INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 22, 241–264 (1996)

CONVECTION TREATMENT USING SPECTRAL ELEMENTS OF DIFFERENT ORDER

EINAR M. RØNQUIST

Nektonics, Inc., 875 Main Street, Cambridge, MA 02139, U.S.A.

SUMMARY

We present here our experiences with using the spectral element methodology to solve convection-dominated problems. Different polynomial approximations are used inside the spectral elements and both conforming and non-conforming interface conditions are investigated. The three spectral element methods that we explore can all be considered to be special cases of the more general mortar element method. We compare the methods for solving incompressible fluid flow and heat transfer problems. Particular attention is given to the convection treatment. The numerical results can be strongly dependent upon whether a conforming or a non-conforming method is used as well as the particular form of the discrete convection operator (convective form versus skew-symmetric form).

KEY WORDS: spectral element; non-conforming; incompressible flows; convection; skew-symmetric

1. INTRODUCTION

In this paper we shall focus our attention on the convection treatment when solving steady and unsteady incompressible fluid flow and heat transfer problems. In terms of spatial discretization we are particularly interested in allowing for domain decompositions (or elemental decompositions) that offer maximum flexibility in terms of resolving geometrical or functional features of the underlying problem. At the same time we would like to use a framework that is well suited for the development of efficient iterative solution algorithms, particularly algorithms suitable for parallel processing.

An interesting and promising approach in this respect is the mortar element method.[1] Independent variational discretizations on the subdomains are here coupled together through an intermediary mortar trace space. The mortar element method allows for both geometrically and functionally non-conforming discretizations and has been proven to be optimal for elliptic problems[1] (optimal in the sense of providing a numerical solution with an error that is bounded by the sum of the local approximation errors). Recently a new non-conforming spectral element method was proposed[2] which allows for p-type refinement and which offers particular advantages in terms of parallel implementation. However, it is not clear how these methods perform for convection-dominated problems.

We remark here that significant progress has been made in recent years regarding adaptivity in the context of the traditional finite element framework.[3,4] Fully adaptive h–p methods currently exist for a whole range of physical applications.[5–8] However, all these methods emphasize a conforming approach.[9] To the author's knowledge there has not been a careful study that compares the impact of using different interface conditions for convection-dominated problems, at least not in the context of

CCC 0271–2091/96/040241–24

Received 18 April 1994
Revised 10 April 1995

the spectral element method or the mortar element method. Neither has there been a study that compares the impact of using different forms of the discrete convection operator (convective form versus skew-symmetric form) in the context of various interface conditions.

The current study has been undertaken in order to better understand the potential as well as the limitation of the mortar element method to solve convection-dominated problems. As a starting point we shall limit our study to geometrically conforming elements, i.e. we shall consider what is also referred to as spectral elements.[10,11] We will, however, allow the polynomial degree to vary between the elements and we will consider both conforming and non-conforming matching conditions on the interfaces between the elements.

Spectral element discretizations for partial differential equations combine spectral methods with domain decomposition techniques to obtain a high-order polynomial approximation of the solution within each subdomain. The key attractive features of this approach are rapid convergence for problems with sufficient regularity, geometric flexibility and suitability to medium-grained parallel implementation.

The rapid convergence rate derives from the good approximation properties of piecewise high-order polynomials to functions with sufficient regularity.[10,11] The geometric flexibility derives from the fact that the computational domain can be decomposed into a collection of elements which is globally unstructured. Each spectral element is a (possibly deformed) quadrilateral element in two space dimensions and a (possibly deformed) hexahedral element in three space dimensions. This type of discretization is thus characterized by a combination of (high-order) locally structured constructs and a globally unstructured representation, a fact that has made the spectral element method particularly well suited for medium-grained parallel computers.[12]

In some respects the spectral element method is very similar to the p-type finite element method.[13,14,9] Within the p-type finite element framework, including the original spectral element framework, the same polynomial approximation (p) is used within all the elements (or subdomains) and continuity of the discrete solution is enforced along the boundaries between adjacent elements for the discretization of second-order elliptic and parabolic equations. This continuity enforcement is consistent with a conforming approach insofar that the discrete space is a subspace of the continuous functional space.[15]

The key differences between the spectral element method and the p-type finite element method are related to how the numerical schemes are derived and solved. The spectral element method puts a particular emphasis on tensor product forms: tensor product nodal bases, tensor product Gauss quadratures and tensor product sum factorization techniques for efficient matrix–vector product evaluations.[16,11] The emphasis on tensor product forms has made it possible to construct efficient iterative solvers for the discrete equations; this is particularly important when solving large three-dimensional problems.[12,17] The classical spectral element method has reached a high degree of maturity over the last few years in terms of both algorithm development and applications.[18–23]

The mortar element method[1,24] is an extension of the original spectral element method to non-conforming discretizations. The main difference compared with the original spectral element method is in terms of the interface treatment between neighbouring elements (or subdomains). In the mortar element method the discrete solution is not restricted to be imbedded in the continuous functional space and this approach thus allows for the construction of more flexible discretizations. For example, local mesh refinement is more readily accommodated. This can be achieved by either splitting a given element into several smaller elements without affecting the neighbouring elements (h-refinement) or by changing the polynomial approximation within a given element (p-refinement). The mortar element method thus provides the necessary framework for the development of fully adaptive h–p refinement procedures[25,26] as well as the coupling of spectral element methods with low-order finite element methods.[27] The mortar element method is particularly attractive in the sense that it represents a domain

decomposition approach in which there is a clean decoupling of local-structure-preserving internal residual evaluations and the transmissions of boundary (or continuity) conditions.

We remark here that a key difference between the mortar element method and the *h–p* finite element method is in terms of the interface treatment between the elements. The mortar element method is based on a *non-conforming* approach through an explicit construction of an intermediary mortar trace space,[1] while the *h–p* finite element method is based on a *conforming* approach through the concept of constrained approximation.[28] In addition to the interface treatment, the mortar element method and the *h–p* finite element method differ in terms of how the numerical schemes are derived and solved.

We would also like to mention here that recently a different modification to the classical spectral element method was proposed[2] for which the key advantage is related to a more efficient parallel implementation. Again the main difference between this approach and the classical spectral element method is related to the interface treatment. Here the continuity requirement on the solution is relaxed on the entire element interface, including the element vertices.

In this paper we consider two modifications to the classical spectral element method that allow for a different polynomial approximation within each element or subdomain, i.e. we limit our discussion to spatial discretization procedures appropriate for *p*-type refinement. We shall investigate the difference in behaviour between the classical spectral element method SECLAS and the two modified methods SECONF and SENONC for the solution of second-order elliptic and parabolic problems. We note that all three methods that we consider in this study represent special cases of the more general mortar element method.[1]

Since SECLAS assumes both geometric and functional conformity and employs the same polynomial approximation within all spectral elements, there is no ambiguity about how to enforce the continuity requirement on the solution along elemental interfaces. This is no longer the case when we consider spectral element discretizations for which geometric conformity is enforced, but a different polynomial approximation is employed within each subdomain. We now have a choice between whether we still want to enforce functional conformity, SECONF, or whether we want to consider a non-conforming approach based upon the mortar element method, SENONC. For example, when considering the edge between two adjacent two-dimensional quadrilateral elements, SECONF will choose the edge values belonging to the element with the lowest polynomial degree as the real (master) degrees of freedom and then interpolate these values onto the edge with the higher polynomial degree. In SENONC the edge values belonging to the element with the highest polynomial degree will be chosen as the master nodes and these values are then projected onto the edge with the lower polynomial degree. Following the mortar element approach, the vertex values are kept the same, while the internal edge values are computed by requiring that the jump in the edge values be as small as possible; these projected internal edge nodes are denoted as slave nodes.

The ultimate objective with this work is to be able to use the mortar element methodology in the context of fully adaptive *h–p* refinement procedures[7,8] and to apply these more flexible discretizations to solve fluid flow and heat transfer problems. Again, the main issues we focus upon in this paper are related to the convection treatment, and we present numerical results that indicate the suitability of the various discretization methods to solve the steady and unsteady incompressible Navier–Stokes equations, particularly convection-dominated problems. It is the hope that these numerical results can be followed up with a theoretical analysis of the observed phenomena.

The paper is organized as follows. In Section 2 we present the three methods SECLAS, SECONF and SENONC that we investigate here. We present the methods in the context of solving the Poisson equation and we demonstrate the good convergence properties of the methods. In Section 3 we apply the three methods to solve fluid flow and heat transfer problems. In particular, we focus upon the convection treatment for the steady and unsteady incompressible Navier–Stokes equations.

2. SPECTRAL ELEMENT DISCRETIZATION

We consider here the solution of the Poisson problem in a two-dimensional domain Ω,

$$-\nabla^2 u = f \quad \text{in } \Omega, \tag{1}$$

$$u = 0 \quad \text{on } \partial\Omega, \tag{2}$$

where f is the given data and u is the solution. As a point of departure for our numerical discretization we consider the equivalent variational formulation of problem (1), (2): find $u \in H_0^1(\Omega)$ such that

$$\forall v \in H_0^1(\Omega), \quad \int_\Omega \nabla u \nabla v \, dx = \int_\Omega f v \, dx, \tag{3}$$

where $H_0^1(\Omega)$ is the standard Sobolev space.

