NAG Library Routine Document

F08LSF (ZGBBRD)

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

F08LSF (ZGBBRD) reduces a complex m by n band matrix to real upper bidiagonal form.

2 Specification

```
SUBROUTINE F08LSF (VECT, M, N, NCC, KL, KU, AB, LDAB, D, E, Q, LDQ, PT,          &
                   LDPT, C, LDC, WORK, RWORK, INFO)

INTEGER                 M, N, NCC, KL, KU, LDAB, LDQ, LDPT, LDC, INFO
REAL (KIND=nag_wp)      D(min(M,N)), E(min(M,N)-1), RWORK(max(M,N))
COMPLEX (KIND=nag_wp)   AB(LDAB,*), Q(LDQ,*), PT(LDPT,*), C(LDC,*),             &
                        WORK(max(M,N))
CHARACTER(1)            VECT
```

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

3 Description

F08LSF (ZGBBRD) reduces a complex m by n band matrix to real upper bidiagonal form B by a unitary transformation: $A = QBP^{\mathrm{H}}$. The unitary matrices Q and P^{H}, of order m and n respectively, are determined as a product of Givens rotation matrices, and may be formed explicitly by the routine if required. A matrix C may also be updated to give $\tilde{C} = Q^{\mathrm{H}}C$.

The routine uses a vectorizable form of the reduction.

4 References

None.

5 Parameters

1: VECT – CHARACTER(1) *Input*

On entry: indicates whether the matrices Q and/or P^{H} are generated.

VECT = 'N'
: Neither Q nor P^{H} is generated.

VECT = 'Q'
: Q is generated.

VECT = 'P'
: P^{H} is generated.

VECT = 'B'
: Both Q and P^{H} are generated.

Constraint: VECT = 'N', 'Q', 'P' or 'B'.

2: M – INTEGER *Input*

On entry: m, the number of rows of the matrix A.

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

3: N – INTEGER *Input*

On entry: n, the number of columns of the matrix A.

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

4: NCC – INTEGER *Input*

On entry: n_C, the number of columns of the matrix C.

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

5: KL – INTEGER *Input*

On entry: the number of subdiagonals, k_l, within the band of A.

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

6: KU – INTEGER *Input*

On entry: the number of superdiagonals, k_u, within the band of A.

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

7: AB(LDAB,$*$) – COMPLEX (KIND=nag_wp) array *Input/Output*

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

On entry: the original m by n band matrix A.

The matrix is stored in rows 1 to $k_l + k_u + 1$, more precisely, the element A_{ij} must be stored in

$$\mathrm{AB}(k_u + 1 + i - j, j) \qquad \text{for } \max(1, j - k_u) \leq i \leq \min(m, j + k_l).$$

On exit: AB is overwritten by values generated during the reduction.

8: LDAB – INTEGER *Input*

On entry: the first dimension of the array AB as declared in the (sub)program from which F08LSF (ZGBBRD) is called.

Constraint: $\mathrm{LDAB} \geq \mathrm{KL} + \mathrm{KU} + 1$.

9: D$(\min(\mathrm{M},\mathrm{N}))$ – REAL (KIND=nag_wp) array *Output*

On exit: the diagonal elements of the bidiagonal matrix B.

10: E$(\min(\mathrm{M},\mathrm{N}) - 1)$ – REAL (KIND=nag_wp) array *Output*

On exit: the superdiagonal elements of the bidiagonal matrix B.

11: Q(LDQ,$*$) – COMPLEX (KIND=nag_wp) array *Output*

Note: the second dimension of the array Q must be at least $\max(1,\mathrm{M})$ if VECT = 'Q' or 'B', and at least 1 otherwise.

On exit: if VECT = 'Q' or 'B', contains the m by m unitary matrix Q.

If VECT = 'N' or 'P', Q is not referenced.

12: LDQ – INTEGER *Input*

On entry: the first dimension of the array Q as declared in the (sub)program from which F08LSF (ZGBBRD) is called.

Constraints:

if VECT = 'Q' or 'B', LDQ $\geq \max(1, \mathrm{M})$;
otherwise LDQ ≥ 1.

13: PT(LDPT,$*$) – COMPLEX (KIND=nag_wp) array *Output*

Note: the second dimension of the array PT must be at least $\max(1, \mathrm{N})$ if VECT = 'P' or 'B', and at least 1 otherwise.

On exit: the n by n unitary matrix P^{H}, if VECT = 'P' or 'B'. If VECT = 'N' or 'Q', PT is not referenced.

14: LDPT – INTEGER *Input*

On entry: the first dimension of the array PT as declared in the (sub)program from which F08LSF (ZGBBRD) is called.

Constraints:

if VECT = 'P' or 'B', LDPT $\geq \max(1, \mathrm{N})$;
otherwise LDPT ≥ 1.

15: C(LDC,$*$) – COMPLEX (KIND=nag_wp) array *Input/Output*

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

On entry: an m by n_C matrix C.

On exit: C is overwritten by $Q^{\mathrm{H}}C$. If NCC $= 0$, C is not referenced.

