JOURNAL OF COMPUTATIONAL PHYSICS 35, 147–168 (1980)

The Chebychev Method for Solving Nonself-Adjoint Elliptic Equations on a Vector Computer

B. E. McDONALD

U.S. Naval Research Laboratory, Washington, D.C. 20375

Received November 28, 1977

The Chebychev explicit method can be extended to nonsymmetric operators L whose complex eigenvalues lie within an ellipse in the complex plane. The vectorizability of the method results in high execution efficiency on a "pipeline" computer. We derive the method and its convergence rate, and give a comparison with two other methods. The comparison is taken from a 2D plasma turbulence code, in which $L = \nabla^2 + \mathbf{A}(x, y) \cdot \nabla$. The explicit method is approximately three times more efficient than ADI for the model problem solved on a two-pipe Texas Instruments ASC. In some cases, a staggered mesh can be used to gain another factor of 2 in the efficiency of the explicit method. The method has been used successfully on meshes of 34×34, 50×50, and 130×130 points. For grids of 50 or more points on a side, we show in the Appendix that convergence can be speeded considerably by the use of a suitably chosen auxiliary coarse grid, on which long-wavelength components of the error are corrected.

INTRODUCTION

The computational mathematics literature is well stocked with numerical methods for solution of self-adjoint linear operator equations. See, for example, the many methods and references given by Varga [1], Birkhoff [2], and Vichnevetsky [3]. However, many fewer approaches are offered for non-self-adjoint problems. One occasionally encounters the suggestion that the equation $L(\mathbf{r})\,\phi(\mathbf{r}) = S(\mathbf{r})$ (with L the linear operator, and S the known driving term) be made self-adjoint by an extra application of the adjoint of L: $L^\dagger L\phi = L^\dagger S$. There may be cases in which this approach has merit. However, it has two immediate drawbacks: (1) one has to work with a higher-order equation and (2) the ratio of maximum to minimum eigenvalue amplitude, an indicator of the amount of numerical work required, is squared.

An approach that has gained recent attention is a combination of approximate factorization and the conjugate gradient method. See Vichnevetsky [3, p. 60], or Meijerink and van der Vorst [4]. With some adaption, this approach may hold promise for non-self-adjoint problems. But the factorization stage is awkward for vector computers. One must solve upper- and lower-triangular matrix equations, which resemble recursion formulas. In order to obtain a precise inverse, the computer must initiate and complete certain calculations one at a time, rather than doing many in parallel. Some work has been done lately on inverting upper- and lower-triangular

0021-9991/80/050147-22$02.00/0

matrices efficiently on vector processors [5]. However, these methods construct a sequence of iterates by vectorizable algorithms, and thus introduce an extra level of approximation and conceptual complexity into the problem. Thus, we wish to offer an alternative method which allows complete vectorization (parallel processing) for all interior mesh points of a multidimensional grid. In addition to its vectorizability, the Chebychev method guarantees convergence at a rate which is an analytic function of eigenvalue parameters. And unlike the popular alternating direction implicit (ADI) method, its convergence rate holds regardless of whether the operator can be split into one-dimensional operators which commute with each other.

The work to be described here parallels that of Manteuffel [6], who has demonstrated in greater detail some of the properties of the Chebychev polynomials in the complex plane. Whereas Manteuffel offers an iterative method for locating eigenvalues in the complex plane, we are concerned with specifics of the method when the eigenvalues can be estimated in advance. In particular, we give iteration formulas and convergence rates in terms of eigenvalue parameters. We give asymptotic forms for the convergence rate, and give a two-parameter optimization of the convergence rate. We then apply the method to a non-self-adjoint equation encountered in plasma physics and fluid dynamics. For this application we give the eigenvalue parameters essential to the method. We show that these parameters in some cases allow a "hopscotch" or odd–even mesh iteration which would double the efficiency of the method. However, we do not take advantage of this extra factor of two in the results presented here.

Assumptions

We seek an iterative solution to the equation

$$L\phi = S, \tag{1}$$

where S is a known source vector and L is a matrix or finite-difference operator. We assume L has complex eigenvalues $\lambda_r + i\lambda_i$, with all λ_r being of the same sign. We also assume all λ fall within an ellipse in the complex plane (see Fig. 1) whose major or minor axis coincides with the real axis. The intersections of the ellipse with the real axis are $b - a$ and $b + a$, with

$$|b| > a > 0. \tag{2}$$

The limits of the ellipse in the imaginary direction are

$$|\lambda_i|_{\max} = c \tag{3}$$

so the equation of the ellipse is

$$\left(\frac{\lambda_r - b}{a}\right)^2 + \left(\frac{\lambda_i}{c}\right)^2 \leqslant 1. \tag{4}$$

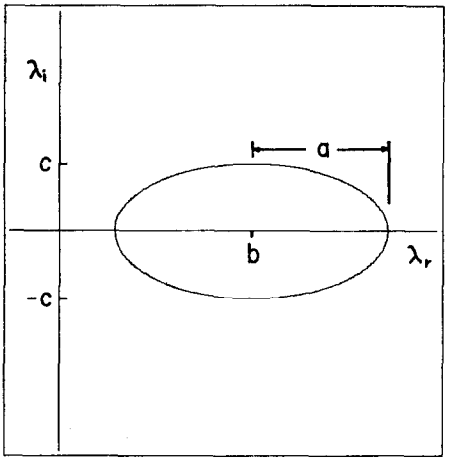

FIG. 1. Ellipse containing complex eigenvalues.

The "best" values of a, b, and c are dependent on the nature of L. For some cases, the error convergence rate may depend strongly on these values. We shall later give a physically motivated example in which "best" values can be estimated straightforwardly. However, if one does not have a priori estimates, values can be obtained numerically by systematic means given elsewhere [6].

We assume that (1) possesses an exact solution Φ. At the end of n iterations, we will have an approximate solution ϕ^n, whose error is defined to be

$$\epsilon^n = \phi^n - \Phi. \tag{5}$$

The iterative method is to be such that

$$\epsilon^n = P_n(L)\, \epsilon^0, \tag{6}$$

where P_n is a polynomial of degree n. Substitution of (5) into (6) gives

$$\phi^n = P_n(L)\, \phi^0 - (P_n(L) - 1)\, \Phi. \tag{7}$$

We do not know Φ in advance, but we do know $L^k\Phi = L^{k-1}S$ for $k > 0$. Thus from (7) we must require that the zero-degree term of $P_n(L)$ be 1:

$$P_n(0) = 1. \tag{8}$$

We wish to choose P_n such that its magnitude is as small as possible everywhere within the ellipse containing eigenvalues of L.

