Journal of Computational Physics **159,** 329–343 (2000)
doi:10.1006/jcph.2000.6448, available online at http://www.idealibrary.com on IDEAL®

Iterative Methods Applied to Matrix Equations Found in Calculating Spheroidal Functions

D. J. Brown* and R. M. Stringfield†

Los Alamos National Laboratory, Los Alamos, New Mexico 87545
E-mail: *djbrown@lanl.gov, †stringfield_r@lanl.gov

Received August 5, 1999; revised January 5, 2000

We look at iterative methods for solving matrix equations, particularly those matrices with small entries. Iterative methods aid computational stability by relying on the topological structure of Banach or Hilbert spaces rather than depending on a calculation's numerical precision. When applicable, they are also quicker than Gaussian elimination. As an example, we use these methods to tabulate the expansion of periodic spheroidal functions in associated Legendre functions, given arbitrary values of the parameters appearing in its defining differential equation. These functions appear in solutions to 3-D Helmholtz equations in oblate and prolate spheroidal coordinates as well as a 1-D Schrödinger equation. © 2000 Academic Press

Key Words: spheroidal wave functions; iterative methods for linear systems; eigenvalue problems.

I. INTRODUCTION

Occasionally, mathematical physics problems present an eigenvalue problem involving a matrix with entries of absolute value less than or equal to 1, with none of the diagonal entries of absolute value 1. In many cases, only the first few eigenvectors are needed, or only low precision of eigenvalues is required. There exist general exact algorithms, such as Jordan or Gaussian elimination and backsubstitution, or the L-U decomposition, which will always provide a solution to the problem if it exists. However, the subtractions and divisions of the small entries occurring in the matrix, necessary in these methods, may require double or higher precision to obtain an accurate result.

In many of these cases, one can benefit by using iterative methods. First, iterative methods can provide meaningful results while using a lower calculation precision. Second, for $n \times n$ matrices meeting certain qualifications, we shall show computation time goes as n^2 multiplied by a factor dependent on the precision chosen, rather than the n^3 in exact methods. However, the applicability of iterative methods is not as general or as straightforward as Gaussian elimination methods. In this paper, we look at a specific variation on the Jacobi iterative scheme; an explanation of the mathematics behind this scheme can be found in Kreyszig [1]. This method seems well suited for solving matrix equations to provide a single fixed-point solution for the vector of expansion coefficients of periodic spheroidal functions in associated Legendre functions.

There are several analyses of the Jacobi and the similar Jacobi overrelaxation methods; the stability of these methods was investigated by Rosanoff and Webel [2], and Udwadia [3], respectively. Manoranjan and Olmos Gomez have looked at a two-step Jacobi method [4]. Several authors have looked at preconditioning the iteration matrix, often by approximately inverse matrices [5–7]; in the last of these, Evans and Okeke categorize various conditioning schemes by the premultiplying conditioning matrix in their Fig. 1.1 and give a thorough stability analysis of the modified preconditioned Jacobi method. Codenotti and Favati [8] have looked at a preconditioning method for a tri-diagonal matrix, which appears in the solution to our problem. However, their method is applied to symmetric matrices. In addition to the Jacobi iteration schemes, there is also significant literature on Gauss–Seidel and related schemes; these algorithms differ from ours. It should be mentioned that we only consider iteration schemes with a single preconditioning of the matrix; iteration schemes such as QR or LR, which produce a sequence of matrices, have not been considered. Our method will only change the n components in the vector of expansion coefficients, and by defining an inner product in this vector space, our method can provide a simple measure of how close a vector is to the desired answer after a number of iterations.

Periodic spheroidal functions are one example of a solution to the Helmholtz equation in a separable coordinate system. Oblate and prolate spheroidal and elliptical cylindrical coordinates allow the Helmholtz equation to be solved by using separation of variables, creating an ordinary differential equation for each coordinate; yet in these cases, at least one of the ordinary differential equations cannot easily be cast as that of a generalized hypergeometric function. A common method of solving these differential equations is by noting the similarities between these coordinate systems and a limiting case, normally spherical coordinates for oblate and prolate spheroidal coordinates, and circular cylindrical for elliptical cylindrical coordinates. The solution to the differential equations found in the more complicated geometries are then expressed as a series expansion of functions which are solutions to the Helmholtz equation in the simpler geometry.

We take a specific example in this paper, the spheroidal functions, solutions to the 3-D Helmholtz equation solved in oblate or prolate spheroidal coordinates. Erdélyi [9] gives the following coordinate transformation for prolate spheroidal coordinates:

$$\begin{aligned} x &= c \sinh u \sin v \cos \phi \\ y &= c \sinh u \sin v \sin \phi \\ z &= c \cosh u \cos v. \end{aligned} \tag{1}$$

Exchange sinh and cosh in (1) to generate the oblate spheroidal transformation. As the descriptive names for the coordinate systems suggest, surfaces of constant u are either prolate or oblate spheroids, and surfaces of constant v are two-sheeted hyperboloids in prolate spheroidal coordinates and one-sheeted hyperboloids in oblate spheroidal coordinates. c is an arbitrary constant representing the separation distance between the two foci of the ellipses in the coordinate system.

In either of these two coordinate systems, the Helmholtz equation separates. In both of these systems, one of the decoupled ordinary differential equations corresponding to the longitudinal angle coordinate ϕ gives ordinary trigonometric functions. All the other differential equations can be shown to be of the form

$$\frac{d^2V}{dv^2} + \cot v \frac{dV}{dv} + [\lambda + (kc \sin v)^2 - (m \csc v)^2]\, V = 0 \tag{2}$$

by an imaginary change of variable, an imaginary change of variable and translation, or an imaginary k. In (2), λ and m are constants of separation; k^2 is the constant appearing in the 3-D Helmholtz equation in Laplacian form. With a change of variable $v = 2\tan^{-1} e^y$, (2) becomes

$$\frac{d^2V}{dy^2} + [\lambda \operatorname{sech}^2 y + (kc)^2 \operatorname{sech}^4 y - m^2]V = 0. \tag{3}$$

We can recognize this as a 1-D Schrödinger equation.

