# 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
double precision    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 $\Lambda$ 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, Philadelphiahttp://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 $= 3$
$BAz = \lambda z$.

2: JOBZ – CHARACTER*1 *Input*

*On entry*: if JOBZ = 'N', compute eigenvalues only.

If JOBZ = 'V', compute eigenvalues and eigenvectors.

*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*: $\mathrm{N} \geq 0$.

5: AP$(*)$ – ***double precision*** 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 $n$ 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 $\mathrm{AP}(i + j(j-1)/2)$ for $i \leq j$;

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

*On exit*: the contents of AP are destroyed.

6: BP$(*)$ – ***double precision*** 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 symmetric matrix $B$, packed by columns.

More precisely,

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

if UPLO = 'L', the lower triangle of $B$ must be stored with element $B_{ij}$ in $\mathrm{BP}(i + (2n-j)(j-1)/2)$ 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) – ***double precision*** array *Output*

*On exit*: if INFO $= 0$, the eigenvalues in ascending order.

8: Z(LDZ,$*$) – ***double precision*** array *Output*

**Note**: the second dimension of the array Z must be at least $\max(1, \mathrm{N})$.

*On exit*: if JOBZ = 'V', then if INFO $= 0$, 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 $\geq 1$.

10: WORK($3 \times \mathrm{N}$) – ***double precision*** 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

```
*     F08TAF Example Program Text
*     Mark 21 Release.  NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX
      PARAMETER        (NMAX=10)
      CHARACTER        UPLO
      PARAMETER        (UPLO='U')
*     .. Local Scalars ..
      DOUBLE PRECISION ANORM, BNORM, EPS, RCOND, RCONDB, T1, T2
      INTEGER          I, INFO, J, N
*     .. Local Arrays ..
      DOUBLE PRECISION AP((NMAX*(NMAX+1))/2), BP((NMAX*(NMAX+1))/2),
     +                 DUMMY(1,1), EERBND(NMAX), W(NMAX), WORK(3*NMAX)
      INTEGER          IWORK(NMAX)
*     .. External Functions ..
      DOUBLE PRECISION F06RDF, X02AJF
      EXTERNAL         F06RDF, X02AJF
*     .. External Subroutines ..
      EXTERNAL         DSPGV, DTPCON
*     .. Intrinsic Functions ..
      INTRINSIC        ABS
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F08TAF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
*
*        Read the upper or lower triangular parts of the matrices A and
*        B from data file
*
         IF (UPLO.EQ.'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.EQ.'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)
*
         CALL DSPGV(1,'No vectors',UPLO,N,AP,BP,W,DUMMY,1,WORK,INFO)
*
         IF (INFO.EQ.0) THEN
*
*           Print solution
*
            WRITE (NOUT,*) 'Eigenvalues'
            WRITE (NOUT,99999) (W(J),J=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.GE.EPS) THEN
               T1 = EPS/RCONDB
               T2 = ANORM/BNORM
               DO 20 I = 1, N
                  EERBND(I) = T1*(T2+ABS(W(I)))
   20          CONTINUE
*
*              Print the approximate error bounds for the eigenvalues
*
               WRITE (NOUT,*)
               WRITE (NOUT,*) 'Error estimates for the eigenvalues'
               WRITE (NOUT,99998) (EERBND(I),I=1,N)
            ELSE
               WRITE (NOUT,*)
               WRITE (NOUT,*) 'B is very ill-conditioned, error ',
     +           'estimates have not been computed'
            END IF
         ELSE IF (INFO.GT.N .AND. INFO.LE.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
      ELSE
         WRITE (NOUT,*) 'NMAX too small'
      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
```

### 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
```

---