OPTIMAL SPECTRAL ELEMENT METHODS FOR THE UNSTEADY THREE-DIMENSIONAL INCOMPRESSIBLE NAVIER-STOKES EQUATIONS

by

Einar Malvin Rønquist

M.S.,Norwegian Institute of Technology(1980)

Submitted to the Department of Mechanical Engineering
in Partial Fulfillment of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 1988

Signature of Author: _____

Department of Mechanical Engineering
May 15, 1988

Certified by: _____

Professor Anthony T. Patera
Thesis Supervisor

Accepted by: _____

Professor Ain A. Sonin
Chairman, Department Committee on Graduate Studies

OPTIMAL SPECTRAL ELEMENT METHODS FOR THE UNSTEADY THREE-DIMENSIONAL INCOMPRESSIBLE NAVIER-STOKES EQUATIONS

by

Einar Malvin Rønquist

Submitted to the Department of Mechanical Engineering
in partial fulfillment for the requirements for the degree of
Doctor of Philosophy

Abstract

The spectral element method is a weighted-residual technique for the solution of partial differential equations that combines the geometric flexibility of low-order h-type finite element methods with the rapid convergence rate of spectral methods. In the spectral element discretization the computational domain is broken up into macro- spectral elements, and the dependent and independent variables are approximated by N^{th} order tensor-product polynomial expansions within each subdomain. Variational projection operators and Gauss numerical quadrature are used to generate the set of discrete equations. Convergence to the exact solution is achieved by increasing the degree N of the polynomial approximation.

This thesis presents a new optimal-order Legendre spectral element discretization for the unsteady Navier-Stokes equations based on consistent velocity/pressure spaces and a non-dissipative general formulation of the convection operator. The resulting discrete equations are solved by a semi-implicit method in time, with the implicit Stokes operator being treated by a global parallelizable procedure consisting of nested preconditioned conjugate gradient/intra-element multigrid iteration. Numerical results are presented that demonstrate the optimality of the discretization and solvers for the solution of incompressible flow problems.

Thesis Supervisor: Professor Anthony T. Patera
Thesis Committee: Professors Yvon Maday and Ain A. Sonin

Contents

Acknowledgements

I would like to thank my advisors Prof. Anthony T. Patera and Prof. Yvon Maday for all the assistance and helpful suggestions they gave me through this project. Especially I would like to thank Prof. Anthony T. Patera who made my work here at MIT the most interesting and enjoyable I have ever done. I am also endebted to the rest of the members of the fluids lab for creating a good social and working environment.

Chapter 1
Introduction

Spectral element methods are high-order weighted-residual techniques for the solution of partial differential equations that combine the geometric flexibility of h-type finite element methods (Strang and Fix [59]; Ciarlet [15]; Girault and Raviart [22]) and p-type spectral techniques (Gottlieb and Orszag [27] ; Canuto, Hussaini, Quarteroni, and Zang [14]). In the spectral element discretization (Patera [53]) the computational domain is broken up into macro- spectral elements, and the dependent and independent variables are approximated by N^{th} order tensor-product polynomial expansions within each subdomain. Variational projection operators and Gauss numerical quadrature are used to generate the discrete equations, which are then solved by iterative procedures using tensor-product sum-factorization techniques. Convergence to the exact solution is achieved by increasing the degree, N, of the polynomial expansions, while keeping the number of elements, K, fixed.

The spectral element method is one particular approach to spectral methods in complex geometries. As the spectral element method relies heavily on variational projection operators, the technique is very similar to the p-type and h-p-type finite element methods used in solid mechanics (Babuška and Dorr [4]), however the latter do not use the tensor-product sum-factorization technique

which is essential for obtaining efficient iterative solution procedures. There has also been much work on non-variational multi-domain spectral methods for complex geometries based on explicit patching techniques (e.g. Gottlieb and Orszag [27], Métivet and Morchoisne [46], Morchoisne [48], Marion and Gay [43], Macaraeg [35], Macaraeg and Street [36], Laurien and Fasel [34], Métivet [45]). The primary difference between the variational and the non-variational approach is the treatment of the elemental boundaries. The variational approach only requires C^0-continuity across elemental interfaces, while higher-order continuity is obtained as part of the convergence process. The non-variational (collocation) method, however, strongly enforces both C^0- and C^1-continuity across elemental boundaries. This difference in approach significantly affects the theoretical analysis of the methods as well as the implementation.

The spectral element methods are optimal in the sense that the error they incur is, at worst, a multiplicative constant away from the best fit possible in the approximating polynomial subspace (e.g. Funaro [19], Funaro, Quarteroni, and Zanolli [20], Maday and Patera [40]). A high-order approach is therefore particularly attractive when aiming for accurate smooth solutions, e.g. solving incompressible fluid flow problems. For this class of problem the spectral element methods are competitive with h-type finite element methods as regards efficiency, especially if the methods are implemented with optimal (resolution-independent) solvers and in a parallel architecture environment (Fischer [18]). This improvement in efficiency is brought about due to the fact that, although a high-order method requires more work per degrees-of-freedom than a low-order method, for a fixed error requirement, many fewer degrees-of-freedom are

required by spectral element methods than by h-type finite element methods. The increased efficiency is not obtained at the cost of decreased generality: spectral element methods allows for geometric flexibility in representing complex geometries by effecting a combination of local domain decomposition and high-order interpolation. In essence, the spectral element methods provides a general, accurate, and efficient treatment of incompressible flow problems in complex geometries.

In this thesis we discuss spectral element discretization and solution procedures for the incompressible Navier-Stokes equations. In Chapter 2 we discuss the discretization of elliptic and parabolic equations, we consider optimal Stokes discretizations based on consistent velocity/pressure spaces, and we construct a non-dissipative discrete convection operator. In Chapter 3 we present global iterative methods for the solution of the fully discretized semi-implicit Navier-Stokes equations. In particular, we apply a Uzawa decoupling procedure in which we solve the implicit Stokes operator based on a nested conjugate gradient iteration/intra-element multigrid approach.

Chapter 2

Spectral Element Discretizations

2.1 Motivation

In this chapter we develop a set of discrete operators which are used to efficiently and accurately solve (integrate) the three-dimensional, unsteady, incompressible Navier-Stokes equations in a complex geometry Ω,

$$\rho[\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}] = -\nabla p + \mu \nabla^2 \mathbf{u} + \mathbf{f} \quad in\ \Omega, \tag{2.1}$$

$$\nabla \cdot \mathbf{u} = 0 \quad in\ \Omega, \tag{2.2}$$

subject to specified initial and boundary conditions for the velocity $\mathbf{u}$. The Navier-Stokes equations represent a set of nonlinear partial differential equations which combine both temporal and spatial derivatives. The emphasis in this chapter will be on developing discrete operators resulting from high-order spectral element spatial discretizations, while temporal discretization is performed using finite difference techniques.

In order to motivate the particular choice of spatial discretization method we must have a way to compare various solution strategies. We shall here define an optimal numerical scheme as a discretization/solution procedure which can

solve a given problem to a specified accuracy with the minimum computational cost using a specified computer architecture. The particular class of problems we are interested in solving here is multi-dimensional incompressible flow problems in complex geometries where we expect the regularity of the solutions to be high. For convection-dominated flow problems we also need a discretization procedure which can accurately resolve steep gradients like thin boundary layers and internal layers, and which introduces minimal numerical diffusion and dispersion. Spectral methods typically have all these properites, however they lack the generality associated with low-order (h-type) finite element methods in terms of being able to represent problems in complex geometries. Hence, the choice of discretization procedure will in general be a domain decomposition technique in which the computational domain is broken up into K subdomain (elements), and the dependent and independent variables are expanded in terms of polynomials of degree N within each element.

In the following we assume that an appropriate time stepping procedure has been chosen in order to meet numerical stability and accuracy requirements, and we are left with solving a boundary-value problem at each time step. For convenience we now define a discretization pair $h = (K, N)$ associated with the spatial discretization of the problem. A low-order finite element method typically achieves convergence to the exact solution by increasing the number of elements K, keeping the polynomial degree N fixed, e.g. $N = 1$; this convergence strategy results in an algebraic convergence rate. In a spectral element method, however, convergence is obtained by increasing the polynomial degree, N, while keeping the number of elements, K, fixed; this convergence strategy results in

exponential convergence for analytic solution and data. Note that in the case of using only one element ($K = 1$), the spectral element method reduces to a pure spectral technique.

It is clear that there exist many different discretization pairs $h = (K, N)$ which can meet the specified accuracy requirement. In order to choose an optimal discretization we define a work function $W_h^d(\varepsilon)$ as the computational cost associated with solving a given problem in $\mathcal{R}^d$ with a specified accuracy ε and using a discretization $h = (K, N)$. The optimization problem then becomes: Given ε and d, find $h = (K, N)$ such that $W_h^d(\varepsilon)$ is minimized. For the class of problems we wish to solve it is our claim is that even for "engineering accuracy" the optimal choice is neither a low-order finite element method nor a spectral method, but rather somewhere in between; the optimal discretization pair $h = (K, N)$ will in general have values $K \neq 1$ and $N \neq 1$.

To motivate this conclusion we shall do a numerical experiment where we solve the one-dimensional convection-diffusion equation

$$\alpha u_{xx} + u_x = f \tag{2.3}$$

on a domain $x \in]0, 1[$ with Dirichlet boundary conditions $u(0) = 1$ and $u(1) = 0$. The particular choice of the data f and the constant α gives the highly regular, but stongly varying solution u as plotted in Fig. 1. We now choose a discretization pair $h = (K, N)$, compute a numerical solution u_h and the associated H^1-error,

$$\varepsilon_h = \{ \int_0^1 [(u - u_h)^2 + (u_x - u_{h,x})^2] \, dx \}^{1/2}, \tag{2.4}$$

and repeat this operation for many different discretization pairs. The convergence rate is algebraic for a low-order method (N fixed, $K \Rightarrow \infty$), while it is exponential for a high-order method (K fixed, $N \Rightarrow \infty$).

In order to compare the computational cost associated with each discretization $h = (K, N)$, we make some simplifications. First, we assume that optimal iterative techniques are used to solve the systems of discrete equations, that is, the iterative convergence rate is independent of the resolution $h = (K, N)$, see Section 3.3. Hence, the work is essentially proportional to the computational cost associated with one matrix-vector multiplication (at least to leading order). Next, we assume that the particular choice of discretization $h = (K, N)$ to reach a specified accuracy ε for the one-dimensional model problem (2.3) extends to solve multi-dimensional problems in $\mathcal{R}^d$. Specifically, the computational cost scales as $W_h^d \sim \mathcal{O}(KN^{d+1})$, or equivalently, $W_h^d \sim \mathcal{O}(\tilde{N}N)$ where $\tilde{N} = KN^d$ is the total number of degrees-of-freedom. The reason why spectral element methods ($N > 1$) are competitive with low-order finite element methods ($N = 1$) is due to the fact that $\tilde{N}$ is much smaller in a high-order approach.

We now plot in Fig. 2 the locus of work-minimum discretization pairs $h = (K, N)$, parameterized by the accuracy ε. The trajectories are given for d =1, 2 and 3. Essentially, the main conclusions are that as the required accuracy ε increases, it becomes more favorable to go towards a high-order method. Next, for the same specified accuracy the high-order approach becomes more optimal as the dimensionality of the problem increases. Finally, for a given problem (d fixed) and a fixed accuracy, the computational cost increases much more strongly near the (vertical) finite element "axis" than near the (horizontal)

spectral "axis", see Fig. 3.

Although these results are based on solving the simple one-dimensional model problem (2.3), they nevertheless demonstrate the correct trends for a more complex problem. In the case of the solution being locally irregular, we could argue that it will pay off to use enough elements so as to confine the effect of the singularities to within relatively small subregions of the total domain; once this is achieved it will again be favorable to increase N in order to obtain higher accuracy. This discussion will be addressed in more detail in a future paper. Note that the scaling we have used for the computational complexity is valid on serial machines. In the context of parallel architectures high-order methods are favored even more strongly. This can be motivated from the fact that for a given d and ϵ, a high-order method results in less total number of degrees-of-freedom than does a low-order method. Hence, on a parallel machine a high-order approach requires less communication and gives a better load-balancing than an h-type finite element method (Fischer [18]).

The main conclusion from the above discussion is that high-order domain-decomposition techniques are attractive in the context of solving accurately problems with smooth solutions. The spectral element method is one particular approach which combines both the good resolution properties associated with high-order discretizations with the geometric flexibility typically associated with low-order methods. The spectral element discretization of (2.1) and (2.2) can be expressed in matrix form as

$$\rho \underline{B}(\frac{\underline{u}_i^{n+1} - \underline{u}_i^n}{\Delta t}) + \rho \sum_{k=0}^{2} \alpha_k \underline{C}^{n-k} \underline{u}_i^{n-k} = \underline{D}_i^T \underline{p}^{n+1} - \mu \underline{A}\, \underline{u}_i^{n+1} + \underline{B}\, \underline{f}_i, \quad i = 1,2,3, \tag{2.5}$$

$$\underline{D}_i \underline{u}_i = 0, \tag{2.6}$$

where $\underline{B}$ is a diagonal mass matrix (the variational equivalent of the identity operator), $\underline{C}$ is the discrete convection operator, $\underline{D}_i, i = 1,2,3$ is the discrete gradient operator, T denotes transpose, $\underline{A}$ represents the discrete Laplace operator, and $\alpha_k, k = 0,1,2$ are constants. In the rest of this chapter we shall develop and analyze each of the discrete spatial operators $\underline{A}$, $\underline{B}$, $\underline{C}$, and $\underline{D}_i$, together with combinations of these. We shall start in Section 2.2 with the spectral element discretization of the elliptic operator, followed by Section 2.3 where we extend the method to solve the heat equation (parabolic). In Section 2.4 we consider optimal saddle Stokes discretizations, and in Section 2.5 we contruct the discrete convection operator (hyperbolic). Finally, in Section 2.6 we shall see how we put all the operators together in order to solve the full Navier-Stokes equations.

2.2 Elliptic Equations

2.2.1 One-Dimensional Problems

To illustrate the basic spectral element concepts we consider the following one-dimensional elliptic Helmholtz problem: Find u(x) defined over the interval

$\Lambda =]-1,1[$ such that

$$-(pu_x)_x + \lambda^2 u = f \qquad x \in \Lambda, \tag{2.7}$$

with homogeneous Dirichlet boundary conditions,

$$u(-1) = u(1) = 0. \tag{2.8}$$

Here subscript x denotes differentiation with respect to x, and we assume that $p(x) > p_0 > 0$, and $\lambda \in \mathcal{R}$. The equation (2.7) shares many of the common features of more complicated elliptic and saddle (Stokes) systems, yet it is sufficiently simple to allow for a clear illustration of the spectral element discretization of elliptic equations. As we shall see later, in order to solve a conduction problem or a Stokes problem, unsteady or steady, we are ultimately faced with solving an elliptic boundary value problem of the form (2.7). In the case of an implicit time stepping procedure, the constant λ^2 typically plays the role of the inverse of the time step.

The basis for our numerical scheme is the variational form of (2.7): Find $u \in \mathcal{H}_0^1(\Lambda)$ such that

$$a(u,v) = (f,v) \qquad \forall v \in \mathcal{H}_0^1(\Lambda) \tag{2.9}$$

where

$$\forall \phi, \psi \in \mathcal{L}^2(\Lambda) \qquad (\phi,\psi) = \int_\Lambda \phi(x)\psi(x)dx, \tag{2.10}$$

$$\forall \phi, \psi \in \mathcal{H}_0^1(\Lambda) \qquad a(\phi,\psi) = \int_\Lambda [\phi_x(x)\psi_x(x) + \lambda^2 \phi(x)\psi(x)]dx. \tag{2.11}$$

Here the function space $\mathcal{L}^2(\Lambda)$ is the space of all functions which are square integrable over Λ, while $\mathcal{H}_0^1(\Lambda)$ is the space of all functions which are square

integrable, whose derivatives are also square integrable over Λ, and which satisfy the homogeneous boundary conditions (2.8).

The spectral element method proceeds by first specifying the discretization pair $h = (K, N)$, and breaking up the interval Λ into K subintervals (spectral elements),

$$\Lambda = \cup_{k=1}^{K} \Lambda_k \tag{2.12}$$

where Λ_k is defined by $a_k \leq x \leq a_k + b$ (for convenience we here choose the elements to be of equal length b). The space of approximation of the solution u is then taken to be a subspace X_h of $\mathcal{H}_0^1(\Lambda)$ consisting of all piecewise high-order polynomials of degree $\leq N$,

$$X_h = Y_h \cap \mathcal{H}_0^1(\Lambda), \tag{2.13}$$

where

$$Y_h = \{\phi \in \mathcal{L}^2(\Lambda),\ \phi_{|\Lambda_k} \in \mathcal{P}_N(\Lambda_k)\}, \tag{2.14}$$

and $\mathcal{P}_N(\Lambda_k)$ is the space of all polynomials of degree $\leq N$ on the interval Λ_k.

The spectral element discretization corresponds to numerical quadrature of the variational form (2.9) restricted to the subspace X_h: Find $u_h \in X_h$ such that

$$a_{h,GL}(u_h, v_h) = (f, v_h)_{h,GL} \qquad \forall v_h \in X_h, \tag{2.15}$$

where $(\cdot,\cdot)_{h,GL}$ and $a_{h,GL}(\cdot,\cdot)$ refer to Gauss-Lobatto quadrature of the inner products defined in (2.10) and (2.11) respectively,

$$(\phi, \psi)_{h,GL} = \frac{b}{2} \sum_{k=1}^{K} \sum_{n=0}^{N} \rho_n \phi(\xi_{n,k}) \psi(\xi_{n,k}), \tag{2.16}$$

$$a_{h,GL}(\phi, \psi) = \frac{b}{2} \sum_{k=1}^{K} \sum_{n=0}^{N} \rho_n \phi_x(\xi_{n,k}) \psi_x(\xi_{n,k}). \tag{2.17}$$

Here the $\xi_{n,k} = a_k + (\xi_n + 1)b/2$, $0 \leq n \leq N$, $1 \leq k \leq K$, are the locations of the local nodes $\{n,k\}$, and the ξ_n, ρ_n, $0 \leq n \leq N$, are the Gauss-Lobatto Legendre quadrature points and weights, respectively (Stroud and Secrest [60]). Note that with the N^{th}-order Gauss-Lobatto quadrature we can integrate exactly polynomials of degree $2N-1$ or less (Stroud and Secrest [60]), implying that for $p(x) = constant$, $a_{h,GL}(u_h, v_h)$ is integrated exactly while $(f, v_h)_{h,GL}$ is not.

A detailed discussion of error estimates for the spectral element discretization (2.15) is given in Maday and Patera [40]. Here, we shall only state the result for the case $p(x) = p_0 = constant$,

$$\| u - u_h \|_1 \leq C\{N^{1-\sigma} \| u \|_\sigma + N^{\frac{1}{2}-\rho} \| f \|_\rho\} \tag{2.18}$$

for $u \in \mathcal{H}_0^\sigma(\Lambda)$ and $f \in \mathcal{H}_0^\rho(\Lambda)$ and where C is a constant. The sources of the discretization error $\| u - u_h \|_1$ are approximation errors, interpolation errors and quadrature errors. From (2.18) we conclude that the spectral element solution u_h converges to the exact solution u as $N \Rightarrow \infty$ for K fixed, with faster than algebraic (exponential) convergence obtaining for infinitely smooth (analytic) data and solutions. This result should be contrasted to finite element h-type methods (N fixed, $K \Rightarrow \infty$) for which only algebraic convergence results.

In order to implement (2.15) it is necessary to choose a basis for the polynomial piecewise-smooth space X_h in (2.13). The particular choice of basis does not effect the (optimal) error estimate (2.18), however it effects the conditioning and structure of the resulting set of algebraic equations, which is very important as regards the computational complexity associated with iterative solution techniques, see Chapter 3. The choice of basis is also important as regards

parallelism since it effects the inter-elemental couplings and hence the communication requirements. We choose an interpolant Gauss-Lobatto Legendre basis to represent $w_h \in X_h$,

$$w_h(x) = \cup_{k=1}^{K} \sum_{p=0}^{N} w_p^k h_p(r) \qquad x \in \Lambda_k \Rightarrow r \in \Lambda \tag{2.19}$$

$$h_p \in \mathcal{P}_N(\Lambda), \quad h_p(\xi_q) = \delta_{pq} \quad \forall p,q \in \{0,...,N\}^2, \tag{2.20}$$

where $w_p^k = w_h(\xi_{p,k})$ is the value of w_h at local node $\{p,k\}$, and δ_{pq} is the Kronecker-delta symbol. Explicit expressions for the Lagrangian interpolants h_i are given in the Appendix. To honor the $\mathcal{H}_0^1$ requirement and the essential boundary conditions (2.8) we further require that

$$w_N^k = w_0^{k+1} \qquad \forall k \in \{1,...,K-1\} \tag{2.21}$$

$$w_0^1 = w_N^K = 0, \tag{2.22}$$

respectively. In addition to a basis for X_h it will also be convenient to represent functions (such as the inhomogeneity f) which are in Y_h but not in X_h. A function $w_h \in Y_h$ is also expressed in terms of the interpolant basis (2.19), but as $w_h \in Y_h$ need not be in $\mathcal{H}^1$ or satisfy the homogeneous boundary conditions (2.8), we do not require the additional conditions (2.21) and (2.22).

The solution (u_h), the testfunctions (v_h), and the data (f) are now all expressed in terms of the nodal basis (2.19) and inserted into (2.15). Choosing each testfunction v_h to be nonzero (unity) at only one global Gauss-Lobatto Legendre collocation point and zero at all the other collocation points, we arrive at the final discrete matrix statement,

$$\sum_{k=1}^{K}{}' \sum_{q=0}^{N} H_{pq}^k u_q^k = \sum_{k=1}^{K}{}' \sum_{q=0}^{N} B_{pq}^k f_q^k, \tag{2.23}$$

where $f_q^k = f(\xi_{q,k})$, and

$$H_{pq}^k = A_{pq}^k + \lambda^2 B_{pq}^k \qquad \forall p,q \in \{0,...,N\}^2 \tag{2.24}$$

$$A_{pq}^k = \frac{2}{b}\sum_{n=0}^{N} \rho_n D_{np} D_{nq} \qquad \forall p,q \in \{0,...,N\}^2 \tag{2.25}$$

$$B_{pq}^k = \frac{b}{2}\rho_p \delta_{pq} \qquad \forall p,q \in \{0,...,N\}^2 \tag{2.26}$$

$$D_{pq} = \frac{dh_q}{dr}(\xi_p) \qquad \forall p,q \in \{0,...,N\}^2. \tag{2.27}$$

Here $\sum'$ denotes "direct stiffness" summation, in which contributions from local nodes $\{p,k\}$ which are physically coincident are summed (enforcing (2.21)), and contributions from local nodes $\{p,k\}$ which correspond to domain boundary points (here $x = \pm 1$) are masked to zero (enforcing (2.22)). In terms of global representation we can also write the spectral element discretization (2.23) of the Helmholtz problem (2.7-2.8) in matrix form as

$$\underline{H}\,\underline{u} = \underline{B}\,\underline{f}, \tag{2.28}$$

$$\underline{H} = \underline{A} + \lambda^2 \underline{B}, \tag{2.29}$$

where $\underline{u}$ is a vector of nodal unknowns, $\underline{f}$ is a vector of given (interpolated) data, $\underline{A}$ is the discrete Laplace-operator, $\underline{H}$ is the discrete Helmholtz-operator, and $\underline{B}$ is a diagonal mass matrix.

To demonstrate numerically the rapid convergence rate promised by error estimates of the form (2.18) we consider the problem (2.7-2.8) with $p(x) = e^x$, $f(x) = e^x(cosx - sinx)$, $\lambda = 0$ on $x =]0,\pi[$, for which the solution is $u = -sinx$. As all the data and the solution are analytic, we expect exponential convergence to the exact solution. This is demonstrated in Fig. 4 where the $\mathcal{L}^\infty$ error is

plotted as a function of N for $K = 2$. It is seen that when the resolution is doubled the error is squared, consistent with exponential convergence. Note that the discrete H^1 error $\| u - u_h \|_{1,GL}$ also goes to zero exponentially fast.

Although the discrete solution u_h is only continuous and not C^1 across elemental boundaries ($u_h \in X_h$), the result in Fig. 4 indicates that the jumps in derivatives at elemental boundaries are correctly taken care of by the variational statement. In fact, it is shown in Maday and Patera [40] that C^1-continuity is weakly imposed by the variational form, which should be contrasted with the spectral collocation (patching) method where C^1-continuity is strongly imposed. In the interior of an element, however, the spectral element method is equivalent to a collocation procedure. To see this we consider the case of $p(x) = 1$ and $\lambda = 0$ in (2.7). We then use the exactness of the Gauss quadrature formula (2.17) to integrate the left-hand-side of (2.15) by parts, and choose the test function v_h to be unity at only one global collocation point. Equation (2.15) now reduces to

$$-u^k_{h,xx}(\xi_{q,k}) = f^k(\xi_{q,k})$$

$$\forall q \in \{1, ..., N-1\}, \quad \forall k \in \{1, ..., K\} \tag{2.30}$$

$$\rho_0[-u^k_{h,xx}(\xi_{0,k+1}) - f^{k+1}(\xi_{0,k+1})] + \rho_N[-u^k_{h,xx}(\xi_{N,k}) - f^k(\xi_{N,k})] =$$

$$[u^{k+1}_{h,x}(\xi_{0,k+1}) - u^k_{h,x}(\xi_{N,k})]$$

$$\forall k \in \{2, ..., K-1\} \tag{2.31}$$

where we have assumed that all the elements are of the same length. Thus in the interior of an element the method is equivalent to a collocation procedure, while on the elemental boundaries the spectral element discretization "naturally"

generates a weak C^1-condition on the solution (note $\rho_0, \rho_N \sim O(1/N^2)$). We will return to this issue in the multi-dimensional case.

2.2.2 Two-Dimensional Problems

Rectilinear Geometry

The spectral element discretization of multi-dimensional elliptic equations corresponds to a tensor-product extension of the one-dimensional method. We consider here the two-dimensional Poisson equation on a rectilinear domain Ω with homogeneous Dirichlet boundary conditions on the domain boundary $\partial\Omega$,

$$-\Delta u = f \qquad in\ \Omega, \tag{2.32}$$

$$u = 0 \qquad on\ \partial\Omega. \tag{2.33}$$

Here $\Delta = \nabla^2$ is the Laplace operator, and a point in Ω is denoted $\mathbf{x} = (x_1, x_2)$. The more general multi-dimensional case in complex geometry is discussed in the next section. The variational statement equivalent to (2.32-2.33) is: Find $u \in \mathcal{H}_0^1(\Omega)$ such that

$$\int_\Omega \nabla u \cdot \nabla v \, d\mathbf{x} = \int_\Omega f v \, d\mathbf{x} \qquad \forall v \in \mathcal{H}_0^1(\Omega), \tag{2.34}$$

or, in abstract form

$$a(u, v) = (f, v) \qquad \forall v \in \mathcal{H}_0^1(\Omega). \tag{2.35}$$

The spectral element discretization proceeds by breaking up the domain Ω into K disjoint rectilinear elements in (x_1, x_2),

$$\overline{\Omega} = \cup_{k=1}^{K} \overline{\Omega}_k, \qquad \Omega_k =]a_k, a_k'[\times]b_k, b_k'[, \tag{2.36}$$

such that the intersection of two adjacent elements is either a whole edge or a vertex, see Fig. 5. We then require that the variational statement (2.35) be satisfied for a polynomial subspace of $\mathcal{H}_0^1$ defined on the Ω_k. As in the one-dimensional case, we first define the space

$$\mathcal{P}_{N,K}(\Omega) = \{\Phi \in \mathcal{L}^2(\Omega); \Phi_{|\Omega_k} \in \mathcal{P}_N(\Omega_k)\} \tag{2.37}$$

where $\mathcal{P}_N(\Omega_k)$ denotes the tensor-product space of all polynomials of degree less than or equal to N with respect to each spatial variable x_1 and x_2. The spectral element space X_h consists of

$$X_h = \mathcal{H}_0^1 \cap \mathcal{P}_{N,K}(\Omega). \tag{2.38}$$

The discrete problem is then given by: Find u_h in X_h such that

$$\int_\Omega \nabla u_h \cdot \nabla v_h \, d\mathbf{x} = \int_\Omega f v_h \, d\mathbf{x} \qquad \forall v_h \in X_h. \tag{2.39}$$

To proceed with the numerical quadrature/interpolation we use an affine mapping: $x_1 \Rightarrow r_1, x_2 \Rightarrow r_2, \mathbf{x} \in \Omega_k \Rightarrow \mathbf{r} \in \Lambda \times \Lambda$, and perform tensor-product Gauss-Lobatto Legendre $\times$ Gauss-Lobatto Legendre quadrature in (r_1, r_2) to obtain from (2.39)

$$\sum_{k=1}^{K}{}' \sum_{p=0}^{N} \sum_{q=0}^{N} \nabla u_h(\xi_{p,k}^1, \xi_{q,k}^2) \cdot \nabla v_h(\xi_{p,k}^1, \xi_{q,k}^2) \rho_{p,k}^1 \rho_{q,k}^2 =$$

$$\sum_{k=1}^{K}{}' \sum_{p=0}^{N} \sum_{q=0}^{N} f v_h(\xi_{p,k}^1, \xi_{q,k}^2) \rho_{p,k}^1 \rho_{q,k}^2 \qquad \forall v_h \in X_h \tag{2.40}$$

or

$$a_{h,GL}(u_h, v_h) = (f, v_h)_{h,GL} \qquad \forall v_h \in X_h. \tag{2.41}$$

Here the $\xi_{p,k}^1, \xi_{q,k}^2$ are the Gauss-Lobatto points, $\xi_{p,k}^1 = a_k + (a_k' - a_k)(\xi_p + 1)/2$, $\xi_{q,k}^2 = b_k + (b_k' - b_k)(\xi_q + 1)/2$, $\rho_{p,k}^1, \rho_{q,k}^2$ are the quadrature weights from (2.16)

including the "geometry", $\rho^1_{p,k} = \rho_p(a'_k - a_k)/2$, $\rho^2_{p,k} = \rho_p(b'_k - b_k)/2$, and ∇ still refers to the gradient operator with respect to the x variable. Theoretical error estimates of the form (2.18) can also be obtained for the multi-dimensional case, see Maday and Patera [40], Funaro [19].

The last remaining point as regards implementation is the choice of basis. For reasons of efficiency, tensor-product forms are virtually required in high-order methods (see Section 3.1), and thus the basis for $w_h \in X_h$ follows naturally from the one-dimensional case as

$$w^k_h(r_1, r_2) = \sum_{i=0}^{N}\sum_{j=0}^{N} w^k_{ij} h_i(r_1) h_j(r_2) \qquad \mathbf{x} \in \Omega_k \Rightarrow \mathbf{r} \in \Lambda \times \Lambda, \tag{2.42}$$

where the h_i are the one-dimensional Gauss-Lobatto Lagrangian interpolants defined in (2.20) and $w^k_{ij} = w_h(\xi^1_{i,k}, \xi^2_{j,k})$. Similar to the one-dimensional case we must also require the polynomials to be C^0-continuous across elemental boundaries and enforce the homogeneous Dirichlet boundary conditions for $w_h \in X_h$. Expressing the discrete solution $u_h \in X_h$, the testfunctions $v_h \in X_h$, and the data $f \in Y_h$ in terms of the basis (2.42) and choosing v_h to be nonzero at only one global collocation point, the discrete formulation (2.41) becomes

$$\sum_{k=1}^{K}{}' \sum_{m=0}^{N}\sum_{n=0}^{N} (A^k_{im} B^k_{jn} + B^k_{im} A^k_{jn}) u^k_{mn} =$$

$$\sum_{k=1}^{K}{}' \sum_{m=0}^{N}\sum_{n=0}^{N} B^k_{im} B^k_{jn} f^k_{mn} \qquad \forall i,j \in \{0,\ldots,N\}^2 \tag{2.43}$$

or

$$\underline{A}\,\underline{u} = \underline{B}\,\underline{f}. \tag{2.44}$$

The one-dimensional elemental matrices A^k_{pq}, B^k_{pq} in (2.43) are defined as in (2.25-2.26), whereas $\underline{A}$, $\underline{B}$ refer to the global multi-dimensional K-element versions of

the discrete Laplace-operator and mass matrix, respectively. The critical feature of (2.43) is the tensor-product factorization of the Laplace-operator that follows from tensor product integration and basis.

In addition to greatly increased complexity and coupling in the matrix equations in two dimensions, there are other noteworthy differences from the one-dimensional case. First, by using the properties of Gauss-quadrature the equivalence of (2.41) to a collocation procedure for points internal to an element can again be obtained. Although in one dimension the interface condition is relatively simple, at cross-points between four elements in two dimensions the weak C^1 condition naturally generated by the variational approach is less trivial. This indicates the simplicity of the variational approach as compared to the "patching" approach; in the latter, the sense in which the normal derivative is to be interpreted at internal corners is less obvious.

Second, whereas in one dimension singularities can enter only through the coefficients or forcing terms of the problem, in two space dimensions even with analytic data the solution can suffer singularities if the domain boundary $\partial\Omega$ is not smooth. Although these singularities are often relatively weak in practice, they must nevertheless be considered when judging the suitability of a high-order approach. In a spectral method ($K = 1$) the singularities will effect the solution over the entire domain, however in a spectral element method ($K > 1$) the effect will be localized to within the elements where the singularities occur. It should also be noted that in the *h-p* type finite element treatment of elliptic problems, exponential convergence can be achieved even in the presence of singularities by effecting suitable refinement procedures (Babuška and Dorr [4]).

Third, whereas geometry does not enter into one-dimensional problems, in higher space dimensions this is often the critical issue. In the discretization procedure we construct isoparametric spectral elements, thus allowing for local elemental mappings similar to, but not as general, as those used in h-type finite element methods. For instance, a tensor-product basis for triangular elements has not yet been established. The next section explains in more detail how problems in deformed geometries are treated.

Deformed Geometry

We consider now the two-dimensional Poisson equation defined in a deformed domain Ω with homogeneous Dirichlet boundary conditions on the domain boundary $\partial\Omega$,

$$-\Delta u = f \qquad in\ \Omega \tag{2.45}$$

$$u = 0 \qquad on\ \partial\Omega. \tag{2.46}$$

The equivalent variational formulation is: Find $u \in \mathcal{H}_0^1(\Omega)$ such that

$$a(u,v) = (f,v) \qquad \forall v \in \mathcal{H}_0^1(\Omega) \tag{2.47}$$

The spectral element discretization proceeds by breaking up the domain into K (generally deformed) disjoint *quadrilateral* elements,

$$\overline{\Omega} = \cup_{k=1}^{K} \overline{\Omega}_k, \tag{2.48}$$

such that the intersection of two adjacent elements is either a whole edge or a vertex. The discrete equations are then generated by applying tensor-product Gauss-Lobatto Legendre numerical quadrature of the variational form (2.47)

restricted to the subspace X_h, where the polynomial subspace $X_h \in \mathcal{H}_0^1$ is defined in (2.37-2.38). The discrete problem then becomes: Find $u_h \in X_h$ such that

$$a_{h,GL}(u_h, v_h) = (f, v_h)_{h,GL}, \tag{2.49}$$

We now rewrite the integrals in (2.49) as a sum of contributions from all the elements Ω_k,

$$\sum_{k=1}^{K} a^k(u,v) = \sum_{k=1}^{K} (f,v)^k, \tag{2.50}$$

where

$$a^k(u,v) = \int_{\Omega_k} \nabla u \cdot \nabla v \, dxdy, \tag{2.51}$$

and

$$(f,v)^k = \int_{\Omega_k} fv \, dxdy. \tag{2.52}$$

The elemental integrals are now mapped into a local (r,s)-system, see Fig. 6: $(x,y) \in \Omega_k \Rightarrow (r,s) \in \Lambda \times \Lambda$, $\Lambda =]-1,1[$, and we rewrite (2.51) as

$$a^k(u,v) = \int_{-1}^{1}\int_{-1}^{1} \tilde{\nabla} u^k(r,s) \cdot \tilde{\nabla} v^k(r,s) \mid J^k(r,s) \mid \, drds \tag{2.53}$$

where $\tilde{\nabla}$ now denotes differentiation with respect to the the local (r,s)-coordinates,

$$\tilde{\nabla} = (\frac{\partial r}{\partial x}\frac{\partial}{\partial r} + \frac{\partial s}{\partial x}\frac{\partial}{\partial s})\hat{e}_x + (\frac{\partial r}{\partial y}\frac{\partial}{\partial r} + \frac{\partial s}{\partial y}\frac{\partial}{\partial s})\hat{e}_y, \tag{2.54}$$

J is the Jacobian,

$$J = (\frac{\partial x}{\partial r}\frac{\partial y}{\partial s} - \frac{\partial x}{\partial s}\frac{\partial y}{\partial r}), \tag{2.55}$$

and superscript k refers to integration over subdomain Ω_k. In (2.54) we note that

$$\frac{\partial r}{\partial x} = \frac{1}{J}\frac{\partial y}{\partial s}, \quad \frac{\partial s}{\partial x} = -\frac{1}{J}\frac{\partial y}{\partial r}, \quad \frac{\partial r}{\partial y} = \frac{1}{J}\frac{\partial x}{\partial r}, \quad \frac{\partial s}{\partial y} = -\frac{1}{J}\frac{\partial x}{\partial s}, \tag{2.56}$$

and we rewrite (2.53) as

$$a^k(u,v) = \int_{-1}^{1}\int_{-1}^{1} \frac{1}{\mid J^k \mid} \hat{\nabla} u^k \cdot \hat{\nabla} v^k \, dr ds, \tag{2.57}$$

where

$$\hat{\nabla} = (y_s \partial_r - y_r \partial_s)\hat{e}_x + (x_r \partial_s - x_s \partial_r)\hat{e}_y, \tag{2.58}$$

and for simplicity we write $x_r = \frac{\partial x}{\partial r}$, $\partial_s = \frac{\partial}{\partial s}$, etc. Similarly, the integral (2.52) can be expressed as

$$(f,v)^k = \int_{\Omega_k} fv \, dxdy = \int_{-1}^{1}\int_{-1}^{1} f^k v^k \mid J^k \mid dr ds. \tag{2.59}$$

The spectral element method now proceeds by writing (2.49) as

$$\sum_{k=1}^{K} a^k_{h,GL}(u_h, v_h) = \sum_{k=1}^{K} (f, v_h)^k_{h,GL}. \tag{2.60}$$

As for the rectilinear case we use a tensor-product interpolant basis to represent a function $w_h \in X_h$,

$$w_h^k(r,s) = \sum_{i=0}^{N}\sum_{j=0}^{N} w_{ij}^k h_i(r) h_j(s), \tag{2.61}$$

where w_{ij}^k is the value of w_h at the (local) point (ξ_i, ξ_j) inside element k, and h_i is the N^{th}-order Lagrangian interpolant through the Gauss-Lobatto Legendre points,

$$h_i(\xi_j) = \delta_{ij}, \tag{2.62}$$

$$\frac{dh_j}{dr}(\xi_i) = D_{ij}. \tag{2.63}$$

The geometry (x,y), the solution (u_h), the test functions (v_h), and the data (f) are all expressed in terms of the nodal basis (2.61) and inserted into (2.60). The elemental integrals in (2.60) then become

$$a^k_{h,GL}(u_h, v_h) = \rho_p \rho_q \frac{1}{\mid J^k_{pq} \mid} \hat{\nabla} u_h^k(\xi_p, \xi_q) \cdot \hat{\nabla} v_h^k(\xi_p, \xi_q) \tag{2.64}$$

where

$$\hat{\nabla} u_h^k(\xi_p, \xi_q) = \hat{\nabla}^k_{pqmn} u^k_{mn}, \tag{2.65}$$

$$\hat{\nabla}_{pqmn} = [(y_s)^k_{pq} D_{pm}\delta_{qn} - (y_r)^k_{pq}\delta_{pm}D_{qn}]\hat{e}_x + [(x_r)^k_{pq}\delta_{pm}D_{qn} - (x_s)^k_{pq}D_{pm}\delta_{qn}]\hat{e}_y, \tag{2.66}$$

$$(x_r)^k_{pq} = D_{pm}\delta_{qn}x^k_{mn} = D_{pm}x_{mq}, \tag{2.67}$$

$$(x_s)^k_{pq} = \delta_{pm}D_{qn}x^k_{mn} = D_{qn}x^k_{pn}, \tag{2.68}$$

$$(y_r)^k_{pq} = D_{pm}\delta_{qn}y^k_{mn} = D_{pm}y_{mq}, \tag{2.69}$$

$$(y_s)^k_{pq} = \delta_{pm}D_{qn}y^k_{mn} = D_{qn}y^k_{pn}, \tag{2.70}$$

$$J^k_{pq} = (x_r)^k_{pq} \cdot (y_s)^k_{pq} - (x_s)^k_{pq} \cdot (y_r)^k_{pq}, \tag{2.71}$$

and summation over repeated indicies is assumed. Similarly,

$$(f, v_h)^k_{h,GL} = \rho_p\rho_q \mid J^k_{pq} \mid f^k_{pq}v^k_{pq}. \tag{2.72}$$

Note that in the case of deformed geometry we incur quadrature errors both in the evaluation of $a_{h,GL}(u_h, v_h)$ and $(f, v_h)_{h,GL}$, however for smooth solutions they are roughly of the same order as the approximation errors and interpolation errors.

Choosing the test functions v_h to be nonzero (unity) at only one global collocation point, and zero at all other collocation points, we can write (2.60) as

$$\sum_{k=1}^{K} A^k_{ijmn}u^k_{mn} = \sum_{k=1}^{K} B^k_{ijmn}f^k_{mn}, \tag{2.73}$$

where

$$A^k_{ijmn} = \rho_p\rho_q \frac{1}{J_{pq}} \hat{\nabla}^k_{pqij} \cdot \hat{\nabla}^k_{pqmn}, \tag{2.74}$$

and

$$B^k_{ijmn} = \rho_i \rho_j \mid J^k_{ij} \mid \delta_{im}\delta_{jn}. \tag{2.75}$$

Although the resulting matrix equations are more complicated than (2.43) they retain their tensor product form. The C^0-mappings (2.61) are sufficient to insure that the spectral element subspace X_h remains in $H^1(\Omega)$.

The isoparametric spectral element case is relatively simple to implement, however there are many new theoretical issues which arise related to the effect of mapping on interpolation and quadrature (Métivet [45]). Although no general error bound is currently available, the numerical evidence to date indicates that spectral accuracy is achieved even in relatively deformed elements. As an example we consider the problem (2.45) on the domain $\Omega = (x_1 \in]0,1[, x_2 \in]0, 1 + \frac{1}{4}sin\pi x_1[)$ with $f = 0$, $u = sinx_1 \cdot e^{-x_2}$. The $K = 4$ spectral element discretization used is shown in Fig. 7, in which the mapping for spectral elements $k = 3, 4$ corresponds to a simple stretching in x_2 of the mapped domain $\Lambda \times \Lambda \Rightarrow \Omega_k$. It is clear that in more complicated domains more sophisticated mappings are required; to date both differential (elasticity) and (Gordon-Hall) blending methods (Gordon and Hall [26]) have been used with success (Métivet [45], Ho [29]).

We plot in Fig. 8 the $\mathcal{L}^\infty$-error as a function of the total number of degrees-of-freedom in one spatial direction, $N_t = 2N + 1$, demonstrating exponential convergence to the analytic solution (Rønquist and Patera [56]). Note that although the domain Ω is relatively deformed compared to the mapped rectilinear problem, high accuracy is nevertheless obtained with relatively few points due to the good interpolation properties of Legendre-Lagrangian interpolants as

applied to piecewice smooth boundaries $\partial\Omega$.

To illustrate the importance of resolving the geometry we again consider the solution of the problem (2.45) with data $f = 0$ and solution $u = sinx_1 \cdot e^{-x_2}$. In Fig. 9 we plot the $\mathcal{L}^\infty$-error as a function of the total number of degrees-of-freedom in one spatial direction, N_t, but now for three different geometries and discretizations. We note that the top element in case (A) in Fig. 9 has a sharp corner on the top edge, resulting in algebraic convergence as N_t is increased. In case (B) the top element is also deformed, however the edges are smooth and exponential convergence is obtained. In case (C) the domain (A) is split into $K = 4$ elements so as to obtain elemental boundaries which coincides with the piecewise smooth boundary $\partial\Omega$. Note that we this time obtain exponential convergence to the exact solution, demonstrating both the necessity and power of a domain decomposition approach.

Finally, we remark that the results presented here for two-dimensional deformed geometry are readily extended to three dimensions. Also note that in the context of the full Stokes problem it appears that (slightly) subparametric elements are preferred to the isoparametric elements presented here, see Section 2.4.

2.3 Parabolic Equations

The spectral element discretization procedure for solving elliptic equations can readily be applied to solve parabolic problems. Consider the unsteady heat

equation

$$\frac{\partial u}{\partial t} = \Delta u + f, \tag{2.76}$$

on a domain Ω with specified initial and boundary conditions. It can be shown that applying a high-order spatial discretization procedure, the spectral radius of the discrete second-order Laplace operator scales like $\lambda_{A,max} \sim O(K_1^2 N^4)$ for large K_1 and N, where K_1 denotes the number of elements in a typical spatial direction and N is the polynomial degree. Note that that λ_A here refers to the eigenvalues of $\underline{A}$ with respect to the mass matrix $\underline{B}$, and *not* with respect to the identity matrix $\underline{I}$. This means that the time step in an explicit time integration scheme must be smaller than $\Delta t \sim O(K_1^{-2} N^{-4})$, which is a severe stability condition. Although K_1 may be large in a low-order finite element method ($N = 1$), the factor N^4 resulting from a high-order scheme will in general give a more severe time step restriction.

In order to avoid this severe time step constraint in the context of spectral element discretizations, an implicit Euler backward time integration scheme is used (we assume here that a sufficiently efficient iterative solver exists, see Sections 3.2 and 3.3),

$$\frac{u^{n+1} - u^n}{\Delta t} = (\Delta u)^{n+1} + f^{n+1}, \tag{2.77}$$

and the spectral element discretization of (2.77) can then be written in matrix form as

$$\underline{H}\,\underline{u} = \underline{B}\,\underline{g}, \tag{2.78}$$

where

$$\underline{H} = \underline{A} + \frac{1}{\Delta t}\underline{B}, \tag{2.79}$$

and

$$\underline{g} = \underline{f} + \frac{\underline{u}^n}{\Delta t}. \tag{2.80}$$

Here $\underline{u}$ is the vector of nodal unknowns, $\underline{f}$ is a vector of given (interpolated) data, $\underline{A}$ is the discrete Laplace-operator, $\underline{H}$ is the discrete Helmholtz-operator, and $\underline{B}$ is the diagonal mass matrix. We see that we recover equation (2.7), with λ^2 playing the role of the inverse of the time step, implying that for each time step we have to solve an elliptic boundary value problem. As an example of a spectral element solution to a multi-dimensional parabolic problem, we refer to Rønquist and Patera [56] where the method is used to solve the Stefan problem. Here, the technique is extended to also account for a time-dependent geometry, that is, the geometry is computed as part of the solution.

