JOURNAL OF COMPUTATIONAL PHYSICS 48, 485–501 (1982)

Spectral Multigrid Methods for Elliptic Equations*

THOMAS A. ZANG

College of William and Mary, Williamsburg, Virginia 23185

AND

YAU SHU WONG AND M. Y. HUSSAINI

Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, Virginia 23665

Received January 13, 1982

This paper examines the multigrid procedures applied to the iterative solution of spectral equations. Spectral multigrid methods are described for selfadjoint elliptic equations with either periodic or Dirichlet boundary conditions. These methods show a substantial improvement over the simplest iterative schemes.

INTRODUCTION

Two computational approaches which achieved substantial popularity during the past decade are spectral methods and multigrid techniques. The former have proven highly efficient for time-dependent smooth flows in simple geometries [1–3]. The latter have been remarkably successful for elliptic equations and some steady state calculations [4–7]. The principal advantage of spectral methods lies in their ability to achieve accurate results with substantially fewer grid points than required by typical finite difference methods. Despite the fact that spectral methods are represented by full matrices, explicit time-stepping algorithms can be implemented nearly as efficiently for them as for finite difference methods on a comparable grid. Transform methods [8] are often the key to this efficiency. For implicit methods or for steady state equations, direct solution of the spectral equations is generally not practical. Iterative schemes for such equations are essential. Orszag [9] has described several attractive methods.

This paper examines an alternative approach which employs multigrid concepts in the iterative solution of spectral equations. In particular, spectral multigrid methods are described for selfadjoint elliptic equations with either periodic or Dirichlet

* Research of the first author was supported by NASA Grant NAG1-109; research of the second and third authors was supported by NASA Contract Nos. NAS1-15810 and NAS1-16394, respectively, while they were in residence at ICASE, NASA Langley Research Center, Hampton, Va. 23665.

0021-9991/82/120485–17$02.00/0

boundary conditions. For realistic fluid calculations, the relevant boundary conditions are likely to be periodic in at least one (angular) coordinate and Dirichlet (or Neumann) in the remaining coordinates. Spectral methods may not always be effective for flows in strictly rectangular geometries, since corners generally introduce singularities into the solution. These singularities can seriously degrade the accuracy of a spectral method. If the boundary is smooth, then mapping techniques [9] can often be used to transform the problem into one with a combination of periodic and Dirichlet boundary conditions. Spectral multigrid methods in these geometries can be devised by combining the techniques presented separately here.

Periodic Problems

Fourier Spectral Approximations

Several types of spectral approximations can be employed. The specific method used here is often termed collocation or pseudospectral approximation. In many cases, this method is easier to implement and is more efficient than the alternative Galerkin and tau approximations. A thorough discussion of all these methods can be found in [3].

For a periodic problem, spectral approximations should be based upon Fourier series. In the collocation approach, the fundamental representation of the solution remains in physical space. The Fourier coefficients are only employed as an intermediate result in the approximate evaluation of derivatives. Consider a function $u(x)$ which is periodic over the interval $[0, 2\pi]$. Use N evenly spaced collocation points

$$x_j = 2\pi j/N, \qquad j = 0, 1,..., N-1, \tag{1}$$

and denote $u(x_j)$ by u_j. The first step in the evaluation of du/dx is the computation of the approximate Fourier coefficients $\hat{u}_p$ via

$$\hat{u}_p = (1/\sqrt{N}) \sum_{j=0}^{N-1} u_j e^{-ipx_j}, \qquad p = -N/2, -N/2+1,..., N/2-1. \tag{2}$$

Since the $u(x_j)$ are real, $\hat{u}_{-N/2}$ is real and $\hat{u}_{-p} = \hat{u}_p^*$ for $|p| < N/2$, where the $*$ denotes complex conjugation. The derivative is then computed via

$$du/dx(x_j) = (1/\sqrt{N}) \sum_{p=-N/2+1}^{N/2-1} ip\hat{u}_p e^{ipx_j}, \qquad j = 0, 1,..., N-1. \tag{3}$$

Both sums can be evaluated in $O(N \ln N)$ operations by the fast Fourier transform [10]. This algorithm is most commonly employed with N chosen to be a power of 2.

Note that the lower limit on the sum in Eq. (3) is not $p = -N/2$ but

$p = -N/2 + 1$. This change is equivalent to setting $\hat{u}_{-N/2} = 0$. The right-hand side of Eq. (3) is necessarily real. The neglected term

$$-i(N/2)\, e^{-i(N/2)x_j}$$

is purely imaginary and cannot contribute to $du/dx(x_j)$. This neglected term represents the familiar *two-point oscillation* in $u(x)$. (Finite difference schemes which use central differences for first derivatives also remove the two-point oscillation.)

