NAG Library Routine Document

F08TAF (DSPGV)

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

F08TAF (DSPGV) computes all the eigenvalues and, optionally, all the eigenvectors of a real generalized symmetric-definite eigenproblem, of the form

$$Az = \lambda Bz, \qquad ABz = \lambda z \qquad \text{or} \qquad BAz = \lambda z,$$

where A and B are symmetric, stored in packed format, and B is also positive definite.

2 Specification

```
SUBROUTINE F08TAF (ITYPE, JOBZ, UPLO, N, AP, BP, W, Z, LDZ, WORK, INFO)

INTEGER              ITYPE, N, LDZ, INFO
REAL (KIND=nag_wp) AP(*), BP(*), W(N), Z(LDZ,*), WORK(3*N)
CHARACTER(1)         JOBZ, UPLO
```

The routine may be called by its LAPACK name ***dspgv***.

3 Description

F08TAF (DSPGV) first performs a Cholesky factorization of the matrix B as $B = U^{\mathrm{T}}U$, when UPLO = 'U' or $B = LL^{\mathrm{T}}$, when UPLO = 'L'. The generalized problem is then reduced to a standard symmetric eigenvalue problem

$$Cx = \lambda x,$$

which is solved for the eigenvalues and, optionally, the eigenvectors; the eigenvectors are then backtransformed to give the eigenvectors of the original problem.

For the problem $Az = \lambda Bz$, the eigenvectors are normalized so that the matrix of eigenvectors, Z, satisfies

$$Z^{\mathrm{T}}AZ = \Lambda \qquad \text{and} \qquad Z^{\mathrm{T}}BZ = I,$$

where Λ is the diagonal matrix whose diagonal elements are the eigenvalues. For the problem $ABz = \lambda z$ we correspondingly have

$$Z^{-1}AZ^{-\mathrm{T}} = \Lambda \qquad \text{and} \qquad Z^{\mathrm{T}}BZ = I,$$

and for $BAz = \lambda z$ we have

$$Z^{\mathrm{T}}AZ = \Lambda \qquad \text{and} \qquad Z^{\mathrm{T}}B^{-1}Z = I.$$

4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D (1999) *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia http://www.netlib.org/lapack/lug

Golub G H and Van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

1: ITYPE – INTEGER *Input*

On entry: specifies the problem type to be solved.

ITYPE = 1
$Az = \lambda Bz$.

ITYPE = 2
$ABz = \lambda z$.

ITYPE = 3
$BAz = \lambda z$.

Constraint: ITYPE = 1, 2 or 3.

2: JOBZ – CHARACTER(1) *Input*

On entry: indicates whether eigenvectors are computed.

JOBZ = 'N'
Only eigenvalues are computed.

JOBZ = 'V'
Eigenvalues and eigenvectors are computed.

Constraint: JOBZ = 'N' or 'V'.

3: UPLO – CHARACTER(1) *Input*

On entry: if UPLO = 'U', the upper triangles of A and B are stored.

If UPLO = 'L', the lower triangles of A and B are stored.

Constraint: UPLO = 'U' or 'L'.

4: N – INTEGER *Input*

On entry: n, the order of the matrices A and B.

Constraint: N ≥ 0.

5: AP($*$) – REAL (KIND=nag_wp) array *Input/Output*

Note: the dimension of the array AP must be at least $\max(1, \mathrm{N} \times (\mathrm{N}+1)/2)$.

On entry: the upper or lower triangle of the n by n symmetric matrix A, packed by columns.

More precisely,

if UPLO = 'U', the upper triangle of A must be stored with element A_{ij} in AP$(i + j(j-1)/2)$ for $i \leq j$;

if UPLO = 'L', the lower triangle of A must be stored with element A_{ij} in AP$(i + (2n - j)(j-1)/2)$ for $i \geq j$.

On exit: the contents of AP are destroyed.

6: BP($*$) – REAL (KIND=nag_wp) array *Input/Output*

Note: the dimension of the array BP must be at least $\max(1, \mathrm{N} \times (\mathrm{N}+1)/2)$.

