NAG Toolbox for MATLAB

f08yk

1 Purpose

f08yk computes some or all of the right and/or left generalized eigenvectors of a pair of real matrices (A, B) which are in generalized real Schur form.

2 Syntax

```
[vl, vr, m, info] = f08yk(side, howmny, select, a, b, vl, vr, mm, 'n',
n)
```

3 Description

f08yk computes some or all of the right and/or left generalized eigenvectors of the matrix pair (A, B) which is assumed to be in generalized upper Schur form. If the matrix pair (A, B) is not in the generalized upper Schur form, then f08xe should be called before invoking f08yk.

The right generalized eigenvector x and the left generalized eigenvector y of (A, B) corresponding to a generalized eigenvalue λ are defined by

$$(A - \lambda B)x = 0$$

and

$$y^{\mathrm{H}}(A - \lambda B) = 0.$$

If a generalized eigenvalue is determined as $0/0$, which is due to zero diagonal elements at the same locations in both A and B, a unit vector is returned as the corresponding eigenvector.

Note that the generalized eigenvalues are computed using f08xe but f08yk does not explicitly require the generalized eigenvalues to compute eigenvectors. The ordering of the eigenvectors is based on the ordering of the eigenvalues as computed by f08yk.

If all eigenvectors are requested, the function may either return the matrices X and/or Y of right or left eigenvectors of (A, B), or the products ZX and/or QY, where Z and Q are two matrices supplied by you. Usually, Q and Z are chosen as the orthogonal matrices returned by f08xe. Equivalently, Q and Z are the left and right Schur vectors of the matrix pair supplied to f08xe. In that case, QY and ZX are the left and right generalized eigenvectors, respectively, of the matrix pair supplied to f08xe.

A must be block upper triangular; with 1 by 1 and 2 by 2 diagonal blocks. Corresponding to each 2 by 2 diagonal block is a complex conjugate pair of eigenvalues and eigenvectors; only one eigenvector of the pair is computed, namely the one corresponding to the eigenvalue with positive imaginary part. Each 1 by 1 block gives a real generalized eigenvalue and a corresponding eigenvector.

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

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

Stewart G W and Sun J-G 1990 *Matrix Perturbation Theory* Academic Press, London

5 Parameters

5.1 Compulsory Input Parameters

1: **side – string**

Specifies the required sets of generalized eigenvectors.

side = 'R'

Only right eigenvectors are computed.

side = 'L'

Only left eigenvectors are computed.

side = 'B'

Both left and right eigenvectors are computed.

Constraint: **side** = 'B', 'L' or 'R'.

2: **howmny – string**

Specifies further details of the required generalized eigenvectors.

howmny = 'A'

All right and/or left eigenvectors are computed.

howmny = 'B'

All right and/or left eigenvectors are computed; they are backtransformed using the input matrices supplied in arrays **vr** and/or **vl**.

howmny = 'S'

Selected right and/or left eigenvectors, defined by the array **select**, are computed.

Constraint: **howmny** = 'A', 'B' or 'S'.

3: **select**$(*)$ **– logical array**

Note: the dimension of the array **select** must be at least $\max(1,\mathbf{n})$ if **howmny** = 'S', and at least 1 otherwise.

Specifies the eigenvectors to be computed if **howmny** = 'S'. To select the generalized eigenvector corresponding to the jth generalized eigenvalue, the jth element of **select** should be set to **true**; if the eigenvalue corresponds to a complex conjugate pair, then real and imaginary parts of eigenvectors corresponding to the complex conjugate eigenvalue pair will be computed.

Constraint: **select**$(j) =$ **true** or **false**, for $j = 1,2,\ldots,n$.

4: **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 pair (A,B) must be in the generalized Schur form. Usually, this is the matrix A returned by f08xe.

5: **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 matrix pair (A,B) must be in the generalized Schur form. If A has a 2 by 2 diagonal block then the corresponding 2 by 2 block of B must be diagonal with positive elements. Usually, this is the matrix B returned by f08xe.

