NAG Fortran Library Routine Document

F08FRF (ZHEEVR)

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

F08FRF (ZHEEVR) computes selected eigenvalues and, optionally, eigenvectors of a complex n by n Hermitian matrix A. Eigenvalues and eigenvectors can be selected by specifying either a range of values or a range of indices for the desired eigenvalues.

2 Specification

```
 SUBROUTINE F08FRF (JOBZ, RANGE, UPLO, N, A, LDA, VL, VU, IL, IU, ABSTOL,
1                   M, W, Z, LDZ, ISUPPZ, WORK, LWORK, RWORK, LRWORK,
2                   IWORK, LIWORK, INFO)
 INTEGER            N, LDA, IL, IU, M, LDZ, ISUPPZ(*), LWORK, LRWORK,
1                   IWORK(*), LIWORK, INFO
 double precision   VL, VU, ABSTOL, W(*), RWORK(*)
 complex*16         A(LDA,*), Z(LDZ,*), WORK(*)
 CHARACTER*1        JOBZ, RANGE, UPLO
```

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

3 Description

The Hermitian matrix is first reduced to a real tridiagonal matrix T, using unitary similarity transformations. Then whenever possible, F08FRF (ZHEEVR) computes the eigenspectrum using Relatively Robust Representations. F08FRF (ZHEEVR) computes eigenvalues by the dqds algorithm, while orthogonal eigenvectors are computed from various 'good' LDL^T representations (also known as Relatively Robust Representations). Gram–Schmidt orthogonalisation is avoided as far as possible. More specifically, the various steps of the algorithm are as follows. For the ith unreduced block of T,

(a) compute $T - \sigma_i I = L_i D_i {L_i}^T$, such that $L_i D_i {L_i}^T$ is a relatively robust representation,

(b) compute the eigenvalues, λ_j, of $L_i D_i {L_i}^T$ to high relative accuracy by the dqds algorithm,

(c) if there is a cluster of close eigenvalues, 'choose' σ_i close to the cluster, and go to step (a),

(d) given the approximate eigenvalue λ_j of $L_i D_i {L_i}^T$, compute the corresponding eigenvector by forming a rank-revealing twisted factorization.

The desired accuracy of the output can be specified by the input parameter ABSTOL. For more details, see Dhillon (1997) and Parlett and Dhillon (2000).

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 URL: http://www.netlib.org/lapack/lug

Barlow J and Demmel J W (1990) Computing accurate eigensystems of scaled diagonally dominant matrices *SIAM J. Numer. Anal.* **27** 762–791

Demmel J W and Kahan W (1990) Accurate singular values of bidiagonal matrices *SIAM J. Sci. Statist. Comput.* **11** 873–912

Dhillon I (1997) A new $O(n^2)$ algorithm for the symmetric tridiagonal eigenvalue/eigenvector problem *Computer Science Division Technical Report No. UCB//CSD-97-971* UC Berkeley

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

Parlett B N and Dhillon I S (2000) Relatively robust representations of symmetric tridiagonals *Linear Algebra Appl.* **309** 121–151

5 Parameters

1: JOBZ – CHARACTER*1 *Input*

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

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

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

2: RANGE – CHARACTER*1 *Input*

On entry: if RANGE = 'A', all eigenvalues will be found.

If RANGE = 'V', all eigenvalues in the half-open interval $(\mathrm{VL}, \mathrm{VU}]$ will be found.

If RANGE = 'I', the ILth to IUth eigenvalues will be found.

For RANGE = 'V' or 'I' and $\mathrm{IU} - \mathrm{IL} < \mathrm{N} - 1$, F08JJF (DSTEBZ) and F08JXF (ZSTEIN) are called.

3: UPLO – CHARACTER*1 *Input*

On entry: if UPLO = 'U', the upper triangle of A is stored.

If UPLO = 'L', the lower triangle of A is stored.

