Contribution II/10

Reduction of the Symmetric Eigenproblem $Ax=\lambda Bx$ and Related Problems to Standard Form*

by R. S. MARTIN and J. H. WILKINSON

1. Theoretical Background

In many fields of work the solution of the eigenproblems $Ax=\lambda Bx$ and $ABx=\lambda x$ (or related problems) is required, where A and B are symmetric and B is positive definite. Each of these problems can be reduced to the standard symmetric eigenproblem by making use of the Cholesky factorization [4] of B.

For if L is defined by

$$LL^T=B \tag{1}$$

then**

$$Ax=\lambda Bx \quad \text{implies} \quad (L^{-1}AL^{-T})(L^Tx)=\lambda(L^Tx) \tag{2}$$

and

$$ABx=\lambda x \quad \text{implies} \quad (L^TAL)(L^Tx) \quad =\lambda(L^Tx). \tag{3}$$

Hence the eigenvalues of $Ax=\lambda Bx$ are those of $Py=\lambda y$, where P is the symmetric matrix $L^{-1}AL^{-T}$, while the eigenvalues of $ABx=\lambda x$ are those of $Qy=\lambda y$, where Q is the symmetric matrix L^TAL. Note that in each case the eigenvector x of the original problem is transformed into L^Tx.

There are a number of closely related problems which are defined by the equations

$$\begin{aligned} y^TAB=\lambda y^T \quad &\text{or} \quad (L^{-1}y)^T(L^TAL)=\lambda(L^{-1}y)^T \\ BAy=\lambda y \quad &\text{or} \quad (L^TAL)(L^{-1}y) \;=\lambda(L^{-1}y) \\ x^TBA=\lambda x^T \quad &\text{or} \quad (L^Tx)^T(L^TAL) \;=\lambda(L^Tx)^T. \end{aligned} \tag{4}$$

Here it is assumed throughout that it is B which is required to be positive definite (A may or may not be positive definite). It will be observed that all the problems can be reduced if we have procedures for computing L^TAL and $L^{-1}AL^{-T}$. Further if we denote an eigenvector of the derived problem by z, we need only procedures for solving

$$L^Tx=z \quad \text{and} \quad L^{-1}y=z \quad \text{i.e.} \quad y=Lz. \tag{5}$$

If z is normalized so that $z^Tz=1$ then x and y determined from Eqs. (5) satisfy

$$x^TBx=1 \quad \text{and} \quad y^TB^{-1}y=1, \quad y^Tx=1 \tag{6}$$

and these are the normalizations that are usually required in practice.

* Prepublished in Numer. Math. 11, 99—110 (1968).

** Throughout this paper we have used L^{-T} in place of the cumbersome $(L^{-1})^T$ and $(L^T)^{-1}$.

Any decomposition $B = XX^T$ may be used in a similar way, but the Cholesky decomposition is the most economical, particularly when carried out as described in Section 5. If the standard eigenproblem for the matrix B is solved by a method *which gives an accurately orthogonal system Y of eigenvectors* then we have

$$B = Y \operatorname{diag}(d_i^2)\, Y^T = Y D^2 Y^T \tag{7}$$

where d_i^2 are the eigenvalues (positive) of B. Hence corresponding to $Ax = \lambda Bx$ we may solve $(D^{-1} Y^T A Y D^{-1})(D Y^T x) = \lambda (D Y^T x)$ and similarly for the other problems. Generally this is much less efficient than the use of the Cholesky decomposition, but it may have slight advantages in very special cases (see the discussion on pp. 337—338, 344 of ref. [7]).

2. Applicability

reduc 1 may be used to reduce the eigenproblem $Ax = \lambda Bx$ to the standard symmetric eigenproblem $Pz = \lambda z$, where $P = L^{-1} A L^{-T}$ and $B = LL^T$.

reduc 2 may be used to reduce the eigenproblems $y^T AB = \lambda y^T$, $BAy = \lambda y$, $ABx = \lambda x$, $x^T BA = \lambda x^T$ to the standard symmetric eigenproblem $Qz = \lambda z$, where $Q = L^T A L$ and $B = LL^T$.