As for the classical spectral element method, we assume that the domain Ω is broken up into K non-overlapping and conforming quadrilateral subdomains Ω_k, $1 \leqslant k \leqslant K$. This implies that the intersection of two elements Ω_k and Ω_l is either empty or reduced to a common vertex or a common edge; in the latter case we define the open interval $\Gamma_{k,l}$ as

$$\bar{\Gamma}_{k,l} = \overline{\partial\Omega}_k \cap \overline{\partial\Omega}_l. \tag{4}$$

Following closely the presentation in Reference 2, we shall focus our attention on the treatment along the interfaces (edges) $\Gamma_{k,l}$. We first rewrite the original problem (3) by introducing the space X defined by[2]

$$\begin{aligned} &X = \{\mathbf{v} = (v_k)_{1\leq k\leq K}, \quad v_k \in H^1(\Omega_k), \\ &\forall k, l, \quad 1 \leq k \leq l \leq K, \quad v_{k|\Gamma_{k,l}} = v_{l|\Gamma_{k,l}}, \quad v_{k|\partial\Omega} = 0\} \end{aligned} \tag{5}$$

and instead consider the continuous problem: find a K-tuple $\mathbf{u} = (u_k = u_{u|\Omega_k})_{1\leq k\leq K} \in X$ such that

$$\forall \mathbf{v} \in X, \quad \sum_{k=1}^{K} \int_{\Omega_k} \nabla u_k \nabla v_k \, dx = \sum_{k=1}^{K} \int_{\Omega_k} f_{|\Omega_k} v_k \, dx. \tag{6}$$

The discretization of problem (6) consists of choosing a finite-dimensional space X_δ that approximates X: find a K- tuple $\mathbf{u}_\delta = (u_{\delta,k} = u_{\delta|\Omega_k})_{1\leq k\leq K} \in X_\delta$ such that

$$\forall \mathbf{v}_\delta \in X_\delta, \quad \sum_{k=1}^{K} \int_{\Omega_k} \nabla u_{\delta,k} \nabla v_{\delta,k} \, dx = \sum_{k=1}^{K} \int_{\Omega_k} f_{|\Omega_k} v_{\delta,k} \, dx. \tag{7}$$

Before we define our choices for the discrete space X_δ, we first define the space Y_δ. Let $\mathbf{N}$ be a K-tuple of integers $\mathbf{N} = (N_k)_{1\leq k\leq K}$, where N_k is the degree of the polynomial approximation inside each subdomain. The space Y_δ is then defined as[2]

$$Y_\delta = \{\mathbf{v}_\delta = (v_{\delta,k})_{1\leq k\leq K}, \quad v_{\delta,k} \in P_{N_k}(\Omega_k)\}. \tag{8}$$

We are now in a position to define the three choices for X_δ that we shall consider througout the rest of this paper. All of them are subspaces of Y_δ.

SECLAS. This method is the classical spectral element method. In the context of the framework just presented, this method is defined by

$$X_\delta = Y_\delta \cap X, \tag{9}$$

$$N_k = N, \quad 1 \leq k \leq K, \tag{10}$$

$$N \geqslant 1. \tag{11}$$

In other words, SECLAS is a conforming method with the same polynomial approximation inside all the subdomains. Note that the condition (11) is necessary in order to meet the requirement (9).

SECONF. This method is defined simply by

$$X_\delta = Y_\delta \cap X, \tag{12}$$

$$N_k \geq 1, \quad 1 \leq k \leq K. \tag{13}$$

This method is still a conforming method, but here we do not put any restrictions on the polynomial approximation inside each subdomain, except for (13).

SENONC. This method is the non-conforming mortar element method[1,24] applied to elements with different polynomial degree inside each subdomain. Before we define the discrete space X_δ, we first define the vertices Nk,l,m, $m = 1, 2$, as the two vertices of the edge $\bar{\Gamma}_{k,l}$ defined in (4). The non-conforming space X_δ is then defined as

$$\begin{aligned} X_\delta = \{ & v_\delta = (v_{\delta,k})_{1 \leqslant k \leqslant K} \in Y_\delta, \qquad v_{\delta,k|\partial\Omega} = 0, \\ & \forall k, l, \quad 1 \leqslant K < l \leqslant K, v_{\delta,k}(N_{k,l,m}) = v_{\delta,l}(N_{k,l,m}), \ m = 1, 2, \\ & \forall k, l \quad 1 \leqslant k < l \leqslant K, \quad \forall \psi \in P_{N^*_{k,l}-2}(\Lambda_{k,l}), \quad \int_{\Gamma_{k,l}} (v_{\delta,k} - v_{\delta,l})\psi \, dl = 0\}, \end{aligned} \tag{14}$$

$$N_k > 2, \quad 1 \leqslant k \leqslant K. \tag{15}$$

In (14) $N^*_{k,l}$ is defined as the minimum of N_k and N_l.

At this point we remark that SECONF and SENONC reduce to SECLAS in the particular case of using a fixed polynomial degree in all the elements, i.e. $N_k = N$, $k = 1, \ldots, K$.

Once the space X_δ has been defined, we have to choose a basis for this space as well as a quadrature rule for evaluating the integrals in (7). To this end we follow the same procedure as for the classical spectral element method, i.e. we construct a tensor product interpolant basis through the Gauss–Lobatto Legendre points associated with each subdomain, and these points are also chosen as the quadrature points. Since the basic implementation is not the focus of this paper, we refer the reader to the literature[1,11,24,29] for more details regarding how to construct the set of algebraic equations, including the extension to variable coefficients and deformed geometries.

2.1. Numerical results: the Poisson problem

To test the convergence properties of the three methods, we consider the Poisson problem in the two-dimensional, square domain $\Omega =]-1, 1[^2$,

$$-\nabla^2 u = f \quad \text{in } \Omega, \tag{16}$$

$$u = 0 \quad \text{on } \partial\Omega. \tag{17}$$

The exact solution u and the forcing function f are given as

$$u(x, y) = (x+1)(1 - e^{\alpha(x-1)})(y+1)(1 - e^{\alpha(y-1)}), \tag{18}$$

$$\begin{aligned} f(x, y) = {} & e^{\alpha(x-1)}[2\alpha + (x+1)\alpha^2](y+1)(1 - e^{\alpha(y-1)}) \\ & + e^{\alpha(y-1)}[2\alpha + (y+1)\alpha^2](x+1)(1 - e^{\alpha(x-1)}). \end{aligned} \tag{19}$$

We break up the computational domain Ω into $K = 4$ equal subdomains (see Figure 1) and we compute the error $\|u - u_h\|$ between the exact solution u and the numerical solution u_h in the discrete

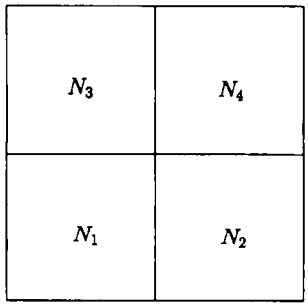


Figure 1. The spectral element mesh used to solve the Poisson problems (16)–(19). The domain is broken up into $K=4$ equal quadrilateral elements, each of order N_k, $k=1, \ldots, K$

seminorm for the different choices of polynomial approximations $\mathbf{N} = (N_k)_{1 \le k \le K}$ within the subdomains. Note that throughout the paper we assume that a spectral element is equivalent to a subdomain. We also remark that for all the convergence results reported in this paper, the error is computed in the discrete seminorm; however, we use a finer mesh for this error calculation in order to avoid quadrature errors. For the results presented for the Poisson equation, we use a polynomial degree $M_k = N_k + 3$ inside each element, $k = 1, \ldots, K$, in the error calculation.

Even though we are considering the solution of the diffusion equation, we choose the forcing function f such that the solution u has the form of a two-dimensional boundary layer. The steepest gradients are close to the upper right corner of the domain (given by the co-ordinates $(x=1, y=1)$) and the parameter α determines how steep these gradients are. In our test case we choose $\alpha = 10$; a contour plot of the solution is depicted in Figure 2. From Figures 1 and 2 we can now readily see that for this particular type of solution and elemental decomposition it should be advantageous to employ a different polynomial order inside the individual spectral elements.

Figure 3 shows the results for the classical spectral element method SECLAS, for which $N_1 = N_2 = N_3 = N_4 = N$. Owing to the fact that the solution u and the data f are analytic, the expected exponential convergence rate is observed as the polynomial degree N is increased. In Figure 3 we also report the convergence results when employing SECONF (conforming) and SENONC (non-conforming), for which we choose $N_{1+i} = N + i$, $i = 0, 1, 2, 3$. Again, as expected, we observe a similar rapid convergence rate as N is increased. We also observe that SECONF and SENONC give almost identical results and that for the same N the errors for SECONF and SENONC are roughly an order of magnitude smaller than the corresponding error for SECLAS.

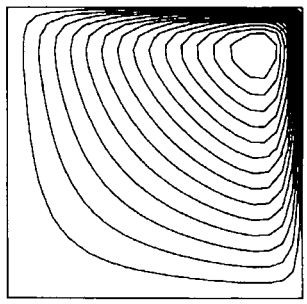


Figure 2. A contour plot of a spectral element solution to the Poisson problem (16)–(19), using the elemental decomposition depicted in Figure 1

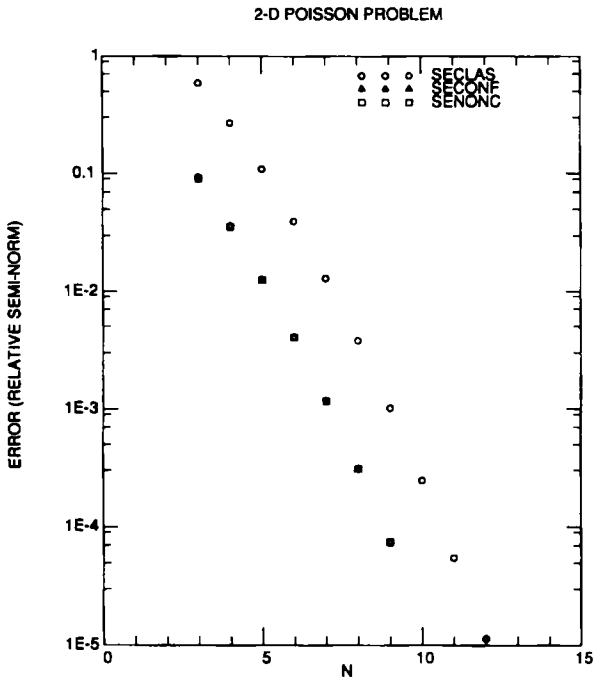


Figure 3. The error between the discrete solution and the analytical solution (18) for the Poisson problem (16)–(19). The plotted error is the relative error in the discrete seminorm as a function of the polynomial approximation N within each of the $K=4$ spectral elements. For SECLAS, $N_1=N_2=N_3=N_4=N$, while for SECONF and SENONC, $N_{1+i}=N+i$, $i=0, 1, 2, 3$

