JOURNAL OF COMPUTATIONAL PHYSICS 37, 70–92 (1980)

Spectral Methods for Problems in Complex Geometries

STEVEN A. ORSZAG

Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Received April 3, 1979; revised August 22, 1979

The properties of spectral methods are surveyed and their extension to solve problems in complex geometries is developed. A new iteration procedure is introduced to solve efficiently the full matrix equations resulting from spectral approximations to nonconstant coefficient boundary-value problems in complex geometries. It is shown that the work required to solve these spectral equations exceeds that of solving the lowest-order finite-difference approximation to the same problem by only $O(N \log N)$.

1. INTRODUCTION

In this paper, we outline some new techniques that permit the efficient application of spectral methods for solving problems in (nearly) arbitrary geometries. The resulting methods are a viable alternative to finite-difference and finite-element methods for these problems. Spectral methods should be particularly attractive for problems in several space dimensions in which high accuracy is required.

Spectral methods are based on representing the solution to a problem as a truncated series of smooth functions of the independent variables. Whereas finite-element methods are based on expansions in local basis functions, spectral methods are based on expansions in global functions. Spectral methods are the extension of the standard technique of separation of variables to the solution of arbitrarily complicated problems.

Let us begin by illustrating spectral methods for the simple one-dimensional heat equation. Consider the mixed initial-boundary value problem

$$\frac{\partial u(x,t)}{\partial t} = K \frac{\partial^2 u(x,t)}{\partial x^2} \qquad (0 < x < \pi,\ t > 0), \tag{1.1}$$

$$u(0,t) = u(\pi,t) = 0 \qquad (t \geqslant 0), \tag{1.2}$$

$$u(x,0) = f(x) \qquad (0 \leqslant x \leqslant \pi). \tag{1.3}$$

The solution to this problem is

$$u(x,t) = \sum_{n=1}^{\infty} a_n(t) \sin nx, \tag{1.4}$$

$$a_n(t) = f_n e^{-Kn^2 t}, \tag{1.5}$$

0021-9991/80/100070–23$02.00/0

where

$$f_n = \frac{2}{\pi} \int_0^\pi f(x) \sin nx \, dx \tag{1.6}$$

are the coefficients of the Fourier sine series expansion of $f(x)$.

A spectral approximation to (1.1)–(1.3) is obtained by simply truncating (1.4) to

$$u_N(x, t) = \sum_{n=1}^{N} a_n(t) \sin nx \tag{1.7}$$

and replacing (1.5) by the evolution equation

$$\frac{da_n}{dt} = -Kn^2 a_n \qquad (n = 1,..., N) \tag{1.8}$$

with the initial conditions $a_n(0) = f_n$ $(n = 1,..., N)$.

The spectral approximation (1.7)–(1.8) to (1.1)–(1.3) is an exceedingly good approximation for any time t greater than zero as $N \to \infty$. In fact, the error $u(x, t) - u_N(x, t)$ satisfies

$$u(x, t) - u_N(x, t) = \sum_{n=N+1}^{\infty} f_n e^{-Kn^2 t} \sin nx = O(e^{-KN^2 t}) \quad (N \to \infty) \tag{1.9}$$

for any $t > 0$. In contrast to (1.9), finite-difference approximations to the heat equation using N grid points in x lead to errors that decay only algebraically with N as $N \to \infty$. Furthermore, this spectral method for the solution of the heat equation is efficiently implementable by the fast Fourier transform (FFT) in $O(N \log N)$ operations.

There are several significant difficulties in extending the simple spectral method employed for (1.1)–(1.3) to more general problems. These difficulties and their solutions will be discussed in the following sections. Some further details are given in the author's monograph [1]. In Section 2, we discuss the difficulty caused by nontrivial boundary conditions. In Section 3, we discuss the difficulty of treating nonlinear and nonconstant coefficient terms. Then, in Section 4, we summarize the properties of spectral methods for problems in simple geometries. In Section 5, we explain how spectral methods can be extended to problems in complicated geometries. In Section 6, a new technique for the efficient solution of spectral equations that arise in complicated geometries is given. Some representative test problems are discussed in Section 7. Then, in Section 8, we summarize our results and provide a glimpse of some other new developments in spectral methods that should find wide application.

2. Boundary Conditions

The Fourier series (1.4) converges rapidly if $u(x, t)$ is infinitely differentiable *and* $u(x, t)$ satisfies the boundary conditions

$$\frac{\partial^{2n} u(x, t)}{\partial x^{2n}} = 0 \qquad (x = 0, \pi) \tag{2.1}$$

for all nonnegative integers n. Under these conditions, the error after N terms

$$\varepsilon_N(x, t) = u(x, t) - \sum_{n=1}^{N} a_n(t) \sin nx$$

goes to xero uniformly in x faster than any power of $1/N$ as $N \to \infty$. On the other hand, if $u(x, t)$ is not infinitely differentiable *or* if any of the conditions (2.1) is violated, then $\varepsilon_N(x, t) = O(1/N^p)$ as $N \to \infty$ for some finite p. For example,

$$1 = \sum_{n=0}^{\infty} (-1)^n \frac{\sin(2n+1)x}{2n+1} \qquad (0 < x < \pi), \tag{2.2}$$

but the error incurred by truncating after N terms is of order $1/N$ for any fixed x, $0 < x < \pi$. Furthermore, the convergence of (2.2) is not uniform in x; (2.2) exhibits Gibb's phenomenon, namely,

$$\varepsilon_N(\xi/N) = 0(1) \qquad (N \to \infty, \xi \text{ fixed}).$$

For any fixed N, there are points x at which the error after N terms of (2.2) is not small. The poor convergence of (2.2) is due to the violation of (2.1) for $n = 0$.

More generally, most eigenfunction expansions of a function $f(x)$ converge faster than algebraically (i.e., the error incurred by truncating after N terms goes to zero faster than any finite power of $1/N$ as $N \to \infty$) only if $f(x)$ is infinitely differentiable *and* $f(x)$ satisfies an infinite number of special boundary conditions. For example, the Fourier–Bessel expansion

$$f(x) = \sum_{n=0}^{\infty} a_n J_0(\lambda_n x) \qquad (0 \leqslant x \leqslant 1),$$

where λ_n is the nth smallest root of $J_0(\lambda) = 0$, converges faster than algebraically only if f is infinitely differentiable and

$$\left[\frac{1}{x}\frac{d}{dx} x \frac{d}{dx}\right]^k f(x) = 0 \qquad \text{at} \quad x = 1 \tag{2.3}$$

for $k = 0, 1, 2, \ldots$.

When a spectral expansion converges only algebraically fast, spectral methods based on these eigenfunction expansions cannot offer significant advantages over

more conventional (finite-difference, finite-element) methods. Eigenfunction expansions of this kind should not normally be used *unless* the boundary conditions of the problem imply all the extra boundary constraints like (2.1) or (2.3). For example, if periodic boundary conditions are compatible with the differential equation to be solved, complex Fourier series are suitable to develop efficient spectral approximations.

