Journal of Scientific Computing, Vol. 6, No. 2, 1991

Hybrid Spectral-Element–Low-Order Methods for Incompressible Flows

Ron Henderson and George Em. Karniadakis[1]

Received June 26, 1991

In this article we present a new formulation for coupling spectral element discretizations to finite difference and finite element discretizations addressing flow problems in very complicated geometries. A general iterative relaxation procedure (*Zanolli patching*) is employed that enforces $\mathscr{C}^1$ continuity along the patching interface between the two differently discretized subdomains. In fluid flow simulations of transitional and turbulent flows the high-order discretization (spectral element) is used in the outer part of the domain where the Reynolds number is effectively very high. Near "rough" wall boundaries (where the flow is effectively very viscous) the use of low-order discretizations provides sufficient accuracy and allows for efficient treatment of the complex geometry. An analysis of the patching procedure is presented for elliptic problems, and extensions to incompressible Navier–Stokes equations are implemented using an efficient high-order splitting scheme. Several examples are given for elliptic and flow model problems and performance is measured on both serial and parallel processors.

KEY WORDS: Incompressible flows; complex geometries; spectral elements; finite differences, finite elements.

1. INTRODUCTION

Over the last two decades, a large number of numerical techniques have been proposed for the solution of the incompressible Navier–Stokes equations. Traditionally, these techniques have fallen into two distinct categories: low-order methods, such as finite differences and finite elements, which may be easily applied to geometrically complex computational domains but suffer from slow, algebraic-type convergence; and high-order

[1] Department of Mechanical and Aerospace Engineering, Program in Applied and Computational Mathematics, Princeton University, Princeton, New Jersey 08544.

854/6/2-1

(spectral) methods based on Fourier or orthogonal polynomial expansions which achieve exponentially fast convergence but are typically limited to simple and smooth domains. Although the differences among these discretization techniques might have initially been very clear, there has been an increasing trend (especially this past decade) toward construction of hybrid algorithms with components that exhibit different properties but typically share a common root. A typical example of such a confluence of numerical algorithms is the spectral element method (Patera, 1984; Karniadakis *et al.*, 1985), which is based on two weighted-residual techniques: finite element and spectral methods. The combination of spectral accuracy with the flexibility in handling complex geometries have made the method quite successfull in a number of applications in fluid dynamics, including simulations of flows in the transitional and turbulent regimes (Karniadakis, 1989). However, a straightforward application of the method in simulating turbulent flows with a very strong disparity in length scales (e.g., fluid flows over arbitrarily roughened surfaces) is prohibitively expensive as the small-scale geometric irregularity imposes severe constraints that result in an extremely low convergence rate for the method.

At the same time, advances in computer hardware as well as the introduction of new parallel supercomputers have forced a revaluation of traditional algorithms. This new class of machines offers the possibility of performing simulations of turbulent flows in regimes and domains that have not been investigated before, i.e., high Reynolds number and complex geometry flows. Another aspect to the hardware issue is the development of specialized computers geared toward a particular numerical algorithm or class of algorithms, such as the RAP1 cellular automaton machine (Clouquer and d'Humières, 1987) or the Navier–Stokes computer (Nosenchuck and Littman, 1986). Such advances in computer technology have made possible the concept of a coprocessor to extend the capabilities of a larger simulation code by providing an environment where not only hybrid algorithms but also *hybrid architectures* may be combined in a way that is in some sense "optimal." This again leads back to the question of how to couple these hybrid schemes efficiently.

In this work, we propose a new class of efficient hybrid discretization schemes appropriate for simulating flows over walls of arbitrary roughness (see Fig. 1). The two main components of the algorithm are a high-order scheme (spectral element method) and a low-order scheme (finite difference or finite element method). The use of the finite difference discretization is essential in geometries with random boundaries, where all discretization techniques based on conformal or isoparametric mappings fail. The use of low-order finite elements can also be useful in a wider class of applications including, for example, flows in unbounded domains, flows over surfaces

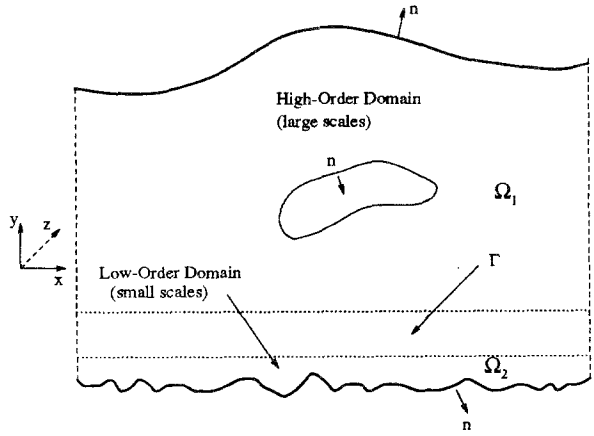

Fig. 1. Geometry definition and computational subdomains for the model flow problem.

with distributed roughness elements, etc. A new general iterative relaxation procedure is applied to allow coupling of two fundamentally different discretizations. In particular, the first component of the hybrid algorithm (spectral element method) is applied to the outer large-scale domain Ω_1, where the effective local Reynolds number is large and thus the spectrallike dispersive properties of the method are effectively utilized. In the near-wall region where an almost laminar flow prevails the second component (a low-order method) is applied providing sufficient resolution to simulate the viscous flow and account for the small-scale irregularity of the domain Ω_2. As regards time discretization, an efficient high-order splitting scheme (Karniadakis *et al.*, 1991) is employed that reduces the problem into solving a series of coupled hyperbolic and elliptic problems. Continuity of the solution along the spectral-element–finite-difference (or finite element) interface (boundary Γ in Fig. 1) is then imposed by requiring continuity of the elliptic components; the latter is accomplished using the iterative "Zanolli" patching procedure and appropriately chosen relaxation parameters (Funaro *et al.*, 1985).

The "Zanolli" patching procedure has been practiced in the past only in the context of similar discretizations on both domains [i.e., spectral collocation (Funaro *et al.*, 1985)]. It basically consists of solving a Dirichlet elliptic problem in domain Ω_1 and subsequently providing a pointwise flux (Neumann) condition for the solution of the corresponding elliptic problem in domain Ω_2; this procedure is then repeated until continuity of the two solutions at the interface is achieved. Convergence to the exact solution is typically obtained after three to five iterations depending on the problem size and the value of the relaxation parameter.

In the current work, we have modified this patching procedure first to accommodate dissimilar discretization schemes across the two domains, and second to allow for a parallel implementation; the latter can be achieved by appropriately modifying the flux condition of the Neumann elliptic problem.

