Analysis of spectral element methods : with application to incompressible flow

Citation for published version (APA):
Timmermans, L. J. P. (1994). *Analysis of spectral element methods : with application to incompressible flow.* [Phd Thesis 1 (Research TU/e / Graduation TU/e), Mechanical Engineering]. Technische Universiteit Eindhoven. https://doi.org/10.6100/IR412687

DOI:
10.6100/IR412687

Document status and date:
Published: 01/01/1994

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:

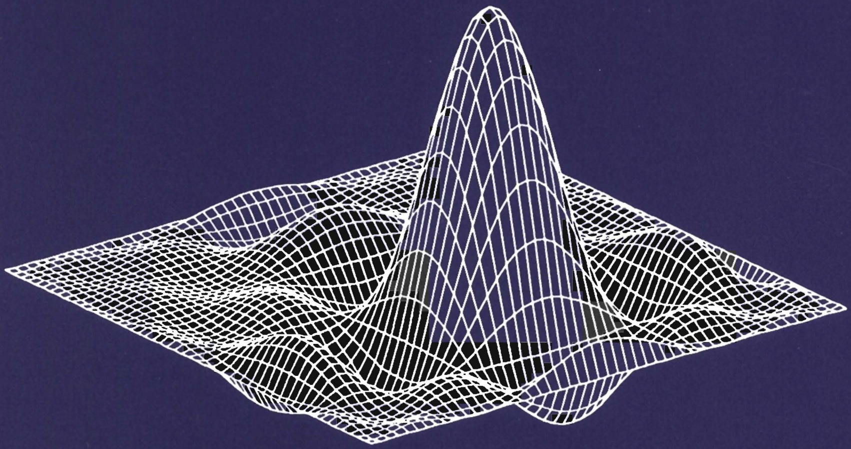
openaccess@tue.nl

providing details and we will investigate your claim.

Download date: 16. Nov. 2023

Analysis of spectral element methods

with application to incompressible flow



Luc Timmermans

Eindhoven University of Technology

Analysis of spectral element methods

with application to incompressible flow

CIP-DATA KONINKLIJKE BIBLIOTHEEK, DEN HAAG

Timmermans, Lucas Johannes Petrus

Analysis of spectral element methods with application to
incompressible flow / Lucas Johannes Petrus Timmermans. -
Eindhoven : Eindhoven University of Technology
Thesis Eindhoven. - With ref. - With summary in Dutch.
ISBN 90-386-0472-6
Subject headings: spectral element methods.

Analysis of spectral element methods

with application to incompressible flow

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Eindhoven
op gezag van de Rector Magnificus, prof.dr. J.H. van Lint,
voor een commissie aangewezen door het College van Dekanen
in het openbaar te verdedigen
op dinsdag 8 maart 1994 om 16.00 uur

door

Lucas Johannes Petrus Timmermans

geboren te Sittard

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr.ir. J.D. Janssen
prof.dr. R.M.M. Mattheij

en de copromotor:

dr.ir. F.N. van de Vosse

Voor Margot

"The things I tell you will not be wrong.
Better to listen than to talk!"

The Giant from 'Twin Peaks'

To start with, I would like to express my gratitude to everyone who has contributed to the research presented in this thesis.

Most of all, I would like to thank Frans van de Vosse for his vast contribution to this work, both qualitively and quantitively; and for the freedom he granted me during the four years of my research. In the early stages of my work discussions with Jos Jansen and Guus Segal are gratefully acknowledged. To the latter I am also thankful for the help provided with issues concerning implementation in SEPRAN. During the last year of my work Peter Minev was strongly involved. I thank him for his great contribution to the last chapters.

I also take this oppurtunity to thank some colleagues who made my stay at the section of Computational (and Experimental) Mechanics an enjoyable one. In particular I am indebted to Jeroen and Michiel for special evenings with BOB.

Finally, I am very grateful to my family and friends for their support; especially to Wim for his sound advice and to my wife Margot for sharing these tumultuous years with me.

Luc Timmermans
Eindhoven, March 1994

This research is supported by the Dutch Foundation of Technology (STW) grant number EWT 88.1442.

Contents

Summary xiii

Notation xv

1 Introduction 1

1.1 Numerical analysis of carotid artery flow 1
1.2 Objectives and methodology employed 3

2 Spectral element methods 7

2.1 Introduction 7
2.2 Spectral element discretization in one dimension 9
2.2.1 p-type weighted residual methods 9
2.2.2 Spectral approximation 11
2.2.3 Pseudospectral approximation 13
2.2.4 A one-dimensional model equation 14
2.3 Spectral elements in more dimensions 18
2.3.1 Tensor product formulation 18
2.3.2 The steady convection-diffusion equation 19
2.4 Numerical results 24
2.4.1 Application to problems with analytical solution . . . 24
2.4.2 The use of SUPG in spectral elements 27
2.4.3 Application to heat-transfer in a Poiseuille flow 30

3 Finite element preconditioning 33

3.1 The conditioning of a spectral matrix 33
3.2 Eigenvalues of the preconditioned matrix 35
3.3 Finite element preconditioned algorithms 40
3.3.1 The basic iterative scheme 40
3.3.2 Acceleration techniques for symmetric systems 40

3.3.3 A Bi-CGSTAB algorithm for non-symmetric systems . 41
3.4 Numerical results 42
3.4.1 Application to a (symmetric) Poisson problem 42
3.4.2 Application to a (non-symmetric) convection-diffusion problem 47
3.5 Conclusions 48

4 Solution of unsteady convection-diffusion problems 51
4.1 Convection-diffusion problems 51
4.2 Operator splitting approach 53
4.3 Taylor–Galerkin time-integration 56
4.3.1 Schemes for linear convection 56
4.3.2 Schemes for non-linear convection 59
4.3.3 Spectral element discretization 60
4.3.4 Numerical results 62
4.4 Operator splitting for convection-diffusion 72
4.4.1 The time-integration scheme 72
4.4.2 Spectral element discretization 72
4.4.3 Numerical results 74
4.5 Conclusions 76

5 Solution of Navier–Stokes problems 79
5.1 Introduction 79
5.2 Governing equations 82
5.3 Solution of the Stokes equations 84
5.3.1 Projection methods 84
5.3.2 The pressure correction method 85
5.3.3 Spectral element discretization 91
5.3.4 Application to an analytical test case 92
5.4 Solution of the Navier–Stokes equations 98
5.4.1 The operator splitting approach 98
5.4.2 The analytical test case revisited 100
5.5 Buoyancy-driven flow in an enclosed cavity 101
5.6 Conclusions 108

6 Discussion 111

A Orthogonal polynomials in $(-1,1)$ **117**
A.1 Sturm–Liouville problems 117
A.2 Orthogonal systems of polynomials 118
A.3 Discrete polynomial expansions 119

B The Legendre polynomials **121**

C Finite element preconditioned algorithms **127**

D Backward differences formulae **129**

E Spectral element pressure correction equations **133**

References **137**

Samenvatting **145**

Curriculum Vitae **147**

Summary

The objective of this thesis is to develop highly accurate and highly efficient solution methods for incompressible flow problems with a dominating convective part, using a high-order spectral element method. The reason to apply spectral elements is the high accuracy that can be achieved when approximating 'sufficiently' smooth phenomena within a geometrically flexible framework.

The first part of this thesis is mainly concerned with the spectral element method. Spectral element methods are high-order p-type weighted residual techniques for the solution of partial differential equations. They combine the geometric flexibility of the well-known h-type finite element method with the attractive approximation properties of spectral methods. For a special class of basis functions, spectral element methods exhibit exponential convergence when approximating sufficiently smooth functions, a clear advantage compared to the algebraic convergence that can be obtained using a conventional low-order method. Of great importance for a numerical method is also the efficiency, both with respect to memory usage and computing time, with which the resulting system of discrete equations can be solved. For the spectral element method efficiency is achieved firstly by the use of iterative solution algorithms using low-order finite element preconditioning, and secondly by the use of a tensorial basis that is derived from the one-dimensional basis. This allows a method to decrease the number of operations needed to compute the residuals in an iterative technique. Using an iterative solver, the full spectral element system matrix no longer needs to be stored, resulting in a more efficient memory usage.

The second part of this thesis deals with the solution of incompressible flow problems. A good starting point is the solution of general unsteady convection-diffusion problems. If the convection part dominates the problem, application of low-order Galerkin methods often leads to spurious oscillations in the numerical solution. The spectral element approximation on the other hand exhibits minimal numerical dispersion and diffusion. For unsteady problems the choice of time-integration becomes also important. Since convection and diffusion are totally different phenomena, it is appropriate to treat them separately using an operator splitting approach that decouples the problem into a pure convection problem and a pure diffusion problem. This choice is influenced by the need for an efficient numerical scheme, since then the diffusion equation can be solved using an implicit time-integration with a large time-step, and the convection equation can be solved using an explicit time-integration with, if necessary, a smaller time-step. The use of a high-order spectral element method has a great advantage because in that case it is valid to apply a diagonal mass matrix, since it is not as restrictive with respect to accuracy as is the case for low-order methods. Consequently, the resulting discrete system does not involve the solution of a system, but only the calculation of matrix-vector products which can be performed on elemental level. The diffusion system can be solved using a standard iterative procedure.

The next step is the solution of the Navier–Stokes equations for incompressible flow. Again, the choice of solution method is largely determined by the need for an efficient method, which is found in a modified continuous pressure correction (or predictor-corrector) scheme. The reason to choose a continuous decoupling algorithm is because in that case the degree of approximation for velocity and pressure can be taken the same, since there is no need to satisfy any form of the Brezzi–Babuška condition. The resulting set of discrete equations is is then very easy to implement. In a pressure correction method the set of equations is split into a set of problems for both velocity and pressure. This results in a convection-diffusion problem for an intermediate velocity field that can be solved using the operator splitting technique. Enforcing the incompressibility constraint yields a Poisson equation for the pressure correction which can be solved using a finite element preconditioning technique. Next, the velocity is 'corrected'. The main advantage of a pressure correction scheme is that the final velocity obtained is guaranteed to be divergence-free. In this thesis a modification to the standard approach is proposed in order to derive a second-order consistent scheme. Using this modification there are no longer any problems relating 'artificial' boundary conditions for the pressure correction.

Notation

Conventions

a, A, α	scalar
$\mathbf{a}$	vector or column
a_i	component of $\mathbf{a}$
$\mathbf{A}$	matrix
A_{ij}	element of $\mathbf{A}$
$\mathbf{A}^T$	transposed of $\mathbf{A}$
$\mathbf{A}^{-1}$	inverse of $\mathbf{A}$
$\lvert\mathbf{A}\rvert$	determinant of $\mathbf{A}$
$\kappa(\mathbf{A})$	condition number of $\mathbf{A}$
$\rho(\mathbf{A})$	spectral radius of $\mathbf{A}$
$\sigma(\mathbf{A})$	spectrum of $\mathbf{A}$
$\boldsymbol{\alpha}$	tensor
α_{ij}	component of $\boldsymbol{\alpha}$
$\mathcal{A}$	operator

Symbols

a	diffusion coefficient (2.2.4)
b	Helmholtz coefficient (2.2.4)
c	unknown scalar function (2.2.1)
c_i	unknown expansion coefficients (2.2.1)
$\tilde{c}_i$	continuous expansion coefficients (2.2.2)
$\hat{c}_i$	discrete expansion coefficients (2.2.3)
c_h	approximate solution to c (2.2.1)
c_{pq}	coefficients in two dimensions (2.3.2)
c_{pqr}	coefficients in three dimensions (2.3.1)
$\mathbf{c}$	(1) column of coefficients (2.2.4)

	(2) unknown vector function (4.1)
$\mathbf{c}^n$	approximation of $\mathbf{c}$ at $t = t^n$ (4.2)
$\mathbf{c}^m$	approximation of $\mathbf{c}$ at $s = s^m$ (4.3.1)
$\tilde{\mathbf{c}}$	unknown of a convection problem for $\mathbf{c}$ (4.2)
d	dimension of space (2.3.1)
$\mathbf{d}^n$	approximation of the diffusive terms at $t = t^n$ (4.4.2)
$\tilde{\mathbf{d}}^n$	end solution of the convection problem for $\mathbf{d}^n$ (4.4.2)
$\tilde{\mathbf{d}}^m$	intermediate approximation of $\tilde{\mathbf{d}}^n$ at $s = s^m$ (4.4.2)
e	reference element $[-1, 1]$ (2.4.4)
$\mathbf{e}$	reference element $[-1, 1] \times [-1, 1]$ (2.3.2)
f	scalar source term (2.2.1)
f_h	approximation of f (2.2.4)
$\mathbf{f}$	(1) righthand side column (2.2.4)
	(2) vector source term (4.1)
$\mathbf{f}^n$	approximation of $\mathbf{f}$ at $t = t^n$ (4.2)
g	acceleration of gravity (5.5)
$\mathbf{g}$	gravitational acceleration vector (5.5)
h	width of an element (2.4.2)
h_n	normal component of $\mathbf{h}$ (5.2)
h_τ	tangential component of $\mathbf{h}$ (5.2)
$\mathbf{h}$	value of surface traction force on $\boldsymbol{\Gamma}_{\boldsymbol{\sigma}}$ (5.2)
k	iteration counter (3.3.1)
$\mathbf{k}$	value of $\mathbf{u}$ at $\boldsymbol{\Gamma}_{\mathbf{u}}$ (5.2)
l	characteristic length (5.5)
m	time counter for convection problems (4.3.1)
n	degree of approximation (2.2.1)
	time counter for diffusion problems (4.2)
n_e	number of elements (2.2.4)
$\mathbf{n}$	outward pointing unit normal (5.2)
p	hydrostatic or kinematic pressure (5.2)
p^n	approximation of p at $t = t^n$ (5.3.2)
p^*	pressure correction (5.3.2)
$\mathbf{p}^n$	pressure coefficients at $t = t^n$ (5.3.3)
$\mathbf{p}^*$	pressure correction coefficients (5.3.3)
p_i	orthogonal polynomial (A.2)
q	Galerkin test function for the pressure (E)
$\mathbf{r}^k$	spectral residual (3.3.1)
s	time in a convection problem (4.2)

s^m	time level for convection problems (4.3.1)
t	time in general or for a diffusion problem (4.1)
t^n	time level for diffusion problems (4.2)
u	scalar velocity (2.4.2)
u_h	approximate scalar velocity (4.3.4)
u_n	normal component of $\mathbf{u}$ (5.2)
u_τ	tangential component of $\mathbf{u}$ (5.2)
$\mathbf{u}$	velocity vector or field (2.3.2)
$\mathbf{u}^n$	approximation of $\mathbf{u}$ at $t = t^n$ (4.4.1)
$\mathbf{u}^m$	approximation of $\mathbf{u}$ at $s = s^m$ (4.3.1)
$\mathbf{u}^*$	intermediate velocity field (5.3.2)
$\tilde{\mathbf{u}}$	unknown of a convection problem for $\mathbf{u}$ (5.4.1)
$\tilde{\mathbf{u}}^n$	end solution of the convection problem for $\mathbf{u}^n$ (4.4.1)
$\tilde{\mathbf{u}}^m$	intermediate approximation of $\tilde{\mathbf{u}}^n$ at $s = s^m$ (4.4.1)
v	Galerkin scalar test function (2.2.1)
v_h	discrete test function (2.2.1)
$\tilde{v}$	SUPG test function (2.4.2)
$\mathbf{v}$	Galerkin vector test function (4.3.3)
w	weighting function (A.1)
x	coordinate in one dimension (2.2.1)
x_k	global interpolation point (2.2.3)
x_l	left boundary point of an interval (2.2.4)
x_r	right boundary point of an interval (2.2.4)
$\mathbf{x}$	coordinate in two dimensions (2.3.2)
y	reference coordinate in one dimension (2.2.4)
$\mathbf{y}$	reference coordinate in two dimensions (2.2.4)

$\mathbf{C}$	convection matrix (2.3.2)
$\mathbf{C}^e$	convection matrix at elemental level (2.3.2)
$\mathbf{C}^m$	convection matrix at time level $s = s^m$ (4.3.3)
$\mathcal{C}$	convection operator (4.2)
$\mathbf{D}$	diffusion or stiffness matrix (2.4.4)
$\mathbf{D}^e$	diffusion matrix at elemental level (2.4.4)
$\mathbf{D}^m$	special diffusion matrix at time level $s = s^m$ (4.3.3)
$\mathcal{D}$	diffusion operator (4.2)
$\mathbf{F}$	finite element system matrix (3.2)
$\mathbf{H}$	Helmholtz matrix (2.2.4)
$\mathbf{H}^e$	Helmholtz matrix at elemental level (2.2.4)

$\mathcal{I}$	identity operator (4.2)
$\mathcal{I}_n$	interpolation operator (2.2.3)
J	Jacobian in one dimension (2.2.4)
J_k	discrete Jacobian in one dimension (2.2.4)
$\mathbf{J}$	Jacobian in two dimensions (2.3.2)
$\mathbf{J}_{kl}$	discrete Jacobian in two dimensions (2.3.2)
$\mathbf{K}$	pressure stiffness matrix (5.3.3)
$\mathbf{K}^e$	pressure stiffness matrix at elemental level (E)
L_n	Legendre polynomial of degree n (2.4.4)
L^{n+1}	divergence of velocity at time level $n+1$ (5.3.2)
$\mathbf{L}$	divergence matrix (5.3.3)
$\mathbf{L}^e$	divergence matrix at elemental level (E)
$\mathcal{L}$	general differential operator (2.2.1)
$\mathbf{M}$	mass matrix (2.4.4)
$\mathbf{M}^e$	mass matrix at elemental level (2.4.4)
N	number of degrees of freedom in one direction (2.4.1)
Nu	Nusselt number (5.5)
$\mathbf{P}^{(m)}$	derivative matrix of order m (2.4.4)
Pr	Prandtl number (5.5)
$\mathbf{Q}$	gradient matrix (5.3.3)
$\mathbf{Q}^e$	gradient matrix at elemental level (E)
$Q_{\mathcal{C}}^{(t^*,t)}$	integrating factor in $\mathcal{C}$ from t to t^* (4.2)
Ra	Rayleigh number (5.5)
$\mathbf{S}$	spectral system matrix (2.3.1)
$\mathbf{S}^e$	spectral system matrix at elemental level (2.3.1)
$\mathcal{S}_n$	continuous expansion operator (2.2.2)
T	temperature (2.4.3)
U	solution space (2.2.1)
U_h	approximate solution space (2.2.1)
V	space of Galerkin test functions (2.2.1)
V_h	approximate space of test functions (2.2.1)

α	relaxation parameter (3.3.1)
α^k	dynamic relaxation parameter (3.3.1)
α_{opt}	optimal relaxation parameter (3.3.2)
β	thermal expansion coefficient (5.5)
β_i	coefficient of backward differences schemes (4.2)
γ_k	Legendre Gauss–Lobatto weights (2.2.4)

δ	thermal diffusivity (5.5)
ε	discrete maximum error (2.4.1)
η	dynamic viscosity (2.3.2)
θ	coefficient of trapezoidal rule (4.2)
ϑ	upwind parameter (2.4.2)
λ	general eigenvalue (3.1)
λ_i	eigenvalue of a Sturm–Liouville problem (2.2.2)
$\lambda_{\max}$	maximum eigenvalue (3.3.2)
$\lambda_{\min}$	minimum eigenvalue (3.3.2)
μ	upwind coefficient (2.4.2)
ν	kinematic viscosity (5.2)
ρ	fluid density (5.2)
$\boldsymbol{\sigma}$	Cauchy stress tensor (5.2)
ξ_k	Legendre Gauss–Lobatto points (2.2.4)
ϕ_i	Lagrangian interpolant (2.2.3)
φ_i	basis function for the solution space (2.2.1)
ψ_k	basis function for the space of test functions (2.2.1)

$\boldsymbol{\Gamma}$	boundary in two dimensions (2.3.2)
$\boldsymbol{\Gamma}_{\mathbf{u}}$	boundary where velocity is prescribed (5.2)
$\boldsymbol{\Gamma}_{\boldsymbol{\sigma}}$	boundary where stress is prescribed (5.2)
Δ^2	discrete second-order BDF operator (5.3.2)
Δs	time step for convection problems (4.2)
Δt	time step for diffusion problems (4.2)
Ω	one-dimensional domain (2.2.1)
Ω_e	one-dimensional element (2.2.4)
$\overline{\Omega}$	closure of Ω (2.2.1)
$\boldsymbol{\Omega}$	two-dimensional domain (2.3.2)
$\boldsymbol{\Omega}_e$	two-dimensional element (2.3.2)
$\overline{\boldsymbol{\Omega}}$	closure of $\boldsymbol{\Omega}$ (2.3.2)

∇	gradient operator (2.3.2)
$\nabla_{\mathbf{y}}$	gradient operator with respect to $\mathbf{y}$ (2.3.2)
$\tilde{\nabla}$	transformed gradient (2.3.2)
$\tilde{\nabla}_{pqrs}$	discrete transformed gradient (2.3.2)

Spaces, inner products and norms

$a(\cdot,\cdot)$	bilinear form (2.2.1)
$a(\cdot,\cdot)_{gl}$	discrete bilinear form (2.2.4)
C^0	space of continuous functions (2.2.1)
C^1	space of continuous differentiable functions (2.2.4)
H_0^1	Sobolev space of order one (2.4.4)
$H_{\mathrm{u},0}^1$	Sobolev space for the velocity (E)
$H_{p,0}^1$	Sobolev space for the pressure (E)
L^2	Lebesgue space (2.2.1)
$(\cdot,\cdot), \|\cdot\|$	inner product, norm in L^2 (2.2.4)
L_w^2	weighted Lebesgue space (A.2)
$(\cdot,\cdot)_{,w}, \; \|\cdot\|_w$	inner product, norm in L_w^2 (A.2)
P_n	space of polynomials of degree $\leq n$ (2.2.4)
P_{2n-1}	space of polynomials of degree $\leq 2n-1$ (2.2.4)
$\mathbf{R}^d$	d-dimensional space of real numbers (4.1)
$(\cdot,\cdot)_V, \|\cdot\|_V$	inner product, norm in V (2.2.1)
$(\cdot,\cdot)_{gl}, \|\cdot\|_{gl}$	discrete inner product, norm (2.2.4)

Chapter 1

Introduction

1.1 Numerical analysis of carotid artery flow

Due to the enormous developments in computer facilities in recent years numerical simulations of physiological flows, which often involve large computations of three-dimensional unsteady phenomena, have become increasingly interesting. An example of the application of large scale numerical computations of physiological flow, or more specific blood flow, can be found in the research project 'Atherosclerosis' at the Eindhoven University of Technology. This project shows that, in addition to *in-vivo* and *in-vitro* experiments, a numerical analysis of blood flow patterns in the carotid artery bifurcation plays an important role in studies of both the genesis and the detectability of atherosclerosis. Atherosclerosis is an arterial disease resulting in localized stiffening and thickening of the arterial wall (atherosclerotic laesions) and may lead to narrowing or even occlusion of the artery affected. Especially arterial bifurcations and bends are found to be sites of preference. Besides biochemical and cytological aspects, hemodynamical aspects are assumed to play an important role in the genesis of the atherosclerotic disease.

In previous studies of this project, both experimental and numerical techniques have been developed and used to analyze the flow in non-diseased arteries. As regards the numerical simulation of the flow field, Van de Vosse *et al.* (1990) studied the steady and unsteady flow in two-dimensional bifurcation geometries with rigid walls. The numerical technique employed was a standard finite element method based on the software package SEPRAN (Segal, 1984). The finite element technique was used to solve the unsteady Navier–Stokes equations for incompressible Newtonian flow; the effect of the non-Newtonian (visco-elastic) properties of blood being currently investigated in this group. Comparison with data obtained from experiments showed that the finite element simulation was accurate for laminar two-dimensional flow, but also demonstrated that fully three-dimensional flow simulations including the analysis of secondary motion are needed for obtaining valid information about the flow phenomena in bifurcation geometries. The extension to three-dimensional simulations of Newtonian flow was described by Rindt *et al.* (1990), who gave detailed information about both the primary and secondary velocity distributions of the steady and unsteady flow in three-dimensional rigid bifurcation geometries. In order to analyze the influence of the distensibility of the arterial wall, the numerical model has been extended by means of a decoupled fluid-structure interaction model. To this end Reuderink *et al.* (1993) developed a wave-propagation model based on pressure-radius relations appropriate for bifurcation geometries. This model was used to compute the motion of the arterial wall. The wall motion serves as as boundary condition for the velocity computations.

Having dealt extensively with the analysis of flow phenomena in non-stenosed bifurcation geometries, the obvious next step is to look more closely at the influence of minor stenoses on the local velocity distributions in the carotid artery. Recently performed experiments (Palmen *et al.*, 1993) confirm information from literature (Ku, 1983) that stenoses may influence the stability of the flow. Since flow instabilities originate from relatively small scaled disturbances, a highly accurate numerical technique with minimal numerical diffusion (or damping) is necessary to ensure an accurate simulation of these flow patterns. Moreover, since flow instabilities are essentially unsteady phenomena, also much attention must be paid to the choice of time-integration for the simulation of unsteady incompressible flow. Former numerical results (Van de Vosse, 1987) showed that standard finite element solution of the incompressible Navier–Stokes equations, may not be accurate enough to resolve the apparent flow instabilities.

Rather than submit to a severe refinement of mesh and time-step, which is not very practical, an alternative numerical technique may be employed in order to obtain a more accurate numerical solution of the flow field. In this study, a possible substitute is found in a high-order spectral element method (Maday and Patera, 1989). This technique combines the geometrical flexibility of the finite element method with the rapid convergence for smooth phenomena (as occur in incompressible fluid dynamics) of spectral methods (Gottlieb and Orszag, 1977; Canuto *et al.*, 1988). The spectral element method is of importance in a wide range of applications where flow instabilities and transition from laminar to turbulent flow occur (Kaiktsis *et al.*, 1991). Moreover, since this technique exhibits minimal numerical damping and dispersion, it is of importance in all numerical simulations where the diffusion is dominated by the convection (differential models in visco-elastic flow, conversion in reactive flows, particle tracking).

1.2 Objectives and methodology employed

The research in this thesis has the following goals:

- to analyze and implement a spectral element method suitable for the approximation of partial differential equations in general.
- to optimize the spectral element method with respect to computing time and memory allocation.
- to compare the spectral element method (where possible) to the finite element method as regards accuracy and computational costs.
- to address important aspects of incompressible flow computations, such as the choice of time-integration and the choice of solution method for the Navier–Stokes equations.

To begin with, in chapter 2 the spectral element method is analyzed from a theoretical point of view. The technique is derived within the general framework of Galerkin weighted residual approximation methods. Of special importance is the choice of basis functions; for a special class of basis functions spectral methods exhibit exponential convergence when approximating smooth functions. In order to get more insight in spectral element methods, the discretization process is described by means of a one-dimensional model equation. The basis for the discretization process is formed by the weak or variational formulation. Important aspects such as the choice of numerical integration and interpolation and the form of the resulting discrete matrix-vector system are addressed. The discretization process in more dimensions is essentially an extension of the process in one dimension. However, several aspects have to be discussed separately, such as the use of isoparametric elements to describe the geometry. The key to efficiency for a high-order spectral method in more dimensions is the use of a tensorial basis; the approximate solution is expanded in a tensor-product of one-dimensional basis functions. As an example the discretization process for the two-dimensional steady convection-diffusion equation is given. In order to verify the theoretical accuracy of spectral methods (in comparison to that of finite element methods), numerical results are presented. The spectral element code is based on an implementation in the software package SEPRAN.

Chapter 3 deals with efficient iterative solvers for spectral element systems. Due to the high-order character of the spectral element approximation, the resulting system matrix is in general fairly full. In order to obtain an efficient solution procedure iterative solvers are virtually necessary, especially in more dimensions. The drawback of an iterative procedure is the conditioning of the spectral matrix. If the degree of approximation becomes large, the condition number also becomes large compared to low-order methods. As a consequence, the number of iterations needed to obtain convergence increases significantly if the degree of approximation increases. A strategy to overcome this problem is the use of finite element preconditioning (Deville and Mund, 1985). In this procedure the spectral matrix is preconditioned by the spectrally close finite element matrix based on the same interpolation points. The preconditioned system is well-conditioned and the condition number is practically independent of the degree of approximation. Moreover, the full spectral matrix no longer needs to be stored since the spectral residual is computed in an element-by-element procedure. The idea of finite element preconditioning is incorporated into several iterative algorithms, both for symmetric and non-symmetric systems. Numerical results are given for various two-dimensional problems. The gain in computing time and memory allocation is also addressed.

As already indicated in the previous section, a real challenge for a high-order spectral element method is formed by numerical simulations in which the diffusion is dominated by the convection. Application of conventional low-order Galerkin methods to such problems often leads to spurious oscillations or 'wiggles' in the numerical solution. The high-order spectral element approximation on the other hand exhibits minimal numerical dispersion and diffusion if the order of the approximation is large enough. In chapter 4 unsteady convection-diffusion problems with dominating convection are considered. It appears that for convection-diffusion problems it is appropriate to decouple the treatment of convection and diffusion using an operator splitting technique. The diffusion part and convection part are then treated by different suitable time-integrations with different time-steps, when needed. The splitting approach provides the possibility to apply an explicit time-integration for the convective terms, resulting in an efficient solution procedure that requires the evaluation of matrix-vector products only.

A useful class of explicit time-integration methods for convection problems is given by Taylor–Galerkin schemes (Donea, 1984). These methods add in a natural way a stabilizing diffusion term to the numerical scheme as part of the time-integration. For the diffusion step standard (semi-)implicit time-integration is chosen. This system can be solved using the iterative techniques described above. The Taylor–Galerkin schemes for convection problems and the splitting scheme for convection-diffusion are extensively tested.

The operator splitting scheme for unsteady convection-diffusion problems with a dominating convective part is the basis for the approximation of the unsteady incompressible Navier–Stokes equations in chapter 5. An essential step in the solution procedure for the Navier–Stokes equations is the decoupling of the velocity and pressure components. This can be done in various ways; in former research of the 'Atherosclerosis' group a penalty function approach was used to decouple velocity and pressure. In the present study a modification to a second-order pressure-correction or projection method (Van Kan, 1986) is proposed for this purpose. In the first step the pressure is taken at the previous time level, yielding an intermediate velocity field that in general is not divergence-free. In this way the computation of the intermediate velocity field results in the solution of a convection-diffusion problem which can be solved using the operator splitting approach described above. The non-linear convective term is treated in an explicit way, thereby prohibiting a linearization. Enforcing the incompressibility constraint yields a Poisson equation for the pressure correction which can be solved using finite element preconditioning. In the next step both the velocity and pressure are 'corrected' to the new time level. In order to derive a second-order consistent scheme, special attention is also given to the choice of boundary conditions for velocity and pressure. All theoretical aspects of the algorithm are tested extensively by means of the approximation of an analytical solution to the Navier–Stokes equations. The numerical technique is further validated by simulating a buoyancy-driven cavity flow and comparing the results to a benchmark solution.

Finally, in chapter 6 conclusions are drawn and recommendations are given. Future research will mainly focus on the extension of the spectral element approximation to complex geometries in three dimensions, and consequently also the use of isoparametric elements. Important implementation aspects such as mesh generation and parallelization of the code are then of importance. As regards the applications within the project 'Atherosclerosis', spectral element simulation of flow patterns in three-dimensional bifurcation geometries can then be performed. Another aspect for future research is the extension of the spectral element method and of the solution algorithms for incompressible flow to visco-elastic flow simulations.

Chapter 2

Spectral element methods

2.1 Introduction

Spectral element methods are domain decomposition methods using high-order p-type weighted residual approximation for the solution of partial differential equations. The key feature of p-type methods is that they achieve convergence by increasing the degree of the approximation. Spectral element methods were first presented by Patera (1984); more recent overviews are given by Rønquist (1988) and by Maday and Patera (1989). Spectral element methods have the geometric flexibility of all other domain decomposition methods, such as the well-known h-type finite element method (Ciarlet, 1978; Axelsson and Barker, 1984; Girault and Raviart, 1986). The technique is very similar to the p-type finite element method proposed by Babuška *et al.* (1981).

The spectral element technique belongs to the large class of spectral methods (Gottlieb and Orszag, 1977; Gottlieb *et al.*, 1984; Canuto *et al.*, 1988; Zang *et al.*, 1989). Spectral methods are characterized by the property that for problems with a solution that is sufficiently smooth, exponential accuracy is obtained by expanding the solution in a series of special expansion functions. These expansion functions are solutions of singular Sturm–Liouville problems. Of particular interest among these solutions are the Chebyshev and the Legendre polynomials.

In the spectral element approximation the domain is divided into non-overlapping, conforming elements. The discretization process is based on a variational formulation of the partial differential equation. The main effect of the variational approach is that the continuity requirements at element boundaries are lowered. The integral equations appearing in the variational formulation are integrated by high-order Legendre Gauss–Lobatto quadrature. The variables in each element are expanded in a series of high-order polynomial basis functions. For reasons of efficiency, in more dimensions a tensorial basis is used. The discrete matrix-vector system is generated in the standard Galerkin way, leading to a block-banded system.

Due to their high-order character, spectral element methods are appropriate for problems in which high-order regularity is guaranteed or (in any case) not the exception, such as is the case for incompressible flow problems. If the regularity of the solution is low, or if the required accuracy is not very high, spectral element methods perform no better than conventional low-order methods. Since spectral element methods are domain decomposition techniques, complex geometries (even singularities) can be treated by local mesh refinement. Moreover physically originated difficulties, such as boundary layers, can be handled by locally increasing the number of elements or even the degree of the approximation (Bernardi *et al.*, 1990).

In section 2.2 the fundamentals of spectral element discretization are described. As a starting point the concept of p-type approximation methods is discussed. Next, the high accuracy of spectral-type methods is explained by analyzing the approximation of a sufficiently smooth function $c(x)$ in terms of a series of expansion functions. Finally, the spectral element discretization process is extensively discussed by means of a one-dimensional linear elliptic model equation.

Section 2.3 deals with spectral elements in more dimensions. Key features that do not occur in one dimension, such as tensor product formulation and the influence of geometry (isoparametric elements), are addressed. The discretization process in more dimensions is shown by applying spectral elements to the two-dimensional steady convection-diffusion equation. In chapter 4 of this thesis attention will be given to the approximation of unsteady convection-diffusion problems.

In section 2.4 numerical results are given. Firstly, the spectral element approximation is applied to several test cases with an analytical solution. Exponential convergence to this solution is shown. Next, the influence of the convective term in a model convection-diffusion problem and the use of upwinding in a spectral element approximation is analyzed. Finally, the problem of heat transfer in a Poiseuille flow is approximated.

2.2 Spectral element discretization in one dimension

2.2.1 p-type weighted residual methods

In this section the concept of p-convergence of a numerical approximation method is discussed from a general point of view. The following partial differential equation in one dimension is taken as a starting point

$$\mathcal{L}c = f \qquad \text{in } \Omega. \tag{2.1}$$

Here $\mathcal{L}$ is a continuous positive-definite differential operator with domain the linear vector space U, a subspace of a Hilbert space V ($\mathcal{L} : U \subset V \to V$) and $f \in C^0(\overline{\Omega})$. The corresponding weighted residual formulation is

Find $c \in U$ such that:

$$(\mathcal{L}c - f, v)_V = 0, \qquad \forall v \in V, \tag{2.2}$$

with $(\cdot,\cdot)_V$ the inner product in V. The functions $v \in V$ are called the test functions. It is a well-known result from the Lax–Milgram lemma, see e.g. Ciarlet (1978), that if the operator $\mathcal{L}$ satisfies certain conditions, problem (2.2) has a unique solution $c \in U$.

The next step is to choose an $(n+1)$-dimensional subspace $U_h \subset U$ with basis φ_i ($i = 0, \ldots, n$). The approximate solution $c_h \in U_h$ is then written as

$$c_h(x) = \sum_{i=0}^{n} c_i \varphi_i(x). \tag{2.3}$$

The coefficients c_i are the unknowns. In an analogous way the space of test functions V is discretized by an $(n+1)$-dimensional subspace V_h with basis ψ_k ($k = 0, \ldots, n$). The discrete weighted residual formulation then reads

Find $c_h \in U_h$ such that:

$$(\mathcal{L}c_h - f, v_h)_V = 0, \qquad \forall v_h \in V_h, \tag{2.4}$$

or equivalently

Find $c_h \in U_h$ such that:

$$(\mathcal{L}c_h - f, \psi_k)_V = 0, \qquad k = 0, \ldots, n. \tag{2.5}$$

Since V_h is also a Hilbert space, the Lax–Milgram lemma again can be applied yielding a unique approximate solution $c_h \in U_h$. If the bases φ_i and ψ_k are the same, i.e. $U_h = V_h$, the method is called a Galerkin method, otherwise the method is called a Petrov–Galerkin method. The Galerkin weighted residual approach is shown in Figure 2.1.

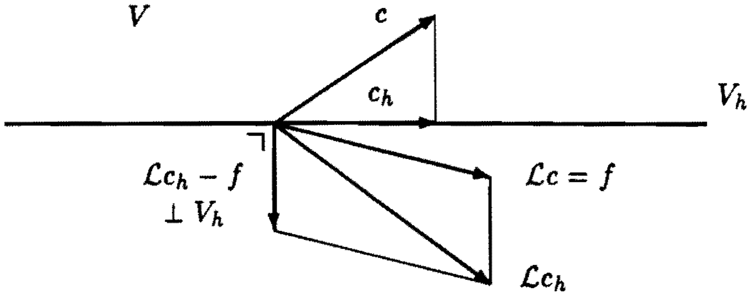


Figure 2.1: Galerkin weighted residual approximation method.

The class of p-type weighted residual techniques can be divided into global methods, such as spectral methods, and methods that use any form of domain decomposition, such as the spectral element method. The main advantage of domain decomposition techniques is that they are able to handle complex geometries. Another consequence of the use of domain decomposition is that the system matrix is block-banded, which results in less computing time and memory storage compared to global methods. Moreover, these techniques provide a very useful application for parallel computing, since a lot of operations can be performed on domain level.

If a Galerkin domain decomposition method is used, the basis for the numerical scheme is the variational or weak formulation to equation (2.1)

Find $c \in V$ such that:

$$a(c, v) = (f, v)_V, \qquad \forall v \in V, \tag{2.6}$$

where the functions $v \in V$ are again the test functions. This is done in order to lower the continuity requirement at element boundaries. If the bilinear form $a(\cdot, \cdot)$ is bounded and positive-definite, application of the Lax–Milgram states that problem (2.6) has a unique solution $c \in V$. Again approximating V by the $(n+1)$-dimensional subspace V_h with basis ψ_k $(k = 0, \ldots, n)$ gives the discrete problem

Find $c_h \in V_h$ such that:

$$a(c_h, v_h) = (f, v_h)_V, \qquad \forall v_h \in V_h, \tag{2.7}$$

or equivalently

Find $c_h \in V_h$ such that:

$$a(c_h, \psi_k) = (f, \psi_k)_V, \qquad k = 0, \dots, n. \tag{2.8}$$

An important result of the variational Galerkin formulation is the following lemma of Céa (Ciarlet, 1978) also known as the 'best approximation result'

$$\|c - c_h\|_V \leq c_0 \|c - v_h\|_V, \qquad \forall v_h \in V_h, \tag{2.9}$$

where $\|\cdot\|_V$ denotes the norm induced by the inner product $(\cdot,\cdot)_V$ and c_0 is a constant. The lemma states that the error $\|c - c_h\|_V$ is bounded by the smallest distance $\|c - v_h\|_V$. In other words, the approximation c_h is the best approximation for c in the space V_h. This result is shown in Figure 2.2.

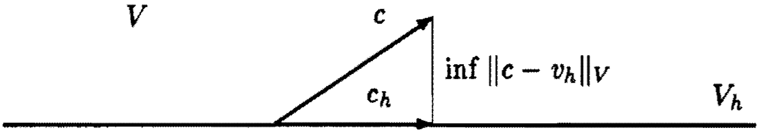


Figure 2.2: Galerkin 'best approximation result'.

The idea behind p-type methods is to obtain a more accurate solution by decreasing the distance $\inf \|c - v_h\|_V$. This is done by increasing n, the dimension of the approximate space V_h (p-convergence). Before describing the spectral element discretization, it is therefore first analyzed how accurate a function $c(x)$ can be approximated by increasing the degree of its approximation.

2.2.2 Spectral approximation

In a weighted residual approximation method the approximate solution to the partial differential equation is expanded in a truncated series of expansion functions

$$\mathcal{S}_n c(x) = \sum_{i=0}^{n} \tilde{c}_i \varphi_i(x), \tag{2.10}$$

with $\tilde{c}_i$ the expansion coefficients and φ_i the expansion functions. In p-type methods convergence is achieved by letting $n \to \infty$, that is by increasing the degree of the expansion.

The essential point behind spectral methods is that the expansion functions are chosen such that the expansion (2.10) has the property of exponential accuracy if the approximated function is sufficiently smooth. The convergence rate is therefore determined by the smoothness of that function. In order to establish this, consider the expansion of an analytical function $c(x)$ in terms of an infinite sequence of orthogonal functions φ_i

$$c(x) = \sum_{i=0}^{\infty} \tilde{c}_i \varphi_i(x). \tag{2.11}$$

If the system of orthogonal functions is complete in a suitable Hilbert space, relation (2.11) can be inverted. Thus the function $c(x)$ can be described both through its values in physical space and through its coefficients in spectral space.

