NAG Toolbox for MATLAB

f08ng

1 Purpose

f08ng multiplies an arbitrary real matrix C by the real orthogonal matrix Q which was determined by f08ne when reducing a real general matrix to Hessenberg form.

2 Syntax

```
[c, info] = f08ng(side, trans, ilo, ihi, a, tau, c, 'm', m, 'n', n)
```

3 Description

f08ng is intended to be used following a call to f08ne, which reduces a real general matrix A to upper Hessenberg form H by an orthogonal similarity transformation: $A = QHQ^{\mathrm{T}}$. f08ne represents the matrix Q as a product of $i_{\mathrm{hi}} - i_{\mathrm{lo}}$ elementary reflectors. Here i_{lo} and i_{hi} are values determined by f08nh when balancing the matrix; if the matrix has not been balanced, $i_{\mathrm{lo}} = 1$ and $i_{\mathrm{hi}} = n$.

This function may be used to form one of the matrix products

$$QC, Q^{\mathrm{T}}C, CQ \text{ or } CQ^{\mathrm{T}},$$

overwriting the result on C (which may be any real rectangular matrix).

A common application of this function is to transform a matrix V of eigenvectors of H to the matrix QV of eigenvectors of A.

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: **side – string**

Indicates how Q or Q^{T} is to be applied to C.

side = 'L'

Q or Q^{T} is applied to C from the left.

side = 'R'

Q or Q^{T} is applied to C from the right.

Constraint: **side** = 'L' or 'R'.

2: **trans – string**

Indicates whether Q or Q^{T} is to be applied to C.

trans = 'N'

Q is applied to C.

trans = 'T'

Q^{T} is applied to C.

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

3: **ilo – int32 scalar**

4: **ihi – int32 scalar**

These **must** be the same parameters **ilo** and **ihi**, respectively, as supplied to f08ne.

Constraints:

if **side** = 'L' and $\mathbf{m} > 0$, $1 \leq \mathbf{ilo} \leq \mathbf{ihi} \leq \mathbf{m}$;
if **side** = 'L' and $\mathbf{m} = 0$, $\mathbf{ilo} = 1$ and $\mathbf{ihi} = 0$;
if **side** = 'R' and $\mathbf{n} > 0$, $1 \leq \mathbf{ilo} \leq \mathbf{ihi} \leq \mathbf{n}$;
if **side** = 'R' and $\mathbf{n} = 0$, $\mathbf{ilo} = 1$ and $\mathbf{ihi} = 0$.

5: **a(lda,∗) – double array**

The first dimension, **lda**, of the array **a** must satisfy

if **side** = 'L', $\mathbf{lda} \geq \max(1, \mathbf{m})$;
if **side** = 'R', $\mathbf{lda} \geq \max(1, \mathbf{n})$.

The second dimension of the array must be at least $\max(1, \mathbf{m})$ if **side** = 'L' and at least $\max(1, \mathbf{n})$ if **side** = 'R'

Details of the vectors which define the elementary reflectors, as returned by f08ne.

6: **tau(∗) – double array**

Note: the dimension of the array **tau** must be at least $\max(1, \mathbf{m} - 1)$ if **side** = 'L' and at least $\max(1, \mathbf{n} - 1)$ if **side** = 'R'.

Further details of the elementary reflectors, as returned by f08ne.

7: **c(ldc,∗) – double array**

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

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

The m by n matrix C.

5.2 Optional Input Parameters

1: **m – int32 scalar**

Default: The first dimension of the array **c**.

m, the number of rows of the matrix C; m is also the order of Q if **side** = 'L'.

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

2: **n – int32 scalar**

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

n, the number of columns of the matrix C; n is also the order of Q if **side** = 'R'.

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

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldc, work, lwork

5.4 Output Parameters

1: **c**(**ldc**,∗) **– double array**

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

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

c contains QC or $Q^{\mathrm{T}}C$ or CQ or CQ^{T} as specified by **side** and **trans**.

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: **side**, 2: **trans**, 3: **m**, 4: **n**, 5: **ilo**, 6: **ihi**, 7: **a**, 8: **lda**, 9: **tau**, 10: **c**, 11: **ldc**, 12: **work**, 13: **lwork**, 14: **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 result differs from the exact result by a matrix E such that

$$\|E\|_2 = O(\epsilon)\|C\|_2,$$

where ϵ is the ***machine precision***.

8 Further Comments

The total number of floating-point operations is approximately $2nq^2$ if **side** = 'L' and $2mq^2$ if **side** = 'R', where $q = i_{\mathrm{hi}} - i_{\mathrm{lo}}$.

The complex analogue of this function is f08nu.

9 Example

```
side = 'Left';
trans = 'No transpose';
ilo = int32(1);
ihi = int32(4);
a = [0.35, -0.1159524296205035, -0.3886010343233214, -0.2941840753473021;
          -0.5140038910358558,  0.1224867524602574,  0.1003597896821503,
0.1125618799705319;
        -0.7284721282927631,  0.6442636185270625,  -0.1357001717571131,  -
0.09768162270493326;
          0.4139046183481608,  -0.1665445794905699,  0.4262443722078449,
0.1632134192968561];
tau = [1.175095950769217;
     1.946022972139863;
```

```
     0];
c = [0.388121783771964, 0.05736203016960297, 0.1493218416107921;
     -0.01428690112019408, 0.4135233160565419, 0.1179454267523731;
     0.8119679816025415, -1.642115525248917e-17, -0.6190595172579693;
     -0.3958676674181024, -0.6041323325818976, 1];
[cOut, info] = f08ng(side, trans, ilo, ihi, a, tau, c)
```

```
cOut =
    0.3881    0.0574    0.1493
   -0.7107    0.0380    0.3956
   -0.3891    0.0778    0.7075
   -0.3996   -0.7270    0.8603
info =
           0
```