

Lawrence Berkeley National Laboratory

Recent Work

Title

ON THE NUMERICAL SOLUTION OF THE EIGENVALUE PROBLEM OF THE LAPLACE OPERATOR
BY A CAPACITANCE MATRIX METHOD

Permalink

<https://escholarship.org/uc/item/7qf3n342>

Author

Proskurowski, Wlodzimierz.

Publication Date

1976-09-01

0 0 7 0 4 6 0 2 3 9 3

Submitted to SIAM Numerical Analysis

LBL-5396
Preprint c.1

ON THE NUMERICAL SOLUTION OF THE EIGENVALUE
PROBLEM OF THE LAPLACE OPERATOR BY A
CAPACITANCE MATRIX METHOD

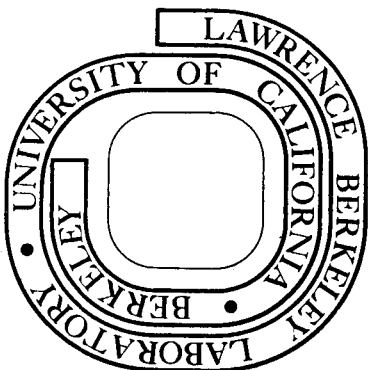
Włodzimierz Proskurowski

September 1976

Prepared for the U. S. Energy Research and
Development Administration under Contract W-7405-ENG-48

For Reference

Not to be taken from this room



LBL-5396

c.1

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.

-iii-

ON THE NUMERICAL SOLUTION OF THE
EIGENVALUE PROBLEM OF THE LAPLACE OPERATOR
BY A CAPACITANCE MATRIX METHOD*

Wlodzimierz Proskurowski

Lawrence Berkeley Laboratory
University of California
Berkeley, California

September 1976

ABSTRACT

The problem of finding several eigenfunctions and eigenvalues of the interior Dirichlet problem for Laplace's equation on arbitrary bounded plane regions is considered. Two fast algorithms are combined: an iterative Block Lanczos method and a capacitance matrix method. The capacitance matrix is generated and factored only once for a given problem. In each iteration of the Block Lanczos method, a discrete Helmholtz equation is solved twice on a rectangle at a cost of the order of $n^2 \log_2 n$ operations where n is the number of mesh points across the rectangle in which the region is imbedded.

*Revised version of TRITA-NA-7609. This work was done with support from the U.S. Energy Research and Development Administration.

1. INTRODUCTION

In spite of its apparent simplicity, the eigenvalue problem for the Laplace operator appears in a great number of applications. There is, therefore, a permanent interest in fast, accurate, and fairly general computational schemes for this problem.

Our method combines two fast algorithms: (1) an iterative Block Lanczos method to compute eigenvalues and eigenvectors of a given operator, and (2) a capacitance matrix method to compute selected eigenvalues and eigenvectors of the Laplace operator on an arbitrary plane bounded region. These two algorithms reflect recent development in their respective fields. The Block Lanczos is due to Underwood [34]. It allows for the simultaneous computation of several eigenvalues and the corresponding eigenvectors at an improved rate of convergence compared to a simple Lanczos method. The choice of a block method also greatly simplifies the treatment of multiple eigenvalues. Our capacitance matrix program is a revision of the one which appeared in Proskurowski and Widlund [25]. Our current version of this program allows for a selective choice of a group of consecutive eigenvalues with the help of a shift. The capacitance matrix is generated and decomposed into triangular factors only once for a given problem and choice of the shift. In each iteration of the Block Lanczos method, the discrete Helmholtz equation is solved at a cost of the order of $n^2 \log_2 n$ operations. The cost of this step is only about twice that of the fast solver on the rectangle in which the region is imbedded.

In Sections 2 and 3 we briefly present the continuous problem and some finite difference schemes. In Sections 4, 5, and 6 we discuss alternative computational techniques and describe the block Lanczos and capacitance matrix methods. In Section 7 results of numerical experiments are given. A listing of the program used in our experiments is given in the original report [24].

Extensive experiments show that our method is suitable for the fast and fairly accurate computation of several eigenvalues and eigenvectors of the Laplace operator on arbitrary bounded plane regions and that it compares favorably with the method recently developed by Kuttler [14] for the same problem, see further discussion in Section 4.

2. STATEMENT OF THE CONTINUOUS PROBLEM

Vibrations of a thin membrane pinned along the edge $\partial\Omega$ of a plane region Ω are governed by the wave equation

$$\Delta U = U_{tt} \quad \text{in } \Omega$$

(2.1)

$$U = 0 \quad \text{on } \partial\Omega.$$

Here Δ is the Laplace operator, $U_{tt} = \partial^2 U / \partial t^2$ and $U = U(x_1, x_2, t)$ represent the deflection of the membrane at the time t . Periodic

motions of the membrane are found by making the Ansatz

$$(2.2) \quad U(x_1, x_2, t) = u(x_1, x_2) \cdot e^{ikt},$$

where k is a real parameter related to the frequency of the vibration.

A substitution of (2.2) into (2.1) shows that u must satisfy the equation

$$(2.3) \quad \begin{aligned} \Delta u + \lambda u &= 0 && \text{in } \Omega \\ u &= 0 && \text{on } \partial\Omega, \end{aligned}$$

where $\lambda = k^2$.

Equation (2.3), the reduced wave equation, arises in a variety of applications.