When A and B are narrow symmetric band matrices of high order these procedures should not be used as they destroy the band form.

The derived standard symmetric eigenproblem may be solved by procedures which are included in this series [1, 3, 5, 6]. The eigenvalues of the derived standard problem are those of the original problem but the vectors are related as indicated by Eq. (2), (3), (4). The eigenvectors of the original problem may then be obtained using the procedures *rebak a* and *rebak b* described below.

rebak a provides eigenvectors x of the problems $Ax = \lambda Bx$, $ABx = \lambda x$, $x^T BA = \lambda x^T$ from the corresponding eigenvectors $z = L^T x$ of the derived standard symmetric problem.

rebak b provides eigenvectors y of the problems $y^T AB = \lambda y^T$, $BAy = \lambda y$ from the corresponding eigenvectors $z = L^{-1} y$ of the derived standard symmetric problem.

3. Formal Parameter List

3.1. Input to procedure *reduc 1*

- n order of matrices A and B, or negative order if L already exists.
- a elements of the symmetric matrix A given as upper-triangle of an $n \times n$ array (strict lower-triangle can be arbitrary).
- b elements of the symmetric positive definite matrix B given as upper-triangle of an $n \times n$ array (strict lower-triangle can be arbitrary).

Output of procedure *reduc 1*

- a elements of symmetric matrix $P = L^{-1} A L^{-T}$ given as lower-triangle of $n \times n$ array. The strict upper-triangle of A is still preserved but its diagonal is lost.

b sub-diagonal elements of matrix L such that $LL^T=B$ stored as strict lower-triangle of an $n\times n$ array.

dl the diagonal elements of L stored as an $n\times 1$ array.

fail the exit used if B, possibly on account of rounding errors, is not positive definite.

3.2. Input to procedure *reduc 2*

n order of matrices A and B or negative order if L already exists.

a elements of the symmetric matrix A given as upper-triangle of an $n\times n$ array (strict lower-triangle can be arbitrary).

b elements of the symmetric positive definite matrix B given as upper-triangle of an $n\times n$ array (strict lower-triangle can be arbitrary).

Output from procedure *reduc 2*

a elements of symmetric matrix $Q=L^TAL$ given as lower-triangle of $n\times n$ array. The strict upper-triangle of A is still preserved but its diagonal is lost.

b sub-diagonal elements of matrix L such that $LL^T=B$ stored as strict lower-triangle of an $n\times n$ array.

dl the diagonal elements of L stored as an $n\times 1$ array.

fail the exist used if B, possibly on account of rounding errors, is not positive definite.

3.3. Input to procedure *rebak a*

n order of matrices A and B.

m1, m2 eigenvectors *m1*, ..., *m2* of the derived standard symmetric eigenproblem have been found.

b the sub-diagonal elements of the matrix L such that $LL^T=B$ stored as the strict lower-triangle of an $n\times n$ array (as given by *reduc 1* or *reduc 2*).

dl the diagonal elements of L stored as an $n\times 1$ array.

z an $n\times(m2-m1+1)$ array containing eigenvectors *m1*, ..., *m2* of the derived symmetric eigenproblem.

Output from procedure *rebak a*

z an $n\times(m2-m1+1)$ array containing eigenvectors of the original problem; output $z=L^{-T}$(input z).

3.4. Input to procedure *rebak b*

n order of matrices A and B.

m1, m2 eigenvectors *m1*, ..., *m2* of the derived standard symmetric eigenproblem have been found.

b the sub-diagonal elements of the matrix L such that $LL^T = B$ stored as the strict lower-triangle of an $n \times n$ array (as given by *reduc 1* or *reduc 2*).

dl the diagonal elements of L stored as an $n \times 1$ array.

z an $n \times (m2 - m1 + 1)$ array containing eigenvectors *m1*, ..., *m2* of the derived symmetric eigenproblem.

