NAG Fortran Library Routine Document

F07HFF (DPBEQU)

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

F07HFF (DPBEQU) computes a diagonal scaling matrix S intended to equilibrate a real n by n symmetric positive-definite band matrix A, with bandwidth $(2k_d + 1)$, and reduce its condition number.

2 Specification

```
SUBROUTINE F07HFF (UPLO, N, KD, AB, LDAB, S, SCOND, AMAX, INFO)
INTEGER              N, KD, LDAB, INFO
double precision     AB(LDAB,*), S(*), SCOND, AMAX
CHARACTER*1          UPLO
```

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

3 Description

F07HFF (DPBEQU) computes a diagonal scaling matrix S chosen so that

$$s_j = 1/\sqrt{a_{jj}}.$$

This means that the matrix B given by

$$B = SAS,$$

has diagonal elements equal to unity. This in turn means that the condition number of B, $\kappa_2(B)$, is within a factor n of the matrix of smallest possible condition number over all possible choices of diagonal scalings (see Corollary 7.6 of Higham (2002)).

4 References

Higham N J (2002) *Accuracy and Stability of Numerical Algorithms* (2nd Edition) SIAM, Philadelphia

5 Parameters

1: UPLO – CHARACTER*1 *Input*

On entry: indicates whether the upper or lower triangular part of A is stored in the array AB, as follows:

UPLO = 'U'

The upper triangle of A is stored.

UPLO = 'L'

The lower triangle of A is stored.

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

2: N – INTEGER *Input*

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

Constraint: N $\geq$ 0.

3: KD – INTEGER *Input*

On entry: k_d, the number of superdiagonals of the matrix A if UPLO = 'U', or the number of subdiagonals if UPLO = 'L'.

Constraint: KD ≥ 0.

4: AB(LDAB,$*$) – ***double precision*** array *Input*

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

On entry: the upper or lower triangle of the symmetric positive-definite band matrix A whose scaling factors are to be computed, stored in the first $(k_d + 1)$ rows of the array AB. The jth column of A is stored in the jth column of the array AB as follows:

if UPLO = 'U', $\mathrm{AB}(k_d + 1 + i - j, j) = a_{ij}$ for $\max(1, j - k_d) \leq i \leq j$;
if UPLO = 'L', $\mathrm{AB}(1 + i - j, j) = a_{ij}$ for $j \leq i \leq \min(n, j + k_d)$.

Only the elements of the array AB corresponding to the diagonal elements of A are referenced. (Row $(k_d + 1)$ of AB when UPLO = 'U', row 1 of AB when UPLO = 'L'.)

5: LDAB – INTEGER *Input*

On entry: the first dimension of the array AB as declared in the (sub)program from which F07HFF (DPBEQU) is called.

Constraint: LDAB $\geq$ KD + 1.

6: S($*$) – ***double precision*** array *Output*

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

On exit: if INFO = 0 on exit, S contains the diagonal elements of the scaling matrix S.

7: SCOND – ***double precision*** *Output*

On exit: if INFO = 0 on exit, SCOND contains the ratio of the smallest value of S(i) to the largest value of S(i). If SCOND ≥ 0.1 and AMAX is neither too large nor too small, it is not worth scaling by S.

8: AMAX – ***double precision*** *Output*

On exit: $\max |a_{ij}|$. If AMAX is very close to overflow or underflow, the matrix A should be scaled.

9: 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. An explanatory message is output, and execution of the program is terminated.

INFO > 0

If INFO $= i$, the ith diagonal element of A is not positive (and hence A cannot be positive-definite).

7 Accuracy

The computed scale factors will be close to the exact scale factors.

8 Further Comments

The complex analogue of this routine is F07HTF (ZPBEQU).

9 Example

To equilibrate the symmetric positive-definite matrix A given by

$$A = \begin{pmatrix} 5.49 & 2.68 \times 10^{10} & 0 & 0 \\ 2.68 \times 10^{10} & 5.63 \times 10^{20} & -2.39 \times 10^{10} & 0 \\ 0 & -2.39 \times 10^{10} & 2.60 & -2.22 \\ 0 & 0 & -2.22 & 5.17 \end{pmatrix}.$$

Details of the scaling factors and the scaled matrix are output.

