

NAG Library Routine Document

F07QNF (ZSPSV)

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

F07QNF (ZSPSV) computes the solution to a complex system of linear equations

$$AX = B,$$

where A is an n by n symmetric matrix stored in packed format and X and B are n by r matrices.

2 Specification

SUBROUTINE F07QNF (UPLO, N, NRHS, AP, IPIV, B, LDB, INFO)

INTEGER N, NRHS, IPIV(N), LDB, INFO
 COMPLEX (KIND=nag_wp) AP(*), B(LDB,*)
 CHARACTER(1) UPLO

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

3 Description

F07QNF (ZSPSV) uses the diagonal pivoting method to factor A as $A = UDU^T$ if UPLO = 'U' or $A = LDL^T$ if UPLO = 'L', where U (or L) is a product of permutation and unit upper (lower) triangular matrices, D is symmetric and block diagonal with 1 by 1 and 2 by 2 diagonal blocks. The factored form of A is then used to solve the system of equations $AX = B$.

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

Higham N J (2002) *Accuracy and Stability of Numerical Algorithms* (2nd Edition) SIAM, Philadelphia

5 Parameters

- 1: UPLO – CHARACTER(1) *Input*
On entry: if UPLO = 'U', the upper triangle of A is stored.
 If UPLO = 'L', the lower triangle of A is stored.
Constraint: UPLO = 'U' or 'L'.
- 2: N – INTEGER *Input*
On entry: n , the number of linear equations, i.e., 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: NRHS ≥ 0 .
- 4: AP(*) – COMPLEX (KIND=nag_wp) array *Input/Output*
Note: the dimension of the array AP must be at least $\max(1, N \times (N + 1)/2)$.
On entry: the n by n symmetric 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 $i \geq j$.
On exit: the block diagonal matrix D and the multipliers used to obtain the factor U or L from the
 factorization $A = UDU^T$ or $A = LDL^T$ as computed by F07QRF (ZSPTRF), stored as a packed
 triangular matrix in the same storage format as A .
- 5: IPIV(N) – INTEGER array *Output*
On exit: details of the interchanges and the block structure of D . More precisely,
 if IPIV(i) = $k > 0$, d_{ii} is a 1 by 1 pivot block and the i th row and column of A were
 interchanged with the k th row and column;
 if UPLO = 'U' and IPIV($i - 1$) = IPIV(i) = $-l < 0$, $\begin{pmatrix} d_{i-1,i-1} & \bar{d}_{i,i-1} \\ \bar{d}_{i,i-1} & d_{ii} \end{pmatrix}$ is a 2 by 2 pivot
 block and the ($i - 1$)th row and column of A were interchanged with the l th row and
 column;
 if UPLO = 'L' and IPIV(i) = IPIV($i + 1$) = $-m < 0$, $\begin{pmatrix} d_{ii} & d_{i+1,i} \\ d_{i+1,i} & d_{i+1,i+1} \end{pmatrix}$ is a 2 by 2 pivot
 block and the ($i + 1$)th row and column of A were interchanged with the m th row and
 column.
- 6: B(LDB,*) – COMPLEX (KIND=nag_wp) array *Input/Output*
Note: the second dimension of the array B must be at least $\max(1, \text{NRHS})$.
Note: To solve the equations $Ax = b$, where b is a single right-hand side, B may be supplied as a
 one-dimensional array with length LDB = $\max(1, N)$.
On entry: the n by r right-hand side matrix B .
On exit: if INFO = 0, the n by r solution matrix X .
- 7: LDB – INTEGER *Input*
On entry: the first dimension of the array B as declared in the (sub)program from which F07QNF
 (ZSPSV) is called.
Constraint: LDB $\geq \max(1, N)$.
- 8: 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 i th argument had an illegal value. An explanatory message is output, and execution of the program is terminated.

INFO > 0

If INFO = i , d_{ii} is exactly zero. The factorization has been completed, but the block diagonal matrix D is exactly singular, so the solution could not be computed.

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) and Chapter 11 of Higham (2002) for further details.

F07QPF (ZSPSVX) is a comprehensive LAPACK driver that returns forward and backward error bounds and an estimate of the condition number. Alternatively, F04DJF solves $AX = B$ and returns a forward error bound and condition estimate. F04DJF calls F07QNF (ZSPSV) to solve the equations.

8 Further Comments

The total number of floating point operations is approximately $\frac{4}{3}n^3 + 8n^2r$, where r is the number of right-hand sides.

The real analogue of this routine is F07PAF (DSPSV). The complex Hermitian analogue of this routine is F07PNF (ZHPSV).

