Report 4710

AL SHIP RESEARCH AND DEVELOPMENT CENTER

Bethesda, Maryland 20034

[redacted]

THE EIGENSYSTEM CAPABILITY OF FOUR SUBROUTINE LIBRARIES AVAILABLE AT DTNSRDC

Donald A. Gignac

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED.

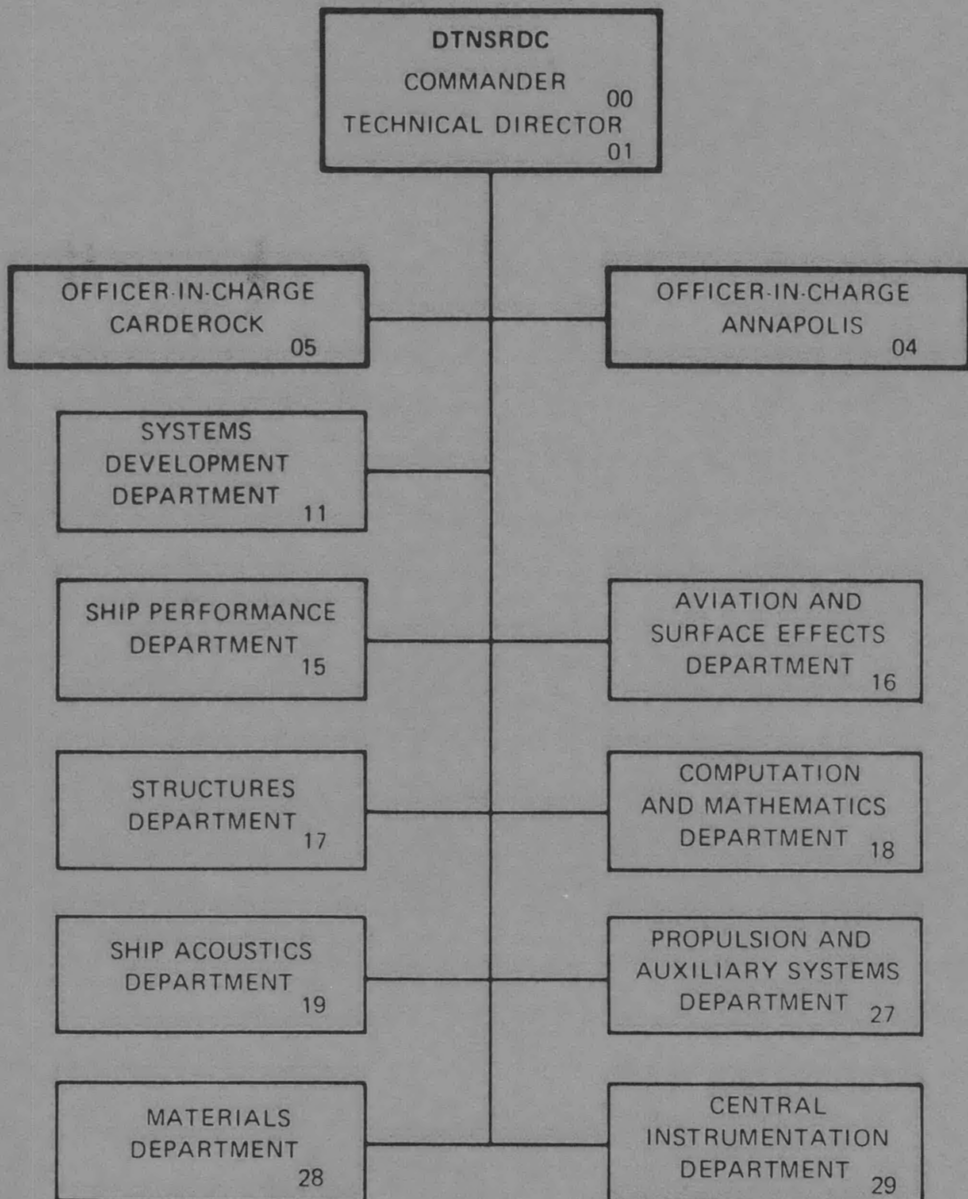
COMPUTATION AND MATHEMATICS DEPARTMENT
RESEARCH AND DEVELOPMENT REPORT

July 1975

Report 4710

THE EIGENSYSTEM CAPABILITY OF FOUR SUBROUTINE LIBRARIES
AVAILABLE AT NSRDC

MAJOR DTNSRDC ORGANIZATIONAL COMPONENTS



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 4710	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) THE EIGENSYSTEM CAPABILITY OF FOUR SUBROUTINE LIBRARIES AVAILABLE AT DTNSRDC		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Donald A. Gignac		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS DAVID TAYLOR NAVAL SHIP RESEARCH & DEVELOPMENT CENTER BETHESDA, MARYLAND 20084		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61153N, R01403, SR 014 03 01, Work Unit 1-1844-004
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE July 1975 13. NUMBER OF PAGES 26
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Matrices
Eigenvalues
Eigenvectors
Subroutine Libraries
Eigensystems

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The need to compute eigenvalues and eigenvectors arises frequently in applied work in the Navy's varied research program. For a given eigensystem problem several algorithms (methods) may be considered and for a given algorithm several Fortran subroutine implementations may be available to choose from. This report offers help in choosing among the options available at DTNSRDC for the solution of eigensystem problems: the MATH SCIENCE LIBRARY

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-014-6601 |

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Item 20. (cont.)

