NAG Fortran Library Routine Document

F07GUF (CPPCON/ZPPCON)

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

F07GUF (CPPCON/ZPPCON) estimates the condition number of a complex Hermitian positive-definite matrix A, where A has been factorized by F07GRF (CPPTRF/ZPPTRF), using packed storage.

2 Specification

```
SUBROUTINE F07GUF(UPLO, N, AP, ANORM, RCOND, WORK, RWORK, INFO)
ENTRY      cppcon (UPLO, N, AP, ANORM, RCOND, WORK, RWORK, INFO)
INTEGER             N, INFO
real                ANORM, RCOND, RWORK(*)
complex             AP(*), 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 F06UDF to compute $\|A\|_1$ and a call to F07GRF (CPPTRF/ZPPTRF) to compute the Cholesky factorization of A. The routine then uses Higham's implementation of Hager's method (see Higham (1988)) to estimate $\|A^{-1}\|_1$.

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: UPLO – CHARACTER*1 *Input*

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

if UPLO = 'U', $A = U^H U$, where U is upper triangular;

if UPLO = 'L', $A = LL^H$, 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: AP(*) – ***complex*** array *Input*

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

On entry: the Cholesky factor of A stored in packed form, as returned by F07GRF (CPPTRF/ZPPTRF).

4: ANORM – ***real*** *Input*

On entry: the 1-norm of the **original** matrix A, which may be computed by calling F06UDF. ANORM must be computed either **before** calling F07GRF (CPPTRF/ZPPTRF) or else from a copy of the original matrix A.

Constraint: ANORM ≥ 0.0.

5: 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.

6: WORK(*) – ***complex*** array *Workspace*

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

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

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

8: 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 $Ax = b$; 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 F07GSF (CPPTRS/ZPPTRS) 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 F07GGF (SPPCON/DPPCON).

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, stored in packed form, and must first be factorized by F07GRF (CPPTRF/ZPPTRF). 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.

```
*     F07GUF Example Program Text
*     Mark 15 Release. NAG Copyright 1991.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX
      PARAMETER        (NMAX=8)
*     .. Local Scalars ..
      real             ANORM, RCOND
      INTEGER          I, INFO, J, N
      CHARACTER        UPLO
*     .. Local Arrays ..
      complex          AP(NMAX*(NMAX+1)/2), WORK(2*NMAX)
      real             RWORK(NMAX)
*     .. External Functions ..
      real             F06UDF, X02AJF
      EXTERNAL         F06UDF, X02AJF
*     .. External Subroutines ..
      EXTERNAL         cppcon, cpptrf
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F07GUF 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
*
*        Compute norm of A
*
         ANORM = F06UDF('1-norm',UPLO,N,AP,RWORK)
*
*        Factorize A
*
         CALL cpptrf(UPLO,N,AP,INFO)
*
         WRITE (NOUT,*)
         IF (INFO.EQ.0) THEN
*
*           Estimate condition number
*
            CALL cppcon(UPLO,N,AP,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

```
F07GUF 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

```
 F07GUF Example Program Results

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