It is useful to make the transformation $x = \cos v$. In the limit $c = 0$, spheroidal coordinates are spherical polar coordinates. In spherical coordinates, the solutions for the latitudinal angle coordinate go as associated Legendre polynomials in $\cos\theta$. Since these polynomials make up an orthogonal expansion set, it may be helpful to formulate the problem to avoid a sinusoidal dependence as an argument of spheroidal functions. This will also help us to see the analogy between v and θ. In this case, the differential equation becomes

$$(1 - x^2)\frac{d^2V}{dx^2} - 2x\frac{dV}{dx} + [\lambda + \gamma^2(1 - x^2) - m^2(1 - x^2)^{-1}]V = 0. \tag{4}$$

We have made the substitution $\gamma^2 = k^2c^2$.

Many authors have examined the problem of calculating spheroidal functions. Most of the original tables were tabulated by Meixner and Schäfke [10], Corbató and Little [11], and Flammer [12]. They expand the angular spheroidal functions in terms of associated Legendre polynomals, and solve the recursion relation for the expansion coefficients via truncating a continued fraction and obtaining numerical values using an overall normalization condition on the spheroidal function. Gianfelice *et al.* [13] write the expansion coefficients and eigenvalues as a power series in γ. Nesterov and Skorodumov [14] truncate the recursion relations and rewrite this as a ratio of polynomials, using a root solver to determine the eigenvalues. Li *et al.* [15] write the recursion relations in matrix form and treat it formally as an eigenvalue problem, solving it using a Mathematica eigenvalue solver. They determine the expansion coefficients by substituting the eigenvalues in the continued fraction and using a normalization condition on the spheroidal function. We shall use a combination of Li *et al.*'s idea to find eigenvalues of matrix and a root solver to calculate the zero of a determinant, similar to Nesterov and Skorodumov. Our method has the advantage of formulating the eigenvalue problem without reference to variables with no set value assigned, as well as directly providing a matrix, which after some manipulations will be used in finding the expansion coefficients.

II. THE ALGORITHMS

To demonstrate the use of iterative methods for solving eigenvalue problems, to solve the differential equation (4) we must reformulate it as a matrix problem. Since (4) is the differential equation for the variable analogous to the latitudinal angle, we restrict our discussion to periodic finite spheroidal functions, although this is not a restriction on the computational method. In the limit of $c = 0$, this differential equation goes to that of the associated Legendre functions for $\lambda = v(v+1)$. The general solution to this Legendre differential equation is

$$AP_v^m(x) + BQ_v^m(x). \tag{5}$$

The only solutions without singularities are the P solutions with v an integer. Clearly, to investigate solutions with certain instabilities at -1 or 1, other solutions can be considered. Thus, a solution to (4) can be formally written as an expansion in associated Legendre polynomials:

$$Ps_n^m(x,\gamma) = \sum_{l=m-n}^{\infty} A_{l,n}^m(\gamma) P_{n+l}^m(x). \tag{6}$$

Substituting this series into the differential equation for spheroidal functions gives the following recursion relations among the expansion coefficients using orthogonality of the expansion set:

$$\begin{aligned}&\left[\lambda - (m+l)(m+l+1) + 2\gamma^2 \frac{(m+l)(m+l+1)+m^2-1}{(2m+2l-1)(2m+2l+3)}\right] A_{l,n}^m \\ &\quad - \gamma^2 \frac{(2m+l+1)(2m+l+2)}{(2m+2l+3)(2m+2l+5)} A_{l+2,n}^m - \gamma^2 \frac{l(l-1)}{(2m+2l-3)(2m+2l-1)} A_{l-2,n}^m = 0.\end{aligned} \tag{7}$$

First, note that the even and odd coefficients are not coupled. It makes sense to break this problem into two functions, one for the even coefficients and one for the odd. The function with even coefficients corresponds to an even function, and that with odd coefficients corresponds to an odd function; for a concise exposition we now focus on the even spheroidal functions. The series of recursion relations (7) can be written using linear algebra in the form of a matrix premultiplying a vector, giving

$$\begin{bmatrix} F_{00} & F_{02} & 0 & 0 & 0 & 0 & 0 & \cdots \\ F_{20} & F_{22} & F_{24} & 0 & 0 & 0 & 0 & \cdots \\ 0 & F_{42} & F_{44} & F_{46} & 0 & 0 & 0 & \cdots \\ 0 & 0 & F_{64} & F_{66} & F_{68} & 0 & 0 & \cdots \\ 0 & 0 & 0 & F_{86} & F_{88} & F_{8,10} & 0 & \cdots \\ 0 & 0 & 0 & 0 & F_{10,8} & F_{10,10} & F_{10,12} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix} \begin{bmatrix} A_0 \\ A_2 \\ A_4 \\ A_6 \\ A_8 \\ A_{10} \\ \vdots \end{bmatrix} = 0, \tag{8}$$

where

$$\begin{aligned} F_{l,l-2} &= -\gamma^2 l(l-1)(2m+2l+3)(2m+2l+5) \\ F_{l,l+2} &= -\gamma^2 (2m+l+1)(2m+l+2)(2m+2l-3)(2m+2l-1) \\ F_{l,l} &= [\lambda - (m+l)(m+l+1)](2m+2l-3)(2m+2l-1)(2m+2l+3) \\ &\quad \times (2m+2l+5) + 2\gamma^2[(m+l)(m+l+1)+m^2-1](2m+2l-3)(2m+2l+5). \end{aligned} \tag{9}$$

Either the vector of expansion coefficients is identically zero, giving the zero solution to (4), or the determinant of the matrix is zero. Since it is difficult to obtain information from the semi-infinite determinant, truncate it after a finite number of terms, creating a $k \times k$ determinant. For example,

$$\begin{aligned} T_1 &= \begin{Vmatrix} F_{00} & F_{02} \\ F_{20} & F_{22} \end{Vmatrix} = F_{00}F_{22} - F_{02}F_{20}; \\ T_2 &= \begin{Vmatrix} F_{00} & F_{02} & 0 \\ F_{20} & F_{22} & F_{24} \\ 0 & F_{42} & F_{44} \end{Vmatrix} = F_{00}F_{22}F_{44} - F_{02}F_{20}F_{44} - F_{00}F_{24}F_{42}. \end{aligned} \tag{10}$$

Substituting values for λ or γ, or a relationship between λ and γ, we will then be able to solve for the dependent parameter after setting the determinant equal to zero. Note that one cannot set both λ and γ; if m is fixed, this would be equivalent to completely specifying both the exact potential and all the eigenvalues of a Schrödinger equation. The resulting polynomial from setting the determinant equal to zero will have multiple roots.

