NAG Fortran 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
double precision      D(*)
complex*16            E(*), B(LDB,*)
```

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

3 Description

A is factored as $A = LDL^H$, and 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 URL: 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($*$) – ***double precision*** 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 $A = LDL^H$.

4: E($*$) – ***complex*16*** array *Input/Output*

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

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

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

5: B(LDB,$*$) – ***complex*16*** array *Input/Output*

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

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

Errors or warnings detected by the routine:

INFO < 0

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

INFO > 0

If INFO $= i$, the leading minor of order i is not positive-definite, and the solution has not been computed. The factorization has not been completed unless $i = N$.

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 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).

9 Example

To solve 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.

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.

```
*     F07JNF Example Program Text
*     Mark 21 Release. NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX
      PARAMETER        (NMAX=8)
*     .. Local Scalars ..
      INTEGER          I, INFO, N
*     .. Local Arrays ..
      COMPLEX *16      B(NMAX), E(NMAX-1)
      DOUBLE PRECISION D(NMAX)
*     .. External Subroutines ..
      EXTERNAL         ZPTSV
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F07JNF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
*
*        Read the lower bidiagonal part of the tridiagonal matrix A and
*        the right hand side b from data file
*
         READ (NIN,*) (D(I),I=1,N)
         READ (NIN,*) (E(I),I=1,N-1)
         READ (NIN,*) (B(I),I=1,N)
*
*        Solve the equations Ax = b for x
*
         CALL ZPTSV(N,1,D,E,B,N,INFO)
```

```
*
         IF (INFO.EQ.0) THEN
*
*           Print solution
*
            WRITE (NOUT,*) 'Solution'
            WRITE (NOUT,99999) (B(I),I=1,N)
*
*           Print details of factorization
*
            WRITE (NOUT,*)
            WRITE (NOUT,*) 'Diagonal elements of the diagonal matrix D'
            WRITE (NOUT,99998) (D(I),I=1,N)
            WRITE (NOUT,*)
            WRITE (NOUT,*)
     +        'Sub-diagonal elements of the Cholesky factor L'
            WRITE (NOUT,99999) (E(I),I=1,N-1)
*
         ELSE
            WRITE (NOUT,99997) 'The leading minor of order ', INFO,
     +        ' is not positive definite'
         END IF
      ELSE
         WRITE (NOUT,*) 'NMAX too small'
      END IF
      STOP
*
99999 FORMAT (4(' (',F8.4,',',F8.4,')',:))
99998 FORMAT ((2X,F7.4,3(11X,F7.4)))
99997 FORMAT (1X,A,I3,A)
      END
```

9.2 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
```

9.3 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)
```