The spectral evaluation of derivatives has a convenient matrix representation. Let $\mathbf{U}$ denote the vector of the solution at the grid, or collocation, points, i.e.,

$$\mathbf{U} = (u_0, u_1, ..., u_{N-1}), \tag{4}$$

let $\mathbf{C}$ represent the discrete Fourier transform, i.e.,

$$\mathbf{C}_{jl} = (1/\sqrt{N})\, e^{-2\pi i l(j-(N/2))/N}, \qquad j, l = 0, 1, ..., N-1, \tag{5}$$

and let $\mathbf{D}$ be the diagonal matrix which represents the first derivative in Fourier space, i.e.,

$$\begin{aligned} \mathbf{D}_{jj} &= i(j - N/2), && \text{for} \quad j = 1, 2, ..., N-1, \\ &= 0, && \text{for} \quad j = 0. \end{aligned} \tag{6}$$

Note that $\mathbf{C}^{-1} = \mathbf{C}^*$, the Hermitian transpose of $\mathbf{C}$. Then the matrix

$$\mathbf{M} = \mathbf{C}^{-1}\mathbf{D}\mathbf{C} \tag{7}$$

represents (in physical space) the spectral evaluation of a first derivative. This matrix is given explicitly by

$$\mathbf{M}_{jl} = \tilde{\mathbf{M}}_{j-l}, \tag{8}$$

where

$$\begin{aligned} \tilde{\mathbf{M}}_j &= 0, && \text{if} \quad j = 0, \pm N, \pm 2N, ..., \\ &= \cos(1 - 1/N)\, \pi j/(2 \sin(\pi j/N)), && \text{otherwise.} \end{aligned} \tag{9}$$

A spectral approximation to the ordinary differential equation

$$(d/dx)\{a(x)\, du/dx\} = f(x) \tag{10}$$

on $[0, 2\pi]$ with periodic boundary conditions, and with $a(x)$ and $f(x)$ infinitely differentiable as well as periodic, satisfies the discrete equation

$$\mathbf{L}\mathbf{U} = \mathbf{F}, \tag{11}$$

where

$$\mathbf{L} = \mathbf{MAM}, \tag{12}$$

$$\mathbf{A}_{jl} = a(x_j)\,\delta_{j,l}, \tag{13}$$

and

$$\mathbf{F} = (f_0, f_1, \ldots, f_{N-1}). \tag{14}$$

Equation (11) may be inverted to yield

$$\mathbf{U} = (\mathbf{C}^{-1}\mathbf{D}^{-1}\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^{-1}\mathbf{D}^{-1}\mathbf{C})\,\mathbf{F}. \tag{15}$$

Although the matrix **D** is technically singular, this merely reflects the usual nonuniqueness of the solution of Eq. (10). All of the matrix multiplies required by the right-hand side of Eq. (15) may be implemented efficiently. There are three diagonal matrices and four Fourier transforms. Thus, the solution to Eq. (11) can be obtained directly in $O(N \ln N)$ operations, even though the matrix **L** is full.

Unfortunately, efficient direct solutions are not available in higher dimensions. Consider the self-adjoint elliptic equation

$$\frac{\partial}{\partial x}\left\{a(x,y)\frac{\partial u}{\partial x}\right\} + \frac{\partial}{\partial y}\left\{b(x,y)\frac{\partial u}{\partial y}\right\} = f \tag{16}$$

on the square $[0, 2\pi] \times [0, 2\pi]$. Again impose periodic boundary conditions and assume that the functions a, b, and f are also periodic as well as infinitely differentiable. A spectral approximation to Eq. (16) will exhibit exponential convergence, i.e., the error will ultimately decrease faster than any finite inverse power of the number of collocation points.

For simplicity, suppose that an $N \times N$ mesh is employed. Define the approximate solution

$$\mathbf{U}_{jl} = u(x_j, y_l) \qquad \text{for} \quad j, l = 0, 1, \ldots, N-1. \tag{17}$$

Define **F** in a similar fashion. The discrete approximation to Eq. (16) is

$$\mathbf{LU} = \mathbf{F}, \tag{18}$$

where the fourth-order tensor L is defined by

$$\mathbf{L} = (\mathbf{M} \otimes \mathbf{I})\,\mathbf{A}(\mathbf{M} \otimes \mathbf{I}) + (\mathbf{I} \otimes \mathbf{M})\,\mathbf{B}(\mathbf{I} \otimes \mathbf{M}), \tag{19}$$

with $\otimes$ denoting a tensor product and **I** representing the identity matrix of order N. The fourth-order tensors **A** and **B** represent the contributions of the variable coefficients $a(x, y)$ and $b(x, y)$, respectively.

The authors are unaware of any efficient method for solving Eq. (18) directly. The

iterative methods described in [9] are one possible solution scheme. A different sort of iterative method—one involving the use of multiple grids—is described below.

Euler Iteration on a Single Grid

The direct solution of the $N^2 \times N^2$ system represented by Eq. (18) would require $O(N^4)$ storage locations and $O(N^6)$ operations. Many iterative schemes require only $O(N^2)$ storage locations and $O(N^2 \ln N)$ operations per step. Perhaps the simplest iterative scheme is the Euler method

$$\mathbf{U} \leftarrow \mathbf{U} - \omega(\mathbf{F} - \mathbf{L}\mathbf{U}), \tag{20}$$

where ω is a relaxation parameter. Aside from the coefficients $a(x_j, y_l)$ and $b(x_j, y_l)$, the only substantial storage required is for the residual (the term in parentheses in Eq. (20)], which is clearly $O(N^2)$. The tensor $\mathbf{L}$ is never explicitly required. The residual itself costs $O(N^2 \ln N)$ operations to compute. Jacobi's method (see below) is also economical in storage and cost per step. Not all iterative schemes used to solve finite difference equations are practical for the spectral equations, however. Gauss–Seidel is an obvious example. The term $\mathbf{L}\mathbf{U}$ can only be evaluated efficiently if it is done all at once.