In the development of spectral methods for general problems, it is important that the rate of convergence of the eigenfunction expansion being used not depend on special properties of the eigenfunctions, like boundary conditions, but rather depend only on the smoothness of the function being expanded. Of course, if the solution to the problem being solved is not smooth, one should not expect errors that decrease faster than algebraically with $1/N$ when global eigenfunction expansions are used. Faster than algebraic rates of convergence may be achieved for these problems by either patching the solution at discontinuities (see Section 5) or pre- and postprocessing of the solution (see [2]).

There is an easy way to ensure that the rate of convergence of a spectral expansion of a function $f(x)$ depends only on the smoothness of $f(x)$, not its boundary properties. The idea is to expand in terms of suitable classes of orthogonal polynomials, including Chebyshev and Legendre polynomials for all those problems in which constraints like (2.1) and (2.3) are unrealistic. These polynomial expansions avoid all difficulties associated with the Gibbs phenomenon provided the solution $f(x)$ is smooth.

From the mathematical point of view, the classical orthogonal polynomials are eigenfunctions of singular Sturm–Liouville problems. It is not hard to show (see [1] for the details) that expansions using eigenfunctions of such singular Sturm–Liouville problems converge at a rate that depends only on the smoothness of $f(x)$, in contrast to eigenfunction expansions based on nonsingular Sturm–Liouville problems that lead to additional boundary constraints like (2.1) on $f(x)$.

These results for othogonal polynomial expansions are easily demonstrated in the case of Chebyshev polynomial expansions. The nth-degree Chebyshev polynomial $T_n(x)$ is defined by

$$T_n(\cos\theta) = \cos n\theta. \tag{2.4}$$

Therefore, if

$$f(x) = \sum_{n=0}^{\infty} a_n T_n(x) \tag{2.5}$$

then

$$g(\theta) = f(\cos\theta) = \sum_{n=0}^{\infty} a_n \cos n\theta \tag{2.6}$$

Thus, the Chebyshev polynomial expansion coefficients a_n of $f(x)$ are just the Fourier

cosine expansion coefficients of the even, periodic function $g(\theta)$. A simple integration by parts argument then shows that

$$n^p a_n \to 0 \qquad (n \to \infty)$$

provided $g(\theta)$ (or, equivalently, $f(x)$) has p continuous derivatives. Since

$$\left| f(x) - \sum_{n=0}^{N} a_n T_n(x) \right| \leqslant \sum_{n=N+1}^{\infty} |a_n| \qquad (|x| \leqslant 1),$$

it follows that the rate of convergence of (2.5) is faster than algebraic if f is smooth.

In summary, spectral expansions should be made using series of orthogonal polynomials unless the boundary conditions of the problem are fully compatible with some other class of eigenfunctions. In practice, Chebyshev and Legendre polynomial expansions are recommended for most applications, supplemented by Fourier series and surface harmonic series when boundary conditions permit.

3. Nonlinear and Nonconstant Coefficient Problems

Another difficulty with general kinds of spectral methods is their application to problems with nonlinear and nonconstant coefficient terms. Before explaining the solution to this problem, let us illustrate the difficulty.

Suppose we wish to solve the partial differential equation

$$\frac{\partial u}{\partial t} = \mathcal{N}(u, u) + \mathcal{L}u, \tag{3.1}$$

where $u = u(\mathbf{x}, t)$ and $\mathcal{N}$ is a bilinear (nonlinear) operator that involves only spatial derivatives and $\mathcal{L}$ is a linear operator that involves only spatial derivatives. The operators $\mathcal{N}$ and $\mathcal{L}$ may depend on both $\mathbf{x}$ and t. A spectral method for the solution of (3.1) is obtained by seeking the solution as a finite spectral expansion:

$$u(\mathbf{x}, t) = \sum_{n=1}^{N} a_n(t)\, \psi_n(\mathbf{x}), \tag{3.2}$$

where we assume for now that $\psi_n(\mathbf{x})$ $(1 \leqslant n < \infty)$ are a complete set of orthogonal functions. If we introduce the reexpansion coefficients c_{nmp} and d_{nm} so that

$$\mathcal{N}(\psi_m, \psi_p) = \sum_{n=1}^{\infty} c_{nmp}(t)\psi_n, \tag{3.3}$$

$$\mathcal{L}(\psi_m) = \sum_{n=1}^{\infty} d_{nm}(t)\psi_n \tag{3.4}$$

and equate coefficients of $\psi_n(\mathbf{x})$ $(n = 1,..., N)$ in (3.1), we obtain

$$\frac{da_n}{dt} = \sum_{m=1}^{N} \sum_{p=1}^{N} c_{nmp}(t) a_m(t) a_p(t) + \sum_{m=1}^{N} d_{nm}(t) a_m(t) \quad (n = 1,..., N). \tag{3.5}$$

Equations (3.5) are the spectral evolution equations for the solution of (3.1). They have one very serious drawback. In general c_{nmp} and d_{nm} are nonzero for typical n, m, p so that evaluation of da_n/dt from (3.5) for all $n = 1,..., N$ requires $O(N^3)$ arithmetic operations for the bilinear term and $O(N^2)$ operations for the linear term. Thus, solution of (3.5) requires order N^3 operations per time step. Since operational spectral calculations now involve $N \gtrsim 10^6$, the computational cost of the direct solution of (3.5) is prohibitive (even if only linear terms are present).

The problem here is one of computational complexity. Finite-difference methods for the solution of (3.1) on N grid points may require only order N operations per time step. If the spectral method really requires order N^3 operations per time step it cannot compete when N is large.

Another example illustrating the computational complexity of spectral methods is given by the nonlinear diffusion equation

$$\frac{\partial u(x, t)}{\partial t} = e^u \frac{\partial^2 u}{\partial x^2}(x, t). \tag{3.6}$$

If we seek the solution as

$$u(x, t) = \sum_{n=1}^{\infty} a_n(t)\, \psi_n(x) \tag{3.7}$$

in terms of the orthonormal functions $\psi_n(x)$, then

$$\frac{da_n}{dt} = \int \psi_n(x) \exp\left[\sum_{m=1}^{N} a_m(t)\, \psi_m(x)\right] \sum_{p=1}^{N} a_p \psi_p''(x)\, dx \tag{3.8}$$

for $n = 1,..., N$. These evolution equations for $\{a_n(t)\}$ have an exponential degree of computational complexity as they are expressed as an integral functional of $\{a_n(t)\}$.

