

F04AHF – NAG Fortran Library Routine Document

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

F04AHF calculates the accurate solution of a set of real linear equations with multiple right-hand sides, $AX = B$, with iterative refinement, where A has been factorized by F03AFF.

2 Specification

```

SUBROUTINE F04AHF(N, IR, A, IA, AA, IAA, P, B, IB, EPS, X, IX, BB,
1          IBB, K, IFAIL)
  INTEGER      N, IR, IA, IAA, IB, IX, IBB, K, IFAIL
  real        A(IA,N), AA(IAA,N), P(N), B(IB,IR), EPS,
1          X(IX,IR), BB(IBM,IR)

```

3 Description

To solve a set of real linear equations $AX = B$, the routine must be preceded by a call to F03AFF which computes an LU factorization of A with partial pivoting, $PA = LU$, where P is a permutation matrix, L is lower triangular and U is unit upper triangular. An approximation to X is found by forward and backward substitution. The residual matrix $R = B - AX$ is then calculated using *additional precision*, and a correction D to X is found by solving $LUD = PR$. X is replaced by $X + D$, and this iterative refinement of the solution is repeated until full machine accuracy has been obtained.

4 References

- [1] 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 A .
- 2: IR — INTEGER *Input*
On entry: r , the number of right-hand sides.
- 3: A(IA,N) — *real* array *Input*
On entry: the n by n matrix A .
- 4: IA — INTEGER *Input*
On entry: the first dimension of the array A as declared in the (sub)program from which F04AHF is called.
Constraint: $IA \geq N$.
- 5: AA(IAA,N) — *real* array *Input*
On entry: details of the LU factorization, as returned by F03AFF.
- 6: IAA — INTEGER *Input*
On entry: the first dimension of the array AA as declared in the (sub)program from which F04AHF is called.
Constraint: $IAA \geq N$.

- 7:** P(N) — *real* array *Input*
On entry: details of the row interchanges as returned by F03AFF.
- 8:** B(IB,IR) — *real* array *Input*
On entry: the n by r right-hand side matrix B .
- 9:** IB — INTEGER *Input*
On entry: the first dimension of the array B as declared in the (sub)program from which F04AHF is called.
Constraint: $IB \geq N$.
- 10:** EPS — *real* *Input*
On entry: EPS must be set to the value of the *machine precision*.
- 11:** X(IX,IR) — *real* array *Output*
On exit: the n by r solution matrix X .
- 12:** IX — INTEGER *Input*
On entry: the first dimension of the array X as declared in the (sub)program from which F04AHF is called.
Constraint: $IX \geq N$.
- 13:** BB(IBB,IR) — *real* array *Output*
On exit: the n by r final residual matrix $R = B - AX$.
- 14:** IBB — INTEGER *Input*
On entry: the first dimension of the array BB as declared in the (sub)program from which F04AHF is called.
Constraint: $IBB \geq N$.
- 15:** K — INTEGER *Output*
On exit: the number of iterations needed in the refinement process.
- 16:** IFAIL — INTEGER *Input/Output*
On entry: IFAIL must be set to 0, -1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.
On exit: IFAIL = 0 unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

Errors detected by the routine:

IFAIL = 1

The matrix A is too ill-conditioned to produce a correctly rounded solution.

7 Accuracy

The computed solutions should be correct to full machine accuracy. For a detailed error analysis see Wilkinson and Reinsch [1] page 106.

8 Further Comments

The time taken by the routine is approximately proportional to n^2r .

9 Example

To solve the set of linear equations $AX = B$ where

$$A = \begin{pmatrix} 33 & 16 & 72 \\ -24 & -10 & -57 \\ -8 & -4 & -17 \end{pmatrix} \text{ and } B = \begin{pmatrix} -359 \\ 281 \\ 85 \end{pmatrix}.$$

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.

```

*      F04AHF Example Program Text
*      Mark 14 Revised.  NAG Copyright 1989.
*      .. Parameters ..
      INTEGER          NMAX, IR, IAA, IA, IB, IX, IBB
      PARAMETER        (NMAX=8,IR=1,IAA=NMAX,IA=NMAX,IB=NMAX,IX=NMAX,
+                      IBB=NMAX)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*      .. Local Scalars ..
      real            D1, EPS
      INTEGER          I, ID, IFAIL, J, K, N
*      .. Local Arrays ..
      real            A(IA,NMAX), AA(IAA,NMAX), B(IB,IR), BB(IBB,IR),
+                      P(NMAX), X(IX,IR)
*      .. External Functions ..
      real            X02AJF
      EXTERNAL         X02AJF
*      .. External Subroutines ..
      EXTERNAL         F03AFF, F04AHF
*      .. Executable Statements ..
      WRITE (NOUT,*) 'F04AHF 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,*) ((AA(I,J),J=1,N),I=1,N)
          DO 40 I = 1, N
              DO 20 J = 1, N
                  A(J,I) = AA(J,I)
20          CONTINUE
40          CONTINUE
          EPS = X02AJF()
          IFAIL = 1
*
*      Crout decomposition
          CALL F03AFF(N, EPS, AA, IAA, D1, ID, P, IFAIL)
*
          IF (IFAIL.NE.0) THEN
              WRITE (NOUT,99998) 'Error in F03AFF. IFAIL =', IFAIL
          ELSE
              READ (NIN,*) ((B(I,J),J=1,IR),I=1,N)
              IFAIL = 1
*
*      Accurate solution of linear equations
          CALL F04AHF(N, IR, A, IA, AA, IAA, P, B, IB, EPS, X, IX, BB, IBB, K, IFAIL)

```

```
*
      IF (IFAIL.NE.0) THEN
        WRITE (NOUT,99998) 'Error in F04AHF. IFAIL =', IFAIL
      ELSE
        WRITE (NOUT,*) ' Solution'
        DO 60 I = 1, N
          WRITE (NOUT,99999) (X(I,J),J=1,IR)
60      CONTINUE
        END IF
      END IF
    ELSE
      WRITE (NOUT,99998) 'N is out of range: N = ', N
    END IF
  STOP
*
99999 FORMAT (1X,8F9.4)
99998 FORMAT (1X,A,I5)
END
```

9.2 Program Data

F04AHF Example Program Data

```
3
 33  16  72
-24 -10 -57
  -8  -4 -17
-359 281  85
```

9.3 Program Results

F04AHF Example Program Results

```
Solution
 1.0000
-2.0000
-5.0000
```