We close this section by remarking that although implicit time integration seems to be natural when solving parabolic equations of the form (2.76), for convection dominated parabolic equations, like the Navier-Stokes equations (2.1-2.2) in the high Reynolds number limit, the time step restriction due to the Courant condition might be a more severe restriction, and hence an explicit time integration scheme is more appropriate. We shall return to this issue in Section 2.5.

2.4 The Stokes Problem

2.4.1 Steady Stokes

Two-Dimensional Rectilinear Geometry

In this section we consider the two-dimensional steady Stokes problem: Find a velocity $\mathbf{u}$ and a pressure p in a rectangularly decomposable domain $\Omega =]-1,1[^2$ such that

$$-\mu\Delta\mathbf{u} + \nabla p = \mathbf{f} \qquad in\ \Omega, \tag{2.81}$$

$$-div\mathbf{u} = 0 \qquad in\ \Omega, \tag{2.82}$$

subject to homogeneous Dirichlet velocity boundary conditions on the domain boundary $\partial\Omega$,

$$\mathbf{u} = 0 \qquad on\ \partial\Omega \tag{2.83}$$

Here $\mathbf{f}$ is the prescribed force and μ is the viscosity. We will further denote a point in Ω as $\mathbf{x} = (x_1, x_2)$. The more general multi-dimensional case in complex geometry is discussed in the next section.

The spectral element discretization of (2.81-2.83) is based the equivalent variational formulation: Find (u,p) in $X \times M$ such that

$$\mu(\nabla\mathbf{u}, \nabla\mathbf{w}) - (p, div\mathbf{w}) = (\mathbf{f}, \mathbf{w}) \qquad \forall\mathbf{w} \in X, \tag{2.84}$$

$$-(q, div\mathbf{u}) = 0 \qquad \forall q \in M, \tag{2.85}$$

where

$$\forall\phi, \psi \in \mathcal{L}^2(\Omega) \qquad (\phi, \psi) = \int_\Omega \phi(\mathbf{x})\psi(\mathbf{x})\, d\mathbf{x}. \tag{2.86}$$

The space for the velocity, X, and the space for the pressure, M, are given as

$$X = [\mathcal{H}_0^1(\Omega)]^2, \tag{2.87}$$

$$M = \mathcal{L}_0^2(\Omega), \qquad \mathcal{L}_0^2 = \{\phi \in \mathcal{L}^2(\Omega) \ \int_\Omega \phi dx = 0\}. \tag{2.88}$$

Since the pressure is only determined up to a constant, we impose the average pressure to be zero.

The discrete formulation of the Stokes problem consists of choosing two discrete approximation spaces, one (X_h) for the velocity and one (M_h) for the pressure. In Bernardi, Maday and Métivet [8] the choice of X_h and M_h as polynomial subspaces of X and M, respectively, is presented in the context of the spectral collocation approximation. It is pointed out that naïvely choosing X_h and M_h to be subspaces of polynomial degree less than or equal to the same constant N leads to an ill-posed problem polluted by spurious pressure modes. For the collocation method an algorithm based on staggered grids is proposed and analyzed in Bernardi and Maday [7] that avoids this problem; the extension of these ideas leads to the well-posed spectral element method we present here (Maday, Patera, and Rønquist [42], Maday and Patera [40], Maday, Patera, and Rønquist [41].

The spectral element discretization of (2.81-2.83) proceeds by first breaking up the domain into K disjoint rectilinear elements in (x_1, x_2),

$$\overline{\Omega} = \cup_{k=1}^{K} \overline{\Omega}_k, \qquad \Omega_k =]a_k, a_k'[\times]b_k, b_k'[, \tag{2.89}$$

such that the intersection of two adjacent elements is either an edge or a vertex. We then require that the variational statement (2.84-2.85) be satisfied for

polynomial subspaces $X_h \subset X$ and $M_h \subset M$ defined on the Ω_k. We choose the subspaces to be

$$X_h = X \cap \mathcal{P}_{N,K}(\Omega) \tag{2.90}$$

$$M_h = M \cap \mathcal{P}_{N-2,K}(\Omega), \tag{2.91}$$

where

$$\mathcal{P}_{N,K}(\Omega) = \{\Phi \in \mathcal{L}^2(\Omega); \Phi_{|\Omega_k} \in \mathcal{P}_N(\Omega_k)\} \tag{2.92}$$

and $\mathcal{P}_N(\Omega_k)$ denotes the space of all polynomials of degree less than or equal to N with respect to each spatial variable x_1 and x_2. We see that the polynomial degree for the pressure is two order lower than for the velocity. Also note that the velocity is continuous across the elemental boundaries, while the pressure is discontinuous.

Next, we define over each subdomain (spectral element) two integration rules, one by taking the tensor-product of Gauss formulas, and the other one by taking the tensor-product of Gauss-Lobatto formulas. First, we denote the (local) Gauss nodes and weights as the pair (ζ_i, ω_i) for $i = 1, ..., N-1$, and the (local) Gauss-Lobatto nodes and weights as the pair (ξ_i, ρ_i) for $i = 0, ..., N$, and then define the global quadrature points and weights as

$$\zeta^1_{i,k} = a_k + (a'_k - a_k)(\zeta_i + 1)/2, \qquad \omega^1_{i,k} = \omega_i \cdot (a'_k - a_k)/2, \tag{2.93}$$

$$\zeta^2_{i,k} = b_k + (b'_k - b_k)(\zeta_i + 1)/2, \qquad \omega^2_{i,k} = \omega_i \cdot (b'_k - b_k)/2, \tag{2.94}$$

$$\xi^1_{i,k} = a_k + (a'_k - a_k)(\xi_i + 1)/2, \qquad \rho^1_{i,k} = \rho_i \cdot (a'_k - a_k)/2, \tag{2.95}$$

$$\xi^2_{i,k} = b_k + (b'_k - b_k)(\xi_i + 1)/2, \qquad \rho^2_{i,k} = \rho_i \cdot (b'_k - b_k)/2, \tag{2.96}$$

We can now define the two integration rules for all $(\phi, \psi) \in [C^0(\overline{\Omega})]^2$,

$$(\phi, \psi)_{h,G} = \sum_{k=1}^{K} \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} \phi(\zeta_{i,k}^1, \zeta_{j,k}^2) \psi(\zeta_{i,k}^1, \zeta_{j,k}^2) \omega_{i,k}^1 \omega_{j,k}^2, \tag{2.97}$$

$$(\phi, \psi)_{h,GL} = \sum_{k=1}^{K} \sum_{i=0}^{N} \sum_{j=0}^{N} \phi(\xi_{i,k}^1, \xi_{j,k}^2) \psi(\rho_{i,k}^1, \rho_{j,k}^2) \omega_{i,k}^1 \omega_{j,k}^2. \tag{2.98}$$

The discrete formulation corresponding to the variational formulation (2.84-2.85) can then be stated as: Find $(\mathbf{u}_h, p_h)$ in $X_h \times M_h$ such that

$$\mu(\nabla \mathbf{u}_h, \nabla \mathbf{w}_h)_{h,GL} - (p_h, div \mathbf{w}_h)_{h,G} = (\mathbf{f}, \mathbf{w}_h)_{h,GL}, \qquad \forall \mathbf{w}_h \in X_h, \tag{2.99}$$

$$-(q_h, div \mathbf{u}_h)_{h,G} = 0, \qquad \forall q_h \in M_h \tag{2.100}$$

In Maday, Patera and Rønquist [42] a detailed theoretical analysis of the discrete formulation (2.99-2.100) is performed in the case of semi-periodic boundary conditions; the dependent variables $(\mathbf{u}, p)$ and the data $\mathbf{f}$ are in this case written in terms of Fourier series in the periodic x_2-direction, a procedure which decouples all the Fourier modes for the Stokes problem. The resulting equations for one Fourier mode are then discretized using spectral element discretization in the non-periodic x_1-direction, see the next section. For the semi-periodic case it is further shown in Maday, Patera, and Rønquist [42] that the discrete solution $(\mathbf{u}_h, p_h)$ is unique, and that the following error estimate holds for the velocity $\mathbf{u}_h$,

$$\| \mathbf{u} - \mathbf{u}_h \|_1 \leq C[N^{1-\sigma}(\| \mathbf{u} \|_\sigma + \| p \|_{\sigma-1}) + N^{1-\rho} \| \mathbf{f} \|_\rho]. \tag{2.101}$$

Here, it is assumed that $\mathbf{u} \in [\mathcal{H}^\sigma(\Lambda)]^2, p \in \mathcal{H}^{\sigma-1}(\Lambda)$, and $\mathbf{f} \in [\mathcal{H}^\rho(\Lambda)]^2$, and the domain Ω is defined as $\Omega =]-1,1[\times]0,2\pi[$, and $\Lambda =]-1,1[$. An error estimate for the pressure is also deduced, and in Maday, Patera, and Rønquist [41] optimal error bound for the velocity is obtained for the non-periodic homogeneous

problem (2.81-2.83). For spectral element discretizations of Stokes problems in rectilinear geometry the main conclusions from the theoretical analysis can be summarized as: (i) The discrete solution is unique, that is, there exist no spurious pressure modes, and (ii) for analytic solution and data exponential convergence to the exact solution is obtained as the polynomial degree, N, is increased for fixed number of elements, K.

In order to implement (2.99-2.100) we must define bases for the solution space $X_h \times M_h$. As for the spectral element discretization of elliptic equations, due to reasons of computational efficiency, we expand the solution in tensor-product form

$$\mathbf{u}_h^k(r_1, r_2) = \sum_{i=0}^{N} \sum_{j=0}^{N} \mathbf{u}_{ij}^k \, h_i(r_1) h_j(r_2) \qquad \mathbf{x} \in \Omega_k \Rightarrow \mathbf{r} \in \Lambda \times \Lambda. \tag{2.102}$$

Here $\mathbf{u}_{ij}^k = \mathbf{u}_{ij}(\xi_{i,k}^1, \xi_{j,k}^2)$ is the velocity at the tensor-product Gauss-Legendre Lobatto points $(\xi_{i,k}^1, \xi_{j,k}^2)$ in subdomain (spectral element) Ω_k, and h_i and h_j are the N^{th} order Lagrangian interpolants through the Lobatto points. The data is expanded similarly to the velocity,

$$\mathbf{f}_h^k(r_1, r_2) = \sum_{i=0}^{N} \sum_{j=0}^{N} \mathbf{f}_{ij}^k \, h_i(r_1) h_j(r_2) \qquad \mathbf{x} \in \Omega_k \Rightarrow \mathbf{r} \in \Lambda \times \Lambda, \tag{2.103}$$

however the pressure is represented at the Gauss points,

$$p_h^k(r_1, r_2) = \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} p_{ij}^k \, \tilde{h}_i(r_1) \tilde{h}_j(r_2) \qquad \mathbf{x} \in \Omega_k \Rightarrow \mathbf{r} \in \Lambda \times \Lambda. \tag{2.104}$$

Here p_{ij}^k is the pressure at the tensor-product Gauss-Legendre points $(\zeta_{i,k}^1, \zeta_{j,k}^2)$ in subdomain (spectral element) Ω_k, and the $\tilde{h}_i$ are the $(N-2)^{th}$ order Lagrangian interpolants through the Gauss points. In Fig. 10 the staggererd mesh is illutrated for $K = 1$ and $N = 7$. Note that in contrast to the finite difference case,

the velocity and pressure collocation points are not shifted relative to each other in a regular fashion. In fact, we see from Fig. 10 that in some regions there are only velocity points with no pressure points in between. In the finite difference case a regularly spaced staggered mesh follows naturally from approximating the differential operators with finite differences. In the spectral element method the staggered mesh is used to represent the (Lagrangian interpolant) bases associated with two different high-order polynomial subspaces, and there is no need for the mesh points to be regularly spaced.

The expansions (2.102-2.104) are now inserted into (2.99-2.100) and the discrete equations are generated by choosing test functions $w_h \in X_h$ in (2.99) which are unity at a single $\xi_{i,k}$ and zero at all other Gauss-Lobatto Legendre points, and test functions $q_h \in M_h$ in (2.100) which are unity at a single $\zeta_{i,k}$ and zero at all other Gauss Legendre points. The resulting set of discrete equations can then be written in matrix form as

$$\mu \underline{A}\,\underline{u}_i - \underline{D}_i^T \underline{p} = \underline{B}\,\underline{f}_i \qquad i = 1,2, \tag{2.105}$$

$$-\underline{D}_i \underline{u}_i = 0. \tag{2.106}$$

Here $\underline{u}_i, i = 1,2$ and $\underline{p}$ represent vectors of nodal unknowns, $\underline{A}$ is the discrete Laplace-operator, $\underline{B}$ is the diagonal mass matrix, $\underline{D}_i, i = 1,2$ is the discrete gradient opertor, and T indicates the transpose operator. The linear algebraic system (2.105-2.106) corresponds to a symmetric saddlepoint problem, see Section 3.4.

We now present numerical results for the spectral element discretization of (2.81-2.82) for a two-dimensional problem with specified Dirichlet boundary con-

ditions. The extension of (2.99-2.100) to non-homogeneous boundary conditions does not cause significant difficulties if the boundary conditions are compatible with a solenoidal velocity field. In particular, we look at the following simple test problem:

$$\mathbf{u} = (u_1, u_2) = (1 - x_2^2, 0) \tag{2.107}$$

$$p = sin\pi x_1 \cdot sin\pi x_2 \tag{2.108}$$

$$\mathbf{f} = (f_1, f_2) = (2 + \pi cos\pi x_1 \cdot sin\pi x_2, \pi sin\pi x_1 \cdot cos\pi x_2), \tag{2.109}$$

on the domain $\Omega =]-1,1[^2$ with $\mathbf{u} = (u_1, u_2)$ specified on the domain boundary. Note that both the solution and the data are analytic. The domain is broken up into $K = 4$ similar subdomains (spectral elements), and the numerical solution is compared to the exact soution for different values of the polynomial degree N. Fig. 11 shows the maximum pointwise error in the velocity and the pressure as a function of the total number of degree-of-freedom in one spatial direction. As expected from the infinitely smooth nature of the solution, the error decreases exponentially fast as N increases for fixed K. Note that although the velocity can be approximated exactly by a second-order polynomial, we still incur an error in $\mathbf{u}_h$ partly because of the coupling between the velocity and the pressure, and partly because the data $\mathbf{f}$ cannot be interpolated exactly.

To solve a problem in more complex geometry we present some results for creeping flow in a wedge. The extension to complex geometry can be derived by following a similar procedure as described for elliptic problems. The geometry for the wedge is shown in Fig. 12a together with the spectral element discretization ($K = 30, N = 8$). The imposed velocity boundary conditions are no-slip conditions on the two vertical side walls, while a unit horizontal velocity is im-

posed along the top (horizontal) side. The spectral element solution in form of streamlines is shown in Fig. 12b, and for comparison we have also included a flow visualization from an experiment by Taneda [62], Fig. 12c. Based on a similarity solution Moffatt [47] derived asymptotic results for the strength and the location of an "infinite" number of eddies. He predicts that in the direction from the top to the bottom of the wedge, the strength of the next eddy should asymptotically be about 405 times weaker than the previous one (for a total wedge angle of 28.5 degrees). With the particular discretization used in this example we are able to resolve 4 eddies. The ratio of the strength of two successive eddies in the computed solution is (from top to bottom) 386, 406 and 411, all in good agreement with Moffatt's asymptotic results. Note that already the third eddy is so weak that is has not been seen experimentally. Moffatt also shows that the distance from the bottom of the wedge to the center of the next eddy is about half the distance to the previous one, again in good agreement with our results.

As a third example we consider parallel Stokes flow past a sphere. This is an axisymmetric problem, and the spectral element discretization procedure for the Cartesian case needs to be modified. This is done in Rønquist, Patera and Deville [58], and we shall here just state the main results. First, the variational form will have an r-weighting, where r is the radial distance from centerline. Second, subparametric representation of the geometry is necessary to ensure the absence of spurious pressure modes. Third, the boundary conditions for the two velocity components are different.

The geometry we consider in this example is shown in Fig. 13. The length

of the domain is 30, and the height is 15. The radius of the sphere is $a = \sqrt{2}$. On the domain boundary we impose exact velocity boundary conditions (White [66]). No-slip conditions are imposed on the surface of the sphere, while on the axis zero Dirichlet condition is imposed for the radial velocity, and zero Neumann condition is imposed for the axial velocity. From the spectral element solution we integrate the total stress over the surface of the sphere to get the total drag. The exact solution for the drag is given as $F = 6\pi\mu U a$ (White [66]), where μ is the viscosity, U is the (axial) free stream velocity far away from the sphere, and a is the radius of the sphere. The computed drag is compared to the exact solution, and in Fig. 14 we plot the relative error as a function of the polynomial degree N, keeping the number of elements, K, fixed. Again, due to the smooth nature of the solution and the geometry, exponential convergence is obtained.

Two-dimensional Semi-Periodic Case

In this section we consider the Stokes problem (2.81-2.82) defined on the domain $\Omega =]-1,1[\times]0,2\pi[$, with semi-periodic boundary conditions of the form

$$\forall x_2 \in]0,2\pi[\qquad \mathbf{u}(-1,x_2) = \mathbf{u}(1,x_2) = 0 \tag{2.110}$$

$$\forall x_1 \in]-1,1[\qquad \mathbf{u}(x_1,0) = \mathbf{u}(x_1,2\pi). \tag{2.111}$$

A detailed theoretical study of this problem can be found in Maday, Patera, and Rønquist [42], however we shall here present the main numerical results. First, we write the dependent variables $(\mathbf{u}, p)$ and the data $\mathbf{f}$ in terms of Fourier series

in the periodic x_2-direction,

$$\mathbf{u}(x_1, x_2) = \sum_{k=-\infty}^{\infty} \hat{\mathbf{u}}^n(x_1) e^{ikx_2}, \tag{2.112}$$

$$p(x_1, x_2) = \sum_{k=-\infty}^{\infty} \hat{p}^n(x_1) e^{ikx_2}, \tag{2.113}$$

$$\mathbf{f}(x_1, x_2) = \sum_{k=-\infty}^{\infty} \hat{\mathbf{f}}^n(x_1) e^{ikx_2}, \tag{2.114}$$

where $i = \sqrt{-1}$. The Fourier representation (2.112-2.114) results in a set of decoupled equations for each Fourier mode k, which are discretized using spectral element discretization in the non-periodic x_1-direction.

To demonstrate numerically the convergence rate in the semi-periodic case, we consider two test problems in which the viscosity μ is equal to unity and the Fourier index $k = 1$ in (2.112-2.114). For the first test problem the exact Fourier coefficients for the solution and data are given as

$$\hat{\mathbf{u}} = (\hat{u}_1, \hat{u}_2) = (-(1 + cos\pi x_1), i\, sin\pi x_1) \tag{2.115}$$

$$\hat{p} = sin\pi x_1 \tag{2.116}$$

$$\hat{\mathbf{f}} = (\hat{f}_1, \hat{f}_2) = (-(1 + cos\pi x_1)/\pi, i\,(2 + \pi^2) sin\pi x_1), \tag{2.117}$$

while the exact Fourier coefficients for the solution and data for the second test problem are

$$\hat{\mathbf{u}} = (\hat{u}_1, \hat{u}_2) = (-(1 + cos\pi x_1), i\, sin\pi x_1) \tag{2.118}$$

$$\hat{p} = sin\pi x_1 + \mid x_1 - 1/2 \mid^{\gamma+2/3} \tag{2.119}$$

$$\hat{f}_1 = [-(1 + cos\pi x_1)/\pi + (\gamma + 2/3) sgn(x_1 - 1/2) \mid x_1 - 1/2 \mid^{\gamma-1/3}] \tag{2.120}$$

$$\hat{f}_2 = i\,[(2 + \pi^2) sin\pi x_1 + \mid x_1 - 1/2 \mid^{\gamma+2/3}] \tag{2.121}$$

on a domain Λ where $x_1 \in \Lambda =]-1,1[$. Note that the solution and the data in the first test problem are analytic, while the regularity of the second test problem is determined by the value of γ, which is assumed to be an integer. In the first test problem Λ is divided into $K = 2$ equal subintervals Λ_1 and Λ_2, while in the second test problem only $K = 1$ element is used. Convergence is achieved by increasing the order of the polynomial expansions, N.

In the first test problem we obtain exponential convergence as the order N is increased. Fig. 15 shows the error in the velocity and the pressure as a function of the total number of degrees-of-freedom (Gauss-Lobatto Legendre points) in the x_1-direction. This rapid convergence rate is expected due to the fact that the solution and the data are analytic. In the second test problem we obtain algebraic convergence as the order N of the polynomial expansions is increased. Fig. 16 shows the error in the pressure as a function of the total number of degrees-of-freedom (Gauss-Lobatto Legendre points) in the x_1-direction for $\gamma = 3$ and $\gamma = 5$. The convergence rate is given approximately as $N^{-(\gamma+1)}$, which in this particular case is slightly better than the theoretical estimate of $N^{\frac{1}{2}-\gamma}$ predicted by (2.101).

Multi-Dimensional Deformed Geometry

Using the idea of tensor-product expansions the spectral element discretization for the two-dimensional steady Stokes problem can readily be extended to solve three-dimensional problems in rectilinear geometry. We have further extended the method to allow for treatment of general deformed geometries.

The only subtlety as regards the general-geometry formulation for the Stokes problem is insuring the absence of spurious pressure modes, and preserving the (physical) hydrostatic mode $p = constant$. To this end we use *subparametric* mappings, that is, the geometry is represented in the space

$$G_h = [\mathcal{P}_{N-2,K} \cap \mathcal{H}^1(\Omega)]^3. \tag{2.122}$$

If the pressure is constant, $p_h = p_0$, the choice (2.122) for the geometry space insures that the pressure term in (2.99) is zero,

$$(p_0, div\mathbf{w})_{h,G} = 0 \qquad \forall \mathbf{w} \in X_h, \tag{2.123}$$

for arbitrary geometries, as should be the case given the similar property of the continuous equations. This result follows from the fact that all the quadratures in the scalar product (2.123) are exact, and integration by parts is allowed. Hence, the subparametric mapping preserve the hydrostatic pressure mode. Although we currently have no proofs as to lack of spurious modes in the general geometry $(\mathbf{u}_h, p_h, \mathbf{x}_h) \in X_h \times M_h \times G_h$ formulation, we have yet to encounter difficulties in practice.

2.4.2 Unsteady Stokes

The unsteady Stokes equations are given by

$$-\mu\Delta\mathbf{u} + \nabla p + \rho\frac{\partial \mathbf{u}}{\partial t} = \mathbf{f} \qquad in\ \Omega, \tag{2.124}$$

$$-div\mathbf{u} = 0 \qquad in\ \Omega, \tag{2.125}$$

with boundary and initial conditions

$$\mathbf{u} = \mathbf{0} \qquad on\ \partial\Omega, \tag{2.126}$$

$$\mathbf{u}(\mathbf{x}, t = 0) = g(\mathbf{x}) \qquad in\ \Omega. \tag{2.127}$$

Here all the variables are defined as for the steady case, with t representing time, and ρ the density of the fluid. Alhough there are physical problems in which the unsteady Stokes equations are relevant (e.g. high frequency phenomena for which the convective terms are small, but the unsteady terms are not), the unsteady problem is primarily of interest as regards its role in unsteady Navier-Stokes calculations.

We proceed directly to the time-discretization of (2.124-2.127) by an implicit Euler backward method (readily extended to Crank-Nicolson),

$$-\mu\Delta\mathbf{u}^{n+1} + \nabla p^{n+1} + \rho\frac{\mathbf{u}^{n+1} - \mathbf{u}^{n}}{\Delta t} = \mathbf{f}^{n+1}, \tag{2.128}$$

$$-div\mathbf{u}^{n+1} = 0, \tag{2.129}$$

in which $(\mathbf{u}^{n+1}, p^{n+1}) = (\mathbf{u}(\mathbf{x}, n\Delta t), p(\mathbf{x}, n\Delta t))$, and Δt is the time step. The variational form is given by: Find $(\mathbf{u}^{n+1}, p^{n+1}) \in X \times M$ such that

$$\mu(\nabla\mathbf{u}^{n+1}, \nabla\mathbf{w}) - (p^{n+1}, div\mathbf{w}) + \rho(\frac{\mathbf{u}^{n+1} - \mathbf{u}^{n}}{\Delta t}, \mathbf{w}) = (\mathbf{f}^{n+1}, \mathbf{w}) \qquad \forall\mathbf{w} \in X, \tag{2.130}$$

$$-(div\mathbf{u}^{n+1}, q) = 0 \qquad \forall q \in M. \tag{2.131}$$

As for the steady case we will consider discretizations based on (2.130-2.131): Find $(\mathbf{u}_h^{n+1}, p_h^{n+1}) \in X_h \times M_h$ such that

$$\mu(\nabla\mathbf{u}_h^{n+1}, \nabla\mathbf{w}_h)_{h,GL} - (p_h^{n+1}, div\mathbf{w}_h)_{h,G} + \rho(\frac{\mathbf{u}_h^{n+1} - \mathbf{u}_h^{n}}{\Delta t}, \mathbf{w}_h)_{h,GL} = (\mathbf{f}^{n+1}, \mathbf{w}_h)_{h,GL}$$

$$\forall w_h \in X_h, \tag{2.132}$$

$$-(div u_h^{n+1}, q_h)_{h,G} = 0 \qquad \forall q_h \in M_h, \tag{2.133}$$

where X_h and M_h are the same polynomial subspaces of X and M as relevant to the steady problem (2.81-2.83). In matrix form (2.132-2.133) are written as:

$$(b_k' - b_k)\mu \underline{A}\,\underline{u}_i^{n+1} - \underline{D}_i^T \underline{p}^{n+1} + \rho \underline{B}(\frac{\underline{u}_i^{n+1} - \underline{u}_i^n}{\Delta t}) = \underline{B}\,\underline{f}_i^{n+1} \qquad i = 1,2,3, \tag{2.134}$$

$$-\underline{D}_i \underline{u}_i^{n+1} = 0. \tag{2.135}$$

Defining the discrete $\mathcal{L}^2$-norm $\| \underline{\phi} \|_{0,GL}$ based on Gauss-Lobatto quadrature as

$$\| \underline{\phi} \|_{0,GL} = (\underline{\phi}^T \underline{B} \underline{\phi})^{1/2}, \tag{2.136}$$

we note that for any discretization for which $\underline{A}$ is symmetric positive-definite (2.134-2.135) are unconditionally stable,

$$\| \underline{u}^{n+1} \|_{0,GL} < \| \underline{u}^n \|_{0,GL}, \tag{2.137}$$

as can be readily demonstrated by multiplying (2.134) with $\underline{u}_i^{n+1}$ and (2.135) by $\underline{p}^{n+1}$, respectively.

2.5 Hyperbolic Equations

The main objective with this section is to construct an accurate spatial discretization of the convection operator in a convection or a convection-diffusion transport equation. Ultimately we will use this operator to represent the advective (nonlinear) term in the *incompressible* Navier-Stokes equations, in which accurate representation of the convection operator is a critical issue when aiming

for accurate *smooth* solutions. In the treatment of diffusion problems and saddle Stokes problem the question about smooth solutions is usually no critical issue (except for maybe induced singularities due to the geometry). In the treatment of convection problems, however, the issue about smoothness deserves extra comment. The key point in the justification of using a high-order method is the fact that we are essentially looking at problems where derivative information about the solution is also important. The solutions may include steep gradients like thin boundary layers in the limit of small viscosity in convection-diffusion problems, however we are always interested in resolving these steep gradients. In fact, when using a high-order discretization scheme it is virtually required to resolve all the structure in the solutions, otherwise global breakdown will occur in form of wiggles. We emphasize that in the following, all our initial conditions are assumed to be smooth; propagation of solutions with weak or strong discontinuities are not treated here.

In direct simulation at moderate Reynolds numbers, numerical diffusion and dispersion induced by the discrete convection operator may effectively produce solutions that are different from the physical solutions. Numerical diffusion will lower the effective Reynolds number in the computed solution, while dispersion errors will cause incorrect propagation of waves with different wave numbers. An energy argument also motivates the use of an accurate spatial representation of the convection operator. For an ideal inviscid, unforced fluid with periodic boundary conditions, the incompressible Navier-Stokes equations conserves both linear momentum and kinetic energy. Similarly, a passive scalar convected by a solenoidal velocity field is conserved both linearly and quadratically. In calcula-

tions where accurate energy representations are crucial, the same conservation properties should hold also in the discrete case.

The arguments mentioned above are the primary motivation for using a high-order spatial discretization technique in the treatment of convection or convection-diffusion equations. In this section we use the spectral element method to construct a discrete high-order convection operator which has (spatially) no numerical diffusion, and has minimal dispersion errors. The discrete operator also conserves quadratic energy as does the continuous counterpart. In a low-order discretization scheme the dispersion errors are large except for very low wavenumbers of the discrete spectrum, however, in a high-order convection operator about 2/3 of the discrete spectrum is accurately represented. This means that in order to avoid dispersion errors, a low-order scheme will need much more resolution than required to resolve the flow, in contrast to our high-order operator where only a small increase in resolution is required. This extra resolution requirement coincides roughly with the additional resolution required if aliasing errors are to be avoided without filtering (2/3-rule, see Orszag [50]). This issue will be further addressed in Section 2.6 in the context of solving the Navier-Stokes equations.

2.5.1 Convection of a Passive Scalar

We consider here the unsteady convection of a passive scalar in an arbitrary domain Ω,

$$\frac{\partial \theta}{\partial t} + L\theta = 0 \qquad in\ \Omega, \tag{2.138}$$

where

$$L\theta = \nabla \cdot (\mathbf{v}\theta) \tag{2.139}$$

is the convection operator, and

$$\nabla \cdot \mathbf{v} = 0. \tag{2.140}$$

Here $\theta(\mathbf{r}, t)$ is a passive scalar (e.g. temperature), $\mathbf{v}(\mathbf{r})$ is a given steady solenoidal velocity field, and $\mathbf{r} \in \Omega$. Equations of the form (2.138) arise in the convection of temperature, mass, or species, and are important in the convection-diffusion equation,

$$\frac{\partial \theta}{\partial t} + L\theta = \alpha \nabla^2 \theta, \tag{2.141}$$

where α is the diffusivity. Equations of the form (2.138) and (2.141) also play a major role in the Euler equations and the incompressible Navier-Stokes equations.

The expression for the convection operator L given in (2.139) is called the conservative form; combining (2.139) and (2.140) gives the equivalent convective form,

$$L\theta = \mathbf{v}\nabla\theta. \tag{2.142}$$

A third way of writing L is as a linear combination of the convective and the conservative forms,

$$L\theta = \frac{1}{2}\mathbf{v}\nabla\theta + \frac{1}{2}\nabla \cdot (\mathbf{v}\theta), \tag{2.143}$$

which is called the skew-symmetric form. In the continuous case all the three forms are equivalent, however as a point of departure for a discrete formulation (2.143) is preferable in multi dimensions. The reason for this will become clear after discussing how the boundary conditions imposed on the domain boundary

$\partial\Omega$ effect the properties of the convection operator L. We start with multiplying (2.139) through by the test function ψ and integrate over the domain Ω,

$$(\psi, L\theta) = (\psi, \mathbf{v}\nabla\theta) \tag{2.144}$$

where the inner product $(\cdot,\cdot)$ is defined as

$$(\Phi, \Psi) = \int_\Omega \Phi\Psi d\Omega. \tag{2.145}$$

After integration by parts and applying the incompressibility constraint (2.140), we arrive at the following result,

$$(\psi, L\theta) = -(\theta,\mathbf{v}\nabla\psi) + \int_{\partial\Omega} \psi\theta\mathbf{v}\cdot\hat{n}dS. \tag{2.146}$$

or

$$(\psi, L\theta) = -(L\psi, \theta) + \int_{\partial\Omega} \psi\theta\mathbf{v}\cdot\hat{n}dS. \tag{2.147}$$

Here $\hat{n}$ is the outward unit normal on the domain boundary $\partial\Omega$. It follows from (2.147) that for the class of homogeneous boundary conditions on θ, ψ for which the surface integral in (2.147) vanishes, that the convection operator L is skew-symmetric. The corresponding set of homogeneous, linear, time-independent boundary conditions will be denoted as

$$\mathcal{B}_{ss}\theta = 0. \tag{2.148}$$

Examples of skew-symmetric boundary conditions include: the specification of θ on some segment of $\partial\Omega$; the imposition of periodicity on θ over some pair of segments on $\partial\Omega$; and no specification of boundary conditions on θ over a segment of $\partial\Omega$ for which $\mathbf{v}\cdot\hat{n} = 0$. We further assume that the imposed boundary conditions lead to a well-posed system. The skew-symmetric property implies

that L has imaginary eigenvalues, reflecting the fact that the convection operator is non-dissipative.

Another important class of boundary conditions of typical interest is "outflow" boundary conditions. In this case no specification of θ is imposed on a segment of $\partial\Omega$ for which $\mathbf{v} \cdot \hat{n} > 0$. The extended set of boundary conditions that includes outflow as well as $\mathcal{B}_{ss}$ will be denoted as

$$\mathcal{B}\theta = 0 \qquad on\ \partial\Omega. \tag{2.149}$$

In the case of outflow conditions the boundary integral in (2.147) does not vanish, and L is no longer skew-symmetric. It can be shown that the eigenvalues of L are now in the right-hand plane, $(\theta, L\theta) > 0$, reflecting the fact that energy is leaving the domain Ω.

We are now in a position to explain why it may be advantageous to use the particular form (2.143) of L instead of (2.139). In deriving the skew-symmetric property for L in the case of $\mathcal{B}_{ss}\theta = 0$, we use the fact that the velocity field is incompressible, however the proof of skew-symmetry for L defined in (2.143) does not rely on the fact that the velocity field be solenoidal; in any discrete scheme this is important for stability since the sense in which the velocity field is incompressible may not be consistent with integration by parts.

2.5.2 One-Dimensional Problems. Skew-Symmetric Boundary Conditions

To show the spectral element discretization procedure for the convection operator, we first consider the one-dimensional version of (2.138), namely

$$\theta_t + \theta_x = 0 \tag{2.150}$$

on the interval $\Lambda =]-1, 1[$ with periodic boundary conditions

$$\theta(-1) = \theta(1). \tag{2.151}$$

Here, subscript t and x denotes differentiation with respect to t and x, respectively. We note that the velocity is unity and that the boundary conditions (2.151) satisfy $\mathcal{B}_{ss}\theta = 0$, that is, we are considering the skew-symmetric case. In fact, periodic boundary conditions are the only skew-symmetric boundary conditions in the one-dimensional case. The continuous solution $\theta(x,t)$ belongs to the space $\mathcal{X}$, where

$$\mathcal{X} = \mathcal{H}^1(\Gamma \times \Lambda) \tag{2.152}$$

where $\Gamma =]0, \tau[$, and $0 \leq t \leq \tau$. The weak form of (2.150) can now be stated as: Find $\theta \in \mathcal{X}$ such that

$$(\psi, \theta_t) + (\psi, L\theta) = 0 \quad \forall \psi \in X, \quad \forall t, 0 \leq t \leq \tau. \tag{2.153}$$

where

$$X = \mathcal{H}^1_{\#}(\Lambda) \tag{2.154}$$

denotes the space of all functions which are square integrable over Λ, whose derivative are also square integrable over Λ, and which satisfy the periodic

boundary conditions (2.151). The space $\mathcal{X}$ ensures that we are only looking at problems with continuous solutions; as mentioned in the introduction to this section we are only considering problems which have (piecewise) smooth solutions. Another reason for looking at solutions which are in $\mathcal{H}^1$ is the treatment of convection-diffusion problems where this is a requirement.

In the one-dimensional case the velocity is constant ($\nabla \cdot \mathbf{v} = 0$) and all the forms (2.139), (2.142) and (2.143) are equivalent, also as a point of departure for a discrete formulation. Using the expression (2.139), and integrating by parts we arrive at

$$(\psi, L\theta) = (\psi, \theta_x) = -(\theta, \psi_x) + \theta\psi \mid_{-1}^{1} = -(\theta, \psi_x) = -(L\psi, \theta). \tag{2.155}$$

In (2.155) the boundary term vanishes due to the periodic boundary conditions (2.151), and the convection operator becomes skew-symmetric.

The basis for the spectral element discretization of (2.150-2.151) is a Galerkin procedure in which we search for a solution $\theta_h(t) \in \mathcal{H}^1(\Gamma; X_h)$ where X_h is a high-order polynomial subspace $X_h \subset X$. The discretization procedure starts with breaking up the domain into K subintervals (spectral elements),

$$\Lambda = \cup_{k=1}^{K} \Lambda_k \tag{2.156}$$

where Λ_k is defined by $a_k \leq x \leq a_k + b$ (for convenience we choose the elements to be of equal length b). The space of approximation for the semi-discrete solution $t \mapsto \theta_h(t, \cdot)$ is then taken to be the subspace $X_h \subset X$ consisting of all piecewise high-order polynomials of degree $\leq N$,

$$X_h = X \cap \mathcal{P}_{N,K} \tag{2.157}$$

where

$$\mathcal{P}_{N,K}(\Omega) = \{\Phi \in \mathcal{L}^2(\Omega); \Phi_{|\Omega_k} \in \mathcal{P}_N(\Omega_k)\} \tag{2.158}$$

and $\mathcal{P}_N(\Lambda_k)$ is the space of all polynomials of degree $\leq N$ on the interval Λ_k.

The spectral element discretization of (2.150) corresponds to numerical quadrature of the Galerkin form (2.153) restricted to the subspace X_h: Find $\theta_h(t) \in \mathcal{H}^1(\Gamma; X_h)$ such that

$$(\psi_h, \theta_{h,t})_{h,GL} + (\psi_h, L\theta_h)_{h,GL} = 0 \qquad \forall \psi_h \in X_h, \tag{2.159}$$

where $(\cdot,\cdot)_{h,GL}$ refers to Gauss-Lobatto quadrature as defined in (2.16). Due to the exactness of the quadratures in the one-dimensional case, it follows from (2.155) that the discrete convection operator can be written as

$$(\psi_h, L\theta_h)_{h,GL} = (\psi_h, \theta_{h,x})_{h,GL} = -(\theta_h, \psi_{h,x})_{h,GL}. \tag{2.160}$$

To implement (2.159) we choose an interpolant basis for X_h. As for elliptic equations the solution θ_h and the test functions ψ_h are expressed in terms of N^{th}-order Lagrangian interpolants through the Gauss-Lobatto Legendre points. The test functions are then chosen to be unity at one global (Gauss-Lobatto) collocation point, and the semi-discrete equations (2.159) can be stated in matrix form as

$$\underline{B}\,\underline{\theta}_t + \underline{C}\,\underline{\theta} = \underline{0}. \tag{2.161}$$

Here $\underline{\theta}$ is a vector of nodal unknowns, $\underline{B}$ is the diagonal mass matrix defined in (2.26), and $\underline{C}$ is the discrete convection operator

$$\underline{C} = \underline{D} = -\underline{D}^T, \tag{2.162}$$

where the expressions for $\underline{D}$ and $\underline{D}^T$ follow from (2.160),

$$\{\underline{D}\,\underline{\theta}\}_p = \sum_{k=1}^{K}{}' \sum_{q=0}^{N} \rho_p D_{pq} \theta_q^k \qquad \forall p \in \{0,...,N\}, \tag{2.163}$$

and T denotes the transpose operator. Here $\{\rho_p\}$ are the Gauss-Lobatto Legendre weights, and the elemental derivative matrix $\{D_{pq}\}$ is defined in (2.27).

If we instead of the convective form (2.139) use the skew-symmetric form (2.143) as a point of departure for our discrete formulation, the discrete convection operator becomes

$$\underline{C} = \frac{1}{2}[\underline{D} - \underline{D}^T], \tag{2.164}$$

which is a more consistent form to use in the multi-dimensional case. The reason why (2.139) and (2.143) are equivalent in the one-dimensional case is due the fact the velocity is constant, and all the quadratures are exact. From (2.162) or (2.164) we immediately see that the discrete convection operator associated with the skew-symmetric boundary conditions (2.151) is also skew-symmetric, that is,

$$\underline{C} = -\underline{C}^T. \tag{2.165}$$

Multiplying (2.161) by $\underline{\theta}^T$ and using the skew-symmetric property (2.165) we get

$$\underline{\theta}^T \underline{B}\,\underline{\theta}_t = -\underline{\theta}^T \underline{C}\,\underline{\theta} = \underline{\theta}^T \underline{C\theta} = 0 \tag{2.166}$$

or

$$\frac{\partial}{\partial t}(\frac{1}{2}\underline{\theta}^T \underline{B}\,\underline{\theta}) = \frac{\partial}{\partial t} \parallel \frac{1}{2}\underline{\theta}^T \underline{\theta} \parallel_{0,GL} = 0, \tag{2.167}$$

implying that

$$\parallel \frac{1}{2}\underline{\theta}^T \underline{\theta} \parallel_{0,GL} = \textit{constant}. \tag{2.168}$$

Here $\|\cdot\|_{0,GL}$ denotes the discrete $\mathcal{L}^2$-norm based on the Gauss-Lobatto Legendre points. Similarly, multiplying (2.161) with the unit vector $\underline{1}^T$ and using the exactness of the quadratures in the one-dimensional case together with the periodic (skew-symmetric) boundary conditions (2.151), we derive that

$$\frac{\partial}{\partial t}(\underline{1}^T \underline{B}\,\underline{\theta}) = 0, \tag{2.169}$$

or

$$\underline{1}^T \underline{B}\,\underline{\theta} = constant. \tag{2.170}$$

The results (2.168) and (2.170) mean that the passive scalar θ is conserved linearly and quadratically also in the discrete sense, that is, the convection operator $\underline{C}$ does not incur any numerical diffusion. Note that this result is based on the semi-discrete equations (2.161), which represent a set of coupled ordinary differential equations. If we include time-discretization as well, the result (2.168) still holds for time discretization schemes which are symmetric like the implicit Crank-Nicolson scheme. In general the results (2.168) and (2.170) hold for any time discretization scheme in the limit as the time step $\Delta t \Rightarrow 0$.

Due to the skew-symmetric property (2.165) all the eigenvalues for $\underline{C}$ will be pure imaginary. To demonstrate the accuracy of the discrete convection operator spectrum, we consider the one-dimensional eigenvalue problem: Find $(\theta, \lambda) \in X \times \mathcal{R}$ such that

$$L\theta = i\lambda\theta \qquad x \in \Lambda =]-1, 1[, \tag{2.171}$$

with periodic boundary conditions $\theta(-1) = \theta(1)$. The spectral element discretzation of (2.171) is: Find $(\theta_h, \lambda_h) \in X_h \times \mathcal{R}$ such that

$$(\psi_h, L\theta_h)_{h,GL} = i\lambda_h(\psi_h, \theta_h)_{h,GL} \qquad \forall \psi_h \in X_h, \tag{2.172}$$

or in matrix form,

$$\underline{C}\,\underline{\theta} = i\lambda_h \underline{B}\,\underline{\theta}. \tag{2.173}$$

The eigenvalues (λ, λ_h) will be real from skew-symmetry.

In Fig. 17 we plot the entire discrete spectrum of (2.173), $\lambda_k' = 2\lambda_k/\pi NK$, for $K = 10$ and $N = 9$. It is seen that a large part (about 2/3) of the spectrum is accurately represented by the spectral element approximation, similarly to earlier results derived for the pure spectral case (Weidemann and Trefethen [65]; Vandeven [63]). For comparison we have also included in Fig. 17 the corresponding h-type finite element result (with lumped mass matrix).

We now turn to discuss the issue of time discretization in more detail. First, we make some remarks regarding solution algorithms. In chapter 3 we shall see that all the iterative solution techniques we present are based on the assumption that the matrix to be inverted is symmetric. Symmetric matrices typically result from discretization of parabolic equations describing diffusion problems, or from *decoupled* saddle problems like the Stokes problem. For convection or convection-diffusion problems unsymmetric matrices result, and many iterative solution techniques fail, or become inefficient to use. In the context of solving the Navier-Stokes equations, the convective term makes the equations unsymmetric and nonlinear, which can significantly complicate the solution algorithm. One way to avoid these problems is to use a semi-implicit or a fully explicit scheme, in which at least the convective (unsymmetric) part of the equations is treated explicitly. This explicit treatment, however, imposes a restriction on the time step which results in high computational cost to reach steady state solutions, in particular for high Reynolds numbers.

The issue of numerical stability is always important in an explicit or semi-implicit time stepping procedure. Of interest is determining the restriction on the time step Δt in order to maintain (numerically) stable solutions, even in the limit as the time $t \Rightarrow \infty$. This time step restriction is closely connected to the spectrum of the discrete spatial operators, in this case the convection operator. The region of absolute stability (Gear [21]) is the region in which the product of the time step Δt and the maximum eigenvalue $\lambda_{max}(\underline{C})$ of the (explicitly treated) spatial operator gives numerically stable solutions. Again, note that $\lambda(\underline{C})$ refers to an eigenvalue of $\underline{C}$ with respect to the mass-matrix $\underline{B}$.

In the case of skew-symmetric boundary conditions all the eigenvalues of the convection operator $\underline{C}$ are pure imaginary, and any explicit time integration scheme must therefore have part of the imaginary axis included in its absolute stability region. In Fig. 18 we compare the absolute stability region for a few time integration schemes. We have chosen to use a third-order Adam Bashforth method, mainly because of the relatively large portion of the imaginary axis included in its stability region. This method is not symmetric in time (like Crank-Nicolson), which means that the fully discrete equations will no longer be skew-symmetric; numerical diffusion on the order of $O(\Delta t^3)$ will be incurred due to the time discretization procedure, however for small time steps this error is negligible.

In Fig. 18 we note that the absolute stability region for a third-order Adam-Bashforth time integration scheme crosses the imaginary axis at a value

$$\Delta t \cdot \lambda_{max}(\underline{C}) = 0.723, \tag{2.174}$$

where the maximum eigenvalue of the discrete convection operator scales like $\lambda_{max}(\underline{C}) \sim O(KN^2)$, see Fig 19. Since the minimum mesh spacing on the computational grid scales like $\Delta x_{min} \sim O(1/KN^2)$, it is expected that

$$\lambda_{max}(\underline{C}) \cdot \Delta x_{min} \Rightarrow \beta = constant \qquad as\ N \Rightarrow \infty. \tag{2.175}$$

In Fig. 20 we plot the product $\lambda_{max}(\underline{C}) \cdot \Delta x_{min}$ for the problem (2.173) as a function of the polynomial degree N for $K = 1$ and $K = 9$. In both cases the product approaches the limit

$$\beta = 1.18 \tag{2.176}$$

for large values of N. Note that this value of β corresponds to a velocity equal to unity. Combining (2.174) and (2.176) the time step restriction in the more general case becomes

$$\Delta t \leq 0.61\ min_p\ \{\frac{\Delta x}{U}\}_p, \tag{2.177}$$

where U is the velocity at the (local) collocation point p. From (2.177) we see that the third-order Adam-Bashforth method is stable if the (local) Courant number is less than 0.61 everywhere in the domain. (The criterion (2.177) is used in the actual implementation of the convection operator in multi dimensions.) In the context of solving the unsteady, incompressible Navier-Stokes equations, the result (2.177) tells us that as the Reynolds number increases and more resolution is required, the time step must decrease according to (2.177). In other words, as it becomes more important to avoid numerical diffusion and dispersion, this is automatically taken care of through the numerical stability restriction (2.177). If we are only interested in steady state solutions, however, the Courant criterion (2.177) is the price we have to pay for treating the convective term explicitly.