While there are several ways to determine roots of polynomials, most of which have been incorporated into standard library functions, we shall use possibly the simplest iterative algorithm, inverse linear interpolation or the secant method. Roughly, this entails keeping only the first term in the Taylor series and approximating the derivative by a secant running through the root:

$$\lambda_{i+1} = \lambda_i - \frac{(\lambda_i - \lambda_{i-1})T_k(\lambda_i)}{T_k(\lambda_i) - T_k(\lambda_{i-1})}. \tag{11}$$

A good guess for the nth root of the polynomial equation might be $n(n+1)$, especially for small γ, as this is the limit as γ goes to zero. Multiplying (7) by the least common denominator, as we did to obtain (9), will eliminate instabilities caused when the denominators pass through zero, making the determinants T infinite. We should also note that by using trial matrices, the final matrix obtained when the root converges also provides a matrix which can be modified to determine the eigenvalues, since this matrix has a non-zero eigenvector solution to (8).

With the eigenvalue part of this matrix problem solved, the eigenvectors must be determined to specify the spheroidal function completely; we use an iterative scheme rather than an exact direct method. Note a matrix equation $\overleftrightarrow{F}\vec{A} = 0$ with a non-zero eigenvector can be rewritten $(\overleftrightarrow{I} - \overleftrightarrow{F})\vec{A} = \vec{A}$. Using this as an iteration matrix is the Jacobi iteration method; however, this matrix must be modified to ensure a stable iterative scheme with a fixed point. We change the iteration matrix $(\overleftrightarrow{I} - \overleftrightarrow{F})$ so no element in the iteration matrix has an absolute value greater than 1.

To modify the iteration matrix, take the final matrix formed in (11) when the eigenvalue converged and note the signs on the diagonal element and the largest element in each row. If the diagonal element has the largest absolute value, divide each term in the row by the negative of the diagonal element. If the diagonal element does not have the largest absolute value and the element with the largest absolute value is of the same sign as the diagonal element, divide each term in the row by the negative of the largest element. If the diagonal element and the element with the largest absolute value are of different sign, divide each term in the row by the largest element. After these divisions, each term should now lie between -1 and 1, and all diagonal elements should lie between -1 and 0. Next we add a unit matrix to this modified matrix; we now call the resultant matrix $\overleftrightarrow{B}$. Note that this matrix was constructed such that $\vec{A} = \overleftrightarrow{B}\vec{A}$, all off-diagonal terms lie between -1 and 1, and the diagonal terms lie between 0 and 1. Thus, this matrix fulfills the requirement of our iteration matrix.

Use the matrix generated in the previous paragraph as a recursion relation between vectors. As a last point to the algorithm, a flag to stop iteration is needed. Conditions can be set on the expansion coefficients themselves, but it will prove more useful to set the condition that the norm of the vector does not change. The orthogonality condition of associated Legendre polynomials gives

$$\int_{-1}^{1} \left[Ps_n^m(x,\gamma)\right]^2 dx = \sum_{l=0}^{\infty} \left[A_{l,n}^m(\gamma)\right]^2 \frac{(2m+l)!}{l!(m+l+1/2)}. \tag{12}$$

There are many ways to normalize periodic spheroidal functions, and there seems to be no standardization in the literature. For our purpose, we use the normalization in Erdélyi, which appears to have been taken from Meixner and Schäfke, as it directly sets the value of the sum in (12). Other normalizations, such as those in Stratton *et al.* or in Flammer, set the value of the spheroidal function at a given point. As the sum in (12) is similar to a dot inner product of a vector, only with a weight function, this will prove more useful to our discussions, since defining a norm or inner product is necessary to analyzing iteration schemes. Finally, we want to emphasize that the normalization only sets the overall constant undetermined in the iteration scheme; the ratios between expansion coefficients must be the same without regard to any normalization scheme. The normalization we shall use is

$$\sum_{l=0}^{\infty} \left[A_{l,n}^m(\gamma)\right]^2 \frac{(2m+l)!}{l!(m+l+1/2)} = \frac{(n+m)!}{(n-m)!(n+1/2)}. \tag{13}$$

To normalize the spheroidal functions, after every iteration calculate

$$t = \frac{(n-m)!(n+1/2)}{(n+m)!} \sum_{l=0}^{k} \left[A_{l,n}^m(\gamma)\right]^2 \frac{(2m+l)!}{l!(m+l+1/2)}; \tag{14}$$

k is the dimension of the iteration matrix used. When t is constant to the precision of the calculation, divide each expansion coefficient by $t^{1/2}$.

Since all the coefficients in (13) come in as the square, there is still a sign ambiguity in the normalization. Select

$$A_{0,n}^m(\gamma) > 0. \tag{15}$$

The expansion coefficients may fluctuate between two sets of values, one of which is the negative of the other. This is not prohibited, since the sign of the expansion coefficients has not yet been set, and the square root can be of either sign. Finally, use (15) to determine completely the periodic spheroidal function; this can be done by inspection.

III. MATHEMATICAL ANALYSIS

There are two parts to the algorithm described above; first find allowed values of λ given γ, then determine the values of the coefficients for the associated Legendre polynomials for one particular allowed value of λ. In determining the eigenvalues, there are two questions of merit. First, under what conditions will approximating the infinite matrix by a finite matrix yield accurate solutions? Second, under what conditions will the root solver give correct roots when setting the determinant equal to zero?

To determine how large a finite matrix will give accurate roots, note that fixing to zero those expansion coefficients of order greater than the dimension of matrix when calculating the norm in Eq. (14) effectively decouples the first k expansion coefficients from the rest. Therefore, an infinite representation of a $k \times k$ partial determinant used to calculate λ is a block diagonal matrix with the first $k \times k$ terms with the same entries as the partial determinant, and the other terms given by a diagonal unit matrix. This infinite representation of the finite determinants implies reasonable answers as long as

$$|F_{k-1,k}/F_{k,k}|, \qquad |F_{k+1,k}/F_{k,k}| \leq \varepsilon, \tag{16}$$

where ε is the set precision of the calculation. This follows since the calculation will not be able to distinguish these matrix elements from zero. Note also that in order to calculate $Ps_n^m(x, \gamma)$, the determinant of the truncated matrix, T_k, must have at least n roots. Preferably, one will pick a k which gives considerably more than n roots, since the last roots of the determinant change dramatically as we move to a greater k.