It is known that there is an infinite sequence of eigenvalues λ_k

$$0 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_k \rightarrow \infty$$

for which Eq. (2.3) has nontrivial solutions $u^{(k)}$. These solutions, the eigenfunctions $u^{(k)}$, can be chosen to be orthonormal and they form a complete set.

The nodal lines of an eigenfunction are the curves along which the membrane remains at rest during the eigenvibration. If several branches of the nodal lines intersect in the interior of a plane region, then they form an equiangular system of rays (see Courant and Hilbert [5]). For further detailed discussion of the eigenvalue problem see Courant and Hilbert [5] and Garabedian [9].

The solution of the eigenvalue problem (2.3) is known explicitly only for certain simple regions for which separation of variables is possible. Briefly, we will present the results for two such regions, a rectangle and a circle.

For a rectangular membrane with sides a and b the eigenvalues are

$$(2.4.1) \quad \lambda^{(m,n)} = \pi^2 \left(\frac{n^2}{a^2} + \frac{m^2}{b^2} \right),$$

and the corresponding normalized eigenfunctions are

$$(2.4.2) \quad u^{(m,n)}(x_1, x_2) = \frac{2}{\sqrt{ab}} \cdot \sin(n\pi x_1/a) \cdot \sin(m\pi x_2/b),$$

where $m, n = 1, 2, \dots$.

These eigenfunctions have nodal lines which are parallel to the coordinate axes. However, in the case of multiple eigenvalues, i.e. when the ratio a/b is rational, many other nodal lines can occur.

For a circular membrane of radius 1 the eigenvalues $\lambda = k^2$ of Eq. (2.3) are the squares of the zeros of the Bessel functions $J_n(kr)$, $n=0,1,\dots$ (a contribution of $-n^2$ is to be added from the trigonometric factors). Each function J_n has infinitely many zeros, which we denote by $k_{n,m}$ ($m = 1,2,\dots$). The eigenfunctions of Eq. (2.3) can be written in the form

$$J_n(k_{n,m}r)(a \cos n\phi + b \sin n\phi).$$

All eigenvalues are multiple, with the exception of those corresponding to $n = 0$. The nodal curves for the eigenfunctions are circles $\rho = \text{const.}$ and radial lines $\phi = \text{const.}$

3. THE FINITE DIFFERENCE EIGENVALUE PROBLEM

In this section we describe the discrete eigenvalue problem of our choice and give a brief review of the literature on the discretization error. The original problem (2.3) is discretized by a finite difference method. The region Ω is first imbedded in a larger region, a rectangle or an infinite parallel strip, which will play a special role in the capacitance matrix method, and a uniform mesh is introduced with the same mesh size h in the two coordinate directions. The set of mesh points is decomposed into three disjoint sets: Ω_h , $\partial\Omega_h$, and $(C\Omega)_h$. The set Ω_h is the set of interior mesh points, i.e., each of its members has all its next neighbors in the open set Ω . The remaining mesh points

in Ω constitute $\partial\Omega_h$, the set of irregular mesh points, while the set $(\Omega)_h$ contains all the remaining, the exterior, mesh points. In addition, we denote by Γ_h the set of points where a mesh line crosses the boundary of the region Ω .

The discrete Laplacian is represented by the five-point difference operator,

$$(3.1) \quad \Delta_h u(x_1, x_2) = \frac{1}{h^2} \left[u(x_1 + h, x_2) + u(x_1 - h, x_2) + u(x_1, x_2 + h) + u(x_1, x_2 - h) - 4u(x_1, x_2) \right],$$

for all points in Ω_h . The related truncation error is $\Delta_h u(x) - \Delta u(x) = O(h^2)$ for $x \in \Omega_h$. For the irregular points we must introduce a formula which also takes the boundary conditions into account. We therefore combine the discrete Laplacian with an interpolation formula. Suppose that the point $x \in \partial\Omega_h$ has its eastern and southern neighbors in $(\Omega)_h$. Then, by applying linear interpolation, we obtain

$$(3.2) \quad \Delta_{h,1} u(x_1, x_2) = \frac{1}{h^2} \left[u(x_1 - h, x_2) + u(x_1, x_2 + h) + \frac{1}{\delta_1} u_e + \frac{1}{\delta_2} u_s - \left(2 + \frac{1}{\delta_1} + \frac{1}{\delta_2} \right) u(x_1, x_2) \right],$$

where $u_e = u(x_1 + \delta_1 h, x_2)$ and $u_s = u(x_1, x_2 - \delta_2 h)$ are the Dirichlet data on Γ_h . Similarly, by applying quadratic interpolation the

Shortley-Weller approximation, $\Delta_{h,2}$, is obtained, see Forsythe and Wasow [7]. We use the approximation (3.2) throughout, because it gives rise to a symmetric discrete operator. We note that most of the error estimates for the operator $\Delta_{h,2}$ also hold for $\Delta_{h,1}$.