The solution to the problem of computational complexity is to use the author's transform methods. Let us illustrate the technique for a pseudospectral (or collocation) approximation to (3.6) [3]. First, we introduce N suitable collocation points $x_1, x_2,..., x_N$ lying within the computational domain. Then, the approximate solution (3.7) is forced to satisfy the partial differential equation (3.6) (or its boundary conditions) exactly at these discrete points at every time t. More specifically, the following three steps are done at each time step t:

(i) Determine N coefficients $a_n(t)$ $(n = 1,..., N)$ so that

$$u(x_j, t) = \sum_{n=1}^{N} a_n(t)\, \psi_n(x_j) \qquad (j = 1,..., N). \tag{3.9}$$

(ii) Evaluate $u_{xx}(x_j, t)$ by

$$u_{xx}(x_j, t) = \sum_{n=1}^{N} a_n(t)\, \psi_n''(x_j) \qquad (j = 1,..., N). \tag{3.10}$$

(iii) Finally, evaluate $\partial u(x_j, t)/\partial t$ by

$$\frac{\partial u(x_j, t)}{\partial t} = e^{u(x_j t)} u_{xx}(x_j, t) \qquad (j = 1,..., N) \tag{3.11}$$

and march forward to the next time step.

The idea of the pseudospectral transform method can be restated as follows: Transform freely between physical (x_j) and spectral (a_n) representations, evaluating each term in whatever representation that term is most accurately, and simply, evaluated. Thus, in (3.11), we evaluate e^u in the physical representation while we compute u_{xx} in the spectral representation by (3.10) because it is most accurately done there.

It should be apparent to the reader that pseudospectral transform methods can be applied to any problem that can be treated by finite-difference methods regardless of the technical complexity of nonlinear and nonconstant coefficient terms.

Let us now examine the computational complexity of pseudospectral transform methods. There are at least three aspects to this question: (i) the computational complexity of differentiation, integration, etc., of spectral series; (ii) the computational complexity of transforming back and forth between physical and spectral representations; and (iii) the computational complexity of solving the resulting equations for the spectral coefficients.

For the expressions of interest, computation of derivatives of an N term spectral expansion requires order N arithmetical operations. For the Fourier series (1.7), this fact is obvious:

$$\frac{d}{dx} \sum_{n=1}^{N} a_n \sin nx = \sum_{n=1}^{N} n a_n \cos nx,$$

$$\frac{d^2}{dx^2} \sum_{n=1}^{N} a_n \sin nx = -\sum_{n=1}^{N} n^2 a_n \sin nx.$$

For the Chebyshev polynomial expansion (2.5), the computational complexity of differentiation is a little less apparent. Since $T_n(\cos\theta) = \cos n\theta$,

$$\frac{T_{n+1}'(x)}{n+1} - \frac{T_{n-1}'(x)}{n-1} = \frac{2}{c_n} T_n(x) \qquad (n \geqslant 0),$$

where $c_0 = 2$, $c_n = 1$ $(n \geqslant 1)$, and $T_0' = T_{-1}' = 0$. Therefore, if

$$\frac{d}{dx} \sum_{n=0}^{N} a_n T_n(x) = \sum_{n=0}^{N} b_n T_n(x),$$

then

$$2\sum_{n=1}^{N} a_n T_n(x) = \sum_{n=0}^{N} c_n b_n \left[\frac{T'_{n+1}}{n+1} - \frac{T'_{n-1}}{n-1}\right]$$
$$= \sum_{n=1}^{N+1} \left[c_{n-1} b_{n-1} - b_{n+1}\right] T'_n(x)/n.$$

Equating coefficients of $T'_n(x)$ for $n = 1,..., N+1$ gives the recurrence relation

$$\begin{aligned} c_{n-1} b_{n-1} - b_{n+1} &= 2na_n \qquad (1 \leqslant n \leqslant N), \\ b_n &= 0 \qquad (n \geqslant N). \end{aligned} \tag{3.12}$$

The solution of (3.12) for b_n given a_n requires only order N arithmetic operations. Similar recurrence relations can be obtained for differentiation of spectral series based on other sets of orthogonal polynomials and functions.

The computational complexity of transforming between spectral and physical space has several interesting aspects. The problem is: How much computational work is necessary to evaluate

$$u_j = \sum_{n=1}^{N} a_n \psi_n(x_j) \qquad (j = 1,..., N) \tag{3.13}$$

given $\{a_n\}$ and, inversely, how much work is necessary to compute the expansion coefficients $\{a_n\}$ given $\{u_j\}$? It is obvious that (3.13) can be evaluated for $\{u_j\}$ in $O(N^2)$ operations while it can be solved for a_n in $O(N^3)$ operations. However, these estimates are much too pessimistic; for many important expansions the operation count to perform the transform (3.13) and its inverse is no larger than $O(N(\log N)^p)$ with $p \leqslant 2$.

In the case of Fourier series, the transform (3.13) and its inverse can be computed in $O(N \log_2 N)$ operations if $N = 2^p$ using the fast Fourier transform. However, most of the computational efficiency of transform methods comes not from the FFT but from the separability of multidimensional transforms. Thus, a three-dimensional discrete Fourier transform can be expressed as three one-dimensional Fourier transforms,

$$\sum_{j=0}^{J-1} \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} a(j,k,l) \exp\left[2\pi i \left(\frac{jm}{J} + \frac{kn}{K} + \frac{lp}{P}\right)\right]$$
$$= \sum_{j=0}^{J-1} e^{2\pi ijm/J} \sum_{k=0}^{K-1} e^{2\pi ikn/K} \sum_{l=0}^{L-1} a(j,k,l) e^{2\pi ilp/L}. \tag{3.14}$$

The left side of (3.14) requires roughly $(JKL)^2$ operations to evaluate at all the points $0 \leqslant m < J$, $0 \leqslant n < K$, $0 \leqslant p < L$. On the other hand, even without the FFT the right side of (3.14) requires only about $(JKL)(J + K + L)$ operations to evaluate at all the points. When the FFT is applied to the one-dimensional transforms on the

right side of (3.14), the number of operations necessary to evaluate (3.14) is reduced further to (JKL) $(\log_2 J + \log_2 K + \log_2 L)$ if J, K, L are powers of 2.

For example, the spectral code CENTICUBE solves the Navier–Stokes equations for three-dimensional incompressible flow with periodic boundary conditions; three-dimensional Fourier series with resolution up to $128 \times 128 \times 128$ are used to represent the flow. The equations for the Fourier components $\mathbf{u}(\mathbf{k}, t)$ of the velocity field involve several convolution sums of the form

$$\sum_{|\mathbf{p}|<64} \mathbf{u}(\boldsymbol{p}, t)\, \mathbf{u}(\mathbf{k} - \mathbf{p}, t).$$

It is not hard to show that direct evaluation of all the required convolution sums on the CRAY-1 computer would require (using optimized code) about 5×10^5 sec of computer time at each simulation time step. On the other hand, the CENTICUBE code executes on the CRAY-1 in less than 20 sec per time step! This speedup by a factor 2.5×10^4 is broken down roughly as follows: a factor 2 for using the FFT instead of an optimized matrix multiply to perform a one-dimensional transform and a factor 10^4 for performing the transforms as a sequence of one-dimensional transforms as in (3.14).