The second part of the eigenvalue problem depends heavily on the root solver utilized. We have chosen inverse linear interpolation because it is simple and familiar, and it allows determinants to be calculated with numbers, not functions, while not requiring storage of several previous iterations. It also provides the matrix useful in the next step of the calculation. One can choose quadratic inverse linear interpolation for greater accuracy, or other methods as the problem may dictate. Error analyses of many standard root solvers can be found in various calculus or numerical analysis texts or Abramowitz and Stegun [16].

One analysis of the calculation of eigenvectors relies on Banach's contraction mapping theorem. This analysis has the advantage of being a generalization for the case of a function of one variable. The iterated vector Bx can be replaced by $Fx = Bx/\|Bx\|$. This new operator changes the algorithm in that normalization must be performed after every iteration, not merely at the very end. The operator F is always a non-linear, non-expansive operator on a complete Hilbert space and maps the closed unit ball into the closed unit ball. Thus, a fixed point exists via Brouwer's theorem. However, Brouwer's theorem does not provide uniqueness information; this is found by determining both whether this fixed point is attractive (unlike $x = 0$ for $x^{1/3}$) and the radius of convergence if attractive. A stable fixed point should have the property that a vector close to the solution will get closer still after an iteration. This can be linearized for an h close to the solution by the Fréchet derivative to obtain a simple equation for stability of a fixed point. Denote L_F as the Fréchet derivative linear operator of F which acts on h. Then

$$\|F(\overline{x}_{\mathrm{sol}} + h) - \overline{x}_{\mathrm{sol}}\| = \|F(\overline{x}_{\mathrm{sol}} + h) - F\overline{x}_{\mathrm{sol}}\| < \|(\overline{x}_{\mathrm{sol}} + h) - \overline{x}_{\mathrm{sol}}\| = \|h\| \tag{17}$$

or

$$\|L_F h\| < \|h\|, \tag{18}$$

which implies the norm of the Fréchet derivative of F must be less than 1. The inequality in (17) is the condition for a contraction mapping; in some sense this is a local contraction mapping. If this condition is a strict inequality, the norm of the operator F in the region where the fixed point is attractive will be less than 1, and all results of the contraction mapping theorem apply locally in the region where the fixed point is attractive.

A more powerful analysis of the stability of the iteration method uses the power method on a Hilbert space; a summary of Fröberg's [17] exposition follows. An arbitrary vector can be expanded in an eigenvector basis, where the eigenvalues and eigenvectors are those of the matrix operator B:

$$y = \sum_i c_i \hat{e}_i. \tag{20}$$

We have ordered the eigenvectors $\hat{e}_i$ such that

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \cdots. \tag{21}$$

By the definition of eigenvalues and eigenvectors, repeated iteration gives, after dividing by the largest eigenvalue,

$$\frac{B^n y}{\lambda_1^n} = c_1 \hat{e}_1 + \sum_{i=2} \left(\frac{\lambda_i}{\lambda_1}\right)^n c_i \hat{e}_i. \tag{22}$$

If the absolute value of the first eigenvalue is strictly greater than the other eigenvalues, we see

$$\lim_{n\to\infty} \frac{B^n y}{\lambda_1^n} = c_1 \hat{e}_1. \tag{23}$$

We know an eigenvalue is 1; this is the definition of a fixed point. If this eigenvalue is the eigenvalue with the largest absolute value, we can simplify (23):

$$\lim_{n\to\infty} B^n y = c_1 \hat{e}_1. \tag{24}$$

Repeated multiplication of an arbitrary vector with the iteration matrix will converge to the desired eigenvector multiplied by a constant. This constant can be eliminated by calculating the norm of the final vector.

Note, however, that the eigenvalue with the largest absolute value must be 1. If the norm converges in this iteration scheme, and the new iterated vector need not be renormalized after each iteration, only at the end, this must be true. This is the case for all the spheroidal functions we calculated. Showing our iteration matrix has no eigenvalue with absolute value greater than 1 is somewhat difficult; we present an argument using Gershgorin's theorem. This theorem states if λ is an eigenvalue of a square $n \times n$ matrix B, then for some j, where $1 \leq j \leq n$,

$$|B_{jj} - \lambda| \leq \sum_{\substack{k=1\\k\neq j}}^{n} |B_{jk}|. \tag{25}$$

For all the matrices we used to calculate the results listed below, there was only one non-zero diagonal element. For the row with a non-zero diagonal element, one of the off-diagonal elements had absolute value 1. For all rows with a zero diagonal element the sum of the absolute value of the off-diagonal elements was less than 1. This last statement, combined with Gershgorin's theorem, implies that all but one eigenvalue must be less than 1; for the row with a non-zero diagonal element, we have

$$|B_{jj} - \lambda| \leq 2; \qquad |B_{jj}| < 1. \tag{26}$$

This last equation certainly allows an eigenvalue of 1, which must exist to have a fixed point. One may also use Gershgorin's theorem and other theorems relating elements of a matrix to eigenvalues as a guide in determining how to precondition a matrix beforehand.

Next we examine some general cases, where the largest eigenvalue is not 1, or there are complex eigenvalues, as well as investigate the circumstances in which the method will fail to converge to a fixed point. First, we look at the case where the largest eigenvalue of the matrix B is not 1. The easiest solution is to find the value of the largest eigenvalue and divide each matrix element by this eigenvalue. This follows from the statement that if the eigenvalues of a matrix M are λ_i, then the eigenvalues of cM are $c\lambda_i$; by dividing by the largest eigenvalue, we ensure that the largest eigenvalue is set to 1. This should not effect the solutions to (8); by multiplying every element by an overall factor c, the resulting vector formed by premultiplying the matrix by an arbitrary vector is merely a multiple c of the vector formed by a matrix which was not multiplied by c. We eliminate all these overall factors when we normalize the final vector. To find the largest eigenvalue for an arbitrary matrix, use the other part of the power method. Note from (21),

$$\frac{(B^n y, B^{n+1} y)}{(B^n y, B^n y)} = \frac{\sum_i |c_i|^2 \lambda_i^{2n+1}}{\sum_i |c_i|^2 \lambda_i^{2n}} = \lambda_1 \frac{1 + \sum_{i=2} \left|\frac{c_i}{c_1}\right|^2 \left(\frac{\lambda_i}{\lambda_1}\right)^{2n+1}}{1 + \sum_{i=2} \left|\frac{c_i}{c_1}\right|^2 \left(\frac{\lambda_i}{\lambda_1}\right)^{2n}}. \tag{27}$$

In the limit n becomes infinite, (27) goes to the largest eigenvalue. One will also obtain this by taking the ratio of norms from one iteration to the next.

