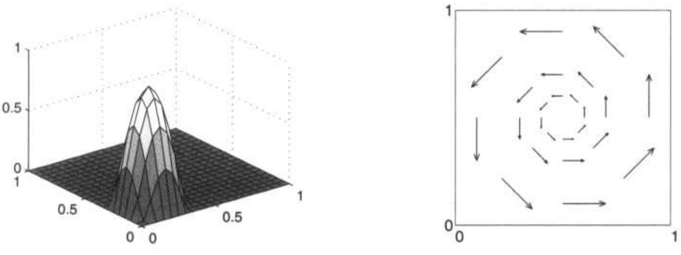
$$\begin{cases}
u_t+\boldsymbol{a}\cdot\boldsymbol{\nabla}u-\boldsymbol{\nabla}\cdot(\nu\boldsymbol{\nabla}u)=s & \text{in } \Omega=]0,1[\times]0,1[\\
u=0 & \text{on } \partial\Omega\\
u=0 & \text{at } t=0
\end{cases}$$

is solved with small diffusion, $\nu = 10^{-5}$, on a uniform 40 × 40 bilinear mesh. The source term and the velocity field are defined as follows (see also Figure 5.19):

$$\boldsymbol{a}=(-y,x)\qquad s=\mathrm{e}^{-t^{10}}\begin{cases}\cos(\pi/2\sqrt{x^2+y^2}) & \text{if } \sqrt{x^2+y^2}\le 1,\\ 0 & \text{otherwise.}\end{cases}$$

Figures 5.20 to 5.24 show the numerical results obtained at $t = \pi$ and $t = 5\pi$ for the Galerkin, SUPG and LS methods. Crank–Nicolson and $\mathsf{R}_{2,2}$ time integration schemes have been used with Courant 1 and 3, respectively.

Boundary layers are present in the solution due to the convective character of the equation and the homogeneous Dirichlet boundary conditions. Thus, the typical instabilities of the Galerkin formulation soon appear. The numerical solution is clearly improved for the stabilized formulations: oscillations are alleviated and almost



Source term s Convection velocity $\mathbf{a}$

Fig. 5.19 Source term and convection velocity

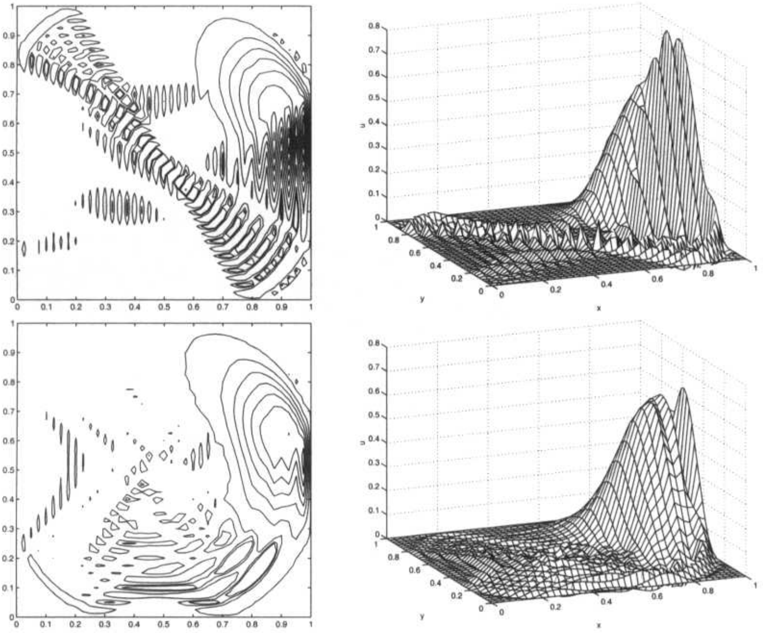


Fig. 5.20 Galerkin with Crank–Nicolson results at $t = \pi$ (top) and $t = 5\pi$ (bottom).

suppressed in the whole domain. However, important phase and amplitude errors can be observed in the numerical solution obtained for the Crank–Nicolson time-stepping scheme. Note that Crank–Nicolson results present negative values, with no physical sense, lagging behind the pulse.

It is also noticeable that both stabilization techniques produce similar results. SUPG presents more oscillations at the boundary layer. But results can easily be improved with the corresponding element-by-element definition of τ. Here the first formula presented in Remark 5.8 is employed. The least-squares results diffuse the spurious oscillations more rapidly but can easily become overly diffusive if Δt or the element size are increased.

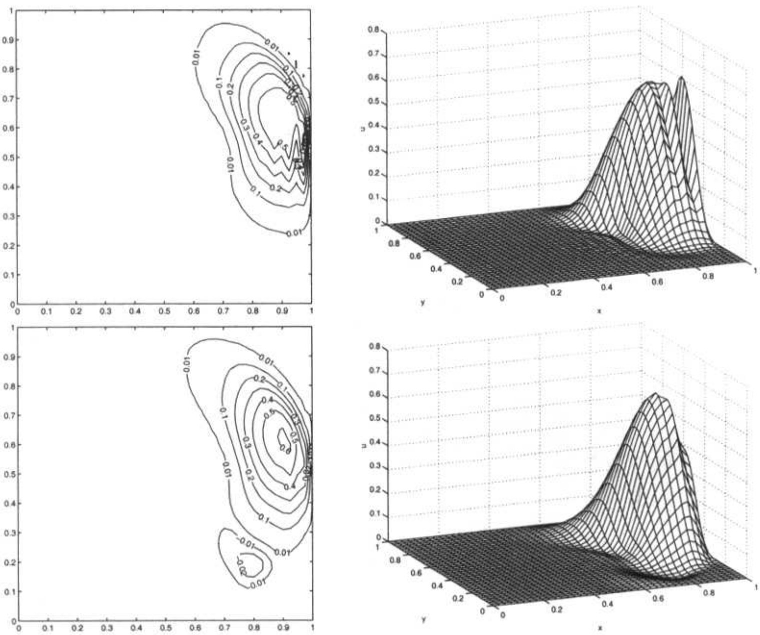


Fig. 5.21 SUPG with Crank–Nicolson results at $t = \pi$ (top) and $t = 5\pi$ (bottom).

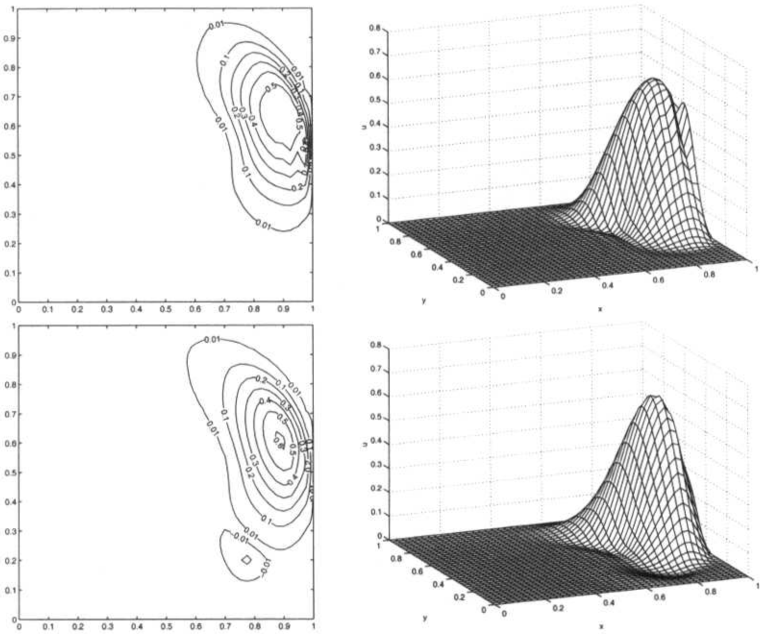


Fig. 5.22 Least-squares with Crank–Nicolson results at $t = \pi$ (top) and $t = 5\pi$ (bottom).

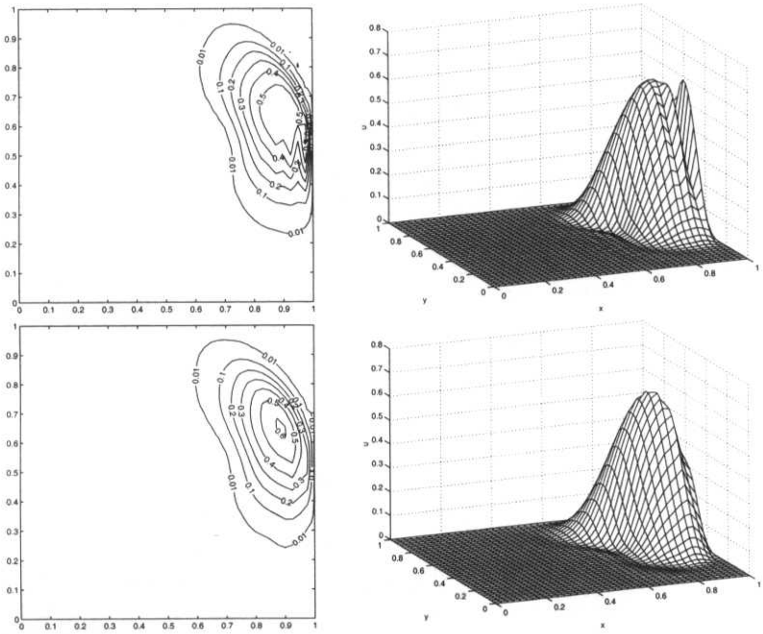


Fig. 5.23 SUPG with $\mathsf{R}_{2,2}$ results at $t = \pi$ (top) and $t = 5\pi$ (bottom).

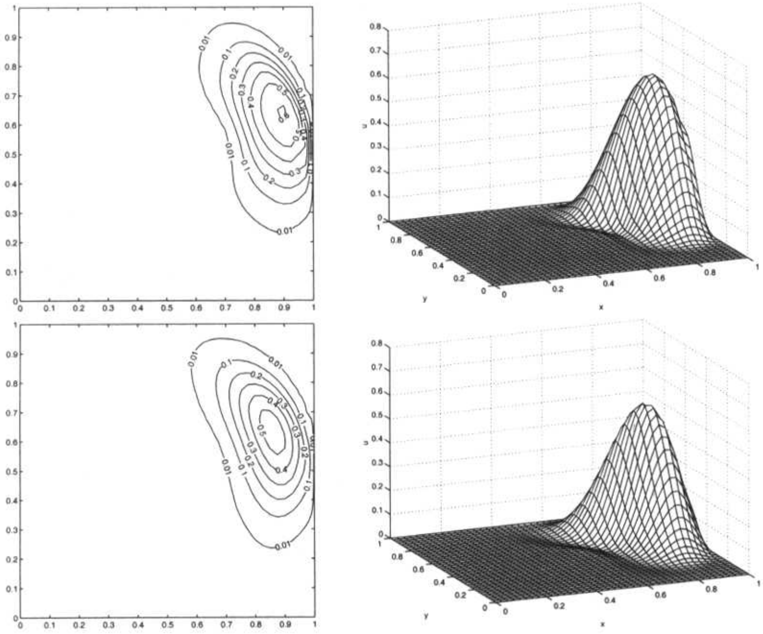


Fig. 5.24 Least-squares with $\mathsf{R}_{2,2}$ results at $t = \pi$ (top) and $t = 5\pi$ (bottom).

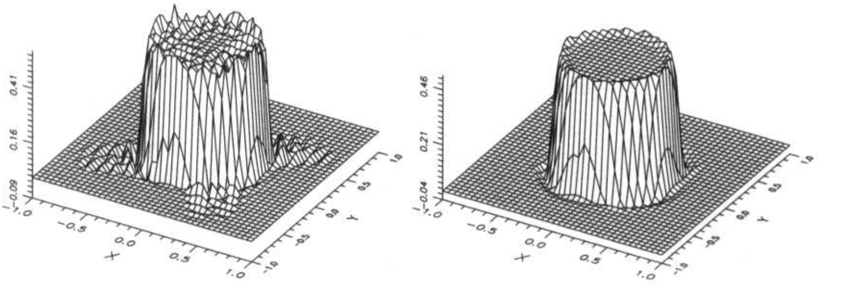


Fig. 5.25 Steady-state solution of the rotating pulse problem using $\mathsf{R}_{2,2}$ with $C = 2$ with Galerkin (left) and least-squares (right) formulations.

5.6.3 Steady rotating pulse problem

This example illustrates the ability of the stabilized Padé schemes discussed in Section 5.4.6 to accurately solve problems with internal boundary layers. The linear convection–diffusion–reaction equation (5.1a) is solved with $\mathsf{R}_{2,2}$ in the square domain $\Omega =]-1,1[\times]-1,1[$ with homogeneous Dirichlet boundary conditions and the following definitions: $\sigma = 2$,

$$a = \phi(\varrho)\begin{pmatrix} -y \\ x \end{pmatrix}, \phi(r) = \begin{cases} 1 - \varrho^2 & \text{if } \varrho \leq 1, \\ 0 & \text{otherwise,} \end{cases} \quad \text{and } s = \begin{cases} 1 & \text{if } \varrho \leq 1/2, \\ 0 & \text{otherwise,} \end{cases}$$

where $\varrho = \sqrt{x^2 + y^2}$. The asymptotic steady-state solution presents a clear pattern with $u \approx 1/2$ if $\varrho \leq 1/2$ and $u \approx 0$ otherwise; the boundary layer is $\mathcal{O}(\sqrt{\nu})$ along the circle $\varrho = 1/2$. A uniform mesh of 40 × 40 bilinear elements is used. Figure 5.25 shows the results obtained with the Courant number $C = 2$ for a Péclet number of 50000. While the Galerkin formulation fails to deliver stable results, one notes that the least-squares stabilization succeeds in eliminating the spurious oscillations characteristic of the Galerkin approach. Nevertheless, some oscillations remain near sharp solution gradients. Nonlinear viscosity of the type discussed in Chapter 4 (shock-capturing schemes) should be locally added to suppress such residual oscillations. Results for other stabilized methods or for higher-order time-stepping schemes ($\mathsf{R}_{3,3}$) present negligible differences.

5.6.4 Nonlinear propagation of a step

This example studies the influence of the numerical time accuracy in the propagation of a vertical front over a nonuniform mesh. Crank–Nicolson and $\mathsf{R}_{2,2}$ are compared and Burgers' equation,

$$u_t + u\, u_x = \nu\, u_{xx},$$

is solved in $]0,1[$ with a Dirichlet condition $u(0,t) = 1$ and a homogeneous Neumann condition of zero total flux at $x = 1$. Note that now we are confronted with a nonlinear

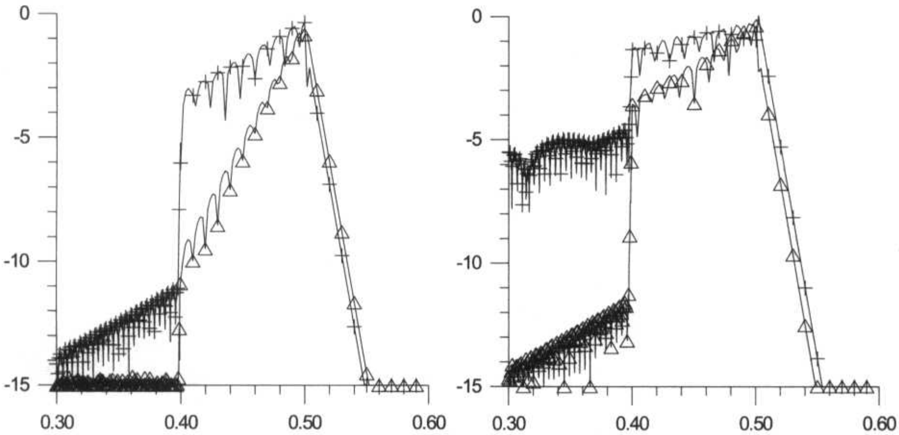


Fig. 5.26 Logarithm of the absolute error at time $t = 1$ for the LS (left) and GLS (right) stabilization. Crank–Nicolson is used with $C = 0.75$ (+) and $\mathsf{R}_{2,2}$ with $C = 3.0$ ($\triangle$).

partial differential equation. The analytical solution given by Sachdev (1987) is

$$u(x,t) = \left(1 + \exp\left(\frac{x - 0.5t}{2\nu}\right)\right)^{-1}.$$

The initial condition is imposed at $t = 0.02$ and a (maximum) Péclet number of 1000 is considered. Thus, the variation of u is extremely sharp and stabilized formulations are required. Here, we also compare the GLS and LS the formulations discussed in Sections 5.4.6.2 and 5.4.6.4, respectively. A piecewise uniform mesh of 1000 linear elements is used to minimize the error due to space discretization and highlight temporal errors. The mesh size is $h = 0.002$ on $[0, 0.2] \cup [0.4, 1]$ and $h/10$ on $]0.2, 0.4[$. As shown in Figure 5.26, the response obtained at time $t = 1$ with the stabilized/Crank–Nicolson is much worse than the one given by the stabilized/$\mathsf{R}_{2,2}$. Since the spatial mesh size is very small, this is clearly due to the inferior phase accuracy of the second-order Crank–Nicolson method. Notice that in both time-stepping schemes the added dissipation introduced by the LS approach increases with the size of the time step.

5.6.5 Burgers' equation in 1D

Burgers' equation is now solved in $]0, 1[$ with homogeneous Dirichlet boundary conditions. The problem is defined by

$$\begin{cases} u_t + uu_x = \nu u_{xx}, \\ u(x,0) = \sin(\pi x), \\ u(0,t) = u(1,t) = 0 \end{cases}$$

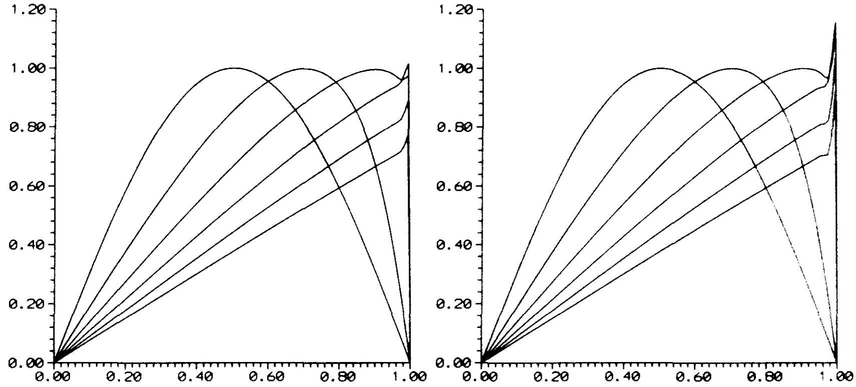


Fig. 5.27 Burgers' equation in 1D: least-squares/R$_{2,2}$ with $C = 2$ for $Pe = 5$ (left) and $Pe = 5000$ (right). Results are shown for $t = 0, 0.2, 0.4, 0.6, 0.8, 1$.

and presents a boundary layer at $x = 1$. The numerical results in Figure 5.27 were obtained using a uniform mesh of 100 linear elements.

A central spatial discretization, such as the Galerkin formulation, is unable to reproduce the solution to this problem for high values of the Péclet number. We note from Figure 5.27 that the LS/R$_{2,2}$ scheme produces an excellent response, except in the boundary layer where it presents a localized oscillation. The amplitude of this oscillation is controlled by the Courant number. A Newton–Raphson procedure has been used to iteratively solve the algebraic system governing the nodal values of the solution and good accuracy was obtained in few iterations.

5.6.6 Two-dimensional Burgers' equation

A 2D Burgers' problem is now considered over the square domain $\Omega =]0, 1[\times]0, 1[$. The problem is defined by the coupled equations

$$\begin{cases} u_t + u\,u_x + v\,u_y = \nu\nabla^2 u, \\ v_t + u\,v_x + v\,v_y = \nu\nabla^2 v, \end{cases}$$

and by the following initial and boundary conditions:

$$\begin{aligned} u(x,y,0) &= \sin(\pi x)\,\cos(\pi y), & v(x,y,0) &= \cos(\pi x)\,\sin(\pi y), \\ u(0,y,t) &= u(1,y,t) = 0, & v(x,0,t) &= v(x,1,t) = 0, \\ \frac{\partial u}{\partial n}(x,0,t) &= \frac{\partial u}{\partial n}(x,1,t) = 0, & \frac{\partial v}{\partial n}(0,y,t) &= \frac{\partial v}{\partial n}(1,y,t) = 0. \end{aligned}$$

A mesh of 30 × 30 bilinear elements is used for the spatial discretization. The problem exhibits various symmetries:

$$u(x,y,t) = v(y,x,t) \quad \text{and} \quad u(x,y,t) = -u(1-x, 1-y, t).$$

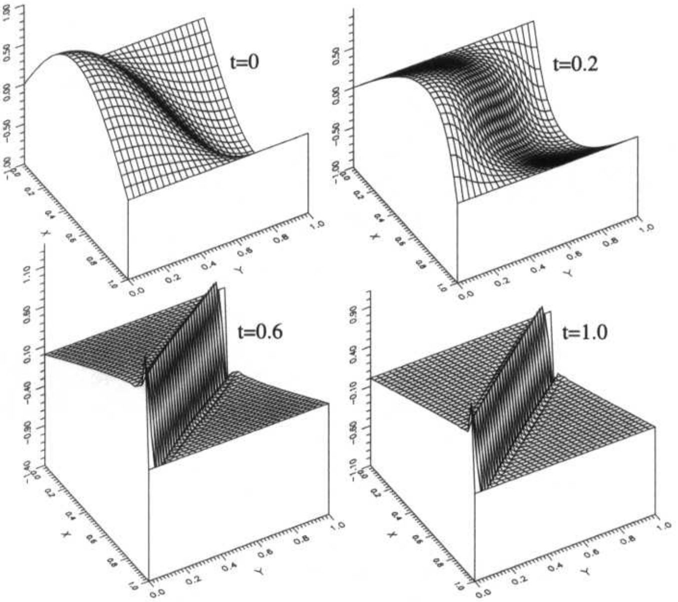


Fig. 5.28 Burgers' equation in 2D: least-squares and $\mathsf{R}_{2,2}$ for $Pe = 5000$ and $C = 2$.

Along the transverse section of the domain they imply that

$$u(x, x, t) = v(x, x, t) \quad \text{and} \quad u(x, x, t) = -u(1 - x, 1 - x, t).$$

When convection dominates the nonlinear transport, the solution includes the formation of a discontinuity along the diagonal of the domain passing through the points $(0, 1)$ and $(1, 0)$.

As in the 1D Burgers' problem, the Galerkin spatial discretization fails to give acceptable results for high values of the Péclet number. Figure 5.28 shows that the LS finite element scheme combined with Padé approximation $\mathsf{R}_{2,2}$ succeeds in capturing the solution with good accuracy. Since the scheme is not monotone, it exhibits residual oscillations near sharp gradients. It does, however, not degrade the solution outside the zones of strong solution gradient.

Appendix Least-squares in transient/relaxation problems

The LS formulation is intrinsically ill-posed for steady convection–diffusion problems. This is not the case of other well-known stabilization techniques such as the SUPG or GLS formulations. For this reason, LS methods were not introduced in Chapter 2. However, LS techniques are commonly used in transient pure convection, see Chapters 3 and 4 and have been extended to transient convection–diffusion in this chapter. Are the intrinsic problems of least-squares in steady convection–diffusion hidden in the transient case? Do they corrupt the transient solution or its steady-state?

Here a simple example/exercise is used to recall the difficulties of LS in a steady-state (elliptic) formulation. We also show that these difficulties do not exist in transient situations. Moreover, the steady solution obtained as the result of a relaxation technique with LS is well-behaved and lacks the inconsistencies of the pure steady solution. Thus, LS can be an alternative to SUPG or GLS in transient (or relaxation) problems because it is also well-posed and can be implemented with standard $\mathcal{C}^0$ finite elements without introducing additional variables (Huerta and Donea, 2002).

The steady-state convection–diffusion equation. The model problem, which is studied in order to show the inherent difficulties of the LS formulation in steady-state problems, is the homogenous 1D convection–diffusion equation,

$$a\,u_x - \nu\,u_{xx} = 0. \tag{A.1}$$

A particular example suffices to illustrate the inconsistency of the LS formulation in steady-state problems, in particular,

$$\begin{cases} a\,u_x - \nu\,u_{xx} = 0 & \text{in } \Omega :=]0,1[, \\ u(0) = 0 \text{ and } u(1) = 1. \end{cases} \tag{A.2}$$

The exact solution of (A.2) is:

$$u_{\mathsf{exact}}(x) = \frac{1-\exp(ax/\nu)}{1-\exp(a/\nu)}, \tag{A.3}$$

where, as expected, a/ν controls the relative importance of the convection and diffusion term, see Figure A.1.

The variational form for the LS formulation of (A.2) is: find u in the appropriate trial space such that

$$\big(a\,v_x - \nu\,v_{xx}, a\,u_x - \nu\,u_{xx}\big) = 0 \qquad \forall v \in \mathcal{H}_0^2(\Omega). \tag{A.4}$$

Remark A.1 (The trial and test spaces). The variational form involves integrals of second derivatives of both u and v. Thus, it is necessary that the spaces of trial and test functions are subspaces of $\mathcal{H}^2$. On one hand, the space of the test (weighting) functions is, as usual, chosen such that homogenous conditions are verified on the Dirichlet portion of the boundary, Γ_D. Thus, in general, the test functions belong to

$$\mathcal{H}^2_{\Gamma_D}(\Omega) := \{v \in \mathcal{H}^2(\Omega) \mid v = 0 \text{ on } \Gamma_D\}.$$

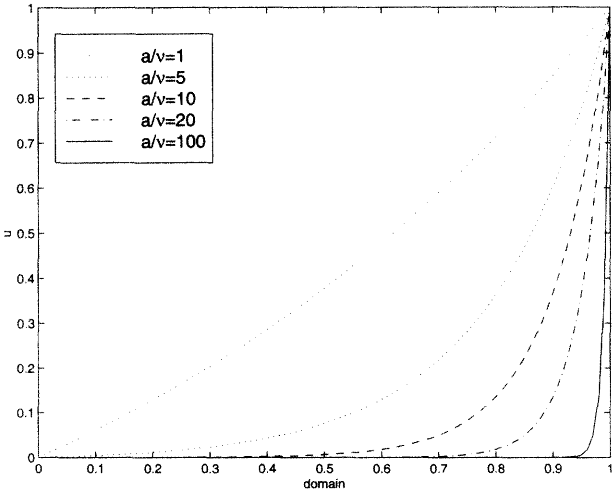


Fig. A.1 Exact solution of problem (A.2) for increasing values of a/ν.

In the particular case studied here, see (A.2), only essential boundary conditions are imposed, consequently,

$$\mathcal{H}^2_{\Gamma_D} = \mathcal{H}^2_0.$$

On the other hand, the trial solutions, u, must satisfy the Dirichlet (forced) boundary conditions; thus, the space of trial solutions is, in general,

$$\{u \in \mathcal{H}^2(\Omega) \mid u = u_D \text{ on } \Gamma_D\} \equiv \mathcal{H}^2_0(\Omega) + \{\overline{u}_D\},$$

where u_D are the Dirichlet boundary data on Γ_D, and $\overline{u}_D$ is any function of $\mathcal{H}^2(\Omega)$ satisfying the Dirichlet boundary conditions.

Remark A.2 (The actual steady problem). The extra regularity in the LS weak form also implies extra boundary conditions for the strong form of the problem. These extra boundary conditions are naturally enforced in (A.4). In fact, if equation (A.4) is integrated by parts recalling that $v \in \mathcal{H}^2_0$, the Euler–Lagrange equation and the natural boundary conditions associated with (A.4) become apparent, namely

$$\begin{cases} a^2\, u_{xx} - \nu^2\, u_{xxxx} = 0 & \text{in } \Omega :=]0,1[, \\ u(0) = 0 \text{ and } u(1) = 1, & \\ \nu(a\, u_x - \nu u_{xx}) = 0 & \text{at } x = 0 \text{ and } x = 1. \end{cases} \tag{A.5}$$

Note that this problem is consistent with the original one, see (A.2).

Remark A.3. It is well known that the LS formulation is formally equivalent to a higher-order problem. This has originated a debate on boundary conditions in LS problems, see for instance the books by Zienkiewicz and Morgan (1983, Sec. 6.8.) and Jiang (1998, p 25). Here it is clear that, in finite elements, the extra boundary conditions are natural conditions which do not appear explicitly in the weak form. But they are imposed implicitly! If other conditions, different from these natural ones, are imposed on the boundary spurious solutions may be obtained, as noted by Zienkiewicz and Morgan (1983). Thus, as observed by Jiang (1998), if other numerical techniques are used, such as finite differences, the extra (natural) boundary conditions must be enforced explicitly.

The objective now is to study the stabilization effects of the LS formulation for convection-dominated situations. Thus, the limit when ν approaches 0 is analyzed, in this case, the variational form (A.4) becomes: find $u \in \mathcal{H}_0^1(\Omega) + \{\overline{u}_D\}$, such that

$$\left(v_x, u_x\right) = 0, \qquad \forall v \in \mathcal{H}_0^1(\Omega). \tag{A.6}$$

However, the equivalent strong form (i.e., the Euler–Lagrange equation associated with the variational form and the boundary conditions induced by the functional spaces) associated with (A.6) is

$$\begin{cases} u_{xx} = 0 & \text{in } \Omega =]0,1[\\ u(0) = 0 \text{ and } u(1) = 1. & \end{cases} \tag{A.7}$$

The unique solution of this problem is $u(x) = x$, which is obviously non consistent with the limit case of (A.3) for ν approaching zero. In fact, this solution coincides with $\lim_{\nu \to \infty} u_{\mathsf{exact}}(x) = x$. Thus, the steady LS solution will converge in the limit case $\nu = 0$ to the exact solution for the limit case $\nu \to \infty$. This is intrinsic to the LS formulation in steady problems. This is why LS are not used as a stabilization technique for the steady convection–diffusion equation.

It is important to note that the problem defined by (A.7) corresponds to the limit problem of (A.5). Thus, the difficulties of the LS formulation are also present for small values of ν as it will be seen in the following numerical examples.

These conclusions can be confirmed numerically. Figure A.2 shows numerical solutions of (A.4) for a uniform mesh of 10 $\mathcal{C}^1$ finite elements (Hermite elements of degree 3) as a/ν grows. The Galerkin approximations with the same mesh are also plotted for comparison. As expected, when convection becomes dominant, the Galerkin formulation becomes unstable and the LS formulation approaches the "non-physical" solution $u = x$.

Figure A.3 presents the influence of the element size in this LS approximation. As expected, as the number of elements increases the Péclet number decreases and the approximation improves. Note however the poor results obtained with 100 elements (200 degrees of freedom).

If standard stabilization techniques for steady problems, such as SUPG and GLS, are used, this intrinsic problem in the limit $\nu \to 0$ does not appear, see Chapter 2. The model problem presented in (A.2), can be rewritten in weak form using the SUPG

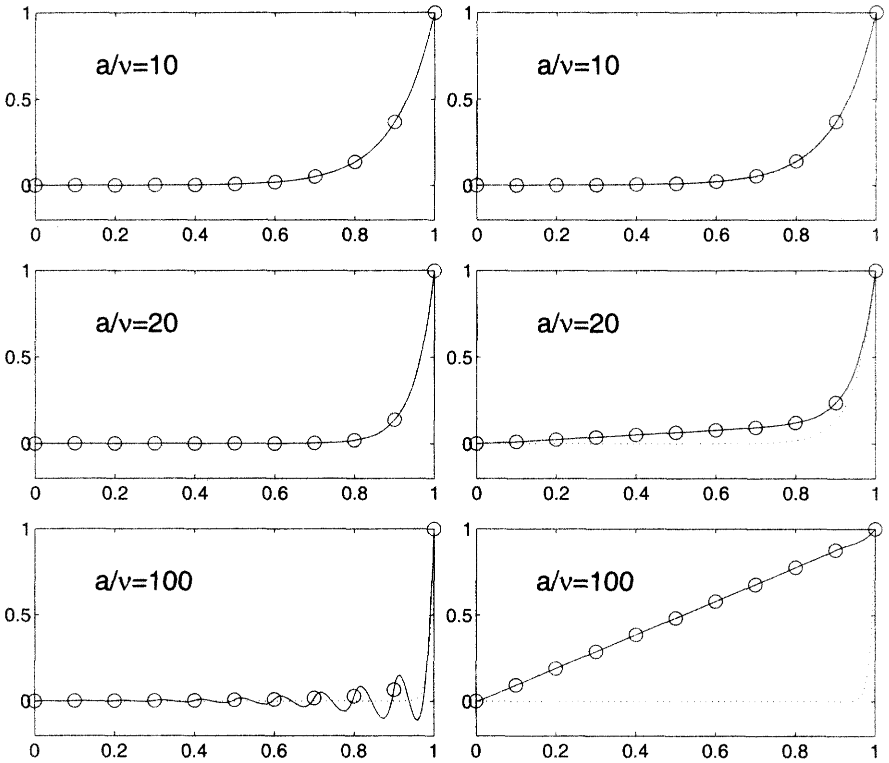


Fig. A.2 Galerkin (left) and Least-Squares (right) approximations (solid lines) to the exact (dotted) solution of the model problem for increasing values of a/ν.

formulation as, find u such that

$$\underbrace{\left(v, a\, u_x\right) + \left(v_x, \nu\, u_x\right)}_{\text{Galerkin}} + \underbrace{\sum_e \tau\left(a\, v_x, a\, u_x - \nu\, u_{xx}\right)_{\Omega^e}}_{\text{SUPG Stabilization}} = 0 \quad \forall v \in \mathcal{H}_0^1(\Omega), \tag{A.8}$$

where the Galerkin term and the stabilization term are easily recognized.

Remark A.4. Note that u must have the first derivative integrable in the whole domain and the second derivative integrable inside the elements. Thus, u belongs to

$$\{u \in \mathcal{H}^1(\Omega) \mid u = u_D \text{ on } \Gamma_D,\ u|_{\Omega^e} \in \mathcal{H}^2(\Omega^e) \text{ for every element } \Omega^e\}$$
$$\equiv \mathcal{H}_0^{1+}(\Omega) + \{\bar{u}_D\},$$

where

$$\mathcal{H}_0^{1+}(\Omega) = \{v \in \mathcal{H}_0^1(\Omega) \mid v|_{\Omega^e} \in \mathcal{H}^2(\Omega^e) \text{ for every element } \Omega^e\} \tag{A.9}$$

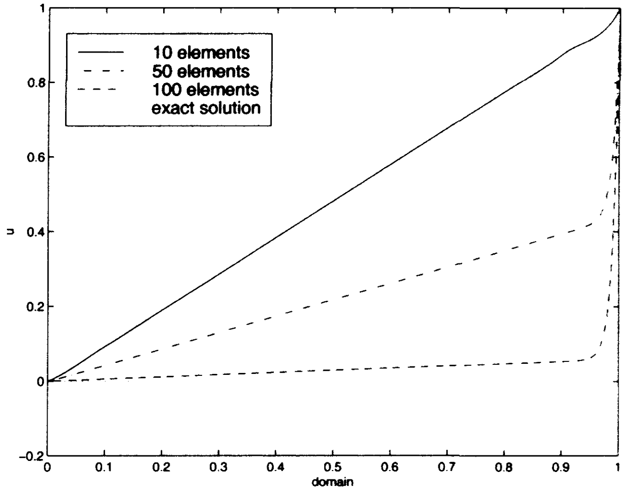


Fig. A.3 Influence of the element size on the LS approximations for $a/\nu = 100$.

and $\overline{u}_D$ is any function of $\mathcal{H}^{1+}(\Omega)$ satisfying the Dirichlet boundary conditions.

As previously discussed for the LS formulation, the limit $\nu \to 0$ is analyzed. In this case, the previous weak form becomes

$$\big(v, a\,u_x\big) + \tau\big(a\,v_x, a\,u_x\big) = 0 \quad \forall v \in \mathcal{H}_0^1(\Omega), \tag{A.10}$$

which can be interpreted as the Galerkin discretization of

$$\begin{cases} a\,u_x - a^2\tau\,u_{xx} = 0 & \text{in } \Omega \\ u(0) = 0 \text{ and } u(1) = 1. \end{cases} \tag{A.11}$$

As expected, the SUPG formulation stabilizes the solution. It introduces an artificial diffusion (viscosity) $\bar{\nu} = a^2\tau$. Figure A.4 shows the SUPG approximation for 10 $\mathcal{C}^1$ finite elements, $a/\nu = 100$, and the optimal intrinsic time for linear elements, $\tau = (\coth(Pe) - 1/Pe)h/(2a)$, which, as expected, is not optimal in this case because Hermite elements of degree 3 are used.

Remark A.5 (The GLS formulation). Similar conclusions are drawn if a GLS formulation is employed. Equation (A.8) becomes

$$\underbrace{\big(v, a\,u_x\big) + \big(v_x, \nu\,u_x\big)}_{\text{Galerkin}} + \underbrace{\sum_e \tau\big(a\,v_x - \nu\,v_{xx}, a\,u_x - \nu\,u_{xx}\big)_{\Omega^e}}_{\text{GLS Stabilization}} = 0,$$

for all $v \in \mathcal{H}_0^{1+}(\Omega)$, which induces exactly the same weak, (A.10), and strong, (A.11), forms as SUPG in the limiting case of zero viscosity.

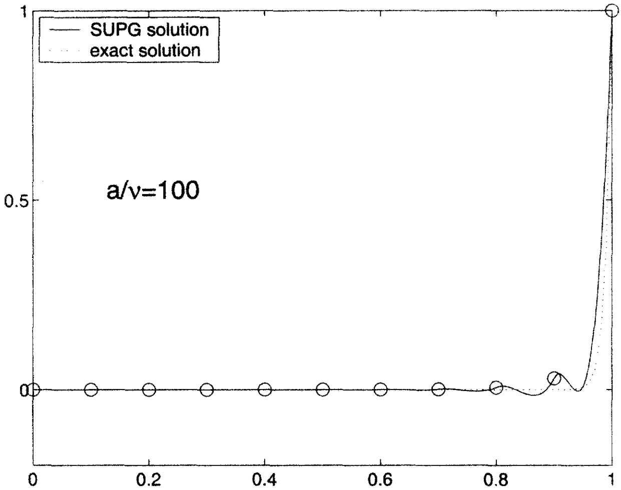


Fig. A.4 SUPG approximation for $a/\nu = 100$.

The transient convection–diffusion equation. It is well known that for the transient convection–diffusion equation the LS formulation works properly. Recall that this is also the case for both SUPG and GLS stabilization techniques. In order to understand this behavior, the variational form obtained for each method will be presented. A homogenous 1D transient convection–diffusion equation,

$$u_t + a\,u_x - \nu\,u_{xx} = 0, \tag{A.12}$$

is used to study the different stabilization techniques. The strong form of the model problem in the transient case is simply

$$\begin{cases} u_t + a\,u_x - \nu\,u_{xx} = 0 & \text{in } \Omega \times \mathbb{R}^+ :=]0,1[\times]0,\infty[, \\ u(0,t) = u_0 \text{ and } u(1,t) = u_1 & \text{for } t \in \mathbb{R}^+ \text{ (boundary conditions)}, \\ u(x,0) = f(x) & \text{for } x \in \Omega \text{ (initial condition)}, \end{cases} \tag{A.13}$$

where the initial condition $f(x)$ must be a smooth function.