The problem analogous to (2.3) is to find the eigenvalues $\lambda_h^{(k)}$ and corresponding mesh eigenvectors $u_h^{(k)}$ which satisfy

$$\begin{aligned} \Delta_{h,1} u_h^{(k)} + \lambda_h^{(k)} u_h^{(k)} &= 0 && \text{in } \Omega_h \cup \delta\Omega_h \\ u_h^{(k)} &= 0 && \text{on } \Gamma_h. \end{aligned} \quad (3.3)$$

Kuttler [15] has given a very elegant and general result which, when applied to the operators $\Delta_{h,1}$ and $\Delta_{h,2}$, shows that if $u^{(k)} \in C^4(\bar{\Omega})$ then there are positive constants c_k and h_k such that

$$\begin{aligned} \left| \lambda_h^{(k)} - \lambda^{(k)} \right| &\leq c_k h^2 \quad \text{and} \\ \max_{\Omega_h \cup \delta\Omega_h} |u_h^{(k)} - u^{(k)}| &\leq c_k h^2 \quad \text{for } h < h_k. \end{aligned} \quad (3.4)$$

Bramble and Hubbard [2] obtained similar results for the symmetric operator $\Delta_{h,1}$. They also considered plane regions whose boundaries are composed of analytic arcs with no reentrant cusps. If π/α , $\alpha > 1/2$, is the largest interior angle, then for every $\epsilon > 0$ there exists $K(\epsilon)$ such that

$$(3.5) \quad |\lambda_h^{(k)} - \lambda^{(k)}| \leq K(\epsilon) (h^2 + h^{2\alpha-\epsilon}) \quad \text{and}$$

$$\max_{\Omega_h \cup \partial\Omega_h} |u_h^{(k)} - u^{(k)}| \leq K(\epsilon) (h^2 + h^{\alpha-\epsilon}) .$$

For certain polygonal regions with reentrant corners and $\alpha = 4/5, 2/3, 4/7$, or $1/2$, Moler [17] found that there exist constants $c_1, c_2 > 0$ such that

$$(3.6) \quad -c_1 h^2 \leq \lambda_h^{(k)} - \lambda^{(k)} \leq c_2 h^{2\alpha} .$$

Moler also showed that the eigenvalues of $\Delta_{h,1}$, for general regions, fail to have an asymptotic expansion which precludes the use of Richardson extrapolation to improve the accuracy. For a more detailed discussion, see in particular Moler [17], Kuttler [15] and Bramble and Hubbard [2], and the references given therein. We note that the higher eigenvalues are progressively more difficult to compute since their eigenfunctions become increasingly oscillatory.

It would be of interest to develop a higher order symmetric finite difference scheme for increased efficiency. For asymmetric highly accurate methods for Poisson's equation, see Pereyra, Proskurowski, and Widlund [22].

4. SOME ALTERNATIVE COMPUTATIONAL TECHNIQUES

In this section we will discuss briefly some alternative techniques for the eigenvalue problem (2.3).

A collocation technique was developed by Fox, Henrici, and Moler [8] and has proved successful in particular for certain regions with reentrant corners; see also Ryder and Sanderson [30]. To achieve an accurate approximation it uses Bessel functions and trigonometric polynomials which represent the solution very well close to the corners. This method also produces remarkably tight bounds for eigenvalues.

Another special technique is due to Golub, Jennings, and Yang [11]. An eigenvalue problem for a matrix which is a low-rank modification of a problem which is easily solvable for any given right-hand side, can be converted into a much smaller dense nonlinear eigenvalue problem. We originally intended to base our capacitance matrix technique for the eigenvalue problem on a similar idea. This approach was suggested to us by Professors Golub and Widlund when they visited Stockholm in the spring of 1973. We will now briefly describe this idea.

Consider the eigenvalue problem for the Laplace equation (2.3) as a homogenous Helmholtz equation, $\Delta u + \lambda u = 0$. The Neumann and Dirichlet problems can be reduced to Fredholm integral equations of the second kind. The potential theory for the Laplace equation, described in

Section 2 of Proskurowski and Widlund [26] generalizes, in a straight forward way, to Helmholtz's equation with a constant coefficient zero-order term. The resulting integral equation is approximated by a capacitance matrix equation

$$(4.1) \quad C(\lambda)\rho = 0$$

when a corresponding theory for the discrete problem is developed. The problem (4.1) is a nonlinear eigenvalue problem. The large eigenvalue problem (3.3) is thus reduced to solving the much smaller problem (4.1), i.e., to finding those λ for which $\det(C(\lambda)) = 0$. For this purpose we can use an efficient algorithm for finding a zero of a function of one variable, for example, the one due to Bus and Dekker [3]. The determinant of $C(\lambda)$ is found by Gaussian elimination. The eigenvector of (4.1) corresponding to an eigenvalue λ^* , already computed, can easily be obtained by the inverse iteration method by using the triangular factors of the capacitance matrix $C(\lambda^*)$ found while computing its determinant. The eigenvector for the original problem (3.3) is then found by using the fast solver.

After a series of numerical experiments, with both the Neumann and Dirichlet problems, this method was abandoned. Our reasons were that the computational effort is proportional to the product of n^3 and the number of iterations, and that even for small values of n this effort is greater than what is needed for our present method. Initial bounds

on the discrete eigenvalues, i.e. for a specific mesh, are also required, and the efficiency depends heavily on the accuracy of this information. CPU-time of one iteration on IBM 360/75 was roughly 2 sec for the step size $h=1/12$. One might expect to need at least 10 iterations even if a fairly good initial guess is used. This compares unfavorably with the results in Tables 3 to 6. We also found this method unreliable, as there is a possibility of computing eigenvalues of the complementary problem in $(C\Omega)_h$ instead. (See also Kuttler [14]). In conclusion, this method is generally more costly, and a user without a good preliminary knowledge of the desired eigenvalues might find it confusing.

We later found that the same method had been developed independently by Kuttler [14], who obtained results similar to ours. Nevertheless, our conclusions on the usefulness of the method differ substantially.

