NAG Toolbox for MATLAB

f08we

1 Purpose

f08we reduces a pair of real matrices (A, B), where B is upper triangular, to the generalized upper Hessenberg form using orthogonal transformations.

2 Syntax

```
[a, b, q, z, info] = f08we(compq, compz, ilo, ihi, a, b, q, z, 'n', n)
```

3 Description

f08we is the third step in the solution of the real generalized eigenvalue problem

$$Ax = \lambda Bx.$$

The (optional) first step balances the two matrices using f08wh. In the second step, matrix B is reduced to upper triangular form using the QR factorization function f08ae and this orthogonal transformation Q is applied to matrix A by calling f08ag.

f08we reduces a pair of real matrices (A, B), where B is upper triangular, to the generalized upper Hessenberg form using orthogonal transformations. This two-sided transformation is of the form

$$Q^{\mathrm{T}}AZ = H$$
$$Q^{\mathrm{T}}BZ = T$$

where H is an upper Hessenberg matrix, T is an upper triangular matrix and Q and Z are orthogonal, matrices determined as products of Givens rotations. They may either be formed explicitly, or they may be postmultiplied into input matrices Q_1 and Z_1, so that

$$Q_1AZ_1^{\mathrm{T}} = (Q_1Q)H(Z_1Z)^{\mathrm{T}},$$
$$Q_1BZ_1^{\mathrm{T}} = (Q_1Q)T(Z_1Z)^{\mathrm{T}}.$$

4 References

Golub G H and Van Loan C F 1996 *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

Moler C B and Stewart G W 1973 An algorithm for generalized matrix eigenproblems *SIAM J. Numer. Anal.* **10** 241–256

5 Parameters

5.1 Compulsory Input Parameters

1: **compq – string**

Specifies the form of the computed orthogonal matrix Q.

compq = 'N'

Do not compute Q.

compq = 'I'

The orthogonal matrix Q is returned.

compq = 'V'

q must contain an orthogonal matrix Q_1, and the product Q_1Q is returned.

Constraint: **compq** = 'N', 'I' or 'V'.

2: **compz – string**

Specifies the form of the computed orthogonal matrix Z.

compz = 'N'

Do not compute Z.

compz = 'I'

The orthogonal matrix Z is returned.

compz = 'V'

z must contain an orthogonal matrix Z_1, and the product Z_1Z is returned.

Constraint: **compz** = 'N', 'I' or 'V'.

3: **ilo – int32 scalar**

4: **ihi – int32 scalar**

i_{lo} and i_{hi} as determined by a previous call to f08wh. Otherwise, they should be set to 1 and n, respectively.

Constraints:

if $\mathbf{n} > 0$, $1 \leq \mathbf{ilo} \leq \mathbf{ihi} \leq \mathbf{n}$;
if $\mathbf{n} = 0$, $\mathbf{ilo} = 1$ and $\mathbf{ihi} = 0$.

5: **a(lda,∗) – double array**

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

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

The matrix A of the matrix pair (A, B). Usually, this is the matrix A returned by f08ag.

6: **b(ldb,∗) – double array**

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

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

The upper triangular matrix B of the matrix pair (A, B). Usually, this is the matrix B returned by the QR factorization function f08ae.

7: **q(ldq,∗) – double array**

The first dimension, **ldq**, of the array **q** must satisfy

if **compq** = 'I' or 'V', $\mathbf{ldq} \geq \max(1, \mathbf{n})$;
if **compq** = 'N', $\mathbf{ldq} \geq 1$.

The second dimension of the array must be at least $\max(1, \mathbf{n})$ if **compq** = 'I' or 'V' and at least 1 if **compq** = 'N'

If **compq** = 'V', **q** must contain an orthogonal matrix Q_1.

If **compq** = 'N', **q** is not referenced.

8: **z(ldz,∗) – double array**

The first dimension, **ldz**, of the array **z** must satisfy

if **compz** = 'V' or 'I', **ldz** $\geq \max(1, \mathbf{n})$;
if **compz** = 'N', **ldz** ≥ 1.

The second dimension of the array must be at least $\max(1, \mathbf{n})$ if **compz** = 'I' or 'V' and at least 1 if **compz** = 'N'

If **compz** = 'V', **z** must contain an orthogonal matrix Z_1.

If **compz** = 'N', **z** is not referenced.

