NAG Library Routine Document

F07ASF (ZGETRS)

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

F07ASF (ZGETRS) solves a complex system of linear equations with multiple right-hand sides,

$$AX = B, \quad A^{\mathrm{T}}X = B \quad \text{or} \quad A^{\mathrm{H}}X = B,$$

where A has been factorized by F07ARF (ZGETRF).

2 Specification

```
SUBROUTINE F07ASF (TRANS, N, NRHS, A, LDA, IPIV, B, LDB, INFO)

INTEGER                 N, NRHS, LDA, IPIV(*), LDB, INFO
COMPLEX (KIND=nag_wp) A(LDA,*), B(LDB,*)
CHARACTER(1)            TRANS
```

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

3 Description

F07ASF (ZGETRS) is used to solve a complex system of linear equations $AX = B$, $A^{\mathrm{T}}X = B$ or $A^{\mathrm{H}}X = B$, the routine must be preceded by a call to F07ARF (ZGETRF) which computes the LU factorization of A as $A = PLU$. The solution is computed by forward and backward substitution.

If TRANS = 'N', the solution is computed by solving $PLY = B$ and then $UX = Y$.

If TRANS = 'T', the solution is computed by solving $U^{\mathrm{T}}Y = B$ and then $L^{\mathrm{T}}P^{\mathrm{T}}X = Y$.

If TRANS = 'C', the solution is computed by solving $U^{\mathrm{H}}Y = B$ and then $L^{\mathrm{H}}P^{\mathrm{T}}X = Y$.

4 References

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

5 Parameters

1: TRANS – CHARACTER(1) *Input*

On entry: indicates the form of the equations.

TRANS = 'N'
$AX = B$ is solved for X.

TRANS = 'T'
$A^{\mathrm{T}}X = B$ is solved for X.

TRANS = 'C'
$A^{\mathrm{H}}X = B$ is solved for X.

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

2: N – INTEGER *Input*

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

Constraint: N ≥ 0.

3: NRHS – INTEGER *Input*

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

Constraint: NRHS ≥ 0.

4: 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: the LU factorization of A, as returned by F07ARF (ZGETRF).

5: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F07ASF (ZGETRS) is called.

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

6: IPIV($*$) – INTEGER array *Input*

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

On entry: the pivot indices, as returned by F07ARF (ZGETRF).

7: B(LDB,$*$) – COMPLEX (KIND=nag_wp) array *Input/Output*

Note: the second dimension of the array B must be at least max(1, 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 F07ASF (ZGETRS) is called.

Constraint: LDB $\geq$ max(1, 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.

7 Accuracy

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

$$|E| \leq c(n)\epsilon P|L||U|,$$

$c(n)$ 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

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

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

Note that $\mathrm{cond}(A,x)$ can be much smaller than $\mathrm{cond}(A)$, and $\mathrm{cond}\left(A^{\mathrm{H}}\right)$ (which is the same as $\mathrm{cond}\left(A^{\mathrm{T}}\right)$) can be much larger (or smaller) than $\mathrm{cond}(A)$.

Forward and backward error bounds can be computed by calling F07AVF (ZGERFS), and an estimate for $\kappa_\infty(A)$ can be obtained by calling F07AUF (ZGECON) with NORM = 'I'.

8 Further Comments

The total number of real floating point operations is approximately $8n^2r$.

This routine may be followed by a call to F07AVF (ZGERFS) to refine the solution and return an error estimate.

The real analogue of this routine is F07AEF (DGETRS).

9 Example

This example solves the system of equations $AX = B$, where

$$A = \begin{pmatrix} -1.34+2.55i & 0.28+3.17i & -6.39-2.20i & 0.72-0.92i \\ -0.17-1.41i & 3.31-0.15i & -0.15+1.34i & 1.29+1.38i \\ -3.29-2.39i & -1.91+4.42i & -0.14-1.35i & 1.72+1.35i \\ 2.41+0.39i & -0.56+1.47i & -0.83-0.69i & -1.96+0.67i \end{pmatrix}$$

and

$$B = \begin{pmatrix} 26.26+51.78i & 31.32-6.70i \\ 6.43-8.68i & 15.86-1.42i \\ -5.75+25.31i & -2.15+30.19i \\ 1.16+2.57i & -2.56+7.55i \end{pmatrix}.$$

Here A is nonsymmetric and must first be factorized by F07ARF (ZGETRF).

9.1 Program Text

```
    Program f07asfe

!     F07ASF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, x04dbf, zgetrf, zgetrs
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter                :: nin = 5, nout = 6
      Character (1), Parameter          :: trans = 'N'
!     .. Local Scalars ..
      Integer                           :: i, ifail, info, lda, ldb, n, nrhs
!     .. Local Arrays ..
      Complex (Kind=nag_wp), Allocatable :: a(:,:), b(:,:)
      Integer, Allocatable              :: ipiv(:)
      Character (1)                     :: clabs(1), rlabs(1)
!     .. Executable Statements ..
      Write (nout,*) 'F07ASF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n, nrhs
```

```
      lda = n
      ldb = n
      Allocate (a(lda,n),b(ldb,nrhs),ipiv(n))

!     Read A and B from data file

      Read (nin,*)(a(i,1:n),i=1,n)
      Read (nin,*)(b(i,1:nrhs),i=1,n)

!     Factorize A

!     The NAG name equivalent of zgetrf is f07arf
      Call zgetrf(n,n,a,lda,ipiv,info)

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

!       Compute solution

!       The NAG name equivalent of zgetrs is f07asf
        Call zgetrs(trans,n,nrhs,a,lda,ipiv,b,ldb,info)

!       Print solution

!       ifail: behaviour on error exit
!              =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
        ifail = 0
        Call x04dbf('General',' ',n,nrhs,b,ldb,'Bracketed','F7.4', &
          'Solution(s)','Integer',rlabs,'Integer',clabs,80,0,ifail)

      Else
        Write (nout,*) 'The factor U is singular'
      End If

    End Program f07asfe
```

9.2 Program Data

```
F07ASF Example Program Data
  4  2                                                    :Values of N and NRHS
 (-1.34, 2.55) ( 0.28, 3.17) (-6.39,-2.20) ( 0.72,-0.92)
 (-0.17,-1.41) ( 3.31,-0.15) (-0.15, 1.34) ( 1.29, 1.38)
 (-3.29,-2.39) (-1.91, 4.42) (-0.14,-1.35) ( 1.72, 1.35)
 ( 2.41, 0.39) (-0.56, 1.47) (-0.83,-0.69) (-1.96, 0.67)  :End of matrix A
 (26.26, 51.78) (31.32, -6.70)
 ( 6.43, -8.68) (15.86, -1.42)
 (-5.75, 25.31) (-2.15, 30.19)
 ( 1.16,  2.57) (-2.56,  7.55)                            :End of matrix B
```

9.3 Program Results

```
 F07ASF Example Program Results

 Solution(s)
                     1                   2
 1  ( 1.0000, 1.0000) (-1.0000,-2.0000)
 2  ( 2.0000,-3.0000) ( 5.0000, 1.0000)
 3  (-4.0000,-5.0000) (-3.0000, 4.0000)
 4  ( 0.0000, 6.0000) ( 2.0000,-3.0000)
```