NAG Library Routine Document

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

2 Specification

```
SUBROUTINE F07GUF (UPLO, N, AP, ANORM, RCOND, WORK, RWORK, INFO)

INTEGER                 N, INFO
REAL (KIND=nag_wp)      ANORM, RCOND, RWORK(N)
COMPLEX (KIND=nag_wp)   AP(*), WORK(2*N)
CHARACTER(1)            UPLO
```

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

3 Description

F07GUF (ZPPCON) estimates the condition number (in the 1-norm) of a complex Hermitian positive definite 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 F06UDF to compute $\|A\|_1$ and a call to F07GRF (ZPPTRF) to compute the Cholesky 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 Parameters

1: UPLO – CHARACTER(1) *Input*

On entry: specifies how A has been factorized.

UPLO = 'U'
$A = U^{\mathrm{H}}U$, where U is upper triangular.

UPLO = 'L'
$A = LL^{\mathrm{H}}$, where L is lower triangular.

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

2: N – INTEGER *Input*

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

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

3: AP$(*)$ – COMPLEX (KIND=nag_wp) array *Input*

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

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

4: ANORM – REAL (KIND=nag_wp) *Input*

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

Constraint: ANORM ≥ 0.0.

5: 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.

6: WORK$(2 \times \mathrm{N})$ – COMPLEX (KIND=nag_wp) array *Workspace*

7: RWORK(N) – REAL (KIND=nag_wp) array *Workspace*

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 F07GUF (ZPPCON) 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 (ZPPTRS) 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 F07GGF (DPPCON).

9 Example

This example estimates the condition number in the 1-norm (or ∞-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 (ZPPTRF). The true condition number in the 1-norm is 201.92.

9.1 Program Text

```
    Program f07gufe

!     F07GUF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, x02ajf, zlanhp => f06udf, zppcon, zpptrf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: anorm, rcond
      Integer                          :: i, info, j, n
      Character (1)                    :: uplo
!     .. Local Arrays ..
      Complex (Kind=nag_wp), Allocatable :: ap(:), work(:)
      Real (Kind=nag_wp), Allocatable  :: rwork(:)
!     .. Executable Statements ..
      Write (nout,*) 'F07GUF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n

      Allocate (ap(n*(n+1)/2),work(2*n),rwork(n))

!     Read A from data file

      Read (nin,*) uplo
      If (uplo=='U') Then
        Read (nin,*)((ap(i+j*(j-1)/2),j=i,n),i=1,n)
      Else If (uplo=='L') Then
        Read (nin,*)((ap(i+(2*n-j)*(j-1)/2),j=1,i),i=1,n)
      End If

!     Compute norm of A
!     f06udf is the NAG name equivalent of the LAPACK auxiliary zlanhp
      anorm = zlanhp('1-norm',uplo,n,ap,rwork)

!     Factorize A
!     The NAG name equivalent of zpptrf is f07grf
      Call zpptrf(uplo,n,ap,info)

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

!       Estimate condition number

!       The NAG name equivalent of zppcon is f07guf
        Call zppcon(uplo,n,ap,anorm,rcond,work,rwork,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,*) 'A is not positive definite'
      End If

99999 Format (1X,A,1P,E10.2)
    End Program f07gufe
```

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
```