Since we did not restrict this method to symmetric matrices, there is no guarantee that all the eigenvalues will be real. For most cases, complex eigenvalues will not pose any difficulty. For this discussion, we assume the absolute value of the largest eigenvalue has been set to 1. Rewriting the complex eigenvalue in polar form,

$$\lambda_j^n = r_j^n \exp(in\phi_j); \qquad 0 \leq r_j < 1. \tag{28}$$

Thus for large numbers of iterations, the real multiplier and thus the power of the eigenvalue goes to 0.

This does not address the possibility of several eigenvalues with absolute value 1. The general case for an $n \times n$ matrix with m eigenvalues on the unit circle in the complex plane does not necessarily give fixed points, or fixed cycles. A fixed cycle returns a vector to itself after a finite number of iterations on the vector. For the case of a fixed cycle taking k iterations to return to the initial vector, each of the vectors in the cycle must satisfy $x_i = M^k x_i$, so the eigenvalues are the k roots of unity. In this case, the eigenvectors can be constructed by linear combinations of the m vectors in a cycle. For example, with a cycle of four, we have

$$\begin{aligned}
\hat{v}_1 &= c_1\hat{e}_1 + ic_2\hat{e}_2 - c_3\hat{e}_3 - ic_4\hat{e}_4 & \hat{e}_1 &= (\hat{v}_1 + \hat{v}_2 + \hat{v}_3 + \hat{v}_4)/4c_1 \\
\hat{v}_2 &= c_1\hat{e}_1 - c_2\hat{e}_2 + c_3\hat{e}_3 - c_4\hat{e}_4 & \hat{e}_2 &= (\hat{v}_4 - \hat{v}_2 + i\hat{v}_3 - i\hat{v}_1)/4c_2 \\
\hat{v}_3 &= c_1\hat{e}_1 - ic_2\hat{e}_2 - c_3\hat{e}_3 + ic_4\hat{e}_4 & \hat{e}_3 &= (\hat{v}_2 + \hat{v}_4 - \hat{v}_1 + \hat{v}_3)/4c_3 \\
\hat{v}_4 &= c_1\hat{e}_1 + c_2\hat{e}_2 + c_3\hat{e}_3 + c_4\hat{e}_4 & \hat{e}_4 &= (\hat{v}_4 - \hat{v}_2 + i\hat{v}_1 - i\hat{v}_3)/4c_4.
\end{aligned} \tag{29}$$

Clearly this will only be useful for small cycles.

If the eigenvalues are not roots of unity, and if the difference of the arguments of two eigenvalues with modulus 1 is not a rational number, the iteration scheme will never reach a fixed point or cycle; the method fails. One should try to manipulate the matrix to avoid this situation. It is often useful to manipulate the matrix to avoid fixed cycles as well, and to attempt to work only with fixed points.

Also, numerical instabilities will exist if $|\lambda_1| - |\lambda_2| < \varepsilon$, the precision of the calculation. The computation scheme will not be able to resolve any difference between the first and second eigenvalue; this may artificially force the computation into a fixed cycle (if the difference in the arguments of λ_1 and λ_2 is rational), diverge (if the difference in the arguments is irrational), or give an incorrect result (if the difference is zero).

In concluding the discussion of the validity of the iterative method, we note that it may be possible to identify the eigenvectors with eigenvalues less than 1 with possible failure modes when exact methods give incorrect results. This seems plausible if somehow the first eigenvector cannot be resolved; an eigenvector with lower eigenvalue may then result as the answer.

Next we look at the running time of the iterative algorithm. For each iteration, multiplying an n-component vector by an $n \times n$ matrix produces n^2 multiplications followed by $n(n-1)$ additions. The run time per iteration then is $O(n^2)$. The contraction mapping theorem also provides an estimate of convergence of solutions. One can show

$$\|\bar{y} - y_m\| \leq \frac{b^m}{1-b}\|y_1 - y_0\| \leq \frac{b^m}{1-b}(\|y_1\| + \|y_0\|) \leq \frac{b^m(b+1)}{1-b}\|y_0\|. \tag{30}$$

Here $\bar{y}$ is the fixed point, y_i is the ith iteration of an arbitrary vector, and b is the bound or norm of the Fréchet derivative. The left-hand side of this equation can be set to be the precision of the calculation ε. This gives

$$m \geq \ln \frac{\varepsilon(1-b)}{\|y_0\|(1+b)} \Big/ \ln b. \tag{31}$$

The number of iterations to reach convergence to within a precision ε goes as ln(ε) plus a constant term. However, this does not depend on the size of the matrix, so the run time is $O(n^2 \ln \varepsilon)$.

Similarly, we can analyze the run time using the power method. The number of iterations must be such that

$$|\lambda_2/\lambda_1|^m = |\lambda_2|^m < \varepsilon. \tag{32}$$

Taking the logarithm of both sides gives, after noting $\ln|\lambda_2| < 0$,

$$m > \ln\varepsilon / \ln|\lambda_2|. \tag{33}$$

Once again, the number of iterations goes as ln(ε) plus a constant term.