The matrix eigenvalue problem

$$(4.2) \quad Ax = \lambda x$$

corresponding to Eq. (3.3) has a matrix A which is large, sparse, and symmetric. We will now discuss the use of certain standard techniques. The well-known transformation methods, such as those of Givens or Householder, destroy the sparseness and are therefore inefficient. We also note that we are interested primarily in the first few eigenvalues and corresponding eigenvectors of the problem (4.2).

The inverse iteration method, see Wilkinson [36,38], is widely used for the calculation of eigenvectors from computed eigenvalues, but also for the calculation of the eigenvalues themselves. We note that this method requires the solution of a large sparse system of equations.

An efficient variant of the power method is the simultaneous (block) iteration method developed by Rutishauser [29,37]. Here s eigenvalues are computed simultaneously by carrying s approximate eigenvectors in the iterations. Convergence to the i -th largest eigenvalues μ_i occurs at the improved rate μ_i/μ_{s+1} . Variants of this method for unsymmetric matrices have been developed by Clint and Jennings [5] and Stewart [33].

Other iterative algorithms, such as the conjugate gradient methods and SOR, have been considered by Ruhe [27,28].

Another technique has been developed by Peters and Wilkinson [23]. It exploits Sylvester's inertia theorem: the number of eigenvalues of $(A - kI)$ less than k equals the number of negative pivots in Gaussian elimination. Accurate determination of any eigenvalue of A can be obtained by using a bisection method. This method has a very good reputation as one of the most competitive if the bandwidth is not very large; see Strang and Fix [32]. A recent variant of this method is given by Anderson [1]. He shows that the pure secant method

converges monotonically to the extremal eigenvalues. He also gives a workable deflation strategy. Again, the limiting factor is the bandwidth.

For a discussion of Lanczos methods see the next section.

Of the methods discussed here we feel that the one due to Rutishauser would have been the most viable alternative to the method of our choice.

5. THE ITERATIVE BLOCK LANCZOS METHOD

In 1950 Lanczos [16] published a method for the reduction of a general square matrix to tridiagonal form. His method has been modified into a practical numerical procedure for symmetric problems only in recent years and has found many applications. For a detailed theory of this method see Golub [10], Golub, Underwood, and Wilkinson [12], Paige [18,19,20] and Underwood [34].

Let A be an $n \times n$ symmetric matrix. For a given initial n -vector v_1 , $v_1^T v_1 = 1$ and a given $m \leq n$ the Lanczos algorithm finds a sequence of scalars α_i, β_i and orthonormal vectors v_i , $i = 1, 2, \dots, m$ as follows:

1. $i = 1$, v_1 given.

2. Compute $y_i = Av_i$ and $\alpha_i = v_i^T y_i$.

3. If $i = m$ stop.

4. Compute

$$z_{i+1} = \begin{cases} y_i - \alpha_i v_i & i = 1 \\ y_i - \alpha_i v_i - \beta_i v_{i-1} & \text{if } i > 1 \end{cases}$$

5. Compute $\beta_{i+1} = \sqrt{z_{i+1}^T z_{i+1}}$

6. If $\beta_{i+1} = 0$ stop.

7. Compute $v_{i+1} = z_{i+1} / \beta_{i+1}$.

8. $i = i+1$, go to 2.

In m steps this algorithm produces an orthonormal $n \times m$ matrix $v = (v_1, v_2, \dots, v_m)$ and an $m \times m$ tridiagonal matrix T such that

$$(5.1) \quad AV = VT + (0, \dots, 0, z_{m+1}) .$$

Here

$$T = \begin{bmatrix} \alpha_1 & \beta_1 & & 0 \\ \beta_2 & \alpha_2 & \beta_2 & \\ & \cdot & \cdot & \cdot \\ & & & \beta_m & \alpha_m \end{bmatrix} .$$

Our discussion can easily be modified to account for the possibility that the algorithm might stop early.

Denote by \bar{A} the restriction of A to the subspace spanned by the first m elements of the Krylov sequence $\{v_1, Av_1, A^2v_1, \dots\}$. A basic property of the Lanczos method is that even if m is much less than n a few of the extreme eigenvalues of \bar{A} will give a good approximation to the extreme eigenvalues of A ; see Kaniel [13] and Paige [18]. The algorithm is closely related to the conjugate gradient method; see, for example, Paige and Saunders [21]. The matrix A is only required in terms of a matrix-vector multiplication in each step. The matrix A is thus not transformed which makes the method very suitable for large sparse problems.

Underwood [34] and Cullum and Donath [6] have extended the Lanczos algorithm to work with blocks of vectors $V^{(p)}$, of order $n \times p$, $p > 1$, instead of a single vector v . This often results in less work overall and allows us to compute multiple eigenvalues and eigenvectors at the same time. The main idea behind the block Lanczos method is thus similar to the modification of the power method due to Rutishauser [29, 37] which led to the simultaneous iteration method.

In our experiments we have used the block Lanczos routine developed by Underwood [34]. Our choice was motivated by example 7 in [34], which indicated that his program should be preferred to that of Rutishauser

for an elliptic difference scheme similar to ours.

We will now discuss some of the details of Underwood's program. It is well known that Lanczos methods can suffer from numerical instability as a result of a loss of orthogonality of the Lanczos vectors. Underwood has therefore chosen to reorthogonalize the vectors with respect to those of the previous blocks. This adds to the expense of the algorithm, but this expense is controlled since the method is periodically restarted using the best available approximate eigenvectors as the new initial approximation. All the vectors generated since the previous restart are retained in fast storage. The convergence of the eigenvectors is tested, and once an eigenvector is accepted, we work with the operator restricted to the subspace orthogonal to this vector.

