F04FCFP

NAG Parallel Library Routine Document

Note: Before using this routine, please read the Users' Note for your implementation to check for implementation-dependent details. You are advised to enclose any calls to NAG Parallel Library routines between calls to Z01AAFP and Z01ABFP.

1 Description

F04FCFP calculates the solution of a set of complex Hermitian positive-definite linear equations

$$AX = B$$

with multiple right-hand sides, using a Cholesky factorization, where A and B are n by n and n by r matrices respectively.

The routine first computes a Cholesky factorization of A as $A = LL^H$, where L is lower triangular with real diagonal elements. An approximation to X is found by forward and backward substitution.

2 Specification

```
SUBROUTINE F04FCFP(ICNTXT, N, NB, A, LDA, NRHS, B, LDB, IFAIL)
COMPLEX*16         A(LDA,*), B(LDB,*)
INTEGER            ICNTXT, N, NB, LDA, NRHS, LDB, IFAIL
```

3 Data Distribution

3.1 Definitions

The following definitions are used in describing the data distribution within this document:

m_p	–	the number of rows in the logical processor grid.
n_p	–	the number of columns in the logical processor grid.
p_r	–	the row grid coordinate of the calling processor.
p_c	–	the column grid coordinate of the calling processor.
N_b	–	the blocking factor for the distribution of the rows and columns of the matrix.
numroc$(\alpha, b_\ell, q, s, k)$	–	a function which gives the **num**ber of **r**ows **o**r **c**olumns of a distributed matrix owned by the processor with the row or column coordinate q (p_r or p_c), where α is the total number of rows or columns of the matrix, b_ℓ is the blocking factor used (N_b), s is the row or column coordinate of the processor that possesses the first row or column of the distributed matrix and k is either n_p or m_p. The Library provides the function Z01CAFP (NUMROC) for the evaluation of this function.

3.2 Global and Local Arguments

The input arguments N, NB, NRHS and IFAIL are all global and so must have the same values on entry to the routine on each processor. The output argument IFAIL is global, and will have the same value on exit from the routine on each processor. The remaining arguments are local.

3.3 Distribution Strategy

The matrix A must be partitioned into N_b by N_b square blocks and stored in an array A in a cyclic 2-d block distribution. In this routine, the logical processor {0,0} of the processor grid must always possess the first block of the distributed matrix (i.e., $s = 0$ in the function numroc). This data distribution is described in more detail in the F04 Chapter Introduction. The right-hand sides of the equation, B, must be stored in the array B, also in a cyclic 2-d block distribution.

This routine assumes that the data has already been correctly distributed, and if this is not the case will fail to produce correct results. However, the Library provides some utility routines which assist you in distributing data correctly. Descriptions of these routines can be found in Chapters F01 and X04 of the NAG Parallel Library Manual.

4 Arguments

1: ICNTXT — INTEGER *Local Input*

On entry: the BLACS context used by the communication mechanism, usually returned by a call to Z01AAFP.

2: N — INTEGER *Global Input*

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

Constraint: $\mathrm{N} \geq 0$.

3: NB — INTEGER *Global Input*

On entry: the blocking factor, N_b, used to distribute the rows and columns of the matrices A and B.

Constraints: $\mathrm{NB} \geq 1$.

4: A(LDA,∗) — COMPLEX*16 array *Local Input/Local Output*

Note: the second dimension of the array A must be at least max(1,numroc(N,NB,p_c,0,n_p)).

On entry: the local part of the Hermitian positive-definite matrix A. The lower triangle of A must be stored. The elements of the matrix above the diagonal are not referenced.

On exit: the lower triangle of A is overwritten by the Cholesky factor L.

5: LDA — INTEGER *Local Input*

On entry: the size of the first dimension of the array A as declared in the (sub)program from which F04FCFP is called.

Constraint: LDA $\geq$ max(1,numroc(N,NB,p_r,0,m_p)).