In the case of periodic functions the obvious expansions are Fourier series. A well-known result of the Fourier theory states that if a function $c(x)$ is periodic and analytical with periodic derivatives, the i^{th} coefficient of its Fourier series decays faster than any inverse power of i, that is

$$\forall m > 0 \; \exists i_0 \; \forall i > i_0 : \; \tilde{c}_i < i^{-m}. \tag{2.12}$$

The exponential (or spectral) accuracy can also be obtained for expansions in non-periodic functions. If the function $c(x)$ is expanded in a series of eigenfunctions of singular Sturm–Liouville problems defined on the domain $(-1, 1)$, the i^{th} coefficient of the expansion decays faster than any inverse power of the corresponding eigenvalue of the associated Sturm–Liouville problem, that is

$$\forall m > 0 \; \exists i_0 \; \forall i > i_0 : \; \tilde{c}_i < \lambda_i^{-m}. \tag{2.13}$$

In appendix A.1 more results on Sturm–Liouville expansions are given. In particular polynomial expansions are of importance, for reasons of efficiency. In Canuto *et al.* (1988) it is shown that the only polynomial solutions to singular Sturm–Liouville problems on $(-1, 1)$ are encompassed in the class of Jacobi polynomials. The two most common applications in this class are the Chebyshev and Legendre polynomials. For a detailed survey of properties of these polynomials see Abramowitz and Stegun (1972). Since spectral element methods use Legendre-type expansions, some properties of Legendre polynomials and Legendre expansions are given in appendix B.

It is clear that spectral approximation can be seen as a special case of the expansion of a function $c(x)$ in terms of an infinite sequence of general orthogonal polynomials. The theory hereof is provided in appendix A.2.

2.2.3 Pseudospectral approximation

The approximation (2.11) defines a continuous transform between physical space and spectral space. In general, the spectral expansion coefficients $\tilde{c}_i$ are not computed exactly, since they depend on all values of the function in physical space. In pseudospectral methods a set of approximate coefficients $\hat{c}_i$ is calculated using the values of $c(x)$ at a finite number of interpolation points. The finite series defined by the discrete coefficients $\hat{c}_i$ is then the interpolating polynomial of $c(x)$ at the interpolation points. It is given by

$$\mathcal{I}_n c(x) = \sum_{i=0}^{n} \hat{c}_i \varphi_i(x). \tag{2.14}$$

The interpolating polynomial satisfies

$$\mathcal{I}_n c(x_k) = c(x_k), \qquad k = 0, \ldots, n, \tag{2.15}$$

where the x_k are the interpolation points. Equation (2.15) gives the discrete transform between the values of $c(x)$ and the set of its discrete coefficients. For more details on interpolating polynomials see appendix A.3, in which the more general case of discrete expansions in terms of orthogonal polynomials is described.

It will be convenient if the spectral accuracy is retained in replacing the continuous transform with the discrete transform. In Canuto *et al.* (1988) it is shown that if the interpolation points are Gauss-type quadrature points, the discrete expansion shows spectral accuracy. An overview of Gauss-type quadrature is given by Davis and Rabinowitz (1984). In practice the interpolating polynomial is often written as

$$\mathcal{I}_n c(x) = \sum_{i=0}^{n} c_i \phi_i(x), \tag{2.16}$$

with $c_i = c(x_i)$ and ϕ_i the Lagrangian interpolant in the Gauss-type quadrature points. For Legendre interpolation explicit values for ϕ_i and its derivatives in the collocation points are given in appendix B. For more theory on pseudospectral methods see e.g. Gottlieb *et al.* (1984) and Hussaini *et al.* (1989).

2.2.4 A one-dimensional model equation

In order to establish the concepts of spectral elements, consider the following linear symmetric elliptic boundary value problem in one dimension

$$\begin{cases} -\dfrac{\partial}{\partial x}\left(a(x)\dfrac{\partial c}{\partial x}\right) + b(x)c = f(x) \quad \text{in } \Omega = (x_l, x_r), \\ c(x_l) = c(x_r) = 0. \end{cases} \tag{2.17}$$

Define a_0, a_1 and b_1 such that $a(x), b(x) \in C^0(\overline{\Omega})$ satisfy

$$0 < a_0 < a(x) < a_1, \qquad 0 < b(x) < b_1, \tag{2.18}$$

where $C^0(\overline{\Omega})$ is the space of continuous functions in $\overline{\Omega}$. Also, $f(x) \in L^2(\Omega)$ with the Lebesgue space $L^2(\Omega)$ defined by

$$L^2(\Omega) = \left\{ v \;\middle|\; \int_\Omega v^2 \, dx < \infty \right\}. \tag{2.19}$$

The natural inner product in this space is defined as

$$(v, w) = \int_\Omega v(x)w(x)\, dx, \qquad \forall\, v, w \in L^2(\Omega), \tag{2.20}$$

with induced norm $\|v\|^2 = (v, v)$. The basis for the numerical scheme is the variational equivalent to problem (2.17). The space of acceptable solutions is defined to be the Sobolev space $H_0^1(\Omega)$ given by

$$H_0^1(\Omega) = \left\{ v \in L^2(\Omega) \;\middle|\; \frac{\partial v}{\partial x} \in L^2(\Omega),\ v(x_r) = v(x_l) = 0 \right\}. \tag{2.21}$$

The variational equivalent to problem (2.17) can then be written as

Find $c(x) \in H_0^1(\Omega)$ such that:

$$a(c, v) = (f, v), \qquad \forall\, v \in H_0^1(\Omega), \tag{2.22}$$

where the bilinear continuous form $a(\cdot,\cdot)$ is defined by

$$a(c, v) = \int_\Omega a \frac{\partial c}{\partial x}\frac{\partial v}{\partial x}\, dx + \int_\Omega bc(x)v(x)\, dx, \quad c, v \in H_0^1(\Omega). \tag{2.23}$$

It is easily verified that the bilinear form $a(\cdot,\cdot)$ is bounded and positive-definite. Application of the Lax–Milgram lemma therefore yields that problem (2.22) has a unique solution $c(x) \in H_0^1(\Omega)$.

The first step in the spectral element discretization process is to break up the domain Ω into n_e non-overlapping elements Ω_e

$$\overline{\Omega} = \bigcup_{e=1}^{n_e} \Omega_e. \tag{2.24}$$

The next step is to discretize the space of acceptable solutions $H_0^1(\Omega)$. The space of approximation for the solution $c(x)$ is taken to be the $(n+1)$-dimensional subspace V_h of $H_0^1(\Omega)$ defined as

$$V_h = \left\{ v \in H_0^1(\Omega) \middle| v_{|\Omega_e} \in P_n(\Omega_e) \right\}, \tag{2.25}$$

where $P_n(\Omega_e)$ denotes the space of polynomials in Ω_e of degree $\leq n$. The discrete variational problem can now be written as

Find $c_h \in V_h$ *such that:*

$$\sum_{e=1}^{n_e} a(c_h, v_h)_{\Omega_e} = \sum_{e=1}^{n_e} (f, v_h)_{\Omega_e}, \qquad \forall\, v_h \in V_h, \tag{2.26}$$

where the subscript Ω_e denotes the restriction to element Ω_e. In general equation (2.26) can not be implemented without numerical quadrature. The choice of numerical quadrature corresponds to the choice of a discrete inner product. The high-order discretization suggests a Gauss-type quadrature formula. A Legendre-type quadrature is chosen since it has a natural weight function equal to 1. For reasons to be explained at the end of this section Gauss–Lobatto integration is chosen. This quadrature is defined by

$$\int_{-1}^{1} p(y)\, dy = \sum_{k=0}^{n} p(\xi_k)\gamma_k, \qquad \forall\, p \in P_{2n-1}(-1,1), \tag{2.27}$$

where the ξ_k are the Legendre Gauss–Lobatto points and the γ_k the Legendre Gauss–Lobatto weights. In order to be able to apply the quadrature, an affine transformation is used to map each element Ω_e to the standard element $e = [-1,1]$ ($x \in \Omega_e \rightarrow y \in e$). Due to this transformation the terms in equation (2.26) can be written as

$$a(c_h, v_h)_{\Omega_e} = \int_e a\, \frac{\partial c_h}{\partial y} \frac{\partial v_h}{\partial y}\, J^{-1}\, dy + \int_e b\; c_h(y) v_h(y)\; J\; dy, \tag{2.28}$$

$$(f, v_h)_{\Omega_e} = \int_e f\; v_h(y)\; J\; dy. \tag{2.29}$$

The transformation Jacobian J is given by

$$J = \frac{dx}{dy} \qquad (x \in \Omega_e \to y \in e). \tag{2.30}$$

Applying the Legendre Gauss–Lobatto quadrature to the system (2.26) yields the following fully discrete problem

Find $c_h \in V_h$ such that:

$$\sum_{e=1}^{n_e} a(c_h, v_h)_{gl} = \sum_{e=1}^{n_e} (f, v_h)_{gl}, \qquad \forall\ v_h \in V_h. \tag{2.31}$$

The corresponding discrete inner product $(\cdot,\cdot)_{gl}$ with induced norm $\|\cdot\|_{gl}$ is given by

$$(v, w)_{gl} = \sum_{k=0}^{n} J_k\ v(\xi_k) w(\xi_k)\ \gamma_k, \qquad \forall\ v, w \in C^0(e), \tag{2.32}$$

where $J_k = J(\xi_k)$. Furthermore, the discrete bilinear form $a(\cdot,\cdot)_{gl}$ is given by

$$\begin{aligned} a(c,v)_{gl} &= \sum_{k=0}^{n} \frac{1}{J_k}\ a(x_k) \frac{\partial c(\xi_k)}{\partial y} \frac{\partial v(\xi_k)}{\partial y}\ \gamma_k \\ &+ \sum_{k=0}^{n} J_k\ b(x_k) c(\xi_k) v(\xi_k)\ \gamma_k, \quad \forall\ c, v \in C^1(e), \end{aligned} \tag{2.33}$$

where the x_k are the global points corresponding to the local Gauss–Lobatto points ξ_k. Equation (2.31) can be seen as the complete spectral element discretization of the original differential equation (2.17). Again, application of the Lax–Milgram lemma shows that problem (2.31) has a unique solution $c_h \in V_h$.

In order to implement the discrete system (2.31) it is necessary to choose a basis for the approximation space V_h. The choice of basis affects the form and conditioning of the discrete equations. Moreover, it is important with respect to inter-elemental coupling. Therefore a Legendre Gauss–Lobatto Lagrangian interpolant basis is chosen to represent a function in V_h, since the Lobatto points include the boundary points of the reference element $e = [-1, 1]$. The basis functions $\phi_i(y)$ $(i = 0, \ldots, n)$ satisfy

$$\begin{cases} \phi_i(y) \in P_n(e), & i = 0, \ldots, n, \\ \phi_i(\xi_j) = \delta_{ij}, & i, j = 0, \ldots, n. \end{cases} \tag{2.34}$$

An explicit expression for the basis functions is given by

$$\phi_i(y) = -\frac{1-y^2}{n(n+1)L_n(\xi_i)(y-\xi_i)}\frac{\partial L_n}{\partial y}, \tag{2.35}$$

with $L_n(y)$ the Legendre polynomial of degree n. Plots of the basis functions for several values of n are shown in appendix B. The approximate solution c_h in each element Ω_e can be written as

$$c_h(x) = \sum_{i=0}^{n} c_i \phi_i(y) \qquad (x \in \Omega_e \ \rightarrow \ y \in e), \tag{2.36}$$

where $c_i = c_h(x_i)$. The approximate solution is continuous over element boundaries since $c_h \in H_0^1(\Omega)$. As a consequence the coefficients satisfy

$$c_{n_{|\Omega_{e-1}}} = c_{0_{|\Omega_e}}, \qquad e = 2, \ldots, n_e. \tag{2.37}$$

Moreover, c_h must satisfy the Dirichlet boundary conditions

$$c_{0_{|\Omega_1}} = c_{n_{|\Omega_{n_e}}} = 0. \tag{2.38}$$

It is convenient to represent the source term $f(x)$ in terms of the basis in V_h. Substitution of c_h, f_h and the test function v_h into equation (2.31) in the standard Galerkin way yields the discrete (block-banded) matrix system

$$(\mathbf{D} + \mathbf{H})\mathbf{c} = \mathbf{M}\mathbf{f}, \tag{2.39}$$

or equivalently

$$\sum_{e=1}^{n_e}{}' \sum_{i=0}^{n} \left(D_{ji}^e + H_{ji}^e\right) c_i = \sum_{e=1}^{n_e}{}' \sum_{i=0}^{n} M_{ji}^e f_i, \qquad j = 0, \ldots, n. \tag{2.40}$$

In equation (2.40) $\sum{}'$ denotes direct stiffness summation, i.e. the contributions of corresponding global element boundary points are summed to ensure condition (2.37) and the Dirichlet boundary conditions (2.38) are taken into account by eliminating from the system the rows and columns corresponding to the boundary points (matrix condensation). The elemental matrices are given by

$$\left\{ \begin{array}{lll} D_{ji}^e = \displaystyle\sum_{k=0}^{n} \frac{1}{J_k} P_{ki}^{(1)} P_{kj}^{(1)} a(x_k)\gamma_k, & i,j = 0, \ldots, n, \\ H_{ji}^e = J_i b(x_i)\gamma_i \delta_{ij}, & i,j = 0, \ldots, n, \\ M_{ji}^e = J_i \gamma_i \delta_{ij}, & i,j = 0, \ldots, n, \end{array} \right. \tag{2.41}$$

with

$$P^{(1)}_{ki} = \frac{\partial \phi_i(\xi_k)}{\partial y}, \qquad k, i = 0, \ldots, n. \tag{2.42}$$

Explicit values for the first derivative matrix $\mathbf{P}^{(1)}$ are given in appendix B. A detailed error analysis for linear elliptic problems is given by Maday and Patera (1989). Here it suffices to say that if the solution and data are analytical, exponential convergence to the exact solution $c(x)$ is obtained for c_h if $n \to \infty$.

To conclude this section a remark can be made about the matrix system. As already mentioned, elements only couple at boundary nodes because of the Gauss–Lobatto interpolation. The choice of Gauss–Lobatto numerical integration ensures an exact evaluation of the stiffness or diffusion matrix **D** in the case of a constant coefficient $a(x)$. The matrix **H** and the mass matrix **M** are not computed exactly. However, the use of Gauss–Lobatto integration has two advantages. Firstly, it enables an efficient use of tensor product basis functions in more dimensions, see also section 2.3.1. Moreover, it results in a diagonal mass matrix **M**, a property that is very useful when approximating time-dependent problems with an explicit time-integration, see chapter 4.

2.3 Spectral elements in more dimensions

2.3.1 Tensor product formulation

The efficiency of high-order methods such as the spectral element method in more dimensions depends on the use of a tensorial basis for the approximate space of solution. To show this consider the discretization of a three-dimensional problem. The approximate solution c_h is written as a tensor product of the one-dimensional basis functions ϕ_i $(i = 0, \ldots, n)$

$$c_h = \sum_{p=0}^{n} \sum_{q=0}^{n} \sum_{r=0}^{n} c_{pqr} \phi_p \phi_q \phi_r. \tag{2.43}$$

A discrete matrix-vector system resulting from the spectral element discretization of a three-dimensional problem has the following form

$$\mathbf{S}\mathbf{u} = \mathbf{M}\mathbf{f}, \tag{2.44}$$

where **S** is the system matrix consisting of e.g. a stiffness, convection or Helmholtz part and **M** the mass matrix. Extending the results of the one-dimensional case from section 2.2.4, this can also be written as

$$\sum_{e=1}^{n_e}{}' \sum_{p=0}^{n} \sum_{q=0}^{n} \sum_{r=0}^{n} S^e_{stvpqr} c_{pqr} = \sum_{e=1}^{n_e}{}' \sum_{p=0}^{n} \sum_{q=0}^{n} \sum_{r=0}^{n} M^e_{stvpqr} f_{pqr}, \qquad s,t,v = 0,\ldots,n. \tag{2.45}$$

At first sight it seems that evaluation of the elemental sum

$$\sum_{p=0}^{n} \sum_{q=0}^{n} \sum_{r=0}^{n} S^e_{stvpqr} c_{pqr}, \qquad s,t,v = 0,\ldots,n, \tag{2.46}$$

requires $\mathcal{O}(n^6)$ operations, since one must sum over $p,q,r = 0,\ldots,n$ for $s,t,v = 0,\ldots,n$. Moreover, since S is full, in general the operation requires $\mathcal{O}(n^6)$ storage. This does not look promising in comparison to a low-order h-type method for which the equivalent operation count and storage both scale as $\mathcal{O}(n^3)$.

The use of the tensor product formulation (2.43) however, enables an efficient evaluation of the elemental sum (2.46). This is due to the fact that the one-dimensional basis functions satisfy

$$\phi_i(\xi_k) = \delta_{ik}, \qquad i,k = 0,\ldots,n, \tag{2.47}$$

where the ξ_k are the Legendre Gauss–Lobatto collocation points. Application of property (2.47) in the evaluation of elemental sums gives the possibility to 'skip' a number of summations. In the case that **S** consists of a stiffness part, the operation count can be reduced to $\mathcal{O}(n^4)$ and the storage requirement can be reduced to $\mathcal{O}(n^3)$. For other cases this reduction is even larger; e.g. the spectral element mass matrix is always a diagonal matrix. In the general case, that is if there is a stiffness contribution, the reduction in operation count in d dimensions is found to be $\mathcal{O}(n^{d-1})$.

Finally, it must be noted that the above reasoning only holds, if the integration and interpolation coincide. Since spectral elements use Legendre Gauss–Lobatto basis functions, also Legendre Gauss–Lobatto numerical integration must be applied. Otherwise the basis functions are not evaluated in the Legendre Gauss–Lobatto points and property (2.47) can not be used.

2.3.2 The steady convection-diffusion equation

As an example of the spectral element method in more dimensions, in this section the discretization of the two-dimensional convection-diffusion equation is considered. Results for the three-dimensional case can be deduced in an analogous way. The spectral element discretization process in two dimensions is essentially a straightforward extension of the one-dimensional case. However, several computational details such as tensor product formulation and geometry (aspects that do not appear in the one-dimensional discretization process) need to be addressed.

Consider the two-dimensional steady convection-diffusion equation in a bounded region $\boldsymbol{\Omega}$ with boundary $\boldsymbol{\Gamma}$

$$-(\nabla \cdot \eta \nabla)c + (\mathbf{u} \cdot \nabla)c = f \qquad \text{in } \boldsymbol{\Omega}, \tag{2.48}$$

with homogeneous Dirichlet boundary conditions. Here η is the dynamic viscosity (which is assumed to be constant for simplicity) and $\mathbf{u}$ is the velocity vector. The variational formulation for this differential equation is

Find $c(\mathbf{x}) \in H_0^1(\boldsymbol{\Omega})$ *such that:*

$$a(c, v) = (f, v), \qquad \forall\, v \in H_0^1(\boldsymbol{\Omega}), \tag{2.49}$$

where the bilinear form $a(\cdot, \cdot)$ is given by

$$a(c, v) = \int\limits_{\boldsymbol{\Omega}} \eta \nabla c \cdot \nabla v \; d\mathbf{x} + \int\limits_{\boldsymbol{\Omega}} (\mathbf{u} \cdot \nabla) c \; v \; d\mathbf{x}, \quad c, v \in H_0^1(\boldsymbol{\Omega}). \tag{2.50}$$

The domain $\boldsymbol{\Omega}$ is divided into (generally) deformed disjoint quadrilateral elements with the intersections either a whole edge or a vertex

$$\overline{\boldsymbol{\Omega}} = \bigcup_{e=1}^{n_e} \boldsymbol{\Omega}_e. \tag{2.51}$$

As in the one-dimensional case we approximate the solution space $H_0^1(\boldsymbol{\Omega})$ by the subspace V_h defined as

$$V_h = \left\{ v \in H_0^1(\boldsymbol{\Omega}) \,\middle|\, v_{|\boldsymbol{\Omega}_e} \in P_n(\boldsymbol{\Omega}_e) \right\}, \tag{2.52}$$

where $P_n(\boldsymbol{\Omega}_e)$ is the tensor product space of polynomials of degree $\leq n$ in $\boldsymbol{\Omega}_e$ with respect to each spatial variable x_1 and x_2. Equation (2.49) can then be written as

$$\sum_{e=1}^{n_e} a(c_h, v_h)_{\boldsymbol{\Omega}_e} = \sum_{e=1}^{n_e} (f, v_h)_{\boldsymbol{\Omega}_e}, \qquad \forall\, v_h \in V_h, \tag{2.53}$$

where

$$a(c_h, v_h)_{\boldsymbol{\Omega}_e} = \int\limits_{\boldsymbol{\Omega}_e} \eta \nabla c_h \cdot \nabla v_h \, d\mathbf{x} + \int\limits_{\boldsymbol{\Omega}_e} (\mathbf{u} \cdot \nabla) c_h v_h \, d\mathbf{x}, \tag{2.54}$$

$$(f, v_h)_{\boldsymbol{\Omega}_e} = \int\limits_{\boldsymbol{\Omega}_e} f \; v_h \, d\mathbf{x}. \tag{2.55}$$

Each quadrilateral element $\boldsymbol{\Omega}_e$ is mapped via an isoparametric transformation, see also Figure 2.3, onto the standard element in two dimensions $\mathbf{e} = [-1,1] \times [-1,1]$ $(\mathbf{x} = (x_1, x_2) \in \boldsymbol{\Omega}_e \rightarrow \mathbf{y} = (y_1, y_2) \in \mathbf{e})$.

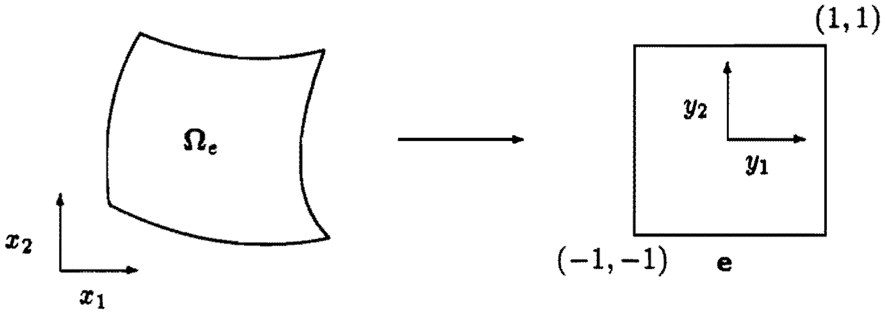


Figure 2.3: Isoparametric transformation from $\boldsymbol{\Omega}_e$ to the standard element **e** in two dimensions.

Due to this transformation equation (2.54) can be rewritten as

$$\begin{aligned} a(c_h, v_h)_{\boldsymbol{\Omega}_e} &= \int\limits_{\mathbf{e}} \eta (\mathbf{J}^{-1} \nabla_{\mathbf{y}} c_h \cdot \mathbf{J}^{-1} \nabla_{\mathbf{y}} v_h) \, |\mathbf{J}| \, d\mathbf{y} \\ &+ \int\limits_{\mathbf{e}} (\mathbf{u} \cdot \mathbf{J}^{-1} \nabla_{\mathbf{y}}) c_h v_h \, |\mathbf{J}| \, d\mathbf{y}. \end{aligned} \tag{2.56}$$

The transformation Jacobian **J** is given by

$$\mathbf{J} = \begin{pmatrix} \dfrac{\partial x_1}{\partial y_1} & \dfrac{\partial x_2}{\partial y_1} \\ \dfrac{\partial x_1}{\partial y_2} & \dfrac{\partial x_2}{\partial y_2} \end{pmatrix}. \tag{2.57}$$

Here $|\mathbf{J}|$ is the determinant of $\mathbf{J}$. With the definition

$$\tilde{\nabla} = \begin{pmatrix} \dfrac{\partial x_2}{\partial y_2}\dfrac{\partial}{\partial y_1} - \dfrac{\partial x_2}{\partial y_1}\dfrac{\partial}{\partial y_2} \\ -\dfrac{\partial x_1}{\partial y_2}\dfrac{\partial}{\partial y_1} + \dfrac{\partial x_1}{\partial y_1}\dfrac{\partial}{\partial y_2} \end{pmatrix}, \tag{2.58}$$

the bilinear form $a(\cdot,\cdot)_{\mathbf{\Omega}_e}$ corresponding to the standard element $\mathbf{e}$ is

$$a(c_h, v_h)_{\mathbf{\Omega}_e} = \int\limits_{\mathbf{e}} \eta \frac{1}{|\mathbf{J}|} \tilde{\nabla} c_h \cdot \tilde{\nabla} v_h \; d\mathbf{y} \; + \int\limits_{\mathbf{e}} (\mathbf{u} \cdot \tilde{\nabla}) c_h v_h \; d\mathbf{y}. \tag{2.59}$$

In an analogous way can be deduced

$$(f, v_h)_{\mathbf{\Omega}_e} = \int\limits_{\mathbf{e}} f \; v_h \; |\mathbf{J}| \; d\mathbf{y}. \tag{2.60}$$

The integral equations can be numerically integrated by applying Legendre Gauss–Lobatto quadrature. This yields

$$\sum_{e=1}^{n_e} a(c_h, v_h)_{gl} = \sum_{e=1}^{n_e} (f, v_h)_{gl}, \tag{2.61}$$

where

$$\begin{aligned} a(c_h, v_h)_{gl} \; &= \; \sum_{k=0}^{n}\sum_{l=0}^{n} \eta \gamma_k \gamma_l \frac{1}{|\mathbf{J}|_{kl}} \tilde{\nabla} c_h(\xi_k, \xi_l) \cdot \tilde{\nabla} v_h(\xi_k, \xi_l) \\ &+ \; \sum_{k=0}^{n}\sum_{l=0}^{n} \gamma_k \gamma_l (\mathbf{u} \cdot \tilde{\nabla}) c_h(\xi_k, \xi_l) v_h(\xi_k, \xi_l), \qquad (2.62) \\ (f, v_h)_{gl} \; &= \; \sum_{k=0}^{n}\sum_{l=0}^{n} \gamma_k \gamma_l \; |\mathbf{J}|_{kl} \; f \; v_h(\xi_k, \xi_l). \qquad (2.63) \end{aligned}$$

The ξ_k and γ_k are the Legendre Gauss–Lobatto points and weights respectively.

The approximate solution $c_h \in V_h$ in each element $\mathbf{\Omega}_e$ is expressed in a tensor product of basis functions

$$c_h(\mathbf{x}) = \sum_{p=0}^{n}\sum_{q=0}^{n} c_{pq} \phi_p(y_1) \phi_q(y_2) \qquad (\mathbf{x} \in \mathbf{\Omega}_e \; \rightarrow \; \mathbf{y} \in \mathbf{e}). \tag{2.64}$$

Substitution in equation (2.61) of c_h, v_h and f_h in the usual way yields the discrete matrix system

$$(\mathbf{D} + \mathbf{C})\mathbf{c} = \mathbf{M}\mathbf{f}, \tag{2.65}$$

or equivalently

$$\sum_{e=1}^{n_e}{}' \sum_{p=0}^{n} \sum_{q=0}^{n} \left(D^e_{rspq} + C^e_{rspq} \right) c_{pq} = \sum_{e=1}^{n_e}{}' \sum_{p=0}^{n} \sum_{q=0}^{n} M^e_{rspq} f_{pq}, \quad r, s = 0, \ldots, n. \tag{2.66}$$

The elemental matrices are given by

$$\begin{cases} D^e_{rspq} = \eta \sum_{k=0}^{n} \sum_{l=0}^{n} \gamma_k \gamma_l \frac{1}{|\mathbf{J}|_{kl}} \tilde{\nabla}_{klpq} \cdot \tilde{\nabla}_{klrs}, \\ C^e_{rspq} = \gamma_r \gamma_s \; \mathbf{u}_{rs} \cdot \tilde{\nabla}_{rspq}, \\ M^e_{rspq} = \gamma_p \gamma_q \; |\mathbf{J}|_{pq} \; \delta_{pr} \delta_{qs}. \end{cases} \tag{2.67}$$

In equation (2.67) the discrete gradient is given by

$$\tilde{\nabla}_{klpq} = \begin{pmatrix} \left(\frac{\partial x_2}{\partial y_2}\right)_{kl} P^{(1)}_{kp} \delta_{lq} - \left(\frac{\partial x_2}{\partial y_1}\right)_{kl} P^{(1)}_{lq} \delta_{kp} \\ \left(\frac{\partial x_1}{\partial y_1}\right)_{kl} P^{(1)}_{lq} \delta_{kp} - \left(\frac{\partial x_1}{\partial y_2}\right)_{kl} P^{(1)}_{kp} \delta_{lq} \end{pmatrix}, \tag{2.68}$$

with the first derivative matrix $\mathbf{P}^{(1)}$ given by equation (2.42) and the partial derivatives given by

$$\left(\frac{\partial x_1}{\partial y_1}\right)_{kl} = \sum_{p=0}^{n} x_{1,pl} P^{(1)}_{kp}, \quad \left(\frac{\partial x_1}{\partial y_2}\right)_{kl} = \sum_{q=0}^{n} x_{1,kq} P^{(1)}_{lq},$$

$$\left(\frac{\partial x_2}{\partial y_1}\right)_{kl} = \sum_{p=0}^{n} x_{2,pl} P^{(1)}_{kp}, \quad \left(\frac{\partial x_2}{\partial y_2}\right)_{kl} = \sum_{q=0}^{n} x_{2,kq} P^{(1)}_{lq}. \tag{2.69}$$

Note that due to the use of tensor products the equations can be evaluated in an efficient way. For the rectilinear case detailed theoretical error estimates can be found in Maday and Patera (1989), again indicating exponential convergence for analytical solution and data. In the case of deformed geometry quadrature errors occur in the evaluation of equations (2.62) and (2.63). However, for smooth solutions they are of the same order as the approximation error. For a general analysis of spectral approximation in the case of deformed geometry see Maday and Rønquist (1990). Another remark on the more-dimensional case concerns the effect of singularities arising from a non-smooth geometry. In a spectral element method this effect will be localized if the number of elements is greater than one. Moreover, it should be noted that by a suitable refining procedure even in the presence of singularities exponential convergence can be achieved, see e.g. Mavriplis (1989).

2.4 Numerical results

2.4.1 Application to problems with analytical solution

The key point of spectral-type methods is that they achieve exponential accuracy for problems with a sufficiently smooth solution. To demonstrate numerically the rapid convergence, several test cases are considered of the partial differential equations described in the previous sections.

The first test case is the one-dimensional linear elliptic problem, described by equation (2.17), on the domain $\Omega = (0,1)$ with homogeneous Dirichlet boundary conditions. The coefficients are given by

$$a(x) = e^{\pi x},\ b(x) = -\pi^2 e^{\pi x},\ f(x) = -\pi^2 e^{\pi x}\cos(\pi x). \tag{2.70}$$

The exact solution to this problem is

$$c(x) = \sin(\pi x). \tag{2.71}$$

Figure 2.4 shows the discrete maximum error $\varepsilon = \|c - c_h\|_{\infty,gl}$ for a spectral element approximation with $n_e = 2$, n varying, and for a finite element approximation with n_e varying, $n = 2$. The subscript ∞, gl means that the maximum error is evaluated in the Gauss–Lobatto points. The spectral element approximation clearly shows exponential convergence; only algebraic convergence is achieved for the finite element approximation.

The second test case is the two-dimensional convection-diffusion equation, described by equation (2.48), on the domain $\Omega = (0,1) \times (0,1)$ with Dirichlet boundary conditions. The velocity $\mathbf{u}$ is taken to be $[-1,0]^T$, the source term $f(\mathbf{x})$ is equal to zero. Two values of the viscosity are considered, $\eta = 1$ and the strongly convective case $\eta = \frac{1}{30}$. The analytical solution to this problem represents a two-dimensional boundary layer, see Strikwerda (1980). It is given by

$$c(\mathbf{x}) = \left(x_2(1-x_2) - 2\eta x_1\right) e^{-\frac{x_1}{\eta}}. \tag{2.72}$$

The analytical solution is shown in Figure 2.5 for $\eta = 1$ (left) and $\eta = \frac{1}{30}$ (right).

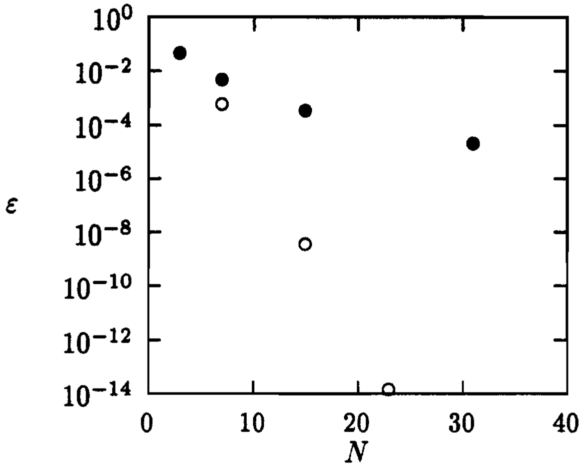


Figure 2.4: One-dimensional linear symmetric elliptic problem. Evolution of the discrete maximum error ε for spectral elements (○) with $n_e = 2, n$ varying, and finite elements (●) with n_e varying, $n = 2$. N is the number of degrees of freedom.

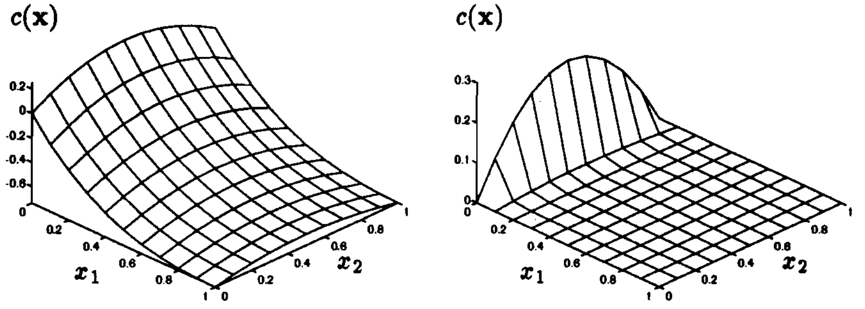


Figure 2.5: Two-dimensional boundary layer, $\eta = 1$ (left), $\eta = \frac{1}{30}$ (right).

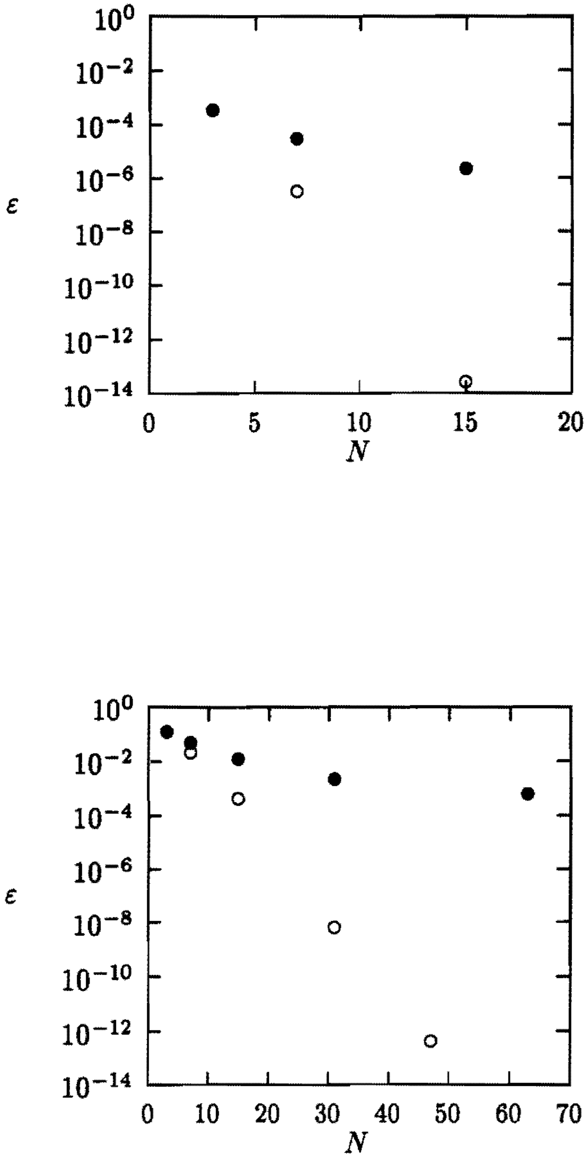


Figure 2.6: Two-dimensional convection-diffusion problem, $\eta = 1$ (top), $\eta = \frac{1}{30}$ (bottom). Evolution of the discrete maximum error ε for spectral elements (○) with $n_e = 4, n$ varying, and finite elements (●) with n_e varying, $n = 2$. N is the number of degrees of freedom in one direction.

Figure 2.6 (top) shows the discrete maximum error for the first case ($\eta = 1$) for a spectral element approximation with $n_e = 4, n$ varying, and for a finite element approximation with n_e varying, $n = 2$. Results for the second case ($\eta = \frac{1}{30}$) are shown in Figure 2.6 (bottom). In both cases exponential convergence is achieved by the spectral element approximation. For the physically more difficult case $\eta = \frac{1}{30}$ however, the degree of approximation in each element is much higher.

2.4.2 The use of SUPG in spectral elements

A very interesting aspect of the approximation of convection-diffusion problems is that the numerical solution is often rather difficult, especially if the convective term dominates the problem. Due to the fact that the resulting system matrix is no longer diagonally dominant because of the non-symmetric convective term, application of standard Galerkin low-order methods usually results in poor solutions that suffer from numerical oscillations or 'wiggles'. A possible strategy to improve these solutions is the use of some kind of upwinding. In Brooks and Hughes (1982) a Streamline Upwind/Petrov–Galerkin (SUPG) technique for finite element schemes is described. The basic idea of the Streamline Upwind method is to add stabilizing diffusion to the numerical scheme which acts only in the direction of the velocity. Extended to a Petrov–Galerkin formulation the standard Galerkin test functions are modified by adding a Streamline Upwind perturbation which again acts only in the flow direction. The modified test function is applied to all terms in the equation resulting in a consistent weighted residual formulation.

Consider again the steady convection-diffusion equation in two dimensions given by equation (2.48). The SUPG method uses discontinuous test functions of the form

$$\tilde{v} = v + \mu(\mathbf{u} \cdot \nabla)v, \tag{2.73}$$

where $v \in V$ is the Galerkin test function and $\mu(\mathbf{u} \cdot \nabla)v$ the Streamline Upwind contribution. The weak formulation of (2.48) is then given by

$$a(c, v) + (\mu(\mathbf{u} \cdot \nabla)v, -(\nabla \cdot \eta\nabla)c + (\mathbf{u} \cdot \nabla)c - f) = (f, v), \tag{2.74}$$

with the bilinear form $a(\cdot, \cdot)$ given by equation (2.50). For finite elements μ is defined as

$$\mu = \frac{h\vartheta}{\|\mathbf{u}\|}. \tag{2.75}$$

For linear elements h is the width of the element in the direction of the flow, for quadratic elements h is the the width divided by 2. The upwind parameter ϑ can be chosen to give optimal values, in some sense, at least for one-dimensional problems, see Christie *et al.* (1976), Brooks and Hughes (1982). A clear disadvantage of the SUPG scheme is that the expensive second-order term must be evaluated.

One of the advantages of the use of a high-order Galerkin method such as the spectral element method, is said to be the minimizing of the numerical oscillations that occur in convection dominated problems (Rønquist, 1988). The application of a high degree of approximation weakens the necessity to use some kind of upwinding in order to obtain good results for convection dominated problems. In this section the use of SUPG in a spectral element approximation is briefly addressed by means of a one-dimensional convection-diffusion problem. In a spectral element method the nodal points are of course not equidistant. However, similar to the finite element case the upwind coefficient μ for a spectral element method that uses a degree of approximation n, can be formulated as

$$\mu = \frac{h\vartheta}{n\|\mathbf{u}\|}, \tag{2.76}$$

with h the width of the element.

Consider the convection-diffusion equation in one dimension with varying viscosity η, velocity $u = 1$ and source term

$$\begin{aligned} f(x) &= \frac{\eta}{\sigma^2}\left(1 - \frac{(x-x_0)^2}{\sigma^2}\right) e^{-\dfrac{(x-x_0)^2}{2\sigma^2}} \\ &- \frac{(x-x_0)}{\sigma^2} e^{-\dfrac{(x-x_0)^2}{2\sigma^2}} . \end{aligned} \tag{2.77}$$

The boundary conditions are of homogeneous Dirichlet type. The exact solution $c(x)$ is a Gaussian hill given by

$$c(x) = e^{-\dfrac{(x-x_0)^2}{2\sigma^2}}, \tag{2.78}$$

centered around $x_0 = 0.4$ with standard deviation $\sigma = 0.04$.

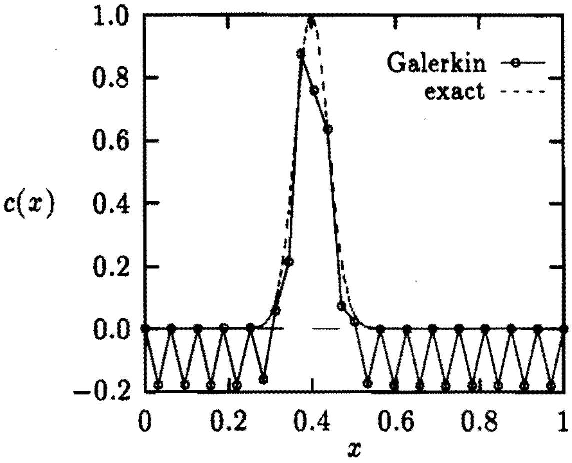


Figure 2.7: One-dimensional convection-diffusion problem. Gaussian hill. Galerkin approximation for $\eta = 10^{-5}$ using $n_e = 16$ elements with degree of approximation $n = 2$.

Table 2.1: Discrete maximum error ε for the Gaussian hill problem for decreasing viscosity η using $n_e = 16$ elements and varying degree of approximation. The use of SUPG is indicated by a + sign.