In order to implement a stabilized formulations (LS, SUPG or GLS) for transient problems we perform first the time discretization of (A.13). For simplicity, a single-step time-integration scheme is used, but higher-order in time schemes can also be implemented. After time discretization the spatial equation that must be solved at each time step is

$$\frac{\Delta u}{\Delta t} + \theta(a\,\Delta u_x - \nu\,\Delta u_{xx}) = -(a\,u_x^n - \nu\,u_{xx}^n) \tag{A.14}$$

with $\Delta u = u^{n+1} - u^n$. Thus, the spatial operator to be used for the LS approach is obtained from the l.h.s. of the previous equation, namely

$$\mathcal{L} = 1/\Delta t + \theta(a\,\partial_x - \nu\,\partial_{xx}). \tag{A.15}$$

Once the spatial operator is known the variational form induced by the LS formulation can be defined, that is, $\forall v \in \mathcal{H}_0^2(\Omega)$,

$$\begin{aligned} 0 &= \left(\mathcal{L}(v), u_t + a\,u_x - \nu\,u_{xx}\right) \\ &= \left(\frac{v}{\Delta t} + \theta(a\,v_x - \nu\,v_{xx}), u_t + a\,u_x - \nu\,u_{xx}\right). \end{aligned} \tag{A.16}$$

Remark A.6. Note that, as usual, the test space is independent of time and the trial space is in this case defined as

$$\begin{aligned} &\{u \mid u(\cdot, t) \in \mathcal{H}^2(\Omega), t \in \mathbb{R}^+ \text{ and } u(x,t) = u_D \text{ for } x \text{ on } \Gamma_D\} \\ &\qquad\qquad \equiv \left\{u \mid u(\cdot, t) \in \mathcal{H}_0^2(\Omega) + \{\overline{u}_D\}, t \in \mathbb{R}^+\right\} \end{aligned}$$

As expected, the LS formulation still requires higher regularity for the solution because of the presence of the second-order spatial derivatives.

Remark A.7 (The actual transient problem). As previously observed in Remark A.2, the extra regularity in the LS weak form also implies extra boundary conditions for the strong form of the problem. This extra boundary conditions are naturally enforced in (A.16). The Euler–Lagrange equation and the boundary conditions associated with (A.16) are

$$\begin{cases} u_t + a\,u_x - \nu\,u_{xx} - a\theta\Delta t(u_t + a\,u_x)_x & \\ \qquad\qquad - \nu\theta\Delta t(u_t - \nu\,u_{xx})_{xx} = 0 & \text{in } \Omega \times \mathbb{R}^+, \\ u(0,t) = u_0 \text{ and } u(1,t) = u_1 & \text{for } t \in \mathbb{R}^+, \\ \nu\theta\Delta t(u_t + a\,u_x - \nu\,u_{xx})\big|_{x=0} = 0 & \text{for } t \in \mathbb{R}^+, \\ \nu\theta\Delta t(u_t + a\,u_x - \nu\,u_{xx})\big|_{x=1} = 0 & \text{for } t \in \mathbb{R}^+, \\ u(x,0) = f(x) & \text{on } x \in \Omega. \end{cases} \tag{A.17}$$

Note that this problem is also consistent with the original one, namely (A.13), hint: use the transient differential equation (A.12) to verify the Euler–Lagrange equation.

In the limit case when $\nu \to 0$ the weak form (A.16) becomes

$$\left(v, u_t + a\,u_x\right)_\Omega + a\,\varepsilon\left(v_x, u_t + a\,u_x\right)_\Omega = 0 \qquad \forall v \in \mathcal{H}_0^1(\Omega), \tag{A.18}$$

where, for convenience, a parameter, $\varepsilon := \theta\Delta t$, is defined. Note that, as previously observed, the regularity conditions are relaxed in this limit case.

The Euler–Lagrange equation associated with the previous weak form is

$$\left[1 - a\,\varepsilon\,\partial_x\right](u_t + a\,u_x) = 0, \tag{A.19}$$

and the strong form induced by (A.18) is

$$\begin{cases} \left[1 - a\,\varepsilon\,\partial_x\right](u_t + a\,u_x) = 0 & \text{in } \Omega \times \mathbb{R}^+ :=]0,1[\times]0,\infty[\\ u(0,t) = u_0 \text{ and } u(1,t) = u_1 & \text{for } t \in \mathbb{R}^+, \\ u(x,0) = f(x) & \text{on } x \in \Omega. \end{cases} \tag{A.20}$$

That is, (A.18) can be interpreted as the Galerkin discretization of this strong form. It is important to note, for consistency, that the strong form defined by (A.20) corresponds to the limit, when ν approaches 0, of problem (A.17).

Remark A.8. Note that in the limit case when ν approaches 0, (A.12) becomes $u_t + a\,u_x = 0$, whose solutions also satisfy (A.19).

Remark A.9 (Stabilizing effect of the LS formulation). At steady-state, that is when $u_t \to 0$, the problem defined by (A.20) reduces to

$$\begin{cases} a\,u_x - a^2\varepsilon\,u_{xx} = 0, & \text{in } \Omega \\ u(0) = u_0 \text{ and } u(1) = u_1, \end{cases}$$

which clearly shows both the stabilizing effect of the LS formulation and the fact that the steady solution of the transient LS formulation is well-posed. Note the equivalence with (A.11) and that the artificial diffusion is controlled, as expected, by the time increment, $\bar{\nu} = a^2\,\varepsilon = a^2\theta\Delta t$.

However, this problem is the transient counterpart to (A.7), whose solution was clearly "non-physical". Thus, the obvious goal now is to determine if (A.20) presents physical solutions, or not. First, however, we wonder if (A.20) is well-posed. Note that this is a hyperbolic equation with prescribed Dirichlet conditions on the whole boundary. Does (A.20) present feasible solutions? These questions will be answered. Before, however, the SUPG and GLS stabilization techniques are studied in order to obtain the transient counterpart to (A.11).

In SUPG and GLS formulations the extra term added to the Galerkin weak form is a function of the residual to ensure consistency of the formulation. The residual of (A.14) is simply

$$\mathcal{R}(\Delta u) := \frac{\Delta u}{\Delta t} + \theta(a\,\Delta u_x - \nu\,\Delta u_{xx}) + a\,u_x^n - \nu\,u_{xx}^n$$

and the stabilized weak form is

$$\big(v, \mathcal{R}(\Delta u)\big) + \sum_e \tau\big(\mathcal{P}(v), \mathcal{R}(\Delta u)\big)_{\Omega^e} = 0 \quad \forall v \in \mathcal{H}_0^{1+}(\Omega). \tag{A.21}$$

In 1D and for SUPG, see Chapter 2, this operator is defined as $\mathcal{P}(v) := \theta a\,v_x$. Thus, the variational form associated with SUPG is, $\forall v \in \mathcal{H}_0^1(\Omega)$,

$$\big(v, u_t + a\,u_x - \nu\,u_{xx}\big) + \sum_e \tau\big(\theta a\,v_x, u_t + a\,u_x - \nu\,u_{xx}\big)_{\Omega^e} = 0. \tag{A.22}$$

Moreover, in the limit case when ν approaches 0 this form becomes

$$\big(v, u_t + a\,u_x\big) + \tau\big(\theta a\,v_x, u_t + a\,u_x\big) = 0 \quad \forall v \in \mathcal{H}_0^1(\Omega), \tag{A.23}$$

where the sum over element interiors is no longer necessary because only first derivatives are present (second derivatives disappear in the limit $\nu \to 0$).

This equation can be interpreted as the standard Galerkin discretization of

$$\left[1 - a\theta\tau\,\partial_x\right](u_t + a\,u_x) = 0,$$

which corresponds to (A.19) for $\varepsilon := \theta\tau$. Thus, the strong form corresponding to the stabilized formulation is identical to (A.20) (where ε now depends on the intrinsic time τ) and the analytical solution developed next will also be valid for the SUPG stabilization.

Remark A.10 (GLS stabilisation). In the GLS formulation, the perturbation operator is defined as $\mathcal{P} := \mathcal{L}$, see equation (A.15). Then, the variational form obtained for the limit case $\nu \to 0$ can be interpreted as the Galerkin discretization of

$$\left[\left(1 + \frac{\tau}{\Delta t}\right) - a\theta\tau\,\partial_x\right](u_t + a\,u_x) = 0,$$

which also corresponds to (A.19) where, in this case, $\varepsilon := \theta\tau\Delta t/(\tau + \Delta t)$.

The analytical solution of the strong form associated with the stabilized formulations is obtained from the general solution of (A.19), which is

$$u(x,t) = F(t)\exp(x/a\,\varepsilon) + G(at - x), \tag{A.24}$$

where functions F and G are be determined imposing the boundary and initial conditions. In this particular case, F and G are piecewise defined functions, for $(n-1)/a \le t \le n/a$

$$F(t) = u_1\frac{1-\exp(\frac{-n}{a\,\varepsilon})}{\exp(\frac{1}{a\,\varepsilon})-1} - u_0\frac{1-\exp\left(\frac{-n+1}{a\,\varepsilon}\right)}{\exp(\frac{1}{a\,\varepsilon})-1} - f(n-at)\exp(\frac{-n}{a\,\varepsilon}),$$

and for $(n-1) \le z \le n$

$$G(z) = u_0\left(1 + \frac{1-\exp\left(\frac{-n+1}{a\,\varepsilon}\right)}{\exp(\frac{1}{a\,\varepsilon})-1}\right) - u_1\frac{1-\exp(\frac{-n}{a\,\varepsilon})}{\exp(\frac{1}{a\,\varepsilon})-1} + f(n-z)\exp\left(\frac{-n}{a\,\varepsilon}\right).$$

Note that the analytical steady-state solution (i.e., when $t \to \infty$) is

$$u(x) = (u_1 - u_0)\frac{\exp(x/a\,\varepsilon)-1}{\exp(1/a\,\varepsilon)-1} + u_0.$$

This equation can be rewritten in terms of an artificial viscosity $\bar{\nu} = a^2\,\varepsilon$, namely

$$u(x) = (u_1 - u_0)\frac{\exp(ax/\bar{\nu})-1}{\exp(a/\bar{\nu})-1} + u_0, \tag{A.25}$$

which clearly shows the boundary layer structure of the solution, see (A.3), and, as expected, only depends on the boundary conditions (not on the initial condition).

Therefore, in the transient case the strong form of the limit problem (i.e., when $\nu \to 0$) is well-posed, has an analytical solution and the steady-state solution presents the desired structure.

The model problem (A.13) is now numerically solved for the particular case

$$\begin{cases} u_t + a\,u_x - \nu\,u_{xx} = 0 & \text{in } \Omega \times \mathbb{R}^+ :=]0,1[\times]0,\infty[, \\ u(0,t) = 0 \text{ and } u(1,t) = 1 & \text{for } t \in \mathbb{R}^+, \\ u(x,0) = x & \text{for } x \in \Omega. \end{cases}$$

In this particular case, the analytical steady solution of the stabilized problem, equation (A.25), is

$$\lim_{t\to\infty} u(x,t) = \frac{\exp(ax/\bar{\nu}) - 1}{\exp(a/\bar{\nu}) - 1}.$$

Note that, the boundary layer is governed by numerical parameters, $\bar{\nu} = a^2\theta\Delta t$ for LS, $\bar{\nu} = a^2\theta\tau$ for SUPG and $\bar{\nu} = a^2\theta\tau\Delta t/(\tau + \Delta t)$ for GLS.

Before presenting the numerical results, the formulation proposed in Section 5.4.6.4 is recalled, see also Huerta and Donea (2002). It allows us to use standard $\mathcal{C}^0$ finite elements. The weak problem is as follows: find $u(\cdot,t) \in \mathcal{H}_0^{1+}(\Omega) + \{\overline{u}_D\}$, for all $v \in \mathcal{H}_0^{1+}(\Omega)$, such that

$$\big(v, u_t + a\,u_x\big) + \big(v_x, \nu\,u_{xx}\big) + \sum_e \theta\Delta t\big(a\,v_x - \nu\,v_{xx}, u_t + a\,u_x - \nu\,u_{xx}\big) = 0.$$

Note that this formulation, as well as the standard LS in $\mathcal{H}^2$, is symmetric, this is not the case of SUPG and GLS.

The model problem is solved for $a/\nu = 100$. Figure A.5 shows numerical results at different instants for LS, SUPG and GLS with a uniform mesh of 10 $\mathcal{C}^0$ elements (left) and a Crank–Nicolson scheme with $\Delta t = 1$ (i.e., $C = 1$). The exact solution is also displayed for comparison. For SUPG and GLS, the intrinsic time is chosen equal to the optimal stabilization parameter for linear elements in the steady convection–diffusion equation, namely

$$\tau = \frac{h}{2a}\left[\coth(Pe) - \frac{1}{Pe}\right]\frac{1}{\theta},$$

where, in this case, $\theta = 1/2$, see also Remark 5.9. Thus the exact nodal solution must be recovered for SUPG at steady-state. The same figure also shows results for 10 $\mathcal{C}^1$ elements (right). As expected both $\mathcal{C}^0$ and $\mathcal{C}^1$ elements produce reasonable results in all cases, in particular, with the LS stabilization.

In summary, standard LS stabilization (in $\mathcal{H}^2$) can not be used directly in steady convection–diffusion problems because it produces non-physical solutions in the pure convection limit. But the transient convection–diffusion equation can be stabilized with LS.

It should also be observed that the LS formulation proposed can be implemented with standard $\mathcal{C}^0$ finite elements and does not necessitate to introduce additional nodal

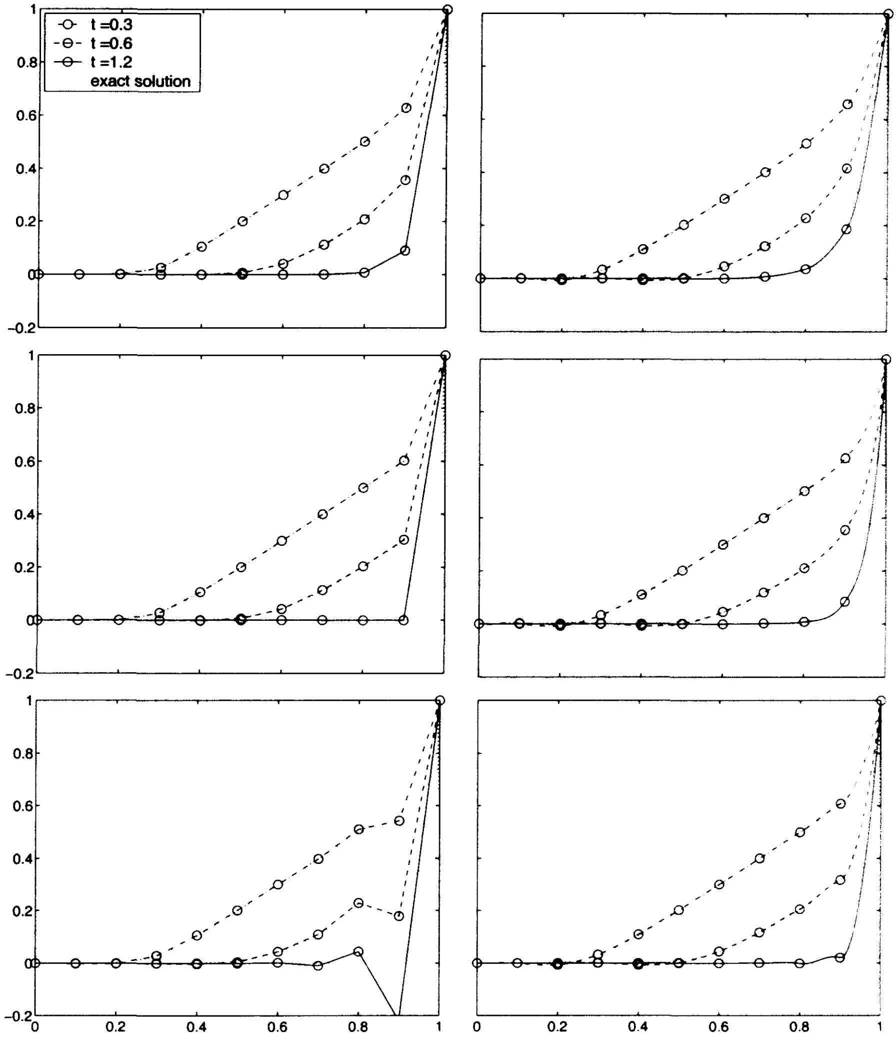


Fig. A.5 Least-Squares (top) SUPG (center) and GLS (bottom) approximations using C^0 (left) and C^1 (right) elements.

variables. In fact, if the steady-state solution is desired, LS produces correct results when a relaxation technique is employed.

This intrinsic difficulty of LS in pure steady-state problems is not present in other stabilization techniques, such as SUPG or GLS, which stabilize properly both steady and transient equations.

6

Viscous incompressible flows

This chapter is concerned with the finite element treatment of viscous incompressible flows governed by the Navier–Stokes equations. Solving these equations numerically meets two difficult problems. The first one is related to the discretization of the nonlinear convective terms, it requires the use of stabilized finite element formulations to properly treat high Reynolds number flows. This issue has been already discussed in previous chapters. The second difficulty is related to the numerical treatment of the saddle-point problem which arises from the variational formulation of the incompressible flow equations with the pressure acting as a Lagrangian multiplier of the incompressibility constraint. Different methods have been devised for the solution of the above numerical difficulties and some of them are discussed in this chapter.

6.1 INTRODUCTION

In this last chapter we consider the finite element modeling of steady and transient viscous, incompressible flows governed by the Navier–Stokes equations. Section 6.2 presents a review of the basic continuum mechanics concepts that are needed for the finite element formulation of incompressible flow problems. The Navier–Stokes equations governing viscous incompressible flows in the laminar regime are then introduced, together with the associated initial and boundary conditions.

The following sections are devoted to the application of finite element techniques to model incompressible flows in the Eulerian description. As shown in Section 6.3, there are two major sources of numerical difficulty in the use of the standard Galerkin finite element method. The first is related to the incompressibility of the

Finite Element Methods for Flow Problems. Jean Donea and Antonio Huerta
Copyright © 2003 John Wiley & Sons, Ltd.
ISBN: 0-471-49666-9

fluid and manifests itself when an inappropriate combination of element interpolation functions for the velocity and pressure is employed. As a consequence, instabilities in the pressure field may appear, and this is independent of the Reynolds number. That is, instabilities may occur even at very "slow" flows (low Reynolds number). A proper combination of interpolation spaces (for velocity and pressure) is needed unless a specific formulation circumventing these instabilities is employed.

A second source of numerical difficulty is due to the presence of nonlinear convective terms in the Navier–Stokes equations. As seen in the preceding chapters, the Galerkin formulation typically lacks stability when convective effects dominate and alternative spatial discretization procedures must be advocated to restore stability without compromising the accuracy. In the present chapter attention will therefore be focused on situations in which convective effects are important, as well as on the numerical difficulties arising from the fluid incompressibility.

In Section 6.4 we define the function spaces that are needed for the approximation of the velocity and pressure fields. Section 6.5 is concerned with the finite element formulation of stationary Stokes problems, thus leaving aside the problems arising from the convective terms in the full Navier–Stokes equations. Emphasis is placed on mixed finite element formulations in which both velocity and pressure are retained as unknowns. Conditions to be satisfied that produce stable approximations with mixed methods in the standard Galerkin formulation are discussed in some detail. Stabilized formulations are then introduced which provide a remedy for the deficiencies of the Galerkin approach. We also discuss the penalty method which uncouples the determination of the velocity and pressure fields through a relaxation of the incompressibility constraint.

Section 6.6 deals with steady Navier–Stokes problems. We underline the additional difficulty introduced by the nonlinear convective terms. Fortunately, stabilization techniques provide a remedy for the deficiencies of the Galerkin approach arising from both the convective terms and the incompressibility constraint.

Transient problems are introduced in Section 6.7. Here, emphasis is placed on the advantages of a fractional-step projection method for the time integration of the Navier–Stokes equations. Simple test problems easily solvable by the interested reader are presented in Section 6.8. In particular, as an illustration, an application of the fractional-step method for natural convection problems is discussed in Section 6.8.4.

Several excellent texts devoted to the finite element modeling of incompressible flow problems are available as complements to the present introductory chapter. Particularly noteworthy are Glowinski (1984), Carey and Oden (1986), Girault and Raviart (1986), Glowinski and Le Tallec (1989), Gresho and Sani (2000), Gunzburger (1989), Pironneau (1989), Quartapelle (1993), Quarteroni and Valli (1994) and Temam (2001). The finite element books by Baker (1983), Jiang (1998), Hughes (2000), Zienkiewicz and Taylor (2000b) and Reddy and Gartling (2001) also contain interesting chapters on incompressible problems.

6.2 BASIC CONCEPTS

6.2.1 Strain rate and spin tensors

The analysis of the relative motion of neighboring particles within a fluid is similar to the theory of deformation of an elastic solid: the rate of strain and the rate of rotation in the fluid take the place of the strain and rotation of the solid. An important variable in the characterization of fluid motion is the *velocity gradient*. The velocity gradient is a second-order tensor defined in a Cartesian coordinate system by

$$\boldsymbol{\nabla v} = \begin{pmatrix} \dfrac{\partial v_1}{\partial x_1} & \dfrac{\partial v_1}{\partial x_2} & \dfrac{\partial v_1}{\partial x_3} \\ \dfrac{\partial v_2}{\partial x_1} & \dfrac{\partial v_2}{\partial x_2} & \dfrac{\partial v_2}{\partial x_3} \\ \dfrac{\partial v_3}{\partial x_1} & \dfrac{\partial v_3}{\partial x_2} & \dfrac{\partial v_3}{\partial x_3} \end{pmatrix}.$$

It may be decomposed into its symmetric and skew-symmetric parts according to

$$\frac{\partial v_i}{\partial x_j} = \frac{1}{2}\Big(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\Big) + \frac{1}{2}\Big(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i}\Big),$$

for $i, j = 1, \dots, \mathrm{n_{sd}}$, or

$$\boldsymbol{\nabla v} = \boldsymbol{\nabla}^{\mathrm{S}}\boldsymbol{v} + \boldsymbol{\nabla}^{\mathrm{W}}\boldsymbol{v}, \quad \text{where} \quad \begin{cases} \boldsymbol{\nabla}^{\mathrm{S}} := \frac{1}{2}(\boldsymbol{\nabla} + \boldsymbol{\nabla}^T), & \text{and} \\ \boldsymbol{\nabla}^{\mathrm{W}} := \frac{1}{2}(\boldsymbol{\nabla} - \boldsymbol{\nabla}^T) \end{cases} \tag{6.1}$$

The symmetric tensor $\boldsymbol{\nabla}^{\mathrm{S}}\boldsymbol{v}$ is called the *rate of deformation (or strain rate) tensor*. We shall frequently use the alternative notation

$$v_{(i,j)} := \frac{1}{2}\Big(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\Big) \quad \text{for } i, j = 1, \dots, \mathrm{n_{sd}}$$

to denote the components $[\boldsymbol{\nabla}^{\mathrm{S}}\boldsymbol{v}]_{ij}$ of the strain rate tensor. The skew-symmetric tensor $\boldsymbol{\nabla}^{\mathrm{W}}\boldsymbol{v}$ is called *vorticity tensor* (or *spin tensor*). Its components are defined by

$$[\boldsymbol{\nabla}^{\mathrm{W}}\boldsymbol{v}]_{ij} = \frac{1}{2}\Big(\frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i}\Big).$$

If the rate of deformation tensor $\boldsymbol{\nabla}^{\mathrm{S}}\boldsymbol{v}$ at a given point is identically zero, the motion in the neighborhood of that point is a rigid body rotation.

Associated with the vorticity tensor is the vorticity vector $\boldsymbol{\omega}$ defined by

$$\boldsymbol{\omega} = \boldsymbol{\nabla} \times \boldsymbol{v},$$

or in component form

$$\omega_1 = \frac{\partial v_3}{\partial x_2} - \frac{\partial v_2}{\partial x_3}, \quad \omega_2 = \frac{\partial v_1}{\partial x_3} - \frac{\partial v_3}{\partial x_1}, \quad \omega_3 = \frac{\partial v_2}{\partial x_1} - \frac{\partial v_1}{\partial x_2}.$$

Note that, a velocity field is said to be irrotational if vorticity vanishes everywhere within the flow field.

Remark 6.1. In 3D, the relation between the vorticity vector ω and the vorticity tensor $\nabla^{\mathrm{W}} \boldsymbol{v}$ is explicitly expressed by

$$\omega = -\mathbf{P} : \nabla^{\mathrm{W}} \boldsymbol{v}, \quad \text{i.e., } \omega_i = -\sum_{j,k=1}^{3} P_{ijk} [\nabla^{\mathrm{W}} \boldsymbol{v}]_{jk},$$

where $\mathbf{P}$ is the unit antisymmetric multilinear form of order 3 (see van der Waerden, 1970, p. 76). $\mathbf{P}$ is defined such that $P_{ijk} = 0$ if one of the indices is repeated and for i, j and k different

$$P_{ijk} = \begin{cases} 1 & \text{if } \pi \text{ is even} \\ -1 & \text{if } \pi \text{ is odd} \end{cases}$$

where π is the permutation taking 123 into ijk.

6.2.2 The stress tensor in a Newtonian fluid

In a general fluid at rest, only normal stresses are present and the stress tensor has the isotropic form

$$\sigma_{ij} = -p\,\delta_{ij},$$

where p is the static fluid pressure and δ_{ij} the Kronecker delta. The situation is different for a fluid in motion. Then, in general, tangential stresses are non-zero, and the normal component of the stress acting across a surface element depends on the direction of the normal to the element. The quantity $-\frac{1}{3}\sigma_{ii}$ (sum on i), which is invariant under rotation of the reference axes and reduces to the static fluid pressure when the fluid is at rest, is used to define the *pressure at a point in a moving fluid*:

$$p = -\frac{1}{3}\sigma_{ii}.$$

Note that this is a purely mechanical definition of pressure, which is thus not connected to the usual definition of pressure in thermodynamics.

It is convenient to decompose the Cauchy stress tensor σ_{ij} into the sum of an isotropic part $-p\,\delta_{ij}$ and a remaining non-isotropic part s_{ij}, the *deviatoric stress tensor*:

$$\sigma_{ij} = -p\,\delta_{ij} + s_{ij}.$$

For a *Newtonian fluid*, it is assumed that the stress tensor and the strain rate tensor are linearly related. The stress–strain rate relationship is given by

$$\sigma_{ij} = -p\,\delta_{ij} + s_{ij} = -p\,\delta_{ij} + \mu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right) + \lambda\,\frac{\partial v_k}{\partial x_k}\delta_{ij},$$

where μ is the fluid dynamic viscosity and λ the so-called second coefficient of viscosity. For an incompressible fluid one has $\boldsymbol{\nabla} \cdot \boldsymbol{v} = 0$ and consequently the above relation reduces to *Stokes' law*

$$\sigma_{ij} = -p\,\delta_{ij} + \mu\Big(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\Big) = -p\,\delta_{ij} + 2\mu\, v_{(i,j)}.$$

In compact form Stokes' law reads

$$\boldsymbol{\sigma} = -p\,\mathbf{I} + 2\mu\boldsymbol{\nabla}^{\mathrm{s}}\boldsymbol{v}, \tag{6.2}$$

or explicitly in 3D:

$$\begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{pmatrix} = -p \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$
$$+ 2\mu \begin{pmatrix} \dfrac{\partial v_1}{\partial x_1} & \dfrac{1}{2}\Big(\dfrac{\partial v_1}{\partial x_2} + \dfrac{\partial v_2}{\partial x_1}\Big) & \dfrac{1}{2}\Big(\dfrac{\partial v_1}{\partial x_3} + \dfrac{\partial v_3}{\partial x_1}\Big) \\ \dfrac{1}{2}\Big(\dfrac{\partial v_2}{\partial x_1} + \dfrac{\partial v_1}{\partial x_2}\Big) & \dfrac{\partial v_2}{\partial x_2} & \dfrac{1}{2}\Big(\dfrac{\partial v_2}{\partial x_3} + \dfrac{\partial v_3}{\partial x_2}\Big) \\ \dfrac{1}{2}\Big(\dfrac{\partial v_3}{\partial x_1} + \dfrac{\partial v_1}{\partial x_3}\Big) & \dfrac{1}{2}\Big(\dfrac{\partial v_3}{\partial x_2} + \dfrac{\partial v_2}{\partial x_3}\Big) & \dfrac{\partial v_3}{\partial x_3} \end{pmatrix}.$$

Remark 6.2. It is easy to prove the identity

$$[\boldsymbol{\nabla}\boldsymbol{v}]_{ij}\,\sigma_{ij} = [\boldsymbol{\nabla}^{\mathrm{S}}\boldsymbol{v}]_{ij}\,\sigma_{ij} = v_{(i,j)}\,\sigma_{ij},$$

which will be repeatedly used in the derivation of the variational form of Stokes and Navier–Stokes problems. The proof (see for instance Hughes, 2000, p. 79) uses the additive decomposition (6.1) and the fact that

$$[\boldsymbol{\nabla}^{\mathrm{W}}\boldsymbol{v}]_{ij}\,\sigma_{ij} = -[\boldsymbol{\nabla}^{\mathrm{W}}\boldsymbol{v}]_{ji}\,\sigma_{ij} = -[\boldsymbol{\nabla}^{\mathrm{W}}\boldsymbol{v}]_{ji}\,\sigma_{ji} = -[\boldsymbol{\nabla}^{\mathrm{W}}\boldsymbol{v}]_{ij}\,\sigma_{ij},$$

in view of the skew-symmetry of $\boldsymbol{\nabla}^{\mathrm{W}}\boldsymbol{v}$ and the symmetry of $\boldsymbol{\sigma}$.

6.2.3 The Navier–Stokes equations

A number of important phenomena in fluid mechanics are described by the Navier–Stokes equations. They are a statement of the dynamical effect of the externally applied forces and the internal forces of a fluid that we shall assume Newtonian. The internal forces are due to the pressure and the viscosity of the fluid. We consider a flow region $\Omega \in \mathbb{R}^{\mathrm{n_{sd}}}$, where $\mathrm{n_{sd}} = 2$ or 3. The domain Ω occupied by the fluid will be assumed bounded (finite size). The boundary $\Gamma = \partial\Omega$ of the fluid domain is assumed to be Lipschitz continuous, meaning that it is a closed and sufficiently regular surface. Then, the time-dependent flow of a viscous incompressible fluid is governed by the following form of the momentum equation (1.15) and the mass-conservation equation (1.11), called the *Navier–Stokes equations*:

$$\begin{aligned} &\rho\,(\boldsymbol{v}_t + (\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v}) = \boldsymbol{\nabla}\cdot\boldsymbol{\sigma} + \rho\boldsymbol{b} && \text{in } \Omega\times]0,T[, && (6.3a)\\ &\boldsymbol{\nabla}\cdot\boldsymbol{v} = 0 && \text{in } \Omega\times]0,T[. && (6.3b)\end{aligned}$$

Here, ρ is the fluid density and $\boldsymbol{b}$ the volume force per unit mass of fluid. Using Stokes' law (6.2), the equation of motion (6.3a) can be expressed in the more convenient form

$$\boldsymbol{v}_t + (\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v} - 2\nu\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}^{\mathrm{s}}\boldsymbol{v} + \boldsymbol{\nabla}p = \boldsymbol{b},$$

where $\nu = \mu/\rho$ is the fluid kinematic viscosity and p the kinematic pressure, that is pressure divided by density. Usually the previous equation is rewritten as

$$\boldsymbol{v}_t + (\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v} - \nu\boldsymbol{\nabla}^2\boldsymbol{v} - \nu\boldsymbol{\nabla}(\boldsymbol{\nabla}\cdot\boldsymbol{v}) + \boldsymbol{\nabla}p = \boldsymbol{b} \tag{6.4}$$

and referred as the velocity–pressure stress-divergence form.

The incompressibility condition (6.3b) is a consequence of the fact that in an incompressible continuum the rate of change of the mass density following the motion is zero, see equation (1.12). Under the incompressibility condition (6.3b), the momentum equation (6.3a), or (6.4), can be transformed to

$$\boldsymbol{v}_t + (\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v} - \nu\boldsymbol{\nabla}^2\boldsymbol{v} + \boldsymbol{\nabla}p = \boldsymbol{b} \qquad \text{in } \Omega\times]0,T[, \tag{6.5}$$

where $\boldsymbol{\nabla}^2\boldsymbol{v}$ denotes the Laplacian operator applied to the velocity vector. Its components are defined by

$$[\boldsymbol{\nabla}^2\boldsymbol{v}]_i = \sum_{j=1}^{\mathrm{n_{sd}}} \frac{\partial^2 v_i}{\partial x_j^2}.$$

The momentum equation is usually expressed in form (6.5), because, apart from the convective term $(\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v}$, these equations uncouple the velocity components.

The Navier–Stokes problem must be completed with suitable initial and boundary conditions to form a well-posed initial boundary value problem. Typical boundary conditions consist of prescribing the value $\boldsymbol{v}_D$ of the velocity on a portion Γ_D of the boundary:

$$\boldsymbol{v}(\boldsymbol{x},t) = \boldsymbol{v}_D(\boldsymbol{x},t), \qquad \boldsymbol{x}\in\Gamma_D, \quad t\in\,]0,T[, \tag{6.6}$$

and boundary traction $\boldsymbol{t}$ on the complementary portion Γ_N:

$$\boldsymbol{n}\cdot\boldsymbol{\sigma}(\boldsymbol{x},t) = \boldsymbol{t}(\boldsymbol{x},t), \qquad \boldsymbol{x}\in\Gamma_N, \quad t\in\,]0,T[, \tag{6.7}$$

where vector $\boldsymbol{n}$ denotes the unit outer normal to the boundary.

Remark 6.3 (Imposing boundary tractions). Note that for fluids obeying Stokes'law, equation (6.7) is equivalent to

$$\boldsymbol{n}\cdot\boldsymbol{\sigma} = -p\boldsymbol{n} + 2\nu\,\boldsymbol{n}\cdot\boldsymbol{\nabla}^{\mathrm{s}}\boldsymbol{v} = \boldsymbol{t}. \tag{6.8}$$

In 2D and with reference to a local system of Cartesian axes $(\boldsymbol{n}, \boldsymbol{\tau})$, it becomes

$$-p + 2\nu \frac{\partial v_{\boldsymbol{n}}}{\partial n} = t_{\boldsymbol{n}},$$

$$\nu\left(\frac{\partial v_{\boldsymbol{\tau}}}{\partial n} + \frac{\partial v_{\boldsymbol{n}}}{\partial \tau}\right) = t_{\boldsymbol{\tau}}.$$

The first condition involves both pressure p and the velocity gradient. It represents a condition of applied normal stress. The second one models applied tangential stresses. Such boundary conditions are frequently employed, in particular, to simulate a no-stress condition at outflow boundaries.

In the case of a time-dependent problem, the value of the velocity field at the initial time $t = 0$ must be given in Ω:

$$\boldsymbol{v}(\boldsymbol{x}, 0) = \boldsymbol{v}_0(\boldsymbol{x}), \qquad \boldsymbol{x} \in \Omega.$$

Moreover, the initial velocity field $\boldsymbol{v}_0$ must be divergence free, that is

$$\boldsymbol{\nabla} \cdot \boldsymbol{v}_0 = 0 \qquad \text{in } \Omega.$$

Remark 6.4. No initial condition needs to be specified for the fluid pressure. This is a consequence of the fact that no time derivative of pressure appears in the governing equations. When Dirichlet conditions are imposed everywhere on the boundary, $\Gamma_N = \emptyset$, pressure is only present by its gradient in the Navier–Stokes equations, and thus it is determined only up to an arbitrary constant. In this case, it is usual to impose the pressure average or the value of the pressure at one point to uniquely define the pressure field.

Remark 6.5. In the case of highly viscous flow, the convective terms in the Navier–Stokes equations can often be neglected if compared with the dominant viscous terms. The resulting equations are identical to the equations of isotropic incompressible elasticity and are called *equations of Stokes flow*:

$$\begin{cases} \boldsymbol{v}_t - \nu \boldsymbol{\nabla}^2 \boldsymbol{v} + \boldsymbol{\nabla} p = \boldsymbol{b} \\ \boldsymbol{\nabla} \cdot \boldsymbol{v} = 0 \end{cases} \qquad \text{in } \Omega \times]0, T[.$$

Remark 6.6. A dimensionless form of the Navier–Stokes equations is obtained by replacing the kinematic viscosity ν by the inverse of the flow Reynolds number defined as $Re = VL/\nu$, where V and L stand, respectively, for a characteristic velocity and a characteristic length of the flow. Reynolds number Re characterizes the ratio between the inertia forces and the viscous forces. In the absence of body forces, the Navier–Stokes equations in primitive variables read

$$\begin{cases} \boldsymbol{v}_t + (\boldsymbol{v} \cdot \boldsymbol{\nabla})\boldsymbol{v} - \dfrac{1}{Re} \boldsymbol{\nabla}^2 \boldsymbol{v} + \boldsymbol{\nabla} p = 0 \\ \boldsymbol{\nabla} \cdot \boldsymbol{v} = 0 \end{cases} \qquad \text{in } \Omega \times]0, T[.$$

Remark 6.7. In the case of 2D motion, $\boldsymbol{v} = (v_1, v_2, 0)$ and v_1 and v_2 are independent of x_3. From the incompressibility condition, $\boldsymbol{\nabla} \cdot \boldsymbol{v} = \partial v_1 / \partial x_1 + \partial v_2 / \partial x_2 = 0$, we may pose

$$v_1 = \frac{\partial \psi}{\partial x_2}, \qquad v_2 = -\frac{\partial \psi}{\partial x_1},$$

where the unknown scalar function $\psi(\boldsymbol{x}, t)$, called *stream function*, is defined by

$$\psi(\boldsymbol{x}, t) - \psi_O = \int_O^P \boldsymbol{\nabla}\psi \cdot d\boldsymbol{x} = \int_O^P [\boldsymbol{v} \times d\boldsymbol{x}]_3,$$

where ψ_0 is a constant and the line integral is taken along an arbitrary curve joining some reference point O to the point P with coordinates $\boldsymbol{x}$; note that $d\boldsymbol{x}$ represents a tangent vector to the arbitrary curve. It is common practice in computational fluid dynamics to provide a picture of a flow field by drawing a family of streamlines (curves at constant ψ and $\boldsymbol{v}$ tangent). In the finite element context, contours of streamlines, with specified intervals in ψ between pairs of neighboring streamlines, are obtained by numerical integration along the element sides or through the solution of a Poisson equation.

