F07FUF (CPOCON/ZPOCON) – NAG Fortran Library Routine Document

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

F07FUF (CPOCON/ZPOCON) estimates the condition number of a complex Hermitian positive-definite matrix A, where A has been factorized by F07FRF (CPOTRF/ZPOTRF).

2 Specification

```
SUBROUTINE F07FUF(UPLO, N, A, LDA, ANORM, RCOND, WORK, RWORK, INFO)
ENTRY      cpocon(UPLO, N, A, LDA, ANORM, RCOND, WORK, RWORK, INFO)
INTEGER            N, LDA, INFO
real               ANORM, RCOND, RWORK(*)
complex            A(LDA,*), WORK(*)
CHARACTER*1        UPLO
```

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

3 Description

This routine estimates the condition number (in the 1-norm) of a complex Hermitian positive-definite matrix A:

$$\kappa_1(A) = \|A\|_1 \|A^{-1}\|_1.$$

Since A is Hermitian, $\kappa_1(A) = \kappa_\infty(A) = \|A\|_\infty \|A^{-1}\|_\infty$.

Because $\kappa_1(A)$ is infinite if A is singular, the routine actually returns an estimate of the **reciprocal** of $\kappa_1(A)$.

The routine should be preceded by a call to F06UCF to compute $\|A\|_1$ and a call to F07FRF (CPOTRF/ZPOTRF) to compute the Cholesky factorization of A. The routine then uses Higham's implementation of Hager's method [1] to estimate $\|A^{-1}\|_1$.

4 References

[1] 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: UPLO — CHARACTER*1 *Input*

On entry: indicates whether A has been factorized as $U^H U$ or LL^H as follows:

if UPLO = 'U', then $A = U^H U$, where U is upper triangular;
if UPLO = 'L', then $A = LL^H$, where L is lower triangular.

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

2: N — INTEGER *Input*

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

Constraint: N ≥ 0.

3: A(LDA,∗) — ***complex*** array *Input*

Note: the second dimension of the array A must be at least max(1,N).

On entry: the Cholesky factor of A, as returned by F07FRF (CPOTRF/ZPOTRF).

4: LDA — INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F07FUF (CPOCON/ZPOCON) is called.

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

5: ANORM — ***real*** *Input*

On entry: the 1-norm of the **original** matrix A, which may be computed by calling F06UCF. ANORM must be computed either **before** calling F07FRF (CPOTRF/ZPOTRF) or else from a copy of the original matrix A.

Constraint: ANORM ≥ 0.0.

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***, then A is singular to working precision.

7: WORK(∗) — ***complex*** array *Workspace*

Note: the dimension of the array WORK must be at least max(1,2∗N).

8: RWORK(∗) — ***real*** array *Workspace*

Note: the dimension of the array RWORK must be at least max(1,N).

9: 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 $= -i$, 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 $Ax = b$; the number is usually 5 and never more than 11. Each solution involves approximately $8n^2$ real floating-point operations but takes considerably longer than a call to F07FSF (CPOTRS/ZPOTRS) with 1 right-hand side, because extra care is taken to avoid overflow when A is approximately singular.

The real analogue of this routine is F07FGF (SPOCON/DPOCON).

9 Example

To estimate the condition number in the 1-norm (or infinity-norm) of the matrix A, where

$$A = \begin{pmatrix} 3.23 + 0.00i & 1.51 - 1.92i & 1.90 + 0.84i & 0.42 + 2.50i \\ 1.51 + 1.92i & 3.58 + 0.00i & -0.23 + 1.11i & -1.18 + 1.37i \\ 1.90 - 0.84i & -0.23 - 1.11i & 4.09 + 0.00i & 2.33 - 0.14i \\ 0.42 - 2.50i & -1.18 - 1.37i & 2.33 + 0.14i & 4.29 + 0.00i \end{pmatrix}.$$

Here A is Hermitian positive-definite and must first be factorized by F07FRF (CPOTRF/ZPOTRF). The true condition number in the 1-norm is 201.92.

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.

```
*     F07FUF Example Program Text
*     Mark 16 Release. NAG Copyright 1993.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, LDA
      PARAMETER        (NMAX=8,LDA=NMAX)
*     .. Local Scalars ..
      real             ANORM, RCOND
      INTEGER          I, INFO, J, N
      CHARACTER        UPLO
*     .. Local Arrays ..
      complex          A(LDA,NMAX), WORK(2*NMAX)
      real             RWORK(NMAX)
*     .. External Functions ..
      real             F06UCF, X02AJF
      EXTERNAL         F06UCF, X02AJF
*     .. External Subroutines ..
      EXTERNAL         cpocon, cpotrf
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F07FUF 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,*) ((A(I,J),J=I,N),I=1,N)
         ELSE IF (UPLO.EQ.'L') THEN
            READ (NIN,*) ((A(I,J),J=1,I),I=1,N)
         END IF
*
*        Compute norm of A
*
         ANORM = F06UCF('1-norm',UPLO,N,A,LDA,RWORK)
*
```

```
*        Factorize A
*
         CALL cpotrf(UPLO,N,A,LDA,INFO)
*
         WRITE (NOUT,*)
         IF (INFO.EQ.0) THEN
*
*           Estimate condition number
*
            CALL cpocon(UPLO,N,A,LDA,ANORM,RCOND,WORK,RWORK,INFO)
*
            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
         ELSE
            WRITE (NOUT,*) 'A is not positive-definite'
         END IF
      END IF
      STOP
*
99999 FORMAT (1X,A,1P,e10.2)
      END
```

9.2 Program Data

```
F07FUF Example Program Data
  4                                                          :Value of N
  'L'                                                        :Value of UPLO
 (3.23, 0.00)
 (1.51, 1.92) ( 3.58, 0.00)
 (1.90,-0.84) (-0.23,-1.11) ( 4.09, 0.00)
 (0.42,-2.50) (-1.18,-1.37) ( 2.33, 0.14) ( 4.29, 0.00)  :End of matrix A
```

9.3 Program Results

```
 F07FUF Example Program Results

 Estimate of condition number =  1.51E+02
```