NAG Fortran Library Routine Document

F04ACF

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

F04ACF calculates the approximate solution of a set of real symmetric positive-definite band equations with multiple right-hand sides, using a Cholesky factorization.

2 Specification

```
SUBROUTINE F04ACF(A, IA, B, IB, N, M, IR, C, IC, RL, IRL, M1, IFAIL)

INTEGER

IA, IB, N, M, IR, IC, IRL, M1, IFAIL

real

A(IA,M1), B(IB,IR), C(IC,IR), RL(IRL,M1)
```

3 Description

Given a set of real linear equations AX = B, where A is a symmetric positive-definite band matrix, the routine computes a Cholesky factorization of A as $A = LL^T$, where L is a lower triangular band matrix. The columns x of the solution X are found by forward and backward substitution in Ly = b and $L^Tx = y$, where b is a column of the right-hand side matrix B.

4 References

Wilkinson J H and Reinsch C (1971) Handbook for Automatic Computation II, Linear Algebra Springer-Verlag

5 Parameters

1: A(IA,M1) - real array

Input

On entry: the lower triangle of the n by n positive-definite symmetric band matrix A, with the diagonal of the matrix stored in the (m+1)th column of the array, and the m sub-diagonals within the band stored in the first m columns of the array. Each row of the matrix is stored in the corresponding row of the array. For example, if n=5 and m=2, the storage scheme is:

$$\begin{pmatrix} * & * & a_{11} \\ * & a_{21} & a_{22} \\ a_{31} & a_{32} & a_{33} \\ a_{42} & a_{43} & a_{44} \\ a_{53} & a_{54} & a_{55} \end{pmatrix}.$$

The elements in the top left corner of the array are not used. The following code may be used to assign elements within the band of the lower triangle of the matrix to the correct elements of the array:

```
DO 20 I = 1, N
        DO 10 J = MAX(1,I-M), I
             A(I,J-I+M+1) = matrix(I,J)
10 CONTINUE
20 CONTINUE
```

2: IA – INTEGER

Input

On entry: the first dimension of the array A as declared in the (sub)program from which F04ACF is called.

[NP3546/20A] F04ACF.1

Constraint: $IA \geq N$.

3: B(IB,IR) - real array

Input

On entry: the n by r right-hand side matrix B. See also Section 8.

4: IB – INTEGER Input

On entry: the first dimension of the array B as declared in the (sub)program from which F04ACF is called.

Constraint: $IB \geq N$.

5: N – INTEGER Input

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

6: M - INTEGER Input

On entry: m, the number of sub-diagonals within the band of A.

7: IR – INTEGER Input

On entry: r, the number of right-hand sides.

8: C(IC,IR) – *real* array

Output

On exit: the n by r solution matrix X. See also Section 8.

9: IC – INTEGER Input

On entry: the first dimension of the array C as declared in the (sub)program from which F04ACF is called.

Constraint: $IC \geq N$.

10: RL(IRL,M1) - real array

Output

On exit: the lower triangular band matrix L stored in the same form as A, except that the reciprocals of the diagonal elements are stored instead of the elements themselves.

11: IRL – INTEGER Input

On entry: the first dimension of the array RL as declared in the (sub)program from which F04ACF is called.

Constraint: $IRL \geq N$.

12: M1 – INTEGER Input

On entry: the value m+1.

13: 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.

F04ACF.2 [NP3546/20A]

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 = 1

A is not positive-definite, possibly due to rounding errors.

7 Accuracy

The accuracy of the computed solutions depend on the conditioning of the original matrix. For a detailed error analysis see page 54 of Wilkinson and Reinsch (1971).

8 Further Comments

The time taken by the routine is approximately proportional to $n(m+1)^2$.

This routine should only be used when $m \ll n$ since as m approaches n, it becomes less efficient to take advantage of the band form.

Unless otherwise stated in the Users' Note for your implementation, the routine may be called with the same actual array supplied for parameters B and C, in which case the solution vectors will overwrite the right-hand sides. However, this is not standard Fortran 77, and may not work on all systems.

9 Example

To solve the set of linear equations AX = B where

$$A = \begin{pmatrix} 5 & -4 & 1 & & & & \\ -4 & 6 & -4 & 1 & & & & \\ 1 & -4 & 6 & -4 & 1 & & & \\ & 1 & -4 & 6 & -4 & 1 & & \\ & & 1 & -4 & 6 & -4 & 1 \\ & & & 1 & -4 & 6 & -4 \\ & & & 1 & -4 & 5 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

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.

```
FO4ACF Example Program Text
Mark 14 Revised. NAG Copyright 1989.
.. Parameters ..
INTEGER
               NMAX, M1MAX, IA, IB, IC, IRL
                 (NMAX=10,M1MAX=5,IA=NMAX,IB=NMAX,IC=NMAX,
PARAMETER
                 IRL=NMAX)
INTEGER
                NIN, NOUT
PARAMETER
                (NIN=5,NOUT=6)
.. Local Scalars ..
                 I, IFAIL, IR, J, M, M1, N
INTEGER
.. Local Arrays ..
real
                A(IA,M1MAX), B(IB,1), C(IC,1), RL(IRL,M1MAX)
.. External Subroutines ..
EXTERNAL
                F04ACF
.. Executable Statements ..
WRITE (NOUT,*) 'F04ACF Example Program Results'
Skip heading in data Ûle
READ (NIN, *)
READ (NIN,*) N, M1
WRITE (NOUT, *)
```

[NP3546/20A] F04ACF.3

```
IF (N.GT.O .AND. N.LE.NMAX .AND. M1.GT.O .AND. M1.LE.M1MAX) THEN
         READ (NIN,*) ((A(I,J),J=1,M1),B(I,1),I=1,N)
         M = M1 - 1
         IFAIL = 1
         CALL FO4ACF(A, IA, B, IB, N, M, IR, C, IC, RL, IRL, M1, IFAIL)
         IF (IFAIL.NE.O) THEN
           WRITE (NOUT, 99999) 'Error in FO4ACF. IFAIL =', IFAIL
         ELSE
           WRITE (NOUT,*) ' Solution'
            WRITE (NOUT, 99998) (C(I,1), I=1,N)
         END IF
     ELSE
         WRITE (NOUT, 99999) 'N or M1 is out of range: N = ', N,
       ' M1 = ', M1
     END IF
     STOP
99999 FORMAT (1X,A,I5,A,I5)
99998 FORMAT (1X,F9.4)
     END
```

9.2 Program Data

```
FO4ACF Example Program Data
 7 3
    0
        0
              5
    0
        -4
              6
                   Ω
        -4
    1
              6
                   0
        -4
    1
              6
                   1
        -4
    1
              6
                   0
        -4
    1
              6
                   0
        -4
                   0
    1
              5
```

9.3 Program Results

```
F04ACF Example Program Results

Solution
    4.0000
    7.5000
    10.0000
    11.0000
    10.0000
    7.5000
    4.0000
```

F04ACF.4 (last) [NP3546/20A]