More general transforms can also be implemented efficiently. The author has recently shown [4] that "fast" transforms of N-term series of Legendre polynomials, surface harmonics, Bessel functions, Jacobi polynomials, Gegenbauer polynomials, etc., can be accomplished in $O(N(\log N)^2/\log\log N)$ arithmetic operations while transforms of series of Hermite and Laguerre polynomials can be accomplished in $O(N(\log N)^2)$ operations. At the present time, these transforms are of mostly academic interest—as discussed above the speed of transform methods for problems of realistic size is attributable to the formulation of the problem in terms of separable transforms, not the existence of a fast transform. Nevertheless, speedups of factors 2 and more are significant, so most of our current work with spectral methods uses transforms based on the FFT.

The third question regarding computational complexity of spectral transform methods concerns the complexity of solving the equations for spectral coefficients. In the case of initial-value problems solved by explicit time-stepping methods, the answer is provided by the transform methods discussed above: $O(N(\log N)^p)$ operations are required per time step. The answer to the question is more complicated for the solution of boundary-value problems or implicit time-stepping methods for initial-value problems.

Spectral approximations to general boundary-value problems lead to full $N \times N$ matrix equations for the N expansion coefficients a_n. It would seem that solution of these equations requires $O(N^3)$ arithmetic operations while storage of the matrix requires $O(N^2)$ memory locations. Since typical problems now involve $N \sim 10^6$, the direct solution (or even the direct formulation) of such problems is clearly unworkable now. The solution to this problem is given in Section 6. The solution to the spectral equations requires essentially only $O(N(\log N)^p)$ $(p \leqslant 2)$ operations and only

$O(N)$ memory locations. Solution of spectral equations, even though they lead to formally full matrix problems, can be accomplished with little more work and memory than required to solve the simplest finite-difference equations!

4. Time Differencing and Boundary Layers

Spectral methods based on the classical orthogonal polynomials have another feature that is very attractive for some kinds of problems but leads to difficulties with others. This feature is very high resolution near the boundaries. For example, the collocation points for Chebyshev polynomial pseudospectral methods for problems on $-1 \leqslant x \leqslant 1$ are usually chosen to be $x_j = \cos \pi j/N$ ($j = 0, 1, \ldots, N$). The collocation points x_1 and x_{N-1} are within about $\pi^2/2N^2$ of the boundary points x_0 and x_N, respectively, so that the boundary resolution is $\Delta x = O(1/N^2)$. This leads to extremely good resolution properties of spectral methods for boundary-layer problems (see [1, 5]). While resolution of a problem with a boundary layer of thickness $\varepsilon \ll 1$ would require $O(1/\varepsilon)$ uniformly spaced grid points, it requires only $O(1/\varepsilon^{1/2})$ terms in the Chebyshev spectral series. [Nonuniform grids should, of course, be used in many of these problems. They can also be implemented efficiently in spectral methods using coordinate transformations.]

The high boundary resolution of spectral methods is not directly useful when problems without boundary layers are to be solved. For example, consider the one-dimensional wave equation

$$u_t + u_x = 0 \qquad (-1 \leqslant x \leqslant 1,\ t > 0), \tag{4.1}$$

$$u(-1, t) = f(t) \qquad (t > 0), \tag{4.2}$$

$$u(x, 0) = g(x) \qquad (-1 \leqslant x \leqslant 1). \tag{4.3}$$

Using a Chebyshev polynomial spectral representation

$$u(x, t) = \sum_{n=0}^{N} a_n(t)\, T_n(x), \tag{4.4}$$

where $T_n(x) = \cos(n \cos^{-1} x)$, the spectral-tau equations for the solution of (4.1) are [1]:

$$\frac{da_n}{dt} = -\frac{2}{c_n} \sum_{\substack{p=0 \\ p+n \text{ odd}}}^{N} p a_p \qquad (0 \leqslant n \leqslant N-1), \tag{4.5}$$

$$\sum_{n=0}^{N} (-1)^n a_n(t) = f(t), \tag{4.6}$$

where $c_0 = 2$, $c_n = 1$ $(n \geqslant 1)$. The numerical solution of (4.5)–(4.6) in time can be done using any absolutely stable time integration method for ordinary differential equations, such as one of the Adams methods [6]. For an explicit Adams method (Adams–Bashforth method), absolute stability in the solution of (4.5)–(4.6) requires that

$$\Delta t = O(1/N^2). \tag{4.7}$$

This result may be verified from (4.5)–(4.6) using Gerschgorin's theorem on the distribution of eigenvalues or, more physically, from the classical explicit stability condition $\Delta t = O(\Delta x)$ with $\Delta x = O(1/N^2)$. Specifically, the first-order Adams–Bashforth method (Euler's method) is stable for the solution of (4.5)–(4.6) provided that

$$\Delta t < 8/N^2. \tag{4.8}$$

The time step restrictions (4.7)–(4.8) are qualitatively more severe than those encountered by standard finite-difference methods for (4.1)–(4.3). The solution to (4.1)–(4.3) does not exhibit boundary layer structure (unless $g(x)$ or $f(t)$ has non-uniform variation) so a uniform grid may be employed giving the stability criterion $\Delta t = O(1/N)$ using N grid points. The high boundary resolution of the spectral scheme that leads to the more stringent requirements (4.7) or (4.8) may seem wasted in this problem. In fact, this high boundary resolution is not completely useless; while high-order accurate stable finite-difference schemes for solution of (4.1)–(4.3) on a uniform grid are complicated and require a number of spurious numerical boundary conditions (see, for example, [7]), the infinite-order accurate spectral scheme (4.5)–(4.6) is quite straightforward and requires no spurious boundary conditions to be applied. However, it is also nice to know that the stiffness of the spectral equations can be easily circumvented and time step restrictions like those of finite-difference schemes can be easily obtained.

At the present time, there are three alternative ways to avoid the severe time step restrictions of spectral methods in problems that do not exhibit strong boundary-layer structure. First, a semi-implicit scheme [1] may be employed to relax the time step restrictions (4.7)–(4.8) to $\Delta t = O(1/N)$ as for finite-difference schemes. The idea here is to treat implicitly just those parts of the problem, to wit, the boundary regions, that lead to the stiffness of the spectral equations.

Second, it is possible to formulate *explicit* unconditionally stable time differencing schemes for spectral methods [8]. Domain of dependence arguments may be used to demonstrate the existence of conditional stability bounds for explicit finite-difference methods. These bounds can be avoided in explicit spectral methods because global data is involved at each time step to march the solution forward in time. A simple example of an unconditionally stable explicit spectral method may be given for

$$\frac{\partial u(k, t)}{\partial t} = -iku(k, t), \tag{4.9}$$

which is the equation for a Fourier mode of the solution to $\partial u/\partial t + \partial u/\partial x = 0$ with periodic boundary conditions. Leapfrog time differencing of (4.9) gives

$$\frac{u(k, t+\Delta t) - u(k, t-\Delta t)}{2\Delta t} = -iku(k, t), \tag{4.10}$$

which is second-order accurate in Δt and has the explicit stability bound $|k\Delta t| \leqslant 1$. However, the modified scheme

$$\frac{u(k, t+\Delta t) - u(k, t-\Delta t)}{2\Delta t} = -i\frac{\sin k\Delta t}{\Delta t}u(k, t) \tag{4.11}$$

is still second-order accurate but it is unconditionally stable since $|\sin k\Delta t| \leqslant 1$ for all $k\Delta t$. [By an accidental cancellation (4.11) happens to give the exact solution $e^{-ik\Delta t}$ of (4.9) for all $k\Delta t$.]