η		$n = 2$	$n = 4$	$n = 8$	$n = 16$
10^{-3}	−	$0.55 \cdot 10^{-1}$	$0.16 \cdot 10^{-2}$	$0.68 \cdot 10^{-6}$	$0.67 \cdot 10^{-14}$
	+	$0.52 \cdot 10^{-1}$	$0.12 \cdot 10^{-1}$	$0.95 \cdot 10^{-1}$	$0.92 \cdot 10^{-12}$
10^{-5}	−	$0.22 \cdot 10^{0}$	$0.76 \cdot 10^{-2}$	$0.70 \cdot 10^{-5}$	$0.12 \cdot 10^{-10}$
	+	$0.50 \cdot 10^{-1}$	$0.10 \cdot 10^{-2}$	$0.15 \cdot 10^{-3}$	$0.14 \cdot 10^{-3}$
10^{-7}	−	unstable	$0.71 \cdot 10^{0}$	$0.62 \cdot 10^{-3}$	$0.17 \cdot 10^{-6}$
	+	$0.50 \cdot 10^{-1}$	$0.92 \cdot 10^{-3}$	$0.14 \cdot 10^{-5}$	$0.15 \cdot 10^{-5}$
10^{-9}	−	unstable	unstable	$0.63 \cdot 10^{-2}$	$0.15 \cdot 10^{-6}$
	+	$0.50 \cdot 10^{-1}$	$0.91 \cdot 10^{-3}$	$0.30 \cdot 10^{-6}$	$0.15 \cdot 10^{-7}$

In Table 2.1 results are given of several spectral element approximations using $n_e = 16$ elements for decreasing viscosity. The results confirm that if the viscosity decreases, yielding a more convection dominated problem that is almost hyperbolic, a low-order Galerkin approximation gives rather inaccurate solutions and can even become unstable. This is illustrated in Figure 2.7 where the Galerkin solution for $\eta = 10^{-5}$ using a degree of approximation $n = 2$ is shown. The solution clearly suffers from oscillations. The low-order approximations can be significantly improved by the use of SUPG. On the other hand, if the degree of approximation becomes larger, the Galerkin approximation still proves to be accurate for decreasing viscosity. In that case also, the SUPG extension is more accurate only for the smallest viscosity. Since the SUPG approach is significantly more expensive due to the need to evaluate the second-order derivatives, the high-order Galerkin approximation is in fact preferable.

The results of this test case seem to indicate that for convection dominated problems the application of a SUPG method can improve the solution for low-order approximations. For high-order computations the Galerkin solution seems to be sufficient. In chapter 4 a closer look is taken at unsteady convection or transport problems. It will be seen that in that case a stabilizing diffusion operator can be built into the numerical scheme in a natural way as part of the time-integration, thus excluding the necessity to apply some kind of upwinding.

2.4.3 Application to heat-transfer in a Poiseuille flow

Consider the application of a spectral element discretization to the problem of heat-transfer in a Poiseuille flow between two flat plates. The heat dissipation is neglected and there are no heat sources. This problem is described by the convection-diffusion equation in two dimensions

$$-(\nabla \cdot \eta \nabla)T + (\mathbf{u} \cdot \nabla)T = 0 \qquad \text{in } \boldsymbol{\Omega}, \tag{2.79}$$

where $T(\mathbf{x})$ is the temperature and $\mathbf{u}$ the velocity vector.

The computational domain $\boldsymbol{\Omega}$ with the corresponding boundary conditions is shown in Figure 2.8. For reasons of symmetry only the upper half of the channel is shown. Note that the solution is discontinuous in the upper left corner. The velocity profile is parabolic and taken such that the average velocity is equal to 1

$$\mathbf{u} = [\frac{3}{2}(1 - x_2)(1 + x_2),\ 0]^T \tag{2.80}$$

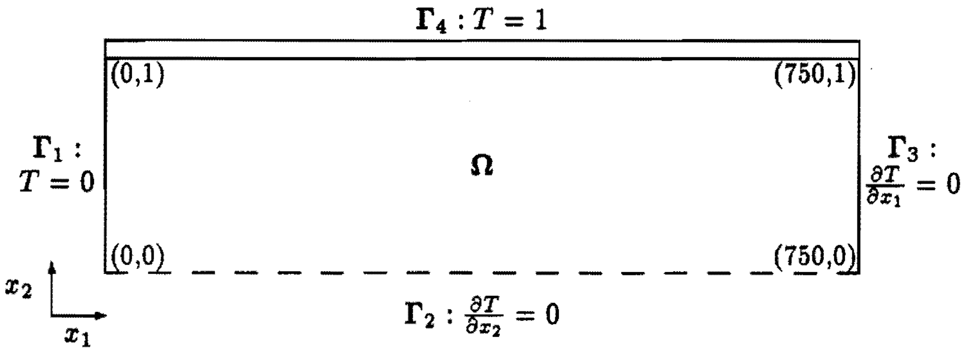


Figure 2.8: Two-dimensional heat transfer in a Poiseuille flow. Computational domain with boundary conditions.

The viscosity is taken to be $\eta = \frac{1}{500}$. The boundary condition on Γ_3 is a homogeneous Neumann boundary condition, as the channel is not long enough for the flow to be of homogeneous temperature on Γ_3.

In Figure 2.9 the isotherms $T(\mathbf{x}) = 0.1k$ ($k = 0, \ldots, 10$) of the approximate solution are shown. In (a)-(b) the number of degrees of freedom of the computation is the same, in (a) $n = 2, n_e = 32$ (8 elements in the x_1 direction, 4 elements in the x_2 direction), in (b) $n = 4, n_e = 8$ (4 elements in the x_1 direction, 2 elements in the x_2 direction). It can be seen that all the computations suffer slightly from the singularity in (0,1). If the degree of the approximation increases, while keeping the total number of degrees of freedom constant, the solution satisfies the homogeneous boundary condition on Γ_2 in a more accurate way. Finally, in (c) the effect of p-convergence is shown. This solution uses the same division of elements as in (b), but the degree of the approximation is now $n = 8$. The effect of the singularity is very small and the boundary conditions are satisfied very accurately.

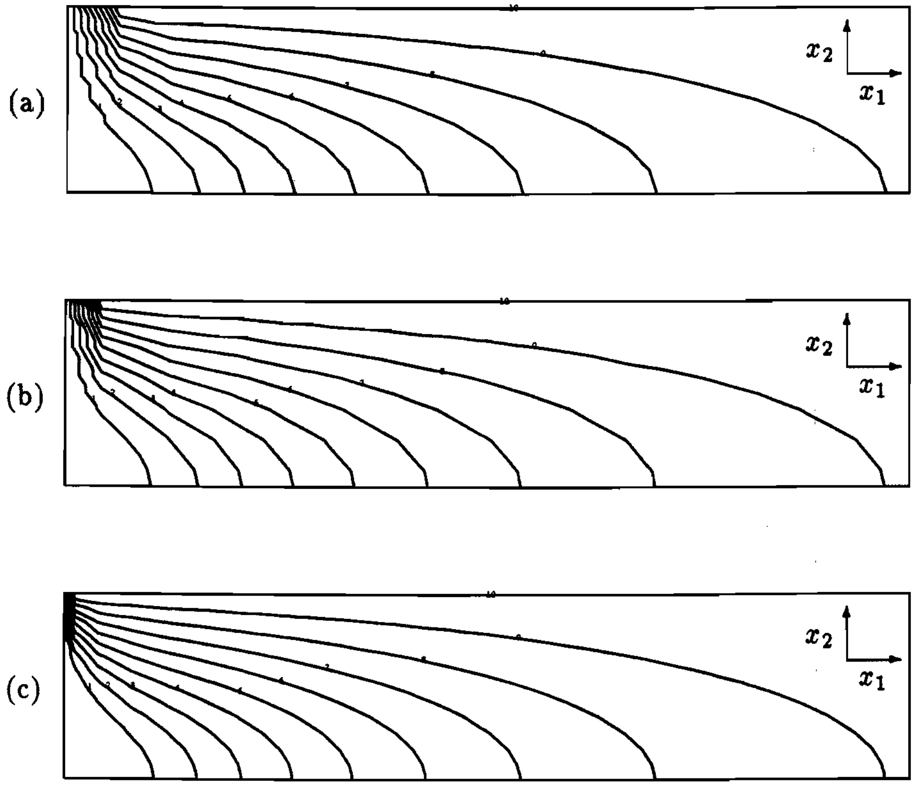


Figure 2.9: Heat transfer in a Poiseuille flow, isotherms of the temperature. (a) $n_e = 32, n = 2$, (b) $n_e = 8, n = 4$, (c) $n_e = 8, n = 8$.

Chapter 3

Finite element preconditioning

3.1 The conditioning of a spectral matrix

Due to the global character of the approximation, the system matrix $\mathbf{S}$ of a high-order or spectral system

$$\mathbf{Sc} = \mathbf{Mf} \tag{3.1}$$

is in general fairly full. Even when applying a spectral domain decomposition method, such as the spectral element method, this will be the case. As a consequence, a direct method for the solution of the spectral system becomes rather costly with respect to both storage and processing time, especially for more-dimensional problems. An efficient solution procedure for the spectral element system therefore requires an iterative algorithm.

The evident drawback of such an iterative procedure is the conditioning of the system matrix $\mathbf{S}$. In Canuto *et al.* (1988) it is shown for spectral methods that there are two constants c_1 and c_2 such that the eigenvalues of the elliptic part of the equation, that is of the second derivative operator, satisfy

$$0 < c_1 \leq -\lambda \leq c_2 N^4, \tag{3.2}$$

with N the maximum of the number of nodes in each spatial direction. Equation (3.2) holds both in the presence of Dirichlet and Neumann boundary conditions, apart from the zero eigenvalue of the Neumann problem. Numerical evidence shows that the largest negative eigenvalues grow like $\mathcal{O}(N^4)$

as $N \to \infty$. Consequently, for a spectral method the condition number $\kappa(\mathbf{S})$ grows like a constant times N^4 ($N \to \infty$). In the case of a spectral element method the condition number is empirically found to be $\mathcal{O}(n_e N^3)$, with n_e the total number of elements in the spectral element discretization, see Maday and Patera (1989). This ill-conditioning of a spectral (element) system is a clear disadvantage compared to the conditioning of low-order h-type finite element and finite difference methods, which scales as $\mathcal{O}(N^2)$, again for the elliptic part of the equation.

For symmetric problems the number of iterations in an iterative solver becomes very large if the degree of approximation is large; e.g. for a Conjugate Gradient iteration the number of iterations needed for convergence scales as $\sqrt{\kappa(\mathbf{S})}$. Therefore some kind of low-order preconditioning has to be applied to improve the condition number. If the equation has a convective contribution, yielding a non-symmetric system matrix, the eigenvalues also have a considerable imaginary part. In that case the condition number is not directly coupled to the convergence of the iterative algorithm. Indeed, there is now no totally satisfactory scheme for non-symmetric problems. However, by looking at the eigenvalues of the system matrix also in this case some indication can be obtained about the quality of the preconditioner.

In Orszag (1980) an iterative procedure is presented that uses finite difference preconditioning of the spectral system matrix. In the present study the idea of finite element preconditioning, proposed simultaneously for Chebyshev pseudospectral systems by Deville and Mund (1985), and by Canuto and Quarteroni (1985), is used to precondition the spectral element system. Both methods are based on the concept of 'spectral equivalence'. Two system matrices of the same problem are said to be spectrally equivalent if they are derived from the same interpolation points, see e.g. Axelsson and Barker (1984). In that case the low-order matrix is a good preconditioner for the high-order matrix.

In section 3.2 the eigenvalues of the preconditioned system matrix are numerically evaluated for two one-dimensional test cases, symmetric and non-symmetric. The condition number of the preconditioned matrix is analyzed. In section 3.3 the basic preconditioned algorithm is given and several acceleration techniques are presented with the idea of finite element preconditioning incorporated. For symmetric systems the point of departure is the Minimum Residual method, in which in each iteration step the Euclidean norm of the residual is minimized. An extension of this algorithm yields the Conjugate Gradient method. For non-symmetric systems, the Bi-CGSTAB algorithm proposed by Van der Vorst (1992) is applied. Finally, in section 3.4

the iterative algorithms are evaluated by applying them to both symmetric and non-symmetric two-dimensional problems.

3.2 Eigenvalues of the preconditioned matrix

An efficient solver for the spectral element system requires an iterative procedure. The evident drawback of the system matrix **S** is its conditioning. In order to improve the ill-conditioning of the spectral system an iterative procedure can be applied that uses low-order preconditioning. Here the lead of finite element preconditioning proposed by Deville and Mund (1985) for Chebyshev pseudospectral methods is followed. The idea is to precondition the full spectral element matrix **S** by the sparse and linear finite element matrix **F** based on the spectral element Legendre Gauss–Lobatto nodes, see Figure 3.1.

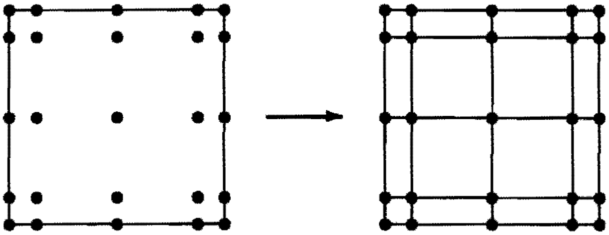


Figure 3.1: A single two-dimensional spectral element containing 25 Legendre Gauss–Lobatto nodes ($n = 4$) (left), and the corresponding linear element mesh containing 16 elements (right).

The reason to expect that the low-order finite element matrix is a good preconditioner for the spectral element matrix, is that the matrices are spectrally equivalent, that is both matrices are derived from the same Legendre Gauss–Lobatto interpolation points. In Axelsson and Barker (1984) a similar concept is described in which a linear element matrix is used as a preconditioner for a quadratic element matrix. Moreover, it is shown that the low-order preconditioning is a good one, in the sense that the condition number of the preconditioned matrix is bounded with respect to the number of degrees of freedom.

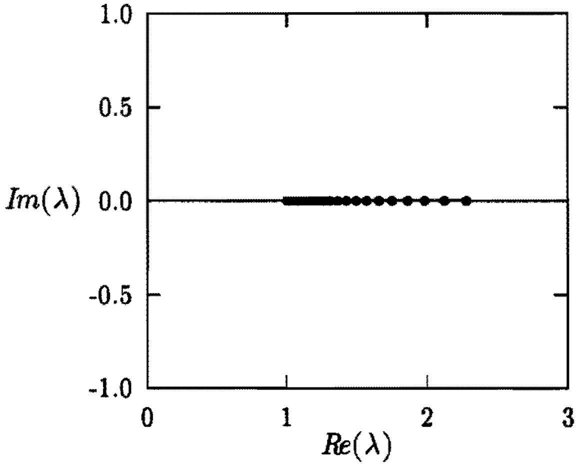


Figure 3.2: Eigenvalues of the preconditioned matrix $\mathbf{F}^{-1}\mathbf{S}$ with $N = 29$, for a symmetric Poisson problem.

Table 3.1: Condition number $\kappa(\mathbf{F}^{-1}\mathbf{S})$ for several values of N for a Poisson problem with two types of boundary conditions (BC).

N	Dirichlet BC	mixed BC
17	2.19	2.20
21	2.23	2.24
25	2.25	2.28
29	2.27	2.30
33	2.30	2.32

In order to analyze the conditioning of the preconditioned matrix $\mathbf{F}^{-1}\mathbf{S}$, the eigenvalues of $\mathbf{F}^{-1}\mathbf{S}$ are numerically evaluated. This is done for two one-dimensional test cases, symmetric and non-symmetric. For simplicity only one element is used ($n_e = 1$) for the moment.

The first test case consists of a Poisson problem with either homogeneous Dirichlet boundary conditions or homogeneous mixed boundary conditions. For the Dirichlet case the eigenvalues of the preconditioned matrix $\mathbf{F}^{-1}\mathbf{S}$ are shown in Figure 3.2. In this case N is equal to 29. The eigenvalues are real and positive. The spectral condition number of the preconditioned matrix, defined as

$$\kappa(\mathbf{F}^{-1}\mathbf{S}) = \frac{|\lambda_{\max}|}{|\lambda_{\min}|}, \qquad \lambda \in \sigma(\mathbf{F}^{-1}\mathbf{S}), \tag{3.3}$$

where $\sigma(\mathbf{F}^{-1}\mathbf{S})$ denotes the eigenvalue spectrum of $\mathbf{F}^{-1}\mathbf{S}$, is given for both types of boundary conditions in Table 3.1 for several values of N. In all cases the condition number increases only very slightly with respect to N, thus confirming that $\mathbf{F}$ is spectrally close to $\mathbf{S}$. Consequently, the rate of convergence hardly depends on the number of nodes used, contrary to the unpreconditioned case.

The second test case consists of a convection-diffusion problem with constant viscosity $\eta = \frac{1}{30}$, see equation (2.48). The boundary conditions are again of homogeneous Dirichlet type or of homogeneous mixed type. This problem yields a non-symmetric spectral element system matrix.

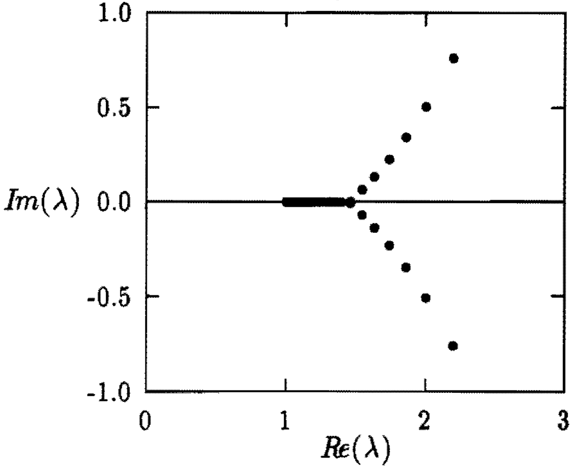


Figure 3.3: Eigenvalues of the preconditioned matrix $\mathbf{F}^{-1}\mathbf{S}$ with $N = 29$, for a non-symmetric convection-diffusion problem.

The eigenvalues of the preconditioned matrix $\mathbf{F}^{-1}\mathbf{S}$ are shown in Figure 3.3 for the Dirichlet case with $N = 29$. For this highly convective problem, the eigenvalues have a relatively large imaginary part. As already stated, at least some information about the quality of the preconditioner can be obtained by looking at the eigenvalues. Although the matrix is now non-symmetric, the condition number $\kappa(\mathbf{F}^{-1}\mathbf{S})$, is still assumed to be given by equation (3.3), see also Demaret and Deville (1989) who also take this approach. The condition number is given in Table 3.2 for several values of N. It is seen that in the non-symmetric case the condition number is bounded with respect to N.

Table 3.2: Condition number $\kappa(\mathbf{F}^{-1}\mathbf{S})$ for several values of N for a convection-diffusion equation with two types of boundary conditions (BC).

N	Dirichlet BC	mixed BC
17	2.20	2.23
21	2.17	2.19
25	2.17	2.18
29	2.17	2.18
33	2.18	2.19

Table 3.3: Condition number $\kappa(\mathbf{S})$ for several values of N for both a Poisson and a convection-diffusion equation.

N	Poisson	convection-diffusion
17	232	10
21	442	19
25	751	34
29	1179	53
33	1604	80

For the sake of completeness, in Table 3.3 the condition number $\kappa(\mathbf{S})$ of the spectral element matrix is given for the corresponding values of N for both problems. Only Dirichlet boundary conditions are considered. It is clear that the spectral element matrix is extremely ill-conditioned, especially in the Poisson case. In both cases, the condition number grows with N. For the case $N = 29$ the eigenvalues of $\mathbf{S}$ are shown in Figure 3.4 and Figure 3.5.

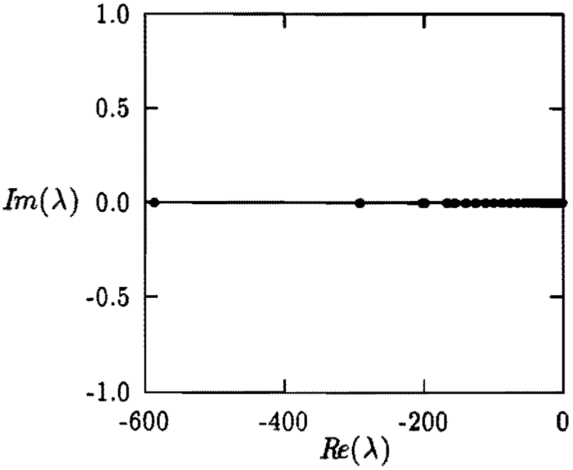


Figure 3.4: Eigenvalues of the spectral element matrix **S** with $N = 29$, for a Poisson problem.

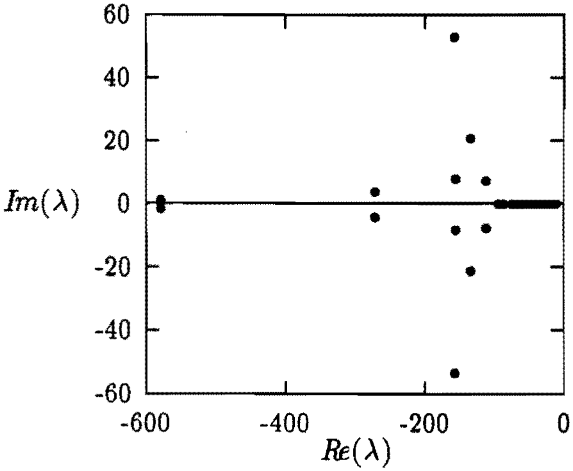


Figure 3.5: Eigenvalues of the spectral element matrix **S** with $N = 29$, for a convection-diffusion problem.

3.3 Finite element preconditioned algorithms

3.3.1 The basic iterative scheme

The iterative algorithm proposed by Deville and Mund (1985) is a simple Richardson scheme. The initial guess is obtained by solving a finite element system. Subsequent approximations are obtained via a correction of the previous solution with the spectral residual

$$\begin{aligned} \mathbf{F}\mathbf{c}^0 &= \mathbf{M}\mathbf{f} \\ \mathbf{F}\mathbf{c}^k &= \mathbf{F}\mathbf{c}^{k-1} + \alpha(\mathbf{M}\mathbf{f} - \mathbf{S}\mathbf{c}^{k-1}) \qquad k = 1, 2, \ldots, \end{aligned} \tag{3.4}$$

with k the iteration counter. It should be noted that since the spectral residual $\mathbf{r}^k = \mathbf{M}\mathbf{f} - \mathbf{S}\mathbf{c}^{k-1}$ is computed in an element-by-element procedure, storage of the full spectral element matrix $\mathbf{S}$ is in this way avoided.

The relaxation parameter α controls the convergence rate of the algorithm. It is well-known that the process converges if the spectral radius of the operator $\mathbf{I} - \alpha\mathbf{F}^{-1}\mathbf{S}$ satisfies

$$\rho(\mathbf{I} - \alpha\mathbf{F}^{-1}\mathbf{S}) < 1. \tag{3.5}$$

In section 3.3.2 a possible choice for α is given, resulting in a Minimum Residual algorithm. In Deville and Mund (1990) the choice of the relaxation parameter is extensively analyzed for Chebyshev methods.

3.3.2 Acceleration techniques for symmetric systems

The results of section 3.2 indicate that the preconditioned matrix $\mathbf{F}^{-1}\mathbf{S}$ is in general well-conditioned. However, to ensure convergence of the algorithm some relaxation is necessary, otherwise equation (3.5) is not satisfied. In general, the determination of a relaxation parameter α that works well for the complete iteration process is rather difficult. An optimal value of the relaxation parameter is given by (Fox and Parker, 1968)

$$\alpha_{\text{opt}} = \frac{2}{|\lambda_{\min}| + |\lambda_{\max}|}, \qquad \lambda \in \sigma(\mathbf{F}^{-1}\mathbf{S}). \tag{3.6}$$

In order to obtain a relaxation parameter α that approximates this optimal value, a fairly accurate estimate of the eigenvalues of $\mathbf{F}^{-1}\mathbf{S}$ is required. However, it is possible to select a sequence of relaxation parameters in a dynamic way, i.e. by picking a new α^k in each iteration step. In this way, one can efficiently obtain relaxation parameters that perform well.

As a starting point for symmetric systems, a Minimum Residual technique is chosen. The same approach is taken by Quarteroni and Zampieri (1992) in the case of Legendre pseudospectral methods. In this method the relaxation parameter is taken to be the real number minimizing the discrete Euclidean norm of the preconditioned residual. In section 3.4.1 numerical values of this relaxation parameter are given. The finite element preconditioned version of this algorithm is given in appendix C.

A substantial improvement in convergence rate can be achieved by the use of conjugate-type methods. For positive-definite symmetric linear systems these methods produce the exact answer in a finite number of steps. Here the well-known Conjugate Gradient method is chosen, see e.g. Reid (1971) and Concus *et al.* (1976). The finite element preconditioned version is also given in appendix C. The performance of these algorithms for several two-dimensional problems is tested in section 3.4.1.

3.3.3 A Bi-CGSTAB algorithm for non-symmetric systems

In the case that the spectral element system to be solved is non-symmetric, the algorithms of the previous section no longer suffice. The subject of iterative schemes for non-symmetric problems has received a lot of attention in the last decade, resulting in methods like Generalized Conjugate Residual (GCR) (Eisenstat *et al.*, 1983), Generalized Minimum Residual (GMRES) (Saad and Schultz, 1986), and Conjugate Gradients Squared (CGS) (Sonneveld, 1989). However, there is no totally satisfactory scheme for non-symmetric problems that is guaranteed to converge under all circumstances.

In this study a variation on the CGS method, the Bi-CGSTAB method is chosen as an iterative solver for non-symmetric systems. This method, proposed by Van der Vorst (1992), can easily be implemented in an efficient way, contrary to e.g. GMRES. The finite element preconditioned version is given in appendix C. It should be noted that the Bi-CGSTAB algorithm can also be applied for the iterative solution of symmetric systems. However, in general this algorithm is more costly than e.g. the Conjugate Gradient algorithm since two finite element systems have to be solved per iteration instead of one. In the next section the Bi-CGSTAB is tested and, for symmetric systems, compared to the algorithms of the previous section: the Minimum Residual method and the Conjugate Gradient method.

3.4 Numerical results

3.4.1 Application to a (symmetric) Poisson problem

In order to evaluate the various preconditioned algorithms presented in the previous sections, numerical results are given for several two-dimensional problems, some of which are adapted from Deville and Mund (1985) and (1992). Some of these results appear also in Timmermans and Van de Vosse (1993^a). This section deals with symmetric two-dimensional problems. The non-symmetric case is discussed in section 3.4.2.

The test problem consists of a two-dimensional Poisson equation

$$-\nabla^2 c(\mathbf{x}) = f(\mathbf{x}) \qquad \text{in } \boldsymbol{\Omega}, \tag{3.7}$$

with domain $\boldsymbol{\Omega} = (0,1) \times (0,1)$ and homogeneous Dirichlet boundary conditions. The source term is chosen such that the exact solution to this problem is given by

$$c(\mathbf{x}) = \sin(4\pi x_1)\sin(4\pi x_2). \tag{3.8}$$

Table 3.4: Two-dimensional Poisson problem. Evolution of the discrete maximum error ε for spectral element approximation with $n_e = 1$ and n varying.

n	discrete maximum error ε
8	$0.14 \cdot 10^{-1}$
16	$0.36 \cdot 10^{-7}$
32	$0.28 \cdot 10^{-13}$

Table 3.4 shows the results of a direct spectral element approximation to this Poisson problem using one element ($n_e = 1$) and a varying degree of approximation n. Again, spectral accuracy is obtained if the degree of the approximation increases.

Consider now the application of the iterative algorithms for symmetric systems to the solution of the Poisson problem for $n = 32$. The condition number of the preconditioned matrix $\mathbf{F}^{-1}\mathbf{S}$ is equal to 2.30. Figure 3.6 shows the performance of the algorithms for the evolution of the discrete maximum error $\varepsilon = \|c - c_h\|_{\infty,gl}$ in the solution with respect to the iteration index k (c_h is the corrected solution in every iteration step). For all three algorithms convergence to the spectral element solution is obtained within few iterations. In Table 3.5 a processing time comparison between the three methods is given.

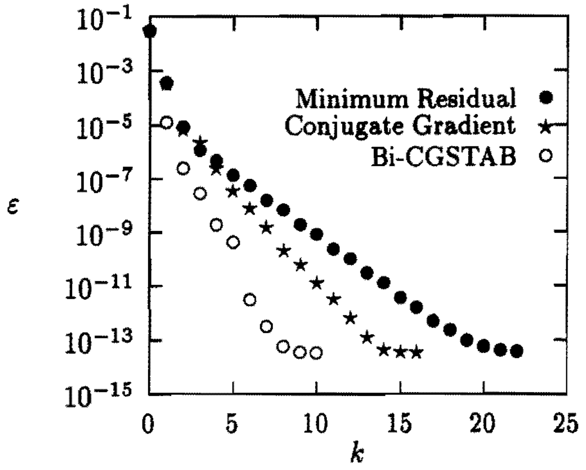


Figure 3.6: Evolution of the discrete maximum error ε with respect to the iteration index k for the two-dimensional Poisson problem ($n_e = 1, n = 32$).

Table 3.5: Two-dimensional Poisson problem ($n_e = 1, n = 32$). Processing time comparison between the Minimum Residual method (MR), the Conjugate Gradient method (CG) and the CGSTAB method.

	MR	CG	CGSTAB
time per iteration	2.60	2.63	5.23
number of iterations	20	14	8
total time	52.00	36.82	41.84

The processing time per iteration of the Minimum Residual method and Conjugate Gradient method is comparable; the number of iterations needed to converge in the Conjugate Gradient method is less. As already stated, the Bi-CGSTAB method takes about twice as much processing time per iteration, although less iterations are required to converge. A comparison for this relatively large two-dimensional problem between the direct solver and the iterative solver shows that, despite the fact that in each iteration step a spectral residual is computed, the iterative solver is 2.1 times faster than the direct solver. Moreover, a storage reduction by a factor 30 is obtained using an iterative procedure.

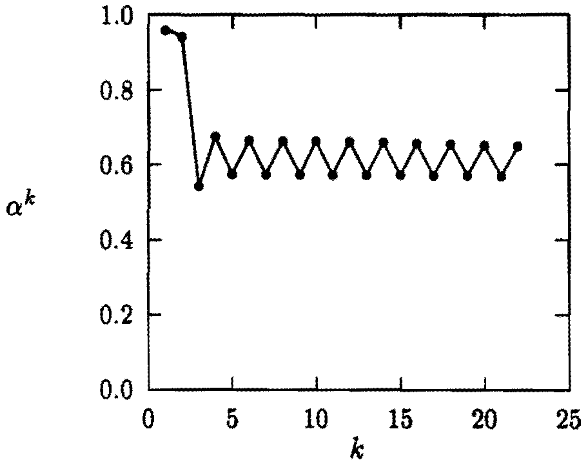


Figure 3.7: Evolution of the relaxation parameter α^k in the Minimum Residual algorithm with respect to the iteration index k for the two-dimensional Poisson problem ($n_e = 1, n = 32$).

Figure 3.7 shows the value of the dynamic relaxation parameter α^k with respect to the iteration index k for the Minimum Residual method. It is seen that after the first iterations the value of α^k oscillates around a value of slightly above 0.6. The optimal value of the Richardson iteration for this problem, given by equation (3.6), is indeed equal to 0.62.

In order to test the influence of domain decomposition consider a Poisson problem with computational domain Ω as shown in Figure 3.8 and homogeneous Dirichlet boundary conditions. The exact solution is given by

$$c(\mathbf{x}) = \sin(2\pi x_1)\sin(2\pi x_2). \tag{3.9}$$

A direct spectral element solution using a degree of approximation $n = 16$ achieves spectral accuracy. The performance of the three algorithms is shown in Figure 3.9.

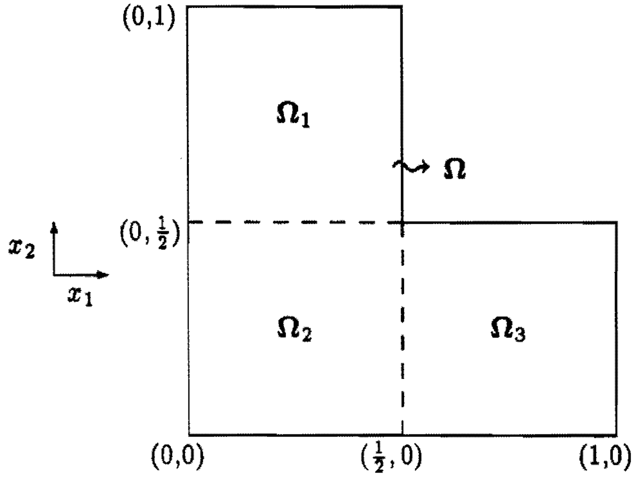


Figure 3.8: Two-dimensional Poisson problem with domain decomposition ($n_e = 3$). Computational domain.

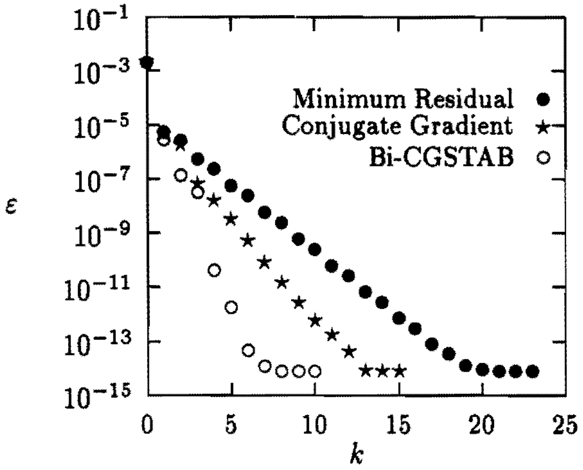


Figure 3.9: Evolution of the discrete maximum error ε with respect to the iteration index k for the two-dimensional Poisson problem ($n_e = 3, n = 16$).

Table 3.6: Two-dimensional Poisson problem ($n_e = 3, n = 16$). Processing time comparison between the Minimum Residual method (MR), the Conjugate Gradient method (CG) and the CGSTAB method.

	MR	CG	CGSTAB
time per iteration	0.84	0.86	1.70
number of iterations	19	13	7
total time	15.96	11.18	11.90

The processing time comparison between the three methods is given in Table 3.6. Again the Conjugate Gradient iteration converges fastest; the processing time of the CGSTAB algorithm is now comparable. The iterative solver is a factor 2.6 faster than the direct spectral element simulation. Also a storage reduction by a factor 8 is obtained using an iterative solver.

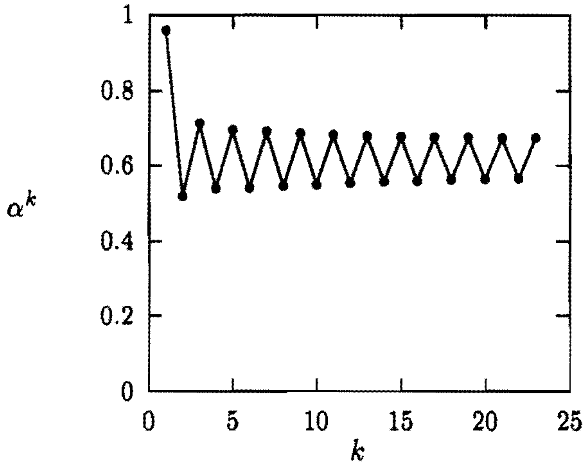


Figure 3.10: Evolution of the relaxation parameter α^k in the Minimum Residual algorithm with respect to the iteration index k for the two-dimensional Poisson problem ($n_e = 3, n = 16$).

Figure 3.10 shows the value of the dynamic relaxation parameter α^k with respect to the iteration index k for the Minimum Residual method. The results are comparable to those in Figure 3.7.

3.4.2 Application to a (non-symmetric) convection-diffusion problem

As an example of a two-dimensional non-symmetric test case, consider again the approximation of a two-dimensional boundary layer described by the convection-diffusion equation, see section 2.4.1. For $\eta = 1$ direct spectral element solution using a degree of approximation $n = 12$ achieves spectral accuracy; for $\eta = \frac{1}{30}$, the highly convective problem, this is the case using a degree of approximation $n = 32$. In both cases $n_e = 1$.

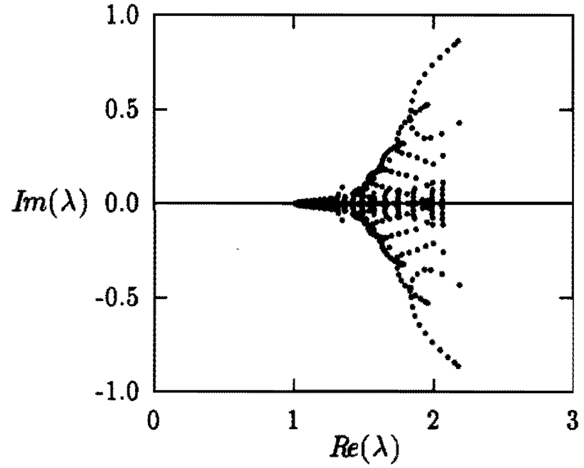


Figure 3.11: Eigenvalues of the preconditioned matrix $\mathbf{F}^{-1}\mathbf{S}$ for a two-dimensional convection-diffusion problem ($\eta = \frac{1}{30}$).

For the case $\eta = \frac{1}{30}$ the eigenvalues of the preconditioned matrix $\mathbf{F}^{-1}\mathbf{S}$ are shown in Figure 3.11. The condition number $\kappa(\mathbf{F}^{-1}\mathbf{S})$ defined by equation (3.3) is equal to 2.07 for $\eta = 1$ and equal to 2.19 for $\eta = \frac{1}{30}$.

Figure 3.12 shows the evolution of the discrete maximum error in the solution with respect to the iteration index k for the Bi-CGSTAB method. It should be noted that even in the case $\eta = 1$, in which the imaginary part of the eigenvalues of the preconditioned matrix can be neglected, the Minimum Residual and Conjugate Gradient method fail to converge; since the preconditioned matrix is non-symmetric, several orthogonality relations in these algorithms are not satisfied. The Bi-CGSTAB method on the other

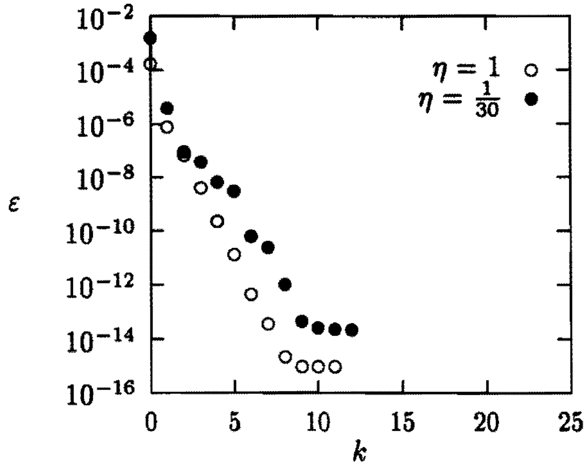


Figure 3.12: Evolution of the discrete maximum error ε with respect to the iteration index k for the two-dimensional convection-diffusion problem.

hand converges to the spectral element solution again within a few iterations.

A comparison between the direct solver and the iterative solver shows that in both cases ($n = 12$ and $n = 32$), a considerable storage reduction is obtained; a factor 8 for the case $n = 12$ and a factor 30 for the case $n = 32$. For the large computation also a gain in processing time is achieved of a factor 2.0.

3.5 Conclusions

Concluding it can be said that the use of low-order finite element preconditioning in iterative solvers for a spectral element system seems to be a good choice. The preconditioned matrix is well-conditioned, also for large values of N, where N is the maximum of the number of degrees of freedom in each spatial direction. Moreover, whereas the condition number of the spectral element matrix grows if the degree of approximation grows, the condition number of the preconditioned matrix remains practically constant, indicating that the finite element and the spectral element systems are spectrally equivalent.

For symmetric problems a Conjugate Gradient algorithm is preferable, whereas for non-symmetric systems the Bi-CGSTAB algorithm gives very good results. Also, by no longer storing the relatively full spectral element matrix a considerable reduction in storage can be obtained. Moreover, for large two-dimensional computations also a gain in processing time is achieved, despite the fact that in each iteration step a spectral residual must be computed. For three-dimensional computations the gain in processing time and storage requirements will be even higher. Finally, it should be noted that when using an iterative procedure, the processing time can also be reduced by stopping the iteration after only a few steps when a desired accuracy is obtained.

Chapter 4

Solution of unsteady convection-diffusion problems

4.1 Convection-diffusion problems

The point of departure in this chapter is the unsteady convection-diffusion equation for divergence-free velocity fields $\mathbf{u}$

$$\begin{cases} \dfrac{\partial \mathbf{c}}{\partial t} + (\mathbf{u} \cdot \nabla)\,\mathbf{c} - (\nabla \cdot \eta \nabla)\mathbf{c} = \mathbf{f} & \text{in } \boldsymbol{\Omega}, \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \overline{\boldsymbol{\Omega}}, \end{cases} \tag{4.1}$$

with $\boldsymbol{\Omega} \subset \mathbf{R}^d$ an open and bounded region with boundary $\boldsymbol{\Gamma}$. The velocity field $\mathbf{u}(\mathbf{x}, t)$ in $\overline{\boldsymbol{\Omega}} = \boldsymbol{\Omega} \cup \boldsymbol{\Gamma}$ is given for $t \geq 0$. Note that the case $\mathbf{c} = \mathbf{u}$ yields a non-linear Burgers equation. For convenience, homogeneous Dirichlet boundary conditions are assumed, given by

$$\mathbf{c}(\mathbf{x}, t) = \mathbf{0}, \qquad \mathbf{x} \text{ on } \boldsymbol{\Gamma},\ t \geq 0. \tag{4.2}$$

The initial conditions are given by

$$\mathbf{c}(\mathbf{x}, 0) = \mathbf{c}_0, \qquad \mathbf{x} \text{ in } \overline{\boldsymbol{\Omega}}. \tag{4.3}$$

Equation (4.1) is important for several classes of problems:

- General convection-diffusion problems (energy equation, constitutive equations in visco-elastic flow). In that case the viscosity η is positive but it can be a function of $\mathbf{c}$.