The fully discrete equations for the one-dimensional convection problem (2.150-2.151) can now be written as:

$$\underline{B}\{\frac{\underline{\theta}^{n+1}-\underline{\theta}^{n}}{\Delta t}\} = -\sum_{k=0}^{2}\alpha_k\{\underline{C}\,\underline{\theta}\}^{n-k}, \tag{2.178}$$

where superscript n denotes time step number n, and $\alpha_k, k = 0, 1, 2$ are the coefficients in the third-order Adam Bashforth scheme; $\alpha_0 = 23/12, \alpha_1 = -16/12, \alpha_2 = 5/12$. In the case of variable time step, these coefficients will depend on the time step. Finally, we remark that the particular form (2.178) of the discrete equations also persist in the multi-dimensional case.

2.5.3 One-Dimensional Problems. Outflow Boundary Conditions

An important class of problems is inflow/outflow problems where we don't have periodicity. In the case of pure convection, the order of the spatial operator is one, and we cannot impose Dirichlet boundary conditions both at inflow and outflow in order to define a well-posed problem. The solution can be described in terms of characteristics where the spatial direction is important. For this class of problem we specify initial conditions and impose Dirichlet boundary conditions at inflow, while no boundary conditions are imposed at outflow.

In the case of convection-diffusion problems, the second-order diffusion operator requires boundary conditions to be imposed both at inflow and outflow in order to formulate a well-posed problem. This is the case even for convection-dominated problems like in high Reynolds flow situations, or for heat transfer

problems involving high Peclet numbers. In many inflow/outflow situations we don't know what to specify as outflow conditions. This is often due to the fact that the outflow boundary is an artificial boundary which is created when we truncate an infinite domain in order to define a finite computational domain. An example of such a situation is flow past a cylinder where we don't know the solution in the wake behind the cylinder. For this class of problems we impose zero Neumann conditions at outflow. It can be shown that in the limit of high Reynolds numbers or Peclet numbers, normal (non-physical) boundary layers of thickness and magnitude $O(Re^{-1})$ or $O(Pe^{-1})$ will form at the (artificial) outflow boundary, while the rest of the computational domain will be unaffected of the outflow conditions. If we instead of Neumann conditions impose Dirichlet conditions at the outflow boundary, normal boundary layers of thickness $O(Re^{-1})$ or $O(Pe^{-1})$ will still form, however in this case we need to *resolve* this boundary layer in order to avoid wiggles and global breakdown of the discrete solution. Since the boundary layer can be very small, we need high resolution in a region of the domain where we are not interested in the solution anyway. Furthermore, since this is a normal boundary layer, the time step restriction due to the Courant condition (2.177) is severe. The reason we don't have to resolve the (artificial) boundary layer in the case of Neumann condition is because this is a much weaker (natural) form of boundary condition, in contrast to the Dirichlet type which is a strong (essential) form.

We shall now discuss how the properties of the convection operator $\underline{C}$ change in the case of outflow conditions. As for the skew-symmetric case we consider

the equation

$$\theta_t + \theta_x = 0 \tag{2.179}$$

on the inteval $\Lambda =]-1, 1[$, but now with Dirichlet boundary condition at inflow,

$$\theta(-1) = 0, \tag{2.180}$$

and specified (smooth) initial conditions. Note that in this case the space $X = \mathcal{H}^{1,0}(\Lambda)$ where the superscript 0 denotes the homogeneous essential (Dirichlet) boundary condition at inflow. The boundary integral in (2.147) does not vanish,

$$(\psi, L\theta) = (\psi, \theta_x) = -(\theta, \psi_x) + \theta(1)\psi(1) \neq -(L\psi, \theta), \tag{2.181}$$

and the convection operator is no longer skew-symmetric. The eigenvalues of L do no longer lie on the imaginary axis, but rather in the right- hand plane, consistent with the fact that energy is leaving the system. The spectral element discretization of (2.181) is given as

$$(\psi_h, L\theta_h)_{h,GL} = (\psi_h, \theta_{h,x})_{h,GL} = -(\theta_h, \psi_{h,x})_{h,GL} + \theta_h(1)\psi_h(1), \tag{2.182}$$

where we as usual express the testfunction ψ_h and the discrete solution θ_h in terms of N^{th} Lagrangian interpolants through the Gauss-Lobatto Legendre points. The expression for the convection operator then becomes (for simplicity, consider the case with only one element, $K = 1$),

$$\{\underline{C}\,\underline{\theta}\}_p = \sum_{q=0}^{N} \rho_p D_{pq} \theta_q = -\sum_{q=0}^{N} \rho_q D_{qp} \theta_q + \sum_{q=0}^{N} \delta_{pN}\delta_{qN}\theta_q \qquad \forall p \in \{0, ..., N\}. \tag{2.183}$$

We note that except for the single node defining the outflow boundary ($p = q = N$), the discrete convection operator $\underline{C}$ in (2.183) is still skew-symmetric,

$$C_{pq} = D_{pq} = -D_{qp} \qquad \forall p, q \in \{\{0, ..., N\}^2 \{\backslash \{p = q = N\}\}. \tag{2.184}$$

Here, $\mathcal{D}$ represents the spectral element discretization of the convective form (2.144) as defined in (2.163). The last matrix element of the convection operator $\underline{C}$ is obtained from (2.183) as follows:

$$C_{NN} = \mathcal{D}_{NN} = -\mathcal{D}_{NN} + 1 = \frac{1}{2} \tag{2.185}$$

Using the results (2.184) and (2.185) we can now easily derive how the linear and quadratic moments of θ change as a function of time,

$$\frac{\partial}{\partial t}(\underline{1}^T \underline{B}\,\underline{\theta}) = -\theta_N \tag{2.186}$$

$$\frac{\partial}{\partial t}(\frac{1}{2}\underline{\theta}^T \underline{B}\,\underline{\theta}) = \frac{\partial}{\partial t} \parallel \frac{1}{2}\underline{\theta}^T \underline{\theta} \parallel_{0,GL} = -\frac{1}{2}\theta_N^2. \tag{2.187}$$

These results are consistent with the amount of linear and quadratic moments leaving the domain at the outflow boundary.

The fact that $\underline{C}$ is skew-symmetric except for the (single) matrix element coupling at the outflow boundary motivates a decoupling of $\underline{C}$ into one skew-symmetric part, $\underline{C}_{ss}$, and one symmetric part, $\underline{C}_s$,

$$\underline{C} = \underline{C}_{ss} + \underline{C}_s. \tag{2.188}$$

The skew-symmetric part is associated with redistribution of energy within the domain, while the symmetric part is associated with energy leaving the domain at outflow. In this decoupling procedure it is convenient to introduce the diagonal Boolean mask matrices $(\underline{M}, \overline{\underline{M}})$ defined as (for simplicity, $K = 1$),

$$M_{pp} = 1 \qquad \forall p \in \{0, ..., N-1\} \tag{2.189}$$

$$M_{pp} = 0 \qquad p = N \tag{2.190}$$

$$\overline{M}_{pp} = 1 - M_{pp} \ . \tag{2.191}$$

The mask matrices makes it possible to split the collocation points into two disjoint groups, one ($\underline{M}$) which is associated with all collocation points *not* on the outflow boundary, and one ($\overline{\underline{M}}$) which is associated with points *on* the outflow boundary. In the one-dimensional case the skew-symmetric forms (2.162) and (2.164) are all equivalent and $\underline{C}_{ss}$ can be written as

$$\underline{C}_{ss} = \underline{M}^T[\frac{1}{2}\underline{D} - \frac{1}{2}\underline{D}^T]\underline{M} + \overline{\underline{M}}^T\underline{D}\,\underline{M} - \underline{M}^T\underline{D}^T\overline{\underline{M}}, \tag{2.192}$$

where we have used the more consistent form (2.164) in the first term. The symmetric part of $\underline{C}$ must be able to represent the kinetic energy $\theta^2/2$ leaving the outflow boundary. Choosing the testfunction $\psi_h = \theta_h$ in (2.182) we find that

$$\underline{C}_s = \overline{\underline{M}}^T[\frac{1}{2}\underline{D} + \frac{1}{2}\underline{D}^T]\overline{\underline{M}}, \tag{2.193}$$

consistent with the result (2.185). We note that this particular form for $\underline{C}$ is used also in the multi-dimensional case.

Finally, we make some remarks regarding inhomogeneous boundary conditions at inflow. In this case the solution belongs to the space $\theta \in \mathcal{H}^{1,E}(\Lambda)$, where superscript E denotes (non-zero) essential inflow boundary condition, while the space for the testfunction ψ, remains in $\psi \in \mathcal{H}^{1,0}(\Lambda)$. However, the energy arguments described for the homogeneous case are still valid, and the main conclusions remain the same as before.

2.5.4 Two-dimensional Convection Operator

We consider first the problem (2.138)-(2.140) in a rectilinear domain Ω with skew-symmetric boundary conditions (2.158). The weak form of (2.138) can be stated as: Find $\theta \in \mathcal{X}$ such that

$$(\psi, \theta_t) + (\psi, L\theta) = 0 \qquad \forall \psi \in X, \tag{2.194}$$

where

$$\mathcal{X} = \mathcal{H}^1(\Gamma \times \Omega), \tag{2.195}$$

$$X = \mathcal{H}^1\Omega), \tag{2.196}$$

$$\Gamma =]0, \tau[,\ 0 \leq t \leq \tau. \tag{2.197}$$

In (2.194) the convection operator can be expressed as

$$(\psi, L\theta) = (\psi, \mathbf{v}\nabla\theta) = -(\theta, \mathbf{v}\nabla\psi), \tag{2.198}$$

where the boundary term in (2.147) vanishes due to the skew-symmetric boundary conditions (2.148).

As usual the spectral element discretization proceeds by breaking up the domain into K disjoint macro- spectral elements,

$$\overline{\Omega} = \cup_{k=1}^{K} \overline{\Omega}_k, \tag{2.199}$$

where the intersection of two elements is either a whole edge or a vertex. The semi-discrete problem is then defined by performing tensor-product Gauss-Lobatto numerical quadrature of the Galerkin form (2.198) restricted to the subspace X_h: Find $\theta_h \in \mathcal{H}^1(\Gamma; X_h)$ such that

$$(\psi_h, \theta_{h,t})_{h,GL} + (\psi_h, L\theta_h)_{h,GL} = 0 \qquad \forall \psi_h \in X_h, \tag{2.200}$$

where

$$X_h = X \cap \mathcal{P}_{N,K}(\Omega), \tag{2.201}$$

$$\mathcal{P}_{N,K}(\Omega) = \{\Phi \in \mathcal{L}^2(\Omega); \Phi_{|\Omega_k} \in \mathcal{P}_N(\Omega_k)\}, \tag{2.202}$$

and $\mathcal{P}_N(\Omega_k)$ denotes the space of all polynomials of degree less than or equal to N in each spatial direction. The testfunction ψ_h and the solution θ_h are now expressed in terms of a tensor-product extension of the one-dimensional interpolant basis, and the discrete convection operator becomes

$$(\psi_h, L\theta_h)_{h,GL} = (\psi_h, \mathbf{v}\nabla\theta_h)_{h,GL}, \tag{2.203}$$

or in matrix form

$$\underline{C}' = \underline{D}, \tag{2.204}$$

where $\underline{D}$ represents the spectral element discretization of the convective form (2.144). The velocity field in (2.203) is appoximated using the same high-order interpolant basis as for the testfunction and the solution. The Gauss quadrature is no longer exact, and the convection operator $\underline{C}'$ as defined in (2.203-2.204) is no longer exactly skew-symmetric,

$$\underline{C}' = \underline{D} \simeq -\underline{D}^T = \underline{C}'. \tag{2.205}$$

However, if we use the form (2.143) for the convection operator we obtain

$$\underline{C} = \frac{1}{2}[\underline{D} - \underline{D}^T], \tag{2.206}$$

and skew-symmetry is now obtained for the discrete operator also in the multi-dimensional (isoparametric) case.

For outflow boundary conditions the discrete spectral element convection operator takes the same form as in the one-dimensional case (2.192-2.193). In

the multi-dimensional case we note that the symmetric part of $\underline{C}$ no longer has only one non-zero matrix element, but becomes a non-diagonal matrix which couples nodes on the outflow boundary. The particular form for $\underline{C}_s$ can again be motivated from the discrete version of (2.146) together with the fact that $\underline{C}_s$ must be able to represent the kinetic energy leaving the outflow boundary. It can be shown that if $\mathbf{v} \cdot \hat{t} = 0$ on the outflow boundary, the matrix $\underline{C}_s$ represents the kinetic energy $\underline{\theta}^T \overline{M}^T \underline{C}_s \overline{M} \underline{\theta}$ leaving the outflow boundary.

To illustrate the accuracy of the spectral element discretization of the convection operator, we consider convection of a two-dimensional Gaussian pulse across the rectilinear domain $\Omega = (x_1 \in]0,3[, x_2 \in]-2.5,2.5[)$. The exact solution as a function of space (x_1, x_2) and time t is given as

$$\theta(x_1, x_2, t) = e^{-2[(x_1-t)^2+x_2^2]}, \tag{2.207}$$

where the velocity $\mathbf{v} = (v_1, v_2) = (1,0)$. The exact solution is imposed as (time-dependent) boundary conditions at inflow ($x_1 = 0$), while outflow conditions are specified at the rest of the domain boundary. The initial condition at time ($t = 0$) corresponds to the situation where half the Gaussian pulse is inside the domain. At time $t = 3$ the pulse has been convected across the domain so that half the pulse remains inside the domain. At this moment we calculate the discrete $\mathcal{H}^1$-error $\| \theta - \theta_h \|_{1,GL}$. We use a small time step Δt in order for the temporal error to be negligible compared to the spatial error. Fig. 21 demonstrate that error $\| \theta - \theta_h \|_{1,GL}$ decreases exponentially fast as the polynomial degree N is increased for fixed $K = 9$. This is also expected due to the smooth nature of the solution. Note that in this test problem we actually solved the convection-diffusion equation (2.141) with a Peclet number equal to 10^7. The magnitude

and thickness of the normal boundary layer at outflow is therefore of order $O(10^{-7})$, which is negigible compared to the smallest discretization error in Fig. 21.

Finally, we remark that the two-dimensional results can easily be extended to solve three-dimensional convection or convection-diffusion problems in complex geometry by using tensor-product forms and isoparametric mappings.

2.6 Navier-Stokes Equations

We present here a combined semi-implicit/fully-explicit method for the solution of the three-dimensional, unsteady, incompressible Navier-Stokes equations based on the elliptic, convective, and Stokes schemes described in the previous sections. We consider the equations

$$-\mu\nabla^2\mathbf{u} + \rho\mathbf{u}_t + \nabla p = -\rho(\mathbf{u}\nabla)\mathbf{u} + \mathbf{f} \qquad in\ \Omega, \tag{2.208}$$

$$-\nabla\cdot\mathbf{u} = 0 \qquad in\ \Omega, \tag{2.209}$$

with rigid wall no-slip boundary conditions of the form

$$\mathbf{u} = 0 \qquad on\ \partial\Omega. \tag{2.210}$$

The presence of the advective term in (2.208) gives rise to many numerical difficulties not present in the steady or unsteady Stokes problem. First, the advective term render the equations unsymmetric and nonlinear, significantly complicating any solution algorithm. As discussed in Section 2.5 we treat the

advection term explicitly using a third-order Adam-Bashforth scheme. Second, the convective-diffusive balance can lead to large numerical errors and stringent resolution requirements in the limit of small viscosity. As demonstrated in Section 2.5 the spectral element convection operator (2.192-2.193) introduces minimal numerical diffusion and dispersion, and the high-order discretization procedure offers good resolution properties of boundary layers. Third, the nonlinearity is directly responsible for unsteadiness, transition, and turbulence, all of which give rise to a wide range of temporal and spatial scales that must be resolved (or modelled) if accurate numerical solutions are to be obtained. Again, the good resolution properties of a high-order (spatial) discretization scheme is attractive in the context of solving problems with *smooth* solutions.

Our algorithm for solving (2.208-2.210) is based on a combined semi-implicit/ fully-explicit discretization procedure. The main reason for this splitting is due to computational efficiency. In a fully explicit treatment the time-step restriction is closely connected to the spectrum of the discrete spatial operators. The maximum eigenvalue of the second-order viscous operator scales like $\lambda_{max}(\underline{A}) \sim O(N^4)$ where N is the polynomial degree in the high-order spectral element discretization, while the maximum eigenvalue of the first-order advection operator scales like $\lambda_{max}(\underline{C}) \sim O(N^2)$. Note that λ here refers to an eigenvalue with respect to the mass matrix $\underline{B}$ and not with respect to the identity matrix $\underline{I}$.

For low Reynolds number flow calculations the second-order viscous operator results in the most stringent time-step restriction in order to maintain numerical ***stability***, however this restriction is not necessary in order to obtain an *accurate*

solutions. For these problems we choose to use a semi-implicit algorithm based on implicit treatment of the Stokes problem and explicit treatment of the non-linear terms. As long as the Courant condition (2.177) is satisfied, the choice of time-step is based on accuracy requirements.

For high Reynolds number flow calculations the advective term results in the most stringent time-step restriction. As explained in Section 2.5 the (local) Courant number must be less than a fixed value (of order one) for the whole computational domain. Note that this time-step restriction is required *both* for accuracy and stability. For convection-dominated flow problems a fully explicit time-stepping procedure is therefore natural. A rough estimate for when the algorithm should switch from a semi-implicit to a fully explicit scheme can be obtained from looking at the scaling of the maximum eigenvalues of the discrete viscous and convective operators. This scaling argument leads to the conclusion that the switching should occur at a Reynolds number $Re \sim O(N^2)$; with the polynomial degree typically taken to of the order of 10, this gives a Reynolds number of the order of 100. In the actual implementation of the algorithm for solving the Navier-Stokes equations, this switching is done automatically and results in savings in terms of computational cost.

The discrete equations in the case of a semi-implicit treatment of (2.208-2.210) can be written as:

$$\mu \underline{A}\,\underline{u}_i^{n+1} + \sigma \underline{B}\,\underline{u}_i^{n+1} - \underline{D}_i^T \underline{p}^{n+1} = \sigma \underline{B}\,\underline{u}_i^n + \rho \sum_{q=0}^{2} \alpha_q \underline{C}^{n-q} \underline{u}_i^n + \underline{B}\,\underline{f}_i^n, \qquad (2.211)$$

$$-\underline{D}_i \underline{u}_i^{n+1} = \underline{0}, \qquad (2.212)$$

where $\sigma = \rho/\Delta t$, n denotes the time-step number, underscore refers to nodal

values (vectors and matrices), $i = 1,2,3$ denotes the spatial directions, and summation over repeated indicies is assumed. Here, a first-order Euler backward time discretization scheme is used for the unsteady Stokes equations.

The corresponding discrete equations in the case of fully explicit treatment of (2.208-2.210) is given as:

$$\sigma \underline{B}\,\underline{u}_i^{n+1} - \underline{D}_i^T \underline{p}^{n+1} = -\mu \underline{A}\,\underline{u}_i^n + \sigma \underline{B}\,\underline{u}_i^n + \rho \sum_{q=0}^{2} \alpha_q \underline{C}^{n-q} \underline{u}_i^n + \underline{B}\,\underline{f}_i^n, \qquad (2.213)$$

$$-\underline{D}_i \underline{u}_i^{n+1} = \underline{0}. \qquad (2.214)$$

In this case a first-order Euler forward time discretization scheme is used in the treatment of the unsteady Stokes equations.

We are now ready to demonstrate the accuracy of the spectral element discretization (2.211-2.214). As a first test problem we consider the following exact two-dimensional solution to the time-dependent, incompressible Navier-Stokes equations (Kim and Moin [31]):

$$\mathbf{u} = (u_1, u_2) = (-cosx_1 \cdot sinx_2 \cdot e^{-2t}, sinx_1 \cdot cosx_2 \cdot e^{-2t}) \qquad (2.215)$$

$$p = -\frac{1}{4}(cos2x_1 + cos2x_2) \cdot e^{-4t}. \qquad (2.216)$$

The problem is solved on a domain $\Omega =]-1,1[^2$, imposing time-dependent velocity boundary conditions according to (2.215), and with initial conditions corresponding to the exact velocity (2.215) at time $t = 0$. The domain is broken up into $K = 4$ similar spectral elements, each element being of order N. A typical solution is shown in Fig. 22 together with the corresponding spectral element discretization. First, we simulate until a fixed time $t = 1$ using different time steps Δt, while keeping N large enough such that the spatial errors are

negligible compared to the temporal errors. As can be seen from Fig. 23 the scheme is clearly first order in time. Next, we simulate until a fixed time $t = 0.5$ using different spatial resolutions ($N = 3, 4, 5$), but now keeping the time step small enough in order to insure that temporal errors are negligible. The results shown in Fig. 24 indicate that the spectral element method gives exponential convergence as the order of the elements, N, is increased.

The next test problem illustrates the effect of aliasing errors due to the nonlinear advection term. In the past various forms of expressing the nonlinear term have been used. Among the various alternatives are the convective form, $\mathbf{u}\cdot\nabla\mathbf{u}$, the conservative (or divergence) form, $\nabla\cdot(\mathbf{uu})$, the skew-symmetric form, $\frac{1}{2}\mathbf{u}\cdot\nabla\mathbf{u}+\frac{1}{2}\nabla\cdot(\mathbf{uu})$, and the rotational form, $\omega\times\mathbf{u}+\frac{1}{2}\nabla\mid\mathbf{u}\mid^2$, where $\omega=\nabla\times\mathbf{u}$ is the vorticity. The first three of these forms have already been discussed in Section 2.5.

For the past decade the rotation form has been a preferred choice for transition and turbulence, see Orszag [49], Canuto, Hussaini, Quarteroni, and Zang [14], mainly due to good conservation properties and economical implementation. The rotational form semi-conserves (global) kinetic energy for inviscid flow also in a discrete implementation, and requires evaluation of fewer derivatives. Recent work by Horiuti [30] and Zang [68], however indicates that the rotational form is less accurate than the skew-symmetric form. The test problem we present here supports this conclusion.

We consider a (simulated) bouyancy-driven flow in a two-dimensional square cavity. The domain together with the spectral element discretization is shown in

Fig. 25. We impose no-slip wall velocity boundary conditions, and specify zero initial conditions for the velocity at time $t = 0$. The external forcing function is $\mathbf{f} = (f_1, f_2) = (-10x_2, 0)$. The density of the fluid is taken to be unity, $\rho = 1$, while two different values for the viscosity is used, $\mu = 0.03$ and $\mu = 0.01$. We simulate until a fixed time $t = 1$ using $K = 9$ spectral elements, while varying the order of the elements, N. At this final time the Reynolds number is about 30 in the case of $\mu = 0.03$, and about 170 in the case of $\mu = 0.01$.

At time $t = 1$ we compute the jump in the spatial derivatives, $\frac{\partial u_i}{\partial x_j}, i = 1,2; j = 1,2$, at the four-element junction indicated in Fig. 25. Since the computed velocity field is in $\mathcal{H}^1$, only C^0-continuity is enforced across elemental boundaries, while continuity of the derivatives are accomplished as part of the convergence process ($N \Rightarrow \infty$). In Fig. 26 and Fig. 27 we plot the standard deviation of these jumps,

$$\varepsilon_d = \{\sum_{k=1}^{4}\sum_{i=1}^{2}\sum_{j=1}^{2}[\frac{\frac{\partial v_i^k}{\partial x_j} - \frac{\partial \bar{u}_i}{\partial x_j}}{\frac{\partial \bar{u}_i}{\partial x_j}}]^2\}^{\frac{1}{2}}, \tag{2.217}$$

as function of the polynomial degree, N, both for $Re \sim 30$ and $Re \sim 170$. For each of the two Reynolds numbers a comparison is made between the rotational form and the skew-symmetric form.

In both cases the skew-symmetric form gives the best results. At Reynolds number $Re \sim 30$ the difference between the two methods is not so large, except for $N \sim 6$ where the error for the skew-symmetric formulation makes a sudden drop; at this particular resolution the flow structure is essentially resolved. If we think of the flow structure in terms of spatial frequencies, the energy in this spectrum is essentially zero above a certain cut-off frequency. Until the spatial

resolution is high enough, aliasing errors due to the evaluation of the nonlinear term is incurred, together with dispersion errors for the highest wave numbers, see Section 2.5.

At Reynolds number $Re \sim 170$ the difference between the two formulations is more significant. A 1% error in the derivatives is obtained with $N = 8$ using the skew-symmetric formulation, while $N = 10$ is required for the rotational form. This translates into a 35% reduction in terms of total number of degrees-of-freedom. The results also clearly indicate that we need higher resolution to resolve the flow structure in this case compared to the lower Reynolds number case; smaller spatial scales have to be resolved and the cut-off frequency is higher.

Note that both the rotational form and the skew-symmetric form result in convergent approximation schemes, in the limit as $N \Rightarrow \infty$ the two formulations should give roughly the same error. What is more interesting is to compare the two formulations when the flow structure is marginally resolved. This particular numerical experiment supports the conclusion reported in Zang [68] that the rotation form produces aliasing errors that are more damaging than those produced by the skew-symmetric form.

Chapter 3

Spectral Element Iterative Solvers

In Chapter 2 we discussed the spectral element discretization of elliptic, parabolic and hyperbolic equations, and demonstrated the rapid rate at which the discrete solution converges to the exact (smooth) solution as the resolution (N) is increased. However, this result is useless unless we can construct methods by which we can *efficiently* invert the resulting systems of algebraic equations (2.211-2.214). Here, we define efficiency in terms of computational work required to achieve some fixed accuracy. Our conclusions in Section 2.1 regarding the choice of optimal discretization $h = (K, N)$ were based on the the use of optimal solvers (resolution independent convergence rate); the purpose of this chapter is to develop such solvers. In particular, we discuss the iterative solution techniques we apply in order to invert the algebraic equations resulting from spectral element discretization of elliptic and saddle Stokes problems. These solution techniques are then applied to solve the fully discretized Navier-Stokes equations (2.211-214). In Chapter 2 we demonstrated the good convergence properties afforded by the spectral element method; we now demonstrate that the resulting systems of discrete equations can also be efficiently inverted.

Earlier spectral element elliptic solvers were based on direct inversion techniques like static condensation (Korczak and Patera [33]) and fast Poisson solvers

(Patera [52]), which are efficient solution techniques for a certain class of problems. A direct method has, however several limitations and is much less flexible than an iterative solution method. First, the memory requirements for a direct method scales like $O(K_1^d N^{2d})$, compared to only $O((K_1^d N^d)$ for an iterative method. Here, K_1 is the number of elements in a typical spatial direction, N is the order (polynomial degree) of each spectral element, and d is the number of spatial dimensions. In order to solve large three-dimensional problems, it is virtually required to use iterative solution procedures due to the severe memory requirements associated with direct solvers. Second, even the operation count $O(KN^{d+1})$ associated with a matrix-vector product evaluation may favor an iterative approach rather than using a direct method. In particular, this will be true if an optimal iterative solver (resolution-independent convergence rate) is used to solve large three-dimensional problems. Third, the choice of inversion algorithm is most important in the context of solving time-dependent problems where the advantage of using a direct solver relies on the fact that the large initial work is amortized over many solves. This approach offers little flexibility in changing the time step dynamically according to stability or accuracy requirements, nor does it allow for solving problems where the geometry is changing as a function of time like free surface problems or moving boundary problems. As adaptive mesh techniques become more important this will also favor iterative solvers. Fourth, in a parallel architecture environment iterative techniques are attractive as they generally require much less communication between processors than direct solvers.

The success of an iterative solution algorithm relies strongly on two important

issues: rapid convergence rate and efficient matrix-vector product evaluation. The convergence rate determines how many (or few) iterations are required to obtain a solution, while the work associated with one matrix-vector product evaluation essentially determines the work per iteration. The product of the number of iterations and the work per iteration gives the total work required to obtain the discrete solution. The computational complexity associated with matrix-vector product evaluation is particularly important in the context of high-order discretization procedures where *tensor-product factorization* (Orszag [51]) is virtually *required*, in fact, the tensor-product factorization is the key to the efficiency of spectral techniques, and we shall discuss this important issue in more detail in the next section.

Note that a high-order discretization scheme like the spectral element method typically results in large-range couplings and therefore larger band-widths than a low-order method, however the structure of the set of algebraic equations is closely related to our particular choice of basis to represent the high-order polynomial subspace $P_{N,K}$. The interpolant Gauss-Lobatto Legendre basis in (2.19) offers C^0-continuity with minimal inter-elemental couplings; combined with Gauss-Lobatto Legendre quadrature it also offers minimal intra-element couplings (the mass matrix is diagonal). In essence, the particular choice of basis gives an optimal structure for the algebraic system of equations.

In this chapter we consider two iterative solution techniques: preconditioned gradient iteration (Golub and van Loan [25]) and spectral element multigrid (Rønquist and Patera [57], Maday and Muñoz [39]). Both of these techniques rely on the assumtion that the system of equations be linear symmetric positive-

definite. This is indeed satisfied for the discrete Laplacian $\underline{A}$ in (2.43) or the discrete Helmholtz operator $\underline{H}$ in (2.79); both operators result from spectral element discretization of linear self-adjoint elliptic problems. The Stokes problem (2.105-2.106) constitute a saddle problem, but as we shall see in Section 3.4 we can use a Uzawa algorithm to decouple this saddle problem into two linear symmetric positive-definite forms which are efficiently inverted by applying the iterative solution techniques in a *nested* fashion. In the context of solving the full Navier-Stokes equations, the convective term is treated explicitly; the reason for this is partly because this procedure greatly simplifies the numerical treatment of the nonlinear term, and partly because the resulting set of equations can be solved using the efficient solution techniques for elliptic and saddle Stokes systems.

We now make some final remarks regarding incomplete iteration. A direct solver offers the discrete solution in a finite number of arithmetic operations, with all operations carried out to machine precision. For a given machine there is no choice as regards how accurate we should solve the algebraic system of equations; the error incurred in the discrete solution will be the machine accuracy multiplied by some constant which depends on the condition number of the system matrix. In an iterative solver, however, the choice of criterion for when to stop iterating is an important issue. One way to proceed is to iterate until the residual has been reduced by several orders of magnitude and then hope for the best. This is not very satisfactory, and can lead to unnecessary computational work. What we need is a relation between the error incurred in the discrete solution due to incomplete iteration and the residual, and then

use this relation to choose a stop criterion. This approach offers the discrete solution to a specified accuracy at the least amount of computational cost. The issue regarding incomplete iteration will be addressed in a not so future paper (Maday, Meiron, Patera and Rønquist [37]).

3.1 Tensor-Product Forms

We consider here a non-constant coefficient Laplace operator in three space dimensions,

$$-\nabla \cdot (p\nabla u) = f \qquad in\ \Omega \in \mathcal{R}^3 \tag{3.1}$$

$$u = 0 \qquad on\ \partial\Omega, \tag{3.2}$$

where $\mathbf{x} = (x_1, x_2, x_3)$. Assuming that Ω is broken up as $\overline{\Omega} = \cup_k \overline{\Omega}_k$, $\Omega_k =]a_k, a_k'[\times]b_k, b_k'[\times]c_k, c_k'[$, the spectral element discretization of (3.1-3.2) is given by

$$\underline{A}\,\underline{u} = \underline{B}\,\underline{f}, \tag{3.3}$$

where

$$\underline{A}\,\underline{u} = \sum_{k=1}^{K}{}' \sum_{l=0}^{N}\sum_{m=0}^{N}\sum_{n=0}^{N} A^k_{\alpha\beta\gamma lmn}\, u^k_{lmn} \qquad \forall \alpha,\beta,\gamma \in \{0,...,N\}^3. \tag{3.4}$$

The elemental sums in (3.4) can be written as (for simplicity we assume cubic elements with $a_k' - a_k = b_k' - b_k = c_k' - c_k = 2$)

$$\underline{A}^k\underline{u}^k = \sum_{l=0}^{N}\sum_{m=0}^{N}\sum_{n=0}^{N} A^k_{\alpha\beta\gamma lmn} u^k_{lmn}$$

$$= \sum_{q=0}^{N} p^k_{qmn} D_{q\alpha} D_{ql} \delta_{\beta m}\delta_{\gamma n}\rho_q\rho_m\rho_n u^k_{lmn}$$

$$+\sum_{q=0}^{N} p^k_{lqn}\delta_{\alpha l}D_{q\beta}D_{qm}\delta_{\gamma n}\rho_l\rho_q\rho_n u^k_{lmn}$$

$$+\sum_{q=0}^{N} p^k_{lmq}\delta_{\alpha l}\delta_{\beta m}D_{q\alpha}D_{ql}\rho_l\rho_m\rho_q u^k_{lmn}. \qquad (3.5)$$

The three terms on the right hand side of (3.5) represent $\partial(p\partial u/\partial x_i)/\partial x_i$, $i = 1,2,3$, respectively, and $p^k_{qrs} = p(\xi^1_{q,k}, \xi^2_{r,k}, \xi^3_{s,k})$. The discretization (3.4)-(3.5) is a direct extension of the two-dimensional schemes described in Section 2.2.

It is important to realize that the computational complexity and memory requirement in implementing a high-order method is essentially determined from the treatment of the matrix-vector products (3.4). A naive evaluation of (3.5) requires $O(N^6)$ operations per element, as we must sum over $l,m,n \in \{0,...,N\}^3$ for $\alpha,\beta,\gamma \in \{0,...,N\}^3$; this naive operation also requires $O(N^6)$ storage as the matrix $A^k_{\alpha\beta\gamma lmn}$ is, in general, full. This operation count and storage is clearly prohibitive, in particular when compared to a low-order (h-type) method for which the equivalent operation count and storage both scale as $O(N^3)$. However, the key point in the evaluation of the matrix-product (3.5) is to realize that each of the terms on the right-hand side can be factored as (we consider here only the first term)

$$[\sum_{q=0}^{N} D_{q\alpha}[p^k_{q\beta\gamma}\rho_q\rho_\beta\rho_\gamma\,[\sum_{l=0}^{N} D_{ql}u^k_{l\beta\gamma}]]] \qquad \alpha,\beta,\gamma \in \{0,...,N\}^3. \qquad (3.6)$$

The computational complexity in evaluating (3.5) is now reduced to an operation count of only $O(N^4)$, and the memory requirement is reduced to $O(N^3)$ since only the elemental vectors $\underline{p}^k$, $\underline{u}^k$, together with the one-dimensional derivative operators are needed. This tensor-product factorization is the key to the efficiency of spectral methods, as first recognized by Orszag [51].

In the general case the operation count and memory requirement scale as $O(KN^{d+1})$ and $O(KN^d)$ for a d-dimensional problem; for the same total number of degrees-of-freedom (KN^d) a high-order method requires N times as much work as a low-order finite element method ($N = 1$). The reason why high-order methods are competitive with low-order methods is the fact that for sufficiently smooth solutions and sufficiently stringent error requirements, a high-order method needs far less degrees-of-freedom than a low-order method in order to meet these requirements, see Section 2.1.

We now make several comments regarding the sum factorization technique. First, we realize that this factorization follows from the tensor-product basis defined in (2.42). Second, we note that to evaluate the matrix-vector products (3.4) the elemental matrices $\underline{A}^k$ are never formed, but rather "computed on the fly". Third, the operation count $O(KN^{d+1})$ in $\mathcal{R}^d$ does not require any prior numerical integration, that is, the numerical quadrature is part of the evaluation. Fourth, the operation count does not rely on the differential operator being separable, as we placed no restriction on $p(\mathbf{x})$. Fifth, the operation count persist also for the isoparametric case; as regards computational complexity, the geometric factors enter into the equations in a fashion similar to the non-constant coefficient term $p(\mathbf{x})$. This implies that the operation count holds even for time-dependent geometries.

3.2 Preconditioned Conjugate Gradient Iteration

Conjugate gradient iteration is a standard iterative inversion technique (Golub and van Loan [25]) which is easy to implement, and works for symmetric positive-definite matrices. The preconditioned conjugate gradient iteration for the system (3.3) takes the form:

$$\begin{aligned}
&\underline{u}_0; \quad \underline{r}_0 = \underline{B}\,\underline{f} - \underline{A}\,\underline{u}_0; \quad \underline{q}_0 = \underline{P}^{-1}\underline{r}_0; \quad \underline{p}_0 = \underline{q}_0; \\
&a_m = (\underline{q}_m, \underline{r}_m)/(\underline{p}_m, \underline{A}\,\underline{p}_m) \\
&\underline{u}_{m+1} = \underline{u}_m + a_m \underline{p}_m \\
&\underline{r}_{m+1} = \underline{r}_m - a_m \underline{A}\,\underline{p}_m \qquad (3.7) \\
&\underline{q}_{m+1} = \underline{P}^{-1}\,\underline{r}_{m+1} \\
&b_m = (\underline{q}_{m+1}, \underline{r}_{m+1})/(\underline{q}_m, \underline{r}_m) \\
&\underline{p}_{m+1} = \underline{r}_{m+1} + b_m \underline{p}_m
\end{aligned}$$

where m refers to iteration number, $\underline{r}_m$ is the residual, $\underline{p}_m$ is the search direction, $\underline{P}$ is the preconditioner, $\underline{q}_m$ is a vector associated with the preconditioning ($\underline{q}_m = \underline{p}_m$ in case of no preconditioning, that is, $\underline{P}$ is the identity matrix), and $(\cdot,\cdot)$ denotes the discrete scalar product. It is clear that the critical step as regards computational complexity is the matrix-vector product $\underline{A}\,\underline{p}$, which can be evaluated in $O(KN^{d+1})$ operations in $\mathcal{R}^d$ using the tensor-product methods described previously.

The number of iterations required to achieve convergence scales like $\sqrt{\kappa_A}$, where $\kappa_A = \lambda^A_{max}/\lambda^A_{min}$ is the condition number of $\underline{A}$, and λ^A_{max} and λ^A_{min} are the maximum and minimum eigenvalues of $\underline{A}$, respectively. Since the condition number increases with the resolution (K, N), it is important to find a good preconditioner as regards efficiency. A good preconditioner should be spectrally close to $\underline{A}$, but at the same time be easy to invert. Earlier work used a preconditioner based on the incomplete Cholesky-factorization of an h-type finite element approximation defined on the Gauss quadrature points (Maday and Patera [40], Fischer [18]), however this choice is not the most attractive in the context of parallel architectures (Fischer, Rønquist, Dewey, and Patera [17]). In the following we shall therefore simply use the diagonal of $\underline{A}$ as a preconditioner, that is, $\underline{P} = \underline{D} = diag(\underline{A})$. For the spectral element equations it can be proved that (Maday and Muñoz [39])

$$\kappa_{B^{-1}A} \sim O(K_1^2 N^4), \tag{3.8}$$

$$\kappa_A \sim O(K_1^2 N^3), \tag{3.9}$$

$$\kappa_{D^{-1}A} \sim O(K_1^2 N^2), \tag{3.10}$$

thus implying that the solution of (3.3) will require $O(K_1 N)$ iterations. Here K_1 is the number of elements in a typical direction. Note that the condition number of $\underline{D}^{-1}\underline{A}$ is a factor $O(N^2)$ less than for $\underline{B}^{-1}\underline{A}$, which is the matrix resulting from a spectral collocation discretization.

The conjugate gradient iteration is a well-documented iterative method, and we shall here not elaborate the discussion any further. For numerical results we refer to the next section where we use convergence histories to compare the

preconditioned conjugate gradient iteration with the spectral element multigrid technique.

3.3 Spectral Element Multigrid

Over the past decade multigrid methods have had tremendous impact on the computational efficiency of elliptic solvers (Brandt [10], Hackbush and Trottenberg [28], Brand, Lemke, and Linden [9]). The multigrid kernel of recursive high-wavenumber smoothing and coarse-grid correction results in an effectively resolution-independent convergence rate, which in turn translates into computational economies.

Multigrid techniques were initially applied primarily to finite difference techniques, but have more recently been cast in a geometrically flexible variational finite element context (Bank and Douglas [5]; Maday and Munoz [39]). However, despite the advantages of multigrid techniques for both finite-difference and finite-element discretizations, application of multigrid techniques to (non-periodic) spectral (Gottlieb and Orszag [27]), spectral element (Patera [53], Ronquist and Patera [56], Maday and Patera [40]), and p-type finite element methods (Babuska and Dorr [4]) has been rather limited. Zang, Wong and Hussaini [69,70] have applied the multigrid techniques to multi-dimensional Chebyshev approximations; however, the success of their technique is primarily due to the good preconditioning properties of h-type incomplete decompositions (Orszag [51], Deville and Mund [16]), and not to the multigrid process *per se*.

In this section we describe an extension of variational h-type multigrid to an intra-element Legendre spectral element multigrid procedure. We demonstrate that essentially resolution- independent convergence rates can be achieved with only diagonal (Jacobi) smoothers, which are highly parallelizable and considerably simpler than previous incomplete-decomposition spectral multigrid procedures.

3.3.1 Nested Spaces

We consider here the three-dimensional Poisson equation in a domain Ω with homogeneous Dirichlet boundary conditions on the domain boundary $\partial\Omega$,

$$-\nabla^2 u = f \qquad in\ \Omega \tag{3.11}$$

$$u = 0 \qquad on\ \partial\Omega. \tag{3.12}$$

The equivalent variational form is: Find $u \in X$ such that

$$a(u,v) = (f,v) \qquad \forall v \in X, \tag{3.13}$$

where

$$X = \mathcal{H}_0^1(\Omega), \tag{3.14}$$

and

$$a(u,v) = \int_\Omega \nabla u \nabla v \, d\Omega, \tag{3.15}$$

$$(f,v) = \int_\Omega f v \, d\Omega. \tag{3.16}$$

The standard spectral element discretization procedure starts by breaking up the domain Ω into K elements,

$$\overline{\Omega} = \cup_{k=1}^{K} \overline{\Omega}_k. \tag{3.17}$$

The discrete equations are then generated by applying tensor-product Gauss-Lobatto Legendre numerical quadrature of the variational form (3.13) restricted to the subspace X_h, and the discrete problem becomes: Find $u_h \in X_h$ such that

$$a_{h,GL}(u_h, v_h) = (f, v_h)_{h,GL} \qquad \forall v_h \in X_h, \tag{3.18}$$

where the finite-dimensional subspace $X_h \subset X$ is defined as

$$X_h = X \cap Y_h, \tag{3.19}$$

$$Y_h = \mathcal{P}_{N,K}(\Omega) = \{\Phi \in \mathcal{L}^2(\Omega); \Phi_{|\Omega_k} \in \mathcal{P}(\Omega_k)\} \tag{3.20}$$

and $\mathcal{P}_N(\Omega_k)$ is the space of all polynomials of degree less than or equal to N with respect to each spatial variable x_1, x_2, and x_3.

We are here interested in solving the problem (3.11-3.12) based on the spectral element discretization corresponding to a given discretization pair $h = (K, N)$. However, to compute the discrete solution $u_h \in X_h$ we shall apply an *intra-element* multigrid concept where the spectral element discretization of (3.11-3.12) is associated with a set of nested spaces X_{h_j} with discretization pairs $h_j = (K, N_j)$. The nested spectral element spaces $X_{h_1} \subset X_{h_2} \subset ... \subset X_{h_J}$ are defined as

$$X_{h_j} = X \cap Y_{h_j} \qquad \forall j \in \{1, ..., J\} \tag{3.21}$$

$$Y_{h_j} = \mathcal{P}_{N_j,K}(\Omega) \qquad \forall j \in \{1, ..., J\}, \tag{3.22}$$

with $j = 1$ the coarsest mesh and $j = J$ the finest mesh. We will typically choose $N_{j+1} \cong 2N_j$. The objective is to compute the solution u_h on the finest mesh, that is, $N = N_J$. Note that the nested spaces correspond to intra-element multigrid in the sense that the subdomains Ω_k remain invariant. This intra-element multigrid concept preserves the natural hierarchy of the spectral element

method; restriction operators and prolongation operators and associated data structures are easily and efficiently defined within the structured Ω_k; the tight coupling within the elements allows the technique to be readily parallelized within the context of a medium-grained paradigm (McBryan and Van de Velde [44], Fischer, Rønquist, Dewey, and Patera [17]); the coarse ($j = 1$) grid is naturally provided as a low-order (e.g., $N = 1$) finite element mesh defined on the macro-elemental "skeleton."