Determination of P_n for a Real Min–Max Problem

The problem of minimizing the maximum value of $|P_n(x)|$ subject to $P_n(0) = 1$ has a well-known solution when x is restricted to real values between $b - a$ and $b + a$ with $b/a > 1$. The standard argument [6] points out that the desired P_n is such that all maxima of $|P_n|$ have the same value. One immediately determines that P_n is proportional to a Chebychev polynomial. The Chebychev polynomials T_n are such that

$$T_n(\cos\alpha) = \cos n\alpha$$

and (9)

$$T_n(\cosh\alpha) = \cosh n\alpha.$$

That $\cos n\alpha$ is a polynomial in $\cos\alpha$ results from the elementary identity

$$\cos(m+1)\alpha = 2\cos\alpha\cos m\alpha - \cos(m-1)\alpha. \tag{10}$$

Equation (10) is also valid when cosh is substituted for cos. Thus if $\cos m\alpha$ and $\cos(m-1)\alpha$ are polynomials in $\cos\alpha$, then so is $\cos(m+1)\alpha$. This is the case for $m = 0$ and 1, so it is also true for all $m > 0$. Equation (10) also gives the recursion formula for the Chebychev polynomials:

$$T_{m+1}(x) = 2xT_m(x) - T_{m-1}(x), \tag{11a}$$

with

$$T_0(x) = 1 \tag{11b}$$

and

$$T_1(x) = x. \tag{11c}$$

Note that $T_n(x)$ is an even or odd function of x, as n is even or odd. From (9) we can see that $|T_n(x)|$ reaches a limiting value of 1, $n+1$ times as x varies from -1 to 1. Thus the solution to the min–max problem for real x is

$$P_n(x) = T_n\left(\frac{x-b}{a}\right)\Big/T_n\left(-\frac{b}{a}\right) \tag{12}$$

and

$$\begin{aligned} |P_n|_{\max} &= \left|T_n\left(-\frac{b}{a}\right)\right|^{-1} \\ &= \frac{1}{\cosh n(\cosh^{-1}(b/a))} < 1. \end{aligned} \tag{13}$$

Determination of P_n for a Complex Min–Max Problem

The Chebychev polynomials also possess optimal properties in the complex plane. With

$$z = \xi + i\eta, \tag{14}$$

we seek a polynomial $P_n(z)$ of degree n such that $P_n(0) = 1$ and such that the maximum $| P_n(z)|$ will be as small as possible within the ellipse

$$\left(\frac{\xi - b}{a}\right)^2 + \frac{\eta^2}{c^2} = 1. \tag{15}$$

For $c/a < 1$, the unique solution to this min–max problem is [6]

$$P_n(z) = T_n\left(\frac{z - b}{(a^2 - c^2)^{1/2}}\right) \Big/ T_n\left(\frac{-b}{(a^2 - c^2)^{1/2}}\right). \tag{16}$$

We shall demonstrate that for a given c/a less than or greater than unity, every maximum of $| P_n |$ in the ellipse (15) has the same value, and that this maximum is less than unity.

For $c/a < 1$, we first point out that all n roots of P_n are pure real. As a result, $| P_n(z)|$ increases monotonically away from the real axis; i.e., there are *no local maxima*. (The proof is elementary.) This means we need examine only the boundary of the ellipse (15) for maxima.

Express the argument of T_n in the numerator of (16) as

$$\begin{aligned} z' &= (z - b)/(a^2 - c^2)^{1/2} \\ &= \cos\alpha \cosh\beta + i \sin\alpha \sinh\beta \\ &= \cos(\alpha - i\beta), \end{aligned} \tag{17}$$

where α and β are real. Proof that an arbitrary complex number can be expressed in this form with real α and β is straightforward and will be omitted. Comparing (14) and (17) and eliminating α one can show that surfaces of constant β are ellipses in the complex plane:

$$\left(\frac{\xi - b}{\cosh\beta(a^2 - c^2)^{1/2}}\right)^2 + \left(\frac{\eta}{\sinh\beta(a^2 - c^2)^{1/2}}\right)^2 = 1. \tag{18}$$

This ellipse is identical to (15) when

$$\tanh\beta = c/a, \tag{19}$$

so that

$$\cosh\beta = (1 - c^2/a^2)^{-1/2}. \tag{20}$$

Note from (17) that z' scans once around the ellipse as α varies from 0 to 2π.

From (17) we find

$$\begin{aligned} T_n(z') &= \cos(n\alpha - \text{in}\,\beta) \\ &= \cos n\alpha \cosh n\beta + i \sin n\alpha \sinh n\beta, \end{aligned} \tag{21}$$

so that

$$|\, T_n(z')|^2 = \cosh^2 n\beta - \sin^2 n\alpha. \tag{22}$$

This shows that on an ellipse of constant β, $|\, T_n(z')|$ reaches the same maximum value, namely, $\cosh n\beta$, $2n + 1$ times as α varies from 0 to 2π. The maximum $|\, P_n(z)|$ can now be obtained from (16):

$$\begin{aligned} |\, P_n(z)|_{\max} &= \frac{\cosh n \cosh^{-1}(1 - c^2/a^2)^{-1/2}}{\cosh n \cosh^{-1}(|\, b\,|/a)(1 - c^2/a^2)^{-1/2}} \\ &< 1. \end{aligned} \tag{23}$$

We have used the even or odd symmetry of T_n to remove the minus sign from the argument in the denominator of (16). The inequality in (23) results from $|\, b\,|/a > 1$ and the monotonicity of the functions cosh and $\cosh^{-1}$

For $c/a > 1$, (16) becomes

$$P_n(z) = T_n\left(\frac{z - b}{i(c^2 - a^2)^{1/2}}\right) \Big/ T_n\left(\frac{-b}{i(c^2 - a^2)^{1/2}}\right). \tag{24}$$

All n roots of P_n are now on a line parallel to the imaginary axis, and there are still no local maxima in $|\, P_n(z)|$. Again, we need examine only the perimeter of the ellipse for maxima. Express the argument in the numerator of (24) as

$$\begin{aligned} z' &= -i(z - b)/(c^2 - a^2)^{1/2} \\ &= \cos\alpha \cosh\beta + i \sin\alpha \sinh\beta \\ &= \cos(\alpha - i\beta) \end{aligned} \tag{25}$$

