NAG Toolbox for MATLAB

f07jp

1 Purpose

f07jp uses the factorization

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

to compute 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. Error bounds on the solution and a condition estimate are also provided.

2 Syntax

```
[df, ef, x, rcond, ferr, berr, info] = f07jp(fact, d, e, df, ef, b, 'n',
n, 'nrhs_p', nrhs_p)
```

3 Description

f07jp performs the following steps:

1. If **fact** = 'N', the matrix A is factorized as $A = LDL^{\mathrm{H}}$, where L is a unit lower bidiagonal matrix and D is diagonal. The factorization can also be regarded as having the form $A = U^{\mathrm{H}}DU$.
2. If the leading i by i principal minor is not positive-definite, then the function returns with **info** $= i$. Otherwise, the factored form of A is used to estimate the condition number of the matrix A. If the reciprocal of the condition number is less than ***machine precision***, **info** $\geq N+1$ is returned as a warning, but the function still goes on to solve for X and compute error bounds as described below.
3. The system of equations is solved for X using the factored form of A.
4. Iterative refinement is applied to improve the computed solution matrix and to calculate error bounds and backward error estimates for it.

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

Higham N J 2002 *Accuracy and Stability of Numerical Algorithms* (2nd Edition) SIAM, Philadelphia

5 Parameters

5.1 Compulsory Input Parameters

1: **fact – string**

Specifies whether or not the factorized form of the matrix A is supplied on entry:

fact = 'F'
df and **ef** contain the factorized form of A. **d**, **e**, **df** and **ef** will not be modified.

fact = 'N'
The matrix A will be copied to **df** and **ef** and factorized.

Constraint: **fact** = 'F' or 'N'.

2: **d**($*$) – **double array**

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

The n diagonal elements of the tridiagonal matrix A.

3: **e**($*$) – **complex array**

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

The $(n - 1)$ subdiagonal elements of the tridiagonal matrix A.

4: **df**($*$) – **double array**

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

If **fact** = 'F', **df** contains the n diagonal elements of the diagonal matrix D from the LDL^{H} factorization of A.

5: **ef**($*$) – **complex array**

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

If **fact** = 'F', **ef** contains the $(n - 1)$ subdiagonal elements of the unit bidiagonal factor L from the LDL^{H} factorization of A.

6: **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 dimension of the array **d** The dimension of the array **df**.

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, ldx, work, rwork

5.4 Output Parameters

1: **df**($*$) – **double array**

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

If **fact** = 'N', **df** contains the n diagonal elements of the diagonal matrix D from the LDL^{H} factorization of A.

2: **ef**($*$) – **complex array**

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

If **fact** = 'N', **ef** contains the $(n - 1)$ subdiagonal elements of the unit bidiagonal factor L from the LDL^{H} factorization of A.

3: **x**(**ldx**,$*$) – **complex array**

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

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

If **info** $= 0$ or **info** $\geq N + 1$, the n by r solution matrix X.

4: **rcond** – **double scalar**

The reciprocal condition number of the matrix A. If **rcond** is less than the ***machine precision*** (in particular, if **rcond** $= 0$), the matrix is singular to working precision. This condition is indicated by a return code of **info** > 0.

5: **ferr**($*$) – **double array**

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

The forward error bound for each solution vector $\mathbf{x}(j)$ (the jth column of the solution matrix X). If x_j is the true solution corresponding to $\mathbf{x}(j)$, $\mathbf{ferr}(j)$ is an estimated upper bound for the magnitude of the largest element in $(\mathbf{x}(j) - x_j)$ divided by the magnitude of the largest element in $\mathbf{x}(j)$.

6: **berr**($*$) – **double array**

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

The component-wise relative backward error of each solution vector $\mathbf{x}(j)$ (i.e., the smallest relative change in any element of A or B that makes $\mathbf{x}(j)$ an exact solution).

7: **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: **fact**, 2: **n**, 3: **nrhs_p**, 4: **d**, 5: **e**, 6: **df**, 7: **ef**, 8: **b**, 9: **ldb**, 10: **x**, 11: **ldx**, 12: **rcond**, 13: **ferr**, 14: **berr**, 15: **work**, 16: **rwork**, 17: **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 and **info** $\leq N$

If **info** $= i$ and $i \leq$ **n**, the leading minor of order i of A is not positive-definite, so the factorization could not be completed, and the solution has not been computed. **rcond** $= 0$ is returned.

info $= N + 1$

U is nonsingular, but **rcond** is less than ***machine precision***, meaning that the matrix is singular to working precision. Nevertheless, the solution and error bounds are computed because there are a number of situations where the computed solution can be more accurate than the value of **rcond** would suggest.

7 Accuracy

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

$$|E| \leq c(n)\epsilon\left|R^{\mathrm{T}}\right||R|, \text{ where } R = D^{\frac{1}{2}}U,$$

$c(n)$ is a modest linear function of n, and ϵ is the ***machine precision***. See Section 10.1 of Higham 2002 for further details.

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

$$\frac{\|x - \hat{x}\|_\infty}{\|\hat{x}\|_\infty} \leq w_c \operatorname{cond}(A, \hat{x}, b)$$

where $\operatorname{cond}(A, \hat{x}, b) = \left\|\left|A^{-1}\right|(|A||\hat{x}| + |b|)\right\|_\infty / \|\hat{x}\|_\infty \leq \operatorname{cond}(A) = \left\|\left|A^{-1}\right||A|\right\|_\infty \leq \kappa_\infty(A)$. If $\hat{x}$ is the jth column of X, then w_c is returned in **berr**(j) and a bound on $\|x - \hat{x}\|_\infty / \|\hat{x}\|_\infty$ is returned in **ferr**(j). See Section 4.4 of Anderson *et al.* 1999 for further details.

8 Further Comments

The number of floating-point operations required for the factorization, and for the estimation of the condition number of A is proportional to n. The number of floating-point operations required for the solution of the equations, and for the estimation of the forward and backward error is proportional to nr, where r is the number of right-hand sides.

The condition estimation is based upon Equation (15.11) of Higham 2002. For further details of the error estimation, see Section 4.4 of Anderson *et al.* 1999.

The real analogue of this function is f07jb.

9 Example

```
fact = 'Not factored';
d = [16;
     41;
     46;
     21];
e = [complex(16, +16);
     complex(18, -9);
     complex(1, -4)];
df = zeros(4, 1);
ef = complex(zeros(3, 1));
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)];
[dfOut, efOut, x, rcond, ferr, berr, info] = f07jp(fact, d, e, df, ef, b)
```

```
dfOut =
```

```
    16
     9
     1
     4
efOut =
   1.0000 + 1.0000i
   2.0000 - 1.0000i
   1.0000 - 4.0000i
x =
   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
rcond =
   1.0862e-04
ferr =
   1.0e-11 *
    0.9038
    0.6093
berr =
     0
     0
info =
           0
```