This test demonstrates the potential advantage of being able to locally refine the solution where additional resolution is needed. It also serves as a verification of the implementation of the spatial discretization methods that we consider in this study. Finally, we remark that in order for these methods to be attractive in an adaptive context, we also need to have good error estimators as well as efficient solution methods for the resulting set of algebraic equations. In this paper, however, we shall limit our discussion to the suitability of the various spatial discretization methods to solve fluid flow and heat transfer problems. In particular, the convection treatment will be our primary focus in what follows. Efficient solution methods are the focus of a separate paper.[30]

3. THE NAVIER–STOKES EQUATIONS

We now turn to the discretization of the unsteady incompressible Navier–Stokes equations

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \nu \nabla^2 \mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega, \tag{20}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega. \tag{21}$$

In (20) and (21) $\mathbf{u}$ is the fluid velocity, p is the pressure, ν is the kinematic viscosity and $\mathbf{f}$ is a body force. We assume prescribed velocity boundary conditions on the domain boundary $\partial\Omega$ as well as initial conditions for the velocity for unsteady problems.

Similarly to the Poisson problem discussed in Section 2, a spectral element discretization of (20) and (21) is based on the equivalent weak form. For two-dimensional problems and homogeneous velocity

boundary conditions we can formulate the problem as: find $\mathbf{u} \in [H_0^1(\Omega)]^2$ and $p \in L_0^2(\Omega)$ such that

$$\left(\frac{\partial \mathbf{u}}{\partial t}, \mathbf{w}\right) + (\mathbf{u} \cdot \nabla \mathbf{u}, \mathbf{w}) + \nu(\nabla \mathbf{u}, \nabla \mathbf{w}) - (p, \nabla \cdot \mathbf{w}) = (\mathbf{f}, \mathbf{w}), \quad \forall \mathbf{w} \in [H_0^1(\Omega)]^2, \tag{22}$$

$$(\nabla \cdot \mathbf{u}, q) = 0, \quad \forall q \in L_0^2(\Omega), \tag{23}$$

where $L_0^2(\Omega)$ is the space of all functions which are square integrable and have zero average over Ω. In (22) and (23) the inner product $(\cdot, \cdot)$ is defined as

$$\forall \Phi, \Psi \in L^2(\Omega), \quad (\Phi, \Psi) = \int_\Omega \Phi \Psi \, d\Omega. \tag{24}$$

3.1. Spatial discretization

For the spatial discretization of (20) and (21) we follow the same approach as employed by the classical spectral element method (SECLAS). We refer the reader to the literature[11,22,29] for more details regarding the theoretical justification, implementation and numerical results. The key features comprise the use of the weak form (22), (23) and the use of compatible approximation spaces for the velocity and pressure. The same procedure is also followed when implementing SECONF and SENONC. The only difference between the three methods is the way the velocity is being treated on the interface between adjacent elements.

Using the notation from Section 2, the approximation space for the discrete (two-dimensional) velocity $\mathbf{u}_\delta$ is

$$\mathbf{u}_\delta \in X_\delta^2. \tag{25}$$

With the definition of the space Y_δ given in (8), we define the related space

$$\tilde{Y}_\delta = \{\mathbf{v}_\delta = (v_{\delta,k})_{1 \le k \le K}, \quad v_{\delta,k} \in P_{N_k - 2}(\Omega_k)\}. \tag{26}$$

The approximation space for the discrete pressure p_δ is then

$$p_\delta \in \tilde{Y}_\delta \cap L_0^2(\Omega), \tag{27}$$

i.e. the polynomial degree for the pressure is two orders lower than for the velocity inside each subdomain (or spectral element).[22,31] We remark that since the pressure needs only be square integrable, no continuity requirement for the pressure is enforced between elements and therefore the pressure treatment is identical for all three methods. We now turn to the convection treatment.

3.1.1. Convection treatment. One of the primary objectives of this work is to investigate and compare the use of conforming and non-conforming spectral elements (SECLAS, SECONF and SENONC) for the solution of the incompressible Navier–Stokes equations, particularly for convection-dominated problems. The ultimate goal is to construct proper multidomain spectral techniques that can be used in the context of adaptive procedures.

The main issue we shall address here is concerned with the proper form of the discrete convection operator. Specifically, we shall compare the use of the convection form and the skew-symmetric form, both in the context of solving the steady Navier–Stokes equations as well as in the context of integrating the unsteady equations.

We first present the different choices for the convection operator in the context of incompressible fluid flow. For the convection of a scalar field θ (e.g. temperature or one of the velocity components) the alternative forms of the convection operator $\mathscr{C}$ are

$$\mathscr{C}\theta = \mathbf{u} \cdot \nabla \theta, \tag{28}$$

$$\mathscr{C}\theta = \nabla \cdot \mathbf{u}\theta, \tag{29}$$

$$\mathscr{C}\theta = \tfrac{1}{2}\mathbf{u} \cdot \nabla \theta + \tfrac{1}{2}\nabla \cdot \mathbf{u}\theta. \tag{30}$$

The first form is called the convective form, the second the conservative form and the third the skew-symmetric form (the average of the first two forms). For solenoidal velocity fields, all three forms are equivalent in the continuous case; however, this is not the case for the associated discrete operators. There have been several studies in the past which have compared the different forms.[29,32–34] These studies seem to indicate that the skew-symmetric form is preferable for multidimensional problems. We remark that for incompressible fluid flow there is also an additional form, the rotational form, that can be used. However, we will not study this form here. The reason is that this form has somewhat similar conservation properties to the skew-symmetric form and the latter form has been shown to be superior in earlier work.

The reason why the skew-symmetric form is attractive will become clearer when we discuss how the boundary conditions imposed on the boundary $\partial\Omega$ affect the properties of the convection operator. To see this, we first multiply (28) by a test function ψ and integrate over the domain Ω; we then integrate by parts and finally use the incompressibility constraint to obtain

$$(\psi,\ \mathbf{u}\cdot\nabla\theta) = -(\mathbf{u}\cdot\nabla\psi,\ \theta) + \int_{\partial\Omega} \psi\theta\mathbf{u}\cdot\hat{\mathbf{n}}\,\mathrm{d}S \tag{31}$$

or

$$(\psi, \mathscr{C}\theta) = -(\mathscr{C}\psi, \theta) + I_{\partial\Omega}(\theta), \tag{32}$$

where $I_{\partial\Omega}(\theta)$ is the surface integral

$$I_{\partial\Omega}(\theta) = \int_{\partial\Omega} \psi\theta\mathbf{u}\cdot\hat{\mathbf{n}}\,\mathrm{d}S \tag{33}$$

and $\hat{\mathbf{n}}$ denotes the outward unit normal on the domain boundary $\partial\Omega$. It follows that for the class of homogeneous boundary conditions on θ and ψ for which the surface integral $I_{\partial\Omega}(\theta)$ vanishes, the convection operator $\mathscr{C}$ is skew-symmetric. The corresponding set of homogeneous, linear boundary conditions will be denoted as

$$\mathscr{B}_{ss}\theta = 0. \tag{34}$$

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 $\mathbf{u}\cdot\hat{\mathbf{n}} = 0$. We further assume that the imposed boundary conditions lead to a well-posed system. The skew-symmetric property implies that $\mathscr{C}$ has imaginary eigenvalues, reflecting the fact that the convection operator is non-dissipative.

Another important class of boundary conditions is the 'outflow' boundary conditions. In this case no specification of θ is imposed on a segment of $\partial\Omega$ for which $\mathbf{u}\cdot\hat{\mathbf{n}} > 0$. The extended set of boundary conditions that includes outflow as well as skew-symmetric boundary conditions will be denoted as

$$\mathscr{B}\theta = 0. \tag{35}$$

In the case of outflow conditions the boundary integral $I_{\partial\Omega}(\theta)$ does not vanish and $\mathscr{C}$ is no longer skew-symmetric. It can be shown that the eigenvalues are now in the right-hand plane $(\theta, \mathscr{C}\theta) > 0$, 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 skew-symmetric form (30) instead of the convective form (28) (or the conservative form (29)). In deriving the skew-symmetric property for the convective form of $\mathscr{C}$ in the case of $\mathscr{B}_{ss}\theta = 0$ (see (31)), we use the fact that the velocity field is incompressible; however, the proof of the skew symmetry for $\mathscr{C}$ as defined in (30) does *not* rely on the fact that the velocity field is 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. The implication of this is that, even in the presence of quadrature errors, a discrete convection operator based upon the skew-symmetric form will also be skew-symmetric for skew-symmetric boundary conditions, i.e. all the discrete eigenvalues will be imaginary. In many cases this is very desirable, at least if energy conservation is important.

In terms of computational cost we remark that the skew-symmetric form is approximately twice as expensive as the convective form in terms of matrix–vector product evaluations; this is primarily due to the fact that the skew-symmetric form requires the evaluation of both the convective form and the conservative form (see (28)–(30)). We also remark that the convective form is easier to implement than the skew-symmetric form. This is particularly true for outflow boundary conditions, in which case the skew-symmetric operator also includes the boundary term (33).