The third method for stabilizing time integrations of spectral methods is to use a full implicit time integration method. With N degrees of freedom used to resolve the spatial dimensions, full implicit schemes give full $N \times N$ sets of linear equations to solve at each time step. An efficient method to solve these equations is presented in Section 6.

5. Formulation of Spectral Methods in Complex Geometries

For most problems in complex geometries, it is both inefficient and disadvantageous to seek special sets of spectral expansion functions tuned to the details of the geometry. First, determination of such a set of special functions is itself a computationally difficult problem in the complex geometry that must be repeated for every new geometry. Second, unless very special care is taken, the resulting spectral series will not be guaranteed of fast convergence properties for the problem of interest. Third, much computer storage will be required to store the required values of the expansion functions themselves and fast, separable, transforms between physical space and transform space will not normally be available.

There are two very general and powerful methods for the formulation of spectral methods in complex geometries that appear to preserve all the nice properties of spectral methods in simpler geometries, namely, mapping and patching.

The idea of mapping is to transform the complex geometry into a simpler one by a smooth transformation. For example, the annular region

$$\begin{aligned} f_1(\theta) \leqslant r \leqslant f_2(\theta), \\ 0 \leqslant \theta < 2\pi, \end{aligned} \tag{5.1}$$

where (r, θ) are polar coordinates, is transformed into the rectangle

$$\begin{aligned} -1 \leqslant z \leqslant 1, \\ 0 \leqslant \theta < 2\pi, \end{aligned} \tag{5.2}$$

by the simple stretching transformation

$$z = 2\frac{r - f_1(\theta)}{f_2(\theta) - f_1(\theta)} - 1. \tag{5.3}$$

In the mapped coordinates (5.2), a spectral expansion using Chebyshev polynomials in z and Fourier series in θ is appropriate (because the solution to the problem must be periodic in θ). The complication of using a coordinate transformation appears in the coefficients of the differential equation in the transformed domain. For example, derivatives are transformed according to

$$\left.\frac{\partial u}{\partial r}\right|_\theta = \frac{2}{f_2(\theta) - f_1(\theta)} \left.\frac{\partial u}{\partial z}\right|_\theta, \tag{5.4}$$

$$\left.\frac{\partial u}{\partial \theta}\right|_r = \left.\frac{\partial u}{\partial \theta}\right|_z - \frac{[z(f_2' - f_1') - (f_2' + f_1')}{f_2 - f_1} \left.\frac{\partial u}{\partial z}\right|_\theta. \tag{5.5}$$

The complication of the equation due to transformation causes no essential difficulty in the implementation of explicit time stepping schemes for initial-value problems because transform methods are still applicable. For boundary-value problems and full implicit treatment of initial-value problems, it is essential that the full matrix equations be solved by the fast iteration procedures introduced in the next section.

The simple stretching transformation (5.3) can only be applied if the boundaries $r = f_1$ and $r = f_2$ are single-valued functions of θ. More general boundaries require more sophisticated mappings. In two dimensions, conformal mapping is sometimes useful. Also, pseudo-Lagrangian marker particles may be used to define a coordinate frame. Many other sources of suitable transformations are possible (see, for example, [9]). In order to maintain spectral accuracy, the mappings should also be constructed using spectral methods. For example, we are currently studying spectral methods to find general curvilinear coordinate systems by techniques similar to those described in Ref. [9].

The idea of patching is to subdivide a complicated region into a number of simpler regions, make spectral expansions in each of the simpler regions, and then solve the problem in the complicated region by applying suitable continuity conditions across the artificial internal boundaries. For example, consider the Poisson equation

$$\nabla^2 u = f$$

in the L-shaped domain

$$\begin{aligned} 0 \leqslant x \leqslant 1, \quad & 0 \leqslant y \leqslant 1, \\ -1 \leqslant x \leqslant 0, \quad & 0 \leqslant y \leqslant 2. \end{aligned} \tag{5.6}$$

We subdivide the domain into three domains:

$$\text{I:} \qquad 0 \leqslant x \leqslant 1, \qquad 0 \leqslant y \leqslant 1; \tag{5.7a}$$

$$\text{II:} \qquad -1 \leqslant x \leqslant 0, \qquad 0 \leqslant y \leqslant 1; \tag{5.7b}$$

$$\text{III:} \qquad -1 \leqslant x \leqslant 0, \qquad 1 \leqslant y \leqslant 2. \tag{5.7c}$$

In each of these regions, we represent $u(x, y)$ as a double Chebyshev series:

$$u_{\mathrm{I}}(x, y) = \sum \sum a^{\mathrm{I}}_{mn} T_m(2x - 1)\, T_n(2y - 1), \tag{5.8a}$$

$$u_{\mathrm{II}}(x, y) = \sum \sum a^{\mathrm{II}}_{mn} T_m(2x + 1)\, T_n(2y - 1), \tag{5.8b}$$

$$u_{\mathrm{III}}(x, y) = \sum \sum a^{\mathrm{III}}_{mn} T_m(2x + 1)\, T_n(2y - 3). \tag{5.8c}$$

Across the internal artificial boundaries, continuity of u and the flux of u is imposed. For example, at the interface between regions I and II, we require that

$$u_{\mathrm{I}}(0, y) = u_{\mathrm{II}}(0, y) \qquad (0 \leqslant y \leqslant 1), \tag{5.9a}$$

$$\frac{\partial u_{\mathrm{I}}}{\partial x}(0, y) = \frac{\partial u_{\mathrm{II}}}{\partial x}(0, y) \qquad (0 \leqslant y \leqslant 1). \tag{5.9b}$$

The system of equations in regions I, II, III together with the continuity conditions of the form (5.9) gives a spectrally patched solution to the Poisson equation in the L-shaped domain (5.6).

The advantage of spectral methods over more conventional methods for patched problems is that the spectral schemes require only physical continuity conditions at the internal interfaces. On the other hand, finite-difference methods require spurious boundary conditions at interfaces whenever the order of the numerical scheme is higher than that of the differential equation to be solved.

6. Spectral Iteration Method

Consider the solution of a general linear differential equation $Lu = f$. (Extensions to nonlinear problems will be discussed later.) Let an N-term spectral approximation to this problem be given by

$$L_{sp} u_N = f_N, \tag{6.1}$$

where f_N is a suitable N-term approximation to f. As mentioned several times earlier the matrix representation of (6.1) is generally a full $N \times N$ matrix so that direct solu-

tion of (6.1) by Gauss elimination methods would require order N^2 storage (for the matrix representation of L_{sp}) and order N^3 arithmetic operations.

