NAG Toolbox for MATLAB

f08kt

1 Purpose

f08kt generates one of the complex unitary matrices Q or P^{H} which were determined by f08ks when reducing a complex matrix to bidiagonal form.

2 Syntax

```
[a, info] = f08kt(vect, k, a, tau, 'm', m, 'n', n)
```

3 Description

f08kt is intended to be used after a call to f08ks, which reduces a complex rectangular matrix A to real bidiagonal form B by a unitary transformation: $A = QBP^{\mathrm{H}}$. f08ks represents the matrices Q and P^{H} as products of elementary reflectors.

This function may be used to generate Q or P^{H} explicitly as square matrices, or in some cases just the leading columns of Q or the leading rows of P^{H}.

The various possibilities are specified by the parameters **vect**, **m**, **n** and **k**. The appropriate values to cover the most likely cases are as follows (assuming that A was an m by n matrix):

1. To form the full m by m matrix Q:

   ```
   [a, info] = f08kt('Q', n, a, tau);
   ```

   (note that the array **a** must have at least m columns).

2. If $m > n$, to form the n leading columns of Q:

   ```
   [a, info] = f08kt('Q', n, a(1:m,1:n), tau);
   ```

3. To form the full n by n matrix P^{H}:

   ```
   [a, info] = f08kt('P', m, a, tau);
   ```

   (note that the array **a** must have at least n rows).

4. If $m < n$, to form the m leading rows of P^{H}:

   ```
   [a, info] = f08kt('P', m, a(1:m,1:n), 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: **vect – string**

Indicates whether the unitary matrix Q or P^{H} is generated.

vect = 'Q'

Q is generated.

vect = 'P'

P^{H} is generated.

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

2: **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: $\mathbf{k} \geq 0$.

3: **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 f08ks.

4: **tau(∗) – complex array**

Note: the dimension of the array **tau** must be at least $(1, \min(\mathbf{m}, \mathbf{k}))$ if **vect** = 'Q' and at least $(1, \min(\mathbf{n}, \mathbf{k}))$ if **vect** = 'P'.

Further details of the elementary reflectors, as returned by f08ks in its parameter **tauq** if **vect** = 'Q', or in its parameter **taup** if **vect** = 'P'.

5.2 Optional Input Parameters

1: **m – int32 scalar**

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

m, the number of rows of the unitary matrix Q or P^{H} to be returned.

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

2: **n – int32 scalar**

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

n, the number of columns of the unitary matrix Q or P^{H} to be returned.

Constraints:

$\mathbf{n} \geq 0$;
if **vect** = 'Q' and $\mathbf{m} > \mathbf{k}$, $\mathbf{m} \geq \mathbf{n} \geq \mathbf{k}$;
if **vect** = 'Q' and $\mathbf{m} \leq \mathbf{k}$, $\mathbf{m} = \mathbf{n}$;
if **vect** = 'P' and $\mathbf{n} > \mathbf{k}$, $\mathbf{n} \geq \mathbf{m} \geq \mathbf{k}$;
if **vect** = 'P' and $\mathbf{n} \leq \mathbf{k}$, $\mathbf{n} = \mathbf{m}$.

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 unitary matrix Q or P^{H}, or the leading rows or columns thereof, as specified by **vect**, **m** and **n**.

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: **m**, 3: **n**, 4: **k**, 5: **a**, 6: **lda**, 7: **tau**, 8: **work**, 9: **lwork**, 10: **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***. A similar statement holds for the computed matrix P^{H}.

8 Further Comments

The total number of real floating-point operations for the cases listed in Section 3 are approximately as follows:

1. To form the whole of Q:

   $\frac{16}{3}n\left(3m^2 - 3mn + n^2\right)$ if $m > n$,

   $\frac{16}{3}m^3$ if $m \leq n$;

2. To form the n leading columns of Q when $m > n$:

   $\frac{8}{3}n^2(3m - n)$;

3. To form the whole of P^{H}:

   $\frac{16}{3}n^3$ if $m \geq n$,

   $\frac{16}{3}m^3\left(3n^2 - 3mn + m^2\right)$ if $m < n$;

4. To form the m leading rows of P^{H} when $m < n$:

   $\frac{8}{3}m^2(3n - m)$.

The real analogue of this function is f08kf.

9 Example

```
vect = 'P';
k = int32(6);
a  =  [complex(-3.087005021051958,  +0),  complex(2.112571007455839,  +0),
complex(0.05433411079440312, ...
```

```
                      +0.4543118496773522),     complex(0.375743827925403,
+0.1070087304094524);
                   complex(0,    +0),    complex(2.066039276679068,    +0),
complex(1.262810106655224, ...
     +0), complex(0.02827717828732752, +0.165005610304937);
        complex(0,  +0),  complex(0,  +0),  complex(1.873128891125711,  +0),
complex(-1.612633872800393, +0);
                complex(0,   +0),   complex(0,   +0),   complex(0,   +0),
complex(2.002182866206992, +0)];
tau = [complex(1.231234531617602, -0.5404263814542949);
     complex(1.262279868392228, -0.928646196138089);
     complex(1.782899043025449, -0.6221487671207239);
     complex(0, +0)];
[aOut, info] = f08kt(vect, k, a, tau)
```

```
aOut =
   1.0000                 0                   0                  0
        0            -0.2312 - 0.5404i    0.1786 - 0.5887i   -0.4048 -
0.3348i
        0             0.4962 + 0.2648i   -0.3556 - 0.6888i    0.2757 -
0.0819i
        0             0.2918 + 0.5030i    0.0485 + 0.1349i   -0.7156 -
0.3596i
info =
           0
```