NAG Toolbox

nag_lapack_zhseqr (f08ps)

1 Purpose

nag_lapack_zhseqr (f08ps) computes all the eigenvalues and, optionally, the Schur factorization of a complex Hessenberg matrix or a complex general matrix which has been reduced to Hessenberg form.

2 Syntax

```
[h, w, z, info] = nag_lapack_zhseqr(job, compz, ilo, ihi, h, z, 'n', n)
[h, w, z, info] = f08ps(job, compz, ilo, ihi, h, z, 'n', n)
```

3 Description

nag_lapack_zhseqr (f08ps) computes all the eigenvalues and, optionally, the Schur factorization of a complex upper Hessenberg matrix H:

$$H = ZTZ^{\mathrm{H}},$$

where T is an upper triangular matrix (the Schur form of H), and Z is the unitary matrix whose columns are the Schur vectors z_i. The diagonal elements of T are the eigenvalues of H.

The function may also be used to compute the Schur factorization of a complex general matrix A which has been reduced to upper Hessenberg form H:

$$\begin{aligned} A &= QHQ^{\mathrm{H}}, \text{ where } Q \text{ is unitary,} \\ &= (QZ)T(QZ)^{\mathrm{H}}. \end{aligned}$$

In this case, after nag_lapack_zgehrd (f08ns) has been called to reduce A to Hessenberg form, nag_lapack_zunghr (f08nt) must be called to form Q explicitly; Q is then passed to nag_lapack_zhseqr (f08ps), which must be called with **compz** = 'V'.

The function can also take advantage of a previous call to nag_lapack_zgebal (f08nv) which may have balanced the original matrix before reducing it to Hessenberg form, so that the Hessenberg matrix H has the structure:

$$\begin{pmatrix} H_{11} & H_{12} & H_{13} \\ & H_{22} & H_{23} \\ & & H_{33} \end{pmatrix}$$

where H_{11} and H_{33} are upper triangular. If so, only the central diagonal block H_{22} (in rows and columns i_{lo} to i_{hi}) needs to be further reduced to Schur form (the blocks H_{12} and H_{23} are also affected). Therefore the values of i_{lo} and i_{hi} can be supplied to nag_lapack_zhseqr (f08ps) directly. Also, nag_lapack_zgebak (f08nw) must be called after this function to permute the Schur vectors of the balanced matrix to those of the original matrix. If nag_lapack_zgebal (f08nv) has not been called however, then i_{lo} must be set to 1 and i_{hi} to n. Note that if the Schur factorization of A is required, nag_lapack_zgebal (f08nv) must **not** be called with **job** = 'S' or 'B', because the balancing transformation is not unitary.

nag_lapack_zhseqr (f08ps) uses a multishift form of the upper Hessenberg QR algorithm, due to Bai and Demmel (1989). The Schur vectors are normalized so that $\|z_i\|_2 = 1$, but are determined only to within a complex factor of absolute value 1.

4 References

Bai Z and Demmel J W (1989) On a block implementation of Hessenberg multishift QR iteration *Internat. J. High Speed Comput.* **1** 97–112

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

5 Parameters

5.1 Compulsory Input Parameters

1: **job** – CHARACTER(1)

Indicates whether eigenvalues only or the Schur form T is required.

job = 'E'
Eigenvalues only are required.

job = 'S'
The Schur form T is required.

Constraint: **job** = 'E' or 'S'.

2: **compz** – CHARACTER(1)

Indicates whether the Schur vectors are to be computed.

compz = 'N'
No Schur vectors are computed (and the array **z** is not referenced).

compz = 'V'
The Schur vectors of A are computed (and the array **z** must contain the matrix Q on entry).

compz = 'I'
The Schur vectors of H are computed (and the array **z** is initialized by the function).

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

3: **ilo** – INTEGER

4: **ihi** – INTEGER

If the matrix A has been balanced by nag_lapack_zgebal (f08nv), then **ilo** and **ihi** must contain the values returned by that function. Otherwise, **ilo** must be set to 1 and **ihi** to **n**.

Constraint: $\mathbf{ilo} \geq 1$ and $\min(\mathbf{ilo}, \mathbf{n}) \leq \mathbf{ihi} \leq \mathbf{n}$.

5: **h**$(ldh, :)$ – COMPLEX (KIND=nag_wp) array

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

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

The n by n upper Hessenberg matrix H, as returned by nag_lapack_zgehrd (f08ns).

6: **z**$(ldz, :)$ – COMPLEX (KIND=nag_wp) 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', $ldz \geq 1$.

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

If **compz** = 'V', **z** must contain the unitary matrix Q from the reduction to Hessenberg form.

If **compz** = 'I', **z** need not be set.

5.2 Optional Input Parameters

1: **n** – INTEGER

Default: the first dimension of the array **h** and the second dimension of the array **h**. (An error is raised if these dimensions are not equal.)

n, the order of the matrix H.

Constraint: **n** ≥ 0.

5.3 Output Parameters

1: **h**$(ldh,:)$ – COMPLEX (KIND=nag_wp) array

The first dimension of the array **h** will be $\max(1,\mathbf{n})$.

The second dimension of the array **h** will be $\max(1,\mathbf{n})$.

If **job** = 'E', the array contains no useful information.

If **job** = 'S', **h** stores the upper triangular matrix T from the Schur decomposition (the Schur form) unless **info** > 0.

