NAG Library Routine Document

F07JNF (ZPTSV)

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

F07JNF (ZPTSV) 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.

2 Specification

```
SUBROUTINE F07JNF (N, NRHS, D, E, B, LDB, INFO)
INTEGER                N, NRHS, LDB, INFO
REAL (KIND=nag_wp)     D(*)
COMPLEX (KIND=nag_wp) E(*), B(LDB,*)
```

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

3 Description

F07JNF (ZPTSV) factors A as $A = LDL^{\mathrm{H}}$. The factored form of A is then used to solve the system of equations.

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

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

5 Parameters

1: N – INTEGER *Input*

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

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

2: NRHS – INTEGER *Input*

On entry: r, the number of right-hand sides, i.e., the number of columns of the matrix B.

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

3: D($*$) – REAL (KIND=nag_wp) array *Input/Output*

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

On entry: the n diagonal elements of the tridiagonal matrix A.

On exit: the n diagonal elements of the diagonal matrix D from the factorization $A = LDL^{\mathrm{H}}$.

4: E($*$) – COMPLEX (KIND=nag_wp) array *Input/Output*

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

On entry: the $(n - 1)$ subdiagonal elements of the tridiagonal matrix A.

On exit: the $(n - 1)$ subdiagonal elements of the unit bidiagonal factor L from the LDL^{H} factorization of A. (E can also be regarded as the superdiagonal of the unit bidiagonal factor U from the $U^{H}DU$ factorization of A.)

5: 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: if INFO $= 0$, the n by r solution matrix X.

6: LDB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F07JNF (ZPTSV) is called.

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

7: 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.

INFO > 0

The leading minor of order $\langle value \rangle$ is not positive definite, and the solution has not been computed. The factorization has not been completed unless N $= \langle value \rangle$.

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.

F07JPF (ZPTSVX) is a comprehensive LAPACK driver that returns forward and backward error bounds and an estimate of the condition number. Alternatively, F04CGF solves $Ax = b$ and returns a forward error bound and condition estimate. F04CGF calls F07JNF (ZPTSV) to solve the equations.

8 Parallelism and Performance

F07JNF (ZPTSV) is not threaded by NAG in any implementation.

F07JNF (ZPTSV) 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 number of floating-point operations required for the factorization of A is proportional to n, and the number of floating-point operations required for the solution of the equations is proportional to nr, where r is the number of right-hand sides.

The real analogue of this routine is F07JAF (DPTSV).

10 Example

This example solves the equations

$$Ax = b,$$

where A is the Hermitian positive definite tridiagonal matrix

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

and

$$b = \begin{pmatrix} 64.0 + 16.0i \\ 93.0 + 62.0i \\ 78.0 - 80.0i \\ 14.0 - 27.0i \end{pmatrix}.$$

Details of the LDL^{H} factorization of A are also output.

10.1 Program Text

```
    Program f07jnfe

!     F07JNF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, zptsv
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter              :: nin = 5, nout = 6
!     .. Local Scalars ..
      Integer                         :: info, n
!     .. Local Arrays ..
      Complex (Kind=nag_wp), Allocatable :: b(:), e(:)
      Real (Kind=nag_wp), Allocatable  :: d(:)
!     .. Executable Statements ..
      Write (nout,*) 'F07JNF Example Program Results'
      Write (nout,*)
!     Skip heading in data file
      Read (nin,*)
```

```
      Read (nin,*) n

      Allocate (b(n),e(n-1),d(n))

!     Read the lower bidiagonal part of the tridiagonal matrix A and
!     the right hand side b from data file

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

!     Solve the equations Ax = b for x
!     The NAG name equivalent of zptsv is f07jnf
      Call zptsv(n,1,d,e,b,n,info)

      If (info==0) Then

!       Print solution

        Write (nout,*) 'Solution'
        Write (nout,99999) b(1:n)

!       Print details of factorization

        Write (nout,*)
        Write (nout,*) 'Diagonal elements of the diagonal matrix D'
        Write (nout,99998) d(1:n)
        Write (nout,*)
        Write (nout,*) 'Sub-diagonal elements of the Cholesky factor L'
        Write (nout,99999) e(1:n-1)

      Else
        Write (nout,99997) 'The leading minor of order ', info, &
          ' is not positive definite'
      End If

99999 Format (4(' (',F8.4,',',F8.4,')':))
99998 Format ((2X,F7.4,3(11X,F7.4)))
99997 Format (1X,A,I3,A)
    End Program f07jnfe
```

10.2 Program Data

```
F07JNF Example Program Data
   4                                                    :Value of N
  16.0          41.0          46.0          21.0        :End of diagonal D
 ( 16.0, 16.0) ( 18.0, -9.0) (  1.0, -4.0)              :End of sub-diagonal E
 ( 64.0, 16.0) ( 93.0, 62.0) ( 78.0,-80.0) ( 14.0,-27.0) :End of vector b
```

10.3 Program Results

```
 F07JNF Example Program Results

 Solution
 (  2.0000,  1.0000) (  1.0000,  1.0000) (  1.0000, -2.0000) (  1.0000, -1.0000)

 Diagonal elements of the diagonal matrix D
  16.0000            9.0000            1.0000            4.0000

 Sub-diagonal elements of the Cholesky factor L
 (  1.0000,  1.0000) (  2.0000, -1.0000) (  1.0000, -4.0000)
```