

## NAG Toolbox

### nag\_lapack\_zhbgvx (f08up)

## 1 Purpose

nag\_lapack\_zhbgvx (f08up) computes selected the eigenvalues and, optionally, the eigenvectors of a complex generalized Hermitian-definite banded eigenproblem, of the form

$$Az = \lambda Bz,$$

where  $A$  and  $B$  are Hermitian and banded, and  $B$  is also positive definite. Eigenvalues and eigenvectors can be selected by specifying either all eigenvalues, a range of values or a range of indices for the desired eigenvalues.

## 2 Syntax

```
[ab, bb, q, m, w, z, jfail, info] = nag_lapack_zhbgvx(jobz, range, uplo, ka, kb,
ab, bb, vl, vu, il, iu, abstol, 'n', n)
[ab, bb, q, m, w, z, jfail, info] = f08up(jobz, range, uplo, ka, kb, ab, bb, vl,
vu, il, iu, abstol, 'n', n)
```

## 3 Description

The generalized Hermitian-definite band problem

$$Az = \lambda Bz$$

is first reduced to a standard band Hermitian problem

$$Cx = \lambda x,$$

where  $C$  is a Hermitian band matrix, using Wilkinson's modification to Crawford's algorithm (see Crawford (1973) and Wilkinson (1977)). The Hermitian eigenvalue problem is then solved for the required eigenvalues and eigenvectors, and the eigenvectors are then backtransformed to the eigenvectors of the original problem.

The eigenvectors are normalized so that

$$z^H A z = \lambda \quad \text{and} \quad z^H B z = 1.$$

## 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>

Crawford C R (1973) Reduction of a band-symmetric generalized eigenvalue problem *Comm. ACM* **16** 41–44

Demmel J W and Kahan W (1990) Accurate singular values of bidiagonal matrices *SIAM J. Sci. Statist. Comput.* **11** 873–912

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

Wilkinson J H (1977) Some recent advances in numerical linear algebra *The State of the Art in Numerical Analysis* (ed D A H Jacobs) Academic Press

## 5 Parameters

### 5.1 Compulsory Input Parameters

1: **jobz** – CHARACTER(1)

Indicates whether eigenvectors are computed.

**jobz** = 'N'

Only eigenvalues are computed.

**jobz** = 'V'

Eigenvalues and eigenvectors are computed.

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

2: **range** – CHARACTER(1)

If **range** = 'A', all eigenvalues will be found.

If **range** = 'V', all eigenvalues in the half-open interval (**vl**, **vu**] will be found.

If **range** = 'I', the **ilth** to **iuth** eigenvalues will be found.

*Constraint:* **range** = 'A', 'V' or 'I'.

3: **uplo** – CHARACTER(1)

If **uplo** = 'U', the upper triangles of *A* and *B* are stored.

If **uplo** = 'L', the lower triangles of *A* and *B* are stored.

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

4: **ka** – INTEGER

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

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

*Constraint:* **ka**  $\geq 0$ .

5: **kb** – INTEGER

If **uplo** = 'U', the number of superdiagonals,  $k_b$ , of the matrix *B*.

If **uplo** = 'L', the number of subdiagonals,  $k_b$ , of the matrix *B*.

*Constraint:* **ka**  $\geq kb \geq 0$ .

6: **ab**(*ldab*, :) – COMPLEX (KIND=nag\_wp) array

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

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

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

The matrix is stored in rows 1 to  $k_a + 1$ , 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**( $k_a + 1 + i - j, j$ ) for  $\max(1, j - k_a) \leq i \leq j$ ;

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  $j \leq i \leq \min(n, j + k_a)$ .

7: **bb**(*ldb*, :) – COMPLEX (KIND=nag\_wp) array

The first dimension of the array **bb** must be at least **kb** + 1.

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

The upper or lower triangle of the  $n$  by  $n$  Hermitian positive definite band matrix  $B$ .

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

if **uplo** = 'U', the elements of the upper triangle of  $B$  within the band must be stored with element  $B_{ij}$  in **bb**( $k_b + 1 + i - j, j$ ) for  $\max(1, j - k_b) \leq i \leq j$ ;

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

- 8: **vl** – REAL (KIND=nag\_wp)  
 9: **vu** – REAL (KIND=nag\_wp)

If **range** = 'V', the lower and upper bounds of the interval to be searched for eigenvalues.

If **range** = 'A' or 'T', **vl** and **vu** are not referenced.

*Constraint:* if **range** = 'V', **vl** < **vu**.