An array of order $n \times q$ is used for the storage of the Lanczos vectors. The algorithm can produce up to $q-1$ eigenvectors and eigenvalues, where q is a value specified by the user. The block size, p , initially also set by the user, is automatically changed by the program to attain an optimal effect. Most often p is smaller than the desired number of eigenvalues. The $n \times q$ array will at any time contain m vectors already accepted as sufficiently accurate. The remaining $q-m$ columns are used for the iteration to find the rest of the required eigenvectors. We can thus take the integer part of $(q-m)/p$ iteration steps before restarting.

The block Lanczos method provides estimates of the residuals $\rho_i = Av_i - \lambda v_i$ while computing the eigenvalues and eigenvectors. An eigenvector is accepted if the norm of the residual is less than a prescribed absolute tolerance.

Underwood's program was primarily designed to be used for the computation of a part of the eigensystem of a sparse matrix A without the benefit of the acceleration obtained by working with $(A - kI)^{-1}$, where k is a shift parameter. He was primarily interested in the eigenvalues at one extreme end of the spectrum, since in most applications the eigenvalues at the other end are of no interest. His program therefore suppresses the information on the eigenvectors at the other end of the spectrum, which is contained in the space spanned by the Lanczos vectors. Since we work with the inverse of discrete Helmholtz operators which can have eigenvalues of both signs, we are often interested in the eigenvalues at both extreme ends of the spectrum. Therefore, in our application this feature of Underwood's program leads to a certain amount of waste; see also Sections 6 and 7.

6. THE CAPACITANCE MATRIX METHOD FOR HELMHOLTZ'S EQUATION

Our method of solving Helmholtz's equation on a bounded plane region is developed in Proskurowski and Widlund [26] and also analyzed in Shieh [31]. For a thorough discussion of capacitance matrix methods we refer to these papers. In this section we will only discuss our

method very briefly, concentrating on the modifications required by the current context.

The fast Helmholtz solver subroutine of our capacitance matrix program gives the solution of Helmholtz's equation on part of an infinite strip. One call of this subroutine requires the order of $m \log_2 n$ operations, where m and n are the number of mesh points along and across the strip, respectively.

The capacitance matrix solver consists of two main parts. Only the second of these has to be repeated when the data of the problem is changed.

In the first part the capacitance matrix C is generated at an expense of the order of p^2 operations and one call of the fast solver on the strip. Here p denotes the number of irregular mesh points and is thus of the order n . The original program, [26], has two options. We can either solve the capacitance matrix equation by a conjugate gradient method or use Gaussian elimination. Since the same problem will be solved repeatedly we have chosen the Gaussian elimination option. We thus once incur an expense of the order of p^3 operations when factoring the matrix C .

In the second main part of the program, we use the fast solver twice and solve the capacitance matrix equation by using the triangular

factors previously computed. This expense is proportional to $m \log_2 n$ and to p^2 and constitutes the dominating term in the cost of one iteration of the block Lanczos method.

The block Lanczos algorithm uses the capacitance matrix solver as a subroutine which computes a matrix vector product $y = Hx$ for a given vector x . This solver returns the solution y of a linear system $Ay = x$, i.e., $y = Hx$, $H = A^{-1}$, for a given x .

The block Lanczos algorithm as implemented by Underwood [34] computes only the smallest eigenvalues of H and not those of maximum modulus. Therefore, in order to compute the least eigenvalues of A we must reverse the sign and make

$$(6.1) \quad y = -A^{-1}x.$$

If it is desirable to obtain a group of eigenvalues all larger than a given value λ_0 , we use a shift in this algorithm. This means that we are computing approximations to the eigenvalues of Helmholtz's equation $(\Delta + \lambda_0)u + \lambda u = 0$. Thus

$$(6.2) \quad y = -(A + \lambda_0 I)^{-1}x.$$

Similarly, if a group of eigenvalues all less than some λ_0 is required, we use

$$(6.3) \quad y = (A + \lambda_0 I)^{-1} x .$$

The capacitance matrix program produces solutions on an extended rectangular region, while the vectors x and y of (6.1) - (6.3) are defined only on the set of mesh points inside the given region Ω . Therefore, at each block Lanczos iteration the useless values at the exterior mesh points are suppressed. Information about the boundary of the region Ω is read in through a user-supplied subroutine. Compact data is then produced which allows for the identification of the mesh points inside the region Ω .

Underwood's block Lanczos program [34] has been used without changes.

7. NUMERICAL EXPERIMENTS

In this section we will report on results from a series of numerical experiments which were carried out on the IBM 360/75 computer at the Royal Institute of Technology in Stockholm. In our experiments we have used the program listed in the original report [24].

In an initial test of the performance of the capacitance matrix program and the block Lanczos routine, we used the latter together with

a Gaussian elimination routine on a problem with a known spectrum. We chose the discrete Laplacian on a square and a coarse mesh size, $h = 1/3$. The results in Table 1, obtained with the capacitance matrix solver and with the Gaussian elimination routine, agreed to 14 decimals.

In Table 2 we have collected results of the simultaneous computation of the eigenvalues for circular regions with different mesh sizes. These results demonstrate that Richardson extrapolation works for this region. The extrapolated values are compared with the exact eigenvalues; see Watson [35]. By Richardson extrapolation we gain almost two decimals in accuracy. The relative discretization error after the extrapolation is of order 10^{-3} to 10^{-4} and grows slowly with the index of the eigenvalues. Within this range of mesh sizes, $h \geq 1/12$, the discretization error is larger than the error from the block Lanczos program if the tolerance is $\epsilon = 10^{-3}$. This indicates that this choice of ϵ is reasonable. We have therefore used this value routinely.

