NAG Toolbox for MATLAB

f07js

1 Purpose

f07js computes the solution to a complex system of linear equations $AX = B$, where A is an n by n Hermitian positive-definite tridiagonal matrix and X and B are n by r matrices, using the LDL^{H} factorization returned by f07jr.

2 Syntax

```
[b, info] = f07js(uplo, d, e, b, 'n', n, 'nrhs_p', nrhs_p)
```

3 Description

f07js should be preceded by a call to f07jr, which computes a modified Cholesky factorization of the matrix A as

$$A = LDL^{\mathrm{H}},$$

where L is a unit lower bidiagonal matrix and D is a diagonal matrix, with positive diagonal elements. f07js then utilizes the factorization to solve the required equations. Note that the factorization may also be expressed as

$$A = U^{\mathrm{H}}DU,$$

where U is a unit upper bidiagonal matrix.

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

5 Parameters

5.1 Compulsory Input Parameters

1: **uplo – string**

Specifies the form of the factorization as follows:

uplo = 'U'

$$A = U^{\mathrm{H}}DU.$$

uplo = 'L'

$$A = LDL^{\mathrm{H}}.$$

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

2: **d**$(*)$ **– double array**

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

Must contain the n diagonal elements of the diagonal matrix D from the LDL^{H} or $U^{\mathrm{H}}DU$ factorization of A.

3: **e**$(*)$ **– complex array**

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

If **uplo** = 'U', **e** must contain the $(n - 1)$ superdiagonal elements of the unit upper bidiagonal matrix U from the $U^{\mathrm{H}}DU$ factorization of A.

If **uplo** = 'L', **e** must contain the $(n - 1)$ subdiagonal elements of the unit lower bidiagonal matrix L from the LDL^{H} factorization of A.

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 matrix of right-hand sides B.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The dimension of the array **d**.

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: **uplo**, 2: **n**, 3: **nrhs_p**, 4: **d**, 5: **e**, 6: **b**, 7: **ldb**, 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.

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 f07ju can be used to estimate the condition number of A and f07jv can be used to obtain approximate error bounds.

8 Further Comments

The total number of floating-point operations required to solve the equations $AX = B$ is proportional to nr.

The real analogue of this function is f07je.

9 Example

```
uplo = 'U';
d = [16;
     9;
     1;
     4];
e = [complex(1, -1);
     complex(2, +1);
     complex(1, +4)];
b = [complex(64, +16), complex(-16, -32);
     complex(93, +62), complex(61, -66);
     complex(78, -80), complex(71, -74);
     complex(14, -27), complex(35, +15)];
[bOut, info] = f07js(uplo, d, e, b)
```

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