

NAG Toolbox for MATLAB

f08wj

1 Purpose

f08wj forms the right or left eigenvectors of the real generalized eigenvalue problem $Ax = \lambda Bx$, by backward transformation on the computed eigenvectors given by f08yk. It is necessary to call this function only if the optional balancing function f08wh was previously called to balance the matrix pair (A, B) .

2 Syntax

```
[v, info] = f08wj(job, side, ilo, ihi, lscale, rscale, v, 'n', n, 'm', m)
```

3 Description

If the matrix pair has been previously balanced using function f08wh then f08wj backtransforms the eigenvector solution given by f08yk. This is usually the sixth and last step in the solution of the generalized eigenvalue problem.

For a description of balancing, see the document for f08wh.

4 References

Ward R C 1981 Balancing the generalized eigenvalue problem *SIAM J. Sci. Stat. Comp.* **2** 141–152

5 Parameters

5.1 Compulsory Input Parameters

1: **job** – string

Specifies the backward transformation step required.

job = 'N'

No transformations are done.

job = 'P'

Only do backward transformations based on permutations.

job = 'S'

Only do backward transformations based on scaling.

job = 'B'

Do backward transformations for both permutations and scaling.

Note: this must be the same parameter **job** as supplied to f08wh.

Constraint: **job** = 'N', 'P', 'S' or 'B'.

2: **side** – string

Indicates whether left or right eigenvectors are to be transformed.

side = 'L'

The left eigenvectors are transformed.

side = 'R'

The right eigenvectors are transformed.

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

3: **ilo** – int32 scalar

4: **ihi** – int32 scalar

i_{lo} and i_{hi} as determined by a previous call to f08wh.

Constraints:

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

5: **lscale**(*) – double array

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

Details of the permutations and scaling factors applied to the left side of the matrices A and B , as returned by a previous call to f08wh.

6: **rscale**(*) – double array

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

Details of the permutations and scaling factors applied to the right side of the matrices A and B , as returned by a previous call to f08wh.

7: **v**(ldv,*) – double array

The first dimension of the array **v** must be at least $\max(1, n)$

The second dimension of the array must be at least $\max(1, m)$

The matrix of right or left eigenvectors, as returned by f08wh.

5.2 Optional Input Parameters

1: **n** – int32 scalar

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

n , the order of the matrices A and B of the generalized eigenvalue problem.

Constraint: $n \geq 0$.

2: **m** – int32 scalar

Default: The second dimension of the array **v**.

m , the required number of left or right eigenvectors.

Constraint: $0 \leq m \leq n$.

5.3 Input Parameters Omitted from the MATLAB Interface

ldv

5.4 Output Parameters

1: **v**(ldv,*) – double array

The first dimension of the array **v** must be at least $\max(1, n)$

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

The transformed right or left eigenvectors.

2: **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: **job**, 2: **side**, 3: **n**, 4: **ilo**, 5: **ihi**, 6: **lscale**, 7: **rscale**, 8: **m**, 9: **v**, 10: **ldv**, 11: **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 errors are negligible, compared with the previous computations.

8 Further Comments

The number of operations is proportional to n^2 .

The complex analogue of this function is f08ww.

9 Example

```

job = 'B';
side = 'R';
ilo = int32(1);
ihi = int32(5);
lscale = [1;
          1;
          0.1;
          0.1;
          0.1];
rscale = [1;
          1;
          0.1;
          0.1;
          0.1];
v = [-0.04937430825069238, -0.2077167167739886, 0.02570235232076236, -
0.07407407407407272, -0.06946618486355889;
      0.1060617937343745, 0.178477357681324, 0.0883247593617138,
0.1354497354497355, 0.1360540981855712;
      -1, -0.5374206868348201, -0.46257931316518, -1, -1;
      0.437609047099613, 0.08027747210915526, 0.1376544281787124,
0.264550264550262, 0.3187945700290964;
      -0.0701924726786497, -0.005597421345643963, -0.02080729905547176, -
0.03703703703703624, -0.03553386129852896];
[vOut, info] = f08wj(job, side, ilo, ihi, lscale, rscale, v)

vOut =
    -0.0494    -0.2077     0.0257    -0.0741    -0.0695
     0.1061     0.1785     0.0883     0.1354     0.1361
    -0.1000    -0.0537    -0.0463    -0.1000    -0.1000

```

0.0438	0.0080	0.0138	0.0265	0.0319
-0.0070	-0.0006	-0.0021	-0.0037	-0.0036
info =				
	0			