6.3 MAIN ISSUES IN INCOMPRESSIBLE FLOW PROBLEMS

Before introducing finite element techniques for the numerical solution of the Navier–Stokes equations, we wish to underline the main difficulties involved in the numerical simulation of incompressible flow problems.

A first difficulty is due to the presence of nonlinear and non-symmetric convective terms in the momentum equation (6.3a). Such difficulty increases with the value of the flow Reynolds number. High Reynolds number flows are convection dominated and, as repeatedly mentioned in the previous chapters, the standard Galerkin formulation is unstable. Stabilization techniques, such as SUPG, GLS, SGS or LS must be used to provide meaningful finite element solutions at high Reynolds numbers. These issues have been discussed in detail in previous chapters.

Another source of numerical difficulty is the incompressibility condition. The continuity equation for an incompressible fluid takes the peculiar form expressed in equation (6.3b). It consists of a constraint on the velocity field which must be divergence free. Then, the pressure has to be considered as a variable not related to any constitutive equation. Its presence in the momentum equation has the purpose of introducing an additional degree of freedom needed to satisfy the incompressibility constraint. The role of the pressure variable is thus to adjust itself instantaneously in order to satisfy the condition of divergence-free velocity. That is, the pressure is acting as a Lagrangian multiplier of the incompressibility constraint and thus there is a coupling between the velocity and the pressure unknowns.

Various formulations have been proposed in the literature to deal with incompressible flow problems. Here emphasis is placed on primitive variable formulations (retaining velocity and pressure as unknowns). Penalty methods are also discussed because they allow us to uncouple the determination of the velocity and pressure fields.

The primitive variable formulation, with both velocity and pressure unknowns, leads to so-called *mixed finite element methods*. Such methods present numerical difficulties caused by the *saddle-point* nature of the resulting variational problem; recall that pressure acts as a Lagrangian multiplier of the incompressibility constraint. Then, the algebraic system for the nodal values of velocity and pressure in a Galerkin formulation is governed by a partitioned matrix with a null submatrix on the diagonal. Solvability of the algebraic system depends on a proper choice of finite element spaces for velocity and pressure. They must satisfy a compatibility condition, the so-called LBB condition. There are, however, alternative finite element formulations that allow us to circumvent the LBB condition and enable the use of velocity–pressure pairs that are unstable in the standard Galerkin formulation.

The *penalty formulation* allows the elimination of the pressure variable from the Navier–Stokes problem through a relaxation of the incompressibility condition. The constraint $\boldsymbol{\nabla} \cdot \boldsymbol{v} = 0$ is replaced by $\boldsymbol{\nabla} \cdot \boldsymbol{v}^{(\lambda)} = -p^{(\lambda)}/\lambda$ where λ is a large parameter. This substitution eliminates the pressure gradient term from the momentum equation. Since it involves only velocities, the penalty method is computationally very attractive. A disadvantage is the presence of the penalty parameter λ, which may cause a loss of accuracy for excessively large values of λ, and prevent convergence to the actual solution for insufficiently large parameters.

As a starting point for the development of finite element models for incompressible flows, we introduce in the next section finite element spaces for velocity and pressure.

6.4 TRIAL SOLUTIONS AND WEIGHTING FUNCTIONS

The weak forms of Stokes and Navier–Stokes problems requires the introduction of classes of functions for the velocity field and the pressure field. With respect to velocity, $\boldsymbol{v}$, the space of trial solutions is denoted by $\boldsymbol{\mathcal{S}}$. As discussed in Section 1.5.2, candidate approximating functions must satisfy a priori Dirichlet boundary conditions, see (6.6), on Γ_D. The trial solution space $\boldsymbol{\mathcal{S}}$ containing the approximating functions for the velocity is thus characterized as follows:

$$\boldsymbol{\mathcal{S}} := \left\{ \boldsymbol{v} \in \boldsymbol{\mathcal{H}}^1(\Omega) \mid \boldsymbol{v} = \boldsymbol{v}_D \text{ on } \Gamma_D \right\} \quad \text{(trial solutions)}, \tag{6.9a}$$

where bold spaces contain vector functions such that each component is in the corresponding space of scalar functions. The weighting functions of the velocity, $\boldsymbol{w}$, belong to $\boldsymbol{\mathcal{V}}$. Functions in this class have the same characteristics as those in class $\boldsymbol{\mathcal{S}}$, except that the weighting functions are required to vanish on Γ_D where the velocity is prescribed. The class $\boldsymbol{\mathcal{V}}$ is thus symbolically defined by

$$\boldsymbol{\mathcal{V}} := \boldsymbol{\mathcal{H}}^1_{\Gamma_D}(\Omega) = \left\{ \boldsymbol{w} \in \boldsymbol{\mathcal{H}}^1(\Omega) \mid \boldsymbol{w} = \boldsymbol{0} \text{ on } \Gamma_D \right\} \quad \text{(weighting functions)}. \tag{6.9b}$$

Finally, we introduce a space of functions, denoted $\mathcal{Q}$, for the pressure. As we shall see, spatial derivatives of pressure do not appear in the weak form of the (Navier–

Stokes) problem; thus functions in $\mathcal{Q}$ are simply required to be square-integrable. Moreover, since there are no explicit boundary conditions on pressure, the space $\mathcal{Q}$,

$$\mathcal{Q} := \mathcal{L}_2(\Omega) \quad \text{(pressure space)}, \tag{6.9c}$$

suffices as the trial solution space and as the weighting function space. Note however that in the case of purely Dirichlet velocity boundary conditions, the pressure is defined up to a constant, see Remark 6.4. In such a case, its value must be prescribed at a given point of the domain Ω and the pressure space $\mathcal{Q}$ is thus replaced by $\mathcal{L}_2(\Omega)/\mathbb{R}$

Remark 6.8 (Construction of a solenoidal velocity field). Additional subspaces accounting for the incompressible constraint are required for the analysis of Stokes and Navier–Stokes problems. They are subspaces of

$$\mathcal{H}(\text{div};\Omega) := \{\boldsymbol{v} \in \mathcal{L}_2(\Omega) \mid \boldsymbol{\nabla}\cdot\boldsymbol{v} \in \mathcal{L}_2(\Omega)\}, \tag{6.10}$$

which is a Hilbert space endowed with the following norm:

$$\|\boldsymbol{v}\|^2_{\text{div};\Omega} = \int_\Omega \boldsymbol{v}\cdot\boldsymbol{v}\,d\Omega + \int_\Omega (\boldsymbol{\nabla}\cdot\boldsymbol{v})^2\,d\Omega = (\boldsymbol{v},\boldsymbol{v}) + (\boldsymbol{\nabla}\cdot\boldsymbol{v},\boldsymbol{\nabla}\cdot\boldsymbol{v}).$$

The inner product for vector-valued functions was introduced in Section 1.5.1.2. Consider, for instance, the space $\mathcal{D}(\Omega)$, or $C_0^\infty(\Omega)$, of infinitely differentiable functions having compact support in Ω. Then, the set of solenoidal vector fields of $\mathcal{D}(\Omega)$ is defined as

$$\mathcal{J}(\Omega) := \{\boldsymbol{v} \in \mathcal{D}(\Omega) \mid \boldsymbol{\nabla}\cdot\boldsymbol{v} = 0\}.$$

Likewise, the set of divergence-free velocity fields in $\mathcal{H}_0^1(\Omega)$ is

$$\mathcal{J}_0^1(\Omega) := \{\boldsymbol{v} \in \mathcal{H}_0^1(\Omega) \mid \boldsymbol{\nabla}\cdot\boldsymbol{v} = 0\}.$$

For instance, to this subspace belongs the solution of the weak form of the Stokes problem with homogenous boundary conditions. In particular, if the domain Ω is bounded and its boundary Γ is Lipschitz continuous, we can define

$$\mathcal{J}_0(\Omega) := \{\boldsymbol{v} \in \mathcal{L}_2(\Omega) \mid \boldsymbol{\nabla}\cdot\boldsymbol{v} = 0 \text{ in } \Omega,\ \boldsymbol{n}\cdot\boldsymbol{v} = 0 \text{ on } \Gamma\}.$$

Note that $\mathcal{H}(\text{div};\Omega)$ is the closure of $\mathcal{D}(\overline{\Omega})$ with respect to the norm $\|\cdot\|_{\text{div};\Omega}$ and $\mathcal{J}_0(\Omega)$ is the closure of $\mathcal{J}(\Omega)$ with respect to the $\mathcal{L}_2(\Omega)$ norm, see Section 1.5.1.1. Since $\mathcal{J}_0(\Omega)$ is a closed subspace of $\mathcal{L}_2(\Omega)$, we can define the decomposition

$$\mathcal{L}_2(\Omega) = \mathcal{J}_0(\Omega) \oplus \mathcal{J}_0^\perp(\Omega),$$

where the characterization of $\mathcal{J}_0^\perp(\Omega)$ derives from a theorem due to Ladyzhenskaya (1969) which states that

$$\mathcal{J}_0^\perp(\Omega) := \{\boldsymbol{w} \in \mathcal{L}_2(\Omega) \mid \boldsymbol{w} = \boldsymbol{\nabla}p,\ p \in \mathcal{H}^1(\Omega)\}.$$

This result is a consequence of the Helmholtz decomposition principle (see for instance Temam, 2001, Chap. I, Sec. 1) which states that any vector field $\boldsymbol{v}$ defined in Ω admits a unique orthogonal decomposition into the sum of a solenoidal field and the gradient of a scalar function. This characterization of $\mathcal{J}_0^{\perp}(\Omega)$ implies, in particular, that for all $\boldsymbol{w} \in \mathcal{L}_2(\Omega)$ orthogonal to any $\boldsymbol{u} \in \mathcal{J}_0(\Omega)$, that is $(\boldsymbol{w}, \boldsymbol{u}) = 0$, there exist p such that $\boldsymbol{w} = \boldsymbol{\nabla} p$. Note that the reciprocal holds.

As shown in Section 6.7, the projection of $\mathcal{L}_2(\Omega)$ onto $\mathcal{J}_0(\Omega)$, denoted by $\mathbb{P}$, is of prime importance for the construction of a solenoidal velocity field in the so-called fractional-step projection methods for solving incompressible flow problems.

6.5 STATIONARY STOKES PROBLEM

Before studying the Navier–Stokes equations, we shall study the steady Stokes problem. That is, we neglect the time-dependent and convective terms of the full Navier–Stokes equations. Two distinct formulations of Stokes problem are considered:

First, the momentum equation is written in terms of the Cauchy stress (also known as stress-divergence form). The Stokes' constitutive law is only invoked after setting the weak form. The advantage of this approach is that it can readily treat problems with fluid constitutive equations more general than the linear Stokes' law.

Second, the problem is directly formulated in terms of velocity and pressure. Use is made of Stokes' law and of the incompressibility condition to express the viscous term as the Laplacian of velocity. This is the standard form of the Stokes equations.

6.5.1 Formulation in terms of Cauchy stress

6.5.1.1 Strong form In differential form, a steady Stokes problem is stated as follows in terms of Cauchy stress: given the body force $\boldsymbol{b}$, prescribed velocities $\boldsymbol{v}_D$ on portion Γ_D of the boundary and imposed boundary tractions $\boldsymbol{t}$ on the remaining portion Γ_N, determine the velocity field $\boldsymbol{v}$ and the pressure field p such that

$$
\begin{aligned}
-\boldsymbol{\nabla}\cdot\boldsymbol{\sigma} &= \boldsymbol{b} && \text{in } \Omega && \text{(equilibrium),} && (6.11a)\\
\boldsymbol{\nabla}\cdot\boldsymbol{v} &= 0 && \text{in } \Omega && \text{(incompressibility),} && (6.11b)\\
\boldsymbol{v} &= \boldsymbol{v}_D && \text{on } \Gamma_D && \text{(Dirichlet b.c.),} && (6.11c)\\
\boldsymbol{n}\cdot\boldsymbol{\sigma} &= \boldsymbol{t} && \text{on } \Gamma_N && \text{(Neumann b.c.).} && (6.11d)
\end{aligned}
$$

Note, that a constitutive equation is needed to close the problem. That is, the Cauchy stress, $\boldsymbol{\sigma}$, must be related to velocity, $\boldsymbol{v}$, and pressure, p, that is $\boldsymbol{\sigma} = \boldsymbol{\sigma}(p, \boldsymbol{v})$, for instance by the linear Stokes' law (6.2).

6.5.1.2 Weak form The Stokes equations (6.11) define a *constrained equilibrium problem.* Techniques from optimization theory are available to treat such problems. In the finite element context standard methods are Lagrange multipliers, penalty, augmented Lagrangian and perturbed Lagrangian methods. A detailed account of such methods, including implementation details, can be found elsewhere (for instance, Glowinski, 1984; Carey and Oden, 1986; Glowinski and Le Tallec, 1989; Brezzi and Fortin, 1991; Belytschko et al., 2000; Quarteroni and Valli, 1994; Hughes, 2000; Gresho and Sani, 2000).

Here, the Lagrange multiplier method is used to solve the Stokes problem. In this method, the minimization of an objective function (the total potential energy) subject to constraints (the incompressibility) corresponds to the stationary points of the sum of the objective function and the constraints weighted by the Lagrange multipliers. This saddle point problem is further discussed in Remark 6.9.

The weak form can also be obtained multiplying the equation of motion (6.11a) by the velocity test function $\boldsymbol{w}$ and integrating by parts the stress term, thereby generating the natural boundary condition (6.11d) on Γ_N. Similarly, the incompressibility condition (6.11b) is multiplied by the pressure test function q and the result integrated over the computational domain Ω. Thus the weak form of Stokes problem becomes: given $\boldsymbol{b}$, $\boldsymbol{v}_D$ and the boundary traction $\boldsymbol{t}$, find the velocity field $\boldsymbol{v} \in \boldsymbol{\mathcal{S}}$ and the pressure field $p \in \mathcal{Q}$, such that for all velocity test functions $\boldsymbol{w} \in \boldsymbol{\mathcal{V}}$ and all pressure test functions $q \in \mathcal{Q}$,

$$\left\{\begin{aligned} &\int_\Omega \boldsymbol{\nabla w} : \boldsymbol{\sigma}\, d\Omega = \int_\Omega \boldsymbol{w}\cdot\boldsymbol{b}\, d\Omega + \int_{\Gamma_N} \boldsymbol{w}\cdot\boldsymbol{t}\, d\Gamma \\ &\int_\Omega q\, \boldsymbol{\nabla}\cdot\boldsymbol{v}\, d\Omega = 0. \end{aligned}\right. \tag{6.12}$$

To further clarify which equations are satisfied by the the weak form (6.12), we integrate by parts the term involving $\boldsymbol{\sigma}$ and use the divergence theorem. This gives, adding both equations,

$$0 = \int_\Omega \boldsymbol{w}\cdot\underbrace{(\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}+\boldsymbol{b})}_{\text{equilibrium}}\, d\Omega + \int_\Omega q\, \underbrace{\boldsymbol{\nabla}\cdot\boldsymbol{v}}_{\text{incomp.}}\, d\Omega - \int_{\Gamma_N} \boldsymbol{w}\cdot\underbrace{(\boldsymbol{n}\cdot\boldsymbol{\sigma}-\boldsymbol{t})}_{\text{Neumann b.c.}}\, d\Gamma.$$

The weighting functions $\boldsymbol{w}$ and q are arbitrary, thus the solution of the variational problem (6.12) verifies the strong form (6.11) of the steady Stokes problem.

If the fluid constitutive equation can be expressed as

$$\sigma_{ij} = -p\,\delta_{ij} + s_{ij}(\boldsymbol{v}),$$

where the deviatoric part of $\boldsymbol{\sigma}$ is denoted by s_{ij}. The variational problem (6.12) can be rewritten in compact form in terms of velocity and pressure, find $(\boldsymbol{v}, p) \in \boldsymbol{\mathcal{S}} \times \mathcal{Q}$, such that

$$\left\{\begin{aligned} &a(\boldsymbol{w},\boldsymbol{v}) + b(\boldsymbol{w},p) = (\boldsymbol{w},\boldsymbol{b}) + (\boldsymbol{w},\boldsymbol{t})_{\Gamma_N} && \forall \boldsymbol{w} \in \boldsymbol{\mathcal{V}} \\ &b(\boldsymbol{v},q) = 0 && \forall q \in \mathcal{Q} \end{aligned}\right. \tag{6.13}$$

or equivalently, find $(\boldsymbol{v}, p) \in \boldsymbol{\mathcal{S}} \times \mathcal{Q}$, such that

$$a(\boldsymbol{w},\boldsymbol{v}) + b(\boldsymbol{w},p) + b(\boldsymbol{v},q) = (\boldsymbol{w},\boldsymbol{b}) + (\boldsymbol{w},\boldsymbol{t})_{\Gamma_N} \quad \forall (\boldsymbol{w},q) \in \boldsymbol{\mathcal{V}} \times \mathcal{Q}$$

with the following definitions of the forms,

$$a(\boldsymbol{w},\boldsymbol{v}) = \int_\Omega w_{(i,j)}\, s_{ij}\, d\Omega \quad \text{and} \quad b(\boldsymbol{v},q) = -\int_\Omega q\boldsymbol{\nabla}\cdot\boldsymbol{v}\, d\Omega,$$

see also (1.27). Note that the symmetry of the stress tensor has been used

$$\boldsymbol{\nabla w}:\boldsymbol{\sigma} = \sum_{i,j=1}^{\mathrm{n_{sd}}} \frac{\partial w_i}{\partial x_j}\sigma_{ij} = \sum_{i,j=1}^{\mathrm{n_{sd}}} w_{(i,j)}\,\sigma_{ij}.$$

Moreover, if the Cauchy stress $\boldsymbol{\sigma}$ is assumed given by the linear Stokes' law (6.2),

$$s_{ij} = C_{ijkl}\, v_{(k,l)}, \quad \text{where } C_{ijkl} = \nu\,(\delta_{ik}\,\delta_{jl} + \delta_{il}\,\delta_{jk}),$$

and this allows us to write the bilinear form $a(\boldsymbol{w},\boldsymbol{v})$ as

$$a(\boldsymbol{w},\boldsymbol{v}) = \int_\Omega w_{(i,j)}\, C_{ijkl}\, v_{(k,l)}\, d\Omega. \tag{6.14}$$

Remark 6.9 (Saddle-point nature of the solution). For homogeneous Dirichlet boundary conditions, $\boldsymbol{v} = 0$ on $\Gamma_D = \partial\Omega$ and $\Gamma_N = \emptyset$, equation (6.13) reads

$$a(\boldsymbol{w},\boldsymbol{v}) + b(\boldsymbol{w},p) + b(\boldsymbol{v},q) = (\boldsymbol{w},\boldsymbol{b}), \qquad \forall(\boldsymbol{w},q)\in\boldsymbol{\mathcal{V}}\times\mathcal{Q}.$$

It can be shown (see for instance Temam, 2001) that the solution $(\boldsymbol{v},p)$ of the previous Stokes problem is a *saddle point* of the Lagrangian functional

$$I(\boldsymbol{w},q) := \frac{1}{2}a(\boldsymbol{w},\boldsymbol{w}) + b(\boldsymbol{w},q) - (\boldsymbol{w},\boldsymbol{b}) \qquad \forall(\boldsymbol{w},q)\in\boldsymbol{\mathcal{V}}\times\mathcal{Q};$$

that is,

$$I(\boldsymbol{v},q) \le I(\boldsymbol{v},p) \le I(\boldsymbol{w},p) \qquad \forall(\boldsymbol{w},q)\in\boldsymbol{\mathcal{V}}\times\mathcal{Q},$$

or equivalently,

$$I(\boldsymbol{v},p) = \min_{\boldsymbol{w}\in\boldsymbol{\mathcal{V}}}\max_{q\in\mathcal{Q}} I(\boldsymbol{w},q)$$

Remark 6.10 (Stress–strain form). It is sometimes interesting to express the bilinear form $a(\boldsymbol{w},\boldsymbol{v})$ in equation (6.14) in terms of the *strain rate vector*

$$\dot{\boldsymbol{\epsilon}}(\boldsymbol{v})^T = \left(\frac{\partial v_1}{\partial x_1}, \frac{\partial v_2}{\partial x_2}, \frac{\partial v_3}{\partial x_3}, \frac{\partial v_1}{\partial x_2} + \frac{\partial v_2}{\partial x_1}, \frac{\partial v_2}{\partial x_3} + \frac{\partial v_3}{\partial x_2}, \frac{\partial v_3}{\partial x_1} + \frac{\partial v_1}{\partial x_3}\right),$$

containing the six strain rate components relevant in 3D analysis. To express $a(\boldsymbol{w},\boldsymbol{v})$ in terms of the strain rate vector $\dot{\boldsymbol{\epsilon}}(\boldsymbol{v})$, we define for 3D the constitutive matrix

$$\mathbf{C}_\nu = \nu\begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0\\ 0 & 2 & 0 & 0 & 0 & 0\\ 0 & 0 & 2 & 0 & 0 & 0\\ 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{6.15}$$

It can then be shown that $w_{(i,j)}\, C_{ijkl}\, v_{(k,l)} = \dot{\boldsymbol{\epsilon}}(\boldsymbol{w})^T\, \mathbf{C}_\nu\, \dot{\boldsymbol{\epsilon}}(\boldsymbol{v})$, so that the bilinear form (6.14) can be rewritten as

$$a(\boldsymbol{w},\boldsymbol{v}) = \int_\Omega \dot{\boldsymbol{\epsilon}}(\boldsymbol{w})^T\, \mathbf{C}_\nu\, \dot{\boldsymbol{\epsilon}}(\boldsymbol{v})\, d\Omega.$$

Note that $\mathbf{C}_\nu\, \dot{\boldsymbol{\epsilon}}(\boldsymbol{v})$ is the deviatoric stress vector $\boldsymbol{s}(\boldsymbol{v})$ with components

$$\boldsymbol{s}(\boldsymbol{v})^T = (s_{11}, s_{22}, s_{33}, s_{12}, s_{23}, s_{31})\,.$$

It follows that

$$a(\boldsymbol{w},\boldsymbol{v}) = \int_\Omega \dot{\boldsymbol{\epsilon}}(\boldsymbol{w})^T\, \boldsymbol{s}(\boldsymbol{v})\, d\Omega$$

in complete analogy with linear elasticity (see the excellent presentation by Belytschko et al., 2000, App. 1).

6.5.2 Formulation in terms of velocity and pressure

6.5.2.1 Strong form The formulation of the Stokes problem in terms of Cauchy stress has the advantage that it is applicable to arbitrary fluid constitutive relations. When the linear Stokes' law can be invoked, it is preferable to start from a strong form of written in terms of velocity and pressure because in this form the velocity components are uncoupled. The steady Stokes problem is usually restated as follows:

$$\begin{array}{rlll}
-\nu\boldsymbol{\nabla}^2\boldsymbol{v} + \boldsymbol{\nabla}p = \boldsymbol{b} & \text{in } \Omega & \text{(equilibrium)}, & (6.16\text{a}) \\
\boldsymbol{\nabla}\cdot\boldsymbol{v} = 0 & \text{in } \Omega & \text{(incompressibility)}, & (6.16\text{b}) \\
\boldsymbol{v} = \boldsymbol{v}_D & \text{on } \Gamma_D & \text{(Dirichlet b.c.)}, & (6.16\text{c}) \\
-p\boldsymbol{n} + \nu(\boldsymbol{n}\cdot\boldsymbol{\nabla})\boldsymbol{v} = \boldsymbol{t} & \text{on } \Gamma_N & \text{(Neumann b.c.)}. & (6.16\text{d})
\end{array}$$

Having used the kinematic viscosity $\nu = \mu/\rho$ and p denoting the dynamic pressure, the ("dynamic" or "scaled") Cauchy stress tensor (normalized by density), if needed, is also given by Stokes' law (6.2), $\boldsymbol{\sigma} = -p\mathbf{I} + 2\nu\,\boldsymbol{\nabla}^{\mathrm{S}}\boldsymbol{v}$.

The equilibrium equation (6.16a) is obtained from (6.11a) using the Stokes' law (6.2) and the incompressibility condition (6.11b), see also the velocity–pressure stress-divergence form (6.4).

In this case, however, $\boldsymbol{t}$ does not correspond to boundary tractions. It is usually called a "pseudo-traction".

> **Remark 6.11 (Laplace or stress-divergence forms).** As noted previously, in the Neumann boundary condition defined by (6.16d) $\boldsymbol{t}$ is not a boundary traction but a "pseudo-traction". Boundary tractions are clearly enforced if the formulation is in terms of Cauchy stress, see for instance (6.11d) or Remark 6.3. In fact, (6.8) is different from (6.16d), that is
>
> $$\boldsymbol{n}\cdot\boldsymbol{\sigma} = -p\boldsymbol{n} + 2\nu\,\boldsymbol{n}\cdot\boldsymbol{\nabla}^{\mathrm{S}}\boldsymbol{v} \neq -p\boldsymbol{n} + \nu(\boldsymbol{n}\cdot\boldsymbol{\nabla})\boldsymbol{v}.$$

This inequality becomes an equality if $(\partial v_j / \partial x_i) n_j = 0$ for $i = 1, \ldots, \mathrm{n_{sd}}$, which, in the case of incompressible fluids, is equivalent to $(\boldsymbol{\tau} \cdot \boldsymbol{\nabla}) \boldsymbol{v} = \boldsymbol{0}$ for all vectors $\boldsymbol{\tau}$ orthogonal to $\boldsymbol{n}$, that is $(\partial v_i / \partial x_j) \tau_j = 0$ for $i = 1, \ldots, \mathrm{n_{sd}}$.

Nevertheless, "pseudo-tractions" are the natural boundary conditions for a weak (velocity–pressure) Laplace formulation and thus, it is often a convenient form to impose open/artificial boundary conditions (see Section 6.8.3).

6.5.2.2 Weak form Applying the divergence theorem to the pressure gradient term and to the second-derivative (viscous) term, the weak formulation of Stokes problem (6.16) is also given by (6.13), namely find $(\boldsymbol{v}, p) \in \boldsymbol{\mathcal{S}} \times \mathcal{Q}$, such that

$$a(\boldsymbol{w}, \boldsymbol{v}) + b(\boldsymbol{w}, p) + b(\boldsymbol{v}, q) = (\boldsymbol{w}, \boldsymbol{b}) + (\boldsymbol{w}, \boldsymbol{t})_{\Gamma_N} \quad \forall (\boldsymbol{w}, q) \in \boldsymbol{\mathcal{V}} \times \mathcal{Q},$$

with the following new definition of the viscous bilinear term:

$$a(\boldsymbol{w}, \boldsymbol{v}) = \int_\Omega \boldsymbol{\nabla} \boldsymbol{w} : \nu \boldsymbol{\nabla} \boldsymbol{v} \, d\Omega. \tag{6.17}$$

Remark 6.12. Depending on the definition of the bilinear form $a(\cdot, \cdot)$, either (6.14) or (6.17), the interpretation of $\boldsymbol{t}$ may be different. If the stress-divergence form is employed, see (6.4) or (6.11a), the viscous bilinear form for an incompressible Newtonian fluid is (6.14) and $\boldsymbol{t}$ are boundary tractions. If the Laplace formulation for the viscous term is preferred, (6.17) defines the viscous bilinear form and $\boldsymbol{t}$ are "pseudo-tractions", see Remark 6.11. Note that these remarks are also pertinent to the Navier-Stokes equations.

6.5.3 Galerkin formulation

The Galerkin formulation of the Stokes problem leads to a *mixed finite element method.* We need to introduce local approximations for both the velocity components v_i^h and pressure p^h, as well as for their associated weighting functions w_i^h and q^h. We denote by $\boldsymbol{\mathcal{S}}^h$ and $\boldsymbol{\mathcal{V}}^h$ the finite dimensional subspaces of $\boldsymbol{\mathcal{S}}$ and $\boldsymbol{\mathcal{V}}$, and $\mathcal{Q}^h$ the finite dimensional subspace of $\mathcal{Q}$.

The velocity approximation $\boldsymbol{v}^h \in \boldsymbol{\mathcal{S}}^h$ admits the representation $\boldsymbol{v}^h = \boldsymbol{u}^h + \boldsymbol{v}_D^h$, where the field $\boldsymbol{v}_D^h$ satisfies (approximates) the Dirichlet boundary condition, see (6.11c), on Γ_D. Thus, the auxiliary velocity $\boldsymbol{u}^h$ belongs to the same space as the test function $\boldsymbol{w}^h$, namely $\boldsymbol{\mathcal{V}}^h$. The Galerkin formulation of the Stokes problem, see (6.13), may then be stated as follows: given $\boldsymbol{b}$, $\boldsymbol{v}_D$ and $\boldsymbol{t}$, find the auxiliary velocity $\boldsymbol{u}^h \in \boldsymbol{\mathcal{V}}^h$ and the pressure field $p^h \in \mathcal{Q}^h$ for all $(\boldsymbol{w}^h, q^h) \in \boldsymbol{\mathcal{V}}^h \times \mathcal{Q}^h$, such that

$$\left\{ \begin{aligned} & a(\boldsymbol{w}^h, \boldsymbol{u}^h) + b(\boldsymbol{w}^h, p^h) = (\boldsymbol{w}^h, \boldsymbol{b}^h) + (\boldsymbol{w}^h, \boldsymbol{t}^h)_{\Gamma_N} - a(\boldsymbol{w}^h, \boldsymbol{v}_D^h) \\ & b(\boldsymbol{u}^h, q^h) = -b(\boldsymbol{v}_D^h, q^h). \end{aligned} \right. \tag{6.18}$$

As will become clear in Section 6.5.6, the success of a mixed finite element formulation crucially depends on a proper choice of the local interpolations of the velocity and the pressure. Since the pressure gradient does not enter the weak form, pressure is not required to be continuous at the interface between elements, a condition which is mandatory for the interpolation of velocity.

The next step in the Galerkin formulation consists in approximating the velocity components $v_i^h = u_i^h + v_{Di}^h$ in terms of shape functions and associated nodal values. Following the terminology introduced in Section 1.5.5, we denote by $\eta = \{1, 2, \ldots, \mathrm{n_{np}}\}$ the set of global velocity node numbers in the finite element mesh. Furthermore, we denote by $\eta_{Di} \subset \eta$ the subset of velocity nodes belonging to the Dirichlet portion of the boundary where component i of the velocity is prescribed.

The velocity components are then approximated as follows:

$$\left.\begin{aligned} u_i^h(\boldsymbol{x}) &= \sum_{A \in \eta \backslash \eta_{Di}} N_A(\boldsymbol{x})\, \mathrm{u}_{iA} \\ v_{Di}^h(\boldsymbol{x}) &= \sum_{A \in \eta_{Di}} N_A(\boldsymbol{x})\, v_{Di}(\boldsymbol{x}_A) \end{aligned}\right\} \quad \text{(no sum on } i\text{)}, \tag{6.19a}$$

where N_A is the shape function associated with global node number A, and u_{iA} the value of u_i^h at node number A. Recall that each velocity node has $\mathrm{n_{sd}}$ degrees of freedom. Thus, in the Galerkin formulation the test functions are defined such that

$$w_i^h \in \mathcal{V}_i^h := \operatorname*{span}_{A \in \eta \backslash \eta_{Di}} \{N_A\}.$$

The vector version of (6.19a) is defined with the aid of the canonical basis of $\mathbb{R}^{\mathrm{n_{sd}}}$, namely $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_{\mathrm{n_{sd}}}\}$. For $\mathrm{n_{sd}} = 3$ the expressions of $\boldsymbol{e}_1$, $\boldsymbol{e}_2$ and $\boldsymbol{e}_3$ are the following:

$$\boldsymbol{e}_1 = (1, 0, 0)^T, \quad \boldsymbol{e}_2 = (0, 1, 0)^T, \quad \boldsymbol{e}_3 = (0, 0, 1)^T.$$

The vector version of the interpolation, (6.19a), and test functions are

$$\begin{aligned} \boldsymbol{u}^h(\boldsymbol{x}) &= \sum_{i=1}^{\mathrm{n_{sd}}} u_i^h(\boldsymbol{x})\, \boldsymbol{e}_i = \sum_{i=1}^{\mathrm{n_{sd}}} \sum_{A \in \eta \backslash \eta_{Di}} N_A(\boldsymbol{x})\, \mathrm{u}_{iA}\, \boldsymbol{e}_i, \\ \boldsymbol{w}^h(\boldsymbol{x}) &= \sum_{i=1}^{\mathrm{n_{sd}}} w_i^h(\boldsymbol{x})\, \boldsymbol{e}_i. \end{aligned}$$

The pressure field is interpolated using a possibly different set of pressure nodes denoted by $\hat{\eta}$ and the shape functions $\hat{N}_{\hat{A}}$, as

$$p^h(\boldsymbol{x}) = \sum_{\hat{A} \in \hat{\eta}} \hat{N}_{\hat{A}}(\boldsymbol{x})\, \mathrm{p}_{\hat{A}},$$

where $\hat{A}$ is the global pressure node number and $\mathrm{p}_{\hat{A}}$ is the pressure value at $\hat{A}$. As previously done for the velocity, we shall use small letters, namely $\hat{a}$ and $\hat{b}$, to denote element pressure node numbers; they range from 1 to $\hat{\mathrm{n}}_{\mathrm{en}}$ pressure nodes. Similarly, the weighting function q^h for the pressure is expressed as

$$q^h \in \mathcal{Q}^h := \operatorname*{span}_{\hat{A} \in \hat{\eta}} \{\hat{N}_{\hat{A}}\}. \tag{6.19b}$$

Table 6.1 Correspondence between matrices and vectors in (6.21) and the variational forms.

Matrix/Vector	Corresponding term of equation (6.13) or (6.18)	
$\mathbf{K}$	$a(\boldsymbol{w},\boldsymbol{u})$	$\int_\Omega \dot{\boldsymbol{\epsilon}}(\boldsymbol{w})^T \mathbf{C}_\nu\, \dot{\boldsymbol{\epsilon}}(\boldsymbol{v})\, d\Omega$ or $\int_\Omega \boldsymbol{\nabla} \boldsymbol{w} : \nu \boldsymbol{\nabla} \boldsymbol{v}\, d\Omega$
$\mathbf{G}$	$b(\boldsymbol{w},p)$	$-\int_\Omega p \boldsymbol{\nabla} \cdot \boldsymbol{w}\, d\Omega$
$\mathbf{G}^T$	$b(\boldsymbol{u},q)$	$-\int_\Omega q \boldsymbol{\nabla} \cdot \boldsymbol{u}\, d\Omega$
$\mathbf{f}$		$(\boldsymbol{w},\boldsymbol{b}) + (\boldsymbol{w},\boldsymbol{t})_{\Gamma_N} - a(\boldsymbol{w},\boldsymbol{v}_D)$
$\mathbf{h}$		$-b(\boldsymbol{v}_D,q)$

6.5.4 Matrix problem

To derive the matrix problem governing Stokes flow in the Galerkin formulation, we introduce the above expressions for the trial and weighting functions into the Galerkin form (6.18). This results in the following set of nodal equations for the unknown components of the velocity field: for each $A \in \eta \setminus \eta_{Di}$ and $1 \le i \le \mathrm{n_{sd}}$

$$\sum_{j=1}^{\mathrm{n_{sd}}} \Big\{ \sum_{B \in \eta \setminus \eta_{Dj}} a(N_A\, \boldsymbol{e}_i, N_B\, \boldsymbol{e}_j) \mathrm{u}_{jB} \Big\} + \sum_{\hat{A} \in \hat{\eta}} b(N_A\, \boldsymbol{e}_i, \hat{N}_{\hat{A}})\, \mathrm{p}_{\hat{A}}$$
$$= (N_A\, \boldsymbol{e}_i, \boldsymbol{b}^h) + (N_A\, \boldsymbol{e}_i, \boldsymbol{t}^h)_{\Gamma_N} - \sum_{j=1}^{\mathrm{n_{sd}}} \Big\{ \sum_{B \in \eta_{Dj}} a(N_A\, \boldsymbol{e}_i, N_B\, \boldsymbol{e}_j)\, v_{Dj} \Big\}. \quad (6.20)$$

Note that the viscous bilinear form can be defined either by (6.14) or by (6.17). Similarly, the following set of discrete equations corresponding to the incompressibility constraint is obtained: for every $\hat{A} \in \hat{\eta}$

$$\sum_{i=1}^{\mathrm{n_{sd}}} \Big\{ \sum_{B \in \eta \setminus \eta_{Di}} b(N_B\, \boldsymbol{e}_i, \hat{N}_{\hat{A}}) \mathrm{u}_{iB} \Big\} = - \sum_{i=1}^{\mathrm{n_{sd}}} \Big\{ \sum_{B \in \eta_{Di}} b(N_B\, \boldsymbol{e}_i, \hat{N}_{\hat{A}}) v_{Di} \Big\}.$$

From these equations one finds that the matrix system which governs the discrete Stokes problem assumes the following partitioned form:

$$\begin{pmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{h} \end{pmatrix}. \quad (6.21)$$

The matrices and vectors in this algebraic system of equations and the corresponding terms in the variational equation (6.18) are identified in Table 6.1.