3.2. Numerical results: the steady Navier–Stokes equations

We now illustrate the spatial convergence rate associated with the spectral element discretization of the steady two-dimensional Navier–Stokes equations. Kovasznay[35] gives an analytical solution to the Navier–Stokes equations which is similar to the two-dimensional flow field behind a periodic array of cylinders:

$$u = 1 - e^{-\lambda x}\cos(2\pi y), \tag{36}$$

$$v = \frac{\lambda}{2\pi}e^{-\lambda x}\sin(2\pi y), \tag{37}$$

$$\lambda = \tfrac{1}{2}Re \pm \sqrt{(\tfrac{1}{4}Re^2 + 4\pi^2)}, \tag{38}$$

where Re is the Reynolds number based on the mean flow velocity and separation between vortices. We solve this problem numerically in the case of $Re = 40$ and $\lambda = \frac{1}{2}Re - \sqrt{(\frac{1}{4}Re^2 + 4\pi^2)}$, imposing the analytical velocity solution on the domain boundary.

In Figure 4 we show the decomposition of the domain $\Omega =]-0{\cdot}5,\ 1{\cdot}0]\times]-0{\cdot}5, 1{\cdot}5[$ into $K = 6$ equal quadrilateral spectral elements. For SECLAS the polynomial approximation N used inside each element is the same for all the elements. For SECONF and SENONC half the elements employ a polynomial approximation equal to N and the other half use $N+3$. Figure 5 depicts the velocity vectors

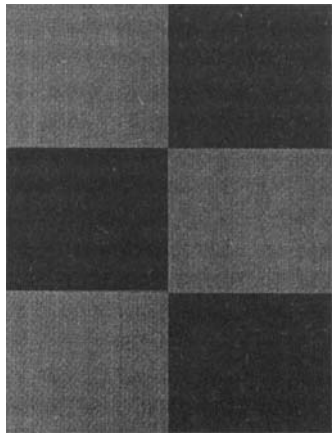


Figure 4. A plot of the spectral element decomposition and the polynomial order that is used within each spectral element when solving the Kovasznay problem (36)–(38) with SECONF (conforming) and SENONC (non-conforming). Half the elements use order N, while the other half employ order $N+3$. For SECLAS the polynomial approximation is N for all the elements

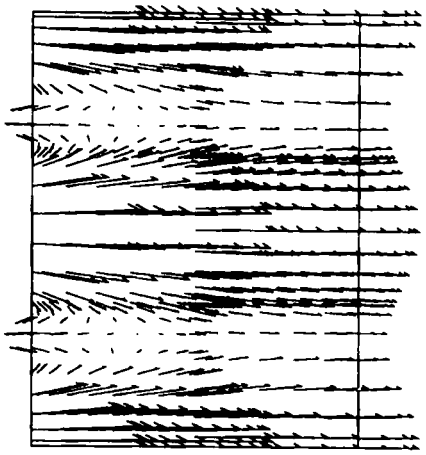


Figure 5. Velocity vectors for the two-dimensional Kovasznay solution (36)–(38). The Reynolds number based on vortex separation, mean flow speed and kinematic viscosity is $Re = 40$. The computational domain is broken up into $K = 6$ quadrilateral spectral elements. The results are obtained with the non-conforming method (SENONC) for a polynomial approximation $N = 7$

when using SENONC with $N = 7$; in the centre of the domain we can clearly see that a different polynomial approximation is used in the adjacent elements.

Similarly to the convergence study presented earlier for the Poisson problem, the errors we present for the Kovasznay problem are also computed in the discrete seminorm. We use a polynomial degree $M_k = N_k + 2$ inside each element, $k = 1, \ldots, K$, in the error calculation.

The set of fully coupled, discrete equations are solved by first linearizing the discrete equations (Newton iteration) and then solving each linearized system by a global iterative procedure. The reason for using an iterative method is primarily due to the relatively large bandwidth associated with high-order methods; using a direct solver is only practical for relatively low values of the polynomial degree N. The iterative procedure we use here comprises a global GMRES iteration[36] in conjunction with a domain-decomposition-based preconditioning. The type of preconditioning we use to solve the set of non-symmetric equations is inspired by the development of iterative substructuring methods for elliptic problems.[37] Although the development of good solution methods represents a very important aspect in the development of adaptive procedures, our focus in this study is limited to the discretization aspect. In particular, we limit our study to finding spectral element spatial discretization methods appropriate for use in the context of p-refinement procedures and suitable for solving convection-dominated fluid flow and heat transfer problems. The particular details regarding the iterative solution algorithm are reported in a separate paper.[30]

We now present the convergence results for SECLAS, SECONF and SENONC, both in the context of using the convective form of the convection operator as well as the skew-symmetric form. Figure 6 shows the relative velocity error as a function of the polynomial degree N for a fixed number of elements, $K = 6$, when using the convective form; Figure 7 shows the corresponding results for the skew-symmetric form. Spectral convergence is clearly achieved for all cases for this smooth solution. However, there are several features worth commenting on. First, there is a significant difference in the convergence rate when comparing the results for the convective form with the results for the skew-symmetric form. For example, for $N = 10$ the error when using the convective form is roughly three orders of magnitude smaller than the corresponding error for the skew-symmetric form.

Even though there is no theoretical analysis available yet, we would like to make a conjecture as to the reason for the 'poor' performance of the skew-symmetric operator. For this problem, we impose Dirichlet velocity boundary conditions on the whole boundary $\partial\Omega$, i.e. $I_{\partial\Omega}(\mathbf{u}) = 0$ and $\mathscr{B}_{ss}\mathbf{u} = \mathbf{0}$.

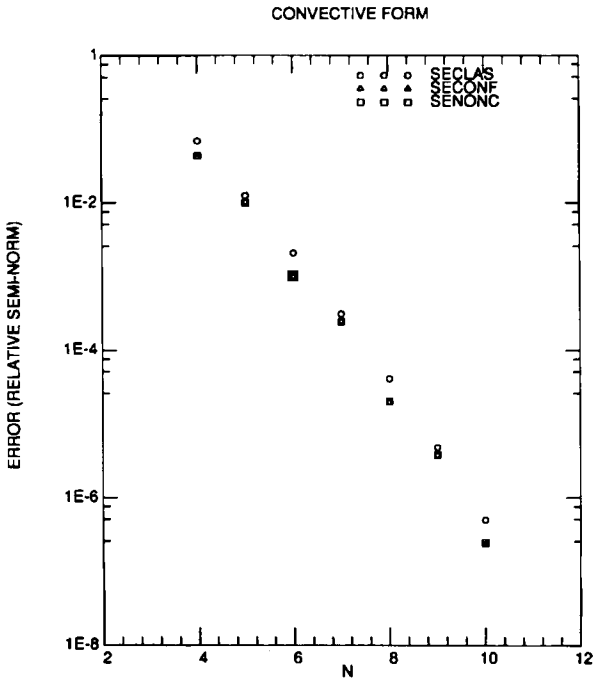


Figure 6. The error between the computed velocity and the analytical solution (36)–(37) for the Kovasznay problem defined in the domain $\Omega =]-0{\cdot}5,\ 1{\cdot}0[\times]-0{\cdot}5,\ 1{\cdot}5[$. The plotted error is the relative error in the discrete seminorm as a function of the polynomial approximation N within each of the $K = 6$ spectral elements. Note that for SECONF and SENONC half the elements employ order $N+3$ as depicted in Figure 4. The convective form of the convection operator is used for SECLAS, SECONF and SENONC

However, the exact velocity that is prescribed on the boundary for the continuous problem cannot be represented in terms of piecewise high-order polynomials (see (36) and (37)), and the boundary conditions are therefore only approximated in the discrete problem. The implication is that $I_{\partial\Omega}(\mathbf{u}_\delta) \neq 0$. We note that when using the skew-symmetric form of the convection operator in conjunction with the skew-symmetric boundary conditions, the associated discrete operator is still skew-symmetric (by construction); however, some accuracy will be lost owing to the imposition of inexact velocity boundary conditions. We also note that the discrete convection operator based upon the convective form does not require $I_{\partial\Omega}(\mathbf{u}) = 0$ and therefore this form is less sensitive to inaccurate representation of the velocity boundary conditions.

To test this hypothesis, we repeat the convergence study, but now considering only the solution inside the lower quarter of the original domain, i.e. $\Omega =]-0{\cdot}5,\ 1{\cdot}0]\times]-0{\cdot}5,\ 0{\cdot}0[$. The implication of this is that the velocity boundary condition is much better approximated in the y-direction where the variation in velocity is the greatest. The convergence results are presented in Figures 8 and 9. Indeed, we now see that the convergence rate for the convection form and the skew- symmetric form is essentially the same.

We now continue the discussion of the convergence results with emphasis on the difference in behaviour between SECLAS, SECONF and SENONC. For the convective form (see Figures 6 and 8) the results indicate that all three methods give comparable results, with SECONF giving the smallest and SECLAS the largest error for a fixed value of N. However, the difference in error between the three methods is less than a factor of two, a fact that perhaps can be explained from a purly approximation point of view. In the case of using SECONF and SENONC, half the elements employ a polynomial

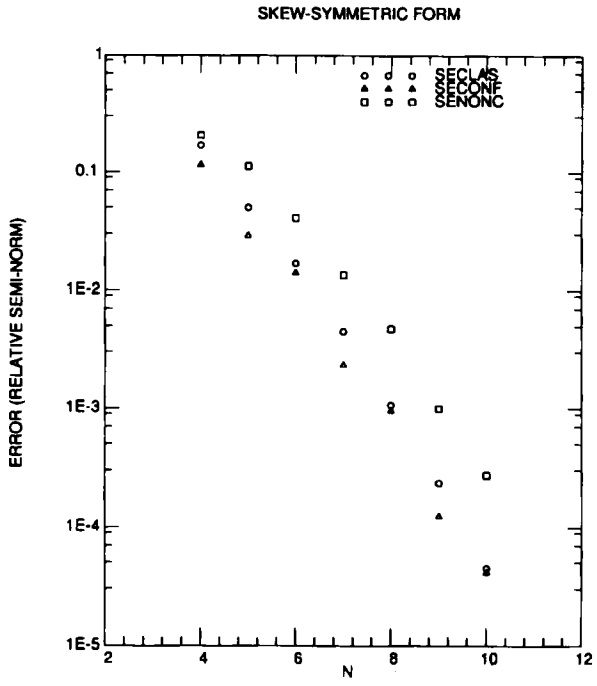


Figure 7. The error between the computed velocity and the analytical solution (36)–(37) for the Kovasznay problem defined in the domain $\Omega =]-0{\cdot}5,\ 1{\cdot}0[\times]-0{\cdot}5,\ 1{\cdot}5[$. The plotted error is the relative error in the discrete seminorm as a function of the polynomial approximation N within each of the $K=6$ spectral elements. Note that for SECONF and SENONC half the elements employ order $N+3$ as depicted in Figure 4. The skew-symmetric form of the convection operator is used for SECLAS, SECONF and SENONC

degree N and the other half $N+3$. The local approximation error inside the elements with a polynomial degree $N+3$ should therefore be much smaller than the corresponding error inside the elements with a polynomial degree N, resulting in at most a factor-of-two reduction in the global approximation error. Since we expect the discretization error to be only a constant away from the best approximation error,[1,11] this could explain the observed results.

