NAG Fortran Library Routine Document

F04MCF

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

F04MCF computes the approximate solution of a system of real linear equations with multiple right-hand sides, $AX = B$, where A is a symmetric positive-definite variable-bandwidth matrix, which has previously been factorized by F01MCF. Related systems may also be solved.

2 Specification

```
 SUBROUTINE F04MCF(N, AL, LAL, D, NROW, IR, B, NRB, ISELCT, X, NRX,
1                  IFAIL)
 INTEGER           N, LAL, NROW(N), IR, NRB, ISELCT, NRX, IFAIL
 real              AL(LAL), D(N), B(NRB,IR), X(NRX,IR)
```

3 Description

The normal use of this routine is the solution of the systems $AX = B$, following a call of F01MCF to determine the Cholesky factorization $A = LDL^T$ of the symmetric positive-definite variable-bandwidth matrix A.

However, the routine may be used to solve any one of the following systems of linear algebraic equations:

1. $LDL^T X = B$ (usual system),
2. $LDX = B$ (lower triangular system),
3. $DL^T X = B$ (upper triangular system),
4. $LL^T X = B$
5. $LX = B$ (unit lower triangular system),
6. $L^T X = B$ (unit upper triangular system).

L denotes a unit lower triangular variable-bandwidth matrix of order n, D a diagonal matrix of order n, and B a set of right-hand sides.

The matrix L is represented by the elements lying within its **envelope**, i.e., between the first non-zero of each row and the diagonal (see Section 9 for an example). The width NROW(i) of the ith row is the number of elements between the first non-zero element and the element on the diagonal inclusive.

4 References

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

5 Parameters

1: N – INTEGER *Input*

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

Constraint: N $\geq$ 1.

2: AL(LAL) – ***real*** array *Input*

On entry: the elements within the envelope of the lower triangular matrix L, taken in row by row order, as returned by F01MCF. The unit diagonal elements of L must be stored explicitly.

3: LAL – INTEGER *Input*

On entry: the dimension of the array AL as declared in the (sub)program from which F04MCF is called.

Constraint: LAL $\geq$ NROW(1) + NROW$(2) + \ldots +$ NROW(n).

4: D(N) – ***real*** array *Input*

On entry: the diagonal elements of the diagonal matrix D. D is not referenced if ISELCT $\geq$ 4.

5: NROW(N) – INTEGER array *Input*

On entry: NROW(i) must contain the width of row i of L, i.e., the number of elements between the first (leftmost) non-zero element and the element on the diagonal, inclusive.

Constraint: $1 \leq$ NROW$(i) \leq i$.

6: IR – INTEGER *Input*

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

Constraint: IR $\geq$ 1.

7: B(NRB,IR) – ***real*** array *Input*

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

8: NRB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F04MCF is called.

Constraint: NRB $\geq$ N.

9: ISELCT – INTEGER *Input*

On entry: ISELCT must specify the type of system to be solved, as follows:

ISELCT = 1:

solve $LDL^TX = B$,

ISELCT = 2:

solve $LDX = B$,

ISELCT = 3:

solve $DL^TX = B$,

ISELCT = 4:

solve $LL^TX = B$,

ISELCT = 5:

solve $LX = B$,

ISELCT = 6:

solve $L^TX = B$.

10: X(NRX,IR) – ***real*** array *Output*

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

11: NRX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which F04MCF is called.

Constraint: NRX $\geq$ N.

12: 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 = 1

On entry, N < 1,
or for some i, NROW(i) < 1 or NROW(i) > i,
or LAL < NROW(1) + NROW(2) + ... + NROW(N).

IFAIL = 2

On entry, IR < 1,
or NRB < N,
or NRX < N.

IFAIL = 3

On entry, ISELCT < 1,
or ISELCT > 6.

IFAIL = 4

The diagonal matrix D is singular, i.e., at least one of the elements of D is zero. This can only occur if ISELCT $\leq$ 3.

IFAIL = 5

At least one of the diagonal elements of L is not equal to unity.

7 Accuracy

The usual backward error analysis of the solution of triangular system applies: each computed solution vector is exact for slightly perturbed matrices L and D, as appropriate (see pages 25–27 and 54–55 of Wilkinson and Reinsch (1971).