4: N – INTEGER *Input*

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

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

5: A(LDA,$*$) – ***complex*16*** 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 Hermitian matrix A.

If UPLO = 'U', the leading n by n upper triangular part of A contains the upper triangular part of the matrix A.

If UPLO = 'L', the leading n by n lower triangular part of A contains the lower triangular part of the matrix A.

On exit: the lower triangle (if UPLO = 'L') or the upper triangle (if UPLO = 'U') of A, including the diagonal, is destroyed.

6: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08FRF (ZHEEVR) is called.

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

7: VL – ***double precision*** *Input*
8: VU – ***double precision*** *Input*

On entry: if RANGE = 'V', the lower and upper bounds of the interval to be searched for eigenvalues.

Constraint: VL $<$ VU.

If RANGE = 'A' or 'I', VL and VU are not referenced.

9: IL – INTEGER *Input*
10: IU – INTEGER *Input*

On entry: if RANGE = 'I', the indices (in ascending order) of the smallest and largest eigenvalues to be returned.

If RANGE = 'A' or 'V', IL and IU are not referenced.

Constraints:

if N $= 0$, IL $= 1$ and IU $= 0$;
if N > 0, $1 \leq$ IL $\leq$ IU $\leq$ N.

11: ABSTOL – ***double precision*** *Input*

On entry: the absolute error tolerance for the eigenvalues. An approximate eigenvalue is accepted as converged when it is determined to lie in an interval $[a, b]$ of width less than or equal to

$$\text{ABSTOL} + \epsilon \max(|a|, |b|),$$

where ϵ is the ***machine precision***. If ABSTOL is less than or equal to zero, then $\epsilon \|T\|_1$ will be used in its place, where T is the tridiagonal matrix obtained by reducing A to tridiagonal form. See Demmel and Kahan (1990).

If high relative accuracy is important, set ABSTOL to X02AMF(), although doing so does not currently guarantee that eigenvalues are computed to high relative accuracy. See Barlow and Demmel (1990) for a discussion of which matrices can define their eigenvalues to high relative accuracy.

12: M – INTEGER *Output*

On exit: the total number of eigenvalues found.

If RANGE = 'A', M = N.

If RANGE = 'I', M = IU − IL + 1.

Constraint: $0 \leq$ M $\leq$ N.

13: W($*$) – ***double precision*** array *Output*

Note: the dimension of the array W must be at least max(1, N).

On exit: the first M elements contain the selected eigenvalues in ascending order.

14: Z(LDZ,$*$) – ***complex*16*** array *Output*

Note: the second dimension of the array Z must be at least max(1, M).

On exit: if JOBZ = 'V', then if INFO $= 0$, the first m columns of Z contain the orthonormal eigenvectors of the matrix A corresponding to the selected eigenvalues, with the ith column of Z holding the eigenvector associated with W(i).

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

Note: the user must ensure that at least max(1, M) columns are supplied in the array Z; if RANGE = 'V', the exact value of M is not known in advance and an upper bound must be used.

15: LDZ – INTEGER *Input*

On entry: the first dimension of the array Z as declared in the (sub)program from which F08FRF (ZHEEVR) is called.

Constraints:

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

16: ISUPPZ$(*)$ – INTEGER array *Output*

Note: the dimension of the array ISUPPZ must be at least $\max(1, 2 \times \mathrm{M})$.

On exit: the support of the eigenvectors in Z, i.e., the indices indicating the nonzero elements in Z. The ith eigenvector is nonzero only in elements ISUPPZ$(2 \times i - 1)$ through ISUPPZ$(2 \times i)$. Implemented only for RANGE $=$ 'A' or 'I' and $\mathrm{IU} - \mathrm{IL} = \mathrm{N} - 1$.

17: WORK$(*)$ – ***complex*16*** array *Workspace*

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

On exit: if $\mathrm{INFO} = 0$, $\mathrm{WORK}(1)$ returns the optimal LWORK.

