NAG Toolbox for MATLAB

f08aw

1 Purpose

f08aw generates all or part of the complex unitary matrix Q from an LQ factorization computed by f08av.

2 Syntax

```
[a, info] = f08aw(a, tau, 'm', m, 'n', n, 'k', k)
```

3 Description

f08aw is intended to be used after a call to f08av, which performs an LQ factorization of a complex matrix A. The unitary matrix Q is represented as a product of elementary reflectors.

This function may be used to generate Q explicitly as a square matrix, or to form only its leading rows.

Usually Q is determined from the LQ factorization of a p by n matrix A with $p \leq n$. The whole of Q may be computed by:

```
[a, info] = f08aw(a, tau);
```

(note that the array **a** must have at least n rows) or its leading p rows by:

```
[a, info] = f08aw(a(1:p,:), tau);
```

The rows of Q returned by the last call form an orthonormal basis for the space spanned by the rows of A; thus f08av followed by f08aw can be used to orthogonalise the rows of A.

The information returned by the LQ factorization functions also yields the LQ factorization of the leading k rows of A, where $k < p$. The unitary matrix arising from this factorization can be computed by:

```
[a, info] = f08aw(a, tau);
```

or its leading k rows by:

```
[a, info] = f08aw(a(1:k,:), tau);
```

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: **a**(**lda,**$*$) **– complex array**

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

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

Details of the vectors which define the elementary reflectors, as returned by f08av.

2: **tau**$(*)$ **– complex array**

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

Further details of the elementary reflectors as returned by f08av.

5.2 Optional Input Parameters

1: **m – int32 scalar**

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

m, the number of rows of the matrix Q.

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

2: **n – int32 scalar**

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

n, the number of columns of the matrix Q.

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

3: **k – int32 scalar**

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

k, the number of elementary reflectors whose product defines the matrix Q.

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

5.3 Input Parameters Omitted from the MATLAB Interface

lda, work, lwork

5.4 Output Parameters

1: **a(lda,$*$) – complex array**

The first dimension of the array **a** 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 Q.

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: **m**, 2: **n**, 3: **k**, 4: **a**, 5: **lda**, 6: **tau**, 7: **work**, 8: **lwork**, 9: **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 matrix Q differs from an exactly unitary matrix by a matrix E such that

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

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

8 Further Comments

The total number of real floating-point operations is approximately $16mnk - 8(m+n)k^2 + \frac{16}{3}k^3$; when $m = k$, the number is approximately $\frac{8}{3}m^2(3n-m)$.

The real analogue of this function is f08aj.

9 Example

```
a  =  [complex(-2.225511177235469,  -0),  complex(0.2438442594612761,  -
0.3082069932914802), ...
                    complex(-0.2741364985191491,     -0.230966496866882),
complex(0.5807709514701289, +0.3468663602164287);
                     complex(0.8207552577961005,    +1.238457046719376),
complex(1.688100989346069, ...
             -0),    complex(-0.1936415258449461,   +0.5429517855236941),
complex(0.2789084851242076, -0.2203175797458332);
                   complex(0.001033470433007241,    -0.6822252862760427),
complex(0.774751330164006, ...
       -0.6154727158531147),  complex(-1.590258250459319,  -0),  complex(-
0.1267668516113225, +0.1109845202357172)];
tau = [complex(1.12581379184436, +0.1617605895141772);
     complex(1.099053669168954, +0.5468590598466785);
     complex(1.132925657311579, -0.9590540478961487)];
[aOut, info] = f08aw(a, tau)
```

```
aOut =
   -0.1258 + 0.1618i   -0.2247 + 0.3864i    0.3460 + 0.2157i   -0.7099 -
0.2966i
   -0.1163 - 0.6380i   -0.3240 + 0.4272i   -0.1995 - 0.5009i   -0.0323 -
0.0162i
   -0.4607 + 0.1090i    0.2171 - 0.4062i    0.2733 - 0.6106i   -0.0994 -
0.3261i
info =
           0
```