NAG Fortran Library Routine Document

F04BHF

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

F04BHF computes the solution to a real system of linear equations $AX = B$, where A is an n by n symmetric matrix and X and B are n by r matrices. An estimate of the condition number of A and an error bound for the computed solution are also returned.

2 Specification

```
      SUBROUTINE F04BHF (UPLO, N, NRHS, A, LDA, IPIV, B, LDB, RCOND, ERRBND,
     1                   IFAIL)
      INTEGER            N, NRHS, LDA, IPIV(*), LDB, IFAIL
      double precision   A(LDA,*), B(LDB,*), RCOND, ERRBND
      CHARACTER*1        UPLO
```

3 Description

The diagonal pivoting method is used 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, and 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 URL: http://www.netlib.org/lapack/lug

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 the matrix A is stored, if UPLO = 'L', the lower triangle of the matrix A is stored.

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

2: N – INTEGER *Input*

On entry: the number of linear equations n, i.e., the order of the matrix A.

Constraint: N $\geq$ 0.

3: NRHS – INTEGER *Input*

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

Constraint: NRHS $\geq$ 0.

4: A(LDA,$*$) – ***double precision*** array *Input/Output*

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

On entry: the n by n symmetric matrix A.

If UPLO = 'U', the leading N by N upper triangular part of the array A contains the upper triangular part of the matrix A, and the strictly lower triangular part of A is not referenced; if UPLO = 'L', the leading N by N lower triangular part of the array A contains the lower triangular part of the matrix A, and the strictly upper triangular part of A is not referenced.

On exit: if IFAIL ≥ 0, 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 F07MDF (DSYTRF).

5: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F04BHF is called.

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

6: IPIV($*$) – INTEGER array *Output*

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

On exit: if IFAIL ≥ 0, details of the interchanges and the block structure of D, as determined by F07MDF (DSYTRF).

If IPIV$(k) > 0$, then rows and columns k and IPIV(k) were interchanged, and d_{kk} is a 1 by 1 diagonal block;
if UPLO = 'U' and IPIV$(k) =$ IPIV$(k-1) < 0$, then rows and columns $k-1$ and $-$IPIV(k) were interchanged and $d_{k-1:k,k-1:k}$ is a 2 by 2 diagonal block;
if UPLO = 'L' and IPIV$(k) =$ IPIV$(k+1) < 0$, then rows and columns $k+1$ and $-$IPIV(k) were interchanged and $d_{k:k+1,k:k+1}$ is a 2 by 2 diagonal block.

7: B(LDB,$*$) – ***double precision*** array *Input/Output*

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

On entry: the n by r matrix of right-hand sides B.

On exit: if IFAIL $= 0$ or N $+ 1$, 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 F04BHF is called.

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

9: RCOND – ***double precision*** *Output*

On exit: if IFAIL ≥ 0, an estimate of the reciprocal of the condition number of the matrix A, computed as RCOND $= 1/\left(\|A\|_1 \|A^{-1}\|_1\right)$.

10: ERRBND – ***double precision*** *Output*

On exit: if IFAIL $= 0$ or N $+ 1$, an estimate of the forward error bound for a computed solution $\hat{x}$, such that $\|\hat{x} - x\|_1 / \|x\|_1 \leq$ ERRBND, where $\hat{x}$ is a column of the computed solution returned in the array B and x is the corresponding column of the exact solution X. If RCOND is less than ***machine precision***, then ERRBND is returned as unity.

11: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

On exit: IFAIL $= 0$ unless the routine detects an error (see Section 6).

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, for users not familiar with this parameter the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

6 Error Indicators and Warnings

If on entry IFAIL $= 0$ or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL < 0 and IFAIL $\neq -999$

If IFAIL $= -i$, the ith argument had an illegal value.

IFAIL $= -999$

Allocation of memory failed. The INTEGER allocatable memory required is N, and the ***double precision*** allocatable memory required is $\max(2 \times \mathrm{N}, \mathrm{LWORK})$, where LWORK is the optimum workspace required by F07MAF (DSYSV). If this failure occurs it may be possible to solve the equations by calling the packed storage version of F04BHF, F04BJF, or by calling F07MAF (DSYSV) directly with less than the optimum workspace (see Chapter F07).

IFAIL > 0 and IFAIL $\leq$ N

If IFAIL $= i$, 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.

IFAIL $=$ N $+ 1$

RCOND is less than ***machine precision***, so that the matrix A is numerically singular. A solution to the equations $AX = B$ has nevertheless been 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 $\kappa(A) = \left\|A^{-1}\right\|_1 \|A\|_1$, the condition number of A with respect to the solution of the linear equations. F04BHF uses the approximation $\|E\|_1 = \epsilon\|A\|_1$ to estimate ERRBND. See Section 4.4 of Anderson *et al.* (1999) for further details.