Comparing (14) and (25) and eliminating α we find

$$\left(\frac{\eta}{\cosh\beta(c^2 - a^2)^{1/2}}\right)^2 + \left(\frac{\xi - b}{\sinh\beta(c^2 - a^2)^{1/2}}\right)^2 = 1. \tag{26}$$

This is identical to (15) when

$$\tanh\beta = a/c \tag{27}$$

or

$$\cosh\beta = (1 - a^2/c^2)^{-1/2} \tag{28}$$

Using (25), we find

$$| T_n(z')|^2 = \cosh^2 n\beta - \sin^2 n\alpha. \tag{29}$$

This gives cosh $n\beta$ as the maximum amplitude of the numerator in (24). We can evaluate the denominator in (24) by setting $z = 0$, $\alpha = \pi/2$, and $\sinh \beta = b/(c^2 - a^2)^{1/2}$ in (25). For even n, (29) gives the amplitude of the denominator as $\cosh n \sinh^{-1} b/(c^2 - a^2)^{1/2}$, while for odd n the result is $\sinh n \sinh^{-1} b/(c^2 - a^2)^{1/2}$. For this reason *we must restrict n to even values* to guarantee convergence when $c/a > 1$. For even n, (24) gives

$$\begin{aligned} | P_n(z)|_{\max} &= \frac{\cosh n \cosh^{-1} (1 - a^2/c^2)^{-1/2}}{\cosh n \sinh^{-1} b/(c^2 - a^2)^{1/2}} \\ &= \frac{\cosh n \cosh^{-1} (1 - a^2/c^2)^{-1/2}}{\cosh n \cosh^{-1} (1 + (b^2 - a^2)/c^2)^{1/2} (1 - a^2/c^2)^{-1/2}} \\ &< 1. \end{aligned} \tag{30}$$

To illustrate the necessity of taking n even, let us consider an example in which $b/c = 0.9$ and $a/c = 0.899$. For $n = 16$, (30) gives $| P |_{\max} = 0.984$. However, with $n = 17$, we must change the cosh function in the denominator of (30) to sinh, and we have $| P_n |_{\max} = 1.003$.

Construction of the Iterative Method

We shall take the polynomial of (16) to be the proper choice for use in constructing the approximate solution to (1) according to (7). We must choose between two approaches for generating the polynomial $P_n(L)$. The first is factorization of P_n into n linear factors. We have tried this approach and found it subject to roundoff error amplification. The reason for this is that the process is equivalent to n overrelaxation steps, some of which reduce the long-wavelength errors at the expense of increasing the short-wavelength errors. The short-wavelength errors are eventually brought back down, but any roundoff error introduced while they were large tends to contaminate the final result.

The second and more preferable approach to constructing $P_n(L)$ is through the use of recursion formulae. The only real drawback to this method is that one extra array of storage is required for retention of earlier iterates. We substitute (16) into (7) to obtain

$$\phi^{n+1} = \frac{T_{n+1}((L - b)/(a^2 - c^2)^{1/2})}{T_{n+1}(-b/(a^2 - c^2)^{1/2})} (\phi^0 - \Phi) + \Phi. \tag{31}$$

In the numerator of this expression let us express T_{n+1} in terms of T_n and

T_{n-1} according to the recursion formula (11), and then express T_n and T_{n-1} in terms of ϕ^n and ϕ^{n-1} according to (31). Then we find

$$\phi^{n+1} = \frac{2T_n(q)}{T_{n+1}(q)\,(a^2 - c^2)^{1/2}}((L - b)\,\phi^n - S) - \frac{T_{n-1}(q)}{T_{n+1}(q)}\,\phi^{n-1}, \tag{32}$$

where

$$q = -b/(a^2 - c^2)^{1/2}. \tag{33}$$

Notice that the coefficient of Φ is zero. Recall that we imposed this condition at the outset in (8).

In order to use this method we must know the first two terms in the recursion. We get these by direct construction of $P_n(L)$, and obtain

$$\begin{aligned} \phi^0 &= \text{arbitrary trail solution,} \\ \phi^1 &= \phi^0 - 1/b(L\phi^0 - S). \end{aligned} \tag{34}$$

Restriction to Real Arithmetic

In case $c > a$, then q in (33) becomes imaginary. We can avoid complex arithmetic in calculating T_n in (29) by use of the Chebychev polynomials of imaginary argument:

$$T_n(ix) \equiv i^n \tau_n(x). \tag{35}$$

Substitution of (35) into (11) gives the recursion formula

$$\tau_{m+1}(x) = 2x\tau_m(x) + \tau_{m-1}(x), \tag{36a}$$

where

$$\tau_0(x) = 1 \tag{36b}$$

and

$$\tau_1(x) = x. \tag{36c}$$

When $c > a$, replace (32) with

$$\phi^{n+1} = \frac{2\tau_n(q')}{\tau_{n+1}(q')\,(c^2 - a^2)^{1/2}}\,((L - b)\,\phi^n - S) + \frac{\tau_{n-1}(q')}{\tau_{n+1}(q')}\,\phi^{n-1}, \tag{37}$$

where $q' = b/(c^2 - a^2)^{1/2}$. What happens in case $c = a$? In the limit $c \to a$, q and q' tend to infinity and both (32) and (37) give

$$\phi^{n+1} = \phi^n - \frac{1}{b}\,(L\phi^n - S). \tag{38}$$

Error Estimates

If B is an eigenvector of L such that

$$LB = \lambda B, \tag{39}$$

where λ is a complex constant, then

$$P_n(L)\, B = P_n(\lambda)\, B. \tag{40}$$

Let us imagine that the error ϵ^0 of (5) is expressed as a linear combination of the eigenvectors of L. We have assumed all eigenvalues of L to lie within an ellipse (15) in the complex plane. Then from (6) and (23) or (30), the coefficients of the eigenvector expansion of ϵ^n have each been decreased in amplitude by at least a factor

$$|\, P_n(\lambda)| = E_n(a, b, c) \equiv |\, T_n(a/(a^2 - c^2)^{1/2})/T_n(b/(a^2 - c^2)^{1/2})| \,. \tag{41}$$

In Fig. 2 we plot this error limit as a function of n for various values of the ratio of maximum to minimum real eigenvalue,

$$R \equiv \frac{|\, b\,| + a}{|\, b\,| - a}, \tag{42}$$

and ellipse axis ratio c/a. One notices that the error limit decreases approximately exponentially with n. In practice, when R is greater than about 10^3, we use a coarse mesh–fine mesh combination [7] to bring down the residual error. The slow convergence at large R is due to the fact that one is attempting to reduce the error by a diffusion process, which is slow for long wavelengths.