We consider now the implementation of the spectral element discretizations associated with the nested spaces X_{h_j}. To represent a function $w_{h_j} \in Y_{h_j}$ we use a tensor-product interpolant basis through the Gauss-Lobatto Legendre points

$$w_{h_j}(r_1, r_2, r_3) = \cup_{k=1}^{K} \sum_{l=0}^{N_j} \sum_{p=0}^{N_j} \sum_{q=0}^{N_j} w_{lpq}^{k} \, h_l^{N_j}(r_1) h_p^{N_j}(r_2) h_q^{N_j}(r_3), \tag{3.23}$$

where $\mathbf{x} = (x_1, x_2, x_3) \in \Omega_k \Rightarrow \mathbf{r} = (r_1, r_2, r_3) \in \Lambda \times \Lambda \times \Lambda$, and $\Lambda =]-1, 1[$. The interpolant basis (3.21) is also used to represent a function $w_{h_j} \in X_{h_j}$, but in addition we must also require the function to be continuous across elemental boundaries and satisfy the homogeneous boundary conditions. Expressing the discrete solution $u_{h_j} \in X_{h_j}$, the testfunctions $v_{h_j} \in X_{h_j}$, and the data $f \in Y_{h_j}$ in terms of the basis (3.21) and choosing v_{h_j} to be nonzero at only one global collocation point, the discrete formulation (3.16) becomes

$$\sum_{k=1}^{K}{}' \sum_{l=0}^{N_j} \sum_{p=0}^{N_j} \sum_{q=0}^{N_j} \tilde{A}_{\alpha\beta\gamma lpq}^{j,k} u_{lpq}^{j,k} = \sum_{k=1}^{K}{}' \sum_{l=0}^{N_j} \sum_{p=0}^{N_j} \sum_{q=0}^{N_j} \tilde{B}_{\alpha\beta\gamma lpq}^{j,k} f_{lpq}^{j,k}, \qquad \forall \alpha\beta\gamma \in \{0, ..., N_j\}^3 \tag{3.24}$$

$$\tilde{A}_{\alpha\beta\gamma lpq}^{j,k} = A_{\alpha l}^{j,k} B_{\beta p}^{j,k} B_{\gamma q}^{j,k} + B_{\alpha l}^{j,k} A_{\beta p}^{j,k} B_{\gamma q}^{j,k} + B_{\alpha l}^{j,k} B_{\beta p}^{j,k} A_{\gamma q}^{j,k}, \tag{3.25}$$

$$\tilde{B}_{\alpha\beta\gamma lpq}^{j,k} = B_{\alpha l}^{j,k} B_{\beta p}^{j,k} B_{\gamma q}^{j,k}, \tag{3.26}$$

where the one-dimensional operators $A_{pq}^{j,k}$, $B_{pq}^{j,k}$ are defined in (2.25) and (2.26), respectively. The only difference in notation is the extra superscript j which is introduced in order to associate the correct space X_{h_j} corresponding to the discretization pair $h_j = (N_j, K)$. The sets of discrete equations can also be written in matrix form as

$$\underline{\underline{A}}^j \underline{u}^j = \underline{\underline{B}}^j \underline{f}^j \qquad \forall j \in \{1, ..., J\}. \tag{3.27}$$

Next, we need a smoothing operator that is effective in eliminating the high wavenumbers on a given mesh. To this end, we use a standard diagonal Jacobi smoother,

$$\underline{\underline{D}}^j = \lambda_{max}^j \, diag(\underline{\underline{A}}^j), \tag{3.28}$$

where

$$\lambda_{max}^j = max\{\frac{\underline{\chi}^T \underline{\underline{A}}^j \underline{\chi}}{\underline{\chi}^T diag(\underline{\underline{A}}^j) \underline{\chi}}\} \qquad \forall \underline{\chi} \in X_{h_j}. \tag{3.29}$$

The eigenvalue λ_{max}^j is used to normalize the spectrum of the smoother so that all the eigenvalues are less than one in absolute value, which is an obvious requirement for convergence. Note that the choice of (3.28) rather than incomplete Cholesky used in previous spectral multigrid investigations (Zang, Wong, and Hussaini [69,70], Phillips [54]) has many advantages: (3.28) is very easily formed and inverted, even in complex geometry; (3.28) is completely parallelizable, unlike the incomplete decomposition. Also note that the maximum eigenvalue in (3.29) can be efficiently evaluated by a power iteration method; this work is easily amortized over future iterations, in particular for time-dependent problems or nested-elliptic Stokes solvers (Maday and Patera [40]; Rønquist and Patera [55]).

We then need to define a restriction operator that takes a residual on $X_{h_{j+1}}$ and projects it onto X_{h_j}. To effect this projection we follow the usual procedure of representing the lower-order basis in terms of the higher-order basis, which gives the elemental multi-dimensional restriction operator

$$\tilde{R}^{j,k}_{\alpha\beta\gamma lpq} = R^{j,k}_{\alpha l} R^{j,k}_{\beta p} R^{j,k}_{\gamma q}$$

$$\forall j \in \{1, ..., J-1\}, \qquad \forall k \in \{1, ..., K\}$$

$$\forall \alpha\beta\gamma \in \{0, ..., N_j\}^3, \qquad \forall lpq \in \{0, ..., N_{j+1}\}^3, \tag{3.30}$$

where

$$R^{j,k}_{\alpha l} = h^{N_j}_{\alpha}(\xi^{N_{j+1}}_{l}). \tag{3.31}$$

Note that the multi-dimensional restriction operator (3.30) is given in terms of the one-dimensional operators (3.31) in tensor-product form. Given the *elemental* residuals $\hat{r}^{j+1,k}_{\alpha\beta\gamma} \in Y_{h_{j+1}}$,

$$\hat{r}^{j+1,k}_{\alpha\beta\gamma} = w^{j+1,k}_{\alpha\beta\gamma} - \sum_{l=0}^{N_{j+1}} \sum_{p=0}^{N_{j+1}} \sum_{q=0}^{N_{j+1}} \tilde{A}^{j+1,k}_{\alpha\beta\gamma lpq} z^{j+1,k}_{lpq}$$

$$\forall k \in \{1, ..., K\} \qquad , \forall \alpha\beta\gamma \in \{0, ..., N_{j+1}\}^3, \tag{3.32}$$

for some data $w^{j+1,k}_{\alpha\beta\gamma}$ and iterate $z^{j+1,k}_{lpq}$, the restricted *global* residual (which will enter as the inhomogeneity at the next multigrid level) is then calculated using the sum-factorization technique described in Section 3.1,

$$\underline{r}^{j} = \sum_{k=1}^{K}{}' \sum_{l=0}^{N_{j+1}} \sum_{p=0}^{N_{j+1}} \sum_{q=0}^{N_{j+1}} \tilde{R}^{j,k}_{\alpha\beta\gamma lpq} \hat{r}^{j+1,k}_{lpq}$$

$$= \sum_{k=1}^{K}{}' \{ \sum_{l=0}^{N_{j+1}} R^{j,k}_{\alpha l} [\sum_{p=0}^{N_{j+1}} R^{j,k}_{\beta p} (\sum_{q=0}^{N_{j+1}} R^{j,k}_{\gamma q} \hat{r}^{j+1,k}_{lpq})] \}$$

$$\forall \alpha\beta\gamma \in \{0, ..., N_j\}^3, \tag{3.33}$$

which can be expressed in global form as

$$\underline{r}^j = \underline{\underline{R}}^{j+1}\underline{r}^{j+1} \tag{3.34}$$

In Rønquist and Patera [57] it is shown that if $\underline{r}^{j+1} = \sum_k \hat{r}_{lpq}^{j+1,k} = \underline{0}$, then the restriction defined by (3.33) will also vanish.

Lastly, we require a prolongation operator that, given a $\underline{z}^j \in X_{h_j}$, extends the function $\underline{z}^j$ onto $X_{h_{j+1}}$. From the definition of the restriction operator in (3.30-3.31) we readily see that the correct choice for prolongation is

$$z_{lpq}^{j+1,k} = \sum_{\alpha=0}^{N_j}\sum_{\beta=0}^{N_j}\sum_{\gamma=0}^{N_j} R_{lpq\alpha\beta\gamma}^{j,k}\, z_{\alpha\beta\gamma}^{j,k}$$

$$\forall k \in \{1,...,K\}, \qquad \forall lpq \in \{0,...,N_{j+1}\}^3, \tag{3.35}$$

which is again defined on an elemental basis. The global statement of (3.35) is

$$\underline{z}^{j+1} = (\underline{\underline{R}}^j)^T \underline{z}^j, \tag{3.36}$$

where it is seen that, as is usually the case in variational multigrid, the prolongation operator is the transpose of the restriction opertor.

3.3.2 Multigrid V-cycle

We now consider a simple V-cycle for solution of

$$\underline{\underline{A}}^J \underline{u}^J = \underline{g}^J \tag{3.37}$$

which corresponds to the spectral element discretization (3.18) of the problem (3.11-3.12) on the finest mesh $j = J$. We first set the V-cycle iteration counter

$l = 0$, and choose an initial guess $\underline{u}^{J,(0)}$. At the highest level, we then perform

$$* \quad MG(J, \underline{u}^{J,(l+1)}, \underline{g}^J, m)$$

$$\underline{u}^{J,(l+1)} = \underline{u}^{J,(l)}$$

$$l = l + 1; \qquad go\ to\ *$$

Here $MG(j, \underline{z}, \underline{w}, m)$ is our multigrid algorithm (a simplified form of the procedure in Bank and Douglas [5]), and $2m$ is the number of smoothings on level j per V-cycle.

The MG algorithm is readily defined in terms of the operators of the previous section. Although in practice all operations are performed *elementally*, for convenience we describe the algorithm in terms of global operators. Any operator or vectors for which the multigrid level is not indicated may be assumed to be at level j.

Algorithm $MG(j, \underline{z}, \underline{w}, m)$:

If $j = 1$ *solve* $\underline{A}\,\underline{z} = \underline{w}$

If $j \neq 1$

(*a*) *Form residual* : $\underline{r} = \underline{w} - \underline{A}\,\underline{z}$

(*b*) *Smooth* : $\underline{z}^{(0)} = \underline{z}, \underline{r}^{(0)} = \underline{r}$

$$\underline{D}\,\underline{z}^{(n+1)} = \underline{D}\,\underline{z}^{(n)} + \underline{r}^{(n)} \qquad n = 0, ..., m-1$$

$$\underline{r}^{(n+1)} = \underline{w} - \underline{A}\,\underline{z}^{(n+1)} \qquad n = 0, ..., m-1$$

$$\underline{z} = \underline{z}^{(m)}, \qquad \underline{r} = \underline{r}^{(m)}$$

(c) *Restrict residual* : $\underline{w}^{j-1} = \underline{R}^{j-1}\underline{r}^{j}$

(d) $\underline{e}^{j-1} = \underline{0}; MG(j-1, \underline{e}^{j-1}, \underline{w}^{j-1}, m)$

(e) *Prolong correction* : $\underline{z} = \underline{z} + (\underline{R}^{j-1})^{T}\underline{e}^{j-1}$

(f) *Form residual* : $\underline{r} = \underline{w} - \underline{A}\,\underline{z}$

(g) *Smooth* : $\underline{z}^{(0)} = \underline{z}, \underline{r}^{(0)} = \underline{r}$

$$\underline{D}\,\underline{z}^{(n+1)} = \underline{D}\,\underline{z}^{(n)} + \underline{r}^{(n)} \qquad n = 0, ..., m-1$$

$$\underline{r}^{(n+1)} = \underline{w} - \underline{A}\,\underline{z}^{(n+1)} \qquad n = 0, ..., m-2$$

$$\underline{z} = \underline{z}^{(m)}$$

(h) *End.*

In order to implement the recursive MG-algorithm in FORTRAN, a special program (also written in FORTRAN) has been been developed in order to produce the multigrid source code. This concept of using a "pre-compiler" to generate the actual source code might be useful in the more general context of implementing complex numerical algorithms.

3.3.3 Convergence Rate

In the previous section we discussed how the condition number of the system matrix $\underline{A}$ effects the convergence rate of the conjugate gradient iteration. The key contribution of the multigrid algorithm is that it successfully addresses the conditioning problem, which is particularly important for high-order methods.

The way this is achieved is essentially by a careful combination of smoothing iterations and coarse grid corrections.

To predict the multigrid convergence rate in the case of spectral element discretizations we start by looking at the (Jacobi) smoothing steps (b) or (g) in the *MG*-algorithm. The error $\underline{e}$ in the solution $\underline{z}$ on grid j can be written as (any operator or vector for which the multigrid level is not indicated may assumed to be at level j)

$$\underline{e}^{n+1} = [\underline{I} - \underline{D}^{-1}\underline{A}]\underline{e}^n \qquad n = 0, ..., m-1. \tag{3.38}$$

Using an eigenfunction expansion for the error,

$$\underline{e} = \sum_i c_i \psi_i \tag{3.39}$$

where

$$\underline{A}\,\underline{\psi}_i = \lambda_i diag(\underline{A})\underline{\psi}_i, \tag{3.40}$$

we get the following reduction in error component e_i with wavenumber λ_i

$$e_i^{n+1} = (1 - \mu_i)e_i^n \tag{3.41}$$

where μ is defined as

$$\mu = \frac{\lambda}{\lambda_{max}} \tag{3.42}$$

Using the result (3.10) from the previous section we get

$$0 < \mu_{min} \leq \mu \leq 1 \tag{3.43}$$

where

$$\mu_{min} = \frac{1}{\kappa_{D^{-1}A}} \sim O(\frac{1}{K_1^2 N^2}). \tag{3.44}$$

From (3.39-3.42) we realize that the high-wavenumber error components, that is, error components corresponding to normalized eigenvalues close to one are damped quickly, while low-wavenumber components are damped at a rate $\rho = 1 - O(N^{-2})$ for large n and N.

In the multigrid algorithm only a few smoothing iterations are performed per V-cycle. Assume that the solution after m smoothing steps is $\underline{z}^j \in X_{h_j}$ with an error $\underline{e}^j \in X_{h_j}$. In steps (c), (d) and (e) a coarse grid correction to the solution is computed. In the one-dimensional case it can be shown (Bank and Douglas [5]) that if we solve exactly on the coarse grid, the coarse grid correction is an orthogonal projection of the error from X_{h_j} to $X_{h_{j-1}}$ with respect to the $a_{h_j}(\cdot,\cdot)$ inner product. After the correction the error $\underline{e}^j$ will then belong to the space $\underline{e}^j \in X_{h_{j-1}}^{\perp} \cap X_{h_j}$, which is associated with the high-wavenumber error components. This remaining error is reduced efficiently by performing more smoothing steps, however, in the multi-dimensional case, the smoothing operation introduces new low-wavenumber errors into the solution and thus more V-cycles are needed. It is this interaction between smoothing and coarse grid correction which makes the multigrid algorithm so effective. Instead of the slow convergence rate associated with the plain Jacobi smoothing (3.38), we obtain

$$e_i^{n+1} = (1 - \mu_i) e_i^n \tag{3.45}$$

where

$$\mu_{mid}^j \leq \mu_i \leq 1 \tag{3.46}$$

and

$$\mu_{mid}^j = \frac{\lambda_{mid}^j}{\lambda_{max}^j} = \frac{\lambda_{max}^{j-1}}{\lambda_{max}^j}. \tag{3.47}$$

In the one-dimensional case it can be proved (Maday and Muñoz [39]) that $\lambda^j_{max} \sim N_j^2$; assuming that $N_j/N_{j-1} = 2$ and substituting into (3.47) gives

$$\mu^j_{mid} \sim \frac{(N_j/2)^2}{(N_j)^2} = \frac{1}{4}. \tag{3.48}$$

Hence, we see that the effect of the coarse grid correction is to effectively lower the condition number on mesh j from $O(K_1^2 N_j^2)$ to 4, independent of the mesh j. If the number of smoothings $m \leq 3$ it can be proved (Maday and Muñoz [39]) that a multigrid convergence rate $\rho = 0.75$ results, *independent* of the resolution $h_J = (K, N_J)$.

In the two-dimensional case it can be shown (Maday and Muñoz [38]) that μ_{mid} is not independent of N like in the one-dimensional case, but scales like

$$\mu_{mid} \sim \frac{C}{N} \tag{3.49}$$

where C is a constant. The multigrid convergence rate will now be dependent of the polynomial degree N,

$$\rho \sim 1 - \frac{C}{N}, \tag{3.50}$$

however it is *independent* of the number of elements K. Note that if intra-subdomain multigrid is used in the context of interface relaxation for the domain decomposition method (Funaro, Quarteroni, and Zanolli [20]) this independence of K will no longer be true.

The convergence rate (3.50) can be improved by using Chebyshev acceleration (Ames [1]). We multiply the smoother in (3.28) with a constant τ which can be different for each smoothing iteration, and rewrite (3.38) and (3.41) as

$$\underline{e}^{n+1} = [\underline{I} - (\tau^{n+1})^{-1} \underline{D}^{-1} \underline{A}]\underline{e}^n \qquad n = 0, ..., m-1 \tag{3.51}$$

and

$$e_i^{n+1} = (1 - (\tau^{n+1})^{-1}\mu_i)e_i^n \qquad n = 0, ..., m-1, \tag{3.52}$$

implying that

$$e_i^m = [\prod_{n=1}^{m}(1 - (\tau^n)^{-1}\mu_i]\, e_i^0 = Q_m(\mu_i)e_i^0. \tag{3.53}$$

where $Q_m(\mu)$ is a polynomial of degree m. We then formulate the following optimization problem: Find coefficients $\tau^n, n = 1, ..., m$ such that

$$max_\mu\, Q_m(\mu) = min \qquad for \qquad \mu_{mid} \leq \mu \leq 1,$$

$$and \qquad Q_m \leq 1 \qquad \forall \mu \in [0, \mu_{mid}], \tag{3.54}$$

subject to the consistency constraint that $Q_m(0) = 1$. Note that we minimize $Q_m(\mu)$ over a range of μ associated with error components belonging to the space $X_{h_{j-1}}^{\perp} \cap X_{h_j}$ rather than minimizing over the whole spectrum of error components. As the low-wavenumber components are efficiently removed by the coarse-grid correction operation, the solution to the minimization problem (3.54) should give a better multigrid convergence rate than minimizing Q_m over the entire spectrum. This acceleration procedure is similar, but a little more intuitive than the one discussed in Bank and Douglas [5].

The solution to the problem (3.54) can be extended from Ames [1] as the polynomial

$$Q_m(\mu) = \frac{T_m(-1 + \frac{2(1-\mu)}{1-\mu_{mid}})}{T_m(-1 + \frac{2}{1-\mu_{mid}})} \tag{3.55}$$

where $T_m(x)$ is the Chebyshev polynomial of degree m. It is seen that although we are minimizing $Q_m(\mu)$ only in the range $\mu_{mid} \leq \mu \leq 1$, $Q_m(\mu) \leq 1$ also for values of μ in the range $0 < \mu < \mu_{mid}$, thus assuring that the low-wavenumber error components are not amplified. Using the expression (3.49) for μ_{mid} we can

derive an average convergence rate for the m smoothing steps which is valid for large polynomial degrees N,

$$\overline{\rho} = 1 - 2(\frac{C}{N})^{1/2}. \tag{3.56}$$

Comparing (3.56) with (3.50) shows that the work to achieve a fixed accuracy will be reduced by a factor $\sqrt{N}$ when applying the Chebyshev acceleration.

3.3.4 Computational Complexity

We now present operation counts for the general (rectilinear) problem in $\mathcal{R}^d$. As explained in Section 3.1, the reason for choosing tensor product spaces, quadratures, and bases is to allow for rapid evaluation of the discrete Laplacian and restriction operators by sum factorization (Orszag [51]). We can easily show that to evaluate the discrete Laplacian in $\mathcal{R}^d$ on mesh j, that is, to evaluate the left hand side of (3.4), requires

$$W_A^j = K(2d)N_j^{d+1} \tag{3.57}$$

operations. (For simplicity we include only multiplications in our operation count.) Note that we have included in (3.57) only the leading-order terms; a direct stiffness summation requires only $O(KN_j^{d-1})$ operations, while a collocation operation requires $O(KN_j^d)$ operations. Both of these operations are generally small compared to the elemental matrix multiplies. [The factor 2 in (3.57) is, in fact, present only in non-rectilinear geometry; however, we retain it here for generality.]

The computational complexity for the restriction (and hence prolongation) operation (3.33) is derived in a similar fashion to the Laplacian. We find that that the leading-order operation count for restriction of the residual from mesh j to mesh $j-1$ is given as

$$W_R^j = K[\sum_{n=1}^{d} (\frac{1}{2})^n] N_j^{d+1} \tag{3.58}$$

where we have again neglected the $O(KN_j^{d-1})$ operation count associated with the direct stiffness operation. Note that the work to evaluate the restriction is less than that for the Laplacian, as the size of the problem shrinks as each directional sum is performed.

We can now readily evaluate the (leading-order) work required per V-cycle on mesh j as

$$W_V^j = \{ (2m+1)W_A^j + 2W_R^j + ...\} \qquad \forall j \in \{2, ..., J\} \tag{3.59}$$

where the first term derives from steps (a), (b), (f), and (g) of $MG(j, \cdot, \cdot, m)$, while the second term derives from steps (c) and (e). By neglecting the restriction and prolongation work, and the work on all the coarser meshes $j = 1, .., J-1$, including the exact coarse-solution work, we arrive at the following simle expression for the *total* work per V-cycle (Rønquist and Patera [57]),

$$W_V \cong (2m+1)W_A^J. \tag{3.60}$$

We now make some final comments regarding spectral element multigrid in the context of vectorization and parallelism. In the above discussion we assume that the scaling of the computational work in terms of operation count

is representative for the scaling of the work in terms of actual cpu time, however this will only be true on serial machines. If the matrix-vector product operation is vectorized, this will give higher speed-up on the finer meshes due to the longer vector-lenghts. Hence, the assumption about neglecting the work on the coarser meshes ($j \neq J$) may not be valid. In particular, the elemental matrix-vector products on the coarsest mesh will typically have vector-lengths of order unity. It should also be noted that even if the computational cost associated with the coarser meshes ($j \neq J$) can be neglected, on a parallel machine the communication cost may be of the same order for a coarse mesh as for a fine mesh, especially if the start-up cost associated with sending messages is high. This may deteriorate the load-balancing, and thus the total potential speed-up using a multigrid strategy. However, continuous improvements in hardware lead us to believe that the multigrid concept will survive also in a parallel architecture environment.

3.3.5 One-Dimensional Results

We consider first the spectral radius of the multigrid iteration matrix corresponding to the $J = 2$ (two-grid) algorithm applied to the one-dimensional version of the problem (3.11-3.12). Defining the iteration error $\underline{e}^{(l)}$ as

$$\underline{e}^{(l)} = \underline{u}^J - \underline{u}^{J,(l)}, \tag{3.61}$$

and the norm

$$|||\ \underline{e}^{(l)}\ ||| = (\underline{e}^{(l)T} \underline{A}^J \underline{e}^{(l)})^{1/2}, \tag{3.62}$$

it can be shown (Bank and Douglas [5] that

$$||| \underline{e}^{(l+1)} |||\leq \rho(\underline{M}) ||| \underline{e}^{(l)} ||| \tag{3.63}$$

where $\rho(\underline{M})$ is the spectral radius of the multigrid iteration matrix

$$\underline{M} = (\underline{I} - \underline{L}^T[\underline{D}^J]^{-1}\underline{L})^m \cdot (\underline{I} - \underline{L}^T[\underline{R}^{J-1}]^T[\underline{A}^{J-1}]^{J-1}]^{-1}\underline{R}^{J-1}\underline{L})$$

$$\cdot (\underline{I} - \underline{L}^T[\underline{D}^J]^{-1}\underline{L})^m, \tag{3.64}$$

and $\underline{L}$, $\underline{L}^T$ are defined by the Cholesky decomposition of the positive definite symmetric matrix $\underline{A}^J$,

$$\underline{A}^J = \underline{L}\,\underline{L}^T. \tag{3.65}$$

From (3.60) we know that the total work is best represented in terms of number of $\underline{A}^J$ matrix-vector products, and we thus define a new iteration counter $l = (2m+1)\bar{l}$, for which

$$||| \underline{e}^{(\bar{l}+1)} |||\leq \bar{\rho} ||| \underline{e}^{(\bar{l})} |||, \tag{3.66}$$

where

$$\bar{\rho} = [\rho(\underline{M})]^{1/(2m+1)}. \tag{3.67}$$

The "work-deflated" spectral radius $\bar{\rho}$ allows us to compare various multigrid schemes while keeping the computational work constant.

In Table I we present $\bar{\rho}$ for various m, K, and N_J (with $N_2 \cong 2N_1$ in all cases). Also indicated is the condition number of $\underline{A}^J$, κ, for each $h = (K, N_J)$. The most important fact to note from Table I is that the multigrid convergence rate $\bar{\rho}$ is bounded from above well below unity, and is sensibly independent of K and N_J. Furthermore, there is an optimal number of smoothings which appears to be on the order of $m = 3$, except for the $K = 1$ "pure spectral" case. These

numerical results are all in good agreement with the theoretical results given in (3.48) and Maday and Muñoz [39].

Our work estimate (3.60) assumes that the exact-solution work on the coarsest ($j = 1$) mesh is negligible. This will clearly only be true if $N_1 \sim O(1)$, which in turn requires several multigrid levels to be used, $J > 2$. We now demonstrate that the convergence rates of Table I for $J = 2$ are also indicative of performance for $J > 2$. In particular, we present convergence histories for the one-dimensional version of (3.11-3.12) in which the inhomogeneity f is chosen such that the solution is given by

$$u = \frac{1}{10} e^{8(x-1)} sin(10\pi x) \qquad x \in \Lambda =]-1, 1[. \tag{3.68}$$

Note that u contains sufficient high and low wave numbers so that neither Jacobi iteration nor coarse-grid solution alone is sufficient to provide rapid convergence. All results presented are for the particular (but representative) discretization $K = 8$, $N_J = 12$ (with $N_{j+1} = 2N_j$).

In Fig. 28 we plot $||| \underline{\tilde{e}}^{\bar{l}} |||$ as a function of $\bar{l}$ for the $J = 2$ and $J = 3$ multigrid algorithms. The results indicate that the spectral radius for the $J = 2$ case is indeed a good predictor for the actual performance both for the $J = 2$ and $J = 3$ case, thus suggesting that the multigrid convergence rates will be insensitive to the number of multigrid levels. In Fig. 29 we also compare the convergence history for the $J = 3$ multigrid scheme with pure Jacobi smoothing and unpreconditioned conjugate gradient iteration, where the saving due to multigrid are clearly seen.

3.3.6 Two-Dimensional Results

Consider now the two-dimensional Laplace equation on a rectilinear domain $\Omega =]-0.5, 0.5[\times]-.25, .25[$, with imposed Dirichlet boundary conditions on the domain boundary $\partial\Omega$ such that the solution is given as $u = 0.5 + x$. The initial guess for the solution in the interior is taken to be zero. The fact that the solution can be exactly represented by a first-order polynomial does not affect the multigrid convergence rate, however it allows us to monitor the error $||| \underline{u} - \underline{u}_A |||$ which will be entirely due to incomplete multigrid convergence, with no contribution from discretization errors.

In the first test we use two grids ($J = 2$) in the multigrid algorithm. The discretization on the fine mesh is $h_2 = (K, N_2) = (50, 8)$ while the discretization on the coarse mesh is $h_1 = (K, N_1) = (50, 4)$, see the two finest grids in Fig. 30. We apply Chebyshev acceleration in the smoothing iteration on the fine mesh, with 10 smoothings per V-cycle, that is, $m = 5$. In Fig. 31 we plot the convergence histories for the multigrid algorithm and the preconditioned conjugate gradient iteration. For comparison we also include the multigrid convergence history without the Chebyshev acceleration. On the horizontal axis we plot the number of matrix-vector products on the fine mesh, thus neglecting the exact coarse-solution work. The convergence rate for the MG-algorithm is impressive compared to the CG-algorithm, especially in the context of the Stokes solvers discussed in the next section where the residuals in the inner elliptic solves only need to be reduced by a few orders of magnitude.

We now change the discretization to $h_2 = (K, N_2) = (50, 12)$ and $h_1 =$

$(K, N_1) = (50, 6)$, that is, the polynomial degree N is increased with a factor of 1.5 on both meshes. In Fig. 32 we see that the convergence rates have decreased both for the multigrid algorithm and the conjugate gradient iteration. To achieve a fixed (high) accuracy, the number of matrix-vector products have increased with a factor ~ 1.5 for the conjugate gradient iteration, consistent with the fact that the number of iterations scales like the square root of the condition number (3.10), that is, the number of iterations is proportional with N_2. The increase in work for the multigrid algorithm is about ~ 1.2, consistent with the result (3.56) that the work should scale as $\sqrt{N_2}$

Next, we investigate how the convergence rate depends on the number of elements K. The same testproblem is again solved with J=2 grids, for discretizations ($K = 50$, $N_2 = 12$) and ($K = 8$, $N_2 = 12$), with $N_2 = 2N_1$. The convergence histories plotted in Fig. 33 demonstrate the important result that the multigrid convergence rate is independent of the number of elements K, consistent with (3.56).

With only two meshes ($J = 2$) the assumption about neglecting the exact coarse-solution work is not good. We therefore repeat the first test, but now using $J = 4$ grids. The discretizations in the multigrid algorithm will then be: $h_4 = (50, 8)$, $h_3 = (50, 4)$, $h_2 = (50, 2)$, and $h_1 = (50, 1)$ with the coarsest mesh ($j = 1$) corresponding to an h-type finite element discretization on the macro-elemental skeleton, see Fig. 30. The multigrid convergence history is plotted in Fig. 34, demonstrating almost no change from the ($J = 2$) result plotted in Fig. 31. Like in the one-dimensional case this result suggest that the convergence rate is almost insensitive to the number of meshes used. We

have also included in Fig. 35 the multigrid convergence history for the case of imposing zero Neumann boundary conditions on the two horizontal edges of $\partial\Omega$, suggesting that the convergence rate is also insensitive to the imposed boundary conditions.

The two-dimensional results presented here are all based on discretizations where the computational domain is broken up into the rectilinear spectral elements with aspect ratio equal to unity. Preliminary tests using spectral elements with aspect ratio different from unity indicate that the convergence rate deteriorates significantly even for an aspect ratio equal to two. We are currently working on making the convergence rate more robust for deformed and elongeted elements.

3.4 Stokes Solvers

In this section we consider solution of the algebraic systems resulting from spectral element discretization of the steady and unsteady Stokes equations; however the solution algorithms we present are equally appropriate also for other variational discretization procedures.

3.4.1 Steady Stokes Solvers

We consider first the solution of the steady Stokes problem in a domain $\Omega \in \mathcal{R}^d$, with Dirichlet velocity boundary conditions on the domain boundary

$\partial\Omega$,

$$-\mu\nabla^2\mathbf{u} + \nabla p = \mathbf{f} \qquad for\ \mathbf{x} \in \Omega, \tag{3.69}$$

$$-\nabla \cdot \mathbf{u} = 0 \qquad for\ \mathbf{x} \in \Omega, \tag{3.70}$$

$$\mathbf{u} = \mathbf{u}_0(\mathbf{x}) \qquad for\ \mathbf{x} \in \partial\Omega. \tag{3.71}$$

The variational discretization of the Stokes problem (3.69-3.71) can be written in the general form

$$\mu \underline{A}\,\underline{u}_i - \underline{D}_i^T p = \underline{B}\,\underline{f}_i \qquad i = 1, .., d, \tag{3.72}$$

$$-\underline{D}_i \underline{u}_i = 0, \tag{3.73}$$

where summation over repeated indicies is assumed, and underscore refer to nodal values. In the three-dimensional case ($d = 3$), $\underline{\mathbf{u}} = (\underline{u}_1, \underline{u}_2, \underline{u}_3)$ is the velocity, $\underline{\mathbf{f}} = (\underline{f}_1, \underline{f}_2, \underline{f}_3)$ is the prescribed force, $\underline{p}$ is the pressure, $\underline{A}$ and $\underline{B}$ are the discrete Laplacian and mass matrices, respectively, and $\underline{D} = (\underline{D}_1, \underline{D}_2, \underline{D}_3)$ is the discrete gradient operator. It is assumed that all boundary conditions are incorporated into the matrix operators. Equations of the form (3.72-3.73) arise in any variational discretization of (3.69-3.71); the particular form of the discrete operators for the case of spectral element discretizations is given in Section 2.4.

Our approach to solve the algebraic system (3.72-3.73) is to use a global iterative decoupling procedure, which is an extension of the classical Uzawa algorithm used in finite element analysis (Arrow, Hurwicz, and Uzawa [2]; Glowinski [23]; Temam [61]; Girault and Raviart [22]; Bristeau, Glowinski and Periaux [12]). The main reasons for choosing this approach are: it is more efficient in terms of computational complexity and memory requirement than a direct approach

(Yamaguchi, Cheng and Brown [67]; Bathe and Dong [6]), especially for large three-dimensional problems; it is a more attractive decoupling algorithm than the Poisson equation approach (Glowinski and Pironneau [24]; Kleiser and Schumann [32]) insofar that it requires no rediscretization of the original problem (3.69-3.71), and no pressure boundary conditions.

By formally solving the momentum equations for the velocity, and then applying the incompressibility constraint (3.73), we arrive at the following *discretely* equivalent statement

$$-\underline{S}\,\underline{p} = \underline{D_i}\underline{A}^{-1}\underline{B}\,\underline{f_i}, \tag{3.74}$$

$$\mu\underline{A}\,\underline{u_i} = \underline{D_i^T}\,\underline{p} + \underline{B}\,\underline{f_i} \qquad i = 1, ..., d \tag{3.75}$$

where

$$\underline{S} = \underline{D_i}\underline{A}^{-1}\underline{D_i^T} \tag{3.76}$$

is a positive semi-definite symmetric matrix. In essence, (3.74-3.76) corresponds to a decoupling of the original symmetric saddle problem (3.72-3.73) into two symmetric positive-definite forms. Note that since the system (3.74-3.76) is equivalent to (3.72-3.73), the theoretical error estimates discussed in Section 2.4 directly apply.

To solve the system (3.74-3.76) we first solve (3.74) for the pressure $\underline{p}$, and then (3.75) is solved for the $\underline{u_i}$ with $\underline{p}$ known. Since the system matrices $\underline{S}$ and $\underline{A}$ both are symmetric positive-(semi)definite, standard elliptic solvers like conjugate gradient iteration or multigrid techniques can readily be applied. Although the steady Stokes pressure matrix $\underline{S}$ is completely full due to the embedded inverse $\underline{A}^{-1}$ (thus clearly necessitating an iterative approach), the important

fact to realize is that $\underline{S}$ is extremely well-conditioned. This can be motivated by noting that $\underline{S}$ is essentially the product of two first-order gradient operators "divided" by the second-order Laplacian, suggesting that $\underline{S}$ is identity-like as regards its spectrum.

The above discussion motivates our approach to invert $\underline{S}$ based on a *nested* global inner/outer iterative procedure. The outer iteration is a preconditioned conjugate gradient iteration of the form (see Section 3.2)

$$\begin{aligned}
&\underline{p}_0; \quad \underline{r}_0 = \underline{D}_i \underline{A}^{-1} \underline{B} \underline{f}_i + \underline{S}\,\underline{p}_0; \quad \underline{q}_0 = \tilde{\underline{B}}^{-1} \underline{r}_0; \quad \underline{p}_0 = \underline{q}_0; \\
&a_m = -(\underline{q}_m, \underline{r}_m)/(\underline{p}_m, \underline{S}\,\underline{p}_m) \\
&\underline{u}_{m+1} = \underline{u}_m + a_m \underline{p}_m \\
&\underline{r}_{m+1} = \underline{r}_m + a_m \underline{S}\,\underline{p}_m \\
&\underline{q}_{m+1} = \tilde{\underline{B}}^{-1} \underline{r}_{m+1} \\
&b_m = (\underline{q}_{m+1}, \underline{r}_{m+1})/(\underline{q}_m, \underline{r}_m) \\
&\underline{p}_{m+1} = \underline{r}_{m+1} + b_m \underline{p}_m
\end{aligned} \tag{3.77}$$

where we have used the *diagonal* mass matrix $\tilde{\underline{B}}$ associated with the pressure mesh (M_h) as a preconditioner. The motivation behind this choice of preconditioner is the fact that the variational equivalent of the identity operator is the mass matrix associated with the scalar product $(\cdot,\cdot)_{h,G}$.

The inner iteration is associated with the evaluation of the the matrix-vector product $\underline{S}\,\underline{q}$ in the outer conjugate gradient iteration,

$$\underline{y}_i = \underline{D}_i^T \underline{q} \qquad i = 1, ..., d \tag{3.78}$$

$$\underline{A}\,\underline{z}_i = \underline{y}_i \qquad i = 1, ..., d \tag{3.79}$$

$$\underline{S}\,\underline{q} = \underline{D}_i \underline{z}_i. \tag{3.80}$$

We note that each matrix-vector product evaluation is composed of tensor-product matrix multiplications and d *standard* elliptic Laplacian solves in $\mathcal{R}^d$. To invert the discrete Laplacian $\underline{A}$ we use conjugate gradient iteration or spectral element multigrid as described in Section 3.2 and 3.3. If the condition number of the matrix $\underline{\tilde{B}}^{-1}\underline{S}$ is order unity, we see that the above algorithm requires only order d elliptic solves, and is therefore an ideal decoupling of the Stokes problem. Finally, we note that the residual $\underline{r}$ in the outer conjugate gradient iteration is precisely the discrete divergence $-\underline{D}_i\underline{u}_i$. This is a useful result as it allows for direct control of the discrete divergence when specifying the tolerance for the outer iteration.

Full Fourier Case

We now consider the conditioning of $\underline{S}$ in more detail, and present theoretical and numerical results for the spectrum of this operator. We start by first considering the trivial case of Fourier discretization in $\mathcal{R}^d$, in which we choose the approximation spaces X_h and M_h to be

$$X_h = M_h^3, \tag{3.81}$$

$$M_h = span\{e^{i\mathbf{k}\cdot\mathbf{x}},\ \mid k_j \mid < K\ \forall j = 1, ..., d\} \tag{3.82}$$

where $\mathbb{k} = (k_1, k_2, k_3)$ is the wavevector, $\mathbf{x} = (x_1, x_2, x_3) \in \Omega$, and K is the maximum wavenumber in each spatial direction. Reality is imposed by conjugate

symmetry. For this Fourier discretization it is clear that

$$\underline{B} \Rightarrow \underline{I} \tag{3.83}$$

$$\underline{D} \Rightarrow ik_j \tag{3.84}$$

$$\underline{A} \Rightarrow -k^2 = -\sum_{j=1}^{d} k_j k_j, \tag{3.85}$$

from which it follows that $\underline{S} = \underline{I}$ independent of K. For the Fourier case the Uzawa algorithm is perfectly conditioned as might be expected.

Semi-Periodic Problem

Next, we return to the analysis of the semi-periodic problem, see Section 2.4. This problem includes boundaries, and is thus much more instructive than the full Fourier case, yet it is sufficiently simple to allow for a complete analysis. The semi-periodic problem corresponds to the domain $\Omega = (x, y) =]-1, 1[\times]0, 2\pi[$, with semi-periodic boundary conditions of the form

$$\forall y \in]0, 2\pi[\qquad \mathbf{u}(-1, y) = \mathbf{u}(1, y) = \mathbf{0} \tag{3.86}$$

$$\forall x \in]-1, 1[\qquad \mathbf{u}(x, 0) = \mathbf{u}(x, 2\pi), \tag{3.87}$$

and associated spaces

$$X = \mathcal{H}_0^1(]-1, 1[; span\{e^{iky}, \mid k \mid < K\}) \tag{3.88}$$

$$M = \mathcal{L}^2(]0, 2\pi[; span\{e^{iky}, \mid k \mid < K\}) \cap \{\phi \in \mathcal{L}^2(\Omega); \int_\Omega \phi \, d\Omega = 0\}. \tag{3.89}$$

We now write the velocity, the pressure and the data as a Fourier series in the periodic y-direction,

$$\mathbf{u}(x, y) = \sum_{k=-K}^{K} \hat{\mathbf{u}}^k(x) e^{iky}, \tag{3.90}$$

$$p(x,y) = \sum_{k=-K}^{K} \hat{p}^k(x) e^{iky}, \tag{3.91}$$

$$\mathbf{f}(x,y) = \sum_{k=-K}^{K} \hat{\mathbf{f}}^k(x) e^{iky}, \tag{3.92}$$

and use the orthogonality of the Fourier modes to reduce the variational form of the steady Stokes problem to a series of decoupled (continuous) problems. It can then be shown (Maday and Patera [40]; Maday, Meiron, Patera, and Rønquist [37]) that the spectrum of the *continuous* operator $\underline{\tilde{B}}^{-1}\underline{S} \Rightarrow S$ is

$$\lambda_0^S(k) = \frac{1}{2} - \frac{k}{sinh\, 2k} \tag{3.93}$$

$$\lambda_1^S(k) = \frac{1}{2} + \frac{k}{sinh\, 2k} \tag{3.94}$$

$$\lambda_l^S(k) = 1 \qquad l > 1, \tag{3.95}$$

with only one non-unity eigenvalue for each boundary.

The decoupled (continuous) equations for each Fourier mode k is now discretized using spectral elements in the non-periodic x-direction (Maday, Patera, and Rønquist [42]), and we arrive at a set of algebraic equations of the form (3.72-3.73), which is then decoupled into the form (3.74-3.76). We now present numerical results demonstrating the good conditioning of the (preconditioned) pressure matrix $\underline{\tilde{B}}^{-1}\underline{S}$ for the semi-periodic problem; in what follows, $\lambda_l^S(k)$, $\kappa^S(k)$ will refer to the spectrum and conditioning of $\underline{\tilde{B}}^{-1}\underline{S}$.

We begin by plotting in Fig. 36 the $\lambda_l^S(k)$ for the spectral element discretization $h = (K,N) = (4,7)$ and wavenumber $k = 1$. The agreement with the continuous operator spectrum is seen to be virtually exact. In Fig. 37 we again plot $\lambda_l^S(k)$ with $h = (K,N) = (4,7)$, but now for a wavenumber

$k = 12$. The low modes of the system are again in good agreement with the continuous spectrum, however at this large value of k the discrete system can no longer resolve exactly the higher modes, resulting in a cluster of eigenvalues at $\lambda_*^S \sim 1.2$. If we investigate the spectrum for $k = 12$, but now using a discretization $h = (K, N) = (4, 14)$, we see in Fig. 38 that the cluster of numerical eigenvalues has almost disappeared due to the higher spatial resolution in x.

We plot in Fig. 39 $\kappa^S(k)$ as a function of k for the spectral element and continuous operators. For small and moderate k the two curves coincide, however as $k \Rightarrow \infty$ the resolution becomes too low and the two curves diverge. For finer resolutions (e.g. larger N) the spectral element and the theoretical results agree over a larger range of wavenumbers, as expected from Fig. 37 and Fig. 38. For large wavenumbers k the condition number $\kappa^S(k)$ for the spectral element discretization is larger than the value predicted by the continuous analysis, however the value is still of order unity as required for fast convergence of the outer iteration.

Multi-Dimensional Case

We consider now the Uzawa decoupling procedure as applied to multidimensional spectral element approximations. In particular, we consider a three-dimensional problem (3.69-3.71) in a domain Ω defined by $x_1 \in]0, 2\Gamma[$, $x_2 \in]-1, 1[$, $x_3 =]-1, 1[$ where Γ can be interpreted as the aspect ratio of the system. The prescribed force $\mathbf{f}$ is such that the exact solution is given by $\mathbf{u} = (u_1, u_2, u_3) = ((1 - x_2^2)(1 - x_3^2), 0, 0)$ and $p = sin\pi x_1/\Gamma \cdot cos\pi x_2 \cdot cos\pi x_3$. For

multi-dimensional problems the calculation of spectra is a difficult task, and we therefore instead produce convergence histories from which appropriate condition numbers can be inferred. In particular, we shall plot the residual $\| \underline{r} \|_{0,G}$ (essentially the root mean square of the divergence) as a function of the number of iterations, m, in the outer conjugate gradient iteration (3.77).

In Fig. 40 we plot $\| \underline{r} \|_{0,G}$ as a function of m for an aspect ratio $\Gamma = 1$ and for spectral element discretizations corresponding to $K = 8$, $N = 7$ and 10. The initial convergence rate is almost independent of N, however the asymptotic convergence rate does appear to be a weak function of N, in good agreement with the theoretical bound derived in Maday, Patera, and Rønquist [41]. The implication of this result is that the constant in the inf-sup stability condition due to Babuška [3] and Brezzi [11] is a weak function of N, that is, in the multi-dimensional case *weak* spurious modes might exist, see Vandeven [64].

In Fig. 41 we repeat the numerical experiment of Fig. 40, but now keeping the discretization parameter $h = (K, N)$ fixed and varying the aspect ratio Γ. As expected from the semi-periodic case, the convergence rate is somewhat lower for $\Gamma = 3$ as compared to $\Gamma = 1$, however the effect is small. These results demonstrate that the good conditioning of the quasi two-dimensional (semi-periodic) model problem do, indeed, extend to multi-dimensional problems.

In all the previous examples preconditioned conjugate gradient iteration was used in the nested inner/outer pressure iteration. In Fig. 42 we show a steady Stokes test problem where preconditioned conjugate gradient iteration in the outer pressure iteration is combined with spectral element multigrid for the

inner Laplacian solves. The test problem is again creeping flow in a "wedge", see Fig. 12, but now with the tip of the wedge removed. The spectral element discretization ($K = 40, N = 8$) is shown in Fig. 42a, and the solution in form of streamlines is shown in Fig. 42b. Note that the strength of the second eddy is almost identical to the case when the tip of the wedge is present, see Fig. 12.

In this test problem we have removed the tip of the wedge in order to be able to break up the computational domain into spectral elements with aspect ratio approximately equal to unity, see Fig. 42a. As discussed in Section 3.3 the convergence rate of the spectral element multigrid algorithm deteriorates significantly as the aspect ratio of the elements become much different from unity. For this particular steady Stokes test problem the total speed-up using multigrid with $J = 4$ meshes instead of preconditioned conjugate gradient iteration for the inner Laplacian solves was about 2.5 (timings on a CRAY-XMP). Note that due to the more inefficient vectorization of the matrix-vector products on the coarser meshes $j = 1$, 2, and 3 compared to the finest mesh ($j = 4$), the computational cost on the coarser meshes ($j \neq J$) cannot be neglected.

3.4.2 Unsteady Stokes Solvers

In Section 2.4.2 we derived a set of algebraic equations (2.134-2.135) resulting from a spectral element discretization of the *implicitly* treated unsteady Stokes problem (2.124-2.127). For reasons of efficiency and rigor, our approach to solve the system (2.134-2.135) will again be based on a global iterative technique. Proceeding in the same fashion as for the steady Stokes case, we arrive at the

following decoupled system equivalent to the saddle problem (2.134-2.135),

$$-\underline{S_t}\underline{p}^{n+1} = \underline{D_i}\underline{H}^{-1}\underline{g_i^n} \tag{3.96}$$

$$\underline{H}\underline{u_i}^{n+1} = \underline{D_i^T}\underline{p}^{n+1} + \underline{g_i^n} \qquad i = 1, ..., d \tag{3.97}$$

where

$$\underline{S_t} = \underline{D_i}\underline{H}^{-1}\underline{D_i^T} \tag{3.98}$$

is the unsteady Stokes pressure operator analogous to the steady operator $\underline{S}$ defined in (3.76),

$$\underline{H} = \mu\underline{A} + \frac{\rho}{\Delta t}\underline{B} \tag{3.99}$$

is the discrete Helmholtz operator, and

$$\underline{g_i^n} = \underline{B}(\underline{f_i^n} + \frac{\rho}{\Delta t}\underline{u_i^n}) \qquad i = 1, ..., d \tag{3.100}$$

represent the inhomogeneities associated with an implicit Euler backward time integration procedure. The advantages of the formulation (3.96-3.98) are similar to those for the steady problem; it represents a complete, general velocity-pressure decoupling which is discretely equivalent to the original discretization (2.134-2.135). First, we solve (3.96) for the pressure, and then (3.97) is solved for each velocity component $\underline{u_i}^{n+1}$ with $\underline{p}^{n+1}$ known.