(Control Data Corporation), the IMSL LIBRARY (International Mathematical & Statistical Libraries, Inc.), the EISPACK LIBRARY (Argonne National Laboratory), and the ARL LINEAR ALGEBRA LIBRARY (Aerospace Research Laboratories, Air Force Systems Command). The report reviews background material and makes recommendations.

TABLE OF CONTENTS

INTRODUCTION

The need to compute eigenvalues and eigenvectors arises frequently in applied work in the Navy's varied research program. Eigenvalues are also referred to in the literature as characteristic values, latent roots, and proper values with similar terminology for eigenvectors. However, the term 'eigenvalue', an unfortunate hybrid of the German "eigen" (proper) and the English "value", is by far the most widely used terminology at the present time.

If A is any real or complex matrix of order n, the eigenvalues of A are those real or complex scalars (numbers) λ which satisfy

$$AX = \lambda X \quad \text{or} \quad (A - \lambda I)X = 0 \tag{1}$$

for some real or complex vector $X \neq 0$. X is said to be an eigenvector for λ with respect to the matrix A. The solution of Equation (1) represents the standard linear eigensystem problem.

The following matrix equations arise if B and C are also real or complex matrices of order n:

$$(A - \lambda B)X = 0 \tag{2}$$

$$(AB - \lambda I)X = 0 \tag{3}$$

$$(\lambda^2 A + \lambda B + C)X = 0 \tag{4}$$

Equations (2) and (3) are non-standard linear eigensystem problems and Equation (4) is a quadratic eigensystem problem. Their scalars λ and vectors X are also referred to as eigenvalues and eigenvectors.

In many instances there is no need to obtain the complete solution consisting of all the eigenvalue-vector pairs of an eigensystem problem. Often it suffices to compute a relatively small number of eigenvalues - for example, the m smallest or largest eigenvalues ($1 \leq m << n$) or eigenvalues in a prescribed interval, with or without their associated eigenvectors. We may refer to such computations as partial eigensystem solutions.

At present DTNSRDC has a wealth of Fortran subroutines for coping with both complete and partial eigensystem solutions in the EISPAK, IMSL, MSL, and ARL libraries for the CDC 6000 series of computers. The purpose of this report is to help the user make intelligent use of these subroutines.

THE LIBRARIES

The four libraries examined in this report are:

- the EISPACK LIBRARY[1] - This library was developed by the Argonne National Laboratory (University of Chicago, Atomic Energy Commission). It consists chiefly of FORTRAN adaptations of ALGOL eigensystem algorithms originally published in Numerische Mathematik.[2]

- the ARL LINEAR ALGEBRA LIBRARY[3] - This library was developed by the Aerospace Research Libraries (Air Force Systems Command, Wright-Patterson Air Force Base). It consists of linear equation solvers and eigensystem subroutines for real symmetric matrices. It is intended to augment EISPACK but with some deliberate overlapping.

---

1 Smith, B.T., et al, "Matrix Eigensystem Routines - EISPACK Guide," Lecture notes in Computer Science, Vol. 6, Springer-Verlag, New York (1974).

2 Wilkinson, J.H. and Reinsch, C., "Linear Algebra," Handbook for Automatic Computation, Vol. II, Springer-Verlag, New York (1971).

3 Nikolai, P.J. and Tsao, N.-K., "The ARL LINEAR ALGEBRA LIBRARY Handbook," Interim Report 1 September 1973-31 March 1974, ARL TR 74-0106 (July 1974), Aerospace Research Laboratories, Air Force Systems Command, Wright-Patterson AFB.

- the IMSL LIBRARY[4] - This proprietary multi-discipline library was produced by International Mathematical and Statistical Libraries, Inc. Its eigensystem capability is heavily dependent on EISPACK subroutines.

- the MATH SCIENCE LIBRARY[5] - This very large proprietary multi-discipline library was developed by the Boeing Company and is distributed by Control Data Corporation. Its eigensystem subroutines come from a variety of sources.

The ARL library was written with the parallel central processor of the CDC 6600 in mind. The others have been suitably modified for the CDC 6000 series of computers. These libraries of compiled subroutines are accessed using the following control cards:

```
ATTACH,LIBRARY
LDSET(LIB=LIBRARY)
LGO.
```

where "LIBRARY" is one of "MSL," "IMSL," "EISPACK," or "ARLMALG" (for the ARL library).

A word of caution: Although the ARL and EISPACK libraries are public domain material, the MSL and IMSL libraries are proprietary! The user can obtain FORTRAN subroutines from the public domain libraries which he can then store indefinitely in one form or another (listings, decks, permanent files), modify for his own special purposes, or distribute to non-DTNSRDC users. He can do NONE of these things

[4] The IMSL LIBRARY 3 Reference Manual, Vol. I, Edition 4 (FORTRAN 2.4) CDC 6200/6400/6500/6600/7600 (1975), revised November 1974, International Mathematical and Statistical Libraries, Inc., Houston, Texas.

