NAG Library Routine Document

F07CUF (ZGTCON)

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

F07CUF (ZGTCON) estimates the reciprocal condition number of a complex n by n tridiagonal matrix A, using the LU factorization returned by F07CRF (ZGTTRF).

2 Specification

```
SUBROUTINE F07CUF (NORM, N, DL, D, DU, DU2, IPIV, ANORM, RCOND, WORK,      &
                   INFO)

INTEGER                 N, IPIV(*), INFO
REAL (KIND=nag_wp)      ANORM, RCOND
COMPLEX (KIND=nag_wp) DL(*), D(*), DU(*), DU2(*), WORK(2*N)
CHARACTER(1)            NORM
```

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

3 Description

F07CUF (ZGTCON) should be preceded by a call to F07CRF (ZGTTRF), which uses Gaussian elimination with partial pivoting and row interchanges to factorize the matrix A as

$$A = PLU,$$

where P is a permutation matrix, L is unit lower triangular with at most one nonzero subdiagonal element in each column, and U is an upper triangular band matrix, with two superdiagonals. F07CUF (ZGTCON) then utilizes the factorization to estimate either $\|A^{-1}\|_1$ or $\|A^{-1}\|_\infty$, from which the estimate of the reciprocal of the condition number of A, $1/\kappa(A)$ is computed as either

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

or

$$1/\kappa_\infty(A) = 1/\left(\|A\|_\infty\|A^{-1}\|_\infty\right).$$

$1/\kappa(A)$ is returned, rather than $\kappa(A)$, since when A is singular $\kappa(A)$ is infinite.

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

4 References

Higham N J (2002) *Accuracy and Stability of Numerical Algorithms* (2nd Edition) SIAM, Philadelphia

5 Parameters

1: NORM – CHARACTER(1) *Input*

On entry: specifies the norm to be used to estimate $\kappa(A)$.

NORM = '1' or 'O'
Estimate $\kappa_1(A)$.

NORM = 'I'
Estimate $\kappa_\infty(A)$.

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

2: N – INTEGER *Input*

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

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

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

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

On entry: must contain the $(n - 1)$ multipliers that define the matrix L of the LU factorization of A.

4: D($*$) – COMPLEX (KIND=nag_wp) array *Input*

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

On entry: must contain the n diagonal elements of the upper triangular matrix U from the LU factorization of A.

5: DU($*$) – COMPLEX (KIND=nag_wp) array *Input*

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

On entry: must contain the $(n - 1)$ elements of the first superdiagonal of U.

6: DU2($*$) – COMPLEX (KIND=nag_wp) array *Input*

Note: the dimension of the array DU2 must be at least $\max(1, \mathrm{N} - 2)$.

On entry: must contain the $(n - 2)$ elements of the second superdiagonal of U.

7: IPIV($*$) – INTEGER array *Input*

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

On entry: must contain the n pivot indices that define the permutation matrix P. At the ith step, row i of the matrix was interchanged with row IPIV(i), and IPIV(i) must always be either i or $(i + 1)$, IPIV(i) $= i$ indicating that a row interchange was not performed.

8: ANORM – REAL (KIND=nag_wp) *Input*

On entry: if NORM = '1' or 'O', the 1-norm of the **original** matrix A.

If NORM = 'I', the ∞-norm of the **original** matrix A.

ANORM may be computed by calling F06UNF with the same value for the parameter NORM.

ANORM must be computed either **before** calling F07CRF (ZGTTRF) or else from a **copy** of the original matrix A (see Section 10).

Constraint: ANORM ≥ 0.0.

9: RCOND – REAL (KIND=nag_wp) *Output*

On exit: contains an estimate of the reciprocal condition number.

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

11: 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

In practice the condition number estimator is very reliable, but it can underestimate the true condition number; see Section 15.3 of Higham (2002) for further details.

8 Parallelism and Performance

F07CUF (ZGTCON) is not threaded by NAG in any implementation.

F07CUF (ZGTCON) 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

The condition number estimation typically requires between four and five solves and never more than eleven solves, following the factorization. The total number of floating-point operations required to perform a solve is proportional to n.

The real analogue of this routine is F07CGF (DGTCON).

10 Example

This example estimates the condition number in the 1-norm of the tridiagonal matrix A given by

$$A = \begin{pmatrix} -1.3+1.3i & 2.0-1.0i & 0 & 0 & 0 \\ 1.0-2.0i & -1.3+1.3i & 2.0+1.0i & 0 & 0 \\ 0 & 1.0+1.0i & -1.3+3.3i & -1.0+1.0i & 0 \\ 0 & 0 & 2.0-3.0i & -0.3+4.3i & 1.0-1.0i \\ 0 & 0 & 0 & 1.0+1.0i & -3.3+1.3i \end{pmatrix}.$$

10.1 Program Text

```
    Program f07cufe

!     F07CUF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: f06unf, nag_wp, x02ajf, zgtcon, zgttrf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: anorm, rcond
      Integer                          :: info, n
!     .. Local Arrays ..
      Complex (Kind=nag_wp), Allocatable :: d(:), dl(:), du(:), du2(:), work(:)
      Integer, Allocatable             :: ipiv(:)
!     .. Executable Statements ..
```

```
      Write (nout,*) 'F07CUF Example Program Results'
      Write (nout,*)
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n

      Allocate (d(n),dl(n-1),du(n-1),du2(n-2),work(2*n),ipiv(n))

!     Read the tridiagonal matrix A from data file

      Read (nin,*) du(1:n-1)
      Read (nin,*) d(1:n)
      Read (nin,*) dl(1:n-1)

!     Compute the 1-norm of A
      anorm = f06unf('1-norm',n,dl,d,du)

!     Factorize the tridiagonal matrix A
!     The NAG name equivalent of zgttrf is f07crf
      Call zgttrf(n,dl,d,du,du2,ipiv,info)

      If (info==0) Then

!       Estimate the condition number of A
!       The NAG name equivalent of zgtcon is f07cuf
        Call zgtcon('1-norm',n,dl,d,du,du2,ipiv,anorm,rcond,work,info)

!       Print the estimated condition number

        If (rcond>=x02ajf()) Then
          Write (nout,99999) 'Estimate of condition number = ', &
            1.0E0_nag_wp/rcond
        Else
          Write (nout,99999) 'A is singular to working precision. RCOND = ', &
            rcond
        End If

      Else
        Write (nout,99998) 'The (', info, ',', info, ')', &
          ' element of the factor U is zero'
      End If

99999 Format (1X,A,1P,E10.2)
99998 Format (1X,A,I3,A,I3,A,A)
    End Program f07cufe
```

10.2 Program Data

```
F07CUF Example Program Data
   5                                                :Value of N
 ( 2.0,-1.0) ( 2.0, 1.0) (-1.0, 1.0) ( 1.0,-1.0) :End of DU
 (-1.3, 1.3) (-1.3, 1.3) (-1.3, 3.3) (-0.3, 4.3)
 (-3.3, 1.3)                                      :End of D
 ( 1.0,-2.0) ( 1.0, 1.0) ( 2.0,-3.0) ( 1.0, 1.0) :End of DL
```

10.3 Program Results

```
 F07CUF Example Program Results

 Estimate of condition number =   1.84E+02
```