NAG Fortran Library Routine Document

F08UQF (ZHBGVD)

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

F08UQF (ZHBGVD) computes all 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. If eigenvectors are desired, it uses a divide-and-conquer algorithm.

2 Specification

```
 SUBROUTINE F08UQF (JOBZ, UPLO, N, KA, KB, AB, LDAB, BB, LDBB, W, Z, LDZ,
1                   WORK, LWORK, RWORK, LRWORK, IWORK, LIWORK, INFO)
 INTEGER            N, KA, KB, LDAB, LDBB, LDZ, LWORK, LRWORK, IWORK(*),
1                   LIWORK, INFO
 double precision   W(*), RWORK(*)
 complex*16         AB(LDAB,*), BB(LDBB,*), Z(LDZ,*), WORK(*)
 CHARACTER*1        JOBZ, UPLO
```

The routine may be called by its LAPACK name ***zhbgvd***.

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 eigenvalues and the eigenvectors, if required, which are then backtransformed to the eigenvectors of the original problem.

The eigenvectors are normalized so that the matrix of eigenvectors, Z, satisfies

$$Z^H AZ = \Lambda \quad \text{and} \quad Z^H BZ = I,$$

where Λ is the diagonal matrix whose diagonal elements are the eigenvalues.

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

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

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

1: JOBZ – CHARACTER*1 *Input*

On entry: if JOBZ = 'N', compute eigenvalues only.

If JOBZ = 'V', compute eigenvalues and eigenvectors.

Constraint: JOBZ = 'N' or 'V'.

2: UPLO – CHARACTER*1 *Input*

On entry: if UPLO = 'U', the upper triangles of A and B are stored.

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

3: N – INTEGER *Input*

On entry: n, the order of the matrices A and B.

Constraint: $\mathrm{N} \geq 0$.

4: KA – INTEGER *Input*

On entry: ka, the number of super-diagonals of the matrix A if UPLO = 'U', or the number of sub-diagonals if UPLO = 'L'.

Constraint: $\mathrm{KA} \geq 0$.

5: KB – INTEGER *Input*

On entry: kb, the number of super-diagonals of the matrix B if UPLO = 'U', or the number of sub-diagonals if UPLO = 'L'.

Constraint: $\mathrm{KB} \geq 0$.

6: AB(LDAB,$*$) – ***complex*16*** array *Input/Output*

Note: the second dimension of the array AB must be at least $\max(1, \mathrm{N})$.

On entry: the upper or lower triangle of the symmetric band matrix A, stored in the first $ka + 1$ rows of the array. The jth column of A is stored in the jth column of the array AB as follows:

if UPLO = 'U', $\mathrm{AB}(ka + 1 + i - j, j) = a_{ij}$ for $\max(1, j - ka) \leq i \leq j$;
if UPLO = 'L', $\mathrm{AB}(1 + i - j, j) = a_{ij}$ for $j \leq i \leq \min(n, j + ka)$.

On exit: the contents of AB are destroyed.

7: LDAB – INTEGER *Input*

On entry: the first dimension of the array AB as declared in the (sub)program from which F08UQF (ZHBGVD) is called.

Constraint: $\mathrm{LDAB} \geq \mathrm{KA} + 1$.

8: BB(LDBB,$*$) – ***complex*16*** array *Input/Output*

Note: the second dimension of the array BB must be at least $\max(1, \mathrm{N})$.

On entry: the upper or lower triangle of the Hermitian band matrix B, stored in the first $kb + 1$ rows of the array. The jth column of B is stored in the jth column of the array BB as follows:

if UPLO = 'U', $\mathrm{BB}(kb + 1 + i - j, j) = b_{ij}$ for $\max(1, j - kb) \leq i \leq j$;
if UPLO = 'L', $\mathrm{BB}(1 + i - j, j) = b_{ij}$ for $j \leq i \leq \min(n, j + kb)$.

On exit: the factor S from the split Cholesky factorization $B = S^H S$, as returned by F08UTF (ZPBSTF).

9: LDBB – INTEGER *Input*

On entry: the first dimension of the array BB as declared in the (sub)program from which F08UQF (ZHBGVD) is called.

Constraint: $\mathrm{LDBB} \geq \mathrm{KB} + 1$.

10: $\mathrm{W}(*)$ – ***double precision*** array *Output*

Note: the dimension of the array W must be at least $\max(1, \mathrm{N})$.

On exit: if $\mathrm{INFO} = 0$, the eigenvalues in ascending order.

