NAG Toolbox for MATLAB

f08wn

1 Purpose

f08wn computes for a pair of n by n complex nonsymmetric matrices (A, B), the generalized eigenvalues and, optionally, the left and/or right generalized eigenvectors using the QZ algorithm.

2 Syntax

```
[a, b, alpha, beta, vl, vr, info] = f08wn(jobvl, jobvr, a, b, 'n', n)
```

3 Description

A generalized eigenvalue for a pair of matrices (A, B) is a scalar λ or a ratio $\alpha/\beta = \lambda$, such that $A - \lambda B$ is singular. It is usually represented as the pair (α, β), as there is a reasonable interpretation for $\beta = 0$, and even for both being zero.

The right generalized eigenvector v_j corresponding to the generalized eigenvalue λ_j of (A, B) satisfies

$$Av_j = \lambda_j Bv_j.$$

The left generalized eigenvector u_j corresponding to the generalized eigenvalues λ_j of (A, B) satisfies

$$u_j^{\mathrm{H}} A = \lambda_j u_j^{\mathrm{H}} B,$$

where u_j^{H} is the conjugate-transpose of u_j.

All the eigenvalues and, if required, all the eigenvectors of the complex generalized eigenproblem $Ax = \lambda Bx$ where A and B are complex, square matrices, are determined using the QZ algorithm. The complex QZ algorithm consists of three stages:

1. A is reduced to upper Hessenberg form (with real, nonnegative subdiagonal elements) and at the same time B is reduced to upper triangular form.
2. A is further reduced to triangular form while the triangular form of B is maintained and the diagonal elements of B are made real and nonnegative. This is the generalized Schur form of the pair (A, B).

   This function does not actually produce the eigenvalues λ_j, but instead returns α_j and β_j such that

   $$\lambda_j = \alpha_j/\beta_j, \qquad j = 1, 2, \ldots, n.$$

   The division by β_j becomes your responsibility, since β_j may be zero, indicating an infinite eigenvalue.
3. If the eigenvectors are required they are obtained from the triangular matrices and then transferred back into the original co-ordinate system.

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

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

Wilkinson J H 1979 Kronecker's canonical form and the QZ algorithm *Linear Algebra Appl.* **28** 285–303

5 Parameters

5.1 Compulsory Input Parameters

1: **jobvl – string**

If **jobvl** = 'N', do not compute the left generalized eigenvectors.

If **jobvl** = 'V', compute the left generalized eigenvectors.

Constraint: **jobvl** = 'N' or 'V'.

2: **jobvr – string**

If **jobvr** = 'N', do not compute the right generalized eigenvectors.

If **jobvr** = 'V', compute the right generalized eigenvectors.

Constraint: **jobvr** = 'N' or 'V'.

3: **a(lda,$*$) – complex 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 in the pair (A,B).

4: **b(ldb,$*$) – complex 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 matrix B in the pair (A,B).

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The first dimension of the arrays **a**, **b** and the second dimension of the arrays **a**, **b**. (An error is raised if these dimensions are not equal.)

n, the order of the matrices A and B.

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

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldb, ldvl, ldvr, work, lwork, rwork

5.4 Output Parameters

1: **a(lda,$*$) – complex 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 has been overwritten.

2: **b(ldb,$*$) – complex 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 has been overwritten.

3: **alpha**($*$) – **complex array**

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

See the description of **beta**.

4: **beta**($*$) – **complex array**

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

alpha(j)/**beta**(j), for $j = 1, \ldots, \mathbf{n}$, will be the generalized eigenvalues.

Note: the quotients **alpha**(j)/**beta**(j) may easily overflow or underflow, and **beta**(j) may even be zero. Thus, you should avoid naively computing the ratio α_j/β_j. However, $\max|\alpha_j|$ will always be less than and usually comparable with $\|\mathbf{a}\|_2$ in magnitude, and $\max|\beta_j|$ will always be less than and usually comparable with $\|\mathbf{b}\|_2$.

5: **vl**(**ldvl**,$*$) – **complex array**

The first dimension, **ldvl**, of the array **vl** must satisfy

if **jobvl** = 'V', **ldvl** $\geq \max(1, \mathbf{n})$;
ldvl ≥ 1 otherwise.

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

If **jobvl** = 'V', the left generalized eigenvectors u_j are stored one after another in the columns of **vl**, in the same order as the corresponding eigenvalues. Each eigenvector will be scaled so the largest component will have $|\text{real part}| + |\text{imag. part}| = 1$.

If **jobvl** = 'N', **vl** is not referenced.

6: **vr**(**ldvr**,$*$) – **complex array**

The first dimension, **ldvr**, of the array **vr** must satisfy

if **jobvr** = 'V', **ldvr** $\geq \max(1, \mathbf{n})$;
ldvr ≥ 1 otherwise.

