NAG Toolbox for MATLAB

f08zf

1 Purpose

f08zf computes a generalized RQ factorization of a real matrix pair (A, B), where A is an m by n matrix and B is a p by n matrix.

2 Syntax

```
[a, taua, b, taub, info] = f08zf(a, b, 'm', m, 'p', p, 'n', n)
```

3 Description

f08zf forms the generalized RQ factorization of an m by n matrix A and a p by n matrix B

$$A = RQ, \qquad B = ZTQ,$$

where Q is an n by n orthogonal matrix, Z is a p by p orthogonal matrix and R and T are of the form

$$R = \begin{cases} \begin{array}{cc} & \begin{array}{cc} n-m & m \end{array} \\ m & \begin{pmatrix} 0 & R_{12} \end{pmatrix} \end{array}; & \text{if } m \leq n, \\[2ex] \begin{array}{cc} & n \\ \begin{array}{c} m-n \\ n \end{array} & \begin{pmatrix} R_{11} \\ R_{21} \end{pmatrix} \end{array}; & \text{if } m > n, \end{cases}$$

with R_{12} or R_{21} upper triangular,

$$T = \begin{cases} \begin{array}{cc} & n \\ \begin{array}{c} n \\ p-n \end{array} & \begin{pmatrix} T_{11} \\ 0 \end{pmatrix} \end{array}; & \text{if } p \geq n, \\[2ex] \begin{array}{cc} & \begin{array}{cc} p & n-p \end{array} \\ p & \begin{pmatrix} T_{11} & T_{12} \end{pmatrix} \end{array}; & \text{if } p < n, \end{cases}$$

with T_{11} upper triangular.

In particular, if B is square and nonsingular, the generalized RQ factorization of A and B implicitly gives the RQ factorization of AB^{-1} as

$$AB^{-1} = \left(RT^{-1}\right)Z^{\mathrm{T}}.$$

4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D 1999 *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia URL: http://www.netlib.org/lapack/lug

Anderson E, Bai Z and Dongarra J 1992 Generalized QR factorization and its applications *Linear Algebra Appl. (Volume 162–164)* 243–271

Hammarling S 1987 The numerical solution of the general Gauss-Markov linear model *Mathematics in Signal Processing* (ed T S Durrani, J B Abbiss, J E Hudson, R N Madan, J G McWhirter, and T A Moore) 441–456 Oxford University Press

Paige C C 1990 Some aspects of generalized QR factorizations . *In Reliable Numerical Computation* (ed M G Cox and S Hammarling) 73–91 Oxford University Press

5 Parameters

5.1 Compulsory Input Parameters

1: **a(lda,$*$) – double 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 A.

2: **b(ldb,$*$) – double array**

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

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

The p by n matrix B.

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 A.

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

2: **p – int32 scalar**

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

p, the number of rows of the matrix B.

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

3: **n – int32 scalar**

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

n, the number of columns of the matrices A and B.

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

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldb, work, lwork

5.4 Output Parameters

1: **a(lda,$*$) – double 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 $\max(1, \mathbf{n})$

If $m \leq n$, the upper triangle of the subarray $\mathbf{a}(1 : m, n - m + 1 : n)$ contains the m by m upper triangular matrix R_{12}.

If $m \geq n$, the elements on and above the $(m - n)$th subdiagonal contain the m by n upper trapezoidal matrix R; the remaining elements, with the array **taua**, represent the orthogonal matrix Q as a product of $\min(m, n)$ elementary reflectors (see Section 3.2.6 in the F08 Chapter Introduction).

2: **taua**($*$) – **double array**

Note: the dimension of the array **taua** must be at least $\max(1, \min(\mathbf{m}, \mathbf{n}))$.

The scalar factors of the elementary reflectors which represent the orthogonal matrix Q.

3: **b**(**ldb**,$*$) – **double array**

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

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

The elements on and above the diagonal of the array contain the $\min(p, n)$ by n upper trapezoidal matrix T (T is upper triangular if $p \geq n$); the elements below the diagonal, with the array **taub**, represent the orthogonal matrix Z as a product of elementary reflectors (see Section 3.2.6 in the F08 Chapter Introduction).

4: **taub**($*$) – **double array**

Note: the dimension of the array **taub** must be at least $\max(1, \min(\mathbf{p}, \mathbf{n}))$.

The scalar factors of the elementary reflectors which represent the orthogonal matrix Z.

5: **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: **p**, 3: **n**, 4: **a**, 5: **lda**, 6: **taua**, 7: **b**, 8: **ldb**, 9: **taub**, 10: **work**, 11: **lwork**, 12: **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 generalized RQ factorization is the exact factorization for nearby matrices $(A + E)$ and $(B + F)$, where

$$\|E\|_2 = O\epsilon\|A\|_2 \qquad \text{and} \qquad \|F\|_2 = O\epsilon\|B\|_2,$$

and ϵ is the ***machine precision***.

8 Further Comments

The orthogonal matrices Q and Z may be formed explicitly by calls to f08cj and f08af respectively. f08ck may be used to multiply Q by another matrix and f08ag may be used to multiply Z by another matrix.

The complex analogue of this function is f08zt.

9 Example

```
a = [1, 0, -1, 0;
     0, 1, 0, -1];
b = [-0.57, -1.28, -0.39, 0.25;
     -1.93, 1.08, -0.31, -2.14;
```

```
     2.3, 0.24, 0.4, -0.35;
     -1.93, 0.64, -0.66, 0.08;
     0.15, 0.3, 0.15, -2.13;
     -0.02, 1.03, -1.43, 0.5];
[aOut, taua, bOut, taub, info] = f08zf(a, b)
```

```
aOut =
   -0.4142         0    1.4142         0
         0   -0.4142         0    1.4142
taua =
    1.7071
    1.7071
bOut =
    3.3264   -0.2354    1.5296   -0.8479
    0.3955    2.0364    0.8978   -1.3588
   -0.4767    0.1207   -1.3372   -0.4674
    0.4573   -0.2835   -0.2594   -2.6447
   -0.0530    0.5095    0.3009   -0.0678
    0.2560   -0.4662    0.3760    0.4784
taub =
    1.2041
    1.2723
    1.5394
    1.6215
info =
           0
```