[5] MATH SCIENCE LIBRARY, Vol. 6, Linear Algebra, Revision A, 3-1-71, Publication No. 60327500A, Control Data Corporation, Sunnyvale, California.

with subroutines from the proprietary libraries. These libraries are private property leased to DTNSRDC for stated intervals of time. Anyone authorized to use the computer facilities at DTNSRDC may make as much use of them as he wishes. However, there is no guarantee that given subroutines from a proprietary library will be available in the indefinite future since the contract for that library may not be renewed or the distributors of that library may alter its contents. Accordingly, due care should be exercised to avoid dependence on such subroutines, especially in production programs or programs which may be distributed outside DTNSRDC.

GENERAL REAL AND COMPLEX MATRICES

A general real matrix is a matrix of order $n \geq 2$ whose elements are real numbers. A general complex matrix is defined similarly. Although the following discussion combines the treatment of the real and complex cases, the user is urged always to distinguish them in practice. Complex subroutines should not be used for a real eigensystem problem because they require extra storage and computation for the complex arithmetic. In general, it is much more efficient to use subroutines which exploit such special matrix properties as symmetry, the Hermitian property, and bandedness.

Let λ and X be an eigenvalue-vector pair for the real or complex matrix A of order n. $(A-\lambda I)X=0$ implies that λ satisfies the n^{th} order polynomial equation

$$\text{determinant } (A-\lambda I) = (-1)^n\lambda^n + \ldots + C_0 = 0 \tag{5}$$

Conversely, for any λ which satisfies Equation (5) there exists an $X \neq 0$ such that $(A-\lambda I)X=0$. Clearly λ, X is an eigenvalue-vector pair for A.

From Equation (5) it follows that

1) each real or complex matrix possesses at least one eigenvalue,

2) but there can be no more than n distinct eigenvalues, and

3) eigenvalues may be repeated.

Since it is obvious that any non-zero scalar multiple of an eigenvector is also an eigenvector, it is clear that an eigenvalue really has a family of eigenvectors associated with it. However, a repeated eigenvalue need not have distinct eigenvector families for each of its multiplicities. Such a matrix is said to have a defective system of eigenvectors.[6]

Next let λ, X be an eigenvalue-vector pair for a real matrix A such that λ is a true complex number. Then it follows from conjugation that $\bar{\lambda}, \bar{X}$ is another eigenvalue-vector pair for A. This, of course, does not hold true when A is a complex matrix.

The following procedure is widely used for computing the eigensystem of a general real or complex matrix:

(1) the matrix is balanced;

(2) the balanced matrix is reduced to Hessenberg form;

(3) the eigenvalues of the Hessenberg matrix are computed;

(4) the eigenvectors of the Hessenberg matrix are computed;

(5) the eigenvectors of the balanced matrix are obtained from the eigenvectors of the Hessenberg matrix;

(6) the eigenvectors of the original matrix are obtained from the eigenvectors of the balanced matrix.

---

6 Wilkinson, J.H., "The Algebraic Eigenvalue Problem," Oxford, Clarendon Press (1965) pp. 9-10.

The balancing procedure (also called equilibration or scaling) of step 1 produces a matrix similar to the original matrix. This new matrix has the properties that its column norms do not differ greatly in order of magnitude and that isolated eigenvalues are recognized as such. As a result the new matrix is better conditioned for the Hessenberg reduction than the original. For details the reader is referred to Parlett and Reinsch.[7]

Now a Hessenberg matrix H is a real or complex matrix of order n such that $h_{ij}=0$ for $i=3,...,n$ and $1 \le j \le i-2$. (A 2x2 matrix is considered to be Hessenberg.) Properly speaking, H is an upper Hessenberg matrix but we shall refer to such matrices as Hessenberg. The details of the reduction to Hessenberg form are to be found in Martin and Wilkinson.[8]

Since the Hessenberg matrix is similar to the balanced matrix, which is in turn similar to the original matrix, the eigenvalues are those of the original matrix. It is clear that the accuracy of the computed eigenvalues is determined jointly by the accuracy of the Hessenberg reduction and the accuracy of the Hessenberg eigenvalue computation. In practice this procedure yields fairly accurate eigenvalues in conjunction with the LR or QR methods.[9] The method of Laguerre[10] is also used with the Hessenberg reduction though

---

[7] Parlett, B.N. and Reinsch, C., "Balancing a Matrix for Calculation of Eigenvalues and Eigenvectors," Numerische Mathematik, 13, 293-304 (1969).

[8] Martin, R.S. and Wilkinson, J.H., "Similarity Reduction of a General Matrix to Hessenberg Form," Numerische Mathematik, 12, 349-368 (1968).

[9] [6], pp. 485-569.

[10] [6], pp. 443-445, pp. 478-482.

its use is greatly overshadowed by that of the LR and QR methods. The LR and QR methods are applied to a Hessenberg matrix rather than to a full matrix because these algorithms involve iterations consisting of matrix multiplication and factorization. Since these algorithms preserve the Hessenberg form throughout and since each iteration requires significantly less work for a Hessenberg matrix than for a full matrix, the Hessenberg reduction proves to be quite advantageous.

