NAG Toolbox for MATLAB

f08kg

1 Purpose

f08kg multiplies an arbitrary real m by n matrix C by one of the real orthogonal matrices Q or P which were determined by f08ke when reducing a real matrix to bidiagonal form.

2 Syntax

```
[c, info] = f08kg(vect, side, trans, k, a, tau, c, 'm', m, 'n', n)
```

3 Description

f08kg is intended to be used after a call to f08ke, which reduces a real rectangular matrix A to bidiagonal form B by an orthogonal transformation: $A = QBP^{\mathrm{T}}$. f08ke represents the matrices Q and P^{T} as products of elementary reflectors.

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

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

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

4 References

Golub G H and Van Loan C F 1996 *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

Note: in the descriptions below, r denotes the order of Q or P^{T}: if **side** = 'L', $r =$ **m** and if **side** = 'R', $r =$ **n**.

5.1 Compulsory Input Parameters

1: **vect – string**

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

vect = 'Q'

Q or Q^{T} is applied to C.

vect = 'P'

P or P^{T} is applied to C.

Constraint: **vect** = 'Q' or 'P'.

2: **side – string**

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

side = 'L'

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

side = 'R'

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

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

3: **trans – string**

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

trans = 'N'

Q or P is applied to C.

trans = 'T'

Q^{T} or P^{T} is applied to C.

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

4: **k – int32 scalar**

If **vect** = 'Q', the number of columns in the original matrix A.

If **vect** = 'P', the number of rows in the original matrix A.

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

5: **a(lda,∗) – double array**

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

if **vect** = 'Q', $\mathbf{lda} \geq \max(1, r)$;
if **vect** = 'P', $\mathbf{lda} \geq \max(1, \min(r, \mathbf{k}))$.

The second dimension of the array must be at least $\max(1, \min(r, \mathbf{k}))$ if **vect** = 'Q' and at least $\max(1, r)$ if **vect** = 'P'

Details of the vectors which define the elementary reflectors, as returned by f08ke.

6: **tau(∗) – double array**

Note: the dimension of the array **tau** must be at least $\max(1, \min(r, \mathbf{k}))$.

Further details of the elementary reflectors, as returned by f08ke in its parameter **tauq** if **vect** = 'Q', or in its parameter **taup** if **vect** = 'P'.

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 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.

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.

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 $C^{\mathrm{T}}Q$ or PC or $P^{\mathrm{T}}C$ or CP or $C^{\mathrm{T}}P$ as specified by **vect**, **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: **vect**, 2: **side**, 3: **trans**, 4: **m**, 5: **n**, 6: **k**, 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

if **side** = 'L' and $m \geq k$, $2nk(2m - k)$;
if **side** = 'R' and $n \geq k$, $2mk(2n - k)$;
if **side** = 'L' and $m < k$, $2m^2n$;
if **side** = 'R' and $n < k$, $2mn^2$,

where k is the value of the parameter **k**.

The complex analogue of this function is f08ku.

9 Example

```
vect = 'Q';
side = 'Right';
trans = 'No transpose';
k = int32(4);
a   =   [3.61767881382524,   1.258711404710556,   -0.4667885339706592,   -
0.4109505449290541;
     0, -2.416055340423328, -1.526154889941117, -0.2094556505641229;
     0, 0.02040919619839952, 1.921310963709999, 1.189466677019391;
     0, -0.3216264186905484, 0.1342281666124217, -1.426501541730861];
tau = [0;
     1.811823817941776;
     1.964603347174235;
     0];
c   =   [-0.1575595925823212,   0.6743815182091041,   -0.4571499642152959,
0.4488516451198497;
          -0.5334912520769823,  -0.3861089893314064,  0.2582527260774866,
0.3898174378676559;
          0.6357667770865594,  -0.2928230899594127,  0.01653842638455846,
0.1929529167768124;
       -0.5334912520769823, -0.1691547055376643, -0.08342735451416805, -
0.2349873193632054;
          0.04146305067955822,  -0.1593027920340203,  0.1474783037934707,
0.7436407646419895;
         -0.005528406757274429,  -0.5063529996817608,  -0.83386806304906,
0.03351284255316388];
[cOut, info] = f08kg(vect, side, trans, k, a, tau, c)
```

```
cOut =
   -0.1576   -0.2690    0.2612    0.8513
   -0.5335    0.5311   -0.2922    0.0184
    0.6358    0.3495   -0.0250   -0.0210
   -0.5335    0.0035    0.1537   -0.2592
    0.0415    0.5572   -0.2917    0.4523
   -0.0055    0.4614    0.8585   -0.0532
info =
           0
```