A Subroutine Package for the Efficient Solution of the Eigenproblem of Real Symmetric Toeplitz Matrices

Marvin J. Goldstein
William A. Babson
Computer and Information Services Department
A Subroutine Package for the Efficient Solution of the Eigenproblem of Real Symmetric Toeplitz Matrices

Marvin Goldstein; William Babson

Naval Underwater Systems Center, New London, CT, 06320

TM 831172

We discuss a subroutine package for solving the eigenproblem of bisymmetric Toeplitz matrices. When the eigenvalues and eigenvectors of a bisymmetric Toeplitz matrix are computed, storage requirements and execution time are reduced by taking advantage of the special structure of the matrix.

16. SECURITY CLASSIFICATION OF:
   a. REPORT
      unclassified
   b. ABSTRACT
      unclassified
   c. THIS PAGE
      unclassified
A SUBROUTINE PACKAGE FOR THE EFFICIENT SOLUTION OF
THE EIGENPROBLEM OF REAL SYMMETRIC TOEPLITZ MATRICES

Date: December 30, 1983
Prepared by: Marvin J. Goldstein
William A. Babson

Approved for public release; distribution unlimited
ABSTRACT

We discuss a subroutine package for solving the eigenproblem of bisymmetric Toeplitz matrices. When the eigenvalues and eigenvectors of a bisymmetric Toeplitz matrix are computed storage requirements and execution time are reduced by taking advantage of the special structure of the matrix.

ADMINISTRATIVE INFORMATION

This memorandum was prepared under Job Order No. 771Y00, Special Projects and Studies. The authors are located at the Naval Underwater Systems Center, New London, Connecticut, 06320.

ACKNOWLEDGEMENT

The authors are grateful to Dr. John Ianniello (Code 3212) for technically reviewing this memorandum.
INTRODUCTION

An $n$-th order real symmetric matrix $M=(M_{jk})$ for which

$$M_{jk} = M_{j+1, k+1} \quad \text{(1)}$$

is called a bisymmetric Toeplitz matrix. Bisymmetric Toeplitz matrices arise, for example, as correlation matrices in the study of discrete stationary random processes. Each element of the correlation matrix is a function of $j-k$; that is,

$$M_{jk} = M_{j-k} = M_{k-j}$$

(e.g. $M_{jk} = \cos(j-k)c$, where $c$ is a real number.)

In [1], using the bisymmetry of matrix $M$, Goldstein shows that the eigenvalues and eigenvectors of matrix $M$ can be computed from two real symmetric matrices one-half its size, if it is of even order. In this memorandum, we provide a FORTRAN subroutine package that is an implementaiton of this reduction. By using this package, one reduces both computer time and storage ordinarily needed to calculate the eigenvalues and eigenvectors of bisymmetric Toeplitz matrices.
The sub-program package in the Appendix computes the eigenvalues and (optionally) the eigenvectors of an even order, \( n \), real symmetric matrix \( M \), whose coefficients satisfy Eq. (1), from two real symmetric matrices:

\[
M^{-j_k}(+) = M_{jk} + M_{j,n+1-k} = M^{-k_j}(+)
\]  

\[(k=1, \ldots, n/2; j=1, \ldots, k)\]  

\[
M^{-j_k}(-) = M_{jk} - M_{j,n+1-k} = M^{-k_j}(-)
\]  

\[(3)\]

one-half the size of matrix \( M \), by [1, 2].

The matrices \( M^+ \) and \( M^- \) are computed from the first row elements of matrix \( M \). For example, by Property (1), Formula (2) reduces to

\[
M^{-j_k}(+) = M_{1,k-(j-1)} + M_{1,n-(k+j)+2}
\]  

\[
= M^{-k_j}(+) \quad k=1, \ldots, n/2; j=1, \ldots, k
\]  

where addition of the coefficients of the first row of matrix \( M \) is replaced by subtraction when computing \( M^-(-) \).
Forming $M^-(+)$ first, the subroutine package calculates the eigenvalues and (optionally) the eigenvectors of $M^-(+)$.

Then $M^-(-)$ is formed and its eigenvalues and (optionally) eigenvectors are computed.