The eigenvectors of the Hessenberg matrix can be obtained using the method of Peters and Wilkinson[11] or inverse iteration.[12] Due to the similarity of the matrices, the eigenvectors of the balanced matrix may be obtained from the corresponding eigenvectors of the Hessenberg matrix by a linear transformation, and likewise the eigenvectors of the original matrix may be obtained from those of the balanced matrix by another linear transformation.

SUBROUTINES FOR GENERAL COMPLEX MATRICES

The easiest subroutine to use is EIGCC(IMSL). It computes all the eigenvalues with an option for all the eigenvectors.

The following sequences of subroutines from the EISPACK library provide more flexibility as regards eigenvector computation. If more than 25% of the eigenvectors are desired then the third sequence should be used - provided that there is sufficient space to store all the eigenvectors. Also the use of the balancing subroutines CBAL and CBABK2

---

[11] Peters, G. and Wilkinson, J.H., "Eigenvectors of Real and Complex Matrices by LR and QR Triangularizations," Numerische Mathematik, 16, pp. 181-204 (1970).

[12] [6], pp. 619-633.

is desirable but not required. If they are omitted, the user must supply appropriate values for the LOW and IGH arguments of subroutine COMHES.

(all eigenvalues)
- CBAL
- COMHES
- COMLR

(all eigenvalues and those eigenvectors corresponding to specified eigenvalues)
- CBAL
- COMHES
- COMLR
- CINVIT
- COMBAK
- CBABK2

(all eigenvalues and eigenvectors)
- CBAL
- COMHES
- COMLR2
- CBABK2

IMSL provides two subroutines QREIGN and VALVEC. These subroutines use the QR method to compute all the eigenvalues while the EISPACK subroutines use the LR method. VALVEC has an eigenvector subset option.

SUBROUTINES FOR GENERAL REAL MATRICES

The easiest subroutine to use is EIGRF(IMSL). It computes all the eigenvalues with an option for all the eigenvectors.

The following sequences of subroutines from the EISPACK library provide more flexibility as regards eigenvector computation. If more than 25% of the eigenvectors are desired, then the third sequence should be used - provided that there is sufficient space to store all the eigenvectors.

Also, there is an option regarding the method to be used for the Hessenberg reduction - the ELMHES, ELTRAN, ELMBAK subroutines implement the elementary transformation method while the ORTHES, ORTRAN, ORTBAK subroutines implement the orthogonal transformation method. The use of the ELMHES, ELTRAN, ELMBAK subroutines is to be preferred over the other set of subroutines. Finally the use of the balancing subroutines BALANC and BALBAK is desirable but not required. If they are omitted, the user must supply appropriate values for the LOW and IGH arguments of ELMHES (ORTHES).

(all eigenvalues)
- BALANC
- ELMHES (ORTHES)
- HQR

(all eigenvalues and those eigenvectors corresponding to specified eigenvalues)
- BALANC
- ELMHES (ORTHES)
- HQR
- INVIT
- ELMBAK (ORTBAK)
- BALBAK

(all eigenvalues and eigenvectors)
- BALANC
- ELMHES (ORTHES)
- ELTRAN (ORTRAN)
- HQR2
- BALBAK

MSL provides two subroutines LATNTR and EIG5. The first uses the QR method (as do the EISPACK subroutines) to compute all the eigenvalues, while the second uses the method of

Laguerre to compute all the eigenvalues or certain subsets of them. Eigenvectors are computed for given real eigenvalues by means of a third subroutine EIGC01 whose use is not recommended for more than one or two eigenvectors. It is the author's opinion that, with regard to eigenvector calculation, the user would be better off with subroutines from IMSL or the EISPACK library.

REAL SYMMETRIC MATRICES

A real symmetric matrix is a real matrix of order $n \geq 2$ such that $a_{ij} = a_{ji}$ for all subscripts i,j. Real symmetric matrices have the property that their eigenvalues are real numbers. A positive definite matrix is a real symmetric matrix whose eigenvalues are positive.

The standard procedure for computing the eigensystem of a real symmetric matrix consists of the following steps:

(1) the matrix is reduced to a real symmetric tridiagonal matrix;

(2) the eigenvalues of the tridiagonal matrix are computed;

(3) the eigenvectors of the tridiagonal matrix are computed;

(4) the eigenvectors of the original matrix are obtained from those of the tridiagonal matrix.

Unlike the general case, real symmetric matrices do not require balancing prior to the tridiagonal reduction. Now a tridiagonal matrix is a matrix of order n where $n \geq 2$ such that the only non-zero elements will be found among the a_{11}, a_{ii}, $a_{i-1,i}$, $a_{i,i-1}$, $i=2,\ldots,n$ elements. This transformation to tridiagonal form results in a considerable reduction of work in the eigensystem computation for the same reasons given in the previous discussion of the general case. The tridiagonalization procedure used is that of Householder.[13]

The LR, QR, and QL algorithms can be used to obtain the complete set of eigenvalues.[9,14,15] The rational QR transformation method with Newton Shift or Sturm sequencing can be used to compute certain subsets of the eigenvalues.[16,17,18] The eigenvectors can be computed by the method of Peters and Wilkinson or inverse iteration. Since the original matrix is similar to the tridiagonal matrix, its eigenvectors are obtained from those of the tridiagonal matrix by a linear transformation.