The article is organized as follows: In Sec. 2, we review and compare through model problems three different procedures for interfacing subdomains in solving second-order elliptic problems: the spectral element method, the alternating Schwarz algorithm, and the Zanolli procedure both in its sequential and parallel form. In Sec. 3, we extend the algorithm to the incompressible Navier–Stokes equations by employing a recently proposed high-order splitting scheme. We then apply the hybrid scheme to nonconforming spectral element simulation of an exact Navier–Stokes solution where the spectral (exponential) convergence is verified; subsequently a time-dependent flow example is simulated in an irregular domain. A brief discussion and conclusions are summarized in Sec. 4.

2. PATCHING PROCEDURES

We consider here numerical solution of a second-order Helmholtz equation of the form

$$\nabla^2\phi - \mu^2\phi = f, \qquad \phi \in \Omega \tag{2.1}$$

with either Dirichlet or Neumann boundary conditions applied on the boundary $\partial\Omega$. This elliptic partial differential equation naturally forms a model problem with which to work as it retains the most important features of the governing Navier–Stokes equations, i.e., the highest derivatives which dictate continuity requirements, stability, and appropriate approximation spaces; moreover computationally it is the most expensive component. Having defined the model problem we now proceed with discretization of (2.1) in a domain subdivided into subdomains using three different procedures: a spectral element method, an alternating Schwarz method, and the "Zanolli" method. While the first approach is only appropriate for subdomains with spectral discretizations, the two other methods are more general and can interface domains with different discretizations, e.g., spectral and finite difference discretizations.

2.1. Spectral Element Method

We first consider the simple one-dimensional version of (2.1) to review some basic steps in the spectral element discretization. Taking $\mu = 0$ the governing equation is

$$\phi_{xx} = f, \qquad x \in \Lambda \equiv [-1, 1] \tag{2.2}$$

with homogeneous Dirichlet boundary conditions:

$$\phi(-1)=\phi(1)=0 \tag{2.3}$$

We solve this equation using a weak form and following a variational approach, i.e.,

$$\alpha(\phi, \psi)=-(f, \psi), \qquad \forall \psi \in \mathscr{H}_0^1(\Lambda) \tag{2.4}$$

where we define the inner products as follows:

$$\begin{aligned}(\phi, \psi)&=\int_\Lambda \phi(x)\,\psi(x)\,dx, \qquad \forall \phi, \psi \in \mathscr{L}^2(\Lambda)\\ \alpha(\phi, \psi)&=\int_\Lambda \phi_x(x)\,\psi_x(x)\,dx, \qquad \forall \phi, \psi \in \mathscr{H}_0^1(\Lambda)\end{aligned} \tag{2.5}$$

The function spaces $\mathscr{L}^2(\Lambda)$ and $\mathscr{H}_0^1(\Lambda)$ are the standard spaces of square-integrable functions and derivatives respectively, i.e.,

$$\begin{aligned}\mathscr{L}^2(\Lambda)&=\left[\psi: \int_\Lambda \psi^2\,dx<\infty\right]\\ \mathscr{H}_0^1(\Lambda)&=[\psi: \psi \in \mathscr{L}^2(\Lambda),\ \psi_x \in \mathscr{L}^2(\Lambda),\ \psi(-1)=\psi(1)=0]\end{aligned} \tag{2.6}$$

The spectral element discretization proceeds by breaking up the domain into K subintervals Λ_k: $\bigcup_{k=1}^K \Lambda_k=\Lambda$, which define the macro-elements. There is also another resolution parameter N, which specifies the interpolation order within each element. The approximation space denoted by $\mathscr{X}_h$ here is a subspace of $\mathscr{H}_0^1$ consisting of all piecewise high-order polynomials of degree up to N. The discrete solution is sought therefore in $\mathscr{X}_h$, and thus (2.4) is replaced by its discrete analog written in terms of the discretization parameter $h \equiv (K, N)$:

$$\alpha_{h,\mathrm{GL}}(\phi_h, \psi)=-(f, \psi)_{h,\mathrm{GL}}, \qquad \forall \psi \in \mathscr{X}_h \tag{2.7}$$

where the subscript GL refers to the numerical Gauss–Lobatto quadrature replacing the integrals in (2.5). The discrete inner products are therefore defined by

$$(\phi, \psi)_{h,\mathrm{GL}}=\frac{l^k}{2}\sum_{k=1}^K \sum_{n=0}^N \rho_n \phi(\xi_n^k)\,\psi(\xi_n^k) \tag{2.8}$$

$$\alpha_{h,\mathrm{GL}}(\phi, \psi)=\frac{l^k}{2}\sum_{k=1}^K \sum_{n=0}^N \rho_n \phi_x(\xi_n^k)\,\psi_x(\xi_n^k) \tag{2.9}$$

where l^k is the length of the kth element, ρ_n are quadrature weights, and ξ_n^k are the corresponding global quadrature points defined in terms of the left and right global coordinates of the end points of the element, i.e.,

$$\xi_n^k = x_L^k + (\xi_n + 1) l^k/2 \qquad \forall 0 \leqslant n \leqslant N, \qquad 1 \leqslant k \leqslant K \tag{2.10}$$

Multidomain spectral methods preserve spectral accuracy as N increases; to see this, we determine the magnitude of the error $\|\phi - \phi_h\|_1$ as N goes to infinity with K (the number of elements) held constant. Expressions for the approximation as well as the interpolation error are given in Maday and Patera (1989); the following error bound is valid assuming that the continuous solution $\phi(x) \in \mathscr{H}^\sigma(\Lambda) \cap \mathscr{H}_0^1(\Lambda)$, and that the forcing has a regularity of order ρ [i.e., $f \in \mathscr{H}^\rho(\Lambda)$],

$$\|\phi - \phi_h\|_1 \leqslant C[N^{1-\sigma} \|\phi\|_\sigma + N^{-\rho} \|f\|_\rho] \tag{2.11}$$

where C is a constant independent of the discretization. This error estimate is independent of the basis selected; however, the choice of basis reflects the sparsity and ease in the implementation of the discrete system. The basis introduced first by Patera (1984) corresponds to a Chebyshev–Lagrangian interpolation, while most recent work employs Legendre–Lagrangian interpolants (Maday and Patera, 1989). In the general case we can therefore write

$$\psi_N^k(x) = \sum_{i=0}^{N} \psi_i^k h_i(r), \qquad x \in \Lambda^k \Rightarrow r \in \Lambda \tag{2.12}$$

where ψ_i^k is the nodal value of the unknown at point i in element k, and the interpolants are defined from $h_i(\xi_j) = \delta_{ij}$, and δ_{ij} is the Kronecker delta symbol.