It is instructive to consider the application of the Euler iteration to the constant coefficient case $a(x, y) = b(x, y) = 1$. The tensor $\mathbf{L}$ simplifies to

$$\mathbf{L} = \mathbf{M}^2 \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{M}^2. \tag{21}$$

The eigenvalues and eigenvectors of $\mathbf{L}$ are

$$\lambda_{pq} = -(p^2 + q^2), \tag{22}$$

$$\xi_{jl}(p, q) = e^{(2\pi i/N)(pj + ql)}, \tag{23}$$

where the eigenvalues and eigenvectors are labelled by p and q, which lie in the range $p, q = -N/2, -N/2 + 1, \ldots, N/2 - 1$. In Eq. (22), if either p or $q = -N/2$, then that term should be replaced by 0 on the right-hand side. A single iteration by Eq. (20) replaces the error component $\xi(p, q)$ with $(1 + \omega\lambda_{pq})\,\xi(p, q)$. There are two eigenvectors which are unaffected by the iteration. One of these—for $p = q = 0$—represents the mean level of the solution. It must be specified for the partial differential equation to have a unique answer. The other term—for $p = q = -N/2$—represents the high-frequency component that is ignored by the discretization. This component should be filtered out of the right-hand side $\mathbf{F}$.

This scheme is convergent if

$$\omega < -2/\lambda_{N/2-1, N/2-1} = 4/(N-2)^2. \tag{24}$$

The smallest spectral radius

$$\rho = (N^2 - 4N + 2)/(N^2 - 4N + 6) \cong 1 - 4/N^2, \tag{25}$$

is obtained when

$$\omega = 4/(N^2 - 4N + 6). \tag{26}$$

According to the usual reasoning, Eq. (25) implies $O(N^2)$ iterations are required. This means a total of $O(N^4 \ln N)$ operations are required in order to solve Eq. (18) in this fashion.

Euler Iteration Using Multiple Grids

Multigrid methods have become a standard means of accelerating convergence for finite difference and finite element discretizations of elliptic equations. The basic processes are the relaxation scheme and the transfer of residuals and corrections between the various grids. In addition to specific choices of relaxation and interpolation procedures, a multigrid algorithm must give rules governing the transfer between grids. A variety of control structures for this latter process have been employed. For examples of some of the control structures, see the flow charts in [5]. The present discussion will focus on the relaxation and interpolation procedures, since they are less arbitrary than the control structure. Moreover, the description will be given for the spectral discretization of the one-dimensional problem (Eqs. (11) and (12)). This is done simply for notational convenience. The performance will be assessed, and numerical examples given, however, for the two-dimensional case.

Define a series of grids (or levels) G_k, for $k = 2, 3, \ldots, K$ covering the interval $[0, 2\pi]$. Let G_k consist of N_k uniformly spaced points, where $N_k = 2^k$. The solution to Eq. (11) is obtained by combining Euler iterations on level K with Euler iterations for related problems on the coarser levels $k < K$. Denote the relevant discrete problem at any level k by

$$\mathbf{L}_k \mathbf{V}_k = \mathbf{F}_k. \tag{27}$$

On the finest level K, $\mathbf{L}_K = \mathbf{L}$, $\mathbf{F}_K = \mathbf{F}$, and the solution $\mathbf{V}_K = \mathbf{U}$, the solution to Eq. (11). At any stage in the iterative solution process for Eq. (27), only an approximation $\mathbf{v}_k$ to the exact answer $\mathbf{V}_k$ is available. If this approximation is deemed adequate, then the approximation on the next-finer level $k + 1$ is corrected via

$$\mathbf{v}_{k+1} \leftarrow \mathbf{v}_{k+1} + \mathbf{P}_{k+1} \mathbf{v}_k. \tag{28}$$

The matrix $\mathbf{P}_k$ represents the coarse-to-fine transfer of corrections from level $k - 1$ to level k. On the other hand, if the approximation $\mathbf{v}_k$ is deemed inadequate, either another relaxation is performed, via

$$\mathbf{v}_k \leftarrow \mathbf{v}_k - \omega_k(\mathbf{F}_k - \mathbf{L}_k \mathbf{v}_k), \tag{29}$$

or else control shifts to a problem on the next-coarser level $k - 1$. The relaxation parameter ω_k on level k is chosen to damp preferentially those error components

which are not represented on coarser grids. The right-hand side of the coarser grid problem is obtained from

$$\mathbf{F}_{k-1} = \mathbf{R}_k(\mathbf{F}_k - \mathbf{L}_k \mathbf{v}_k). \tag{30}$$

The matrix $\mathbf{R}_k$ represents the fine-to-coarse residual transfer from level k to level $k-1$.

