NAG Library Routine Document

F08PAF (DGEES)

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

F08PAF (DGEES) computes the eigenvalues, the real Schur form T, and, optionally, the matrix of Schur vectors Z for an n by n real nonsymmetric matrix A.

2 Specification

```
SUBROUTINE F08PAF (JOBVS, SORT, SELECT, N, A, LDA, SDIM, WR, WI, VS, LDVS,      &
                   WORK, LWORK, BWORK, INFO)

INTEGER            N, LDA, SDIM, LDVS, LWORK, INFO
REAL (KIND=nag_wp) A(LDA,*), WR(*), WI(*), VS(LDVS,*), WORK(max(1,LWORK))
LOGICAL            SELECT, BWORK(*)
CHARACTER(1)       JOBVS, SORT
EXTERNAL           SELECT
```

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

3 Description

The real Schur factorization of A is given by

$$A = ZTZ^{\mathrm{T}},$$

where Z, the matrix of Schur vectors, is orthogonal and T is the real Schur form. A matrix is in real Schur form if it is upper quasi-triangular with 1 by 1 and 2 by 2 blocks. 2 by 2 blocks will be standardized in the form

$$\begin{bmatrix} a & b \\ c & a \end{bmatrix}$$

where $bc < 0$. The eigenvalues of such a block are $a \pm \sqrt{bc}$.

Optionally, F08PAF (DGEES) also orders the eigenvalues on the diagonal of the real Schur form so that selected eigenvalues are at the top left. The leading columns of Z form an orthonormal basis for the invariant subspace corresponding to the selected eigenvalues.

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: JOBVS – CHARACTER(1) *Input*

On entry: if JOBVS = 'N', Schur vectors are not computed.

If JOBVS = 'V', Schur vectors are computed.

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

2: SORT – CHARACTER(1) *Input*

On entry: specifies whether or not to order the eigenvalues on the diagonal of the Schur form.

SORT = 'N'
Eigenvalues are not ordered.

SORT = 'S'
Eigenvalues are ordered (see SELECT).

Constraint: SORT = 'N' or 'S'.

3: SELECT – LOGICAL FUNCTION, supplied by the user. *External Procedure*

If SORT = 'S', SELECT is used to select eigenvalues to sort to the top left of the Schur form.

If SORT = 'N', SELECT is not referenced and F08PAF (DGEES) may be called with the dummy function F08PAZ.

An eigenvalue $\mathrm{WR}(j) + \sqrt{-1} \times \mathrm{WI}(j)$ is selected if $\mathrm{SELECT}(\mathrm{WR}(j), \mathrm{WI}(j))$ is .TRUE.. If either one of a complex conjugate pair of eigenvalues is selected, then both are. Note that a selected complex eigenvalue may no longer satisfy $\mathrm{SELECT}(\mathrm{WR}(j), \mathrm{WI}(j)) =$ .TRUE. after ordering, since ordering may change the value of complex eigenvalues (especially if the eigenvalue is ill-conditioned); in this case INFO is set to $\mathrm{N} + 2$ (see INFO below).

The specification of SELECT is:

```
FUNCTION SELECT (WR, WI)
LOGICAL SELECT
REAL (KIND=nag_wp) WR, WI
```

1: WR – REAL (KIND=nag_wp) *Input*
2: WI – REAL (KIND=nag_wp) *Input*

On entry: the real and imaginary parts of the eigenvalue.

SELECT must either be a module subprogram USEd by, or declared as EXTERNAL in, the (sub)program from which F08PAF (DGEES) is called. Parameters denoted as *Input* must **not** be changed by this procedure.

4: N – INTEGER *Input*

On entry: n, the order of the matrix A.

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

5: A(LDA,$*$) – REAL (KIND=nag_wp) array *Input/Output*

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

On entry: the n by n matrix A.

On exit: A is overwritten by its real Schur form T.

6: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08PAF (DGEES) is called.

Constraint: $\mathrm{LDA} \geq \max(1, \mathrm{N})$.

7: SDIM – INTEGER *Output*

On exit: if SORT = 'N', SDIM = 0.

If SORT = 'S', SDIM = number of eigenvalues (after sorting) for which SELECT is .TRUE.. (Complex conjugate pairs for which SELECT is .TRUE. for either eigenvalue count as 2.)

8: WR($*$) – REAL (KIND=nag_wp) array *Output*

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

On exit: see the description of WI.