18: LWORK – INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F08FRF (ZHEEVR) is called.

For optimal efficiency, $\mathrm{LWORK} \geq (nb + 1) \times \mathrm{N}$, where nb is the largest optimal block size for F08FSF (ZHETRD) and for F08FUF (ZUNMTR).

If $\mathrm{LWORK} = -1$, a workspace query is assumed; the routine only calculates the optimal sizes of the WORK, RWORK and IWORK arrays, returns these values as the first entries of the WORK, RWORK and IWORK arrays, and no error message related to LWORK, LRWORK or LIWORK is issued.

Constraint: $\mathrm{LWORK} \geq \max(1, 2 \times \mathrm{N})$.

19: RWORK$(*)$ – ***double precision*** array *Workspace*

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

On exit: if $\mathrm{INFO} = 0$, $\mathrm{RWORK}(1)$ returns the optimal (and minimal) LRWORK.

20: LRWORK – INTEGER *Input*

On entry: the dimension of the array RWORK as declared in the (sub)program from which F08FRF (ZHEEVR) is called.

If $\mathrm{LRWORK} = -1$, a workspace query is assumed; the routine only calculates the optimal sizes of the WORK, RWORK and IWORK arrays, returns these values as the first entries of the WORK, RWORK and IWORK arrays, and no error message related to LWORK, LRWORK or LIWORK is issued.

Constraint: $\mathrm{LRWORK} \geq \max(1, 24 \times \mathrm{N})$.

21: IWORK$(*)$ – INTEGER array *Workspace*

On exit: if INFO $= 0$, $\mathrm{IWORK}(1)$ returns the optimal (and minimal) LIWORK.

22: LIWORK – INTEGER *Input*

On entry: the dimension of the array IWORK as declared in the (sub)program from which F08FRF (ZHEEVR) is called.

If $\mathrm{LIWORK} = -1$, a workspace query is assumed; the routine only calculates the optimal sizes of the WORK, RWORK and IWORK arrays, returns these values as the first entries of the WORK,

RWORK and IWORK arrays, and no error message related to LWORK, LRWORK or LIWORK is issued.

Constraint: LIWORK $\geq \max(1, 10 \times \mathrm{N})$.

23: 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 $= -i$, the ith argument had an illegal value.

INFO > 0

F08FRF (ZHEEVR) failed to converge.

7 Accuracy

The computed eigenvalues and eigenvectors are exact for a nearby matrix $(A + E)$, where

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

and ϵ is the ***machine precision***. See Section 4.7 of Anderson *et al.* (1999) for further details.

8 Further Comments

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

The real analogue of this routine is F08FDF (DSYEVR).

9 Example

To find the eigenvalues with indices in the range $[2, 3]$, and the corresponding eigenvectors, of the Hermitian matrix

$$A = \begin{pmatrix} 1 & 2-i & 3-i & 4-i \\ 2+i & 2 & 3-2i & 4-2i \\ 3+i & 3+2i & 3 & 4-3i \\ 4+i & 4+2i & 4+3i & 4 \end{pmatrix}.$$

Information on required and provided workspace is also output

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     F08FRF Example Program Text
*     Mark 21 Release. NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NB, NMAX, MMAX
      PARAMETER        (NB=64,NMAX=10,MMAX=5)
      INTEGER          LDA, LDZ, LIWORK, LRWORK, LWORK
      PARAMETER        (LDA=NMAX,LDZ=NMAX,LIWORK=10*NMAX,LRWORK=24*NMAX,
     +                 LWORK=(NB+1)*NMAX)
      DOUBLE PRECISION ZERO
      PARAMETER        (ZERO=0.0D+0)
```

