NAG Toolbox for MATLAB

f07ns

1 Purpose

f07ns solves a complex symmetric system of linear equations with multiple right-hand sides,

$$AX = B,$$

where A has been factorized by f07nr.

2 Syntax

```
[b, info] = f07ns(uplo, a, ipiv, b, 'n', n, 'nrhs_p', nrhs_p)
```

3 Description

f07ns is used to solve a complex symmetric system of linear equations $AX = B$, this function must be preceded by a call to f07nr which computes the Bunch–Kaufman factorization of A.

If **uplo** = 'U', $A = PUDU^{\mathrm{T}}P^{\mathrm{T}}$, where P is a permutation matrix, U is an upper triangular matrix and D is a symmetric block diagonal matrix with 1 by 1 and 2 by 2 blocks; the solution X is computed by solving $PUDY = B$ and then $U^{\mathrm{T}}P^{\mathrm{T}}X = Y$.

If **uplo** = 'L', $A = PLDL^{\mathrm{T}}P^{\mathrm{T}}$, where L is a lower triangular matrix; the solution X is computed by solving $PLDY = B$ and then $L^{\mathrm{T}}P^{\mathrm{T}}X = Y$.

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 how A has been factorized.

uplo = 'U'

$A = PUDU^{\mathrm{T}}P^{\mathrm{T}}$, where U is upper triangular.

uplo = 'L'

$A = PLDL^{\mathrm{T}}P^{\mathrm{T}}$, 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 $\max(1, \mathbf{n})$

Details of the factorization of A, as returned by f07nr.

3: **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, as returned by f07nr.

4: **b**(**ldb**,$*$) – **complex array**

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

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

The n by r right-hand side matrix B.

5.2 Optional Input Parameters

1: **n** – **int32 scalar**

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

n, the order of the matrix A.

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

2: **nrhs_p** – **int32 scalar**

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

r, the number of right-hand sides.

Constraint: $\mathbf{nrhs\_p} \geq 0$.

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldb

5.4 Output Parameters

1: **b**(**ldb**,$*$) – **complex array**

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

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

The n by r solution matrix X.

2: **info** – **int32 scalar**

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

6 Error Indicators and Warnings

Errors or warnings detected by the function:

$\mathbf{info} = -i$

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

1: **uplo**, 2: **n**, 3: **nrhs_p**, 4: **a**, 5: **lda**, 6: **ipiv**, 7: **b**, 8: **ldb**, 9: **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.

7 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

if **uplo** = 'U', $|E| \leq c(n)\epsilon P|U||D|\left|U^{\mathrm{T}}\right|P^{\mathrm{T}}$;
if **uplo** = 'L', $|E| \leq c(n)\epsilon P|L||D|\left|L^{\mathrm{T}}\right|P^{\mathrm{T}}$,

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

If $\hat{x}$ is the true solution, then the computed solution x satisfies a forward error bound of the form

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

where $\mathrm{cond}(A,x) = \left\||A^{-1}||A||x|\right\|_\infty / \|x\|_\infty \leq \mathrm{cond}(A) = \left\||A^{-1}||A|\right\|_\infty \leq \kappa_\infty(A)$.

Note that $\mathrm{cond}(A,x)$ can be much smaller than $\mathrm{cond}(A)$.

Forward and backward error bounds can be computed by calling f07nv, and an estimate for $\kappa_\infty(A)$ $(=\kappa_1(A))$ can be obtained by calling f07nu.

8 Further Comments

The total number of real floating-point operations is approximately $8n^2r$.

This function may be followed by a call to f07nv to refine the solution and return an error estimate.

The real analogue of this function is f07me.

9 Example

```
uplo = 'L';
a = [complex(-0.39, -0.71), complex(0, +0), complex(0, +0), complex(0,
+0);
         complex(-7.86, -2.96), complex(-2.83, -0.03), complex(0, +0),
complex(0, +0);
            complex(0.5278724801640799, -0.3714660014825906), complex(-
0.6078391056683192, ...
     +0.281079647893122), complex(4.407906236731014, +5.399120676796941),
complex(0, +0);
             complex(0.442558238872675, +0.1936483698297402), complex(-
0.4822822975185383, ...
               +0.01498936219105284),   complex(-0.1070821880092683,   -
0.3156780862488454), ...
     complex(-2.095414887840057, -2.201139281440786)];
ipiv = [int32(-3);
     int32(-3);
     int32(3);
     int32(4)];
b = [complex(-55.64, +41.22), complex(-19.09, -35.97);
     complex(-48.18, +66), complex(-12.08, -27.02);
     complex(-0.49, -1.47), complex(6.95, +20.49);
     complex(-6.43, +19.24), complex(-4.59, -35.53)];
[bOut, info] = f07ns(uplo, a, ipiv, b)
```

```
bOut =
   1.0000 - 1.0000i  -2.0000 - 1.0000i
  -2.0000 + 5.0000i   1.0000 - 3.0000i
   3.0000 - 2.0000i   3.0000 + 2.0000i
  -4.0000 + 3.0000i  -1.0000 + 1.0000i
info =
           0
```