NAG Library Routine Document

F08MEF (DBDSQR)

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.

Warning. The specification of the argument WORK changed at Mark 20: the length of WORK needs to be increased.

1 Purpose

F08MEF (DBDSQR) computes the singular value decomposition of a real upper or lower bidiagonal matrix, or of a real general matrix which has been reduced to bidiagonal form.

2 Specification

```
SUBROUTINE F08MEF (UPLO, N, NCVT, NRU, NCC, D, E, VT, LDVT, U, LDU, C,     &
                   LDC, WORK, INFO)
INTEGER            N, NCVT, NRU, NCC, LDVT, LDU, LDC, INFO
REAL (KIND=nag_wp) D(*), E(*), VT(LDVT,*), U(LDU,*), C(LDC,*), WORK(*)
CHARACTER(1)       UPLO
```

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

3 Description

F08MEF (DBDSQR) computes the singular values and, optionally, the left or right singular vectors of a real upper or lower bidiagonal matrix B. In other words, it can compute the singular value decomposition (SVD) of B as

$$B = U\Sigma V^{\mathrm{T}}.$$

Here Σ is a diagonal matrix with real diagonal elements σ_i (the singular values of B), such that

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0;$$

U is an orthogonal matrix whose columns are the left singular vectors u_i; V is an orthogonal matrix whose rows are the right singular vectors v_i. Thus

$$Bu_i = \sigma_i v_i \quad \text{and} \quad B^{\mathrm{T}} v_i = \sigma_i u_i, \quad i = 1, 2, \ldots, n.$$

To compute U and/or V^{T}, the arrays U and/or VT must be initialized to the unit matrix before F08MEF (DBDSQR) is called.

The routine may also be used to compute the SVD of a real general matrix A which has been reduced to bidiagonal form by an orthogonal transformation: $A = QBP^{\mathrm{T}}$. If A is m by n with $m \geq n$, then Q is m by n and P^{T} is n by n; if A is n by p with $n < p$, then Q is n by n and P^{T} is n by p. In this case, the matrices Q and/or P^{T} must be formed explicitly by F08KFF (DORGBR) and passed to F08MEF (DBDSQR) in the arrays U and/or VT respectively.

F08MEF (DBDSQR) also has the capability of forming $U^{\mathrm{T}}C$, where C is an arbitrary real matrix; this is needed when using the SVD to solve linear least squares problems.

F08MEF (DBDSQR) uses two different algorithms. If any singular vectors are required (i.e., if NCVT > 0 or NRU > 0 or NCC > 0), the bidiagonal QR algorithm is used, switching between zero-shift and implicitly shifted forms to preserve the accuracy of small singular values, and switching between QR and QL variants in order to handle graded matrices effectively (see Demmel and Kahan (1990)). If only singular values are required (i.e., if NCVT $=$ NRU $=$ NCC $= 0$), they are computed by the differential qd algorithm (see Fernando and Parlett (1994)), which is faster and can achieve even greater accuracy.

The singular vectors are normalized so that $\|u_i\| = \|v_i\| = 1$, but are determined only to within a factor ± 1.

4 References

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

Fernando K V and Parlett B N (1994) Accurate singular values and differential qd algorithms *Numer. Math.* **67** 191–229

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

5 Arguments

1: UPLO – CHARACTER(1) *Input*

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

UPLO = 'U'
B is an upper bidiagonal matrix.

UPLO = 'L'
B is a lower bidiagonal matrix.

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

2: N – INTEGER *Input*

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

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

3: NCVT – INTEGER *Input*

On entry: $ncvt$, the number of columns of the matrix V^{T} of right singular vectors. Set $\mathrm{NCVT} = 0$ if no right singular vectors are required.

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

4: NRU – INTEGER *Input*

On entry: nru, the number of rows of the matrix U of left singular vectors. Set $\mathrm{NRU} = 0$ if no left singular vectors are required.

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

5: NCC – INTEGER *Input*

On entry: ncc, the number of columns of the matrix C. Set $\mathrm{NCC} = 0$ if no matrix C is supplied.

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

6: D($*$) – REAL (KIND=nag_wp) array *Input/Output*

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

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

On exit: the singular values in decreasing order of magnitude, unless $\mathrm{INFO} > 0$ (in which case see Section 6).

