NAG Toolbox for MATLAB

f04cg

1 Purpose

f04cg 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. An estimate of the condition number of A and an error bound for the computed solution are also returned.

2 Syntax

```
[d, e, b, rcond, errbnd, ifail] = f04cg(d, e, b, 'n', n, 'nrhs_p',
nrhs_p)
```

3 Description

A is factorized as $A = LDL^{\mathrm{H}}$, where L is a unit lower bidiagonal matrix and D is a real diagonal matrix, and the factored form of A is then used to solve the system of 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

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

5 Parameters

5.1 Compulsory Input Parameters

1: **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 tridiagonal matrix A.

2: **e**$(*)$ **– complex array**

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

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

3: **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 $\max(1, \mathbf{nrhs\_p})$. To solve the equations $Ax = b$, where b is a single right-hand side, **b** may be supplied as a one-dimensional array with length $\mathbf{ldb} = \max(1, \mathbf{n})$

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 number of linear equations n, i.e., the order of the matrix A.

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

2: **nrhs_p – int32 scalar**

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

The number of right-hand sides r, 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: **d**$(*)$ **– double array**

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

If **ifail** $= 0$ or $Np1$, **d** contains the n diagonal elements of the diagonal matrix D from the LDL^{H} factorization of A.

2: **e**$(*)$ **– complex array**

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

If **ifail** $= 0$ or $Np1$, **e** contains the $(n-1)$ subdiagonal elements of the unit lower bidiagonal matrix L from the LDL^{H} factorization of A. (**e** can also be regarded as the conjugate of the superdiagonal of the unit upper bidiagonal factor U from the $U^{\mathrm{H}}DU$ factorization of A.)

3: **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 $\max(1, \mathbf{nrhs\_p})$. To solve the equations $Ax = b$, where b is a single right-hand side, **b** may be supplied as a one-dimensional array with length $\mathbf{ldb} = \max(1, \mathbf{n})$

If **ifail** $= 0$ or $Np1$, the n by r solution matrix X.

4: **rcond – double scalar**

If **ifail** $= 0$ or $Np1$, an estimate of the reciprocal of the condition number of the matrix A, computed as $\mathbf{rcond} = 1/\left(\|A\|_1, \|A^{-1}\|_1\right)$.

5: **errbnd – double scalar**

If **ifail** $= 0$ or $Np1$, an estimate of the forward error bound for a computed solution $\hat{x}$, such that $\|\hat{x} - x\|_1 / \|x\|_1 \leq \mathbf{errbnd}$, where $\hat{x}$ is a column of the computed solution returned in the array **b** and x is the corresponding column of the exact solution X. If **rcond** is less than ***machine precision***, then **errbnd** is returned as unity.

6: **ifail – int32 scalar**

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

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail < 0 and **ifail** $\neq -999$

If **ifail** $= -i$, the ith argument had an illegal value.

ifail $= -999$

Allocation of memory failed. The double allocatable memory required is **n**. In this case the factorization and the solution X have been computed, but **rcond** and **errbnd** have not been computed.

ifail > 0 and **ifail** $\leq N$

If **ifail** $= i$, the leading minor of order i of A is not positive-definite. The factorization could not be completed, and the solution has not been computed.

ifail $= N + 1$

rcond is less than ***machine precision***, so that the matrix A is numerically singular. A solution to the equations $AX = B$ has nevertheless been computed.

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. f04cg uses the approximation $\|E\|_1 = \epsilon\|A\|_1$ to estimate **errbnd**. See Section 4.4 of Anderson *et al.* 1999 for further details.

8 Further Comments

The total number of floating-point operations required to solve the equations $AX = B$ is proportional to nr. The condition number estimation requires $O(n)$ floating-point operations.

See Section 15.3 of Higham 2002 for further details on computing the condition number of tridiagonal matrices.

The real analogue of f04cg is f04bg.

9 Example

```
d = [16;
     41;
     46;
     21];
e = [complex(16, +16);
     complex(18, -9);
     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)];
[dOut, eOut, bOut, rcond, errbnd, ifail] = f04cg(d, e, b)
```

```
dOut =
    16
     9
     1
     4
eOut =
   1.0000 + 1.0000i
   2.0000 - 1.0000i
   1.0000 - 4.0000i
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
rcond =
   1.0862e-04
errbnd =
   1.0231e-12
ifail =
           0
```