6: **vl(ldvl,$*$) – double array**

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

if **side** = 'L' or 'B', **ldvl** $\geq \max(1, \mathbf{n})$;
if **side** = 'R', **ldvl** ≥ 1.

The second dimension of the array must be at least $\max(1, \mathbf{mm})$ if **side** = 'L' or 'B' and at least 1 if **side** = 'R'

If **howmny** = 'B' and **side** = 'L' or 'B', **vl** must be initialized to an n by n matrix Q. Usually, this is the orthogonal matrix Q of left Schur vectors returned by f08xe.

7: **vr(ldvr,$*$) – double array**

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

if **side** = 'R' or 'B', **ldvr** $\geq \max(1, \mathbf{n})$;
if **side** = 'L', **ldvr** ≥ 1.

The second dimension of the array must be at least $\max(1, \mathbf{mm})$ if **side** = 'R' or 'B' and at least 1 if **side** = 'L'

If **howmny** = 'B' and **side** = 'R' or 'B', **vr** must be initialized to an n by n matrix Z. Usually, this is the orthogonal matrix Z of right Schur vectors returned by f08xe.

8: **mm – int32 scalar**

The number of columns in the arrays **vl** and/or **vr**.

Constraints:

if **howmny** = 'A' or 'B', **mm** $\geq$ **n**;
if **howmny** = 'S', **mm** must not be less than the number of requested eigenvectors.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The first dimension of the arrays **vl**, **vr**. (An error is raised if these dimensions are not equal.)

n, the order of the matrices A and B.

Constraint: **n** ≥ 0.

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldb, ldvl, ldvr, work

5.4 Output Parameters

1: **vl(ldvl,$*$) – double array**

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

if **side** = 'L' or 'B', **ldvl** $\geq \max(1, \mathbf{n})$;
if **side** = 'R', **ldvl** ≥ 1.

The second dimension of the array must be at least $\max(1, \mathbf{mm})$ if **side** = 'L' or 'B' and at least 1 if **side** = 'R'

If **side** = 'L' or 'B', **vl** contains:

if **howmny** = 'A', the matrix Y of left eigenvectors of (A, B);
if **howmny** = 'B', the matrix QY;
if **howmny** = 'S', the left eigenvectors of (A, B) specified by **select**, stored consecutively in the columns of the array **vl**, in the same order as their corresponding eigenvalues.

A complex eigenvector corresponding to a complex eigenvalue is stored in two consecutive columns, the first holding the real part, and the second the imaginary part.

2: **vr**(**ldvr**,$*$) **– double array**

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

if **side** $=$ 'R' or 'B', **ldvr** $\geq \max(1, \mathbf{n})$;
if **side** $=$ 'L', **ldvr** ≥ 1.

The second dimension of the array must be at least $\max(1, \mathbf{mm})$ if **side** $=$ 'R' or 'B' and at least 1 if **side** $=$ 'L'

If **side** $=$ 'R' or 'B', **vr** contains:

if **howmny** $=$ 'A', the matrix X of right eigenvectors of (A, B);
if **howmny** $=$ 'B', the matrix ZX;
if **howmny** $=$ 'S', the right eigenvectors of (A, B) specified by **select**, stored consecutively in the columns of the array **vr**, in the same order as their corresponding eigenvalues.

A complex eigenvector corresponding to a complex eigenvalue is stored in two consecutive columns, the first holding the real part, and the second the imaginary part.

3: **m – int32 scalar**

The number of columns in the arrays **vl** and/or **vr** actually used to store the eigenvectors. If **howmny** $=$ 'A' or 'B', **m** is set to **n**. Each selected real eigenvector occupies one column and each selected complex eigenvector occupies two columns.

4: **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: **side**, 2: **howmny**, 3: **select**, 4: **n**, 5: **a**, 6: **lda**, 7: **b**, 8: **ldb**, 9: **vl**, 10: **ldvl**, 11: **vr**, 12: **ldvr**, 13: **mm**, 14: **m**, 15: **work**, 16: **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 > 0

If **info** $= i$, the 2 by 2 block $(\mathbf{info} : \mathbf{info} + 1)$ does not have complex eigenvalues.