The eigenvalues and eigenvectors of matrices $M^-(+)$ and $M^-(-)$ determine the eigenvalues and eigenvectors of matrix $M$. If $z$ is an eigenvector of $M^-(+)\) with corresponding eigenvalue $d$, and $J$ is the identity matrix of order $n/2$ with its columns written in reverse order, then

$$
\begin{bmatrix}
  z \\
  z
\end{bmatrix}
= d
\begin{bmatrix}
  Jz \\
  Jz
\end{bmatrix}.
$$

Similarly, if $z$ is an eigenvector of $M^-(\cdot)$ with corresponding eigenvalue $d$, then

$$
\begin{bmatrix}
  z \\
  z
\end{bmatrix}
= d
\begin{bmatrix}
  -Jz \\
  -Jz
\end{bmatrix}.
$$
STORAGE CONSIDERATIONS

Note that in (5) and (6), since premultiplying a vector by matrix J just reverses the order of the vector's components, all the information about the eigenvectors of matrix M is given by the eigenvectors of matrices $M^{-}(+) \text{ and } M^{-}(-)$; therefore, only $n/2$ of the components of each eigenvector of matrix $M$ have to be stored, and a fortiori, only $n^2/2$ entries (instead of $n^2$) are required to store all the eigenvectors of matrix $M$.

Furthermore, since the eigenvalues and eigenvectors of $M^{-}(+)$ are computed independently of those for $M^{-}(-)$, matrices $M^{-}(+)$ and $M^{-}(-)$ may share the same storage area for a matrix of order $n/2$. Therefore, the reduction requires a maximum of $(3/4)n^2 + n$ storage locations to accommodate the matrices $M^{-}(+)$, $M^{-}(-)$ and their eigenvalues and eigenvectors, instead of the $2n^2 + n$ required when the eigensystem is solved directly from matrix $M$. In particular, only $(1/4)n^2 + n$ storage locations are used to accommodate $M^{-}(+)$ and $M^{-}(-)$ and their eigenvalues, if only the eigenvalues of matrix $M$ are required.

TIMING

Computational results indicate that the eigenproblem for a large matrix $M$ can be solved by computer at least four times faster this way than directly from matrix $M$. For example, the following table gives representative execution times (in seconds) to compute the eigenvalues of bisymmetric Toeplitz matrices of orders 16, 32, 64, 128, 256 by the reduction SUBROUTINE
MATRED (in the Appendix) and directly by the IMSL library SUBROUTINE EIGRS [3] on the VAX 11/780 in double precision arithmetic.

<table>
<thead>
<tr>
<th>Order (n)</th>
<th>MATRED (using EIGRS)</th>
<th>EIGRS</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>.03</td>
<td>.07</td>
</tr>
<tr>
<td>32</td>
<td>.13</td>
<td>.38</td>
</tr>
<tr>
<td>64</td>
<td>.73</td>
<td>2.64</td>
</tr>
<tr>
<td>128</td>
<td>4.97</td>
<td>21.00</td>
</tr>
<tr>
<td>256</td>
<td>38.53</td>
<td>248.63</td>
</tr>
</tbody>
</table>

The following table gives representative execution times (in seconds) for a single precision version of SUBROUTINE MATRED and a single precision version of EIGRS on the UNIVAC 1100/62:* 

<table>
<thead>
<tr>
<th>Order (n)</th>
<th>Single Precision MATRED</th>
<th>Single Precision EIGRS</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>.02</td>
<td>.05</td>
</tr>
<tr>
<td>32</td>
<td>.10</td>
<td>.33</td>
</tr>
<tr>
<td>64</td>
<td>.60</td>
<td>2.24</td>
</tr>
<tr>
<td>128</td>
<td>4.08</td>
<td>16.76</td>
</tr>
<tr>
<td>256</td>
<td>30.16</td>
<td>129.04</td>
</tr>
</tbody>
</table>

* Since only a single precision version of EIGRS is available in the IMSL library on the UNIVAC, the UNIVAC version of MATRED is single precision.
In multi-program computer environments, variations in the execution time of a program can occur due to differences in machine load. Therefore, in order to smooth out these variations, subroutines MATRED and EIGRS were run a number of times on different matrices of order $n$ and the average execution time for each order was recorded in the tables.

**USER PROCEDURE**

The calling sequence of SUBROUTINE MATRED (double precision version) is defined in its commentary in the Appendix. The relocatable code, MATRD.OBJ, is in the VAX 11/780 directory [MJG] on node 2. The UNIVAC single precision version of SUBROUTINE MATRED is in the UNIVAC 1100/62 (node 2) file MGOLDSTN*MATRD and has the element name MATRD. Its calling sequence is the same as the double precision version on the VAX, but none of the arguments in the sequence are double precision.