On entry: the upper or lower triangle of the n by n symmetric matrix B, packed by columns.

More precisely,

if UPLO = 'U', the upper triangle of B must be stored with element B_{ij} in BP$(i + j(j-1)/2)$ for $i \leq j$;

if UPLO = 'L', the lower triangle of B must be stored with element B_{ij} in BP$(i + (2n-j)(j-1)/2)$ for $i \geq j$.

On exit: the triangular factor U or L from the Cholesky factorization $B = U^{\mathrm{T}}U$ or $B = LL^{\mathrm{T}}$, in the same storage format as B.

7: W(N) – REAL (KIND=nag_wp) array *Output*

On exit: the eigenvalues in ascending order.

8: Z(LDZ,$*$) – REAL (KIND=nag_wp) array *Output*

Note: the second dimension of the array Z must be at least $\max(1, \mathrm{N})$ if JOBZ = 'V', and at least 1 otherwise.

On exit: if JOBZ = 'V', Z contains the matrix Z of eigenvectors. The eigenvectors are normalized as follows:

if ITYPE $= 1$ or 2, $Z^{\mathrm{T}}BZ = I$;

if ITYPE $= 3$, $Z^{\mathrm{T}}B^{-1}Z = I$.

If JOBZ = 'N', Z is not referenced.

9: LDZ – INTEGER *Input*

On entry: the first dimension of the array Z as declared in the (sub)program from which F08TAF (DSPGV) is called.

Constraints:

if JOBZ = 'V', LDZ $\geq \max(1, \mathrm{N})$;
otherwise LDZ ≥ 1.

10: WORK($3 \times \mathrm{N}$) – REAL (KIND=nag_wp) array *Workspace*

11: INFO – INTEGER *Output*

On exit: INFO $= 0$ unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

INFO < 0

If INFO $= -i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

INFO > 0

F07GDF (DPPTRF) or F08GAF (DSPEV) returned an error code:

$\leq$ N if INFO $= i$, F08GAF (DSPEV) failed to converge; i off-diagonal elements of an intermediate tridiagonal form did not converge to zero;

$>$ N if INFO $= \mathrm{N} + i$, for $1 \leq i \leq \mathrm{N}$, then the leading minor of order i of B is not positive definite. The factorization of B could not be completed and no eigenvalues or eigenvectors were computed.

7 Accuracy

If B is ill-conditioned with respect to inversion, then the error bounds for the computed eigenvalues and vectors may be large, although when the diagonal elements of B differ widely in magnitude the eigenvalues and eigenvectors may be less sensitive than the condition of B would suggest. See Section 4.10 of Anderson *et al.* (1999) for details of the error bounds.

The example program below illustrates the computation of approximate error bounds.

8 Further Comments

The total number of floating point operations is proportional to n^3.

The complex analogue of this routine is F08TNF (ZHPGV).

9 Example

This example finds all the eigenvalues and eigenvectors of the generalized symmetric eigenproblem $Az = \lambda Bz$, where

$$A = \begin{pmatrix} 0.24 & 0.39 & 0.42 & -0.16 \\ 0.39 & -0.11 & 0.79 & 0.63 \\ 0.42 & 0.79 & -0.25 & 0.48 \\ -0.16 & 0.63 & 0.48 & -0.03 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 4.16 & -3.12 & 0.56 & -0.10 \\ -3.12 & 5.03 & -0.83 & 1.09 \\ 0.56 & -0.83 & 0.76 & 0.34 \\ -0.10 & 1.09 & 0.34 & 1.18 \end{pmatrix},$$

together with an estimate of the condition number of B, and approximate error bounds for the computed eigenvalues and eigenvectors.

The example program for F08TCF (DSPGVD) illustrates solving a generalized symmetric eigenproblem of the form $ABz = \lambda z$.

