NAG Toolbox

nag_lapack_zhbevd (f08hq)

1 Purpose

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

3 Description

nag_lapack_zhbevd (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

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

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 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** – CHARACTER(1)

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** – CHARACTER(1)

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** – INTEGER

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 (KIND=nag_wp) array

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

The second dimension of the array **ab** 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 **ab**$(k_d + 1 + i - j, j)$ for $\max(1, j - k_d) \leq i \leq j$;

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

5.2 Optional Input Parameters

1: **n** – INTEGER

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 Output Parameters

1: **ab**$(ldab,:)$ – COMPLEX (KIND=nag_wp) array

The first dimension of the array **ab** will be **kd** $+ 1$.

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

ab stores values generated during the reduction to tridiagonal form.

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

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

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

The eigenvalues of the matrix A in ascending order.

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

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

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

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

If **job** = 'V', **z** stores 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** – 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: **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 $i \bmod (\mathbf{n}+1)$.

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 nag_lapack_dsbevd (f08hc).

9 Example

This example computes all the eigenvalues and eigenvectors of the Hermitian band matrix A, where

$$A = \begin{pmatrix} 1+0i & 2-1i & 3-1i & 0+0i & 0+0i \\ 2+1i & 2+0i & 3-2i & 4-2i & 0+0i \\ 3+1i & 3+2i & 3+0i & 4-3i & 5-3i \\ 0+0i & 4+2i & 4+3i & 4+0i & 5-4i \\ 0+0i & 0+0i & 5+3i & 5+4i & 5+0i \end{pmatrix}.$$

9.1 Program Text

```
function f08hq_example

fprintf('f08hq example results\n\n');

% Hermitian band matrix A, stored on symmetric banded format
uplo = 'L';
kd = nag_int(2);
ab = [1 + 0i,  2 + 0i,  3 + 0i,  4 + 0i,  5 + 0i;
      2 + 1i,  3 + 2i,  4 + 3i,  5 + 4i,  0 + 0i;
      3 + 1i,  4 + 2i,  5 + 3i,  0 + 0i,  0 + 0i];

% All eigenvalues and eigenvectors of A
job = 'V';
[abf, w, z, info] = f08hq( ...
                           job, uplo, kd, ab);

% Normalize: largest elements are real
for i = 1:5
  [~,k] = max(abs(real(z(:,i)))+abs(imag(z(:,i))));
  z(:,i) = z(:,i)*conj(z(k,i))/abs(z(k,i));
end

disp('Eigenvalues');
disp(w');
[ifail] = x04da( ...
                 'General', ' ', z, 'Eigenvectors');
```

9.2 Program Results

```
f08hq example results

Eigenvalues
   -6.4185   -1.4094    1.4421    4.4856   16.9002

 Eigenvectors
          1       2       3       4       5
 1  -0.2534 -0.4188 -0.2560  0.4767  0.1051
    -0.0538  0.4797  0.3721 -0.2748 -0.0983

 2  -0.0662 -0.0122  0.5344  0.5524  0.2516
     0.4301 -0.3529  0.0000  0.0000 -0.1789

 3   0.5274  0.4621 -0.4245  0.2076  0.4994
     0.0000  0.0000  0.0915 -0.0660 -0.1513

 4   0.1061 -0.1642  0.4964 -0.1379  0.5611
    -0.4981  0.3146 -0.1546  0.1026  0.0000

 5  -0.4519 -0.0360 -0.1979 -0.2651  0.4837
     0.0424 -0.3593 -0.1114 -0.4948  0.2509
```