- Pure convection problems (particle tracking, free-boundary problems). In the case $\eta = 0$ equation (4.1) results in a convection (or transport) problem and describes the convection of the quantity $\mathbf{c}$. A convection problem also arises if an operator splitting method is applied to equation (4.1). This case is discussed in the subsequent sections.
- Splitting methods and pressure correction (projection) methods for the incompressible Navier–Stokes equations. In that case the equations are non-linear ($\mathbf{c} = \mathbf{u}$). This will be discussed in chapter 5.

Standard Galerkin approximations of convection-diffusion problems are often unstable and show spurious oscillations or 'wiggles' related to the fact that the resulting system matrix is no longer diagonally dominant. Especially if the convection part dominates the problem, these oscillations can only be suppressed by severe mesh and time-step refinement, which is often not practical.

An overview of recently developed methods that stabilize the numerical approximation of convection dominated problems is given by Donea (1991). For steady problems a common approach is the use of some kind of upwinding. Among such techniques are Streamline Upwind methods (SU) (Hughes, 1978), Streamline Upwind/Petrov–Galerkin methods (SUPG) (Brooks and Hughes, 1982; Johnson, 1987) and Galerkin Least Squares methods (GLS) (Hughes *et al.*, 1989).

For unsteady problems various generalized Galerkin methods have been developed in the last decade. An important class of these methods is formed by the Taylor–Galerkin methods. These methods, introduced for finite elements by Donea (1984), add in a natural way a stabilizing diffusion term to the numerical scheme, using Taylor series expansions in time including second- and third-order terms. In the finite difference context, similar techniques are provided by Lax–Wendroff schemes (Lax and Wendroff, 1960). In contrast to the general procedure, Taylor–Galerkin methods usually consider the time-integration before the spatial discretization. A recent analysis of Taylor–Galerkin schemes is given by Donea and Quartapelle (1992).

In this chapter Taylor–Galerkin time-integration is combined with a spectral element spatial discretization for the approximation of convection-diffusion problems. In section 4.2 an operator splitting approach is described, which decouples a convection-diffusion problem into a pure convection problem and a pure diffusion problem. The splitting method is based on the classical splitting methods (Yanenko, 1971; Marchuk, 1975), but also follows an approach proposed by Maday *et al.* (1990). In section 4.3 several

Taylor–Galerkin schemes are deduced that are specificly suited for (non-linear) convection problems with time-dependent and divergence-free velocity. The schemes are applied to several convection problems, both linear and non-linear. In section 4.4 the full operator splitting scheme is given for convection-diffusion problems. Numerical results are given also for this scheme. The approach for convection-diffusion problems is the basis for the treatment of the Navier–Stokes equations in chapter 5. Finally, in section 4.5 some conclusions are drawn.

4.2 Operator splitting approach

With respect to the choice of time-integration it is preferable to decouple the treatment of convection and diffusion. For spectral-type methods the eigenvalues of the diffusion system are real and strictly negative and grow like $\mathcal{O}(N^4)$, if $N \to \infty$, see chapter 3. Here N is the maximum of the number of degrees of freedom in each spatial direction. As a consequence, the diffusion (or stiff) part requires a time-integration with a stability area that includes the negative real axis; in order to satisfy this requirement a time-integration needs to be A-stable or at least $A(\alpha)$-stable, see appendix D. As is well-known, some type of (semi-)implicit time-integration is then in order. For the diffusive part the system matrix does not depend on time and suitable iterative solvers like finite element preconditioning can be applied in each time-step. On the other hand, especially in the case that the problem is non-linear or in the case that the velocity $\mathbf{u}$ is time-dependent, an implicit time-integration of the convective terms requires an evaluation of the convective matrix in each time-step. Therefore, explicit time-integration is virtually necessary to ensure efficiency of the numerical scheme; in that case the solution of the system only involves matrix-vector products which can be evaluated in an element-by-element procedure. Moreover, for the convective part the eigenvalues are complex and grow 'only' like $\mathcal{O}(N^2)$, if $N \to \infty$ (Canuto *et al.*, 1988). The restriction to the time-step is therefore not too severe if explicit time-integration is applied.

In view of this, it appears appropriate to treat unsteady convection-diffusion problems by an operator splitting technique in which the problem is decomposed in a pure convection problem and a pure diffusion problem. Both problems are then solved by suitable time-integrations with different time-steps, if necessary.

Thereto the convection-diffusion problem (4.1) is rewritten as follows

$$\frac{\partial \mathbf{c}}{\partial t} = \mathcal{D}\mathbf{c} + \mathcal{C}\mathbf{c} + \mathbf{f}, \tag{4.4}$$

where $\mathcal{D} = (\nabla \cdot \eta \nabla)$ is the diffusion operator and $\mathcal{C} = -(\mathbf{u} \cdot \nabla)$ is the convection operator. Following the idea of Maday *et al.* (1990), equation (4.4) is written in terms of an integrating factor in $\mathcal{C}$

$$\frac{\partial}{\partial t}\left(\mathcal{Q}_{\mathcal{C}}^{(t^*,t)}\mathbf{c}(t)\right) = \mathcal{Q}_{\mathcal{C}}^{(t^*,t)}(\mathcal{D}\mathbf{c} + \mathbf{f}), \tag{4.5}$$

with t^* an arbitrary fixed time. The integrating factor $\mathcal{Q}_{\mathcal{C}}^{(t^*,t)}$ is defined by

$$\frac{\partial}{\partial t}\mathcal{Q}_{\mathcal{C}}^{(t^*,t)} = -\mathcal{Q}_{\mathcal{C}}^{(t^*,t)}\mathcal{C}, \quad \mathcal{Q}_{\mathcal{C}}^{(t^*,t^*)} = \mathcal{I}, \tag{4.6}$$

where $\mathcal{I}$ is the identity operator. Equation (4.5) can be viewed as a diffusion problem for the new variable $\mathcal{Q}_{\mathcal{C}}^{(t^*,t)}\mathbf{c}$. It is integrated by a suitable time-integration for the diffusion operator $\mathcal{D}$. A useful class of $A(\alpha)$-stable time-integration methods is given by the so-called 'backward differences formulae' (Hairer *et al.*, 1987). These schemes are accurate for all components around the origin in the stability diagram and absolutely stable away from the origin in the left imaginary plane. Thus it is possible to use high-order backward differences schemes without the severe constraints on the time-step that are needed for other high-order multistep schemes like the Adams–Moulton methods, which are not $A(\alpha)$-stable for any order higher than 2. Stability regions of backward differences schemes can be found in appendix D.

Taking $t^* = t^{n+1}$, application of a backward differences scheme with a time-step $\Delta t = t^{n+1} - t^n$ to equation (4.5) gives the following semi-discrete system

$$\frac{\beta_0 \mathbf{c}^{n+1} - \sum_{i=1}^{k} \beta_i \mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n+1-i})}\mathbf{c}^{n+1-i}}{\Delta t} = \mathcal{D}\mathbf{c}^{n+1} + \mathbf{f}^{n+1}, \tag{4.7}$$

where e.g. the superscript $n+1$ denotes the approximation at time t^{n+1}. Note that use has been made of the fact that $\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n+1})} = \mathcal{I}$. For consistency it is required that

$$\beta_0 = \sum_{i=1}^{k} \beta_i. \tag{4.8}$$

The coefficients of the various schemes are listed in appendix D.

Obviously, integration of the diffusion equation (4.5) requires the evaluation of integrating terms of the form $\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n+1-i})}\mathbf{c}^{n+1-i}(i = 1, 2, \ldots)$. To avoid explicit construction of $\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n+1-i})}$ an auxiliary variable $\tilde{\mathbf{c}}(s)$ is introduced that satisfies the following associated initial value problem

$$\begin{cases} \dfrac{\partial \tilde{\mathbf{c}}(s)}{\partial s} = \mathcal{C}\tilde{\mathbf{c}}(s), \quad 0 < s < i\Delta t, \\ \tilde{\mathbf{c}}(0) = \mathbf{c}^{n+1-i}. \end{cases} \tag{4.9}$$

It then follows that

$$\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n+1-i})}\mathbf{c}^{n+1-i} = \tilde{\mathbf{c}}(i\Delta t). \tag{4.10}$$

Problem (4.9), accounting for the convection part, can be solved using a suitable (and preferably explicit) scheme with a time-step Δs which can be taken different from Δt. In the next section several Taylor–Galerkin time-integration methods, appropriate for both linear and non-linear convection problems are proposed and discussed. It is important to note that the integrating factor $\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n+1-i})}$ is never constructed explicitly; rather, the 'action' of the integrating factor is evaluated through solution of the associated convection problem (4.9).

Remark

An alternative approach for the diffusion step is to use the θ-method (or the trapezoidal method). The semi-discrete equation for the diffusion operator then becomes

$$\begin{aligned} \frac{\mathbf{c}^{n+1} - \mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n})}\mathbf{c}^{n}}{\Delta t} &= \theta(\mathcal{D}\mathbf{c}^{n+1} + \mathbf{f}^{n+1}) \\ &+ (1-\theta)\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n})}(\mathcal{D}\mathbf{c}^{n} + \mathbf{f}^{n}). \end{aligned} \tag{4.11}$$

The terms $\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n})}\mathbf{c}^{n}$ and $\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n})}(\mathcal{D}\mathbf{c}^{n} + \mathbf{f}^{n})$ are calculated according to a convection problem similar to problem (4.9).

For $\theta = \frac{1}{2}$ this scheme results in a second-order accurate Crank–Nicolson method. This scheme is commonly used for diffusion problems. In approximations of Navier–Stokes problems it is frequently applied to the viscous and pressure terms. Although the Crank–Nicolson scheme is $A(\alpha)$-stable for such terms, it has the disadvantage that it damps high frequency components very weakly, whereas these components in reality decay very rapidly. In cases where this is undesirable, a possible strategy is to use $\theta = \frac{1}{2} + \delta\Delta t$, where δ is a small positive constant. This method damps all components of the solution and is formally second-order in time.

4.3 Taylor–Galerkin time-integration

4.3.1 Schemes for linear convection

Taylor–Galerkin time-integration methods are an extension of some typical time-stepping methods on the basis of Taylor series expansions including up to third-order terms. They are appropriate for pure convection problems, both linear and non-linear, since a stabilizing diffusion term is added to the numerical scheme in a natural way as part of the time-integration. Taylor–Galerkin methods were first proposed by Donea (1984) for linear convection equations with both constant and variable, but not time-dependent, velocity.

In this section several Taylor–Galerkin for linear convection problems are discussed (the Galerkin discretization is given in section 4.3.3). For reasons mentioned in the previous section, explicit schemes are preferable. An overview is given of second-order methods that lead to explicit schemes with special emphasis to convection problems with a time-dependent and divergence-free velocity field. For velocity fields that are not divergence-free some of the schemes are not applicable. Keeping in mind possible applications, e.g. the incompressible Navier–Stokes equations, this seems to be a valid assumption. In Donea (1984) third-order methods are given; however, these result in semi-implicit schemes.

Consider the linear convection equation

$$\left\{\begin{array}{ll} \dfrac{\partial \mathbf{c}}{\partial s} = -(\mathbf{u}\cdot\nabla)\mathbf{c} & \text{in } \boldsymbol{\Omega}, \\ \nabla\cdot\mathbf{u} = 0 & \text{in } \overline{\boldsymbol{\Omega}}, \end{array}\right. \tag{4.12}$$

with initial and boundary conditions. Following the notation of the previous section the variable s is used for the time in a convection problem. The velocity field $\mathbf{u}(\mathbf{x}, s)$ can depend both on space and on time.

Explicit Euler Taylor–Galerkin (EETG)

The schemes for linear convection are Euler schemes, i.e. the Taylor series of the time-derivative is based on forward differences. The approximation of

c at $s = m\Delta s$ is denoted by $\mathbf{c}^m$. The discretization in time is derived from the explicit Taylor series

$$\mathbf{c}^{m+1} = \mathbf{c}^m + \Delta s \frac{\partial \mathbf{c}}{\partial s}|^m + \frac{\Delta s^2}{2}\frac{\partial^2 \mathbf{c}}{\partial s^2}|^m + \mathcal{O}(\Delta s^3). \tag{4.13}$$

Using the original differential equation (4.12), the second derivative can be expressed as

$$\frac{\partial^2 \mathbf{c}}{\partial s^2} = -\left(\frac{\partial \mathbf{u}}{\partial s}\cdot\nabla\right)\mathbf{c} + (\mathbf{u}\cdot\nabla)(\mathbf{u}\cdot\nabla)\mathbf{c}. \tag{4.14}$$

Substitution of equation (4.14) and of the original equation (4.12) into equation (4.13) yields a second-order EETG method

$$\begin{aligned}\mathbf{c}^{m+1} &= \mathbf{c}^m - \Delta s\left(\left(\mathbf{u}^m + \frac{\Delta s}{2}\frac{\partial \mathbf{u}}{\partial s}|^m\right)\cdot\nabla\right)\mathbf{c}^m \\ &+ \frac{\Delta s^2}{2}(\mathbf{u}^m\cdot\nabla)(\mathbf{u}^m\cdot\nabla)\mathbf{c}^m + \mathcal{O}(\Delta s^3). \end{aligned}\tag{4.15}$$

The third term on the righthand side is not to be thought of as an artificial diffusion term, but rather as part of the time-integration. It possesses the tensorial structure similar to the Streamline Upwind/Petrov–Galerkin formulation proposed by Brooks and Hughes (1982).

In fact, the derivation of this scheme is similar to that of the schemes proposed by Donea *et al.* (1984) for convection-diffusion problems. However, if there is a diffusion term, the method becomes a semi-implicit one. The coefficient of the convective term in equation (4.15), containing the time-derivative of the velocity, can either be approximated implicitly as

$$\begin{aligned}\mathbf{u}^m + \frac{\Delta s}{2}\frac{\partial \mathbf{u}}{\partial s}|^m &= \mathbf{u}^m + \frac{\Delta s}{2}\left(\frac{\mathbf{u}^{m+1}-\mathbf{u}^m}{\Delta s} + \mathcal{O}(\Delta s)\right) \\ &= \frac{\mathbf{u}^m + \mathbf{u}^{m+1}}{2} + \mathcal{O}(\Delta s^2) \\ &= \mathbf{u}^{m+\frac{1}{2}} + \mathcal{O}(\Delta s^2),\end{aligned}\tag{4.16}$$

or explicitly as

$$\begin{aligned}\mathbf{u}^m + \frac{\Delta s}{2}\frac{\partial \mathbf{u}}{\partial s}|^m &= \mathbf{u}^m + \frac{\Delta s}{2}\left(\frac{\mathbf{u}^{m}-\mathbf{u}^{m-1}}{\Delta s} + \mathcal{O}(\Delta s)\right) \\ &= \frac{3\mathbf{u}^m - \mathbf{u}^{m-1}}{2} + \mathcal{O}(\Delta s^2) \\ &= \mathbf{u}^{m-\frac{1}{2}} + \mathcal{O}(\Delta s^2).\end{aligned}\tag{4.17}$$

In equation (4.17) the notation $\mathbf{u}^{m-\frac{1}{2}}$ is introduced for simplicity. If the implicit approximation (4.16) is applied, only one old velocity $\mathbf{u}^m$ has to be stored; in the explicit case (4.17) two old velocities $\mathbf{u}^m$ and $\mathbf{u}^{m-1}$ must be stored. However, it is not always possible to have the velocity at the time-level $m+1$ available, e.g. in non-linear or coupled problems. In that case the explicit approximation has to be applied.

The two-step EETG scheme
Equation (4.13) can be rewritten as

$$\mathbf{c}^{m+1} = \mathbf{c}^m + \Delta s \frac{\partial \mathbf{c}}{\partial s}\Big|^{m+\frac{1}{2}} + \mathcal{O}(\Delta s^3), \tag{4.18}$$

where

$$\frac{\partial \mathbf{c}}{\partial s}\Big|^{m+\frac{1}{2}} = \frac{\partial \mathbf{c}}{\partial s}\Big|^{m} + \frac{\Delta s}{2}\frac{\partial^2 \mathbf{c}}{\partial s^2}\Big|^{m}. \tag{4.19}$$

Substituting the original equation (4.12) at the time-level $m+\frac{1}{2}$ into equation (4.18) yields

$$\mathbf{c}^{m+1} = \mathbf{c}^m - \Delta s\ (\mathbf{u}^{m+\frac{1}{2}} \cdot \nabla)\mathbf{c}^{m+\frac{1}{2}} + \mathcal{O}(\Delta s^3). \tag{4.20}$$

The explicit Taylor series for $\mathbf{c}^{m+\frac{1}{2}}$ reads

$$\mathbf{c}^{m+\frac{1}{2}} = \mathbf{c}^m + \frac{\Delta s}{2}\frac{\partial \mathbf{c}}{\partial s}\Big|^{m} + \mathcal{O}(\Delta s^2). \tag{4.21}$$

Again substituting the original equation, now at time-level m, gives

$$\mathbf{c}^{m+\frac{1}{2}} = \mathbf{c}^m - \frac{\Delta s}{2}(\mathbf{u}^m \cdot \nabla)\mathbf{c}^m. \tag{4.22}$$

Equations (4.20) and (4.22) give a second-order accurate two-step explicit scheme. Morgan *et al.* (1991) propose this scheme for general linear conservation problems. The same scheme is classified by Hirsch (1988) as a second-order Runge-Kutta predictor-corrector method. For convection problems the two-step scheme is not asymptotically stable. However, it is quite useful for integration over a fixed time-interval. Since the growth rate of the numerical solution is proportional to Δs^3, one need merely choose the time-step sufficiently small so that the numerical growth of the solution is insignificant. This property is often referred to as weak instability, see e.g. Canuto *et al.* (1988).

The two-step scheme possesses the same accuracy as the EETG method. However, there is no need to evaluate the 'expensive' second-order term. Moreover, application of a Galerkin spatial discretization (spectral elements) to this scheme is simpler than is the case for the EETG scheme. This aspect will be discussed in section 4.3.3.

4.3.2 Schemes for non-linear convection

In this section Taylor–Galerkin schemes for non-linear convection problems are discussed. Although most schemes are practically the same as the schemes for linear convection, several aspects must be addressed separately. Consider the non-linear convection equation (Burgers equation with zero diffusivity)

$$\begin{cases} \dfrac{\partial \mathbf{u}}{\partial s} = -(\mathbf{u}\cdot\nabla)\,\mathbf{u} & \text{in } \Omega, \\ \nabla\cdot\mathbf{u} = 0 & \text{in } \overline{\Omega}, \end{cases} \tag{4.23}$$

with initial and boundary conditions.

Explicit Euler Taylor–Galerkin (EETG)
The derivation of a second-order accurate explicit Euler Taylor–Galerkin scheme for the non-linear Burgers equation for divergence-free velocity is completely analogous to the linear case. The scheme reads

$$\begin{aligned} \mathbf{u}^{m+1} &= \mathbf{u}^m - \Delta s\left(\left(\mathbf{u}^m + \frac{\Delta s}{2}\frac{\partial \mathbf{u}}{\partial s}|^m\right)\cdot\nabla\right)\mathbf{u}^m \\ &+ \frac{\Delta s^2}{2}(\mathbf{u}^m\cdot\nabla)(\mathbf{u}^m\cdot\nabla)\mathbf{u}^m + \mathcal{O}(\Delta s^3). \end{aligned} \tag{4.24}$$

In order to keep the scheme explicit, the coefficient of the convective term must now be approximated according to equation (4.17).

A different Taylor–Galerkin scheme for non-linear convection equations is proposed by Laval and Quartapelle (1990). Their scheme is more general in the sense that the velocity is not required to be divergence-free. Moreover, the treatment of the time-derivative of the velocity differs from the one proposed in equation (4.17). As the equations are non-linear, in their scheme the original equation (4.23) is substituted for the time-derivative of the velocity; this leads to a rather complicated scheme, especially if a Galerkin spatial discretization is applied. The treatment of the time-derivative of the velocity proposed here is quite simple and does not lower the second-order accuracy of the original scheme.

The two-step EETG scheme
Again analogous to the linear case, the two-step Euler Taylor–Galerkin scheme can be written as

$$\begin{aligned}\mathbf{u}^{m+\frac{1}{2}} &= \mathbf{u}^m - \frac{\Delta s}{2}(\mathbf{u}^m \cdot \nabla)\,\mathbf{u}^m,\\ \mathbf{u}^{m+1} &= \mathbf{u}^m - \Delta s\left(\mathbf{u}^{m+\frac{1}{2}} \cdot \nabla\right)\mathbf{u}^{m+\frac{1}{2}}.\end{aligned} \tag{4.25}$$

For reasons of stability, in chapter 5 this method is extended to a three-step scheme. For non-linear problems another way to construct a two-step EETG scheme is to substitute in the second equation of (4.25) the following Taylor series for $\mathbf{u}^{m+\frac{1}{2}}$

$$\begin{aligned}\mathbf{u}^{m+\frac{1}{2}} &= \mathbf{u}^m + \frac{\Delta s}{2}\frac{\partial \mathbf{u}}{\partial s}|^m + \mathcal{O}(\Delta s^2)\\ &= \mathbf{u}^m - \frac{\Delta s}{2}(\mathbf{u}^m \cdot \nabla)\mathbf{u}^m + \mathcal{O}(\Delta s^2).\end{aligned} \tag{4.26}$$

Then for divergence-free velocity fields the following scheme is obtained

$$\begin{aligned}\mathbf{u}^{m+\frac{1}{2}} = \mathbf{u}^m &- \frac{\Delta s}{2}(\mathbf{u}^m \cdot \nabla)\,\mathbf{u}^m,\\ \mathbf{u}^{m+1} = \mathbf{u}^m &- \Delta s\left(\mathbf{u}^{m+\frac{1}{2}} \cdot \nabla\right)\mathbf{u}^m\\ &+ \frac{\Delta s^2}{2}(\mathbf{u}^m \cdot \nabla)(\mathbf{u}^m \cdot \nabla)\mathbf{u}^m.\end{aligned} \tag{4.27}$$

This scheme can be viewed as a stabilized version of the original two-step scheme (4.25). The second equation of (4.27) is similar to the one-step EETG scheme (4.24). The difference is that, since the problem is non-linear, it is possible to derive the coefficient for the convective term using the original equation. The most efficient and easiest to implement scheme is, also for non-linear convection problems, the two-step scheme (4.25); especially in an operator splitting procedure.

4.3.3 Spectral element discretization

Application of the Galerkin spectral element discretization to the semi-discrete systems of the previous sections is performed in the standard way. For details of the spectral element discretization process in more dimensions and the resulting matrix-vector system see section 2.3.2, where the application to the steady convection-diffusion equation is given. Since the schemes for linear convection can be derived from those for non-linear convection, only the latter case is considered. As already stated in section 4.1, for simplicity homogeneous boundary conditions are considered.

The Galerkin weighted residual formulation of the EETG method (4.24) is given by

$$\begin{aligned} (\mathbf{u}^{m+1}, \mathbf{v}) &= (\mathbf{u}^{m}, \mathbf{v}) - \Delta s((\mathbf{u}^{m-\frac{1}{2}} \cdot \nabla)\mathbf{u}^{m}, \mathbf{v}) \\ &+ \frac{\Delta s^2}{2}((\mathbf{u}^{m} \cdot \nabla)(\mathbf{u}^{m} \cdot \nabla)\mathbf{u}^{m}, \mathbf{v}), \end{aligned} \tag{4.28}$$

with $\mathbf{v}$ the standard Galerkin test function. Application of Green's formula to the second-order term and substituting $\nabla \cdot \mathbf{u}^m = 0$ gives the variational or weak formulation

$$\begin{aligned} (\mathbf{u}^{m+1}, \mathbf{v}) &= (\mathbf{u}^{m}, \mathbf{v}) - \Delta s((\mathbf{u}^{m-\frac{1}{2}} \cdot \nabla)\mathbf{u}^{m}, \mathbf{v}) \\ &- \frac{\Delta s^2}{2}((\mathbf{u}^{m} \cdot \nabla)\mathbf{u}^{m}, (\mathbf{u}^{m} \cdot \nabla)\mathbf{v}). \end{aligned} \tag{4.29}$$

For non-homogeneous boundary conditions also a non-trivial boundary integral must be taken into account. Applying the spectral element discretization then leads to

$$\mathbf{u}^{m+1} = \mathbf{u}^{m} - \Delta s \mathbf{M}^{-1}\mathbf{C}^{m-\frac{1}{2}}\mathbf{u}^{m} - \frac{\Delta s^2}{2}\mathbf{M}^{-1}\mathbf{D}^{m}\mathbf{u}^{m}. \tag{4.30}$$

The matrix $\mathbf{C}^{m-\frac{1}{2}}$ is the convective matrix with velocity $\mathbf{u}^{m-\frac{1}{2}}$ (see equation (4.17)); the matrix $\mathbf{D}^m$ is a symmetric diffusion matrix arising in a natural way as part of the time-integration. The matrix $\mathbf{M}$ is the mass matrix, which is diagonal due to the Legendre Gauss–Lobatto quadrature. Consequently, the solution of equation (4.30) (and of every other subsequent system) does not involve the inversion of a matrix but only matrix-vector products which are evaluated in an element-by-element procedure. In general, for low-order methods the consistent mass matrix approach has advantages as regards the accuracy of the numerical scheme, see e.g. Gresho and Chan (1990). For high-order methods it is shown numerically by Timmermans and Van de Vosse (1993^b) that the use of a diagonal mass matrix approach is a valid approach with relatively little loss of accuracy, compared to low-order methods.

In an analogous way the spectral element discretization of the two-step EETG scheme (4.25) reads

$$\begin{aligned}\mathbf{u}^{m+\frac{1}{2}} &= \mathbf{u}^{m} - \frac{\Delta s}{2}\mathbf{M}^{-1}\mathbf{C}^{m}\mathbf{u}^{m},\\ \mathbf{u}^{m+1} &= \mathbf{u}^{m} - \Delta s\mathbf{M}^{-1}\mathbf{C}^{m+\frac{1}{2}}\mathbf{u}^{m+\frac{1}{2}}.\end{aligned} \tag{4.31}$$

And likewise for the extended stabilized two-step scheme (4.27)

$$\begin{aligned}\mathbf{u}^{m+\frac{1}{2}} &= \mathbf{u}^{m} - \frac{\Delta s}{2}\mathbf{M}^{-1}\mathbf{C}^{m}\mathbf{u}^{m},\\ \mathbf{u}^{m+1} &= \mathbf{u}^{m} - \Delta s\mathbf{M}^{-1}\mathbf{C}^{m+\frac{1}{2}}\mathbf{u}^{m} - \frac{\Delta s^2}{2}\mathbf{M}^{-1}\mathbf{D}^{m}\mathbf{u}^{m}.\end{aligned} \tag{4.32}$$

For the two-step scheme (4.31) no boundary integral arises in the case of non-homogeneous boundary conditions, due to the absence of the second-order term. This scheme is therefore not only the most efficient but also the most easiest to implement.

4.3.4 Numerical results

In this section the performance of the explicit Taylor-Galerkin methods and the spectral element approximation is tested by applying the numerical schemes to several more or less difficult convection problems, see also Timmermans *et al.* (1993). Firstly, the proposed schemes are compared by means of a one-dimensional linear test case, the convection of a Gaussian hill. As an example of one-dimensional non-linear convection, the Burgers problem is approximated. In more dimensions the efficiency of the scheme becomes rather important. Therefore the fast two-step EETG scheme is applied to several two-dimensional linear problems. The first one consists of an unsteady rotation of a Gaussian hill. Since this a smooth problem it can be expected that in the spectral element approximation better results can be obtained by increasing the degree of approximation n (p-convergence) than by increasing the number of elements n_e (h-convergence). The second problem consists of the rotation of a non-smooth cone. In that case some optimum has to be found between p- and h-convergence.

One-dimensional linear convection

Consider as a test case for the Taylor–Galerkin schemes for linear convection problems in one dimension the convection of a Gaussian hill described by

$$c(x,t) = e^{-\dfrac{(x-x_0-ut)^2}{2\sigma^2}}. \tag{4.33}$$

The initial hill ($t = 0$) is centered around $x_0 = 0.15$ and has a standard deviation of $\sigma = 0.04$. The hill is convected with constant velocity $u = 1$ and $t \in [0, 0.6]$.

For this problem the one-step EETG scheme and the two-step EETG scheme for linear convection are compared with a Crank–Nicolson time-integration. The spatial discretization is performed by a spectral element method using $n_e = 16$ elements of degree of approximation $n = 2, 4$ and 8. The discrete maximum error $\varepsilon = \|c - c_h\|_{\infty,gl}$ for these cases is given in Table 4.1. The exact solution and the approximation for $n_e = 16, n = 4$ for 256 time-steps is shown in Figure 4.1.

Table 4.1: Linear convection in one dimension. Discrete maximum error ε for the convection of a Gaussian hill. $n_e = 16$ elements with varying degree of approximation n. Two-step EETG scheme (TG2), one-step EETG scheme (TG1) or Crank–Nicolson scheme (CN).

method	n	number of time-steps			
		128	256	512	1024
TG2	2	$0.20 \cdot 10^0$	$0.21 \cdot 10^0$	$0.21 \cdot 10^0$	$0.21 \cdot 10^0$
	4	$0.44 \cdot 10^{-1}$	$0.10 \cdot 10^{-1}$	$0.90 \cdot 10^{-2}$	$0.91 \cdot 10^{-2}$
	8	unstable	unstable	$0.30 \cdot 10^{-2}$	$0.74 \cdot 10^{-3}$
TG1	2	$0.16 \cdot 10^0$	$0.19 \cdot 10^0$	$0.20 \cdot 10^0$	$0.21 \cdot 10^0$
	4	$0.47 \cdot 10^{-1}$	$0.10 \cdot 10^{-1}$	$0.77 \cdot 10^{-2}$	$0.84 \cdot 10^{-2}$
	8	unstable	$0.12 \cdot 10^{-1}$	$0.30 \cdot 10^{-2}$	$0.74 \cdot 10^{-3}$
CN	2	$0.22 \cdot 10^0$	$0.22 \cdot 10^0$	$0.21 \cdot 10^0$	$0.21 \cdot 10^0$
	4	$0.30 \cdot 10^{-1}$	$0.13 \cdot 10^{-1}$	$0.93 \cdot 10^{-2}$	$0.92 \cdot 10^{-2}$
	8	$0.24 \cdot 10^{-1}$	$0.59 \cdot 10^{-2}$	$0.15 \cdot 10^{-2}$	$0.37 \cdot 10^{-3}$

The results show that the one-step and two-step EETG method are comparable with respect to accuracy. Both schemes need more time-steps to become stable if the degree of approximation of the spatial discretization increases; the one-step scheme being a little more stable. Note also that the solution becomes much more accurate if the degree of approximation increases. The Crank–Nicolson scheme is only slightly more accurate. All

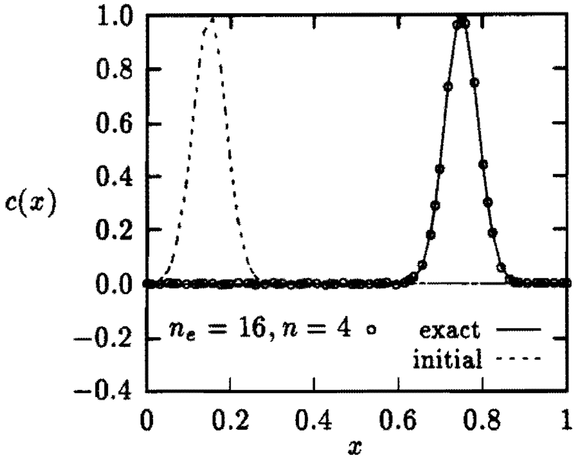


Figure 4.1: Linear convection in one dimension. Convection of a Gaussian hill. Exact solution and two-step EETG approximation for $n_e = 16, n = 4$ with 256 time-steps.

schemes show second-order accuracy if the degree of the approximation is large enough ($n = 8$); for the approximations $n = 2$ and $n = 4$ the error is determined by the spatial discretization. Taking into account that the explicit Taylor–Galerkin schemes require far less processing time, it is obvious that they are in fact preferable for this problem.

One-dimensional non-linear convection
Consider the one-dimensional non-linear Burgers equation (4.23) with domain is $\Omega = (0,4)$ and $t \in [0,2]$. The initial condition is given by

$$u(x,0) = g(x) = \begin{cases} c_0 - c_1 \cos(2\pi x), & 0 \le x \le 1, \\ c_0 - c_1, & \text{elsewhere}, \end{cases} \tag{4.34}$$

with $c_0 = 1, c_1 = 0.01$. The boundary conditions are given by

$$u(0,t) = u(4,t) = c_0 - c_1. \tag{4.35}$$

The exact solution to this problem is given by (Whitham, 1974)

$$u(x,t) = g(y), \quad x = y + u(g(y))t. \tag{4.36}$$

For this initial solution no shock arises in the given time-segment.

This non-linear is problem is solved with the explicit two-step EETG scheme and compared to a time-linearized Crank–Nicolson scheme. The latter scheme was in fact used to solve this problem in Timmermans and Van de Vosse (1993^b). Since the boundary conditions are non-homogeneous, for this case the two-step scheme is easier to implement than the one-step EETG scheme (4.30) and the extended two-step scheme (4.32) both of which involve the evaluation of a boundary integral. The spectral element method uses the same number of elements and degree of approximation as in the linear case. Since the solution only varies over an interval of 0.02, the numerical solution is verified with respect to the following relative discrete maximum error

$$\varepsilon_r = \frac{\|u - u_h\|_{\infty,gl}}{0.02}. \tag{4.37}$$

Table 4.2: Non-linear convection in one dimension. Relative discrete maximum error ε_r for the Burgers problem. $n_e = 16$ elements with varying degree of approximation n. Two-step EETG scheme (TG2) or Crank–Nicolson scheme (CN).

method	n	number of time-steps			
		128	256	512	1024
TG2	2	$0.92 \cdot 10^{-1}$	$0.99 \cdot 10^{-1}$	$0.10 \cdot 10^{0}$	$0.10 \cdot 10^{0}$
	4	$0.21 \cdot 10^{-1}$	$0.14 \cdot 10^{-1}$	$0.11 \cdot 10^{-1}$	$0.11 \cdot 10^{-1}$
	8	unstable	unstable	$0.33 \cdot 10^{-2}$	$0.16 \cdot 10^{-2}$
CN	2	$0.11 \cdot 10^{0}$	$0.10 \cdot 10^{0}$	$0.10 \cdot 10^{0}$	$0.10 \cdot 10^{0}$
	4	$0.22 \cdot 10^{-1}$	$0.15 \cdot 10^{-1}$	$0.12 \cdot 10^{-1}$	$0.12 \cdot 10^{-1}$
	8	$0.15 \cdot 10^{-1}$	$0.57 \cdot 10^{-2}$	$0.29 \cdot 10^{-2}$	$0.18 \cdot 10^{-2}$

The results for the relative error of the three different spectral element discretizations are shown in Table 4.2. Figure 4.2 shows the exact solution and the approximation for $n_e = 16, n = 4$ using 128 time-steps.

For non-linear convection the results are much the same as for the linear convection problem, although for $n = 8$ no clear second-order accuracy is achieved due to the non-linearity; for $n = 2$ and $n = 4$ the error is again determined by the spatial discretization. The two-step EETG scheme is

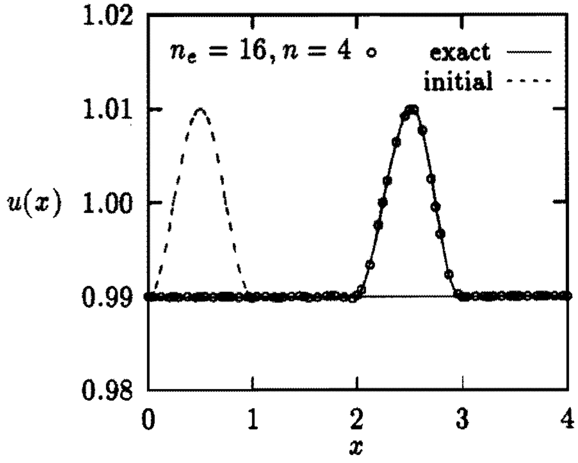


Figure 4.2: Non-linear convection in one dimension (Burgers problem). Exact solution and two-step EETG approximation for $n_e = 16, n = 4$ with 128 time-steps.

quite comparable in accuracy to the Crank–Nicolson scheme. Again, for an increasing degree of approximation the solution becomes much more accurate; but then also more time-steps are needed to obtain a stable numerical scheme. However, as was already stated in the linear convection case, due to the efficiency of the two-step scheme it is more suited for this problem than the Crank–Nicolson method.

Two-dimensional linear convection

In more dimensions the choice of the time-integration becomes more and more important with respect to efficiency. From the previous sections it appears that the two-step EETG scheme (4.31) is the most suitable for large more-dimensional problems. In order to check the performance of the two-step scheme, consider the unsteady rotation of a Gaussian hill described by the convection equation in two dimensions with domain $\Omega = (-1,1)\times(-1,1)$ and $t \in [0, 0.5]$. The time-dependent velocity is given by

$$\mathbf{u}(\mathbf{x},t) = [-\pi^2 \sin(2\pi t)x_2,\ \pi^2 \sin(2\pi t)x_1]^T. \tag{4.38}$$

The initial solution is given by

$$c(\mathbf{x},t) = 0.01^{4((x_1+\frac{1}{2})^2+x_2^2)}. \tag{4.39}$$

It represents a smooth Gaussian hill with height equal to 1 and with radius equal to $\frac{1}{4}$ centered at $(-\frac{1}{2}, 0)$. At $t = 0.5$ the hill is rotated halfway without diffusion, and therefore without loss of shape.

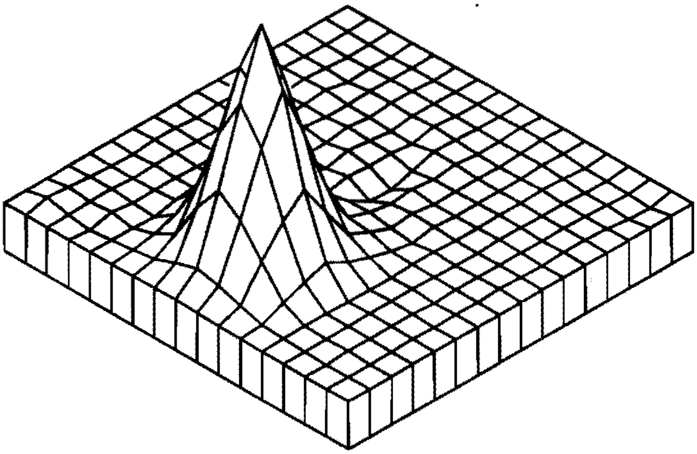
The problem is solved using the two-step EETG scheme. Two types of convergence are examined. To check the p-convergence the number of elements is kept fixed at $n_e = 4$; the degree of approximation is varying ($n = 4, 8, 12, 16$). To check the h-convergence the degree of approximation is kept fixed at $n = 2$ and the number of elements varies ($n_e = 16, 64, 144, 256$). The total number of degrees of freedom in the corresponding discretizations is the same. The results for the discrete maximum error $\varepsilon = \|c - c_h\|_{\infty,gl}$ for the first discretization are given in Table 4.3; for the second discretization they are found in Table 4.4.

Table 4.3: Linear convection in two dimensions. Discrete maximum error ε for the rotation of a Gaussian hill. Number of elements $n_e = 4$ fixed with varying degree of approximation n.

time-steps	$n = 4$	$n = 8$	$n = 12$	$n = 16$
256	$0.33 \cdot 10^{0}$	$0.67 \cdot 10^{-1}$	$0.17 \cdot 10^{-1}$	unstable
512	$0.33 \cdot 10^{0}$	$0.67 \cdot 10^{-1}$	$0.29 \cdot 10^{-2}$	$0.29 \cdot 10^{-2}$
1024	$0.33 \cdot 10^{0}$	$0.67 \cdot 10^{-1}$	$0.29 \cdot 10^{-2}$	$0.33 \cdot 10^{-3}$

Table 4.4: Linear convection in two dimensions. Discrete maximum error ε for the rotation of a Gaussian hill. Degree of approximation $n = 2$ fixed with varying number of elements n_e.

time-steps	$n_e = 16$	$n_e = 64$	$n_e = 144$	$n_e = 256$
256	$0.53 \cdot 10^{0}$	$0.18 \cdot 10^{0}$	$0.77 \cdot 10^{-1}$	$0.34 \cdot 10^{-1}$
512	$0.53 \cdot 10^{0}$	$0.19 \cdot 10^{0}$	$0.82 \cdot 10^{-1}$	$0.37 \cdot 10^{-1}$
1024	$0.53 \cdot 10^{0}$	$0.19 \cdot 10^{0}$	$0.82 \cdot 10^{-1}$	$0.38 \cdot 10^{-1}$



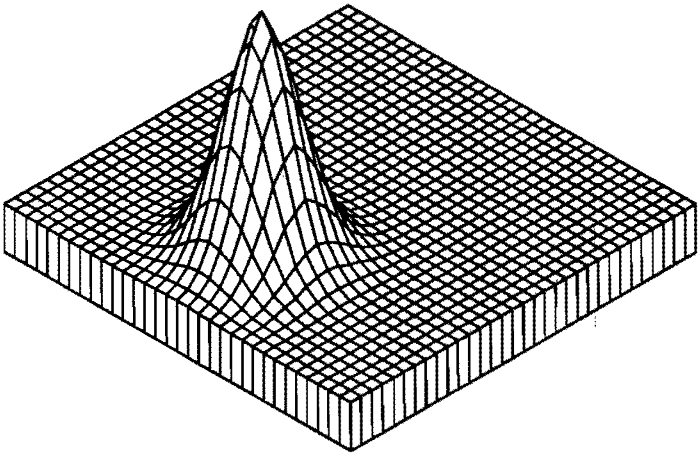


Figure 4.3: Linear convection in two dimensions. Unsteady rotation of a smooth Gaussian hill. Two-step EETG approximation using 1024 time-steps for $n_e = 4, n = 8$ (top) and for $n_e = 4, n = 16$ (bottom).