In this Section, we shall describe a method that permits solution of (6.1) using order N storage locations with the number of arithmetic operations of order the larger of $N \log N$ and the number of operations required to solve $Lu = f$ by a *first-order* finite-difference method. The important conclusion is that *spectral methods for general problems in general geometries can be implemented efficiently with operation costs and storage not much larger than those of the simplest finite-difference approximation to the problem with the same number of degrees of freedom.* Since spectral methods require many fewer degrees of freedom to achieve given accuracy (or, nearly equivalently, spectral methods achieve much higher accuracy for a given number of degrees of freedom) than required by finite-order difference approximations, important computational efficiencies result from the new method.

The idea of the iteration method is as follows: Suppose we are able to construct an approximation L_{ap} to the spectral operator L_{sp} that has the following properties:

(i) L_{ap} has a sparse matrix representation so that it can be represented using only $O(N)$ storage locations;

(ii) L_{ap} is efficiently invertible in the sense that the equation

$$L_{ap} u_N = f_N \tag{6.2}$$

is soluble as efficiently as a first-order finite-difference approximation to the problem;

$$0 < m \leqslant \| L_{ap}^{-1} L_{sp} \| \leqslant M < \infty \tag{6.3}$$

for suitable constants m, M as $N \to \infty$. Roughly speaking, (6.3) requires that the eigenvalues of $L_{ap}^{-1} L_{sp}$ be bounded from above and below as $N \to \infty$. Methods for constructing suitable operator approximations L_{ap} will be discussed below. Now we indicate how L_{ap} can be used to solve the spectral equation (6.1).

There are several iteration procedure using L_{ap} that converge efficiently to the solution of (6.1). We describe three of these procedures here:

Richardson (Jacobi) Iteration

Consider the iteration scheme

$$L_{ap} u_N^{(n+1)} = L_{ap} u_N^{(n)} - \alpha (L_{sp} u_N^{(n)} - f_N). \tag{6.4}$$

If

$$0 < \alpha < 2/M \tag{6.5}$$

then

$$u_N^{(n)} \to u_N \qquad (n \to \infty), \tag{6.6}$$

where u_N satisfies (6.1): $L_{sp}u_N = f_N$. The proof is elementary: If $\varepsilon^{(n)} = u_N^{(n)} - u_N$ is the error, then

$$\varepsilon^{(n+1)} = \varepsilon^{(n)} - \alpha L_{ap}^{-1} L_{sp} \varepsilon^{(n)}.$$

Therefore, noting (6.3),

$$\|\varepsilon^{(n+1)}\| \leqslant \max(|1 - \alpha m|, |1 - \alpha M|)\, \|\varepsilon^{(n)}\| \tag{6.7}$$

so $\|\varepsilon^{(n)}\| \to 0$ as $n \to \infty$ if (6.5) holds.

The optimal rate of convergence of Richardson's iteration is normally achieved by choosing α to minimize the factor $\max(|1 - \alpha m|, |1 - \alpha M|)$ appearing in (6.7). This gives

$$\alpha_{\text{opt}} = \frac{2}{M + m}. \tag{6.8}$$

so

$$\frac{\|\varepsilon^{(n+1)}\|}{\|\varepsilon^{(n)}\|} \leqslant \frac{M - m}{M + m}. \tag{6.9}$$

Since, as shown below, we can find L_{ap} such that $M \leqslant 2.5$ and $m \geqslant 1$ for nearly arbitrary spectral operators L_{sp}, it follows that Richardson's method decreases the norm of the error in the solution of (6.1) by at least a factor $(5/2 + 1)/(5/2 - 1) = 7/3$ at each iteration. (Here $\alpha_{\text{opt}} = 4/7$.) The accuracy of a typical initial approximation to u_N is improved by a factor 10^6 after about 16 iterations *independent of the resolution N.*

How much computational work is required per iteration? The right-hand side of (6.4) can be evaluated in $O(N \log N)$ operations by transform methods because $u_N^{(n)}$ is known. Also the solution of (6.4) for $u_N^{(n+1)}$ can be accomplished efficiently because of assumption (ii) above.

Chebyshev Iteration

The rate of convergence of the scheme (6.4) can be accelerated using Chebyshev acceleration [10]. The scheme is

$$L_{ap}u_N^{(n+1)} = L_{ap}[\omega_n u_N^{(n)} + (1 - \omega_n)u_N^{(n-1)}] - \alpha\omega_n(L_{sp}u_N^{(n)} - f_N), \tag{6.10}$$

where

$$\omega_n = \frac{2\beta T_n(\beta)}{T_{n+1}(\beta)} \tag{6.11}$$

and $\beta = \min(|1 - \alpha m|^{-1}, |1 - \alpha M|^{-1})$. Here $T_n(\beta)$ is the Chebyshev polynomial of degree n and m, M are given in (6.3). It is not hard to show that the error in the solu-

tion of (6.1) decreases by at least the factor $\beta + (\beta^2 - 1)^{1/2}$ after each iteration of (6.10).

If $M = 2.5$, $m = 1$, then choosing $\alpha = 4/7$ gives $\beta = 7/3$, so the error decreases by at least $(7 + 2(10)^{1/2})/3 \simeq 4.44$ at each iteration. Terefore the error in the solution of (6.1) is decreased by a factor 10^6 after about nine iterations of (6.10), nearly a factor two faster than the Richardson method (6.4). The penalty of using the Chebyshev acceleration method is that two levels of iterates, $u_N^{(n)}$ and $u_N^{(n-1)}$, must be stored.

Conjugate Gradient Iterations

For many problems, it is possible to accelerate the convergence of the iteration method for solving (6.1) still further by using the conjugate gradient method [11]. The applicability of this method to the solution of spectral equations will be studied in depth elsewhere. Here we report that for general Dirichlet problems for elliptic partial differential equations, the error is decreased by 10^4 after only three conjugate gradient iterations and by 10^7 after only seven iterations.

Let us now address the construction of a suitable approximate operator L_{ap}. The idea is akin to an idea of D'yakonov [12] but differs in one essential respect. D'yakonov considers the solution of the elliptic partial differential equation

$$Du = \nabla \cdot K(x, y) \nabla u = f \tag{6.12}$$

in some region R with, say, Dirichlet boundary conditions on δR. If

$$0 < K_{\min} \leqslant K(x, y) \leqslant K_{\max} < \infty, \tag{6.13}$$

then D'yakonov proposes the iteration scheme

$$\nabla^2 u^{(n+1)} = \nabla^2 u^{(n)} - \alpha(Du^{(n)} - f), \tag{6.14}$$

where $\alpha = 2/(K_{\min} + K_{\max})$. The error $\|u^{(n)} - u\|$ decreases by at least the factor $(K_{\max} - K_{\min})/(K_{\max} + K_{\min})$ at each iteration of (6.14). D'yakonov's method involves approximating the differential operator by a new, simpler, operator whose spectrum is bounded in terms of the original operator.

We propose to construct L_{ap} from L_{sp} by changing the discretization operator either in addition to or in place of approximating the differential operator. Thus, we construct L_{ap} by a suitable *low-order finite-difference approximation to L*.