For the variational formulation adopted in the spectral element analysis an $\mathscr{H}^1$ continuity condition should be honored at interfacial boundaries. For the discrete system this is accomplished by the requirement

$$\psi_N^k = \psi_0^{k+1} \qquad \forall k = 1, \ldots, K-1 \tag{2.13}$$

while the homogeneous boundary conditions (2.3) are imposed by requiring

$$\psi_0^1 = \psi_N^K = 0 \tag{2.14}$$

These two conditions are imposed in practice using direct stiffness summation, in which contributions from local nodes that are physically coincident

are summed, while contributions from nodes located at the domain boundary are masked to zero. Introducing a notation $\sum_k'$ for such an operation, we can then write the discrete one-dimensional equations after inserting (2.8) and (2.9) and (2.12) into (2.7),

$$\sum_{k=1}^{K}{}' \sum_{i=0}^{N} \sum_{j=0}^{N} A_{ij}^k \phi_j^k = \sum_{k=0}^{K}{}' \sum_{i=0}^{N} \sum_{j=0}^{N} B_{ij}^k f_j^k \tag{2.15}$$

where the forcing on the right-hand side is interpolated as in (2.12) (this interpolation is also bounded spectrally). Here, the stiffness matrix A_{ij}^k and the mass matrix B_{ij}^k are defined as

$$A_{ij}^k = \frac{2}{l^k} \sum_{n=0}^{N} \rho_n D_{ni} D_{nj}, \qquad \text{where} \quad D_{ij} = \left.\frac{dh_j}{dr}\right|_{\xi_i} \tag{2.16}$$

$$B_{ij}^k = \frac{l^k}{2} \rho_i \delta_{ij} \tag{2.17}$$

The mass matrix is therefore diagonal due to the fact that the quadrature points coincide with the collocation points in each element, and thus the interpolation between the two sets of points in the identity operator. For two-dimensional extensions the reader is to referred to Patera (1984), Karniadakis *et al.* (1985), Karniadakis (1989), Maday and Patera (1989), and Gottlieb and Orszag (1977).

One drawback of the spectral element method is that each element must be mapped isoparametrically to a smooth, rectangular or cubic domain in two and three dimensions, respectively, thus precluding its use in geometries with random or small-scale roughness. In this case, we propose to use low-order methods (e.g., finite difference discretizations). Although low-order schemes achieve only algebraic convergence, the computational work associated with them is typically $\mathcal{O}(N^d)$ (where d is the dimension) in contrast to $\mathcal{O}(N^{d+1})$ for spectral methods. Also, because of the local character of finite differences, techniques such as block gridding may be used more effectively to describe highly irregular boundaries. In their use here, finite difference domains may be thought of as local refinement patches which extend the capabilites of the spectral element solver by offering a new type of boundary condition, bridging the gap between the physical boundary and the one numerically imposed by the limitations of the spectral element method. The reader is referred to standard references such as Roache (1982) for a further description of finite difference techniques.

2.2. Alternating Schwarz Method

The classical alternating Schwarz method is a technique for coupling domains that have some amount of overlap. First introduced as a means of proving the existence of solutions to elliptic boundary value problems, the method gained new popularity dcuring recent years for computations on parallel computers and has also been used as a means of constructing composite grids for Navier–Stokes calculations as introduced by van der Wijngaart (1990).

2.2.1. Description of the Algorithm

For simplicity, the algorithm will be described for the one-dimensional case depicted in Fig. 2. We seek a solution on the domain $\Omega(a, b)$ that satisfies the differential equation

$$\phi_{xx} - \mu^2 \phi = f, \qquad \phi \in \Omega(a, b) \tag{2.18}$$

with homogeneous boundary conditions

$$\phi(a) = \phi(b) = 0 \tag{2.19}$$

The global domain is to be partitioned into two subdomains $\Omega_1(a, \beta)$ and $\Omega_2(\alpha, b)$ where $\delta = \beta - \alpha$ is the length of the overlap region. The solution will be computed iteratively, and ϕ_i^n denotes the restriction of the solution to domain Ω_i at iteration n.

Choosing an arbitrary initial guess for the solution $\phi \in \Omega$, we compute the sequence of functions $\{\phi_1^n\} \in \Omega_1$ and $\{\phi_2^n\} \in \Omega_2$ for $n \geqslant 1$ as

$$\phi_1^n(\beta) = \phi_2^{n-1}(\beta)$$
$$\phi_{1,xx}^n - \mu^2 \phi_1^n = f \qquad \text{in } \Omega_1 \tag{2.20}$$

$$\phi_2^n(\alpha) = \phi_1^n(\alpha)$$
$$\phi_{2,xx}^n - \mu^2 \phi_2^n = f \qquad \text{in } \Omega_2 \tag{2.21}$$

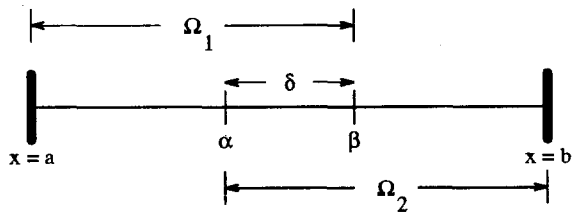

Fig. 2. Domain partitioning using the alternating Schwarz method.

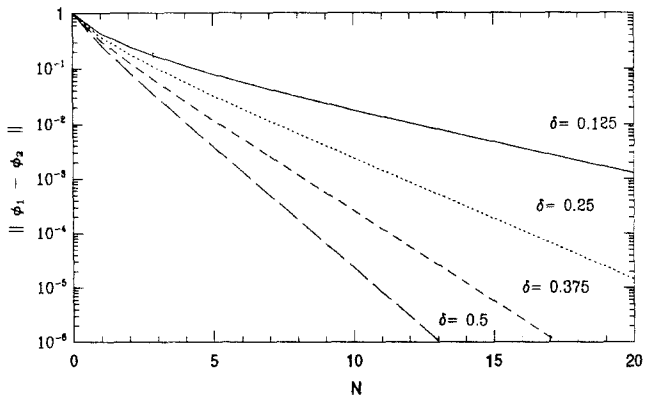

Fig. 3. Convergence rates for a mixed spectral–finite-difference approximation. Here, N is the global iteration number and δ refers to the size of the overlap between the two subdomains $\Omega_1(-1, \delta/2)$ and $\Omega_2(-\delta/2, +1)$.

As $n \to \infty$ this sequence converges geometrically to the solution ϕ of (2.18), and the convergence rate is a monotonically decreasing function of the overlap δ; for a more precise explanation and proofs of this statement see Canuto and Funaro (1987). This performance is illustrated in Fig. 3 for the case of a spectral approximation to (2.20) and a finite-difference approximation to (2.21) for a simple one-dimensional problem. It has been previously shown (Canuto and Funaro, 1987) that the Schwarz method achieves exponential convergence in the case of spectral collocation approximations to (2.20) and (2.21), and in Fig. 4 we see that similar behavior is observed even for mixed discretizations.