Matrix $\mathbf{K}$ is the *viscosity matrix* and results from the discretization of $a(\cdot,\cdot)$, see (6.14) or (6.17). It is obtained, as usual, from the assembly of element contributions,

$$\mathbf{K} = \overset{e}{\mathbf{A}} \mathbf{K}^e.$$

Each velocity node has as many degrees of freedom as spatial dimensions. Thus, $\mathbf{K}^e$ is a square matrix whose dimension is given by $\mathrm{n_{ee}} = \mathrm{n_{en}} \cdot \mathrm{n_{sd}}$, where $\mathrm{n_{en}}$ is the number of element nodes and $\mathrm{n_{sd}}$ is the number of space dimensions. Each element of this matrix, K^e_{rs}, is characterized by indices r and s (element equation numbers) related to the element node numbers, a and b, and the components i and j of the velocity:

$$\begin{matrix} r = \mathrm{n_{sd}}(a-1)+i \\ s = \mathrm{n_{sd}}(b-1)+j \end{matrix} \quad \text{with} \quad \begin{matrix} 1 \le r, s \le \mathrm{n_{ee}} \\ 1 \le a, b \le \mathrm{n_{en}} \\ 1 \le i, j \le \mathrm{n_{sd}}. \end{matrix} \tag{6.22}$$

Each element of the viscosity matrix is defined as $K^e_{rs} = e_i^T \mathbf{K}^e_{ab}\, e_j$, where the particular expression of $\mathbf{K}^e_{ab}$ depends on the formulation chosen (recall the relation between r and s and a, b, i and j given in the previous equation). In 3D, a formulation based on the Cauchy stress gives

$$\mathbf{K}^e_{ab} = \int_{\Omega^e} \mathbf{B}_a^T \mathbf{C}_\nu \mathbf{B}_b \, d\Omega, \quad \text{with}$$
$$\mathbf{B}_a^T = \begin{pmatrix} \dfrac{\partial N_a}{\partial x_1} & 0 & 0 & \dfrac{\partial N_a}{\partial x_2} & 0 & \dfrac{\partial N_a}{\partial x_3} \\ 0 & \dfrac{\partial N_a}{\partial x_2} & 0 & \dfrac{\partial N_a}{\partial x_1} & \dfrac{\partial N_a}{\partial x_3} & 0 \\ 0 & 0 & \dfrac{\partial N_a}{\partial x_3} & 0 & \dfrac{\partial N_a}{\partial x_2} & \dfrac{\partial N_a}{\partial x_1} \end{pmatrix}; \tag{6.23}$$

where the strain rate–velocity matrix $\mathbf{B}$ is identical to the strain–displacement matrix in elasticity, and the matrix $\mathbf{C}_\nu$ is defined in (6.15). In fact, the viscosity matrix is equivalent to the stiffness matrix in elasticity; it is symmetric and positive definite. A velocity–pressure formulation induces a matrix with the same properties and the following expression:

$$\mathbf{K}^e_{ab} = \int_{\Omega^e} \mathbf{B}_a^T \nu \, \mathbf{B}_b \, d\Omega, \quad \text{with}$$
$$\mathbf{B}_a^T = \begin{pmatrix} \dfrac{\partial N_a}{\partial x_1} & \dfrac{\partial N_a}{\partial x_2} & \dfrac{\partial N_a}{\partial x_3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \dfrac{\partial N_a}{\partial x_1} & \dfrac{\partial N_a}{\partial x_2} & \dfrac{\partial N_a}{\partial x_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \dfrac{\partial N_a}{\partial x_1} & \dfrac{\partial N_a}{\partial x_2} & \dfrac{\partial N_a}{\partial x_3} \end{pmatrix}. \tag{6.24}$$

Matrix $\mathbf{G}$ is the discrete *gradient operator*, and $\mathbf{G}^T$ is the discrete *divergence operator*. Matrix $\mathbf{G}$ arises from the discretization of the term $b(w^h, p^h)$ in the Galerkin variational form (6.18):

$$\mathbf{G} = \mathop{\mathbf{A}}^{e} \mathbf{G}^e, \qquad G^e_{r\hat{a}} = b(N_a\, e_i, \hat{N}_{\hat{a}})_{\Omega^e} = -\big(\hat{N}_{\hat{a}}, \boldsymbol{\nabla}\cdot(N_a\, e_i)\big)_{\Omega^e},$$

where $\hat{a}$ is an element pressure node number and r an element equation number. Note that $\boldsymbol{\nabla}\cdot(N_a e_i) = \partial N_a / \partial x_i$.

Vectors $\mathbf{f}$ and $\mathbf{h}$ incorporate the effect of the velocity v_D prescribed on the Dirichlet portion Γ_D of the boundary. Vector $\mathbf{f}$ also includes the contributions emanating from the applied body force $\boldsymbol{b}$ and the prescribed traction $\boldsymbol{t}$ on the Neumann portion of the boundary. They take the form

$$
\begin{aligned}
\mathbf{f} &= \overset{e}{\mathbf{A}}\,\mathbf{f}^e, \quad f_r^e = \left(N_a, b_i^h\right)_{\Omega^e} + \left(N_a, t_i^h\right)_{\Gamma_N^e} - \sum_{q=1}^{\mathrm{n_{ee}}} K_{rs}^e\, v_{Ds}^e, \\
\mathbf{h} &= \overset{e}{\mathbf{A}}\,\mathbf{h}^e, \quad h_{\hat{a}}^e = \sum_{p=1}^{\mathrm{n_{ee}}} \left(\hat{N}_{\hat{a}}, \boldsymbol{\nabla}\cdot(N_a \boldsymbol{e}_i)\right)_{\Omega^e} v_{Dr}^e.
\end{aligned}
\tag{6.25}
$$

Here, $v_{Ds}^e = v_{Djb}^e$ if v_{Dj} is prescribed at node b, and equals zero otherwise.

6.5.5 Solvability condition and solution procedure

The partitioned system (6.21) could in principle be solved in several ways. These include global methods in which the original system is solved iteratively, for instance using the Uzawa method (Brezzi and Fortin, 1991), as well as the so-called pressure-matrix method in which an independent linear system is generated for the pressure.

However, first of all, given the peculiar form of the Stokes system (6.21) with a null submatrix on the diagonal, the following critical question can be raised: under which condition can the algebraic system governing the velocity and pressure be safely solved? It can be shown that, provided the kernel (null space) of matrix $\mathbf{G}$ is zero, the global matrix (6.21) is non-singular, that is $\mathbf{u}$ and $\mathbf{p}$ are uniquely defined. Recall that the kernel of $\mathrm{n_{eq}} \times \hat{\mathrm{n}}_{\mathrm{eq}}$ matrix $\mathbf{G}$ is the set of all vectors $\mathbf{q}$ ($\hat{\mathrm{n}}_{\mathrm{eq}}$ components) such that $\mathbf{Gq} = \mathbf{0}$, namely

$$
\ker \mathbf{G} := \{\mathbf{q} \mid \mathbf{q} \in \mathbb{R}^{\hat{\mathrm{n}}_{\mathrm{eq}}} \text{ and } \mathbf{Gq} = \mathbf{0}\},
$$

where $\mathrm{n_{eq}}$ is the number of velocity unknowns, $\mathrm{n_{eq}} = \mathrm{n_{np}} \cdot \mathrm{n_{sd}} - \sum_{i=1}^{\mathrm{n_{sd}}} \dim(\eta_{Di})$, and $\hat{\mathrm{n}}_{\mathrm{eq}}$ is the number of pressure unknowns, $\hat{\mathrm{n}}_{\mathrm{eq}} = \dim(\hat{\eta}) = \dim(\mathcal{Q}^h)$.

In turn, to have $\ker \mathbf{G} = \{\mathbf{0}\}$, the velocity and pressure interpolations must satisfy a compatibility condition, called the LBB condition, to be discussed in the next section. Inappropriate combinations of velocity and pressure interpolations, namely pairs of spaces not satisfying the LBB compatibility condition, may render the discrete divergence matrix, $\mathbf{G}^T$, rank deficient.

Provided the kernel of matrix $\mathbf{G}$ is zero, the following pressure-matrix method can be employed to solve the partitioned Stokes system. From the first equation in (6.21), one obtains

$$
\mathbf{u} = \mathbf{K}^{-1}\left(\mathbf{f} - \mathbf{G}\mathbf{p}\right). \tag{6.26}
$$

The introduction of this result in the second equation of (6.21) yields an algebraic system for the pressure based on the so-called Schur complement matrix:

$$
\left(\mathbf{G}^T\mathbf{K}^{-1}\mathbf{G}\right)\mathbf{p} = \mathbf{G}^T\mathbf{K}^{-1}\mathbf{f} - \mathbf{h}. \tag{6.27}
$$

Note that the pressure matrix $\left(\mathbf{G}^T\mathbf{K}^{-1}\mathbf{G}\right)$ is symmetric, but full due to the presence of $\mathbf{K}^{-1}$. It is positive definite if $\ker \mathbf{G} = \{\mathbf{0}\}$. Once the pressure nodal values, $\mathbf{p}$, are determined from (6.27), the velocity field is obtained using equation (6.26). Usually this approach is combined with an iterative solver for the resulting linear systems and therefore ends up with two nested iterative loops which make the solution strategy quite expensive. To reduce the solution cost, alternative solution strategies have been proposed. They are described in detail by Quarteroni and Valli (1994) and Gresho and Sani (2000). Some of them are based on splitting the original problem into successive subproblems cheaper to solve. An example of an alternative solution strategy is presented in Section 6.7 in connection with the modeling of time-dependent Navier–Stokes problems.

6.5.6 The LBB compatibility condition

We have seen that the symmetric pressure matrix $(\mathbf{G}^T\mathbf{K}^{-1}\mathbf{G})$ is positive definite only if $\ker \mathbf{G} = \{\mathbf{0}\}$. If this is the case, the partitioned matrix (6.21) is non-singular and delivers uniquely defined velocity and pressure fields. If this is not the case, a stable and convergent velocity field might be obtained, but the pressure field is likely to present spurious and oscillatory results.

Ladyzhenskaya (1969), Babuška (1970/71) and Brezzi (1974) have determined the compatibility condition, known as the LBB (or inf-sup) condition that continuous and discrete spaces must satisfy to guarantee the stability of a mixed method. A complete discussion of the theory of mixed methods is beyond the scope of the present text, see Girault and Raviart (1986) or Brezzi and Fortin (1991) for more details. We shall limit ourselves to give a flavor of the numerical difficulties associated with these methods.

The LBB condition states that velocity and pressure spaces cannot be chosen arbitrarily, a link between them is necessary. To illustrate this let us consider once more the partitioned matrix system (6.21) governing steady Stokes flow, namely

$$\begin{pmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{h} \end{pmatrix},$$

where for the present purposes $\mathbf{K}$ is a square matrix ($\mathrm{n}_{\mathrm{eq}} \times \mathrm{n}_{\mathrm{eq}}$), $\mathbf{G}$ a rectangular matrix ($\mathrm{n}_{\mathrm{eq}} \times \hat{\mathrm{n}}_{\mathrm{eq}}$), and the vectors $\mathbf{u}$, $\mathbf{p}$, $\mathbf{f}$ and $\mathbf{h}$ have the corresponding dimensions. Note that to ensure a unique solution, the matrix in the previous system, equation (6.21), should have a *full rank*. Recall that the rank of a matrix is the order of the largest square array within that matrix, formed by deleting certain rows and columns, whose determinant does not vanish. In the present case, since $\mathbf{K}$ is regular ($\operatorname{rank}\mathbf{K} = \mathrm{n}_{\mathrm{eq}}$), the row vectors of $(\mathbf{G}^T, \mathbf{0})$ must be linearly independent, that is $\operatorname{rank}\mathbf{G}^T = \hat{\mathrm{n}}_{\mathrm{eq}}$. Matrix $\mathbf{G}$ has n_{eq} rows. Thus, a necessary condition for $\operatorname{rank}\mathbf{G} = \hat{\mathrm{n}}_{\mathrm{eq}}$ is that $\hat{\mathrm{n}}_{\mathrm{eq}} \leq \mathrm{n}_{\mathrm{eq}}$. This means that in order for $\mathbf{u}$ and $\mathbf{p}$ to be uniquely determined from (6.21), a necessary, but not sufficient, condition is that

$$\dim \mathcal{Q}^h \leq \dim \boldsymbol{\mathcal{V}}^h.$$

Thus we have found a necessary relation between the dimension of the discrete velocity and pressure spaces.

The sufficient condition linking these spaces is given by the Ladyzhenskaya–Babuška–Brezzi (LBB) compatibility condition, which states that: *The existence of a stable finite element approximate solution $(\mathbf{u}^h, \mathbf{p}^h)$ to the steady Stokes problem* *depends on choosing a pair of spaces* $\mathcal{V}^h$ *and* Q^h, *such that the following* inf-sup condition *holds*:

$$\inf_{q^h \in Q^h} \sup_{\boldsymbol{w}^h \in \mathcal{V}^h} \frac{\left(q^h, \boldsymbol{\nabla} \cdot \boldsymbol{w}^h\right)}{\|q\|_0 \|\, \boldsymbol{w}^h\|_1} \geq \alpha > 0,$$

where α is independent of the mesh size h. If the LBB compatibility condition is satisfied, then there exists a unique $\boldsymbol{u}^h \in \mathcal{V}^h$ and a $p^h \in Q^h$ (determined up to an arbitrary constant in the case of purely Dirichlet boundary conditions).

If the velocity–pressure pair satisfies the LBB condition, the discrete gradient operator $\mathbf{G}$ in the partitioned matrix (6.21) is such that $\ker \mathbf{G} = \{\mathbf{0}\}$. Hence, the pressure matrix $\left(\mathbf{G}^T \mathbf{K}^{-1} \mathbf{G}\right)$ in (6.27) is positive definite and the partitioned matrix (6.21) is non-singular. Existence and uniqueness of the solution are thus guaranteed.

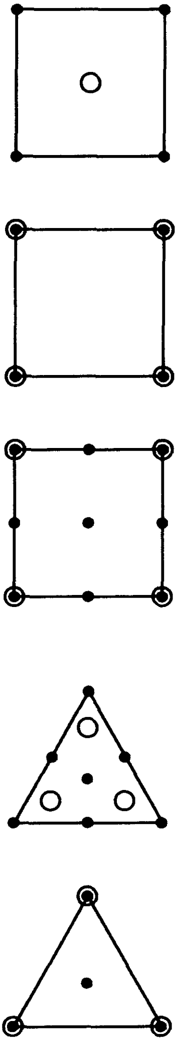
6.5.7 Some popular velocity–pressure couples

In the finite element context, it is by no means easy to prove whether or not a given velocity–pressure pair satisfies the LBB compatibility condition. Although several numerical techniques can help, from the simplest constraint ratio (see for instance Hughes, 2000, Sec. 4.3.7) to the more recent numerical inf-sup testing by Brezzi and Bathe (1990) or Bathe et al. (2000) (see also Brezzi and Fortin, 1991, Sec. II.3.2). Possible combinations of the velocity and pressure approximations leading to stable results are given in Figure 6.1, together with seemingly natural combinations which fail to satisfy the LBB condition.

Among the stable elements listed in Figure 6.1, the triangular and quadrilateral Taylor–Hood elements (Taylor and Hood, 1973) appear to be the most natural ones, in that they only make use of the base element nodes. The velocity and pressure interpolations are both piecewise continuous, the former being quadratic and the latter linear. In addition to being stable, the Taylor–Hood elements exhibit optimal quadratic convergence. The mini element proposed by Arnold, Brezzi and Fortin (1984) and the elements developed by Crouzeix and Raviart (1973) are perhaps less appealing from the viewpoint of computer implementation because they include bubble functions. In the mini element, the piecewise linear velocity field is enriched with a bubble function, which is actually a cubic function in 2D (product of all barycentric coordinates) that vanishes on the boundary. The pressure is continuous and piecewise linear. The mini element exhibits a linear convergence and gives poor pressure approximations in 3D.

The Crouzeix–Raviart triangular element in Figure 6.1 is actually part of a family of stable elements developed by the same authors. It is based on a piecewise linear, but discontinuous, pressure representation. The velocity field is described by a continuous quadratic polynomial enriched with a bubble (as in the mini element product of the barycentric coordinates, i.e. cubic in 2D, quartic in 3D). In spite of its extra cost (certain versions allow more economical implementations), it is considered an accurate and effective element.

A few other simple elements, not listed in Figure 6.1, do satisfy the LBB condition. These include the quadrilateral element with continuous piecewise biquadratic velocity interpolation and discontinuous linear pressure (as in the Crouzeix–Raviart



Q1P0 element:
Continuous bilinear velocity,
Discontinuous constant pressure,
Does not satisfy LBB condition.
(same for linear/constant triangle)

Q1Q1 element:
Continuous bilinear velocity,
Continuous bilinear pressure,
Does not satisfy LBB condition.
(same for linear/linear triangle)

Q2Q1 element:
(Taylor–Hood element)
Continuous biquadratic velocity,
Continuous bilinear pressure,
Satisfies LBB condition,
Quadratic convergence.
(same for quadratic/linear triangle)

Crouzeix–Raviart element:
Velocity: continuous quadratic
+ cubic bubble function,
Pressure: discontinuous linear,
Satisfies LBB condition,
Quadratic convergence.

Mini element:
Velocity: continuous linear
+ cubic bubble function,
Pressure: continuous linear,
Satisfies LBB condition,
Linear convergence.

Nodes: ● Velocity

○ Pressure

Fig. 6.1 Examples of 2D stable and unstable velocity and pressure interpolations.

triangular element). This element enjoys quadratic convergence properties. Also stable is the triangular element with continuous piecewise quadratic velocity and elementwise constant pressure. This element has a linear convergence rate.

6.5.8 Stabilization of the Stokes problem

In recent years, the efforts of researchers in the area of mixed methods have been directed towards circumventing the LBB condition, thus opening the way to the use of velocity–pressure pairs which are not stable in the standard Galerkin formulation.

The basic idea behind stabilization procedures is to enforce the positive definiteness of matrix $\left(\begin{smallmatrix}\mathbf{K} & \mathbf{G}\\ \mathbf{G}^T & \mathbf{0}\end{smallmatrix}\right)$, see equation (6.21). This can, for instance, be accomplished through a modification of the weak form of the incompressibility condition in order to render non-zero the diagonal term resulting from the incompressibility condition.

It is now standard to stabilize incompressible flow problems using techniques inspired from stabilized formulations, such as SUPG and Galerkin/Least-squares (GLS), introduced in Chapter 2.

Let us illustrate the GLS stabilization proposed for the Stokes problem by Hughes and Franca (1987). The method consists of modifying the variational form of the Stokes problem (6.16), that is equations (6.13), by the addition of the terms emanating from the minimization of the least-squares form

$$L_{\mathrm{S}}(\boldsymbol{v},p) := \big(-\nu\boldsymbol{\nabla}^2\boldsymbol{v} + \boldsymbol{\nabla}p - \boldsymbol{b}, -\nu\boldsymbol{\nabla}^2\boldsymbol{v} + \boldsymbol{\nabla}p - \boldsymbol{b}\big),$$

that is, the square of the residual of the momentum equation, see (6.16a). The stationarity condition of $L_{\mathrm{S}}(\boldsymbol{v},p)$ implies that

$$\frac{dL_{\mathrm{S}}(\boldsymbol{v}+\epsilon\boldsymbol{w},p+\epsilon q)}{d\epsilon}\bigg|_{\epsilon=0} = 0$$

for all values of $\boldsymbol{w}$ and q, variations of the velocity and of the pressure, respectively. The derivation foreseen in the previous equation implies that

$$\big(-\nu\boldsymbol{\nabla}^2\boldsymbol{w} + \boldsymbol{\nabla}q, -\nu\boldsymbol{\nabla}^2\boldsymbol{v} + \boldsymbol{\nabla}p - \boldsymbol{b}\big) = 0 \qquad \forall(\boldsymbol{w},q)\in\boldsymbol{\mathcal{V}}\times\mathcal{Q},$$

or equivalently,

$$\begin{cases} \big(-\nu\boldsymbol{\nabla}^2\boldsymbol{w}, -\nu\boldsymbol{\nabla}^2\boldsymbol{v} + \boldsymbol{\nabla}p - \boldsymbol{b}\big) = 0 & \forall\boldsymbol{w}\in\boldsymbol{\mathcal{V}}\\ \big(\boldsymbol{\nabla}q, -\nu\boldsymbol{\nabla}^2\boldsymbol{v} + \boldsymbol{\nabla}p - \boldsymbol{b}\big) = 0 & \forall q\in\mathcal{Q}. \end{cases}$$

The stabilization of the Stokes problem is obtained by adding to the Galerkin weak form (6.13) the previous equations emanating from the least-squares form. This entails a modification of both the momentum and the continuity equations. To avoid additional continuity requirements due to the presence of second spatial derivatives, the terms added to the Galerkin weak form act on the element interiors only. These terms depend on the residual of the momentum equation and therefore ensure the consistency of the stabilized formulation. The stabilized discrete problem is then formulated as: find $\boldsymbol{v}^h \in \boldsymbol{\mathcal{S}}^h$ and $p^h \in \mathcal{Q}^h$, for all $(\boldsymbol{w}^h, q^h) \in \boldsymbol{\mathcal{V}}^h \times \mathcal{Q}^h$, such that

$$\begin{cases} a(\boldsymbol{w}^h, \boldsymbol{v}^h) + b(\boldsymbol{w}^h, p^h) + \displaystyle\sum_{e=1}^{n_{\mathrm{el}}} \tau_e\big(-\nu\boldsymbol{\nabla}^2\boldsymbol{w}^h, -\nu\boldsymbol{\nabla}^2\boldsymbol{v}^h + \boldsymbol{\nabla}p^h - \boldsymbol{b}^h\big)_{\Omega^e} \\ \qquad\qquad\qquad\qquad = \big(\boldsymbol{w}^h, \boldsymbol{b}^h\big) + \big(\boldsymbol{w}^h, \boldsymbol{t}^h\big)_{\Gamma_N}, \\ b(\boldsymbol{v}^h, q^h) - \displaystyle\sum_{e=1}^{n_{\mathrm{el}}} \tau_e\big(\boldsymbol{\nabla}q^h, -\nu\boldsymbol{\nabla}^2\boldsymbol{v}^h + \boldsymbol{\nabla}p^h - \boldsymbol{b}^h\big)_{\Omega^e} = 0, \end{cases}$$

where τ_e is the stabilization parameter. Note that the presence in the second equation of the term $\big(\boldsymbol{\nabla}q^h, \boldsymbol{\nabla}p^h\big)$ introduces a non-zero diagonal term in the partitioned matrix resulting from the spatial discretization of the above GLS weak form. This produces the desired stabilization of the pressure field.

For linear elements the GLS stabilization does not affect the weak form of the momentum equation because the terms involving the second derivatives of the weighting function $\boldsymbol{w}$ vanish. The GLS weak formulation then reduces to the following variational problem: find $\boldsymbol{v}^h \in \boldsymbol{\mathcal{S}}^h$ and $p^h \in \mathcal{Q}^h$, such that, for all $(\boldsymbol{w}^h, q^h) \in \boldsymbol{\mathcal{V}}^h \times \mathcal{Q}^h$,

$$\begin{cases} a(\boldsymbol{w}^h, \boldsymbol{v}^h) + b(\boldsymbol{w}^h, p^h) = \big(\boldsymbol{w}^h, \boldsymbol{b}^h\big) + \big(\boldsymbol{w}^h, \boldsymbol{t}^h\big)_{\Gamma_N}, \\ b(\boldsymbol{v}^h, q^h) - \displaystyle\sum_{e=1}^{n_{\mathrm{el}}} \tau_e\big(\boldsymbol{\nabla}q^h, \boldsymbol{\nabla}p^h\big)_{\Omega^e} = -\displaystyle\sum_{e=1}^{n_{\mathrm{el}}} \tau_e\big(\boldsymbol{\nabla}q^h, \boldsymbol{b}^h\big)_{\Omega^e}. \end{cases} \tag{6.28}$$

Note that the second term in the second equation indicates that a Poisson equation has been generated for the pressure field. An interesting consequence of the GLS stabilization of the Stokes problem is that elements with equal order interpolations, which are unstable in the Galerkin formulation, now become stable. This is, in particular, the case for the quadrilateral element in Figure 6.1 with continuous piecewise bilinear interpolations, as well as for the linear/linear three-node triangle. The stabilization parameter is chosen as

$$\tau_e = \alpha_0 \frac{h_e^2}{4\nu},$$

where h_e is a measure of the element size. Since the Stokes equations do not contain convective terms, the only component of τ_e is the viscous contribution. The choice $\alpha_0 = 1/3$ appears to be optimal for linear elements.

6.5.9 Penalty method

The penalty method may be interpreted as enabling a relaxation of the incompressibility constraint in the sense that the incompressible problem is approximated by means of a slightly compressible formulation.

In the mixed finite element approach pressure is an unknown. However, the penalty method allows its elimination and thus reduces the size of the matrix problem. There is a vast literature on penalty methods for incompressible flow problems. In the finite element framework, initial studies of the method were performed, among others, by Temam (2001), Girault and Raviart (1986), Malkus and Hughes (1978), Zienkiewicz and Godbole (1975), Bercovier (1978) and Bercovier and Engelman (1979). Articles by Hughes, Liu and Brooks (1979) and Heinrich, Marshall and Zienkiewicz (1978) also provide a detailed presentation of the penalty formulation for the Stokes problem, including computer implementation details.

The starting point for the penalty method consists in replacing the incompressibility constraint $\nabla \cdot v = 0$ by

$$\nabla \cdot v^{(\lambda)} = -p^{(\lambda)}/\lambda, \tag{6.29}$$

where λ is a mesh-size- and problem-independent parameter taken of the order of $10^7 \sim 10^8$ in double-precision calculations. The fluid constitutive equation (6.2) is then replaced by the following relation:

$$\sigma_{ij}^{(\lambda)} = -p^{(\lambda)}\delta_{ij} + 2\nu\, v_{(i,j)}^{(\lambda)},$$

where pressure is now defined by (6.29), namely $p^{(\lambda)} = -\lambda \nabla \cdot v^{(\lambda)}$.

The strict incompressibility constraint (6.11b) is thus abandoned. More precisely, it is introduced in the momentum equation as a penalized term emanating from relation (6.29). Consequently, the boundary value problem for Stokes flow is formulated as:

$$\begin{cases} -\nabla \cdot \sigma^{(\lambda)} = b & \text{in } \Omega, \\ v^{(\lambda)} = v_D & \text{on } \Gamma_D, \\ n \cdot \sigma^{(\lambda)} = t & \text{on } \Gamma_N. \end{cases}$$

The weak form for the penalty method is easily obtained in the primitive variable, v,

$$a(w, v) + \lambda \left(\nabla \cdot w, \nabla \cdot v\right) = (w, b) + (w, t)_{\Gamma_N} \qquad \forall w \in \mathcal{V}.$$

Note the presence of a new viscous-type term associated with the penalty parameter. The incompressibility equation is not needed because the pressure in the momentum equation is replaced using expression (6.29). As in Section 6.5.3, we use the representation $v^h = u^h + v_D^h$ for the finite dimensional approximation of the velocity. The discrete formulation is then given as: find $u^h \in \mathcal{V}^h$, for all $w^h \in \mathcal{V}^h$, such that

$$\begin{aligned} a(w^h, u^h) + \lambda \left(\nabla \cdot w^h, \nabla \cdot u^h\right) = \left(w^h, b^h\right) + \left(w^h, t^h\right)_{\Gamma_N} \\ - a(w^h, v_D^h) - \lambda \left(\nabla \cdot w^h, \nabla \cdot v_D^h\right). \end{aligned}$$

After spatial discretization, this equation produces the following set of nodal equations for the components of the auxiliary velocity: for each $A \in \eta \setminus \eta_{Di}$ and $1 \le i \le \mathrm{n_{sd}}$

$$\begin{aligned} \sum_{j=1}^{\mathrm{n_{sd}}} \Big\{ \sum_{B \in \eta \setminus \eta_{Dj}} \left[a(N_A\, e_i, N_B\, e_j) + \lambda\big(\nabla \cdot (N_A e_i), \nabla \cdot (N_B e_j)\big)\right] \mathrm{u}_{jB} \Big\} \\ = \left(N_A\, e_i, b^h\right) + \left(N_A\, e_i, t^h\right)_{\Gamma_N} \\ - \sum_{j=1}^{\mathrm{n_{sd}}} \Big\{ \sum_{B \in \eta_{Dj}} \left[a(N_A\, e_i, N_B\, e_j) + \lambda \left(\nabla \cdot (N_A e_i), \nabla \cdot (N_B e_j)\right)\right] v_{Dj} \Big\}. \end{aligned}$$

From this set of algebraic equations, the following matrix system is obtained:

$$\left(\mathbf{K} + \mathbf{K}^{\lambda}\right) \mathbf{u}^{(\lambda)} = \mathbf{f}, \tag{6.30}$$

which governs the penalty formulation of the Stokes problem. $\mathbf{K}$ is the viscosity matrix arising from the term $a\left(w^h, u^h\right)$. It is identical to the viscosity matrix obtained in Section 6.5.4 for the mixed method. $\mathbf{K}^{\lambda}$ is the so-called *penalty matrix*, which has the same structure as $\mathbf{K}$ and is defined as

$$\mathbf{K}^{\lambda} = \mathbf{A}^{e}[\mathbf{K}^{\lambda}]^{e}, \qquad [K^{\lambda}]^{e}_{rs} = \lambda\left(\nabla \cdot (N_a e_i), \nabla \cdot (N_b e_j)\right)_{\Omega^e},$$

recall the definition of the indices given by (6.22). Thus, it is identical to expression (6.24) but with λ replacing ν. Finally, the nodal vector $\mathbf{f}$ is defined in (6.25) (where K^{e}_{rs} now accounts for the sum of the viscosity and penalty matrices).

Observe that the global matrices $\mathbf{K}$ and $\mathbf{K}^{\lambda}$ are proportional to ν and λ, respectively. In order to impose incompressibility, parameter λ must be selected very large. However, when λ is very large, the penalty matrix plays a dominant role in system (6.30). Then, if matrix $\mathbf{K}^{\lambda}$ is regular, only the trivial solution $\mathbf{u}^{(\lambda)} = \mathbf{0}$ of the homogeneous form of (6.30) is possible. It follows that matrix $\mathbf{K}^{\lambda}$ must be singular to obtain a meaningful solution, that is

$$\mathbf{K}^{\lambda}\,\mathbf{u}^{(\lambda)} = \mathbf{0} \qquad \text{and} \qquad \mathbf{u}^{(\lambda)} \neq \mathbf{0}.$$

Standard numerical integration rules used to evaluate finite element matrices are such that, with conforming elements, they practically always result in a regular $\mathbf{K}^{\lambda}$ matrix. To render the penalty matrix singular, one needs to lower the order of the quadrature used to evaluate the element integrals involving the λ term. Note that the rank of system (6.30) is conserved using the required (full) quadrature for the viscous diffusion term $\mathbf{K}^{\nu}$. Figure 6.2, reproduced from the book by Hughes (2000), gives the Gaussian integration rules to be used for the viscous and penalty terms with some popular elements in 2D and 3D.

The convergence of the solution of the Stokes problem obtained by the penalty method has been proved by Temam (2001). The main advantage of the penalty formulation is the uncoupling of the velocity and pressure solutions and the elimination of the incompressibility condition from the variational formulation. The reduction in the number of variables renders the penalty method quite attractive. However, an improper choice of the value of parameter λ might cause numerical problems. If λ is too small compressibility and pressure errors will occur. An excessively large value may result in numerical ill conditioning. For Stokes flow, Hughes et al. (1979) suggest selecting λ according to the relation $\lambda = c\mu$, where μ is the fluid dynamic viscosity and c is a constant of the order of 10^7 for double-precision calculations.

Remark 6.13 (The need for under-integration of the penalty term). As an illustration, consider the case of a local approximation of the velocity u^h based upon a nine-node biquadratic interpolation in the plane $(x_1 . x_2)$. The components (u_1^h, u_2^h) of the velocity are polynomials of the form

$$\begin{aligned} u_j^h = {} & a_{1,j} + a_{2,j}\, x_1 + a_{3,j}\, x_2 + a_{4,j}\, x_1 x_2 \\ & + a_{5,j}\, x_1^2 + a_{6,j}\, x_2^2 + a_{7,j}\, x_1^2 x_2 + a_{8,j}\, x_1 x_2^2 + a_{9,j}\, x_1^2 x_2^2, \end{aligned}$$

where $a_{i,j}$, $i = 1, \dots, 9$, are the nine coefficients of the biquadratic polynomial for each component ($j = 1, 2$) of the local velocity within the element. When λ is very large the flow is incompressible and the divergence-free condition $\partial u_1^h / \partial x_1 + \partial u_2^h / \partial x_2 = 0$ becomes

$$\begin{aligned} & a_{2,1} + a_{4,1}\, x_2 + 2a_{5,1}\, x_1 + 2a_{7,1}\, x_1 x_2 + a_{8,1}\, x_2^2 + 2a_{9,1}\, x_1 x_2^2 \\ & \qquad + a_{3,2} + a_{4,2}\, x_1 + 2a_{6,2}\, x_2 + a_{7,2}\, x_1^2 + 2a_{8,2}\, x_1 x_2 + 2a_{9,2}\, x_1^2 x_2 = 0. \end{aligned}$$

If the penalty term is integrated exactly the above relationship must be satisfied at every point (x_1, x_2) of the element and a total of *eight constraints* are thus imposed for each element:

$$\begin{array}{llll} a_{2,1} + a_{3,2} = 0, & a_{4,1} + 2a_{6,2} = 0, & a_{8,1} = 0, & a_{7,2} = 0, \\ 2a_{5,1} + a_{4,2} = 0, & 2a_{7,1} + 2a_{8,2} = 0, & a_{9,1} = 0, & a_{9,2} = 0. \end{array}$$

This coincides with the number of velocity degrees of freedom per element in an infinite plane mesh,

$$\lim_{n \to \infty} \frac{2\,(2n+1)^2}{n^2} = 8.$$

Thus, in the case of full integration of the penalty term, the constraints are seen to consume a great number of degrees of freedom and too few are left to properly simulate the flow problem at hand. To cure this situation a 2 × 2 Gaussian integration rule must be used to evaluate the penalty term, while the normal 3 × 3 rule for the biquadratic element should be used for the viscous diffusion term. In this way, only four constraints are imposed per element. Incompressibility is therefore satisfied in the mean, but now four degrees of freedom per element are available to satisfy the equations of motion.

Success of the penalty method crucially depends on an appropriate reduction of the number of constraints introduced by the penalty term. The use of a reduced integration rule for the penalty term leads to a proper balance between the number of degrees available to satisfy the momentum equations and those consumed to approximate the incompressibility of the flow.

Remark 6.14 (Equivalence of penalty and mixed methods). Malkus and Hughes (1978) (see also Hughes, 2000) have established this important result. They demonstrate the *equivalence*, for incompressible problems, between the reduced integration of the penalty term and a mixed finite element method if the pressure nodes coincide with the integration points of the reduced rule. The equivalence between mixed elements and penalty-type elements with selective integration is illustrated in Figure 6.2.

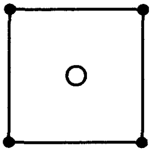
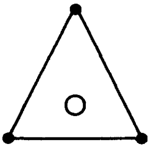
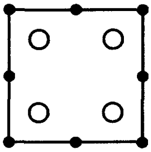
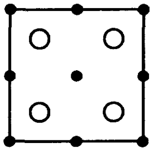
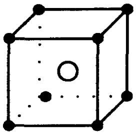
	Mixed elements		Equivalent elements with selective integration	
	Velocity interpolation	Pressure interpolation	Standard integration (Term **K**)	Reduced integration (Term $\mathbf{K}^{\lambda}$)
	Bilinear	Constant	2×2	1 point
	Linear	Constant	1 point	1 point
	Serendipity	Bilinear	3×3	2×2
	Biquadratic	Bilinear	3×3	2×2
	Trilinear	Constant	2×2×2	1 point

Fig. 6.2 Equivalence of mixed and reduced/selective integration elements.

6.6 STEADY NAVIER–STOKES PROBLEM

We shall now consider the Navier–Stokes equations for steady flow. From the definitions of Section 6.2.3 and, in particular, equation (6.5) the strong form of the boundary value problem is stated as follows: find the velocity field $\boldsymbol{v}$ and the pressure field p, such that

$$
\begin{aligned}
-\nu\boldsymbol{\nabla}^2\boldsymbol{v} + (\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v} + \boldsymbol{\nabla}p &= \boldsymbol{b} && \text{in } \Omega, && (6.31a)\\
\boldsymbol{\nabla}\cdot\boldsymbol{v} &= 0 && \text{in } \Omega, && (6.31b)\\
\boldsymbol{v} &= \boldsymbol{v}_D && \text{on } \Gamma_D, && (6.31c)\\
-p\boldsymbol{n} + \nu(\boldsymbol{n}\cdot\boldsymbol{\nabla})\boldsymbol{v} &= \boldsymbol{t} && \text{on } \Gamma_N. && (6.31d)
\end{aligned}
$$

Note that the Cauchy stress is again assumed to be given by Stokes' law.

6.6.1 Weak form and Galerkin formulation

The Navier–Stokes problem differs from the Stokes problem because of the presence of the nonlinear convective term $(\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v}$. Considering the same spaces $\mathcal{S}$, $\mathcal{V}$ and $\mathcal{Q}$ defined by (6.9), the weak formulation of problem (6.31) becomes: find $\boldsymbol{v} \in \mathcal{S}$ and $p \in \mathcal{Q}$, such that

$$
\begin{cases}
a(\boldsymbol{w},\boldsymbol{v}) + c(\boldsymbol{v};\boldsymbol{w},\boldsymbol{v}) + b(\boldsymbol{w},p) = (\boldsymbol{w},\boldsymbol{b}) + (\boldsymbol{w},\boldsymbol{t})_{\Gamma_N} & \forall \boldsymbol{w} \in \mathcal{V},\\
b(\boldsymbol{v},q) = 0 & \forall q \in \mathcal{Q},
\end{cases}
$$

where, apart from the bilinear form used in the weak form (6.16) of the Stokes problem, we have introduced the trilinear form, see Section 1.5.3,

$$
c(\boldsymbol{a};\boldsymbol{w},\boldsymbol{v}) = \big(\boldsymbol{w},(\boldsymbol{a}\cdot\boldsymbol{\nabla})\boldsymbol{v}\big) = \int_\Omega \boldsymbol{w}\cdot(\boldsymbol{a}\cdot\boldsymbol{\nabla})\boldsymbol{v}\, d\Omega,
$$

which, in this case, is associated with the nonlinear convective term in the momentum equation. Introducing the finite dimensional subspaces $\mathcal{S}^h$, $\mathcal{V}^h$ and $\mathcal{Q}^h$, and proceeding as for Stokes flow, we obtain the Galerkin counterpart of the previous weak formulation. Define $\boldsymbol{v}_D^h \in \mathcal{S}^h$ such that $\boldsymbol{v}^h = \boldsymbol{u}^h + \boldsymbol{v}_D^h$, then find the auxiliary velocity field $\boldsymbol{u}^h \in \mathcal{V}^h$ and the pressure $p^h \in \mathcal{Q}^h$, such that, for all $(\boldsymbol{w}^h, q^h) \in \mathcal{V}^h \times \mathcal{Q}^h$,

$$
\begin{cases}
a(\boldsymbol{w}^h,\boldsymbol{u}^h) + c(\boldsymbol{v}^h;\boldsymbol{w}^h,\boldsymbol{u}^h) + b(\boldsymbol{w}^h,p^h) = (\boldsymbol{w}^h,\boldsymbol{b}^h) + (\boldsymbol{w}^h,\boldsymbol{t}^h)_{\Gamma_N},\\
\qquad\qquad\qquad\qquad\qquad - a(\boldsymbol{w}^h,\boldsymbol{v}_D^h) - c(\boldsymbol{v}^h;\boldsymbol{w}^h,\boldsymbol{v}_D^h)\\
b(\boldsymbol{u}^h,q^h) = -b(\boldsymbol{v}_D^h,q^h).
\end{cases}
$$

Note that now, the r.h.s. term depends on the unknown because $\boldsymbol{v}^h = \boldsymbol{u}^h + \boldsymbol{v}_D^h$.

