NAG Toolbox for MATLAB

f07cs

1 Purpose

f07cs computes the solution to a complex system of linear equations $AX = B$ or $A^{\mathrm{T}}X = B$ or $A^{\mathrm{H}}X = B$, where A is an n by n tridiagonal matrix and X and B are n by r matrices, using the LU factorization returned by f07cr.

2 Syntax

```
[b, info] = f07cs(trans, dl, d, du, du2, ipiv, b, 'n', n, 'nrhs_p', nrhs_p)
```

3 Description

f07cs should be preceded by a call to f07cr, which uses Gaussian elimination with partial pivoting and row interchanges to factorize the matrix A as

$$A = PLU,$$

where P is a permutation matrix, L is unit lower triangular with at most one nonzero subdiagonal element in each column, and U is an upper triangular band matrix, with two superdiagonals. f07cs then utilizes the factorization to solve the required equations.

4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D 1999 *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia URL: http://www.netlib.org/lapack/lug

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: **trans – string**

Specifies the equations to be solved as follows:

trans = 'N'

Solve $AX = B$ for X.

trans = 'T'

Solve $A^{\mathrm{T}}X = B$ for X.

trans = 'C'

Solve $A^{\mathrm{H}}X = B$ for X.

Constraint: **trans** = 'N', 'T' or 'C'.

2: **dl**$(*)$ **– complex array**

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

Must contain the $(n - 1)$ multipliers that define the matrix L of the LU factorization of A.

3: **d**$(*)$ **– complex array**

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

Must contain the n diagonal elements of the upper triangular matrix U from the LU factorization of A.

4: **du**$(*)$ **– complex array**

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

Must contain the $(n - 1)$ elements of the first superdiagonal of U.

5: **du2**$(*)$ **– complex array**

Note: the dimension of the array **du2** must be at least $\max(1, \mathbf{n} - 2)$.

Must contain the $(n - 2)$ elements of the second superdiagonal of U.

6: **ipiv**$(*)$ **– int32 array**

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

Must contain the n pivot indices that define the permutation matrix P. At the ith step, row i of the matrix was interchanged with row **ipiv**(i), and **ipiv**(i) must always be either i or $(i + 1)$, **ipiv**$(i) = i$ indicating that a row interchange was not performed.

7: **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 matrix of right-hand sides B.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The dimension of the array **d** 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, i.e., the number of columns of the matrix B.

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

5.3 Input Parameters Omitted from the MATLAB Interface

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**

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: **trans**, 2: **n**, 3: **nrhs_p**, 4: **dl**, 5: **d**, 6: **du**, 7: **du2**, 8: **ipiv**, 9: **b**, 10: **ldb**, 11: **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

The computed solution for a single right-hand side, $\hat{x}$, satisfies an equation of the form

$$(A + E)\hat{x} = b,$$

where

$$\|E\|_1 = O(\epsilon)\|A\|_1$$

and ϵ is the ***machine precision***. An approximate error bound for the computed solution is given by

$$\frac{\|\hat{x} - x\|_1}{\|x\|_1} \leq \kappa(A)\frac{\|E\|_1}{\|A\|_1},$$

where $\kappa(A) = \left\|A^{-1}\right\|_1\|A\|_1$, the condition number of A with respect to the solution of the linear equations. See Section 4.4 of Anderson *et al.* 1999 for further details.

Following the use of this function f07cu can be used to estimate the condition number of A and f07cv can be used to obtain approximate error bounds.

8 Further Comments

The total number of floating-point operations required to solve the equations $AX = B$ or $A^{\mathrm{T}}X = B$ or $A^{\mathrm{H}}X = B$ is proportional to nr.

The real analogue of this function is f07ce.

9 Example

```
trans = 'No transpose';
dl = [complex(-0.78, -0.26);
      complex(0.162, -0.4859999999999999);
      complex(-0.04516923076923077, -0.0009538461538460446);
      complex(-0.3978553846153843, -0.0562030769230711)];
d = [complex(1, -2);
      complex(1, +1);
      complex(2, -3);
      complex(1, +1);
      complex(-1.339863692307691, +0.2875264615384604)];
du = [complex(-1.3, +1.3);
      complex(-1.3, +3.3);
      complex(-0.3, +4.3);
      complex(-3.3, +1.3)];
du2 = [complex(2, +1);
```

```
     complex(-1, +1);
     complex(1, -1)];
ipiv = [int32(2);
     int32(3);
     int32(4);
     int32(5);
     int32(5)];
b = [complex(2.4, -5), complex(2.7, +6.9);
     complex(3.4, +18.2), complex(-6.9, -5.3);
     complex(-14.7, +9.699999999999999), complex(-6, -0.6);
     complex(31.9, -7.7), complex(-3.9, +9.300000000000001);
     complex(-1, +1.6), complex(-3, +12.2)];
[bOut, info] = f07cs(trans, dl, d, du, du2, ipiv, b)
```

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