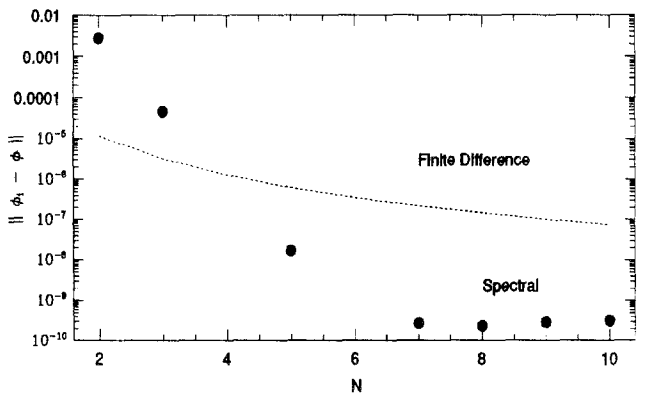

Fig. 4. Convergence to the exact solution for a mixed spectral–finite-difference approximation with respect to the polynomial order (N) and the number of finite difference points ($5N$).

The procedure of (2.20) and (2.21) may be parallelized with little effect on the convergence rate. However, because the convergence rate is very slow as $\delta \to 0$ the method is inefficient as acceptable preformance requires a significant portion of the global domain to be duplicated. Additionally, the need to perform interpolation on the interior of subdomains translates into a possibly heavier workload for transmission of data. As a final remark on the Schwarz method, we note that the convergence behavior seen in Fig. 4 must be retained by any viable patching method as it justifies the heavier workload associated with the high-order method. It should be mentioned that recent modifications to the classic Schwarz algorithm, such as those introduced by Lions (1989) and Douglas (1989), may help overcome some of these difficulties.

2.3. Zanolli Patching

Another approach to domain decomposition is to eliminate overlapping subdomains and perform matching of the solution and its derivatives along a single interface. One such procedure was introduced in 1985 by Funaro, Quarteroni, and Zanolli (1985) and is referred to here as "Zanolli" patching. The method consists of solving a sequence of alternating Dirichlet/Neumann problems, maintaining the continuity of derivatives at all times while relaxing function values to achieve $\mathscr{C}^0$ continuity to within some predefined tolerance; this procedure generally results in very fast convergence.

2.3.1. Description of the Sequential Algorithm

In applying the Zanolli algorithm, the global domain $\Omega(a, b)$ is partitioned into two nonoverlapping subdomains $\Omega_1(a, \delta)$ and $\Omega_2(\delta, b)$, δ denoting the location of the patched interface (Fig. 5). In comparison with the Schwarz algorithm, this is the extreme case where $\alpha = \beta$ and $\delta \to 0$. The solution will be computed iteratively, and ϕ_i^n denotes the restriction of the solution to domain Ω_i at iteration n. The differential equation and boundary conditions are the same as (2.18)–(2.19).

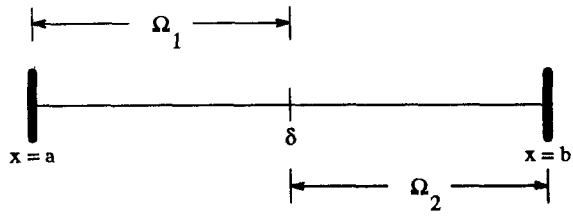

Fig. 5. Domain partitioning usinig the Zanolli patching procedure.

The Zanolli patching procedure is applied as follows. Choosing λ^1 as an initial guess for the solution on the interface δ, we now compute the sequence of functions $\{\phi_1^n\} \in \Omega_1$ and $\{\phi_2^n\} \in \Omega_2$ for $n \geqslant 1$ as

$$\begin{aligned} \phi_1^n(\delta) &= \lambda^n \\ \phi_{1,xx}^n - \mu^2 \phi_1^n &= f \qquad \text{in } \Omega_1 \end{aligned} \tag{2.22}$$

$$\begin{aligned} \phi_{2,x}^n(\delta) &= \phi_{1,x}^n(\delta) \\ \phi_{2,xx}^n - \mu^2 \phi_2^n &= f \qquad \text{in } \Omega_2 \end{aligned} \tag{2.23}$$

where subsequent λ^n's are computed as

$$\lambda^{n+1} = \theta \cdot \phi_2^n(\delta) + (1-\theta) \cdot \lambda^n \tag{2.24}$$

In this context, θ is a *relaxation parameter* which under certain conditions guarantees that the procedure (2.22) and (2.23) will always converge.

One of the important results of Funaro *et al.* (1985) is a theoretical prediction of optimal θ's and a method for choosing θ *dynamically* so as to accelerate convergence. Defining error functions

$$\begin{aligned} e_1^n &= \phi_1^n - \phi_1^{n-1} \\ e_2^n &= \phi_2^n - \phi_2^{n-1} \\ z^n(\theta) &= \theta \cdot \phi_2^n + (1-\theta) \cdot \phi_1^n \end{aligned} \tag{2.25}$$

the unique real number θ which minimizes $\|z^n(\theta) - z^{n-1}(\theta)\|^2$ is given by

$$\theta^n = \frac{(e_1^n, e_1^n - e_2^n)}{\|e_1^n - e_2^n\|^2} \tag{2.26}$$

where $(\cdot, \cdot)$ is the normal inner product in $\mathscr{L}^2$ and $\|\cdot\|$ is the associated norm.

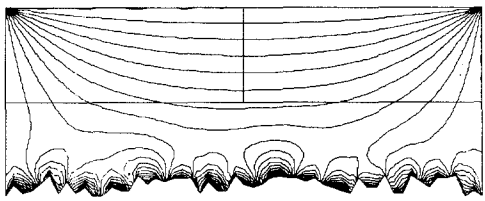

Fig. 6. Contours of a solution to the Helmholtz equation. The upper two rectangular regions are discretized using high-order spectral elements, while the lower region represents a finite difference grid.

Several examples of the performance of this algorithm for spectral collocation are given in Funaro *et al.* (1985). Here we demonstrate its effectiveness for two-dimensional problems with mixed discretizations by solving (2.1) in a complex domain (see Fig. 6). In Fig. 7 we show convergence of the solution at the interface for several values of θ, including the case where θ is updated dynamically. The performance seen here is typical, namely 5–10 iterations for convergence independent of the complexity of the the solution.

