

NAG Library Routine Document

F07JSF (ZPTTRS)

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

F07JSF (ZPTTRS) computes the solution to a complex system of linear equations $AX = B$, where A is an n by n Hermitian positive definite tridiagonal matrix and X and B are n by r matrices, using the LDL^H factorization returned by F07JRF (ZPTTRF).

2 Specification

SUBROUTINE F07JSF (UPLO, N, NRHS, D, E, B, LDB, INFO)

INTEGER N, NRHS, LDB, INFO
 REAL (KIND=nag_wp) D(*)
 COMPLEX (KIND=nag_wp) E(*), B(LDB,*)
 CHARACTER(1) UPLO

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

3 Description

F07JSF (ZPTTRS) should be preceded by a call to F07JRF (ZPTTRF), which computes a modified Cholesky factorization of the matrix A as

$$A = LDL^H,$$

where L is a unit lower bidiagonal matrix and D is a diagonal matrix, with positive diagonal elements. F07JSF (ZPTTRS) then utilizes the factorization to solve the required equations. Note that the factorization may also be regarded as having the form $U^H DU$, where U is a unit upper bidiagonal matrix.

4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D (1999) *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia <http://www.netlib.org/lapack/lug>

5 Parameters

1: UPLO – CHARACTER(1) *Input*

On entry: specifies the form of the factorization as follows:

UPLO = 'U'
 $A = U^H DU$.

UPLO = 'L'
 $A = LDL^H$.

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

2: N – INTEGER *Input*

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

Constraint: $N \geq 0$.

- 3: NRHS – INTEGER *Input*
On entry: r , the number of right-hand sides, i.e., the number of columns of the matrix B .
Constraint: $\text{NRHS} \geq 0$.
- 4: D(*) – REAL (KIND=nag_wp) array *Input*
Note: the dimension of the array D must be at least $\max(1, N)$.
On entry: must contain the n diagonal elements of the diagonal matrix D from the LDL^H or $U^H DU$ factorization of A .
- 5: E(*) – COMPLEX (KIND=nag_wp) array *Input*
Note: the dimension of the array E must be at least $\max(1, N - 1)$.
On entry: if $\text{UPLO} = 'U'$, E must contain the $(n - 1)$ superdiagonal elements of the unit upper bidiagonal matrix U from the $U^H DU$ factorization of A .
 If $\text{UPLO} = 'L'$, E must contain the $(n - 1)$ subdiagonal elements of the unit lower bidiagonal matrix L from the LDL^H factorization of A .
- 6: B(LDB, *) – COMPLEX (KIND=nag_wp) array *Input/Output*
Note: the second dimension of the array B must be at least $\max(1, \text{NRHS})$.
On entry: the n by r matrix of right-hand sides B .
On exit: the n by r solution matrix X .
- 7: LDB – INTEGER *Input*
On entry: the first dimension of the array B as declared in the (sub)program from which F07JSF (ZPTTRS) is called.
Constraint: $\text{LDB} \geq \max(1, N)$.
- 8: INFO – INTEGER *Output*
On exit: $\text{INFO} = 0$ unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

$\text{INFO} < 0$

If $\text{INFO} = -i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

The computed solution for a single right-hand side, \hat{x} , satisfies an equation of the form

$$(A + E)\hat{x} = b,$$

where

$$\|E\|_1 = O(\epsilon)\|A\|_1$$

and ϵ is the *machine precision*. An approximate error bound for the computed solution is given by

$$\frac{\|\hat{x} - x\|_1}{\|x\|_1} \leq \kappa(A) \frac{\|E\|_1}{\|A\|_1},$$

where $\kappa(A) = \|A^{-1}\|_1 \|A\|_1$, the condition number of A with respect to the solution of the linear equations. See Section 4.4 of Anderson *et al.* (1999) for further details.

Following the use of this routine F07JUF (ZPTCON) can be used to estimate the condition number of A and F07JVF (ZPTRFS) can be used to obtain approximate error bounds.

8 Parallelism and Performance

F07JSF (ZPTTRS) is not threaded by NAG in any implementation.

F07JSF (ZPTTRS) 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 total number of floating-point operations required to solve the equations $AX = B$ is proportional to nr .

The real analogue of this routine is F07JEF (DPTTRS).

10 Example

This example solves the equations

$$AX = B,$$

where A is the Hermitian positive definite tridiagonal matrix

$$A = \begin{pmatrix} 16.0 & 16.0 - 16.0i & 0 & 0 \\ 16.0 + 16.0i & 41.0 & 18.0 + 9.0i & 0.0 \\ 0 & 18.0 - 9.0i & 46.0 & 1.0 + 4.0i \\ 0 & 0 & 1.0 - 4.0i & 21.0 \end{pmatrix}$$

and

$$B = \begin{pmatrix} 64.0 + 16.0i & -16.0 - 32.0i \\ 93.0 + 62.0i & 61.0 - 66.0i \\ 78.0 - 80.0i & 71.0 - 74.0i \\ 14.0 - 27.0i & 35.0 + 15.0i \end{pmatrix}.$$

10.1 Program Text

```

Program f07jsfe

!       F07JSF Example Program Text

!       Mark 25 Release. NAG Copyright 2014.

!       .. Use Statements ..
Use nag_library, Only: nag_wp, x04dbf, zpttrf, zpttrs
!       .. Implicit None Statement ..
Implicit None
!       .. Parameters ..
Integer, Parameter          :: nin = 5, nout = 6
Character (1), Parameter   :: uplo = 'U'
!       .. Local Scalars ..
Integer                    :: i, ifail, info, ldb, n, nrhs
!       .. Local Arrays ..
Complex (Kind=nag_wp), Allocatable :: b(:,,:), e(:)
Real (Kind=nag_wp), Allocatable   :: d(:)
Character (1)                  :: clabs(1), rlabs(1)
!       .. Executable Statements ..
Write (nout,*) 'F07JSF Example Program Results'

```

```

      Write (nout,*)
      Flush (nout)
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n, nrhs
      ldb = n
      Allocate (b(ldb,nrhs),e(n-1),d(n))

!     Read the upper bidiagonal part of the tridiagonal matrix A from
!     data file

      Read (nin,*) e(1:n-1)
      Read (nin,*) d(1:n)

!     Read the right hand matrix B

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

!     Factorize the tridiagonal matrix A
!     The NAG name equivalent of zpttrf is f07jrf
      Call zpttrf(n,d,e,info)

      If (info==0) Then

!         Solve the equations AX = B
!         The NAG name equivalent of zpttrs is f07jsf
          Call zpttrs(uplo,n,nrhs,d,e,b,ldb,info)

!         Print the 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',' ','Solution(s)', &
            'Integer',rlabs,'Integer',clabs,80,0,ifail)

      Else
          Write (nout,99999) 'The leading minor of order ', info, &
            ' is not positive definite'
      End If

99999 Format (1X,A,I3,A)
      End Program f07jsfe

```

10.2 Program Data

```

F07JSF Example Program Data
  4          2          :Values of N and NRHS
          ( 16.0,-16.0) ( 18.0,  9.0) (  1.0,  4.0) :End of superdiagonal E
 16.0      41.0      46.0      21.0      :End of diagonal D
( 64.0, 16.0) (-16.0,-32.0)
( 93.0, 62.0) ( 61.0,-66.0)
( 78.0,-80.0) ( 71.0,-74.0)
( 14.0,-27.0) ( 35.0, 15.0)          :End of matrix B

```

10.3 Program Results

F07JSF Example Program Results

```

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

```