5.2 Optional Input Parameters

1: **n – int32 scalar**

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

n, the order of the matrices A and B.

Constraint: **n** ≥ 0.

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldb, ldq, ldz

5.4 Output Parameters

1: **a(lda,∗) – double array**

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

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

a contains the upper Hessenberg matrix H.

2: **b(ldb,∗) – double array**

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

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

b contains the upper triangular matrix T.

3: **q(ldq,∗) – double array**

The first dimension, **ldq**, of the array **q** must satisfy

if **compq** = 'I' or 'V', **ldq** $\geq \max(1, \mathbf{n})$;
if **compq** = 'N', **ldq** ≥ 1.

The second dimension of the array must be at least $\max(1, \mathbf{n})$ if **compq** = 'I' or 'V' and at least 1 if **compq** = 'N'

If **compq** = 'I', **q** contains the orthogonal matrix Q.

Iif **compq** = 'V', **q** contains $Q_1 Q$.

4: **z(ldz,∗) – double array**

The first dimension, **ldz**, of the array **z** must satisfy

if **compz** = 'V' or 'I', **ldz** $\geq \max(1, \mathbf{n})$;
if **compz** = 'N', **ldz** ≥ 1.

The second dimension of the array must be at least $\max(1, \mathbf{n})$ if **compz** = 'I' or 'V' and at least 1 if **compz** = 'N'

If **compz** = 'I', **z** contains the orthogonal matrix Z.

If **compz** = 'V', **z** contains $Z_1 Z$.

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: **compq**, 2: **compz**, 3: **n**, 4: **ilo**, 5: **ihi**, 6: **a**, 7: **lda**, 8: **b**, 9: **ldb**, 10: **q**, 11: **ldq**, 12: **z**, 13: **ldz**, 14: **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 reduction to the generalized Hessenberg form is implemented using orthogonal transformations which are backward stable.

8 Further Comments

This function is usually followed by f08xe which implements the QZ algorithm for computing generalized eigenvalues of a reduced pair of matrices.

The complex analogue of this function is f08ws.

9 Example

```
compq = 'N';
compz = 'N';
ilo = int32(1);
ihi = int32(5);
a   =   [-2.189810707792021,  -3.860243234889781,  -0.7832784454794537,  -
1.868357725686654, -5.398725629594945;
         -0.6852853888947079,  -3.288784152686995,  -1.438693822365931,  -
6.499961199366684, -30.43126686913852;
          -0.4675345497569942,  -0.9460296237838147,  0.05890247520307763,
1.68772079635865, 11.83388168132396;
        0.1275112017793676,  0.08562118570231017,  -0.08916578356814751,  -
0.5024032653213368, -1.524287626980205;
          -0.01628919921740115,  0.0108594661449348,  0.01466027929566115,
0.03366434504929487, -0.05049651757395134];
b   =   [-1.424780684877501,   -4.604217385613993,   -1.088588592238918,   -
2.347027886812987, -4.825304043612226;
          0.4124084319199036,  -3.498740096950594,  -2.422558190366834,  -
9.651981131278539, -28.51976667786654;
           0.04124084319199037,  0.1104973584069892,  0.6928828841397809,
3.972051174738771, 13.55775258164351;
            0.04124084319199037,  0.2780329357492619,  0.015085691276236,
0.3071315763993982, 1.792521439103921;
          0.04124084319199037,  0.6131040904338072,  -0.8759673660952852,
```

```
0.1236241913362016, 0.03257839843480646];
q = [-0.4524812702907142];
z = [-0.4521589283762347];
[aOut, bOut, qOut, zOut, info] = f08we(compq, compz, ilo, ihi, a, b, q,
z)
```

```
aOut =
   -2.1898   -0.3181    2.0547    4.7371   -4.6249
   -0.8395   -0.0426    1.7132    7.5194  -17.1850
         0   -0.2846   -1.0101   -7.5927   26.4499
         0         0    0.0376    1.4070   -3.3643
         0         0         0    0.3813   -0.9937
bOut =
   -1.4248   -0.3476    2.1175    5.5813   -3.9269
         0   -0.0782    0.1189    8.0940  -15.2928
         0         0    1.0021  -10.9356   26.5971
         0         0         0    0.5820   -0.0730
         0         0         0         0    0.5321
qOut =
   -0.4525
zOut =
   -0.4522
info =
           0
```