NAG Fortran Library Routine Document

F07UGF (STPCON/DTPCON)

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

F07UGF (STPCON/DTPCON) estimates the condition number of a real triangular matrix, using packed storage.

2 Specification

```
SUBROUTINE F07UGF(NORM, UPLO, DIAG, N, AP, RCOND, WORK, IWORK, INFO)
ENTRY      stpcon (NORM, UPLO, DIAG, N, AP, RCOND, WORK, IWORK, INFO)
INTEGER            N, IWORK(*), INFO
real               AP(*), RCOND, WORK(*)
CHARACTER*1        NORM, UPLO, DIAG
```

The ENTRY statement enables the routine to be called by its LAPACK name.

3 Description

This routine estimates the condition number of a real triangular matrix A, in either the 1-norm or the infinity-norm, using packed storage:

$$\kappa_1(A) = \|A\|_1\|A^{-1}\|_1 \quad \text{or} \quad \kappa_\infty(A) = \|A\|_\infty\|A^{-1}\|_\infty.$$

Note that $\kappa_\infty(A) = \kappa_1(A^T)$.

Because the condition number is infinite if A is singular, the routine actually returns an estimate of the ***reciprocal*** of the condition number.

The routine computes $\|A\|_1$ or $\|A\|_\infty$ exactly, and uses Higham's implementation of Hager's method (Higham (1988)) to estimate $\|A^{-1}\|_1$ or $\|A^{-1}\|_\infty$.

4 References

Higham N J (1988) FORTRAN codes for estimating the one-norm of a real or complex matrix, with applications to condition estimation *ACM Trans. Math. Software* **14** 381–396

5 Parameters

1: NORM – CHARACTER*1 *Input*

On entry: indicates whether $\kappa_1(A)$ or $\kappa_\infty(A)$ is estimated as follows:

if NORM = '1' or 'O', $\kappa_1(A)$ is estimated;

if NORM = 'I', $\kappa_\infty(A)$ is estimated.

Constraint: NORM = '1', 'O' or 'I'.

2: UPLO – CHARACTER*1 *Input*

On entry: indicates whether A is upper or lower triangular as follows:

if UPLO = 'U', A is upper triangular;

if UPLO = 'L', A is lower triangular.

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

3: DIAG – CHARACTER*1 *Input*

On entry: indicates whether A is a non-unit or unit triangular matrix as follows:

if DIAG = 'N', A is a non-unit triangular matrix;

if DIAG = 'U', A is a unit triangular matrix; the diagonal elements are not referenced and are assumed to be 1.

Constraint: DIAG = 'N' or 'U'.

4: N – INTEGER *Input*

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

Constraint: N ≥ 0.

5: AP(*) – ***real*** array *Input*

Note: the dimension of the array AP must be at least $\max(1, \mathrm{N} * (\mathrm{N}+1)/2)$.

On entry: the n by n triangular matrix A, packed by columns. More precisely, if UPLO = 'U', the upper triangle of A must be stored with element a_{ij} in AP$(i+j(j-1)/2)$ for $i \leq j$; if UPLO = 'L', the lower triangle of A must be stored with element a_{ij} in AP$(i+(2n-j)(j-1)/2)$ for $i \geq j$. If DIAG = 'U', the diagonal elements of the matrix are not referenced and are assumed to be 1; the same storage scheme is used whether DIAG = 'N' or 'U'.

6: RCOND – ***real*** *Output*

On exit: an estimate of the reciprocal of the condition number of A. RCOND is set to zero if exact singularity is detected or the estimate underflows. If RCOND is less than ***machine precision***, A is singular to working precision.

7: WORK(*) – ***real*** array *Workspace*

Note: the dimension of the array WORK must be at least $\max(1, 3 * \mathrm{N})$.

8: IWORK(*) – INTEGER array *Workspace*

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

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 parameter had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

The computed estimate RCOND is never less than the true value ρ, and in practice is nearly always less than 10ρ, although examples can be constructed where RCOND is much larger.

8 Further Comments

A call to this routine involves solving a number of systems of linear equations of the form $Ax = b$ or $A^T x = b$; the number is usually 4 or 5 and never more than 11. Each solution involves approximately n^2

floating-point operations but takes considerably longer than a call to F07UEF (STPTRS/DTPTRS) with one right-hand side, because extra care is taken to avoid overflow when A is approximately singular.

The complex analogue of this routine is F07UUF (CTPCON/ZTPCON).

9 Example

To estimate the condition number in the 1-norm of the matrix A, where

$$A = \begin{pmatrix} 4.30 & 0.00 & 0.00 & 0.00 \\ -3.96 & -4.87 & 0.00 & 0.00 \\ 0.40 & 0.31 & -8.02 & 0.00 \\ -0.27 & 0.07 & -5.95 & 0.12 \end{pmatrix},$$

using packed storage. The true condition number in the 1-norm is 116.41.

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     F07UGF Example Program Text
*     Mark 15 Release. NAG Copyright 1991.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX
      PARAMETER        (NMAX=8)
      CHARACTER        NORM, DIAG
      PARAMETER        (NORM='1',DIAG='N')
*     .. Local Scalars ..
      real             RCOND
      INTEGER          I, INFO, J, N
      CHARACTER        UPLO
*     .. Local Arrays ..
      real             AP(NMAX*(NMAX+1)/2), WORK(3*NMAX)
      INTEGER          IWORK(NMAX)
*     .. External Functions ..
      real             X02AJF
      EXTERNAL         X02AJF
*     .. External Subroutines ..
      EXTERNAL         stpcon
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F07UGF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
*
*        Read A from data file
*
         READ (NIN,*) UPLO
         IF (UPLO.EQ.'U') THEN
            READ (NIN,*) ((AP(I+J*(J-1)/2),J=I,N),I=1,N)
         ELSE IF (UPLO.EQ.'L') THEN
            READ (NIN,*) ((AP(I+(2*N-J)*(J-1)/2),J=1,I),I=1,N)
         END IF
*
*        Estimate condition number
*
         CALL stpcon(NORM,UPLO,DIAG,N,AP,RCOND,WORK,IWORK,INFO)
*
         WRITE (NOUT,*)
         IF (RCOND.GE.X02AJF()) THEN
            WRITE (NOUT,99999) 'Estimate of condition number =',
     +        1.0e0/RCOND
         ELSE
            WRITE (NOUT,*) 'A is singular to working precision'
```

```
         END IF
      END IF
      STOP
*
99999 FORMAT (1X,A,1P,e10.2)
      END
```

9.2 Program Data

```
F07UGF Example Program Data
  4                           :Value of N
  'L'                         :Value of UPLO
  4.30
 -3.96  -4.87
  0.40   0.31  -8.02
 -0.27   0.07  -5.95   0.12   :End of matrix A
```

9.3 Program Results

```
 F07UGF Example Program Results

 Estimate of condition number =  1.16E+02
```