While not a general trait of matrix problems, the calculation of spheroidal functions involves an infinite matrix. There are two additional issues encountered; first whether the solution generated by the expansion in associated Legendre functions converges, and second, whether the sequence of iterative solutions for finite dimensional matrices converge in the limit of an infinite matrix. For our example, the first problem is usually proved in texts discussing spheroidal functions via continued fractions. Since the equivalence of determinants of truncated matrices and terminated continued fractions is discussed in texts on continued fractions such as Wall [18], and proofs of convergence for continued fractions can be found in Stratton, we do not discuss it further here and instead turn our attention to the convergence of the sequence of iterative solutions. Denote the iterative solution of the infinite matrix by $\overline{x}_{\text{sol}}$ and that of the $n \times n$ finite matrix by x_n. Call the operator formed by using an $n \times n$ matrix B_n. y is an arbitrary vector of norm one. Extend B_n to an infinite dimensional space by appending 0's for off-diagonal terms with one index greater than n, and appending 1's for on-diagonal terms with indices greater than n. Similarly extend x_n to an infinite dimensional space by appending zeros for vector components with indices greater than n:

$$\begin{aligned}
\|\overline{x}_{\text{sol}} - x_n\| &\equiv \lim_{m\to\infty}\lim_{n\to\infty}\left\|B^m y - B_n^m y\right\| \\
&\le \lim_{m\to\infty}\lim_{n\to\infty}\sum_{i=0}^{m-1}\left\|B^{m-1-i}(B - B_n)B_n^i y\right\| \\
&\le \lim_{m\to\infty}\lim_{n\to\infty}\sum_{i=0}^{m-1}\|B^{m-1-i}\|\|B - B_n\|\left\|B_n^i y\right\| \\
&= \lim_{m\to\infty}\sum_{i=0}^{m-1}\|B^{m-1-i}\|0\left\|B_n^i y\right\| = 0.
\end{aligned} \tag{34}$$

This follows from the triangle inequality and the boundedness of B. Thus, a unique fixed point exists for operators produced by the finite matrices, and the sequence of these fixed points will converge to a solution of the infinite matrix.

One final comment is in order. Our example of spheroidal functions does not really take the best advantage of the calculation speed mentioned above; both exact and iterative methods should run in linear time on a tri-diagonal matrix. However, accuracy is ensured by the topological properties of a Hilbert space, and this is independent of the accuracy of the mathematical processor. Thus, assuming the processor is capable of obtaining the precision desired or better, the results of iterations will be forced to the fixed point, regardless of cumulative computational errors. This feature of iterative methods is always present when the iteration matrices satisfy the conditions discussed above.

IV. RESULTS

We now present results of our algorithm for the case of $Ps_n^m(x, \gamma)$, and compare them with Flammer's and Little and Corbató's tables. All computations were done using Maple V, Release V on a Windows NT machine with an Intel Pentium II processor. In Table I, we list eigenvalues from Flammer and the same numbers calculated from the iteration scheme (11). Note that the differential equation for angular spheroidal functions in Flammer is slightly different from that in Meixner and Schäfke; Flammer's equation is

$$(1 - x^2)\frac{d^2V}{dx^2} - 2x\frac{dV}{dx} + [\lambda - c^2x^2 - m^2(1 - x^2)^{-1}]V = 0. \tag{35}$$

TABLE I
Eigenvalues: Flammer's and Ours

γ	$\lambda_{00,\text{Flammer}}$	λ_{00}	γ	$\lambda_{01,\text{Flammer}}$	λ_{01}	γ	$\lambda_{02,\text{Flammer}}$	λ_{02}
0	0	0	0	2	2	0	6	6
0.2	0.01331	0.01330965977	0.2	2.02399	2.023989025	0.2	6.02096	6.020968591
0.4	0.05296	0.0529560091	0.4	2.09582	2.095824209	0.4	6.08407	6.084067407
0.6	0.11810	0.1181021565	0.6	2.21511	2.215108515	0.6	6.18986	6.189864446
0.8	0.20739	0.2073904977	0.8	2.38118	2.381175780	0.8	6.33927	6.339268662
1.0	0.31900	0.3190000551	1.0	2.59308	2.593084580	1.0	6.53347	6.533471800
1.2	0.45073	0.4507283534	1.2	2.84961	2.849610685	1.2	6.77387	6.773865201
1.4	0.60010	0.600096432	1.4	3.14924	3.149239296	1.4	7.06193	7.061932809
1.6	0.76447	0.764471910	1.6	3.49016	3.490158703	1.6	7.39913	7.399125343
1.8	0.94120	0.941200653	1.8	3.87026	3.870257570	1.8	7.78673	7.786724914
2.0	1.12773	1.127734064	2.0	4.28713	4.287128543	2.0	8.22572	8.225713001
γ	$\lambda_{03,\text{Flammer}}$	λ_{03}	γ	$\lambda_{11,\text{Flammer}}$	λ_{11}	γ	$\lambda_{12,\text{Flammer}}$	λ_{12}
0	12	12	0	2	2	0	6	6
0.2	12.02045	12.02044972	0.2	2.00799	2.007992694	0.2	6.01714	6.017136638
0.4	12.08186	12.08186235	0.4	2.03188	2.031883469	0.4	6.06847	6.068471974
0.6	12.18443	12.18442966	0.6	2.07141	2.071413195	0.6	6.15378	6.153782607
0.8	12.32848	12.32847557	0.8	2.12616	2.126159145	0.8	6.27270	6.272697364
1.0	12.51446	12.51446215	1.0	2.19555	2.195548355	1.0	6.42470	6.424699144
1.2	12.74300	12.74299682	1.2	2.2788755	2.278875547	1.2	6.60913	6.609127680
1.4	13.01484	13.01483980	1.4	2.37533	2.375324823	1.4	6.82518	6.825183428
1.6	13.33091	13.33090985	1.6	2.48399	2.483994144	1.6	7.07193	7.071932764
1.8	13.69229	13.69228618	1.8	2.60392	2.603921437	1.8	7.34832	7.348314733
2.0	14.10020	14.10020388	2.0	2.7341110	2.734111025	2.0	7.65315	7.653149562
γ	$\lambda_{13,\text{Flammer}}$	λ_{13}	γ	$\lambda_{22,\text{Flammer}}$	λ_{22}	γ	$\lambda_{23,\text{Flammer}}$	λ_{23}
0	12	12	0	6	6	0	12	12
0.2	12.01867	12.01866884	0.2	6.00571	6.005711178	0.2	12.012333	12.01332974
0.4	12.07470	12.07470112	0.4	6.02281	6.022807533	0.4	12.05328	12.05327592
0.6	12.16817	12.16817139	0.6	6.05118	6.051178349	0.6	12.11971	12.11970969
0.8	12.29920	12.29919503	0.8	6.09064	6.090641815	0.8	12.21242	12.21241728
1.0	12.46792	12.46791533	1.0	6.14095	6.140948992	1.0	12.33110	12.33110151
1.2	12.6744862	12.67448617	1.2	6.20179	6.201789172	1.2	12.47538	12.47538389
1.4	12.91905	12.91905116	1.4	6.27280	6.272796426	1.4	12.64481	12.64480735
1.6	13.20172	13.20172024	1.6	6.35356	6.353557109	1.6	12.83884	12.83883962
1.8	13.52255	13.52254490	1.8	6.44362	6.443618072	1.8	13.05688	13.05687728
2.0	13.8814934	13.88149342	2.0	6.54250	6.542495274	2.0	13.29825	13.29825047

Thus, Flammer's c is our γ, but Flammer's λ is our $\lambda + \gamma^2$. We have already accounted for this difference in our tables. The average number of iterations to 10-digit convergence is seven.