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

If **jobvr** = 'V', the right generalized eigenvectors v_j are stored one after another in the columns of **vr**, in the same order as the corresponding eigenvalues. Each eigenvector will be scaled so the largest component will have $|\text{real part}| + |\text{imag. part}| = 1$.

If **jobvr** = 'N', **vr** is not referenced.

7: **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: **jobvl**, 2: **jobvr**, 3: **n**, 4: **a**, 5: **lda**, 6: **b**, 7: **ldb**, 8: **alpha**, 9: **beta**, 10: **vl**, 11: **ldvl**, 12: **vr**, 13: **ldvr**, 14: **work**, 15: **lwork**, 16: **rwork**, 17: **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.

info $= 1$ to N

The *QZ* iteration failed. No eigenvectors have been calculated, but **alpha**(j) and **beta**(j) should be correct for $j = \textbf{info} + 1, \ldots, \textbf{n}$.

info $= N + 1$

Unexpected error returned from f08xs.

info $= N + 2$

Error returned from f08yx.

7 Accuracy

The computed eigenvalues and eigenvectors are exact for a nearby matrices $(A + E)$ and $(B + F)$, where

$$\|(E,F)\|_F = O(\epsilon)\|(A,B)\|_F,$$

and ϵ is the ***machine precision***. See Section 4.11 of Anderson *et al.* 1999 for further details.

Note: interpretation of results obtained with the *QZ* algorithm often requires a clear understanding of the effects of small changes in the original data. These effects are reviewed in Wilkinson 1979, in relation to the significance of small values of α_j and β_j. It should be noted that if α_j and β_j are **both** small for any j, it may be that no reliance can be placed on **any** of the computed eigenvalues $\lambda_i = \alpha_i/\beta_i$. You are recommended to study Wilkinson 1979 and, if in difficulty, to seek expert advice on determining the sensitivity of the eigenvalues to perturbations in the data.

8 Further Comments

The total number of floating-point operations is proportional to n^3.

The real analogue of this function is f08wa.

9 Example

```
jobvl = 'No left vectors';
jobvr = 'Vectors (right)';
a = [complex(-21.1, -22.5), complex(53.5, -50.5), complex(-34.5, +127.5),
complex(7.5, +0.5);
      complex(-0.46, -7.78), complex(-3.5, -37.5), complex(-15.5, +58.5),
complex(-10.5, -1.5);
       complex(4.3, -5.5), complex(39.7, -17.1), complex(-68.5, +12.5),
complex(-7.5, -3.5);
        complex(5.5, +4.4), complex(14.4, +43.3), complex(-32.5, -46),
complex(-19, -32.5)];
b = [complex(1, -5), complex(1.6, +1.2), complex(-3, +0), complex(0, -1);
     complex(0.8, -0.6), complex(3, -5), complex(-4, +3), complex(-2.4, -
3.2);
     complex(1, +0), complex(2.4, +1.8), complex(-4, -5), complex(0, -3);
      complex(0, +1), complex(-1.8, +2.4), complex(0, -4), complex(4, -
5)];
[aOut, bOut, alpha, beta, vl, vr, info] = f08wn(jobvl, jobvr, a, b)
```

```
aOut =
   1.0e+02 *
    0.1903 - 0.5710i    0.5359 - 0.8982i   -0.8131 - 0.6323i    1.0666 -
0.4479i
         0                  0.1188 - 0.2970i    0.0356 + 0.2763i   -0.0067 -
0.1642i
         0                       0                  0.1096 - 0.0365i   -0.2502 -
0.0820i
         0                       0                       0                 0.2187 -
0.2734i
```

```
bOut =
    6.3443                3.3986 + 0.7119i  -0.5152 - 2.3820i   6.5818 +
2.4299i
         0               5.9409             -2.4480 - 0.3427i   5.7385 -
0.7017i
         0                    0              3.6536             -1.4096 -
3.9326i
        0                  0                  0             5.4681
alpha =
  19.0329 -57.0986i
  11.8818 -29.7045i
  10.9609 - 3.6536i
  21.8722 -27.3403i
beta =
    6.3443
    5.9409
    3.6536
    5.4681
vl =
  4.4316e-195 - 1.3018e-54i
vr =
   -0.8238 - 0.1762i    0.6397 + 0.3603i    0.9775 + 0.0225i   -0.9062 +
0.0938i
   -0.1530 + 0.0707i    0.0042 - 0.0005i    0.1591 - 0.1137i   -0.0074 +
0.0069i
   -0.0707 - 0.1530i    0.0402 + 0.0226i    0.1209 - 0.1537i    0.0302 -
0.0031i
    0.1530 - 0.0707i   -0.0226 + 0.0402i    0.1537 + 0.1209i   -0.0146 -
0.1410i
info =
           0
```