NAG Library Routine Document

F08WTF (ZGGHD3)

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

F08WTF (ZGGHD3) reduces a pair of complex matrices (A, B), where B is upper triangular, to the generalized upper Hessenberg form using unitary transformations.

2 Specification

```
SUBROUTINE F08WTF (COMPQ, COMPZ, N, ILO, IHI, A, LDA, B, LDB, Q, LDQ, Z,    &
                   LDZ, WORK, LWORK, INFO)

INTEGER               N, ILO, IHI, LDA, LDB, LDQ, LDZ, LWORK, INFO
COMPLEX (KIND=nag_wp) A(LDA,*), B(LDB,*), Q(LDQ,*), Z(LDZ,*),               &
                      WORK(max(1,LWORK))
CHARACTER(1)          COMPQ, COMPZ
```

The routine may be called by its LAPACK name ***zgghd3***.

3 Description

F08WTF (ZGGHD3) is usually the third step in the solution of the complex generalized eigenvalue problem

$$Ax = \lambda Bx.$$

The (optional) first step balances the two matrices using F08WVF (ZGGBAL). In the second step, matrix B is reduced to upper triangular form using the QR factorization routine F08ASF (ZGEQRF) and this unitary transformation Q is applied to matrix A by calling F08AUF (ZUNMQR). The driver, F08WQF (ZGGEV3), solves the complex generalized eigenvalue problem by combining all the required steps including those just listed.

F08WTF (ZGGHD3) reduces a pair of complex matrices (A, B), where B is triangular, to the generalized upper Hessenberg form using unitary transformations. This two-sided transformation is of the form

$$Q^{\mathrm{H}}AZ = H,$$
$$Q^{\mathrm{H}}BZ = T$$

where H is an upper Hessenberg matrix, T is an upper triangular matrix and Q and Z are unitary matrices determined as products of Givens rotations. They may either be formed explicitly, or they may be postmultiplied into input matrices Q_1 and Z_1, so that

$$Q_1AZ_1^{\mathrm{H}} = (Q_1Q)H(Z_1Z)^{\mathrm{H}},$$
$$Q_1BZ_1^{\mathrm{H}} = (Q_1Q)T(Z_1Z)^{\mathrm{H}}.$$

4 References

Golub G H and Van Loan C F (2012) *Matrix Computations* (4th Edition) Johns Hopkins University Press, Baltimore

Moler C B and Stewart G W (1973) An algorithm for generalized matrix eigenproblems *SIAM J. Numer. Anal.* **10** 241–256

5 Arguments

1: COMPQ – CHARACTER(1) *Input*

On entry: specifies the form of the computed unitary matrix Q.

COMPQ = 'N'
: Do not compute Q.

COMPQ = 'I'
: The unitary matrix Q is returned.

COMPQ = 'V'
: Q must contain a unitary matrix Q_1, and the product Q_1Q is returned.

Constraint: COMPQ = 'N', 'I' or 'V'.

2: COMPZ – CHARACTER(1) *Input*

On entry: specifies the form of the computed unitary matrix Z.

COMPZ = 'N'
: Do not compute Z.

COMPZ = 'V'
: Z must contain a unitary matrix Z_1, and the product Z_1Z is returned.

COMPZ = 'I'
: The unitary matrix Z is returned.

Constraint: COMPZ = 'N', 'V' or 'I'.

3: N – INTEGER *Input*

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

Constraint: N $\geq$ 0.

4: ILO – INTEGER *Input*
5: IHI – INTEGER *Input*

On entry: i_{lo} and i_{hi} as determined by a previous call to F08WVF (ZGGBAL). Otherwise, they should be set to 1 and n, respectively.

Constraints:

if N > 0, $1 \leq$ ILO $\leq$ IHI $\leq$ N;
if N $= 0$, ILO $= 1$ and IHI $= 0$.