Output from procedure *rebak b*

z an $n \times (m2 - m1 + 1)$ array containing eigenvectors of the original problem. output $z = L$(input z).

4. ALGOL Programs

```
procedure reduc1 (n) trans: (a, b) result: (dl) exit: (fail);
value n; integer n; array a, b, dl; label fail;
comment Reduction of the general symmetric eigenvalue problem
                    A × x = lambda × B × x,
        with symmetric matrix A and symmetric positive definite matrix B,
        to the equivalent standard problem P × z = lambda × z.
            The upper triangle, including diagonal elements, of A and B are
        given in the arrays a[1:n, 1:n] and b[1:n, 1:n].
            L (B=L×LT) is formed in the remaining strictly lower triangle
        of the array b with its diagonal elements in the array dl[1:n], and the
        lower triangle of the symmetric matrix P (P=inv(L)×A×inv(LT))
        is formed in the lower triangle of the array a, including the diagonal
        elements. Hence the diagonal elements of A are lost.
            If n<0, it is assumed that L has already been formed.
            The procedure will fail if B, perhaps on account of rounding errors,
        is not positive definite;
begin   integer i, j, k;
        real x, y;
        if n<0 then n := -n
        else
        for i := 1 step 1 until n do
        for j := i step 1 until n do
        begin x := b[i, j];
              for k := i-1 step -1 until 1 do
              x := x - b[i, k] × b[j, k];
              if i=j then
              begin if x ≦ 0 then go to fail;
                    y := dl[i] := sqrt(x)
              end
              else b[j, i] := x/y
        end ji;
        comment L has been formed in the array b;
        for i := 1 step 1 until n do
        begin y := dl[i];
              for j := i step 1 until n do
```

```
            begin x := a[i, j];
                  for k := i-1 step -1 until 1 do
                  x := x-b[i, k]×a[j, k];
                  a[j, i] := x/y
            end j
      end i;
      comment The transpose of the upper triangle of inv(L)×A has been
                  formed in the lower triangle of the array a;
      for j := 1 step 1 until n do
      for i := j step 1 until n do
      begin x := a[i, j];
            for k := i-1 step -1 until j do
            x := x-a[k, j]×b[i, k];
            for k := j-1 step -1 until 1 do
            x := x-a[j, k]×b[i, k];
            a[i, j] := x/dl[i]
      end ij
end reduc1;

procedure reduc2 (n) trans: (a, b) result: (dl) exit: (fail);
value n; integer n; array a, b, dl; label fail;
comment Reduction of the general symmetric eigenvalue problems
            A×B×x = lambda×x,     yT×A×B = yT×lambda,
            B×A×y = lambda×y,     xT×B×A = xT×lambda,
      with symmetric matrix A and symmetric positive definite matrix B,
      to the equivalent standard problem Q×z = lambda×z.
         The upper triangle, including diagonal elements, of A and B are
      given in the arrays a[1:n, 1:n] and b[1:n, 1:n].
         L (B=L×LT) is formed in the remaining strictly lower triangle of
      the array b with its diagonal elements in the array dl[1:n], and the
      lower triangle of the symmetric matrix Q (Q=LT×A×L) is formed
      in the lower triangle of the array a, including the diagonal elements.
      Hence the diagonal elements of A are lost.
         If n<0, it is assumed that L has already been formed.
         The procedure will fail if B, perhaps on account of rounding errors,
      is not positive definite;
begin integer i, j, k;
      real x, y;
      if n<0 then n := -n
      else
      for i := 1 step 1 until n do
      for j := i step 1 until n do
      begin x := b[i, j];
            for k := i-1 step -1 until 1 do
            x := x-b[i, k]×b[j, k];
            if i=j then
```

```
                begin if x<0 then go to fail;
                      y := dl[i] := sqrt(x)
                end
                else b[j, i] := x/y
        end ji;
        comment L has been formed in the array b;
        for i := 1 step 1 until n do
        for j := 1 step 1 until i do
        begin x := a[j, i]×dl[j];
              for k := j+1 step 1 until i do
              x := x+a[k, i]×b[k, j];
              for k := i+1 step 1 until n do
              x := x+a[i, k]×b[k, j];
              a[i, j] := x
        end ji;
        comment The lower triangle of A×L has been formed in the lower
                triangle of the array a;
        for i := 1 step 1 until n do
        begin y := dl[i];
              for j := 1 step 1 until i do
              begin x := y×a[i, j];
                    for k := i+1 step 1 until n do
                    x := x+a[k, j]×b[k, i];
                    a[i, j] := x
              end j
        end i
end reduc2;

procedure rebaka (n) data: (m1, m2, b, dl) trans: (z);
value n, m1, m2; integer n, m1, m2; array b, dl, z;
comment This procedure performs, on the matrix of eigenvectors, Z, stored in
        the array z[1:n, m1:m2], a backward substitution LT×X=Z, over-
        writing X on Z.
           The diagonal elements of L must be stored in the array dl[1:n],
        and the remaining triangle in the strictly lower triangle of the array
        b[1:n, 1:n]. The procedures reduc1 and reduc2 leave L in this desired
        form.
           If x denotes any column of the resultant matrix X, then x satisfies
        xT×B×x=zT×z, where B=L×LT;
begin   integer i, j, k;
        real x;
        for j := m1 step 1 until m2 do
        for i := n step -1 until 1 do
```

```
            begin x := z[i, j];
                  for k := i+1 step 1 until n do
                  x := x - b[k, i] × z[k, j];
                  z[i, j] := x/dl[i]
            end ij
end rebaka;

procedure rebakb (n) data: (m1, m2, b, dl) trans: (z);
value n, m1, m2; integer n, m1, m2; array b, dl, z;
comment This procedure performs, on the matrix of eigenvectors, Z, stored in
            the array z[1:n, m1:m2], a forward substitution Y=L×Z, over-
            writing Y and Z.
               The diagonal elements of L must be stored in the array dl[1:n],
            and the remaining triangle in the strictly lower triangle of the array
            b[1:n, 1:n]. The procedures reduc1 and reduc2 leave L in this desired
            form.
               If y denotes any column of the resultant matrix Y, then y satis-
            fies yT × inv(B) × y = zT × z, where B=L × LT;
begin       integer i, j, k;
            real x;
            for j := m1 step 1 until m2 do
            for i := n step -1 until 1 do
            begin x := dl[i] × z[i, j];
                  for k := i-1 step -1 until 1 do
                  x := x + b[i, k] × z[k, j];
                  z[i, j] := x
            end ij
end rebakb;
```