11: Z(LDZ,$*$) – ***complex*16*** array *Output*

Note: the second dimension of the array Z must be at least $\max(1, \mathrm{N})$.

On exit: if JOBZ = 'V', then if $\mathrm{INFO} = 0$, Z contains the matrix Z of eigenvectors, with the ith column of Z holding the eigenvector associated with $\mathrm{W}(i)$. The eigenvectors are normalized so that $Z^H BZ = I$.

If JOBZ = 'N', Z is not referenced.

12: LDZ – INTEGER *Input*

On entry: the first dimension of the array Z as declared in the (sub)program from which F08UQF (ZHBGVD) is called.

Constraints:

if JOBZ = 'V', $\mathrm{LDZ} \geq \max(1, \mathrm{N})$;
$\mathrm{LDZ} \geq 1$ otherwise.

13: $\mathrm{WORK}(*)$ – ***complex*16*** array *Workspace*

Note: the dimension of the array WORK must be at least $\max(1, \mathrm{LWORK})$.

On exit: if $\mathrm{INFO} = 0$, $\mathrm{WORK}(1)$ returns the optimal LWORK.

14: LWORK – INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F08UQF (ZHBGVD) is called.

If $\mathrm{LWORK} = -1$, a workspace query is assumed; the routine only calculates the optimal sizes of the WORK, RWORK and IWORK arrays, returns these values as the first entries of the WORK, RWORK and IWORK arrays, and no error message related to LWORK, LRWORK or LIWORK is issued.

Constraints:

if $\mathrm{N} \leq 1$, $\mathrm{LWORK} \geq 1$;
if JOBZ = 'N' and $\mathrm{N} > 1$, $\mathrm{LWORK} \geq \max(1, \mathrm{N})$;
if JOBZ = 'V' and $\mathrm{N} > 1$, $\mathrm{LWORK} \geq \max\left(1, \mathrm{N}^2\right)$.

15: $\mathrm{RWORK}(*)$ – ***double precision*** array *Workspace*

Note: the dimension of the array RWORK must be at least $\max(1, \mathrm{LRWORK})$.

On exit: if $\mathrm{INFO} = 0$, $\mathrm{RWORK}(1)$ returns the optimal LRWORK.

16: LRWORK – INTEGER *Input*

On entry: the first dimension of the array RWORK as declared in the (sub)program from which F08UQF (ZHBGVD) is called.

If $\mathrm{LRWORK} = -1$, a workspace query is assumed; the routine only calculates the optimal sizes of the WORK, RWORK and IWORK arrays, returns these values as the first entries of the WORK, RWORK and IWORK arrays, and no error message related to LWORK, LRWORK or LIWORK is issued.

Constraints:

if $\mathrm{N} \leq 1$, $\mathrm{LRWORK} \geq 1$;
if JOBZ = 'N' and $\mathrm{N} > 1$, $\mathrm{LRWORK} \geq \max(1, \mathrm{N})$;
if JOBZ = 'V' and $\mathrm{N} > 1$, $\mathrm{LRWORK} \geq 1 + 5 \times \mathrm{N} + 2 \times \mathrm{N}^2$.

17: IWORK($*$) – INTEGER array *Workspace*

Note: the dimension of the array IWORK must be at least $\max(1, \mathrm{LIWORK})$.

On exit: if $\mathrm{INFO} = 0$, IWORK(1) returns the optimal LIWORK.

18: LIWORK – INTEGER *Input*

On entry: the dimension of the array IWORK as declared in the (sub)program from which F08UQF (ZHBGVD) is called.

If $\mathrm{LIWORK} = -1$, a workspace query is assumed; the routine only calculates the optimal sizes of the WORK, RWORK and IWORK arrays, returns these values as the first entries of the WORK, RWORK and IWORK arrays, and no error message related to LWORK, LRWORK or LIWORK is issued.

Constraints:

if JOBZ = 'N' or $\mathrm{N} \leq 1$, $\mathrm{LIWORK} \geq 1$;
if JOBZ = 'V' and $\mathrm{N} > 1$, $\mathrm{LIWORK} \geq 3 + 5 \times \mathrm{N}$.

19: INFO – INTEGER *Output*

On exit: $\mathrm{INFO} = 0$ unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

$\mathrm{INFO} < 0$

If $\mathrm{INFO} = -i$, the ith argument had an illegal value.

$\mathrm{INFO} > 0$

If $\mathrm{INFO} = i$ and $i \leq \mathrm{N}$, the algorithm failed to converge: i off-diagonal elements of an intermediate tridiagonal form did not converge to zero.