7: E($*$) – REAL (KIND=nag_wp) array *Input/Output*

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

On entry: the off-diagonal elements of the bidiagonal matrix B.

On exit: E is overwritten, but if INFO > 0 see Section 6.

8: VT(LDVT, $*$) – REAL (KIND=nag_wp) array *Input/Output*

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

On entry: if NCVT > 0, VT must contain an n by $ncvt$ matrix. If the right singular vectors of B are required, $ncvt = n$ and VT must contain the unit matrix; if the right singular vectors of A are required, VT must contain the orthogonal matrix P^{T} returned by F08KFF (DORGBR) with VECT = 'P'.

On exit: the n by $ncvt$ matrix V^{T} or $V^{\mathrm{T}}P^{\mathrm{T}}$ of right singular vectors, stored by rows.

If NCVT $= 0$, VT is not referenced.

9: LDVT – INTEGER *Input*

On entry: the first dimension of the array VT as declared in the (sub)program from which F08MEF (DBDSQR) is called.

Constraints:

if NCVT > 0, LDVT $\geq \max(1, \mathrm{N})$;
otherwise LDVT ≥ 1.

10: U(LDU, $*$) – REAL (KIND=nag_wp) array *Input/Output*

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

On entry: if NRU > 0, U must contain an nru by n matrix. If the left singular vectors of B are required, $nru = n$ and U must contain the unit matrix; if the left singular vectors of A are required, U must contain the orthogonal matrix Q returned by F08KFF (DORGBR) with VECT = 'Q'.

On exit: the nru by n matrix U or QU of left singular vectors, stored as columns of the matrix.

If NRU $= 0$, U is not referenced.

11: LDU – INTEGER *Input*

On entry: the first dimension of the array U as declared in the (sub)program from which F08MEF (DBDSQR) is called.

Constraint: LDU $\geq \max(1, \mathrm{NRU})$.

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

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

On entry: the n by ncc matrix C if NCC > 0.

On exit: C is overwritten by the matrix $U^{\mathrm{T}}C$. If NCC $= 0$, C is not referenced.

13: LDC – INTEGER *Input*

On entry: the first dimension of the array C as declared in the (sub)program from which F08MEF (DBDSQR) is called.

Constraints:

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

14: WORK($*$) – REAL (KIND=nag_wp) array *Workspace*

Note: the dimension of the array WORK must be at least $\max(1, 2 \times \mathrm{N})$ if $\mathrm{NCVT} = 0$ and $\mathrm{NRU} = 0$ and $\mathrm{NCC} = 0$, and at least $\max(1, 4 \times \mathrm{N})$ otherwise.

15: INFO – INTEGER *Output*

On exit: $\mathrm{INFO} = 0$ unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

$\mathrm{INFO} < 0$

If $\mathrm{INFO} = -i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

$\mathrm{INFO} > 0$

The algorithm failed to converge and INFO specifies how many off-diagonals did not converge. In this case, D and E contain on exit the diagonal and off-diagonal elements, respectively, of a bidiagonal matrix orthogonally equivalent to B.

7 Accuracy

Each singular value and singular vector is computed to high relative accuracy. However, the reduction to bidiagonal form (prior to calling the routine) may exclude the possibility of obtaining high relative accuracy in the small singular values of the original matrix if its singular values vary widely in magnitude.

If σ_i is an exact singular value of B and $\tilde{\sigma}_i$ is the corresponding computed value, then

$$|\tilde{\sigma}_i - \sigma_i| \leq p(m,n)\epsilon\sigma_i$$

where $p(m,n)$ is a modestly increasing function of m and n, and ϵ is the ***machine precision***. If only singular values are computed, they are computed more accurately (i.e., the function $p(m,n)$ is smaller), than when some singular vectors are also computed.

If u_i is the corresponding exact left singular vector of B, and $\tilde{u}_i$ is the corresponding computed left singular vector, then the angle $\theta(\tilde{u}_i, u_i)$ between them is bounded as follows:

$$\theta(\tilde{u}_i, u_i) \leq \frac{p(m,n)\epsilon}{relgap_i}$$

where $relgap_i$ is the relative gap between σ_i and the other singular values, defined by

$$relgap_i = \min_{i \neq j} \frac{|\sigma_i - \sigma_j|}{(\sigma_i + \sigma_j)}.$$

A similar error bound holds for the right singular vectors.