6.6.2 Matrix problem

We then express the finite dimensional approximations in terms of the velocity and pressure shape functions defined in (6.19). Proceeding as in Section 6.5.4, the matrix system governing the discrete Navier–Stokes problem is obtained in the following partitioned form:

$$\begin{pmatrix} \mathbf{K}+\mathbf{C}(\boldsymbol{v}) & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f}(\boldsymbol{v}) \\ \mathbf{h} \end{pmatrix}, \tag{6.32}$$

where, as in the Stokes problem, matrices $\mathbf{K}$ and $\mathbf{G}$ are, respectively, the viscosity matrix and the discrete gradient operator. $\mathbf{C} = \mathbf{C}(\boldsymbol{v})$ is the convection matrix defined by

$$\mathbf{C} = \mathop{\mathbf{A}}^{e} \mathbf{C}^e, \qquad \text{with} \quad C^e_{rs} = \boldsymbol{e}_i^T \int_{\Omega^e} N_a \, \boldsymbol{v}^h \cdot \boldsymbol{\nabla} N_b \, d\Omega \; \boldsymbol{e}_j. \tag{6.33}$$

Vector $\mathbf{f}$ is the same as for the Stokes problem, see (6.25), plus an added term: the product of the previous matrix by the Dirichlet velocities. Recall the definition of the indices given by (6.22).

Here again, the saddle-point nature of the problem, with pressure acting as a Lagrangian multiplier of the incompressibility constraint, yields a coupling between velocity and pressure. System (6.32) is nonlinear and non-symmetric (due to the convective terms), an appropriate iterative solution technique must be used. An in-depth discussion of solution algorithms for the nonlinear Navier–Stokes equations may be found in the textbooks mentioned in the introduction to this chapter.

There are two potential sources of numerical instability in the Galerkin finite element solution of steady Navier–Stokes problems. The first is due to the treatment of the convective term and manifests itself in high Reynolds number flows when unresolved internal or boundary layers are present in the solution. The second source of potential instability, already encountered in Stokes flow, is an inappropriate combination of interpolation functions for velocity and pressure. Fortunately, stabilization procedures for the Navier–Stokes equations capable of curing both types of numerical instability are available. They are presented in Section 6.7.2.

6.7 UNSTEADY NAVIER–STOKES EQUATIONS

Let us now consider the case of unsteady viscous incompressible flows. Here, emphasis will be placed on stabilized finite element formulations and on the advantages of a fractional-step projection method for time integration of the Navier–Stokes equations. Again, using the definitions of Section 6.2.3, the governing equations and associated initial/boundary conditions are:

$$\begin{aligned} \boldsymbol{v}_t - \nu\boldsymbol{\nabla}^2\boldsymbol{v} + (\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v} + \boldsymbol{\nabla}p &= \boldsymbol{b} && \text{in } \Omega\times]0,T[, && (6.34a)\\ \boldsymbol{\nabla}\cdot\boldsymbol{v} &= 0 && \text{in } \Omega\times]0,T[, && (6.34b)\\ \boldsymbol{v} &= \boldsymbol{v}_D && \text{on } \Gamma_D\times]0,T[, && (6.34c)\\ -p\boldsymbol{n} + \nu(\boldsymbol{n}\cdot\boldsymbol{\nabla})\boldsymbol{v} &= \boldsymbol{t} && \text{on } \Gamma_N\times]0,T[, && (6.34d)\\ \boldsymbol{v}(\boldsymbol{x},0) &= \boldsymbol{v}_0(\boldsymbol{x}) && \text{in } \Omega. && (6.34e) \end{aligned}$$

Note that the initial velocity field is assumed solenoidal: $\boldsymbol{\nabla}\cdot\boldsymbol{v}_0 = 0$.

6.7.1 Weak formulation and spatial discretization

The weak formulation is obtained, as usual, by projection of the equations (6.34) onto a space of weighting functions $\boldsymbol{w} \in \boldsymbol{\mathcal{V}}$ for the momentum equation and $q \in \mathcal{Q}$ for the incompressibility condition. The end result is the following variational problem: given $\boldsymbol{b}, \boldsymbol{v}_D, \boldsymbol{t}$ and $\boldsymbol{v}_0$, find $\boldsymbol{v}(\boldsymbol{x},t) \in \boldsymbol{\mathcal{S}}\times]0,T[$ and $p(\boldsymbol{x},t) \in \mathcal{Q}\times]0,T[$, such that, for all $(\boldsymbol{w},q) \in \boldsymbol{\mathcal{V}} \times \mathcal{Q}$,

$$\begin{cases} (\boldsymbol{w},\boldsymbol{v}_t) + a(\boldsymbol{w},\boldsymbol{v}) + c(\boldsymbol{v};\boldsymbol{w},\boldsymbol{v}) + b(\boldsymbol{w},p) = (\boldsymbol{w},\boldsymbol{b}) + (\boldsymbol{w},\boldsymbol{t})_{\Gamma_N}, \\ b(\boldsymbol{v},q) = 0 \end{cases}$$

with $\boldsymbol{v}(0) = \boldsymbol{v}_0$.

The Galerkin spatial discretization of this time-dependent problem proceeds as previously. For each $t \in]0,T[$, we define $\boldsymbol{v}_D^h(t) \in \boldsymbol{\mathcal{S}}^h$ such that $\boldsymbol{v}^h(t) = \boldsymbol{u}^h(t) + \boldsymbol{v}_D^h(t)$. Then, we seek the auxiliary velocity field $\boldsymbol{u}^h(\cdot,t) \in \boldsymbol{\mathcal{V}}^h$ and pressure $p^h(\cdot,t) \in \mathcal{Q}^h$, such that, for all $(\boldsymbol{w}^h,q^h) \in \boldsymbol{\mathcal{V}}^h \times \mathcal{Q}^h$,

$$\begin{cases} (\boldsymbol{w}^h,\boldsymbol{u}_t^h) + a(\boldsymbol{w}^h,\boldsymbol{u}^h) + c(\boldsymbol{v}^h;\boldsymbol{w}^h,\boldsymbol{u}^h) + b(\boldsymbol{w}^h,p^h) \\ \qquad\qquad = (\boldsymbol{w}^h,\boldsymbol{b}^h) + (\boldsymbol{w}^h,\boldsymbol{t}^h)_{\Gamma_N} - a(\boldsymbol{w}^h,\boldsymbol{v}_D^h) - c(\boldsymbol{v}^h;\boldsymbol{w}^h,\boldsymbol{v}_D^h) \\ b(\boldsymbol{u}^h,q^h) = -b(\boldsymbol{v}_D^h,q^h) \end{cases}$$

with $\boldsymbol{v}^h(0) = \boldsymbol{v}_0^h$.

Finally, the finite element discretization of this weak form yields the system of semi-discrete equations for $t \in]0,T[$

$$\begin{cases} \mathbf{M}\,\dot{\mathbf{u}}(t) + \left[\mathbf{K} + \mathbf{C}(\boldsymbol{v}(t))\right]\mathbf{u}(t) + \mathbf{G}\,\mathbf{p}(t) = \mathbf{f}\left(t,\boldsymbol{v}(t)\right) \\ \mathbf{G}^T\mathbf{u}(t) = \mathbf{h}(t) \\ \mathbf{u}(0) = \mathbf{v}_0 - \mathbf{v}_D(0) \end{cases} \tag{6.35}$$

where $\mathbf{M}$ is the standard finite element mass matrix.

To trace the transient response, this system of semi-discrete equations can be advanced in time by suitable finite difference schemes such as the θ family methods introduced in Section 3.4.2.3. Note that a fully implicit method requires the solution of a nonlinear algebraic system at each time step. Semi-implicit methods in which the convection matrix $\mathbf{C}(\boldsymbol{v}(t))$ and $\mathbf{f}\left(t,\boldsymbol{v}(t)\right)$ are treated explicitly are thus generally preferred. In Section 6.7.3 we shall consider the time discretization of the unsteady Navier–Stokes equations by means of a fractional-step procedure.

Remark 6.15. An in-depth analysis of the semi-discrete system induced by the unsteady Navier–Stokes equations can be found in the series of papers by Heywood and Rannacher (1982; 1986; 1988; 1990). In order to analyze stability and obtain an energy estimate, these authors use the nowadays standard skew-symmetric form of the convection term, see for instance Temam (2001). That is, the trilinear form $c(\boldsymbol{v};\boldsymbol{w},\boldsymbol{u})$ is replaced by

$$\widehat{c}(\boldsymbol{v};\boldsymbol{w},\boldsymbol{u}) = \frac{1}{2}\Big(c(\boldsymbol{v};\boldsymbol{w},\boldsymbol{u}) - c(\boldsymbol{v};\boldsymbol{u},\boldsymbol{w})\Big),$$

which is skew-symmetric, i.e. $\widehat{c}(\boldsymbol{v};\boldsymbol{w},\boldsymbol{w}) = 0$. This modified trilinear term is consistent with the original unsteady incompressible Navier–Stokes problem. It is easy to verify that the replacement of the original trilinear term, $c(\cdot;\cdot,\cdot)$, by the new skew-symmetric form is equivalent to the replacement of the original convective term, $(\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v}$, by $(\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v}+\frac{1}{2}(\boldsymbol{\nabla}\cdot\boldsymbol{v})\boldsymbol{v}$. Note that for a divergence-free velocity field, such as the incompressible solution of (6.34), this modification is legitimate.

Finally, it is important to remark that the skew-symmetric form of the convection term is also exploited in the computational schemes in order to ensure unconditional time stability, see Remark 6.18.

6.7.2 Stabilized finite element formulation

Recall that the Galerkin finite element method leads to central approximations of the convective terms and is thus not optimal when convection dominate diffusion (the viscosity effects), that is for high Reynolds number flows. In such cases, use should be made of a stabilized finite element formulation to obtain reliable numerical solutions. In addition, as was the case for Stokes flow, the stability of the Galerkin method applied to the incompressible Navier–Stokes equations depends on satisfying of the LBB condition. It is nevertheless possible to circumvent this condition, as already seen for the Stokes problem, by making use of a stabilization technique. There are therefore two major reasons for stabilizing the incompressible Navier–Stokes equations.

The extension of the stabilized formulation of the Stokes problem discussed in Section 6.5.8 to the incompressible Navier–Stokes equations (6.34) has been studied in a series of papers by Johnson and Saranen (1986), Hughes, Franca and Hulbert (1989), Hansbo and Szepessy (1990), Franca et al. (1992), Franca and Frey (1992), Franca and Hughes (1993), Tezduyar, Mittal, Ray and Shih (1992) and Tezduyar and Osawa (2000), among others. See also the review paper by Tezduyar (1992) on finite element stabilization methods for incompressible flow computations.

The stabilized finite element formulation of the Navier–Stokes problem (6.34) proposed by Tezduyar and Osawa (2000) is given as follows: find $\boldsymbol{v}^h \in \boldsymbol{\mathcal{S}}^h\times]0,T[$ and $p^h \in \mathcal{Q}^h\times]0,T[$, for all $(\boldsymbol{w}^h, q^h) \in \boldsymbol{\mathcal{V}}^h \times \mathcal{Q}^h$, such that

$$\left\{\begin{aligned} &(\boldsymbol{w}^h,\boldsymbol{v}_t^h) + a(\boldsymbol{w}^h,\boldsymbol{v}^h) + c(\boldsymbol{v}^h;\boldsymbol{w}^h,\boldsymbol{v}^h) + b(\boldsymbol{w}^h,p^h) - (\boldsymbol{w}^h,\boldsymbol{b}^h) - (\boldsymbol{w}^h,\boldsymbol{t}^h)_{\Gamma_N}\\ &\qquad + \sum_{e=1}^{n_{\mathrm{el}}} \tau_{\mathrm{SUPG}}\big((\boldsymbol{v}^h\cdot\boldsymbol{\nabla})\boldsymbol{w}^h, \mathcal{R}(\boldsymbol{v}^h)\big)_{\Omega^e} + \tau_{\mathrm{LSIC}}\big(\boldsymbol{\nabla}\cdot\boldsymbol{w}^h, \boldsymbol{\nabla}\cdot\boldsymbol{v}^h\big)_{\Omega^e} = 0.\\ &b(\boldsymbol{v}^h,q^h) + \sum_{e=1}^{n_{\mathrm{el}}} \tau_{\mathrm{PSPG}}\big(\boldsymbol{\nabla}q^h, \mathcal{R}(\boldsymbol{v}^h)\big)_{\Omega^e} = 0, \end{aligned}\right.$$

where

$$\mathcal{R}(\boldsymbol{v}^h) = \boldsymbol{v}_t^h + (\boldsymbol{v}^h\cdot\boldsymbol{\nabla})\boldsymbol{v}^h - \nu\boldsymbol{\nabla}^2\boldsymbol{v}^h + \boldsymbol{\nabla}p^h - \boldsymbol{b}^h$$

is the residual of the momentum equation.

In the variational form of the momentum equation, the first line represents the standard Galerkin formulation. Similarly, the first term in the weak form of the incompressibility constraint is the usual Galerkin term.

The terms involving the element-level integrals are the added stabilization terms. These terms depend on three stabilization parameters. The terms involving parameter τ_{SUPG} stabilize the Galerkin formulation in the presence of a dominating convective term in the momentum equation, see Sections 2.4, 5.4.5 or 5.4.6. Note that other stabilization techniques such as GLS, SGS or LS can also be employed.

Parameter τ_{LSIC} is in fact an artificial diffusion. Its dimension is length square over time (same as the kinematic viscosity ν), and a typical choice is $\|\boldsymbol{v}^h\| h_e/2$. Tezduyar and Osawa (2000) propose to use this term as a least-squares stabilization on the incompressibility constraint. It provides additional stability for flows at large Reynolds numbers.

The terms associated with parameter τ_{PSPG} (pressure-stabilizing/Petrov–Galerkin) allow the use of mixed elements with equal-order interpolations for the velocity and pressure, see Section 6.5.8, in particular equations (6.28). Note that all stabilization terms are weighted residuals, therefore ensuring the consistency of the formulation.

Tezduyar and Osawa (2000) suggest evaluating the stabilization parameters using element-level matrices and vectors which automatically account for the local length scales, advection field and flow Reynolds number Re. The reader interested in the actual construction of element-based stability parameters should consult the above-mentioned article. Note that alternative definitions of τ_{SUPG} have been proposed by Codina (2000) and Shakib et al. (1991), see Sections 2.4.3, 5.4.5 and 5.4.6.

6.7.3 Time discretization by fractional-step methods

A popular method for the time discretization of the unsteady Navier–Stokes equations is the fractional-step method in which, as already mentioned in Section 5.3.2, the time advancement is decomposed into a sequence of two or more steps. Fractional-step methods for the incompressible Navier–Stokes equations were originated independently by Chorin (1968; 1969) and Temam (1969; 2001).

As we shall see, the fractional-step approach to time integration allows us to alleviate the numerical difficulties related to the saddle-point problem which arises from the variational formulation of the Navier–Stokes equations. The basic idea is to split the numerical treatment of the various operators in the equations, thus decomposing the initially difficult problem into relatively easier substeps. There are several ways to perform such splitting and therefore a variety of fractional-step methods for the unsteady Navier–Stokes equations do exist. The books by Quartapelle (1993), Quarteroni and Valli (1994), Gresho and Sani (2000), and references therein, should be consulted for a detailed exposition of fractional-step methods.

Fractional-step methods described here perform time discretization before the spacial discretization. When this approach is adopted, a controversy arises on which boundary conditions must be imposed at each step, because the intermediate semi-discrete problems must be well-posed. Another important feature in fractional-step methods is the overall order of accuracy with respect to time discretization. Most methods are first-order accurate, but some second-order accurate methods have also been developed. Both types of methods will be considered in this section.

While the first step in the time discretization which treats the convective and viscous terms in the Navier–Stokes equations might be treated by an explicit algorithm, the final step describing the pressure/incompressibility phase must necessarily be treated by an implicit time integration scheme. Here, an approach is adopted which gives the choice of either an explicit, a semi-implicit, or a fully implicit time-stepping scheme for the first step. Note, however, that adopting a fully implicit scheme for the first step considerably increases the computational burden due to the nonlinearity of the convective terms in the Navier–Stokes equations. Semi-implicit methods are therefore generally preferred in large-scale computations.

6.7.3.1 Chorin–Temam projection method The principle of the projection method is to compute the velocity and pressure fields separately through the computation of an intermediate velocity, which is then projected onto the subspace of the solenoidal vector functions. A detailed account of the method can be found in the textbook by Temam (2001). Basic to the derivation of projection methods is a theorem of orthogonal decomposition due to Ladyzhenskaya (1969), which is based on the Helmholtz decomposition principle. The theorem states that any vector field $\boldsymbol{w}$ in Ω admits the unique orthogonal decomposition

$$\boldsymbol{w} = \boldsymbol{v} + \boldsymbol{\nabla}\phi \tag{6.36}$$

into a solenoidal field, $\boldsymbol{v}$, with zero normal component on the domain boundary (i.e., $\boldsymbol{\nabla}\cdot\boldsymbol{v} = 0$ and $\boldsymbol{n}\cdot\boldsymbol{v} = 0$ on Γ) and the gradient of some scalar function ϕ, see also the Remark 6.8. In the present context, an intermediate velocity field, $\boldsymbol{v}_{\text{int}}^{n+1}$, is decomposed into the sum of a solenoidal velocity field, $\boldsymbol{v}^{n+1}$, and the gradient of a scalar function proportional to the unknown pressure, namely, $\boldsymbol{\nabla}p^{n+1}$.

For simplicity, we shall consider a purely Dirichlet problem: that is, we prescribe the condition $\boldsymbol{v} = \boldsymbol{v}_D$ on the boundary Γ of the computational domain Ω. The Chorin–Temam projection method includes two basic steps as follows.

The *first step* includes the viscous and convective terms in the Navier–Stokes equations (6.34) and, given the previous time-step velocity field $\boldsymbol{v}^n$, consists of finding an intermediate velocity field, $\boldsymbol{v}_{\text{int}}^{n+1}$, such that

$$\begin{cases} \dfrac{\boldsymbol{v}_{\text{int}}^{n+1} - \boldsymbol{v}^n}{\Delta t} + (\boldsymbol{v}^* \cdot \boldsymbol{\nabla})\,\boldsymbol{v}^{**} - \nu\boldsymbol{\nabla}^2\boldsymbol{v}^{**} = \boldsymbol{b}^{n+1} & \text{in } \Omega, \\ \boldsymbol{v}_{\text{int}}^{n+1} = \boldsymbol{v}_D^{n+1} & \text{on } \Gamma, \end{cases} \tag{6.37}$$

where the velocities $\boldsymbol{v}^*$ and $\boldsymbol{v}^{**}$ must be chosen suitably for the treatment of the nonlinear convective term, possible options are

$$\begin{aligned} &\boldsymbol{v}^* = \boldsymbol{v}^{**} = \boldsymbol{v}^n && \text{for the explicit Euler method,} \\ &\boldsymbol{v}^* = \boldsymbol{v}^n \quad \text{and} \quad \boldsymbol{v}^{**} = \boldsymbol{v}_{\text{int}}^{n+1} && \text{for a semi-implicit method,} \\ &\boldsymbol{v}^* = \boldsymbol{v}^{**} = \boldsymbol{v}_{\text{int}}^{n+1} && \text{for the implicit Euler method.} \end{aligned}$$

Note that the complete Dirichlet boundary conditions are imposed in this first step. This is due to the fact that this step includes the viscous term.

To construct a finite element version of the fractional-step method, a weak form of the first-step equations (6.37) is necessary. The problem requires to find the intermediate velocity $\boldsymbol{v}_{\text{int}}^{n+1} \in \boldsymbol{\mathcal{S}}_{\text{int}}$, such that for all $\boldsymbol{w} \in \boldsymbol{\mathcal{V}}_{\text{int}}$

$$\left(\boldsymbol{w}, \frac{\boldsymbol{v}_{\text{int}}^{n+1} - \boldsymbol{v}^{n}}{\Delta t}\right) + c(\boldsymbol{v}^{*}; \boldsymbol{w}, \boldsymbol{v}^{**}) + a(\boldsymbol{w}, \boldsymbol{v}^{**}) = (\boldsymbol{w}, \boldsymbol{b}^{n+1}), \tag{6.38}$$

where the trilinear and bilinear forms have already been defined, see for instance Section 6.6.1 and equation 6.17. Note that the functional spaces $\boldsymbol{\mathcal{S}}_{\text{int}}$ and $\boldsymbol{\mathcal{V}}_{\text{int}}$ are such that the complete Dirichlet boundary conditions are verified, namely $\boldsymbol{v}_{\text{int}}^{n+1} = \boldsymbol{v}_{D}^{n+1}$ on Γ.

For the semi-implicit and fully implicit cases, the algebraic system resulting from the finite element discretization is

$$\mathbf{M}_{1}\left(\frac{\mathbf{v}_{\text{int}}^{n+1} - \mathbf{v}^{n}}{\Delta t}\right) + \big(\mathbf{C}(\boldsymbol{v}^{*}) + \mathbf{K}\big)\, \mathbf{v}_{\text{int}}^{n+1} = \mathbf{f}^{n+1}, \tag{6.39}$$

where $\mathbf{M}_1$ is the consistent mass matrix, $\mathbf{C}$ is the convection matrix defined in (6.33), $\mathbf{K}$ is the viscosity matrix identical to the one already defined in Section 6.5.4, and vector $\mathbf{f}^{n+1}$ accounts for the applied body force $\boldsymbol{b}$ and the Dirichlet boundary conditions.

Note the computational complexity of the fully implicit option, $\boldsymbol{v}^{*} = \boldsymbol{v}_{\text{int}}^{n+1}$, for time integration, which requires repeated computations of the inverse of the nonlinear and non-symmetric matrix $\mathbf{M}_1 + \Delta t \left(\mathbf{C}(\boldsymbol{v}_{\text{int}}^{n+1}) + \mathbf{K}\right)$. In this case, predictor–corrector methods are usually employed to solve (6.39). In the semi-implicit case, $\boldsymbol{v}^{*} = \boldsymbol{v}^{n}$, a modification of the convective term is required to maintain unconditional stability as explained in Remark 6.18.

The *second step* of the Chorin–Temam method determines the end-of-step velocity $\boldsymbol{v}^{n+1}$ and pressure p^{n+1} solving

$$\begin{cases} \dfrac{\boldsymbol{v}^{n+1} - \boldsymbol{v}_{\text{int}}^{n+1}}{\Delta t} + \boldsymbol{\nabla} p^{n+1} = 0 & \text{in } \Omega, \\ \boldsymbol{\nabla} \cdot \boldsymbol{v}^{n+1} = 0 & \text{in } \Omega, \\ \boldsymbol{n} \cdot \boldsymbol{v}^{n+1} = \boldsymbol{n} \cdot \boldsymbol{v}_{D}^{n+1} & \text{on } \Gamma. \end{cases} \tag{6.40}$$

Note that this second step includes the remaining term (pressure) and equation (incompressibility) of the Navier–Stokes equations. Now, the boundary condition only prescribes the normal component of the velocity (not the tangential components). This is a crucial aspect of this fractional-step method: the tangential components of the velocity cannot be controlled on the boundary in accordance with the Helmholtz decomposition principle, which only allows us to prescribe a condition on the normal component of the velocity. The first equation in (6.40) is

$$\boldsymbol{v}_{\text{int}}^{n+1} = \boldsymbol{v}^{n+1} + \Delta t\, \boldsymbol{\nabla} p^{n+1}, \quad \text{or} \quad \boldsymbol{v}^{n+1} = \boldsymbol{v}_{\text{int}}^{n+1} - \Delta t\, \boldsymbol{\nabla} p^{n+1}. \tag{6.41}$$

In the particular case of homogeneous boundary values for the normal component of the velocity (i.e., $\boldsymbol{n}\cdot\boldsymbol{v}_D = 0$), this is precisely the orthogonal decomposition applied to the intermediate velocity $\boldsymbol{v}_{\text{int}}^{n+1} = \boldsymbol{w}$ in (6.36). Thus, $\boldsymbol{\nabla}\cdot\boldsymbol{v}^{n+1} = 0$ and $\boldsymbol{n}\cdot\boldsymbol{v}^{n+1} = 0$ on Γ. This shows that the fractional-step method involves an orthogonal projection operator $\mathbb{P}$, see Remark 6.8, such that

$$\boldsymbol{v}^{n+1} = \mathbb{P}\,\boldsymbol{v}_{\text{int}}^{n+1} \quad \text{and} \quad \Delta t\,\boldsymbol{\nabla} p^{n+1} = (\mathbf{I}-\mathbb{P})\,\boldsymbol{v}_{\text{int}}^{n+1}.$$

The weak form of the second-step equations (6.40) is given as follows: find the end-of-step velocity $\boldsymbol{v}^{n+1} \in \boldsymbol{\mathcal{S}}$ and the pressure $p^{n+1} \in \mathcal{Q}$, such that, for all $(\boldsymbol{w}, q) \in \boldsymbol{\mathcal{V}}\times\mathcal{Q}$,

$$\begin{cases} \left(\boldsymbol{w}, \dfrac{\boldsymbol{v}^{n+1} - \boldsymbol{v}_{\text{int}}^{n+1}}{\Delta t}\right) + b(\boldsymbol{w}, p^{n+1}) = 0, \\ b(\boldsymbol{v}^{n+1}, q) = 0. \end{cases} \tag{6.42}$$

Now, the functional spaces $\boldsymbol{\mathcal{S}}$ and $\boldsymbol{\mathcal{V}}$ are such that the solution verifies the prescribed boundary conditions, $\boldsymbol{n}\cdot\boldsymbol{v}^{n+1} = \boldsymbol{n}\cdot\boldsymbol{v}_D^{n+1}$ on Γ.

The discrete equations emanating from the discretization of (6.42) induce the following system of algebraic equations:

$$\begin{cases} \mathbf{M}_2\left(\dfrac{\mathbf{v}^{n+1} - \mathbf{v}_{\text{int}}^{n+1}}{\Delta t}\right) + \mathbf{G}\,\mathbf{p}^{n+1} = 0, \\ \mathbf{G}^T\mathbf{v}^{n+1} = 0, \end{cases}$$

or equivalently,

$$\begin{pmatrix} \mathbf{M}_2/\Delta t & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{v}^{n+1} \\ \mathbf{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{M}_2\mathbf{v}_{\text{int}}^{n+1}/\Delta t \\ \mathbf{0} \end{pmatrix}.$$

Note that this system has the same structure as the one obtained for the Stokes problem, see (6.21), where the viscous matrix is now replaced by a mass matrix. As already discussed in Section 6.5.5 this system can be solved in two steps, first compute the pressure field from

$$(\mathbf{G}^T\mathbf{M}_2^{-1}\mathbf{G})\,\mathbf{p}^{n+1} = \frac{1}{\Delta t}\mathbf{G}^T\,\mathbf{v}_{\text{int}}^{n+1}$$

and then compute the end-of-step velocity from

$$\mathbf{M}_2\mathbf{v}^{n+1} = \mathbf{M}_2\mathbf{v}_{\text{int}}^{n+1} - \Delta t\,\mathbf{G}\mathbf{p}^{n+1},$$

see also (6.41). Recall that when Dirichlet conditions are precibed on the whole boundary, a reference value of pressure must be specified at an arbitrary pressure node because $\mathcal{Q} = \mathcal{L}_2(\Omega)/\mathbb{R}$.

The solvability issues discussed in Section 6.5.5 are automatically verified here. That is, this algorithm provides a unique solution which converges to the exact one for any velocity–pressure pair provided that the boundary is smooth (the intermediate velocity also converges to the exact one). This property and the fact that velocity and pressure are uncoupled (recall that the mass matrix is adequately approximated by the lumped one) have contributed to the popularity of this method.

Nevertheless, this scheme is first order in time; although modified schemes, for instance with iterative procedures see Bell, Colella and Glaz (1989), have been developed for higher-order methods. Another drawback is that the end-of-step velocity $\boldsymbol{v}^{n+1}$ does not satisfy exactly the Dirichlet boundary conditions, this may cause an non-physical boundary layer of $\mathcal{O}\left(\sqrt{\nu\Delta t}\right)$, see Remark 6.20 and the papers by Gresho (1990) and Temam (1991).

Finally, it is important to note that the condition on the smoothness of boundary Γ is, in some cases, crucial. For instance, the cavity flow problem, which is studied in Section 6.8.2, does not present a smooth boundary. One can verify that the bilinear equal-interpolation for velocity and pressure, that is the Q1Q1 element (see Figure 6.1), induces a non regular $\mathbf{G}^T\mathbf{M}_2^{-1}\mathbf{G}$ matrix. Note that this is not the case for the same problem and another element not passing the LBB condition: the Q1P0 element.

Remark 6.16 (Incremental projection scheme). The original Chorin–Temam method can be modified to incorporate the pressure increment $p^{n+1}-p^n$, instead of the total pressure, in the pressure/incompressibility phase. This leads to a projection method with improved convergence properties. The first step (6.37) of the incremental projection scheme is modified to include the pressure gradient term $\boldsymbol{\nabla} p^n$ and becomes

$$\left\{\begin{array}{ll}\dfrac{\boldsymbol{v}_{\text{int}}^{n+1}-\boldsymbol{v}^n}{\Delta t}+(\boldsymbol{v}^*\cdot\boldsymbol{\nabla})\,\boldsymbol{v}^{**}-\nu\boldsymbol{\nabla}^2\boldsymbol{v}^{**}=\boldsymbol{b}^{n+1}-\boldsymbol{\nabla} p^n & \text{in }\Omega,\\ \boldsymbol{v}_{\text{int}}^{n+1}=\boldsymbol{v}_D^{n+1} & \text{on }\Gamma,\end{array}\right.$$

while the projection step is replaced by

$$\left\{\begin{array}{ll}\dfrac{\boldsymbol{v}^{n+1}-\boldsymbol{v}_{\text{int}}^{n+1}}{\Delta t}+\boldsymbol{\nabla}(p^{n+1}-p^n)=0 & \text{in }\Omega,\\ \boldsymbol{\nabla}\cdot\boldsymbol{v}^{n+1}=0 & \text{in }\Omega,\\ \boldsymbol{n}\cdot\boldsymbol{v}^{n+1}=\boldsymbol{n}\cdot\boldsymbol{v}_D^{n+1} & \text{on }\Gamma.\end{array}\right.$$

Remark 6.17 (Treatment of Neumann-type boundary conditions). When Neumann boundary conditions, see (6.34d), are present, the condition

$$-p\boldsymbol{n}+\nu(\boldsymbol{n}\cdot\boldsymbol{\nabla})\boldsymbol{v}=\boldsymbol{t}$$

is decomposed in order to include the viscous part of the prescribed boundary traction in the first step and the pressure contribution in the second step. The process is illustrated in a paper by Laval and Quartapelle (1990). This splitting is also feasible when a velocity–pressure stress-divergence formulation is employed, recall Remarks 6.3, 6.11 and 6.12.

Remark 6.18 (Treatment of convective terms). As previously discussed, in the first step, convective terms can be treated in various ways, from fully explicit to fully implicit. The drawback of implicit integration is the need to solve nonlinear and non-symmetric systems at each time step. Despite their conditional stability, explicit schemes are advantageous because the linear algebraic systems that must be solved are symmetric and positive definite. Thus, effective iterative solvers can be employed.

The interesting features of a semi-implicit approach are the linearization of the convective term, the possible unconditional stability of the scheme, and, as shown by Guermond and Quartapelle (1998b), the possibility of eliminating the velocity update in the incompressibility step. To guarantee unconditional stability in the case of a semi-implicit time integration, the skew-symmetric form of the convective term must be used, see Remark 6.15.

Remark 6.19 (Elimination of the end-of-step velocity). Since the intermediate velocity also converges to the exact one, using the skew-symmetric form of the convective term, Guermond and Quartapelle (1997; 1998b) rewrite the first step, see (6.37), in the semi-implicit form

$$\begin{cases} \dfrac{\boldsymbol{v}_{\text{int}}^{n+1} - \boldsymbol{v}^n}{\Delta t} + (\boldsymbol{v}_{\text{int}}^n \cdot \boldsymbol{\nabla})\boldsymbol{v}_{\text{int}}^{n+1} + \dfrac{1}{2}(\boldsymbol{\nabla} \cdot \boldsymbol{v}_{\text{int}}^n)\boldsymbol{v}_{\text{int}}^{n+1} - \nu \boldsymbol{\nabla}^2 \boldsymbol{v}_{\text{int}}^{n+1} = \boldsymbol{b}^{n+1} & \text{in } \Omega, \\ \boldsymbol{v}_{\text{int}}^{n+1} = \boldsymbol{v}_D^{n+1} & \text{on } \Gamma, \end{cases}$$

where the convective velocity is $\boldsymbol{v}_{\text{int}}^n$ instead of $\boldsymbol{v}^n$. Using now the momentum equation of the second step, equation (6.41), at t^n (i.e., $\boldsymbol{v}^n = \boldsymbol{v}_{\text{int}}^n - \Delta t\, \boldsymbol{\nabla} p^n$) the first equation can be expressed in terms of the intermediate velocity only:

$$\frac{\boldsymbol{v}_{\text{int}}^{n+1} - \boldsymbol{v}_{\text{int}}^n}{\Delta t} + (\boldsymbol{v}_{\text{int}}^n \cdot \boldsymbol{\nabla})\, \boldsymbol{v}_{\text{int}}^{n+1} + \frac{1}{2}(\boldsymbol{\nabla} \cdot \boldsymbol{v}_{\text{int}}^n)\boldsymbol{v}_{\text{int}}^{n+1} - \nu \boldsymbol{\nabla}^2 \boldsymbol{v}_{\text{int}}^{n+1} = \boldsymbol{b}^{n+1} - \boldsymbol{\nabla} p^n.$$

Since this algorithm has eliminated the end-of-step velocity completely, if needed it is computed from (6.41). Finally, observe that this fractional-step scheme requires the evaluation of pressure at each step. This can be done with a pressure Poisson equation, see the next remark, with no evaluation of the end-of-step velocity.

Remark 6.20 (Pressure Poisson equation). Observe that the equations of the second step can be reformulated in terms of a Poisson equation for the pressure. In fact, applying the divergence operator to the first equation in (6.40), since $\boldsymbol{\nabla} \cdot \boldsymbol{v}^{n+1} = 0$, we obtain the following Neumann problem for the pressure:

$$\begin{cases} \boldsymbol{\nabla}^2 p^{n+1} = \dfrac{1}{\Delta t} \boldsymbol{\nabla} \cdot \boldsymbol{v}_{\text{int}}^{n+1} & \text{in } \Omega, \\ \boldsymbol{n} \cdot \boldsymbol{\nabla} p^{n+1} = 0 & \text{on } \Gamma. \end{cases}$$

Once the end-of-step pressure p^{n+1} is determined, the end-of-step velocity $\boldsymbol{v}^{n+1}$, if needed, can be computed by the explicit relation (6.41), that is the first equation in (6.40).

The Neumann boundary conditions for the end-of-step pressure are obtained using (6.41) in the boundary condition of the second step, namely $\boldsymbol{n} \cdot \boldsymbol{v}^{n+1} = \boldsymbol{n} \cdot \boldsymbol{v}_D^{n+1}$, and using the fact that $\boldsymbol{v}_{\text{int}}^{n+1} = \boldsymbol{v}_D^{n+1}$ from the first step. There has been some controversy on these non-physical boundary conditions for pressure. Note that the exact pressure for sufficiently smooth conditions does not verify homogeneous boundary conditions (see Gresho and Sani, 1987). Together with the weak representation of the tangential velocity on the boundary, the non-physical boundary conditions for the pressure represent a potential drawback of the Chorin–Temam method.

Remark 6.21 (Viscosity splitting fractional-step method). To alleviate the difficulties regarding the imposition of Dirichlet boundary conditions in the second step of the Chorin–Temam projection method, Blasco, Codina and Huerta (1997; 1998) introduced a viscosity splitting fractional-step method in which the second step avoids using the projection idea. Instead, they introduce a diffusion term in the momentum equation of the second step, which consequently loses its inviscid character responsible for preventing control of the prescribed tangential component of the velocity at the boundary. The second step then consists of determining the end-of-step velocity $\boldsymbol{v}^{n+1}$ and pressure p^{n+1} as being the solution of

$$\begin{cases} \dfrac{\boldsymbol{v}^{n+1} - \boldsymbol{v}_{\text{int}}^{n+1}}{\Delta t} - \nu \boldsymbol{\nabla}^2 \left(\boldsymbol{v}^{n+1} - \boldsymbol{v}_{\text{int}}^{n+1}\right) + \boldsymbol{\nabla} p^{n+1} = 0 & \text{in } \Omega, \\ \boldsymbol{\nabla} \cdot \boldsymbol{v}^{n+1} = 0 & \text{in } \Omega, \\ \boldsymbol{v}^{n+1} = \boldsymbol{v}_D^{n+1} & \text{on } \Gamma. \end{cases}$$

When combined with the equations (6.37) of the first step, this formulation of the second step allows the imposition of the original Dirichlet boundary conditions in both phases of the fractional-step method. This method, however, requires finite elements passing the LBB condition.

Remark 6.22 (The LBB condition). It was initially thought that Poisson-based projection techniques could be used with velocity–pressure pairs not satisfying the LBB condition and that this family of methods could thus be considered as pertaining to the class of stabilized finite element methods. This is unfortunately not always true. Guermond and Quartapelle (1998a; 1998b) performed an in-depth study of the stability and convergence properties of fractional-step projection methods in which the pressure/incompressibility step is recast in terms of a Poisson equation for the pressure. Their conclusion is that in the case of the incremental projection scheme, the velocity–pressure pairs must satisfy the LBB compatibility condition to obtain non-oscillatory numerical results. By contrast, when the total pressure is employed in the second step, they found that equal-order interpolations could be safely used, provided the time step is not too small with respect to the spatial mesh size, in the sense that $\Delta t \geq c\, h^k$, k being the degree of the velocity interpolation and h a measure of the mesh size. Unfortunately, the non-incremental projection method has inferior convergence properties with respect to the incremental scheme. Minev (2001) presents a discussion on which fractional-step methods require an LBB compliant approximation and which schemes do not.