In Tables 3 and 4 we report on the CPU-time for our program when run on an IBM 360/75 computer using a FORTRAN H level 2.1 optimizing compiler. The results in Table 3 refer to the smallest eigenvalue of Helmholtz's equation computed individually, and Table 4 to a group of 15 eigenvalues computed simultaneously. We note that the rectangle in which the region is imbedded is unnecessarily large, 73.5% of the mesh points are exterior and a different choice of step size would

have been more economical. The largest eigenvalues of the inverse of the Laplace operator are better separated than the smaller. Therefore, when we use a shift λ_0 , the convergence is slower, in accordance with the theoretical estimate; see Underwood [34]. It is therefore difficult to give a very precise comparison between the efficiency of the block Lanczos and simple Lanczos methods. Nevertheless, the gain is substantial, and we strongly recommend the use of the block Lanczos method.

The share of the time used to solve Helmholtz's equation with the capacitance matrix solver dominates these computations. In the experiments where 15 eigenvalues were computed simultaneously it constitutes over 2/3 of the total computing time, while the time for the generation and factorization of the capacitance matrix is about 2.0 to 2.5% of the total for this choice of mesh. We can therefore ignore this contribution in an estimate of the execution time. We conjecture that the time grows as $\text{const} \cdot n^2 \log_2 n$. Here n is the number of mesh points across the rectangle in which the region is imbedded. When we test this conjecture by using the values from Table 4 we find that the total execution time grows by a factor 3.65 when n is doubled. This shows that we could do problems with finer mesh sizes at a reasonable cost. In Table 5 we present results of experiments for an unsymmetric, nonconvex, palette-like region with three mesh sizes $h = 1/6$, $1/12$, and $1/24$. For this problem second order accuracy is apparent only for the first eigenvalue.

In Table 6 we report on results obtained on circular regions with a slit. We can solve such problems with our method without any special treatment, and with a computational effort which is close to that for the other regions; see Tables 4 and 5. It is interesting to note that the eigenvalues corresponding to the eigenfunctions which have a zero value along the slit agree very closely with certain eigenvalues for the circular region without a slit; see Table 4. In general, we must expect a slow convergence of the discrete eigensystem for a problem of this nature.

In certain applications we might be interested in a higher accuracy of the eigenvalues and eigenvectors than what we obtained in our experiments, or in decreasing the cost. We have already noted, in Section 4, that it would be interesting to develop finite difference schemes which are more accurate and which still give rise to symmetric matrices. Some speed could be gained by modifying our current fast Poisson solver on the strip; see Table 7 in Proskurowski and Widlund [26]. A block Lanczos method without reorthogonalization might be preferable, since operations are saved and previous blocks can be stored on secondary storage devices. The previous blocks are then accessed only when the eigenvalues have converged. Alternatively, the same information could be recomputed from the same initial block in order to obtain the eigenvectors. Rutishauser's method might also prove a better choice than our Lanczos method. The reliability and rates of convergence of these methods will of course differ, and only extensive numerical experiments could settle these questions.

ACKNOWLEDGMENTS

I would like to thank: Professor Olof Widlund for his consistent support and contribution throughout my work, Professor Gene Golub for his share in initiating this project, and Professor Germund Dahlquist for several valuable suggestions. I would also like to thank Dr. Richard Underwood, formerly of the computer science department at Stanford University, for his excellent block Lanczos program.

REFERENCES

- [1] N. Anderson, *On computing eigenvalues of matrices with real eigenvalues by the secant method*, TRITA-NA-7513 Report, 1975, RIT Stockholm.
- [2] J.H. Bramble and B.E. Hubbard, *Effects of boundary regularity on the discretization error in the fixed membrane eigenvalue problem*, SIAM J. Numer. Anal., 5(1968), pp. 835-863.
- [3] J.C.P. Bus and T.J. Dekker, *Two efficient algorithms with guaranteed convergence for finding a zero of a function*, Report NW 13/74, Mathematisch Centrum, Amsterdam, 1974.
- [4] M. Clint and A. Jennings, *A simultaneous iteration method for the unsymmetric eigenvalue problem*, J. Inst. Math. Applics., 8 (1971), pp. 111-121.

- [5] R. Courant and D. Hilbert, *Methods of Mathematical Physics, Vol. 1*, Interscience, 1953.
- [6] J. Cullum and W.E. Donath, *A block generalization of the symmetric S-step Lanczos algorithm*, Report #RC 4845 (#21570), IBM Thomas J. Watson Research Center, Yorktown Heights, New York, May 1974.
- [7] G.E. Forsythe and W.R. Wasow, *Finite-Difference Methods for Partial Differential Equations*, Wiley, New York, 1960.
- [8] L. Fox, P. Henrici, and C. Moler, *Approximations and bounds for eigenvalues of elliptic operations*, *SIAM J. Numer. Anal.*, 4 (1967), pp. 89-102.
- [9] P.R. Garabedian, *Partial Differential Equations*, Wiley, New York, 1964.
- [10] G.H. Golub, *Some uses of the Lanczos algorithm in numerical linear algebra*, Stanford Report CS-72-302, Aug. 1972.
- [11] G.H. Golub, L. Jennings, and W.H. Yang, *Waves in periodically structured media*, *J. Comp. Phys.*, 17 (1975), pp. 1-9.
- [12] G.H. Golub, R. Underwood, and J.H. Wilkinson, *The Lanczos algorithm for the symmetric $Ax = \lambda Bx$ problem*, Stanford Report CS-72-270, March 1972.