2.4. Description of the Parallel Algorithm

One of the drawbacks to the Zanolli procedure is that it is a "serial algorithm." Because of the coupling (through the derivative term) between subdomains the solution within each subdomain must be computed in a specific order, limiting the application of this procedure to sequential processing. A simple modification, which allows the computations to proceed *in parallel*, is

$$\begin{aligned} \phi_1^n(\delta) &= \theta \cdot \phi_2^{n-1}(\delta) + (1-\theta) \cdot \phi_1^{n-1} \\ \phi_{2,x}^n(\delta) &= \phi_{1,x}^{n-1}(\delta) \end{aligned} \tag{2.27}$$

$$\begin{aligned} \phi_{1,xx}^n - \mu^2 \phi_1^n &= f \qquad \text{in } \Omega_1 \\ \phi_{2,xx}^n - \mu^2 \phi_2^n &= f \qquad \text{in } \Omega_2 \end{aligned} \tag{2.28}$$

Note that (2.27) denotes a communication and (2.28) a computation step. While altering the convergence properties of the original scheme, this

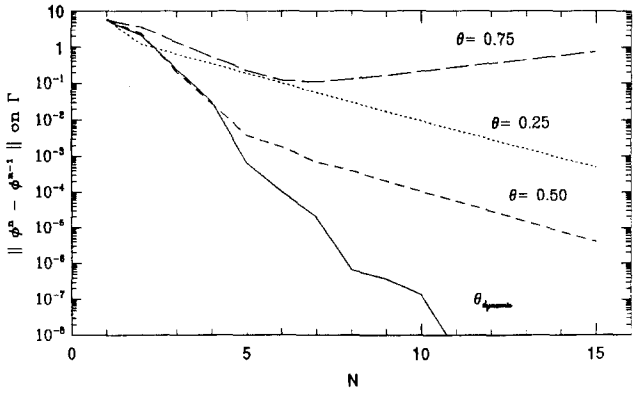

Fig. 7. Interface convergence for Zanolli patching. θ refers to the relaxation parameter applied to function values along the interface.

allows for each subdomain to be updated simultaneously. On a medium- to fine-grained machine, it also allows each Ω_i to be distributed over an appropriate collection of processors such that domains with differing workloads may be updated in the same amount of "real" time.

A simple convergence analysis is possible for the one-dimensional case. Defining error functions as in (2.25) we find that in the unrelaxed case ($\theta = 1$) the error at the interface obeys

$$e_1^n(\delta) = -\frac{b}{a} e_1^{n-1}(\delta) \tag{2.29}$$

Following the procedure outlined in Funaro *et al.* (1985) we can define an optimal θ which yields $e_1^3(\delta) = e_2^4(\delta) \equiv 0 \rightarrow$ convergence in 4 iterations. Figure 8 shows convergence rates for a typical two-dimensional configuration for various fixed value of θ, and Fig. 9 shows the corresponding rates for the parallel procedure. Note that in the unrelaxed case ($\theta = 1$) the parallel procedure indeed produces two uncoupled solutions which converge at approximately half the rate of the serial version. However, a value of θ exists which yields acceptable convergence at better than half the serial rate. This result suggests that the amount of work represented by individual subdomains should be significant for the parallel procedure to be efficient.

To illustrate the effects of distributed parallelism, we solve the same problem as in Fig. 6 by applying the procedure of (2.27) and (2.28) on an Intel iPSC/860 parallel supercomputer. In Fig. 10 we show parallel efficiency, defined as $\varepsilon = T_s/P \cdot T_p$, where T_s is the time to solve the

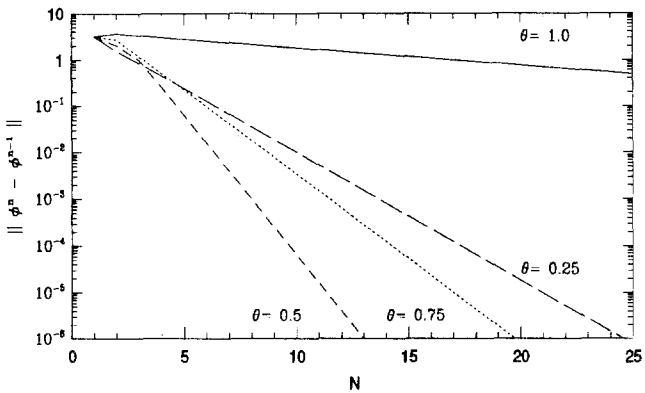

Fig. 8. Spectral-element–finite-difference convergence rates for the sequential Zanolli patching algorithm.

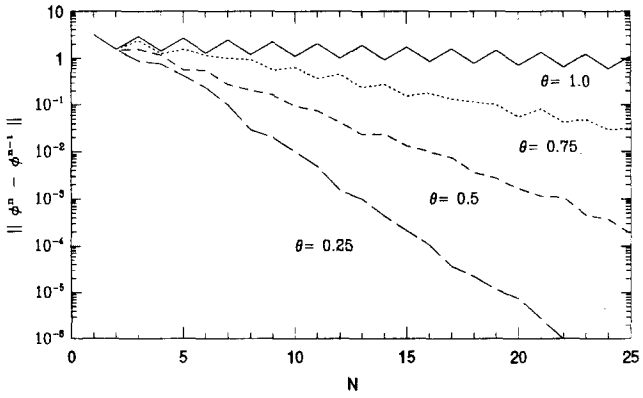

Fig. 9. Spectral-element–finite-difference convergence for the modified (parallel) Zanolli algorithm.

problem using the "best" sequential algorithm, and T_p is the time to solve the problem using the parallel algorithm on P processors. Figure 10 may be interpreted as follows: with only 2 processors, the computation may only proceed at the speed of the processor with the heavier workload, and since the amount of work associated with the high-order domain is much

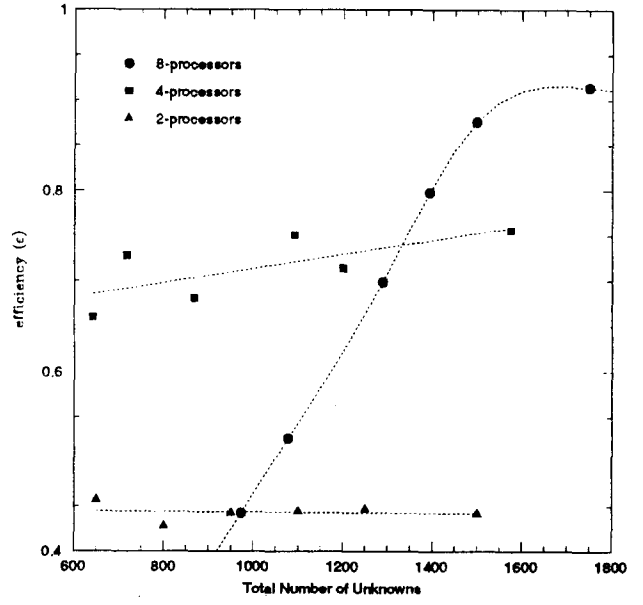

Fig. 10. Efficiency measurement for the parallel Zanolli patching algorithm applied to two subdomains with differing workloads on a Intel iPSC/860-32 hypercube.

larger than that of the low-order domain this translates into poor efficiencies. With a larger number of processors, resources within the machine may be distributed in such a way that each subdomain is updated in roughly the same amount of "real time" and the overall procedure achieves a speedup close to linear. This is confirmed in Fig. 10 by the computations with 4 and 8 processors.

