NAG Library Routine Document

F08MBF (DBDSVDX)

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

F08MBF (DBDSVDX) computes all or selected singular values and, optionally, the corresponding left and right singular vectors of a real n by n (upper or lower) bidiagonal matrix B.

2 Specification

```
SUBROUTINE F08MBF (UPLO, JOBZ, RANGE, N, D, E, VL, VU, IL, IU, NS, S, Z,    &
                   LDZ, WORK, IWORK, INFO)

INTEGER             N, IL, IU, NS, LDZ, IWORK(12*N), INFO
REAL (KIND=nag_wp) D(N), E(N-1), VL, VU, S(N), Z(LDZ,*), WORK(14*N)
CHARACTER(1)        UPLO, JOBZ, RANGE
```

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

3 Description

F08MBF (DBDSVDX) computes the singular value decomposition (SVD) of a real n by n (upper or lower) bidiagonal matrix B as

$$B = USV^{\mathrm{T}},$$

where S is a diagonal matrix with non-negative diagonal elements (the singular values of B), and U and V^{T} are orthogonal matrices of left and right singular vectors, respectively.

Given an upper bidiagonal B with diagonal $\boldsymbol{d} = \begin{pmatrix} d_1 & d_2 & \ldots & d_n \end{pmatrix}$ and superdiagonal $\boldsymbol{e} = \begin{pmatrix} e_1 & e_2 & \ldots & e_{N-1} \end{pmatrix}$, F08MBF (DBDSVDX) computes the singular value decompositon of B through the eigenvalues and eigenvectors of the $n \times 2$ by $n \times 2$ tridiagonal matrix

$$TGK = \begin{pmatrix} 0 & d_1 & & & \\ d_1 & 0 & e_1 & & \\ & e_1 & 0 & d_2 & \\ & & d_2 & . & . \\ & & & . & . \end{pmatrix}.$$

If (s, u, v) is a singular triplet of B with $\|u\| = \|v\| = 1$, then (s, q) and $(-s, q)$, $\|q\| = 1$, are eigenpairs of TGK, with $q = (v_1, u_1, v_2, u_2, \ldots, v_n, u_n)/\sqrt{2}$ for s, and $q = (-v_1, u_1, -v_2, u_2, \ldots, -v_n, u_n)/\sqrt{2}$ for $-s$.

Given a TGK matrix, one can either

(i) compute $-s, -v$ and change signs so that the singular values (and corresponding vectors) are already in descending order (as in F08KBF (DGESVD)) or

(ii) compute s, v and reorder the values (and corresponding vectors).

F08MBF (DBDSVDX) implements (i) by calling F08JBF (DSTEVX) (bisection plus inverse iteration, to be replaced with a version of the Multiple Relative Robust Representation algorithm. (See Williams and Lang (2013).)

Alternative to computing all singular values of A, a selected set can be computed. The set is either those singular values lying in a given interval, $\sigma \in (v_l, v_u]$, or those whose index (counting from largest to smallest in magnitude) lies in a given range $1 \leq i_l, \ldots, i_u \leq n$. In these cases, the corresponding left and right singular vectors can optionally be computed.

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

Williams P and Lang B (2013) A framework for the MR^3 algorithm: theory and implementation *SIAM J. Sci. Comput.* **35** 740–766

5 Arguments

1: UPLO – CHARACTER(1) *Input*

On entry: indicates whether B is upper or lower bidiagonal.

UPLO = 'U'
B is upper bidiagonal.

UPLO = 'L'
B is lower bidiagonal.

Constraint: UPLO = 'U' or 'L'.

2: JOBZ – CHARACTER(1) *Input*

On entry: indicates whether singular vectors are computed.

JOBZ = 'N'
Only singular values are computed.

JOBZ = 'V'
Singular values and singular vectors are computed.

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

3: RANGE – CHARACTER(1) *Input*

On entry: indicates which singular values should be returned.

RANGE = 'A'
All singular values will be found.

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

RANGE = 'I'
The ILth through IUth singular values will be found.

Constraint: RANGE = 'A', 'V' or 'I'.

4: N – INTEGER *Input*

On entry: n, the order of the bidiagonal matrix B.

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

5: D(N) – REAL (KIND=nag_wp) array *Input*

On entry: the diagonal elements $\boldsymbol{d}$ of the bidiagonal matrix B.

6: E(N − 1) – REAL (KIND=nag_wp) array *Input*

On entry: the $(n-1)$ superdiagonal elements $\boldsymbol{e}$ of the bidiagonal matrix B.

7: VL – REAL (KIND=nag_wp) *Input*

On entry: if RANGE = 'V', the lower bounds of the interval to be searched for singular values.

If RANGE = 'A' or 'I', VL is not referenced.

Constraint: if RANGE = 'V', $0.0 \leq$ VL.

8: VU – REAL (KIND=nag_wp) *Input*

On entry: if RANGE = 'V', the upper bounds of the interval to be searched for singular values.

If RANGE = 'A' or 'I', VU is not referenced.

Constraint: if RANGE = 'V', VL $<$ VU.

9: IL – INTEGER *Input*

10: IU – INTEGER *Input*

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

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

Constraints:

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

11: NS – INTEGER *Output*

On exit: the total number of singular values found. $0 \leq$ NS $\leq$ N.

If RANGE = 'A', NS = N.

If RANGE = 'I', NS = IU − IL + 1.

12: S(N) – REAL (KIND=nag_wp) array *Output*

On exit: the first NS elements contain the selected singular values in ascending order.

