NAG Library Routine Document

F07MUF (ZHECON)

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

F07MUF (ZHECON) estimates the condition number of a complex Hermitian indefinite matrix A, where A has been factorized by F07MRF (ZHETRF).

2 Specification

```
SUBROUTINE F07MUF (UPLO, N, A, LDA, IPIV, ANORM, RCOND, WORK, INFO)
INTEGER                 N, LDA, IPIV(*), INFO
REAL (KIND=nag_wp)      ANORM, RCOND
COMPLEX (KIND=nag_wp) A(LDA,*), WORK(2*N)
CHARACTER(1)            UPLO
```

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

3 Description

F07MUF (ZHECON) estimates the condition number (in the 1-norm) of a complex Hermitian indefinite matrix A:

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

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

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 F07MRF (ZHETRF) to compute the Bunch–Kaufman factorization of A. The routine then uses Higham's implementation of Hager's method (see Higham (1988)) to estimate $\left\|A^{-1}\right\|_1$.

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 Arguments

1: UPLO – CHARACTER(1) *Input*

On entry: specifies how A has been factorized.

UPLO = 'U'
: $A = PUDU^{\mathrm{H}}P^{\mathrm{T}}$, where U is upper triangular.

UPLO = 'L'
: $A = PLDL^{\mathrm{H}}P^{\mathrm{T}}$, where L is lower triangular.

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

2: N – INTEGER *Input*

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

Constraint: $N \geq 0$.

3: A(LDA, $*$) – COMPLEX (KIND=nag_wp) array *Input*

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

On entry: details of the factorization of A, as returned by F07MRF (ZHETRF).

4: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F07MUF (ZHECON) is called.

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

5: IPIV($*$) – INTEGER array *Input*

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

On entry: details of the interchanges and the block structure of D, as returned by F07MRF (ZHETRF).

6: ANORM – REAL (KIND=nag_wp) *Input*

On entry: the 1-norm of the **original** matrix A, which may be computed by calling F06UCF with its argument NORM = '1'. ANORM must be computed either **before** calling F07MRF (ZHETRF) or else from a **copy** of the original matrix A.

Constraint: $ANORM \geq 0.0$.

7: RCOND – REAL (KIND=nag_wp) *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.

8: WORK($2 \times N$) – COMPLEX (KIND=nag_wp) array *Workspace*

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 $= -i$, argument i 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 Parallelism and Performance

F07MUF (ZHECON) makes calls to BLAS and/or LAPACK routines, which may be threaded within the vendor library used by this implementation. Consult the documentation for the vendor library for further information.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9 Further Comments

A call to F07MUF (ZHECON) 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 F07MSF (ZHETRS) with one right-hand side, because extra care is taken to avoid overflow when A is approximately singular.

The real analogue of this routine is F07MGF (DSYCON).

10 Example

This example estimates the condition number in the 1-norm (or ∞-norm) of the matrix A, where

$$A = \begin{pmatrix} -1.36 + 0.00i & 1.58 + 0.90i & 2.21 - 0.21i & 3.91 + 1.50i \\ 1.58 - 0.90i & -8.87 + 0.00i & -1.84 - 0.03i & -1.78 + 1.18i \\ 2.21 + 0.21i & -1.84 + 0.03i & -4.63 + 0.00i & 0.11 + 0.11i \\ 3.91 - 1.50i & -1.78 - 1.18i & 0.11 - 0.11i & -1.84 + 0.00i \end{pmatrix}.$$

Here A is Hermitian indefinite and must first be factorized by F07MRF (ZHETRF). The true condition number in the 1-norm is 9.10.

10.1 Program Text

```
    Program f07mufe

!     F07MUF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, x02ajf, zhecon, zhetrf, zlanhe => f06ucf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: anorm, rcond
      Integer                          :: i, info, lda, lwork, n
      Character (1)                    :: uplo
!     .. Local Arrays ..
      Complex (Kind=nag_wp), Allocatable :: a(:,:), work(:)
      Real (Kind=nag_wp), Allocatable  :: rwork(:)
      Integer, Allocatable             :: ipiv(:)
!     .. Executable Statements ..
      Write (nout,*) 'F07MUF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n
      lda = n
      lwork = 64*n
      Allocate (a(lda,n),work(lwork),rwork(n),ipiv(n))

!     Read A from data file

      Read (nin,*) uplo
      If (uplo=='U') Then
        Read (nin,*)(a(i,i:n),i=1,n)
      Else If (uplo=='L') Then
        Read (nin,*)(a(i,1:i),i=1,n)
      End If

!     Compute norm of A
```

```
!     f06ucf is the NAG name equivalent of the LAPACK auxiliary zlanhe
      anorm = zlanhe('1-norm',uplo,n,a,lda,rwork)

!     Factorize A
!     The NAG name equivalent of zhetrf is f07mrf
      Call zhetrf(uplo,n,a,lda,ipiv,work,lwork,info)

      Write (nout,*)
      If (info==0) Then

!       Estimate condition number
!       The NAG name equivalent of zhecon is f07muf
        Call zhecon(uplo,n,a,lda,ipiv,anorm,rcond,work,info)

        If (rcond>=x02ajf()) Then
          Write (nout,99999) 'Estimate of condition number =',                 &
            1.0_nag_wp/rcond
        Else
          Write (nout,*) 'A is singular to working precision'
        End If
      Else
        Write (nout,*) 'The factor D is singular'
      End If

99999 Format (1X,A,1P,E10.2)
    End Program f07mufe
```

10.2 Program Data

```
F07MUF Example Program Data
  4                                                       :Value of N
  'L'                                                     :Value of UPLO
 (-1.36, 0.00)
 ( 1.58,-0.90) (-8.87, 0.00)
 ( 2.21, 0.21) (-1.84, 0.03) (-4.63, 0.00)
 ( 3.91,-1.50) (-1.78,-1.18) ( 0.11,-0.11) (-1.84, 0.00)  :End of matrix A
```

10.3 Program Results

```
 F07MUF Example Program Results

 Estimate of condition number =  6.68E+00
```