6: NRHS — INTEGER *Global Input*

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

Constraint: $\mathrm{NRHS} \geq 0$.

7: B(LDB,∗) — COMPLEX*16 array *Local Input/Local Output*

Note: the second dimension of the array B must be at least max(1,numroc(NRHS,NB,p_c,0,n_p)).

On entry: the local part of the the n by r right-hand side matrix B.

On exit: the n by r solution matrix X distributed in the same cyclic 2-d block distribution.

8: LDB — INTEGER *Local Input*

On entry: the size of the first dimension of the array B as declared in the (sub)program from which F04FCFP is called.

Constraint: LDB $\geq$ max(1,numroc(N,NB,p_r,0,m_p)).

9: IFAIL — INTEGER *Global Input/Global Output*

On entry: IFAIL must be set to 0, -1 or 1. For users not familiar with this parameter (described in the Essential Introduction) the recommended values are:

IFAIL = 0, if multigridding is **not** employed;
IFAIL = -1, if multigridding is employed.

On exit: IFAIL = 0 unless the routine detects an error (see Section 5).

5 Errors 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 detected by the routine:

IFAIL $= -2000$

The routine has been called with an invalid value of ICNTXT on one or more processors.

IFAIL $= -1000$

The logical processor grid and library mechanism (Library Grid) have not been correctly defined, see Z01AAFP.

IFAIL $= -i$

On entry, the ith argument had an invalid value. This error occurred either because a global argument did not have the same value on all the logical processors (see Section 3.2), or because its value was incorrect. An explanatory message distinguishes between these two cases.

IFAIL $= 1$

The matrix A is either not positive-definite, or is nearly singular.

6 Further Comments

The total number of floating-point operations is approximately $\frac{4}{3}n^3 + 8n^2r$.

6.1 Algorithmic Detail

The algorithm used by this routine is described in Chapter 3 of [1].

6.2 Parallelism Detail

The Level 3 BLAS operations used in this routine are carried out in parallel.

6.3 Accuracy

For each right-hand side vector b, the computed solution x is the exact solution of a perturbed system of equations $(A+E)x = b$, where

$$\| E \| \leq \epsilon c(n) \| A \|,$$

$c(n)$ is a modest function of n, ϵ is the ***machine precision***.

If $\hat{x}$ is the true solution, then the computed solution x satisfies the bound

$$\frac{\|x - \hat{x}\|}{\|x\|} \leq \epsilon(n)\mathrm{cond}(A)\epsilon$$

where $\mathrm{cond}(A) = \|A\| \cdot \|A^{-1}\|$.

7 References

[1] Anderson E, Bai Z, Bischof C, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A, Ostrouchov S and Sorensen D (1995) *LAPACK Users' Guide* (2nd Edition) SIAM, Philadelphia

[2] Golub G H and Van Loan C F (1989) *Matrix Computations* Johns Hopkins University Press (2nd Edition), Baltimore

8 Example

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

$$A = \begin{pmatrix} 3.23+0.00i & 1.51-1.92i & 1.90+0.84i & 0.42+2.50i \\ 1.51+1.92i & 3.58+0.00i & -0.23+1.11i & -1.18+1.37i \\ 1.90-0.84i & -0.23-1.11i & 4.09+0.00i & 2.33-0.14i \\ 0.42-2.50i & -1.18-1.37i & 2.33+0.14i & 4.29+0.00i \end{pmatrix} \text{ and}$$

$$B = \begin{pmatrix} 3.93{-}6.14i & 1.48{+}6.58i \\ 6.17{+}9.42i & 4.65{-}4.75i \\ -7.17{-}21.83i & -4.91{+}2.29i \\ 1.99{-}14.38i & 7.64{-}10.79i \end{pmatrix}.$$

The example uses a 2 by 2 logical processor grid and a block size of 2.

Note: the listing of the Example Program presented below does not give a full pathname for the data file being opened, but in general the user must give the full pathname in this and any other OPEN statement.