- [13] S. Kaniel, *Estimates for some computational techniques in linear algebra*, Math. Comp., 20 (1966), pp. 369-378.

- [14] J.R. Kuttler, *Direct methods for computing eigenvalues of the finite-difference Laplacian*, SIAM J. Numer. Anal., 11 (1974), pp. 732-740.

- [15] J.R. Kuttler, *Finite-difference approximations for eigenvalues of uniformly elliptic operators*, SIAM J. Numer. Anal., 7 (1970), pp. 206-232.

- [16] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Nat. Bur. Standards, 45 (1950), pp. 255-283.

- [17] C.B. Moler, *Finite difference methods for the eigenvalues of Laplace's operator*, Stanford Report CS-22, 1965.

- [18] C.C. Paige, *The computation of eigenvalues and eigenvectors of very large sparse matrices*, Ph.D. dissertation, The University of London, 1971.

- [19] C.C. Paige, *Computational variants of the Lanczos method for the eigenproblem*, J. Inst. Maths. Applics., 10 (1972), pp. 373-381.

- [20] C.C. Paige, *Error analysis of the Lanczos algorithm for tridiagonalization of a symmetric matrix*, J. Inst. Math. Appl., 1976, to appear.

- [21] C.C. Paige and M.A. Saunders, *Solutions of sparse indefinite systems of equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617-629.
- [22] V. Pereyra, W. Proskurowski, and O. Widlund, *High order fast Laplace solver for the Dirichlet problem on general regions*, Math. Comp., Jan. 1977, to appear.
- [23] G. Peters and J.H. Wilkinson, *Eigenvalues of $Ax = \lambda Bx$ with band symmetric A and B* , Computer J., 12 (1969), pp. 398-404.
- [24] W. Proskurowski, *On the numerical solution of the eigenvalue problem of the Laplace operator by the capacitance matrix method*, TRITA-NA-7609 Report, RIT Stockholm.
- [25] W. Proskurowski and O. Widlund, *On the numerical solution of Helmholtz's equation by the capacitance matrix method*, ERDA Report COO-3077-99, New York University, Nov. 1975.
- [26] W. Proskurowski and O. Widlund, *On the numerical solution of Helmholtz's equation by the capacitance matrix method*, Math. Comp. 30 (1976), pp. 433-468.
- [27] A. Ruhe, *Iterative eigenvalue algorithms for large symmetric matrices*, in L. Collatz (ed.) *Numerische Behandlung Eigenwertaufgaben*, Birkhäuser, 1974.

- [28] A. Ruhe, *Iterative eigenvalue algorithms based on convergent splittings*, J. Comp. Phys., 19 (1975), pp. 110-120.
- [29] H. Rutishauser, *Simultaneous iteration method for symmetric matrices*, Numer. Math., 16 (1970), pp. 205-223.
- [30] B.G. Ryder and J.G. Sanderson, *A program for computing the eigenvalues of Laplace's equation*, CST Report No. 20, Bell Laboratories, July 1974.
- [31] A. Shieh, *Fast Poisson solver on nonrectangular domains*, Ph.D. thesis, New York University, June 1976.
- [32] G. Strang and G.J. Fix, *An Analysis of the Finite Element Method*, Prentice Hall, 1973.
- [33] G.W. Stewart, *Simultaneous iteration for computing invariant subspaces of non-Hermitian matrices*, Numer. Math., 25 (1976), pp. 123-136.
- [34] R. Underwood, *An iterative block Lanczos method for the solution of large sparse symmetric eigenproblems*, Stanford Report CS-496, May 1975.

- [35] G.N. Watson, *A treatise on the theory of Bessel functions*, Cambridge Univ. Press, 1952.
- [36] J.H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.
- [37] J.H. Wilkinson and C. Reinsch, *Handbook for Automatic Computation, Vol. II, Linear Algebra, Part 2*, Springer-Verlag, New York, and Heidelberg, Berlin, 1971.
- [38] J.H. Wilkinson, *Inverse iteration in theory and in practice*, Inst. Naz. di Alta Mathem. *Symposia Mathematica*, 10 (1972), pp. 361-379.

Table 1. The accuracy of six eigenvalues of the discrete Laplacian on a square region with a side 2 and the step size $h=1/3$ obtained by the block Lanczos method.

No.	Discrete eigenvalues λ_h		Norms of residuals
	computed	exact	
1	4.8230854638	4.8230854638	$2.49 \cdot 10^{-5}$
2	11.4115427322	} 11.4115427319	$2.75 \cdot 10^{-4}$
3	11.4115427319		$2.07 \cdot 10^{-5}$
4	18.0000041635	18.0000000000	$3.98 \cdot 10^{-4}$
5	20.4116831420	} 20.4115427319	$1.24 \cdot 10^{-3}$
6	20.4115387031		$1.38 \cdot 10^{-3}$

Table 2. Eigenvalues of the discrete Laplacian on the unit circle with the step size h equal to $1/3$, $1/6$, and $1/12$. Extrapolated values are compared with the exact eigenvalues.