A simple example is given by the second-order differential equation

$$Lu = f(x) u''(x) + g(x) u'(x) + h(x) u(x) = v(x) \quad (0 \leqslant x < 2\pi) \tag{6.15}$$

with periodic boundary conditions $u(x + 2\pi) = u(x)$ and assuming that L^{-1} exists and $f(x) > 0$. A spectral approximation is approximately sought as the finite Fourier series

$$u(x) = \sum_{|k| < K} a_k e^{ikx}. \tag{6.16}$$

If the Fourier coefficients of $f(x)$, $g(x)$, $h(x)$, $v(x)$ are denoted f_k, g_k, h_k, v_k, respectively, then the spectral (Galerkin) equations for a_k are

$$L_{sp}u = \sum_{\substack{|p|<K \\ |k-p|<K}} [-p^2 f_{k-p} + ipg_{k-p} + h_{k-p}]a_p = v_k. \tag{6.17}$$

Clearly, these equations have, in general, a full matrix representation that requires $O(K^2)$ storage locations and $O(K^3)$ operations to invert.

A suitable approximate operator L_{ap} is constructed using the collocation points $x_j = 2\pi j/N$ $(j = 0, 1,..., N-1)$, where $N = 2K$. In the physical space representation, we use the finite-difference approximation

$$L_{ap}u|_{x_j} = f(x_j)\frac{u_{j+1} - 2u_j + u_{j-1}}{(\Delta x)^2} + g(x_j)\frac{u_{j+1} - u_{j-1}}{2\Delta x} + h(x_j)u_j \tag{6.18}$$

where $u_j = u(x_j)$ and $\Delta x = 2\pi/N$. Obviously, L_{ap} is sparse and efficiently invertible. To verify (6.3) we use the following elementary argument (that may be made more rigorous but no more correct by more involved WKB-like arguments.) If λ is an eigenvalue of $L_{ap}^{-1}L_{sp}$ then there exists a function $u(x)$ such that

$$L_{sp}u = \lambda L_{ap}u. \tag{6.19}$$

If $u(x)$ is a smooth function of x (in the limit $N \to \infty$), then both $L_{ap}u$ and $L_{sp}u$ should be good approximations to $Lu(x)$ so (6.19) implies $\lambda \sim 1$. On the other hand, if $u(x)$ is a highly oscillatory function of x (in the limit $N \to \infty$) then

$$u'' \gg u' \gg u \qquad (N \to \infty). \tag{6.20}$$

Therefore,

$$L_{ap}u \sim f(x_j)\frac{u_{j+1} - 2u_j + u_{j-1}}{(\Delta x)^2} \tag{6.21}$$

and, if transform (pseudospectral) methods are used to evaluate $L_{sp}u$,

$$L_{sp}u \sim f(x_j)\sum_{|k|<K}(-k^2)a_k e^{ikx_j} \tag{6.22}$$

so (6.19) gives

$$f(x_j)\sum_{|k|<K}(-k^2)a_k e^{ikx_j} \sim \lambda f(x_j)\frac{u_{j+1} - 2u_j + u_{j-1}}{(\Delta x)^2}. \tag{6.23}$$

The eigenfunctions of (6.23) are

$$u_j = e^{iqx_j} \qquad (|q| < K) \tag{6.24}$$

and the associated eigenvalues are

$$\lambda = \frac{(q\Delta x)^2}{4 \sin^2 \frac{1}{2} q\Delta x}. \tag{6.25}$$

Since $|q| < K$ with $K = \frac{1}{2}N = \pi/\Delta x$, we obtain

$$1 \leqslant \lambda \leqslant \pi^2/4 \tag{6.26}$$

Thus, (6.3) holds with $m = 1$ and $M = \pi^2/4 \approx 2.5$.

There are several extensions of the above method for constructing L_{ap} that are important in practice. First, in the case of Chebyshev-spectral methods, it is appropriate to construct L_{ap} using finite-difference approximations based on the collocation points $x_j = \cos \pi j/N$. In this case, the operator bounds (6.3) continue to hold with $M = 2.5$, $m = 1$, for a wide variety of operators L. Second, higher-order equations are best treated by writing them as a system of lower-order equations. Thus, direct construction of L_{ap} for $L = \nabla^4$ gives

$$1 \leqslant \| L_{ap}^{-1} L_{sp} \| \lesssim 6 \approx (\pi^2/4)^2. \tag{6.27}$$

However, if we introduce $v = \nabla^2 u$ and define the second-order operator K by

$$K \begin{pmatrix} u \\ v \end{pmatrix} = \begin{matrix} \nabla^2 u - v \\ \nabla^2 v \end{matrix}, \tag{6.28}$$

then direct construction of K_{ap} as a finite-difference operator gives

$$1 \leqslant \| K_{ap}^{-1} K_{sp} \| \lesssim 2.5. \tag{6.29}$$

Third, odd-order operators, initial-value problems, and problems of mixed type are best treated by constructing L_{ap} on a grid that is roughly 50 % finer than that used in construction of L_{sp} by collocation. In this case the spectral bounds (6.3) with $M \lesssim 2.5$ continue to hold for most problems. For example, the operator $\partial/\partial x$ with periodic boundary conditions has spectrum ik while its centered finite-difference approximation has spectrum $i \sin(k\Delta x)/\Delta x$ so

$$\| L_{ap}^{-1} L_{sp} \| = O(k\Delta x / \sin k\Delta x),$$

which is unbounded for $|k\Delta x| < \pi$, but bounded by $4\pi/3(3)^{1/2} \approx 2.4$ if $|k\Delta x| < 2\pi/3$.

7. Examples

Consider the solution of the heat equation

$$\frac{\partial u}{\partial t} = \nabla^2 u - e^x - e^y, \qquad (x, y) \in D, \tag{7.1}$$

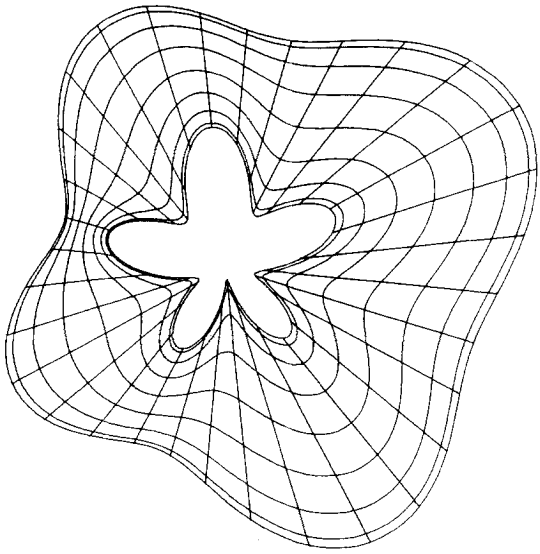

FIG. 1. A plot of the region D defined by (7.3)–(7.4) in which the diffusion equation (7.1)–(7.2) is solved. The collocation grid is also plotted for 32 collocation points in θ and 7 interior collocation points in z.

