NAG Toolbox for MATLAB

f08hq

1 Purpose

f08hq computes all the eigenvalues and, optionally, all the eigenvectors of a complex Hermitian band matrix. If the eigenvectors are requested, then it uses a divide-and-conquer algorithm to compute eigenvalues and eigenvectors. However, if only eigenvalues are required, then it uses the Pal–Walker–Kahan variant of the *QL* or *QR* algorithm.

2 Syntax

```
[ab, w, z, info] = f08hq(job, uplo, kd, ab, 'n', n)
```

3 Description

f08hq computes all the eigenvalues and, optionally, all the eigenvectors of a complex Hermitian band matrix A. In other words, it can compute the spectral factorization of A as

$$A = Z\Lambda Z^{\mathrm{H}},$$

where Λ is a real diagonal matrix whose diagonal elements are the eigenvalues λ_i, and Z is the (complex) unitary matrix whose columns are the eigenvectors z_i. Thus

$$Az_i = \lambda_i z_i, \qquad i = 1, 2, \ldots, n.$$

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

5 Parameters

5.1 Compulsory Input Parameters

1: **job – string**

Indicates whether eigenvectors are computed.

job = 'N'

Only eigenvalues are computed.

job = 'V'

Eigenvalues and eigenvectors are computed.

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

2: **uplo – string**

Indicates whether the upper or lower triangular part of A is stored.

uplo = 'U'

The upper triangular part of A is stored.

uplo = 'L'

The lower triangular part of A is stored.

Constraint: **uplo** = 'U' or 'L'.

3: **kd – int32 scalar**

If **uplo** = 'U', the number of superdiagonals, k_d, of the matrix A.

If **uplo** = 'L', the number of subdiagonals, k_d, of the matrix A.

Constraint: **kd** ≥ 0.

4: **ab**(**ldab**,$*$) **– complex array**

The first dimension of the array **ab** must be at least **kd** $+ 1$

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

The upper or lower triangle of the n by n Hermitian band matrix A.

The matrix is stored in rows 1 to $k_d + 1$, more precisely,

if **uplo** = 'U', the elements of the upper triangle of A within the band must be stored with element A_{ij} in $\mathbf{ab}(k_d + 1 + i - j, j)$ for $\max(1j - k_d) \leq i \leq j$;
if **uplo** = 'L', the elements of the lower triangle of A within the band must be stored with element A_{ij} in $\mathbf{ab}(1 + i - j, j)$ for $j \leq i \leq \min(nj + k_d)$.

5.2 Optional Input Parameters

1: **n – int32 scalar**

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

n, the order of the matrix A.

Constraint: **n** ≥ 0.

5.3 Input Parameters Omitted from the MATLAB Interface

ldab, ldz, work, lwork, rwork, lrwork, iwork, liwork

5.4 Output Parameters

1: **ab**(**ldab**,$*$) **– complex array**

The first dimension of the array **ab** must be at least **kd** $+ 1$

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

ab contains values generated during the reduction to tridiagonal form.

The first superdiagonal and the diagonal of the tridiagonal matrix T are returned in **ab** using the same storage format as described above.

2: **w**($*$) **– double array**

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

The eigenvalues of the matrix A in ascending order.

3: **z(ldz,∗) – complex array**

The first dimension, **ldz**, of the array **z** must satisfy

if **job** = 'V', **ldz** $\geq \max(1, \mathbf{n})$;
if **job** = 'N', **ldz** ≥ 1.

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

If **job** = 'V', **z** contains the unitary matrix Z which contains the eigenvectors of A. The ith column of Z contains the eigenvector which corresponds to the eigenvalue $\mathbf{w}(i)$.

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

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: **job**, 2: **uplo**, 3: **n**, 4: **kd**, 5: **ab**, 6: **ldab**, 7: **w**, 8: **z**, 9: **ldz**, 10: **work**, 11: **lwork**, 12: **rwork**, 13: **lrwork**, 14: **iwork**, 15: **liwork**, 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** $= i$ and **job** = 'N', the algorithm failed to converge; i elements of an intermediate tridiagonal form did not converge to zero; if **info** $= i$ and **job** = 'V', then the algorithm failed to compute an eigenvalue while working on the submatrix lying in rows and column $i/(\mathbf{n}+1)$ through $\mathrm{mod}(i, \mathbf{n}+1)$.

7 Accuracy

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

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

and ϵ is the ***machine precision***. See Section 4.7 of Anderson *et al.* 1999 for further details.

8 Further Comments

The real analogue of this function is f08hc.

9 Example

```
job = 'V';
uplo = 'L';
kd = int32(2);
ab = [complex(1, +0), complex(2, +0), complex(3, +0), complex(4, +0),
complex(5, +0);
       complex(2, +1), complex(3, +2), complex(4, +3), complex(5, +4),
complex(0, 0);
        complex(3, +1), complex(4, +2), complex(5, +3), complex(0, 0),
complex(0, 0)];
```

```
[abOut, w, z, info] = f08hq(job, uplo, kd, ab)
```

```
abOut =
  Columns 1 through 4
   1.0000            5.2000            7.8051           -0.0720
   3.8730            8.5456            6.2114            1.9595
   3.0000 + 1.0000i   7.2377 + 3.8317i   5.4275 + 0.2013i       0
  Column 5
   1.0669
        0
        0
w =
   -6.4185
   -1.4094
    1.4421
    4.4856
   16.9002
z =
  Columns 1 through 4
   0.2591           -0.6367            0.4516           -0.5503
   -0.0245 - 0.4344i    0.2578 - 0.2413i   -0.3029 - 0.4402i   -0.4785 -
0.2759i
   -0.5159 + 0.1095i    0.3039 + 0.3481i    0.3160 + 0.2978i   -0.2128 -
0.0465i
   -0.0004 + 0.5093i   -0.3450 + 0.0832i   -0.4088 - 0.3213i    0.1707 -
0.0200i
    0.4333 - 0.1353i    0.2469 - 0.2634i    0.0204 + 0.2262i   -0.0175 +
0.5611i
  Column 5
   0.1439
   0.3060 + 0.0411i
   0.4681 + 0.2306i
   0.4098 + 0.3832i
   0.1819 + 0.5136i
info =
           0
```