It is evident that the two-step scheme and the spectral element discretization perform very well for this smooth problem. The results of Table 4.3 show that the Gaussian hill is convected very accurately if the degree of the approximation n increases (p-convergence). These results are in good agreement with those for the steady rotation of a Gaussian hill presented by Timmermans and Van de Vosse (1993^b). From Table 4.4 it can be deduced that also h-convergence is obtained, the solution becomes more accurate if the number of elements n_e increases; the solutions obtained by increasing the degree of approximation however, are much more accurate. In Figure 4.3 (top) the solution for $n_e = 4, n = 8$ using 1024 time-steps is shown. There are still some 'wiggles' visible in this solution. Figure 4.3 (bottom) shows the solution for $n_e = 4, n = 16$ using 1024 time-steps. This approximation is convected in an extremely accurate way.

It is also interesting to observe how the two-step scheme and the spectral element discretization perform if the solution is no longer smooth. Consider again a linear convection problem in two dimensions with domain $\mathbf{\Omega} = (-1,1) \times (-1,1)$ and $t \in [0,1]$. The initial solution is given by

$$c(x,0) = \begin{cases} 1 - 4\sqrt{(x_1 - \frac{1}{2})^2 + x_2^2}, & (x_1 - \frac{1}{2})^2 + x_2^2 \le \frac{1}{16}, \\ 0, & (x_1 - \frac{1}{2})^2 + x_2^2 > \frac{1}{16}. \end{cases} \tag{4.40}$$

It represents a non-smooth cone with height equal to 1 and with radius equal to $\frac{1}{4}$ centered at $(\frac{1}{2}, 0)$. The velocity is now given by

$$\mathbf{u}(\mathbf{x},t) = [-2\pi x_2,\ 2\pi x_1]^T, \tag{4.41}$$

resulting in a steady rotation of the initial cone. As there is no diffusion present the end solution at $t = 1$ is exactly the same as the initial solution.

Again both p- and h-convergence are examined. The discretizations are the same as in the approximation of the Gaussian hill. The results for the first discretization are given in Table 4.5; for the second discretization they are found in Table 4.6.

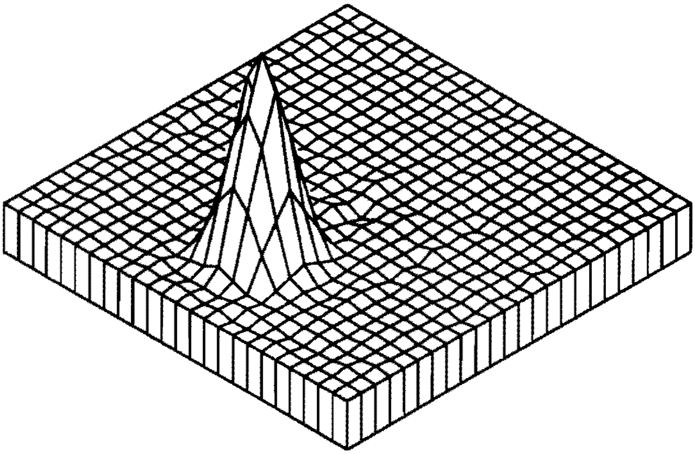
The results of Table 4.5 show that for this problem no clear p-convergence is obtained. This is of course due to the fact that the solution is non-smooth. However, good results are still obtained if the degree of approximation and the number of time-steps are large enough, as can be seen in Figure 4.4, where the solution is shown for $n_e = 4, n = 12$ using 1024 time-steps (top) and for $n_e = 4, n = 16$ using 2048 time-steps (bottom). As regards the results of Table 4.6, it is seen that no clear h-convergence is obtained also. The solution improves slightly if the number of elements is increased. With respect to accuracy and efficiency one may conclude that the high-order elements are preferable up to degree $n = 8$. For a higher degree of approximation the solution is indeed more accurate but also more expensive than the solutions obtained with elements of degree 2. Concluding it can be said that one has to look for an optimal combination of p- and h-convergence when approximating non-smooth problems. The spectral element method can then be a very useful tool.

Table 4.5: Linear convection in two dimensions. Discrete maximum error ε for the rotation of a cone. Number of elements $n_e = 4$ fixed with varying degree of approximation n.

time-steps	$n = 4$	$n = 8$	$n = 12$	$n = 16$
256	$0.58 \cdot 10^0$	$0.30 \cdot 10^0$	unstable	unstable
512	$0.58 \cdot 10^0$	$0.23 \cdot 10^0$	$0.32 \cdot 10^1$	unstable
1024	$0.58 \cdot 10^0$	$0.21 \cdot 10^0$	$0.17 \cdot 10^0$	$0.38 \cdot 10^1$
2048	$0.58 \cdot 10^0$	$0.21 \cdot 10^0$	$0.17 \cdot 10^0$	$0.97 \cdot 10^{-1}$

Table 4.6: Linear convection in two dimensions. Discrete maximum error ε for the rotation of a cone. Degree of approximation $n = 2$ fixed with varying number of elements n_e.

time-steps	$n_e = 16$	$n_e = 64$	$n_e = 144$	$n_e = 256$
256	$0.11 \cdot 10^1$	$0.54 \cdot 10^0$	$0.26 \cdot 10^0$	unstable
512	$0.11 \cdot 10^1$	$0.54 \cdot 10^0$	$0.29 \cdot 10^0$	$0.14 \cdot 10^0$
1024	$0.11 \cdot 10^1$	$0.54 \cdot 10^0$	$0.29 \cdot 10^0$	$0.15 \cdot 10^0$
2048	$0.11 \cdot 10^1$	$0.54 \cdot 10^0$	$0.29 \cdot 10^0$	$0.15 \cdot 10^0$



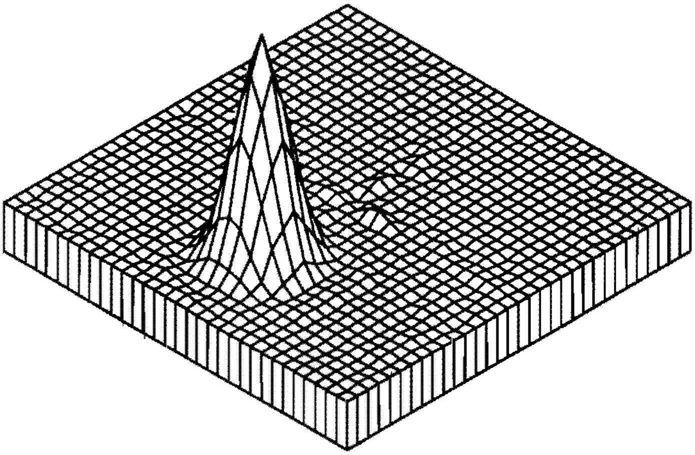


Figure 4.4: Linear convection in two dimensions. Steady rotation of a cone. Two-step EETG approximation using 1024 time-steps for $n_e = 4, n = 12$ (top) and using 2048 time-steps for $n_e = 4, n = 16$ (bottom).

4.4 Operator splitting for convection-diffusion

4.4.1 The time-integration scheme

Combining the semi-implicit backward differences schemes for the diffusion operator with the explicit Taylor–Galerkin time-integration of the previous section, a full operator splitting scheme for convection-diffusion problems can be derived, based on the theory of section 4.2. As an example the two-step Taylor–Galerkin method is chosen for the associated convection problem (4.9).

According to equation (4.7) the diffusion step reads (again only the non-linear case is considered)

$$\frac{\beta_0 \mathbf{u}^{n+1} - \sum_{i=1}^{k_1} \beta_i \mathcal{Q}_C^{(t^{n+1},t^{n+1-i})} \mathbf{u}^{n+1-i}}{\Delta t} = \mathcal{D}\mathbf{u}^{n+1} + \mathbf{f}^{n+1}. \tag{4.42}$$

The terms $\mathcal{Q}^{(t^{n+1-i},t^n)}\mathbf{u}^n (i = 1, 2, \ldots)$ are calculated according to the two-step explicit EETG scheme, with initial condition $\tilde{\mathbf{u}}^0 = \mathbf{u}^{n+1-i}$ using a time-step Δs such that $\Delta t = j\Delta s$ with j an integer. The semi-discrete convection step then reads

$$\begin{aligned} \tilde{\mathbf{u}}^{m+\frac{1}{2}} &= \tilde{\mathbf{u}}^m - \frac{\Delta s}{2}(\tilde{\mathbf{u}}^m \cdot \nabla)\tilde{\mathbf{u}}^m, \\ \tilde{\mathbf{u}}^{m+1} &= \tilde{\mathbf{u}}^m - \Delta s(\tilde{\mathbf{u}}^{m+\frac{1}{2}} \cdot \nabla)\tilde{\mathbf{u}}^{m+\frac{1}{2}}. \end{aligned} \tag{4.43}$$

Analogous to equation (4.10) it then follows that

$$\tilde{\mathbf{u}}^{n+1-i} = \tilde{\mathbf{u}}^{i(j+1)}, \tag{4.44}$$

where the simpler notation $\tilde{\mathbf{u}}^{n+1-i} = \mathcal{Q}^{(t^{n+1-i},t^n)}\mathbf{u}^{n+1-i}$ is introduced. The above splitting scheme for convection-diffusion problems is the basis for the treatment of the incompressible Navier–Stokes equations described in chapter 5.

4.4.2 Spectral element discretization

The application of a Galerkin spectral element discretization to the splitting scheme of the previous section is straightforward. The weighted residual formulation of the diffusion step is given by (again homogeneous boundary conditions are assumed)

$$\left(\beta_0 \mathbf{u}^{n+1} - \sum_{i=1}^{k_1} \beta_i \tilde{\mathbf{u}}^{n+1-i}, \mathbf{v}\right) = \Delta t \left(\mathcal{D}\mathbf{u}^{n+1} + \mathbf{f}^{n+1}, \mathbf{v}\right), \tag{4.45}$$

leading to

$$\mathbf{M}\left(\beta_0 \mathbf{u}^{n+1} - \sum_{i=1}^{k_1} \beta_i \tilde{\mathbf{u}}^{n+1-i}\right) = \Delta t \left(-\mathbf{D}\mathbf{u}^{n+1} + \mathbf{M}\mathbf{f}^{n+1}\right). \tag{4.46}$$

The terms $\tilde{\mathbf{u}}^{n+1-i}$ are evaluated through

$$\begin{aligned} \tilde{\mathbf{u}}^{m+\frac{1}{2}} &= \tilde{\mathbf{u}}^{m} - \frac{\Delta s}{2}\mathbf{M}^{-1}\mathbf{C}^{m}\tilde{\mathbf{u}}^{m}, \\ \tilde{\mathbf{u}}^{m+1} &= \tilde{\mathbf{u}}^{m} - \Delta s\mathbf{M}^{-1}\mathbf{C}^{m+\frac{1}{2}}\tilde{\mathbf{u}}^{m+\frac{1}{2}}, \end{aligned} \tag{4.47}$$

with initial condition $\tilde{\mathbf{u}}^0 = \mathbf{u}^{n+1-i}$.

Remark
If a Crank–Nicolson scheme, see equation (4.11), is used to treat the diffusion step, the discrete Galerkin system for this step becomes

$$\mathbf{M}(\mathbf{u}^{n+1} - \tilde{\mathbf{u}}^{n}) = \frac{1}{2}\Delta t \left(-\mathbf{D}\mathbf{u}^{n+1} + \mathbf{M}\mathbf{f}^{n+1}\right) + \frac{1}{2}\Delta t\tilde{\mathbf{d}}^{n}, \tag{4.48}$$

where

$$\begin{aligned} \tilde{\mathbf{u}}^{n} &= \mathcal{Q}_C^{(t^{n+1},t^n)}\mathbf{u}^{n}, \\ \tilde{\mathbf{d}}^{n} &= \mathcal{Q}_C^{(t^{n+1},t^n)}(-\mathbf{D}\mathbf{u}^{n} + \mathbf{M}\mathbf{f}^{n}). \end{aligned} \tag{4.49}$$

The term $\tilde{\mathbf{u}}^n$ is evaluated through solution of the two-step scheme (4.47) with initial condition $\tilde{\mathbf{u}}^0 = \mathbf{u}^n$. The term $\tilde{\mathbf{d}}^n$ is is evaluated according to the two-step scheme

$$\begin{aligned} \tilde{\mathbf{d}}^{m+\frac{1}{2}} &= \tilde{\mathbf{d}}^{m} - \frac{\Delta s}{2}\mathbf{M}^{-1}\mathbf{C}^{m}\tilde{\mathbf{d}}^{m}, \\ \tilde{\mathbf{d}}^{m+1} &= \tilde{\mathbf{d}}^{m} - \Delta s\mathbf{M}^{-1}\mathbf{C}^{m+\frac{1}{2}}\tilde{\mathbf{d}}^{m+\frac{1}{2}}. \end{aligned} \tag{4.50}$$

The initial condition to equation (4.50) reads

$$\tilde{\mathbf{d}}^0 = \mathbf{d}^n = -\mathbf{M}^{-1}\mathbf{D}\mathbf{u}^n + \mathbf{f}^n. \tag{4.51}$$

If the term $\tilde{\mathbf{d}}^n$ in equation (4.48) is replaced by $\mathbf{d}^n$, such as is the case in splitting methods based on the classical splitting approach, the second-order accuracy of the diffusion step is lost. This has been found empirically by Maday *et al.* (1990) and will be shown also in the next section. Note that in the case of a backward differences scheme there is no need to convect the diffusion operator itself, thus prohibiting 'expensive' evaluations like (4.51).

4.4.3 Numerical results

In order to test the performance of the operator splitting approach, in this section a one-dimensional convection-diffusion problem is solved using an implicit backward differences time-integration for the diffusion step and the explicit two-step EETG scheme (4.31) for the convection step.

Consider as a test case for the operator splitting scheme the problem of a Gaussian hill in one dimension travelling with a constant velocity $u = 1$ and spreading isotropically with a viscosity $\eta = \frac{1}{200}$. This problem is adapted from a test case in Donea *et al.* (1984). The exact solution has the form

$$c(x,t) = \frac{\sigma(0)}{\sigma(t)} e^{-\frac{(x - x_0 - ut)^2}{2\sigma(t)^2}}, \tag{4.52}$$

with the time-dependent standard deviation given by

$$\sigma(t) = \sqrt{\sigma(0) + 2\eta t}. \tag{4.53}$$

The initial hill at $t = 0$ is centered around $x_0 = 0.15$ and has a standard deviation of $\sigma(0) = 0.04$. The hill is convected with constant velocity $u = 1$ and $t \in [0, 0.3]$.

This problem is solved using the operator splitting scheme described above for both a first-order and a second-order backward differences (BDF) scheme and for a Crank–Nicolson scheme. Moreover, it is checked whether the second-order accuracy of the Crank–Nicolson splitting scheme is lost if the term $\tilde{\mathbf{d}}^n$ in equation (4.48) is replaced by $\mathbf{d}^n$. This approach shall be indicated with 'classical' Crank–Nicolson. The number of time-steps for the associated convection problem is equal to 64. The spectral element discretization uses $n_e = 16$ elements with degree of approximation $n = 4$. The discrete maximum error is given in Table 4.7. Figure 4.5 shows the exact solution and the approximation for $n_e = 16, n = 4$ using a second-order backward differences scheme with 4 diffusion time-steps.

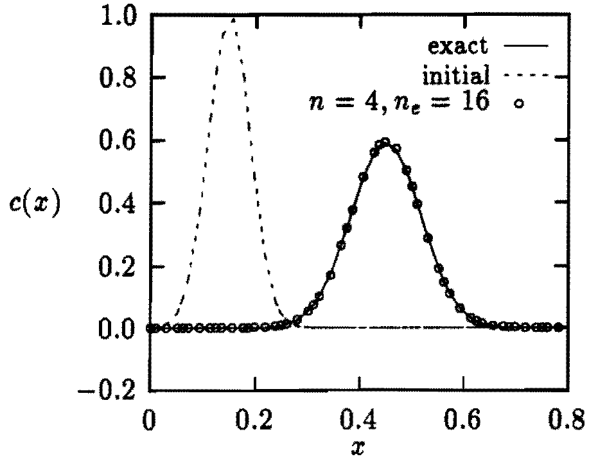


Figure 4.5: Linear convection-diffusion in one dimension. Convection and diffusion of a Gaussian hill. Exact solution and two-step EETG/second-order BDF approximation for $n_e = 16, n = 4$ with 4 diffusion steps containing 64 convection steps each.

Table 4.7: Linear convection-diffusion in one dimension. Discrete maximum error ε for the convection and diffusion of a Gaussian hill. $n_e = 16$ elements of degree $n = 4$. First-order BDF scheme (BDF1), second-order BDF scheme (BDF2), Crank–Nicolson scheme (CN) or 'classical' Crank–Nicolson (CCN).

diff. steps	BDF1	BDF2	CN	CCN
2	$0.42 \cdot 10^{-1}$	$0.24 \cdot 10^{-1}$	$0.58 \cdot 10^{-2}$	$0.23 \cdot 10^{0}$
4	$0.22 \cdot 10^{-1}$	$0.39 \cdot 10^{-2}$	$0.16 \cdot 10^{-2}$	$0.26 \cdot 10^{0}$
8	$0.11 \cdot 10^{-1}$	$0.87 \cdot 10^{-3}$	$0.42 \cdot 10^{-3}$	$0.11 \cdot 10^{0}$
16	$0.58 \cdot 10^{-2}$	$0.31 \cdot 10^{-3}$	$0.24 \cdot 10^{-3}$	$0.43 \cdot 10^{-1}$
32	$0.30 \cdot 10^{-2}$	$0.17 \cdot 10^{-3}$	$0.17 \cdot 10^{-3}$	$0.20 \cdot 10^{-1}$

The performance of the operator splitting scheme is quite good. Only very few expensive diffusion steps are needed to obtain accurate solutions. It can also be seen that the backward differences schemes and the Crank–Nicolson scheme achieve the theoretical order of accuracy for sufficient diffusion steps. The performance of the 'classical' Crank–Nicolson approach is very bad compared to the other results. For the small number of diffusion steps that are needed to obtain accuracy for the other schemes, the solution is not very accurate. The large number of convection steps in each diffusion cycle does not require much extra processing time, since each convection step is solved explicitly.

4.5 Conclusions

In this chapter a closer look has been taken at the application of a Galerkin high-order spectral element method to convection-diffusion problems with a time-dependent and divergence-free velocity field. In order to decouple the treatment of the convection operator and the stiff diffusion operator, an operator splitting integrating factor approach has been presented. The decoupled problems are then solved with suitable time-integrations using, if necessary, different time-steps.

For pure convection problems explicit time-integration is virtually necessary to obtain an efficient numerical scheme; especially in the case that the problem is non-linear. A possible strategy to stabilize the numerical oscillations that often occur in the solution of convection dominated problems, is to make use of so-called Taylor–Galerkin methods. Several explicit second-order schemes have been proposed and analyzed for both linear and non-linear convection problems with time-dependent and divergence-free velocity. It should be noted that the second-order accuracy of the time-integration can only be obtained if the degree of approximation of the spatial discretization is large enough for the spatial error to be negligible with respect to the time error. The schemes have been tested by means of several model test cases. With respect to accuracy the Taylor–Galerkin schemes are quite comparable to an implicit Crank–Nicolson scheme; as regards efficiency, the explicit schemes are much faster than the implicit time-integration. Also for non-linear and for more-dimensional problems the Taylor–Galerkin schemes appear to be very well suited.

For convection-diffusion problems the operator splitting technique gives good results. The splitting scheme has been applied to a one-dimensional test case. The convection part is treated with a suitable Taylor–Galerkin scheme. The diffusion or stiff part of the equation can be solved with a second-order (semi-)implicit backward differences scheme or with a trapezoidal method (e.g. Crank–Nicolson), the former of the two being the more efficient one. Only very few diffusion steps are needed to obtain a second-order accurate splitting scheme. The number of convection steps can be taken relatively large, since the Taylor–Galerkin schemes are explicit.

Finally, with respect to the spectral element spatial discretization it can be concluded that if the number of time-steps is large enough a clear p-convergence for smooth problems to the exact solution is seen. In the operator splitting technique here proposed the larger number of time-steps required for a higher-order spectral element approximation is not a problem with respect to efficiency. The number of 'expensive' diffusion steps is very low compared to the number of 'cheap' convection steps. Moreover, since in a high-order method the use of a diagonal mass matrix is not as restrictive with respect to accuracy than in the case of a low-order method, the explicit convection steps only involve matrix-vector products and not the inversion of a matrix to solve the system. If the problem is no longer smooth, an optimum has to be found between p- and h-convergence. In that case the spectral element method is a very useful tool, since both types of convergence can be realized by it.

Chapter 5

Solution of Navier–Stokes problems

5.1 Introduction

The solution of the Navier–Stokes equations for unsteady incompressible fluid flow is still a major challenge in the field of computational fluid dynamics. The aim of this chapter is to derive not only a solution algorithm that guarantees high accuracy, but also high efficiency through the use of iterative techniques. This means that it must be possible to incorporate the methods described in the previous chapters. As already stated in chapter 4, the Navier–Stokes equations form a specific case of general unsteady convection-diffusion problems for divergence-free velocity fields. The operator splitting approach for these problems will also prove to be very useful for the Navier–Stokes equations. Then also it is possible to derive the solution algorithm in such a way that the iterative finite element preconditioning techniques of chapter 3 can be applied to the resulting system of discrete equations.

An overview of the most important aspects with respect to the solution of the incompressible Navier–Stokes equations can be found in Fortin and Glowinski (1983), Peyret and Taylor (1983), Temam (1984), Girault and Raviart (1986) and Gresho (1991). A numerical solution algorithm for these equations faces several difficulties. Firstly, the Navier–Stokes equations consist not only of the momentum equation but also of the continuity equation (or conservation of mass). For incompressible flow the latter equation reduces to the incompressibility constraint, requiring that the velocity is divergence-free instantaneously in the whole computational domain. Secondly, the Navier–Stokes equations form a set of coupled equations for both velocity and pressure (or better the gradient of the pressure). The pressure is not a thermodynamic variable as there is no equation of state for an incompressible fluid. It is an implicit variable which instantaneously 'adjusts itself' in such a way that the velocity remains divergence-free. The gradient of the pressure on the other hand is a relevant physical quantity: a force per unit volume. The mathematical importance of the pressure in an incompressible flow lies in the theory of saddle-point problems (of which the steady Stokes equations are an example), where it acts as a Lagrangian multiplier that constrains the velocity to remain divergence-free (Fortin and Glowinski, 1983; Girault and Raviart, 1986).

There are numerous approaches to solve the Navier–Stokes equations. The set of equations can be solved directly in the coupled form (Bathe and Dong, 1987). However, an important disadvantage is that the system of equations becomes very large. Consequently, a lot of computing time is required. Moreover, since there are no pressure degrees of freedom in the continuity equation, the system matrix contains zeros on the main diagonal, which makes partial pivoting necessary, by which also the banded structure of the discrete system is distorted. Again, this causes an increase in computing time. Therefore it is preferable to adopt a solution procedure that decouples the treatment of velocity and pressure. One approach is to apply a penalty function method to the discrete Navier–Stokes system (Carey and Krishnan, 1984; Cuvelier *et al.*, 1986). In a penalty function method the continuity equation is perturbed by a small parameter times the pressure. Next the pressure can be eliminated from the momentum equation, thus decoupling the momentum equation and the continuity equation. A disadvantage of the penalty function method is that the perturbation parameter must be chosen carefully. It must be taken small enough for the approximation of the continuity equation to be accurate enough. Unfortunately, this results in an ill-conditioned system matrix, precluding the use of iterative techniques. These restrictions can partly be taken away if iterative versions of the penalty function method are applied, such as the often used Uzawa algorithm (Girault and Raviart, 1986; Maday *et al.*, 1993). However, in general these techniques seem to be more suited for steady Navier–Stokes computations, since for unsteady problems the system matrix again becomes ill-conditioned due to the contribution of the unsteady inertia forces.

For the solution of unsteady Navier–Stokes flow perhaps one of the most successful approaches to-date is provided by the class of projection methods

(Chorin, 1968; Gresho, 1990). In a projection method the velocity and pressure are decoupled by taking the divergence of the momentum equation. This results in a general convection-diffusion equation for the velocity that can be solved using the operator splitting approach of chapter 4, and a Poisson equation for the pressure that can be solved using a finite element preconditioning technique described in chapter 3. While the pressure is well-defined up to an arbitrary constant by the original equations, it is less so when directly expressed in terms of a Poisson equation. This is because in the latter case the necessity arises to formulate a non-trivial boundary condition for the pressure. The choice of the pressure boundary condition is an aspect that is much discussed in literature, see e.g. Orszag *et al.* (1986), Gresho and Sani (1987) and Karniadakis *et al.* (1991). The obvious theoretical choice for the pressure boundary condition is a Neumann condition derived from the normal component of the momentum equation. The form in which this boundary condition is implemented is important not only because of the overall accuracy, but also because of the efficiency of the numerical scheme. This aspect is still very much open for improvement.

In this chapter a pressure correction method, a particular application of the projection method, is used to decouple the velocity and pressure treatment. The technique presented here follows the ideas of Van Kan (1986) who, for finite difference methods, first presented a second-order pressure correction scheme. In the finite element context a related scheme is presented by Hawken *et al.* (1990). In this thesis a modification to the algorithm is suggested. It will be shown that the solution of the decoupled equations coincides with that of the original equations. Section 5.2 presents the governing equations for incompressible fluid flow: the Navier–Stokes equations. In section 5.3 the solution method for these equations is discussed. Before presenting the algorithm, the choice of solution method is explained. In order to discuss the theoretical background of the pressure correction scheme, it is first applied to the Stokes equations. For the Navier–Stokes problem the equations are first split according to an operator splitting procedure similar to the approach described in chapter 4 for convection-diffusion problems, including the pressure term temporarily in the viscous part of the equations. This is discussed in section 5.4. Next, the velocity and pressure treatment is decoupled by applying the pressure correction algorithm. The properties of the algorithm are extensively analyzed by means of an analytical test case. In section 5.5 more numerical results are presented for the problem of a buoyancy-driven flow in an enclosed cavity. Finally, in section 5.6 conclusions are drawn.

5.2 Governing equations

In this section two-dimensional incompressible transient Newtonian flow is considered, without thermal effects. The Navier–Stokes equations are then given by the momentum equation

$$\rho\frac{\partial \mathbf{u}}{\partial t}+\rho(\mathbf{u}\cdot\nabla)\mathbf{u}-\nabla\cdot\boldsymbol{\sigma}=\rho\mathbf{f} \tag{5.1}$$

and the continuity equation, which implies that the flow is incompressible always and everywhere

$$\nabla\cdot\mathbf{u}=0. \tag{5.2}$$

For a detailed derivation of these equations, see e.g. Batchelor (1967). In these equations $\mathbf{u}(\mathbf{x},t)=(u_1,u_2)^T$ is the velocity vector, $\mathbf{f}(\mathbf{x},t)$ the body force vector, ρ the fluid density and $\boldsymbol{\sigma}(\mathbf{x},t)$ the Cauchy stress tensor with components

$$\sigma_{ij}=-p\delta_{ij}+\eta\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right),\qquad i,j=1,2. \tag{5.3}$$

Here $p(\mathbf{x},t)$ is the pressure and η is the dynamic viscosity.

Consider the equations (5.1) and (5.2) in an open and bounded domain $\Omega\subset\mathbf{R}^d$ with boundary Γ for $t\geq 0$. In order for the problem to be well-posed boundary and initial conditions have to be imposed. With respect to boundary conditions it is necessary to prescribe either velocity (essential boundary conditions) or surface traction force (natural boundary conditions) in the normal and tangential direction (Temam, 1984; Gresho and Sani, 1987). Suppose for convenience that the boundary Γ is composed of two non-overlapping parts $\Gamma_{\mathbf{u}}$ and $\Gamma_{\boldsymbol{\sigma}}$ and assume that on each part either the velocity or the stress is prescribed. The boundary conditions can then be formulated as

$$\mathbf{u}=\mathbf{k},\qquad \mathbf{x}\text{ on }\Gamma_{\mathbf{u}},\ t\geq 0, \tag{5.4}$$

$$\boldsymbol{\sigma}\cdot\mathbf{n}=\mathbf{h},\qquad \mathbf{x}\text{ on }\Gamma_{\boldsymbol{\sigma}},\ t\geq 0, \tag{5.5}$$

where $\mathbf{n}$ is the outward unit normal. Using equation (5.3) the boundary condition (5.5) can also be written as

$$\left\{\begin{aligned} h_n&=-p+2\eta\frac{\partial u_n}{\partial n}, &&\mathbf{x}\text{ on }\Gamma_{\boldsymbol{\sigma}},\\ h_\tau&=\eta\left(\frac{\partial u_n}{\partial \tau}+\frac{\partial u_\tau}{\partial n}\right), &&\mathbf{x}\text{ on }\Gamma_{\boldsymbol{\sigma}},\end{aligned}\right. \tag{5.6}$$

with h_n, u_n the normal components and h_τ, u_τ the tangential components of the stress and the velocity respectively.

In general the application of pressure boundary conditions is inconsistent with equation (5.2). In an incompressible flow the pressure is an implicit variable whose value is (pointwise) determined by the requirement that the incompressibility constraint $\nabla \cdot \mathbf{u} = 0$ is satisfied there. The pressure can be prescribed in an indirect way via boundary condition (5.5). If however u_n is specified on all of $\boldsymbol{\Gamma}$, thus prohibiting the boundary condition (5.5), the pressure is only obtainable up to an arbitrary additive constant. In that case one can further require that

$$\int_{\boldsymbol{\Omega}} p \, d\boldsymbol{\Omega} = 0. \tag{5.7}$$

Finally, initial conditions have to be imposed on the system. These read

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0, \qquad \mathbf{x} \text{ in } \overline{\boldsymbol{\Omega}}, \tag{5.8}$$

where $\mathbf{u}_0$ must satisfy

$$\nabla \cdot \mathbf{u}_0 = 0, \qquad \mathbf{x} \text{ in } \overline{\boldsymbol{\Omega}}, \tag{5.9}$$

$$\mathbf{n} \cdot \mathbf{u}_0 = \mathbf{n} \cdot \mathbf{k}(\mathbf{x}, 0), \qquad \mathbf{x} \text{ on } \boldsymbol{\Gamma}_{\mathbf{u}}. \tag{5.10}$$

If either (5.9) or (5.10) is omitted, the problem (5.1)–(5.2) is ill-posed, see e.g. Temam (1984).

Substitution of the Cauchy stress tensor according to equation (5.3) into equation (5.1) and dividing by ρ yields for the momentum equation

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - (\nabla \cdot \nu \nabla)\mathbf{u} + \nabla p = \mathbf{f}, \tag{5.11}$$

in which p denotes the kinematic pressure (pressure divided by density) and ν the kinematic viscosity. The complete set of Navier–Stokes equations for incompressible flow can thus be written as

$$\left\{ \begin{array}{ll} \dfrac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - (\nabla \cdot \nu \nabla)\mathbf{u} + \nabla p = \mathbf{f} & \text{in } \boldsymbol{\Omega}, \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \overline{\boldsymbol{\Omega}}, \\ \mathbf{u} = \mathbf{k} & \text{on } \boldsymbol{\Gamma}_{\mathbf{u}}, \\ \boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{h} & \text{on } \boldsymbol{\Gamma}_{\boldsymbol{\sigma}}, \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0 & \text{in } \overline{\boldsymbol{\Omega}}, \end{array} \right. \tag{5.12}$$

with $\mathbf{u}_0$ such that equations (5.9) and (5.10) are satisfied.

5.3 Solution of the Stokes equations

5.3.1 Projection methods

Projection methods, first proposed by Chorin (1968), have been developed as a useful way for obtaining an efficient solution algorithm for unsteady incompressible flow. By decoupling the treatment of velocity and pressure terms, a set of easier to solve equations arises: a convection-diffusion problem for the velocity, yielding an intermediate velocity, and a Poisson equation for the pressure. There are essentially two approaches for projection methods: pressure correction methods and fractional step methods.

Pressure correction methods (Van Kan, 1986; Hawken *et al.*, 1990) consist of a basic predictor-corrector procedure between the velocity and the pressure fields. Using an initial approximation of the pressure, the momentum equation can be solved to obtain an intermediate velocity field. This velocity in general does not satisfy the divergence-free constraint and must therefore be corrected. Since this correction has an impact on the pressure field, a related pressure correction is defined, obtained by enforcing that the corrected velocity satisfies the continuity equation. This leads to a Poisson equation for the pressure correction. The Poisson equation for the pressure correction is solved with Neumann boundary conditions on that part of the boundary where velocities are prescribed. This can be realized by taking the normal component of the momentum equation. On that part of the boundary where stress is prescribed, a Dirichlet boundary condition for the pressure correction must be given.

The fractional step method (Donea *et al.*, 1982; Laval and Quartapelle, 1990; Karniadakis *et al.*, 1991) is based on a full splitting of the treatment of the pressure/incompressibility constraint and the diffusion in different substeps. The intermediate step leads to a Poisson equation for the pressure at the new time-level. The remark on boundary conditions also holds here, but now of course for the pressure itself, instead of the pressure correction. An important difference with the pressure correction approach is that the divergence-free constraint is imposed on the intermediate velocity field, not on the velocity field on the new time-level, which is therefore not guaranteed to satisfy the continuity equation.

The solution algorithms mentioned above can be applied either to the continuous or the discrete system of equations. In the latter case, the procedure does not involve a rediscretization of the original equations. Consequently, the boundary conditions are then already built in directly in the weak or variational formulation, thereby prohibiting the need to formulate a specific boundary condition for the discrete pressure 'Poisson' equation. In this case the choice of the element for the velocity and the pressure is important with respect to the well-posedness of the system. As is well-known from the theory of saddle-point problems, a discrete form of the Brezzi–Babuška condition (Babuška, 1971; Brezzi, 1974), must then be satisfied for obtaining a unique velocity and pressure. For a high-order spectral element approximation this means that the degree of approximation for the pressure must be taken two degrees lower than that of the velocity (Maday *et al.*, 1987).

On the other hand, applying the decoupling procedure to the continuous equations leads to a more straightforward scheme, since in that case the original problem is reformulated into several new (and simpler) problems. The theory of saddle-point problems is then no longer applicable; as a consequence the degree of approximation for velocity and pressure can be taken the same, yielding a simpler to implement numerical scheme. Moreover, then also the iterative finite element preconditioning techniques described in chapter 3 can easily be applied to the resulting discrete equations. In that case however, the Poisson equation for the pressure (correction) requires a non-trivial boundary condition. In this thesis the continuous approach is chosen as it can be applied more easily within the framework of the operator splitting approach described in chapter 4. Moreover, it will be shown that the use of a homogeneous Neumann boundary condition for the pressure correction Poisson equation is essential in obtaining a divergence-free velocity field. For the standard pressure correction approach this leads to problems, since then the pressure is not allowed to change in a boundary layer. Here, a modification to the standard approach is proposed in which this problem is circumvented.

5.3.2 The pressure correction method

In a pressure correction algorithm an intermediate velocity is computed by taking the pressure at the previous time-level. This intermediate velocity is in general not divergence-free. By taking the divergence of the momentum equation and enforcing the incompressibility constraint (5.2), a Poisson equation for the pressure correction (the difference between the new and the old pressure) is obtained. Using the pressure correction the new (divergence-free) velocity and the new pressure can then be computed. In this section the properties of a pressure correction method are analyzed by means of its application to the linear Stokes problem. Firstly, the algorithm similar to the one used in Hawken *et al.* (1990) is deduced; next, a modification and its consequences are discussed.

Consider the unsteady Stokes equations with, for simplicity, only essential boundary conditions

$$\begin{cases} \dfrac{\partial \mathbf{u}}{\partial t} - \nu\nabla^2\mathbf{u} + \nabla p = \mathbf{f} & \text{in } \mathbf{\Omega}, \\ \nabla\cdot\mathbf{u} = 0 & \text{in } \overline{\mathbf{\Omega}}, \\ \mathbf{u} = \mathbf{k} & \text{on } \mathbf{\Gamma}, \\ \mathbf{u}(\mathbf{x},0) = \mathbf{u}_0 & \text{in } \overline{\mathbf{\Omega}}, \end{cases} \tag{5.13}$$

where for convenience it is assumed that the viscosity ν is constant. If this is not the case, e.g. for generalized Newtonian or non-Newtonian flows, the theory becomes more complicated but still applicable. The momentum equation can be written in the following form

$$\frac{\partial \mathbf{u}}{\partial t} = \mathcal{D}\mathbf{u} - \nabla p + \mathbf{f}, \tag{5.14}$$

with $\mathcal{D} = \nu\nabla^2$ the diffusion operator. Equation (5.14) is integrated using an implicit backward differences scheme with time-step Δt. This yields the following semi-discrete system

$$\frac{\beta_0\mathbf{u}^{n+1} - \sum_{i=1}^{k}\beta_i\mathbf{u}^{n+1-i}}{\Delta t} = \mathcal{D}\mathbf{u}^{n+1} - \nabla p^{n+1} + \mathbf{f}^{n+1}. \tag{5.15}$$

Again, the approximation of a quantity at time $t^{n+1} = (n+1)\Delta t$ is denoted by the superscript $n+1$. The coefficients β_i of the backward differences schemes are listed in appendix D. For a second-order backward differences scheme, the following semi-discrete system results

$$\begin{cases} \dfrac{3}{2}\mathbf{u}^{n+1} - \Delta t\mathcal{D}\mathbf{u}^{n+1} = 2\mathbf{u}^{n} - \dfrac{1}{2}\mathbf{u}^{n-1} & \\ \qquad -\Delta t\nabla p^{n+1} + \Delta t\mathbf{f}^{n+1} & \text{in } \mathbf{\Omega}, \\ \nabla\cdot\mathbf{u}^{n+1} = 0 & \text{in } \overline{\mathbf{\Omega}}, \\ \mathbf{u}^{n+1} = \mathbf{k}^{n+1} & \text{on } \mathbf{\Gamma}. \end{cases} \tag{5.16}$$

The pressure correction scheme proceeds as follows:

- Calculate an intermediate velocity field $\mathbf{u}^*$ by choosing the pressure at the previous time-level

$$\frac{3}{2}\mathbf{u}^* - \Delta t\mathcal{D}\mathbf{u}^* = 2\mathbf{u}^n - \frac{1}{2}\mathbf{u}^{n-1} - \Delta t\nabla p^n + \Delta t\mathbf{f}^{n+1}. \tag{5.17}$$

  The intermediate velocity field $\mathbf{u}^*$ is in general not divergence-free.

- Calculate the pressure correction $p^* = p^{n+1} - p^n$ by subtracting the equation for $\mathbf{u}^*$ from the original equation (5.16). This yields

$$\frac{3}{2}\left(\mathbf{u}^{n+1} - \mathbf{u}^*\right) - \Delta t\left(\mathcal{D}\mathbf{u}^{n+1} - \mathcal{D}\mathbf{u}^*\right) = -\Delta t\nabla p^*. \tag{5.18}$$

  Neglecting the term $\mathcal{D}\mathbf{u}^{n+1} - \mathcal{D}\mathbf{u}^*$, taking the divergence of equation (5.18) and using the incompressibility constraint at $t = t^{n+1}$ then gives the following Poisson equation for the pressure correction

$$\nabla^2 p^* = \frac{3}{2}\frac{\nabla\cdot\mathbf{u}^*}{\Delta t}. \tag{5.19}$$

- Calculate the velocity at the time-level $n+1$ from equation (5.18) by

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{2}{3}\Delta t\nabla p^*. \tag{5.20}$$

- Calculate the pressure at the time-level $n+1$ by

$$p^{n+1} = p^n + p^*. \tag{5.21}$$

Some comments on boundary conditions are in order. A general choice is to adopt for the intermediate velocity $\mathbf{u}^*$ the original boundary conditions at the time-level $n+1$, that is to choose $\mathbf{u}^* \equiv \mathbf{u}^{n+1}$ on Γ. Due to the continuous formulation an 'artificial' boundary condition must be formulated for equation (5.19). The obvious way to formulate this boundary condition would be by taking either the normal or the tangential component of the momentum equation. However, in a pressure correction scheme a homogeneous Neumann boundary condition arises naturally for the Poisson equation for the pressure correction

$$\frac{\partial p^*}{\partial n} = 0 \qquad \text{on } \Gamma, \tag{5.22}$$

due to the assumption that $\mathbf{u}^* = \mathbf{u}^{n+1}$ on Γ. Moreover, this is the only condition that can guarantee the velocity to be divergence-free because of equations (5.19) and (5.20). For a fractional step method, in which a Poisson equation for the pressure arises, the need does exist to use the normal component of the momentum equation as a boundary condition for the pressure. As a consequence, in such an algorithm the velocity is not guaranteed to be divergence-free.