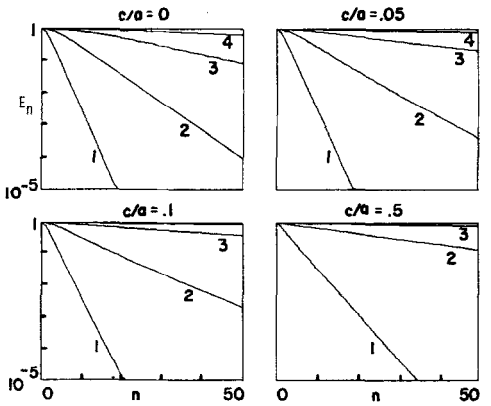

FIG. 2. Error estimate E_n versus n as given by (41) for various ellipse parameters (a, b, c). The curves are labeled with $\log_{10}R$, as defined in (42).

Let us derive an approximate expression for (41) in the limit of large R. The results to be given here are valid for arbitrary $c/a \neq 1$. It is convenient to define

$$\begin{aligned} \delta &= |b|/a - 1 \\ &= 2/(R-1), \end{aligned} \tag{43}$$

and

$$\rho = c/a.$$

From (9) and the identity,

$$\cosh^{-1} x = \log(x + (x^2 - 1)^{1/2}), \tag{44}$$

we have from (23) or (30), as appropriate,

$$\begin{aligned} E_n &= \frac{\cosh(n \log Q_1)}{\cosh(n \log Q_1 + n \log Q_2)} \\ &= \frac{1}{\cosh(n \log Q_2) + \tanh(n \log Q_1) \sinh(n \log Q_2)} \end{aligned} \tag{45}$$

where

$$\begin{aligned} Q_1 &= \frac{1 - \rho}{|1 - \rho^2|^{1/2}}, \\ Q_2 &= \frac{1 + \delta + (2\delta + \delta^2 + \rho^2)^{1/2}}{1 + \rho}. \end{aligned} \tag{46}$$

Since Q_1 and Q_2 are both greater than 1, (45) yields

$$2 \exp(-n \log Q_2) > E_n > \exp(-n \log Q_2). \tag{47}$$

Thus the estimate

$$E_n \approx 2^{1/2} \left(\frac{1 + \delta + (2\delta + \delta^2 + \rho^2)^{1/2}}{1 + \rho} \right)^{-n} \tag{48}$$

is always correct within a factor of $2^{1/2}$. Defining the convergence rate to be

$$C = -\frac{\partial \log E_n}{\partial n}, \tag{49}$$

we can find simple expressions for C in the limit $\delta \ll 1$. For $\rho \ll \delta$, $C \approx (2\delta)^{1/2}$. When $\rho \gg \delta$, then $C \approx \delta/\rho$. This shows that convergence slows down when the eigenvalues of the operator L have significant imaginary parts.

APPLICATION TO A MODEL PROBLEM

As a test of the method (32) and the error estimate (41) we shall solve numerically the two-dimensional diffusion-advection equation

$$\nabla^2\phi + \mathbf{A}(x, y) \cdot \nabla\phi = S(x, y) \tag{50}$$

on a rectangular grid with constant mesh spacing subject to doubly periodic boundary conditions. The analysis to follow will be such that other boundary conditions may be treated by changing the value of a single parameter σ (to be defined in (72)). Equations of this form arise in plasma physics when charge neutrality is imposed upon a plasma with a nonisotropic electrical conductivity, with ϕ being the electrostatic potential. For an application to the earth's ionosphere, see McDonald *et al.* [8, 9].

The mesh is assumed to be $K_x \times K_y$ interior mesh points. We choose the x direction so that

$$K_x \geqslant K_y . \tag{51}$$

We also assume the mesh intervals to be

$$\delta x = \delta y = \text{const.} \tag{52}$$

We use redundant guard cells around the perimeter of the mesh for efficient handling of the boundary conditions. This results in a complete mesh of $(K_x + 2) \times (K_y + 2)$ points. We use a second-order five-point representation of the derivatives in (50):

$$\begin{aligned} L\phi_{I,J} \equiv\ & (\phi_{I+1,J} + \phi_{I-1,J} + \phi_{I,J+1} + \phi_{I,J-1} - 4\phi_{I,J})/\delta x^2 \\ & + AX_{I,J}(\phi_{I+1,J} - \phi_{I-1,J})/2\,\delta x \\ & + AY_{I,J}(\phi_{I,J+1} - \phi_{I,J-1})/2\,\delta x. \end{aligned} \tag{53}$$

Here I and J are the x and y mesh point indices, and AX and AY are the x and y components of $\mathbf{A}$, respectively.

In order to estimate the eigenvalues of L it is necessary to consider $\mathbf{A}$ locally constant. Then the eigenfunctions are complex exponentials,

$$\phi_{\mathbf{k}} = \exp\left(2\pi i\left(\frac{k_x I}{K_x} + \frac{k_y J}{K_y}\right)\right), \tag{54}$$

where

$$k_x = 0, 1, 2, \ldots, K_x - 1, \text{ etc. for } k_y . \tag{55}$$

The eigenvalues are

$$\begin{aligned} \lambda_{\mathbf{k}} = & \frac{2}{\delta x^2}\left(\cos\frac{2\pi k_x}{K_x} + \cos\frac{2\pi k_y}{K_y} - 2\right) \\ & + i\left(\frac{AX}{\delta x}\sin\frac{2\pi k_x}{K_x} + \frac{AY}{\delta x}\sin\frac{2\pi k_y}{K_y}\right). \end{aligned} \tag{56}$$

Construction of the Ellipse

Any set of eigenvalues of bounded modulus can be enclosed in a sufficiently large ellipse. However, for the Chebychev iteration to have a positive convergence rate, the ellipse must exclude the origin. In the analysis to follow we are motivated by the need to produce an algorithm for solving the variable coefficient equation (50) at each time step in a plasma physics code [8, 9]. The coefficients change in time, so it does not seem appropriate to spend a great deal of effort in calculating precise iteration parameters for a particular $\mathbf{A}(x, y)$. Rather we will assume that an appropriate global measure of $|\mathbf{A}|$ is available, and from this construct parameters optimized for the most unfavorable orientation of $\mathbf{A}$. The resulting convergence rate is positive, but its proximity to the maximum convergence rate for specific distributions $\mathbf{A}(x, y)$ has not been investigated in a systematic way.