For the skew-symmetric form we made the observation that an inaccurate representation of the velocity boundary conditions seems to result in a significantly reduced convergence rate compared with the use of the convective form (see Figures 6 and 7). In addition, the results in Figure 7 for the original, larger domain indicate that the error for the non-conforming method SENONC, is generally two to five times larger than for the classical spectral element method SECLAS, while SECONF generally gives the best result. The comment we would like to make here is that for the terms in the weak form that are derived using integration by parts, the associated boundary terms that result are defined on all elemental boundaries, both along the external boundary $\partial\Omega$ as well as along the interelemental boundaries which we will denote as Γ (all $\Gamma_{k,l}$ as defined in (4)). For example, for the spectral element decomposition considered here, the surface integral in (33) is defined not only over $\partial\Omega$ but also along Γ. For the conforming methods SECLAS and SECONF, the surface integral along an edge $\Gamma_{k,l}$ will be identically zero owing to the cancellation of the contribution from the adjacent elements Ω_k and Ω_l. For the non-conforming method SENONC, the surface integral along Γ will not be zero, i.e. $I_\Gamma(\mathbf{u}_\delta) \neq 0$. It appears from the numerical results in Figure 7 that if the approximation of the Dirichlet velocity boundary conditions is poor ($I_{\partial\Omega}(\mathbf{u}_\delta)$ is large in a certain sense), this is aggravated by the fact that $I_\Gamma(\mathbf{u}_\delta) \neq 0$.

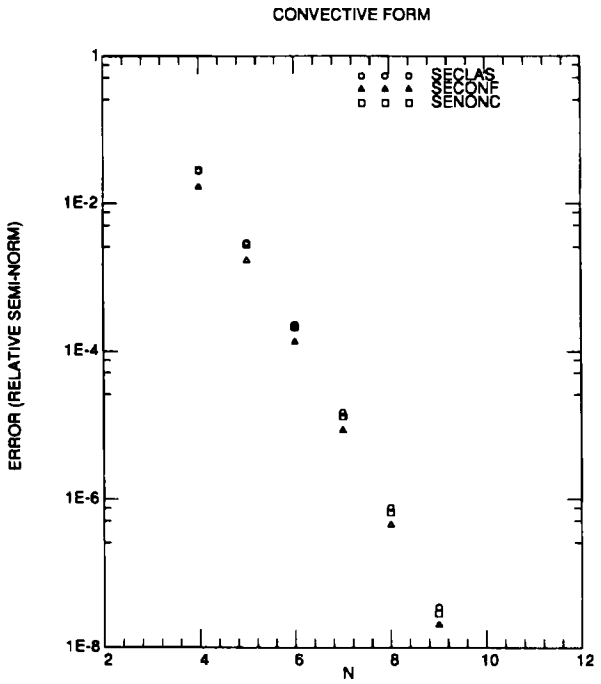


Figure 8. The error between the computed velocity and the analytical solution (36)–(37) for the Kovasznay problem defined in the reduced domain $\Omega =]-0{\cdot}5,\ 1{\cdot}0[\times]-0{\cdot}5,\ 0{\cdot}0[$. The plotted error is the relative error in the discrete seminorm as a function of the polynomial approximation N within each of the $K = 6$ spectral elements. Note that for SECONF and SENONC half the elements employ order $N+3$ as depicted in Figure 4. The convective form of the convection operator is used for SECLAS, SECONF and SENONC

Finally, when we consider the more well-resolved case (including boundary conditions) corresponding to the skew-symmetric results in Figure 9, we notice that all three methods again give comparable results and that these results are almost identical with the corresponding results when using the convective form (see Figure 8).

3.3. Temporal discretization

The temporal discretization of (20) and (21) that we shall use here is characterized by a semi-implicit operator-splitting approach.[38] The treatment of each time step is broken up into three substeps: an explicit convection step, an implicit viscous step and an implicit pressure step. This decoupling of the spatial operators is motivated by the fact that the operators can typically be treated computationally more efficiently by handling them separately rather than as an ensemble.

Following such an operator-splitting approach, the convection operator is typically treated explicitly. The reason for this is that for large two- and three-dimensional problems, iterative solvers are currently the only viable approach, particularly for high-order methods.[12,17,21] Since iterative solvers for non-symmetric systems are less mature compared with their symmetric counterparts, an explicit convection treatment has been preferred in the past, although the resulting Courant limitation on the allowable time step can sometimes be restrictive. One approach for the convection step is to use a third-order Adams–Bashforth multistep scheme,[39] which only results in a partial decoupling from the viscous and pressure steps (the time step has to be the same). An alternative approach for the explicit convection step is to use the method of characteristics,[40] in which several (inexpensive) convection substeps can be completed before acting with the (expensive) discrete Stokes operator.[38]

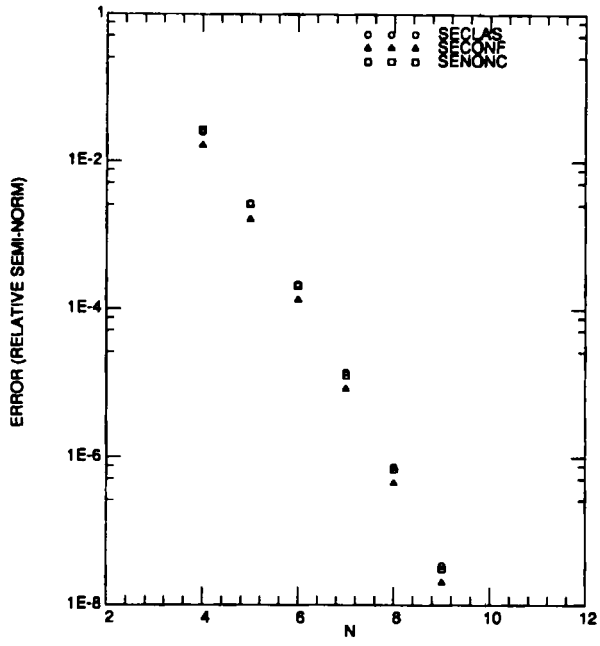


Figure 9. The error between the computed velocity and the analytical solution (36)–(37) for the Kovasznay problem defined in the reduced domain $\Omega =]-0{\cdot}5,\ 1{\cdot}0[\times]-0{\cdot}5,\ 0{\cdot}0[$. The plotted error is the relative error in the discrete seminorm as a function of the polynomial approximation N within each of the $K = 6$ spectral elements. Note that for SECONF and SENONC half the elements employ order $N+3$ as depicted in Figure 4. The skew-symmetric form of the convection operator is used for SECLAS, SECONF and SENONC

3.4. Numerical results: the unsteady Navier–Stokes equations

The purpose of this subsection is to investigate the behaviour of SECLAS, SECONF and SENONC defined in Section 2 for solving unsteady convection–diffusion problems. This investigation was motivated by the fact that we sometimes experienced stability problems when employing SENONC.

3.4.1. Flow past a cylinder. As an illustrative example we consider first the flow past a cylinder at $Re = 200$ based on the cylinder diameter. The computational domain is depicted in Figure 10 together

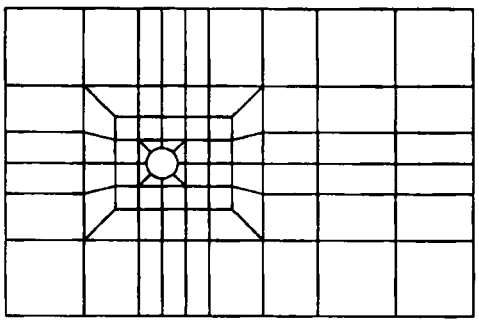


Figure 10. A plot of the computational domain and the spectral element decomposition for solving the flow past a cylinder at $Re = 200$ based on cylinder diameter, inlet velocity and kinematic viscosity. The flow is from left to right. The velocity is specified along the inflow boundary, no-slip boundary conditions are specified along the cylinder and outflow (zero Neumann) boundary conditions are specified on the rest of the domain boundary. The domain is broken up into $K = 76$ quadrilateral spectral elements (some of them deformed)

with the spectral element decomposition. At the inlet we specify Dirichlet velocity boundary conditions: the x-component of the velocity is set equal to unity, while the y-component is set equal to zero. No-slip boundary conditions are specified along the cylinder, while zero Neumann (outflow) boundary conditions are specified along the rest of the domain boundary. We remark that these boundary conditions do *not* correspond to skew-symmetric boundary conditions.

We also solve an associated convection–diffusion heat transfer problem in which the Peclet number is equal to the Reynolds number. The boundary conditions for this problem are zero temperature along the inflow boundary, unit temperature specified along the cylinder wall and zero flux everywhere else. The heat transfer problem is included here for visualization purposes.