The above algorithm has two disadvantages. Firstly, it can easily be shown that its solution $(\mathbf{u}^{n+1}, p^{n+1})$ is not consistent with the solution of the system (5.16), due to the fact that the second-order term $\mathcal{D}\mathbf{u}^{n+1} - \mathcal{D}\mathbf{u}^*$ in equation (5.18) is neglected. Secondly, since $p^* = p^{n+1} - p^n$ and p^* satisfies equation (5.22), the pressure is not allowed to change in a boundary layer. To overcome these disadvantages, the following modified scheme is proposed:

- Calculate $\mathbf{u}^*$ from

  $$\frac{3}{2}\mathbf{u}^* - \Delta t \mathcal{D}\mathbf{u}^* = 2\mathbf{u}^n - \frac{1}{2}\mathbf{u}^{n-1} - \Delta t \nabla p^n + \Delta t \mathbf{f}^{n+1}, \tag{5.23}$$

  with $\mathbf{u}^* = \mathbf{u}^{n+1}$ on Γ.

- Calculate p^* from

  $$\nabla^2 p^* = \frac{3}{2}\frac{\nabla \cdot \mathbf{u}^*}{\Delta t}, \tag{5.24}$$

  with $\frac{\partial p^*}{\partial n} = 0$ on Γ.

- Calculate $\mathbf{u}^{n+1}$ from

  $$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{2}{3}\Delta t \nabla p^*. \tag{5.25}$$

- Calculate p^{n+1} from

  $$p^{n+1} = p^n + p^* - \nu \nabla \cdot \mathbf{u}^*. \tag{5.26}$$

Note that in the modified scheme both the velocity and the pressure are first predicted in the first step and then corrected in the remaining three steps. It is clear from equation (5.26) that the pressure can now also adapt in the boundary layer.

It can easily be shown that the solution $(\mathbf{u}^{n+1}, p^{n+1})$ of the modified scheme is consistent with that of the original system (5.16). Equation (5.25) also ensures that the normal component of the boundary conditions for $\mathbf{u}^{n+1}$ is satisfied on the boundary; the tangential component of the boundary condition can not be satisfied exactly. However, it can be shown that if the acceleration $\frac{\partial \mathbf{u}}{\partial t}$ and the source term are continuous in time (sudden starts and sudden sources are not allowed), the tangential boundary condition for the velocity is satisfied with accuracy $\mathcal{O}(\Delta t^2)$.

Applying the Laplace operator to equation (5.26), taking the divergence of equation (5.23) and using equation (5.24) yields that p^{n+1} satisfies the pressure Poisson equation (PPE)

$$\nabla^2 p^{n+1} = \nabla \cdot \mathbf{f}^{n+1} \qquad \text{in } \boldsymbol{\Omega}. \tag{5.27}$$

Taking the normal derivative of equation (5.26), substituting the pressure derivatives from equations (5.23) and (5.25) and using equation (5.24) gives

$$\frac{\partial p^{n+1}}{\partial n} = \mathbf{n} \cdot \left(\mathbf{f}^{n+1} - \Delta^2 \mathbf{u}^{n+1} + \nu \nabla^2 \mathbf{u}^{n+1}\right) \qquad \text{on } \boldsymbol{\Gamma}, \tag{5.28}$$

where $\Delta^2 \mathbf{u}^{n+1}$ denotes the second-order backward differences discretization of the time-derivative of the velocity. Equation (5.28) is in fact the Neumann boundary condition for the pressure which can be derived from the original system (5.16). Now using continuity in time for the acceleration and the source term it follows that p^{n+1} satisfies

$$\begin{cases} \nabla^2 p^{n+1} = \nabla \cdot \mathbf{f}^n + \mathcal{O}(\Delta t) & \text{in } \boldsymbol{\Omega}, \\ \dfrac{\partial p^{n+1}}{\partial n} = \mathbf{n} \cdot \left(\mathbf{f}^n - \Delta^2 \mathbf{u}^n + \nu \nabla^2 \mathbf{u}^n\right) + \mathcal{O}(\Delta t) & \text{on } \boldsymbol{\Gamma}, \end{cases} \tag{5.29}$$

leading to

$$p^{n+1} = p^n + \mathcal{O}(\Delta t). \tag{5.30}$$

On the other hand equation (5.23) implies that also $\mathbf{u}^* = \mathbf{u}^n + \mathcal{O}(\Delta t)$ and therefore applying the incompressibility constraint and using equation (5.26) yields

$$\nabla p^* = \mathcal{O}(\Delta t). \tag{5.31}$$

Finally, using equation (5.25) yields the important result

$$\mathbf{u}^{n+1} = \mathbf{u}^* + \mathcal{O}(\Delta t^2). \tag{5.32}$$

This means that the tangential component of $\mathbf{u}^{n+1}$ is satisfied with second-order accuracy in time, yielding a second-order consistent predictor-corrector scheme.

A way to impose the exact no-slip boundary condition for $\mathbf{u}^{n+1}$ could be to solve the momentum equation of system (5.16) (instead of using equation (5.25)) with p^{n+1} obtained from (5.26). Then however, the strict incompressibility imposed by equation (5.25) will be lost. In that case it can be shown, under certain conditions of smoothness, that $L^{n+1} \equiv \nabla \cdot \mathbf{u}^{n+1}$ satisfies the Helmholtz equation

$$\frac{3}{2}\frac{L^{n+1}}{\Delta t} = \nu \nabla^2 L^{n+1}, \tag{5.33}$$

which has solution $L^{n+1} \equiv 0$ only if $L^{n+1} = 0$ or $\frac{\partial L^{n+1}}{\partial n} = 0$ on Γ. Since this can not be ensured, $\nabla \cdot \mathbf{u}^{n+1}$ decays exponentially beyond a boundary layer of thickness $(\nu\Delta t)^{\frac{1}{2}}$, see also Orszag *et al.* (1986), which poses a severe restriction to the time-step. Numerical experiments have indeed shown that $\nabla \cdot \mathbf{u}^{n+1}$ is worse compared to the results obtained with the modified pressure correction scheme and that the maximum is achieved in the boundary layer. So with respect to divergence-freedom the present algorithm is better suited.

To conclude this section some remarks are made regarding natural boundary conditions. On that part of the boundary where the normal component of the stress is prescribed, that is on Γ_σ, the pressure can be described through equation (5.6). For instance, in the case of an outflow boundary it is customary to prescribe zero stress there. In formulations where a pressure Poisson equation has to be solved to obtain the pressure, the most frequently used outflow boundary condition for the pressure problem is $p = 0$ in combination with homogeneous Neumann conditions for the velocity components, since it results in no coupling between the pressure and the velocity calculations. The Neumann conditions for the velocity are implemented naturally in the variational formulation; also, Neumann conditions for the velocity on the outflow perform better than Dirichlet conditions because they are less restrictive.

Finally, with respect to initial conditions the initial velocity is given by equation (5.8); for the pressure correction scheme also an initial condition for the pressure is required. The most common way to obtain the initial pressure is to solve the PPE with the Neumann boundary condition obtained from the momentum equation at $t = 0$.

5.3.3 Spectral element discretization

Application of a Galerkin spectral element discretization to the semi-discrete pressure correction equations is performed in the standard way. As already stated in section 5.3.1, there is no need to satisfy any form of the discrete Brezzi–Babuška condition as the decoupling procedure is applied to the continuous equations, leading to uncoupled problems for both velocity and pressure. Therefore the degree of approximation for the pressure can be taken equal to that for the velocity, resulting in a numerical algorithm that is simple to implement. The fully discrete form of the pressure correction scheme thus becomes:

- Calculate $\mathbf{u}^*$ by solving

  $$\begin{aligned}\left(\frac{3}{2}\mathbf{M} + \Delta t\mathbf{D}\right)\mathbf{u}^* &= 2\mathbf{M}\mathbf{u}^n - \frac{1}{2}\mathbf{M}\mathbf{u}^{n-1} \\ &- \Delta t\mathbf{Q}\mathbf{p}^n + \Delta t\mathbf{M}\mathbf{f}^{n+1}, \qquad (5.34)\end{aligned}$$

  with **M** the (diagonal) mass matrix, **D** the diffusion or stiffness matrix and **Q** the gradient matrix. The column $\mathbf{p}^n$ contains the pressure components at $t = t^n$. The column **f** contains also the contribution of non-homogeneous boundary conditions.

- Calculate $\mathbf{p}^*$ by solving

  $$\mathbf{K}\mathbf{p}^* = -\frac{3}{2}\frac{\mathbf{L}\mathbf{u}^*}{\Delta t}, \qquad (5.35)$$

  with **K** the pressure stiffness matrix and **L** the divergence matrix.

- Calculate $\mathbf{u}^{n+1}$ via

  $$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{2}{3}\Delta t\mathbf{M}^{-1}\mathbf{Q}\mathbf{p}^*. \qquad (5.36)$$

- Calculate p^{n+1} via

  $$\mathbf{p}^{n+1} = \mathbf{p}^n + \mathbf{p}^* - \nu\mathbf{M}^{-1}\mathbf{L}\mathbf{u}^*, \qquad (5.37)$$

where ν is assumed to be constant. In appendix E these equations are derived in a more detailed way. Moreover, explicit expression for the system matrices are given there. From the above system it can be seen that it is essential that the mass matrix **M** is diagonal, since then the equations (5.36) and (5.37) do not involve the solution of a system, but only the calculation of matrix-vector products which can be performed on elemental level. Problems (5.34) and (5.35), finally, can be solved using an iterative finite element preconditioning technique for symmetric systems as described in chapter 3.

5.3.4 Application to an analytical test case

In order to test the performance of the pressure correction schemes presented in the previous sections, here a Stokes problem with analytical solution is analyzed. This analytical solution is also used by e.g. Kim and Moin (1985) and Rønquist (1988) for the Navier–Stokes problem. Later in this chapter this will be done also.

Consider the Stokes problem (5.13) with $\nu = 1$. The source term is chosen such that the exact velocity and pressure are given by

$$\begin{aligned} u_1(\mathbf{x},t) &= -\cos(x_1)\sin(x_2)e^{-2t}, \\ u_2(\mathbf{x},t) &= \sin(x_1)\cos(x_2)e^{-2t}, \\ p(\mathbf{x},t) &= -\frac{1}{4}\left(\cos(2x_1)+\cos(2x_2)\right)e^{-4t}. \end{aligned} \tag{5.38}$$

A plot of the exact velocity and pressure at $t = 1$ is shown in Figure 5.1.

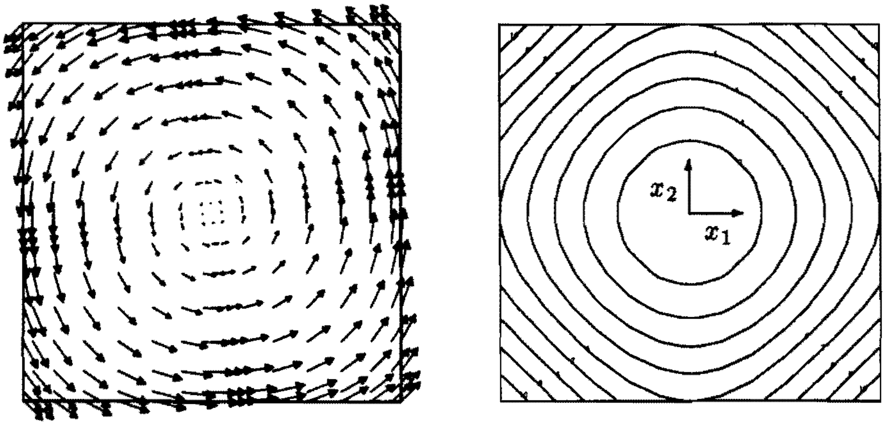


Figure 5.1: Stokes problem with analytical solution. Velocity field (left) and contourlines of the pressure (right) at $t = 1$.

The problem is solved on the domain $\Omega = (-1,1) \times (-1,1)$ with time-dependent boundary conditions for the velocity according to equation (5.38) and with initial velocity and pressure according to equation (5.38) at $t = 0$. The domain Ω is divided into $n_e = 4$ elements of degree n. As a first test, this problem is approximated until $t = 1$ using a degree of approximation n such that the spatial errors are negligible compared to the errors due to the time-integration.

Table 5.1: Stokes problem with analytical solution. Standard pressure-correction method. Discrete maximum error ε for several quantities, spatial discretization fixed, number of time-steps varying.

time-steps	discrete maximum error			
	u	**Lu**	**p**	**Qp**
8	$0.12 \cdot 10^{-2}$	$0.18 \cdot 10^{-4}$	$0.16 \cdot 10^{0}$	$0.42 \cdot 10^{-2}$
16	$0.16 \cdot 10^{-3}$	$0.90 \cdot 10^{-5}$	$0.74 \cdot 10^{-1}$	$0.22 \cdot 10^{-2}$
32	$0.50 \cdot 10^{-4}$	$0.12 \cdot 10^{-4}$	$0.38 \cdot 10^{-1}$	$0.94 \cdot 10^{-3}$
64	$0.12 \cdot 10^{-4}$	$0.29 \cdot 10^{-5}$	$0.26 \cdot 10^{-1}$	$0.18 \cdot 10^{-3}$

Table 5.2: Stokes problem with analytical solution. Modified pressure-correction method. Discrete maximum error ε for several quantities, spatial discretization fixed, number of time-steps varying.

time-steps	discrete maximum error			
	u	**Lu**	**p**	**Qp**
8	$0.47 \cdot 10^{-3}$	$0.82 \cdot 10^{-5}$	$0.27 \cdot 10^{0}$	$0.17 \cdot 10^{-2}$
16	$0.10 \cdot 10^{-3}$	$0.12 \cdot 10^{-5}$	$0.19 \cdot 10^{0}$	$0.27 \cdot 10^{-3}$
32	$0.24 \cdot 10^{-4}$	$0.19 \cdot 10^{-6}$	$0.18 \cdot 10^{0}$	$0.50 \cdot 10^{-4}$
64	$0.58 \cdot 10^{-5}$	$0.47 \cdot 10^{-7}$	$0.10 \cdot 10^{0}$	$0.13 \cdot 10^{-4}$

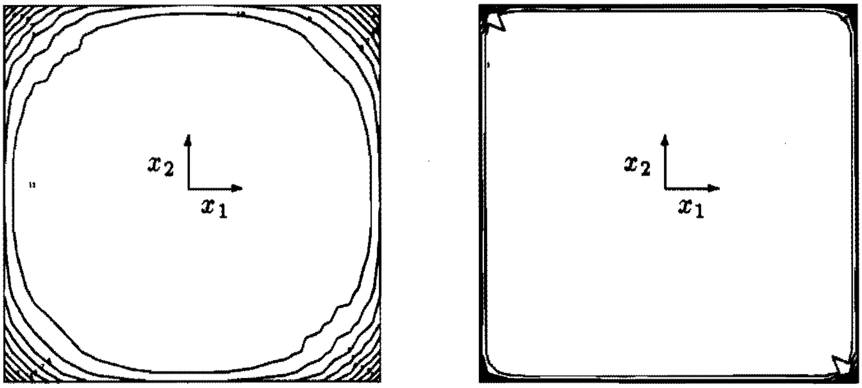


Figure 5.2: Contourplot of the pressure error for the Stokes problem with the standard pressure correction method, using 8 time-steps (left) and 32 time-steps (right).

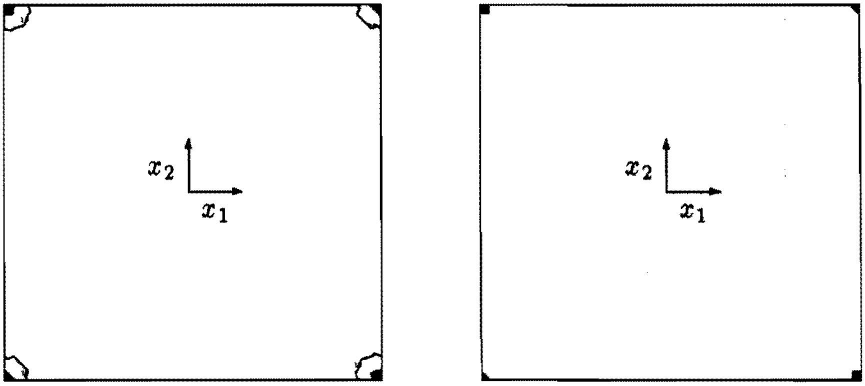


Figure 5.3: Contourplot of the pressure error for the Stokes problem with the modified pressure correction method, using 8 time-steps (left) and 32 time-steps (right).

In Table 5.1 results are presented for a varying number of time-steps for the velocity and its discrete divergence, and for the pressure and its discrete gradient, for the standard pressure correction approach; in Table 5.2 the same is done for the modified scheme presented in this thesis. Both schemes are second-order accurate in time for the velocity. For the modified scheme however, the results are more accurate, especially with respect to divergence-freedom. For the standard pressure correction scheme the results for the divergence do not improve much if the number of time-steps is increased. With respect to the gradient of the pressure the standard scheme does not obtain a clear second-order convergence in time, contrary to the modified scheme.

The results for the pressure itself are really bad for both schemes, realizing that the maximum value of the pressure at $t = 1$ is $\mathcal{O}(10^{-2})$. For the standard scheme, which clearly has disadvantages for the computing the pressure since it is not allowed to change in the boundary layer, this need not be a great shock. However, for the modified scheme, of which it is shown that the pressure satisfies the PPE with the correct Neumann condition, and is therefore consistent with the original Stokes equation, this may seem surprising. The picture becomes more clear if it is known where the error in the pressure achieves its maximum value. In Figure 5.2 a contour plot of the pressure error is given for several time-steps using the standard pressure correction approach. In Figure 5.3 the same is done for the modified pressure correction scheme. For the standard scheme the maximum error is achieved within a boundary layer, which can be explained by the fact that the pressure can not change there. For the modified scheme however, the maximum error is not obtained alongside the whole boundary, but only near the cornerpoints. The reason that the computed pressure is not correct is because for this test case the boundary is not smooth, due to the corners. The use of a Neumann condition overdetermines the problem in the cornerpoints.

It is proven by Gresho and Sani (1987) that for a sufficiently smooth boundary the pressure obtained from the PPE with the Neumann condition will also satisfy the Dirichlet boundary condition which follows from the tangential component of the momentum equation. If the Dirichlet boundary condition is used, the Poisson problem for the pressure is well-posed and should yield the correct pressure. To emphasize this point the same computations are made with the correct Dirichlet boundary conditions for the pressure correction Poisson equation. Table 5.3 shows the results for the standard pressure correction method and Table 5.4 gives the results obtained with the modificated scheme. For the standard method the pressure still is not correct. For the modified scheme however, the pressure now clearly is correct. Moreover, the pressure gradient has improved considerably. It can also be seen that since the zero Neumann boundary condition no longer holds now, the velocity and its divergence-freedom have deteriorated, compared to the results in Table 5.2.

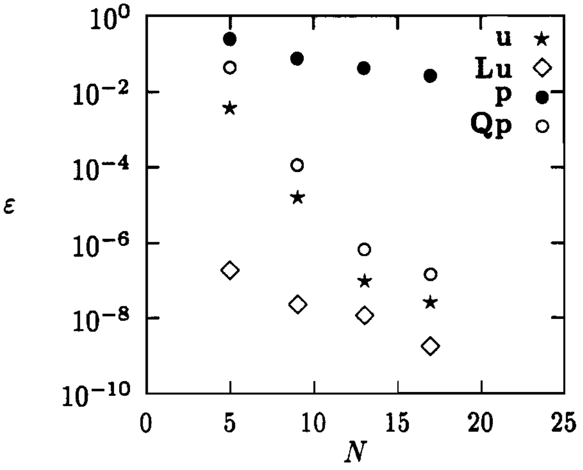
Table 5.3: Stokes problem with analytical solution. Standard pressure-correction method with Dirichlet boundary conditions for the pressure. Discrete maximum error ε for several quantities, spatial discretization fixed, number of time-steps varying.

time-steps	discrete maximum error			
	u	Lu	p	Qp
8	$0.74 \cdot 10^{-3}$	$0.53 \cdot 10^{-4}$	$0.49 \cdot 10^{-1}$	$0.50 \cdot 10^{-2}$
16	$0.20 \cdot 10^{-3}$	$0.17 \cdot 10^{-4}$	$0.22 \cdot 10^{-1}$	$0.26 \cdot 10^{-2}$
32	$0.67 \cdot 10^{-4}$	$0.13 \cdot 10^{-4}$	$0.11 \cdot 10^{-1}$	$0.83 \cdot 10^{-3}$
64	$0.21 \cdot 10^{-4}$	$0.32 \cdot 10^{-5}$	$0.68 \cdot 10^{-2}$	$0.21 \cdot 10^{-3}$

Table 5.4: Stokes problem with analytical solution. Modified pressure-correction method with Dirichlet boundary conditions for the pressure. Discrete maximum error ε for several quantities, spatial discretization fixed, number of time-steps varying.

time-steps	discrete maximum error			
	u	Lu	p	Qp
8	$0.14 \cdot 10^{-2}$	$0.23 \cdot 10^{-3}$	$0.34 \cdot 10^{-3}$	$0.40 \cdot 10^{-4}$
16	$0.64 \cdot 10^{-3}$	$0.77 \cdot 10^{-4}$	$0.13 \cdot 10^{-3}$	$0.56 \cdot 10^{-5}$
32	$0.99 \cdot 10^{-4}$	$0.11 \cdot 10^{-4}$	$0.70 \cdot 10^{-5}$	$0.47 \cdot 10^{-6}$
64	$0.14 \cdot 10^{-4}$	$0.13 \cdot 10^{-5}$	$0.33 \cdot 10^{-5}$	$0.14 \cdot 10^{-6}$

In the second test the problem is approximated until $t = 0.5$ using $n_e = 4$ elements with varying degree of freedom n, but now keeping the time-step small enough to ensure that temporal errors are negligible. The results for the modified scheme are presented in Figure 5.4 (top). Spectral convergence is obtained for the velocity and for the discrete gradient of the pressure; the pressure itself is not correct. Again performing the same computation, but now with the correct Dirichlet boundary conditions for the pressure also gives spectral convergence for the pressure, as is shown in Figure 5.4 (bottom).



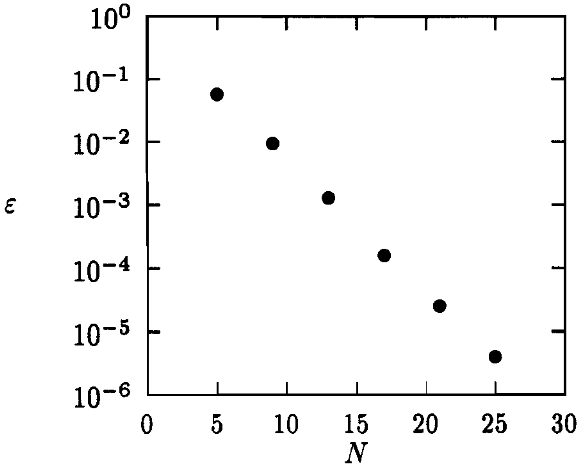


Figure 5.4: Stokes problem. Evolution of the discrete maximum error ε for several quantities (top) and for the pressure in the case of Dirichlet conditions (bottom) with $n_e = 4, n$ varying. N is the number of degrees of freedom in one direction.

5.4 Solution of the Navier–Stokes equations

5.4.1 The operator splitting approach

Consider again the Navier–Stokes equations plus boundary and initial conditions given by the system (5.12). The first step in the solution method is to apply a similar operator splitting technique as is described in chapter 4 for unsteady convection-diffusion problems, including the pressure term temporarily in the viscous part of the equation. Thereto the momentum equation is written in the following form

$$\frac{\partial \mathbf{u}}{\partial t} = \mathcal{D}\mathbf{u} + \mathcal{C}\mathbf{u} - \nabla p + \mathbf{f}, \tag{5.39}$$

with $\mathcal{D} = \nu\nabla^2$ the diffusion operator and $\mathcal{C} = -(\mathbf{u}\cdot\nabla)$ the non-linear convection operator. Equation (5.39) is written in terms of an integrating factor in $\mathcal{C}$

$$\frac{\partial}{\partial t}\left(\mathcal{Q}_{\mathcal{C}}^{(t^*,t)}\mathbf{u}\right) = \mathcal{Q}_{\mathcal{C}}^{(t^*,t)}(\mathcal{D}\mathbf{u} - \nabla p + \mathbf{f}). \tag{5.40}$$

For the definition of $\mathcal{Q}_{\mathcal{C}}^{(t^*,t)}$ see equation (4.6). The 'Stokes' equation (5.40) is integrated using an implicit backward differences scheme with time-step Δt. This yields the following semi-discrete system

$$\begin{aligned}\frac{\beta_0\mathbf{u}^{n+1} - \sum_{i=1}^{k}\beta_i\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n+1-i})}\mathbf{u}^{n+1-i}}{\Delta t} &= \mathcal{D}\mathbf{u}^{n+1}\\ &- \nabla p^{n+1} + \mathbf{f}^{n+1}.\end{aligned} \tag{5.41}$$

To evaluate the terms $\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n+1-i})}\mathbf{u}^{n+1-i}(i = 1,2,\ldots)$ the following associated initial value problem is solved

$$\begin{cases}\dfrac{\partial\tilde{\mathbf{u}}(s)}{\partial s} = \mathcal{C}\tilde{\mathbf{u}}(s), \quad 0 < s < i\Delta t,\\ \tilde{\mathbf{u}}(0) = \mathbf{u}^{n+1-i},\end{cases} \tag{5.42}$$

from which it follows that

$$\mathcal{Q}_{\mathcal{C}}^{(t^{n+1},t^{n+1-i})}\mathbf{u}^{n+1-i} = \tilde{\mathbf{u}}(i\Delta t). \tag{5.43}$$

Problem (5.43) according for the non-linear convection is solved using a three-step version of the explicit Taylor–Galerkin scheme described by equation (4.25). This scheme, also used by Jiang and Kawahara (1993), does not suffer from weak instability, contrary to the two-step scheme. The initial condition is $\tilde{\mathbf{u}}^0 = \mathbf{u}^{n+1-i}$; a time-step Δs such that $\Delta t = j\Delta s$ with j an integer is used. The semi-discrete convection step then becomes

$$\begin{aligned} \tilde{\mathbf{u}}^{m+\frac{1}{3}} &= \tilde{\mathbf{u}}^m - \frac{\Delta s}{3}(\tilde{\mathbf{u}}^m \cdot \nabla)\tilde{\mathbf{u}}^m, \\ \tilde{\mathbf{u}}^{m+\frac{1}{2}} &= \tilde{\mathbf{u}}^m - \frac{\Delta s}{2}(\tilde{\mathbf{u}}^{m+\frac{1}{3}} \cdot \nabla)\tilde{\mathbf{u}}^{m+\frac{1}{3}}, \\ \tilde{\mathbf{u}}^{m+1} &= \tilde{\mathbf{u}}^m - \Delta s(\tilde{\mathbf{u}}^{m+\frac{1}{2}} \cdot \nabla)\tilde{\mathbf{u}}^{m+\frac{1}{2}}. \end{aligned} \tag{5.44}$$

After introduction of the simpler notation $\tilde{\mathbf{u}}^{n+1-i} = \mathcal{Q}^{(t^{n+1-i},t^n)}\mathbf{u}^{n+1-i}$ (see also chapter 4), equation (5.43) leads to

$$\tilde{\mathbf{u}}^{n+1-i} = \tilde{\mathbf{u}}^{i(j+1)}. \tag{5.45}$$

The further deduction of the pressure correction scheme is completely analogous to the Stokes case; also the theory concerning boundary conditions and consistency of the computed solution holds now. The only difference consists of the equation for the intermediate velocity. For a second-order backward differences integration this part reads:

- Calculate $\mathbf{u}^*$ from

  $$\frac{3}{2}\mathbf{u}^* - \Delta t\mathcal{D}\mathbf{u}^* = 2\tilde{\mathbf{u}}^n - \frac{1}{2}\tilde{\mathbf{u}}^{n-1} - \Delta t\nabla p^n + \Delta t\mathbf{f}^{n+1}, \tag{5.46}$$

  with $\mathbf{u}^* = \mathbf{u}^{n+1}$ on Γ. The quantities $\tilde{\mathbf{u}}^n$ and $\tilde{\mathbf{u}}^{n-1}$ are calculated according to the associated convection problem (5.44), (5.45).

The spectral element discretization of equation (5.46) is given by:

- Calculate $\mathbf{u}^*$ by solving

  $$\begin{aligned} \left(\frac{3}{2}\mathbf{M} + \Delta t\mathbf{D}\right)\mathbf{u}^* &= 2\mathbf{M}\tilde{\mathbf{u}}^n - \frac{1}{2}\mathbf{M}\tilde{\mathbf{u}}^{n-1} \\ &- \Delta t\mathbf{Q}\mathbf{p}^n + \Delta t\mathbf{M}\mathbf{f}^{n+1}, \end{aligned} \tag{5.47}$$

  The columns $\tilde{\mathbf{u}}^n$ and $\tilde{\mathbf{u}}^{n-1}$ are calculated through the solution of

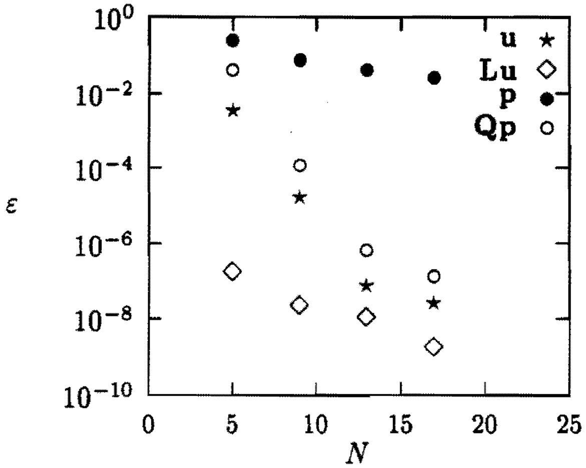
  $$\begin{aligned} \tilde{\mathbf{u}}^{m+\frac{1}{3}} &= \tilde{\mathbf{u}}^m - \frac{\Delta s}{3}\mathbf{M}^{-1}\mathbf{C}^m\tilde{\mathbf{u}}^m, \\ \tilde{\mathbf{u}}^{m+\frac{1}{2}} &= \tilde{\mathbf{u}}^m - \frac{\Delta s}{2}\mathbf{M}^{-1}\mathbf{C}^{m+\frac{1}{3}}\tilde{\mathbf{u}}^{m+\frac{1}{3}}, \\ \tilde{\mathbf{u}}^{m+1} &= \tilde{\mathbf{u}}^m - \Delta s\mathbf{M}^{-1}\mathbf{C}^{m+\frac{1}{2}}\tilde{\mathbf{u}}^{m+\frac{1}{2}}, \end{aligned} \tag{5.48}$$

  where $\mathbf{C}^{m+\frac{1}{3}}$ and $\mathbf{C}^{m+\frac{1}{2}}$ denote the convection matrix at time levels $m + \frac{1}{3}$ and $m + \frac{1}{2}$ respectively.

The derivation of the system (5.48) is analogous to that of the two-step scheme described in section 4.3.3. Again, the use of a diagonal mass matrix $\mathbf{M}$ ensures an efficient evaluation without the need to solve the system.

5.4.2 The analytical test case revisited

Consider again the analytical velocity and pressure given by equation (5.38). For $\nu = 1$ this solution satisfies the Navier–Stokes equations with zero source term. The numerical results for this test case are quite similar to those obtained for the Stokes problem, as can be expected. To emphasize that the modified pressure correction scheme performs well also in the presence of the non-linear convective term, the second test of section 5.3.4 is repeated. The problem is approximated until $t = 0.5$ using $n_e = 4$ elements of varying degree of approximation n, keeping the time-step small enough to ensure that temporal errors are negligible. Figure 5.5 (top) shows the results for the modified scheme using the obligated Neumann condition for the pressure correction Poisson equation. Spectral accuracy is obtained for the velocity and the discrete gradient of the pressure. For the pressure itself again spectral accuracy can be achieved using the correct Dirichlet boundary condition, as is shown in Figure 5.5 (bottom).



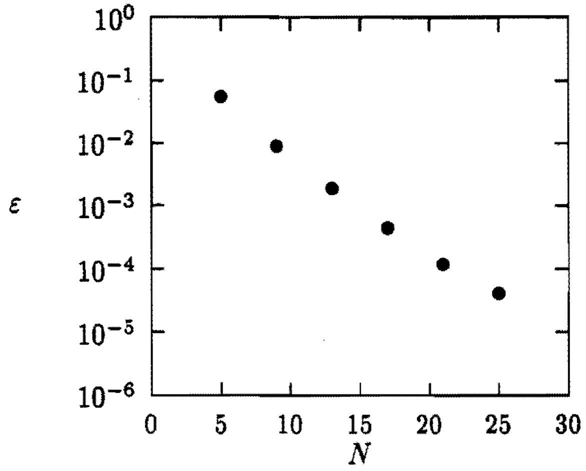


Figure 5.5: Navier–Stokes problem. Evolution of the discrete maximum error ε for several quantities (top) and for the pressure in the case of Dirichlet conditions (bottom) with $n_e = 4, n$ varying. N is the number of degrees of freedom in one direction.

5.5 Buoyancy-driven flow in an enclosed cavity

The simulation of cavity flow is an often used test case for the validation of a numerical technique for incompressible flow. In this section a buoyancy-driven free-convection flow in an enclosed cavity is analyzed, a problem well-documented in literature, see De Vahl Davis (1983). The driving force behind this flow is the gravity field acting on density differences caused by non-uniform temperature distributions. The numerical modelling of a free-convection flow is the more challenging, since it is is additionally complicated by the presence of the energy equation, a convection-diffusion equation for the temperature, which is coupled to the system of Navier–Stokes equations. Rønquist (1988) uses a similar test case for the algorithm presented in his thesis, however with a temperature that is assumed to be constant. In the context of the research presented in this thesis, the natural convection problem is very interesting since it couples the solution of the Navier–Stokes equations for incompressible flow to the solution of general convection-diffusion problems, described in chapter 4.

The equations for natural convection in a domain Ω presuming an incompressible flow are the momentum equation, the continuity equation and the energy equation. Compared to equation (5.12) the source term is replaced with the buoyancy term which, assuming constant fluid properties except for the density in the buoyancy term, is supposed to be linearly dependent on the temperature. In dimensionless form the full set of equations reads

$$\begin{cases} \dfrac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} - Pr\nabla^2\mathbf{u} + \nabla p = Ra Pr T\mathbf{g}, \\ \nabla\cdot\mathbf{u} = 0, \\ \dfrac{\partial T}{\partial t} + (\mathbf{u}\cdot\nabla)T = \nabla^2 T, \end{cases} \tag{5.49}$$

with $T(\mathbf{x})$ the dimensionless temperature and $\mathbf{g} = [0,1]^T$. The Rayleigh number Ra and the Prandtl number Pr are defined by

$$Ra = \frac{g\beta\Delta T l^3}{\delta\nu}, \qquad Pr = \frac{\nu}{\delta}, \tag{5.50}$$

with g the acceleration of gravity, β the thermal expansion coefficient, ΔT the characteristic temperature difference, l the characteristic length, δ the thermal diffusivity and ν the kinematic viscosity of the fluid.

The solution procedure for the system (5.49) is straightforward. Each time-step the energy equation is solved using an operator splitting scheme for convection-diffusion problems as described in chapter 4. For the diffusion part a second-order accurate backward differences scheme is used. The convection part is integrated using the three-step explicit scheme described in section 5.4.1. The velocity appearing in the convective term is extrapolated from the previous time-step using a second-order accurate in time extrapolation scheme. More details can be found in Minev *et al.* (1994). Next, the set of momentum equation and continuity equation is solved with the modified pressure correction scheme described in this chapter.

The computational domain with boundary conditions is shown in Figure 5.6. The cavity aspect ratio and the Prandtl number are chosen to be 1 and 0.71 respectively. Simulations are performed at 4 different values of Rayleigh numbers ($Ra = 10^3, 10^4, 10^5, 10^6$). Under the same conditions this problem has been investigated by Lankhorst (1991). For Rayleigh numbers larger than $Ra = 10^6$ no benchmark solution exists. Computations are performed until a steady state is reached using a time-step that varies from $\Delta t = 4\cdot 10^{-2}$ at $Ra = 10^3$ to $\Delta t = 10^{-4}$ at $Ra = 10^6$. The number of

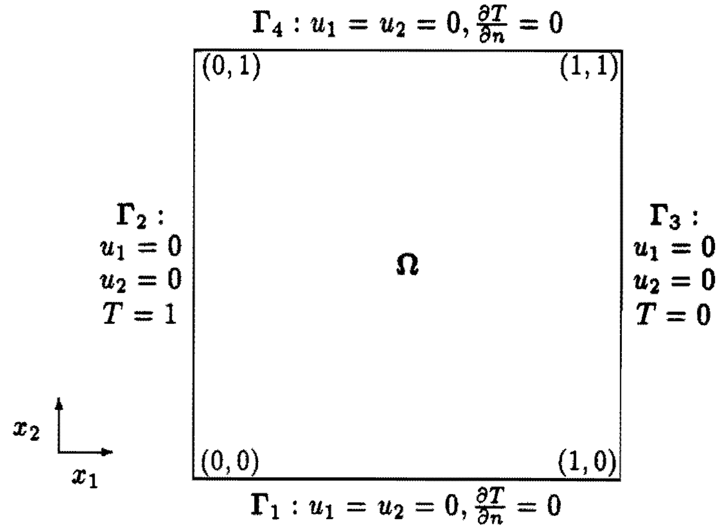

$\Gamma_4 : u_1 = u_2 = 0, \frac{\partial T}{\partial n} = 0$
(0,1)
(1,1)
Γ_2 :
$u_1 = 0$
$u_2 = 0$
$T = 1$
Ω
Γ_3 :
$u_1 = 0$
$u_2 = 0$
$T = 0$
x_2
x_1
(0,0)
(1,0)
$\Gamma_1 : u_1 = u_2 = 0, \frac{\partial T}{\partial n} = 0$

Figure 5.6: Buoyancy-driven flow in an enclosed cavity. Computational domain with boundary conditions.

elements in the spectral element discretization is $n_e = 16$; the degree of approximation in each element is $n = 8$.

The resulting streamfunction contours and isotherms are plotted in Figures 5.7 through 5.10 for increasing Rayleigh number. The qualitative agreement with the available information of Lankhorst (1991) is very good. The flow clearly follows the same development when the Rayleigh number is increased. At $Ra = 10^3$ and $Ra = 10^4$ a clockwise rotating primary flow with the rotation center, or stagnation point, at the cavity midpoint is observed. At $Ra = 10^5$ the center has been split up into two clockwise rotating secondary rolls. At $Ra = 10^6$ the flow is further split up into three, though very weak, rolls rotating in a clockwise direction. The thickness of the wall boundary layer also decreases with increasing Rayleigh number. This structure of the flow highly influences the resulting temperature field. At $Ra = 10^3$ the conduction is still dominant. For increasing Rayleigh number the problem for the temperature becomes highly convective; the temperature gradient normal to the isothermal walls at the left lower and right upper corners also increases due to the intensification of the convective heat transfer.

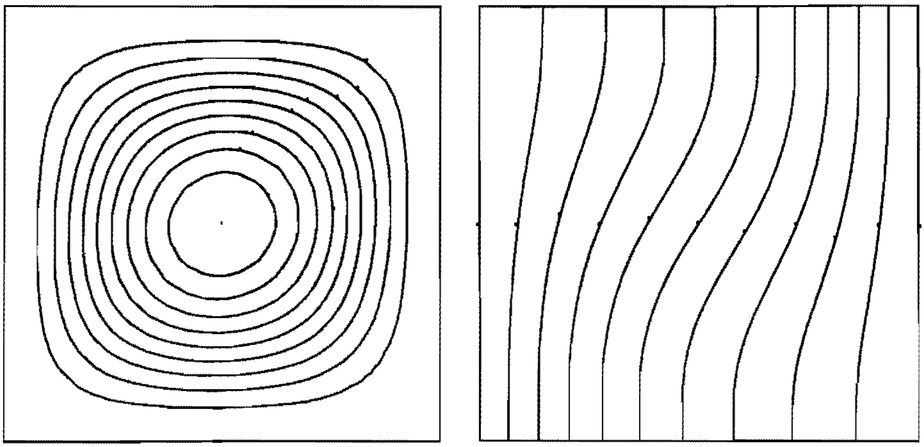


Figure 5.7: Streamlines (left) and isotherms (right) for the buoyancy-driven flow in an enclosed cavity at $Ra = 10^3$. Modified pressure correction/operator splitting scheme using $n_e = 16$ elements of degree $n = 8$.

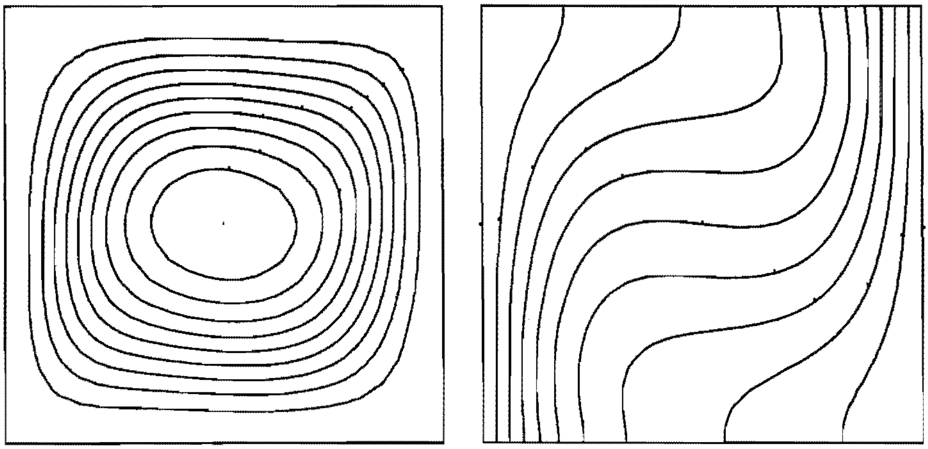


Figure 5.8: Streamlines (left) and isotherms (right) for the buoyancy-driven flow in an enclosed cavity at $Ra = 10^4$. Modified pressure correction/operator splitting scheme using $n_e = 16$ elements of degree $n = 8$.