In Table II we compare the ratios of expansion coefficients for selected $Ps_n^m(x, \gamma)$ with $\gamma = 1$ and 2 using the eigenvalues determined above with ratios computed from Little and Corbató's tables. We use Flammer's tables for those cases where he gives greater than the seven places in Little and Corbató. We look at the ratios because these quantities are independent of any normalization selected. Our expansion coefficients are calculated by iterating until the value of the norm is constant to at least nine places. This takes less than 30 iterations.

Given that both Little and Corbató and Flammer state their results may be off by one or more in the last place, and that these errors propagate through the division, our results are not inconsistent.

TABLE II

Ratios of Expansion Coefficients: Flammer's, Little and Corbató's, and Ours

	$Ps_2^1(x,1)_{LC}$	$Ps_2^1(x,1)$		$Ps_2^1(x,2)_{LC}$	$Ps_2^1(x,2)$
A_1/A_3	−81.98272	−81.9826884	A_1/A_3	−20.77072	−20.7707097
A_3/A_5	−178.5232	−178.523183	A_3/A_5	−44.91608	−44.9160701
A_5/A_7	−306.7511	−306.751005	A_5/A_7	−76.97433	−76.9743615
A_7/A_9	−466.8760	−466.876119	A_7/A_9	−117.0052	−117.005117
			A_9/A_{11}	−165.0239	−165.023904
			A_{11}/A_{13}	−221.0365	−221.036485
	$Ps_3^1(x,1)_{LC}$	$Ps_3^1(x,1)$		$Ps_3^1(x,2)_{Fl}$	$Ps_3^1(x,2)$
A_0/A_2	+3.339112e-2	+3.33911151e-2	A_0/A_2	+0.12375847	+0.123758461
A_2/A_4	−94.59394	−94.5939523	A_2/A_4	−23.684528	−23.6845291
A_4/A_6	−209.8502	−209.850179	A_4/A_6	−52.527261	−52.5272607
A_6/A_8	−355.3031	−355.303200	A_6/A_8	−88.901066	−88.9010681
A_8/A_{10}	−532.1282	−532.127963	A_8/A_{10}	−133.11223	−133.112225
			A_{10}/A_{12}	−185.24788	−185.247878
			A_{12}/A_{14}	−245.34235	−245.342354
			A_{14}/A_{16}	−313.41190	−313.411914
			A_{16}/A_{18}	−389.46523	−389.465258
			A_{18}/A_{20}	−473.50745	−473.507459
	$Ps_2^2(x,1)_{LC}$	$Ps_2^2(x,1)$		$Ps_2^2(x,2)_{LC}$	$Ps_2^2(x,2)$
A_0/A_2	−249.5560	−249.555972	A_0/A_2	−65.83287	−65.8328673
A_2/A_4	−299.5804	−299.580418	A_2/A_4	−76.85448	−76.8544958
A_4/A_6	−431.1593	−431.159206	A_4/A_6	−109.4333	−109.433241
A_6/A_8	−601.8268	−601.826844	A_6/A_8	−151.9579	−151.957977
			A_8/A_{10}	−203.1344	−203.134407
			A_{10}/A_{12}	−262.6083	−262.608289
	$Ps_3^2(x,1)_{LC}$	$Ps_3^2(x,1)$		$Ps_3^2(x,2)_{LC}$	$Ps_3^2(x,2)$
A_1/A_3	−190.0881	−190.088070	A_1/A_3	−48.37032	−48.3703357
A_3/A_5	−315.5598	−315.559843	A_3/A_5	−79.64133	−79.6413227
A_5/A_7	−474.4665	−474.466384	A_5/A_7	−119.3187	−119.318729
A_7/A_9	−665.8549	−665.854870	A_7/A_9	−167.1353	−167.135318
			A_9/A_{11}	−223.0166	−223.016562
	$Ps_0^0(x,1)_{LC}$	$Ps_0^0(x,1)$		$Ps_0^0(x,2)_{LC}$	$Ps_0^0(x,2)$
A_0/A_2	−9.302358	−9.30236137	A_0/A_2	−2.594044	−2.59404295
A_2/A_4	−58.87578	−58.8757937	A_2/A_4	−15.22100	−15.2209994
A_4/A_6	−139.2044	−139.204417	A_4/A_6	−35.36820	−35.3681952
A_6/A_8	−251.3487	−251.348675	A_6/A_8	−63.43610	−63.4361232
A_8/A_{10}	−395.4301	−395.430041	A_8/A_{10}	−99.47543	−99.4754256
			A_{10}/A_{12}	−143.5010	−143.501067