---

[13] Martin, R.S., Reinsch, C., and Wilkinson, J.H., "Householder's Tridiagonalization of a Symmetric Matrix," Numerische Mathematik, 11, 181-195 (1968).

[14] Bowdler, H., Martin, R.S., Reinsch, C., and Wilkinson, J.H., "The QR and QL Algorithms for Symmetric Matrices," Numerische Mathematik, 11, 293-306 (1968).

[15] Dubrulle, A., Martin, R.S., and Wilkinson, J.H., "The Implicit QL Algorithm," Numerische Mathematik, 12, 377-383 (1968).

[16] Reinsch, C., and Bauer, F.L., "Rational QR Transformation with Newton Shift for Symmetric Tridiagonal Matrices," Numerische Mathematik, 11, 264-272 (1968).

[17] Barth, W., Martin, R.S., and Wilkinson, J.H., "Calculation of the Eigenvalues of a Symmetric Tridiagonal Matrix by the Method of Bisertion," Numerische Mathematik, 9, 386-393 (1967).

[18] Peters, G., Wilkinson, J.H., "The Calculation of Specified Eigenvectors," [2], pp. 418-439.

SUBROUTINES FOR REAL SYMMETRIC MATRICES

The easiest subroutine to use is EIGRS(IMSL). It computes all the eigenvalues with an option for all the eigenvectors. However, the matrix must be input in a symmetric storage mode. The IMSL sequence EHOUSS, EQRIS will provide the M smallest or largest eigenvalues. The IMSL sequence EHOUSS, EQR3S will provide the M smallest eigenvalues where M is the smallest integer such that the sum of the absolute values of the eigenvalues is not less than a given value.

The EISPACK sequences TRED1, IMTQL1 or TRED2, IMTQL2 will produce all the eigenvalues or all the eigenvalues and their eigenvectors, respectively. The IMTQL1, IMTQL2 subroutines are implementations of the implicit QL algorithm; there are also corresponding EISPACK subroutines TQL1, TQL2 which are implementations of the regular QL algorithm. However, the use of IMTQL1, IMTQL2 is recommended for tridiagonal matrices whose elements may vary considerably in magnitude. Such variation may well occur after reduction to tridiagonal form.

The EISPACK sequences TRED1, RATQR or TRED1, RATQR, TINVIT, TRBAK1 will produce the M smallest or largest eigenvalues or the M smallest or largest eigenvalues and their eigenvectors, respectively, using the rational QR transformation with Newton shift for the eigenvalues and inverse iteration for the eigenvectors. If more than 25% of the eigenvalues are required, it is more efficient to use the IMTQL1 subroutine instead of RATQR. Also, if more than 25% of the eigenvectors are required, it is more efficient to use IMTQL2 (provided that there is enough room to store all the eigenvectors) instead of RATQR, TINVIT, TRBAK1.

The EISPACK sequence TRED1, TSTURM computes all the eigenvalues and their eigenvectors lying in a given interval using Sturm sequencing for the eigenvalues and inverse iteration for the eigenvectors. As before, if more than 25% of these eigenvalues and eigenvectors are required, it is more efficient to use the TRED2, IMTQL2 sequence. The EISPACK sequence TRED1, BISECT computes all the eigenvalues lying in a given interval using Sturm sequencing. If more than 25% of the eigenvalues are required, it is more efficient to use the TRED1, IMTQL1 sequence. Eigenvectors for specified eigenvalues are obtained using the TINVIT, TRBAK1 subroutines after calling BISECT.

The ARL sequence TRI1, BISEC is recommended for computing a small selection of eigenvalues for a matrix of moderate size (<<50) using Sturm sequencing. The ARL sequence TRI1, RNQL1 will compute the entire set of eigenvalues for this order of matrix. The INIT and BAC1 subroutines will provide eigenvectors for specified eigenvalues computed by these two sequences. TRI1 and BAC1 are the tridiagonal reduction and eigenvector recovery subroutines, respectively. These subroutines store matrices in two-dimensional arrays. TRI2 and BAC2 are analogous subroutines which store matrices in linear arrays. Their substitution in the above sequences enables one to cope with matrices of larger order (<50).

The ARL sequence TRI3, IMQL1 is recommended for the complete eigensystem solution for matrices of order <50. TRI4 and IMQL2 are analogous subroutines which may be used for matrices of order >50.

The MSL subroutine EIGSYM computes all the eigenvalues with an option for computing the eigenvectors of the M largest or smallest eigenvalues. The user has his choice of the LR method, the QR method, or Sturm sequencing for eigenvalue computation. The eigenvectors are computed by inverse iteration. Although Sturm sequencing is quite reliable and very accurate, it is slow and thus the Sturm sequencing option is not recommended for computing all the eigenvalues. However, the MSL sequence TRIDI, SEPAR2 is suggested for computing a very few eigenvalues using Sturm sequencing. The MSL subroutine VECTOR will provide eigenvectors for given eigenvalues.