3. NAVIER–STOKES ALGORITHM

In this section, we extend the iterative patching procedure to the incompressible Navier–Stokes equations for simulations of flows in arbitrarily complex domains. The governing equations for Newtonian fluids are

$$\begin{aligned} \frac{D\mathbf{v}}{Dt} &= -\frac{\nabla p}{\rho} + R^{-1}\nabla^2\mathbf{v} \qquad \text{in } \Omega \\ \nabla\cdot\mathbf{v} &= 0 \qquad \text{in } \Omega \end{aligned} \tag{3.1}$$

where $\mathbf{v}(\mathbf{x}, t)$ is the velocity field, p is the static pressure, R is the Reynolds number, ρ is the density, and D denotes total derivative.

3.1. High-Order Splitting Scheme

Numerical solution of the above system of equations will be obtained in the domains Ω_1 and Ω_2 shown in Fig. 1. Having defined the computational domain Ω we now proceed with the discretization of the system (3.1). The time-discretization employs a high-order splitting algorithm based on mixed stiffly stable schemes (Karniadakis *et al.*, 1991). Considering first the nonlinear terms we obtain

$$\frac{\hat{\mathbf{v}} - \sum_{q=0}^{J-1} \alpha_q \mathbf{v}^{n-q}}{\Delta t} = -\sum_{q=0}^{J-1} \beta_q \mathbf{N}(\mathbf{v}^{n-q}) \tag{3.2}$$

where $\mathbf{N}(\mathbf{v}^n) = \frac{1}{2}[\mathbf{v}^n\cdot\nabla\mathbf{v}^n + \nabla\cdot(\mathbf{v}^n\cdot\mathbf{v}^n)]$ represents the nonlinear contributions written in skew-symmetric form at time level $t = n\,\Delta t$, and α_q, β_q are implicit/explicit weight coefficients for the stiffy stable scheme of order J (see Karniadakis *et al.*, 1991). The next substep incorporates the pressure equation and enforces the incompressibility constraint as follows:

$$\begin{aligned} \hat{\hat{\mathbf{v}}} - \hat{\mathbf{v}} &= -\Delta t \nabla p^{n+1} \\ \nabla\cdot\hat{\hat{\mathbf{v}}} &= 0 \end{aligned} \tag{3.3}$$

Finally, the last substep includes the viscous corrections and the imposition of the boundary conditions, i.e.,

$$\frac{\gamma_0 \mathbf{v}^{n+1} - \hat{\hat{\mathbf{v}}}}{\Delta t} = R^{-1} \nabla^2 \mathbf{v}^{n+1} \tag{3.4}$$

where γ_0 is a weight-coefficient of the backwards differentiation scheme employed (Karniadakis *et al.*, 1991).

The above time treatment of the system of equation (3.1) results in a very efficient calculation procedure as it decouples the pressure and velocity equations as in (3.3) and (3.4). As regards time accuracy of this splitting scheme a key element in this approach is the specific treatment of the pressure equation (3.3), which can be recast in the form

$$\nabla^2 p^{n+1} = \nabla \cdot \left(\frac{\hat{\mathbf{v}}}{\Delta t} \right) \tag{3.5}$$

along with the consistent high-order pressure boundary condition (see Karniadakis *et al.*, 1991)

$$\frac{\partial p^{n+1}}{\partial n} = \mathbf{n} \cdot \left[- \sum_{q=0}^{J-1} \beta_q \mathbf{N}(\mathbf{v}^{n-q}) - R^{-1} \sum_{q=0}^{J-1} \beta_q [\nabla \times (\nabla \times \mathbf{v}^{n-q})] \right] \tag{3.6}$$

where $\mathbf{n}$ denotes the unit normal to the boundary $\partial\Omega$. Equations (3.3) and (3.5) therefore are Poisson equations with constant coefficients; the pressure equation, for example, can be rewritten in the standard form

$$\nabla^2 \phi = g(\mathbf{x}) \tag{3.7}$$

where we have defined $\phi = p^{n+1}$, and $g(\mathbf{x}) = \nabla \cdot (\hat{\mathbf{v}}/\Delta t)$. Standard spectral element and finite difference discretizations can then be applied in the two subdomains Ω_1 and Ω_2, while the Zanolli patching procedure is employed for both the pressure and the viscous correction terms. In this formulation (where the nonlinear terms are considered explicitly in time) the algorithms developed in the previous section are directly applicable.

3.2. Nonconforming Spectral Elements

We now consider the solution to (3.1) where the domain Ω is subdivided into two domains Ω_1 and Ω_2 and spectral element discretizations applied in both. By applying the Zanolli patching procedure, we can relax the usual constraint of physically coincident collocation points along element boundaries and consider both nonconforming elements and poly-

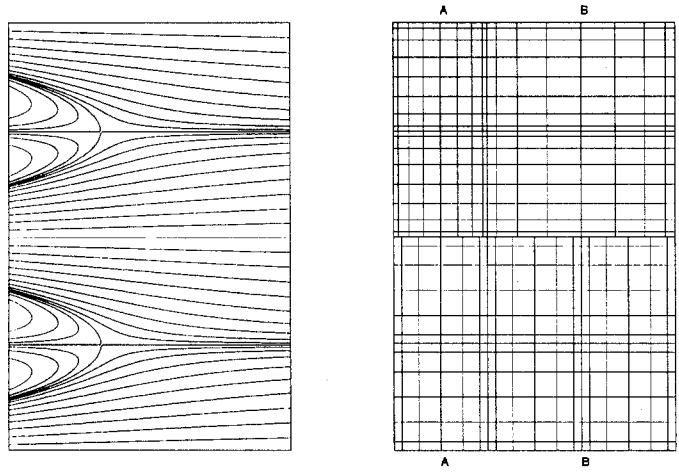

Fig. 11. Streamlines and nonconforming spectral element mesh for the Kovasznay flow problem.