```
*     .. Local Scalars ..
      DOUBLE PRECISION ABSTOL, VL, VU
      INTEGER          I, IFAIL, IL, INFO, IU, J, LIWOPT, LRWOPT, LWOPT,
     +                 M, N
*     .. Local Arrays ..
      COMPLEX *16      A(LDA,NMAX), WORK(LWORK), Z(LDZ,MMAX)
      DOUBLE PRECISION RWORK(LRWORK), W(NMAX)
      INTEGER          ISUPPZ(2*MMAX), IWORK(LIWORK)
*     .. External Subroutines ..
      EXTERNAL         X04DAF, ZHEEVR
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F08FRF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file and read N and the lower and upper
*     indices of the smallest and largest eigenvalues to be found
      READ (NIN,*)
      READ (NIN,*) N, IL, IU
      IF (N.LE.NMAX .AND. (IU-IL+1).LE.MMAX) THEN
*
*        Read the upper triangular part of the matrix A from data file
*
         READ (NIN,*) ((A(I,J),J=I,N),I=1,N)
*
*        Set the absolute error tolerance for eigenvalues.  With ABSTOL
*        set to zero, the default value is used instead
*
         ABSTOL = ZERO
*
*        Solve the symmetric eigenvalue problem
*
         CALL ZHEEVR('Vectors','I','Upper',N,A,LDA,VL,VU,IL,IU,ABSTOL,M,
     +               W,Z,LDZ,ISUPPZ,WORK,LWORK,RWORK,LRWORK,IWORK,
     +               LIWORK,INFO)
         LWOPT = WORK(1)
         LRWOPT = RWORK(1)
         LIWOPT = IWORK(1)
*
         IF (INFO.EQ.0) THEN
*
*           Print solution
*
            WRITE (NOUT,*) 'Selected eigenvalues'
            WRITE (NOUT,99999) (W(J),J=1,M)
*
            IFAIL = 0
            CALL X04DAF('General',' ',N,M,Z,LDZ,'Selected eigenvectors',
     +                  IFAIL)
         ELSE
            WRITE (NOUT,99998) 'Failure in ZHEEVR. INFO =', INFO
         END IF
*
*        Print workspace information
*
         IF (LWORK.LT.LWOPT) THEN
            WRITE (NOUT,*)
            WRITE (NOUT,99997) 'Optimum complex workspace required = ',
     +        LWOPT, 'Complex workspace provided         = ', LWORK
         END IF
         IF (LRWORK.LT.LRWOPT) THEN
            WRITE (NOUT,*)
            WRITE (NOUT,99997) 'Real workspace required  = ', LRWOPT,
     +        'Real  workspace provided = ', LRWORK
         END IF
         IF (LIWORK.LT.LIWOPT) THEN
            WRITE (NOUT,*)
            WRITE (NOUT,99997) 'Integer workspace required = ', LIWOPT,
     +        'Integer workspace provided = ', LIWORK
         END IF
      ELSE
         WRITE (NOUT,*) 'NMAX and/or MMAX too small'
      END IF
```

```
      STOP
*
99999 FORMAT (3X,(8F8.4))
99998 FORMAT (1X,A,I5)
99997 FORMAT (1X,A,I5,/1X,A,I5)
      END
```

9.2 Program Data

```
F08FRF Example Program Data

   4           2           3                          :Values of N, IL and IU

  (1.0, 0.0)  (2.0,-1.0)  (3.0,-1.0)  (4.0,-1.0)
              (2.0, 0.0)  (3.0,-2.0)  (4.0,-2.0)
                          (3.0, 0.0)  (4.0,-3.0)
                                      (4.0, 0.0)     :End of matrix A
```

9.3 Program Results

```
 F08FRF Example Program Results

 Selected eigenvalues
    -0.6886  1.1412
 Selected eigenvectors
          1       2
 1  -0.3975 -0.3746
     0.5105 -0.2414

 2   0.3953  0.2895
    -0.3238 -0.4917

 3  -0.4309  0.3768
     0.0383  0.3994

 4   0.3648 -0.4175
     0.0000  0.0000
```