HERMITIAN MATRICES

A Hermitian matrix is a complex matrix such that $a_{ij} = \bar{a}_{ji}$ (where the bar represents complex conjugation) for all subscripts i,j. Hermitian matrices have the property that all their eigenvalues are real. The algorithm for computing the eigensystem of a Hermitian matrix is as follows:

(1) the matrix is reduced to a real symmetric tridiagonal matrix;

(2) the eigensystem of the tridiagonal matrix is computed;

(3) the eigenvectors of the Hermitian matrix are recovered from those of the tridiagonal matrix.

Householder's method is used for the tridiagonal reduction.[19] The eigensystem of the tridiagonal matrix is computed as before. The method for recovery of the eigenvectors is similar to that used in the real symmetric case.

The eigensystem problem for a Hermitian matrix can be reduced to the real symmetric eigensystem problem at the cost of doubling the order of the matrix.[20]

---

[19] Mueller, D.J., "Householder's Method for Complex Matrices and Eigensystems of Hermitian Matrices," Numerische Mathematik, 8, 72-92 (1966).

[20] [6], pp. 174-5.

SUBROUTINES FOR HERMITIAN MATRICES

The easiest subroutines to use are EIGCH(IMSH) and TCDIAG(MSL). In the EISPACK sequences for real symmetric matrices the HTRIDI subroutine can be substituted for TRED1 or TRED2 and the HIRIBK subroutine for TRBAK1 to obtain equivalent sequences for Hermitian matrices.

CERTAIN REAL NON-SYMMETRIC TRIDIAGONAL MATRICES

Certain real non-symmetric tridiagonal matrices, i.e., those with the property that the products of pairs of corresponding off-diagonal elements are non-negative, are similar to real symmetric tridiagonal matrices. This similarity provides an algorithm for computing the eigensystems of such matrices.

(1) The real non-symmetric matrix is transformed to a real symmetric tridiagonal matrix.

(2) The eigensystem problem is solved for that matrix.

(3) The eigenvectors of the original matrix are recovered from those of the last matrix.

SUBROUTINES FOR CERTAIN REAL NON-SYMMETRIC TRIDIAGONAL MATRICES

The substitution of the FIGI subroutine for TRED1 and the FIGI2 subroutine for TRED2 and the BAKVEC subroutine for TRBAK1 in the EISPACK sequences for real symmetric matrices provides corresponding sequences for these non-symmetric tridiagonal matrices.

BANDED REAL SYMMETRIC MATRICES

A banded real symmetric matrix is a real symmetric matrix such that $a_{ij} = 0$ if $|i-j| \geq m$ for some $m \leq n$. m is said to be the bandwidth of A.

SUBROUTINES FOR BANDED REAL SYMMETRIC MATRICES

The EISPACK subroutine BANDR reduces a banded real symmetric matrix to a real symmetric tridiagonal matrix. The EISPACK subroutines IMTQL1, BISECT, or RATQR can then be used to compute all or part of the eigenvalues: IMTQL2 will provide the entire eigensystem. The BANDV subroutine will compute eigenvectors for given eigenvalues. BANDV requires the original banded matrix which must be copied before it is destroyed by BANDR.

NON-STANDARD EIGENSYSTEM PROBLEMS

Consider the problem $(A-\lambda B)X=0$. If B is not singular, the problem is merely that of solving $(B^{-1}A-\lambda I)X=0$. If B is singular but A is not, then the problem is that of solving $(A^{-1}B- 1/\lambda I)X=0$. When B is singular, $A^{-1}B$ must be singular also. Thus $A^{-1}B$ has p zero eigenvalues where $p \geq 1$. In this case $(A-\lambda B)X=0$ possesses n-p eigenvalues which are the reciprocals of the non-zero eigenvalues of $A^{-1}B$. The eigenvectors of this problem are those of $A^{-1}B$ or $B^{-1}A$, respectively.

The expressions $A^{-1}B$ or $B^{-1}A$ should not be obtained by computing a matrix inverse and then performing a matrix multiplication. It is faster and more accurate to obtain these quantities by solving the systems of linear equations

AX=B or BX=A. Even so there is the possibility that the desired quantity may not be computed with enough significance to provide an eigensystem of the desired accuracy. Then too, there is the case when both A and B are singular and the above approach is not possible.

One procedure for the general case is the QZ algorithm of Moler and Stewart.[21] The QZ algorithm consists of two stages. In the first stage the matrices A and B are simultaneously reduced to Hessenberg form and upper triangular form, respectively. In the second stage zeros are introduced into the line of elements below the diagonal of the Hessenberg matrix so that that matrix becomes essentially triangular while the upper triangular matrix keeps that form. At this point the eigenvalues are obtained as the ratio of corresponding diagonal elements of the two reduced matrices. The eigenvectors of the original problem are related to those of the reduced problem by a linear transformation.

If both A and B are real symmetric matrices and one of them is positive definite, then the eigenvalues of $(A-\lambda B)X=0$ are real. In this case there is a procedure[22] which provides a real symmetric matrix C whose eigenvalues are those of $(A-\lambda B)X=0$. Then the eigensystem of C is computed by some procedure for real symmetric matrices. As usual the eigenvectors of the original problem are related to those of the reduced problem by a simple linear transformation.