As for the steady Stokes problem the matrix $\underline{S_t}$ is completely full, and thus solving (3.96) requires an iterative approach. Unfortunately, wheras the steady pressure operator $\underline{S}$ is naturally well-conditioned ($\underline{\tilde{B}}^{-1}\underline{S}$ is identity like), the same is not true for $\underline{S_t}$. In particular, for large time steps we can express $\underline{S_t}$ as

$$\Delta t \Rightarrow \infty, \quad \underline{S_t} \Rightarrow \frac{1}{\mu}\underline{S} \tag{3.101}$$

and is thus well-conditioned, however, for small time steps $\underline{S}_t$ goes to the pseudo-Laplacian $\underline{E}$,

$$\Delta t \Rightarrow 0, \quad \underline{S}_t \Rightarrow \frac{\Delta t}{\rho}\underline{E},$$

where

$$\underline{E} = \underline{D}_i \underline{B}^{-1} \underline{D}_i^T \tag{3.104}$$

is poorly conditioned. The matrix $\underline{E}$ is, in fact, the discrete consistent Poisson operator resulting from spectral element discretization of the *explicitly* treated unsteady Navier-Stokes problem (2.208-2.210). The algorithm described for the steady case therefore needs to be modified.

Earlier spectral element solvers used a two-level Richardson inner/outer iteration scheme to solve the discrete unsteady Stokes and Navier-Stokes equations (Rønquist and Patera [55]; Maday and Patera [40]). We shall here use a similar approach as Cahouet and Chabard [13], where we precondition the unsteady pressure operator $\underline{S}_t$ directly. The preconditioner we use is

$$\underline{P}^{-1} = \mu \tilde{\underline{B}}^{-1} + \frac{\rho}{\Delta t}\underline{E}^{-1}, \tag{3.105}$$

which can be motivated by looking at the two limits of very small and very large time steps. In both of these cases we expect $\underline{P}^{-1}\underline{S}_t$ to be close to the identity operator.

Full Fourier case

We shall now give a more convincing argument for the particular choice (3.105) as a preconditioner for $\underline{S}_t$ by considering the Fourier discretization (3.81-

3.82) in $\mathcal{R}^d$. From (3.83-3.85) it follows that the spectrum of $\underline{S}_t$ is given as

$$\lambda_k^{S_t} = \frac{-k^2}{-\mu k^2 + \frac{\rho}{\Delta t}} = [\mu + \frac{\rho}{\Delta t}(-k^2)]^{-1} \tag{3.106}$$

while the spectrum of the preconditioner (3.105) becomes

$$\lambda_k^{P^{-1}} = [\mu + \frac{\rho}{\Delta t}(-k^2)]. \tag{3.107}$$

In the Fourier case we therefore obtain

$$\lambda_k^{P^{-1}S_t} = 1, \tag{3.108}$$

implying that the unsteady pressure operator is perfectly preconditioned.

Multi-Dimensional Case

Our approach to invert the unsteady pressure operator is the same as for the steady case, namely a *nested* global inner/outer iterative procedure. Given the preconditioner (3.105) the outer conjugate gradient iteration takes the form

$$\begin{gathered}
\underline{p}_0; \quad \underline{r}_0 = \underline{D}_i \underline{H}^{-1} \underline{B} \underline{g}_i + \underline{S}_t \underline{p}_0; \quad \underline{q}_0 = \underline{P}^{-1} \underline{r}_0; \quad \underline{p}_0 = \underline{q}_0; \\
a_m = -(\underline{q}_m, \underline{r}_m)/(\underline{p}_m, \underline{S}_t \underline{p}_m) \\
\underline{u}_{m+1} = \underline{u}_m + a_m \underline{p}_m \\
\underline{r}_{m+1} = \underline{r}_m + a_m \underline{S}_i \underline{p}_m \\
\underline{q}_{m+1} = \underline{P}^{-1} \underline{r}_{m+1} \\
b_m = (\underline{q}_{m+1}, \underline{r}_{m+1})/(\underline{q}_m, \underline{r}_m) \\
\underline{p}_{m+1} = \underline{r}_{m+1} + b_m \underline{p}_m
\end{gathered} \tag{3.109}$$

where we for convenience have dropped the superscript n referring to the particular time step.

The inner iteration associated with the evaluation of the matrix-vector product $\underline{S}_i\underline{q}$ in the outer iteration now becomes

$$\underline{y}_i = \underline{D}_i^T \underline{q} \qquad i = 1, ..., d \tag{3.110}$$

$$\underline{H}\,\underline{z}_i = \underline{y}_i \qquad i = 1, ..., d \tag{3.111}$$

$$\underline{S}_i\underline{q} = \underline{D}_i\underline{z}_i, \tag{3.112}$$

where the discrete Helmholtz operator H is inverted by conjugate gradient iteration or spectral element multigrid. We note that the *structure* in the solution procedure is similar to the steady case, however the computational complexity associated with the preconditioning in the outer iteration is very different. For the steady case the inversion of the *diagonal* mass matrix $\underline{\tilde{B}}$ is trivial, wheras the unsteady case requires the inversion of the pseudo-Laplacian $\underline{E}$. If we count the inversion of the $\underline{E}$-matrix as one standard elliptic solve, each iteration in the outer conjugate gradient iteration takes $d + 1$ standard elliptic solves, as compared to d for the steady case. If the condition number of the matrix $\underline{P}^{-1}\underline{S}_i$ is order unity, we see that the algorithm (3.109) to compute the pressure again requires only order d elliptic solves. Once the pressure is known, another d elliptic solves is required to compute the velocity (3.97).

We now make some final remarks regarding the $\underline{E}$-matrix. Since the $\underline{E}$-matrix is essentially a second-order operator with Neumann boundary conditions, the convergence rate is slower than for the inversion of a standard Laplacian $\underline{A}$ or Helmholtz operator $\underline{H}$ with Dirichlet (velocity) boundary conditions. This has

also been our experience from numerical simulations. To this end, standard conjugate gradient iteration has been used to invert $\underline{E}$, although a multigrid approach is in good progress.

To demonstrate the effect of the preconditioner $\underline{P}$, we plot in Fig. 43 the convergence history for the residual $\| \underline{r} \|_{0,G}$ in the outer iteration (3.109). The residual is here monitored during the first time step in solving the (simulated) buoyancy-driven flow in the two-dimensional square cavity shown in Fig. 25. These results clearly indicate that the condition number for $\underline{S}_t$ is indeed of order unity.

3.4.3 Navier-Stokes Solvers

As dicussed in Section 2.6 we use a combined semi-implicit/ fully explicit method to solve the incompressible Navier-Stokes equations; the particular time stepping procedure depends on the flow field (Reynolds number). The semi-implicit algorithm is based on implicit treatment of the unsteady Stokes problem and explicit treatment of the nonlinear advective term. The solution algorithm for the Navier-Stokes equations can in this case be viewed as an implicit unsteady Stokes scheme with an "augmented" forcing term including the explicit convective contributions. The decoupled equations for the pressure and velocity therefore become the same as (3.96-3.98), with the inhomogeneity $\underline{g}_i^n$ now expressed as

$$\underline{g}_i^n = \underline{B}(\underline{f}_i^n + \frac{\rho}{\Delta t}\underline{u}_i^n) + \sum_{q=0}^{2} \alpha_q \underline{C}^{n-k}\underline{u}_i^n \qquad i = 1, ..., d. \tag{3.113}$$

In (3.113) $\underline{C}$ is the convection opertor derived in Section 2.5.

In the fully explicit algorithm (high Reynolds numbers) we follow the same decoupling approach as before, and we arrive at the following decoupled system equivalent to the saddle problem (2.213-2.214),

$$-\underline{E}\,\underline{p}^{n+1} = \underline{D}_i \underline{B}^{-1} \underline{g}_i^n \tag{3.114}$$

$$\underline{B}\,\underline{u}_i^{n+1} = \underline{D}_i^T \underline{p}^{n+1} + \underline{g}_i^n \qquad i = 1, ..., d, \tag{3.115}$$

where $\underline{E}$ is the pseudo-Laplacian defined in (3.104), and the inhomogeneity $\underline{g}_i^n$ given as

$$\underline{g}_i^n = \underline{B}(\underline{f}_i^n + \frac{\rho}{\Delta t}\underline{u}_i^n) + \underline{A}\,\underline{u}_i^n + \sum_{q=0}^{2} \alpha_q \underline{C}^{n-q}\underline{u}_i^n \qquad i = 1, ..., d. \tag{3.116}$$

We note that in the fully explicit case there is only one elliptic solve per time step, as only the pseudo-Laplacian $\underline{E}$ needs to be inverted. No nested iteration is required since the inversion of the diagonal mass matrix $\underline{B}$ is trivial.

Bibliography

[1] W.F. Ames. *Numerical Methods for Partial Differential Equations.* Academic Press, 1977.

[2] K. Arrow, L. Hurwicz, and H. Uzawa. *Studies in Nonlinear Programming.* Stanford University Press, Stanford, 1968.

[3] I. Babuska. Error bounds for the finite element method. *Numer. Math.*, 16:322, 1971.

[4] I. Babuška and M.R. Dorr. Error estimates for the combined h- and p-version of the finite element method. *Numer. Math.*, 37:257, 1981.

[5] R.E. Bank and C.C. Douglas. Sharp estimates for multigrid rates of convergence with general smoothing and acceleration. *SIAM J. Numer. Anal.*, 22:617, 1985.

[6] K.J. Bathe and J. Dong. Solution of incompressible viscous fluid flow with heat transfer. *J. Computers and Structures*, to appear.

[7] C. Bernardi and Y. Maday. A collocation method over staggered grids for the Stokes problem. *Int. J. Numer. Methods in Fluids*, 1988.

[8] C. Bernardi, Y. Maday, and B. Métivet. Spectral approximation of the periodic-nonperiodic Navier-Stokes equations. *Numer. Math.*, 1988.

[9] K. Brand, M. Lemke, and J. Linden. *Multigrid Bibliography.* Arbeitspapiere der GMD 206, Gesellschaft fur Mathematik und Datenverarbeitung, 1977.

[10] A. Brandt. Multi-level adaptive solutions to boundary value problems. *Math. Comp.*, 31:330, 1977.

[11] F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers. *Rairo Anal. Numer.*, 8 R2:129, 1974.

[12] M.O. Bristeau, R. Glowinski, and J. Periaux. Numerical methods for the Navier-Stokes equations. Applications to the simulation of compressible and incompressible viscous flows. *Computer Physics Report*, to appear.

[13] J. Cahouet and J.P. Chabard. Multi-domains and multi-solvers finite element approach for the Stokes problem. In R.P. Shaw, editor, *Proc. Fourth International Symposium on Innovative Numerical Methods in Engineering*, page 317, Springer, 1986.

[14] C. Canuto, M. Hussaini, A. Quarteroni, and T. Zang. *Spectral Methods in Fluid Dynamics.* Springer-Verlag, 1987.

[15] P. Ciarlet. *The Finite Element Method for Elliptic Problems.* North Holland, 1978.

[16] M.O. Deville and E.H. Mund. Chebyshev pseudospectral solution of second order elliptic equations with finite element preconditioning. *J. Comput Phys.*, 60:517–533, 1985.

[17] P. Fischer, E. Rønquist, D. Dewey, and A.T. Patera. Spectral element methods: Algorithms and Architectures. In *First Int. Symp. on Domain Decomposition Methods for Partial Differential Equations, SIAM*, page 173, 1987.

[18] P.F. Fischer. *Parallel spectral elements, algorithms and applications.* PhD thesis, Massachusetts Institute of Technology, 1989.

[19] D. Funaro. A multi-domain spectral approximation of elliptic equations. *Numer. Meth. for Partial Differential Equations*, 2:187, 1986.

[20] D. Funaro, A. Quarteroni, and P. Zanolli. An iterative procedure with interface ralaxation for the domain decomposition method. *to appear*, 1987.

[21] C.W. Gear. *Numerical initial value problems in ordinary differential equations.* Prentice-Hall, Englewood Cliffs, New Jersey, 1971.

[22] V. Girault and P.A. Raviart. *Finite Element Approximation of the Navier-Stokes Equations.* Springer, 1986.

[23] R. Glowinski. *Numerical Methods for Nonlinear Variational Problems.* Springer, 1984.

[24] R. Glowinski and O. Pironneau. On a mixed finite element approximation of the Stokes problem. *Numer. Math.*, 33:397, 1979.

[25] G.H. Golub and C.F. Van Loan. *Matrix computations.* John Hopkins University Press, Maltimore, Maryland, 1983.

[26] W.J. Gordon and C.A. Hall. Transfinite element methods. Blending function interpolation over arbitrary curved element domains. *Numer. Math.*,

21:109, 1973.

[27] D. Gottlieb and S.A. Orszag. *Numerical analysis of spectral methods.* SIAM, Philadelphia, 1977.

[28] W. Hackbusch and U. Trottenberg, editors. *Multigrid Methods*, Springer-Verlag, New York, 1982.

[29] L.W. Ho. *A spectral element stress formulation of the Navier-Stokes equations.* PhD thesis, Massachusetts Institute of Technology, 1989.

[30] K. Horiuti. Comparison of conservative an rotational forms in large eddy simulation of turbulent channel flow. *J. Comput. Phys.*, 71:343, 1987.

[31] J. Kim and P. Moin. Application of a fractional-step method to incompressible Navier-Stokes equations. *J. Comput. Phys.*, 59:308, 1985.

[32] L. Kleiser and U. Schumann. Spectral simulation of the laminar turbulent transition process in plane Poiseuille flow. In R.G. Voigt, D. Gottlieb, and M.Y. Hussaini, editors, *Spectral Methods for Partial Differential Equations*, SIAM, 1984.

[33] K.Z. Korczak and A.T. Patera. An isoparametric spectral element method for solution of the Navier-Stokes equations in complex geometry. *J. Comput. Phys.*, 62:361, 1986.

[34] E. Laurien and H. Fasel. Numerical investigation of the onset of chaos beween rotating cylinders. In *Proc. GAMM Annual Meeting*, Stuttart, 1987.

[35] M. Macaraeg. A spectral multi-domain technique with application to generalized curvilinear coordinates. In *Sixth International Symposium on Finite*

Element Methods in Flow Problems, Antibes, 1986.

[36] M.C. Macaraeg and C.L. Street. Improvements in spectral collocation through a multiple domain technique. *Appl. Numer. Math.*, 2:95, 1986.

[37] Y. Maday, D.I. Meiron, A.T. Patera, and E.M. Rønquist. Iterative saddleproblem decomposition methods for the steady and unsteady Stokes equations. in preparation.

[38] Y. Maday and R. Muñoz. Analysis of spectral element multigrid. In *Proceedings of the Eleventh International Conferance on Numerical Methods in Fluid Dynamics*, Springer-Verlag, to appear.

[39] Y. Maday and R. Muñoz. Spectral element multigrid. II. Theoretical justification. *J. Scient. Comput.*, to appear.

[40] Y. Maday and A.T. Patera. Spectral element methods for the Navier-Stokes equations. In A.K. Noor, editor, *State-of-the-art surveys in computational mechanics*, ASME, New York, 1988.

[41] Y. Maday, A.T. Patera, and E.M. Rønquist. Optimal Legendre spectral element methods for the multi-dimensional Stokes problem. in preparation.

[42] Y. Maday, A.T. Patera, and E.M. Rønquist. Optimal Legendre spectral element methods for the Stokes semi-periodic problem. *SIAM J. Numer. Anal.*, submitted.

[43] Y. Marion and B. Gay. Coordination by partition in subdomains and spectral methods. In *Fourth International Symposium on Innovative Numerical Methods in Engineering*, Springer, 1986.

[44] O.A. McBryan and E.F. Van de Velde. The multigrid method on parallel processors. In A. Dold and B. Eckmann, editors, *Multigrid Methods II. Proceedings of the 2nd European Conference on Multigrid Methods*, pages 232–260, Springer-Verlag, New York, 1985.

[45] B. Métivet. Resolution des equations de Navier-Stokes par methodes spectrales. 1987. These d'Etat, University Pierre and Marie Curie.

[46] B. Métivet and Y. Morchoisne. Multi-domain spectral technique for viscous flow calculations. In *Fourth GAMM Conference on Numerical Methods in Fluid Mechanics, Paris*, 1982.

[47] H.K. Moffatt. Viscous and resistive eddies near a sharp corner. *J. Fluid Mechanics*, 18:1, 1964.

[48] Y. Morchoisne. Resolution des equations de Navier-Stokes par une methode spectrale de sous-domaines. In P. Lascaux, editor, *Third International Conference on Numerical Methods in Science and Engineering*, 1983.

[49] S.A. Orszag. Numerical simulation of incompressible flows within simple boundaries: Accuracy. *J. Fluid Mech.*, 49:75, 1971.

[50] S.A. Orszag. On the elimination of aliasing in finite-difference schemes by filtering high-wavenumber components. *J. Atmospheric Sciences*, 28:1074, 1971.

[51] S.A. Orszag. Spectral methods for problems in complex geometries. *J. Comput. Phys.*, 37:70, 1980.

[52] A.T. Patera. Fast direct Poisson solvers for high-order finite element discretizations in rectangularly decomposable domains. *J. Comput. Phys.*, 65:474, 1986.

[53] A.T. Patera. A spectral element method for Fluid Dynamics; Laminar flow in a channel expansion. *J. Comput. Phys.*, 54:468, 1984.

[54] T.N. Phillips. Relaxation schemes for spectral multigrid methods. *J. Comput. Appl. Math.*, 18:149–162, 1987.

[55] E.M. Rønquist and A.T. Patera. A Legendre spectral element method for the incompressible Navier-Stokes equations. In *Proceedings of the Seventh GAMM Conferance on Numerical Methods in Fluid Mechanics*, Vieweg, to appear.

[56] E.M. Rønquist and A.T. Patera. A Legendre spectral element method for the Stefan problem. *Int. J. Num. Methods Eng.*, 24:2273, 1987.

[57] E.M. Rønquist and A.T. Patera. Spectral element multigrid. I. Formulation and numerical results. *J. Scient. Comput.*, 2(4):389, 1987.

[58] E.M. Rønquist, A.T. Patera, and M.O. Deville. Legendre spectral element solution of the axisymmetric Stokes problem. *J. Comput. Phys.*, in preparation.

[59] G. Strang and G. Fix. *An Analysis of the Finite Element Method.* Prentice-Hall, 1973.

[60] A.H. Stroud and D. Secrest. *Gaussian Quadrature Formulas.* Prentice Hall, Englewood Cliffs, New Jersey, 1966.

[61] R. Temam. *Navier-Stokes Equations. Theory and Numerical Analysis.* North-Holland, Amsterdam, 1984.

[62] M. Van Dyke. *An album of fluid motion.* The Parabolic Press, Stanford, California, 1982.

[63] H. Vandeven. Analysis of the eigenvalues of spectral differentiation operators. *SIAM J. Numer. Anal.*, submitted.

[64] H. Vandeven. Compatibilité des espaces discrets pour l'approximation spectrale du problème de Stokes périodique non périodique. M^2AN, to appear.

[65] J.A.C. Weidemann and L.N. Trefethen. The eigenvalues of second-order spectral differentiation matrices. *SIAM J. Numer. Anal.*, submitted.

[66] F.M. White. *Viscous Fluid Flow.* McGraw-Hill, 1974.

[67] Y. Yamaguchi, C.J. Chang, and R.A. Brown. Multiple bouyancy-driven flows in a vertical cylinder heated from below. *Phil. Trans. Roy. Soc. London*, A 312:519, 1984.

[68] T.A. Zang. On the rotation and skew-symmetric forms for incompressible flow simulations. *Applied Num. Math.*, to appear, 1988.

[69] T.A. Zang, Y.S. Wong, and M.Y. Hussaini. Spectral multigrid methods for elliptic equations. *J. Comput. Phys*, 48:485–501, 1982.

[70] T.A. Zang, Y.S. Wong, and M.Y. Hussaini. Spectral multigrid methods for elliptic equations II. *J. Comput. Phys*, 54:489–507, 1982.

Appendix A

Appendix: Legendre polynomials and Legendre-Lagrangian interpolants

The Legendre polynomials, $L_n(z)$, satisfy the singular Sturm-Liouville problem,

$$\frac{d}{dz}(1-z^2)\frac{dL_n(z)}{dz} + n(n+1)L_n(z) = 0,$$

$$L_n(1) = 1, \tag{A.1}$$

they are orthogonal with respect to a unity weighting,

$$\int_{-1}^{1} L_m(z)L_n(z)\,dz = \frac{2}{2m+1}\delta_{mn}, \tag{A.2}$$

and they satisfy the following three-term recurrence relation,

$$(n+1)L_n(z) = (2n+1)zL_n(z) - nL_{n-1}(z),$$

$$L_1(z) = z,$$

$$L_0(z) = 1. \tag{A.3}$$

In the Legendre spectral element method the $N+1$ local Gauss-Lobatto Legendre collocation points are defined as

$$z_0 = -1 \tag{A.4}$$

$$L'_N(z_i) = 0 \qquad i = 1, ..., N-1, \tag{A.5}$$

$$z_N = +1, \tag{A.6}$$

where prime denotes differentiation. With this choice of collocation points, the nodal Lagrangian interpolants, h_j, can be written as

$$h_j(z) = -\frac{(1-z^2)L'_N(z)}{\alpha_N L_N(z_j)(z-z_j)}, \tag{A.7}$$

where $\alpha_N = N(N+1)$. The Lagrangian interpolants $h_j(z)$ have the properties of being N^{th} order polynomials, and satisfying the relation

$$h_j(z_i) = \delta_{ij} \qquad \forall i,j \in \{0,...,N\}^2, \tag{A.8}$$

where δ is the Kronecker delta symbol. The first property follows from the fact that the numerator is a polynomial of degree $N+1$, while the denominator is a first-order polynomial. The property (A.7) is easily proven by using L'Hopital's rule, and then applying (A.1).

From the properties of the Lagrangian interpolants, any N^{th} order polynomial, $u(z)$, on $z \in]-1,1[$ can be written as

$$u(z) = \sum_{j=0}^{N} h_j(z)u(z_j). \tag{A.9}$$

The recurrence relation (A.3) is used to evaluate the Lagrangian interpolants, h_j, given by (A.7). It should be noted that (A.3) is subject to round-off errors for high-order expansions, but for the (relatively) low-order expansions used in spectral element calculations this is not a serious problem.

The first derivative of the polynomial $u(z)$ at the nodal points z_i is given by

$$u'(z_i) = \sum_{j=0}^{N} h'_j(z_i)u(z_j) = \sum_{j=0}^{N} D_{ij}u_j, \tag{A.10}$$

where $u_j = u(z_j)$, prime denotes differentiation, and the nodal interpolant derivative matrix, $\underline{D}$, is defined by

$$D_{ij} = \frac{dh_j}{dz}(z_i). \tag{A.11}$$

Using the expression (A.7) for the Lagrangian interpolants, the matrix elements D_{ij} can be written as

$$D_{ij} = \frac{L_N(z_i)}{L_N(z_j)(z_i - z_j)}, \quad i \neq j, \tag{A.12}$$

$$D_{ii} = 0, \quad i \neq 0, N \tag{A.13}$$

$$D_{00} = -\alpha_N/4, \tag{A.14}$$

$$D_{NN} = +\alpha_N/4 \tag{A.15}$$

Tables and Figures

Table 1. Table of the work-deflated J=2 multigrid matrix spectral radius $\bar{\rho}$ for various values of K, N_J, and smoothings m. The condition number of the A^J matrix, κ, is also given for each case. The convergence rate is extremely insensitive to h=(K,N_J), in particular for K>1 and moderate N_J.

K=1

m\N_2\|N_1	8|4	12|6	16|8	19|10	41|19
1	.745	.775	.788	.772	.839
2	.702	.736	.752	.733	.810
3	.685	.720	.737	.717	.798
4	.675	.711	.728	.708	.791
5	.669	.706	.723	.703	.787
10	.657	.694	.712	.691	.778
κ	35	103	232	381	3630

K=4

m\N_2\|N_1	8|4	12|6	16|8	19|10	41|19
1	.759	.779	.790	.773	
2	.718	.741	.754	.734	
3	.701	.725	.739		
4	.709	.720	.730		
5	.727	.733	.738		
10	.791	.788	.787		
κ	1151	3665	8469	14023	

K=8

m\N_2\|N_1	8|4	12|6	16|8	19|10	41|19
1	.760	.779	.790		
2	.719	.741	.754		
3	.702	.726	.739		
4	.710				
5	.731				
10	.794				
κ	4603	14622	33828		

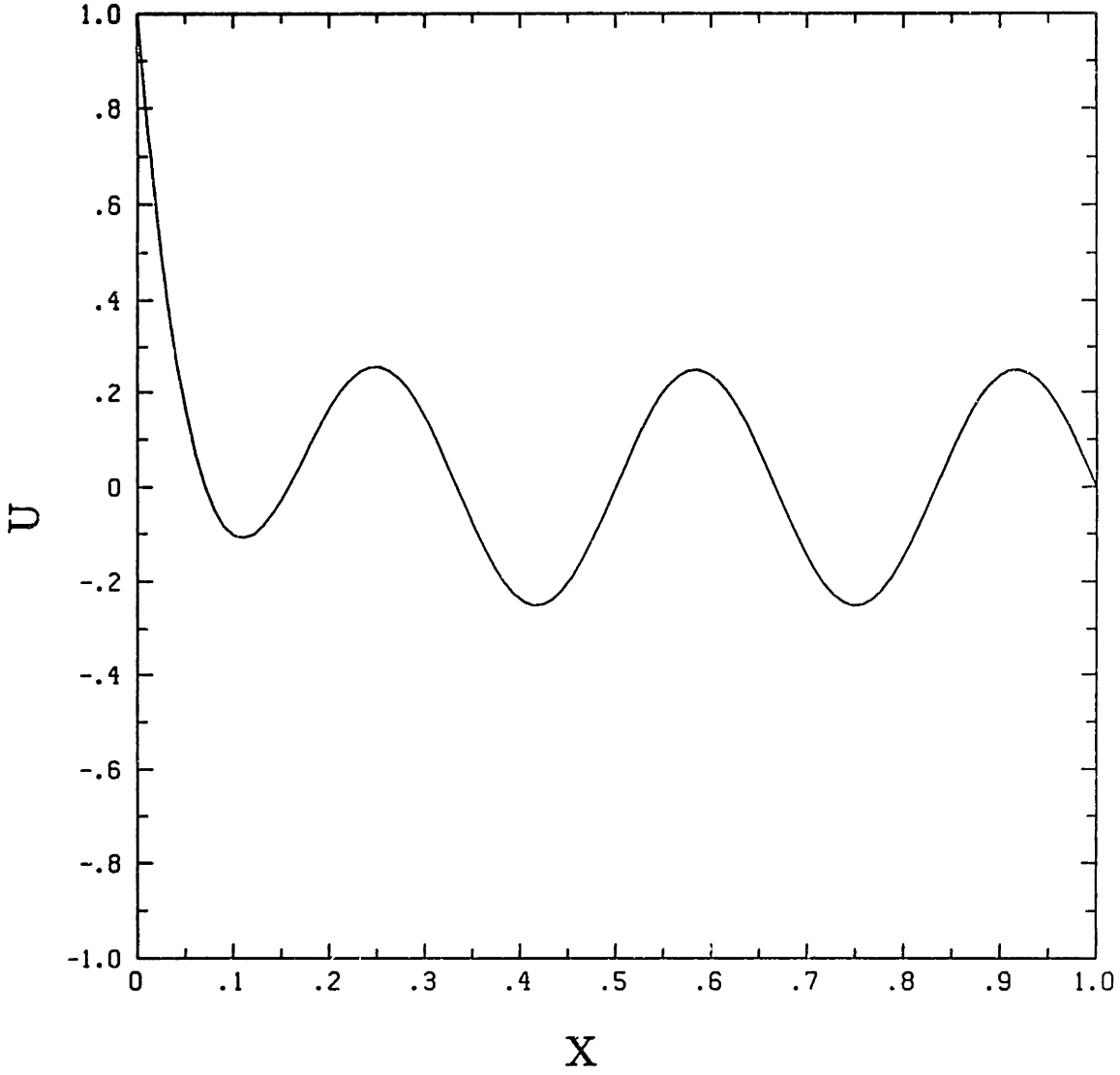


Figure 1. A plot of the solution $u(x)$ to the convection-diffusion equation $\alpha u_{xx} + u_x = f$, $u(0) = 1$, $u(1) = 0$, with data $f = 1.5\pi cos(6\pi x) - 0.45\pi^2 sin(6\pi x)$, and $\alpha = 0.05$.

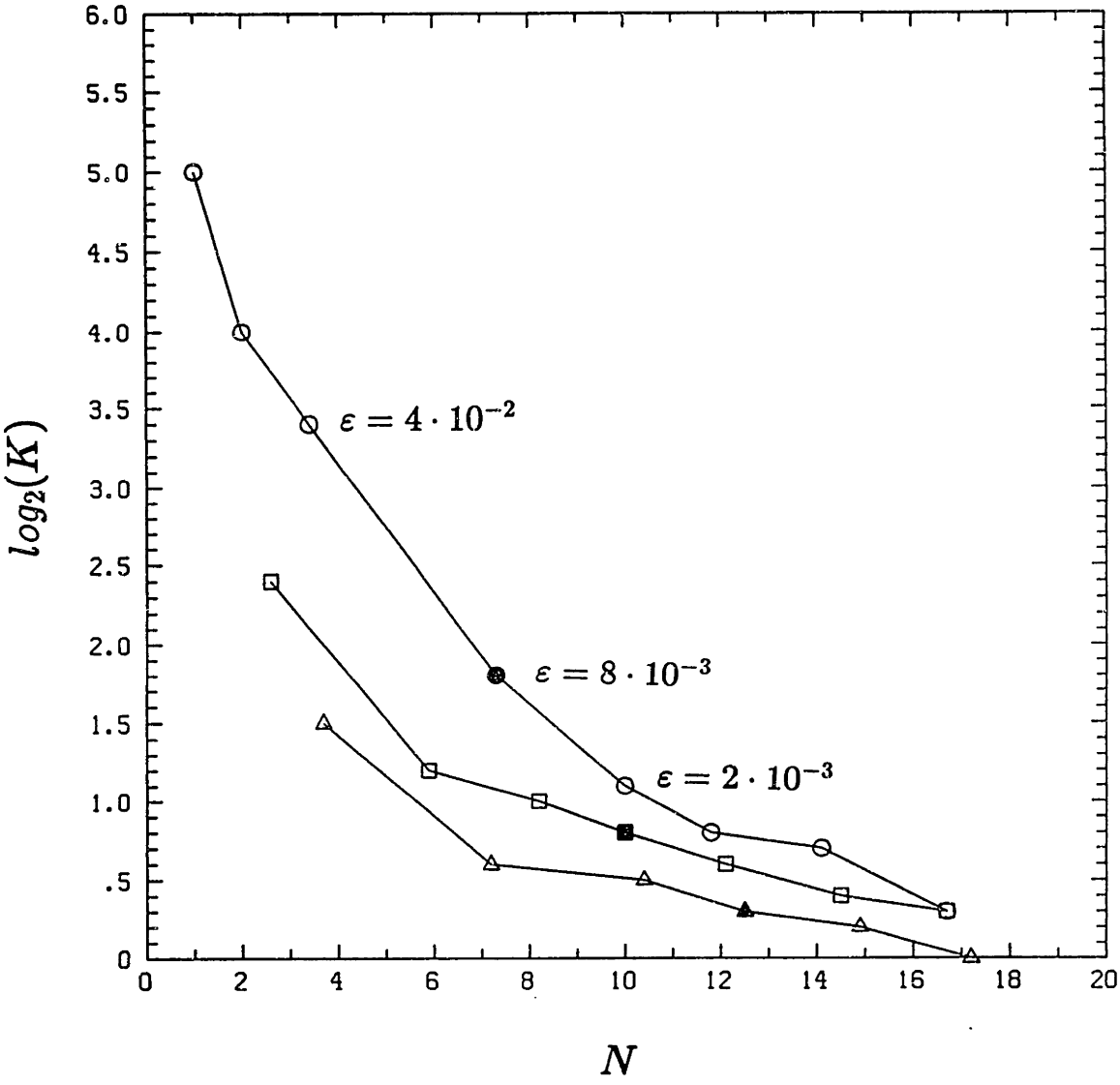


Figure 2. The plot shows the trajectories of optimal discretization pairs $h = (K, N)$ when solving a given problem in $\mathcal{R}^d$, $d = 1$ (○), $d = 2$ (□), $d = 3$ (△), to a specified error tolerance $\varepsilon = \| u - u_h \|_1$. Here, K is the number of elements and N is the polynomial degree. The filled symbols show optimal discretizations for fixed $\varepsilon = 8 \cdot 10^{-3}$ for $d = 1$, 2, and 3.

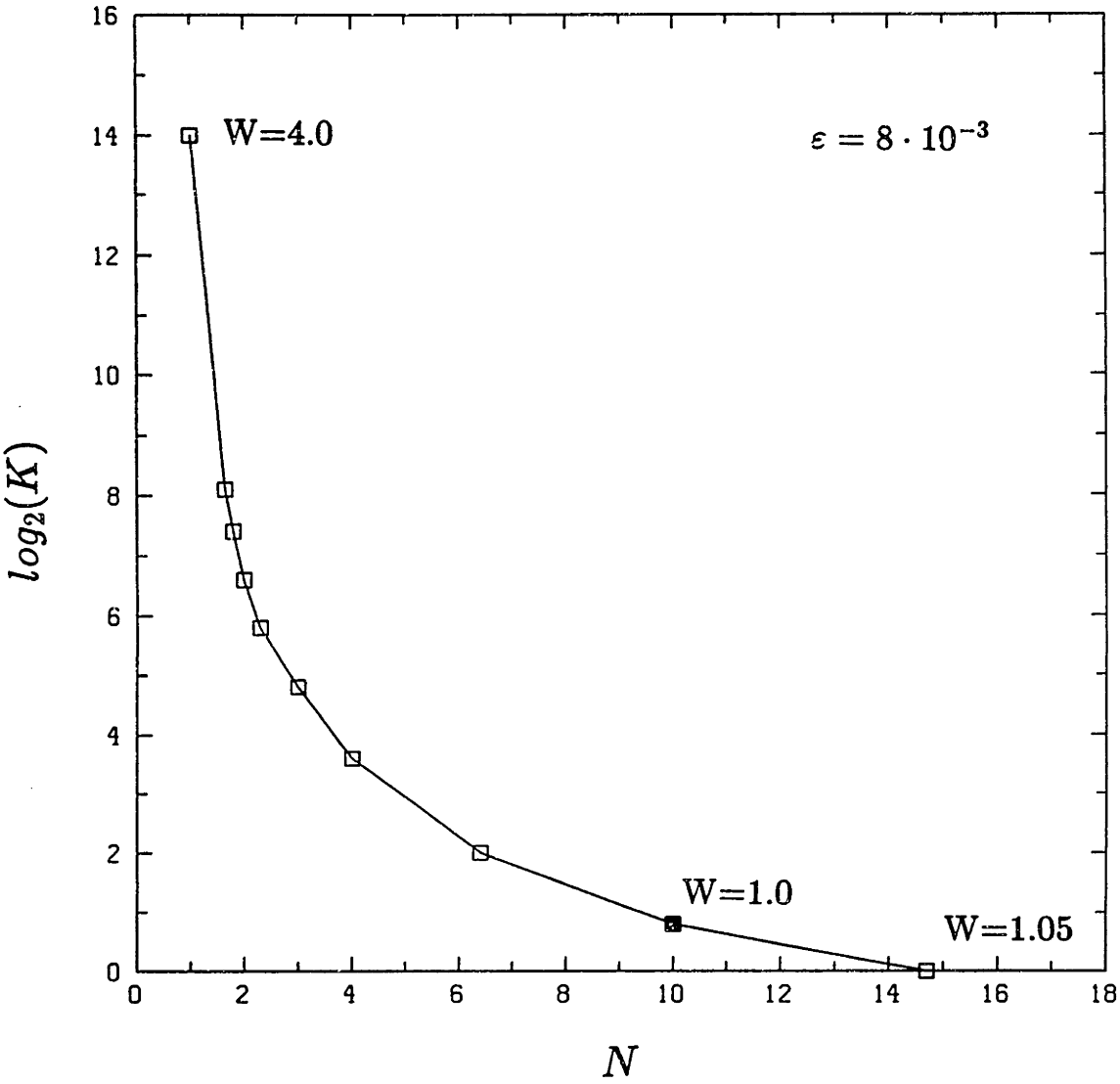


Figure 3. A plot of the computational work W as a function of the discretization $h = (K, N)$ when solving a two-dimensional problem $(d = 2)$ to a specified tolerance $\varepsilon = 8 \cdot 10^{-3}$. The computational work is given relative to the the optimal discretization where W is normalized to unity.

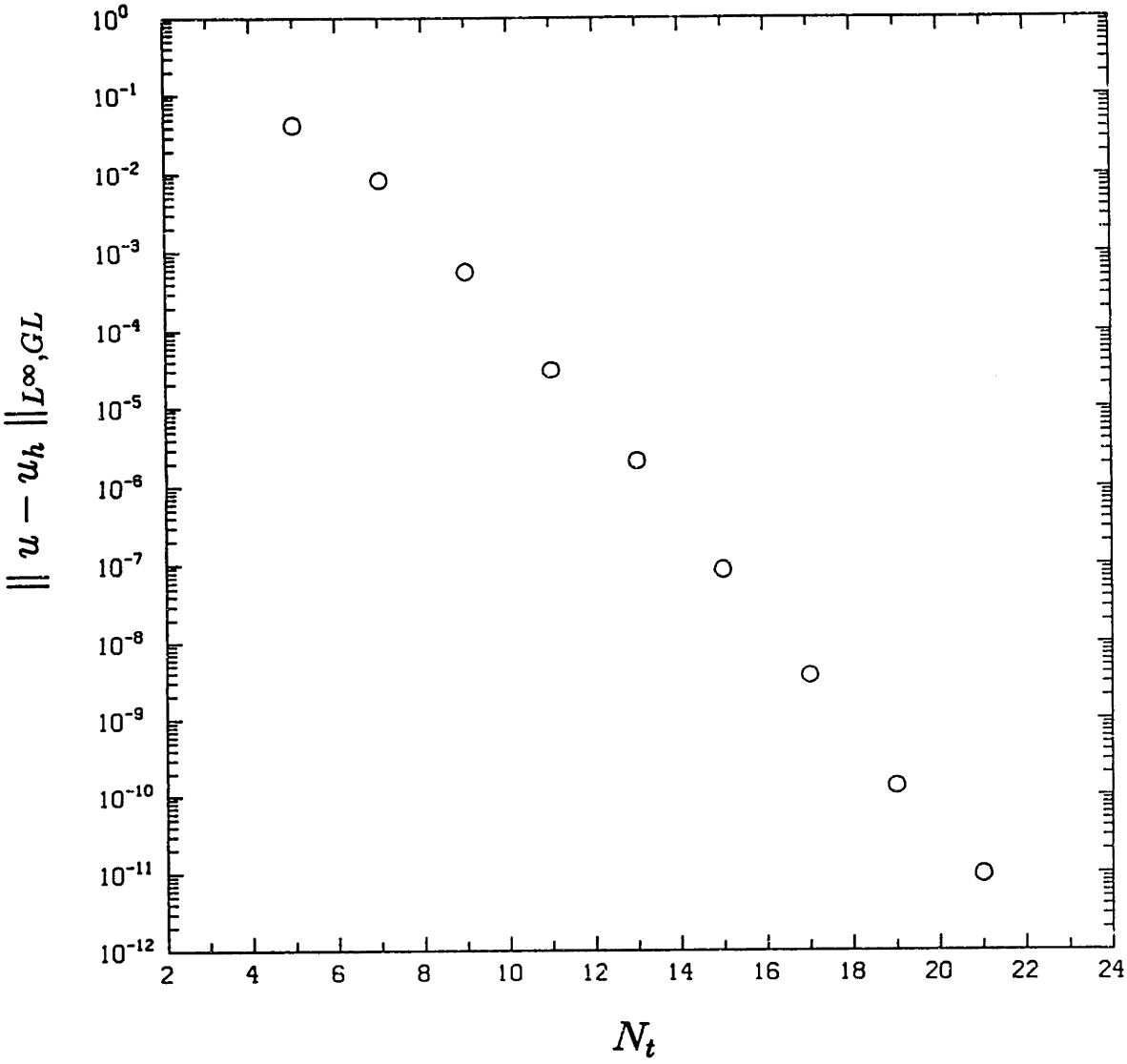


Figure 4. A plot of the maximum pointwise error in the spectral element solution of the differential equation (2.7) as a function of the total number of degrees-of-freedom, N_t. In this test case $\lambda = 0$, $p(x) = e^x$, $f(x) = e^x(\cos x - \sin x)$, for which the solution is $u(x) = -\sin x$ on $x \in]0, \pi[$. The domain is divided into $K = 2$ spectral elements of equal length. Exponential convergence is achieved as the polynomial degree of the fixed elements is increased.

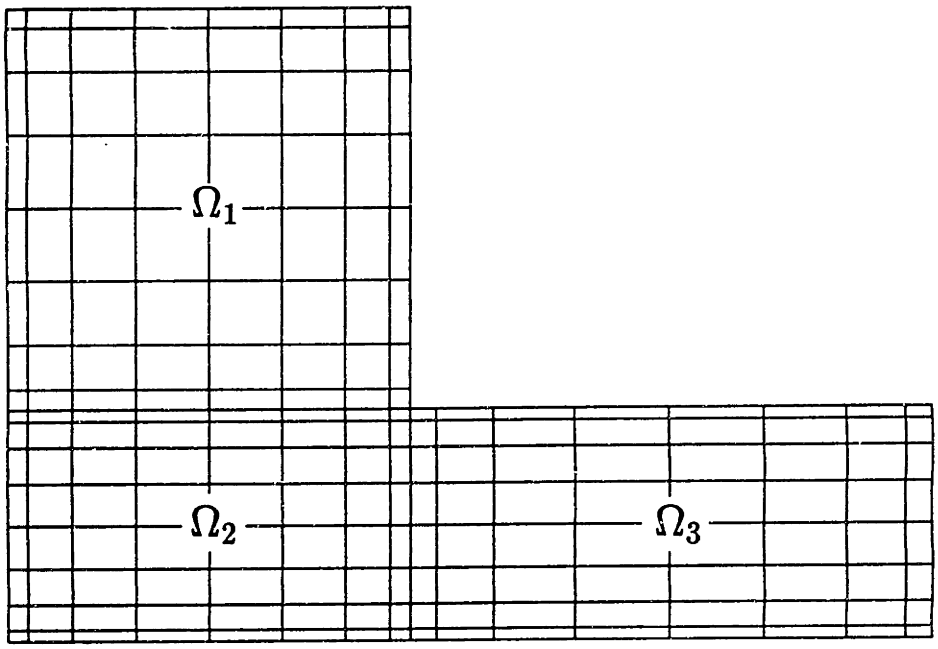


Figure 5. The two-dimensional computational domain is broken up into three rectilinear elements Ω_1, Ω_2, and Ω_3.

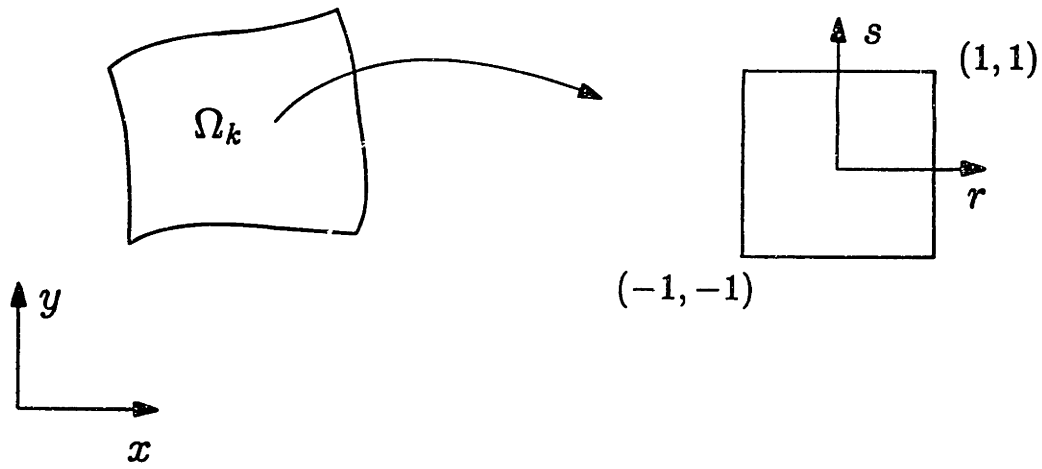


Figure 6. Each quadrilateral spectral element is mapped into a local (r,s)-system: $(x,y) \in \Omega_k \Rightarrow (r,s) \in]-1,1[^2$.

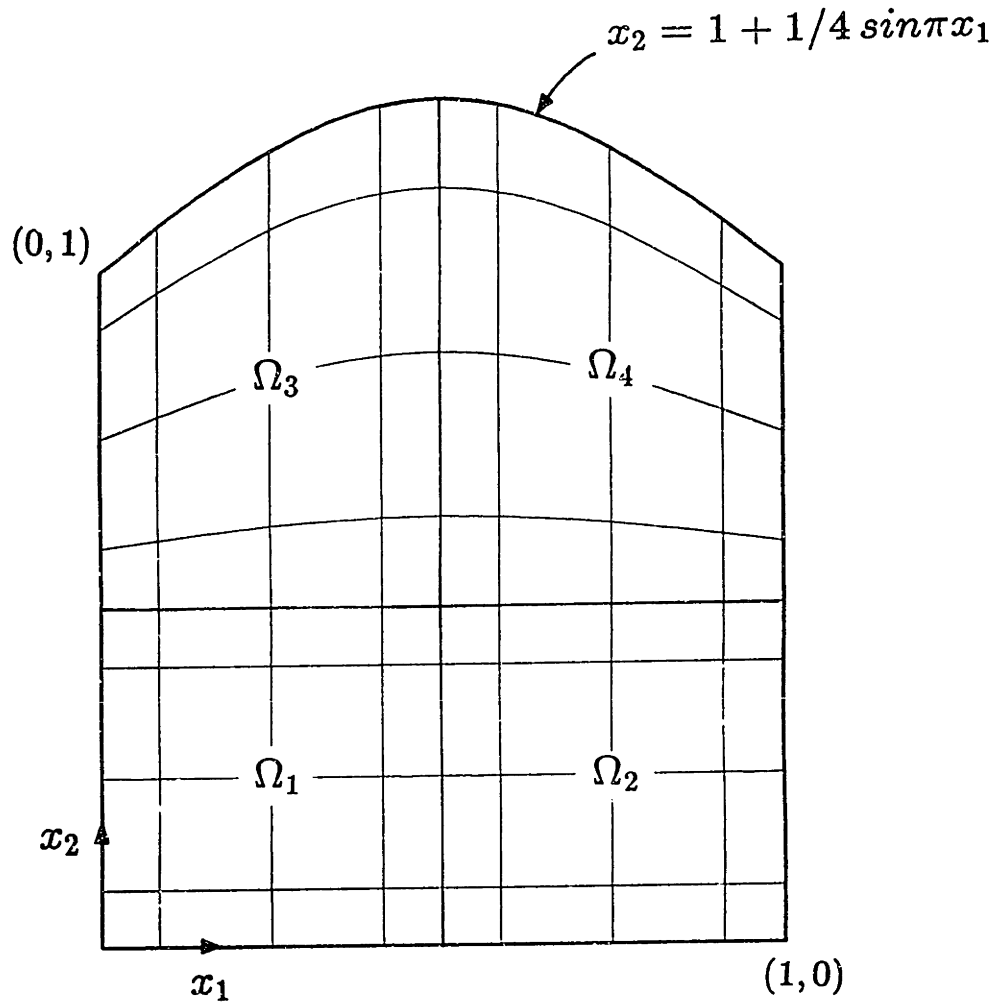


Figure 7. The domain and spatial discretization for solution of the Poisson equation (2.45) with $f = 0$. Dirichlet boundary conditions are imposed on $\partial\Omega$ such that the solution is $u(x_1, x_2) = sinx_1 \cdot e^{-x_2}$. The domain is divided into $K = 4$ spectral elements.