9: WI($*$) – REAL (KIND=nag_wp) array *Output*

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

On exit: WR and WI contain the real and imaginary parts, respectively, of the computed eigenvalues in the same order that they appear on the diagonal of the output Schur form T. Complex conjugate pairs of eigenvalues will appear consecutively with the eigenvalue having the positive imaginary part first.

10: VS(LDVS,$*$) – REAL (KIND=nag_wp) array *Output*

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

On exit: if JOBVS = 'V', VS contains the orthogonal matrix Z of Schur vectors.

If JOBVS = 'N', VS is not referenced.

11: LDVS – INTEGER *Input*

On entry: the first dimension of the array VS as declared in the (sub)program from which F08PAF (DGEES) is called.

Constraints:

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

12: WORK($\max(1,\mathrm{LWORK})$) – REAL (KIND=nag_wp) array *Workspace*

On exit: if INFO = 0, WORK(1) contains the minimum value of LWORK required for optimal performance.

13: LWORK – INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F08PAF (DGEES) is called.

If LWORK $= -1$, a workspace query is assumed; the routine only calculates the optimal size of the WORK array, returns this value as the first entry of the WORK array, and no error message related to LWORK is issued.

Suggested value: for optimal performance, LWORK must generally be larger than the minimum, say $3 \times \mathrm{N} + nb \times \mathrm{N}$, where nb is the optimal ***block size*** for F08NEF (DGEHRD)

Constraint: LWORK $\geq \max(1, 3 \times \mathrm{N})$.

14: BWORK($*$) – LOGICAL array *Workspace*

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

If SORT = 'N', BWORK is not referenced.

15: 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 $= 1$ to N

If INFO $= i$ and $i \leq$ N, the QR algorithm failed to compute all the eigenvalues.

INFO $=$ N $+ 1$

The eigenvalues could not be reordered because some eigenvalues were too close to separate (the problem is very ill-conditioned).

INFO $=$ N $+ 2$

After reordering, roundoff changed values of some complex eigenvalues so that leading eigenvalues in the Schur form no longer satisfy SELECT $=$ .TRUE.. This could also be caused by underflow due to scaling.

7 Accuracy

The computed Schur factorization satisfies

$$A + E = ZTZ^{\mathrm{T}},$$

where

$$\|E\|_2 = O(\epsilon)\|A\|_2,$$

and ϵ is the ***machine precision***. See Section 4.8 of Anderson *et al.* (1999) for further details.

8 Further Comments

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

The complex analogue of this routine is F08PNF (ZGEES).

9 Example

This example finds the Schur factorization of the matrix

$$A = \begin{pmatrix} 0.35 & 0.45 & -0.14 & -0.17 \\ 0.09 & 0.07 & -0.54 & 0.35 \\ -0.44 & -0.33 & -0.03 & 0.17 \\ 0.25 & -0.32 & -0.13 & 0.11 \end{pmatrix},$$

such that the real eigenvalues of A are the top left diagonal elements of the Schur form, T.

Note that the block size (NB) of 64 assumed in this example is not realistic for such a small problem, but should be suitable for large problems.