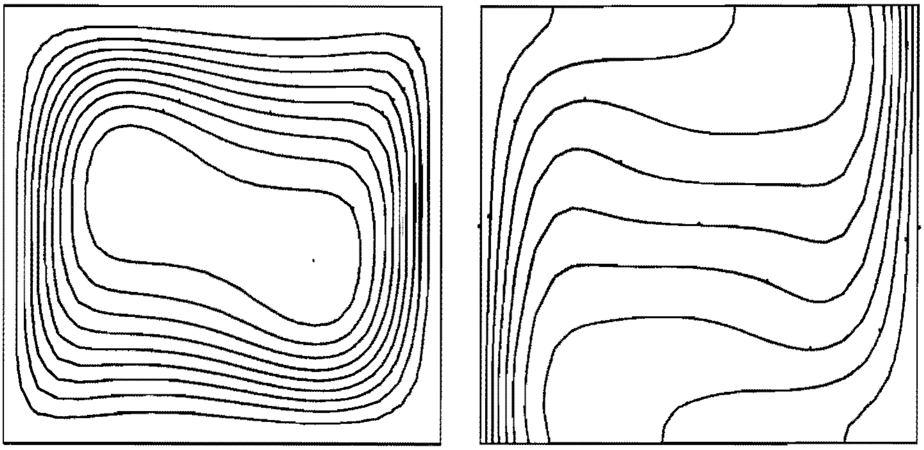


Figure 5.9: Streamlines (left) and isotherms (right) for the buoyancy-driven flow in an enclosed cavity at $Ra = 10^5$. Modified pressure correction/operator splitting scheme using $n_e = 16$ elements of degree $n = 8$.

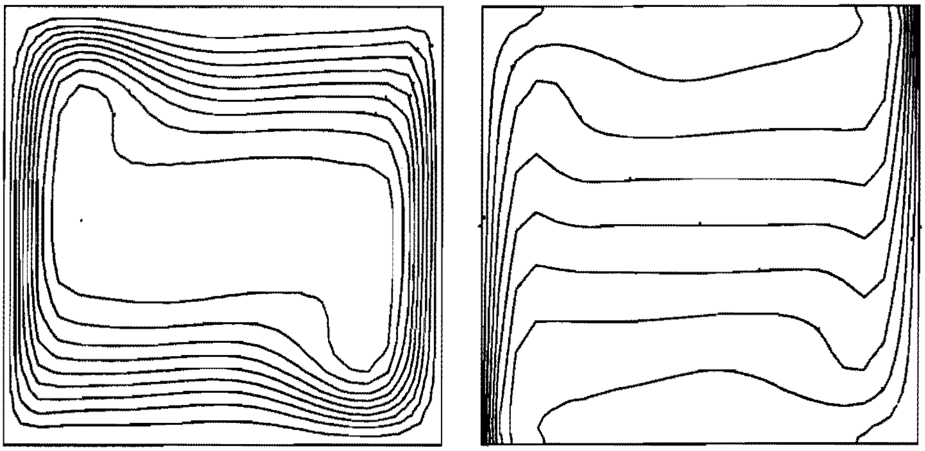


Figure 5.10: Streamlines (left) and isotherms (right) for the buoyancy-driven flow in an enclosed cavity at $Ra = 10^6$. Modified pressure correction/operator splitting scheme using $n_e = 16$ elements of degree $n = 8$.

Table 5.5: Buoyancy-driven flow in an enclosed cavity. Present results (P) compared with the benchmark numerical solution (B) and the deviation (D) for $Ra = 10^3$ through $Ra = 10^6$. Modified pressure correction/operator splitting scheme using $n_e = 16$ elements of degree $n = 8$.

variable	source	$Ra = 10^3$	$Ra = 10^4$	$Ra = 10^5$	$Ra = 10^6$
$u_{1,\max}$	B	3.649	16.178	34.73	64.63
	P	3.630	16.171	34.15	63.02
	D (%)	−0.5	0.0	−1.6	−2.3
$x_2(u_1)$	B	0.813	0.823	0.855	0.850
	P	0.830	0.830	0.875	0.830
$u_{2,\max}$	B	3.697	19.617	68.59	219.39
	P	3.693	19.604	66.85	219.69
	D (%)	−0.1	−0.1	−2.5	+0.1
$x_1(u_2)$	B	0.178	0.119	0.066	0.0379
	P	0.170	0.125	0.079	0.0404
$Nu_{\max}$	B	1.505	3.528	7.717	17.925
	P	1.507	3.531	7.717	17.350
	D (%)	+0.1	+0.1	0.0	−3.2
$x_2(Nu)$	B	0.092	0.143	0.081	0.0378
	P	0.080	0.125	0.080	0.0404
$Nu_{\min}$	B	0.692	0.586	0.729	0.989
	P	0.692	0.586	0.726	0.972
	D (%)	0.0	0.0	−0.3	−1.7
$x_2(Nu)$	B	1.0	1.0	1.0	1.0
	P	1.0	1.0	1.0	1.0

In Table 5.5 results are shown for Rayleigh numbers $Ra = 10^3$ through $Ra = 10^6$. Presented are the following characteristic values of the present method (P) compared with the benchmark values (B) of De Vahl Davis (1983) and the deviations (D) in terms of percentage from the benchmark solution:

- The maximum horizontal velocity on the vertical mid-line $u_{1,\max}$ and its x_2-location $x_2(u_1)$. This is illustrated in Figure 5.11 for the case $Ra = 10^5$.
- The maximum vertical velocity on the horizontal mid-line $u_{2,\max}$ and its x_1-location $x_1(u_2)$. This is illustrated in Figure 5.11 for the case $Ra = 10^5$.
- The maximum Nusselt number Nu at the 'hot' wall $x_1 = 0$ and its x_2-location $x_2(Nu)$.
- The minimum Nusselt number Nu at the 'hot' wall $x_1 = 0$ and its x_2-location $x_2(Nu)$.

The Nusselt number is defined by

$$Nu = \left.\frac{\partial T}{\partial x_1}\right|_{x_1=0} . \tag{5.51}$$

As can be seen the difference between the values of the present computation and the benchmark values are quite small. There is a tendency towards larger differences for increasing Rayleigh number. For instance, the maximum deviation is found at $Ra = 10^6$ for the maximum value of the Nusselt number. However, this may be expected because if the Rayleigh number increases, the sensitivity for the determination of the maximum (or minimum) of the Nusselt number to the distribution of nodal points in the vertical direction also increases. Unless accidentally one of the grid lines is located at exactly the right position, a too low value (or too high) value for the Nusselt number is found. This holds also for the determination of the maximum of the velocity components. Finally, it is remarked that the results also compare very well to the results obtained by Lankhorst (1991), who uses a considerably finer mesh, 45x45 nodes against 33x33 nodes for the present computation.

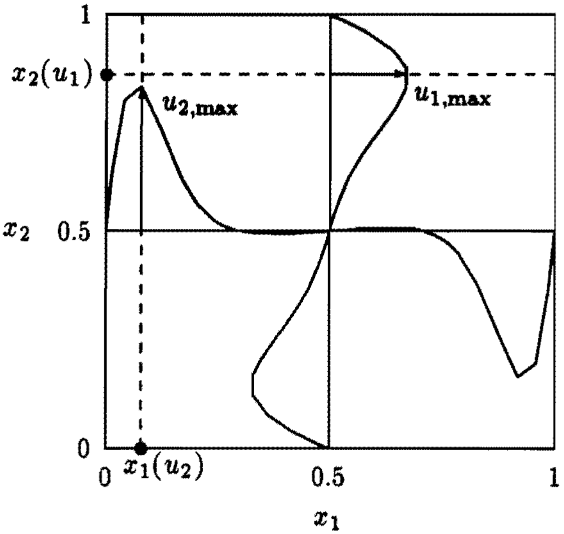


Figure 5.11: Buoyancy-driven cavity flow. Horizontal velocity profile on the vertical mid-line and vertical velocity profile on the horizontal mid-line for the case $Ra = 10^5$.

5.6 Conclusions

This chapter has dealt extensively with the solution of the Navier–Stokes equations for unsteady incompressible flow. The choice of solution method is largely determined by the need for an efficient numerical scheme. The solution method presented consists of a continuous pressure correction scheme with second-order accuracy in time. Pressure correction schemes are a special application of the class of projection methods. In a projection method the original set of Navier–Stokes equations is split into a set of problems that are simple to solve for both velocity and pressure. In the pressure correction approach the velocity and pressure are decoupled by taking the pressure at the previous time-level, resulting in a convection-diffusion problem for an intermediate velocity field that can be solved using the operator splitting technique presented in chapter 4. Enforcing the incompressibility constraint yields a Poisson equation for the pressure correction which can be solved using a finite element preconditioning technique of chapter 3.

In the standard pressure correction approach the computed solution is not consistent with the solution of the original Navier–Stokes system. Moreover, the pressure is not allowed to change in a boundary layer. In this chapter the standard pressure correction scheme has been modified. The modification consists of an addition to the equation for the computation of the new pressure in such a way that the disadvantages mentioned above no longer hold. This results results in a second-order consistent algorithm. Gresho and Sani (1987) emphasize the following weak points of a 'projection' approach of which only the first one applies to the present algorithm:

- *More-than-originally necessary smoothness for velocity and pressure are required.*
- *It is difficult to derive a boundary condition for the pressure, since it involves the calculation of second-order derivatives of the velocity.* In the present algorithm this problem does not exist. The boundary condition (a homogeneous Neumann condition) to be formulated is for the pressure correction and not for the pressure itself.
- *The velocity is not strictly guaranteed to be divergence-free, since the incompressibility constraint is applied to the intermediate velocity.* In the present algorithm the constraint is applied to the new velocity on each time-level thus guaranteeing always divergence-freedom.
- *The general solvability constraint for the Navier–Stokes equations*

  $$\int_{\Gamma} \mathbf{u}^{n+1}\, d\Gamma = 0, \tag{5.52}$$

  is often difficult to satisfy. In the present case it is automatically satisfied because of the choice of boundary conditions for the intermediate velocity.

The standard and modified algorithm have been thoroughly analyzed by means of a test case with analytical velocity and pressure. Especially with respect to divergence-freedom the modified scheme gives much better results. Moreover, the modified scheme is clearly second-order consistent in time. A 'practical' drawback for the computation of the pressure is a smoothness requirement for the boundary. For the test case the Poisson problem for the pressure correction is overdetermined in the corner points because of the use of the (obligated) Neumann condition. If the correct Dirichlet boundary condition is used the modified scheme yields the correct pressure, contrary to the standard scheme. Therefore, a way to compute the pressure in the case of a non-smooth boundary could be to retrieve it through post-processing of the discrete gradient on the boundary, yielding an accurate approximation of the Dirichlet boundary condition. The modified scheme has been further validated by means of its application to the problem of a buoyancy-driven cavity flow, which for increasing Rayleigh number results in a highly convective convection-diffusion problem for the temperature. Good results were obtained also for this problem.

With respect to the spectral element discretization the above algorithm holds several advantages. Firstly, the application of the decoupling algorithm to the continuous Navier–Stokes results in a set of equations that is simple to implement. The degree of approximation for velocity and pressure can be taken the same, as there is no need to satisfy any form of the Brezzi–Babuška condition. Secondly, the use of a diagonal mass matrix, which is a valid approach in a high-order method, is essential with respect to efficiency since in that case both the convective equations for the velocity and the correction equations for velocity and pressure do not involve the solution of a system, but only the calculation of matrix-vector products which can be performed on elemental level. Finally, for the analytical test case it was also shown that spectral accuracy for both velocity and pressure can be achieved if the temporal errors are negligible.

Chapter 6

Discussion

In the last chapter the accomplishments of the research presented in this thesis are discussed and some ideas for future research are given. The aim of this work is to solve convection-dominated flow problems, in particular governed by the Navier–Stokes equations, using a high-order spectral element method. The reason to apply spectral elements is the high accuracy that can be achieved when approximating 'sufficiently' smooth phenomena, such as occur in incompressible fluids, within a geometrically flexible framework. Therefore, the first part of this thesis (chapters 2 and 3) is mainly concerned with the spectral element method. In the second part (chapters 4 and 5) the application of the spectral element method to problems for unsteady incompressible flow is analyzed. Then not only the spatial discretization is of importance but also aspects such as the choice of time-integration and the choice of decoupling algorithm for the Navier–Stokes equations.

After describing the spectral element technique (chapter 2), using a thorough theoretical basis, in the first instance the property of exponential or spectral accuracy is checked by applying the method to several relatively simple test cases for elliptic equations. For such problems spectral elements prove to be much better suited than the conventional low-order methods as regards the accuracy of the approximate solution. For the approximation of somewhat less trivial solutions (but still smooth), such as highly convective phenomena in more dimensions the remark on accuracy still holds. However, a perhaps equally important criterion for a numerical method is the efficiency, both with respect to memory usage and computing time, with which the resulting system of discrete equations can be solved (chapter 3). For a high-order spectral-type method efficiency is achieved firstly by the use of iterative solution algorithms, and secondly by the use of a tensorial basis that is derived from the one-dimensional basis. This allows a method to significantly decrease the number of operations needed to compute the residuals in an iterative technique. The success of iterative solvers strongly depends on the quality of the preconditioner. A very powerful tool in the iteration is provided by the use of finite element preconditioning to improve the conditioning of the spectral element system matrix. Using this type of preconditioner only a few iterations are needed to obtain convergence.

At that point, having obtained not only a highly accurate numerical method but also an efficient iterative solver, the possibility exists to further deepen the analysis of the spectral element method by investigating such matters as the extension to complex three-dimensional geometries or the use of non-conforming elements. Since these properties have already been examined (with excellent results) in other researches (Bernardi *et al.*, 1990; Maday and Rønquist, 1990), here priority is given to the application of spectral elements to the specific class of incompressible flow problems, mainly because of the future field of application (flows in the transition from laminar to turbulent, visco-elastic flows). The above aspects will have to be addressed in future research if the application has need of them.

The solution of general unsteady convection-diffusion problems is considered a good starting point for validating a numerical method that is going to be applied to incompressible flow problems (chapter 4). Moreover, examples of the convection-diffusion equation can be found in a wide range of problems, such as the energy equation for temperature problems and the pure convection equation for particle tracking. For unsteady problems the choice of time-integration becomes also important. Since convection and diffusion are, also from a mathematical point of view, totally different phenomena, it is appropriate to treat them separately by using an operator splitting approach that decouples the problem into a pure convection problem followed by a pure diffusion problem. Again, this choice is influenced by the need for an efficient numerical scheme, since then the diffusion equation can be solved using an implicit time-integration with a large time-step, and the convection equation can be solved using an explicit time-integration with, if necessary, a smaller time-step. The use of a high-order spectral element method has a great advantage because in that case it is valid to apply a diagonal mass matrix, since it is not as restrictive with respect to accuracy as is the case for low-order methods. Consequently, the resulting discrete system does not involve the solution of a system, but only the calculation of matrix-vector products which can be performed on elemental level. The diffusion system can be solved using an iterative procedure described above.

The numerical results for the spectral element approximation of both linear and non-linear convection-diffusion equations are excellent, even if the convective part is dominant. For smooth phenomena clear spectral convergence (or p-convergence) is obtained if the time-step is small enough; again the method proves to be much more accurate than the conventional low-order finite element method. The opportunity has also been taken to approximate a non-smooth solution. In that case no spectral accuracy is obtained, as is to be expected. However, the use of high-order elements still has an advantage with respect to accuracy over low-order elements. As regards efficiency some balance will have to be found between p-convergence and h-convergence, both of which types can be provided for by the spectral element method. In the context of convection-diffusion problems the performance of the spectral element technique described in this thesis has also been compared in detail to other numerical methods, see Vreugdenhil and Koren (1993), Timmermans and Van de Vosse (1993^b). There the spectral element method gives excellent results in obtaining accuracy combined with an efficient algorithm, also for the approximation of a strongly non-linear and discontinuous propagation of a shock.

The solution of the Navier–Stokes equations for incompressible flow is the obvious next step (chapter 5). Again, the choice of solution method is largely determined by the need for an efficient method, which is found in a modified continuous pressure correction (or predictor-corrector) scheme. The set of equations is split into a set of problems for both velocity and pressure that are simple to solve. This results in a convection-diffusion problem for an intermediate velocity field that can be solved using the operator splitting technique mentioned above. Enforcing the incompressibility constraint yields a Poisson equation for the pressure correction which can be solved using a finite element preconditioning technique. Next, the velocity is 'corrected'. The main advantage of a pressure correction is that the final velocity obtained is guaranteed to be divergence-free. In this thesis a modification to the standard approach is proposed in order to derive a second-order consistent scheme. This also introduces a correction for the pressure which allows the pressure to change in the boundary layer. Using this modification there are no longer any problems relating 'artificial' boundary conditions for the pressure correction Poisson equation (Orszag *et al.*, 1986; Gresho and Sani, 1987; Karniadakis *et al.*, 1991).

The reason to choose a continuous decoupling algorithm is because in that case the degree of approximation for velocity and pressure can be taken the same, since there is no need to satisfy any form of the Brezzi–Babuška condition. The resulting set of discrete equations is then very easy to implement; moreover, the use of finite element preconditioning to solve the Poisson problem for the pressure is then straightforward. Again, the use of a diagonal spectral element mass matrix gives great advantages, also in the pressure correction scheme. Using the modified scheme numerical results for the Navier–Stokes equations are also very good showing a second-order consistency in time. Spectral convergence for both velocity, pressure and the gradient of the pressure can be obtained, if the time-step is chosen such that temporal errors are negligible. The only drawback is that the correct pressure must be retrieved through post-processing of the discrete pressure gradient (Gresho and Sani, 1987), if the boundary is not smooth. Concluding, at this moment there exists a highly accurate and highly efficient solution method for the approximation of incompressible flow problems.

As already indicated, with respect to the analysis of the spectral element method still work has to be done concerning the treatment of complex three-dimensional geometries. Another job for future research is to further speed up the iterative solver. At the moment, still the (sparse) LU-decomposition of the linear element matrix, that is used for preconditioning, is stored. The ultimate goal is to achieve a matrix-free solver by also solving the linear element system using an iterative procedure. A possible strategy could then be to make use of so-called element-by-element preconditioners, based on the product of the inverse of the element matrices. Since this may lead to a great loss of connectivity between the elements, an alternative is provided for by the use of clusters of elements (Tezduyar *et al.*, 1992). Such a procedure would fit naturally in the context of spectral elements, since every spectral element can be seen as a cluster of linear elements. As a consequence, the same part of the mesh is used for both computing the spectral residuals and inverting the finite element systems. This leads not only to a highly parallelizable iterative solver but also to a highly efficient data transport between the processors (Johan *et al.*, 1992). Also, the application of spectral elements enables a more flexible use of the available processors than low-order techniques; the fewer the processors, the higher the order of elements has to be. The above procedure seems to be more promising than the use of 'intra-element' multigrid (Rønquist and Patera, 1987; Maday and Munoz, 1988), since the communication between the subsequent levels of the multigrid cycle is complicated by the non-equidistant distribution of the grid points.

Future research must also focus on application of the techniques described in this thesis to flows in the transitional area between laminar and turbulent flow where often physically originated instabilities occur. These (smooth) flow phenomena are very difficult to simulate since (as already stated in chapter 1) they originate from relatively small scaled disturbances. Application of a highly-accurate spectral element method that introduces minimal numerical damping can prove to be very successful. Already, in this area work has been done (Kaiktsis *et al.*, 1991). Another very interesting application for spectral elements is the field of visco-elastic flow problems (Baaijens, 1992; Tamaddon-Jahromi *et al.*, 1992; Phillips and Roberts, 1993). Computations in this field often lead to large systems even for two-dimensional flows, since compared to the Navier–Stokes system also the extra stresses are unknowns. The use of spectral elements will not only enhance the computational accuracy, but as already stated they are more flexible than low-order methods with respect to parallelization. Moreover, as is also shown in this thesis spectral elements are well-suited to deal with highly convective problems, without the use of stabilizing upwind techniques, such as occur in visco-elastic problems when some splitting algorithm is applied to decouple the stresses and pressure from the velocity.

Appendix A

Orthogonal polynomials in (−1,1)

A.1 Sturm–Liouville problems

Consider the following eigenvalue problem on the domain $(-1,1)$

$$\begin{cases} -\dfrac{\partial}{\partial x}\left(a(x)\dfrac{\partial \varphi_i}{\partial x}\right) + b(x)\varphi_i = \lambda_i w(x)\varphi_i & \text{in } (-1,1), \\ \text{plus suitable boundary conditions for } \varphi_i. \end{cases} \tag{A.1}$$

Problem (A.1) is a Sturm–Liouville eigenvalue problem on $(-1,1)$. The solution $\varphi_i(x)$ is called the eigenfunction with eigenvalue λ_i. The coefficient $a(x)$ is continuously differentiable and strictly positive in $(-1,1)$, the coefficient $b(x)$ and the weight function $w(x)$ are non-negative in $(-1,1)$. The problem is called singular, or more specific, singular at the boundary points if

$$a(-1) = a(1) = 0. \tag{A.2}$$

As indicated in section 2.2.2 those eigenfunctions that form a complete orthogonal polynomial set in a Hilbert space are of special importance. First consider the property of completeness. Therefore define the weighted Lebesgue space

$$L^2_w(-1,1) = \left\{ v \;\middle|\; \int_{-1}^{1} v^2 \; w \; dx < \infty \right\}, \tag{A.3}$$

which is a Hilbert space. The natural inner product in this space is

$$(c, v)_w = \int_{-1}^{1} c(x)v(x)w(x)\,dx, \qquad \forall c, v \in L^2_w(-1,1), \tag{A.4}$$

with induced norm $\|v\|^2_w = (v, v)_w$. It can be shown, see e.g. Courant and Hilbert (1953), that the eigenfunctions φ_i form a complete set on the Lebesgue space $L^2_w(-1,1)$ if and only if

$$\lambda_i \to \infty \qquad (i \to \infty). \tag{A.5}$$

Consider now the expansion of an analytical function $c(x)$ in terms of eigenfunctions φ_i with eigenvalues λ_i that satisfy equation (A.5). In Canuto *et al.* (1988) it is shown that this expansion has the property of spectral accuracy, i.e. the coefficients $\tilde{c}_i$ decay faster than any inverse power of the eigenvalues λ_i

$$\forall m > 0\ \exists i_0\ \forall i > i_0 :\ \tilde{c}_i < \lambda_i^{-m}. \tag{A.6}$$

It should be noted here that spectral accuracy is generally not obtained for eigenfunction expansions of a non-singular or of a half-singular Sturm–Liouville problem. In these cases the coefficients decay algebraically unless the function $c(x)$ satisfies an infinite number of special boundary conditions. A proof of these properties is given in Gottlieb and Orszag (1977).

Of special interest are those orthogonal eigenfunctions that are polynomials, because of the efficiency with which they can be evaluated and differentiated numerically. In section 2.2.2 it is seen that the polynomial eigenfunctions are the Jacobi polynomials, see e.g. Abramowitz and Stegun (1972). In practice only the Chebyshev and Legendre polynomials are used in spectral methods. A survey of Legendre polynomials is given in appendix B.

A.2 Orthogonal systems of polynomials

In section 2.2.2 the spectral approximation of an analytical function $c(x)$ in terms of orthogonal eigenfunctions of Sturm–Liouville problems is discussed. In this section the expansion of $c(x)$ in terms of a system of orthogonal polynomials shall be discussed from a general point of view.

First define $P_n(-1,1)$ to be the space of polynomials of degree $\leq n$ on $(-1,1)$. Let p_i $(i = 0, 1 \ldots)$ be a system of algebraic polynomials that are mutually orthogonal in $(-1,1)$ with respect to a weight function $w(x)$

$$\int_{-1}^{1} p_i(x)p_j(x)w(x)\,dx = 0, \qquad i \neq j. \tag{A.7}$$

The classical Weierstrass theorem implies that the system p_i is complete in the Lebesgue space $L^2_w(-1,1)$ defined by equation (A.3). A function $c(x)$ can be expanded in terms of the polynomials p_i. This formal series is denoted by

$$c(x) = \sum_{i=0}^{\infty} \tilde{c}_i p_i(x). \tag{A.8}$$

The coefficients $\tilde{c}_i$ satisfy

$$\tilde{c}_i = \frac{1}{\|p_i\|^2_w} \int_{-1}^{1} c(x)p_i(x)w(x)\,dx. \tag{A.9}$$

Consider now the truncated series expansion defined for any integer $n > 0$ by

$$\mathcal{S}_n c(x) = \sum_{i=0}^{n} \tilde{c}_i p_i(x). \tag{A.10}$$

Due to the orthogonality relation (A.7), $\mathcal{S}_n c$ is the orthogonal projection of $c(x)$ upon $P_n(-1,1)$ with respect to the inner product (A.4). The completeness of the system p_i is equivalent to the property

$$\|c - \mathcal{S}_n c\|_w \to 0, \qquad \forall c \in L^2_w(-1,1) \quad (n \to \infty). \tag{A.11}$$

Equation (A.11) is a well-known result of classical functional analysis. It states that the function $c(x)$ is equivalent to its series expansion (A.8).

A.3 Discrete polynomial expansions

The coefficients $\tilde{c}_i$ depend on all the values of $c(x)$ in physical space and are therefore rarely computed exactly. A finite number of discrete coefficients can easily be computed using the values of $c(x)$ at Gauss-type quadrature points. The finite series defined by the discrete coefficients is actually the interpolating polynomial of $c(x)$ at these points.

Let x_k and w_k be the Gauss-type quadrature points and weights. The interpolating polynomial can be written as

$$\mathcal{I}_n c(x) = \sum_{i=0}^{n} \hat{c}_i p_i(x). \tag{A.12}$$

It is uniquely defined by

$$\mathcal{I}_n c(x_k) = c(x_k), \qquad k = 0, \ldots, n. \tag{A.13}$$

The $\hat{c}_i$ are the discrete coefficients. They satisfy

$$\hat{c}_i = \frac{1}{\|p_i\|_n^2} \sum_{j=0}^{n} c(x_j) p_i(x_j) w_j \; dx, \tag{A.14}$$

with $\|\cdot\|_n$ the norm induced by the discrete inner product

$$(c, v)_n = \sum_{k=0}^{n} c(x_k) v(x_k) w_k, \qquad \forall c, v \in C^0(-1, 1). \tag{A.15}$$

For more information on orthogonal systems of polynomials see Canuto *et al.* (1988).

Appendix B

The Legendre polynomials

The Legendre polynomials are eigenfunctions of a singular Sturm–Liouville problem on $(-1,1)$ which reads

$$-\frac{\partial}{\partial x}\left(\left(1-x^2\right)\frac{\partial L_n(x)}{\partial x}\right)=n(n+1)L_n(x). \tag{B.1}$$

In terms of equation (A.1) this implies that $\alpha(x)=(1-x^2)$, $\beta(x)=0$ and $w(x)=1$ and that the eigenvalues satisfy $\lambda_n=n(n+1)$. The Legendre polynomials satisfy the recurrence relation

$$L_{n+1}(x)=\frac{2n+1}{n+1}xL_n(x)-\frac{n}{n+1}L_{n-1}(x), \tag{B.2}$$

with $L_0(x)=1$ and $L_1(x)=x$. The following properties are of importance

$$L_n(\pm1)=(\pm1)^n, \tag{B.3}$$

$$\frac{\partial L_n(\pm1)}{\partial x}=(\pm1)^{n+1}\frac{1}{2}n(n+1), \tag{B.4}$$

$$\frac{\partial^2 L_n(\pm1)}{\partial x^2}=(\pm1)^n\frac{(n+2)(n+1)n(n-1)}{8}, \tag{B.5}$$

$$(2n+1)L_n(x)=\frac{\partial L_{n+1}}{\partial x}-\frac{\partial L_{n-1}}{\partial x}, \tag{B.6}$$

$$(1-x^2)\frac{\partial L_n}{\partial x}=nL_{n-1}(x)-nxL_n(x). \tag{B.7}$$

Let $x_0 = -1$, $x_n = 1$, and let x_k $(k = 1, \ldots, n-1)$ be the Legendre Gauss–Lobatto points defined as the roots of $\frac{\partial L_n}{\partial x}$. The interpolation polynomial for a given function $c(x)$ on $(-1,1)$ can be constructed as

$$\mathcal{I}_n c(x) = \sum_{i=0}^{n} c(x_i)\phi_i(x), \tag{B.8}$$

where the $\phi_i(x)$ denote the Lagrange interpolation polynomials which are 1 at x_i and 0 at all other Gauss–Lobatto points. The functions $\phi_i(x)$ are given by

$$\phi_i(x) = \frac{-1}{n(n+1)L_n(x_i)} \frac{1-x^2}{x-x_i} \frac{\partial L_n}{\partial x}. \tag{B.9}$$

In Figures B.1 through B.4 the Lagrange polynomials $\phi_i(x)$ are shown for $n = 2, 4, 8, 16$. Also shown are the corresponding Legendre Gauss–Lobatto points.

The derivatives of $\mathcal{I}_n c(x)$ in the interpolation points are given by

$$\left.\frac{\partial^{(m)} \mathcal{I}_n c(x)}{\partial x^{(m)}}\right|_{x=x_k} = \sum_{i=0}^{n} c(x_i) P_{ki}^{(m)}, \tag{B.10}$$

with

$$P_{ki}^{(m)} = \left.\frac{\partial^{(m)} \phi_i(x)}{\partial x^{(m)}}\right|_{x=x_k}. \tag{B.11}$$

Using equations (B.6) and (B.7) the Lagrange polynomials can also be written as

$$\phi_i(x) = \frac{-1}{(2n+1)L_n(x_i)} \frac{L_{n-1}(x) - L_{n+1}(x)}{x-x_i}. \tag{B.12}$$

This form easily leads to

$$\frac{\partial \phi_i}{\partial x} = \frac{1}{x-x_i}\left(\frac{L_n(x)}{L_n(x_i)} - \phi_i(x)\right). \tag{B.13}$$

Using $\phi_i(x_k) = \delta_{ik}$ one can derive for the case that $k \neq i$

$$P_{ki}^{(1)} = \frac{\partial \phi_i(x_k)}{\partial x} = \frac{L_n(x_k)}{L_n(x_i)} \frac{1}{(x_k - x_i)}, \qquad k \neq i. \tag{B.14}$$

Substitution of Taylor expansions around x_i in equation (B.13) yields

$$P_{ii}^{(1)} = \frac{1}{2L_n(x_i)}\frac{\partial L_n(x_i)}{\partial x}. \tag{B.15}$$

Using the properties (B.4) and (B.5) and recalling that $\frac{\partial L_n}{\partial x}|_{x=x_i} = 0$ for $i = 1, \ldots, n-1$ finally yields

$$\begin{cases} P_{ki}^{(1)} = \dfrac{L_n(x_k)}{L_n(x_i)}\dfrac{1}{(x_k - x_i)}, & k, i = 1, \ldots, n-1,\ k \neq i, \\ P_{ii}^{(1)} = 0, & i = 1, \ldots, n-1, \\ P_{00}^{(1)} = -\dfrac{1}{4}n(n+1), & \\ P_{nn}^{(1)} = \dfrac{1}{4}n(n+1). & \end{cases} \tag{B.16}$$

The second derivatives can be derived from equation (B.13)

$$\begin{aligned} \frac{\partial^2 \phi_i}{\partial x^2} &= \frac{1}{(x - x_i)}\left(\frac{1}{L_n(x_i)}\frac{\partial L_n}{\partial x} - \frac{\partial \phi_i}{\partial x}\right) \\ &- \frac{1}{(x - x_i)^2}\left(\frac{L_n(x)}{L_n(x_i)} - \phi_i(x)\right). \end{aligned} \tag{B.17}$$

Using the same methods as for the first derivative one can obtain

$$P_{ii}^{(2)} = \frac{1}{3}\frac{1}{L_n(x_i)}\frac{\partial^2 L_n(x_i)}{\partial x^2}. \tag{B.18}$$

And thus

$$\begin{cases} P_{ki}^{(2)} = -2\dfrac{L_n(x_k)}{L_n(x_i)}\dfrac{1}{(x_k - x_i)^2}, \quad k \neq i, \\ \qquad\qquad k = 1, \ldots, n-1,\ i = 0, \ldots, n, \\ P_{ki}^{(2)} = \dfrac{(\pm 1)^{n+1} n(n+1)}{2L_n(x_i)(\pm 1 - x_i)} - 2\dfrac{(\pm 1)^n}{L_n(x_i)(\pm 1 - x_i)^2}, \\ \qquad\qquad k = 0, n,\ i = 0, \ldots, n,\ i \neq k, \\ P_{ii}^{(2)} = -\dfrac{1}{3}\dfrac{n(n+1)}{1 - x_i^2}, \quad i = 1, \ldots, n-1, \\ P_{00}^{(2)} = P_{nn}^{(2)} = \dfrac{1}{24}(n+2)(n+1)n(n-1). \end{cases} \tag{B.19}$$

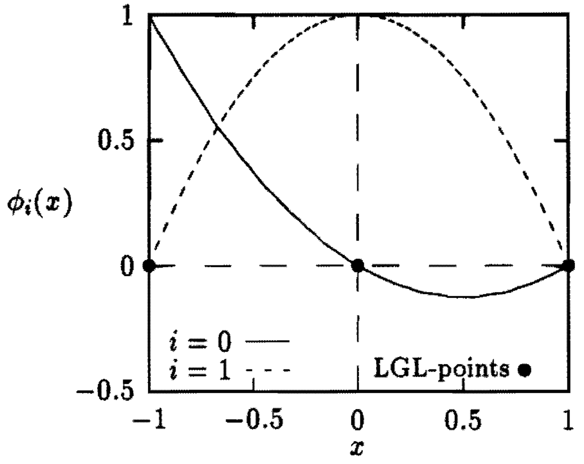


Figure B.1: Lagrange interpolants $\phi_i(x)$ ($i = 0, i = 1$) through the Legendre Gauss–Lobatto points (•) for $n = 2$.

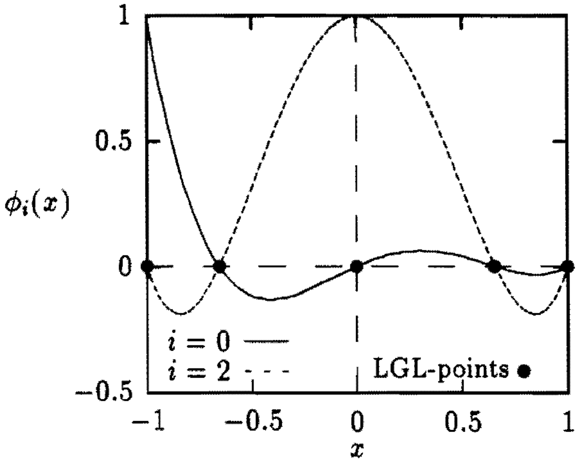


Figure B.2: Lagrange interpolants $\phi_i(x)$ ($i = 0, i = 2$) through the Legendre Gauss–Lobatto points (•) for $n = 4$.

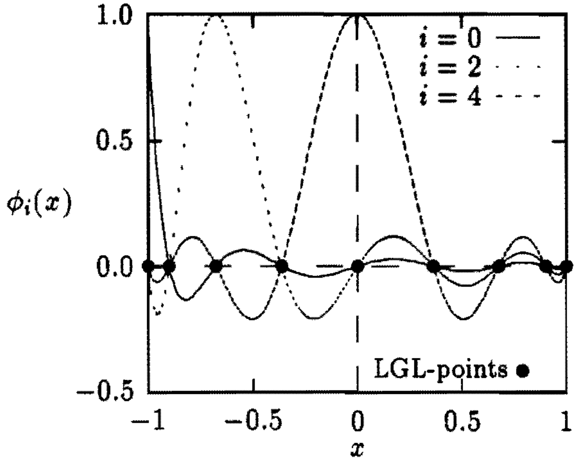


Figure B.3: Lagrange interpolants $\phi_i(x)$ $(i = 0, i = 2, i = 4)$ through the Legendre Gauss–Lobatto points (•) for $n = 8$.

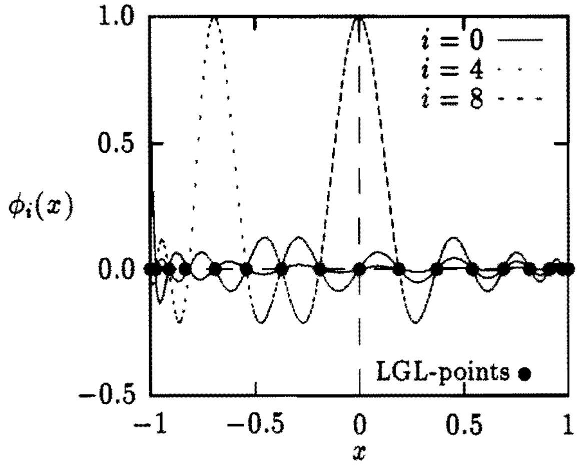


Figure B.4: Lagrange interpolants $\phi_i(x)$ $(i = 0, i = 4, i = 8)$ through the Legendre Gauss–Lobatto points (•) for $n = 16$.

Appendix C

Finite element preconditioned algorithms

In this appendix the iterative algorithms with the idea of finite element preconditioning incorporated, that are analyzed in chapter 3, are presented. The spectral element system is denoted by

$$\mathbf{Sc} = \mathbf{Mf}, \tag{C.1}$$

with **S** the system matrix and **M** the diagonal mass matrix. The linear finite element preconditioner, based on the same Legendre Gauss–Lobatto interpolation points is denoted by **F**.

Minimum Residual

```
solve $\mathbf{Fc}^0 = \mathbf{Mf}$
$\mathbf{r}^0 = \mathbf{Mf} - \mathbf{Sc}^0$
solve $\mathbf{Fb}^0 = \mathbf{r}^0$
do until $\|\mathbf{r}^{k+1}\|$ small enough
    $\alpha^k = \dfrac{(\mathbf{r}^k, \mathbf{Sb}^k)}{(\mathbf{Sb}^k, \mathbf{Sb}^k)}$                (C.2)
    $\mathbf{c}^{k+1} = \mathbf{c}^k + \alpha^k \mathbf{b}^k$
    $\mathbf{r}^{k+1} = \mathbf{r}^k - \alpha^k \mathbf{Sb}^k$
    solve $\mathbf{Fb}^{k+1} = \mathbf{r}^{k+1}$
enddo
```

Conjugate Gradient

```
solve $\mathbf{F}\mathbf{c}^0 = \mathbf{M}\mathbf{f}$
$\mathbf{r}^0 = \mathbf{M}\mathbf{f} - \mathbf{S}\mathbf{c}^0$
solve $\mathbf{F}\mathbf{z}^0 = \mathbf{r}^0$
$\mathbf{b}^0 = \mathbf{z}^0$
do until $\|\mathbf{r}^{k+1}\|$ small enough
    $\alpha^k = \frac{(\mathbf{r}^k, \mathbf{z}^k)}{(\mathbf{b}^k, \mathbf{S}\mathbf{b}^k)}$
    $\mathbf{c}^{k+1} = \mathbf{c}^k + \alpha^k \mathbf{b}^k$                    (C.3)
    $\mathbf{r}^{k+1} = \mathbf{r}^k - \alpha^k \mathbf{S}\mathbf{b}^k$
    solve $\mathbf{F}\mathbf{z}^{k+1} = \mathbf{r}^{k+1}$
    $\beta^k = \frac{(\mathbf{r}^{k+1}, \mathbf{z}^{k+1})}{(\mathbf{r}^k, \mathbf{z}^k)}$
    $\mathbf{b}^{k+1} = \mathbf{z}^{k+1} + \beta^k \mathbf{b}^k$
enddo
```

Bi-CGSTAB

```
solve $\mathbf{F}\mathbf{c}^0 = \mathbf{M}\mathbf{f}$
$\mathbf{r}^0 = \mathbf{M}\mathbf{f} - \mathbf{S}\mathbf{c}^0$
$\rho^0 = \alpha = \omega^0 = 1$
$\mathbf{v}^0 = \mathbf{p}^0 = \mathbf{0}$
do until $\|\mathbf{r}^{k+1}\|$ small enough
    $\rho^{k+1} = (\mathbf{r}^0, \mathbf{r}^k)$
    $\beta = \frac{\rho k + 1}{\rho^k} \frac{\alpha}{\omega^k}$
    $\mathbf{p}^{k+1} = \mathbf{r}^k + \beta \left(\mathbf{p}^k - \omega^k \mathbf{v}^k\right)$
    solve $\mathbf{F}\mathbf{y} = \mathbf{p}^k$
    $\mathbf{v}^{k+1} = \mathbf{S}\mathbf{y}$                    (C.4)
    $\alpha = \frac{\rho k + 1}{(\mathbf{r}^0, \mathbf{v}^{k+1})}$
    $\mathbf{s} = \mathbf{r}^k - \alpha \mathbf{v}^{k+1}$
    solve $\mathbf{F}\mathbf{z} = \mathbf{s}$
    $\mathbf{t} = \mathbf{S}\mathbf{z}$
    $\omega^{k+1} = \frac{(\mathbf{t}, \mathbf{s})}{(\mathbf{t}, \mathbf{t})}$
    $\mathbf{c}^{k+1} = \mathbf{c}^k + \alpha \mathbf{y}$
    $\mathbf{r}^{k+1} = \mathbf{s} - \omega^{k+1} \mathbf{t}$
enddo
```

Appendix D

Backward differences formulae

In chapter 4 an operator splitting method is discussed that decouples a convection-diffusion problem into a pure convection and a pure diffusion problem. For spectral-type methods the eigenvalues of the diffusion or stiff system are real and negative and grow like $\mathcal{O}(N^4)$, if $N \to \infty$. Here N is the maximum of the number of degrees of freedom in each spatial direction. As a consequence, the diffusion part requires a time-integration with a stability area that includes the negative real axis.