- 10: **il** – INTEGER  
 11: **iu** – INTEGER

If **range** = 'I', the indices (in ascending order) of the smallest and largest eigenvalues to be returned.

If **range** = 'A' or 'V', **il** and **iu** are not referenced.

*Constraints:*

if **range** = 'I' and **n** = 0, **il** = 1 and **iu** = 0;  
 if **range** = 'I' and **n** > 0,  $1 \leq \mathbf{il} \leq \mathbf{iu} \leq \mathbf{n}$ .

- 12: **abstol** – REAL (KIND=nag\_wp)

The absolute error tolerance for the eigenvalues. An approximate eigenvalue is accepted as converged when it is determined to lie in an interval  $[a, b]$  of width less than or equal to

$$\mathbf{abstol} + \epsilon \max(|a|, |b|),$$

where  $\epsilon$  is the **machine precision**. If **abstol** is less than or equal to zero, then  $\epsilon \|T\|_1$  will be used in its place, where  $T$  is the tridiagonal matrix obtained by reducing  $C$  to tridiagonal form. Eigenvalues will be computed most accurately when **abstol** is set to twice the underflow threshold  $2 \times \text{x02am}( )$ , not zero. If this function returns with **info** = 1 to **n**, indicating that some eigenvectors did not converge, try setting **abstol** to  $2 \times \text{x02am}( )$ . See Demmel and Kahan (1990).

## 5.2 Optional Input Parameters

- 1: **n** – INTEGER

*Default:* the second dimension of the arrays **ab**, **bb**.

$n$ , the order of the matrices  $A$  and  $B$ .

*Constraint:*  $n \geq 0$ .

## 5.3 Output Parameters

- 1: **ab**(*ldab*, :) – COMPLEX (KIND=nag\_wp) array

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

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

The contents of **ab** are overwritten.

2: **bb**(*ldbb*,:) – COMPLEX (KIND=nag\_wp) array

The first dimension of the array **bb** will be **kb** + 1.

The second dimension of the array **bb** will be max(1, **n**).

The factor  $S$  from the split Cholesky factorization  $B = S^H S$ , as returned by nag\_lapack\_zpbstf (f08ut).

3: **q**(*ldq*,:) – COMPLEX (KIND=nag\_wp) array

The first dimension, *ldq*, of the array **q** will be

if **jobz** = 'V', *ldq* = max(1, **n**);  
otherwise *ldq* = 1.

The second dimension of the array **q** will be max(1, **n**) if **jobz** = 'V' and 1 otherwise.

If **jobz** = 'V', the  $n$  by  $n$  matrix,  $Q$  used in the reduction of the standard form, i.e.,  $Cx = \lambda x$ , from symmetric banded to tridiagonal form.

If **jobz** = 'N', **q** is not referenced.

4: **m** – INTEGER

The total number of eigenvalues found.  $0 \leq \mathbf{m} \leq \mathbf{n}$ .

If **range** = 'A', **m** = **n**.

If **range** = 'I', **m** = **iu** – **il** + 1.

5: **w(n)** – REAL (KIND=nag\_wp) array

The eigenvalues in ascending order.

6: **z**(*ldz*,:) – COMPLEX (KIND=nag\_wp) array

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

if **jobz** = 'V', *ldz* = max(1, **n**);  
otherwise *ldz* = 1.

The second dimension of the array **z** will be max(1, **n**) if **jobz** = 'V' and 1 otherwise.

If **jobz** = 'V', **z** contains the matrix  $Z$  of eigenvectors, with the  $i$ th column of  $Z$  holding the eigenvector associated with **w**(*i*). The eigenvectors are normalized so that  $Z^H B Z = I$ .

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

7: **jfail**(:) – INTEGER array

The dimension of the array **jfail** will be max(1, **n**)

If **jobz** = 'V', then

if **info** = 0, the first **m** elements of **jfail** are zero;

if **info** = 1 to **n**, **jfail** contains the indices of the eigenvectors that failed to converge.

If **jobz** = 'N', **jfail** is not referenced.

8: **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: **jobz**, 2: **range**, 3: **uplo**, 4: **n**, 5: **ka**, 6: **kb**, 7: **ab**, 8: **ldab**, 9: **bb**, 10: **ldbb**, 11: **q**, 12: **ldq**, 13: **vl**, 14: **vu**, 15: **il**, 16: **iu**, 17: **abstol**, 18: **m**, 19: **w**, 20: **z**, 21: **ldz**, 22: **work**, 23: **rwork**, 24: **iwork**, 25: **jfail**, 26: **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** = 1 to **n** (*warning*)

If **info** =  $i$ , then  $i$  eigenvectors failed to converge. Their indices are stored in array **jfail**. Please see **abstol**.

