NAG Fortran Library Routine Document

F07USF (CTPTRS/ZTPTRS)

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

F07USF (CTPTRS/ZTPTRS) solves a complex triangular system of linear equations with multiple right-hand sides, $AX = B$, $A^T X = B$ or $A^H X = B$, using packed storage.

2 Specification

```
SUBROUTINE F07USF(UPLO, TRANS, DIAG, N, NRHS, AP, B, LDB, INFO)
ENTRY      ctptrs  (UPLO, TRANS, DIAG, N, NRHS, AP, B, LDB, INFO)
INTEGER            N, NRHS, LDB, INFO
complex            AP(*), B(LDB,*)
CHARACTER*1        UPLO, TRANS, DIAG
```

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

3 Description

This routine solves a complex triangular system of linear equations $AX = B$, $A^T X = B$ or $A^H X = B$ using packed storage.

4 References

Golub G H and van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

Higham N J (1989) The accuracy of solutions to triangular systems *SIAM J. Numer. Anal.* **26** 1252–1265

5 Parameters

1: UPLO – CHARACTER*1 *Input*

On entry: indicates whether A is upper or lower triangular as follows:

if UPLO = 'U', A is upper triangular;

if UPLO = 'L', A is lower triangular.

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

2: TRANS – CHARACTER*1 *Input*

On entry: indicates the form of the equations as follows:

if TRANS = 'N', the equations are of the form $AX = B$;

if TRANS = 'T', the equations are of the form $A^T X = B$;

if TRANS = 'C', the equations are of the form $A^H X = B$.

Constraint: TRANS = 'N', 'T' or 'C'.

3: DIAG – CHARACTER*1 *Input*

On entry: indicates whether A is a non-unit or unit triangular matrix as follows:

if DIAG = 'N', A is a non-unit triangular matrix;

if DIAG = 'U', A is a unit triangular matrix; the diagonal elements are not referenced and are assumed to be 1.

Constraint: DIAG = 'N' or 'U'.

4: N – INTEGER *Input*

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

Constraint: N ≥ 0.

5: NRHS – INTEGER *Input*

On entry: r, the number of right-hand sides.

Constraint: NRHS ≥ 0.

6: AP(*) – ***complex*** array *Input*

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

On entry: the n by n triangular matrix A, packed by columns. More precisely, if UPLO = 'U', the upper triangle of A must be stored with element a_{ij} in AP$(i+j(j-1)/2)$ for $i \leq j$; if UPLO = 'L', the lower triangle of A must be stored with element a_{ij} in AP$(i+(2n-j)(j-1)/2)$ for $i \geq j$. If DIAG = 'U', the diagonal elements of the matrix are not referenced and are assumed to be 1; the same storage scheme is used whether DIAG = 'N' or 'U'.

7: B(LDB,*) – ***complex*** array *Input/Output*

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

On entry: the n by r right-hand side matrix B.

On exit: the n by r solution matrix X.

8: LDB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F07USF (CTPTRS/ZTPTRS) is called.

Constraint: LDB $\geq \max(1, \mathrm{N})$.

9: 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.

INFO > 0

If INFO $= i$, a_{ii} is zero and the matrix A is singular.

7 Accuracy

The solutions of triangular systems of equations are usually computed to high accuracy. See Higham (1989).

For each right-hand side vector b, the computed solution x is the exact solution of a perturbed system of equations $(A + E)x = b$, where

$$|E| \leq c(n)\epsilon|A|,$$

$c(n)$ is a modest linear function of n, and ϵ is the ***machine precision***.

If $\hat{x}$ is the true solution, then the computed solution x satisfies a forward error bound of the form

$$\frac{\|x - \hat{x}\|_\infty}{\|x\|_\infty} \leq c(n)\,\mathrm{cond}(A, x)\epsilon, \quad \text{provided} \quad c(n)\,\mathrm{cond}(A, x)\epsilon < 1,$$

where $\mathrm{cond}(A, x) = \||A^{-1}|\,|A|\,|x|\|_\infty / \|x\|_\infty$.

Note that $\mathrm{cond}(A, x) \leq \mathrm{cond}(A) = \||A^{-1}|\,|A|\|_\infty \leq \kappa_\infty(A)$; $\mathrm{cond}(A, x)$ can be much smaller than $\mathrm{cond}(A)$ and it is also possible for $\mathrm{cond}(A^H)$, which is the same as $\mathrm{cond}(A^T)$, to be much larger (or smaller) than $\mathrm{cond}(A)$.