7 Accuracy

It is beyond the scope of this manual to summarize the accuracy of the solution of the generalized eigenvalue problem. Interested readers should consult Section 4.11 of the LAPACK Users' Guide (see Anderson *et al.* 1999) and Chapter 6 of Stewart and Sun 1990.

8 Further Comments

f08yk is the sixth step in the solution of the real generalized eigenvalue problem and is called after f08xe.

The complex analogue of this function is f08yx.

9 Example

```
side = 'B';
howmny = 'B';
select = [false];
a  =  [-0.4330439202822806,  0.8882467533979291,  0.6987419307539519,  -
2.78606012720401, 33.3000678734937;
          0,  2.272041918502348,  -0.778564278012238,  -3.789565374915556,
2.810326368299267;
        0,  0.6959451777892208,  0.01598047470420195,  -2.211594557807314,
3.695758182229978;
     0, 0, 0, 0.7767255566261145, -3.926125447765922;
     0, 0, 0, 0, -0.1777200369901831];
b  =  [0.1777200369901909,  -1.884221925375466,  -0.3451471567162397,  -
5.770754287406221, 33.3000678734937;
     0, 1.95938614900662, -0, -4.397648586028289, 2.331281190880262;
     0, 0, 0.2950650455399987, -1.496434947184499, 1.891993010362068;
     0, 0, 0, 0.7767255566261325, -0.1576200206221604;
     0, 0, 0, 0, 0.4330439202822933];
vl  =  [0.0137935688646433,  0.6395783098103517,  -0.7492306336949638,  -
0.1654939390665379, 0.0448835910260 4707;
          -0.002219113006760248,  0.7648879322844937,  0.5998307886848795,
0.2141230003947479, -0.09641520746563347;
        0.01146387551054054,  0.06623895480509465,  0.1915886650277972,  -
0.3638783083660775, 0.9090474908155874;
         0.1858689994006417,  0.03577152029604508,  0.2032724508670227,  -
0.8744156286739607, -0.3978074062241045;
         0.9824083202088234,  -0.01479310203181619,  -0.02881972640567903,
0.1724905268864221, 0.06380829116266883];
vr  =  [-0.04488359102604732,  0.5436567257530147,  -0.8349106547137655,  -
0.07181061135741797, 0.01379356886464299;
         0.09641520746563365,  0.4776249909256048,  0.3736731484120213,  -
0.7892667036597866, -0.002219113006760028;
       -0.9090474908155884, -0.2595381867122672, -0.09308800212270843, -
0.3122111344892641, 0.01146387551054191;
         0.3978074062241038,  -0.628980937908613,  -0.3836535012414873,  -
0.5141934979633679, 0.1858689994006436;
          -0.06380829116266826,  0.1154757202728654,  0.08623915379472402,
0.1001526871512249, 0.9824083202088231];
mm = int32(5);
[vlOut, vrOut, m, info] = f08yk(side, howmny, select, a, b, vl, vr, mm)
```

```
vlOut =
   -0.0695   -0.2092   -0.0053   -0.0741    0.0494
    0.1361    0.1635    0.1137    0.1354   -0.1061
   -1.0000   -0.4631   -0.5369   -1.0000    1.0000
    0.3188    0.0591    0.1480    0.2646   -0.4376
   -0.0355   -0.0025   -0.0214   -0.0370    0.0702
vrOut =
   -0.0494   -0.2077    0.0257   -0.0741   -0.0695
    0.1061    0.1785    0.0883    0.1354    0.1361
   -1.0000   -0.5374   -0.4626   -1.0000   -1.0000
    0.4376    0.0803    0.1377    0.2646    0.3188
   -0.0702   -0.0056   -0.0208   -0.0370   -0.0355
m =
           5
info =
           0
```