NAG Toolbox for MATLAB

f08nw

1 Purpose

f08nw transforms eigenvectors of a balanced matrix to those of the original complex general matrix.

2 Syntax

```
[v, info] = f08nw(job, side, ilo, ihi, scale, v, 'n', n, 'm', m)
```

3 Description

f08nw is intended to be used after a complex general matrix A has been balanced by f08nv, and eigenvectors of the balanced matrix A''_{22} have subsequently been computed.

For a description of balancing, see the document for f08nv. The balanced matrix A'' is obtained as $A'' = DPAP^{\mathrm{T}}D^{-1}$, where P is a permutation matrix and D is a diagonal scaling matrix. This function transforms left or right eigenvectors as follows:

if x is a right eigenvector of A'', $P^{\mathrm{T}}D^{-1}x$ is a right eigenvector of A;
if y is a left eigenvector of A'', $P^{\mathrm{T}}Dy$ is a left eigenvector of A.

4 References

None.

5 Parameters

5.1 Compulsory Input Parameters

1: **job – string**

This **must** be the same parameter **job** as supplied to f08nv.

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**

The values i_{lo} and i_{hi}, as returned by f08nv.

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: **scale**$(*)$ **– double array**

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

Details of the permutations and/or the scaling factors used to balance the original complex general matrix, as returned by f08nv.

6: **v**(**ldv,**$*$) **– complex array**

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

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

The matrix of left or right eigenvectors to be transformed.

5.2 Optional Input Parameters

1: **n – int32 scalar**

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

n, the number of rows of the matrix of eigenvectors.

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

2: **m – int32 scalar**

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

m, the number of columns of the matrix of eigenvectors.

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

5.3 Input Parameters Omitted from the MATLAB Interface

ldv

5.4 Output Parameters

1: **v**(**ldv,**$*$) **– complex array**

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

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

The transformed eigenvectors.

2: **info – int32 scalar**

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: **job**, 2: **side**, 3: **n**, 4: **ilo**, 5: **ihi**, 6: **scale**, 7: **m**, 8: **v**, 9: **ldv**, 10: **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.

8 Further Comments

The total number of real floating-point operations is approximately proportional to nm.

The real analogue of this function is f08nj.

9 Example

```
job = 'Both';
side = 'Right';
ilo = int32(2);
ihi = int32(3);
scale = [3;
     1;
     1;
     1];
v  =  [complex(1,  +0),  complex(0.6032225283394844,  -0.3967774716605157),
complex(0.2982663996834333, ...
                    +0.7017336003165667),     complex(0.7768255861335958,
+0.2231744138664043);
     complex(0, +0), complex(-0.06157074124970285, +0.04126986941002295),
...
         complex(0.4613150078539515,  -1.765599476879182e-17),  complex(-
0.2071937862358928, -0.2450198886726209);
                  complex(0,   +0),   complex(0.08223214401049841,   +0),
complex(0.4251208555461429, ...
                    +0.284951615584963),     complex(-0.01188956374147339,
+0.4371839616835084);
                complex(0,   +0),   complex(0,   +0),   complex(0,   +0),
complex(0.1452103084889974, +0)];
[vOut, info] = f08nw(job, side, ilo, ihi, scale, v)
```

```
vOut =
        0                  0                  0             0.1452
          0              -0.0616 + 0.0413i    0.4613 - 0.0000i   -0.2072 -
0.2450i
    1.0000                0.6032 - 0.3968i    0.2983 + 0.7017i    0.7768 +
0.2232i
          0               0.0822               0.4251 + 0.2850i   -0.0119 +
0.4372i
info =
           0
```