For the spectral multigrid method, the natural interpolation operators represent trigonometric rather than polynomial interpolation. For the one-dimensional case,

$$\mathbf{R}_k = \mathbf{C}_{k-1}^{-1} \mathbf{E}_{k-1} \mathbf{C}_k, \tag{31}$$

$$\mathbf{P}_k = \mathbf{C}_k^{-1} \mathbf{E}_{k-1}^{\mathrm{T}} \mathbf{C}_{k-1}, \tag{32}$$

where the $N_k \times N_{k+1}$ matrix

$$\mathbf{E}_k = (0|\mathbf{I}_k|0) \tag{33}$$

(with $\mathbf{I}_k$ the identity matrix of order N_k), $\mathbf{E}_k^{\mathrm{T}}$ is its transpose, and $\mathbf{C}_k$ is the matrix given in Eq. (5) for $N = N_k$. The matrix $\mathbf{E}_k$ represents the dropping of the high-frequency Fourier coefficients in the trigonometric interpolation from the fine grid to the coarse grid. Note that $\mathbf{P}_k = \mathbf{R}_k^*$. The generalization to higher dimensions is straightforward.

For the constant-coefficient one-dimensional case, the finest grid relaxation operator

$$\mathbf{L}_K = \mathbf{C}_K^{-1} \mathbf{D}^2 \mathbf{C}_K, \tag{34}$$

and it is natural to use

$$\mathbf{L}_k = \mathbf{C}_k^{-1} \mathbf{D}_k^2 \mathbf{C}_k \tag{35}$$

for $k < K$. It is easy to show that

$$\mathbf{L}_{k-1} = \mathbf{R}_k \mathbf{L}_k \mathbf{P}_k. \tag{36}$$

The description of the variable coefficient relaxation operator is more complicated and the details will be published elsewhere. The procedure used in the numerical experiments reported below amounts to performing the collocation operations in an alias-free fashion.

For the two-dimensional Poisson equation discussed in the previous section, the level k relaxation parameter ω_k is chosen to maximize the smoothing of all the modes except those for which $|p|, |q| < N_k/4$,

$$\omega_k = 2/((9/16)\, N_k^2 - 2N_k + 2). \tag{37}$$

This choice produces a smoothing rate for the high-frequency modes of

$$\mu_k = 1 - 2N_k^2/(9N_k^2 - 32N_k + 32): \tag{38}$$

TABLE I

Convergence Rates for Euler Iteration in Two Dimensions

N	Single grid spectral radius	Multigrid smoothing rate
4	0.3333	0.3333
8	0.8947	0.6364
16	0.9798	0.7193
32	0.9956	0.7510
64	0.9990	0.7649
∞	1.0000	0.7778

after one relaxation, the amplitude of each high-frequency mode is at most μ_k times its previous amplitude. This smoothing rate is listed in Table I alongside the spectral radius for the single grid Euler method. The advantage of multiple gridding is apparent. For large $N_k, \mu_k \simeq 7/9$. Thus, according to the usual multigrid argument, the number of iterations needed to obtain a given reduction in the residual should be independent of the number of grid points on the finest grid. This assumes, of course, no untoward effects of the interpolation process. But the trigonometric interpolation procedure used here is ideally suited to minimize the spurious generation of high-frequency components at these stages.

Alternatives to Euler Relaxation

A straightforward improvement upon the simple relaxation scheme described in the preceding subsection is to make it nonstationary. This approach has been used for accelerating point-Jacobi iterations for finite difference multigrid algorithms (see [11]). The nonstationary Euler iteration consists of using n relaxation parameters $\omega_{k,1}, \omega_{k,2}, \ldots, \omega_{k,n}$ in a cyclic fashion on each level k. These parameters are determined from the solution of a standard minimax problem over the interval covered by the high-frequency eigenvalues.

For the two-dimensional Poisson equation, this eigenvalue range is from $-(N_k/4)^2$ to $-(N_k/2-1)^2$. The results are only changed slightly if the upper limit of this range is changed to $-(N_k/2)^2$. Then the optimal parameters are given by

$$\omega_{k,j} = (32/N_k^2)/(7\cos(j-1/2)\,\pi/n + 9) \tag{39}$$

and the total smoothing of the high frequencies after the full n relaxations is $1/|T_n(-9/7)|$, where $T_n(x)$ is the Chebyshev polynomial of degree n. Then the effective smoothing rate is

$$\bar{\mu}_k = 1/|T_n(-9/7)|^{1/n}, \tag{40}$$

which is the average smoothing per single step in the cyclic relaxation. The values are given in Table II along with the corresponding effective smoothing rates for a finite

TABLE II

Smoothing Rates for Euler Iteration on Poisson's Equation

Number of parameters	Spectral smoothing rate	Finite difference smoothing rate
1	0.7778	0.6000
2	0.6585	0.4685
3	0.5995	0.4198
4	0.5676	0.3964
5	0.5485	0.3749

difference multigrid method which also is relaxed with Euler iteration. The spectral smoothing rates are larger than the finite difference ones because the ratio of the largest high-frequency eigenvalue to the smallest high-frequency eigenvalue is 8 in the former case and only 4 in the latter. This ratio may be termed the multigrid condition number. The higher smoothing rate for the spectral method suggests that a larger number of distinct relaxation parameters should be used here than for the finite difference case.