8 Further Comments

The total number of floating-point operations required to solve the equations $AX = B$ is proportional to $\left(\frac{1}{3}n^3 + 2n^2r\right)$. The condition number estimation typically requires between four and five solves and never more than eleven solves, following the factorization.

In practice the condition number estimator is very reliable, but it can underestimate the true condition number; see Section 15.3 of Higham (2002) for further details.

The complex analogues of F04BHF are F04CHF for complex Hermitian matrices, and F04DHF for complex symmetric matrices.

9 Example

To solve the equations

$$AX = B,$$

where A is the symmetric indefinite matrix

$$A = \begin{pmatrix} -1.81 & 2.06 & 0.63 & -1.15 \\ 2.06 & 1.15 & 1.87 & 4.20 \\ 0.63 & 1.87 & -0.21 & 3.87 \\ -1.15 & 4.20 & 3.87 & 2.07 \end{pmatrix}$$

and

$$B = \begin{pmatrix} 0.96 & 3.93 \\ 6.07 & 19.25 \\ 8.38 & 9.90 \\ 9.50 & 27.85 \end{pmatrix}.$$

An estimate of the condition number of A and an approximate error bound for the computed solutions are also printed.

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.

```
*     F04BHF Example Program Text
*     Mark 21 Release. NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, NRHSMX
      PARAMETER        (NMAX=8,NRHSMX=2)
      INTEGER          LDA, LDB
      PARAMETER        (LDA=NMAX,LDB=NMAX)
*     .. Local Scalars ..
      DOUBLE PRECISION ERRBND, RCOND
      INTEGER          I, IERR, IFAIL, J, N, NRHS
*     .. Local Arrays ..
      DOUBLE PRECISION A(LDA,NMAX), B(LDB,NRHSMX)
      INTEGER          IPIV(NMAX)
*     .. External Subroutines ..
      EXTERNAL         F04BHF, X04CAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F04BHF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N, NRHS
      IF (N.LE.NMAX .AND. NRHS.LE.NRHSMX) THEN
*
*        Read the upper triangular part of A from data file
*
```

```
         READ (NIN,*) ((A(I,J),J=I,N),I=1,N)
*
*        Read B from data file
*
         READ (NIN,*) ((B(I,J),J=1,NRHS),I=1,N)
*
*        Solve the equations AX = B for X
*
         IFAIL = -1
         CALL F04BHF('Upper',N,NRHS,A,LDA,IPIV,B,LDB,RCOND,ERRBND,IFAIL)
*
         IF (IFAIL.EQ.0) THEN
*
*           Print solution, estimate of condition number and approximate
*           error bound
*
            IERR = 0
            CALL X04CAF('General',' ',N,NRHS,B,LDB,'Solution',IERR)
*
            WRITE (NOUT,*)
            WRITE (NOUT,*) 'Estimate of condition number'
            WRITE (NOUT,99999) 1.0D0/RCOND
            WRITE (NOUT,*)
            WRITE (NOUT,*)
     +        'Estimate of error bound for computed solutions'
            WRITE (NOUT,99999) ERRBND
         ELSE IF (IFAIL.EQ.N+1) THEN
*
*           Matrix A is numerically singular.  Print estimate of
*           reciprocal of condition number and solution
*
            WRITE (NOUT,*)
            WRITE (NOUT,*) 'Estimate of reciprocal of condition number'
            WRITE (NOUT,99999) RCOND
*
            WRITE (NOUT,*)
            IERR = 0
            CALL X04CAF('General',' ',N,NRHS,B,LDB,'Solution',IERR)
*
         ELSE IF (IFAIL.GT.0 .AND. IFAIL.LE.N) THEN
*
*           The upper triangular matrix U is exactly singular.  Print
*           details of factorization
*
            WRITE (NOUT,*)
            IERR = 0
            CALL X04CAF('Upper','Non-unit diagonal',N,N,A,LDA,
     +                  'Details of factorization',IERR)
*
*           Print pivot indices
*
            WRITE (NOUT,*)
            WRITE (NOUT,*) 'Pivot indices'
            WRITE (NOUT,99998) (IPIV(I),I=1,N)
         END IF
      ELSE
         WRITE (NOUT,*) 'NMAX and/or NRHSMX too small'
      END IF
      STOP
*
99999 FORMAT (6X,1P,E9.1)
99998 FORMAT ((1X,7I11))
      END
```

9.2 Program Data

```
F04BHF Example Program Data

  4      2                 :Values of N and NRHS

 -1.81   2.06   0.63  -1.15
         1.15   1.87   4.20
               -0.21   3.87
                       2.07 :End of matrix A

  0.96   3.93
  6.07  19.25
  8.38   9.90
  9.50  27.85               :End of matrix B
```

9.3 Program Results

```
 F04BHF Example Program Results

 Solution
             1          2
 1     -5.0000     2.0000
 2     -2.0000     3.0000
 3      1.0000     4.0000
 4      4.0000     1.0000

 Estimate of condition number
        7.6E+01

 Estimate of error bound for computed solutions
        8.4E-15
```