F08PXF (CHSEIN/ZHSEIN) – NAG Fortran Library Routine Document

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

F08PXF (CHSEIN/ZHSEIN) computes selected left and/or right eigenvectors of a complex upper Hessenberg matrix corresponding to specified eigenvalues, by inverse iteration.

2 Specification

```
 SUBROUTINE F08PXF(JOB, EIGSRC, INITV, SELECT, N, H, LDH, W, VL,
1                  LDVL, VR, LDVR, MM, M, WORK, RWORK, IFAILL,
2                  IFAILR, INFO)
 ENTRY      chsein(JOB, EIGSRC, INITV, SELECT, N, H, LDH, W, VL,
1                  LDVL, VR, LDVR, MM, M, WORK, RWORK, IFAILL,
2                  IFAILR, INFO)
 INTEGER           N, LDH, LDVL, LDVR, MM, M, IFAILL(*), IFAILR(*),
1                  INFO
 real              RWORK(*)
 complex           H(LDH,*), W(*), VL(LDVL,*), VR(LDVR,*), WORK(*)
 LOGICAL           SELECT(*)
 CHARACTER*1       JOB, EIGSRC, INITV
```

The ENTRY statement enables the routine to be called by its LAPACK name.

3 Description

This routine computes left and/or right eigenvectors of a complex upper Hessenberg matrix H, corresponding to selected eigenvalues.

The right eigenvector x, and the left eigenvector y, corresponding to an eigenvalue λ, are defined by:

$$Hx = \lambda x \text{ and } y^H H = \lambda y^H \text{ (or } H^H y = \bar{\lambda} y\text{)}.$$

The eigenvectors are computed by inverse iteration. They are scaled so that $\max(|\mathrm{Re}(x_i)| + |\mathrm{Im}(x_i)|) = 1$.

If H has been formed by reduction of a complex general matrix A to upper Hessenberg form, then the eigenvectors of H may be transformed to eigenvectors of A by a call to F08NUF (CUNMHR/ZUNMHR).

4 References

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

5 Parameters

1: JOB — CHARACTER*1 *Input*

On entry: indicates whether left and/or right eigenvectors are to be computed as follows:

if JOB = 'R', then only right eigenvectors are computed;
if JOB = 'L', then only left eigenvectors are computed;
if JOB = 'B', then both left and right eigenvectors are computed.

Constraint: JOB = 'R', 'L' or 'B'.

2: EIGSRC — CHARACTER*1 *Input*

On entry: indicates whether the eigenvalues of H (stored in W) were found using F08PSF (CHSEQR/ZHSEQR) as follows:

if EIGSRC = 'Q', then the eigenvalues of H were found using F08PSF (CHSEQR/ZHSEQR); thus if H has any zero sub-diagonal elements (and so is block triangular), then the jth eigenvalue can be assumed to be an eigenvalue of the block containing the jth row/column. This property allows the routine to perform inverse iteration on just one diagonal block;
if EIGSRC = 'N', then no such assumption is made and the routine performs inverse iteration using the whole matrix.

Constraint: EIGSRC = 'Q' or 'N'.

3: INITV — CHARACTER*1 *Input*

On entry: indicates whether the user is supplying initial estimates for the selected eigenvectors as follows:

if INITV = 'N', then no initial estimates are supplied;
if INITV = 'U', then initial estimates are supplied in VL and/or VR.

Constraint: INITV = 'N' or 'U'.

4: SELECT(∗) — LOGICAL array *Input*

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

On entry: SELECT specifies which eigenvectors are to be computed. To select the eigenvector corresponding to the eigenvalue W(j), SELECT(j) must be set .TRUE..

5: N — INTEGER *Input*

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

Constraint: N $\geq$ 0.

6: H(LDH,∗) — ***complex*** array *Input*

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

On entry: the n by n upper Hessenberg matrix H.

7: LDH — INTEGER *Input*

On entry: the first dimension of the array H as declared in the (sub)program from which F08PXF (CHSEIN/ZHSEIN) is called.

Constraint: LDH $\geq$ max(1,N).

8: W(∗) — ***complex*** array *Input/Output*

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

On entry: the eigenvalues of the matrix H. If EIGSRC = 'Q', the array **must** be exactly as returned by F08PSF (CHSEQR/ZHSEQR).

On exit: the real parts of some elements of W may be modified, as close eigenvalues are perturbed slightly in searching for independent eigenvectors.

