NAG Toolbox

nag_lapack_dtzrzf (f08bh)

1 Purpose

nag_lapack_dtzrzf (f08bh) reduces the m by n ($m \leq n$) real upper trapezoidal matrix A to upper triangular form by means of orthogonal transformations.

2 Syntax

```
[a, tau, info] = nag_lapack_dtzrzf(a, 'm', m, 'n', n)
[a, tau, info] = f08bh(a, 'm', m, 'n', n)
```

3 Description

The m by n ($m \leq n$) real upper trapezoidal matrix A given by

$$A = \begin{pmatrix} R_1 & R_2 \end{pmatrix},$$

where R_1 is an m by m upper triangular matrix and R_2 is an m by $(n-m)$ matrix, is factorized as

$$A = \begin{pmatrix} R & 0 \end{pmatrix} Z,$$

where R is also an m by m upper triangular matrix and Z is an n by n orthogonal matrix.

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 http://www.netlib.org/lapack/lug

5 Parameters

5.1 Compulsory Input Parameters

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

The first dimension of the array **a** must be at least max(1, **m**).

The second dimension of the array **a** must be at least max(1, **n**).

The leading m by n upper trapezoidal part of the array **a** must contain the matrix to be factorized.

5.2 Optional Input Parameters

1: **m** – INTEGER

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

m, the number of rows of the matrix A.

Constraint: **m** ≥ 0.

2: **n** – INTEGER

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

n, the number of columns of the matrix A.

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

5.3 Output Parameters

1: **a**$(lda, :)$ – REAL (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 leading m by m upper triangular part of **a** contains the upper triangular matrix R, and elements $\mathbf{m}+1$ to **n** of the first m rows of **a**, with the array **tau**, represent the orthogonal matrix Z as a product of m elementary reflectors (see Section 3.2.6 in the F08 Chapter Introduction).

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

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

The scalar factors of the elementary reflectors.

3: **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: **a**, 4: **lda**, 5: **tau**, 6: **work**, 7: **lwork**, 8: **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 factorization is the exact factorization of a nearby matrix $A + E$, where

$$\|E\|_2 = O\epsilon\|A\|_2$$

and ϵ is the ***machine precision***.

8 Further Comments

The total number of floating-point operations is approximately $4m^2(n-m)$.

The complex analogue of this function is nag_lapack_ztzrzf (f08bv).

9 Example

This example solves the linear least squares problems

$$\min_x \left\|b_j - Ax_j\right\|_2, \quad j = 1, 2$$

for the minimum norm solutions x_1 and x_2, where b_j is the jth column of the matrix B,

$$A = \begin{pmatrix} -0.09 & 0.14 & -0.46 & 0.68 & 1.29 \\ -1.56 & 0.20 & 0.29 & 1.09 & 0.51 \\ -1.48 & -0.43 & 0.89 & -0.71 & -0.96 \\ -1.09 & 0.84 & 0.77 & 2.11 & -1.27 \\ 0.08 & 0.55 & -1.13 & 0.14 & 1.74 \\ -1.59 & -0.72 & 1.06 & 1.24 & 0.34 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 7.4 & 2.7 \\ 4.2 & -3.0 \\ -8.3 & -9.6 \\ 1.8 & 1.1 \\ 8.6 & 4.0 \\ 2.1 & -5.7 \end{pmatrix}.$$

The solution is obtained by first obtaining a QR factorization with column pivoting of the matrix A, and then the RZ factorization of the leading k by k part of R is computed, where k is the estimated rank of A. A tolerance of 0.01 is used to estimate the rank of A from the upper triangular factor, R.

Note that the block size (NB) of 64 assumed in this example is not realistic for such a small problem, but should be suitable for large problems.

9.1 Program Text

```
     function f08bh_example

fprintf('f08bh example results\n\n');

% Going to solve Least squares problem Ax = b, m>n.
m = 6;
n = 5;
a = [-0.09,  0.14, -0.46,  0.68,  1.29;
     -1.56,  0.2,   0.29,  1.09,  0.51;
     -1.48, -0.43,  0.89, -0.71, -0.96;
     -1.09,  0.84,  0.77,  2.11, -1.27;
      0.08,  0.55, -1.13,  0.14,  1.74;
     -1.59, -0.72,  1.06,  1.24,  0.34];

b = [ 7.4,  2.7;
      4.2, -3.0;
     -8.3, -9.6;
      1.8,  1.1;
      8.6,  4.0;
      2.1, -5.7];

% QR factorization of A with column pivoting = Q*(R1 R2 )*(P^T)
%                                              (0  R22)

[qr, jpvt, tau, info] = f08bf( ...
        a, zeros(n,1,nag_int_name));

% QRP'X = B, => RP'X = Q'B = C
% Compute C = Q'B
[c, info] = f08ag( ...
    'Left', 'Transpose', qr, tau, b);

% Determine the rank, k, of R relative to tol;
% Choose tol to reflect the relative accuracy of the input data
tol = 0.01;
k = find(abs(diag(qr)) <= tol*abs(qr(1,1)));
if numel(k) == 0
  k = numel(diag(qr));
else
  k = k(1)-1;
end

fprintf('Tolerance used to estimate the rank of A\n     %11.2e\n', tol);
fprintf('Estimated rank of A\n        %d\n', k);

% Compute the RZ (TZ) factorization of the first k rows of (R1 R2)
[rz, taurz, info] = f08bh( ...
    qr(1:k,:));

% Now, (TZ)P'X = C on first k rows of C
% Let ZP'X = T^{-1}C = Y (on first k rows)
```

```
y = zeros(n, 2);
y(1:k, :) = inv(triu(rz(1:k,1:k)))*c(1:k,:);

% ZP'X = Y => P'X = Z^T Y = W; Form W = Z^TY.
[w, info] = f08bk( ...
    'Left', 'Transpose', nag_int(n-k), rz, taurz, y);

% P'X = W => X = PW, '
x = zeros(n, 2);
for i=1:n
   x(jpvt(i), :) = w(i, :);
end
fprintf('\nLeast-squares solution(s)\n');
disp(x);

% Compute estimates of the square roots of the residual sums of squares
rnorm = [norm(c(k+1:6,1)), norm(c(k+1:6,2))];
fprintf('Square root(s) of the residual sum(s) of squares\n');
disp(rnorm);
```

9.2 Program Results

```
     f08bh example results

Tolerance used to estimate the rank of A
        1.00e-02
Estimated rank of A
        4

Least-squares solution(s)
    0.6344    3.6258
    0.9699    1.8284
   -1.4402   -1.6416
    3.3678    2.4307
    3.3992    0.2818

Square root(s) of the residual sum(s) of squares
    0.0254    0.0365
```