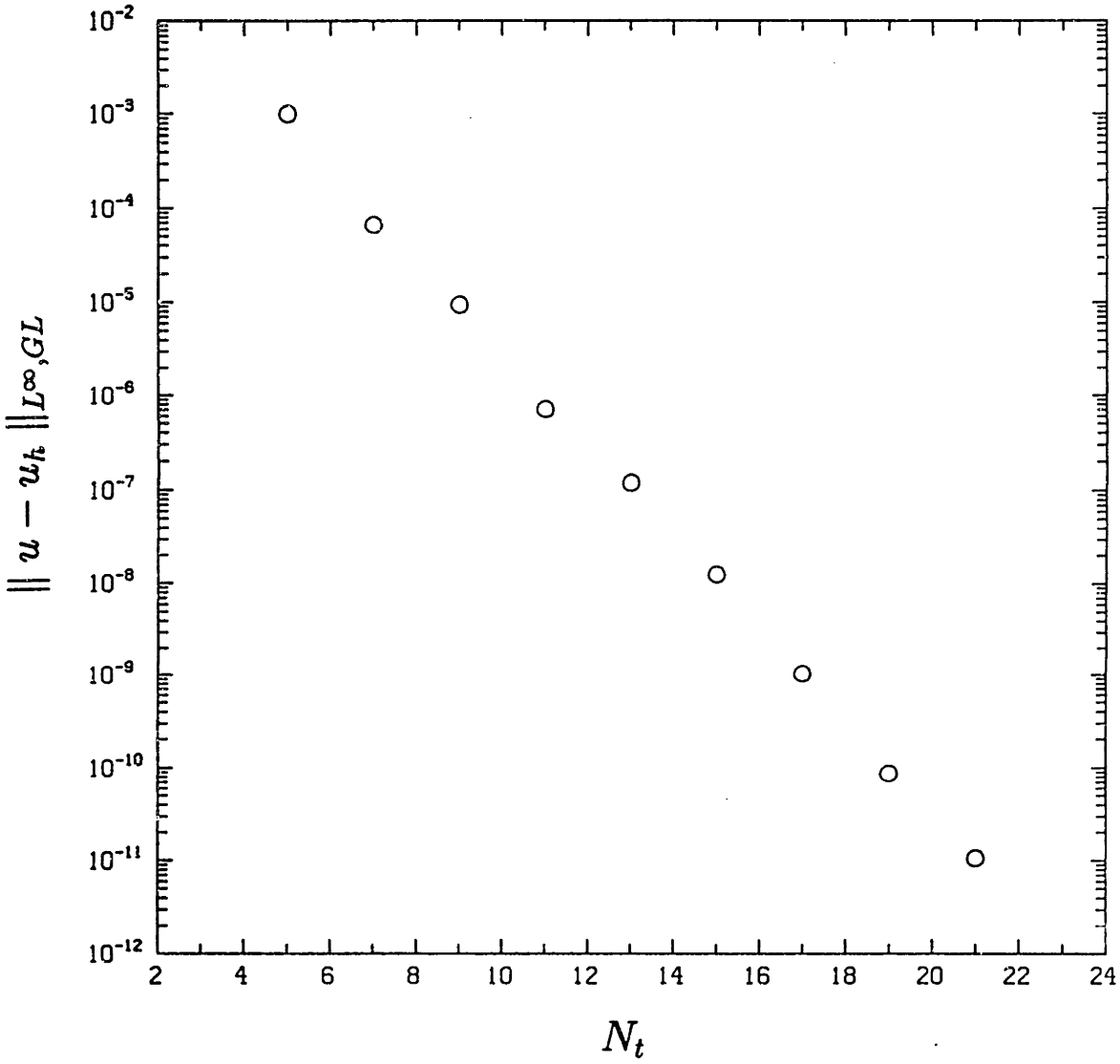


Figure 8. A plot of the maximum pointwise error in the Legendre spectral element solution to the Poisson equation (2.45) as a function of the total number of degrees-of-freedom in one spatial direction, N_t. Here, the data $f = 0$ and the solution $u = sinx_1 \cdot e^{-x_2}$. Exponential convergence is achieved as the degree of the elements, N, is increased for fixed number of elements, $K = 4$.

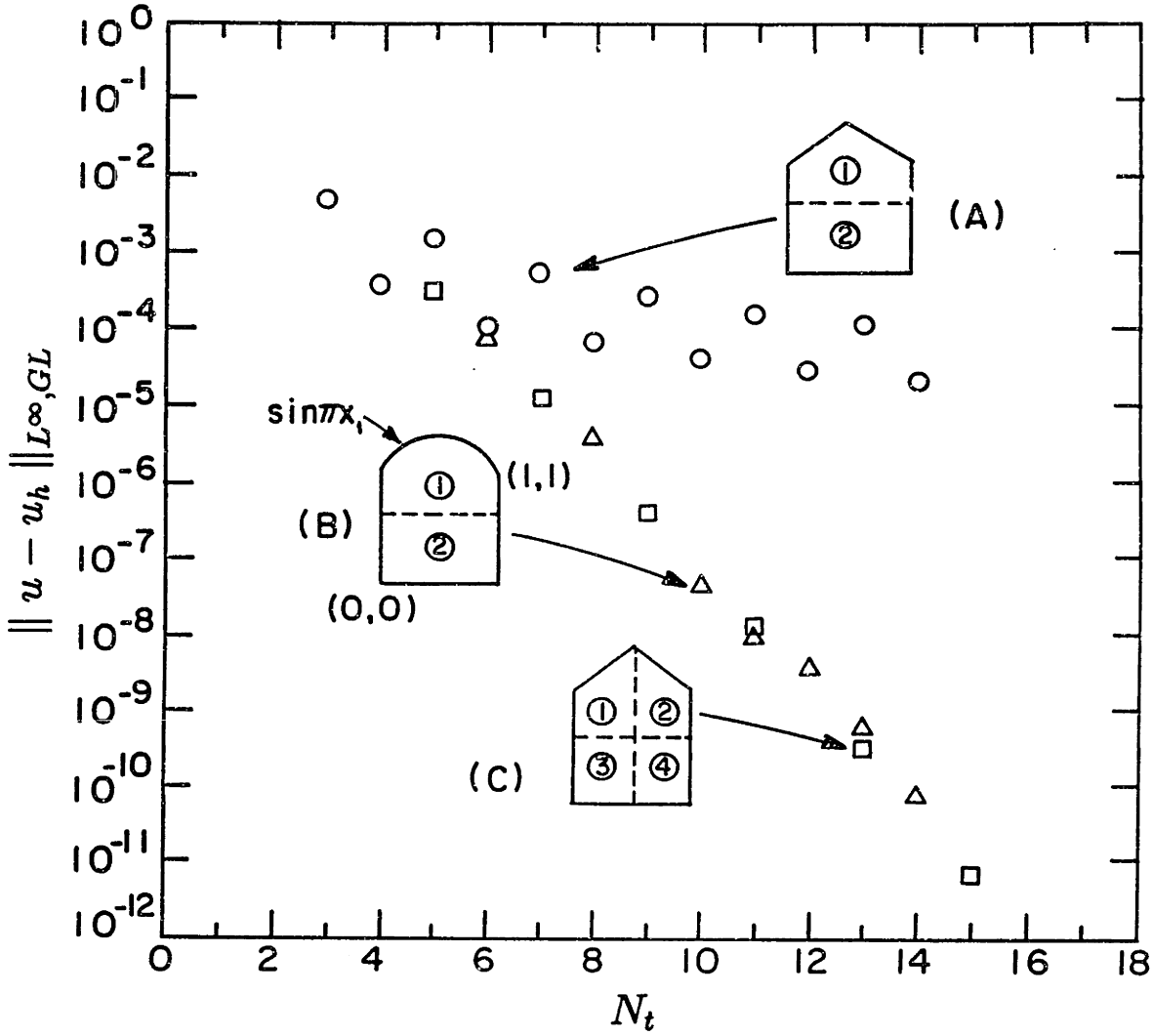


Figure 9. A plot of the maximum pointwise error in the Legendre spectral element solution to the Poisson equation (2.45) as a function of the total number of degrees-of-freedom in the x_1-direction, N_i. Here, the data $f = 0$ and the solution $u = sinx_1 \cdot e^{-x_2}$. The problem is solved for three different geometries and discretizations, (A), (B), and (C), where the element numbering is indicated.

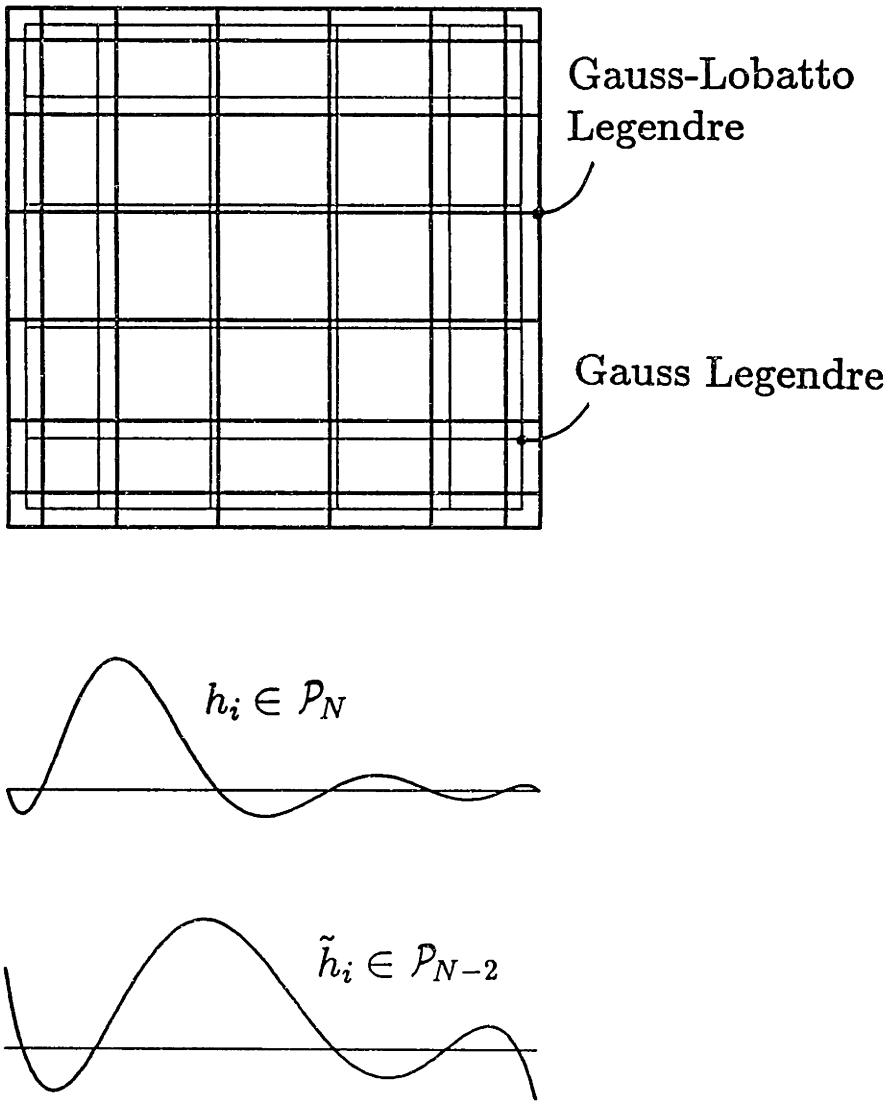


Figure 10. A plot of the staggered mesh for $K = 1$ and $N = 7$. Examples of the one-dimensional Lagrangian interpolants h_i and $\tilde{h}_i$ are also shown.

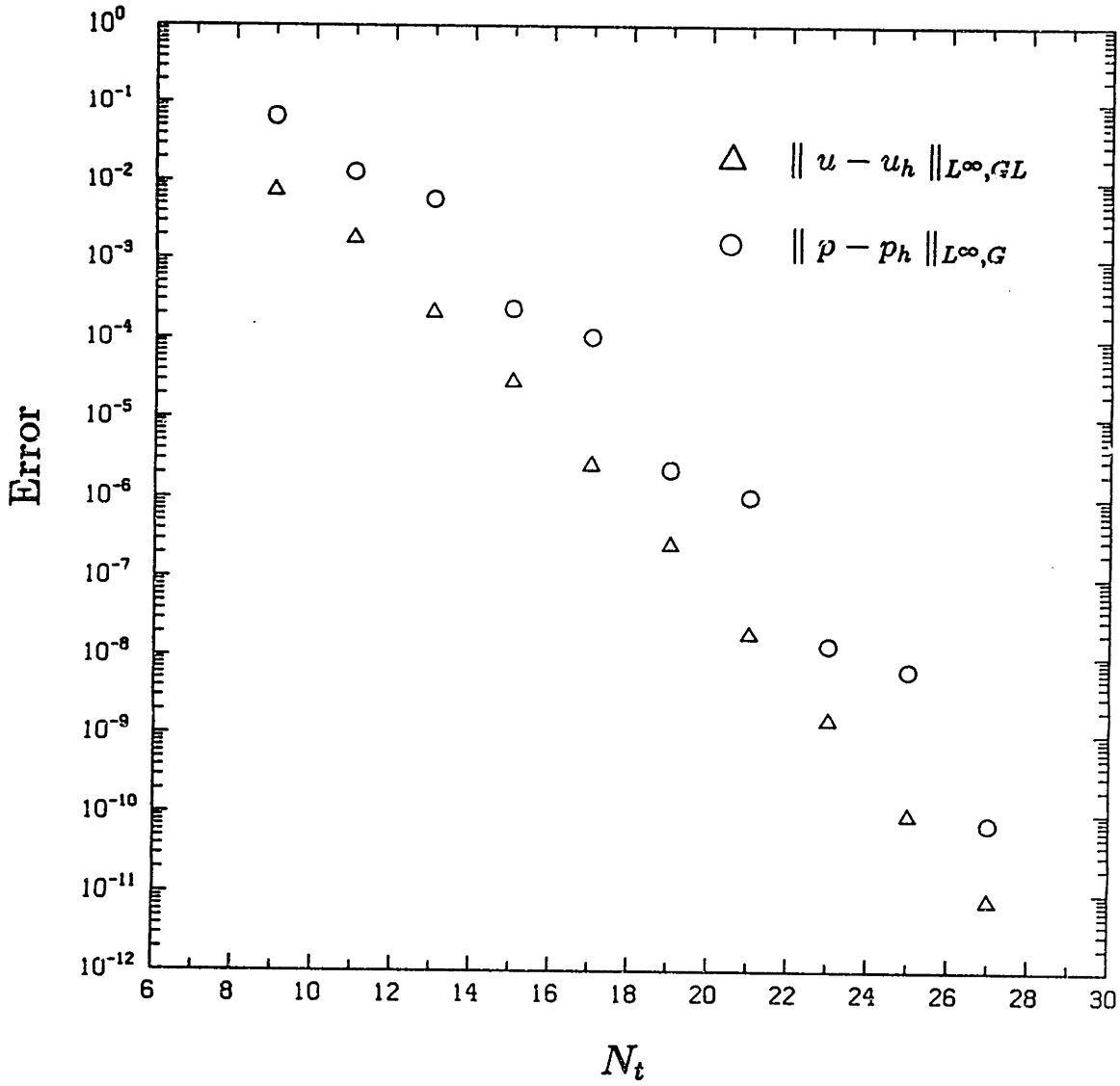


Figure 11. A plot of the maximum pointwise error in the solution to the two-dimensional steady Stokes problem (2.81-2.82) with non-homogeneous boundary conditions as a function of the total number of degrees-of-freedom in one spatial direction, N_t. The problem is defined on the domain $\Omega =]-1,1[^2$ with the exact solution given as $\mathbf{u} = (1-x_2^2, 0)$, and $p = sin\,\pi x \cdot sin\,\pi x_2$. The domain is broken up into $K = 4$ similar spectral elements. Exponential convergence is achieved as the degree N of fixed elements is increased.

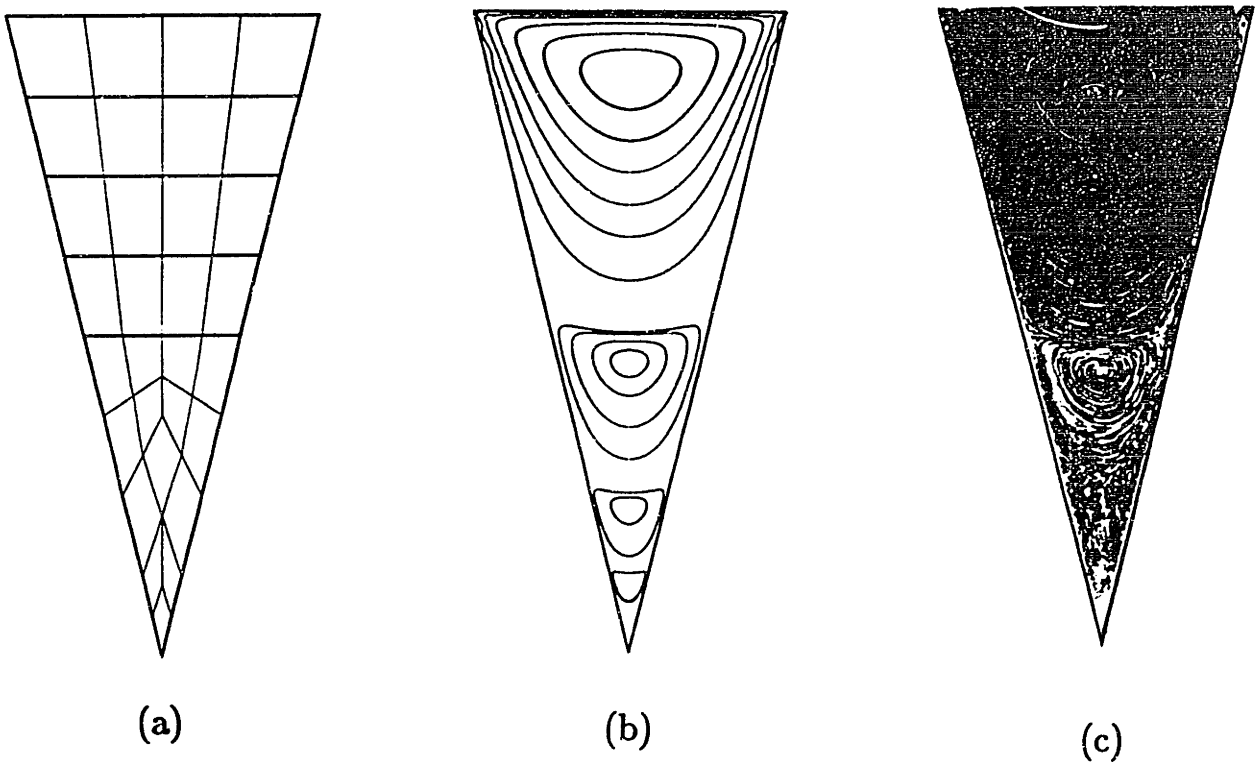


Figure 12. Creeping flow in a wedge. The imposed velocity boundary conditions are no-slip conditions on the two vertical side walls and a unit horizontal velocity on the top side. Fig. 12a shows the spectral element discretization ($K = 30$, $N = 8$), while (b) shows the solution in form of streamlines. For comparison a flow visualization from an experiment by Taneda [59] is included in (c).

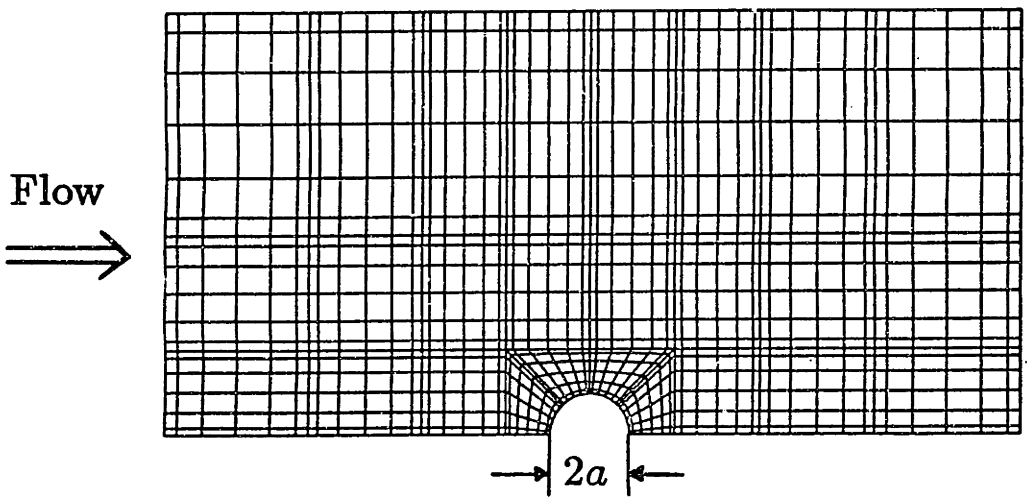


Figure 13. The plot shows a particular spectral element discretization ($K = 26$, $N = 6$) when solving steady Stokes flow past a sphere (axisymmetric). The length of the domain is 30 (axial direction) and the height is 15 (radial direction). The radius of the sphere is $a = \sqrt{2}$.

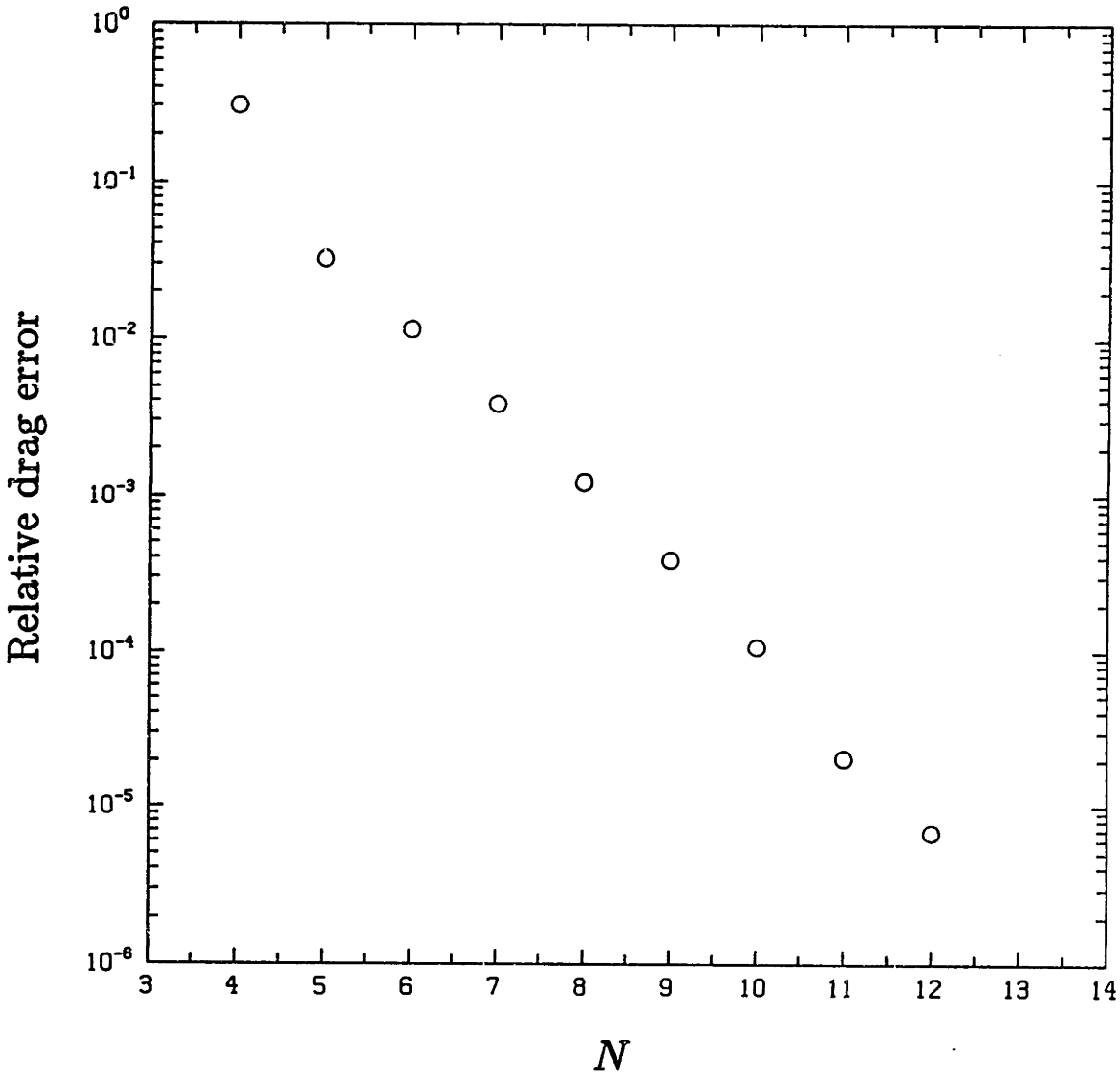


Figure 14. A plot of the relative error in the drag as a function of the polynomial degree N (for fixed $K = 26$) for Stokes flow past a sphere. Exponential convergence is obtained as N is increased.

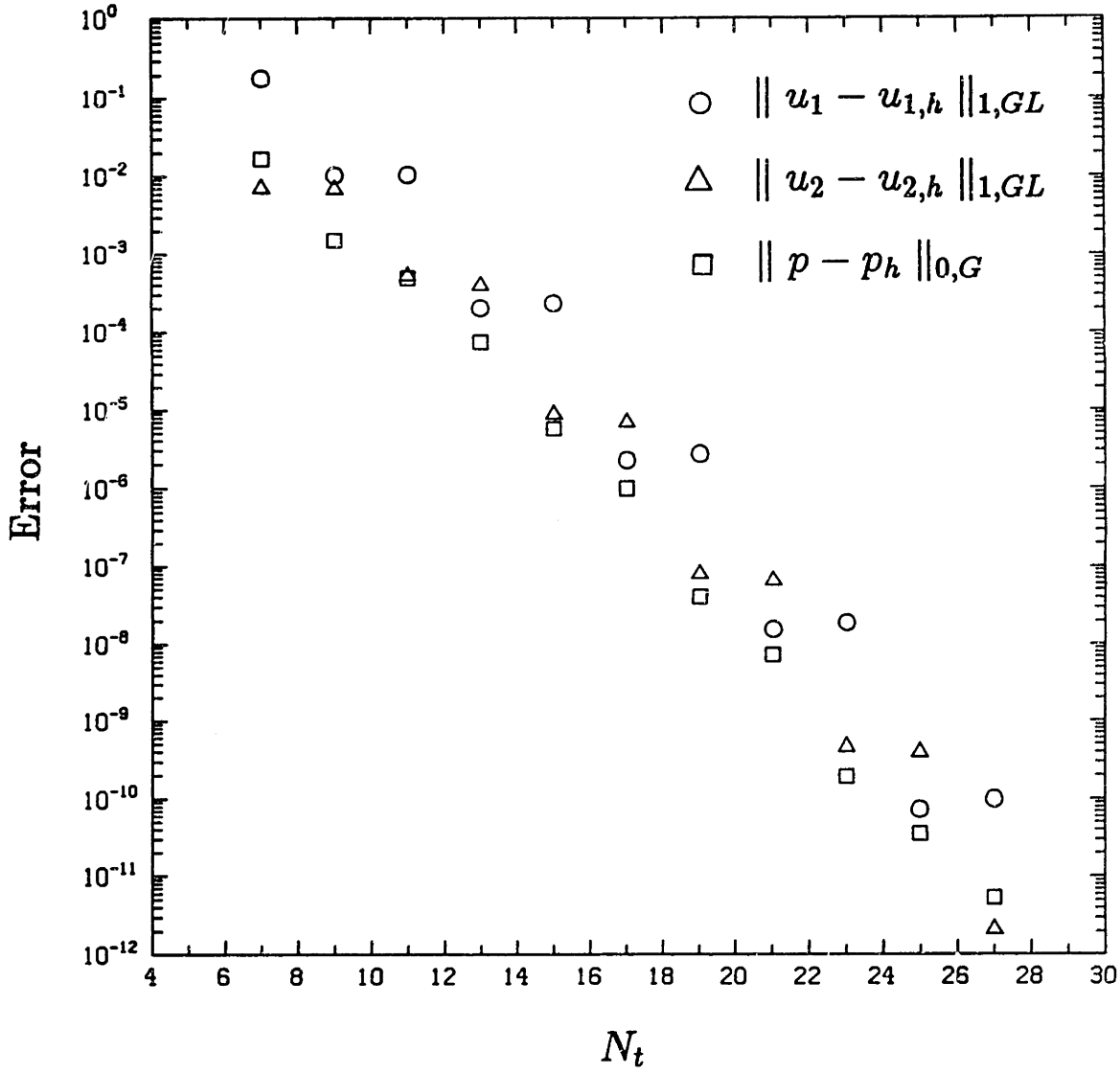


Figure 15. The error in the velocity $\mathbf{u}_h = (u_{1,h}, u_{2,h})$ and the pressure p_h as a function of the total number of degrees-of-freedom (Gauss-Lobatto Legendre points) in the x_1-direction, N_t, when solving the test problem (2.115-2.117). The total interval $\Lambda =]-1,1[$ is divided into $K = 2$ spectral elements $\Lambda_1 =]-1,0[$ and $\Lambda_2 =]0,1[$. Exponential convergence is achieved as the polynomial degree, N, is increased.

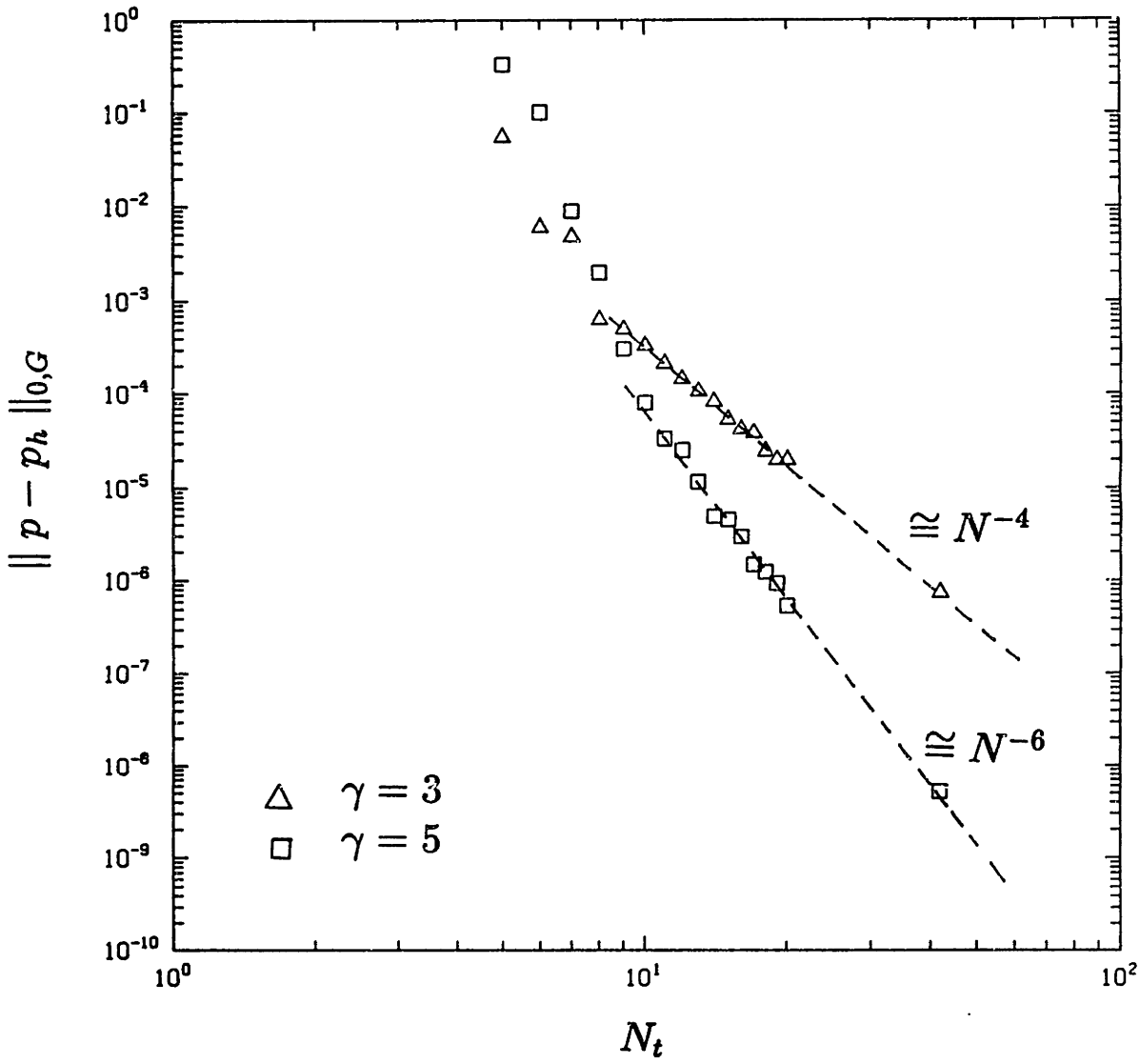


Figure 16.. The error in the pressure p_h as a function of the total number of degrees-of-freedom (Gauss-Lobatto Legendre points) in the x_1-direction, N_t, when solving the spectral element equations corresponding to the test problem (2.118-2.121). The error is given for $\gamma = 3$ ($\triangle$) and for $\gamma = 5$ ($\square$). The total interval $\Lambda =]-1, 1[$ is not divided into subintervals, that is, $K = 1$. Algebraic convergence is achieved asymptotically (the plot is log-log), although for small N significantly faster convergence is observed.

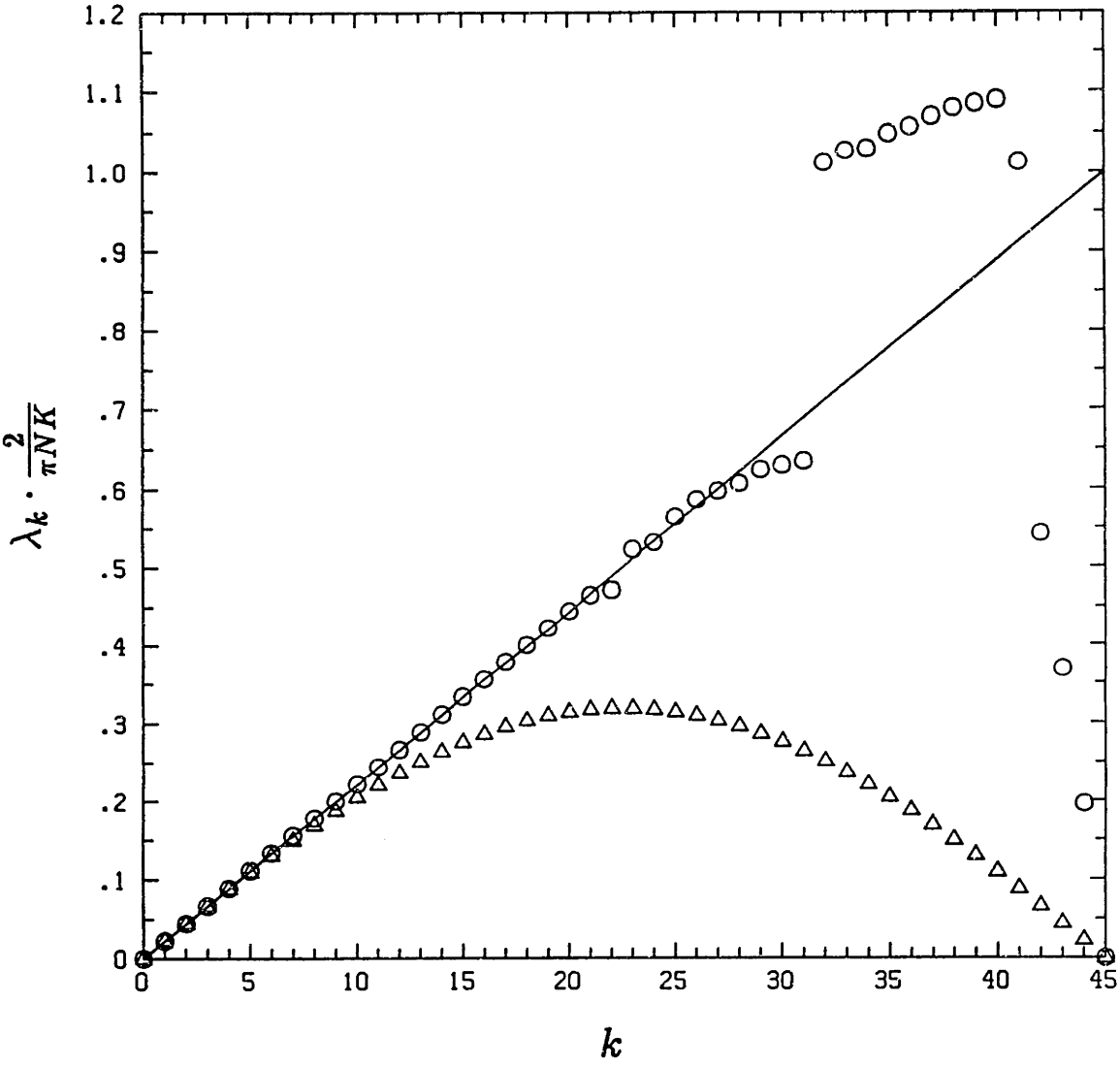


Figure 17. A plot of the spectrum $2\lambda_k/\pi NK$ of the convection operator (2.171) for the spectral element approximation (2.172) with $K = 10$ and $N = 9$ ($\bigcirc$), and for the linear-basis finite element method ($K = 90$, $N = 1$) ($\triangle$). The exact solution is given by the solid line.

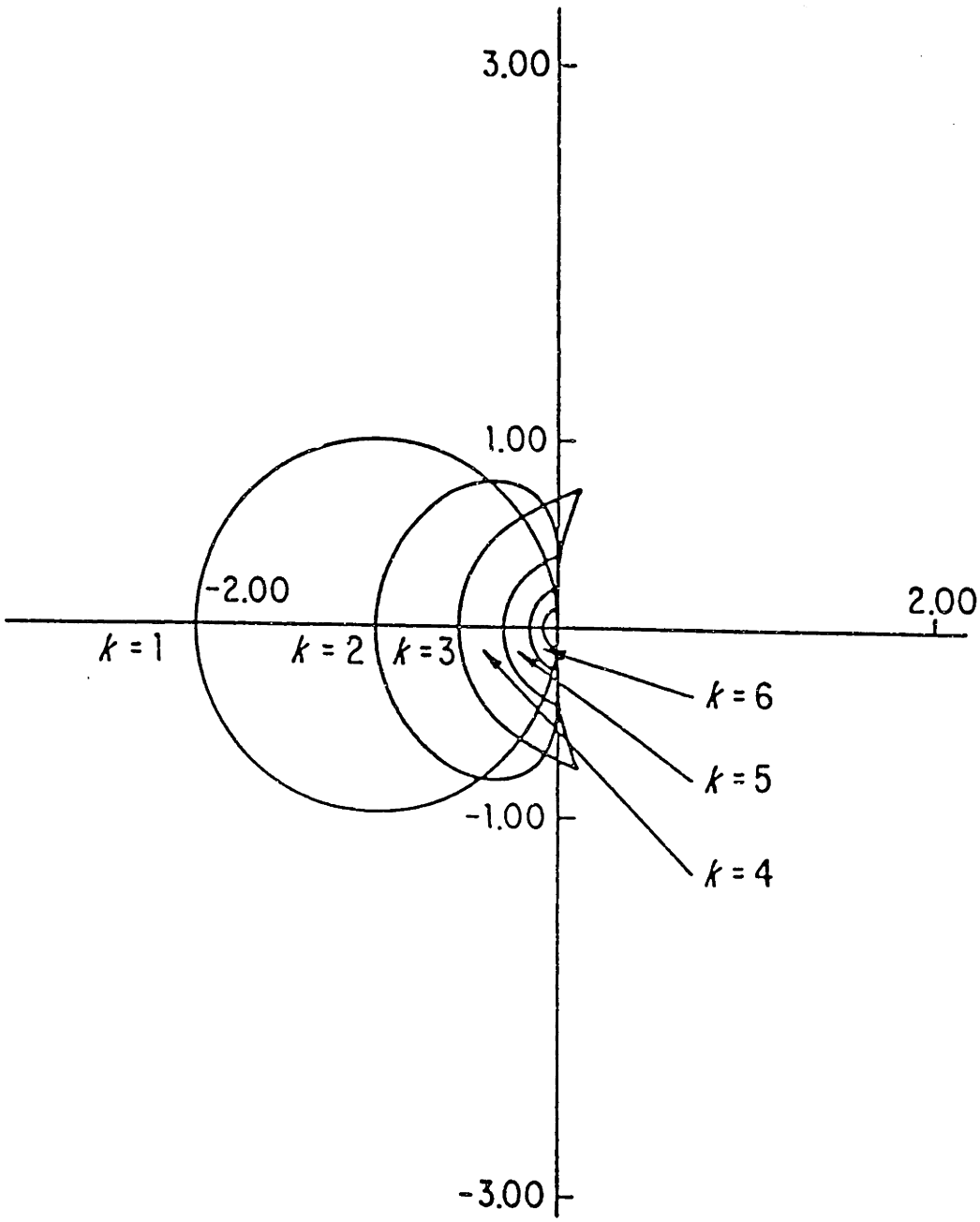


Figure 18. Stability regions for Adam-Bashforth methods. Method of order k is stable inside the indicated region.

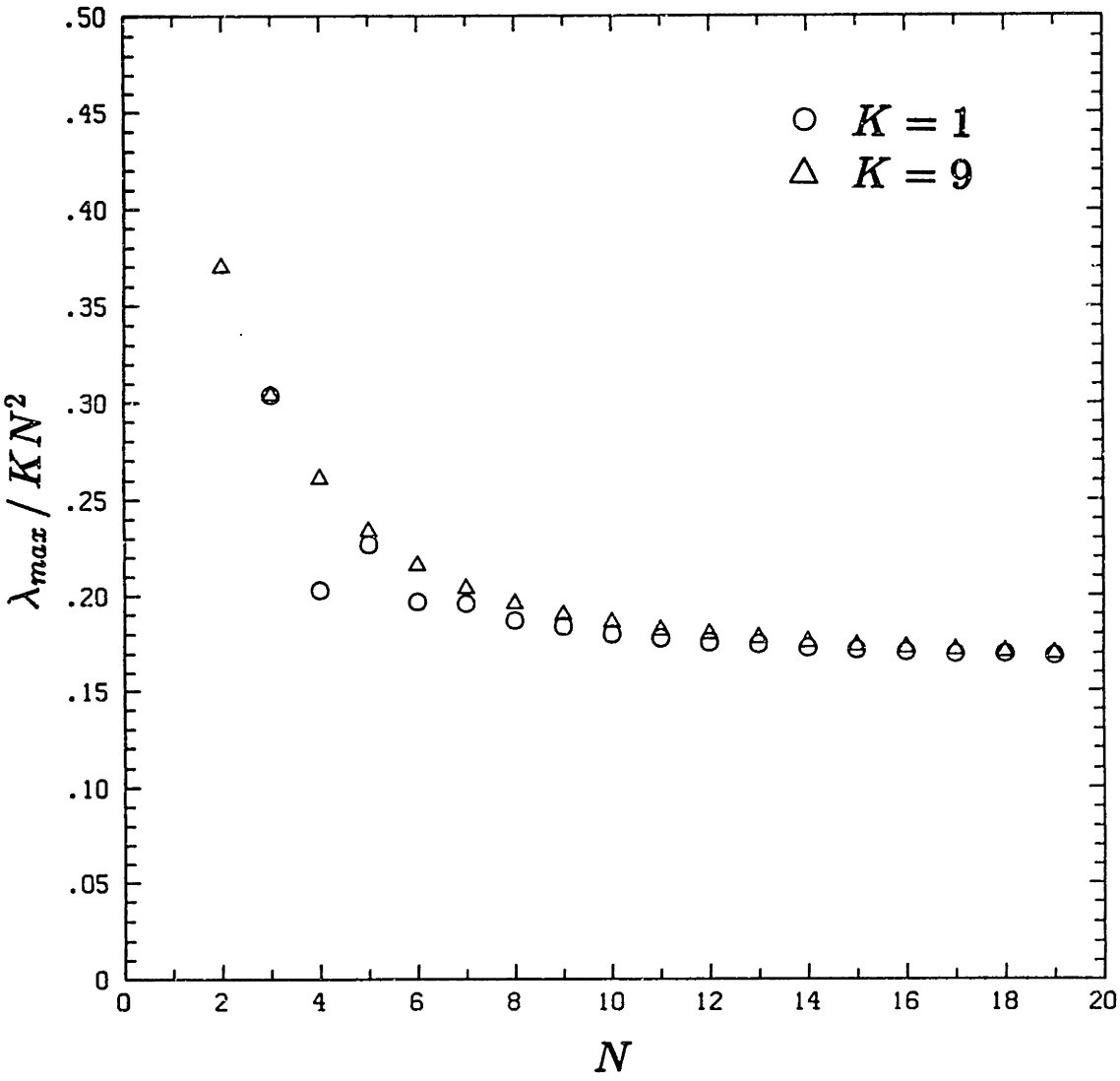


Figure 19. A plot of the maximum eigenvalue $\lambda_{max}(\underline{C})$ of the discrete convection operator $\underline{C}$ in (2.173) for different values of K and N. The maximum eigenvalue is scaled with KN^2.