Table II—*Contiuned*

	$Ps_1^0(x,1)_{LC}$	$Ps_1^0(x,1)$		$Ps_1^0(x,2)_{LC}$	$Ps_1^0(x,2)$
A_1/A_3	−24.78931	−24.7893217	A_1/A_3	−6.075177	−6.07517888
A_3/A_5	−87.91655	−87.9165486	A_3/A_5	−21.82251	−21.8225076
A_5/A_7	−183.5455	−183.545478	A_5/A_7	−45.71781	−45.7178151
A_7/A_9	−311.3402	−311.340241	A_7/A_9	−77.66054	−77.6605406
			A_9/A_{11}	−177.6244	−117.624388
			A_{11}/A_{13}	−165.5995	−165.599419
	$Ps_2^0(x,1)_{LC}$	$Ps_2^0(x,1)$		$Ps_2^0(x,2)_{LC}$	$Ps_2^0(x,2)$
A_0/A_2	+2.150491e-2	+2.15048961e-2	A_0/A_2	+7.738010e-2	+7.73801443e-2
A_2/A_4	−40.74947	−40.7494916	A_2/A_4	−10.04196	−10.0419562
A_4/A_6	−118.6964	−118.696388	A_4/A_6	−29.51113	−29.5111296
A_6/A_8	−229.7089	−229.708879	A_6/A_8	−57.25654	−57.2565390
A_8/A_{10}	−373.1270	−373.126923	A_8/A_{10}	−93.10669	−93.1066893
			A_{10}/A_{12}	−137.0079	−137.007878
	$Ps_3^0(x,1)_{LC}$	$Ps_3^0(x,1)$		$Ps_3^0(x,2)_{LC}$	$Ps_3^0(x,2)$
A_1/A_3	+1.729076e-2	+1.72907588e-2	A_1/A_3	+7.069070e-2	+7.06907088e-2
A_3/A_5	−56.66337	−56.6633750	A_3/A_5	−14.09139	−14.0913927
A_5/A_7	−149.7652	−149.765237	A_5/A_7	−37.36378	−37.3637874
A_7/A_9	−276.2019	−276.201884	A_7/A_9	−68.97131	−68.9713194
A_9/A_{11}	−435.2224	−435.222381	A_9/A_{11}	−108.7254	−108.725415
			A_{11}/A_{13}	−156.5569	−156.556794
	$Ps_1^1(x,1)_{LC}$	$Ps_1^1(x,1)$		$Ps_1^1(x,2)_{Fl}$	$Ps_1^1(x,2)$
A_0/A_2	−77.01811	−77.0180852	A_0/A_2	−20.814235	−20.8142346
A_2/A_4	−148.5253	−148.525238	A_2/A_4	−38.321262	−38.3212629
A_4/A_6	−258.8156	−258.815563	A_4/A_6	−65.813092	−65.8130912
A_6/A_8	−402.0793	−402.079341	A_6/A_8	−101.59196	−101.591963
			A_8/A_{10}	−145.46764	−145.467638
			A_{10}/A_{12}	−197.38776	−197.387768
			A_{12}/A_{14}	−257.33203	−257.332048
			A_{14}/A_{16}	−325.29091	−325.290929
			A_{16}/A_{18}	−401.25934	−401.259320

V. CONCLUSION

We have found that we can use iterative methods to calculate quickly and accurately the eigenvalues and eigenvectors necessary to solve for the periodic spheroidal functions, using reverse linear interpolation and a variant of Jacobi's method. Our results are in agreement with the tables of Flammer and Stratton. We have investigated the circumstances in which the modified Jacobi's method can be applied to general matrices. The run time of this method goes as $O(n^2)$, with an additional factor of order the logarithm of the calculation precision. We have determined various conditions on the eigenvalues of the matrix in Jacobi's method where the iteration scheme will fail, and we have discussed possible ways to resolve some failure modes. We conclude that using iterative methods to solve eigenvalue and eigenvector problems is an excellent way to avoid inaccuracies due to propagated errors from finite precision by relying on the topology of Hilbert spaces, as well as accelerating computation time, for those iteration matrices where the absolute value of the largest eigenvalue is significantly greater than other eigenvalues.

ACKNOWLEDGMENTS

The authors thank Eric M. Nelson for his assistance with this paper. This work was partially funded by the U.S. Department of Energy.

REFERENCES

1. E. Kreyszig, *Introductory Functional Analysis with Applications* (Wiley, New York, 1978).
2. R. Rosanoff and H. Webel, On the convergence rate of iterative methods for the solution of positive definite linear equations, *Comput. Meth. Appl. Mech. Eng.* **7**, 369 (1976).
3. F. Udwadia, Some convergence results related to the JOR iterative method for symmetric, positive-definite matrices, *Appl. Math. Comput.* **47**, 37 (1992).
4. V. Manoranjan and M. Olmos Gomez, A two-step Jacobi-type iterative method, *Comput. Math. Appl.* **34**, 1 (1997).
5. D. Evans, and N. Missialis, On the preconditioned Jacobi method for solving large linear systems, *Computing* **29**, 167 (1982).
6. M. Benson, J. Krettmann, and M. Wright, Parallel algorithms for the solution of certain large sparse linear systems, *Int. J. Comput. Math.* **16**, 245 (1984).
7. D. Evans, and C. Okeke, The modified preconditioned Jacobi method for iterative solution of linear systems of equations, *Int. J. Comput. Math.* **44**, 53 (1992).
8. B. Codenotti, and P. Favati, New techniques of the solution of linear systems by iterative methods, *Comput. Math. Appl.* **14**, 425 (1987).
9. A. Erdélyi, Ed., *Bateman Manuscript Project: Higher Transcendental Functions* (McGraw–Hill, New York, 1953).
10. J. Meixner and F. W. Schäfke, *Mathieusche Funktionen und Sphäroidfunktionen mit Anwendugen auf physikalishe und technishe Probleme* (Springer-Verlag, Berlin, 1954).
11. J. A. Stratton, P. M. Morse, L. J. Chu, J. D. C. Little, and F. J. Corbató, *Spheroidal Wave Functions* (Technology Press of MIT, Cambridge Wiley, New York, 1956).
12. C. Flammer, *Spheroidal Wave Functions* (Stanford Univ. Press, Stanford, 1957).
13. E. Gianfelice, L. Verolino, L. Palumbo, and V. G. Vaccaro, New methods for treatment of spheroidal functions in the solutions of scalar wave equation, *Nuovo Cimento B* **105**, 771 (1990).
14. S. M. Nesterov and I. A. Skorodumov, Scattering of electromagnetic waves by a metallic disk, *Radioteknika Electronika* **8**, 1351 (1992).
15. L.-W. Li, M.-S. Leong, T.-S. Yeo, P.-S. Kooi, and K.-T. Tan, Computations of spheroidal harmonics with complex arguments: A review with an algorithm, *Phys. Rev. E* **58**, 5 (1998).
16. C.-E. Fröberg, *Introduction to Numerical Analysis* (Addison–Wesley, Reading, MA, 1969).
17. M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions* (Dover, New York, 1972; 1st ed. published by National Bureau of Standards, U.S. Government Printing Office, Washington, DC, 1964).
18. H. S. Wall, *Analytic Theory of Continued Fractions* (Van Nostrand, New York, 1948).