6.7.3.2 Algebraic splitting The Chorin–Temam projection method, which is based on the Helmholtz decomposition principle, can be viewed as a *physical splitting* of the original incompressible Navier–Stokes equations. As an alternative method for solving the Navier–Stokes problem in a sequence of simpler steps, Perot (1993) and Abdallah (1995) in a finite volume context and more recently Quarteroni, Saleri and Veneziani (2000) in the finite element context introduced an *algebraic splitting*. It is based on an incomplete (or approximate) block LU factorization of the original partitioned matrix system arising from the space–time discretization of the unsteady Navier–Stokes equations (6.35), namely

$$\begin{pmatrix} \mathbf{B} & \mathbf{G} \\ \mathbf{G}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}^{n+1} \\ \mathbf{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{f}^* \\ \mathbf{h} \end{pmatrix}, \tag{6.43}$$

where $\mathbf{f}^*$ accounts for the r.h.s. term in the first equation in (6.35) and all the known terms (evaluated at t^n) resulting from the time discretization of the momentum equation. To simplify the notation, we have posed

$$\mathbf{B} = \frac{1}{\Delta t}\mathbf{M} + \big(\mathbf{K} + \mathbf{C}(\boldsymbol{v}^{n+1})\big). \tag{6.44}$$

As already mentioned in Section 6.5.5 for the Stokes problem, the partitioned system (6.43) could in principle be solved in two steps (6.26) and (6.27), namely

$$\begin{cases} \left(\mathbf{G}^T\mathbf{B}^{-1}\mathbf{G}\right)\mathbf{p}^{n+1} = \mathbf{G}^T\mathbf{B}^{-1}\mathbf{f}^* - \mathbf{h}, \\ \mathbf{B}\mathbf{u}^{n+1} = \mathbf{f}^* - \mathbf{G}\,\mathbf{p}^{n+1}. \end{cases}$$

The second equation is obtained making the first equation in (6.43) explicit with respect to $\mathbf{u}^{n+1}$ and the pressure equation, which is solved first, is obtained after this expression for $\mathbf{u}^{n+1}$ is introduced in the second equation of (6.43).

Unfortunately, this strategy is computationally unaffordable in engineering applications. The reduction in computational complexity is usually limited to replace $\mathbf{B}^{-1}$ by a simpler matrix. From (6.44) we have

$$\mathbf{B}^{-1} = \Delta t\Big(\mathbf{I} + \Delta t\mathbf{M}^{-1}\big(\mathbf{K} + \mathbf{C}(\boldsymbol{v}^{n+1})\big)\Big)^{-1}\mathbf{M}^{-1}.$$

The standard approach is to use the first-order approximation of matrix $\mathbf{B}^{-1}$ with respect to Δt given by

$$\mathbf{B}^{-1} \approx \mathbf{H} = \Delta t\mathbf{M}^{-1}.$$

Since, by contrast with matrix $\mathbf{B}^{-1}$, matrix $\mathbf{M}^{-1}$ remains invariant during the transient calculation, a significant reduction of the computational effort is obtained by this approximation. Higher-order expressions can also be devised, the second-order approximation is

$$\mathbf{H} = \Delta t(\mathbf{I} - \Delta t\mathbf{M}^{-1}(\mathbf{K} + \mathbf{C}))\mathbf{M}^{-1}$$

and the third-order one becomes

$$\mathbf{H} = \Delta t\Big(\mathbf{I} - \Delta t\mathbf{M}^{-1}(\mathbf{K}+\mathbf{C}) + \Delta t^2\big(\mathbf{M}^{-1}(\mathbf{K}+\mathbf{C})\big)^2\Big)\mathbf{M}^{-1},$$

where $\mathbf{C}$ is non-symmetric and varies with time. Note that only the inverse of the mass matrix is required (recall that the lumped mass matrix is a good approximation of the consistent one).

Perot (1993) and Quarteroni et al. (2000) generalize this approach using the exact LU factorization of the matrix in (6.43),

$$\begin{pmatrix} \mathbf{B} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{B} & \mathbf{0} \\ \mathbf{G}^T & -\mathbf{G}^T\mathbf{B}^{-1}\mathbf{G} \end{pmatrix} \begin{pmatrix} \mathbf{I} & \mathbf{B}^{-1}\mathbf{G} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}.$$

This process is indeed equivalent to a direct solution of system (6.43). Its interest lies in the fact that suitable approximations of matrix $\mathbf{B}^{-1}$ allow the construction of more economic solution schemes. Now $\mathbf{B}^{-1}$ can be replaced by two simpler matrices, one in the L-block, $\mathbf{H}_1$, and one in the U-block, $\mathbf{H}_2$. Thus, the original matrix is approximated by

$$\begin{pmatrix} \mathbf{B} & \mathbf{0} \\ \mathbf{G}^T & -\mathbf{G}^T\mathbf{H}_1\mathbf{G} \end{pmatrix} \begin{pmatrix} \mathbf{I} & \mathbf{H}_2\mathbf{G} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} = \begin{pmatrix} \mathbf{B} & \mathbf{B}\mathbf{H}_2\mathbf{G} \\ \mathbf{G}^T & \mathbf{G}^T(\mathbf{H}_2 - \mathbf{H}_1)\mathbf{G} \end{pmatrix}. \tag{6.45}$$

From this inexact factorization three strategies become apparent.

First, if $\mathbf{H} = \mathbf{H}_1 = \mathbf{H}_2$ the previously discussed methodology is recovered. Note that this strategy is the only one that preserves mass because the lower-left block of the inexact factorization is a null matrix, see (6.45). The momentum equation, however, is modified by the fact that $\mathbf{H}_2 \neq \mathbf{B}^{-1}$. Projection fractional-step methods, such as the ones discussed in the previous section, belong to this strategy.

Second, if $\mathbf{H}_2 = \mathbf{B}^{-1}$ and $\mathbf{H}_1 \neq \mathbf{H}_2$ momentum is preserved because the momentum equation is not modified but the mass conservation equation is modified. Quasi-compressible (i.e., pressure-stabilized, penalty, etc.) methods belong to this category, see for instance (6.28) and (6.29).

Third, if $\mathbf{H}_2 \neq \mathbf{B}^{-1}$ and $\mathbf{H}_1 \neq \mathbf{H}_2$ a more general strategy is devised where both the momentum and continuity equations are perturbed, see for instance Section 6.7.2.

The inexact factorization (6.45) in equation (6.43) allows the computation of $\mathbf{u}^{n+1}$ and $\mathbf{p}^{n+1}$ by means of the following sequence of steps:

$$\text{L-step}\begin{cases} \mathbf{B}\,\mathbf{u}_{\text{int}}^{n+1} = \mathbf{f}^*, \\ \mathbf{G}^T\,\mathbf{u}_{\text{int}}^{n+1} - \mathbf{G}^T\mathbf{H}_1\mathbf{G}\,\mathbf{p}_{\text{int}}^{n+1} = \mathbf{h}, \end{cases} \qquad \text{U-step}\begin{cases} \mathbf{u}^{n+1} + \mathbf{H}_2\mathbf{G}\,\mathbf{p}^{n+1} = \mathbf{u}_{\text{int}}^{n+1}, \\ \mathbf{p}^{n+1} = \mathbf{p}_{\text{int}}^{n+1}, \end{cases}$$

or, equivalently

$$\begin{cases} \mathbf{B}\,\mathbf{u}_{\text{int}}^{n+1} = \mathbf{f}^* & \text{(intermediate velocity computation)}, \\ \mathbf{G}^T\mathbf{H}_1\mathbf{G}\,\mathbf{p}^{n+1} = \mathbf{G}^T\,\mathbf{u}_{\text{int}}^{n+1} - \mathbf{h} & \text{(pressure computation)}, \\ \mathbf{u}^{n+1} = \mathbf{u}_{\text{int}}^{n+1} - \mathbf{H}_2\mathbf{G}\,\mathbf{p}^{n+1} & \text{(end-of-step velocity computation)}. \end{cases}$$

A number of other schemes based on the inexact factorization (6.45) are investigated by Quarteroni et al. (2000).

6.8 APPLICATIONS AND SOLVED EXERCICES

6.8.1 Stokes flow with analytical solution

In order to illustrate the behavior of selected mixed finite elements in the solution of stationary Stokes flow, see Section 6.5.2, we consider a two-dimensional problem in the square domain $\Omega =]0,1[\times]0,1[$, which possesses a closed-form analytical solution. The problem consists of determining the velocity field $\boldsymbol{v} = (v_1, v_2)$ and the pressure p such that

$$\begin{cases} -\nu\nabla^2\boldsymbol{v} + \boldsymbol{\nabla} p = \boldsymbol{b} & \text{in } \Omega, \\ \boldsymbol{\nabla}\cdot\boldsymbol{v} = 0 & \text{in } \Omega, \\ \boldsymbol{v} = 0 & \text{on } \Gamma, \end{cases}$$

where the fluid viscosity is taken as $\nu = 1$. The components of the body force $\boldsymbol{b}$ are prescribed as

$$\begin{aligned} b_1 &= (12 - 24\,y)\,x^4 + (-24 + 48\,y)\,x^3 + (-48\,y + 72\,y^2 - 48\,y^3 + 12)\,x^2 \\ &\qquad + (-2 + 24\,y - 72\,y^2 + 48\,y^3)\,x + 1 - 4\,y + 12\,y^2 - 8\,y^3, \\ b_2 &= (8 - 48\,y + 48\,y^2)\,x^3 + (-12 + 72\,y - 72\,y^2)\,x^2 \\ &\qquad + (4 - 24\,y + 48\,y^2 - 48\,y^3 + 24\,y^4)\,x - 12\,y^2 + 24\,y^3 - 12\,y^4, \end{aligned}$$

where, for simplicity, we have used in this 2D problem the notation $(x, y) := (x_1, x_2)$. With this prescribed body force, the exact solution is

$$\begin{aligned} v_1(x,y) &= x^2\,(1-x)^2\,(2\,y - 6\,y^2 + 4\,y^3), \\ v_2(x,y) &= -y^2\,(1-y)^2\,(2\,x - 6\,x^2 + 4\,x^3), \\ p(x,y) &= x\,(1-x). \end{aligned}$$

The square domain is discretized with different uniform meshes: 20 × 20 Q1P0 elements, 20 × 20 Q1Q1 elements, 40 × 40 `Mini` elements, and finally 10 × 10 Q2Q1 elements (the characteristic distance between velocity nodes is kept constant). Figures 6.3 and 6.4 show the pressure field and velocity vectors obtained with the various elements and a Galerkin formulation. Clearly, elements Q1P0 and Q1Q1, which do not satisfy the LBB compatibility condition, exhibit a spurious pressure response. However, one should note that they deliver an acceptable velocity response, see the excellent discussion by Brezzi and Fortin (1991, Sec. II.3.3) entitled “Is the inf-sup condition so important?”. The results obtained with the `Mini` element and the Q2Q1 element, which satisfy the LBB condition, deliver good answers for both velocity and pressure.

These conclusions are clarified in the convergence plots, see Figures 6.5 and 6.6, where uniform meshes have been used with characteristic lengths: $h = 1/10$, $1/20$, $1/30$, and $1/40$. The velocity field converges to the exact solution for all the elements as shown in Figure 6.5 and the convergence rates coincide with the theoretical ones. This is not the case for the pressure field. Figure 6.6 shows two plots, the left one

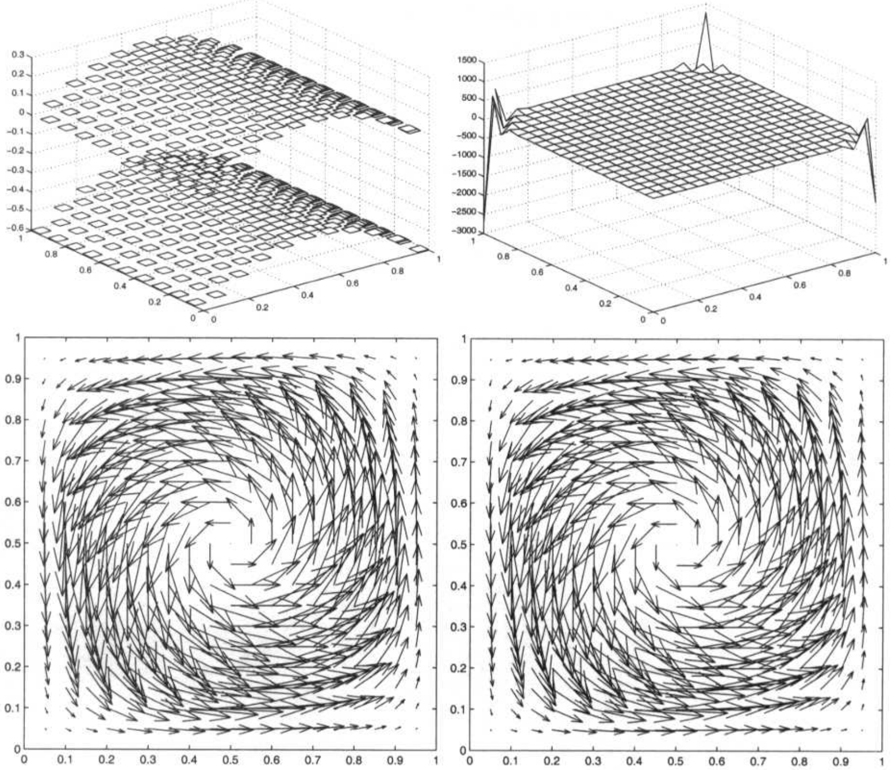


Fig. 6.3 Analytical Stokes: Pressure and velocity for Q1P0 (left) and Q1Q1 (right).

shows the variation of the maximum error norm for all four elements, note that the error scale ranges from 10^{-6} to 10^{6} (one trillion orders of magnitude). The plot on the right only shows the LBB compliant elements and clearly indicates that the pressure field converges to the exact one.

6.8.2 Cavity flow problem

This example has become a standard benchmark test for incompressible flows. We will show here results for the Stokes and Navier–Stokes problems. Figure 6.7 shows a schematic representation of the problem statement. It models a plane flow of an isothermal fluid in a square lid-driven cavity. The upper side of the cavity moves in its own plane at unit speed, while the other sides are fixed.

The boundary conditions are indicated in Figure 6.7. There is a discontinuity in the boundary conditions at the two upper corners of the cavity. Two cases can be envisioned: the two upper corners are either considered as belonging to the top mobile side (leaky cavity), or they are assumed to belong to the fixed vertical walls

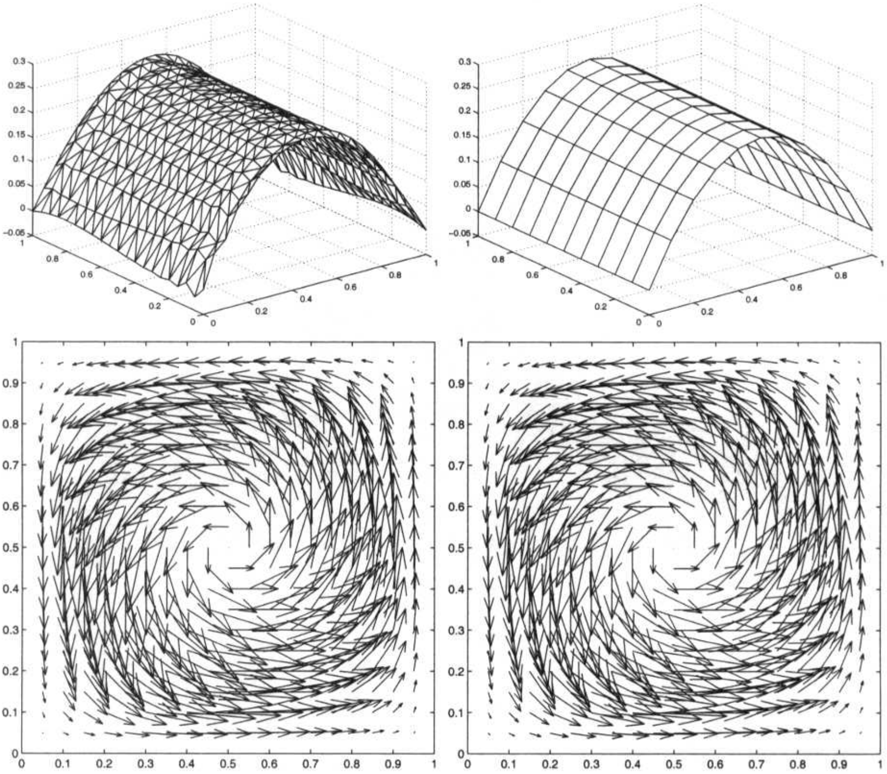


Fig. 6.4 Analytical Stokes: Pressure and velocity for `Mini` (left) and Q2Q1 (right).

(non-leaky). The former case is adopted here. It introduces a singulary in the pressure field precisely at those two upper corners.

Finally, it should be noticed that Dirichlet boundary conditions are imposed on every boundary in this example. As commented earlier this implies that pressure is known up to a constant ($p \in \mathcal{L}_2(\Omega)/\mathbb{R}$). Thus at an arbitrary point, the lower left corner of the cavity, the reference value $p = 0$ is prescribed.

First, we solve the lid-driven cavity for the Stokes problem and the standard Galerkin formulation. The main features in this case are the symmetry with respect to the vertical centerline and the pressure singularity at the two upper corners. In fact, no shear layers are present in the Stokes problem, but results (the pressure jump between both corners) improve if a nonuniform mesh is employed. The cavity is discretized with a nonuniform mesh of 30 × 30 Q1P0 elements, 30 × 30 Q1Q1 elements, 60 × 60 `Mini` elements, and finally 15 × 15 Q2Q1 elements. Thus, all these meshes have the same characteristic element size: $h = 1/30$.

Figure 6.8 shows the symmetric streamlines for the Q2Q1 element. This distribution of streamlines is very similar to the distributions obtained for the other elements. Figure 6.9 shows the pressure field for the non LBB compliant elements.

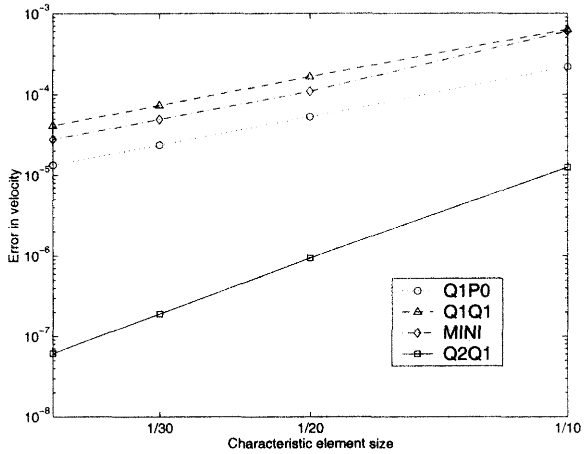


Fig. 6.5 Analytical Stokes: velocity error versus element size.

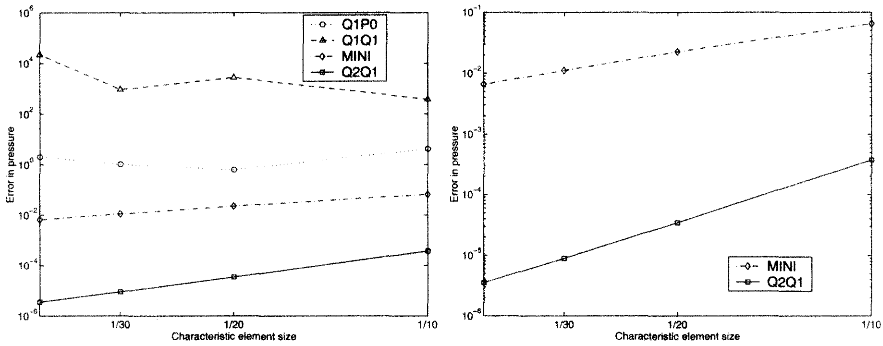


Fig. 6.6 Analytical Stokes: pressure error versus element size.

As expected they present inaccurate pressure results. Both the Q1P0 and the Q1Q1 elements present oscillations which are more pronounced in the corners. The element-to-element oscillations are more obvious on uniform meshes. Both the `Mini` and the Q2Q1, which are LBB compliant, show, as expected, reasonable results for pressure, see the corresponding pressure fields in Figure 6.10.

The stationary Navier–Stokes solution, which is the objective now, is entirely characterized by the Reynolds number,

$$Re = \frac{V_{\mathrm{ref}}\, L_{\mathrm{ref}}}{\nu},$$

ν being the kinematic viscosity of the fluid. The reference velocity used in the Reynolds number is the velocity of the mobile side: $V_{\mathrm{ref}} = 1$. The reference length is the side of the cavity: $L_{\mathrm{ref}} = 1$. The influence of the Reynolds number can be clearly

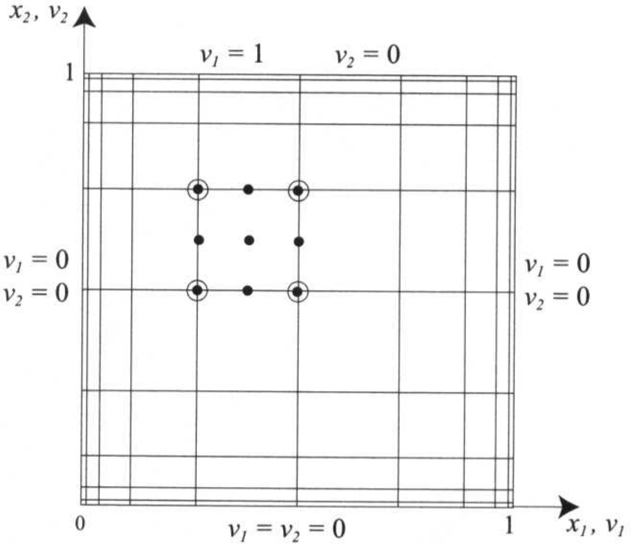


Fig. 6.7 Lid-driven cavity: problem statement with boundary conditions and schematic discretization with Q2Q1 elements. Dots indicate velocity nodes and circles denote pressure nodes.

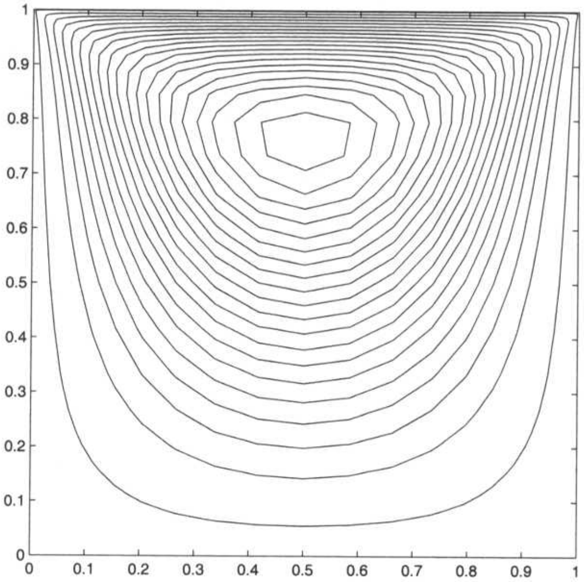


Fig. 6.8 Stokes cavity flow: streamlines for the Q2Q1 element.

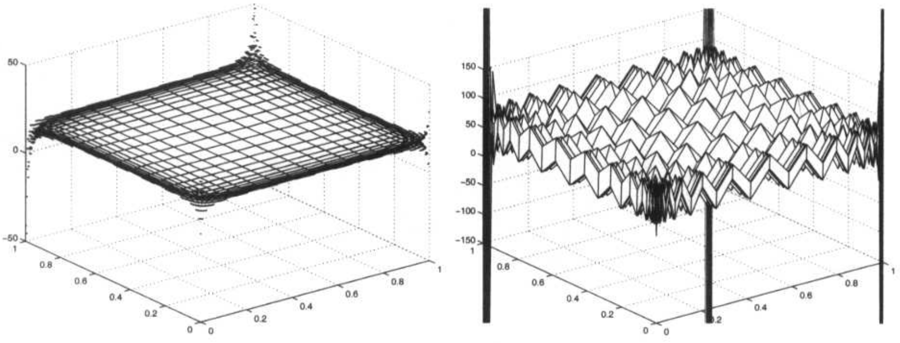


Fig. 6.9 Stokes cavity flow: pressure field for Q1P0 (left) and Q1Q1 (right) elements.

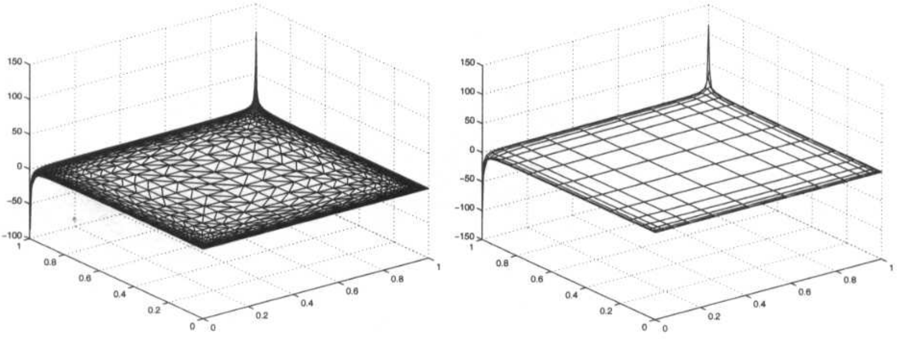


Fig. 6.10 Stokes cavity flow: pressure for `Mini` (left) and Q2Q1 (right) elements.

seen in Figure 6.11 where the velocity profile at the vertical centerline is depicted for the Stokes flow and for Navier–Stokes with $Re = 100$, $Re = 400$ and $Re = 1000$. As the Reynolds number increases boundary layers are more obvious and the variations in the velocity profile become sharper.

We show velocity and pressure results for Reynolds numbers of 100 and 1000. Note that computations are performed using the standard Galerkin finite element method. Stabilized formulations, see Section 6.7.2, must be employed for larger values of the Reynolds number or coarser meshes. An iterative technique (Picard or Newton-Raphson methods) must be employed to iteratively solve the resulting system of nonlinear algebraic equations, equations (6.32).

Results for the cavity flow are displayed in graphical form in Figures 6.12 and 6.13 which allow us to visualize the streamlines and the pressure response. The value and position of the main vortex are indicated in Table 6.2; a comparison with some reference solutions from the literature is also indicated. A satisfactory agreement is observed for all values of the Reynolds number.

As can be seen in these figures and in the table, the position of the main vortex moves towards the center of the cavity when the Reynolds number increases. The

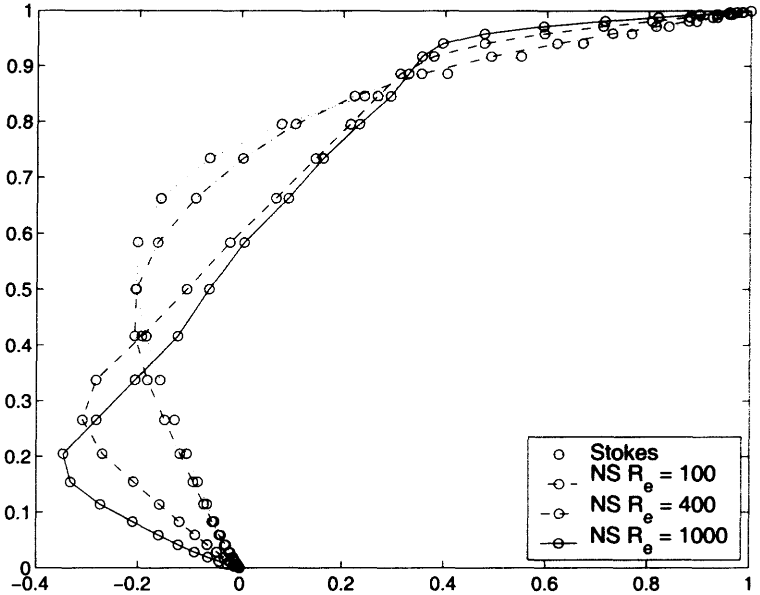


Fig. 6.11 Stokes cavity flow: velocity profiles at the vertical centerline for Stokes and Navier–Stokes with different values of *Re*.

Table 6.2 Position and strength of the cavity main vortex as a function of Reynolds number.

Square cavity		x_1	x_2	Stream Function
$Re = 100$	Present simulation	0.62	0.74	0.103
	Burggraf (1966)	0.62	0.74	0.101
	Tuann and Olson (1978)	0.61	0.722	0.104
$Re = 400$	Present simulation	0.568	0.606	0.110
	Burggraf (1966)	0.560	0.620	0.101
	Tuann and Olson (1978)	0.506	0.583	0.1213
	Ozawa (1975)	0.559	0.614	0.1083
$Re = 1000$	Present simulation	0.540	0.573	0.110
	Ozawa (1975)	0.533	0.569	0.118
	Goda (1979)	0.538	0.575	—

development of a secondary vortex in the right bottom corner of the cavity becomes progressively apparent and a third vortex appears at the lower left corner.

Elevated velocity gradients develop near the cavity walls for large values of the flow Reynolds number. This generates non-physical oscillations in the Galerkin solution for the velocity. A stabilized formulation would then be required, see Section 6.7.2. Solutions of the lid-driven cavity problem obtained with stabilized finite element

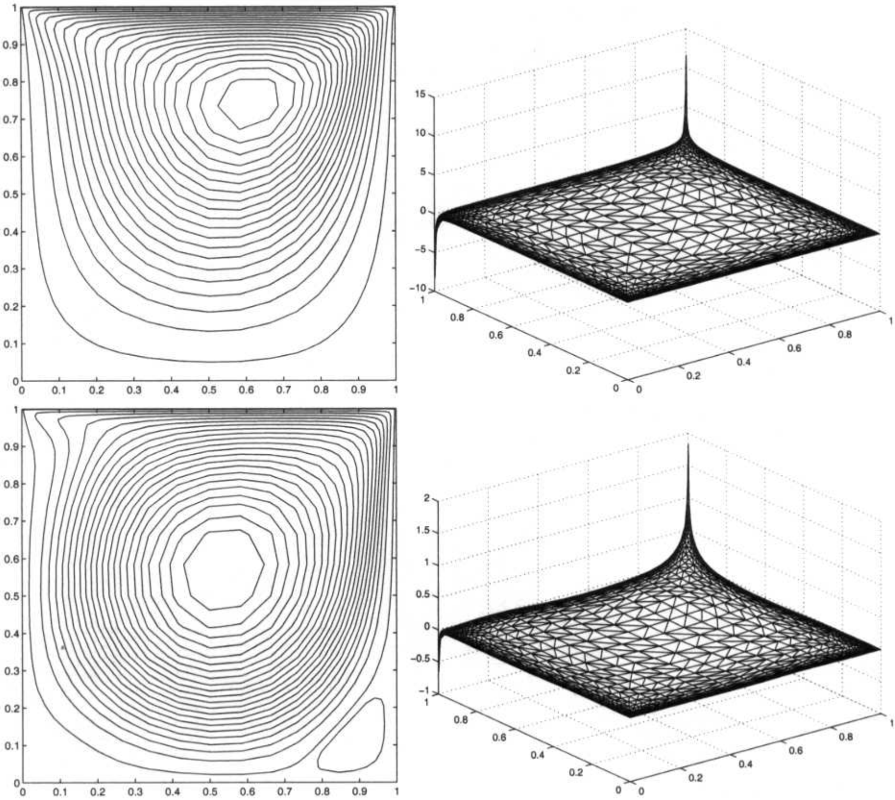


Fig. 6.12 Cavity: `Mini` element, streamlines and pressure for $Re = 100$ (top) and 1000 (bottom).

formulations are reported by Tezduyar et al. (1992), Franca and Frey (1992), Hannani, Stanislas and Dupont (1995), among others.

6.8.3 Plane jet simulation

As an example of a truly transient situation, a plane jet problem is now considered in which the flow domain is the right half-space, namely $\{\boldsymbol{x} \mid x_1 > 0, -\infty < x_2 < \infty\}$. The computational domain is limited to the square defined by $]0,1[\times]0,1[$, which is discretized by a 16×16 uniform mesh of Q2Q1 elements.

The jet aperture is centered at $\boldsymbol{x} = (0.0, 0.5)$ and is $1/16$ wide. The jet profile is parabolic with a maximum velocity of one. The fluid viscosity is taken to be $\nu = 5 \times 10^{-4}$ and its density $\rho = 1$ (non-dimensional variables are used). The boundary conditions simulating the actual flow domain are as follows:

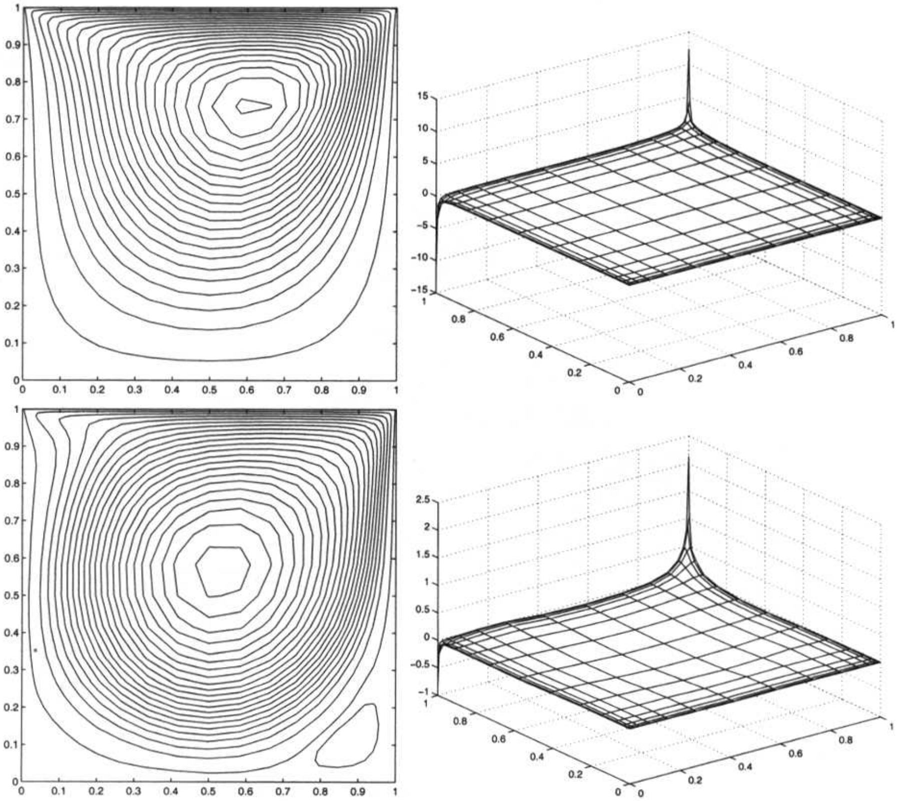
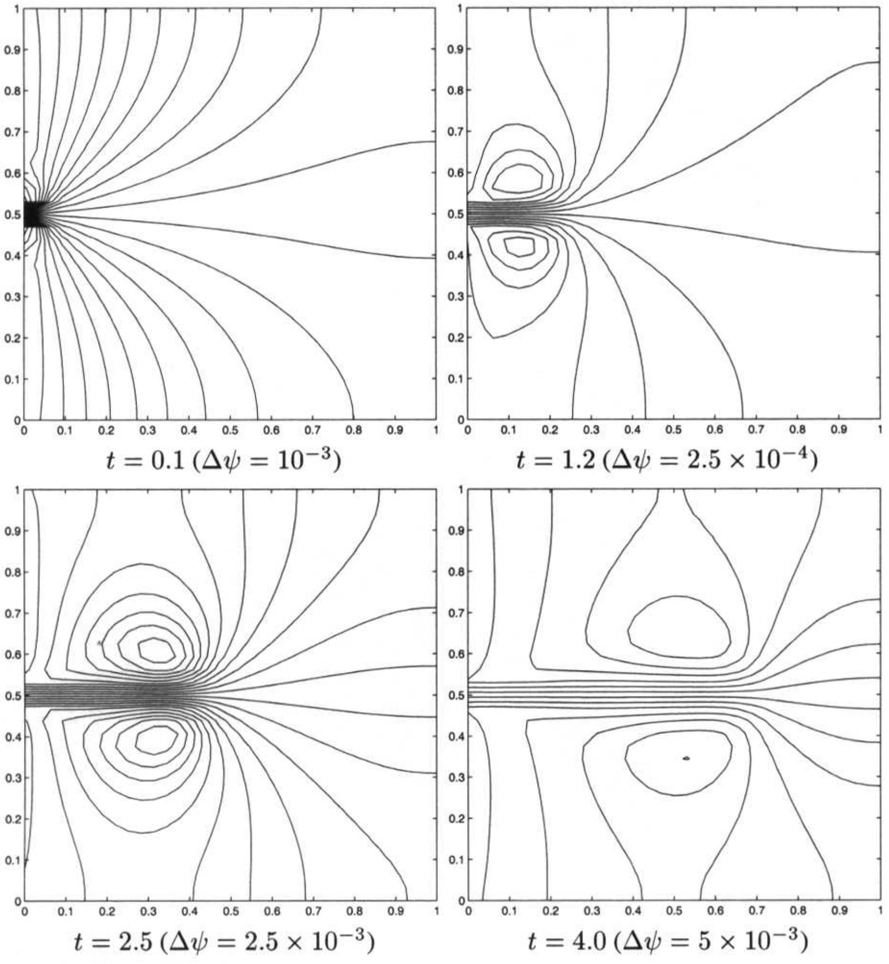


Fig. 6.13 Cavity: Q2Q1, streamlines and pressure for $Re = 100$ (top) and 1000 (bottom).