9.1 Program Text

```
!   F08PAF Example Program Text
!   Mark 24 Release. NAG Copyright 2012.

    Module f08pafe_mod

!     F08PAF Example Program Module:
!            Parameters and User-defined Routines

!     .. Use Statements ..
      Use nag_library, Only: nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter                   :: nb = 64, nin = 5, nout = 6
    Contains
      Function select(wr,wi)

!       Logical function select for use with DGEES (F08PAF)
!       Returns the value .TRUE. if the eigenvalue is real

!       .. Function Return Value ..
        Logical                              :: select
!       .. Scalar Arguments ..
        Real (Kind=nag_wp), Intent (In)      :: wi, wr
!       .. Local Scalars ..
        Logical                              :: d
!       .. Executable Statements ..
        If (wi==0.0_nag_wp) Then
          d = .True.
        Else
          d = .False.
        End If

        select = d

        Return
      End Function select
    End Module f08pafe_mod
    Program f08pafe

!     F08PAF Example Main Program

!     .. Use Statements ..
      Use nag_library, Only: dgees, dgemm, dlange => f06raf, nag_wp, x02ajf,   &
                             x04caf
      Use f08pafe_mod, Only: nb, nin, nout, select
!     .. Implicit None Statement ..
      Implicit None
!     .. Local Scalars ..
      Real (Kind=nag_wp)                   :: alpha, beta, norm
      Integer                              :: i, ifail, info, lda, ldc, ldd,   &
                                              ldvs, lwork, n, sdim
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable      :: a(:,:), c(:,:), d(:,:), vs(:,:), &
                                              wi(:), work(:), wr(:)
      Real (Kind=nag_wp)                   :: dummy(1), rwork(1)
      Logical, Allocatable                 :: bwork(:)
!     .. Intrinsic Procedures ..
      Intrinsic                            :: max, nint
!     .. Executable Statements ..
      Write (nout,*) 'F08PAF Example Program Results'
      Write (nout,*)
      Flush (nout)
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n
      lda = n
      ldc = n
      ldd = n
```

```
      ldvs = n
      Allocate (a(lda,n),c(ldc,n),d(ldd,n),vs(ldvs,n),wi(n),wr(n),bwork(n))

!     Use routine workspace query to get optimal workspace.
      lwork = -1
!     The NAG name equivalent of dgees is f08paf
      Call dgees('Vectors (Schur)','Sort',select,n,a,lda,sdim,wr,wi,vs,ldvs, &
        dummy,lwork,bwork,info)

!     Make sure that there is enough workspace for blocksize nb.
      lwork = max((nb+2)*n,nint(dummy(1)))
      Allocate (work(lwork))

!     Read the matrix A from data file
      Read (nin,*)(a(i,1:n),i=1,n)

!     Copy A into D
      d(1:n,1:n) = a(1:n,1:n)

!     Print Matrix A
!     ifail: behaviour on error exit
!             =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
      ifail = 0
      Call x04caf('General',' ',n,n,a,lda,'Matrix A',ifail)

      Write (nout,*)
      Flush (nout)

!     Find the Schur factorization
!     The NAG name equivalent of dgees is f08paf
      Call dgees('Vectors (Schur)','Sort',select,n,a,lda,sdim,wr,wi,vs,ldvs, &
        work,lwork,bwork,info)

      If (info==0 .Or. info==(n+2)) Then

!       Compute A - Z*T*Z^H from the factorization of A and store in matrix D
!       The NAG name equivelent of dgemm is f06yaf
        alpha = 1.0_nag_wp
        beta = 0.0_nag_wp
        Call dgemm('N','N',n,n,n,alpha,vs,ldvs,a,lda,beta,c,ldc)
        alpha = -1.0_nag_wp
        beta = 1.0_nag_wp
        Call dgemm('N','T',n,n,n,alpha,c,ldc,vs,ldvs,beta,d,ldd)

!       Find norm of matrix D and print warning if it is too large
!       f06raf is the NAG name equivalent of the LAPACK auxiliary dlange
        norm = dlange('O',ldd,n,d,ldd,rwork)
        If (norm>x02ajf()**0.8_nag_wp) Then
          Write (nout,*) 'Norm of A-(Z*T*Z^T) is much greater than 0.'
          Write (nout,*) 'Schur factorization has failed.'
        Else
!         Print eigenvalues.
          Write (nout,*) 'Eigenvalues'
          Write (nout,99998)(' (',wr(i),',',wi(i),')',i=1,n)
        End If

      Else
        Write (nout,99999) 'Failure in DGEES.  INFO = ', info
      End If

99999 Format (1X,A,I4)
99998 Format (1X,A,F8.4,A,F8.4,A)

    End Program f08pafe
```

9.2 Program Data

```
F08PAF Example Program Data

  4                          :Value of N

  0.35   0.45  -0.14  -0.17
  0.09   0.07  -0.54   0.35
 -0.44  -0.33  -0.03   0.17
  0.25  -0.32  -0.13   0.11 :End of matrix A
```

9.3 Program Results

```
 F08PAF Example Program Results

 Matrix A
          1       2       3       4
 1   0.3500  0.4500 -0.1400 -0.1700
 2   0.0900  0.0700 -0.5400  0.3500
 3  -0.4400 -0.3300 -0.0300  0.1700
 4   0.2500 -0.3200 -0.1300  0.1100

 Eigenvalues
  (  0.7995,  0.0000)
  ( -0.1007,  0.0000)
  ( -0.0994,  0.4008)
  ( -0.0994, -0.4008)
```