It should be kept in mind that this larger eigenvalue ratio for the spectral method occurs because this method represents the larger eigenvalues of the partial differential equation much better than finite difference methods. Indeed, it is just this property which is responsible for the exponential convergence rate of spectral methods as N is increased and for their low phase error in time-dependent calculations.

Another obvious relaxation scheme is point-Jacobi. The actual implementation of this method requires that the diagonal elements of the matrix $\mathbf{L}$ be known explicitly. Consider the one-dimensional situation, where $\mathbf{L}$ is given by Eq. (12) for the general case. It would appear that the evaluation of the elements $\mathbf{L}_{jj}$ requires $O(N^2)$ operations. This would be impractical since the results of the previous section suggested that only $O(N \ln N)$ operations are needed to get the solution itself.

Nonetheless, Jacobi relaxation is worth considering since transform methods may be employed to compute the requisite diagonal elements in only $O(N \ln N)$ operations. It is clear from Eq. (9) that $\tilde{\mathbf{M}}_j$ is odd in j. Thus,

$$\mathbf{L}_{jj} = -\sum_{l=o}^{N-1} \tilde{\mathbf{M}}_{j-l}\, a_l. \tag{41}$$

But this is a convolution sum and may be evaluated efficiently by the transform methods described in [8]. Therefore, even for nonlinear problems, Jacobi relaxation may be implemented efficiently.

The spectral multigrid method was implemented for the two-dimensional problem (Eq. (16)) for which the coefficients are

$$a(x, y) = b(x, y) = 1 + \varepsilon\, e^{\cos(x+y)} \tag{42}$$

and the solution itself is

$$u(x, y) = \sin(\pi \cos x + \pi/4) \sin(\pi \cos y + \pi/4). \tag{43}$$

The Fourier coefficients of this function may be expressed in terms of Bessel functions. This function is used in [3, pp. 35–37] to illustrate exponential convergence. The term $\pi/4$ serves to make all the Fourier coefficients nonzero. The constant ε in Eq. (42) measures the departure of the equation from the strictly Poisson form.

A simple control structure was selected for the multigrid algorithm: start on the finest level; perform a single relaxation followed by a fine-to-coarse residual transfer on each level in turn until the coarsest level $k = 2$; there iterate until convergence; then work back up to the finest level, using the coarse-to-fine transfer of corrections followed by a single relaxation on each intermediate level. This process is repeated until the desired accuracy is achieved. This algorithm requires more frequent interpolation but is less arbitrary than many alternatives. Despite the necessity for employing the fast Fourier transform in the trigonometric interpolations, this portion of the computations takes less than 10% of the total computation time.

The root mean square (or discrete l_2 norm) results of calculations for which the finest level K is 5 are shown in Tables III and IV. The nonstationary Euler iteration used three distinct parameters. The transfer between grids does not occur until all three relaxations have been performed. The residuals are listed in the tables after every three relaxations on the finest grid. The number in parentheses is the exponent of the residual. For comparison purposes, note that Euler iteration on a single grid exhibits a residual of about 10 after 15 relaxations. The multigrid results are a marked improvement.

On a 32×32 grid, the true solution of Fourier collocation Eq. (18) has an RMS error of 5.08 (−10) compared with the exact solution of Eq. (43) for $\varepsilon = 0.0$. The RMS error of the nonstationary iteration after 15 fine-grid relaxations is 2.20(−7). To get the full accuracy out of a spectral method, it may be necessary to reduce the residual by many orders of magnitude. By contrast, a second-order finite difference

TABLE III

RMS Residuals for Fourier Spectral Multigrid Using Stationary Euler Iteration

Relaxation number	$\varepsilon = 0.0$	$\varepsilon = 0.1$	$\varepsilon = 0.2$
3	2.92 (1)	3.23 (0)	3.72 (0)
6	2.27 (−1)	2.49 (−1)	3.12 (−1)
9	3.24 (−2)	3.52 (−2)	4.40 (−2)
12	1.02 (−2)	1.11 (−2)	1.37 (−2)
15	4.00 (−3)	4.37 (−3)	5.55 (−3)

TABLE IV

RMS Residuals for Fourier Spectral Multigrid
Using Nonstationary Euler Iteration

Relaxation number	$\varepsilon = 0.0$	$\varepsilon = 0.1$	$\varepsilon = 0.2$
3	2.82 (1)	3.12 (1)	3.47 (1)
6	2.42 (−1)	2.10 (−1)	2.56 (−1)
9	6.35 (−3)	3.68 (−3)	5.57 (−3)
12	4.56 (−4)	3.19 (−4)	6.30 (−4)
15	8.30 (−5)	5.36 (−5)	1.17 (−5)

approximation on a 32×32 grid gives an RMS error of 7.64(−2) for the $\varepsilon = 0.0$ problem. Even a fourth-order method gives only 5.04(−3). For this problem, at least, it seems worthwhile to accept the less advantageous smoothing rate of the spectral multigrid method (see Table II), since a far smaller grid can be used than for a finite difference method.