- On the inflow side, $x_1 = 0$, Dirichlet conditions for the velocity are imposed. The vertical component of the velocity is prescribed to be zero, in addition to the obvious conditions on the horizontal component.
- On the other sides, we want to simulate the situation at the outlet of the computational domain, that is *open/artificial boundary conditions*. The issue of open boundary conditions for the Navier–Stokes equations is still open and has been avoided in this text. It is however accepted, in general, that a good alternative is to impose on the primary variables the condition $\boldsymbol{n} \cdot \boldsymbol{\nabla}(\cdot) = 0$. In the context of the fractional-step method, such a boundary condition is easy to implement. In the first step the homogenous natural boundary condition is simply $\nu(\boldsymbol{n} \cdot \boldsymbol{\nabla})\boldsymbol{v} = \boldsymbol{0}$, which imposes the desired open boundary condition on the velocity. Note that in 2D and with reference to a local system of Cartesian axes $(\boldsymbol{n}, \boldsymbol{\tau})$, this is simply

$$\frac{\partial v_n}{\partial n} = 0 \qquad \text{and} \qquad \frac{\partial v_\tau}{\partial n} = 0.$$



$t = 0.1\ (\Delta\psi = 10^{-3})$

$t = 1.2\ (\Delta\psi = 2.5 \times 10^{-4})$

$t = 2.5\ (\Delta\psi = 2.5 \times 10^{-3})$

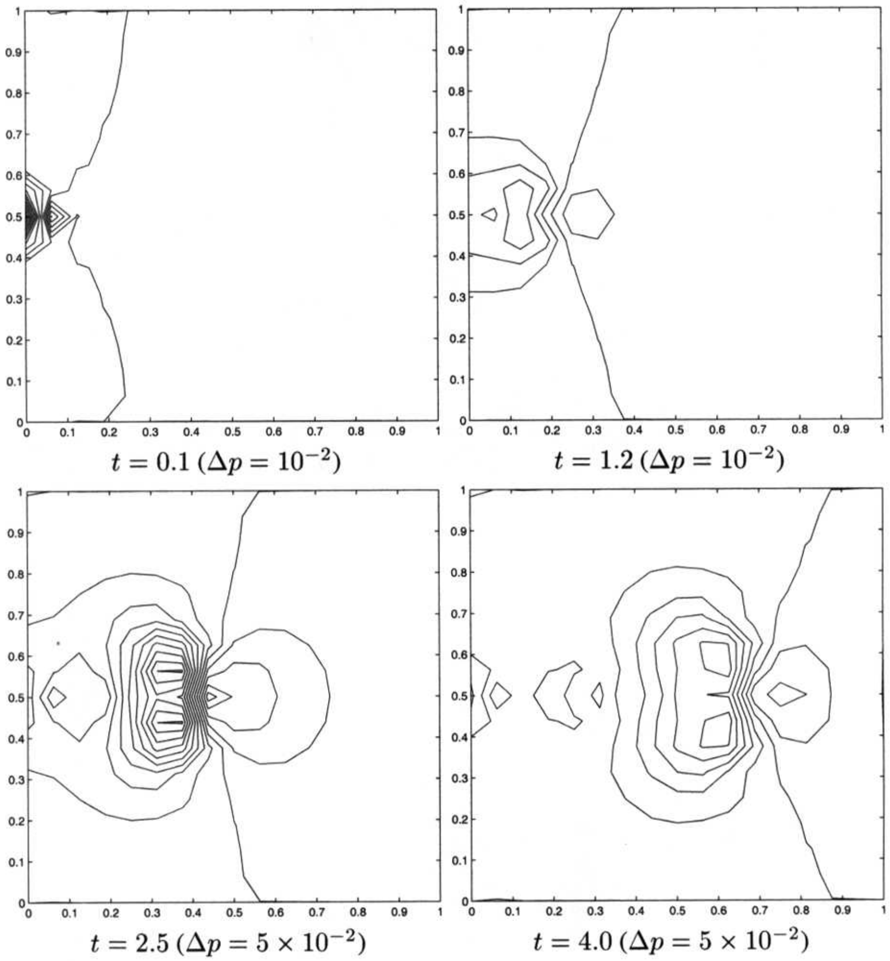
$t = 4.0\ (\Delta\psi = 5 \times 10^{-3})$

Fig. 6.14 Streamlines for plane jet problem

> If the pressure Poisson equation is employed in the second step, see Remark 6.20, again the homogenous natural boundary condition, $\boldsymbol{n} \cdot \boldsymbol{\nabla} p = 0$, imposes the desired open boundary condition on the pressure.

The Chorin–Temam projection method described in Section 6.7.3.1 is used for marching in time. The idea is to present a solution strategy that could easily be programmed by the interested reader. With this in mind, the computational schemes have been selected so as to be characterized by a very simple algorithmic structure. The streamlines at different instants are shown in Figure 6.14.

The fluid is at rest at $t = 0$. The explicit Euler scheme is used for time integration of the first-step equations (although a backward Euler method can also be used to avoid instabilities and would induce similar results). To make explicit schemes



$t = 0.1\ (\Delta p = 10^{-2})$

$t = 1.2\ (\Delta p = 10^{-2})$

$t = 2.5\ (\Delta p = 5 \times 10^{-2})$

$t = 4.0\ (\Delta p = 5 \times 10^{-2})$

Fig. 6.15 Pressure contours for plane jet problem

efficient, they are combined with a lumped (diagonal) mass representation obtained by the classical row-sum technique. The backward Euler method is employed in the necessarily implicit second step (pressure/incompressibility step). As will be apparent from the numerical results, in addition to its simplicity, the proposed strategy is capable of producing accurate results. The time step is taken to be $\Delta t = 0.01$ as done by Laval and Quartapelle (1990).

The flow pattern, particularly the vortex creation close to the jet aperture, is represented at different instants by the streamlines in Figure 6.14. The corresponding pressure contours are shown in Figure 6.15. The results are in good agreement with those obtained by Laval and Quartapelle (1990) using a fractional-step Taylor–Galerkin method. Their method consists of three phases as follows. A pure convection phase is solved first using a second-order explicit Taylor–Galerkin method. This is followed by a viscous diffusion phase solved using the forward Euler method. Finally, the pressure/incompressibility phase is solved using the backward Euler scheme. The present results can also be compared with those obtained by Blasco, Codina and Huerta (1997) using a predictor-multicorrector algorithm.

6.8.4 Natural convection in a square cavity

To illustrate the use of the Chorin–Temam fractional-step method in another incompressible flow problem, we shall describe the splitting-up approximate solution of unsteady natural convection problems. Such problems involve a coupling between the Navier–Stokes equations describing the fluid motion and the thermal energy equation governing the space–time evolution of the temperature. The forces which induce natural convection are in fact spatially variable gravity forces generated by (buoyancy) density variations in the fluid due to the non-uniformity of the temperature.

6.8.4.1 The Navier–Stokes and temperature equations In natural convection problems the mathematical description of fluid motion is given by the following form of the Navier–Stokes equations:

$$\varrho\big(\boldsymbol{v}_t + (\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v}\big) = \boldsymbol{\nabla}\cdot\boldsymbol{\sigma} + \varrho\,\boldsymbol{b} \qquad \text{in } \Omega\times]0,T[, \tag{6.46}$$

where

$$\boldsymbol{b} = \big(1-\beta(\mathrm{T}-\mathrm{T}_0)\big)\,\boldsymbol{g} \tag{6.47}$$

is the gravity force per unit mass derived on the basis of Boussinesq approximation, vector $\boldsymbol{g}$ denotes the gravity field, β is the coefficient of thermal expansion of the fluid, T is the temperature field, T_0 is the reference temperature, and ϱ is the reference density of the fluid (i.e., the fluid density corresponding to the reference temperature, when $\mathrm{T} = \mathrm{T}_0$).

The fluid is assumed incompressible according to Boussinesq approximation. That is, its density is assumed constant, except in the gravity force term where it depends on temperature according to the indicated linear law, see (6.47).

Under the usual assumption of a Newtonian fluid, the components σ_{ij} of Cauchy stress are linearly related to the strain rate by Stokes' law. The incompressibility condition is expressed in the standard form $\boldsymbol{\nabla}\cdot\boldsymbol{v} = 0$, and the Navier–Stokes equations (6.46) are completed with the usual initial and boundary conditions.

The Boussinesq body forces, see (6.47), introduce a coupling between the Navier–Stokes equations and the thermal energy equation

$$\varrho c\big(\mathrm{T}_t + \boldsymbol{v}\cdot\boldsymbol{\nabla}\mathrm{T}\big) = \boldsymbol{\nabla}\cdot(k\boldsymbol{\nabla}\mathrm{T}) + Q \qquad \text{in } \Omega\times]0,T[. \tag{6.48}$$

Here, c denotes the specific heat of the fluid at constant volume, k is the thermal conductivity (assumed isotropic in this case) and Q represents volumetric heat generation.

Equation (6.48) combined with boundary conditions such as

$$\begin{aligned} \mathsf{T}(\boldsymbol{x},t) &= \mathsf{T}_D(\boldsymbol{x},t) && \text{on } \Gamma_D^{\mathsf{T}} \times]0,T[, \\ k\frac{\partial \mathsf{T}(\boldsymbol{x},t)}{\partial n} &= q_N(\boldsymbol{x},t) && \text{on } \Gamma_N^{\mathsf{T}} \times]0,T[, \end{aligned}$$

where T_D represents a prescribed temperature, while q_N denotes a prescribed normal heat flux. The initial condition $\mathsf{T}(\boldsymbol{x},0) = \mathsf{T}_0(\boldsymbol{x})$ for all $\boldsymbol{x} \in \Omega$, completes the initial boundary value problem for the temperature.

6.8.4.2 A fractional-step method for Navier–Stokes and temperature

The implicit character of pressure in an incompressible fluid precludes the use of purely explicit time integration algorithms in the Navier–Stokes equations. As seen in Section 6.7.3, it is nevertheless possible to take advantage of the algorithmic simplicity of explicit time-stepping schemes by using a fractional-step approach. That is, the convective and Boussinesq terms could be evaluated explicitly. Here a simple example is shown and to simplify the exposition, we assume that Dirichlet conditions for the velocity are prescribed on the whole boundary of the fluid domain.

The first step considers the convective, viscous and Boussinesq terms of equation (6.46), see also (6.37). Thus, the first-step equation reads

$$\boldsymbol{v}_t + (\boldsymbol{v} \cdot \boldsymbol{\nabla})\boldsymbol{v} - \boldsymbol{\nabla} \cdot (2\nu \boldsymbol{\nabla}^{\mathrm{S}} \boldsymbol{v}) = \boldsymbol{b}.$$

Note that this equation is normalized by the reference density, ϱ, of the fluid. The simplest option for this first step is an explicit time integration scheme. For instance, we can use the explicit Euler method or the second-order Adams–Bashforth method described in Section 5.3.1. The result is an intermediate velocity field $\boldsymbol{v}_{\text{int}}^{n+1}$, satisfying the Dirichlet boundary conditions, that is $\boldsymbol{v}_{\text{int}}^{n+1} = \boldsymbol{v}_D^{n+1}$.

The second step of the time integration of the Navier–Stokes equations is the pressure/incompressibiblity step. The end-of-step velocity field $\boldsymbol{v}^{n+1}$ is determined adding the intermediate velocity $\boldsymbol{v}_{\text{int}}^{n+1}$ and the effect of the pressure gradient. Recall that pressure is such that the end-of-step velocity is solenoidal. Time integration is necessarily implicit and the simplest algorithm is given by the backward Euler method, see equations (6.40).

A two-step procedure can also be used to update temperature from T^n to T^{n+1}. The convective term is integrated in the first step and the diffusion and source terms in the second one, see equation (6.48). Thus, the first step reduces to the hyperbolic equation

$$\mathsf{T}_t + \boldsymbol{v} \cdot \boldsymbol{\nabla} \mathsf{T} = 0.$$

The third-order explicit Taylor–Galerkin method, see Section 3.6.2, can be used for the time integration of this equation. An intermediate temperature field, T_{int}, is then obtained. Since this is a pure convection equation Dirichlet boundary condition $\mathsf{T}_{\text{int}} = \mathsf{T}_D^{n+1}$ are only prescribed on the inflow portion of Γ_D^{T} (i.e., where $\boldsymbol{n} \cdot \boldsymbol{v} < 0$).

The second step, which accounts for the diffusion and source terms, reads

$$\varrho c \mathsf{T}_t = \boldsymbol{\nabla} \cdot (k \boldsymbol{\nabla} \mathsf{T}) + Q.$$

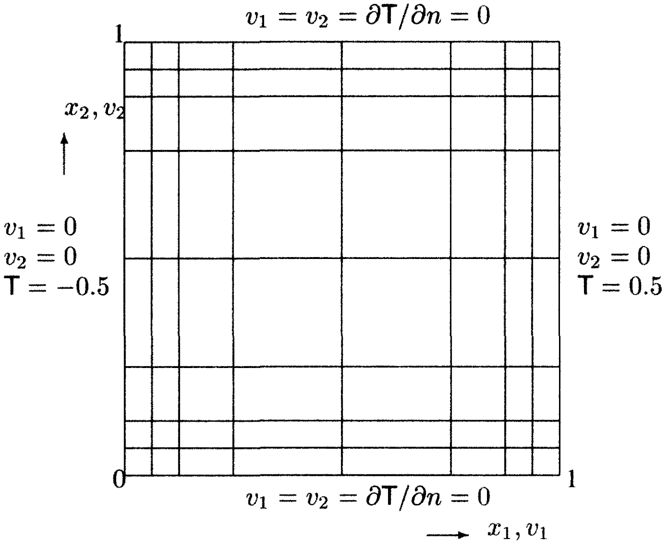

$v_1 = v_2 = \partial \mathsf{T}/\partial n = 0$
1
x_2, v_2
$v_1 = 0$
$v_2 = 0$
$\mathsf{T} = -0.5$
$v_1 = 0$
$v_2 = 0$
$\mathsf{T} = 0.5$
0
1
$v_1 = v_2 = \partial \mathsf{T}/\partial n = 0$
x_1, v_1

Fig. 6.16 Natural convection: problem statement and discretization.

The complete (Dirichlet and Neumann) boundary conditions are imposed. In this parabolic phase an implicit time-stepping algorithm is to be preferred since the stable time step for explicit methods decreases with the square of the mesh size. For instance, the second-order Crank–Nicolson method can be a reasonable choice.

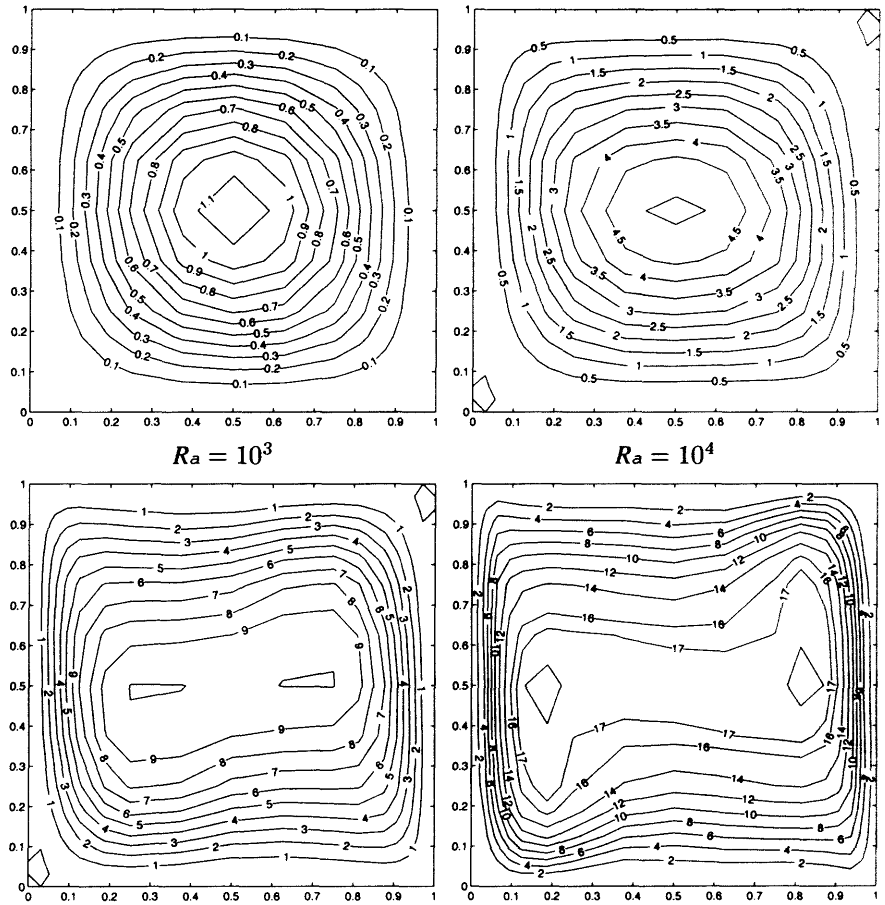
6.8.4.3 The numerical example The classical test problem of a differentially heated square cavity is chosen to illustrate the use of the fractional-step method for solving natural convection flows in enclosures.

The problem statement is depicted in Figure 6.16. At the initial time, a fluid at uniform temperature $\mathsf{T}_0 = 0$ is at rest in a square cavity with unit sides. At the start of the calculation, the temperature of the left wall of the cavity is suddenly lowered by 0.5, while the temperature of the right wall is increased by the same quantity. The horizontal walls are assumed to be thermally insulated (zero heat flux). No-slip and no penetration velocity boundary conditions are prescribed on all walls.

The motion induced by natural convection is governed by two dimensionless numbers, the Rayleigh and the Prandtl numbers defined by

$$\mathcal{R}a = \frac{g\beta L_{\text{ref}}^3\, \Delta\mathsf{T}}{\nu\, \kappa} \qquad \text{and} \qquad \mathcal{P}r = \frac{\nu}{\kappa},$$

where g is the gravitational acceleration, β the coefficient of thermal expansion, $\Delta\mathsf{T}$ the temperature difference between the hot and cold walls, ν denotes the kinematic viscosity and κ is the thermal diffusivity of the fluid. The reference length is $L_{\text{ref}} = 1$, the length of the cavity side.



$Ra = 10^3$

$Ra = 10^4$

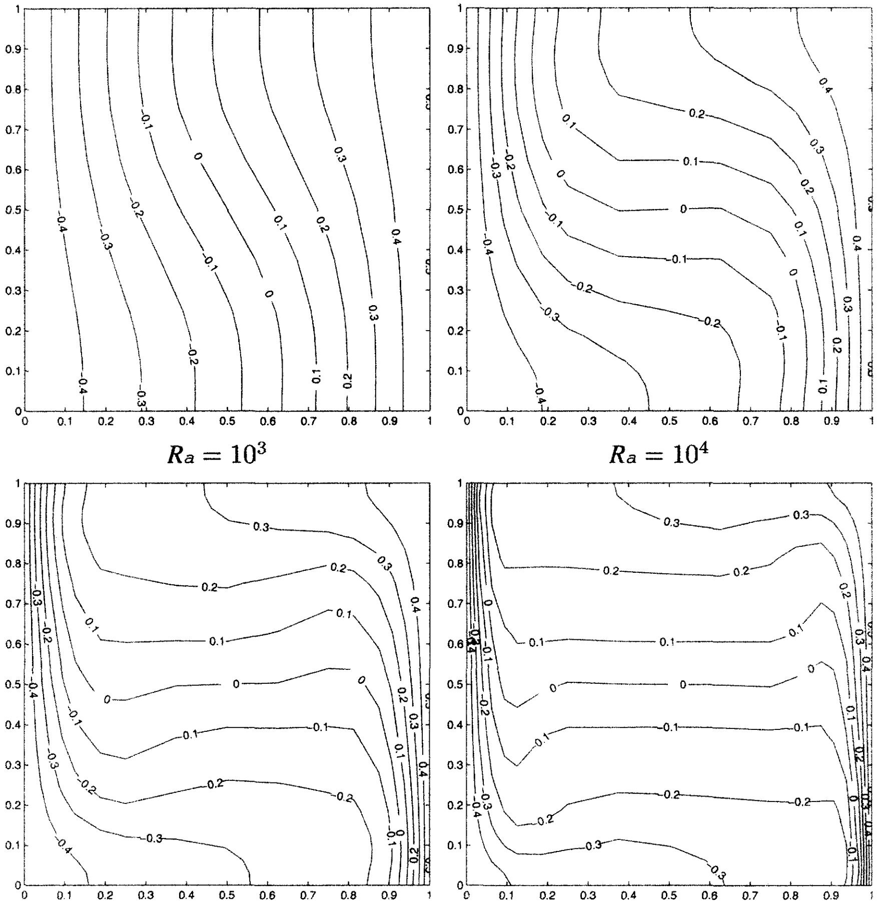
$Ra = 10^5$

$Ra = 10^6$

Fig. 6.17 Natural convection: streamlines in the thermal cavity for different Rayleigh numbers and Pr = 1.

The problem is symmetric with respect to the center of the cavity. This property can be exploited to reduce the cost of the simulation or to test the quality of the numerical results. We select the second option. The finite element computation is performed using the non-uniform mesh of 8 × 8 Q2Q1 elements for the mixed interpolation of velocity and pressure, see Figure 6.16. A biquadratic interpolation is used for the temperature. The time integration based upon the fractional-step approach described in Section 6.8.4.2 is used until a steady state is reached.

The results are for a Prandtl number of one and Rayleigh numbers from $10^{\,3}$ to 10^6. The development and the structure of the flow vary considerably with the Rayleigh number. The main characteristics of the flow are illustrated in Figures 6.17 and 6.18:



$Ra = 10^3$ $Ra = 10^4$

$Ra = 10^5$ $Ra = 10^6$

Fig. 6.18 Natural convection: isotherms in the thermal cavity for different Rayleigh numbers and Pr = 1.

1. A secondary flow develops at a Rayleigh number situated between 10^4 and 10^5.
2. A boundary layer appears progressively in the vicinity of the vertical walls of the cavity.
3. The secondary flow evolves at the center of the cavity in the form of "eyes of a cat".

The computed streamlines are plotted in Figure 6.17. For $Ra < 10^5$, the flow is mono-cellular. The fluid rises along the hot side of the cavity (right wall) and comes down along the cold wall. For $Ra = 10^5$, the development of a secondary flow starts in the central portion of the cavity. The circulation direction of the secondary flow is the same as that of the base flow. The value of the stream function at the center

Table 6.3 Stream function values at the center of the thermal cavity

Rayleigh number		10^3	10^4	10^5	10^6
Present simulation	(8 × 8)	1.19	5.15	9.74	17.99
Heinrich et al. (1978)	(4 × 4)	1.18	5.13	9.44	—
Marshall et al. (1978)	(8 × 8)	—	5.12	9.54	17.32

of the cavity gives a measure of the importance of the recirculation. The results are displayed in Table 6.3 in comparison with other simulations from the literature.

The isotherms in the cavity are displayed in Figure 6.18. At low values of the Rayleigh number they are practically straight lines, in accordance with a diffusion-dominated heat transfer process within the cavity. When the Rayleigh number increases, the convective effects become more and more important. The isotherms deform progressively and take an S shape. The distortion of the isotherms generates high temperature gradients near the vertical walls of the cavity.

References

Abdallah, S. (1995), 'Comments on the fractional step method: "An analysis of the fractional step method" [J. Comput. Phys. **108** (1993), no. 1, 51–58] by J. B. Perot', *J. Comput. Phys.* **117**(1), 179–180.

Adams, R. A. (1975), *Sobolev spaces*, Vol. 65 of *Pure and Applied Mathematics*, Academic Press, New York.

Ainsworth, M. (2001), 'Essential boundary conditions and multi-point constraints in finite element analysis', *Comput. Methods Appl. Mech. Eng.* **190**(48), 6323–6339.

Ainsworth, M. and Oden, J. T. (2000), *A posteriori error estimation in finite element analysis*, John Wiley & Sons, Chichester.

Allievi, A. and Bermejo, R. (2000), 'Finite element modified method of characteristics for the Navier-Stokes equations', *Int. J. Numer. Methods Fluids* **32**(4), 439–464.

Ames, W. F. (1992), *Numerical methods for partial differential equations*, third edn, Academic Press, New York.

Arnold, D. N., Brezzi, F. and Fortin, M. (1984), 'A stable finite element for the Stokes equations', *Calcolo* **21**(4), 337–344.

Babuška, I. (1970/71), 'Error-bounds for finite element method', *Numer. Math.* **16**, 322–333.

Baiocchi, C., Brezzi, F. and Franca, L. (1993), 'Virtual bubbles and the Galerkin-least-squares method', *Comput. Methods Appl. Mech. Eng.* **105**(1), 125–141.

Finite Element Methods for Flow Problems. Jean Donea and Antonio Huerta

ISBN: 0-471-49666-9

Baker, A. J. (1983), *Finite element computational fluid mechanics*, Hemisphere, Washington, D.C.

Batchelor, G. K. (1999), *An introduction to fluid dynamics*, Cambridge University Press, Cambridge.

Bathe, K.-J., Hendriana, D., Brezzi, F. and Sangalli, G. (2000), 'Inf-sup testing of upwind methods', *Int. J. Numer. Methods Eng.* **48**(5), 745–760.

Baumann, C. E. and Oden, J. T. (1999), 'A discontinuous hp finite element method for convection-diffusion problems', *Comput. Methods Appl. Mech. Eng.* **175**(3–4), 311–341.

Baumann, C. E. and Oden, J. T. (2000), 'An adaptive-order discontinuous Galerkin method for the solution of the Euler equations of gas dynamics', *Int. J. Numer. Methods Eng.* **47**(1–3), 61–73.

Bell, J. B., Colella, P. and Glaz, H. M. (1989), 'A second-order projection method for the incompressible Navier-Stokes equations', *J. Comput. Phys.* **85**(2), 257–283.

Belytschko, T. (1983), 'An overview of semidiscretization and time integration procedures', *in* T. Belytschko and T. J. R. Hughes, eds, *Computational methods for transient analysis*, Vol. 1 of *Mechanics and Mathematical Methods. A Series of Handbooks. Subseries: Computational Methods in Mechanics*, Elsevier Scientific, Amsterdam, Chap. 1, pp. 1–65.

Belytschko, T. and Eldib, I. (1979), 'Analysis of a finite element upwind scheme', *in* T. J. R. Hughes, ed., *Finite element methods for convection dominated flows*, AMD – Vol. 34, Presented at the Winter Annual Meeting of the ASME, Amer. Soc. Mech. Engrs. (ASME), New York, pp. 195–200.

Belytschko, T. and Kennedy, J. M. (1978), 'Computer methods for subassembly simulation', *Nucl. Eng. Des.* **49**, 17–38.

Belytschko, T., Kennedy, J. M. and Schoeberle, D. F. (1978), 'Quasi-Eulerian finite element formulation for fluid-structure interaction', Proceedings of the Joint ASME/CSME Pressure Vessels and Piping Conference, Amer. Soc. Mech. Engrs. (ASME), New York, p. 13. ASME paper 78-PVP-60.

Belytschko, T., Krongauz, Y., Organ, D., Fleming, M. and Krysl, P. (1996), 'Meshless methods: an overview and recent developments', *Comput. Methods Appl. Mech. Eng.* **139**(1–4), 3–48.

Belytschko, T., Liu, W. K. and Moran, B. (2000), *Nonlinear finite elements for continua and structures*, John Wiley & Sons, Chichester.

Belytschko, T., Lu, Y. Y. and Gu, L. (1994), 'Element-free Galerkin methods', *Int. J. Numer. Methods Eng.* **37**(2), 229–256.

Benqué, J. P., Ibler, B., Keramsi, A. and Labadie, G. (1980), 'A finite element method for the Navier–Stokes equations', Proceedings from the Third International Conference on Finite Elements in Flow Problems, Banff, Alberta, Canada, June 10–13, 1980, Vol. I, pp. 110–120.

Benqué, J. P., Labadie, G. and Ronat, J. (1982), 'A new finite element method for the Navier–Stokes equations coupled with a temperature equation', in T. Kawai, ed., Finite element flow analysis, Proceedings of the Fourth International Symposium, (Tokyo 1982), North-Holland, Amsterdam, pp. 295–302.

Bercovier, M. (1978), 'Perturbation of mixed variational problems. Application to mixed finite element methods', *RAIRO Anal. Numér.* **12**(3), 211–236.

Bercovier, M. and Engelman, M. (1979), 'A finite element for the numerical solution of viscous incompressible flows', *J. Comput. Phys.* **30**(2), 181–201.

Bermejo, R. (1995), 'A Galerkin-characteristic algorithm for transport-diffusion equations', *SIAM J. Numer. Anal.* **32**(2), 425–454.

Blasco, J., Codina, R. and Huerta, A. (1997), 'Analysis of fractional step finite element methods for the incompressible Navier-Stokes equations', Technical Report 38, International Center for Numerical Methods in Engineering (CIMNE), Barcelona.

Blasco, J., Codina, R. and Huerta, A. (1998), 'A fractional-step method for the incompressible Navier-Stokes equations related to a predictor-multicorrector algorithm', *Int. J. Numer. Methods Fluids* **28**(10), 1391–1419.

Bonet, J. and Wood, R. D. (1997), *Nonlinear continuum mechanics for finite element analysis*, Cambridge University Press, Cambridge.

Boris, J. P. and Book, D. L. (1997), 'Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works [J. Comput. Phys. **11** (1973), no. 1, 38–69]', *J. Comput. Phys.* **135**(2), 170–186. With an introduction by Steven T. Zalesak, Commemoration of the 30th anniversary of *J. Comput. Phys.*

Brezzi, F. (1974), 'On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers', *RAIRO Anal. Numér.* **8**(R-2), 129–151.

Brezzi, F. and Bathe, K.-J. (1990), 'A discourse on the stability conditions for mixed finite element formulations', *Comput. Methods Appl. Mech. Eng.* **82**(1-3), 27–57.

Brezzi, F. and Fortin, M. (1991), *Mixed and hybrid finite element methods*, Vol. 15 of *Springer Series in Computational Mathematics*, Springer-Verlag, New York.

Brezzi, F., Franca, L. P. and Russo, A. (1998), 'Further considerations on residual-free bubbles for advective-diffusive equations', *Comput. Methods Appl. Mech. Eng.* **166**(1–2), 25–33.

Brooks, A. N. and Hughes, T. J. R. (1982), 'Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations', *Comput. Methods Appl. Mech. Eng.* **32**(1–3), 199–259.

Bung, H., Casadei, F., Halleux, J.-P. and Lepareux, M. (1989), 'PLEXIS-3C: a computer code for fast dynamic problems in structures and fluids', Proceedins of the 10th SMIRT Conference, Anaheim, USA.

Burggraf, O. R. (1966), 'Analytical and numerical studies of the structure of steady separated flows', *J. Fluid Mechanics* **24**(1), 113–152.

Carette, J.-C. (1997), 'Adaptive unstructured mesh algorithms and SUPG finite element method for compressible high Reynolds number flows', PhD thesis, Université Libre de Bruxelles, Belgium.

Carey, G. F. and Jiang, B. N. (1988), 'Least-squares finite elements for first-order hyperbolic systems', *Int. J. Numer. Methods Eng.* **26**(1), 81–93.

Carey, G. F. and Oden, J. T. (1983), *Finite elements. A second course*, Vol. II of *The Texas Finite Element Series*, Prentice Hall, Englewood Cliffs, NJ.

Carey, G. F. and Oden, J. T. (1986), *Finite elements. Fluid mechanics*, Vol. VI of *The Texas Finite Element Series*, Prentice Hall, Englewood Cliffs, NJ.

Carey, G. F., Shen, Y. and McLay, R. T. (1998), 'Parallel conjugate gradient performance for least-squares finite elements and transport problems', *Int. J. Numer. Methods Fluids* **28**(10), 1421–1440.

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

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

Casadei, F. and Sala, A. (1999), 'Finite element and finite volume simulation of industrial fast transient fluid-structure interactions', European Conference on Computational Mechanics, ECCM'99, Munich.

Chorin, A. J. (1968), 'Numerical solution of the Navier-Stokes equations', *Math. Comput.* **22**, 745–762.

Chorin, A. J. (1969), 'On the convergence of discrete approximations to the Navier-Stokes equations', *Math. Comput.* **23**, 341–353.

Christie, I. (1985), 'Upwind compact finite difference schemes', *J. Comput. Phys.* **59**(2), 353–368.

Christie, I., Griffiths, D. F., Mitchell, A. R. and Zienkiewicz, O. C. (1976), 'Finite element methods for second order differential equations with significant first derivatives', *Int. J. Numer. Methods Eng.* **10**(6), 1389–1396.

Christon, M. A. (1991), 'The influence of the mass matrix on the dispersive nature of the semi-discrete, second-order wave equation', *Comput. Methods Appl. Mech. Eng.* **173**(1–2), 147–166.

Ciarlet, P. G. (1978), *The finite element method for elliptic problems*, Vol. 4 of *Studies in Mathematics and its Applications*, North-Holland, Amsterdam.

Cockburn, B. (1998), 'An introduction to the discontinuous Galerkin method for convection-dominated problems', *in* A. Quarteroni, ed., *Advanced numerical approximation of nonlinear hyperbolic equations*, Vol. 1697 of *Lecture Notes in Mathematics*, Springer-Verlag, Berlin, pp. 151–268. Papers from the C.I.M.E. Summer School held in Cetraro, June 23–28, 1997.

Cockburn, B. and Shu, C.-W. (2001), 'Runge-Kutta discontinuous Galerkin methods for convection-dominated problems', *J. Sci. Comput.* **16**(3), 173–261.

Codina, R. (1993a), 'A discontinuity-capturing crosswind-dissipation for the finite element solution of the convection-diffusion equation', *Comput. Methods Appl. Mech. Eng.* **110**(3–4), 325–342.

Codina, R. (1993b), 'A finite element formulation for the numerical solution of the convection-diffusion equation', Technical Report 14, International Center for Numerical Methods in Engineering (CIMNE), Barcelona.

Codina, R. (1998), 'Comparison of some finite element methods for solving the diffusion-convection-reaction equation', *Comput. Methods Appl. Mech. Eng.* **156**(1–4), 185–210.

Codina, R. (2000), 'On stabilized finite element methods for linear systems of convection-diffusion-reaction equations', *Comput. Methods Appl. Mech. Eng.* **188**(1–3), 61–82.

Codina, R., Vázquez, M. and Zienkiewicz, O. C. (1998), 'A general algorithm for compressible and incompressible flows. III. The semi-implicit form', *Int. J. Numer. Methods Fluids* **27**(1–4, Special Issue), 13–32.

Courant, R., Friedrichs, K. and Lewy, H. (1967), 'On the partial difference equations of mathematical physics', *IBM J. Res. Develop.* **11**, 215–234. English translation of an article originally published in German in *Math. Ann.* **100** (1928), 32–74.

Crouzeix, M. and Raviart, P.-A. (1973), 'Conforming and nonconforming finite element methods for solving the stationary Stokes equations. I', *RAIRO Anal. Numér.* **7**(R-3), 33–75.

Dahmen, W., Kurdila, A. J. and Oswald, P. (eds) (1997), *Multiscale wavelet methods for partial differential equations*, Academic Press, San Diego, CA.

Davis, G. D. V. and Mallinson, G. (1976), 'An evaluation of upwind and central difference approximations by a study of recirculating flow', *Comput. Fluids* **4**, 29–43.

Deconinck, H., Hirsch, C. and Peuteman, J. (1986), 'Characteristic decomposition methods for the multidimensional Euler equations', *Tenth international conference on numerical methods in fluid dynamics (Beijing, 1986)*, Vol. 264 of *Lecture Notes in Phys.*, Springer-Verlag, Berlin, pp. 216–221.

Donea, J. (1983), 'Arbitrary Lagrangian-Eulerian finite element methods', *in* T. Belytschko and T. J. R. Hughes, eds, *Computational methods for transient analysis*, Vol. 1 of *Mechanics and Mathematical Methods. A Series of Handbooks. Subseries: Computational Methods in Mechanics*, Elsevier Scientific, Amsterdam, Chap. 10, pp. 473–516.

Donea, J. (1984), 'A Taylor-Galerkin method for convective transport problems', *Int. J. Numer. Methods Eng.* **20**(24), 101–120.

Donea, J., Belytschko, T. and Smolinski, P. (1985), 'A generalized Galerkin method for steady convection–diffusion problems with application to quadratic shape functions', *Comput. Methods Appl. Mech. Eng.* **48**(1), 25–43.

Donea, J., Fasoli-Stella, P. and Giuliani, S. (1977), 'Lagrangian and Eulerian finite element techniques for transient fluid-structure interaction problems', Transactions of the 4th SMIRT Conference, Vol. B. paper B1/2, San Francisco, 15-19 August.

Donea, J. and Giuliani, S. (1981), 'A simple method to generate high-order accurate convection operators for explicit schemes based on linear finite elements', *Int. J. Numer. Methods Fluids* **1**(1), 63–79.

Donea, J., Giuliani, S. and Halleux, J.-P. (1982), 'An Arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions', *Comput. Methods Appl. Mech. Eng.* **33**, 689–723.

Donea, J., Giuliani, S., Laval, H. and Quartapelle, L. (1982), 'Finite element solution of the unsteady Navier-Stokes equations by a fractional step method', *Comput. Methods Appl. Mech. Eng.* **30**, 53–73.

Donea, J. and Quartapelle, L. (1992), 'An introduction to finite element methods for transient advection problems', *Comput. Methods Appl. Mech. Eng.* **95**(2), 169–203.

Donea, J., Quartapelle, L. and Selmin, V. (1987), 'An analysis of time discretization in the finite element solution of hyperbolic problems', *J. Comput. Phys.* **70**(2), 463–499.

Donea, J., Roig, B. and Huerta, A. (1998), 'High-order accurate time-stepping schemes for convection-diffusion problems', Technical Report 42, International Center for Numerical Methods in Engineering (CIMNE), Barcelona.

Donea, J., Roig, B. and Huerta, A. (2000), 'High-order accurate time-stepping schemes for convection-diffusion problems', *Comput. Methods Appl. Mech. Eng.* **182**(3–4), 249–275.

Donea, J., Selmin, V. and Quartapelle, L. (1988), 'Recent developments of the Taylor-Galerkin method for the numerical solution of hyperbolic problems', *Numerical methods for fluid dynamics, III (Proceedings of the 3rd Conference, Oxford, 1988)*, Vol. 17 of *Inst. Math. Appl. Conf. Ser. New Ser.*, Oxford Univ. Press, New York, pp. 171–185.

Douglas, Jr, J. and Russell, T. F. (1982), 'Numerical methods for convection-dominated diffusion problems based on combining the method of characteristics with finite element or finite difference procedures', *SIAM J. Numer. Anal.* **19**(5), 871–885.

Duarte, C. A. and Oden, J. T. (1996), 'H-p clouds—an h-p meshless method', *Numer. Meth. Part. Differ. Equ.* **12**(6), 673–705.

Felippa, C. A. (2001), 'A historical outline of matrix structural analysis: a play in three acts', *Comput. Struct.* **79**(14), 1313–1324.

Franca, L. P. and Frey, S. L. (1992), 'Stabilized finite element methods: II. The incompressible Navier-Stokes equations', *Comput. Methods Appl. Mech. Eng.* **99**(2–3), 209–233.

Franca, L. P., Frey, S. L. and Hughes, T. J. R. (1992), 'Stabilized finite element methods. I. Application to the advective-diffusive model', *Comput. Methods Appl. Mech. Eng.* **95**(2), 253–276.

Franca, L. P. and Hughes, T. J. R. (1993), 'Convergence analyses of Galerkin least-squares methods for symmetric advective-diffusive forms of the Stokes and incompressible Navier-Stokes equations', *Comput. Methods Appl. Mech. Eng.* **105**(2), 285–298.

Franca, L. P., Nesliturk, A. and Stynes, M. (1998), 'On the stability of residual-free bubbles for convection-diffusion problems and their approximation by a two-level finite element method', *Comput. Methods Appl. Mech. Eng.* **166**(1–2), 35–49.

Galeão, A. C. and do Carmo, E. G. D. (1988), 'A consistent approximate upwind Petrov-Galerkin method for convection dominated problems', *Comput. Methods Appl. Mech. Eng.* **68**(1), 83–95.

Gear, C. W. (1971), *Numerical initial value problems in ordinary differential equations*, Prentice Hall, Englewood Cliffs, NJ.

Giles, M. B. (1997), 'Stability analysis of a Galerkin/Runge-Kutta Navier-Stokes discretisation on unstructured tetrahedral grids', *J. Comput. Phys.* **132**(2), 201–214.

Girault, V. and Raviart, P.-A. (1986), *Finite element methods for Navier-Stokes equations. Theory and algorithms*, Springer-Verlag, Berlin.