9.1 Program Text

```
    Program f08tafe

!     F08TAF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: dspgv, dtpcon, f06rdf, nag_wp, x02ajf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
      Character (1), Parameter         :: uplo = 'U'
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: anorm, bnorm, eps, rcond, rcondb,    &
                                          t1, t2
      Integer                          :: i, info, j, n
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: ap(:), bp(:), eerbnd(:), w(:), work(:)
      Real (Kind=nag_wp)               :: dummy(1,1)
      Integer, Allocatable             :: iwork(:)
!     .. Intrinsic Procedures ..
      Intrinsic                        :: abs
!     .. Executable Statements ..
      Write (nout,*) 'F08TAF Example Program Results'
      Write (nout,*)
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n

      Allocate (ap((n*(n+1))/2),bp((n*(n+1))/2),eerbnd(n),w(n),work(3*n),iwork &
        (n))
```

```
!     Read the upper or lower triangular parts of the matrices A and
!     B from data file

      If (uplo=='U') Then
        Read (nin,*)((ap(i+(j*(j-1))/2),j=i,n),i=1,n)
        Read (nin,*)((bp(i+(j*(j-1))/2),j=i,n),i=1,n)
      Else If (uplo=='L') Then
        Read (nin,*)((ap(i+((2*n-j)*(j-1))/2),j=1,i),i=1,n)
        Read (nin,*)((bp(i+((2*n-j)*(j-1))/2),j=1,i),i=1,n)
      End If

!     Compute the one-norms of the symmetric matrices A and B

      anorm = f06rdf('One norm',uplo,n,ap,work)
      bnorm = f06rdf('One norm',uplo,n,bp,work)

!     Solve the generalized symmetric eigenvalue problem
!     A*x = lambda*B*x (itype = 1)

!     The NAG name equivalent of dspgv is f08taf
      Call dspgv(1,'No vectors',uplo,n,ap,bp,w,dummy,1,work,info)

      If (info==0) Then

!       Print solution

        Write (nout,*) 'Eigenvalues'
        Write (nout,99999) w(1:n)

!       Call DTPCON (F07UGF) to estimate the reciprocal condition
!       number of the Cholesky factor of B.  Note that:
!       cond(B) = 1/rcond**2

        Call dtpcon('One norm',uplo,'Non-unit',n,bp,rcond,work,iwork,info)

!       Print the reciprocal condition number of B

        rcondb = rcond**2
        Write (nout,*)
        Write (nout,*) 'Estimate of reciprocal condition number for B'
        Write (nout,99998) rcondb

!       Get the machine precision, eps, and if rcondb is not less
!       than eps**2, compute error estimates for the eigenvalues

        eps = x02ajf()
        If (rcond>=eps) Then
          t1 = eps/rcondb
          t2 = anorm/bnorm
          Do i = 1, n
            eerbnd(i) = t1*(t2+abs(w(i)))
          End Do

!         Print the approximate error bounds for the eigenvalues

          Write (nout,*)
          Write (nout,*) 'Error estimates for the eigenvalues'
          Write (nout,99998) eerbnd(1:n)
        Else
          Write (nout,*)
          Write (nout,*) 'B is very ill-conditioned, error ', &
            'estimates have not been computed'
        End If
      Else If (info>n .And. info<=2*n) Then
        i = info - n
        Write (nout,99997) 'The leading minor of order ', i, &
          ' of B is not positive definite'
      Else
        Write (nout,99996) 'Failure in DSPGV. INFO =', info
      End If
```

```
99999 Format (3X,(6F11.4))
99998 Format (4X,1P,6E11.1)
99997 Format (1X,A,I4,A)
99996 Format (1X,A,I4)
    End Program f08tafe
```

9.2 Program Data

```
F08TAF Example Program Data

  4                         :Value of N

  0.24   0.39   0.42  -0.16
        -0.11   0.79   0.63
               -0.25   0.48
                      -0.03 :End of matrix A

  4.16  -3.12   0.56  -0.10
         5.03  -0.83   1.09
                0.76   0.34
                       1.18 :End of matrix B
```

9.3 Program Results

```
 F08TAF Example Program Results

 Eigenvalues
       -2.2254    -0.4548     0.1001     1.1270

 Estimate of reciprocal condition number for B
        5.8E-03

 Error estimates for the eigenvalues
        4.7E-14    1.2E-14    5.6E-15    2.5E-14
```