2: **w**$(:)$ – COMPLEX (KIND=nag_wp) array

The dimension of the array **w** will be $\max(1,\mathbf{n})$

The computed eigenvalues, unless **info** > 0 (in which case see Section 6). The eigenvalues are stored in the same order as on the diagonal of the Schur form T (if computed).

3: **z**$(ldz,:)$ – COMPLEX (KIND=nag_wp) array

The first dimension, ldz, of the array **z** will be

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

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

If **compz** = 'V' or 'I', **z** contains the unitary matrix of the required Schur vectors, unless **info** > 0.

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

4: **info** – INTEGER

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: **compz**, 3: **n**, 4: **ilo**, 5: **ihi**, 6: **h**, 7: **ldh**, 8: **w**, 9: **z**, 10: **ldz**, 11: **work**, 12: **lwork**, 13: **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

The algorithm has failed to find all the eigenvalues after a total of $30 \times (\mathbf{ihi} - \mathbf{ilo} + 1)$ iterations. If **info** $= i$, elements $1, 2, \ldots, \mathbf{ilo} - 1$ and $i + 1, i + 2, \ldots, n$ of **w** contain the eigenvalues which have been found.

If **job** $=$ 'E', then on exit, the remaining unconverged eigenvalues are the eigenvalues of the upper Hessenberg matrix $\hat{H}$, formed from $\mathbf{h}(\mathbf{ilo} : \mathbf{info}, \mathbf{ilo} : \mathbf{info})$, i.e., the **ilo** through **info** rows and columns of the final output matrix H.

If **job** $=$ 'S', then on exit

$$(*) \quad H_i U = U \tilde{H}$$

for some matrix U, where H_i is the input upper Hessenberg matrix and $\tilde{H}$ is an upper Hessenberg matrix formed from $\mathbf{h}(\mathbf{info} + 1 : \mathbf{ihi}, \mathbf{info} + 1 : \mathbf{ihi})$.

If **compz** $=$ 'V', then on exit

$$Z_{\text{out}} = Z_{\text{in}} U$$

where U is defined in $(*)$ (regardless of the value of **job**).

If **compz** $=$ 'I', then on exit

$$Z_{\text{out}} = U$$

where U is defined in $(*)$ (regardless of the value of **job**).

If **info** > 0 and **compz** $=$ 'N', then **z** is not accessed.

7 Accuracy

The computed Schur factorization is the exact factorization of a nearby matrix $(H + E)$, where

$$\|E\|_2 = O(\epsilon)\|H\|_2,$$

and ϵ is the ***machine precision***.

If λ_i is an exact eigenvalue, and $\tilde{\lambda}_i$ is the corresponding computed value, then

$$\left|\tilde{\lambda}_i - \lambda_i\right| \leq \frac{c(n)\epsilon\|H\|_2}{s_i},$$

where $c(n)$ is a modestly increasing function of n, and s_i is the reciprocal condition number of λ_i. The condition numbers s_i may be computed by calling nag_lapack_ztrsna (f08qy).

8 Further Comments

The total number of real floating-point operations depends on how rapidly the algorithm converges, but is typically about:

$25n^3$ if only eigenvalues are computed;

$35n^3$ if the Schur form is computed;

$70n^3$ if the full Schur factorization is computed.

The real analogue of this function is nag_lapack_dhseqr (f08pe).

9 Example

This example computes all the eigenvalues and the Schur factorization of the upper Hessenberg matrix H, where

$$H = \begin{pmatrix} -3.9700 - 5.0400i & -1.1318 - 2.5693i & -4.6027 - 0.1426i & -1.4249 + 1.7330i \\ -5.4797 + 0.0000i & 1.8585 - 1.5502i & 4.4145 - 0.7638i & -0.4805 - 1.1976i \\ 0.0000 + 0.0000i & 6.2673 + 0.0000i & -0.4504 - 0.0290i & -1.3467 + 1.6579i \\ 0.0000 + 0.0000i & 0.0000 + 0.0000i & -3.5000 + 0.0000i & 2.5619 - 3.3708i \end{pmatrix}.$$

See also Section 10 in nag_lapack_zunghr (f08nt), which illustrates the use of this function to compute the Schur factorization of a general matrix.

9.1 Program Text

```
     function f08ps_example

fprintf('f08ps example results\n\n');

a = [ -3.97 - 5.04i, -4.11 + 3.70i, -0.34 + 1.01i,  1.29 - 0.86i;
       0.34 - 1.50i,  1.52 - 0.43i,  1.88 - 5.38i,  3.36 + 0.65i;
       3.31 - 3.85i,  2.50 + 3.45i,  0.88 - 1.08i,  0.64 - 1.48i;
      -1.10 + 0.82i,  1.81 - 1.59i,  3.25 + 1.33i,  1.57 - 3.44i];

% Reduce (all of) A to upper Hessenberg Form
ilo = nag_int(1);
ihi = nag_int(4);
[H, tau, info] = f08ns(ilo, ihi, a);

% Form Q
[Q, info] = f08nt(ilo, ihi, H, tau);

% Schur factorize H = Y*T*Y' and form Z = QY  A = QY*T*(QQY)'
job   = 'Schur form';
compz = 'Vectors';
[~, w, Z, info] = f08ps( ...
                         job, compz, ilo, ihi, H, Q);

disp('Eigenvalues of A');
disp(w);
```

9.2 Program Results

```
     f08ps example results

Eigenvalues of A
  -6.0004 - 6.9998i
  -5.0000 + 2.0060i
   7.9982 - 0.9964i
   3.0023 - 3.9998i
```