A desirable property for a multistep time-integration for the diffusion part is so-called A-stability (Hairer and Wanner, 1991). A method is called A-stable if the region of absolute stability includes the negative complex half plane. It is proven by Dahlquist (1963) that a multistep method that is A-stable can not have order greater than 2, and that the method of order 2 with the smallest error constant is the trapezoidal method. Since for the diffusion part only the negative real axis has to be included in the stability region, the requirement of A-stability can be relaxed. For this purpose the property of $A(\alpha)$-stability is introduced. A method is $A(\alpha)$-stable if the region of absolute stability includes the wedge in Figure D.1, thus always including the negative real axis.

A useful class of $A(\alpha)$-stable time-integration methods is given by the so-called 'backward differences formulae' (Hairer *et al.*, 1987). In a backward differences scheme the time-derivative is approximated via backward differentiation, leading to

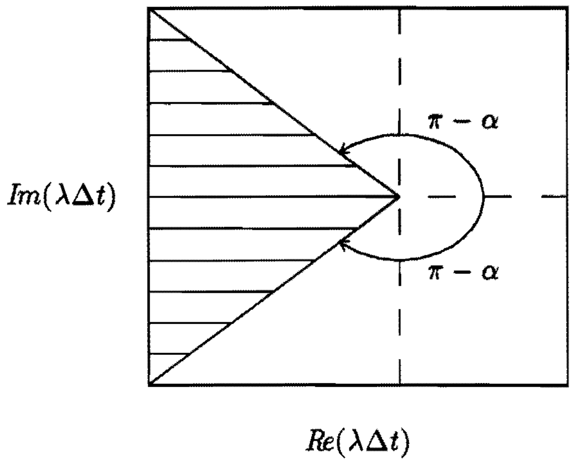


Figure D.1: $A(\alpha)$-stability region.

$$\frac{\partial c}{\partial t} = \frac{\beta_0 c^{n+1} - \sum_{i=1}^{k} \beta_i c^{n+1-i}}{\Delta t}, \tag{D.1}$$

where e.g. the superscript $n+1$ denotes the approximation at the time $t^{n+1} = (n+1)\Delta t$ with Δt the time-step. For consistency it is required that

$$\beta_0 = \sum_{i=1}^{k} \beta_i. \tag{D.2}$$

The coefficients of the schemes of order $k = 1$ through $k = 4$ are given in Table D.1.

With respect to stability, backward differences schemes are accurate for all components around the origin in the stability diagram and absolutely stable away from the origin in the left imaginary plane. Backward differences schemes are implicit and available up to eleventh order; the schemes of order $k = 1$ through $k = 6$ are $A(\alpha)$-stable. In Figure D.2 the regions of absolute stability for the backward differences schemes of order $k = 1$ through $k = 4$ are shown. The methods are stable outside of the closed contours.

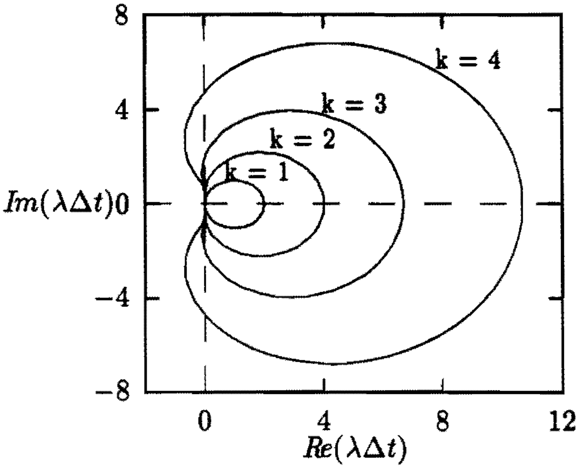


Figure D.2: Regions of absolute stability of the backward differences schemes of order $k = 1$ through $k = 4$. The methods are stable outside of the closed contours.

Table D.1: Coefficients of the backward differences schemes of order $k = 1$ through $k = 4$.

coefficient	$k = 1$	$k = 2$	$k = 3$	$k = 4$
β_0	1	$\frac{3}{2}$	$\frac{11}{6}$	$\frac{25}{12}$
β_1	1	2	3	4
β_2	–	$-\frac{1}{2}$	$-\frac{3}{2}$	-3
β_3	–	–	$\frac{1}{3}$	$\frac{4}{3}$
β_4	–	–	–	$-\frac{1}{4}$

Appendix E

Spectral element pressure correction equations

The operator splitting pressure correction scheme for the solution of the continuous Stokes equations in a bounded domain $\mathbf{\Omega}$, as derived in chapter 5, consists of the following steps:

1. Calculate $\mathbf{u}^*$ from
$$\frac{3}{2}\mathbf{u}^* - \Delta t\mathcal{D}\mathbf{u}^* = 2\mathbf{u}^n - \frac{1}{2}\mathbf{u}^{n-1} - \Delta t\nabla p^n + \Delta t\mathbf{f}^{n+1}. \tag{E.1}$$
2. Calculate p^* from
$$\nabla^2 p^* = \frac{3}{2}\frac{\nabla\cdot\mathbf{u}^*}{\Delta t}. \tag{E.2}$$
3. Calculate $\mathbf{u}^{n+1}$ from
$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{2}{3}\Delta t\nabla p^*. \tag{E.3}$$
4. Calculate p^{n+1} from
$$p^{n+1} = p^n + p^* - \nu\nabla\cdot\mathbf{u}^*. \tag{E.4}$$

For simplicity the following boundary conditions for the pressure correction scheme are assumed

$$\mathbf{u}^* = \mathbf{0},\ \frac{\partial p^*}{\partial n} = 0 \qquad \text{on } \mathbf{\Gamma}. \tag{E.5}$$

The solution space for the velocity is the space $(H^1_{\mathbf{u},0}(\boldsymbol{\Omega}))^2$ defined by

$$H^1_{\mathbf{u},0}(\boldsymbol{\Omega}) = \left\{ v,\ \frac{\partial v}{\partial x_i} \in L^2(\boldsymbol{\Omega})\,|\, v = 0 \text{ on } \boldsymbol{\Gamma} \right\}. \tag{E.6}$$

The solution space for the pressure is the space $H^1_{p,0}(\boldsymbol{\Omega})$ defined by

$$H^1_{p,0}(\boldsymbol{\Omega}) = \left\{ q, \frac{\partial q}{\partial x_i} \in L^2(\boldsymbol{\Omega}) \middle| \frac{\partial q}{\partial n} = 0 \text{ on } \boldsymbol{\Gamma}, \int_{\boldsymbol{\Omega}} q \, d\boldsymbol{\Omega} = 0 \right\}. \tag{E.7}$$

First consider the discretization of step 1. The weighted residual formulation of equation (E.1) reads

Find $\mathbf{u}^* \in H^1_{\mathbf{u},0}(\boldsymbol{\Omega})$ *such that:*

$$\begin{aligned} \left(\frac{3}{2}\mathbf{u}^* - \Delta t \mathcal{D}\mathbf{u}^*, \mathbf{v}\right) &= \left(2\mathbf{u}^n - \frac{1}{2}\mathbf{u}^{n-1}, \mathbf{v}\right) - \Delta t \left(\nabla p^n, \mathbf{v}\right) \\ &+ \Delta t \left(\mathbf{f}^{n+1}, \mathbf{v}\right), \qquad \forall \mathbf{v} \in (H^1_{\mathbf{u},0}(\boldsymbol{\Omega}))^2. \end{aligned} \tag{E.8}$$

Application of Green's formula to the diffusion term leads to the weak or variational formulation

Find $\mathbf{u}^* \in H^1_{\mathbf{u},0}(\boldsymbol{\Omega})$ *such that:*

$$\begin{aligned} \left(\frac{3}{2}\mathbf{u}^* + \Delta t \nu \nabla \mathbf{u}^*, \nabla \mathbf{v}\right) &= \left(2\mathbf{u}^n - \frac{1}{2}\mathbf{u}^{n-1}, \mathbf{v}\right) - \Delta t \left(\nabla p^n, \mathbf{v}\right) \\ &+ \Delta t \left(\mathbf{f}^{n+1}, \mathbf{v}\right), \quad \forall \mathbf{v} \in (H^1_{\mathbf{u},0}(\boldsymbol{\Omega}))^2. \end{aligned} \tag{E.9}$$

Application of the Lax-Milgram lemma, see also chapter 2, yields that problem (E.9) has a unique solution $\mathbf{u}^* \in H^1_{\mathbf{u},0}(\boldsymbol{\Omega})$. The discretization of equation (E.9) is similar to that of the steady convection-diffusion equation described in section 2.3.2 and to that of the unsteady convection-diffusion equation described in section 4.4.2. It is of importance that since the pressure correction scheme is applied to the continuous Stokes equations, the degree of approximation n for velocity and pressure can be taken the same. As a consequence the discretization of the pressure term in equation (E.9) becomes rather simple. The resulting matrix-vector system reads

$$\begin{aligned} \left(\frac{3}{2}\mathrm{M} + \Delta t \mathrm{D}\right) \mathbf{u}^* &= 2\mathrm{M}\mathbf{u}^n - \frac{1}{2}\mathrm{M}\mathbf{u}^{n-1} \\ &- \Delta t \mathbf{Q}\mathbf{p}^n + \Delta t \mathrm{M}\mathbf{f}^{n+1}, \end{aligned} \tag{E.10}$$

or equivalently

$$\begin{aligned}\sum_{e=1}^{n_e}{}' \sum_{p=0}^{n} \sum_{q=0}^{n} \left(\frac{3}{2} M^e_{rspq} + \Delta t D^e_{rspq} \right) \mathbf{u}^*_{pq} \\ = \sum_{e=1}^{n_e}{}' \sum_{p=0}^{n} \sum_{q=0}^{n} 2 M^e_{rspq} \mathbf{u}^n_{pq} - \frac{1}{2} M^e_{rspq} \mathbf{u}^{n-1}_{pq} - \Delta t Q^e_{pqrs} p_{pq} \\ + \Delta t M^e_{rspq} \mathbf{f}^{n+1}_{pq}, \qquad r, s = 0, \ldots, n.\end{aligned} \tag{E.11}$$

Here $\mathbf{M}$ is the mass matrix, $\mathbf{D}$ is the diffusion or stiffness matrix and $\mathbf{Q}$ is the gradient matrix. Explicit expressions for these matrices in an element $\mathbf{\Omega}_e$ are

$$\begin{cases} M^e_{rspq} = \gamma_p \gamma_q \, |\mathbf{J}|_{pq} \, \delta_{pr} \delta_{qs}, \\ D^e_{rspq} = \nu \displaystyle\sum_{k=0}^{n} \sum_{l=0}^{n} \gamma_k \gamma_l \frac{1}{|\mathbf{J}|_{kl}} \tilde{\nabla}_{klpq} \cdot \tilde{\nabla}_{klrs}, \\ Q^e_{rspq} = \gamma_r \gamma_s \, \tilde{\nabla}_{rspq}, \end{cases} \tag{E.12}$$

with γ_p $(p = 0, \ldots, n)$ the Gauss–Lobatto weights, see equation (2.27), $|\mathbf{J}|_{pq}$ $(p, q = 0, \ldots, n)$ the discrete transformation Jacobian, see equation (2.57), and $\tilde{\nabla}_{rspq}$ $(r, s, p, q = 0, \ldots, n)$ the discrete gradient, see equation (2.68).

Next, consider the discretization of step 2. The weak formulation of equation (E.2) is given by

Find $\mathbf{p}^* \in H^1_{p,0}(\mathbf{\Omega})$ *such that:*

$$(\nabla p^*, \nabla q) = -\frac{3}{2} \frac{1}{\Delta t} (\nabla \cdot \mathbf{u}^*, q), \qquad \forall q \in H^1_{p,0}(\mathbf{\Omega}). \tag{E.13}$$

Again, application of the Lax-Milgram lemma yields that problem (E.13) has a unique solution $\mathbf{p}^* \in H^1_{p,0}(\mathbf{\Omega})$. The resulting matrix-vector system reads

$$\mathbf{K}\mathbf{p}^* = -\frac{3}{2} \frac{1}{\Delta t} \mathbf{L}\mathbf{u}^*. \tag{E.14}$$

This can equivalently be written as

$$\begin{aligned}\sum_{e=1}^{n_e}{}' \sum_{p=0}^{n} \sum_{q=0}^{n} K^e_{rspq} p^*_{pq} = -\frac{3}{2} \frac{1}{\Delta t} \sum_{e=1}^{n_e}{}' \sum_{p=0}^{n} \sum_{q=0}^{n} L^e_{rspq} \cdot \mathbf{u}^*_{pq}, \\ r, s = 0, \ldots, n.\end{aligned} \tag{E.15}$$

Here **K** is the pressure stiffness matrix and **L** is the divergence matrix. Explicit expression for these matrices in an element Ω_e are

$$\begin{cases} K^e_{rspq} = \displaystyle\sum_{k=0}^{n}\sum_{l=0}^{n} \gamma_k \gamma_l \frac{1}{|\mathbf{J}|_{kl}} \tilde{\nabla}_{klpq} \cdot \tilde{\nabla}_{klrs}, \\ L^e_{pqrs} = \gamma_r \gamma_s \, \tilde{\nabla}_{pqrs}. \end{cases} \tag{E.16}$$

The discretized versions of step 3 and 4 (equations (E.3) and (E.4)), finally, are straightforward. The complete spectral element pressure correction equations thus read:

1. Solve for $\mathbf{u}^*$ from

$$\begin{aligned} \left(\frac{3}{2}\mathrm{M} + \Delta t \mathbf{D}\right) \mathbf{u}^* &= 2\mathbf{M}\mathbf{u}^n - \frac{1}{2}\mathbf{M}\mathbf{u}^{n-1} \\ &- \Delta t \mathbf{Q}\mathbf{p}^n + \Delta t \mathbf{M}\mathbf{f}^{n+1}. \end{aligned} \tag{E.17}$$

2. Solve for $\mathbf{p}^*$ from

$$\mathbf{K}\mathbf{p}^* = -\frac{3}{2}\frac{1}{\Delta t}\mathbf{L}\mathbf{u}^*. \tag{E.18}$$

3. Calculate $\mathbf{u}^{n+1}$ from

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{2}{3}\Delta t \mathrm{M}^{-1}\mathbf{Q}\mathbf{p}^*. \tag{E.19}$$

4. Calculate $\mathbf{p}^{n+1}$ from

$$\mathbf{p}^{n+1} = \mathbf{p}^n + \mathbf{p}^* - \nu \mathrm{M}^{-1}\mathbf{L}\mathbf{u}^*. \tag{E.20}$$

References

[1] Abramowitz, M. and Stegun, I.A. (eds.) (1972), *Handbook of mathematical functions with formulas, graphs and mathematical tables* (Gov. Printing Office, Washington D.C.)

[2] Axelsson, O. and Barker, V.A. (1984), *Finite element solution of boundary value problems: theory and applications* (Academic Press, Orlando Florida).

[3] Baaijens, F.P.T. (1992), Numerical analysis of unsteady viscoelastic flow, *Comp. Meth. Appl. Mech. Eng.* **94** pp. 285-299.

[4] Babuška, I. (1971), Error-bounds for finite element method, *Numer. Math.* **16** pp. 322-333.

[5] Babuška, I., Szabo, B.A. and Katz, I.N. (1981), The p-version of the finite element method, *SIAM J. Numer. Anal.* **18** pp. 515-545.

[6] Batchelor, G.K. (1967), *An introduction to fluid dynamics* (Cambridge University Press, Cambridge).

[7] Bathe, K. and Dong, J. (1987), Solution of incompressible viscous fluid flow with heat transfer, *Computers and Structures* **26** pp. 17-31.

[8] Bernardi, C., Maday, Y. and Patera A.T. (1990), A new nonconforming approach to domain decomposition: the mortar element method, in: *Nonlinear partial differential equations and their applications (eds. H. Brezis and J.L. Lions)* (Pitman, London).

[9] Brezzi, F. (1974), On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers, *RAIRO* **8** pp. 129-151.

[10] Brooks, A.N. and Hughes, T.J.R. (1982), Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with special emphasis on the incompressible Navier–Stokes equations, *Comp. Meth. Appl. Mech. Eng.* **32** pp. 199-259.

[11] Canuto, C. and Quarteroni, A. (1985), Preconditioner minimal residual methods for Chebyshev spectral calculations, *J. Comp. Phys.* **60** pp. 315-337.

[12] Canuto, C., Hussaini, M.Y., Quarteroni, A. and Zang, T.A. (1988), *Spectral methods in fluid dynamics* (Springer–Verlag, New York, Berlin).

[13] Carey, G.F. and Krishnan, R. (1984), Penalty finite element method for the Navier–Stokes equations, *Comp. Meth. Appl. Mech. Eng.* **47** pp. 183-224.

[14] Chorin, A.J. (1968), Numerical solution of the Navier–Stokes equations, *Math. Comp.* **22** pp. 745-761.

[15] Christie, I., Griffiths, D.F. and Mitchell, A.R. (1976), Finite element methods for second order differential equations with significant first derivatives, *Int. J. Numer. Meth. Eng.* **10** pp. 1389-1396.

[16] Ciarlet, P.G. (1978), *The finite element method for elliptic problems* (North–Holland, Amsterdam, New York).

[17] Concus, P., Golub, G.H. and O'Leary, D.P. (1976), A generalized Conjugate Gradient method for the numerical solution of elliptic partial differential equations, in: *Sparse matrix computations (eds. J.R. Bunch and D.J. Rose)* (Academic Press, New York).

[18] Courant, R. and Hilbert, D. (1953), *Methods of mathematical physics, vol. 1* (Wiley–Interscience, New York).

[19] Cuvelier, C., Segal, A. and Van Steenhoven, A.A. (1986), *Finite element methods and Navier–Stokes equations* (D. Reidel Publishing Company, Dordrecht, Boston, Lancaster, Tokyo).

[20] Dahlquist, G. (1963), A special stability problem for linear multistep methods, *BIT* **3** pp. 27-43.

[21] Davis, P.J. and Rabinowitz, P. (1984), *Methods of numerical integration* (Academic Press, Orlando Florida).

[22] Demaret, P. and Deville, M.O. (1989), Chebyshev pseudospectral solution of the Stokes equation, *J. Comp. Phys.* **83** pp. 463-484.

[23] Deville, M. and Mund, E. (1985), Chebyshev pseudospectral solution of second order elliptic equations with finite element preconditioning, *J. Comp. Phys.* **60** pp. 517-533.

[24] Deville, M.O. and Mund, E.H. (1990), Finite element preconditioning for pseudospectral solutions of elliptic problems, *SIAM J. Sci. Stat. Comp.* **11** pp. 311-342.

[25] Deville, M.O. and Mund, E.H. (1992), Finite element preconditioning of collocation schemes for advection-diffusion equations, in: *Iterative methods in linear algebra, proc. IMACS Int. Symp. iterative methods in linear algebra (eds. R. Beauwens and P. de Groen)* (North–Holland, Amsterdam).

[26] Donea, J., Giulani, S., Laval, H. and Quartapelle, L. (1982), Finite element solution of the unsteady Navier–Stokes equations by a fractional step method, *Comp. Meth. Appl. Mech. Eng.* **30** pp. 53-73.

[27] Donea, J. (1984), A Taylor–Galerkin method for convective transport problems, *Int. J. Numer. Meth. Eng.* **20** pp. 101-119.

[28] Donea, J., Giulani, S., Laval, H. and Quartapelle, L. (1984), Time-accurate solution of advection-diffusion problems by finite elements, *Comp. Meth. Appl. Mech. Eng.* **45** pp. 123-145.

[29] Donea, J. (1991), Generalized Galerkin methods for convection dominated transport phenomena, *Appl. Mech. Rev.* **44** pp. 205-214.

[30] Donea, J. and Quartapelle, L. (1992), An introduction to finite element methods for transient advection problems, *Comp. Meth. Appl. Mech. Eng.* **95** pp. 169-203.

[31] Eisenstat, S.C., Elman, H.C. and Schultz, M.H. (1983), Variational iterative methods for nonsymmetric systems of linear equations, *SIAM J. Numer. Anal.* **20** pp. 345-357.

[32] Fortin, M. and Glowinski, R. (1983), *Augmented Lagrangian methods: applications to the numerical solution of boundary value problems* (North-Holland, Amsterdam, New York).

[33] Fox, L. and Parker, I.B. (1968), *Chebyshev polynomials in numerical analysis* (Oxford University Press, London).

[34] Girault, V. and Raviart, P.A. (1986), *Finite element methods for Navier–Stokes equations* (Springer–Verlag, New York, Berlin).

[35] Gottlieb, D. and Orszag, S.A. (1977), *Numerical analysis of spectral methods* (SIAM, Philadelphia).

[36] Gottlieb, D., Hussaini, M.Y. and Orszag, S.A. (1984), Theory and applications of spectral methods, in: *Spectral methods for partial differential equations (eds. R. Voigt, D. Gottlieb and M.Y. Hussaini)* (SIAM, Philadelphia).

[37] Gresho, P.M. and Sani, R.L. (1987), On pressure boundary conditions for the incompressible Navier-Stokes equations, *Int. J. Numer. Meth. Fluids* **7** pp. 1111-1145.

[38] 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. Part 1: Theory, *Int. J. Numer. Meth. Fluids* **11** pp. 587-620.

[39] Gresho, P.M. and Chan, S.T. (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. Part 2: Implementation, *Int. J. Numer. Meth. Fluids* **11** pp. 621-659.

[40] Gresho, P.M. (1991), Some current CFD issues relevant to the incompressible Navier–Stokes equations, *Comp. Meth. Appl. Mech. Eng.* **87** pp. 201-252.

[41] Hairer, E., Wanner, G. and Noersett, S.P. (1987), *Solving ordinary differential equations I. Nonstiff problems* (Springer, Berlin).

[42] Hairer, E. and Wanner, G. (1991), *Solving ordinary differential equations II. Stiff and differential-algebraic problems* (Springer, Berlin).

[43] Hawken, D.M., Tamaddon-Jahromi, H.R., Townsend, P. and Webster, M.F. (1990), A Taylor-Galerkin based algorithm for viscous incompressible flow, *Int. J. Numer. Meth. Fluids* **10** pp. 327-351.

[44] Hirsch, C. (1988), *Numerical computation of internal and external flows. Volume 1: Fundamentals of numerical discretization* (John Wiley & Sons, Chicester, New York, Brisbane, Toronto, Singapore).

[45] Hughes, T.J.R. (1978), A simple scheme for developing 'upwind' finite elements, *Int. J. Numer. Meth. Eng.* **12** pp. 1359-1365.

[46] Hughes, T.J.R., Franca, L.P. and Hulbert, G.M. (1989), A new finite element method for computational fluid dynamics: VIII. The Galerkin/Least Squares methods for advective-diffusive systems, *Comp. Meth. Appl. Mech. Eng.* **73** pp. 173-189.

[47] Hussaini, M.Y., Kopriva, D.A. and Patera, A.T. (1989), Spectral collocation methods, *Appl. Numer. Anal.* **5** pp. 171-209.

[48] Jiang, C.B. and Kawahara, M. (1993), A three-step finite element method for unsteady incompressible flows, *Comp. Mech.* **11** pp. 355-370.

[49] Johan, Z., Hughes, T.J.R., Mathur, K.K. and Johnsson, S.L. (1992), A data parallel finite element method for computational fluid dynamics on the Connection Machine system, *Comp. Meth. Appl. Mech. Eng.* **99** pp. 113-134.

[50] Johnson, C. (1987), *Numerical solutions of partial differential equations by the finite element method* (Cambridge University Press, Cambridge).

[51] Kaiktsis, L., Karniadakis, G.E. and Orszag, S.A. (1991), Onset of three-dimensionality, equilibria, and early transition in flow over a backward facing step, *J. Fluid Mech.* **231** pp. 501-528.

[52] Van Kan, J. (1986), A second-order accurate pressure-correction scheme for viscous incompressible flow, *SIAM J. Sci. Stat. Comp.* **7** pp. 870-891.

[53] Karniadakis, G.E., Israeli, M. and Orszag S.A. (1991), High-order splitting schemes for the incompressible Navier–Stokes equations, *J. Comp. Phys.* **97** pp. 414-443.

[54] Kim, J. and Moin, P. (1985), Application of a fractional-step method to incompressible Navier–Stokes equations, *J. Comp. Phys.* **59** pp. 308-323.

[55] Ku, D.N. (1983), Hemodynamics and atherogenesis at the human carotid bifurcation, *Ph.D. Thesis* (Georgia Institute of Technology, Georgia).

[56] Lankhorst, A.M. (1991), Laminar and turbulent natural convection in cavities, *Ph.D. Thesis* (Delft University of Technology, Delft).

[57] Laval, H. and Quartapelle, L. (1990), A fractional-step Taylor–Galerkin method for unsteady incompressible flows, *Int. J. Numer. Meth. Fluids* **11** pp. 501-513.

[58] Lax, P.D. and Wendroff, B. (1960), Systems of conservation laws, *Comm. Pure Appl. Math.* **13** pp. 217-237.

[59] Maday, Y., Patera, A.T. and Rønquist, E.M. (1987), A well-posed optimal spectral element approximation for the Stokes problem, *ICASE Report, 87-48* (NASA Langley Research Center, Hampton Virginia).

[60] Maday, Y. and Munoz, F. (1988), Spectral element multigrid. II. Theoretical justification, *J. Sci. Comp.* **3** pp. 323-354.

[61] Maday, Y. and Patera, A.T. (1989), Spectral element methods for the incompressible Navier–Stokes equations, in: *State-of-the-Art surveys on computational mechanics (ed. A. Noor)* (ASME, New York).

[62] Maday. Y., Patera, A.T. and Rønquist, E.M. (1990), An operator-integration-factor splitting method for time-dependent problems: application to incompressible fluid flow, *J. Sci. Comp.* **5** pp. 263-292.

[63] Maday, Y. and Rønquist, E.M. (1990), Optimal error analysis of spectral methods with emphasis on non-constant coefficients and deformed geometries, *Comp. Meth. Appl. Mech. Eng.* **80** pp. 91-115.

[64] Maday, Y., Meiron, D., Patera, A.T. and Rønquist, E.M. (1993), Analysis of iterative methods for the steady and unsteady Stokes problem: application to spectral element discretizations, *J. Sci. Comp.* **14** pp. 310-337.

[65] Marchuk, G.I. (1975), *Methods of numerical mathematics* (Springer, Berlin, Heidelberg, New York).

[66] Mavriplis, C.A. (1989), Nonconforming discretizations and a posteriori error estimators for adaptive spectral element techniques, *Ph.D. Thesis* (Massachusetts Institute of Technology, Cambridge Massachusetts).

[67] Minev, P.D., Timmermans, L.J.P., Van de Vosse, F.N., Rindt, C.C.M. and Van Steenhoven, A.A. (1994), A splitting technique for thermally-driven problems using spectral elements, submitted to *3rd International Conference on advanced computational methods in heat transfer* (Southampton).

[68] Morgan, K., Peraire, J., Peiro, J. and Hassan, O. (1991), The computation of three-dimensional flows using unstructured grids, *Comp. Meth. Appl. Mech. Eng.* **87** pp. 335-352.

[69] Orszag, S.A. (1980), Spectral methods for problems in complex geometries, *J. Comp. Phys.* **37** pp. 70-92.

[70] Orszag, S.A., Isreali, M. and Deville, M.O. (1986), Boundary conditions for incompressible flows, *J. Sci. Comp.* **1** pp. 75-111.

[71] Palmen, D.E.M., Van de Vosse, F.N., Janssen, J.D. and Van Dongen, M.E.H. (1993), The influence of minor stenoses on the flow in the carotid artery bifurcation - Hydrogen-bubble visualisation, accepted for publication in *J. Biomech.*

[72] Patera, A.T. (1984), A spectral element method for fluid dynamics: laminar flow in a channel expansion, *J. Comp. Phys.* **54** pp. 468-488.

[73] Peyret, R. and Taylor, T. (1983), *Computational methods for fluid flow* (Springer, New York).

[74] Phillips, T.N. and Roberts, G.W. (1993), The treatment of spurious pressure modes in spectral incompressible flow calculations, *J. Comp. Phys.* **105** pp. 150-164.

[75] Quarteroni, A. and Zampieri, E. (1992), Finite element preconditioning for Legendre collocation approximations to elliptic equations and systems, *SIAM J. Numer. Anal.* **29** pp. 917-936.

[76] Reid, J.K. (1971), On the method of Conjugate Gradients for the solution of large sparse systems of linear equations, in: *Large sparse sets of linear equations (ed. J.K. Reid)* (Academic Press, London, New York).

[77] Reuderink, P.J., Van de Vosse, F.N., Van Steenhoven, A.A., Van Dongen, M.E.H. and Janssen, J.D. (1993), Incompressible low-speed-ratio flow in non-uniform distensible tubes, *Int. J. Numer. Meth. Fluids* **16** pp. 597-612.

[78] Rindt, C.C.M., Van Steenhoven, A.A., Janssen, J.D. and Segal, A. (1990), A numerical analysis of the steady flow in a three-dimensional model of the carotid artery bifurcation, *J. Biomech.* **23** pp. 461-473.

[79] Rønquist, E.M. and Patera, A.T. (1987), Spectral element multigrid. I. Formulation and numerical results, *J. Sci. Comp.* **2** pp. 389-406.

[80] Rønquist, E.M. (1988), Optimal spectral element methods for the unsteady three-dimensional Navier–Stokes equations, *Ph.D. Thesis* (Massachusetts Institute of Technology, Cambridge Massachusetts).

[81] Saad, Y. and Schultz, M.H. (1986), GMRES: A generalized minimum residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Stat. Comp.* **7** pp. 856-869.

[82] Segal, A. (1984), *SEPRAN user manual, standard problems and programmers guide* (Ingenieursburo SEPRA, Leidschendam).

[83] Sonneveld, P. (1989), CGS, a fast Lanczos-type solver for nonsymmetric linear systems, *SIAM J. Sci. Stat. Comp.* **10** pp. 36-52.

[84] Strikwerda, J. (1980), Iterative methods for the numerical solution of second-order elliptic equations with large first-order terms, *SIAM J. Sci. Sta. Comp.* **1** pp. 119-145.

[85] Tahmaddon–Jahromi, H.R., Ding, D., Webster, M.F. and Townsend, P. (1992), A Taylor–Galerkin finite element method for non-Newtonian flows, *Int. J. Numer. Meth. Fluids* **34** pp. 741-757.

[86] Temam, R. (1984), *Navier-Stokes equations* (North-Holland, Amsterdam, New York).

[87] Tezduyar, T.E., Behr, M., Aliabadi, S.K., Mittal, S. and Ray, S.E. (1992), A new mixed preconditioning method for finite-element computations, *Comp. Meth. Appl. Mech. Eng.* **99** pp. 27-42.

[88] Timmermans, L.J.P. and Van de Vosse, F.N. (1993), Finite element preconditioned spectral element methods for convection-diffusion problems, in: *Topics in applied mechanics: integration of theory and application in applied mechanics (eds. J.F. Dijksman and F.T.M. Nieuwstadt)* (Kluwer Academic Publishers, Dordrecht, Boston, London).

[89] Timmermans, L.J.P. and Van de Vosse, F.N. (1993), Spectral methods for advection-diffusion problems, in: *Notes on numerical fluid mechanics: numerical methods for advection-diffusion problems (eds. C.B. Vreugdenhil and B. Koren)* (Vieweg, Braunschweig).

[90] Timmermans, L.J.P., Van de Vosse, F.N. and Minev, P.D. (1993), Taylor–Galerkin based spectral element methods for convection-diffusion problems, accepted for publication in *Int. J. Numer. Meth. Fluids.*

[91] De Vahl Davis, G. (1983), Natural convection of air in a square cavity: a benchmark numerical solution, *Int. J. Numer. Meth. Fluids* **3** pp. 870-891.

[92] Van der Vorst, H.A. (1992), Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, *SIAM J. Sci. Stat. Comp.* **13** pp. 631-644.

[93] Van de Vosse, F.N. (1987), Numerical analysis of carotid artery flow, *Ph.D. Thesis* (Eindhoven University of Technology, Eindhoven).

[94] Van de Vosse, F.N., Van Steenhoven, A.A., Janssen, J.D. and Reneman, R.S. (1990), A two-dimensional numerical analysis of unsteady flow in the carotid artery bifurcation, *Biorheol.* **27** pp. 163-189.

[95] Vreugdenhil, C.B. and Koren, B. (eds.) (1993), *Notes on numerical fluid mechanics: numerical methods for advection-diffusion problems* (Vieweg, Braunschweig).

[96] Whitham, G.B. (1974), *Linear and nonlinear waves* (Wiley–Interscience, London).

[97] Yanenko, N.N. (1971), *The method of fractional steps* (Springer, New York).

[98] Zang, T.A., Streett, C.L. and Hussaini, M.Y. (1989), Spectral methods for computational fluid dynamics, *ICASE report 89-13* (NASA Langley Research Center, Hampton Virginia).

Samenvatting

Het doel van dit proefschrift is de ontwikkeling van nauwkeurige en efficiënte oplosmethoden voor incompressibele stromingsproblemen met een dominant convectief gedeelte. De reden om spectraal elementen toe te passen is de hoge nauwkeurigheid die bereikt wordt, als 'voldoende' gladde stomingsverschijnselen benaderd worden, binnen een raamwerk van geometrische flexibiliteit.

In het eerste gedeelte van dit proefschrift wordt de spectraal elementen methode nader bekeken. Spectraal elementen methoden zijn hoge-orde p-type gewogen residuen technieken om partiële differentiaal vergelijkingen op te lossen. Ze combineren de geometrische flexibiliteit van de conventionele h-type eindige elementen methode met de hoge nauwkeurigheid van spectraal methoden. Voor een specifieke klasse van basisfuncties tonen spectraal elementen methoden exponentiële convergentie als gladde functies benaderd worden. Dit is een groot voordeel t.o.v. conventionele lage-orde methoden, die slechts algebraisch convergeren. Van groot belang voor een numerieke methode is ook de efficiëntie waarmee het resulterende discrete stelsel opgelost kan worden, zowel met betrekking tot geheugengebruik als rekentijd. Voor een spectraal elementen methode wordt dit op de eerste plaats bereikt door het gebruik van een iteratieve oplosmethode, en in de tweede plaats door het gebruik van een tensoriële basis die afgeleid is van de basis in één dimensie. Het aantal operaties nodig om de residuen in een iteratieve techniek uit te rekenen wordt op deze manier sterk verminderd. Nog een voordeel is dat de volle spectrale systeem matrix niet langer opgeslagen hoeft te worden, hetgeen resulteert in een efficiënt geheugengebruik.

In het tweede gedeelte wordt het oplossen van incompressibele stromingen besproken. Als uitgangspunt wordt gekeken naar algemene tijdsafhankelijke convectie-diffusie problemen. Toepassing van lage-orde methoden leidt vaak tot oscillaties in de numerieke oplossing, als de convectie dominant is. De spectraal elementen methode daarentegen vertoont een zeer geringe numerieke demping en dispersie. Voor tijdsafhankelijke problemen is ook de tijdsintegratie van groot belang. Aangezien convectie en diffusie verschillende verschijnselen zijn, ligt het voor de hand om ze apart te behandelen m.b.v. een 'operator splitting' techniek, waarmee het probleem gesplitst wordt in een puur convectie- en een puur diffusie-probleem. Hierdoor wordt een efficiënte methode bereikt, omdat in dat geval het diffusie-gedeelte opgelost kan worden met een impliciete tijdsintegratie en een grote tijdsstap; het convectie-gedeelte wordt opgelost met een expliciete methode met een kleinere tijdsstap, indien nodig. De spectraal elementen methode biedt hier een groot voordeel door het gebruik van een diagonale massa-matrix, die bijzonder eenvoudig te inverteren is hetgeen leidt tot grote efficiëntie. Het diffusie-gedeelte kan opgelost worden met een iterative procedure.

De volgende stap is het oplossen van de Navier–Stokes vergelijkingen voor incompressibele stromingen. Ook hier wordt de keuze van oplosmethode mede bepaald door de behoefte aan een efficiënte numerieke methode: een gemodificeerde continue druk-correctie methode. Er is gekozen voor een continue methode omdat dan niet aan de Brezzi–Babuška conditie voldaan hoeft te worden, zodat de approximatiegraad voor snelheid en druk gelijk genomen kan worden. De discrete vergelijkingen zijn dan eenvoudig te implementeren. In een druk-correctie methode worden de vergelijkingen gesplitst in een stelsel problemen voor zowel snelheid als druk. Dit resulteert in een convectie-diffusie probleem voor een tijdelijk snelheidsveld, dat opgelost wordt met de 'operator splitting' zoals boven vermeld. Voor de druk-correctie resteert een Poisson vergelijking die opgelost wordt met een eindige elementen gepreconditioneerd iteratief schema. Vervolgens wordt de snelheid gecorrigeerd. Het grote voordeel van een druk-correctie is de garantie dat de benaderde stroming altijd incompressibel is. In dit proefschrift wordt een modificatie op de standaard druk-correctie techniek voorgesteld waarmee een tweede-orde consistent schema afgeleid kan worden. Dan zijn er ook geen problemen met betrekkking tot de formulering van een randvoorwaarde voor de druk-correctie.

Curriculum Vitae

Luc Timmermans

- 6-12-1964: Born in Sittard, The Netherlands.
- 1977–1983: Serviam Scholengemeenschap Sittard: Gymnasium β.
- 1983–1989: Eindhoven University of Technology: Studies at the Department of Mathematics and Computer Science.
- 1989–1994: Eindhoven University of Technology: Research assistant at the Department of Mechanical Engineering, Section of Computational and Experimental Mechanics.

Stellingen

behorende bij het proefschrift

Analysis of spectral element methods

with application to incompressible flow

1. Toepassing van een hoge-orde Galerkin spectraal elementen methode op sterk convectieve problemen leidt ook zonder de toevoeging van een stabilizerende upwind term tot zeer goede resultaten, zelfs in het geval van een sterk niet-lineaire propagatie van een schok.

   - Dit proefschrift, hoofdstukken 2 en 4.
   - Vreugdenhil, C.B. and Koren, B. (eds.), *Numerical methods for advection-diffusion problems* (Vieweg, Braunschweig), 1993.

2. Door toevoeging van een term aan de vergelijking voor de druk op het nieuwe tijdsniveau is de druk-correctie methode consistent met de originele Navier–Stokes vergelijkingen. Dan zijn er ook geen moeilijkheden m.b.t. de formulering van een randvoorwaarde voor de druk.

   - Dit proefschrift, hoofdstuk 5.

3. Het gebruik van een diagonale massa-matrix, in plaats van een consistente massa-matrix, leidt voor een hoge-orde methode tot een veel kleiner verlies van nauwkeurigheid dan voor een lage-orde methode. Als gevolg hiervan biedt de spectraal elementen methode grote voordelen bij het creëren van efficiënte oplosmethoden voor tijdsafhankelijke problemen.

   - Dit proefschrift, hoofdstukken 4 en 5.
   - Timmermans, L.J.P and Van de Vosse, F.N., Spectral methods for advection-diffusion problems, in: *Numerical methods for advection-diffusion problems (eds. C.B. Vreugdenhil and B. Koren)* (Vieweg, Braunschweig), 1993.

4. Verreweg de belangrijkste grootheid in simulaties van incompressibele stromingen is een divergentie-vrij snelheidsveld. Hiervan kan op eenvoudige wijze worden afgeleid: druk, versnelling, vorticiteit, stroomfunctie etc.

   - Dit proefschrift, hoofdstuk 5.
   - Gresho, P.M., Some current CFD issues relevant to the incompressible Navier–Stokes equations, *Comp. Meth. Appl. Mech. Eng.* **87** pp. 201-252, 1991.

5. Onderzoekers die voornamelijk publiceren om een maximaal aantal publicaties te bereiken, tonen een gebrek aan beroepsmoraal.

6. Het is niet duidelijk of de verwaarlozing van het verschil van de impliciete tweede-orde termen in de standaard druk-correctie aanpak toegestaan is, indien zoals door Hawken *et al.* het algorithme toegepast wordt op het continue stelsel Navier–Stokes vergelijkingen.

   - Hawken, D.M., Tamaddon-Jahromi, H.R., Townsend, P. and Webster, M.F., A Taylor-Galerkin based algorithm for viscous incompressible flow, *Int. J. Numer. Meth. Fluids* **10** pp. 327-351, 1990.

7. De onderliggende gedachte bij het gebruik van projectie-methoden voor het oplossen van de Navier–Stokes vergelijkingen is: het mag eigenlijk niet, maar probeer het toch maar.

   - Gresho, P.M., 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. Part 1: Theory, *Int. J. Numer. Meth. Fluids* **11** pp. 587-620, 1990.

8. De goede begeleider vormt een spiegel voor de promovendus. Hij reflecteert en dwingt zo de promovendus zichzelf te evalueren. De promovendus mag onder geen beding worden geleid.

9. De ondraaglijke lichtheid van het bestaan ligt hierin, dat een volledig gebrek aan gewicht of druk het bewijs ontneemt dat men voelt en dus leeft.

   - Milan Kundera, *The unbearable lightness of being* (Unwin Paperbacks, London), 1989.

10. Problemen m.b.t. het diskgebruik binnen een sterk numeriek gerichte vakgroep worden niet opgelost door uitbreiding van de computerfaciliteiten, maar door het regelmatig instrueren en controleren van gebruikers.

Eindhoven, 8 maart 1994 Luc Timmermans