Giuliani, S. (1982), 'An algorithm for continuous rezoning of the hydrodynamic grid in Arbitrary Lagrangian-Eulerian computer codes', *Nucl. Eng. Des.* **72**, 205–212.

Glowinski, R. (1984), *Numerical methods for nonlinear variational problems*, Springer-Verlag, New York.

Glowinski, R. and Le Tallec, P. (1989), *Augmented Lagrangian and operator-splitting methods in nonlinear mechanics*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA.

Godlewski, E. and Raviart, P.-A. (1996), *Numerical Approximation of Hyperbolic Systems of Conservation Laws*, Vol. 118 of *Springer Series in Applied Mathematical Sciences*, Springer-Verlag, Berlin.

Godunov, S. K. (1959), 'A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics', *Mat. Sb. (N.S.)* **47** **(89)**, 271–306.

Gresho, P. M. (1990), 'On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. I. Theory', *Int. J. Numer. Methods Fluids* **11**(5), 587–620.

Gresho, P. M. and Lee, R. L. (1979), 'Don't suppress the wiggles—they're telling you something!', *in* T. J. R. Hughes, ed., *Finite element methods for convection dominated flows*, AMD – Vol. 34, Presented at the Winter Annual Meeting of the ASME, Amer. Soc. Mech. Engrs. (ASME), New York, pp. 37–61.

Gresho, P. M. and Sani, R. L. (1987), 'On pressure boundary conditions for the incompressible Navier-Stokes equations', *Int. J. Numer. Methods Fluids* **7**(10), 1111–1145.

Gresho, P. M. and Sani, R. L. (2000), *Incompressible flow and the finite element method. Vol. 1: Advection diffusion. Vol. 2: Isothermal laminar flow*, John Wiley & Sons, Chichester.

Griffiths, D. F. and Mitchell, A. R. (1979), 'On generating upwind finite element methods', *in* T. J. R. Hughes, ed., *Finite element methods for convection dominated flows*, AMD – Vol. 34, Presented at the Winter Annual Meeting of the ASME, Amer. Soc. Mech. Engrs. (ASME), New York, pp. 91–104.

Guermond, J.-L. (1999a), 'Stabilisation par viscosit é de sous-maille pour l'approximation de Galerkin des opérateurs linéaires monotones', *Comptes Rendus Acad. Sci. Ser. I-Math.* **328**(7), 617–622.

Guermond, J.-L. (1999b), 'Stabilization of Galerkin approximations of transport equations by subgrid modeling', *ESAIM-Math. Model. Numer. Anal.* **33**(6), 1293–1316.

Guermond, J.-L. and Quartapelle, L. (1997), 'Calculation of incompressible viscous flow by an unconditionally stable projection FEM', *J. Comput. Phys.* **132**(1), 12–33.

Guermond, J.-L. and Quartapelle, L. (1998a), 'On stability and convergence of projection methods based on pressure Poisson equation', *Int. J. Numer. Methods Fluids* **26**(9), 1039–1053.

Guermond, J.-L. and Quartapelle, L. (1998b), 'On the approximation of the unsteady Navier-Stokes equations by finite element projection methods', *Numer. Math.* **80**(2), 207–238.

Gunzburger, M. D. (1989), *Finite element methods for viscous incompressible flows. A guide to theory, practice, and algorithms*, Academic Press, Boston, MA.

Hairer, E., Nørsett, S. P. and Wanner, G. (1993), *Solving ordinary differential equations. Vol. I. Nonstiff problems*, second edn, Springer-Verlag, Berlin.

Hairer, E. and Wanner, G. (1996), *Solving ordinary differential equations. Vol. II. Stiff and differential–algebraic problems*, second edn, Springer-Verlag, Berlin.

Hannani, S. K., Stanislas, M. and Dupont, P. (1995), 'Incompressible Navier-Stokes computations with SUPG and GLS formulations', *Comput. Methods Appl. Mech. Eng.* **124**(1–2), 153–170.

Hansbo, P. (1993), 'Explicit streamline diffusion finite element methods for the compressible Euler equations in conservation variables', *J. Comput. Phys.* **109**(2), 274–288.

Hansbo, P. and Johnson, C. (1991), 'Adaptive streamline diffusion methods for compressible flows using conservation variables', *Comput. Methods Appl. Mech. Eng.* **87**(2–3), 267–280.

Hansbo, P. and Szepessy, A. (1990), 'A velocity-pressure streamline diffusion finite element method for the incompressible Navier-Stokes equations', *Comput. Methods Appl. Mech. Eng.* **84**(2), 175–192.

Harari, I., Frey, S. and Franca, L. P. (2002), 'A note on a recent study of stabilized finite element computations for heat conduction', *Comput. Mech.* **28**(1), 63–65.

Harari, I. and Hughes, T. J. R. (1994), 'Stabilized finite element methods for steady advection-diffusion with production', *Comput. Methods Appl. Mech. Eng.* **115**(1–2), 165–191.

Harten, A. (1983), 'High resolution schemes for hyperbolic conservation laws', *J. Comput. Phys.* **49**(3), 357–393.

Harten, A. (1997), 'High resolution schemes for hyperbolic conservation laws', *J. Comput. Phys.* **135**(2), 259–278. With an introduction by Peter Lax, Commemoration of the 30th anniversary of *J. Comput. Phys.*

Harten, A., Hyman, J. M. and Lax, P. D. (1976), 'On finite-difference approximations and entropy conditions for shocks', *Commun. Pure Appl. Math.* **29**(3), 297–322. With an appendix by B. Keyfitz.

Harten, A. and Tal-Ezer, H. (1981), 'On a fourth order accurate implicit finite difference scheme for hyperbolic conservation laws. I. Nonstiff strongly dynamic problems', *Math. Comput.* **36**(154), 353–373.

Hauke, G. and Hughes, T. J. R. (1998), 'A comparative study of different sets of variables for solving compressible and incompressible flows', *Comput. Methods Appl. Mech. Eng.* **153**(1–2), 1–44.

Heinrich, J. C., Huyakorn, P. S., Zienkiewicz, O. C. and Mitchell, A. R. (1977), 'An "upwind" finite element scheme for two-dimensional convective transport equation', *Int. J. Numer. Methods Eng.* **11**(1), 131–143.

Heinrich, J. C., Marshall, R. S. and Zienkiewicz, O. C. (1978), 'Penalty function solution of coupled convective and conductive heat transfer', *in* C. Taylor, K. Morgan and C. Brebbia, eds, *Numerical Methods in Laminar and Turbulent Flows*, Pentech Press, pp. 435–447.

Heinrich, J. C. and Zienkiewicz, O. C. (1979), 'The finite element method and "upwinding" techniques in the numerical solution of convection dominated flow problems', *in* T. J. R. Hughes, ed., *Finite element methods for convection dominated flows*, AMD – Vol. 34, Presented at the Winter Annual Meeting of the ASME, Amer. Soc. Mech. Engrs. (ASME), New York, pp. 105–136.

Heywood, J. G. and Rannacher, R. (1982), 'Finite element approximation of the nonstationary Navier-Stokes problem. I. Regularity of solutions and second-order error estimates for spatial discretization', *SIAM J. Numer. Anal.* **19**(2), 275–311.

Heywood, J. G. and Rannacher, R. (1986), 'Finite element approximation of the nonstationary Navier-Stokes problem. II. Stability of solutions and error estimates uniform in time', *SIAM J. Numer. Anal.* **23**(4), 750–777.

Heywood, J. G. and Rannacher, R. (1988), 'Finite element approximation of the nonstationary Navier-Stokes problem. III. Smoothing property and higher order error estimates for spatial discretization', *SIAM J. Numer. Anal.* **25**(3), 489–512.

Heywood, J. G. and Rannacher, R. (1990), 'Finite-element approximation of the nonstationary Navier-Stokes problem. IV. Error analysis for second-order time discretization', *SIAM J. Numer. Anal.* **27**(2), 353–384.

Hirsch, C. (1990), *Numerical computation of internal and external flows. Vol. 2: Computational methods for inviscid and viscous flows*, Wiley Series in Numerical Methods in Engineering, John Wiley & Sons, Chichester.

Hirt, C. W., Amsden, A. A. and Cook, J. L. (1974), 'An Arbitrary Lagrangian-Eulerian computing method for all flow speeds', *J. Comput. Phys.* **14**(3), 227–253.

Hirt, C. W., Amsden, A. A. and Cook, J. L. (1997), 'An Arbitrary Lagrangian-Eulerian computing method for all flow speeds', *J. Comput. Phys.* **135**(2), 203–216. Reprinted from ibid. **14**, 227–253 (1974).

Huerta, A. and Donea, J. (2002), 'Time-accurate solution of stabilized convection-diffusion-reaction equations: I — Time and space discretization', *Commun. Numer. Methods Eng.* **18**(8), 565–573.

Huerta, A. and Fernández-Méndez, S. (2000), 'Enrichment and coupling of the finite element and meshless methods', *Int. J. Numer. Methods Eng.* **48**(11), 1615–1636.

Huerta, A. and Fernández-Méndez, S. (2003), 'Time accurate consistently stabilized mesh-free methods for convection dominated problems', *Int. J. Numer. Methods Eng.* **59**(9), 1225–1242.

Huerta, A. and Liu, W. K. (1988), 'Viscous flow with large free surface motion', *Comput. Methods Appl. Mech. Eng.* **69**(3), 277–324.

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

Huerta, A., Roig, B. and Donea, J. (2002), 'Time-accurate solution of stabilized convection-diffusion-reaction equations: II — Accuracy analysis and examples', *Commun. Numer. Methods Eng.* **18**(8), 575–584.

Hughes, T. J. R. (1978), 'A simple scheme for developing "upwind" finite elements', *Int. J. Numer. Methods Eng.* **12**(9), 1359–1365.

Hughes, T. J. R. (1995), 'Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods', *Comput. Methods Appl. Mech. Eng.* **127**(1–4), 387–401.

Hughes, T. J. R. (2000), *The finite element method: linear static and dynamic finite element analysis*, Dover Publications, New York. Corrected reprint of the 1987 original [Prentice Hall, Englewood Cliffs, NJ].

Hughes, T. J. R. and Brooks, A. (1979), 'A multidimensional upwind scheme with no crosswind diffusion', *in* T. J. R. Hughes, ed., *Finite element methods for convection dominated flows*, AMD – Vol. 34, Presented at the Winter Annual Meeting of the ASME, Amer. Soc. Mech. Engrs. (ASME), New York, pp. 19–35.

Hughes, T. J. R. and Brooks, A. N. (1982), 'A theoretical framework for Petrov-Galerkin methods with discontinuous weighting functions: application to the streamline-upwind procedure', *in* R. H. Gallagher, D. H. Norrie, J. T. Oden and O. C. Zienkiewicz, eds, *Finite Elements in Fluids*, Vol. 4, Selected papers from the Third International Conference on Finite Elements in Flow Problems, Banff, Alberta, Canada, June 10–13, 1980, John Wiley & Sons, New York, pp. 47–65.

Hughes, T. J. R., Feijóo, G. R., Mazzei, L. and Quincy, J.-B. (1998), 'The variational multiscale method - a paradigm for computational mechanics', *Comput. Methods Appl. Mech. Eng.* **166**(1–2), 3–24.

Hughes, T. J. R. and Franca, L. P. (1987), 'A new finite element formulation for computational fluid dynamics. VII. The Stokes problem with various well-posed boundary conditions: symmetric formulations that converge for all velocity/pressure spaces', *Comput. Methods Appl. Mech. Eng.* **65**(1), 85–96.

Hughes, T. J. R., Franca, L. P. and Hulbert, G. M. (1989), 'A new finite element formulation for computational fluid dynamics. VIII. The Galerkin/least-squares method for advective-diffusive equations', *Comput. Methods Appl. Mech. Eng.* **73**(2), 173–189.

Hughes, T. J. R., Franca, L. P. and Mallet, M. (1987), 'A new finite element formulation for computational fluid dynamics. VI. Convergence analysis of the generalized SUPG formulation for linear time-dependent multidimensional advective-diffusive systems', *Comput. Methods Appl. Mech. Eng.* **63**(1), 97–112.

Hughes, T. J. R., Liu, W. K. and Brooks, A. (1979), 'Finite element analysis of incompressible viscous flows by the penalty function formulation', *J. Comput. Phys.* **30**(1), 1–60.

Hughes, T. J. R., Liu, W. K. and Zimmermann, T. K. (1978), 'Lagrangian-Eulerian finite element formulation for incompressible viscous flows', U.S.-Japan Seminar on Interdisciplinary Finite Element Analysis, Cornell University, Ithaca, NY.

Hughes, T. J. R. and Mallet, M. (1986a), 'A new finite element formulation for computational fluid dynamics. III. The generalized streamline operator for multidimensional advective-diffusive systems', *Comput. Methods Appl. Mech. Eng.* **58**(3), 305–328.

Hughes, T. J. R. and Mallet, M. (1986b), 'A new finite element formulation for computational fluid dynamics. IV. A discontinuity-capturing operator for multidimensional advective-diffusive systems', *Comput. Methods Appl. Mech. Eng.* **58**(3), 329–336.

Hughes, T. J. R., Mazzei, L. and Jansen, K. E. (2000), 'Large eddy simulation and variational multiscale method', *Computing and Visualization in Science* **3**(1/2), 47–59.

Hughes, T. J. R., Mazzei, L., Oberai, A. O. and Wray, A. A. (2001), 'The multiscale formulation of large eddy simulation: decay of homogeneous isotropic turbulence', *Phys. Fluids* **13**(2), 505–512.

Hughes, T. J. R. and Tezduyar, T. E. (1984), 'Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations', *Comput. Methods Appl. Mech. Eng.* **45**(1–3), 217–284.

Hughes, T. J. R., Tezduyar, T. E. and Brooks, A. N. (1982), 'A Petrov Galerkin finite element formulation for systems of conservation law with special reference to the compressible Euler equations', *in* K. W. Morton and M. J. Baines, eds, *Numerical methods for fluid dynamics, (Reading 1982)*, Academic Press, London, pp. 97–125.

Iannelli, G. S. and Baker, A. J. (1991), 'A globally well-posed finite element algorithm for aerodynamic applications', *Int. J. Numer. Methods Fluids* **12**(5), 407–441.

Idelsohn, S., Nigro, N., Storti, M. and Buscaglia, G. (1996), 'A Petrov-Galerkin formulation for advection-reaction-diffusion problems', *Comput. Methods Appl. Mech. Eng.* **136**(1–2), 27–46.

Ilinca, F., Hétu, J.-F. and Peletier, D. (2000), 'On stabilized finite element formulations for incompressible advective-diffusive transport and fluid flow problems', *Comput. Methods Appl. Mech. Eng.* **188**(1–3), 235–255.

Isaacson, E. and Keller, H. B. (1994), *Analysis of numerical methods*, Dover Publications, New York. Corrected reprint of the 1966 original [John Wiley & Sons, New York].

Jameson, A. (1985), 'Numerical solution of the Euler equations for compressible inviscid fluids', *in* F. Angrand, A. Dervieux, J. A. Desideri and R. Glowinski, eds, *Numerical methods for the Euler equations of fluid dynamics*, SIAM, Philadelphia, pp. 199–245. Proceedings of the INRIA Workshop, Rocquencourt, 1983.

Jameson, A., Schmidt, W. and Turkel, E. (1981), 'Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time stepping schemes', AIAA Computational Fluid Dynamics Conference, San Diego. Paper 81-1259.

Jamet, P. (1978), 'Galerkin-type approximations which are discontinuous in time for parabolic equations in a variable domain', *SIAM J. Numer. Anal.* **15**(5), 912–928.

Jansen, K. E., Collis, S. S., Whiting, C. and Shakib, F. (1999), 'A better consistency for low-order stabilized finite element methods', *Comput. Methods Appl. Mech. Eng.* **174**(1–2), 153–170.

Jiang, B.-n. (1998), *The least-squares finite element method. Theory and applications in computational fluid dynamics and electromagnetics*, Springer-Verlag, Berlin.

Jiang, C. B. and Kawahara, M. (1993), 'The analysis of unsteady incompressible flows by a three-step finite element method', *Int. J. Numer. Methods Fluids* **16**(9), 793–811.

John, F. (1991), *Partial differential equations*, fourth edn, Springer-Verlag, New York.

Johnson, C. (1987), *Numerical solution of partial differential equations by the finite element method*, Cambridge University Press, Cambridge.

Johnson, C., Nävert, U. and Pitkäranta, J. (1984), 'Finite element methods for linear hyperbolic equations', *Comput. Methods Appl. Mech. Eng.* **45**(1–3), 285–312.

Johnson, C. and Saranen, J. (1986), 'Streamline diffusion methods for the incompressible Euler and Navier-Stokes equations', *Math. Comput.* **47**(175), 1–18.

Johnson, C., Szepessy, A. and Hansbo, P. (1990), 'On the convergence of shock-capturing streamline diffusion finite element methods for hyperbolic conservation laws', *Math. Comput.* **54**(189), 107–129.

Kehr-Candille, V. and Ohayon, R. (1992), 'Elastoacoustic damped vibrations. Finite element and modal reduction methods', *in* P. Ladevèze and O. Zienkiewicz, eds, *New advances in computational structural mechanics (Giens, 1991)*, Vol. 32 of *Studies in Applied Mechanics*, Elsevier, Amsterdam, pp. 321–334.

Kelly, D. W., Nakazawa, S., Zienkiewicz, O. C. and Heinrich, J. C. (1980), 'A note of upwinding and anisotropic balancing dissipation in finite element approximation to convective diffusion problems', *Int. J. Numer. Methods Eng.* **15**(9), 1705–1711.

Ladevèze, P. and Pelle, J.-P. (2001), *La maîtrise du calcul en mécanique linéaire et non linéaire*, Hermes Science, Paris.

Ladyzhenskaya, O. A. (1969), *The mathematical theory of viscous incompressible flow*, Gordon and Breach Science, New York.

Lamb, H. (1993), *Hydrodynamics*, sixth edn, Cambridge University Press, Cambridge. Reprint of the 1932 edition.

Lambert, J. D. (1991), *Numerical methods for ordinary differential equations. The initial value problem*, John Wiley & Sons, Chichester.

Landau, L. D. and Lifshitz, E. M. (1959), *Fluid mechanics*, Pergamon Press, London.

Lapidus, A. (1967), 'A detached shock calculation by second order finite differences.', *J. Comput. Phys.* **2**, 154–177.

Lasaint, P. and Raviart, P.-A. (1974), 'On a finite element method for solving the neutron transport equation', *Mathematical aspects of finite elements in partial differential equations*, Math. Res. Center, University of Wisconsin-Madison, Academic Press, New York, pp. 89–123. Publication No. 33. Proceedings of the Symposium, Madison, 1974.

Laval, H. (1988), 'Taylor–Galerkin solution of the time-dependent Navier–Stokes equations', *in* H. Niki and M. Kawahara, eds, *Computational Methods in Flow Analysis*, Okayama University of Science, pp. 414–421.

Laval, H. and Quartapelle, L. (1990), 'A fractional-step Taylor–Galerkin method for unsteady incompressible flows', *Int. J. Numer. Methods Fluids* **11**(5), 501–513.

Leonard, B. P. (1979), 'A survey of finite differences of opinion on numerical muddling of the incomprehensible defective confusion equation', *in* T. J. R. Hughes, ed., *Finite element methods for convection dominated flows*, AMD – Vol. 34, Presented at the Winter Annual Meeting of the ASME, Amer. Soc. Mech. Engrs. (ASME), New York, pp. 1–17.

LeVeque, R. J. (1992), *Numerical methods for conservation laws*, second edn, Birkhäuser-Verlag, Basel.

Li, C. W. (1990), 'Least-squares characteristics and finite elements for advection-dispersion simulation', *Int. J. Numer. Methods Eng.* **29**(6), 1343–1358.

Liu, F. and Jameson, A. (1993), 'Multigrid Euler calculations for three-dimensional cascades', *AIAA J.* **31**(10), 1785–1791.

Liu, W. K. and Chang, H. (1986), 'On a numerical method for liquid filled systems', *Comput. Struct.* **23**(5), 671–677.

Liu, W. K. and Chang, H. G. (1985), 'A method of computation for fluid structure interaction', *Comput. Struct.* **20**(1–3), 311–320.

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

Liu, W. K., Jun, S., Li, S., Adee, J. and Belytschko, T. (1995), 'Reproducing kernel particle methods for structural dynamics', *Int. J. Numer. Methods Eng.* **38**(10), 1655–1679.

Löhner, R., Morgan, K. and Peraire, J. (1985), 'A simple extension to multidimensional problems of the artificial viscosity due to Lapidus', *Commun. Numer. Methods Eng.* **1**(4), 141–147.

Malkus, D. S. and Hughes, T. J. R. (1978), 'Mixed finite element methods - reduced and selective integration techniques: a unification of concepts', *Comput. Methods Appl. Mech. Eng.* **15**(1), 63–81.

Marchuk, G. I. (1982), *Methods of numerical mathematics*, second edn, Springer-Verlag, New York. Translated from the Russian by Arthur A. Brown.

Marchuk, G. I. (1990), 'Splitting and alternating direction methods', *in* P. G. Ciarlet and J.-L. Lions, eds, *Handbook of numerical analysis, Vol. I*, North-Holland, Amsterdam, pp. 197–462.

Marshall, R. S., Heinrich, J. C. and Zienkiewicz, O. C. (1978), 'Natural convection in a square enclosure by a finite element penalty function method using primitive fluid variables', *J. Num. Methods in Heat Transfer* **1**, 315–330.

Masud, A. and Hughes, T. J. R. (1997), 'A space-time Galerkin/least-squares finite element formulation of the Navier-Stokes equations for moving domain problems', *Comput. Methods Appl. Mech. Eng.* **146**(1–2), 91–126.

Melenk, J. M. and Babuška, I. (1996), 'The partition of unity finite element method: basic theory and applications', *Comput. Methods Appl. Mech. Eng.* **139**(1–4), 289–314.

Minev, P. D. (2001), 'A stabilized incremental projection scheme for the incompressible Navier-Stokes equations', *Int. J. Numer. Methods Fluids* **36**(4), 441–464.

Mitchell, A. R. and Griffiths, D. F. (1980), *The finite difference method in partial differential equations*, John Wiley & Sons, Chichester.

Moran, H. J. P. and Ohayon, R. (1979), 'Substructure variational analysis of the vibrations of coupled fluid-structure systems. Finite element results', *Int. J. Numer. Methods Eng.* **14**(5), 741–755.

Morton, K. W. (1982), 'Schock capturing, fitting and recovery', *in* E. Krause, ed., *Eigth International Conference on Numerical Methods in Fluid Dynamics*, Vol. 170 of *Lecture Notes in Phys.*, Springer-Verlag, Berlin, pp. 77–93.

Morton, K. W. (1983), 'Characteristing Galerkin methods for hyperbolic problems', in M. Pandolfi and R. Piva, eds, Fifth GAMM Conference on Numerical Methods in Fluid Dynamics, Vieweg, Braunschweig, pp. 243–250.

Morton, K. W. (1985), 'Generalised Galerkin methods for hyperbolic problems', *Comput. Methods Appl. Mech. Eng.* **52**(1–3), 847–871.

Morton, K. W. (1996), *Numerical Solution of Convection-Diffusion Problems*, Vol. 12 of *Applied mathematics and mathematical computation*, R.J. Knops and K.W. Morton, eds, Chapman & Hall, London.

Morton, K. W. and Parrott, A. K. (1980), 'Generalised Galerkin methods for first-order hyperbolic equations', *J. Comput. Phys.* **36**(2), 249–270.

Nayroles, B., Touzot, G. and Villon, P. (1992), 'Generalizing the finite element method: diffuse approximation and diffuse elements', *Comput. Mech.* **10**(5), 307–318.

Nguyen, H. and Reynen, J. (1984), 'A space–time least-squares finite element scheme for advection–diffusion equations', *Comput. Methods Appl. Mech. Eng.* **42**(3), 331–342.

Noh, W. F. (1964), 'Cel: A time-dependent, two-space-dimensional, coupled eulerian-lagrange code', *in* B. Alder, S. Fernbach and M. Rotenberg, eds, *Methods in computational physics. Advances in research and applications. Fundamental methods in hydrodynamics*, Vol. 3, Academic Press, New York, pp. 117–179.

Oden, J. T. and Reddy, J. N. (1976), *An introduction to the mathematical theory of finite elements*, Pure and Applied Mathematics, John Wiley & Sons, New York.

Oleĭnik, O. (1957), 'Discontinuous solutions of non-linear differential equations', *Usp. Mat. Nauk (N.S.)* **12**, 3–73 (in Russian), translated in *Am. Math. Soc. Transl. (Ser. 2)* **26** (1963), 95–172.

Oñate, E. (1998), 'Derivation of stabilized equations for numerical solution of advective-diffusive transport and fluid flow problems', *Comput. Methods Appl. Mech. Eng.* **151**(1–2), 233–265.

Oñate, E. and Manzán, M. (1999), 'A general procedure for deriving stabilized space-time finite element methods for advective-diffusive problems', *Int. J. Numer. Methods Fluids* **31**(1), 203–221.

Ozawa, S. (1975), 'Numerical studies of steady flow in a two-dimensional square cavity at high reynolds numbers', *J. Phys. Soc. Jpn.* **38**(3), 889–895.

Paillère, H. (1995), 'Multidimensional upwind residual distribution schemes for the Euler and Navier-Stokes equations on unstructured grids', PhD thesis, Universit é Libre de Bruxelles and von Karman Institute, Belgium.

Park, N.-S. and Liggett, J. A. (1990), 'Taylor-least-squares finite element for two-dimensional advection-dominated unsteady advection-diffusion problems', *Int. J. Numer. Methods Fluids* **11**(1), 21–38.

Park, N. S. and Liggett, J. A. (1991), 'Application of Taylor-least squares finite element to three dimensional advection-diffusion equation', *Int. J. Numer. Methods Fluids* **13**(6), 759–773.

Peraire, J. (1986), 'A finite element method for convection dominated flows', PhD thesis, University College of Swansea, Wales.

Peraire, J., Zienkiewicz, O. C. and Morgan, K. (1986), 'Shallow water problems: a general explicit formulation', *Int. J. Numer. Methods Eng.* **22**(3), 547–574.

Pereira, J. M. C., Kobayashi, M. H. and Pereira, J. C. F. (2001), 'A fourth-order-accurate finite volume compact method for the incompressible Navier-Stokes solutions', *J. Comput. Phys.* **167**(1), 217–243.

Perot, J. B. (1993), 'An analysis of the fractional step method', *J. Comput. Phys.* **108**(1), 51–58.

Perrochet, P. and Azérad, P. (1995), 'Space-time integrated least-squares: solving a pure advection equation with a pure diffusion operator', *J. Comput. Phys.* **117**(2), 183–193.

Pironneau, O. (1981/82), 'On the transport-diffusion algorithm and its applications to the Navier-Stokes equations', *Numer. Math.* **38**(3), 309–332.

Pironneau, O. (1989), *Finite element methods for fluids*, John Wiley & Sons, Chichester.

Pudykiewicz, J. and Staniforth, A. (1984), 'Some properties and comparative performance of semi-Lagrangian method of Robert in the solution of the advection–diffusion equation', *Atmos.-Ocean* **22**(3), 283–308.

Purnell, D. K. (1976), 'Solution of the advective equation by upstream interpolation with a cubic spline', *Mon. Weather Rev.* **104**, 42–48.

Quartapelle, L. (1993), *Numerical solution of the incompressible Navier-Stokes equations*, Vol. 113 of *International Series of Numerical Mathematics*, Birkh äuser-Verlag, Basel.

Quartapelle, L. and Rebay, S. (1990), 'Numerical solution of two-point boundary value problems', *J. Comput. Phys.* **86**(2), 314–354.

Quarteroni, A., Saleri, F. and Veneziani, A. (2000), 'Factorization methods for the numerical approximation of Navier-Stokes equations', *Comput. Methods Appl. Mech. Eng.* **188**(1–3), 505–526.

Quarteroni, A. and Valli, A. (1994), *Numerical Approximation of Partial Differential Equations*, Vol. 23 of *Springer Series in Computational Mathematics*, Springer-Verlag, Berlin.

Reddy, J. N. and Gartling, D. K. (2001), *The finite element method in heat transfer and fluid dynamics*, second edn, CRC Press, Boca Raton, FL.

Richtmyer, R. D. and Morton, K. W. (1967), *Difference Methods for Initial-Value Problems*, Vol. 4 of *Interscience tracts in pure and applied mathematics*, second edn, L. Bers, R. Courant and J.J. Stoker, eds, John Wiley & Sons, New York.

Robert, A. (1981), 'A stable numerical integration scheme for the primitive meteorological equations', *Atmos.-Ocean* **19**(1), 35–46.

Robert, A. (1982), 'A semi-Lagrangian and semi-implicit numerical integration scheme for the primitive meteorological equations', *J. Meteorol. Soc. Jpn* **60**, 319–325.

Rodríguez-Ferran, A. and Huerta, A. (1999), 'Adapting Broyden method to handle linear constraints imposed via Lagrange multipliers', *Int. J. Numer. Methods Eng.* **46**(12), 2011–2026.

Roe, P. L. (1981), 'Approximate Riemann solvers, parameter vectors, and difference schemes', *J. Comput. Phys.* **43**(2), 357–372.

Roe, P. L. (1984), 'Generalized formulation of TVD Lax-Wendroff schemes', Technical Report n. 84-53, ICASE, Hampton.

Roe, P. L. (1985), 'Some contributions to the modelling of discontinuous flows', in B. Engquist, S. Osher and R. Somerville, eds, Large-scale computations in fluid mechanics, Part 2 (La Jolla, Calif., 1983), American Mathematical Society, Providence, RI, pp. 163–193.

Roe, P. L. (1986), 'Discrete models for the numerical analysis of time-dependent multidimensional gas dynamics', *J. Comput. Phys.* **63**(2), 458–476.

Sachdev, P. L. (1987), *Nonlinear diffusive waves*, Cambridge University Press, Cambridge.

Sampaio, P. A. B. D. and Moreira, M. L. (2000), 'A new finite element formulation for both compressible and nearly incompressible fluid dynamics', *Int. J. Numer. Methods Fluids* **32**(1), 51–78.

Selmin, V. (1987), 'Third-order finite element schemes for the solution of hyperbolic problems', Technical Report 707, INRIA, France.

Selmin, V., Donéa, J. and Quartapelle, L. (1985), 'Finite element methods for nonlinear advection', *Comput. Methods Appl. Mech. Eng.* **52**(1–3), 817–845.

Shakib, F. (1989), 'Finite element analysis of the compressible Euler and Navier-Stokes equations', PhD thesis, Stanford University, USA.

Shakib, F. and Hughes, T. J. R. (1991), 'A new finite element formulation for computational fluid dynamics. IX. Fourier analysis of space-time Galerkin/least-squares algorithms', *Comput. Methods Appl. Mech. Eng.* **87**(1), 35–58.

Shakib, F., Hughes, T. J. R. and Johan, Z. (1991), 'A new finite element formulation for computational fluid dynamics. X. The compressible Euler and Navier-Stokes equations', *Comput. Methods Appl. Mech. Eng.* **89**(1–3), 141–219.

Sod, G. A. (1978), 'A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws', *J. Comput. Phys.* **27**(1), 1–31.

Soulaïmani, A. and Fortin, M. (1994), 'Finite element solution of compressible viscous flows using conservative variables', *Comput. Methods Appl. Mech. Eng.* **118**(3–4), 319–350.

Staniforth, A. and Côté, J. (1991), 'Semi-Lagrangian integration schemes for atmospheric models - A review', *Mon. Weather Rev.* **119**(9), 2206–2223.

Staniforth, A. and Pudykiewicz, J. (1985), 'Reply to comments on and addenda to "Some properties and comparative performance of semi-Lagrangian method of Robert in the solution of the advection–diffusion equation"', *Atmos.-Ocean* **23**, 195–200.

Steger, J. L. and Warming, R. F. (1981), 'Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods', *J. Comput. Phys.* **40**(2), 263–293.

Stein, L. R., Gentry, R. A. and Hirt, C. (1977), 'Computational simulation of transient blast loading on three-dimensional structures', *Comput. Methods Appl. Mech. Eng.* **11**, 57–74.

Strang, G. and Fix, G. J. (1973), *An analysis of the finite element method*, Prentice Hall Series in Automatic Computation, Prentice Hall, Englewood Cliffs, NJ.

Strouboulis, T., Copps, K. and Babuška, I. (2000), 'The generalized finite element method: an example of its implementation and illustration of its performance', *Int. J. Numer. Methods Eng.* **47**(8), 1401–1417.

Sweby, P. K. (1984), 'High resolution schemes using flux limiters for hyperbolic conservation laws', *SIAM J. Numer. Anal.* **21**(5), 995–1011.

Tanguay, M., Simard, A. and Staniforth, A. (1989), 'A three-dimensional semi-Lagrangian integration scheme for the Canadian regional finite-element forecast model', *Mon. Weather Rev.* **117**, 1861–1871.

Taylor, C. and Hood, P. (1973), 'A numerical solution of the Navier-Stokes equations using the finite element technique', *Comput. Fluids* **1**(1), 73–100.

Temam, R. (1969), 'Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires. II', *Arch. Ration. Mech. Anal.* **33**, 377–385.

Temam, R. (1991), 'Remark on the pressure boundary condition for the projection method', *Theor. Comput. Fluid Dyn.* **3**(3), 181–184.

Temam, R. (2001), *Navier-Stokes equations. Theory and numerical analysis*, AMS Chelsea Publishing, Providence, RI. Corrected reprint of the 1984 edition [North-Holland, Amsterdam, 1984].

Tezduyar, T. E. (1992), 'Stabilized finite element formulations for incompressible flow computations', *Adv. Appl. Mech.* **28**, 1–44.

Tezduyar, T. E. and Ganjoo, D. K. (1986), 'Petrov-Galerkin formulations with weighting functions dependent upon spatial and temporal discretization: applications to transient convection-diffusion problems', *Comput. Methods Appl. Mech. Eng.* **59**(1), 49–71.

Tezduyar, T. E., Mittal, S., Ray, S. E. and Shih, R. (1992), 'Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements', *Comput. Methods Appl. Mech. Eng.* **95**(2), 221–242.

Tezduyar, T. E., Mittal, S. and Shih, R. (1991), 'Time-accurate incompressible flow computations with quadrilateral velocity-pressure elements', *Comput. Methods Appl. Mech. Eng.* **87**(2–3), 363–384.

Tezduyar, T. E. and Osawa, Y. (2000), 'Finite element stabilization parameters computed from element matrices and vectors', *Comput. Methods Appl. Mech. Eng.* **190**(3–4), 411–430.

Tezduyar, T. E. and Park, Y. (1986), 'Discontinuity-capturing finite element formulations for nonlinear convection-diffusion-reaction equations', *Comput. Methods Appl. Mech. Eng.* **59**(3), 307–325.

Trulio, J. G. (1966), 'Theory and structure of the AFTON codes', Technical Report AFWL-TR-66-19, Air Force Weapons Lab.

Tuann, S. Y. and Olson, M. D. (1978), 'Review of computing methods for recirculating flows', *J. Comput. Phys.* **29**(1), 1–19.

van der Waerden, B. L. (1970), *Algebra. Vol. 1*, Frederick Ungar, New York.

van Leer, B. (1974), 'Towards the ultimate conservative difference scheme II. Monotonicity and conservation combined in a second order scheme', *J. Comput. Phys.* **14**, 361–370.

van Leer, B. (1982), 'Flux-vector splitting for the Euler equations', *in* E. Krause, ed., *Eigth International Conference on Numerical Methods in Fluid Dynamics*, Vol. 170 of *Lecture Notes in Phys.*, Springer-Verlag, Berlin, pp. 507–512.

von Neumann, J. and Richtmyer, R. D. (1950), 'A method for the numerical calculation of hydrodynamic shocks', *J. Appl. Phys.* **21**, 232–237.

Wait, R. and Mitchell, A. R. (1985), *Finite element analysis and applications*, John Wiley & Sons, New York.

Warming, R. F. and Hyett, B. J. (1974), 'The modified equation approach to the stability and accuracy analysis of finite-difference method', *J. Comput. Phys.* **14**, 159–179.

Wilkins, M. L. (1969), 'Calculation of elastic-plastic flow', Technical Report UCRL-7322 Rev 1, University of California.

Wong, J. S., Darmofal, D. L. and Peraire, J. (2001), 'The solution of the compressible Euler equations at low Mach numbers using a stabilized finite element algorithm', *Comput. Methods Appl. Mech. Eng.* **190**(43–44), 5719–5737.

Woodward, P. and Colella, P. (1984), 'The numerical simulation of two-dimensional fluid flow with strong shocks', *J. Comput. Phys.* **54**(1), 115–173.

Yanenko, N. N. (1971), *The method of fractional steps. The solution of problems of mathematical physics in several variables*, Springer-Verlag, Berlin. Translated from the Russian by T. Cheron. English translation edited by M. Holt.

Yee, H. C. (1987), 'Construction of explicit and implicit symmetric TVD schemes and their applications', *J. Comput. Phys.* **68**(1), 151–179.

Yosida, K. (1995), *Functional analysis*, Springer-Verlag, Berlin. Reprint of the sixth (1980) edition.

Zienkiewicz, O. C. and Codina, R. (1995), 'A general algorithm for compressible and incompressible flow. I. The split, characteristic-based scheme', *Int. J. Numer. Methods Fluids* **20**(8–9), 869–885.

Zienkiewicz, O. C. and Godbole, P. N. (1975), 'Viscous, incompressible flow with special reference to non-Newtonian (plastic) fluids', *in* R. H. Gallagher, J. T. Oden, C. Taylor and O. C. Zienkiewicz, eds, *Finite elements in fluids, Vol. 1: Viscous Flow and Hydrodynamics*, International Symposium on the Finite Element Method in Flow Problems held at Swansea, Wales, John Wiley & Sons, Chichester, pp. 25–55.

Zienkiewicz, O. C. and Morgan, K. (1983), *Finite elements and approximation*, John Wiley & Sons, New York.

Zienkiewicz, O. C., Morgan, K., Satya Sai, B. V. K., Codina, R. and V ázquez, M. (1995), 'A general algorithm for compressible and incompressible flow. II. Tests on the explicit form', *Int. J. Numer. Methods Fluids* **20**(8–9), 887–913.

Zienkiewicz, O. C. and Taylor, R. L. (2000a), *The finite element method. Vol. 1 The basis*, fifth edn, Butterworth Heinemann, Oxford.

Zienkiewicz, O. C. and Taylor, R. L. (2000b), *The finite element method. Vol. 3 Fluid dynamics*, fifth edn, Butterworth Heinemann, Oxford.

Index

Finite Element Methods for Flow Problems. Jean Donea and Antonio Huerta

ISBN: 0-471-49666-9