Dirichlet Problems

Chebyshev Spectral Approximations

For problems with Dirichlet (or Neumann) boundary conditions, spectral approximations should be based upon Chebyshev series. The standard interval is $|-1, 1|$ and the collocation points are

$$x_j = \cos(2\pi j/N), \qquad j = 0, 1, \ldots, N. \tag{44}$$

The analog to Eq. (7) with Dirichlet boundary conditions may be written in the form of Eqs. (11)–(14), where now

$$\mathbf{C}_{jl} = (2/N\bar{c}_j\bar{c}_l)\cos(\pi jl/N), \qquad j, l = 0, 1, \ldots, N, \tag{45}$$

$$\begin{aligned} \bar{c}_j &= 2, && j = 0 \text{ or } j = N, \\ &= 1, && 1 \leqslant j \leqslant N, \end{aligned} \tag{46}$$

$$\begin{aligned} \mathbf{D}_{jl} &= 2l/c_j, && l \geqslant j+1 \quad \text{and } l = j+1 \pmod 2, \\ &= 0, && \text{otherwise}, \end{aligned} \tag{47}$$

and

$$\begin{aligned} c_j &= 2, && j = 0, \\ &= 1, && j \geqslant 1. \end{aligned} \tag{48}$$

Many details about Chebyshev collocation can be found in [3]. The matrix **M**, which represents the Chebyshev approximation to a first derivative, is again given by Eq. (7), where now

$$\begin{aligned} \mathbf{M}_{jl} &= \bar{c}_j(\tilde{\mathbf{M}}_{j+l} + \tilde{\mathbf{M}}_{j-l})/(\bar{c}_l \sin(\pi j/N)) \qquad \text{for} \quad 1 \leqslant j \leqslant N-1, \\ \mathbf{M}_{00} &= -\mathbf{M}_{NN} = (2N^2+1)/6, \\ \mathbf{M}_{0l} &= -M_{Nl} = 2(-1)^l/(1-\cos(\pi l/N)) \qquad \text{for} \quad 1 \leqslant l \leqslant N-1, \end{aligned} \tag{49}$$

where

$$\begin{aligned} \tilde{\mathbf{M}}_j &= 0, && j = 0, \pm 2N, \pm 4N, \ldots, \\ &= (1/2)(-1)^{j+1}\cot(\pi j/N), && \text{otherwise.} \end{aligned} \tag{50}$$

Once more **M** is a full matrix but the product **MU** can be evaluated in $O(N \ln N)$ operations.

Preconditioned Euler Iteration Using Multiple Grids

The direct analog of the Euler iteration method described in the preceding section is not practical for the Dirichlet problem. The difficulty is that for the Chebyshev second derivative operator, the multigrid condition number grows as N^2. In the one-dimensional case Gershgorin's Theorem can be used to show that the largest eigenvalue grows as N^4 [3]. All but the several largest eigenvalues are good approximations to the eigenvalues of the continuous problem. Thus, the smallest high-frequency eigenvalue grows as N^2. (Direct numerical computation of the eigenvalues supports these conclusions.) Since the ratio of these two eigenvalues (the multigrid condition number) is N^2, the smoothing rate of a straightforward Chebyshev–Euler multigrid method is of the same order as the spectral radius of the Fourier–Euler iteration on a single grid (see Table I). The nonstationary Chebyshev–Euler multigrid method has the same problem.

Clearly, preconditioning is essential for an effective Chebyshev spectral multigrid algorithm based on Euler iteration. Thus, in place of Eq. (20), the relaxation scheme is

$$\mathbf{U} \leftarrow \mathbf{U} - \omega \mathbf{H}^{-1}(\mathbf{F} - \mathbf{L}\mathbf{U}), \tag{51}$$

where the preconditioning matrix is denoted by **H**. An effective preconditioning matrix has been devised by Orszag [9] for finding solutions iteratively on a single grid to Chebyshev spectral approximations. That preconditioning matrix, denoted here by **S**, is a full finite difference approximation to the spectral matrix **L**. Orszag noted that the conventional condition number of the matrix $\mathbf{S}^{-1}\mathbf{L}$ should be about 2.4, regardless of N.

The preconditioning matrix employed in the present spectral multigrid calculations

is a cheaper but less precise version of **S**. Instead of using **S** itself, an approximate lower-triangular/upper-triangular decomposition of **S** is used as **H**, i.e.,

$$\mathbf{H} = \mathscr{L}\mathscr{U}, \tag{52}$$

where script letters are used to denote the lower-triangular ($\mathscr{L}$) and upper-triangular ($\mathscr{U}$) factors. This matrix **H** is cheaper to employ than **S** because $\mathbf{H}^{-1}$ can be found by simple forward- and back-substitutions, whereas finding $\mathbf{S}^{-1}$ amounts to computing the solution to a finite difference discretization of the problem.