9: VL(LDVL,∗) — ***complex*** array *Input/Output*

Note: the second dimension of the array VL must be at least max(1,MM) if JOB = 'L' or 'B' and at least 1 if JOB = 'R'.

On entry: if INITV = 'U' and JOB = 'L' or 'B', VL must contain starting vectors for inverse iteration for the left eigenvectors. Each starting vector must be stored in the same column as will be used to store the corresponding eigenvector (see below). If INITV = 'N', VL need not be set.

On exit: if JOB = 'L' or 'B', VL contains the computed left eigenvectors (as specified by SELECT). The eigenvectors are stored consecutively in the columns of the array, in the same order as their eigenvalues.

VL is not referenced if JOB = 'R'.

10: LDVL — INTEGER *Input*

On entry: the first dimension of the array VL as declared in the (sub)program from which F08PXF (CHSEIN/ZHSEIN) is called.

Constraints:

LDVL $\geq$ max(1,N) if JOB = 'L' or 'B',
LDVL $\geq$ 1 if JOB = 'R'.

11: VR(LDVR,$*$) — ***complex*** array *Input/Output*

Note: the second dimension of the array VR must be at least max(1,MM) if JOB = 'R' or 'B' and at least 1 if JOB = 'L'.

On entry: if INITV = 'U' and JOB = 'R' or 'B', VR must contain starting vectors for inverse iteration for the right eigenvectors. Each starting vector must be stored in the same column as will be used to store the corresponding eigenvector (see below). If INITV = 'N', VR need not be set.

On exit: if JOB = 'R' or 'B', VR contains the computed right eigenvectors (as specified by SELECT). The eigenvectors are stored consecutively in the columns of the array, in the same order as their eigenvalues.

VR is not referenced if JOB = 'L'.

12: LDVR — INTEGER *Input*

On entry: the first dimension of the array VR as declared in the (sub)program from which F08PXF (CHSEIN/ZHSEIN) is called.

Constraints:

LDVR $\geq$ max(1,N) if JOB = 'R' or 'B',
LDVR $\geq$ 1 if JOB = 'L'.

13: MM — INTEGER *Input*

On entry: the number of columns in the arrays VL and/or VR. The actual number of columns required, m, is equal to the number of selected eigenvectors (see SELECT); $0 \leq m \leq n$.

Constraint: MM $\geq m$.

14: M — INTEGER *Output*

On exit: m, the number of selected eigenvectors.

15: WORK($*$) — ***complex*** array *Workspace*

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

16: RWORK($*$) — ***real*** array *Workspace*

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

17: IFAILL($*$) — INTEGER array *Output*

Note: the dimension of the array IFAILL must be at least max(1,MM) if JOB = 'L' or 'B' and at least 1 if JOB = 'R'.

On exit: if JOB = 'L' or 'B', then IFAILL(i) = 0 if the selected left eigenvector converged and IFAILL(i) = $j > 0$ if the eigenvector stored in the ith column of VL (corresponding to the jth eigenvalue) failed to converge.

IFAILL is not referenced if JOB = 'R'.

18: IFAILR(∗) — INTEGER array *Output*

Note: the dimension of the array IFAILR must be at least max(1,MM) if JOB = ’R’ or ’B’ and at least 1 if JOB = ’L’.

On exit: if JOB = ’R’ or ’B’, then IFAILR(i) = 0 if the selected right eigenvector converged and IFAILR(i) = $j > 0$ if the eigenvector stored in the ith column of VR (corresponding to the jth eigenvalue) failed to converge.

IFAILR is not referenced if JOB = ’L’.

19: INFO — INTEGER *Output*

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

6 Error Indicators and Warnings

INFO < 0

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

INFO > 0

If INFO = i, then i eigenvectors (as indicated by the parameters IFAILL and/or IFAILR above) failed to converge. The corresponding columns of VL and/or VR contain no useful information.

7 Accuracy

Each computed right eigenvector x_i is the exact eigenvector of a nearby matrix $A + E_i$, such that $\| E_i \| = O(\epsilon) \| A \|$. Hence the residual is small:

$$\| Ax_i - \lambda_i x_i \| = O(\epsilon) \| A \|.$$

However eigenvectors corresponding to close or coincident eigenvalues may not accurately span the relevant subspaces.

Similar remarks apply to computed left eigenvectors.

8 Further Comments

The real analogue of this routine is F08PKF (SHSEIN/DHSEIN).

9 Example

See the example for F08NUF (CUNMHR/ZUNMHR).