5. Organisational and Notational Details

In *reduc 1* and *reduc 2* the symmetric matrices A and B are stored in $n\times n$ arrays though only the upper-triangles need be stored. The procedures *choldet 1* and *cholsol 1* [4] could have been used but since we wish to take full advantage of symmetry in computing the derived symmetric matrices special purpose procedures are preferable.

The matrix L such that $LL^T=B$ is determined in either case. Its strict lower-triangle is stored in the strict lower-triangle of B and its diagonal elements are stored in the linear array dl. Hence full information on B is retained. Since the factorization of B may already be known (if, for example several matrices A are associated with a fixed matrix B) provision is made for omitting the Cholesky factorization by inputting $-n$ instead of n.

In *reduc 1* the matrix $P=L^{-1}AL^{-T}$ is formed in the two steps

$$LX=A, \qquad PL^T=X. \tag{8}$$

The matrix X is not symmetric but only half of it is required for the computation of P. The transpose of the upper-triangle of X is computed and written on the lower-triangle of A. The diagonal of A is therefore destroyed and must be copied before using *reduc 1* if the residuals or Rayleigh quotients are required. The lower triangle of P is then determined and overwritten on X (see pp. 337–340 of ref. [7]).

Similarly in *reduc 2* the matrix $Q=L^T A L$ is computed in the steps

$$Y=AL, \quad Q=L^T Y. \tag{9}$$

Here only the lower-triangle of Y is required (although it is not symmetric). It is overwritten on the lower-triangle of A. The lower-triangle of Q is then formed and overwritten on Y. Again the diagonal of A is lost and a copy must be made independently if the residuals or Rayleigh quotients are required.

