NAG Fortran Library Routine Document

F07FTF (ZPOEQU)

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

F07FTF (ZPOEQU) computes a diagonal scaling matrix S intended to equilibrate a complex n by n Hermitian positive-definite matrix A and reduce its condition number.

2 Specification

```
SUBROUTINE F07FTF (N, A, LDA, S, SCOND, AMAX, INFO)
INTEGER              N, LDA, INFO
double precision     S(*), SCOND, AMAX
complex*16           A(LDA,*)
```

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

3 Description

F07FTF (ZPOEQU) computes the 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: N – INTEGER *Input*

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

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

2: A(LDA,$*$) – ***complex*16*** array *Input*

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

On entry: the matrix A whose scaling factors are to be computed. Only the diagonal elements of the array A are referenced.

3: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F07FTF (ZPOEQU) is called.

Constraint: $\mathrm{LDA} \geq \max(1, \mathrm{N})$.

4: 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.

5: SCOND – ***double precision*** *Output*

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

6: AMAX – ***double precision*** *Output*

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

7: 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 real analogue of this routine is F07FFF (DPOEQU).

9 Example

To equilibrate the Hermitian positive-definite matrix A given by

$$A = \begin{pmatrix} 3.23 & 1.51 - 1.92i & (1.90 + 0.84i) \times 10^5 & 0.42 + 2.50i \\ 1.51 + 1.92i & 3.58 & (-0.23 + 1.11i) \times 10^5 & -1.18 + 1.37i \\ (1.90 - 0.84i) \times 10^5 & (-0.23 - 1.11i) \times 10^5 & 4.09 \times 10^{10} & (2.33 - 0.14i) \times 10^5 \\ 0.42 - 2.50i & -1.18 - 1.37i & (2.33 + 0.14i) \times 10^5 & 4.29 \end{pmatrix}.$$

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

9.1 Program Text

```
*     F07FTF Example Program Text
*     Mark 21 Release. NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX
      PARAMETER        (NMAX=8)
      INTEGER          LDA
      PARAMETER        (LDA=NMAX)
*     .. Local Scalars ..
```

```
      DOUBLE PRECISION AMAX, BIG, SCOND, SJ, SMALL
      INTEGER          I, IFAIL, INFO, J, N
*     .. Local Arrays ..
      COMPLEX *16      A(LDA,NMAX)
      DOUBLE PRECISION S(NMAX)
      CHARACTER        CLABS(1), RLABS(1)
*     .. External Functions ..
      DOUBLE PRECISION X02AJF, X02AMF
      INTEGER          X02BHF
      EXTERNAL         X02AJF, X02AMF, X02BHF
*     .. External Subroutines ..
      EXTERNAL         X04DBF, ZPOEQU
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F07FTF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
*
*        Read the upper triangular part of the matrix A from data file
*
         READ (NIN,*) ((A(I,J),J=I,N),I=1,N)
*
*        Print the matrix A
*
         IFAIL = 0
         CALL X04DBF('Upper','Non-unit',N,N,A,LDA,'Bracketed',
     +              '1P,E10.2','Matrix A','Integer',RLABS,'Integer',
     +              CLABS,80,0,IFAIL)
         WRITE (NOUT,*)
*
*        Compute diagonal scaling factors
*
         CALL ZPOEQU(N,A,LDA,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
*
               DO 40 J = 1, N
                  SJ = S(J)
                  DO 20 I = 1, J
                     A(I,J) = S(I)*A(I,J)*SJ
   20             CONTINUE
   40          CONTINUE
*
*              Print the scaled matrix
*
               IFAIL = 0
               CALL X04DBF('Upper','Non-unit',N,N,A,LDA,'Bracketed',' ',
     +                    'Scaled matrix','Integer',RLABS,'Integer',
     +                    CLABS,80,0,IFAIL)
```

```
*
            END IF
         END IF
      ELSE
         WRITE (NOUT,*) 'NMAX 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

```
F07FTF Example Program Data
   4                                                              :Value of N
 ( 3.23, 0.00) ( 1.51,-1.92) ( 1.90D+05, 0.84D+05) ( 0.42D+00, 2.50D+00)
               ( 3.58, 0.00) (-0.23D+05, 1.11D+05) (-1.18D+00, 1.37D+00)
                             ( 4.09D+10, 0.00D+00) ( 2.33D+05,-0.14D+05)
                                                   ( 4.29D+00, 0.00D+00)
                                                                :End of matrix A
```

9.3 Program Results

```
 F07FTF Example Program Results

 Matrix A
                            1                         2                         3
 1  (  3.23E+00,  0.00E+00) (  1.51E+00, -1.92E+00) (  1.90E+05,  8.40E+04)
 2                           (  3.58E+00,  0.00E+00) ( -2.30E+04,  1.11E+05)
 3                                                   (  4.09E+10,  0.00E+00)
 4

                            4
 1  (  4.20E-01,  2.50E+00)
 2  ( -1.18E+00,  1.37E+00)
 3  (  2.33E+05, -1.40E+04)
 4  (  4.29E+00,  0.00E+00)

 SCOND = 8.9E-06, AMAX = 4.1E+10

 Diagonal scaling factors
     5.6E-01    5.3E-01    4.9E-06    4.8E-01

 Scaled matrix
                         1                    2                    3
 1  (  1.0000,  0.0000) (  0.4441, -0.5646) (  0.5227,  0.2311)
 2                      (  1.0000,  0.0000) ( -0.0601,  0.2901)
 3                                          (  1.0000,  0.0000)
 4

                         4
 1  (  0.1128,  0.6716)
 2  ( -0.3011,  0.3496)
 3  (  0.5562, -0.0334)
 4  (  1.0000,  0.0000)
```