If $\mathrm{INFO} = i$ and $i > \mathrm{N}$, if $\mathrm{INFO} = \mathrm{N} + i$, for $1 \leq i \leq \mathrm{N}$, then F08UTF (ZPBSTF) returned '$\mathrm{INFO} = i$: 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, assuming that $n \gg k_a$, is approximately proportional to $n^2 k_a$ otherwise.

The real analogue of this routine is F08UCF (DSBGVD).

9 Example

To find all the eigenvalues 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

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     F08UQF Example Program Text
*     Mark 21.  NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, KAMAX, KBMAX
      PARAMETER        (NMAX=20,KAMAX=5,KBMAX=5)
      INTEGER          LDAB, LDBB, LIWORK, LRWORK, LWORK
      PARAMETER        (LDAB=KAMAX+1,LDBB=KBMAX+1,LIWORK=1,LRWORK=NMAX,
     +                 LWORK=NMAX)
      CHARACTER        UPLO
      PARAMETER        (UPLO='U')
*     .. Local Scalars ..
      INTEGER          I, INFO, J, KA, KB, N
*     .. Local Arrays ..
      COMPLEX *16      AB(LDAB,NMAX), BB(LDBB,NMAX), DUMMY(1,1),
     +                 WORK(LWORK)
      DOUBLE PRECISION RWORK(LRWORK), W(NMAX)
      INTEGER          IWORK(LIWORK)
*     .. External Subroutines ..
      EXTERNAL         ZHBGVD
*     .. Intrinsic Functions ..
      INTRINSIC        MAX, MIN
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F08UQF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N, KA, KB
      IF (N.LE.NMAX .AND. KA.LE.KAMAX .AND. KB.LE.KBMAX) THEN
*
*        Read the upper or lower triangular parts of the matrices A and
*        B from data file
*
         IF (UPLO.EQ.'U') THEN
            READ (NIN,*) ((AB(KA+1+I-J,J),J=I,MIN(N,I+KA)),I=1,N)
            READ (NIN,*) ((BB(KB+1+I-J,J),J=I,MIN(N,I+KB)),I=1,N)
         ELSE IF (UPLO.EQ.'L') THEN
            READ (NIN,*) ((AB(1+I-J,J),J=MAX(1,I-KA),I),I=1,N)
            READ (NIN,*) ((BB(1+I-J,J),J=MAX(1,I-KB),I),I=1,N)
         END IF
*
*        Solve the generalized Hermitian band eigenvalue problem
*        A*x = lambda*B*x
*
```

```
         CALL ZHBGVD('No vectors',UPLO,N,KA,KB,AB,LDAB,BB,LDBB,W,DUMMY,
     +               1,WORK,LWORK,RWORK,LRWORK,IWORK,LIWORK,INFO)
*
         IF (INFO.EQ.0) THEN
*
*           Print solution
*
            WRITE (NOUT,*) 'Eigenvalues'
            WRITE (NOUT,99999) (W(J),J=1,N)
         ELSE IF (INFO.GT.N .AND. INFO.LE.2*N) THEN
            I = INFO - N
            WRITE (NOUT,99998) 'The leading minor of order ', I,
     +        ' of B is not positive definite'
         ELSE
            WRITE (NOUT,99997) 'Failure in ZHBGVD. INFO =', INFO
         END IF
      ELSE
         WRITE (NOUT,*) 'NMAX too small'
      END IF
      STOP
*
99999 FORMAT (3X,(6F11.4))
99998 FORMAT (1X,A,I4,A)
99997 FORMAT (1X,A,I4)
      END
```

9.2 Program Data

```
F08UQF Example Program Data

   4             2             1                          :Values of N, KA and KB

 (-1.13, 0.00) ( 1.94,-2.10) (-1.40, 0.25)
               (-1.91, 0.00) (-0.82,-0.89) (-0.67, 0.34)
                             (-1.87, 0.00) (-1.10,-0.16)
                                           ( 0.50, 0.00) :End of matrix A

 ( 9.89, 0.00) ( 1.08,-1.73)
               ( 1.69, 0.00) (-0.04, 0.29)
                             ( 2.65, 0.00) (-0.33, 2.24)
                                           ( 2.17, 0.00) :End of matrix B
```

9.3 Program Results

```
 F08UQF Example Program Results

 Eigenvalues
       -6.6089    -2.0416     0.1603     1.7712
```