The $(AB-\lambda I)X=0$ problem is similar. If both A and B are symmetric and one of them is positive definite, there is a procedure which reduces this problem to one of computing the eigensystem of a real symmetric matrix.[23]

---

[21] Stewart, G.W., "Introduction to Matrix Computations," Academic Press, New York (1973), pp. 387-394.

[22] [6], pp. 34-5.

[23] [6], pp. 337-9.

With regard to the quadratic problem $(\lambda^2 A + \lambda B + C)X=0$ we assume that A is not singular. If A is singular and C is not, then we may divide through by λ^2 and solve for the reciprocal of λ. The simple substitution of $Y=\lambda X$ after multiplication by A^{-1} results in the linear eigensystem problem

$$\left(\begin{array}{c|c} 0 & I \\ \hline -A^{-1}C & -A^{-1}B \end{array}\right)\binom{X}{Y} = \lambda\binom{X}{Y}$$

of order 2n. Multiplication by A^{-1}, completion of the square, and the substitution $Y = (\lambda A + \frac{1}{2}B)X$ yields another linear eigensystem problem

$$\left(\begin{array}{c|c} -\frac{1}{2}A^{-1}B & A^{-1} \\ \hline \frac{1}{4}BA^{-1}B-C & -\frac{1}{2}BA^{-1} \end{array}\right)\binom{X}{Y} = \lambda\binom{X}{Y}$$

of order 2n.[24] As noted previously, terms of the form $A^{-1}B$ should be obtained by solving AX=B rather than by matrix inversion and multiplication.

SUBROUTINES FOR NON-STANDARD EIGENSYSTEM PROBLEMS

The IMSL sequence EQZHF, EQZIF, EQZVAF computes all the eigenvalues of the general non-linear problems $(A-\lambda B)X=0$. The IMSL sequence EQZHSF, EQZISF, EQZVEF computes the entire eigensystem for this problem. Both sequences use the method of Moler and Stewart.

---

[24] Gignac, D.A., "Solution of a Complex Quadratic Eigenvalue Problem Related to Pipe Flow," Naval Ship Research and Development Center Report 4503 (August 1974).

If A is real symmetric and B is positive definite, the MSL subroutine REDSY1 can be used to reduce $(A-\lambda B)X=0$ to the form $(C-\lambda I)Y=0$. The MSL subroutine RECOV1 can be used to obtain the eigenvectors of the original problem from those of C. The reader must provide an MSL subroutine to compute the eigensystem of C. If A in addition has a narrow bandwidth and is non-negative definite (i.e., has no negative eigenvalues) and B is diagonal, then the MSL subroutine BANEIG can be used to obtain the M^{th} smallest eigenvalues (which are non-negative due to the restriction on A) and their eigenvectors.

With regard to the problem $(AB-\lambda I)X=0$, if A is real symmetric and B is positive definite, the MSL subroutine REDSY2 can be used to reduce the problem to one of the form $(C-\lambda I)Y=0$. The MSL subroutine RECOV2 can be used to obtain the eigenvectors.

OTHER SUBROUTINES

RITZIT

The ARL subroutine RITZIT computes the M largest eigenvalues with eigenvectors for a real symmetric matrix A. The main feature of the RITZIT subroutine is that the matrix A need not be stored in core. The user merely provides a subroutine OP which computes Y=AX where Y and X are real vectors. Thus RITZIT is well suited for large sparse matrices.

In the case of $(A-\lambda B)X=0$ or $(AB-\lambda I)X=0$, where A,B are real symmetric matrices and B is positive definite, the reduced matrix is of the form $L^{-1}A(L^{-1})^T$ or L^TAL where L and L^T (L transposed) are the Cholesky factors of B.[25] L is a

[25] [2], pp. 9-11.

lower triangular matrix. If B has a narrow bandwidth (as is often the case), then L has the same bandwidth though C is in general a full matrix. RITZIT circumvents the fullness of C and utilizes the bandwidth of B in the case of $C=L^{-1}A(L^{-1})^T$ by computing Y=CX thus

(1) $U=(L^{-1})^T X$ is obtained by solving $L^T U=X$ by means of backsubstitution.

(2) V=AU is provided by the OP subroutine.

(3) $Y=L^{-1}V$ is obtained by solving LY=V by means of backsubstitution.

For $C=L^T AL$ the computation of Y=CX consists merely of three consecutive multiplications.

However, the user must provide his own coding to obtain L and compute Y=CX. Also the user is required to provide a subroutine INF to monitor to the computation of RITZIT. For these reasons RITZIT is unfortunately not a subroutine that can be recommended for general use.