For all the results reported below, the computational domain is broken up into $K = 76$ subdomains or spectral elements (see Figure 10). The unsteady incompressible Navier–Stokes equations are solved using SECLAS with $N = 7$ until a von Karman vortex street has developed. The solution at this time is used as an initial condition for the simulation results that we now present for SECONF and SENONC. The viscous term is treated with a first-order backward differentiation scheme and the convection step is treated by the method of characteristics, in which a fourth-order explicit Runge–Kutta scheme is employed.[38,39] Each simulation result we present is the result we obtain after 1000 time steps, using a time step equal to $\Delta t = 0{\cdot}05$ and starting from the initial conditions including an already developed von Karman vortex street.

For all the test cases we report below, we choose the polynomial approximation in each element in the upstream part of the computational domain to be the same and equal to N^-, while the polynomial approximation in the downstream part of the domain is equal to N^+ (see Figure 11). For SECLAS we always have $N^- = N^+ = N$. For SECONF and SENONC we choose either $N^- = N$ and $N^+ = N+1$ or $N^- = N+1$ and $N^+ = N$.

First we consider the case with $N^- = 9$ and $N^+ = 8$. In Figure 12(a) we show the temperature contours when using SECONF in conjunction with the convective form of the convection operator. In this case SECONF and SENONC are both stable regardless of whether the convective or the skew-symmetric form is used. The results that are obtained in all these cases look almost identical and hence we only show one of them here.

Next we reverse the situation by choosing $N^- = 8$ and $N^+ = 9$. For this problem SECONF is again stable regardless of whether the convective or the skew-symmetric form is used. However, when using SENONC in conjunction with the convective form of the convection operator, the solution becomes

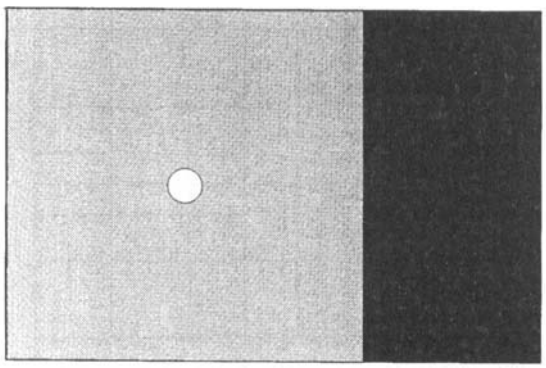
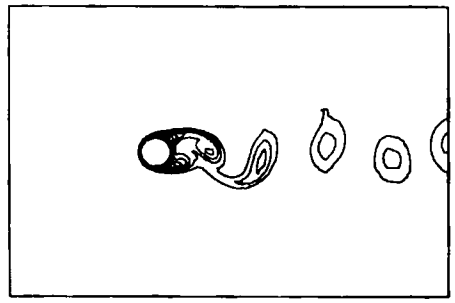
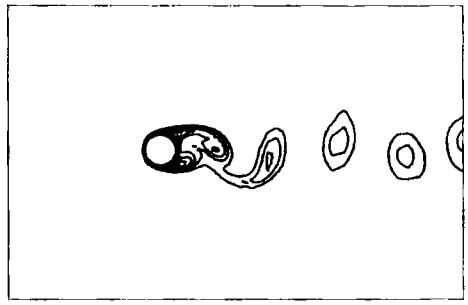


Figure 11. The distribution of the polynomial order that is used within each spectral element when solving the flow past a cylinder at $Re = 200$. The polynomial approximation in each element in the upstream part of the computational domain is N^-, while the polynomial approximation in the downstream part of the domain is N^+ (dark region). For SECLAS we always have $N^- = N^+ = N$. For SECONF and SENONC we choose either $N^- = N$ and $N^+ = N+1$ or $N^- = N+1$ and $N^+ = N$



(a)



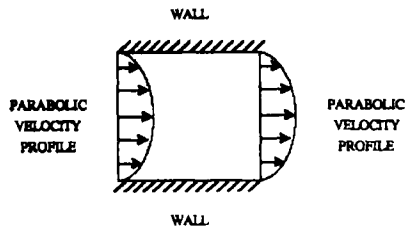
(b)

Figure 12. Results for flow past a cylinder at $Re = 200$. For visualization purposes we show the temperature contours for the associated convection–diffusion problem at $Pe = 200$. The presence of a von Karman street is clearly seen. In (a) we show the result using SECONF with $N^- = 9$ and $N^- = 8$ and using the convective form of the convection operator. In (b) we show the result using SENONC with $N^- = 8$ and $N^+ = 9$ and using the skew-symmetric form of the convection operator

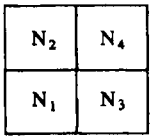
unstable at roughly the same time independently of the choice of time step. The reason for this is that the explicitly treated convection operator appears to be unstable when the flow is going from a lower-order element into a higher-order element. However, if we repeat the experiment with SENONC but now using the skew-symmetric form of the convection operator, the solution process is stable and the result after 1000 time steps (see Figure 12(b)) looks almost identical with the result displayed in Figure 12(a).

3.4.2. Two-dimensional channel flow. In order to investigate the cause of the observed instability, we consider the model problem of solving the unsteady Navier–Stokes equations in the square domain $\Omega =]-1, 1[^2$ (see Figure 13(a)), specifying a parabolic velocity profile at the inlet and outlet and no-slip boundary conditions on the top and bottom walls. The initial conditions for the velocity are set to zero, except for the non-homogeneous velocity boundary conditions.

The computational domain is broken up into $K = 4$ equal squares Ω_k, $k = 1, 2, 3, 4$, and the polynomial approximation used inside each subdomain is N_k, $k = 1, 2, 3, 4$ (see Figure 13(b)). The unsteady Navier–Stokes equations are integrated for different values of the Reynolds number $Re = VH/\nu$. Here V is the maximum velocity at the inlet and H is the channel half-width. We repeat the calculations for all three methods SECLAS, SECONF and SENONC.



(a)



(b)

Figure 13. (a) Boundary conditions and (b) domain decomposition for the two-dimensional channel flow problem

We choose the polynomial approximation in each subdomain to be $N_1 = N_2 = N^-$ and $N_3 = N_4 = N^+$. Here the signs $-$ and $+$ are associated with the upstream part and the downstream part relative to the centre of the domain respectively ($x < 0$ and $x > 0$). For SECLAS we always have $N^- = N^+ = N$. For SECONF and SENONC we choose either $N^- = N$ and $N^+ = N + 1$ or $N^- = N + 1$ and $N^+ = N$.

First we integrate the unsteady Navier–Stokes equations by following an operator-splitting approach in which the viscous operator is treated by a first-order (implicit) backward differentiation scheme and the convection operator is treated either by a third-order explicit Adams–Bashforth scheme[11] or by a fourth-order explicit Runge–Kutta scheme.[38,39] Our experience has been the following: when the Reynolds number is small ($Re < 100$), all three methods are stable. However, for $Re = 200$, say, SENONC is unstable if $N^- < N^+$ and the convective form of the convection operator is used. The instability seems to occur roughly at the same time independently of the choice of time step.

Secondly we integrate the unsteady Navier–Stokes equations using a fully coupled, fully implicit approach (a first-order backward differentiation scheme). In this case no stability problems are observed for any of the three methods or for any form of the discrete convection operator.

These results, together with the simulation results for the flow past a cylinder, suggest that the observed instability for non-conforming discretizations is associated with the use of an explicit time integration scheme as well as the use of the convective form of the convection operator. Since the stability of an explicit time integration scheme is closely linked to the eigenvalues of the associated discrete spatial operator, we now compute the entire set of eigenvalues for the various discrete convection operators.

We consider the discrete eigenvalue problem

$$-\mathbf{C}\boldsymbol{\psi} = \lambda \mathbf{B}\boldsymbol{\psi}, \tag{39}$$

where $\mathbf{C}$ is the discrete two-dimensional convector operator, $\mathbf{B}$ is the mass matrix, $\boldsymbol{\psi}$ represents a discrete eivenvector and λ represents the corresponding discrete eigenvalue. In (39) we choose the fully developed (parabolic) velocity field for channel flow as the convecting field. Figure 14 shows the entire spectrum when using SECLAS with $N = 4$. We see that all the eigenvalues lie on the imaginary axis,

which explains why the explicit treatment of the convection operator is stable as long as the time step is small enough so that $\lambda\Delta t$ lies within the absolute stability region of the pertinent explicit time integration scheme.[39]

The results in Figure 14, as well as the following results that we will present, are all obtained with the convective form of the convection operator. By construction, for skew-symmetric boundary conditions, as in this case, the skew-symmetric form always results in a discrete convection operator with purely imaginary eigenvalues, and the spectra associated with this operator are therefore not shown (they all look similar to the results in Figure 14).

Next we consider the case with $N^- = 4$ and $N^+ = 5$. Figure 15 shows the entire spectrum when using SECONF, while Figure 16 shows the corresponding results for SENONC. We notice that the eigenvalues are no longer lying exactly on the imaginary axis. The conforming approach SECONF moves some of the eigenvalues into the left-hand plane, which explains why this approach still is stable as long as the Courant condition is satisfied. The non-conforming approach, however, moves some of the discrete eigenvalues into the right-hand plane, which explains the observed instability. In particular we observe one eigenvalue λ with the property $\mathrm{Im}(\lambda) = 0$ and $\mathrm{Re}(\lambda) > 0$. Since this eigenvalue cannot be moved inside the absolute stability regions for the explicit Adams–Bashforth methods or the explicit Runge–Kutta method for any choice of the time step, SENONC will always cause instability for this choice of spatial discretization. The fact that no instability is observed for low Reynolds numbers is most likely due to the fact that the diffusion is large enough to act as a stabilizer (or that the grid Peclet number is small enough).