16: LDC – INTEGER *Input*

On entry: the first dimension of the array C as declared in the (sub)program from which F08LSF (ZGBBRD) is called.

Constraints:

if NCC > 0, LDC $\geq \max(1, \mathrm{M})$;
if NCC $= 0$, LDC ≥ 1.

17: WORK($\max(\mathrm{M}, \mathrm{N})$) – COMPLEX (KIND=nag_wp) array *Workspace*

18: RWORK($\max(\mathrm{M}, \mathrm{N})$) – REAL (KIND=nag_wp) array *Workspace*

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

7 Accuracy

The computed bidiagonal form B satisfies $QBP^{\mathrm{H}} = A + E$, where

$$\|E\|_2 \leq c(n)\epsilon\|A\|_2,$$

$c(n)$ is a modestly increasing function of n, and ϵ is the ***machine precision***.

The elements of B themselves may be sensitive to small perturbations in A or to rounding errors in the computation, but this does not affect the stability of the singular values and vectors.

The computed matrix Q differs from an exactly unitary matrix by a matrix F such that

$$\|F\|_2 = O(\epsilon).$$

A similar statement holds for the computed matrix P^{H}.

8 Further Comments

The total number of real floating point operations is approximately the sum of:

$20n^2k$, if VECT = 'N' and NCC $= 0$, and

$10n^2n_C(k-1)/k$, if C is updated, and

$10n^3(k-1)/k$, if either Q or P^{H} is generated (double this if both),

where $k = k_l + k_u$, assuming $n \gg k$. For this section we assume that $m = n$.

The real analogue of this routine is F08LEF (DGBBRD).

9 Example

This example reduces the matrix A to upper bidiagonal form, where

$$A = \begin{pmatrix} 0.96-0.81i & -0.03+0.96i & 0.00+0.00i & 0.00+0.00i \\ -0.98+1.98i & -1.20+0.19i & -0.66+0.42i & 0.00+0.00i \\ 0.62-0.46i & 1.01+0.02i & 0.63-0.17i & -1.11+0.60i \\ 0.00+0.00i & 0.19-0.54i & -0.98-0.36i & 0.22-0.20i \\ 0.00+0.00i & 0.00+0.00i & -0.17-0.46i & 1.47+1.59i \\ 0.00+0.00i & 0.00+0.00i & 0.00+0.00i & 0.26+0.26i \end{pmatrix}.$$

9.1 Program Text

```
    Program f08lsfe

!     F08LSF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, zgbbrd
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
      Character (1), Parameter         :: vect = 'N'
!     .. Local Scalars ..
      Integer                          :: i, info, j, kl, ku, ldab, ldc, ldpt, &
                                          ldq, m, n, ncc
!     .. Local Arrays ..
      Complex (Kind=nag_wp), Allocatable :: ab(:,:), c(:,:), pt(:,:), q(:,:),  &
                                            work(:)
      Real (Kind=nag_wp), Allocatable  :: d(:), e(:), rwork(:)
!     .. Intrinsic Procedures ..
      Intrinsic                        :: max, min
!     .. Executable Statements ..
      Write (nout,*) 'F08LSF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) m, n, kl, ku, ncc
      ldab = kl + ku + 1
      ldc = m
      ldpt = n
      ldq = m
      Allocate (ab(ldab,n),c(m,ncc),pt(ldpt,n),q(ldq,m),work(m+n),d(n),e(n-1), &
        rwork(m+n))
```

```
!     Read A from data file

      Read (nin,*)((ab(ku+1+i-j,j),j=max(i-kl,1),min(i+ku,n)),i=1,m)

!     Reduce A to upper bidiagonal form

!     The NAG name equivalent of zgbbrd is f08lsf
      Call zgbbrd(vect,m,n,ncc,kl,ku,ab,ldab,d,e,q,ldq,pt,ldpt,c,ldc,work, &
        rwork,info)

!     Print bidiagonal form

      Write (nout,*)
      Write (nout,*) 'Diagonal'
      Write (nout,99999) d(1:min(m,n))
      Write (nout,*) 'Super-diagonal'
      Write (nout,99999) e(1:min(m,n)-1)

99999 Format (1X,8F9.4)
    End Program f08lsfe
```

9.2 Program Data

```
F08LSF Example Program Data
  6  4  2  1  0                                :Values of M, N, KL, KU and NCC
 ( 0.96,-0.81) (-0.03, 0.96)
 (-0.98, 1.98) (-1.20, 0.19) (-0.66, 0.42)
 ( 0.62,-0.46) ( 1.01, 0.02) ( 0.63,-0.17) (-1.11, 0.60)
               ( 0.19,-0.54) (-0.98,-0.36) ( 0.22,-0.20)
                             (-0.17,-0.46) ( 1.47, 1.59)
                                           ( 0.26, 0.26)   :End of matrix A
```

9.3 Program Results

```
 F08LSF Example Program Results

 Diagonal
    2.6560   1.7501   2.0607   0.8658
 Super-diagonal
    1.7033   1.2800   0.1467
```