**SUMMARY**

A subroutine package for computing the eigenvalues and (optionally) the eigenvectors of a bisymmetric Toeplitz matrix is available which reduces storage requirements and execution time by taking advantage of the special structure of the matrix. In particular, if only the eigenvalues of the matrix are required storage requirements for arrays and execution time are reduced by 75% when the matrix is sufficiently large.
REFERENCES


SUBROUTINE MATRED(IVEC,MR1,N,IFULST,MTILFL,IFUL,MTILSY,D,Z,IZ,WK)

****************************************•··············•••**********

A SUBROUTINE PACKAGE FOR THE EFFICIENT SOLUTION

OF THE EIGENPROBLEM OF REAL SYMMETRIC TOEPLITZ MATRICES

BY

MARVIN GOLSTEIN AND WILLIAM BABSON

*** THIS SUB-PROGRAM PACKAGE COMPUTES THE EIGENVALUES AND

*** (OPTIONALLY) THE EIGENVECTORS OF AN EVEN ORDER (n) REAL SYM-

*** METRIC TOEPLITZ MATRIX:

***FROM TWO REAL SYMMETRIC MATRICES:

***ONE-HALF THE SIZE OF MATRIX M, BY [1, 2).

*** THE MATRICES M-(+) AND M-(-) CAN BE COMPUTED FROM THE

*** FIRST ROW ELEMENTS OF MATRIX M. FOR EXAMPLE, BY PROPERTY (1),***

*** FORMULA (2) REDUCES TO

***

***WHERE ADDITION OF THE COEFFICIENTS OF THE FIRST ROW OF MATRIX ***

*** IS REPLACED BY SUBTRACTION WHEN COMPUTING M-(-).***

*** IF z IS AN EIGENVECTOR OF M-(+) WITH CORRESPONDING EIGEN-

*** VALUE \( \alpha \), AND J IS THE IDENTITY MATRIX OF ORDER \( n/2 \) WITH ITS ***

*** COLUMNS WRITTEN IN REVERSE ORDER, THEN

*** SIMILARLY, IF z IS AN EIGENVECTOR OF M-(-) WITH CORRESPONDING ***

*** EIGENVALUE \( \alpha \), THEN

*** COMPUTATIONAL RESULTS SHOW THAT THE EIGENPROBLEM FOR

*** A MATRIX THAT SATISFIES (1) CAN BE SOLVED BY COMPUTER AT LEAST***

*** FOUR TIMES FASTER THIS WAY THAN BY SOLVING THE EIGENPROBLEM ***

***DIRECTLY FROM M FOR LARGE VALUES OF n.***
**FURTHERMORE, SINCE THE EIGENVALUES AND EIGENVECTORS OF Z
**ARE COMPUTED INDEPENDENTLY OF THOSE FOR M(-), ONLY
**A MAXIMUM OF (3/4)n**2 + n STORAGE LOCATIONS ARE USED TO AC-
**OMODATE THESE MATRICES, THEIR EIGENVALUES AND EIGENVECTORS.
**VARIABLE ORDER IN THE PROGRAM THAT HOLDS EIGENVECTORS OF M(+)
**AND M(-). IZ IS ONE-HALF THE DIMENSION OF MRT IN THE CALLING PROGRAM IF IVEC=1;
**OTHERWISE, IZ IS ONE. IZ WILL BE ONE-HALF THE DIMENSION OF MRT IN THE CALLING PROGRAM IF IVECO=
**OUTPUT TO CALLING PROGRAM:
**D DOUBLE PRECISION REAL VECTOR OF AT LEAST LENGTH N.
**C

---

**REFERENCES**

2. Goldstein, Marvin, "Further Decomposition of the Pseudoinverse of a Hermitian Pseudosymmetric Matrix," ACM
INTEGER N,IER,IFULST,IFUL,IZ,IVEC,NDIV2,POINT
DOUBLE PRECISION MR1,MTILFL,MTILSY,D,Z,WK,SIGN
DIMENSION MK1111,MTILFL(IFUL1),MTILSY(I),D(1),Z((Z,1),WK(1))

COMPUTE ORDER OF REDUCED MATRICES

NDIV2=N/2

SET SIGN TO COMPUTE M-(+)
SIGN=1.0
DO 15 1=1,2
   IF(1FULST.LT.1) GO TO 10
   GO TO 20
10   CALL MTILSY(NDIV2,MR1,MTILSY,SIGN)
   GO TO 30
20   CALL EIGRS(MTILSY,NDIV2,IVEC,D(POINT),Z(1,POINT),IZ,WK,IER)
   CALL MTILFL(NDIV2,MR1,MTILFL,NDIV2,SIGN)
   CALL EIGRS(MTILFL,NDIV2,IVEC,D(POINT),Z(1,POINT),IZ,WK,IER)
   GO TO 30
15   CONTINUE