The derived standard symmetric eigenproblem may be solved in a variety of ways and the algorithms are given elsewhere in this series [1, 3, 5, 6]. Assuming eigenvectors of the derived problem have been found by some method and are normalized according to the Euclidean norm, then the corresponding eigenvectors of the original problem may be obtained using *rebak a* or *rebak b*, as appropriate (see 2).

6. Discussion of Numerical Properties

In general, the accuracy of the procedures described here is very high. The computed $\bar{L}$ satisfies

$$\bar{L}\bar{L}^T=B+E \tag{10}$$

and we have discussed the bounds for E in [4]. The errors made in computing $\bar{L}A\bar{L}^T$ or $\bar{L}^{-1}A\bar{L}^{-T}$ are comparatively unimportant. In general the errors in the eigenvalues are such as could be produced by very small perturbations in A and B. However, with the eigenproblem $Ax=\lambda Bx$ small perturbations in A and B can correspond to quite large perturbations in the eigenvalues if B is ill-conditioned with respect to inversion and this can lead to the inaccurate determination of eigenvalues.

It might be felt that in this case such a loss of accuracy is inherent in the data but this is not always true. Consider, for example, the case

$$A=\begin{bmatrix}2 & 2\\ 2 & 1\end{bmatrix}, \quad B=\begin{bmatrix}1 & 2\\ 2 & 4+\varepsilon\end{bmatrix} \tag{11}$$

for which the eigenvalues are the roots of

$$-2-\lambda(1+2\varepsilon)+\varepsilon\lambda^2=0$$

i.e.

$$\lambda=[+(1+2\varepsilon)\pm(1+12\varepsilon+4\varepsilon^2)^{\frac{1}{2}}]/2\varepsilon. \tag{12}$$

As $\varepsilon\to 0$ one root tends to -2 and the other to $+\infty$. It may be readily verified that the root -2 is not at all sensitive to small random changes in the elements of A and B.

Rather surprisingly, well-conditioned eigenvalues such as the -2 of the previous example are often given quite accurately, in spite of the ill-condition of B. In the above example for instance the true matrix $P=L^{-1}AL^{-T}$ is

$$\begin{bmatrix}2 & -2/\varepsilon^{\frac{1}{2}}\\ -2/\varepsilon^{\frac{1}{2}} & 1/\varepsilon\end{bmatrix}, \tag{13}$$

and for small ε it has some very large elements. When rounding errors are involved one might easily obtain a matrix $\bar{P}$ of the form

$$\bar{P}=\begin{bmatrix} 2/(1+\varepsilon_1) & -2(1+\varepsilon_2)/(\varepsilon+\varepsilon_3)^{\frac{1}{2}} \\ -2(1+\varepsilon_2)/(\varepsilon+\varepsilon_3)^{\frac{1}{2}} & (1+\varepsilon_4)/(\varepsilon+\varepsilon_3) \end{bmatrix} \tag{14}$$

where the ε_i are of the order of machine precision. If ε itself is of this order the computed $\bar{P}$ will differ substantially from the true matrix and the large elements may well have a high relative error. However $\bar{P}$ has an eigenvalue close to -2 for all relevant values of the ε_i.

As regards the eigenvectors, there are procedures which give an eigensystem of the derived problem which is accurately orthogonal. However, when transforming back to the eigensystem of the original eigenproblem, the computed set of vectors will be ill-conditioned when B (and hence L) is ill-conditioned. This deterioration would appear to be inherent in the data.

In our experience the considerations have not been important in practice since the matrices B which arise are, in general, extremely well-conditioned.