To determine **H**, one starts with **S** as a standard finite difference approximation to Eq. (16) on the nonuniform grid of the Chebyshev collocation points. The matrices $\mathscr{L}$ and $\mathscr{U}$ are determined by the row sum agreement factorization which enforces the following conditions:

(1) $\mathscr{L}$ and $\mathscr{U}$ have nonzero elements only in those positions which correspond to the nonzero elements in the lower- and upper-triangular part of **S** itself.

(2) Whenever $\mathbf{S}_{jl} \neq 0$ and $j \neq l$, then $\mathbf{H}_{jl} = \mathbf{S}_{jl}$. (The off-diagonal elements of **H** whose locations correspond to the nonzero off-diagonal elements of **S** are set to those values.)

(3) The row sums of **H** are the same as those of **S**.

For further details on this sort of preconditioning, see [12].

The decreased accuracy of the matrix **H** is indicated in Table V, which lists the smallest and largest eigenvalues of the preconditioned matrix $\mathbf{H}^{-1}\mathbf{L}$. In contrast to the matrix $\mathbf{S}^{-1}\mathbf{L}$, for which the largest eigenvalue is roughly 2.4, the largest eigenvalue here shows a slow growth with N, evidently increasing as $\sqrt{N}$. Both matrices yield essentially the same value for the smallest eigenvalue. Moreover, the smallest high-frequency eigenvalue of $\mathbf{H}^{-1}\mathbf{L}$ stays roughly constant—at about 1.45—as N increases. Thus, the multigrid condition number of this preconditioned Euler method increases slowly with N.

The eigenvalue results given above suggest that an Euler iteration scheme using the approximate $\mathscr{L}\mathscr{U}$ factorization form of preconditioning will have the convergence rates listed in Table VI. The advantage of using multiple grids here is not as great as in the periodic case. The basic problem is the slow growth of the multi-grid condition number with N. Clearly, better forms of preconditioning are needed.

TABLE V

Extreme Eigenvalues of the Preconditioned Matrices

N	$\mathbf{S}^{-1}\mathbf{L}$		$\mathbf{H}^{-1}\mathbf{L}$	
4	1.000	1.757	1.037	1.781
8	1.000	2.131	1.061	2.877
16	1.000	2.305	1.043	4.241
32	1.000	2.363	1.034	6.392

TABLE VI

Convergence Rates for Euler Iteration in Two Dimensions

N	Single grid spectral radius	Multigrid smoothing rate
4	0.264	0.264
8	0.462	0.330
16	0.605	0.490
32	0.725	0.630

The interpolation for this multigrid scheme can be based upon the Chebyshev polynomial expansions of the solution. Expressions analogous to Eqs. (31)–(33) can be employed, where Eq. (45) is now used for the matrix **C** and the expression for the matrix **E** is altered accordingly. If the boundary conditions are homogeneous, then **C** can easily be manipulated into a selfadjoint form.

Nonstationary Euler iteration will, of course, improve the multigrid smoothing rates. The use of four distinct parameters reduces the smoothing rates of the $N = 16$ and $N = 32$ cases to 0.30 and 0.40, respectively.

Point-Jacobi is a viable alternative here as well. The present form of the matrix (Eq. (49)) also permits the diagonal elements of variable coefficient (or nonlinear) problems to be computed efficiently by transform methods. Two convolution sums now appear in the analog of Eq. (41). The portion involving $\tilde{\mathbf{M}}_{j-l}$ can be evaluated in the usual manner after allowing for special treatment of the terms for which $j = 0$ and $j = N$. The portion involving $\tilde{\mathbf{M}}_{j+l}$ appears in transform space as the product of the transform of $\tilde{\mathbf{M}}_j$ and the complex conjugate of the transform of the variable coefficient term a_l.

Numerical Example

The test problem for the Chebyshev multigrid method has the coefficients

$$a(x, y) = b(x, y) = 1 + \varepsilon(x^2 + y^2) \tag{53}$$

for the exact solution

$$u(x, y) = \sin(\pi \cos x) \sin(\pi \cos y). \tag{54}$$

Some of the results using the finest level $K = 5$ are listed in Table VII. On a single grid, the residual for the $\varepsilon = 0.0$ case is $8.39(-1)$ after 15 relaxations. The exact solution to the discrete equations for this case has an error that is essentially round-off error. There is relatively little content in the high-frequency component. The multigrid approach to this problem makes its biggest gains by using the coarser grids to damp out the low-frequency components.

An example similar to Eq. (54) was examined in [13], where two schemes were given for solving the constant coefficient Chebyshev equations exactly. The results of

TABLE VII

RMS Residuals for Chebyshev Spectral Multigrid
Using Stationary Euler Iteration

Relaxation number	$\varepsilon = 0.0$	$\varepsilon = 0.1$	$\varepsilon = 0.2$
3	1.25 (0)	1.29 (0)	1.32 (0)
6	2.14 (−1)	1.89 (−1)	1.67 (−1)
9	4.68 (−2)	3.81 (−2)	3.16 (−2)
12	1.18 (−2)	9.14 (−3)	7.34 (−3)
15	3.32 (−3)	2.47 (−3)	1.93 (−3)

a recent note [14] suggest that greater accuracy can be achieved, especially on problems with singularities, by subdividing the original domain and patching the individual Chebyshev spectral solutions together along the internal boundaries. The spectral multigrid method can be applied to patched collocation approximations as well. Moreover, the multigrid approach would appear to present a noticeable improvement over the admittedly inefficient schemes used in [14].

