NAG Library Routine Document

F04AGF

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

F04AGF calculates the approximate solution of a set of real symmetric positive definite linear equations with multiple right-hand sides, $AX = B$, where A has been factorized by F03AEF.

2 Specification

```
SUBROUTINE F04AGF (N, IR, A, LDA, P, B, LDB, X, LDX)

INTEGER            N, IR, LDA, LDB, LDX
REAL (KIND=nag_wp) A(LDA,N), P(N), B(LDB,IR), X(LDX,IR)
```

3 Description

To solve a set of real linear equations $AX = B$ where A is symmetric positive definite, F04AGF must be preceded by a call to F03AEF which computes a Cholesky factorization of A as $A = LL^{\mathrm{T}}$, where L is lower triangular. The columns x of the solution X are found by forward and backward substitution in $Ly = b$ and $L^{\mathrm{T}}x = y$, where b is a column of the right-hand sides.

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 A.

Constraint: N $\geq$ 1.

2: IR – INTEGER *Input*

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

3: A(LDA,N) – REAL (KIND=nag_wp) array *Input*

On entry: the upper triangle of the n by n positive definite symmetric matrix A, and the subdiagonal elements of its Cholesky factor L, as returned by F03AEF.

On exit: is used as internal workspace, but is restored on exit.

4: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F04AGF is called.

Constraint: LDA $\geq$ N.

5: P(N) – REAL (KIND=nag_wp) array *Input*

On entry: the reciprocals of the diagonal elements of L, as returned by F03AEF.

On exit: is used as internal workspace, but is restored on exit.

6: B(LDB,IR) – REAL (KIND=nag_wp) array *Input*

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

7: LDB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F04AGF is called.

Constraint: LDB $\geq$ N.

8: X(LDX,IR) – REAL (KIND=nag_wp) array *Output*

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

9: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which F04AGF is called.

Constraint: LDX $\geq$ N.

6 Error Indicators and Warnings

If an error is detected in an input parameter F04AGF will act as if a soft noisy exit has been requested (see Section 3.3.4 in the Essential Introduction).

7 Accuracy

The accuracy of the computed solutions depends on the conditioning of the original matrix. For a detailed error analysis see page 39 of Wilkinson and Reinsch (1971).

8 Further Comments

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

9 Example

This example solves the set of linear equations $AX = B$ where

$$A = \begin{pmatrix} 5 & 7 & 6 & 5 \\ 7 & 10 & 8 & 7 \\ 6 & 8 & 10 & 9 \\ 5 & 7 & 9 & 10 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 23 \\ 32 \\ 33 \\ 31 \end{pmatrix}.$$

9.1 Program Text

```
    Program f04agfe

!     F04AGF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: f03aef, f04agf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: d1
```

```
      Integer                          :: i, id, ifail, ir, lda, ldb, ldx, n
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: a(:,:), b(:,:), p(:), x(:,:)
!     .. Executable Statements ..
      Write (nout,*) 'F04AGF Example Program Results'
      Write (nout,*)
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n
      ir = 1
      lda = n
      ldb = n
      ldx = n
      Allocate (a(lda,n),b(ldb,ir),p(n),x(ldx,ir))
      Read (nin,*)(a(i,1:n),i=1,n), (b(i,1:ir),i=1,n)

!     ifail: behaviour on error exit
!             =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
      ifail = 0
!     Cholesky decomposition
      Call f03aef(n,a,lda,p,d1,id,ifail)

!     Approximate solution of linear equations
      ifail = 0
      Call f04agf(n,ir,a,lda,p,b,ldb,x,ldx)

      Write (nout,*) ' Solution'
      Do i = 1, n
        Write (nout,99999) x(i,1:ir)
      End Do

99999 Format (1X,8F9.4)
    End Program f04agfe
```

9.2 Program Data

```
F04AGF Example Program Data
  4                      : n
    5    7    6    5
    7   10    8    7
    6    8   10    9
    5    7    9   10
   23   32   33   31    : matrices A and B
```

9.3 Program Results

```
 F04AGF Example Program Results

  Solution
    1.0000
    1.0000
    1.0000
    1.0000
```