7. Test Results

To give a formal test of the procedures described here the matrices

$$F=\begin{bmatrix} 10 & 2 & 3 & 1 & 1 \\ 2 & 12 & 1 & 2 & 1 \\ 3 & 1 & 11 & 1 & -1 \\ 1 & 2 & 1 & 9 & 1 \\ 1 & 1 & -1 & 1 & 15 \end{bmatrix}, \quad G=\begin{bmatrix} 12 & 1 & -1 & 2 & 1 \\ 1 & 14 & 1 & -1 & 1 \\ -1 & 1 & 16 & -1 & 1 \\ 2 & -1 & -1 & 12 & -1 \\ 1 & 1 & 1 & -1 & 11 \end{bmatrix}$$

were used. Since both F and G are positive definite either matrix can be used as A or B in either of the main procedures. When used in conjunction with procedures for solving the standard eigenproblems this gives quite extensive checks on the overall performance. Since both A and B are well-conditioned with respect to inversion good agreement should be achieved between computed matrices $L^{-1}AL^{-T}$, L^TAL, eigenvalues and eigenvectors obtained on different computers.

A selection from the results obtained on KDF 9, a computer using a 39 binary digit mantissa are given in Tables 1–4. The eigensystems of the derived standard eigenproblems were found using the procedures *tred 1* [3], *bisect* [1], *tridi inverse iteration* [6] and *rebak 1* [3]. Any accurate procedures for dealing with the standard eigenproblem could be used including the procedure *jacobi* [5] already published.

Table 1 gives the output matrices a obtained from *reduc 1* and *reduc 2* taking $A=F$, $B=G$.

Table 2 gives the output matrices b and vectors dl obtained from *reduc 1* and *reduc 2* taking $A=F$, $B=G$. Since the computations involved in the two procedures are identical in every detail the output matrices are identical.

Table 3 gives the eigenvalues of $F-\lambda G$ and those of $G-\lambda F$ obtained using *reduc 1*; these involve the Cholesky factorizations of G and F respectively. The product of corresponding eigenvalues is equal to unity to about eleven decimal places. It also gives the eigenvalues of $FG-\lambda I$ and $GF-\lambda I$ calculated using

Table 1

Column 1	*Column 2*	*Column 3*	*Column 4*	*Column 5*
		Output matrix a from reduc 1 with $A=F$, $B=G$		
$+8.3333\,3333\,334_{10}-1$;	$+2.0000\,0000\,000_{10}+0$;	$+3.0000\,0000\,000_{10}+0$;	$+1.0000\,0000\,000_{10}+0$;	$+1.0000\,0000\,000_{10}+0$;
$+9.0279\,3771\,290_{10}-2$;	$+8.4331\,3373\,256_{10}-1$;	$+1.0000\,0000\,000_{10}+0$;	$+2.0000\,0000\,000_{10}+0$;	$+1.0000\,0000\,000_{10}+0$;
$+2.7151\,9129\,955_{10}-1$;	$-4.4710\,9421\,464_{10}-3$;	$+7.2690\,8512\,779_{10}-1$;	$+1.0000\,0000\,000_{10}+0$;	$-1.0000\,0000\,000_{10}+0$;
$-3.3434\,1703\,880_{10}-2$;	$+2.1334\,5628\,682_{10}-1$;	$+6.2497\,8723\,923_{10}-2$;	$+8.1234\,1947\,476_{10}-1$;	$+1.0000\,0000\,000_{10}+0$;
$-1.6323\,2422\,021_{10}-2$;	$+2.4171\,4784\,237_{10}-2$;	$-1.5020\,8557\,974_{10}-1$;	$+1.3606\,1578\,747_{10}-1$;	$+1.4260\,4956\,979_{10}+0$;
		Output matrix a from reduc 2 with $A=F$, $B=G$		
$+1.3100\,0000\,000_{10}+2$;	$+2.0000\,0000\,000_{10}+0$;	$+3.0000\,0000\,000_{10}+0$;	$+1.0000\,0000\,000_{10}+0$;	$+1.0000\,0000\,000_{10}+0$;
$+4.4649\,6005\,171_{10}+1$;	$+1.6856\,8862\,275_{10}+2$;	$+1.0000\,0000\,000_{10}+0$;	$+2.0000\,0000\,000_{10}+0$;	$+1.0000\,0000\,000_{10}+0$;
$+3.1565\,2935\,415_{10}+1$;	$+2.6097\,6424\,615_{10}+1$;	$+1.7183\,5449\,373_{10}+2$;	$+1.0000\,0000\,000_{10}+0$;	$-1.0000\,0000\,000_{10}+0$;
$+2.8625\,9509\,072_{10}+1$;	$+1.5509\,5804\,016_{10}+1$;	$+8.7809\,0481\,507_{10}+0$;	$+1.0313\,4644\,279_{10}+2$;	$+1.0000\,0000\,000_{10}+0$;
$+2.9269\,1592\,858_{10}+1$;	$+2.2283\,8464\,931_{10}+1$;	$-1.1468\,4872\,835_{10}+0$;	$-3.9498\,1287\,698_{10}+0$;	$+1.6046\,1044\,072_{10}+2$;