SINCE IMSL ROUTINE EIGRS REQUIRES THAT MTILFL BE EXACTLY N/2 BY N/2
THE FOURTH ARGUMENT IN THE MTILFL CALL HAS BEEN REPLACED BY NDIV2.
RESTORE THIS ARGUMENT TO IFUL1 IF REQUIRED BY AN EIGENVALUE ROUTINE.

CALL MTILFUL(NDIV2,MR1,MTILFL,NDIV2,SIGN)
CALL EIGRS(MTILFL,NDIV2,IVEC,D(POINT),Z(1,POINT),IZ,WK,IER)
30   CONTINUE

SET SIGN TO COMPUTE M-(-)
SIGN=-1.0
DO 40 1=1,2
   IF(1FULST.LT.1) GO TO 10
   GO TO 20
10   CALL MTILSY(NDIV2,MR1,MTILSY,SIGN)
40   CONTINUE
20   CALL EIGRS(MTILSY,NDIV2,IVEC,D(POINT),Z(1,POINT),IZ,WK,IER)
15   CONTINUE

RETURN
END
SUBROUTINE MTLSYM(NDIV2, MR1, MTILDA, SIGN)
C
*** THIS SUBROUTINE RETURNS M- (+) OR M- (-) STORED IN ***
C
*** SYMMETRIC STORAGE MODE IN THE ONE-DIMENSIONAL, DOUBLE ***
C
*** PRECISION ARRAY MTILDA. M- (+) AND M- (-) ARE COMPUTED ***
C
*** FROM THE FIRST ROW OF MATRIX M, NAMELY MR1, WHEN SIGN ***
C
*** IS +1 AND -1, RESPECTIVELY. ***
C
********************************************************************
C
INTEGER I, J, K, NDIV2, N, L1, L2
DOUBLE PRECISION MR1, MTILDA, SIGN
DIMENSION MR1(1), MTILDA(1)

N=NDIV2*2
I=0
DO 20 K=1, NDIV2
DO 10 J=1, K
I=I+1
L1=K-J+1
L2=N+2-K-J
MTILDA(I)=MR1(L1)+MR1(L2)*SIGN
10 CONTINUE
20 CONTINUE
RETURN
END

SUBROUTINE MTLFUL(NDIV2, MR1, MTILDA, IFUL, SIGN)
C
*** THIS SUBROUTINE RETURNS M- (+) OR M- (-) STORED IN ***
C
*** FULL STORAGE MODE IN THE TWO-DIMENSIONAL, DOUBLE PRE- ***
C
***CISION ARRAY MTILDA. M- (+) OR M- (-) ARE COMPUTED FROM***
C
*** THE FIRST ROW OF MATRIX M, NAMELY MR1, WHEN SIGN IS ***
C
*** +1 AND -1, RESPECTIVELY. ***
C
********************************************************************
C
INTEGER N, NDIV2, I, IFUL, J, K, L1, L2
DOUBLE PRECISION MR1, MTILDA, SIGN
DIMENSION MR1(1), MTILDA(IFUL,1)

N=NDIV2*2
I=0
DO 20 K=1, NDIV2
DO 10 J=1, K
I=I+1
L1=K-J+1
L2=N+2-K-J
MTILDA(J,K)=MR1(L1)+MR1(L2)*SIGN
MTILDA(K,J)=MTILDA(J,K)
10 CONTINUE
20 CONTINUE
RETURN
END
A SUBROUTINE PACKAGE FOR THE EFFICIENT SOLUTION OF
THE EIGENPROBLEM OF REAL SYMMETRIC TOEPLITZ MATRICES
Marvin J. Goldstein, Computer and Information Services Dept.
William Babson, Computer and Information Services Dept.
TM No. 831172
UNCLASSIFIED

DISTRIBUTION LIST

External:

Mr. I. L. Avrunin
David Taylor Naval Ship
Research & Development
Center
Headquarters
Bethesda, MD 20084

Dr. Martin Schultz
Research Center for Scientific Comp.
Yale Univ.
P. O. Box 2158 Yale Station
New Haven, CT 06520

Mr. Ken Kunec
Naval Data Automation
Command
Washington Navy Yard
Washington, DC 20374