9 Example

This example solves the equations

$$Ax = b,$$

where A is the complex symmetric matrix

$$A = \begin{pmatrix} -0.56 + 0.12i & -1.54 - 2.86i & 5.32 - 1.59i & 3.80 + 0.92i \\ -1.54 - 2.86i & -2.83 - 0.03i & -3.52 + 0.58i & -7.86 - 2.96i \\ 5.32 - 1.59i & -3.52 + 0.58i & 8.86 + 1.81i & 5.14 - 0.64i \\ 3.80 + 0.92i & -7.86 - 2.96i & 5.14 - 0.64i & -0.39 - 0.71i \end{pmatrix}$$

and

$$b = \begin{pmatrix} -6.43 + 19.24i \\ -0.49 - 1.47i \\ -48.18 + 66.00i \\ -55.64 + 41.22i \end{pmatrix}.$$

Details of the factorization of A are also output.

9.1 Program Text

```

Program f07qnfe

!      F07QNF Example Program Text

!      Mark 24 Release. NAG Copyright 2012.

!      .. Use Statements ..
Use nag_library, Only: nag_wp, x04ddf, zspsv
!      .. Implicit None Statement ..
Implicit None
!      .. Parameters ..
Integer, Parameter          :: nin = 5, nout = 6
Character (1), Parameter   :: uplo = 'U'
!      .. Local Scalars ..
Integer                    :: i, ifail, info, j, n
!      .. Local Arrays ..
Complex (Kind=nag_wp), Allocatable :: ap(:), b(:)
Integer, Allocatable       :: ipiv(:)
Character (1)              :: clabs(1), rlabs(1)
!      .. Executable Statements ..
Write (nout,*) 'F07QNF Example Program Results'
Write (nout,*)
!      Skip heading in data file
Read (nin,*)
Read (nin,*) n

Allocate (ap((n*(n+1))/2),b(n),ipiv(n))

!      Read the upper or lower triangular part of the matrix A from
!      data file

If (uplo=='U') Then
  Read (nin,*)((ap(i+(j*(j-1))/2),j=i,n),i=1,n)
Else If (uplo=='L') Then
  Read (nin,*)((ap(i+((2*n-j)*(j-1))/2),j=1,i),i=1,n)
End If

!      Read b from data file

Read (nin,*) b(1:n)

!      Solve the equations Ax = b for x
!      The NAG name equivalent of zspsv is f07qnf
Call zspsv(uplo,n,1,ap,ipiv,b,n,info)

If (info==0) Then

!      Print solution

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

!      Print details of factorization

Write (nout,*)
Flush (nout)

!      ifail: behaviour on error exit
!              =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
ifail = 0
Call x04ddf(uplo,'Non-unit diagonal',n,ap,'Bracketed','F7.4', &
'Details of factorization','Integer',rlabs,'Integer',clabs,80,0, &
ifail)

!      Print pivot indices

```

```

      Write (nout,*)
      Write (nout,*) 'Pivot indices'
      Write (nout,99998) ipiv(1:n)

      Else
        Write (nout,99997) 'The diagonal block ', info, ' of D is zero'
      End If

99999 Format ((3X,4(' (',F7.4,',',F7.4,')':)))
99998 Format (1X,7I11)
99997 Format (1X,A,I3,A)
      End Program f07qnfe

```

9.2 Program Data

F07QNF Example Program Data

```

      4                                     :Value of N
( -0.56,  0.12) ( -1.54, -2.86) (  5.32, -1.59) (  3.80,  0.92)
                ( -2.83, -0.03) ( -3.52,  0.58) ( -7.86, -2.96)
                                (  8.86,  1.81) (  5.14, -0.64)
                                                ( -0.39, -0.71) :End matrix A
( -6.43, 19.24) ( -0.49, -1.47) (-48.18, 66.00) (-55.64, 41.22) :End vector b

```

9.3 Program Results

F07QNF Example Program Results

Solution

```
(-4.0000, 3.0000) ( 3.0000,-2.0000) (-2.0000, 5.0000) ( 1.0000,-1.0000)
```

Details of factorization

```

      1          2          3          4
1 (-2.0954,-2.2011) (-0.1071,-0.3157) (-0.4823, 0.0150) ( 0.4426, 0.1936)
2 ( 4.4079, 5.3991) (-0.6078, 0.2811) ( 0.5279,-0.3715)
3 (-2.8300,-0.0300) (-7.8600,-2.9600)
4 (-0.3900,-0.7100)

```

Pivot indices

```

      1          2          -2          -2

```
