NAG Toolbox for MATLAB

f07mr

1 Purpose

f07mr computes the Bunch–Kaufman factorization of a complex Hermitian indefinite matrix.

2 Syntax

```
[a, ipiv, info] = f07mr(uplo, a, 'n', n)
```

3 Description

f07mr factorizes a complex Hermitian matrix A, using the Bunch–Kaufman diagonal pivoting method. A is factorized either as $A = PUDU^{\mathrm{H}}P^{\mathrm{T}}$ if **uplo** = 'U' or $A = PLDL^{\mathrm{H}}P^{\mathrm{T}}$ if **uplo** = 'L', where P is a permutation matrix, U (or L) is a unit upper (or lower) triangular matrix and D is an Hermitian block diagonal matrix with 1 by 1 and 2 by 2 diagonal blocks; U (or L) has 2 by 2 unit diagonal blocks corresponding to the 2 by 2 blocks of D. Row and column interchanges are performed to ensure numerical stability while keeping the matrix Hermitian.

This method is suitable for Hermitian matrices which are not known to be positive-definite. If A is in fact positive-definite, no interchanges are performed and no 2 by 2 blocks occur in D.

4 References

Golub G H and Van Loan C F 1996 *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

5.1 Compulsory Input Parameters

1: **uplo – string**

Indicates whether the upper or lower triangular part of A is stored and how A is to be factorized.

uplo = 'U'

The upper triangular part of A is stored and A is factorized as $PUDU^{\mathrm{H}}P^{\mathrm{T}}$, where U is upper triangular.

uplo = 'L'

The lower triangular part of A is stored and A is factorized as $PLDL^{\mathrm{H}}P^{\mathrm{T}}$, where L is lower triangular.

Constraint: **uplo** = 'U' or 'L'.

2: **a(lda,∗) – complex array**

The first dimension of the array **a** must be at least $\max(1, \mathbf{n})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

The n by n Hermitian indefinite matrix A.

If **uplo** = 'U', the upper triangular part of A must be stored and the elements of the array below the diagonal are not referenced.

If **uplo** = 'L', the lower triangular part of A must be stored and the elements of the array above the diagonal are not referenced.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The second dimension of the array **a**.

n, the order of the matrix A.

Constraint: $\mathbf{n} \geq 0$.

5.3 Input Parameters Omitted from the MATLAB Interface

lda, work, lwork

5.4 Output Parameters

1: **a(lda,$*$) – complex array**

The first dimension of the array **a** must be at least $\max(1, \mathbf{n})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

The upper or lower triangle of A contains details of the block diagonal matrix D and the multipliers used to obtain the factor U or L as specified by **uplo**.

2: **ipiv($*$) – int32 array**

Note: the dimension of the array **ipiv** must be at least $\max(1, \mathbf{n})$.

Details of the interchanges and the block structure of D. More precisely,

if $\mathbf{ipiv}(i) = k > 0$, d_{ii} is a 1 by 1 pivot block and the ith row and column of A were interchanged with the kth row and column;

if **uplo** = 'U' and $\mathbf{ipiv}(i-1) = \mathbf{ipiv}(i) = -l < 0$, $\begin{pmatrix} d_{i-1,i-1} & \bar{d}_{i,i-1} \\ \bar{d}_{i,i-1} & d_{ii} \end{pmatrix}$ is a 2 by 2 pivot block and the $(i-1)$th row and column of A were interchanged with the lth row and column;

if **uplo** = 'L' and $\mathbf{ipiv}(i) = \mathbf{ipiv}(i+1) = -m < 0$, $\begin{pmatrix} d_{ii} & d_{i+1,i} \\ d_{i+1,i} & d_{i+1,i+1} \end{pmatrix}$ is a 2 by 2 pivot block and the $(i+1)$th row and column of A were interchanged with the mth row and column.

3: **info – int32 scalar**

info $= 0$ unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

info $= -i$

If **info** $= -i$, parameter i had an illegal value on entry. The parameters are numbered as follows:

1: **uplo**, 2: **n**, 3: **a**, 4: **lda**, 5: **ipiv**, 6: **work**, 7: **lwork**, 8: **info**.

It is possible that **info** refers to a parameter that is omitted from the MATLAB interface. This usually indicates that an error in one of the other input parameters has caused an incorrect value to be inferred.

info > 0

If **info** $= i$, $d(i,i)$ is exactly zero. The factorization has been completed, but the block diagonal matrix D is exactly singular, and division by zero will occur if it is used to solve a system of equations.

7 Accuracy

If **uplo** = 'U', the computed factors U and D are the exact factors of a perturbed matrix $A + E$, where

$$|E| \leq c(n)\epsilon P|U||D|\left|U^{\mathrm{H}}\right|P^{\mathrm{T}},$$

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

If **uplo** = 'L', a similar statement holds for the computed factors L and D.

8 Further Comments

The elements of D overwrite the corresponding elements of A; if D has 2 by 2 blocks, only the upper or lower triangle is stored, as specified by **uplo**.

The unit diagonal elements of U or L and the 2 by 2 unit diagonal blocks are not stored. The remaining elements of U or L are stored in the corresponding columns of the array **a**, but additional row interchanges must be applied to recover U or L explicitly (this is seldom necessary). If $\mathbf{ipiv}(i) = i$, for $i = 1, 2, \ldots, n$ (as is the case when A is positive-definite), then U or L is stored explicitly (except for its unit diagonal elements which are equal to 1).

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

A call to f07mr may be followed by calls to the functions:

f07ms to solve $AX = B$;

f07mu to estimate the condition number of A;

f07mw to compute the inverse of A.

The real analogue of this function is f07md.

9 Example

```
uplo = 'L';
a = [complex(-1.36, +0), complex(0, +0), complex(0, 0), complex(0, 0);
       complex(1.58, -0.9), complex(-8.87, 0), complex(0, 0), complex(0,
0);
        complex(2.21, +0.21), complex(-1.84, +0.03), complex(-4.63, +0),
complex(0, 0);
       complex(3.91, -1.5), complex(-1.78, -1.18), complex(0.11, -0.11),
complex(-1.84, +0)];
[aOut, ipiv, info] = f07mr(uplo, a)
```

```
aOut =
  -1.3600                  0                  0                  0
   3.9100 - 1.5000i  -1.8400                  0                  0
   0.3100 + 0.0433i   0.5637 + 0.2850i  -5.4176                  0
  -0.1518 + 0.3743i   0.3397 + 0.0303i   0.2997 + 0.1578i  -7.1028
ipiv =
          -4
          -4
           3
           4
info =
           0
```