Conclusion

The spectral multigrid methods described here exhibited a substantial improvement over the simplest iterative schemes. It has not yet been checked whether this specific algorithm is more efficient than the best available iterative methods. There is still room for improvement, of course, in the spectral multigrid methods. This is especially true for the Chebyshev methods, for which better preconditioning procedures would help considerably.

It is technically straightforward to extend this solution technique to two-dimensional incompressible Navier–Stokes equations, particularly in the vorticity-streamfunction formulation, since the problem addressed in this paper is representative of the advection-diffusion equation. Present efforts are directed towards using the spectral multigrid method to compute the classical problem of flow past a circular cylinder. The appropriate method for this geometry combines a Fourier approximation in angle and a Chebyshev approximation in radius.

APPENDIX: Nomenclature

A diagonal matrix of PDE coefficients at collocation points
a variable coefficient in PDE
B diagonal matrix of PDE coefficients at collocation points
b variable coefficient in PDE
C matrix representing Fourier transform

$\bar{c}_j$ constants used in describing the discrete cosine transform
$\mathbf{D}$ matrix representing first derivative operator in transform space
$\mathbf{E}$ matrix describing trigonometric interpolation in transform space
F right hand side terms of PDE at collocation points
f right-hand side term of PDE
G_k grid on level k
$\mathbf{H}$ preconditioning matrix
K finest level of the multiple grids
k any grid (or level) k of multiple grids
$\mathbf{L}$ matrix representing spectral approximation to PDE operator
$\mathscr{L}$ lower-triangular matrix
$\mathbf{M}$ matrix representing first derivative operator in physical space
$\tilde{\mathbf{M}}$ vector used for describing $\mathbf{M}$
N number of collocation points (in one coordinate direction)
n number of distinct relaxation parameters
$\mathbf{P}$ matrix representing coarse-to-fine grid interpolation
$\mathbf{R}$ matrix representing fine-to-coarse grid interpolation
$\mathbf{S}$ matrix representing finite difference approximation to PDE
T Chebyshev polynomial of degree n
$\mathbf{U}$ vector of solution at collocation points
$\mathscr{U}$ upper-triangular matrix
u solution to PDE
$\hat{u}$ Fourier transform of solution to PDE
$\mathbf{V}$ vector of corrections in multigrid scheme
$\mathbf{v}$ approximate solution to $\mathbf{V}$
x, y physical space coordinates
$\delta_{j,l}$ Keonecker delta function
ε amplitude in variable coefficient term
λ eigenvalue
ξ eigenvector
ρ spectral radius
ω relaxation parameter
μ smoothing rate
$\bar{\mu}$ average smoothing rate

REFERENCES

1. STEVEN A. ORSZAG, *J. Fluid Mech.* **49** (1) (1971), 75.
2. STEVEN A. ORSZAG AND MOSHE ISRAELI, *Ann. Rev. Fluid Mech.* **6** (1974), 281.
3. DAVID GOTTLIEB AND STEVEN A. ORSZAG, *in* "CBMS-NSF Regional Conference Series in Applied Mathematics," Society for Industrial and Applied Mathematics, 1977.
4. R. P. FEDORENKO, *Zh. Vychisl. Mat. Mat. Fix.* **1** (1961), 922.
5. A. BRANDT, *Math. Comput.* **31** (138) (1977), 333.
6. R. A. NICOLAIDES, *J. Comput. Phys.* **19** (4) (1975), 418.
7. W. HACKBUSCH, "Ein Iteratives Verfahren zur Schnellen Auflosung Elliptischer Randwertprobleme," Math. Inst., Universitat zu Koln, Report 76-12, November 1976.
8. STEVEN A. ORSZAG, *Stud. Appl. Math.* **50** (4) (1971), 293.
9. STEVEN A. ORSZAG, *J. Comput. Phys.* **37** (1) (1980), 70.
10. J. W. COOLEY AND J. W. TUKEY, *Math. Comput.* **19** (90) (1965), 297.

11. HARVARD LOMAX, *in* "Proceedings, AIAA 5th Computational Fluid Dynamics Conference, Palo Alto, Ca., June 22–23, 1981," Paper No. 81-0994-CP.
12. Y. S. WONG, "Conjugate Gradient Type Methods for Unsymmetric Matrix Problems," Institute of Applied Mathematics and Statistics, Univ. of British Columbia, Technical Report No. 79-36, 1979.
13. DALE B. HAIDVOGEL AND THOMAS ZANG, *J. Comput. Phys.* **30** (2) (1979), 167.
14. A. MCKERRELL, C. PHILLIPS, AND L. M. DELVES, *J. Comput. Phys.* **41** (2) (1981), 444.