8 Parallelism and Performance

F08MEF (DBDSQR) is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

F08MEF (DBDSQR) 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

The total number of floating-point operations is roughly proportional to n^2 if only the singular values are computed. About $6n^2 \times nru$ additional operations are required to compute the left singular vectors and about $6n^2 \times ncvt$ to compute the right singular vectors. The operations to compute the singular values must all be performed in scalar mode; the additional operations to compute the singular vectors can be vectorized and on some machines may be performed much faster.

The complex analogue of this routine is F08MSF (ZBDSQR).

10 Example

This example computes the singular value decomposition of the upper bidiagonal matrix B, where

$$B = \begin{pmatrix} 3.62 & 1.26 & 0.00 & 0.00 \\ 0.00 & -2.41 & -1.53 & 0.00 \\ 0.00 & 0.00 & 1.92 & 1.19 \\ 0.00 & 0.00 & 0.00 & -1.43 \end{pmatrix}.$$

See also the example for F08KFF (DORGBR), which illustrates the use of the routine to compute the singular value decomposition of a general matrix.

10.1 Program Text

```
    Program f08mefe

!     F08MEF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: dbdsqr, f06qhf, nag_wp, x04caf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Real (Kind=nag_wp), Parameter    :: one = 1.0_nag_wp
      Real (Kind=nag_wp), Parameter    :: zero = 0.0_nag_wp
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Integer                          :: ifail, info, ldc, ldu, ldvt, n
      Character (1)                    :: uplo
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: c(:,:), d(:), e(:), u(:,:), vt(:,:), &
                                          work(:)
!     .. Executable Statements ..
      Write (nout,*) 'F08MEF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n
      ldc = 1
      ldu = n
      ldvt = n
      Allocate (c(ldc,1),d(n),e(n-1),u(ldu,n),vt(ldvt,n),work(4*n))

!     Read B from data file

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

      Read (nin,*) uplo

!     Initialize U and VT to be the unit matrix
      Call f06qhf('General',n,n,zero,one,u,ldu)
      Call f06qhf('General',n,n,zero,one,vt,ldvt)

!     Calculate the SVD of B
!     The NAG name equivalent of dbdsqr is f08mef
```

```
      Call dbdsqr(uplo,n,n,n,0,d,e,vt,ldvt,u,ldu,c,ldc,work,info)

      Write (nout,*)
      If (info>0) Then
        Write (nout,*) 'Failure to converge.'
      Else

!       Print singular values, left & right singular vectors

        Write (nout,*) 'Singular values'
        Write (nout,99999) d(1:n)
        Write (nout,*)
        Flush (nout)

!       ifail: behaviour on error exit
!              =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
        ifail = 0
        Call x04caf('General',' ',n,n,vt,ldvt,'Right singular vectors, by row' &
          ,ifail)

        Write (nout,*)
        Flush (nout)

        ifail = 0
        Call x04caf('General',' ',n,n,u,ldu,'Left singular vectors, by column' &
          ,ifail)

      End If

99999 Format (3X,(8F8.4))
    End Program f08mefe
```

10.2 Program Data

```
F08MEF Example Program Data
  4                           :Value of N
  3.62  -2.41   1.92  -1.43
  1.26  -1.53   1.19          :End of matrix B
  'U'                         :Value of UPLO
```

10.3 Program Results

```
 F08MEF Example Program Results

 Singular values
     4.0001  3.0006  1.9960  0.9998

 Right singular vectors, by row
          1       2       3       4
 1   0.8261  0.5246  0.2024  0.0369
 2   0.4512 -0.4056 -0.7350 -0.3030
 3   0.2823 -0.5644  0.1731  0.7561
 4   0.1852 -0.4916  0.6236 -0.5789

 Left singular vectors, by column
          1       2       3       4
 1   0.9129  0.3740  0.1556  0.0512
 2  -0.3935  0.7005  0.5489  0.2307
 3   0.1081 -0.5904  0.6173  0.5086
 4  -0.0132  0.1444 -0.5417  0.8280
```