8.1 Example Text

```
*     F04FCFP Example Program Text
*     NAG Parallel Library Release 2. NAG Copyright 1996.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NB
      PARAMETER        (NB=2)
      INTEGER          NMAX, LDA, LDB, NRHMAX, LW
      PARAMETER        (NMAX=8,LDA=NMAX,LDB=NMAX,NRHMAX=2,LW=NMAX)
*     .. Local Scalars ..
      INTEGER          ICNTXT, IFAIL, MP, N, NP, NRHS
      LOGICAL          ROOT
      CHARACTER*80     FORMAT
*     .. Local Arrays ..
      COMPLEX*16       A(LDA,NMAX), B(LDB,NRHMAX), WORK(LW)
*     .. External Functions ..
      LOGICAL          Z01ACFP
      EXTERNAL         Z01ACFP
*     .. External Subroutines ..
      EXTERNAL         F04FCFP, X04BVFP, X04BWFP, Z01AAFP, Z01ABFP
*     .. Executable Statements ..
      ROOT = Z01ACFP()
      IF (ROOT) WRITE (NOUT,*) 'F04FCFP Example Program Results'
*
      MP = 2
      NP = 2
      IFAIL = 0
*
      CALL Z01AAFP(ICNTXT,MP,NP,IFAIL)
*
      OPEN (NIN,FILE='f04fcfpe.d')
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N, NRHS
      READ (NIN,*) FORMAT
*
      IF (N.LE.NMAX .AND. NRHS.LE.NRHMAX) THEN
*
         IFAIL = 0
```

```
*
*       Read in matrices A and B
*
         CALL X04BVFP(ICNTXT,NIN,N,N,NB,A,LDA,IFAIL)
*
         CALL X04BVFP(ICNTXT,NIN,N,NRHS,NB,B,LDB,IFAIL)
*
         CALL F04FCFP(ICNTXT,N,NB,A,LDA,NRHS,B,LDB,IFAIL)
*
*       Print solution(s)
*
         IF (ROOT) THEN
            WRITE (NOUT,*)
            WRITE (NOUT,*) 'Solution(s)'
            WRITE (NOUT,*)
         END IF
*
         CALL X04BWFP(ICNTXT,NOUT,N,NRHS,NB,B,LDB,FORMAT,WORK,IFAIL)
*
      END IF
*
      CLOSE (NIN)
*
      IFAIL = 0
      CALL Z01ABFP(ICNTXT,'N',IFAIL)
*
      STOP
      END
```

8.2 Example Data

```
F04FCFP Example Program Data
 4 2                                                         :Values of N and NRHS
 '(F7.4)'                                                    :Value of FORMAT
 (3.23, 0.00) ( 0.0 , 0.0 ) ( 0.0 , 0.0 ) ( 0.0 , 0.0 )
 (1.51, 1.92) ( 3.58, 0.00) ( 0.0 , 0.0 ) ( 0.0 , 0.0 )
 (1.90,-0.84) (-0.23,-1.11) ( 4.09, 0.00) ( 0.0 , 0.0 )
 (0.42,-2.50) (-1.18,-1.37) ( 2.33, 0.14) ( 4.29, 0.00)  :End of matrix A
 ( 3.93, -6.14) ( 1.48,  6.58)
 ( 6.17,  9.42) ( 4.65, -4.75)
 (-7.17,-21.83) (-4.91,  2.29)
 ( 1.99,-14.38) ( 7.64,-10.79)                           :End of matrix B
```

8.3 Example Results

```
 F04FCFP Example Program Results

 Solution(s)

 ( 1.0000,-1.0000) (-1.0000, 2.0000)
 ( 0.0000, 3.0000) ( 3.0000,-4.0000)
 (-4.0000,-5.0000) (-2.0000, 3.0000)
 ( 2.0000, 1.0000) ( 4.0000,-5.0000)
```