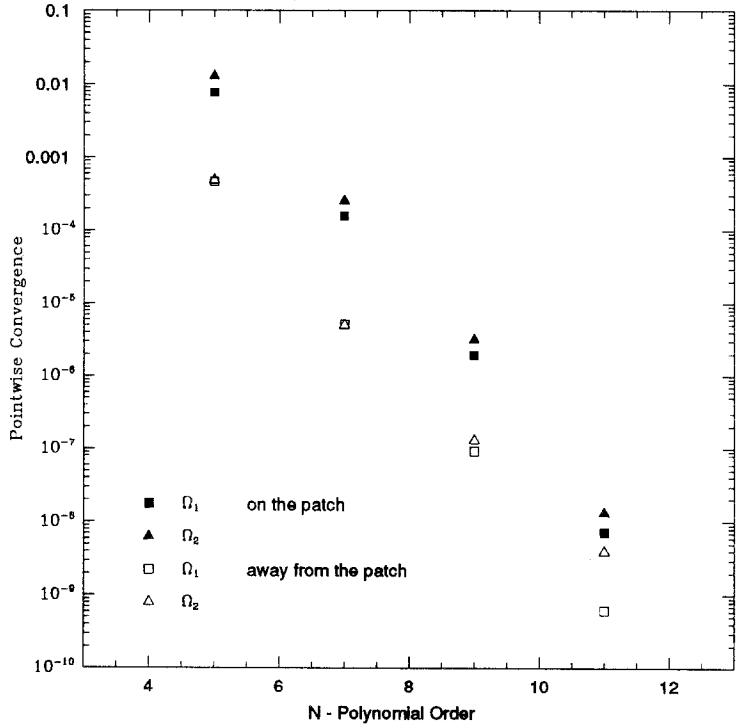

Fig. 12. Convergence for the Kovasznay flow problem. As the polynomial order of the expansion within domains Ω_1 and Ω_2 is increased, we obtain exponential convergence both on and away from the patched interface.

nomial expansions in the regions Ω_1 and Ω_2. As a test case we will solve the flow problem proposed by Kovasznay (1948) where the solution is given by

$$u = 1 - e^{\lambda x} \cos 2\pi y, \qquad v = \frac{\lambda}{2\pi} e^{\lambda x} \sin 2\pi y, \qquad p = \frac{1}{2}(1 - e^{2\lambda x}) \quad (3.8)$$

where $\lambda = R/2 - (R^2/4 + 4\pi^2)^{1/2}$. This solution is shown in the form of streamlines in Fig. 11 and may represent steady, low-Reynolds-number flow in the wake of a row of cylinders. The solution domain $\Omega = [-1/2, 1] \times [-1/2, 3/2]$ is subdivided into $\Omega_1 = [-1/2, 1] \times [1/2, 3/2]$ and $\Omega_2 = [-1/2, 1] \times [-1/2, 1/2]$. Within these subdomains we further subdivide the solution space into a number of spectral elements; a typical mesh is shown in Fig. 11. In Fig. 12 we see that as the polynomial order N is increased simultaneously in Ω_1 and Ω_2 this nested decomposition results in *exponential* convergence both on and away from the patched interface. This convergence is critical in the case of high-order schemes as it justifies the additional work usually associated with them. Even in the case of dissimilar polynomial expansions in the two subdomains we still obtain excellent results, as shown in Fig. 13 for the particular case $N_1 = 9$ and $N_2 = 7$.

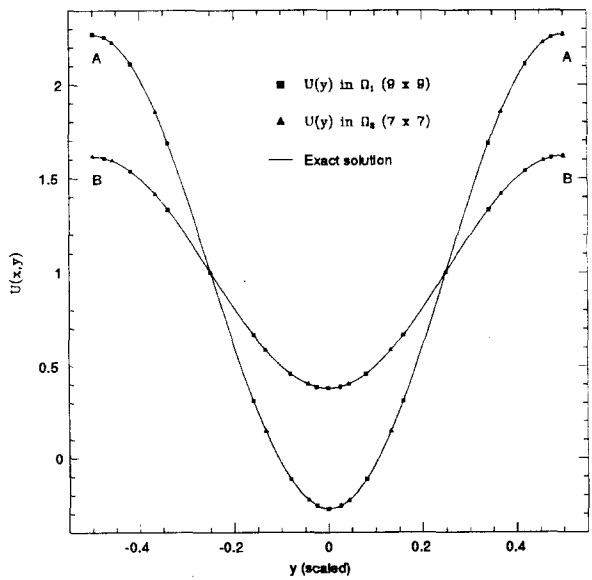

Fig. 13. Profiles of the streamwise velocity component at locations $x = -0.25$ (A) and $x = 0.50$ (B) for the Kovasznay problem.

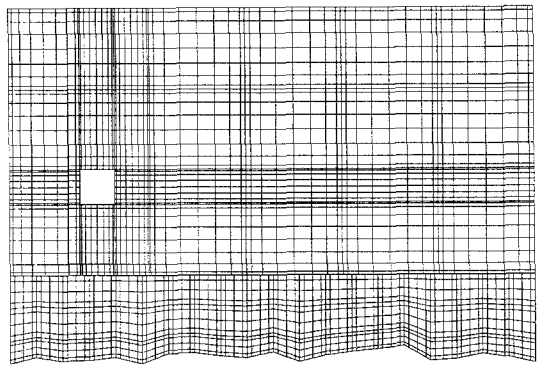

Fig. 14. Nonconforming spectral element mesh for flow past a square cylinder in a channel. High-order spectral elements are used in the upper domain, while in the lower domain finite elements provide a more efficient means of representing the irregular boundary.

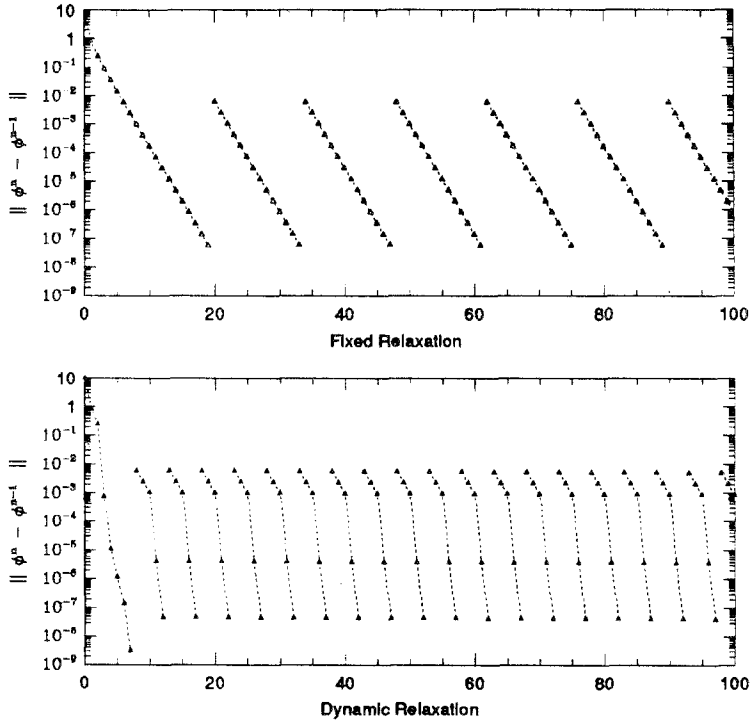

Fig. 15. Convergence rates for fixed and dynamic relaxation in the streamwise velocity calculations. Over the course of 100 "global" iterations, the procedure with fixed relaxation achieves 6 times steps while the dynamically relaxed procedure achieves 20.