In order to find the tightest ellipse containing the complex eigenvalues for arbitrary orientation of $\mathbf{A}$, let us adopt the following abbreviated notation:

$$
\begin{aligned}
c_x &= \cos\frac{2\pi k_x}{K_x}, \\
s_x &= \sin\frac{2\pi k_x}{K_x}, \\
a_x &= AX\,\delta x, \\
a_y &= AY\,\delta y, \\
\xi' &= c_x + c_y, \\
\eta' &= a_x s_x + a_y s_y .
\end{aligned} \tag{57}
$$

From (56), $\lambda_k = 2/\delta x^2(\xi' - 2) + i/\delta x^2\eta'$. To find the envelope of these eigenvalues we need only maximize $|\eta'|$ for fixed ξ'. The orientation of (a_x, a_y) must be allowed to be arbitrary (we are solving a variable coefficient equation and must allow for the "worst case"). Maximizing on the orientation of (a_x, a_y) and on c_x, we find the envelope

$$
\lambda_e = 2/\delta x^2(\xi' - 2) \pm i/\delta x^2\bar{a}(2 - \tfrac{1}{2}\xi'^2)^{1/2}, \tag{58}
$$

where

$$
\bar{a} = (a_x^2 + a_y^2)^{1/2}. \tag{59}
$$

This is an ellipse with parameters $(a, b, c) = (4, -4, \bar{a}2^{1/2})/\delta x^2$. It is the tightest ellipse containing *all* the eigenvalues for arbitrary K_x, K_y, a_x, and a_y subject to fixed $\bar{a}$. Unfortunately, it passes through the origin and thus results in a zero convergence rate. By modifying the ellipse as follows, we can obtain a positive convergence rate.

MAXIMIZING THE CONVERGENCE RATE

We must exclude from the ellipse the "mean value" component $(k_x, k_y) = (0, 0)$, which has no effect upon (50). Let us then construct a family of ellipses passing through

$$\lambda_{0r} \pm i\lambda_{0i} \equiv \text{eigenvalue pair with real part nearest zero.} \tag{60}$$

Maximizing the convergence rate with respect to the two remaining free-ellipse parameters amounts to maximizing Q_2 in (46). A significant simplification results if we approximate (46) as follows for $\delta \ll 1$:

$$Q_2 \approx 1 + (2\delta + \rho^2)^{1/2} - \rho. \tag{61}$$

Equation (61) is valid to lowest order if δ for all $\rho \geqslant 0$. The family of ellipses passing through (60) has

$$a = ((\lambda_{0r} - b)^2 + (\lambda_{0i}/\rho)^2)^{1/2} \tag{62}$$

and

$$\delta = ((1 - \lambda_{0r}/b)^2 + (\lambda_{0i}/\rho b)^2)^{-1/2} - 1 \tag{63}$$

so that

$$\delta \approx \lambda_{0r}/b - \tfrac{1}{2}(\lambda_{0i}/\rho b)^2. \tag{64}$$

The assumption $\delta \ll 1$ requires both terms on the right-hand side of (64) to be small. Maximizing (61) with respect to ρ then gives

$$\theta^3 + \mu^2(3\theta - \tfrac{1}{2}) = 0, \tag{65}$$

where

$$\theta = \frac{\lambda_{0i}^2}{4\rho^2 b\lambda_{0r}} \tag{66}$$

and

$$\mu = \tfrac{1}{4} \mid \lambda_{0i}/\lambda_{0r} \mid . \tag{67}$$

Equation (65) has only one real root, since its θ — derivative is never zero for real θ. This root is

$$\theta = (\mu/4)^{1/3}\{[(16\mu^2 + 1)^{1/2} + 1]^{1/3} - [(16\mu^2 + 1)^{1/2} - 1]^{1/3}\}. \tag{68}$$

As μ increases from zero, θ increases monotonically from zero to its limiting value of $\frac{1}{6}$. This is shown in Fig. 3.

Maximizing (61) with respect to b is equivalent to maximizing (63). The result, invoking (66), is

$$b_{\max} = \lambda_{0r}/(1 - 4\theta) \lesssim 3\lambda_{0r}\,. \tag{69}$$

However this value is unacceptably small since many eigenvalues of high mode numbers would fall outside the ellipse. Thus we take the smallest allowable value for b:

$$b = -4/\delta x^2. \tag{70}$$

This choice excludes from the ellipse the "odd–even" mode $(k_x, k_y) = (K_x/2, K_y/2)$. This choice allows an effective doubling of the computational efficiency (to be demonstrated below) at the expense of having to perform a single follow-up iteration to eliminate the "odd–even" mode:

$$\phi \to \phi + \tfrac{1}{2}\delta x^2(L\phi - S). \tag{71}$$

From (53) and (70), the midpoint of the ellipse, b, is just equal to the center coefficient of the finite-difference operator L in (53). The recursion formulas for the iterative solution, (32) or (37) as appropriate, involve the operator $L - b$, whose central coefficient is zero. This results in a natural separation of the finite-difference solution into odd and even components, separated according to whether the sum of the indices $I + J$ is odd or even. In fact this odd–even separation extends to an arbitrary number of spatial dimensions. This separation would allow one to define ϕ^n for even (odd) n only on even (odd) gridpoints. One could then refine the solution in a "hopscotch" fashion, updating only half the points at a given iteration.

We have not taken advantage of this even–odd grid separation in any of the results presented here. An interested user should keep in mind two caveats: if the eigenvalue ellipse is prolate ($c/a > 1$), convergence is not guaranteed for odd-numbered iterates (see (30)); and to overcome the inherent fetch-speed limitation of a pipeline computer,

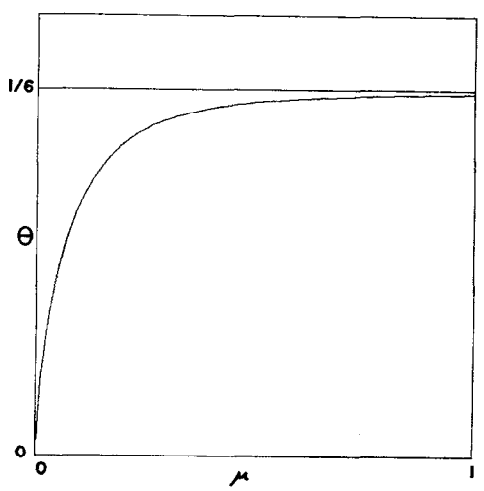

FIG. 3. Optimal tuning parameter as determined from (68).

one must separate even and odd grid point quantities into contiguously stored arrays.