**info** > **n**

`nag_lapack_dpbstf` (f08uf) returned an error code; i.e., if **info** =  $\mathbf{n} + i$ , for  $1 \leq i \leq \mathbf{n}$ , then the leading minor of order  $i$  of  $B$  is not positive definite. The factorization of  $B$  could not be completed and no eigenvalues or eigenvectors were computed.

## 7 Accuracy

If  $B$  is ill-conditioned with respect to inversion, then the error bounds for the computed eigenvalues and vectors may be large, although when the diagonal elements of  $B$  differ widely in magnitude the eigenvalues and eigenvectors may be less sensitive than the condition of  $B$  would suggest. See Section 4.10 of Anderson *et al.* (1999) for details of the error bounds.

## 8 Further Comments

The total number of floating-point operations is proportional to  $n^3$  if **jobz** = 'V' and **range** = 'A', and assuming that  $n \gg k_a$ , is approximately proportional to  $n^2 k_a$  if **jobz** = 'N'. Otherwise the number of floating-point operations depends upon the number of eigenvectors computed.

The real analogue of this function is `nag_lapack_dsbgvx` (f08ub).

## 9 Example

This example finds the eigenvalues in the half-open interval (0.0, 2.0], and corresponding eigenvectors, of the generalized band Hermitian eigenproblem  $Az = \lambda Bz$ , where

$$A = \begin{pmatrix} -1.13 & 1.94 - 2.10i & -1.40 + 0.25i & 0 \\ 1.94 + 2.10i & -1.91 & -0.82 - 0.89i & -0.67 + 0.34i \\ -1.40 - 0.25i & -0.82 + 0.89i & -1.87 & -1.10 - 0.16i \\ 0 & -0.67 - 0.34i & -1.10 + 0.16i & 0.50 \end{pmatrix}$$

and

$$B = \begin{pmatrix} 9.89 & 1.08 - 1.73i & 0 & 0 \\ 1.08 + 1.73i & 1.69 & -0.04 + 0.29i & 0 \\ 0 & -0.04 - 0.29i & 2.65 & -0.33 + 2.24i \\ 0 & 0 & -0.33 - 2.24i & 2.17 \end{pmatrix}.$$

## 9.1 Program Text

```

function f08up_example

fprintf('f08up example results\n\n');

% Hermitian banded matrices A and B stored in symmetric banded format
uplo = 'U';
ka = nag_int(2);
ab = [ 0,           0 + 0i,   -1.40 + 0.25i, -0.67 + 0.34i;
       0 + 0i,   1.94 - 2.10i, -0.82 - 0.89i, -1.10 - 0.16i;
      -1.13 + 0i, -1.91 + 0i,   -1.87 + 0i,     0.50 + 0i];
kb = nag_int(1);
bb = [ 0,           1.08 - 1.73i, -0.04 + 0.29i, -0.33 + 2.24i;
      9.89 + 0i,   1.69 + 0i,     2.65 + 0i,     2.17 + 0i];

% Eigenvalues in range [0,2] and corresponding eigenvectors of Ax = lambda Bx
jobz = 'Vectors';
range = 'Values in range';
vl = 0;             vu = 2;
il = nag_int(0);  iu = il;
abstol = 0;
[~,~,~,m,w,z,jfail,info] = ...
f08up(..., ...
jobz, range, uplo, ka, kb, ab, bb, vl, vu, il, iu, abstol);

disp('Selected eigenvalues');
disp(w(1:m));

% Normalize vectors, largest element is real and positive.
for i = 1:m
    [~,k] = max(abs(real(z(:,i)))+abs(imag(z(:,i))));
    z(:,i) = z(:,i)*conj(z(k,i))/abs(z(k,i));
end

disp('Corresponding eigenvectors');
disp(z(:,1:m));

```

## 9.2 Program Results

f08up example results

Selected eigenvalues  
0.1603    1.7712

Corresponding eigenvectors  
-0.1026 - 0.1614i    0.0068 + 0.0492i  
-0.0028 - 0.1738i    -0.4307 + 0.2705i  
-0.0466 + 0.0890i    -0.1123 - 0.9664i  
0.3586 + 0.0000i    1.3241 + 0.0000i

---