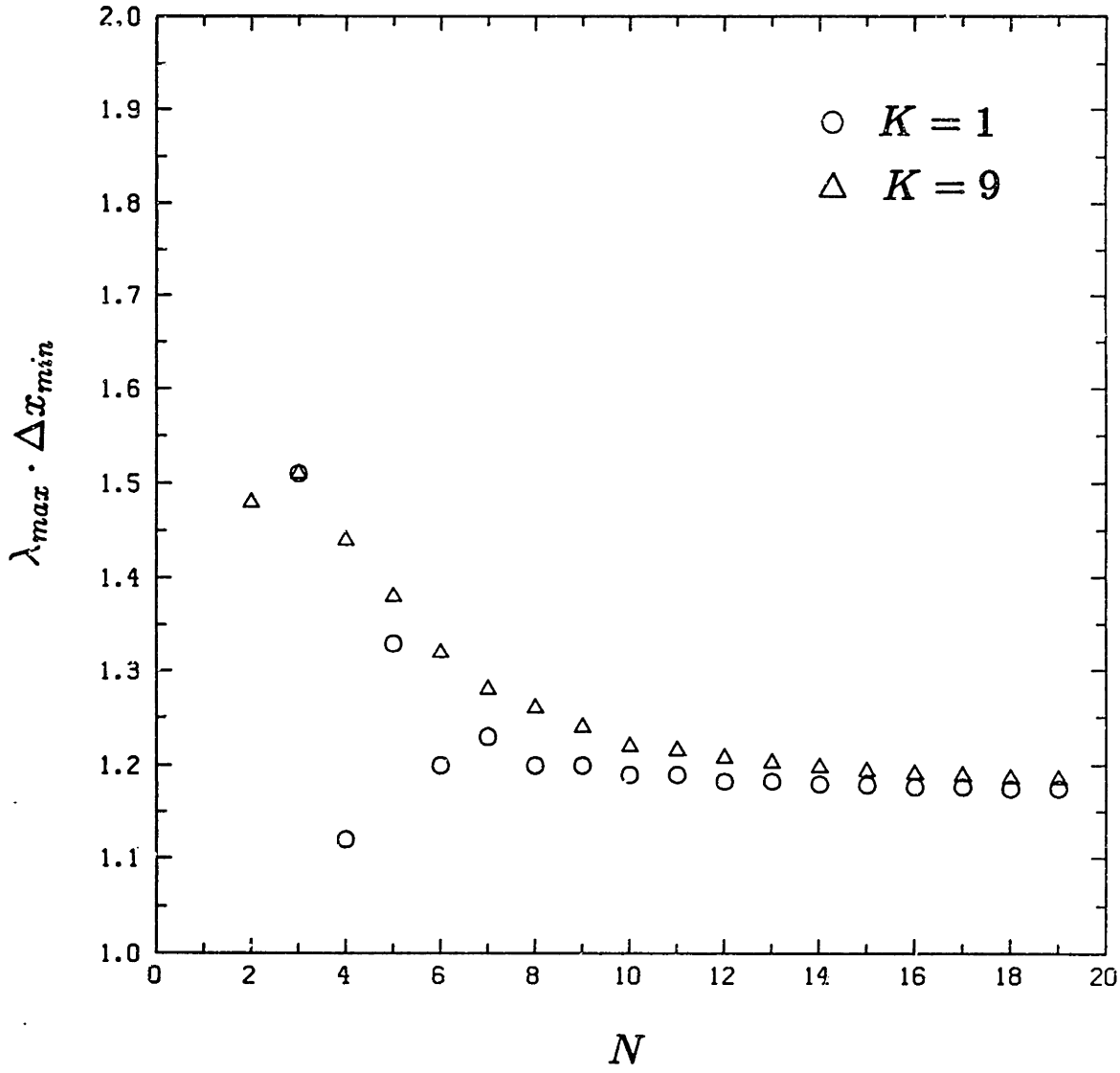


Figure 20. A plot of the product between the maximum eigenvalue of the discrete convection operator, $\lambda_{max}(\underline{C})$, and the minimum mesh spacing on the computational grid, Δx_{min}, for different values of K and N.

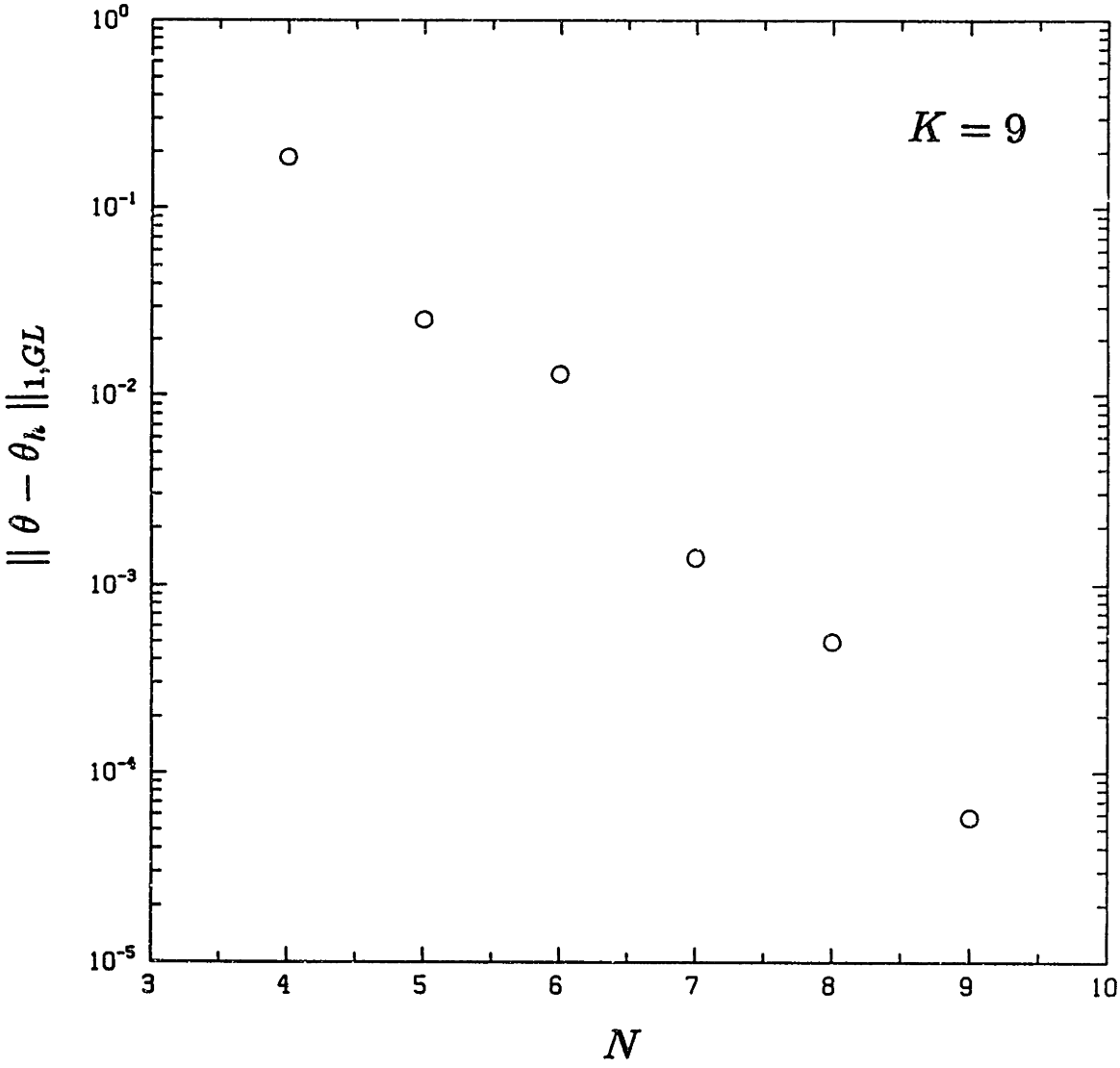


Figure 21. A plot of the spatial discretization error $\| \theta - \theta_0 \|_{1,GL}$ for convection of the two-dimensional Gaussian pulse (2.207), as a function of the polynomial degree, N, for fixed number of elements, $K = 9$. Exponential convergence is achieved as N is increased.

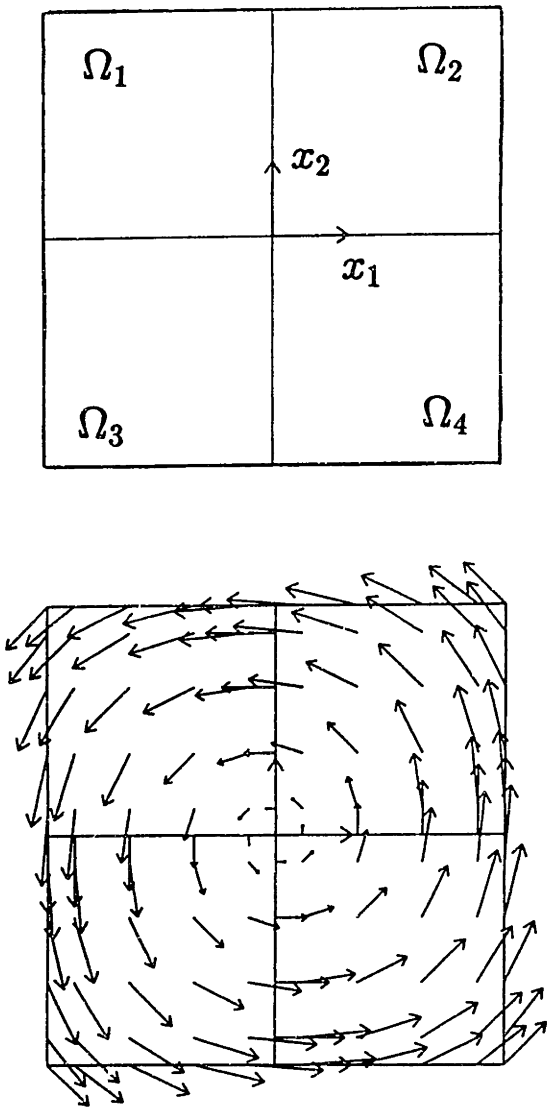


Figure 22. The plot shows the spectral element discretization and a typical solution for the velocity when solving the two-dimensional, unsteady, incompressible Navier-Stokes equations with exact solution given by (2.215-2.216).

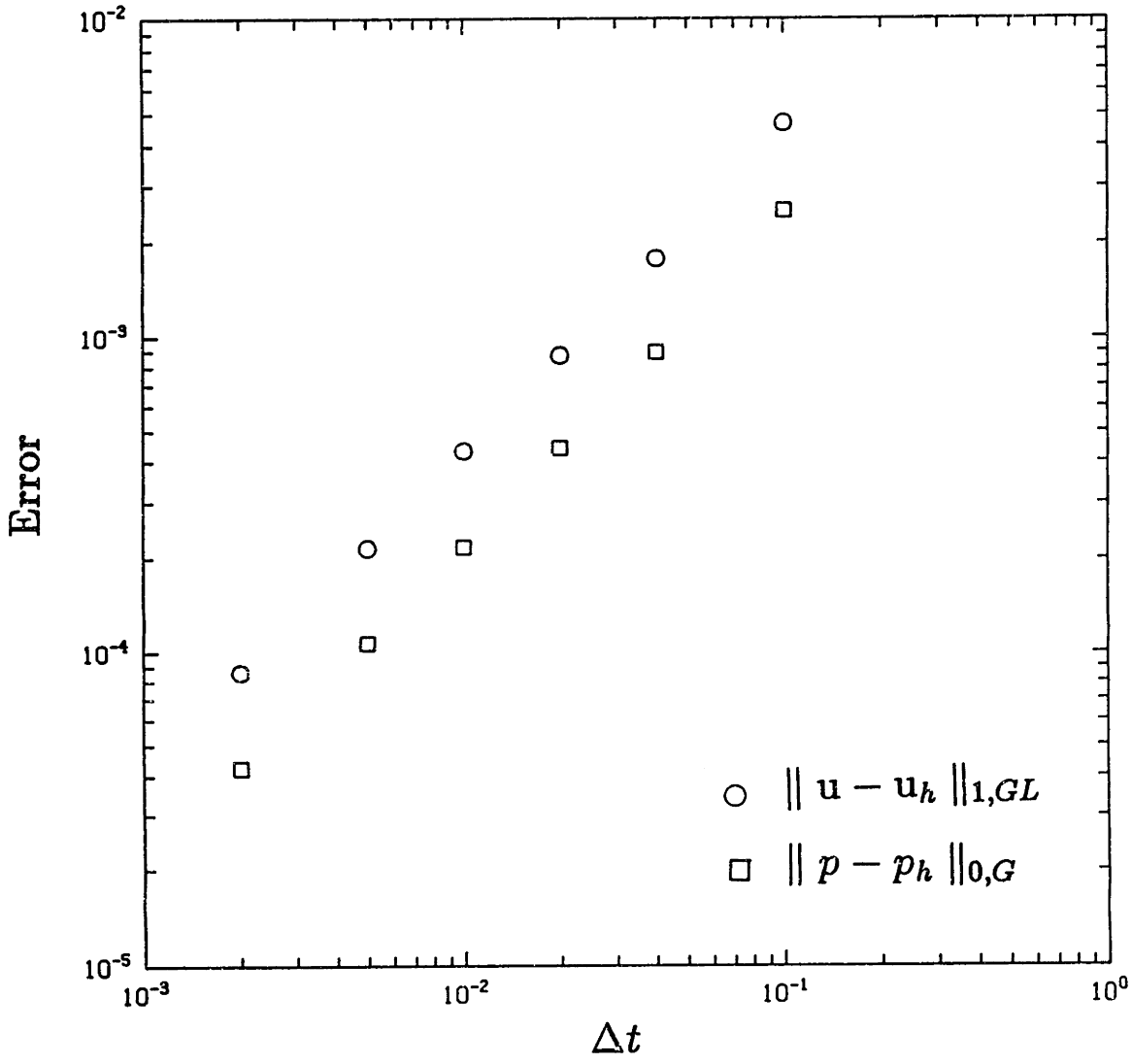


Figure 23. A plot of the discretization errors $\| \mathbf{u} - \mathbf{u}_h \|_1$ (○) and $\| \underline{p} - \underline{p}_h \|_1$ (□) as a function of the time step Δt when solving the two-dimensional, unsteady, incompressible Navier-Stokes equations with exact solution given by (2.215-2.216). Here, the spatial errors are negligible compared to the temporal errors. The results indicate that the scheme is first-order accurate in time.

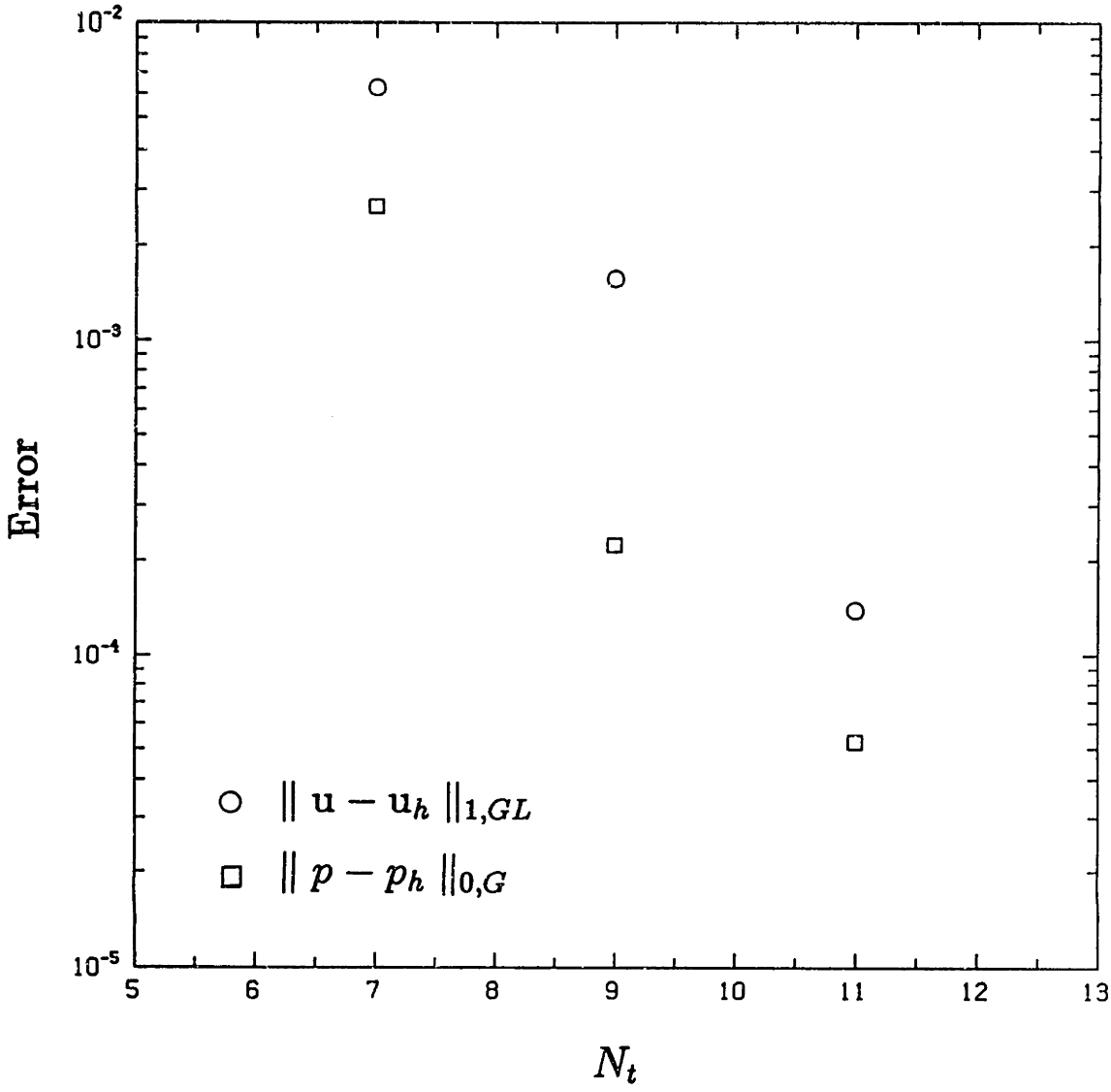


Figure 24. A plot of the discretization errors $\| \mathbf{u} - \mathbf{u}_h \|_1$ (○) and $\| \underline{p} - \underline{p}_h \|_1$ (□) as a function of the total number of degrees-of-freedom, N_t, in one spatial direction when solving the two-dimensional, unsteady, incompressible Navier-Stokes equations with exact solution given by (2.215-2.216). Here the temporal errors are negligible compared to the spatial errors. The results indicate that the scheme gives exponential convergence as the order of the elements, N, is increased.

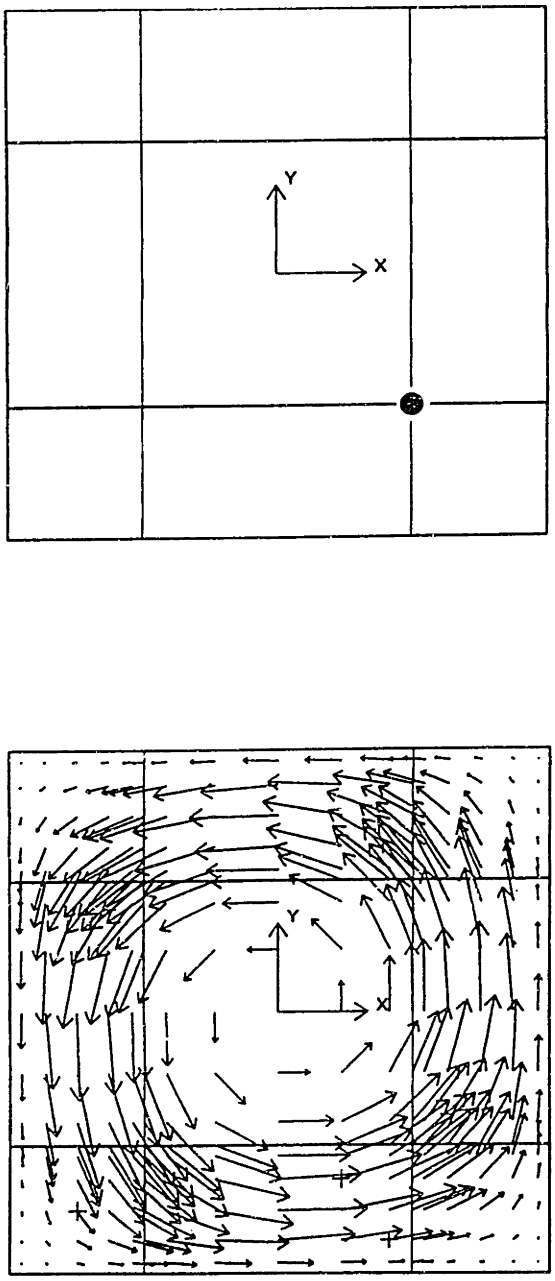


Figure 25. Simulated bouyancy-driven flow in a square cavity. The plot shows the spectral element discretization together with a typical solution for the velocity at time $t = 1$. The error in the first derivative of the velocity is measured at the indicated four-element junction.

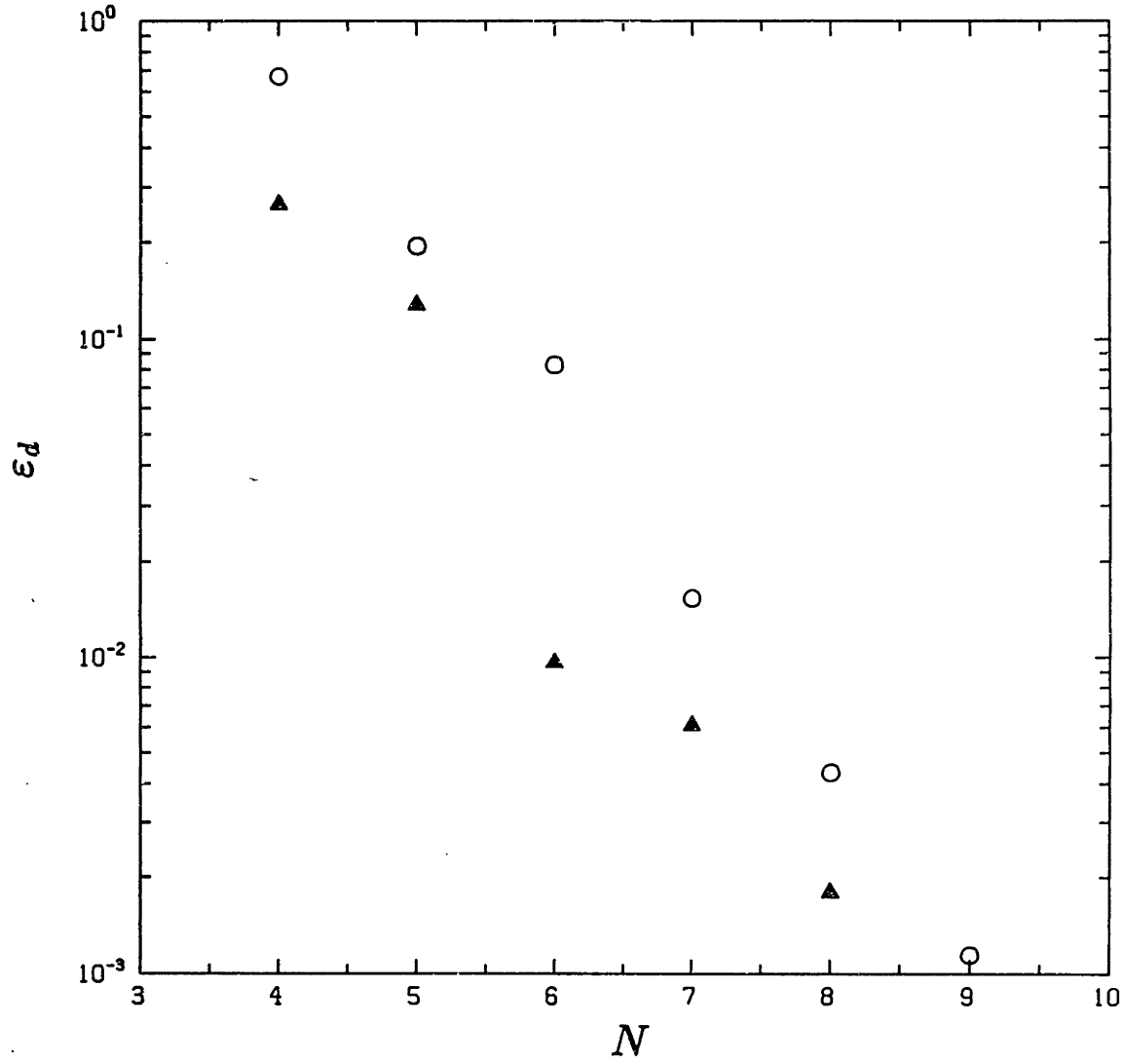


Figure 26. A plot of the error in the first derivative of the velocity, ε_d in (2.217), as a function of the polynomial degree, N, for fixed number of elements, $K = 9$. The results compare the skew-symmetric form of the convection operator (▲) with the rotational form (○). The error is measured at the four-element junction indicated in Fig. 25, and the flow field corresponds to a Reynolds number of about 30.

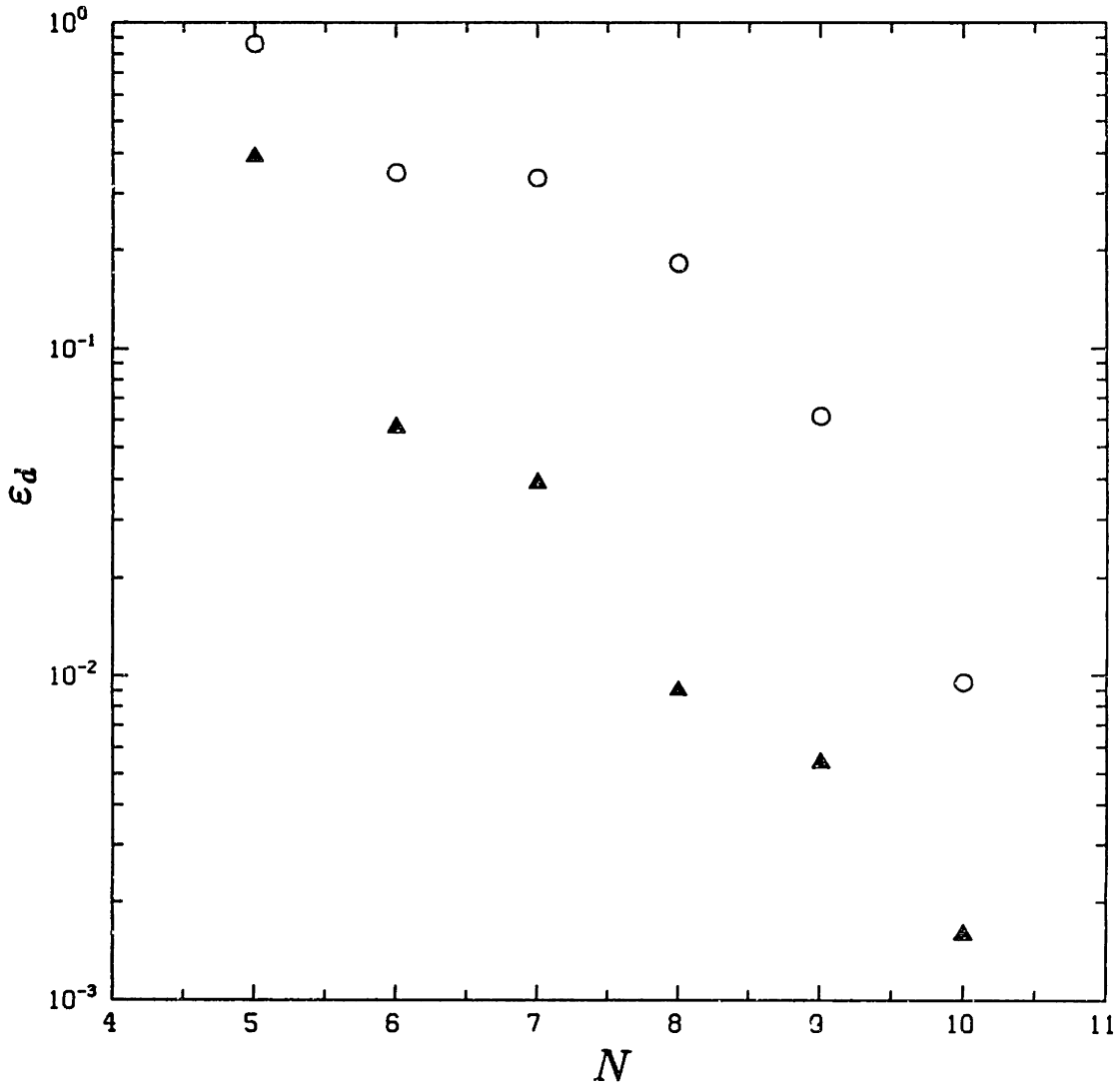


Figure 27. A plot of the error in the first derivative of the velocity, ε_d in (2.217), as a function of the polynomial degree, N, for fixed number of elements, $K = 9$. The results compare the skew-symmetric form of the convection operator (▲) with the rotational form (○). The error is measured at the four-element junction indicated in Fig. 25, and the flow field corresponds to a Reynolds number of about 170.

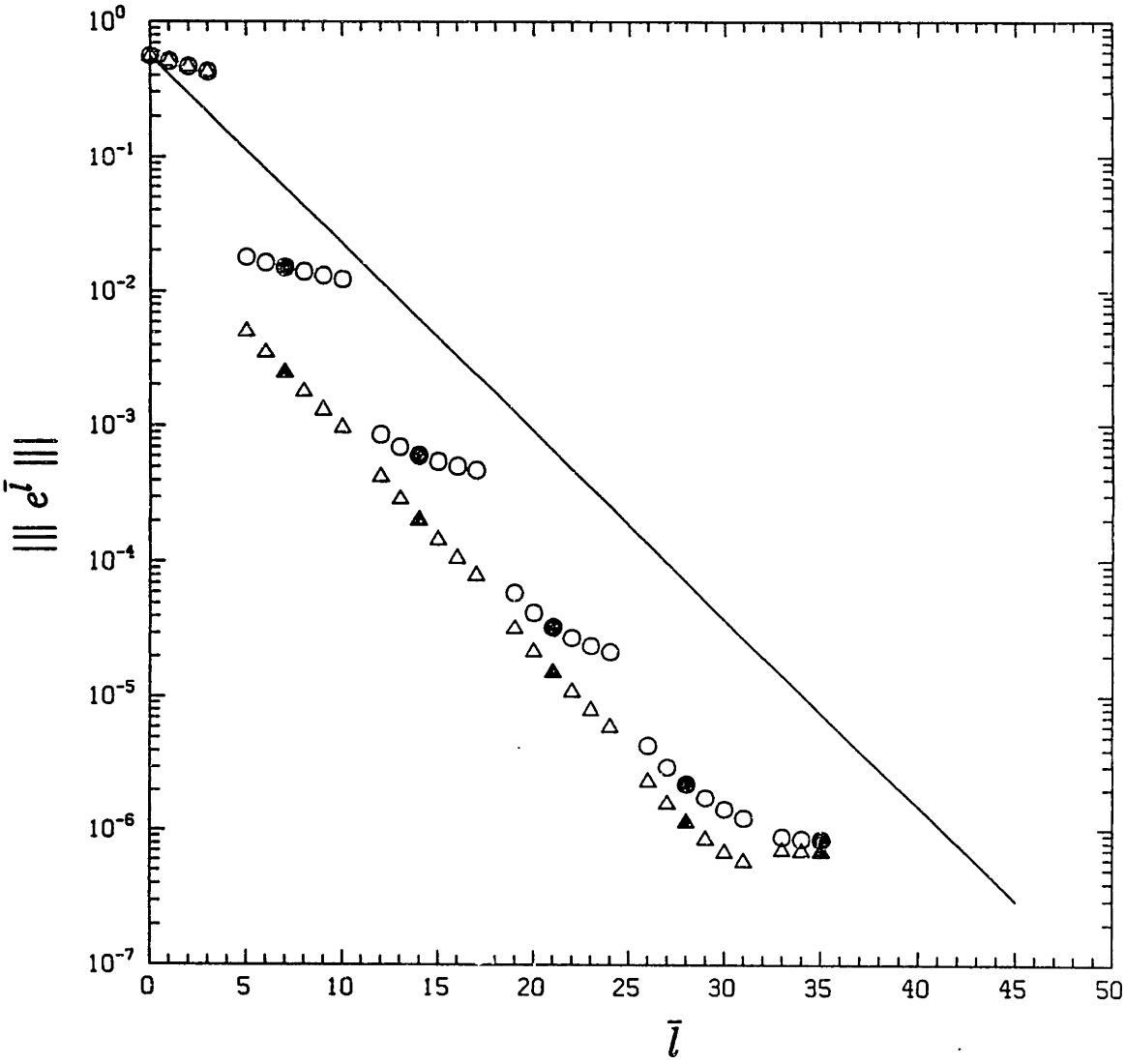


Figure 28. A convergence history plot of $|||\ \underline{e}^{\bar{l}}\ |||$ vs. $\bar{l}$ for the $J = 2$ ($\triangle$) and $J = 3$ ($\bigcirc$) multigrid schemes with $K = 8$, $N_J = 12$, and $m = 3$ smoothings. Here $|||\ \underline{e}^{\bar{l}}\ |||$ is interpreted as the iteration error after each fine-mesh smoothing; the solid symbols ($\blacktriangle$, $\bullet$ for $J = 2$, 3, respectively) indicate the end of each V-cycle. The solid line represents an upper bound on the $J = 2$ iteration error based on $\bar{\rho} = 0.726$.

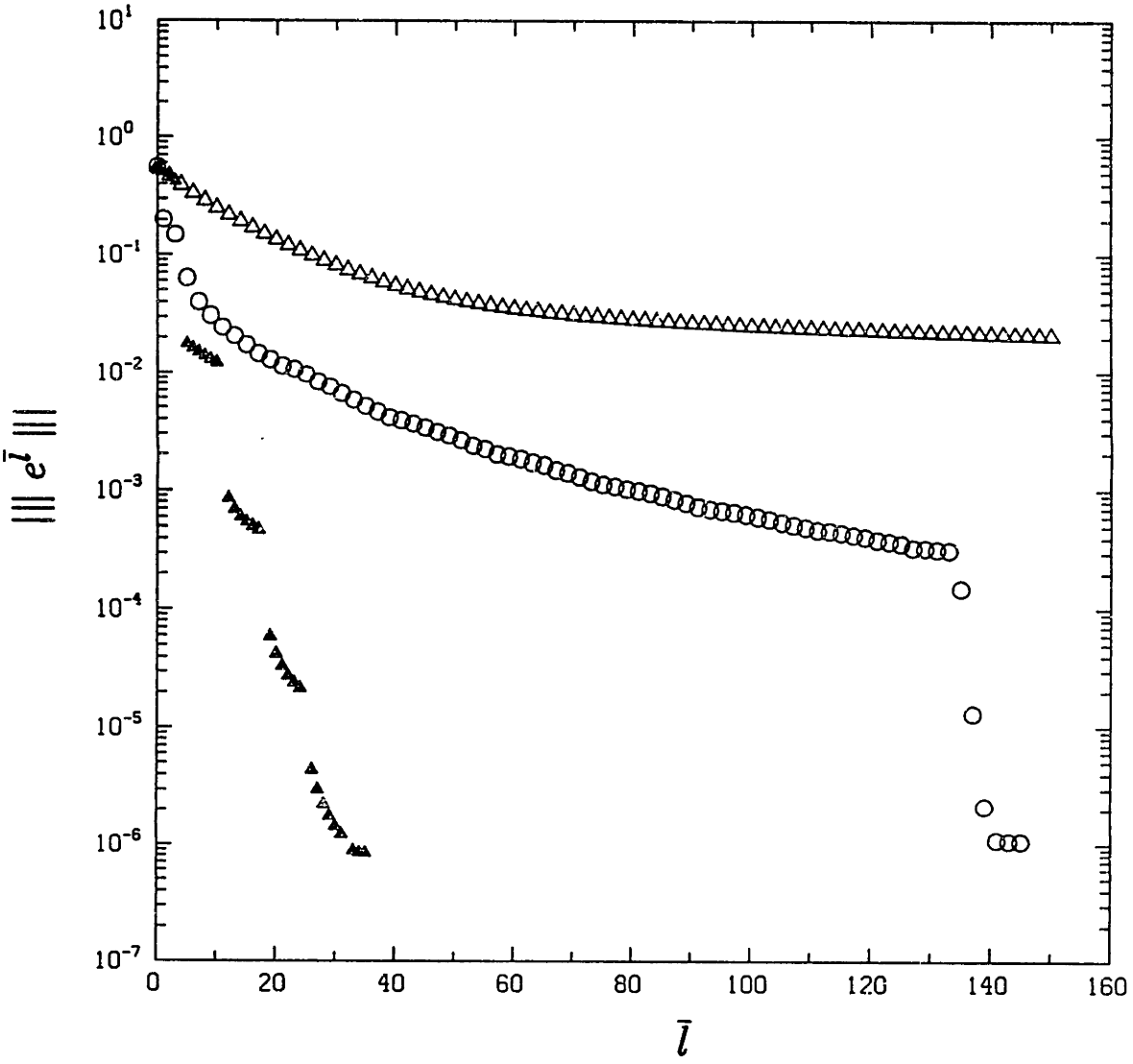
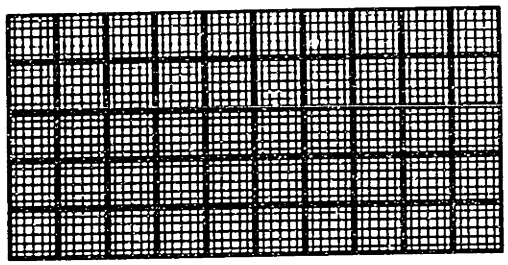
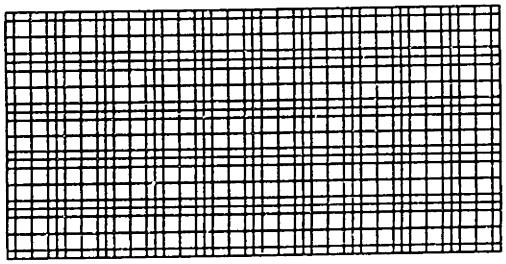


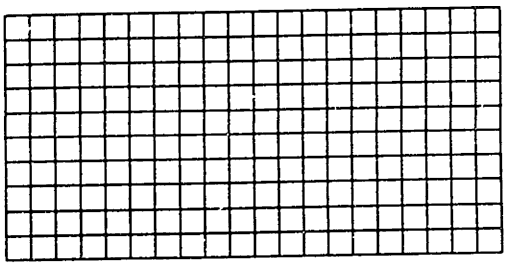
Figure 29. A convergence history plot of $|||\ \underline{e}^{\bar{l}}\ |||$ vs. $\bar{l}$ for the $J = 3$ multigrid scheme with $m = 3$ smoothings (▲), as compared to plain Jacobi iteration (△), and unpreconditioned conjugate gradient iteration (○). The final rapid convergence of the conjugate gradient iteration is due to the fact that all of the modes of the system have been exhausted.



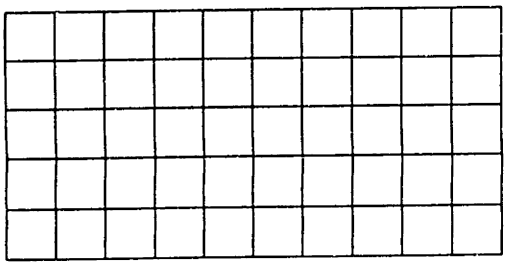
$K = 50, N_4 = 8$



$K = 50, N_3 = 4$



$K = 50, N_2 = 2$



$K = 50, N_1 = 1$

Figure 30. A plot of the $J = 4$ grids used in the spectral element multigrid algorithm when solving the two-dimensional Poisson equation (3.11) with data $f = 0$ and solution $u = 0.5 + x$.

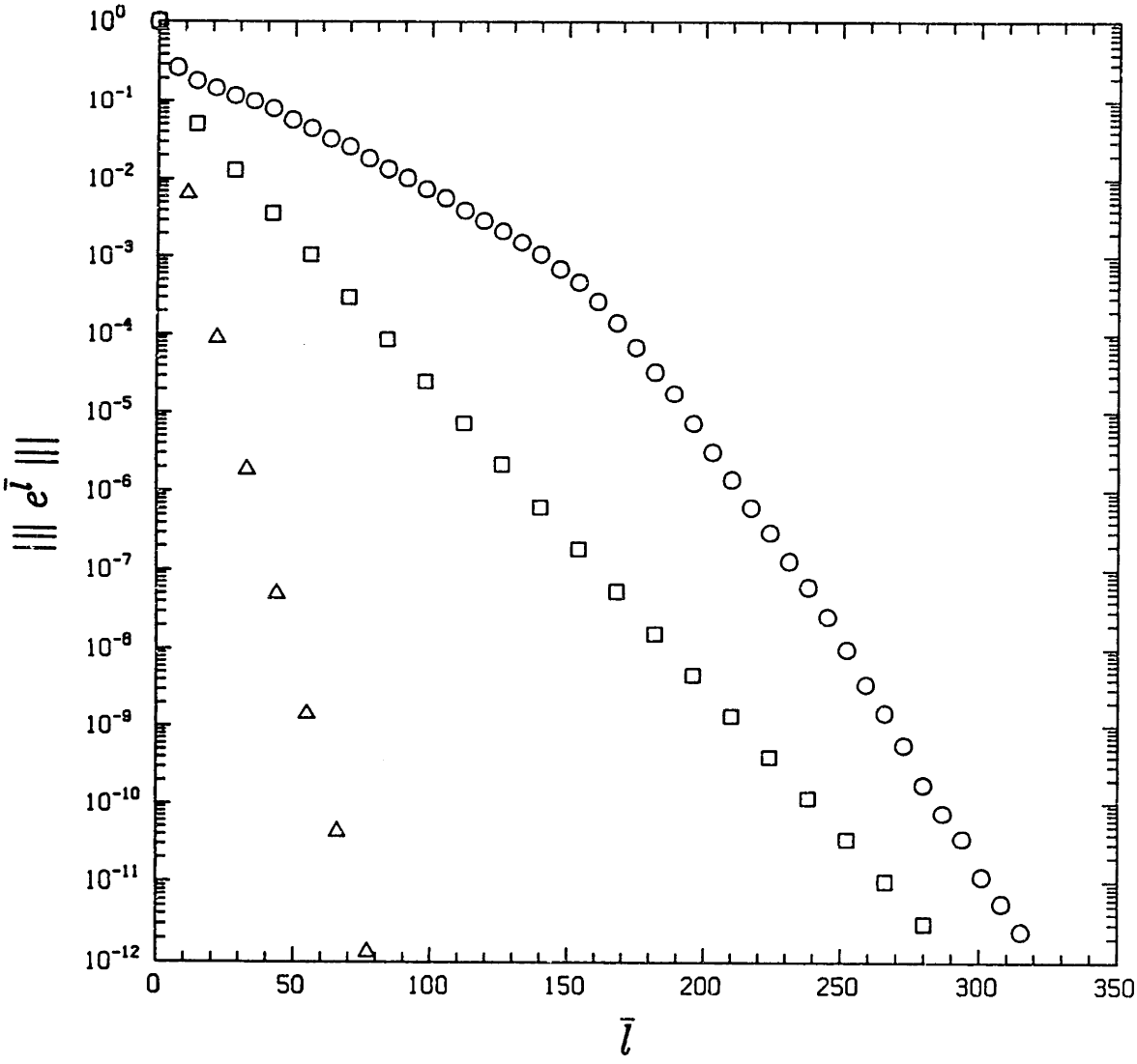


Figure 31. A plot of the iteration error $|||\ \underline{e}^{\bar{l}}\ |||$ as a function of $\bar{l}$, the number of matrix-vector products $\underline{A}\,\underline{u}$ on the fine grid ($K = 50$, $N_J = 8$), when solving the two-dimensional Poisson equation (3.11) with data $f = 0$ and solution $u = 0.5 + x$. The plot compares convergence histories for the ($J = 2$, $m = 5$) spectral element multigrid with Chebyshev acceleration (△), multigrid without Chebyshev acceleration (□), and preconditioned conjugate gradient iteration (○).

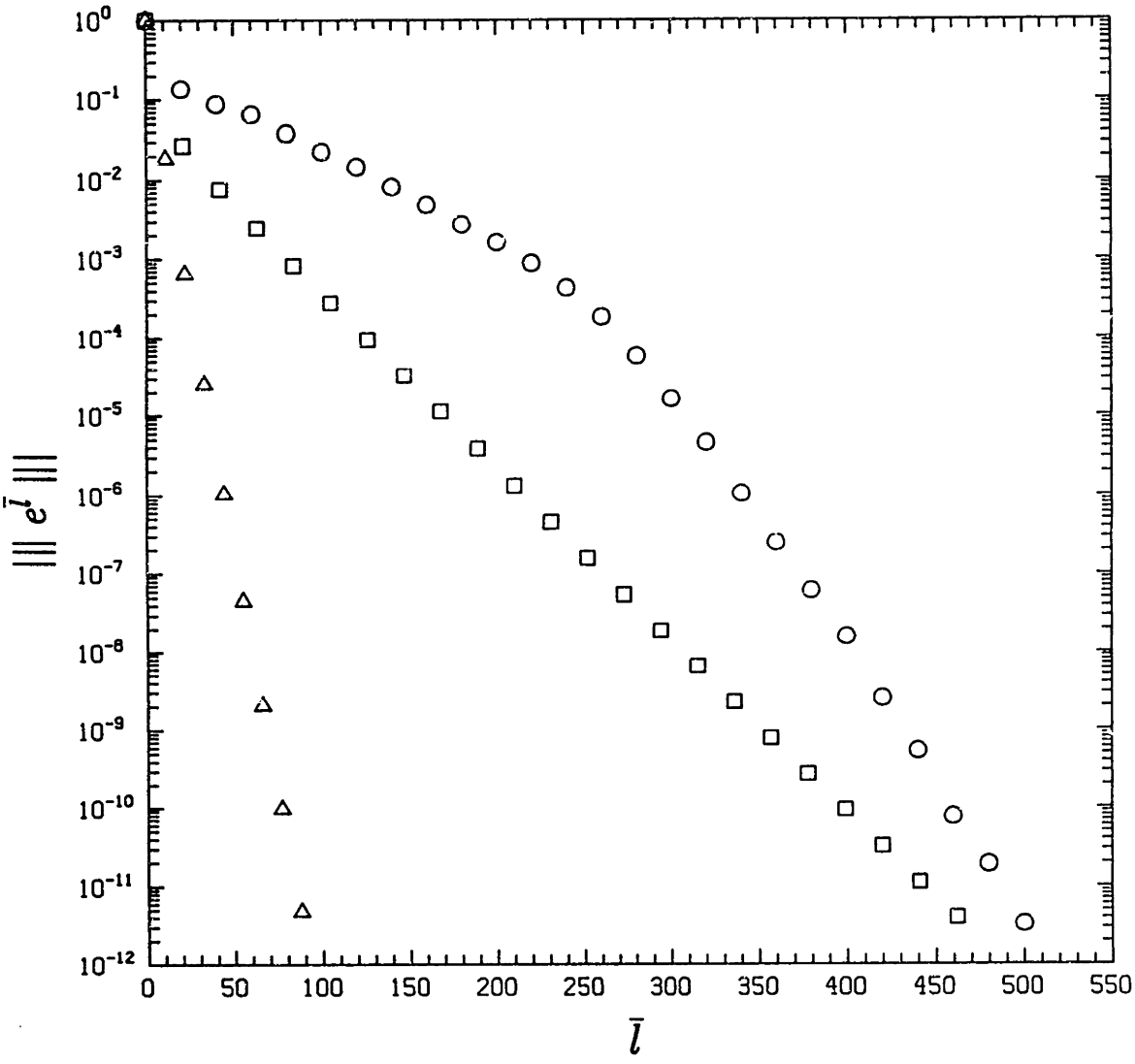


Figure 32. A plot of the iteration error $|||\ \underline{e}^{\bar{l}}\ |||$ as a function of $\bar{l}$, the number of matrix-vector products $\underline{A}\,\underline{u}$ on the fine grid ($K = 50$, $N_J = 12$), when solving the two-dimensional Poisson equation (3.11) with data $f = 0$ and solution $u = 0.5 + x$. The plot compares convergence histories for the ($J = 2$, $m = 5$) spectral element multigrid with Chebyshev acceleration (△), multigrid without Chebyshev acceleration (□), and preconditioned conjugate gradient iteration (○).