Forward and backward error bounds can be computed by calling F07UVF (CTPRFS/ZTPRFS), and an estimate for $\kappa_\infty(A)$ can be obtained by calling F07UUF (CTPCON/ZTPCON) with NORM = 'I'.

8 Further Comments

The total number of real floating-point operations is approximately $4n^2r$.

The real analogue of this routine is F07UEF (STPTRS/DTPTRS).

9 Example

To solve the system of equations $AX = B$, where

$$A = \begin{pmatrix} 4.78 + 4.56i & 0.00 + 0.00i & 0.00 + 0.00i & 0.00 + 0.00i \\ 2.00 - 0.30i & -4.11 + 1.25i & 0.00 + 0.00i & 0.00 + 0.00i \\ 2.89 - 1.34i & 2.36 - 4.25i & 4.15 + 0.80i & 0.00 + 0.00i \\ -1.89 + 1.15i & 0.04 - 3.69i & -0.02 + 0.46i & 0.33 - 0.26i \end{pmatrix}$$

and

$$B = \begin{pmatrix} -14.78 - 32.36i & -18.02 + 28.46i \\ 2.98 - 2.14i & 14.22 + 15.42i \\ -20.96 + 17.06i & 5.62 + 35.89i \\ 9.54 + 9.91i & -16.46 - 1.73i \end{pmatrix},$$

using packed storage for A.

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.

```
*     F07USF Example Program Text
*     Mark 15 Release. NAG Copyright 1991.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, NRHMAX, LDB
      PARAMETER        (NMAX=8,NRHMAX=NMAX,LDB=NMAX)
      CHARACTER        TRANS, DIAG
      PARAMETER        (TRANS='N',DIAG='N')
*     .. Local Scalars ..
```

```
      INTEGER          I, IFAIL, INFO, J, N, NRHS
      CHARACTER        UPLO
*     .. Local Arrays ..
      complex          AP(NMAX*(NMAX+1)/2), B(LDB,NRHMAX)
      CHARACTER        CLABS(1), RLABS(1)
*     .. External Subroutines ..
      EXTERNAL         ctptrs, X04DBF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F07USF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N, NRHS
      IF (N.LE.NMAX .AND. NRHS.LE.NRHMAX) THEN
*
*        Read A and B 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
         READ (NIN,*) ((B(I,J),J=1,NRHS),I=1,N)
*
*        Compute solution
*
         CALL ctptrs(UPLO,TRANS,DIAG,N,NRHS,AP,B,LDB,INFO)
*
*        Print solution
*
         WRITE (NOUT,*)
         IF (INFO.EQ.0) THEN
            IFAIL = 0
            CALL X04DBF('General',' ',N,NRHS,B,LDB,'Bracketed','F7.4',
     +                  'Solution(s)','Integer',RLABS,'Integer',CLABS,
     +                  80,0,IFAIL)
         ELSE
            WRITE (NOUT,*) 'A is singular'
         END IF
      END IF
      STOP
*
      END
```

9.2 Program Data

```
F07USF Example Program Data
  4  2                                                    :Values of N and NRHS
  'L'                                                     :Value of UPLO
 ( 4.78, 4.56)
 ( 2.00,-0.30) (-4.11, 1.25)
 ( 2.89,-1.34) ( 2.36,-4.25) ( 4.15, 0.80)
 (-1.89, 1.15) ( 0.04,-3.69) (-0.02, 0.46) ( 0.33,-0.26)  :End of matrix A
 (-14.78,-32.36) (-18.02, 28.46)
 (  2.98, -2.14) ( 14.22, 15.42)
 (-20.96, 17.06) (  5.62, 35.89)
 (  9.54,  9.91) (-16.46, -1.73)                          :End of matrix B
```

9.3 Program Results

```
 F07USF Example Program Results

 Solution(s)
                     1                   2
 1  (-5.0000,-2.0000) ( 1.0000, 5.0000)
 2  (-3.0000,-1.0000) (-2.0000,-2.0000)
 3  ( 2.0000, 1.0000) ( 3.0000, 4.0000)
 4  ( 4.0000, 3.0000) ( 4.0000,-3.0000)
```