6: A(LDA, $*$) – COMPLEX (KIND=nag_wp) array *Input/Output*

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

On entry: the matrix A of the matrix pair (A, B). Usually, this is the matrix A returned by F08AUF (ZUNMQR).

On exit: A is overwritten by the upper Hessenberg matrix H.

7: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08WTF (ZGGHD3) is called.

Constraint: LDA $\geq \max(1, \mathrm{N})$.

8: B(LDB, $*$) – COMPLEX (KIND=nag_wp) array *Input/Output*

Note: the second dimension of the array B must be at least max(1, N).

On entry: the upper triangular matrix B of the matrix pair (A, B). Usually, this is the matrix B returned by the QR factorization routine F08ASF (ZGEQRF).

On exit: B is overwritten by the upper triangular matrix T.

9: LDB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F08WTF (ZGGHD3) is called.

Constraint: LDB $\geq$ max(1, N).

10: Q(LDQ, $*$) – COMPLEX (KIND=nag_wp) array *Input/Output*

Note: the second dimension of the array Q must be at least max(1, N) if COMPQ = 'I' or 'V' and at least 1 if COMPQ = 'N'.

On entry: if COMPQ = 'V', Q must contain a unitary matrix Q_1.

If COMPQ = 'N', Q is not referenced.

On exit: if COMPQ = 'I', Q contains the unitary matrix Q.

Iif COMPQ = 'V', Q is overwritten by Q_1Q.

11: LDQ – INTEGER *Input*

On entry: the first dimension of the array Q as declared in the (sub)program from which F08WTF (ZGGHD3) is called.

Constraints:

if COMPQ = 'I' or 'V', LDQ $\geq$ max(1, N);
if COMPQ = 'N', LDQ $\geq$ 1.

12: Z(LDZ, $*$) – COMPLEX (KIND=nag_wp) array *Input/Output*

Note: the second dimension of the array Z must be at least max(1, N) if COMPZ = 'V' or 'I' and at least 1 if COMPZ = 'N'.

On entry: if COMPZ = 'V', Z must contain a unitary matrix Z_1.

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

On exit: if COMPZ = 'I', Z contains the unitary matrix Z.

If COMPZ = 'V', Z is overwritten by Z_1Z.

13: LDZ – INTEGER *Input*

On entry: the first dimension of the array Z as declared in the (sub)program from which F08WTF (ZGGHD3) is called.

Constraints:

if COMPZ = 'V' or 'I', LDZ $\geq$ max(1, N);
if COMPZ = 'N', LDZ $\geq$ 1.

14: WORK(max(1, LWORK)) – COMPLEX (KIND=nag_wp) array *Workspace*

On exit: if INFO = 0, the real part of WORK(1) contains the minimum value of LWORK required for optimal performance.

15: LWORK – INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)routine from which F08WTF (ZGGHD3) is called.

If LWORK $= -1$, a workspace query is assumed; the routine only calculates the optimal size of the WORK array, returns this value as the first entry of the WORK array, and no error message related to LWORK is issued.

Suggested value: for optimal performance, LWORK must generally be larger than the minimum; increase workspace by, say, $nb \times (\mathrm{N} \times 6)$, where nb is the optimal ***block size***.

16: INFO – INTEGER *Output*

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

6 Error Indicators and Warnings

INFO < 0

If INFO $= -i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

The reduction to the generalized Hessenberg form is implemented using unitary transformations which are backward stable.

8 Parallelism and Performance

F08WTF (ZGGHD3) makes calls to BLAS and/or LAPACK routines, which may be threaded within the vendor library used by this implementation. Consult the documentation for the vendor library for further information.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9 Further Comments

This routine is usually followed by F08XSF (ZHGEQZ) which implements the QZ algorithm for computing generalized eigenvalues of a reduced pair of matrices.

The real analogue of this routine is F08WFF (DGGHD3).

10 Example

See Section 10 in F08XSF (ZHGEQZ) and F08YXF (ZTGEVC).