We now repeat the above experiments for SECONF and SENONC (again in conjunction with the convective form of the convection operator), but this time we choose $N^- = 5$ and $N^+ = 4$. Both methods are now stable and the associated spectra for the discrete convection operators are shown in

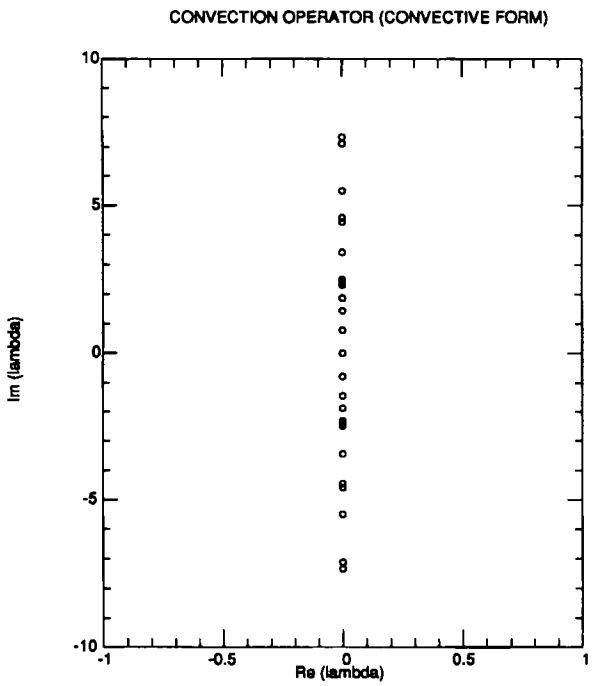


Figure 14. Eigenvalues of the discrete convection operator for two-dimensional channel flow. The results are for SECLAS with $N = 4$ for all $K = 4$ spectral elements

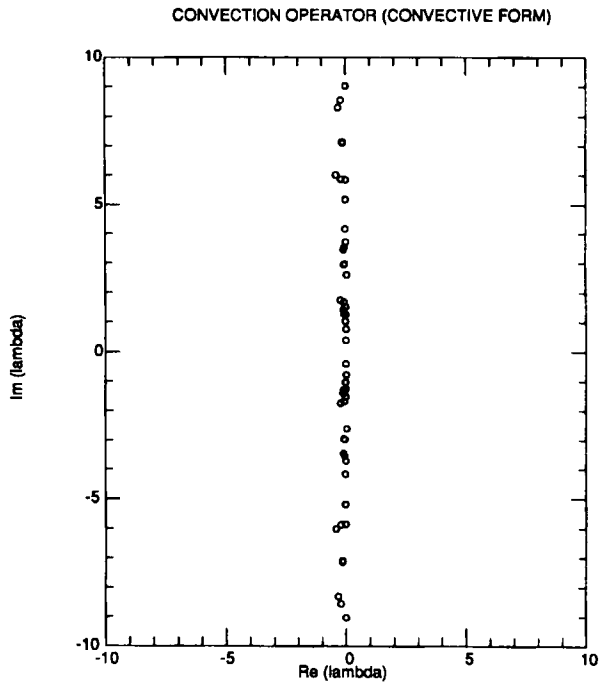


Figure 15. Eigenvalues of the discrete convection operator for two-dimensional channel flow. The results are for SECONF with $N_1 = N_2 = N^- = 4$ and $N_3 = N_4 = N^+ = 5$

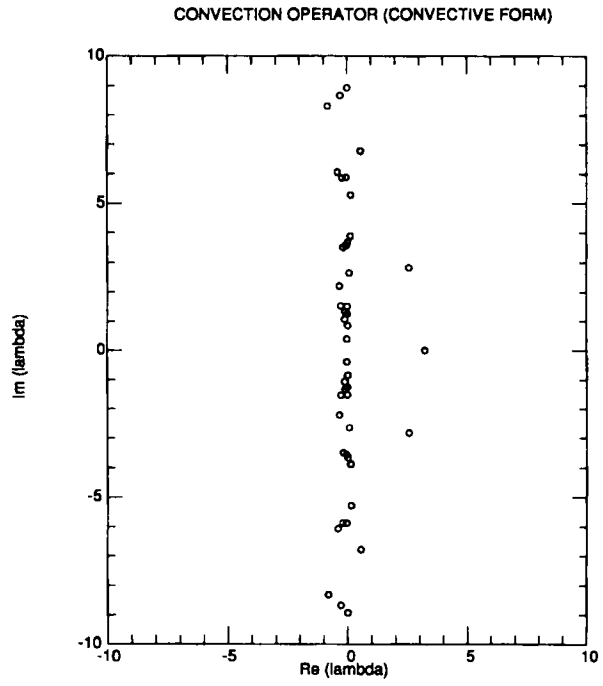


Figure 16. Eigenvalues of the discrete convection operator for two-dimensional channel flow. The results are for SENONC with $N_1 = N_2 = N^- = 4$ and $N_3 = N_4 = N^+ = 5$

Figures 17 and 18. In particular we notice that the spectra in Figures 17 and 18 are the mirror images of the spectra in Figures 15 and 16 around the imaginary axis.

Finally we remark that the above results also explain the fact that no stability problems are observed when solving the discrete equations using a fully coupled, fully implicit approach.

4. CONCLUSIONS

In this study we have solved convection–diffusion problems using spectral elements of equal or different order. We have compared the use of conforming and non-conforming methods as well as the use of a convective form and a skew-symmetric form of the discrete convection operator. Both steady and unsteady problems have been solved numerically.

For *steady* convection–diffusion problems the convective form of the convection operator seems to be the best choice given the fact that this is the simplest form in terms of evaluation and implementation, it gives good accuracy and it seems to be stable for the conforming and non-conforming methods investigated in this work (regardless of flow direction).

Earlier work indicated that the skew-symmetric form is more accurate than other forms of the discrete convection operator. For one of the test problems presented in this study (the Kovasznay problem), some of the results indicate that the convective form and the skew-symmetric form give comparable errors for the same spatial resolution. However, some of the results also indicate that the skew-symmetric form can be substantially *less* accurate than the convective form in the case of inexact representation of the velocity (Dirichlet) boundary conditions. Fortunately, the specification of the velocity on part of the (or the whole) domain boundary typically consists of the specification of a relatively smooth function (e.g. constant, parabolic, etc.). Hence, in many practical cases, the velocity boundary conditions are well resolved and the difference in accuracy between the convective form and the skew-symmetric form might not be significant.

For *unsteady* problems, our initial experience has been that care has to be taken when using a non-conforming method, particularly for convection-dominated problems. The numerical results indicate that if the flow is going from a lower-order element into a higher-order element, and if a convective form of the convection operator is used, instabilities can occur with an *explicit* treatment of the convection operator regardless of the choice of time step. In contrast, the numerical results indicate that the combination of a skew-symmetric form and explicit time integration schemes is conditionally stable (CFL condition) in the context of using both conforming and non-conforming methods.

If *implicit* time-stepping procedures are used, the results indicate that both the convective form and the skew-symmetric form will work well in combination with both conforming and non-conforming discretizations.

The three variants of the spectral element method that we have used in this study can all be regarded as special cases of the mortar element method. We are ultimately interested in developing fast and robust domain decomposition algorithms where the full generality of the mortar element method is exploited. Since the general case comprises both geometrically conforming and non-conforming discretizations, the general case dictates the need for robust convection treatment for non-conforming discretizations. If the conclusions from this study also apply to the general case, a fully non-conforming discretization based on the mortar element method could potentially be used with success for solving steady and unsteady convection-dominated problems. Based on the results from this study, we would favour the use of a convective form of the convection operator and implicit time-stepping techniques for unsteady problems.

We hope that the results presented in this study will be followed up with a theoretical analysis in order to confirm the findings, to better understand the results and perhaps also to construct improved high-order discretization methods for convection–diffusion problems.

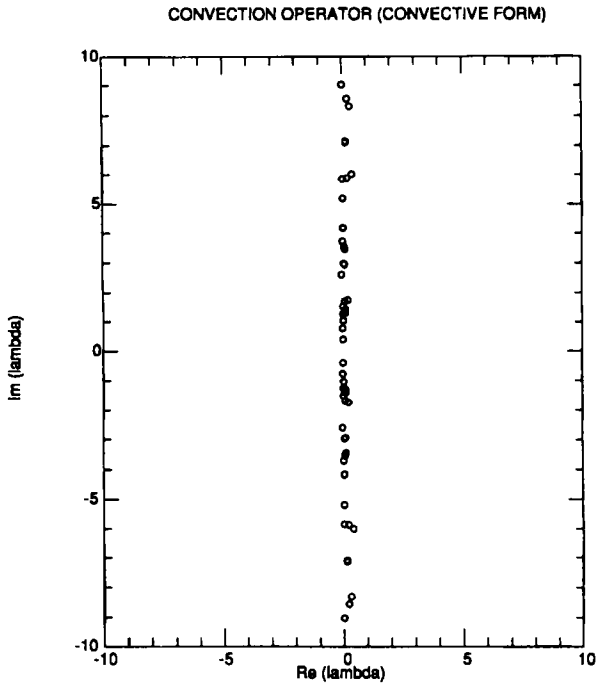


Figure 17. Eigenvalues of the discrete convection operator for two-dimensional channel flow. The results are for SECONF with $N_1 = N_2 = N^- = 5$ and $N_3 = N_4 = N^+ = 4$