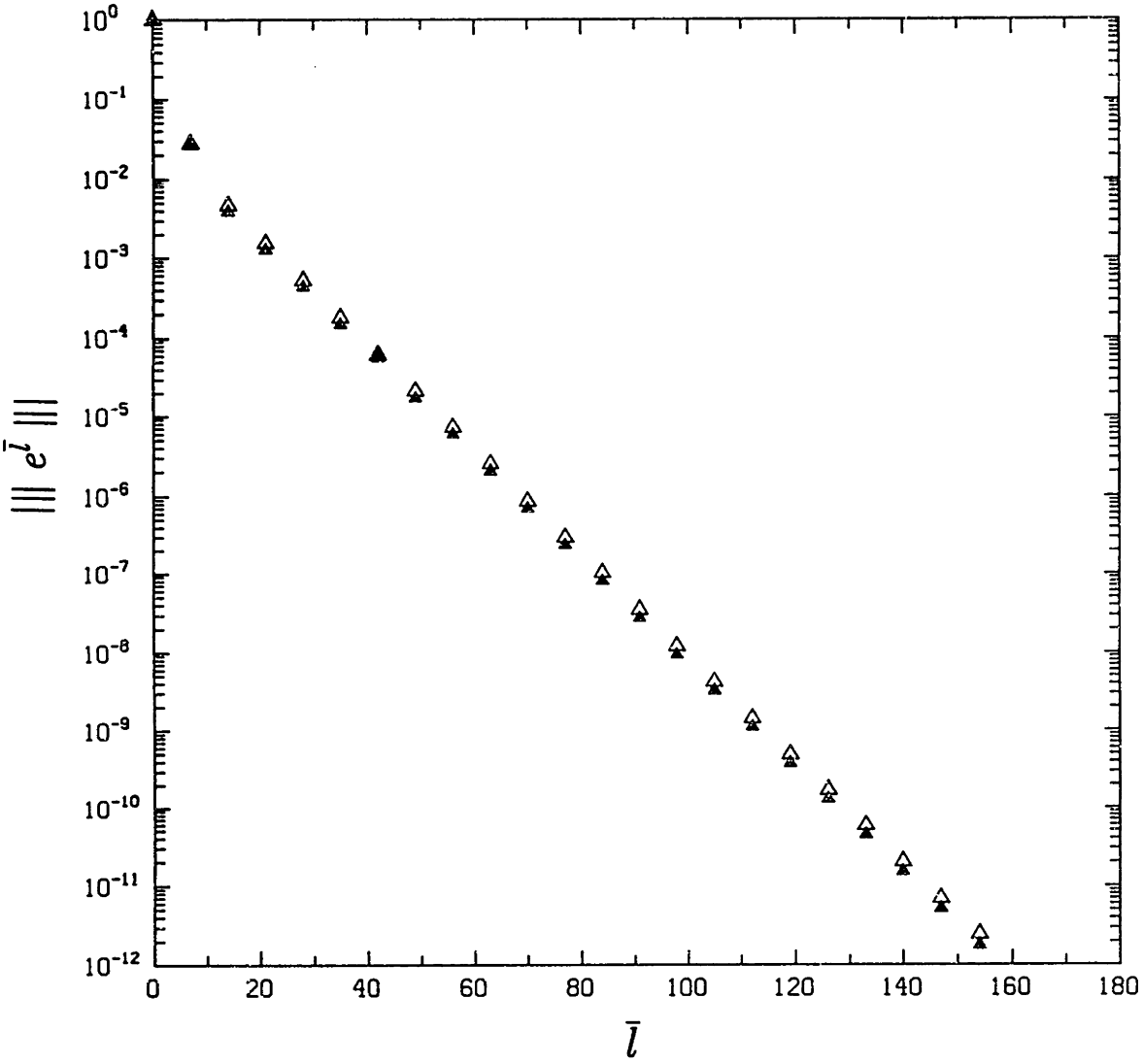


Figure 33. A plot of the iteration error $|||\ \underline{e}^{\bar{l}}\ |||$ as a function of $\bar{l}$, the number of matrix-vector products $\underline{A}\,\underline{u}$ on the fine grid, when solving the two-dimensional Poisson equation (3.11) with data $f = 0$ and solution $u = 0.5 + x$. The plot compares convergence histories for the ($J = 2$, $m = 5$) spectral element multigrid with (non-optimal) Chebyshev acceleration for two different values of K, $K = 8$ (▲) and $K = 50$ (△), with fixed $N_J = 8$.

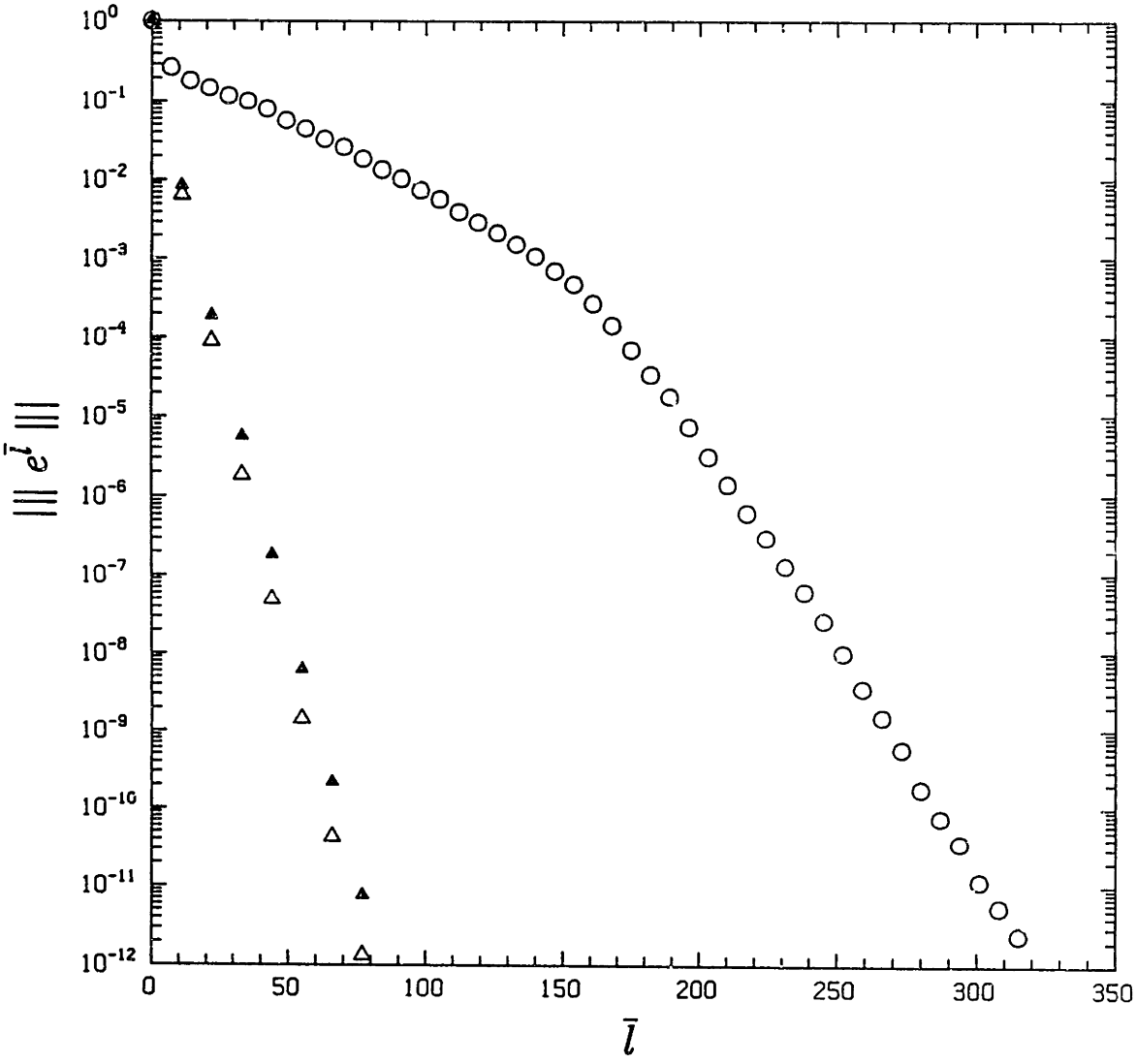


Figure 34. A plot of the iteration error $|||\ \underline{e}^{\bar{l}}\ |||$ as a function of $\bar{l}$, the number of matrix-vector products $\underline{A}\,\underline{u}$ on the fine grid, when solving the two-dimensional Poisson equation (3.11) with data $f = 0$ and solution $u = 0.5 + x$. The plot compares convergence histories for spectral element multigrid with Chebyshev acceleration when using $J = 2$ grids (△) and $J = 4$ grids (▲). The convergence history for preconditioned conjugate gradient iteration is also indicated (○).

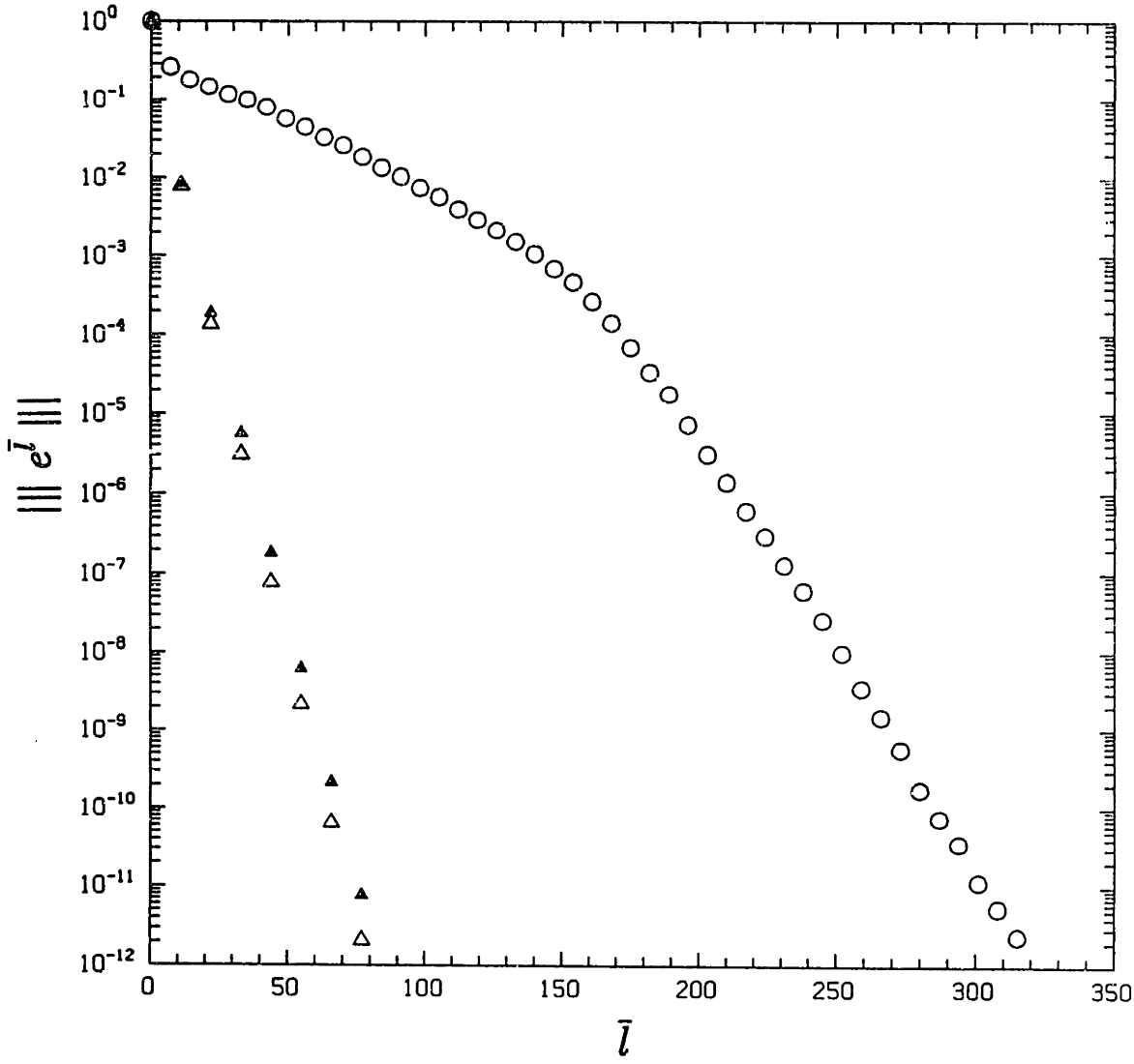


Figure 35. A plot of the iteration error $|||\ \underline{e}^{\bar{l}}\ |||$ as a function of $\bar{l}$, the number of matrix-vector products $\underline{A}\,\underline{u}$ on the fine grid, when solving the two-dimensional Poisson equation (3.11) with data $f = 0$ and solution $u = 0.5 + x$. The plot compares convergence histories for spectral element multigrid with Chebyshev acceleration ($J = 4$, $m = 5$) when imposing Dirichlet boundary conditions (▲) and mixed Dirichlet/Neumann boundary conditions (△). The convergence history for preconditioned conjugate gradient iteration is also indicated (○).

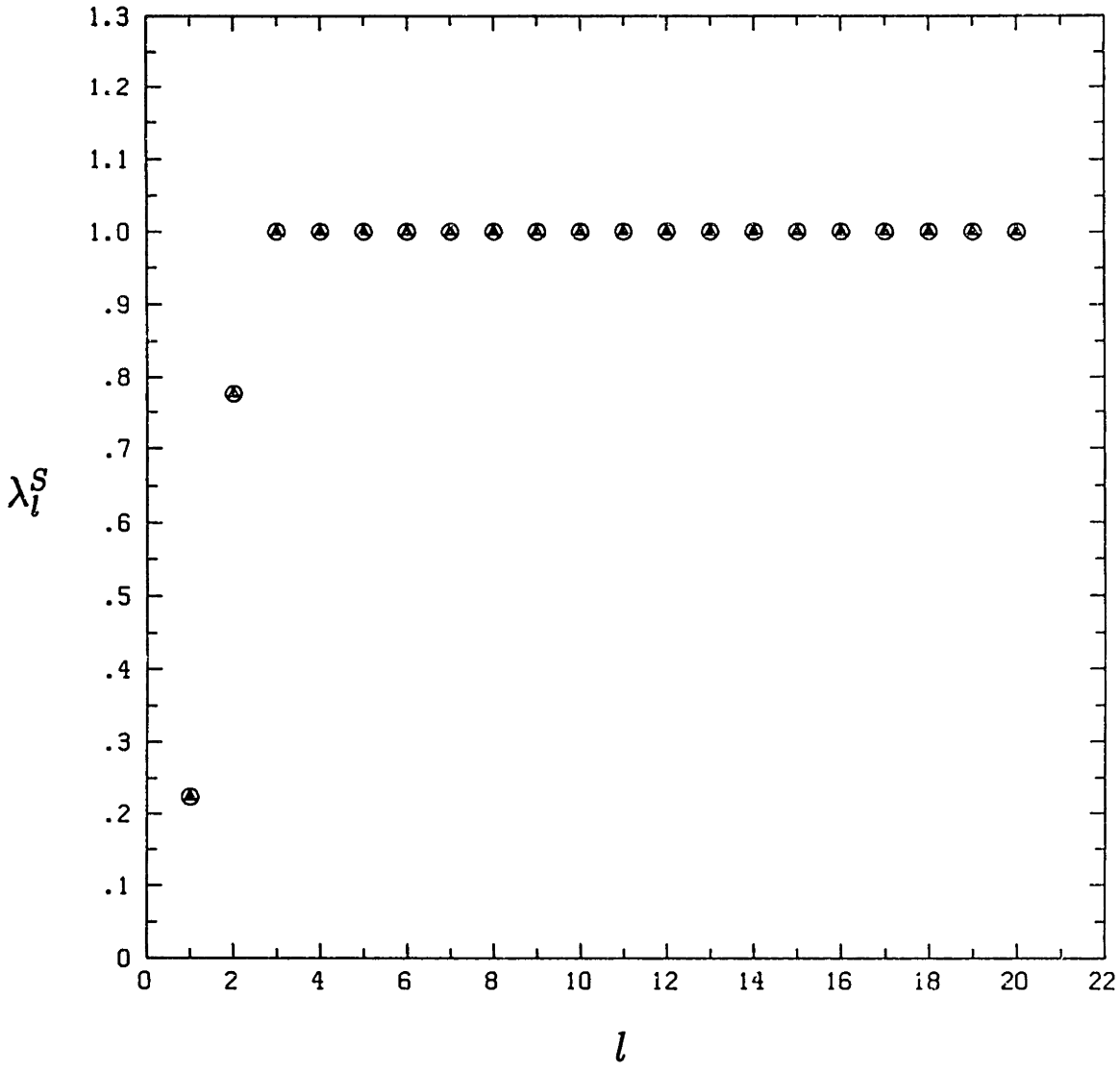


Figure 36. A plot of the spectrum $\lambda_l^S(k)$ of the preconditioned steady Stokes pressure matrix $\underline{\tilde{B}}^{-1}\underline{S}$, where $\underline{S}$ is the pressure matrix given in (3.76) and $\underline{\tilde{B}}$ is the mass matrix defined on the Gauss pressure mesh. The spectrum (▲) corresponds to a spectral element discretization ($K = 4$, $N = 7$) for a wavenumber $k = 1$; the agreement with the continuous operator spectrum λ_l^S of (3.93-3.95) (○) is very good.

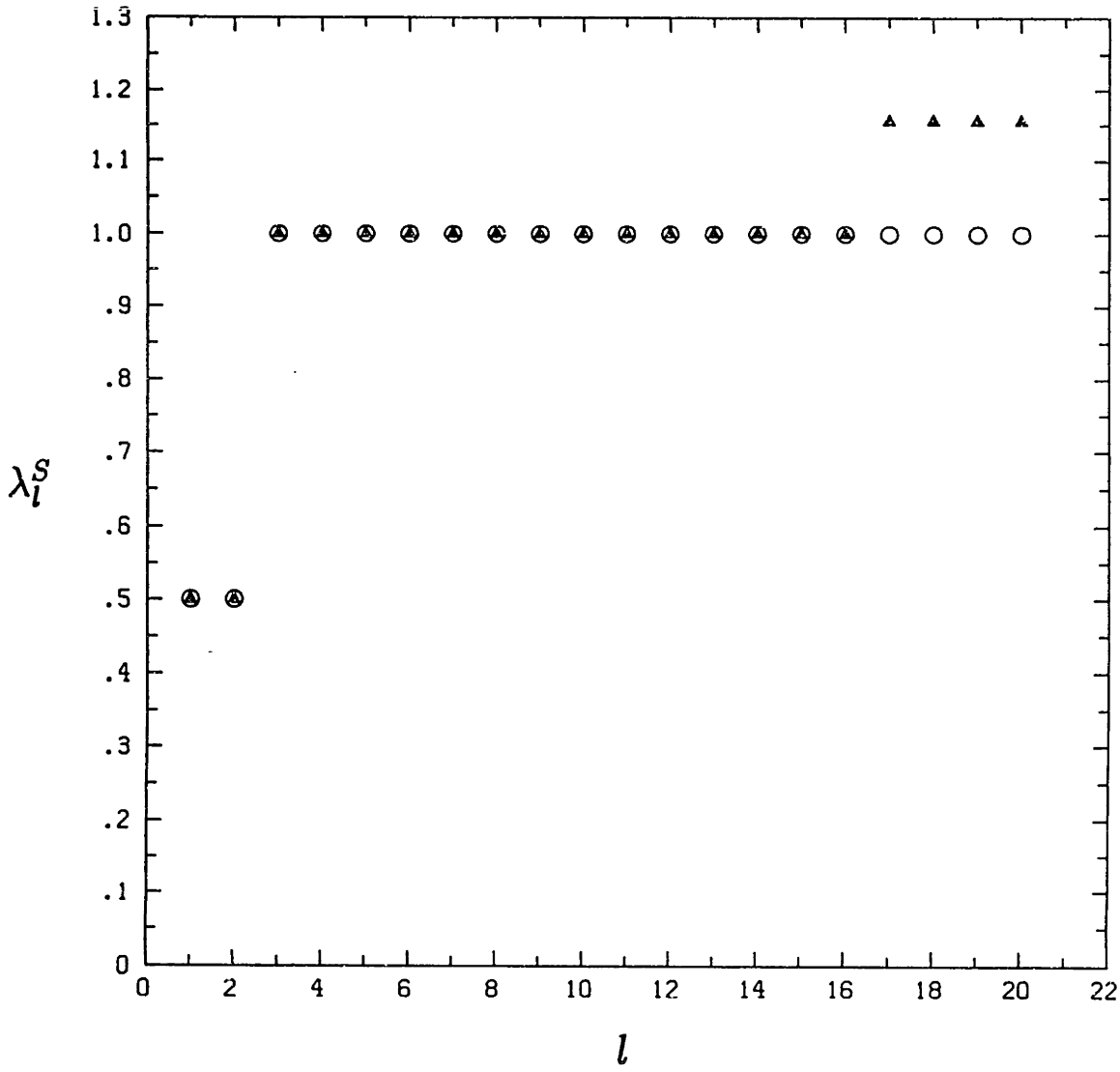


Figure 37. A plot of the spectrum $\lambda_l^S(k)$ of the preconditioned pressure matrix $\underline{\tilde{B}}^{-1}\underline{S}$. The spectrum (▲) corresponds to a spectral element discretization ($K = 4$, $N = 7$) for a wavenumber $k = 12$; for this large value of k the discrete system can no longer resolve the higher continuous modes (○).

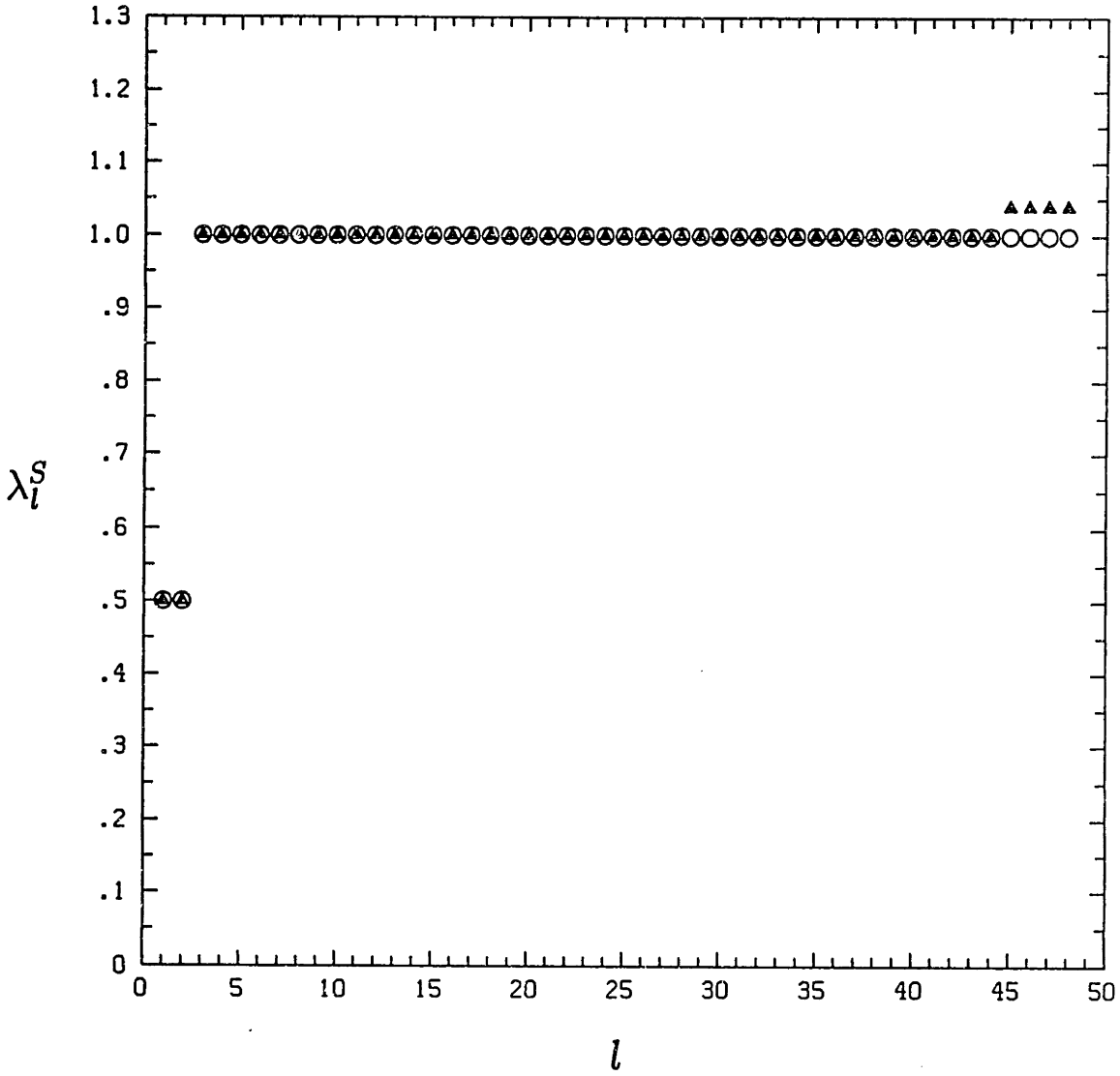


Figure 38. A plot of the spectrum $\lambda_l^S(k)$ of the preconditioned pressure matrix $\underline{\tilde{B}}^{-1}\underline{S}$. The spectrum (▲) corresponds to a spectral element discretization ($K = 4$, $N = 14$) for a wavenumber $k = 12$; due to the higher spatial resolution the agreement of the discrete spectrum with the continuous operator spectrum (○) is improved.

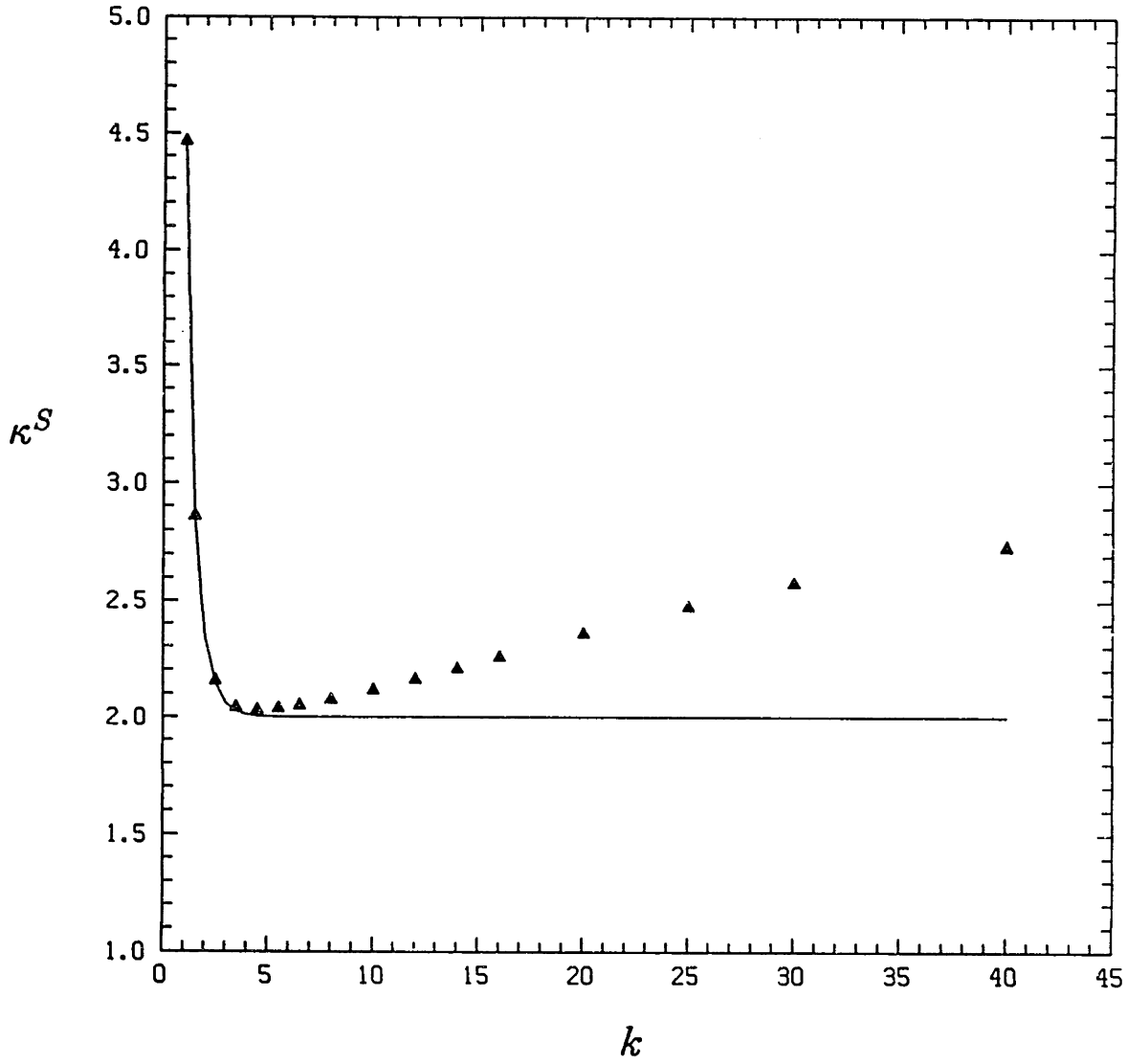


Figure 39. A plot of the condition number $\kappa^S(k)$ for the spectral element operator $\underline{\tilde{B}}^{-1}\underline{S}$ (▲) and the continuous operator κ^S of (3.93-3.95), as a function of Fourier wavenumber k. The spectral element discretization $K = 4$, $N = 10$ is used. For small and moderate k the two curves coincide, however as $k \Rightarrow \infty$ the two results diverge due to the finite resolution of the spectral element mesh.

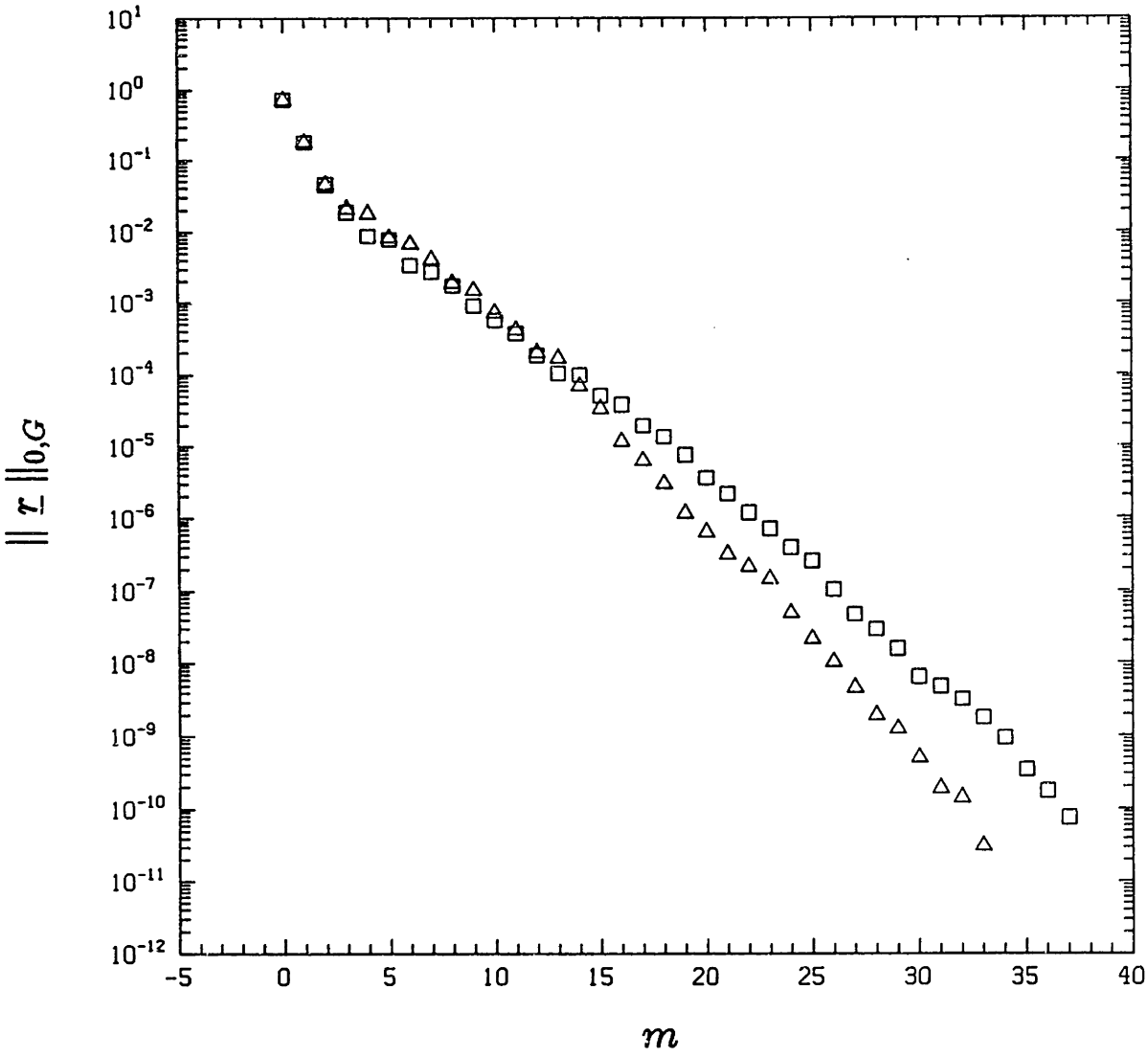


Figure 40. A plot of the residual $\| \underline{r} \|_{0,G}$ (the root-mean-square of the divergence) from (3.77) as a function of the number of outer conjugate gradient iterations, m, when solving the three-dimensional steady Stokes problem with solution $\mathbf{u} = [(1-x_2^2)(1-x_3^2), 0, 0]$, $p = sin\pi x_1/\Gamma \cdot sin\pi x_2 \cdot cos\pi x_3$ on the domain $\Omega =]-1,1[\times]-1,1[\times]0,2\Gamma[$ with $\Gamma = 1$. The domain is broken up into $K = 8$ equal spectral elements, with convergence histories shown for $N = 7$ ($\triangle$) and $N = 10$ ($\square$). The convergence rate decreases slightly with increasing N.

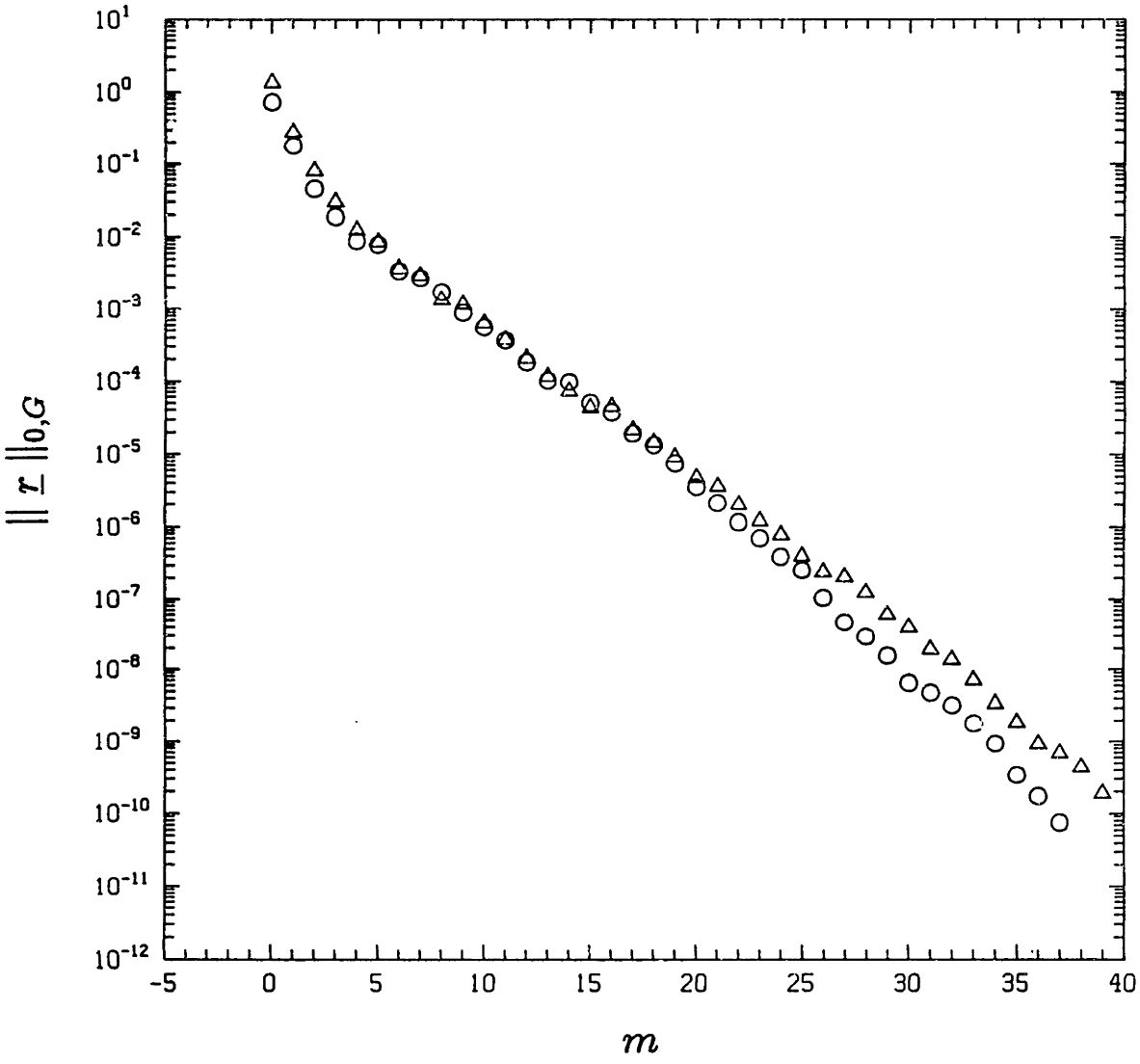


Figure 41. A plot of the residual $\| \underline{r} \|_{0,G}$ (the root-mean-square of the divergence) from (3.77) as a function of the number of outer conjugate gradient iterations, m, when solving the three-dimensional steady Stokes problem with solution $\mathbf{u} = [(1-x_2^2)(1-x_3^2), 0, 0]$, $p = sin\pi x_1/\Gamma \cdot sin\pi x_2 \cdot cos\pi x_3$ on the domain $\Omega =]-1,1[\times]-1,1[\times]0,2\Gamma[$ with $\Gamma = 1$ (○) and $\Gamma = 3$ (△). Both domains are broken up into $K = 8$ equal spectral elements, each of order $N = 10$. The convergence rate decreases slightly as the aspect ratio Γ increases.

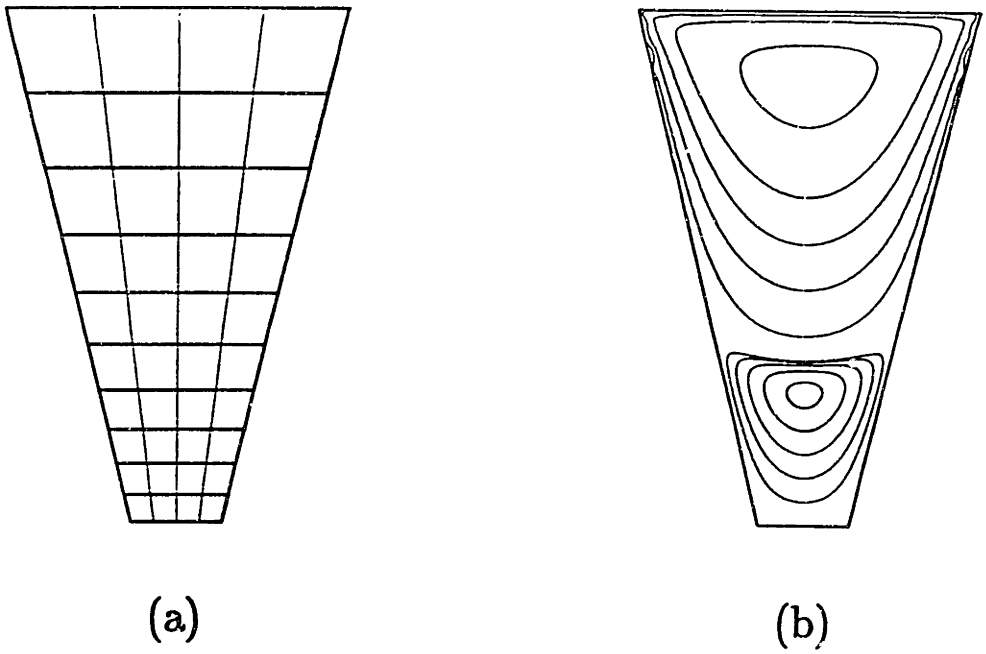


Figure 42. Creeping flow in a "wedge" where the tip of the wedge is removed. The imposed velocity boundary conditions are no-slip conditions on the two side walls and at the bottom, with a unit horizontal velocity imposed on the top side. Fig. 42a shows the spectral element discretization ($K = 40$, $N = 8$), while (b) shows the solution in form of streamlines.

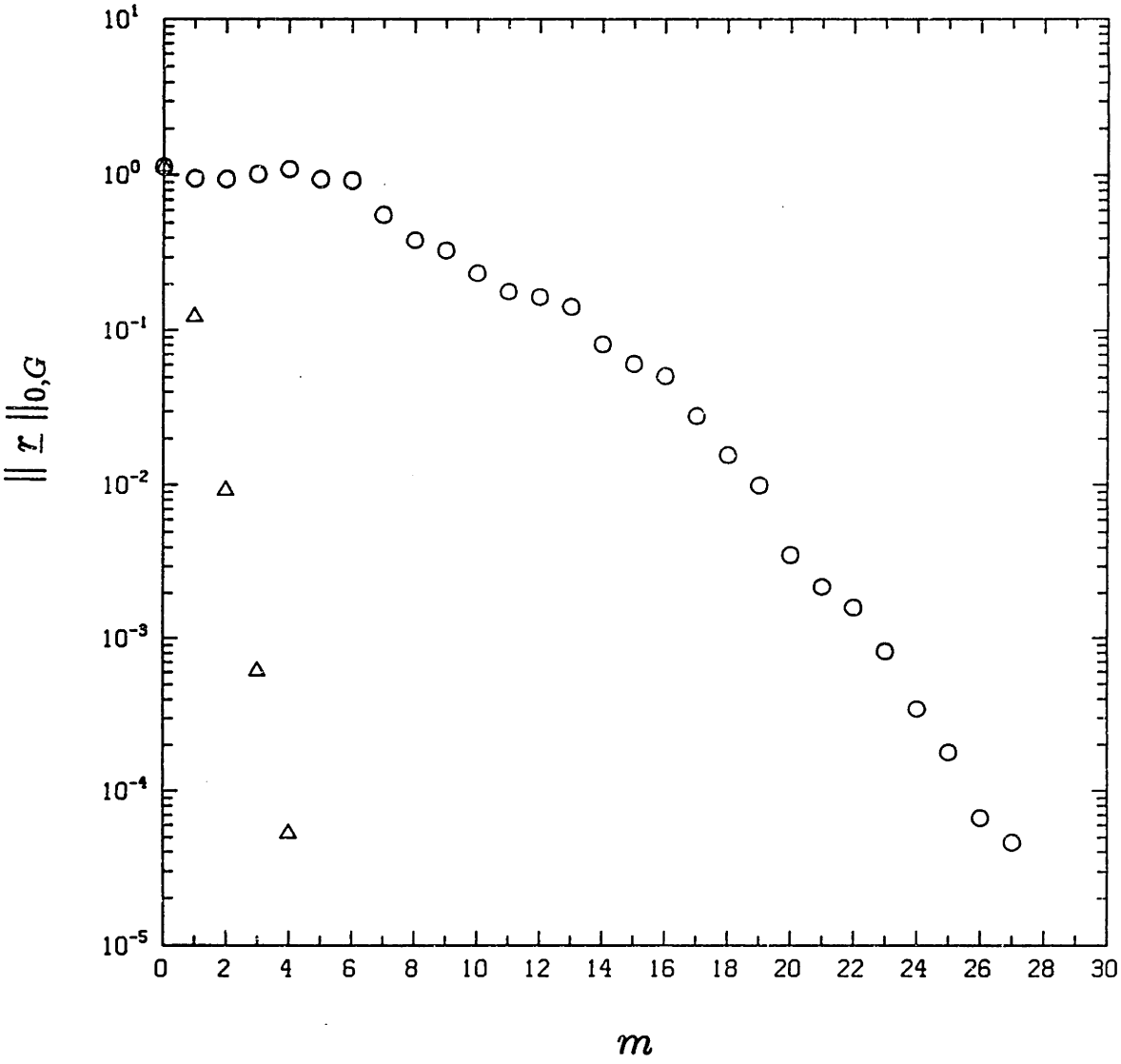


Figure 43. A plot of the residual $\| \underline{r} \|_{0,G}$ (the root-mean-square of the divergence) from (3.109) as a function of the number of outer conjugate gradient iterations, m, when solving for the first time step of the (simulated) buoyancy-driven flow shown in Fig. 25. The plot shows the convergence history with no preconditioning of the unsteady pressure operator $\underline{S}_t$ ($\bigcirc$), and when using the preconditioner from (3.105) ($\triangle$).