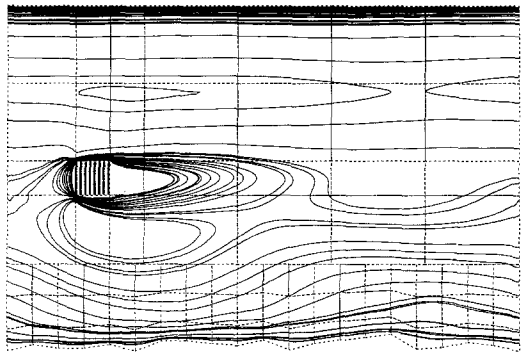

Fig. 16. Streamwise velocity contours for the unsteady flow past a square cylinder at a Reynolds number of approximately 110 based on the cylinder diameter. The slightly irregular lower wall results in a flow structure much more complicated than that seen in the upper half of the channel.

To address the issue of computational efficiency for time-dependent calculations, we simulate unsteady flow past a square cylinder in a channel with a rough wall. The computational mesh is shown in Fig. 14; the boundary conditions are periodicity at inflow and outflow and no slip along the upper and lower walls. A relatively small number of high-order spectral elements have been used to decompose the regular domain Ω_1 while a larger number of low-order finite elements have been used to represent the irregular domain Ω_2. Again, this emphasizes the generality of the Zanolli patching procedure in resolving the two solutions. In Fig. 15 we compare fixed versus dynamic relaxation for the streamwise velocity calculations. Clearly, the fixed relaxation is inappropriate in this case but dynamic relaxation yields convergence to the tolerance of 10^{-8} in only 5 iterations per time step. A plot of streamwise velocity contours at a simulation time of $t = 153.2$ is shown in Fig. 16.

4. DISCUSSION

We have presented a general procedure for solving elliptic problems in irregular domains using an iterative patching algorithm which allows for the coupling of fundamentally different discretization techniques. In particular, hybrid schemes which combine high-order spectral elements, finite elements, and finite difference discretizations have been formulated. Results for the Helmholtz equation were presented and performance of both serial and parallel implementations were discussed. Extension to the Navier–Stokes equations showed these results to carry through to complex numerical simulations including unsteady fluid flows. From these results it

is clear that iterative patching methods offer an efficient way to couple dissimilar discretizations in subdomains. For both elliptic problems and Navier–Stokes calculations presented here we have shown that in hybrid discretizations coupled through the Zanolli procedure high accuracy and spectrallike convergence is achieved in the spectral domain and depends only weakly on the low-order discretized subdomain. In addition, good performance has been achieved on both serial and parallel processors.

Currently, we are working on fully parallel implementations of these algorithms as well as extending these hybrid schemes to address the issues of local refinement and composite grids, adaptive gridding, time-dependent domains, and three-dimensional flow over surfaces of arbitrary (random) roughness.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under grants Nos. CTS-8906432, CTS-8906911, and CTS-8914422, and by ONR under grant No. N00014-82-C-0451. Computations were performed on the facilities of NAS at NASA Ames and on the Intel iPSC/860-32 Hypercube at Princeton University.

REFERENCES

Patera, A. T. (1984). A spectral element method for fluid dynamics: Laminar flow in a channel expansion. *J. Comput. Phys.* **54**, 468.

Karniadakis, G. E., Bullister, E. T., and Patera, A. T. (1985). A spectral element method for solution of two- and three-dimensional time dependent Navier–Stokes equations. In *Finite Element Methods for Nonlinear Problems*, Springer-Verlag, New York, p. 803.

Karniadakis, G. E. (1989). Spectral element simulations of laminar and turbulent flows in complex geometries, *Applied Numer. Math.* **6**, 85.

Clouqueur, A., and d'Humières, D. (1987). RAP1, a cellular automaton machine for fluid dynamics, *Complex Syst.* **1**, 585–597.

Nosenchuck, D. M., and Littman, M. G. (1986). The coming age of the parallel processor, In Proc. 23rd Annual Space Conference, NASA.

Karniadakis, G. E., Israeli, M., and Orszag, S. A. (1991). High-order splitting methods for the incompressible Navier–Stokes equations, *J. Comput. Phys.* 1991, to appear.

Funaro, D., Quarteroni, A., and Zanolli, P. (1985). An interative procedure with interface relaxation for domain decomposition methods. Technical Report 530, Instituto di Analisi Numerica del Consiglio Nazionale delle Ricerche, Pavia.

Maday, Y., and Patera, A. T. (1989). Spectral element methods for the Navier–Stokes equations, *State of the Art Surveys in Computational Mechanics*, ASME.

Gottlieb, D., and Orszag, S. A. (1977). *Numerical Analysis of Spectral Methods: Theory and Applications*, SIAM, Philadelphia.

Roache, P. J. (1982). *Computational Fluid Dynamics*, Hermosa Publishers.

van der Wijngaart, R. G. (1990). Composite-Grid Techniques and Adaptive Mesh Refinement in Comptational Fluid Dynamics, report No. CLaSSiC-90-07, Stanford University, January.

Canuto, C., and Funaro, D. (1987). The Schwarz algorithm for spectral methods, *SIAM J. Numer. Anal.*

Lions, P. L. (1989). On the Schwarz alternating method III: A variant for non-overlapping subdomains, in Chan *et al.* (1989), 202.

Douglas, C. C. (1989). A variation of the Schwarz alternating method: The domain decomposition reduction method, in Chan *et al.* (1989), p. 191.

Kovasznay, L. I. G. (1948). Laminar flow behind a two-dimensional grid, in *Proc. Cambridge Phil. Society* **1948**, 44.

Chan, T. F., Glowinski, R., Periaux, J., and Widlund, O. B. (eds.) (1989). *Third Int. Symp. on Domain Decomposition Methods for Partial Differential Equations*, Houston, Texas, March 1989, SIAM, Philadelphia.