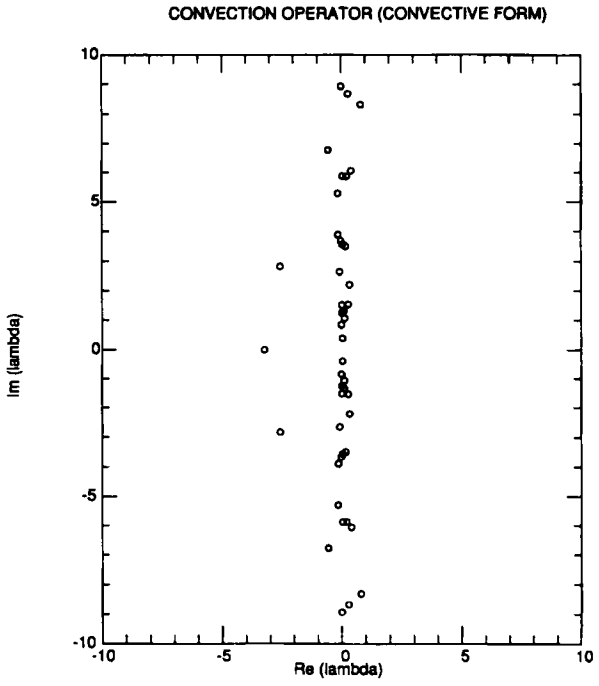


Figure 18. Eigenvalues of the discrete convection operator for two-dimensional channel flow. The results are for SENONC with $N_1 = N_2 = N^- = 5$ and $N_3 = N_4 = N^+ = 4$

ACKNOWLEDGEMENTS

The author would like to thank NASA Goddard Space Flight Center for the financial support of most of this work (contract NAS5-32432). In addition, the author would like to thank Fluent, Inc. for providing the CORTEX user interface and Dr Edward T. Bullister for helping out with the visualization of the computational results. Part of this work was supported by Nektonics, Inc. The author would also like to thank the referees for their constructive criticism and Professors Yvon Maday and Anthony T. Patera for their valuable comments on an earlier version of the manuscript.

REFERENCES

1. C. Bernardi, Y. Maday and A. T. Patera, 'A new nonconforming approach to domain decomposition; the mortar element method', Publications du Laboratoire d'Analyse Numerique, *No. R 89027*, Universite Pierre et Marie Curie, 1990.
2. F. B. Belgacem and Y. Maday, 'A spectral element methodology tuned to parallel implementations', *Comput. Methods Appl. Mech. Eng.*, **116**, 59–67 (1994).
3. I. Babuska and M. Suri, 'The *p*- and *h–p* version of the finite element method, an overview', *Comput. Methods Appl. Mech. Eng.*, **80**, 5–26 (1990).
4. J. T. Oden and L. Demkowicz, '*h–p* adaptive finite element methods in computational fluid dynamics', *Comput. Methods Appl. Mech. Eng.*, **89**, 11–40 (1991).
5. J. T. Oden, L. Demkowicz and W. Rachowicz, 'A new finite element method for solving compressible Navier–Stokes equations based on an operator splitting method and *h–p* adaptivity', *Comput. Methods Appl. Mech. Eng.*, **84**, 275–326 (1990).
6. A Safjan, L. Demkowicz and J. T. Oden, 'Adaptive finite element methods for hyperbolic systems with application to transient acoustics', *Int. j. numer. methods eng.*, **32**, 677–707 (1991).
7. J. T. Oden, 'Theory and implementation of high-order adaptive *h–p* methods for the analysis of incompressible viscous flows', in S. N. Atluri (ed.), Computational Nonlinear Mechanics in Aerospace Engineering, AIAA Progress in Aeronautics and Astronautics Series, Vol. 146, AIAA, New York, 1992, Chap. 9, pp. 321–363.
8. J. T. Oden, W. Wu and M. Ainsworth, 'An *a posteriori* error estimate for finite element approximations of the Navier–Stokes equations', *Comput. Methods Appl. Mech. Eng.*, **111**, 185–202 (1994).
9. I. Babuska and M. Suri, 'The *p* and *h–p* versions of the finite element method, basic principles and properties', *SIAM Rev.*, **36**, 578–632 (1994).
10. A. T. Patera, 'A spectral element method for fluid dynamics, laminar flow in a channel expansion', *J. Comput. Phys.*, **54**, 468–488 (1984).
11. Y. Maday and A. T. Patera, 'Spectral element methods for the Navier–Stokes equations', in A. K. Noor (ed.), State of the Art Surveys in Computational Mechanics, ASME, New York, 1989, pp. 71–143.
12. P. F. Fischer and A. T. Patera, 'Parallel spectral element solution to the Stokes problem', *J. Comput. Phys.*, **92**, 380–421 (1991).
13. I. Babuska, B. A. Szabo and I. N. Katz, 'The *p*-version of the finite element method', *SIAM J. Numer. Anal.*, **18**, 515–545 (1981).
14. I. Babuska and M. R. Dorr, 'Error estimates for combined *h* and *p* versions of the finite element method', *Numer. Math.*, **37**, 257–277 (1981).
15. G. Strang and G. Fix, *An Analysis of the Finite Element Method*, Prentice-Hall, Englewood Cliffs, NJ, 1973.
16. S. A. Orszag, 'Spectral methods for problems in complex geometries', *J. Comput. Phys.*, **37**, 70–92 (1980).
17. P. F. Fischer and E. M. Rønquist, 'Spectral element methods for large scale parallel Navier–Stokes calculations', *Comput. Methods Appl. Mech. Eng.*, **116**, 69–76 (1994).
18. L. W. Ho and A. T. Patera, 'Variational formulation of three-dimensional viscous free-surface flows: natural imposition of surface tension boundary conditions', *Int. j. numer. methods fluids*, **13**, 691–698 (1991).
19. L. W. Ho and E. M. Rønquist, 'Spectral element solution of steady incompressible viscous free-surface flows', *Finite Elements Analysis Design*, **16**, 207–227 (1994).
20. G. E. Karniadakis, S. A. Orszag, E. M. Rønquist and A. T. Patera, 'Spectral element and lattice gas methods for incompressible fluid dynamics', in M. D. Gunzburger and R. A. Nicolaides (eds), *Incompressible Computational Fluid Dynamics*, Cambridge University Press, Cambridge, 1993, pp. 203–266.
21. Y. Maday, D. Meiron, A. T. Patera and E. M. Rønquist, 'Analysis of iterative methods for the steady and unsteady Stokes problem: application to spectral element discretizations', *SIAM J. Sci. Stat. Comput.*, **14**, 310–337 (1993).
22. Y. Maday, A. T. Patera and E. M. Rønquist, 'The $P_N \times P_{N-2}$ method for the approximation of the Stokes problem', Numer. Math., in press.
23. V. Van Kemenade and M. O. Deville, 'Application of spectral elements to viscoelastic creeping flows', *J. Non-Newtonian Fluid Mech.*, in press.
24. G. Anagnostou, Y. Maday, C. Mavriplis and A. T. Patera, 'On the mortar element method; generalizations and implementation', in T. F. Chan, R. Glowinski, J. Periaux and O. B. Widlund (eds), *Proc. 3rd Int. Symp. on Domain Decomposition Methods*, SIAM, Philadelphia, PA, 1989, pp. 157–173.

25. C. Mavriplis, 'Adaptive mesh strategies for the spectral element method', *Comput. Methods Appl. Mech. Eng.*, **116**, 77–86 (1994).
26. R. D. Henderson and G. E. Karniadakis, 'Unstructured spectral element methods for the incompressible Navier–Stokes equations', in *Proc. Finite Elements in Fluids: New Trends and Applications*, eds. K. Morgan, E. Onate, J. Perriaux and O. C. Zienkewicz, Pineridge Press, 1993.
27. C. Bernardi, N. Debit and Y. Maday, 'Coupling spectral and finite element methods: first results', *Math. Comput.*, **54**, 21 (1990).
28. L. Demkowicz, J. T. Oden, W. Rachowicz and O. Hardy, 'Toward a universal *h–p* adaptive finite element strategy, Part 1. Constrained approximation and data structure', *Comput. Methods Appl. Mech. Eng.*, **77**, 79–112 (1989).
29. E. M. Rønquist, 'Optimal spectral element methods for the unsteady three-dimensional incompressible Navier–Stokes equations', *Ph.D. Thesis*, Massachusetts Institute of Technology, 1988.
30. E. M. Rønquist, 'A domain decomposition solver for the steady Navier–Stokes equations', in *Proc. 3rd Int. Conf. on Spectral and High-order Methods*, Houston, Texas, June 1995, ed. L. Ridgway Scott, in preparation.
31. C. Bernardi, Y. Maday and B. Metivet, 'Spectral approximation of the periodic–nonperiodic Navier–Stokes equations', *Numer. Math.*, **51**, 655–700 (1987).
32. K. Horiuti, 'Comparison of conservative and rotational forms in large eddy simulation of turbulent channel flow', *J. Comput. Phys.*, **71**, 343–370 (1987).
33. T. A. Zang, 'On the rotation and skew-symmetric forms for incompressible flow simulations', *Appl. Numer. Math.*, **7**, 27–40 (1991).
34. P. M. Gresho, 'Incompressible fluid dynamics: some fundamental formulation issues', *Ann. Rev. Fluid. Mech.*, **23**, 413–453 (1991).
35. L. I. G. Kovasznay, 'Laminar flow behind a two-dimensional grid', *Proc. Camb. Philos. Soc.*, **44**, 58–62 (1948).
36. Y. Saad and M. H. Schultz, 'GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems', *SIAM J. Sci. Stat. Comput.*, **7**, 856–869 (1986).
37. L. F. Pavarino and O. B. Widlund, 'Iterative substructuring methods for spectral elements in three dimensions', in M. Krizek, P. Neittaanmaki and R. Stenberg (eds), *The Finite Element Method: Fifty Years of the Courant Element*, Marcel Dekker, New York, 1994, pp. 345–355.
38. Y. Maday, A. T. Patera and E. M. Rønquist, 'An operator–integration-factor splitting method for time-dependent problems: application to incompressible fluid flow', *J. Sci. Comput.*, **5**, 263–292 (1991).
39. C. W. Gear, *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1971.
40. O. Pironneau, 'On the transport-diffusion algorithm and its application to the Navier–Stokes equations', *Numer. Math.*, **38**, 309–332 (1982).