8 Further Comments

The time taken by the routine is approximately proportional to pr, where $p = \text{NROW}(1) + \text{NROW}(2) + \ldots + \text{NROW}(n)$.

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

9 Example

To solve the system of equations $AX = B$, where

$$A = \begin{pmatrix} 1 & 2 & 0 & 0 & 5 & 0 \\ 2 & 5 & 3 & 0 & 14 & 0 \\ 0 & 3 & 13 & 0 & 18 & 0 \\ 0 & 0 & 0 & 16 & 8 & 24 \\ 5 & 14 & 18 & 8 & 55 & 17 \\ 0 & 0 & 0 & 24 & 17 & 77 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 6 & -10 \\ 15 & -21 \\ 11 & -3 \\ 0 & 24 \\ 51 & -39 \\ 46 & 67 \end{pmatrix}$$

Here A is symmetric and positive-definite and must first be factorized by F01MCF.

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.

```
*     F04MCF Example Program Text
*     Mark 14 Revised.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NMAX, IRMAX, NRB, NRX, LALMAX
      PARAMETER        (NMAX=6,IRMAX=2,NRB=NMAX,NRX=NMAX,LALMAX=14)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, IR, ISELEC, K, K1, K2, LAL, N
*     .. Local Arrays ..
      real             A(LALMAX), AL(LALMAX), B(NRB,IRMAX), D(NMAX),
     +                 X(NRX,IRMAX)
      INTEGER          NROW(NMAX)
*     .. External Subroutines ..
      EXTERNAL         F01MCF, F04MCF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F04MCF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      WRITE (NOUT,*)
      IF (N.GT.0 .AND. N.LE.NMAX) THEN
         READ (NIN,*) (NROW(I),I=1,N)
         K2 = 0
         DO 20 I = 1, N
            K1 = K2 + 1
            K2 = K2 + NROW(I)
            READ (NIN,*) (A(K),K=K1,K2)
   20    CONTINUE
         READ (NIN,*) IR
         IF (IR.GT.0 .AND. IR.LE.IRMAX) THEN
            READ (NIN,*) ((B(I,K),K=1,IR),I=1,N)
            LAL = K2
            IF (LAL.LE.LALMAX) THEN
               IFAIL = 1
*
               CALL F01MCF(N,A,LAL,NROW,AL,D,IFAIL)
*
               IF (IFAIL.EQ.0) THEN
                  ISELEC = 1
                  IFAIL = 1
*
                  CALL F04MCF(N,AL,LAL,D,NROW,IR,B,NRB,ISELEC,X,NRX,
     +                        IFAIL)
*
```

```
                  IF (IFAIL.EQ.0) THEN
                     WRITE (NOUT,*) ' Solution'
                     DO 40 I = 1, N
                        WRITE (NOUT,99998) (X(I,K),K=1,IR)
   40                CONTINUE
                  ELSE
                     WRITE (NOUT,99999) 'F04MCF fails with IFAIL =',
     +                 IFAIL
                  END IF
               ELSE
                  WRITE (NOUT,99999) 'F01MCF fails with IFAIL =', IFAIL
               END IF
            ELSE
               WRITE (NOUT,*)
               WRITE (NOUT,99999) 'LAL is out of range: LAL = ', LAL
            END IF
         ELSE
            WRITE (NOUT,*)
            WRITE (NOUT,99999) 'IR is out of range: IR = ', IR
         END IF
      ELSE
         WRITE (NOUT,99999) 'N is out of range: N = ', N
      END IF
      STOP
*
99999 FORMAT (1X,A,I5)
99998 FORMAT (1X,8F9.3)
      END
```

9.2 Program Data

```
F04MCF Example Program Data
  6
  1  2  2  1  5  3
   1.0
   2.0   5.0
   3.0  13.0
  16.0
   5.0  14.0  18.0   8.0  55.0
  24.0  17.0  77.0
  2
   6.0 -10.0
  15.0 -21.0
  11.0  -3.0
   0.0  24.0
  51.0 -39.0
  46.0  67.0
```

9.3 Program Results

```
 F04MCF Example Program Results

  Solution
    -3.000    4.000
     2.000   -2.000
    -1.000    3.000
    -2.000    1.000
     1.000   -2.000
     1.000    1.000
```