NAG Toolbox

nag_lapack_zungqr (f08at)

1 Purpose

nag_lapack_zungqr (f08at) generates all or part of the complex unitary matrix Q from a QR factorization computed by nag_lapack_zgeqrf (f08as), nag_lapack_zgeqpf (f08bs) or nag_lapack_zgeqp3 (f08bt).

2 Syntax

```
[a, info] = nag_lapack_zungqr(a, tau, 'm', m, 'n', n, 'k', k)
[a, info] = f08at(a, tau, 'm', m, 'n', n, 'k', k)
```

3 Description

nag_lapack_zungqr (f08at) is intended to be used after a call to nag_lapack_zgeqrf (f08as), nag_lapack_zgeqpf (f08bs) or nag_lapack_zgeqp3 (f08bt), which perform a QR 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 columns.

Usually Q is determined from the QR factorization of an m by p matrix A with $m \geq p$. The whole of Q may be computed by:

```
[a, info] = f08at(a, tau);
```

(note that the array **a** must have m columns) or its leading p columns by:

```
[a, info] = f08at(a(1:p,:), tau);
```

The columns of Q returned by the last call form an orthonormal basis for the space spanned by the columns of A; thus nag_lapack_zgeqrf (f08as) followed by nag_lapack_zungqr (f08at) can be used to orthogonalize the columns of A.

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

```
[a, info] = f08at(a, tau);
```

or its leading k columns by:

```
[a, info] = f08at(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 (KIND=nag_wp) array

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

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

Details of the vectors which define the elementary reflectors, as returned by nag_lapack_zgeqrf (f08as), nag_lapack_zgeqpf (f08bs) or nag_lapack_zgeqp3 (f08bt).

2: **tau**(:) – COMPLEX (KIND=nag_wp) array

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

Further details of the elementary reflectors, as returned by nag_lapack_zgeqrf (f08as), nag_lapack_zgeqpf (f08bs) or nag_lapack_zgeqp3 (f08bt).

5.2 Optional Input Parameters

1: **m** – INTEGER

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

m, the order of the unitary matrix Q.

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

2: **n** – INTEGER

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

n, the number of columns of the matrix Q.

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

3: **k** – INTEGER

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

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

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

5.3 Output Parameters

1: **a**$(lda, :)$ – COMPLEX (KIND=nag_wp) array

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

The second dimension of the array **a** will be $\max(1, \mathbf{n})$.

The m by n matrix Q.

2: **info** – INTEGER

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

6 Error Indicators and Warnings

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 $n = k$, the number is approximately $\frac{8}{3}n^2(3m-n)$.

The real analogue of this function is nag_lapack_dorgqr (f08af).

9 Example

This example forms the leading 4 columns of the unitary matrix Q from the QR factorization of the matrix A, where

$$A = \begin{pmatrix} 0.96 - 0.81i & -0.03 + 0.96i & -0.91 + 2.06i & -0.05 + 0.41i \\ -0.98 + 1.98i & -1.20 + 0.19i & -0.66 + 0.42i & -0.81 + 0.56i \\ 0.62 - 0.46i & 1.01 + 0.02i & 0.63 - 0.17i & -1.11 + 0.60i \\ -0.37 + 0.38i & 0.19 - 0.54i & -0.98 - 0.36i & 0.22 - 0.20i \\ 0.83 + 0.51i & 0.20 + 0.01i & -0.17 - 0.46i & 1.47 + 1.59i \\ 1.08 - 0.28i & 0.20 - 0.12i & -0.07 + 1.23i & 0.26 + 0.26i \end{pmatrix}.$$

The columns of Q form an orthonormal basis for the space spanned by the columns of A.

9.1 Program Text

```
     function f08at_example

fprintf('f08at example results\n\n');

a = [ 0.96 - 0.81i, -0.03 + 0.96i, -0.91 + 2.06i, -0.05 + 0.41i;
     -0.98 + 1.98i, -1.20 + 0.19i, -0.66 + 0.42i, -0.81 + 0.56i;
      0.62 - 0.46i,  1.01 + 0.02i,  0.63 - 0.17i, -1.11 + 0.60i;
     -0.37 + 0.38i,  0.19 - 0.54i, -0.98 - 0.36i,  0.22 - 0.20i;
      0.83 + 0.51i,  0.20 + 0.01i, -0.17 - 0.46i,  1.47 + 1.59i;
      1.08 - 0.28i,  0.20 - 0.12i, -0.07 + 1.23i,  0.26 + 0.26i];
[m,n] = size(a);

% Compute the QR factorization of A
[qr, tau, info] = f08as(a);

% Form Q
[Q, info] = f08at(qr, tau);

mtitle = sprintf('The leading %2d columns of Q\n',n);
disp(mtitle);
disp(Q(:,1:n));
```

9.2 Program Results

```
     f08at example results

The leading  4 columns of Q

  -0.3110 + 0.2624i  -0.3175 + 0.4835i   0.4966 - 0.2997i  -0.0072 - 0.3718i
   0.3175 - 0.6414i  -0.2062 + 0.1577i  -0.0793 - 0.3094i  -0.0282 - 0.1491i
  -0.2008 + 0.1490i   0.4892 - 0.0900i   0.0357 - 0.0219i   0.5625 - 0.0710i
   0.1199 - 0.1231i   0.2566 - 0.3055i   0.4489 - 0.2141i  -0.1651 + 0.1800i
  -0.2689 - 0.1652i   0.1697 - 0.2491i  -0.0496 + 0.1158i  -0.4885 - 0.4540i
  -0.3499 + 0.0907i  -0.0491 - 0.3133i  -0.1256 - 0.5300i   0.1039 + 0.0450i
```