13: Z(LDZ, $*$) – REAL (KIND=nag_wp) array *Output*

Note: the second dimension of the array Z must be at least $\min(\text{N}, \text{NS} + 1)$ if JOBZ = 'V', and at least 1 otherwise.

On exit: If JOBZ = 'V', then if INFO = 0 the first NS columns of Z contain the singular vectors of the matrix B corresponding to the selected singular values, with U in rows 1 to N and V in rows $N + 1$ to $N \times 2$, i.e.,

$$Z = \begin{pmatrix} U \\ V \end{pmatrix}.$$

If JOBZ = 'V', then Z is not referenced.

Note: the user must ensure that at least $K = \text{NS} + 1$ columns are supplied in the array Z; if RANGE = 'V', the exact value of NS is not known in advance and an upper bound must be used.

14: LDZ – INTEGER *Input*

On entry: the first dimension of the array Z as declared in the (sub)program from which F08MBF (DBDSVDX) is called.

Constraints:

if JOBZ = 'V', LDZ $\geq \max(2, \text{N} \times 2)$;
otherwise LDZ ≥ 1.

15: WORK($14 \times$ N) – REAL (KIND=nag_wp) array *Workspace*

16: IWORK($12 \times$ N) – INTEGER array *Workspace*

On exit: if JOBZ = 'V', then

if INFO $= 0$, the first NS elements of IWORK are zero;

if INFO > 0, IWORK contains the indices of the eigenvectors that failed to converge in F08JBF (DSTEVX).

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

17: 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.

INFO > 0

The algorithm failed to converge; $\langle$*value*$\rangle$ eigenvectors of the associated eigenproblem did not converge. Their indices are stored in array IWORK.

7 Accuracy

Each computed singular value of B is accurate to nearly full relative precision, no matter how tiny the singular value. The ith computed singular value, $\hat{s}_i$, satisfies the bound

$$|\hat{s}_i - s_i| \leq p(n)\epsilon s_i$$

where ϵ is the ***machine precision*** and $p(n)$ is a modest function of n.

For bounds on the computed singular values, see Section 4.9.1 of Anderson *et al.* (1999). See also F08FLF (DDISNA).

8 Parallelism and Performance

F08MBF (DBDSVDX) is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

F08MBF (DBDSVDX) makes calls to BLAS and/or LAPACK routines, which may be threaded within the vendor library used by this implementation. Consult the documentation for the vendor library for further information.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9 Further Comments

If only singular values are required, the total number of floating-point operations is approximately proportional to n^2. When singular vectors are required the number of operations is bounded above by approximately the same number of operations as F08MEF (DBDSQR), but for large matrices F08MBF (DBDSVDX) is usually much faster.

There is no complex analogue of F08MBF (DBDSVDX).

10 Example

This example computes the singular value decomposition of the upper bidiagonal matrix

$$B = \begin{pmatrix} 3.62 & 1.26 & 0 & 0 \\ 0 & -2.41 & -1.53 & 0 \\ 0 & 0 & 1.92 & 1.19 \\ 0 & 0 & 0 & -1.43 \end{pmatrix}.$$

10.1 Program Text

```
    Program f08mbfe

!     F08MBF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: dbdsvdx, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: vl, vu
      Integer                          :: il, info, iu, ldz, n, ns
      Character (1)                    :: range
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: d(:), e(:), s(:), work(:), z(:,:)
      Integer, Allocatable             :: iwork(:)
!     .. Executable Statements ..
      Write (nout,*) 'F08MBF Example Program Results'
      Write (nout,*)
      Flush (nout)
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n
      ldz = 2*n
      Allocate (d(n),e(n-1),s(n),z(ldz,n+1),work(14*n),iwork(12*n))

!     Read the bidiagonal matrix B from data file, first
!     the diagonal elements, and then the off diagonal elements

      Read (nin,*) d(1:n)
      Read (nin,*) e(1:n-1)

!     Read range for selected singular values
      Read (nin,*) range

      If (range=='I' .Or. range=='i') Then
        Read (nin,*) il, iu
      Else If (range=='V' .Or. range=='v') Then
        Read (nin,*) vl, vu
      End If

!     Calculate the singular values and singular vectors of B.

!     The NAG name equivalent of dbdsvdx is f08mbf
      Call dbdsvdx('Upper','V',range,n,d,e,vl,vu,il,iu,ns,s,z,ldz,work,iwork,  &
        info)

      If (info==0) Then
!       Print the singular values of B.

        If (range=='I' .Or. range=='i') Then
          Write (nout,99999) ns, il, iu
        Else If (range=='V' .Or. range=='v') Then
          Write (nout,99998) ns, vl, vu
        End If
```

```
        Write (nout,99997) s(1:ns)
      Else
        Write (nout,99996) '** F08MBF/DBDSVDX failed with INFO = ', info
      End If

99999 Format (1X,I2,1X,'singular values of B in the index range [',I2,',',I2,  &
        ']:')
99998 Format (1X,I2,1X,'singular values of B in the range [',F7.3,',',F7.3,    &
        ']:')
99997 Format (1X,4(3X,F11.4))
99996 Format (1X,A,I10)
    End Program f08mbfe
```

10.2 Program Data

```
F08MBF Example Program Data

  4                           : n

  3.62  -2.41   1.92  -1.43  : diagonal elements,  d
  1.26  -1.53   1.19         : off-diagonal elements, e

  'V'
  0.00   4.00
```

10.3 Program Results

```
 F08MBF Example Program Results

  3 singular values of B in the range [  0.000,  4.000]:
        3.0006        1.9960        0.9998
```