Table 2

		Output matrix b from reduc 1 and reduc 2 with $A=F$, $B=G$		
$+1.2000\,0000\,000_{10}+1$;	$+1.0000\,0000\,000_{10}+0$;	$-1.0000\,0000\,000_{10}+0$;	$+2.0000\,0000\,000_{10}+0$;	$+1.0000\,0000\,000_{10}+0$;
$+2.8867\,5134\,595_{10}-1$;	$+1.4000\,0000\,000_{10}+1$;	$+1.0000\,0000\,000_{10}+0$;	$-1.0000\,0000\,000_{10}+0$;	$+1.0000\,0000\,000_{10}+0$;
$-2.8867\,5134\,595_{10}-1$;	$+2.9039\,8583\,546_{10}-1$;	$+1.6000\,0000\,000_{10}+1$;	$-1.0000\,0000\,000_{10}+0$;	$+1.0000\,0000\,000_{10}+0$;
$+5.7735\,0269\,190_{10}-1$;	$-3.1273\,6936\,127_{10}-1$;	$-1.8660\,9059\,553_{10}-1$;	$+1.2000\,0000\,000_{10}+1$;	$-1.0000\,0000\,000_{10}+0$;
$+2.8867\,5134\,595_{10}-1$;	$+2.4572\,1878\,385_{10}-1$;	$+2.5433\,0089\,229_{10}-1$;	$-3.0692\,1313\,649_{10}-1$;	$+1.1000\,0000\,000_{10}+1$;

Output vector dl from reduc 1 and reduc 2 with $A=F$, $B=G$

$+3.4641\,0161\,514_{10}+0$;
$+3.7305\,0488\,093_{10}+0$;
$+3.9789\,8672\,143_{10}+0$;
$+3.3961\,8010\,923_{10}+0$;
$+3.2706\,8845\,017_{10}+0$;

Table 3. *Eigenvalues*

$A-\lambda B,\ A=F,\ B=G$	$A-\lambda B,\ A=G,\ B=F$	$AB-\lambda I,\ A=F,\ B=G$	$AB-\lambda I,\ A=G,\ B=F$
$+4.3278\,7211\,020_{10}-1$;	$+6.7008\,2644\,107_{10}-1$;	$+7.7697\,1911\,953_{10}+1$;	$+7.7697\,1911\,963_{10}+1$;
$+6.6366\,2748\,402_{10}-1$;	$+9.0148\,1958\,801_{10}-1$;	$+1.1215\,4193\,247_{10}+2$;	$+1.1215\,4193\,246_{10}+2$;
$+9.4385\,9004\,670_{10}-1$;	$+1.0594\,8027\,732_{10}-0$;	$+1.3468\,6463\,320_{10}+2$;	$+1.3468\,6463\,320_{10}+2$;
$+1.1092\,8454\,002_{10}+0$;	$+1.5067\,8940\,837_{10}+0$;	$+1.6748\,4878\,917_{10}+2$;	$+1.6748\,4878\,915_{10}+2$;
$+1.4923\,5323\,254_{10}+0$;	$+2.3106\,0432\,137_{10}+0$;	$+2.4297\,7273\,320_{10}+2$;	$+2.4297\,7273\,319_{10}+2$;