VARAH1 and VARAH2

The VARAH1 subroutine[26] computes the eigensystem of a real matrix using the double step QR algorithm for eigenvalues and inverse iteration for eigenvectors after reduction to Hessenberg form. The VARAH2 subroutine[26] refines a given eigensystem and provides error bounds for the improved eigensystem. VARAH1 and VARAH2 are FORTRAN Extended adaptations for the CDC 6000 series of computers of ALGOL procedures of J.M. Varah. These two subroutines suffer from complicated calling sequences and large core requirements. They are part of the NSRDC library and are loaded by the following control

---

[26] Gignac, D.A., "VARAH1 and VARAH2: Two Eigensystem Programs for General Real Matrices," Naval Ship Research and Development Center Report 3549 (February 1971).

cards:

```
ATTACH,DTNSRDC.
LDSET=(LIB=DTNSRDC)
LGO.
```

REDUCB1

The REDUCB1 subroutine[27] transforms the eigensystem problem $(A-\lambda B)X=0$, where A and B are real symmetric matrices and B is positive definite, to the eigensystem problem $(C-\lambda I)Y=0$, where C is a real symmetric matrix, by means of the Lanczos tridiagonalization process. It is further assumed that A and B are sparse; that is, these matrices have relatively few non-zero elements. REDUCB1 has the drawback that its calling sequence is slightly complicated. REDUCB1 is a FORTRAN Extended adaptation for the CDC 6000 series of computers of an ALGOL procedure of Golub, Underwood and Wilkinson with modification for sparse matrix core storage.

DOUBLE PRECISION SUBROUTINES

Certain EISPACK subroutines have been modified so that double precision arithmetic is used throughout. This approach was found to be necessary to obtain satisfactory answers to a certain problem.[24] These modified subroutines should be regarded as experimental and not as standard subroutines. The question of scaling was also briefly examined with regard to this problem.

REFINEMENT SUBROUTINES

The MSL library provides two refinement subroutines. EIGCHK improves an eigenvalue-vector pair λ_i, X_i for a real symmetric matrix given λ_{i-1} and λ_{i+1}. EIGCO1 converges to

---

[27] Gignac, D.A., "REDUCB1, A Lanczos Algorithm Subroutine for (A- B)X=0," Naval Ship Research and Development Center Report 4064 (February 1973).

the closest eigenvalue-vector pair from an approximate eigenvalue for a real matrix with real and distinct eigenvalues.

CONCLUSIONS

Of the four libraries the EISPACK library stands out as the best. The programmer looking for subroutines to handle eigensystem problems would be well advised to review the EISPACK library first as a matter of policy.

Although the eigensystem capability of the IMSL library is an adaptation of the EISPACK library, it is somewhat easier to use. It provides subroutines which call a sequence of subroutines while the EISPACK library requires the user to call the several subroutines himself.

For a subroutine library of its scope, the MSL library has some surprising deficiencies, particularly with regard to eigenvector calculation. However, it is completely and extensively documented.

The ARL library is restricted to symmetric real matrices and is intended to augment the EISPACK library.

ACKNOWLEDGMENTS

The author wishes to express his indebtedness to Dr. Elizabeth Cuthill (Code 1805) and Mrs. Sharon Good (Code 1892).

INITIAL DISTRIBUTION

Copies:
2 CHONR
  1 430/R. Lundegard
  1 432/L.D. BRAM

1 NRL
  1 8441/J. Hansen

2 USNA
  1 Dept Math
  1 Library

4 NAVPGSCOL
  1 59C1/G. Cantin
  1 M. Kelleher
  1 Math Dept
  1 Library

1 NAVWARCOL

1 NROTC & NAVADMINU, MIT

2 NSWC White Oak
  1 331/E. Cohen
  1 331/M. Vander Vorst

Copies:
1 NAVSHIPYD BREM/LIB

1 NAVSHIPYD CHASN/LIB

1 NAVSHIPYD MARE/LIB

1 NAVSHIPYD NORVA/LIB

1 NAVSHIPYD PEARL/LIB

1 NAVSHIPYD PHILA/LIB

1 NAVSHIPYD PTSMH/LIB

12 DDC

1 AIR FORCE AERO RES LAB
  1 P. Nikolai

CENTER DISTRIBUTION

Copies:

1	0000	DAVIS MICHAEL C
4	1725	JONES REMBERT F JR
		RODERICK JOAN E
		ROTH PETER N
		GIFFORD LEROY N JR
1	1745	NG CHRISTOPHER
1	1800	GLEISSNER GENE H
3	1802	SHANKS DANIEL
		FRENKIEL FRANCOIS
		THEILHEIMER FEODOR
1	1805	CUTHILL ELIZABETH H
1	1830	
1	1835	CULPEPPER LINWOOD M
1	1840	LUGT HANS J
2	1843	SCHOT JOANNA W
		EDDY ROBERT P
4	1844	DHIR SURENDRA K
		EVERSTINE GORDON C
		HENDERSON FRANCIS M
		MATULA PETRO
10	1844	GIGNAC DONALD A
1	1850	CORIN THOMAS
1	1860	SULIT ROBERT A
1	1880	CAMARA ABEL W
1	1890	GRAY GILBERT R
1	1890	TAYLOR NORA M
3	1892	GOOD SHARON E
		MUELLER LOUIS
		SMITH GEORGE H
2	1966	CASPAR JOHN R
		LIU YUAN-NING
30	5211	REPORTS DISTRIBUTION
1	5221	LIBRARY (C)
1	5222	LIBRARY (A)