It remains to give expressions for λ_{0r} and λ_{0i} for the model problem. Defining

$$\sigma = (s_x^2 + s_y^2)_{\min}^{1/2} \approx 2\pi/K_x \qquad \text{(for periodic boundaries)}, \tag{72}$$

(58) gives, for $\sigma \ll 1$,

$$\lambda_{0r} \approx -\sigma^2/\delta x^2, \qquad \lambda_{0i} = \bar{a}\sigma/\delta x^2. \tag{73}$$

In (72), "min" refers to a minimum over all wavenumbers excluding the (0, 0) mode. Ellipse parameters a and b are now specified by (62) and (70). We obtain c from (66) and (43):

$$c = a\lambda_{0i}/(4\theta b\lambda_{0r})^{1/2}. \tag{74}$$

Convergence Rates

Taking into consideration (46), (48), (49), and (69), we have the approximate convergence rate

$$C \approx (2\delta + \rho^2)^{1/2} - \rho. \tag{75}$$

We can use (67), (68), and (73) to express (75) in terms of μ. The result is

$$C \approx 2^{-1/2}\sigma C_0(\mu), \tag{76}$$

where

$$C_0(\mu) = 2^{1/2}[(\tfrac{1}{2} - \theta + \mu^2/\theta)^{1/2} - \mu\theta^{1/2}]. \tag{77}$$

Recall that $\sigma \approx 2\pi/K_x$ for doubly periodic boundaries. The dependence of C_0 upon μ is shown in Fig. 4. Also plotted in Fig. 4 (dashed line) is the approximation

$$C_0(\mu) \approx (9.5\mu + 1)^{-1}. \tag{78}$$

Using (71), we find the following limiting forms for C:

$$\begin{aligned} C &\approx \sigma 2^{-1/2}(1 - 3(\mu^2/2)^{1/3}), && \mu \ll 1 \\ C &\approx \sigma\mu^{-1}6^{-3/2}, && \mu \gtrsim 1. \end{aligned} \tag{79}$$

This shows that as μ increases from 0 to 1, the convergence rate drops by approximately a factor of 10.

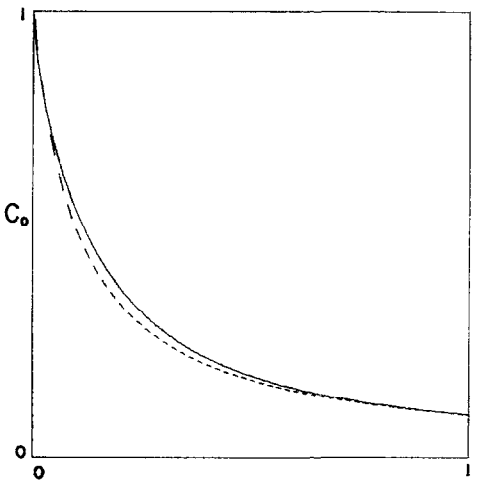

FIG. 4. Normalized convergence rate (77) (solid); approximation (78) (dashed).

REMOVAL OF LONG-WAVELENGTH ERROR ON COARSE GRID

We can see from (72) and (76) that the number of iterations required to reach a certain level of error reduction is proportional to K_x. Thus the total number of operations required is proportional to $K_x^2 K_y$. Therefore, it is advantageous to correct the long-wavelength components of the solution on a coarse mesh and interpolate onto a fine mesh before completing the solution. The importance of regridding as a means of accelerating convergence of explicit relaxation schemes has been recognized for some time (see Brandt [7] and references contained therein). The interpolation between coarse and fine meshes introduces truncation error into the solution, so that the error estimates derived earlier may not hold on the fine mesh. For this reason, it is best to alternate between fine and coarse meshes more than once, attempting only a modest error reduction with each pass. The procedure to be described below was developed through practical experience in simulating physical systems. Some of the details of this procedure can be justified by a simple model given in the Appendix. In practice, use of a coarse grid becomes important for $K_x \geqslant 50$. For a grid of 128×128 interior points, a coarse grid of 32×32 interior points is used. First, the residual $L\phi^0 - S$ is extracted on the fine grid. Then this residual and **A** are defined on the coarse grid using block averages of 16 fine grid points per coarse grid point. A course grid correction is initialized to zero and a large number of iterations are performed. Typically we use three times as many iterations on the coarse grid as on the fine grid. Then the potential correction is defined on the fine grid using bilinear interpolation, and the result is subtracted from ϕ^0. Then iterations are performed on the fine grid, making no attempt to make significant improvements in the lowest 5 to 10 modes of the solution. That is, we arbitrarily increase σ in (72) by a factor of 5 to 10. This

results in improved convergence of the higher modes. The effectiveness of this fine grid-coarse grid approach may be improved significantly in a time dependent problem by two-level extrapolation for the trial solution ϕ^0.

Numerical Results

As a test of the iterative procedure (32) and the convergence estimate (76) we have solved (50) on meshes of 32×32 and 48×48 interior grid points without regriding; and on a 128×128 mesh with a reduced mesh of 32×32 interior points. All tests used doubly periodic boundary conditions. The numerical convergence rate comparisons were carried out as follows. Arbitrary forms were adopted for $\mathbf{A}$ and a reference solution Φ. Then a source term was generated from Φ numerically using the difference operator (53). This source term was used in the iteration (32), with the approximate solution being initialized to zero. After a large number N of iterations (usually $N = 40$), a relative error E was defined to be the root mean square residual of (50) divided by the root mean square of the source, S. The average convergence rate was then taken to be $-\log E/N$.

In all cases convergence was fast enough to be consistent with (76), and in most cases was faster than (76). The one case in which convergence was just equal to (76) for all μ was for $\Phi_{IJ} = \sin(2\pi I)/K_x$, and $A_x = \text{const}$, $A_y = 0$.