9.1 Program Text

```
*     F07HFF Example Program Text
*     Mark 21 Release. NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          KDMAX, NMAX
      PARAMETER        (KDMAX=4,NMAX=8)
      INTEGER          LDAB
      PARAMETER        (LDAB=KDMAX+1)
      CHARACTER        UPLO
      PARAMETER        (UPLO='U')
*     .. Local Scalars ..
      DOUBLE PRECISION AMAX, BIG, SCOND, SJ, SMALL
      INTEGER          I, IFAIL, INFO, J, JJ, KD, N
*     .. Local Arrays ..
      DOUBLE PRECISION AB(LDAB,NMAX), S(NMAX)
*     .. External Functions ..
      DOUBLE PRECISION X02AJF, X02AMF
      INTEGER          X02BHF
      EXTERNAL         X02AJF, X02AMF, X02BHF
*     .. External Subroutines ..
      EXTERNAL         DPBEQU, X04CEF
*     .. Intrinsic Functions ..
      INTRINSIC        MAX, MIN
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F07HFF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N, KD
      IF (N.LE.NMAX .AND. KD.LE.KDMAX) THEN
*
*        Read the upper or lower triangular part of the band matrix A
*        from data file
*
         IF (UPLO.EQ.'U') THEN
            DO 20 I = 1, N
               READ (NIN,*) (AB(KD+1+I-J,J),J=I,MIN(N,I+KD))
   20       CONTINUE
         ELSE IF (UPLO.EQ.'L') THEN
            DO 40 I = 1, N
               READ (NIN,*) (AB(1+I-J,J),J=MAX(1,I-KD),I)
   40       CONTINUE
         END IF
*
*        Print the matrix A
*
         IFAIL = 0
         IF (UPLO.EQ.'U') THEN
            CALL X04CEF(N,N,0,KD,AB,LDAB,'Matrix A',IFAIL)
         ELSE IF (UPLO.EQ.'L') THEN
```

```
            CALL X04CEF(N,N,KD,0,AB,LDAB,'Matrix A',IFAIL)
         END IF
         WRITE (NOUT,*)
*
*        Compute diagonal scaling factors
*
         CALL DPBEQU(UPLO,N,KD,AB,LDAB,S,SCOND,AMAX,INFO)
*
         IF (INFO.GT.0) THEN
            WRITE (NOUT,99999) 'Diagonal element', INFO,
     +        ' of A is non positive'
         ELSE
*
*           Print SCOND, AMAX and the scale factors
*
            WRITE (NOUT,99998) 'SCOND = ', SCOND, ', AMAX = ', AMAX
            WRITE (NOUT,*)
            WRITE (NOUT,*) 'Diagonal scaling factors'
            WRITE (NOUT,99997) (S(I),I=1,N)
            WRITE (NOUT,*)
*
*           Compute values close to underflow and overflow
*
            SMALL = X02AMF()/(X02AJF()*X02BHF())
            BIG = 1.0D0/SMALL
            IF ((SCOND.LT.0.1D0) .OR. (AMAX.LT.SMALL) .OR. (AMAX.GT.BIG)
     +          ) THEN
*
*              Scale A
*
               IF (UPLO.EQ.'U') THEN
                  DO 80 J = 1, N
                     SJ = S(J)
                     JJ = KD + 1 - J
                     DO 60 I = MAX(1,J-KD), J
                        AB(I+JJ,J) = S(I)*AB(I+JJ,J)*SJ
   60                CONTINUE
   80             CONTINUE
               ELSE IF (UPLO.EQ.'L') THEN
                  DO 120 J = 1, N
                     SJ = S(J)
                     JJ = 1 - J
                     DO 100 I = J, MIN(N,J+KD)
                        AB(I+JJ,J) = S(I)*AB(I+JJ,J)*SJ
  100                CONTINUE
  120             CONTINUE
               END IF
*
*              Print the scaled matrix
*
               IFAIL = 0
               IF (UPLO.EQ.'U') THEN
                  CALL X04CEF(N,N,0,KD,AB,LDAB,'Scaled matrix',IFAIL)
               ELSE IF (UPLO.EQ.'L') THEN
                  CALL X04CEF(N,N,KD,0,AB,LDAB,'Scaled matrix',IFAIL)
               END IF
            END IF
         END IF
      ELSE
         WRITE (NOUT,*) 'NMAX and/or KDMAX too small'
      END IF
      STOP
*
99999 FORMAT (1X,A,I4,A)
99998 FORMAT (1X,2(A,1P,E7.1))
99997 FORMAT ((1X,1P,7E11.1))
      END
```

9.2 Program Data

```
F07HFF Example Program Data
  4  1                                      :Values of N and KD
  5.49E+00   2.68E+10
             5.63E+20  -2.39E+10
                        2.60E+00  -2.22E+00
                                   5.17E+00 :End of matrix A
```

9.3 Program Results

```
 F07HFF Example Program Results

 Matrix A
              1            2            3            4
 1    5.4900E+00   2.6800E+10
 2                 5.6300E+20  -2.3900E+10
 3                              2.6000E+00  -2.2200E+00
 4                                           5.1700E+00

 SCOND = 6.8E-11, AMAX = 5.6E+20

 Diagonal scaling factors
     4.3E-01    4.2E-11    6.2E-01    4.4E-01

 Scaled matrix
            1          2          3          4
 1      1.0000     0.4821
 2                 1.0000    -0.6247
 3                            1.0000    -0.6055
 4                                       1.0000
```