with Dirichlet boundary conditions

$$u(x, y) = e^x + e^y, \qquad (x, y) \in \partial D, \tag{7.2}$$

where D is the annular region plotted in Fig. 1 whose inner and outer boundaries are, respectively (in polar coordinates),

$$r = f_1(\theta) = 0.3 + 0.1 \sin \theta + 0.15 \sin 5\theta \tag{7.3}$$

and

$$r = f_2(\theta) = 1 + 0.2 \cos \theta + 0.15 \sin 4\theta. \tag{7.4}$$

TABLE I

Errors in Steady-State Solution of (7.1)–(7.4)

Number of angular modes (θ)	Number of "radial" Chebyshev modes (z)	Maximum relative error
16	4	1.4×10^{-2}
32	8	2.8×10^{-5}
64	16	2.5×10^{-10}

This problem is solved spectrally using the stretching transformation (5.3) to transform D into the rectangular domain (5.2) and then using Fourier series in θ and Chebyshev series in z to represent $u(x, y)$.

As $t \to \infty$, the solution to (7.1)–(7.4) approaches the steady-state solution

$$u(x, y) = e^x + e^y, \qquad (x, y) \in D. \tag{7.5}$$

In Table I, we list the maximum pointwise errors in the spectral solution of this problem as $t \to \infty$. Evidently the rapid convergence of the spectral solution to the exact steady state is preserved in this complex geometry problem. It is not very surprising that, for a given total number of retained modes, the best accuracy is achieved with about 4 times more "angular" (θ) than "radial" (z) modes; after all, the "annulus" (7.3)–(7.4) is on average about π times longer in circumference than in radius.

Another example of the technique suggested in Section 6 is the solution of the boundary-layer equations

$$\frac{\partial^3 f}{\partial \eta^3} + f\frac{\partial^2 f}{\partial \eta^2} + \beta(\xi)\left[1 - \left(\frac{\partial f}{\partial \eta}\right)^2\right] = 2\xi\left[\frac{\partial f}{\partial \eta}\frac{\partial^2 f}{\partial \xi\, \partial \eta} - \frac{\partial^2 f}{\partial \eta^2}\frac{\partial f}{\partial \xi}\right], \tag{7.6}$$

where $f(\xi, \eta)$ is the dimensionless stream function and is subject to the boundary conditions

$$f(\xi, 0) = \frac{\partial f}{\partial \eta}(\xi, 0) = 0, \tag{7.7}$$

$$\frac{\partial f}{\partial \eta}(\xi, \eta) \to 1 \qquad (\eta \to \infty). \tag{7.8}$$

Here (ξ, η) are the Levy–Lees coordinates; ξ increases in the free-stream direction and η increases away from the wall. The non-self-similar solution of these equations for Howarth's flow in which the pressure-gradient parameter $\beta(\xi)$ is given by

$$\beta(\xi) = \frac{\xi}{\xi - 4} \tag{7.9}$$

is a standard test problem [13]. We solve (7.6)–(7.9) using Chebyshev series in the transformed variable

$$s = 2\eta/R - 1 \qquad (-1 \leqslant s \leqslant 1),$$

where R is chosen so $\eta = R$ is in the free stream and a Crank–Nicolson scheme is used in ξ. The finite-difference errors in ξ are reduced by Richardson extrapolation [13]. The results for the viscous wall stress at two downstream locations near the separation point $\xi \approx 0.901$ are given in Table II together with some corresponding finite-difference results from Ref. [13]. These results illustrate the high accuracy of the spectral schemes with modest resolution.

TABLE II

Viscous Wall Stresses for Howarth's Flow

Method	Truncation $R, 0 \leqslant s \leqslant R$	Number of grid points (modes) in η	Number of grid points in ξ	Total number of points	A	B
Spectral	8	23	6	138	0.21239	0.04454
Spectral	8	23	11	253	0.20913	0.03471
Spectral with Richardson extrapolation in ξ	8	23	{6,11}	253	0.20718	0.03064
Finite difference [13]	6	121	51	6171	0.20723	0.03083
Finite difference with Richardson extrapolation in ξ and/or η [13]	6	19	16	489	0.20710	0.03171
	6	61	16	1708	0.20741	0.03121
	6	121	51	9282	0.20714	0.03053

$A = \frac{\partial^2 f}{\partial \eta^2}(\xi = 0.7, \eta = 0) \approx 0.20724$ $\quad$ $B = \frac{\partial^2 f}{\partial \eta^2}(\xi = 0.894, \eta = 0) \approx 0.03072$

8. Other Developments

In this paper, we have explained how to apply spectral methods efficiently to [redacted] and results will be presented elsewhere.

There have been several other recent developments of spectral methods that relate to these applications but have independent interest. First, Lagrangian spectral methods have been developed for the solution of high-speed flows. In this case, Lagrangian marker particles are used to provide the coordinate transformations necessary for the methods introduced in this paper. Second, fast conformal mapping techniques have been developed. These techniques may be attractive for the solution of free-surface flow problems by spectral methods.

Acknowledgments

This work was supported by the Defense Advanced Research Projects Agency and the Office of Naval Research under Contract N00014-77-C-0138, the National Science Foundation under Grant No. ATM-7817092, and the Los Alamos Scientific Laboratory. The boundary-layer calculations reported in Section 7 were done with the assistance of Dr. L. Jayne and were partially supported by NASA under Contract NAS1-15372.

References

1. D. Gottlieb and S. A. Orszag, "Numerical Analysis of Spectral Methods," NSF-CBMS Monograph No. 26, Soc. Ind. and Appl. Math., Philadelphia, 1977.
2. A. Majda, J. McDonough, and S. Osher, *Math. Comput.* **32** (1978), 1041.
3. S. A. Orszag, *Stud. Appl. Math.* **50** (1971), 293.
4. S. A. Orszag, to be published.
5. S. A. Orszag and M. Israeli, *Ann. Rev. Fluid Mech.* **6** (1974), 281.
6. C. W. Gear, "Numerical Initial Value Problems in Ordinary Differential Equations," Prentice–Hall, Englewood Cliffs, N.J., 1971.
7. J. Oliger, *Math. Comput.* **28** (1974), 15.
8. E. Turkel and D. Gottlieb, *Stud. Appl. Math.*, in press.
9. J. F. Thompson, F. C. Thames, and C. W. Mastin, *J. Comput. Phys.* **15** (1974), 299.
10. R. S. Varga, "Matrix Iterative Analysis," Prentice–Hall, Englewood Cliffs, N.J., 1962.
11. M. R. Hestenes and E. Stiefel, *J. Res. Nat. Bur. Stand.* **49** (1952), 409.
12. E. G. D'yakonov, *Dokl. Akad. Nauk SSSR* **138** (1961), 522.
13. H. B. Keller and T. Cebeci, *in* "Proceedings, Second Intl. Conf. on Num. Meth. in Fluid Dynamics," p. 92, Springer-Verlag, Berlin, 1971.