Figure 5 shows convergence rates per second of computer time, $-\log E/t$, obtained

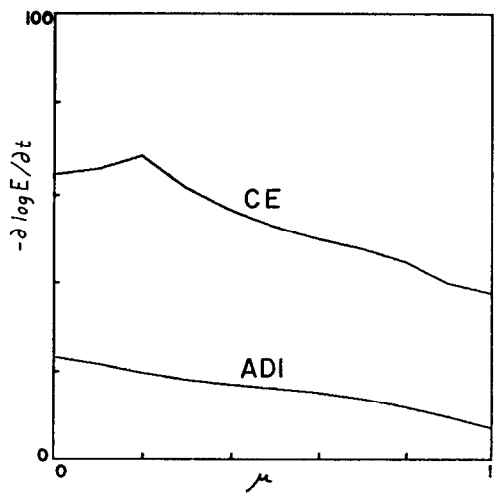

FIG. 5. Convergence rates per second of computer time, $-\partial \log E/\partial t$, for Chebychev explicit (CE) and alternating direction implicit (ADI) methods on a two-pipe ASC.

using a two-pipe Texas Instruments *ASC* to solve the same set of problems with the Chebychev explicit method (CE) (upper curve) and with an alternating direction implicit method (ADI) (lower curve). These comparisons were made on a 50×50 mesh using data from the EJET plasma turbulence code [8, 9]. The vector **A** was computed from a turbulent plasma density distribution $n_e(x, y)$ having a spectral power index of approximately -3.5:

$$A_x = f\frac{\partial n_e}{\partial y}, \qquad A_y = -f\frac{\partial n_e}{\partial x}. \tag{80}$$

The proportionality constant f was adjusted to give a desired value of μ (see (67)) from 0 to 1. Figure 5 summarizes 11 separate tests with **A** generated from the same n_e, but scaled to give 11 equally spaced μ values. For these tests Φ was the electrostatic potential appropriate to n_e [8, 9].

The ADI solution used logarithmically spaced iteration parameters [1] and a partially vectorized tridiagonal solver. Although ADI converged faster than CE *per iteration*, the limited vectorizability of the method resulted in slow execution. Thus the result in Fig. 5 is machine dependent. The execution times per iteration were 25.8 ms for ADI and 1.32 ms for the Chebychev explicit method. To illustrate the computational efficiency of the explicit method, the lower limit on execution time per iteration (neglecting overhead and boundary value resetting) would be (48 × 48 [illegible]) the explicit method runs at 76 % efficiency on a modest 50 × 50 mesh. ADI could be made more competitive by the use of cyclic reduction rather than tridiagonal solution in the integration direction, but it is unlikely that any improvement would bring the convergence rate per second up to that of the explicit method. In addition the boundary conditions are much easier to change in the explicit method than in the implicit one. Recall, too, that when the eigenvalue ellipse is oblate, one can use "hopscotch" updating and achieve in principle a doubling of the explicit method's efficiency.

Another comparison was made using a recently proposed method employing the conjugate gradient (CG) algorithm [10]. This particular CG method requires inversion of the self-adjoint part of L (i.e., $(L + L^T)/2$) once per iteration. For this problem an optimized complex fast Fourier transform (FFT) is used to invert the Laplacian operator. The results of the comparison are not shown in Fig. 5, but are comparable to the ADI results for $\mu \geqslant 0.1$. For $\mu = 0$ the method is direct and thus superior to the others. However for $\mu \geqslant 10^{-4}$ the convergence of CG is no greater than that of CE. Use of a fully optimized real transform would increase the overall speed of CG by approximately a factor of 2, leading to results intermediate to the two curves of Fig. 5. However this particular CG method is not as generally applicable as ADI or CE in that it may become unwieldy when the self-adjoint portion of L contains varying coefficients.

APPENDIX: A MODEL FOR OPTIMAL REGRIDDING

We shall derive properties of the "best" coarse mesh for use in refining long wavelength components of the solution. Let us consider a test problem requiring the solution of Poisson's equation subject to doubly periodic boundaries on a square domain of N by N interior points:

$$\nabla^2\phi = S(x, y). \tag{A1}$$

The model to be developed can be made more elaborate to encompass more general problems such as (50), and to include startup and fetch increment penalties inherent in vector processors. However, the results from the more elaborate model are qualitatively the same as those from the simple model below.

Given an initial trial solution, our goal is to reduce the residual error by a factor of ϵ in two stages. First, we enlarge the grid spacing by a factor m in each direction, so that the coarse mesh consists of $N/m \times N/m$ points. On this mesh we reduce the residual by a factor $\epsilon/2$. Second we interpolate corrections to the solution from the coarse mesh back onto the fine mesh The interpolation process introduces an error of magnitude $\epsilon/2$ or less into some number of n of the lowest modes of the system Thus the errors in modes 1 through n have been reduced by at least ϵ. Errors in modes higher than mode n have been decreased somewhat, but not necessarily by a factor ϵ. So next we relax the errors in modes higher than n on the fine mesh. This is done by increasing σ in (72) by a factor n.

The regridding is now defined by three variables ϵ, m, and n. These are respectively the overall error reduction factor to be achieved, the ratio of coarse to fine grid spacing, and the number of long-wavelength modes for which truncation error introduced by interpolation is small enough that further error relaxation on the fine mesh is unnecessary. An estimate for the truncation error in a particular mode will allow n to be defined in terms of ϵ and m. These two remaining independent variables will then be adjusted to give maximum overall convergence rate.

Let D be the following second-order representation of the second derivative:

$$Df(x_l) = (f_{l+1} - 2f_l + f_{l-1})/\delta x^2. \tag{A2}$$

Suppose δx is increased by a factor m. Then for the basic function e^{ikx} we have

$$\begin{aligned} D_m e^{ikx} &= 2\,(\cos mk\,\delta x - 1)/m^2\,\delta x^2\, e^{ikx} \\ &= (-k^2 + \tfrac{1}{12}k^4 m^2\,\delta x^2 - \cdots)\, e^{ikx}, \end{aligned} \tag{A3}$$

where the subscript m denotes the increase in the mesh interval and k an arbitrary wavenumber. To lowest order in $k\delta x$, the fractional truncation error in (A3) is of magnitude $\frac{1}{12}(mk\delta x)^2$. To find the mode for which the truncation error is $\epsilon/2$, we set $k = n\,2\pi/N\delta x$ and find

$$n = \frac{N}{2\pi m}(6\epsilon)^{1/2}. \tag{A4}$$

We now compute the residual error in (A1),

$$R = \nabla^2\phi - S, \tag{A5}$$

and define block averages of R on a coarse mesh of $N/m \times N/m$ interior points.