Table 4. *Eigenvectors*

1st eigenvector of $F-\lambda G$	*last eigenvector of* $G-\lambda F$	*1st eigenvector of* $FG-\lambda I$ *via* $G=LL^T$	*1st eigenvector of* $FG-\lambda I$ *via* $F=LL^T$
$+1.3459\,0573\,962_{10}-1$;	$-2.0458\,6718\,183_{10}-1$;	$+2.3491\,1413\,526_{10}-1$;	$+2.0706\,5038\,597_{10}+0$;
$-6.1294\,7224\,718_{10}-2$;	$+9.3172\,0977\,419_{10}-2$;	$-4.1091\,5167\,469_{10}-2$;	$-3.6220\,5325\,515_{10}-1$;
$-1.5790\,2562\,211_{10}-1$;	$+2.4002\,2507\,111_{10}-1$;	$-3.8307\,5945\,797_{10}-2$;	$-3.3766\,6162\,397_{10}-1$;
$+1.0946\,5787\,725_{10}-1$;	$-1.6639\,5354\,480_{10}-1$;	$-2.0590\,0367\,490_{10}-1$;	$-1.8149\,2958\,995_{10}+0$;
$-4.1473\,0117\,966_{10}-2$;	$+6.3041\,7653\,099_{10}-2$;	$-7.3470\,7965\,853_{10}-2$;	$-6.4761\,5758\,762_{10}-1$;

reduc 2 and again using the Cholesky factorization of G and F respectively. Corresponding eigenvalues have a maximum disagreement of 1 in the eleventh decimal.

Table 4 gives the first eigenvector of $F - \lambda G$ and the last eigenvector of $G - \lambda F$. These should be the same apart from the normalization. The first is normalized so that $x^T G x = 1$ and the second so that $x^T F x = 1$. The angle between the vectors is smaller than 10^{-11} radians. It also gives the first eigenvector of $FG - \lambda I$ obtained via the factorization of G and via the factorization of F. Again the angle between the two vectors is smaller than 10^{-11} radians. These results are typical of those obtained for all five eigenvectors in each case.

The complete sets of results were subjected to extensive cross checking and proved generally to be "best possible" for a 39 digit binary computer. A second test was carried out by C. REINSCH at the Technische Hochschule, München, and confirmed the above results.

Acknowledgements. The authors wish to thank Dr. C. REINSCH of the Mathematisches Institut der Technischen Hochschule, München, for testing these algorithms and suggesting several important modifications.

The work described above has been carried out at the National Physical Laboratory.

References

1. BARTH, W., R. S. MARTIN, and J. H. WILKINSON: Calculation of the eigenvalues of a symmetric tridiagonal matrix by the method of bisection. Numer. Math. **9**, 386—393 (1967). Cf. II/5.
2. BOWDLER, H., R. S. MARTIN, C. REINSCH, and J. H. WILKINSON: The *QR* and *QL* algorithms for symmetric matrices. Numer. Math. **11**, 293—306 (1968). Cf. II/3.
3. MARTIN, R. S., C. REINSCH, and J. H. WILKINSON: Householder's tridiagonalization of a symmetric matrix. Numer. Math. **11**, 181—195 (1968). Cf. II/2.
4. — G. PETERS, and J. H. WILKINSON: Symmetric decomposition of a positive definite matrix. Numer. Math. **7**, 362—383 (1965). Cf. I/1.
5. RUTISHAUSER, H.: The Jacobi method for real symmetric matrices. Numer. Math. **9**, 1—10 (1966). Cf. II/1.
6. Wilkinson, J. H.: Calculation of the eigenvectors of a symmetric tridiagonal matrix by inverse iteration. Numer. Math. **4**, 368—376 (1962). See also: PETERS, G., and J. H. WILKINSON: The calculation of specified eigenvectors by inverse iteration. Cf. II/18.
7. — The algebraic eigenvalue problem. London: Oxford University Press 1965.