No. Mode	Eigen-values λ_h	Extra-polated eigenv. $\lambda_h^{(1)}$	Exact eigenv. λ	Absolute error	Relative error
1 (0,1)	5.5845 5.7351 5.7713	5.7853 5.7834	5.7832	$2 \cdot 10^{-4}$	$3.46 \cdot 10^{-5}$
2,3 (1,1)	13.004 14.336 14.596	14.779 14.681	14.682	$1 \cdot 10^{-3}$	$6.81 \cdot 10^{-5}$
4,5 (2,1)	20.997 24.980 26.024	26.308 26.372	26.375	$3 \cdot 10^{-3}$	$1.14 \cdot 10^{-4}$
6 (0,2)	24.531 28.928 30.081	30.394 30.465	30.471	$6 \cdot 10^{-3}$	$1.97 \cdot 10^{-4}$
7,8 (3,1)	- 37.959 40.009	40.698	40.706	$8 \cdot 10^{-3}$	$1.97 \cdot 10^{-4}$
9,10 (1,2)	- 45.274 48.195	49.169	49.218	$49 \cdot 10^{-3}$	$9.96 \cdot 10^{-4}$
11,12 (4,1)	- 52.283 56.230	57.546	57.583	$37 \cdot 10^{-3}$	$6.43 \cdot 10^{-4}$
13,14 (2,2)	- 62.294 68.644	70.761	70.850	$89 \cdot 10^{-3}$	$1.26 \cdot 10^{-3}$
15 (0,3)	- 63.628 75.512	75.473	74.887	.586	$7.83 \cdot 10^{-3}$

Table 3. CPU-time in seconds for a problem on a circular domain on an IBM 360/75. Only one eigenvalue was computed at a time. The shifts were chosen in order to find $\lambda_h^{(1)}$ and $\lambda_h^{(6)}$ respectively.

Step size h	1/6		1/12	
Number of mesh points in the rectangle imbedding the region, $n \times m$	16×16		32×32	
Number of irregular mesh points p	32		64	
Number of interior mesh points N	109		437	
The shift CON	0	-28	0	-28
Number of block Lanczos iterations	6	9	9	13
Number of calls of the capacitance matrix solver	10	13	15	19
Preprocessing time t_p	0.38	0.38	1.84	1.84
Time of the block Lanczos iterations t_L	0.25	0.26	0.68	1.00
Time for the capacitance matrix solver t_c	1.21	1.90	6.28	10.72
Total execution time t	1.84	2.54	8.80	12.56
t_c/t	0.658	0.748	0.714	0.791
t_p/t	0.207	0.150	0.209	0.137

Table 4. CPU-time in seconds for a problem on a circular domain on an IBM 360/75. 15 eigenvalues were simultaneously computed ($q=20$, $p=5$).

Step size h	1/6	1/12
Number of mesh points in the rectangle imbedding the region, $n \times m$	16 \times 16	32 \times 32
Number of irregular mesh points p	32	64
Number of interior mesh points N	109	437
Number of block Lanczos iterations	76	84
Number of calls of the capacitance matrix solver N_0	101	109
Preprocessing time t_p	0.42	1.84
Time of the block Lanczos iterations t_L	6.40	20.74
Time for the capacitance matrix solver t_c	13.43	51.30
Total execution time t	20.25	73.88
t_c/t	0.663	0.694
t_p/t	0.0207	0.0249
t_c/N_0	0.133	0.471

Table 5. Eigenvalues and CPU-time in seconds for an unsymmetric non-convex, palette-like region on an IBM 360/75. Six eigenvalues were simultaneously computed ($q=9$, $p=3$).

Step size h	1/6	1/12	1/24
Number of mesh points in the rectangle imbedding the region, $n \times m$	16×16	32×32	64×64
Number of irregular mesh points p	27	56	117
Number of interior mesh points N	68	272	1107
Density of imbedding $N/n \times m$	0.265	0.266	0.270
λ_1^h	10.7068	10.8681	10.9138
λ_2^h	22.602	22.878	23.497
λ_3^h	27.59	28.08	28.91
λ_4^h	36.53	37.88	38.62
λ_5^h	47.30	50.12	51.16
λ_6^h	49.52	53.99	55.11
Number of calls of the capacitance matrix solver N_0	48	52	54
Time for the capacitance matrix solver t_c	6.41	25.77	104.58
Total execution time t	7.61	30.87	126.86
t_c/t	0.84	0.83	0.82
t_c/N_0	0.134	0.496	1.94

Table 6. Eigenvalues and CPU-time in seconds for a circular region with a slit on an IBM 360/75. Six eigenvalues were simultaneously computed ($q = 9$, $p = 3$).

Step size	1/6	1/12
Number of mesh points in the rectangle imbedding the region, $n \times m$	16 \times 16	32 \times 32
Number of irregular mesh points p	42	86
Number of interior mesh points N	103	425
λ_1^h	10.310	10.127
λ_2^h	14.336	14.595
λ_3^h	19.505	20.020
λ_4^h	25.514	26.176
λ_5^h	31.44	32.84
λ_6^h	37.96	40.01
Number of calls on the capacitance matrix solver N_0	54	58
Time for the capacitance matrix solver t_c	7.18	32.28
Total execution time t	10.22	42.99
t_c/t	0.70	0.75

This report was done with support from the United States Energy Research and Development Administration. Any conclusions or opinions expressed in this report represent solely those of the author(s) and not necessarily those of The Regents of the University of California, the Lawrence Berkeley Laboratory or the United States Energy Research and Development Administration.

TECHNICAL INFORMATION DIVISION
LAWRENCE BERKELEY LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720