The asymptotic convergence rate for the Chebychev relaxation on the coarse mesh is $2^{1/2}\pi m/N$, so the total number of operations required to reduce the residual by $\epsilon/2$ is approximately

$$T_c = \left[-\log(\epsilon/2)\Big/\left(\frac{2^{1/2}\pi m}{N}\right)\right] 11(N/m)^2. \tag{A6}$$

After performing the required number of iterations on the coarse mesh, the correction is transferred to the fine mesh by bilinear interpolation and subtracted from the fine mesh solution. The additional truncation error introduced by bilinear interpolation consists mainly of high wavenumber modes. These are damped very effectively by the follow-up iterations on the fine mesh. Increasing σ in (72) by a factor n gives a convergence rate $2^{1/2}\pi n/N$, so that the required fine mesh operation count is

$$T_f = [-\log \epsilon/(2^{1/2}\pi n/N)]\, 11N^2. \tag{A7}$$

The total operation count is the sum of T_c and T_f and all operations necessary for regridding. The regridding can be optimized for the simple uniform mesh problem under consideration to require one operation per fine mesh point in obtaining block sums, and eight for the bilinear interpolation and correction of the solution. Thus the total operation count is

$$T = 9N^2 + 11N^2\left(-\frac{m\log\epsilon}{(6\epsilon)^{1/2}} - \frac{\log\epsilon/2}{2\pi}\frac{N}{m^3}\right) 2^{1/2}. \tag{A8}$$

We have eliminated n from (A8) by using (A4). In deriving (A8) we have set the operation count per point to 11 for the Chebychev iteration. One could take advantage of the constancy of coefficients in the Poisson problem to reduce the operation count to 7. However, our purpose here is to illustrate properties of regridding for a more general class of problems. The number 11 is taken as a representative value for the Chebychev iteration applied to five-point operators. This number can take on values from 7 (for all constant coefficients) to 13 (for all variable coefficients).

We now define the convergence rate per operation:

$$C_{op} = -\log \epsilon/T. \tag{A9}$$

Let us maximize (A9) with respect to m and ϵ simultaneously. We find

$$m^4 = \frac{3N}{2\pi}(6\epsilon)^{1/2}\left(1 - \frac{\log 2}{\log \epsilon}\right) \tag{A10}$$

and

$$0 = \left(\frac{9}{2^{1/2}} + \frac{11N}{2\pi m^3}\log 2\right)\Big/(\log \epsilon)^2 - 11m/(24\epsilon)^{1/2}. \tag{A11}$$

Equations (A10), (A6), and (A7) show that for the optimal regrid $T_f = 3T_c$. Allowing for regridding operations, the optimal solution thus spends less than one-fourth of the time carrying out coarse grid iterations. Results from the numerical solution of (A10) and (A11) are given in Table I. These results are only approximate, since we have used approximate expressions for convergence rates and truncation errors. Table I shows that the optimal coarse grid has between two and four times fewer points in each direction than the original mesh. For large meshes ($N \geqslant 50$) one can take $m = 4$ and achieve near-optimal results. For $N < 50$, $m = 2$ is near optimal.

The optimal error reduction factor is never greatly different from $\frac{1}{3}$, so that one may have to repeat the coarse mesh–fine mesh procedure many times to achieve a substantial error reduction. Table I also shows that for large meshes, the use of a coarse grid for correction of long-wavelength errors may increase converge by a factor of 4 to 7. This may be improved by the use of a hierarchy of coarse grids. Brandt [7] claims convergence rates in excess of those in Table I for a multigrid solution of the Poisson equation. However, it has not yet been demonstrated how well the multigrid method treats variable coefficient equations. The constant coefficient model presented here and its results are descriptive of the balance between convergence and generation of truncation error for a general two-grid system. The results of the present model are similar to those of a more elaborate model tailored to Eq. (50).

TABLE I

Optimal Regrid Parameters m, ϵ, n for Solution of Poisson's Equation on $N \times N$ Mesh[a]

N	m	ϵ	n	C_m[b]	C_2/C_m[c]	C_4/C_m[c]	C_1/C_m[d]
16	2.073	0.352	1.785	1.02E−04	1.00	0.73	0.96
25	2.339	0.367	2.523	3.84E−05	0.97	0.79	0.67
32	2.500	0.374	3.054	2.23E−05	0.93	0.83	0.55
50	2.819	0.389	4.310	8.36E−06	0.84	0.89	0.39
100	3.397	0.410	7.350	1.82E−06	0.65	0.97	0.22
200	4.091	0.431	12.513	3.94E−07	0.44	1.00	0.13

[a] N, m, ϵ, and n are respectively the grid size, grid reduction factor, error reduction factor, and factor by which σ in (72) is to be increased on the $N \times N$ grid.

[b] C_m is the maximum convergence rate per operation from (A9)–(A11).

[c] C_2 and C_4 are obtained from (A8) and (A9) with $m = 2$ and 4, respectively.

[d] C_1 is the convergence rate per operation without regridding: $C_1 = 2^{1/2}\pi/11N^3$.

ACKNOWLEDGMENT

This work was supported by the Office of Naval Research and the Defense Nuclear Agency. Preliminary results were presented at the Second IMACS Symposium on Computer Methods for Partial Differential Equations, June 22–24, 1977, Lehigh University.

REFERENCES

1. R. S. VARGA, "Matrix Iterative Analysis," Prentice-Hall, Englewood Cliffs, N. J., 1962.
2. G. BIRKHOFF, The numerical solution of elliptic equations, presented at Regional Conference Series in Applied Mathematics, SIAM, Philadelphia, 1971.
3. R. VICHNEVETSKY, "Advances in Computer Methods for Partial Differential Equations" (R. Vichnevetsky, Ed.), AICA, Rutgers University, New Brunswick, N. J., 1975.
4. J. A. MEIJERINK AND H. A. VAN DER VORST, An Iterative Solution Method for Linear Systems of which the Coefficient Matrix is a Symmetric M-Matrix, Academic Computer Center, Budapestlaan 6, de Uithof-Utrecht, Netherlands Technical Report TR-1, 1976.
5. P. F. DUBOIS, A. GREENBAUM, AND G. H. RODRIGUE, *SIAM Review* **20** (1978), 625.
6. T. A. MANTEUFFEL, *Numer. Math.* **28** (1977), 307.
7. A. BRANDT, *Math. Comp.* **31** (1977), 333.
8. B. E. MCDONALD, T. P. COFFEY, S. OSSAKOW, AND R. N. SUDAN, *J. Geophys. Res.* **79** (1974), 2551.
9. B. E. MCDONALD, T. P. COFFEY, S. OSSAKOW, AND R. N. SUDAN, *Radio Sci.* **10** (1975), 247.
10. P. CONCUS AND G. GOLUB, A Generalized Conjugate Gradient Method for Non-Symmetric Systems of Linear Equations, Stanford Report STAN-CS-76-535, 1976.