NAG Library Routine Document

F08YFF (DTGEXC)

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

F08YFF (DTGEXC) reorders the generalized Schur factorization of a matrix pair in real generalized Schur form.

2 Specification

```
 SUBROUTINE F08YFF(WANTQ, WANTZ, N, A, LDA, B, LDB, Q, LDQ, Z, LDZ, IFST,
1                  ILST, WORK, LWORK, INFO)
 INTEGER           N, LDA, LDB, LDQ, LDZ, IFST, ILST, LWORK, INFO
 double precision  A(LDA,*), B(LDB,*), Q(LDQ,*), Z(LDZ,*),
1                  WORK(max(1,LWORK))
 LOGICAL           WANTQ, WANTZ
```

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

3 Description

F08YFF (DTGEXC) reorders the generalized real n by n matrix pair (S,T) in real generalized Schur form, so that the diagonal element or block of (S,T) with row index i_1 is moved to row i_2, using an orthogonal equivalence transformation. That is, S and T are factorized as

$$S = \hat{Q}\hat{S}\hat{Z}^{\mathrm{T}}, \qquad T = \hat{Q}\hat{T}\hat{Z}^{\mathrm{T}},$$

where $\left(\hat{S},\hat{T}\right)$ are also in real generalized Schur form.

The pair (S,T) are in real generalized Schur form if S is block upper triangular with 1 by 1 and 2 by 2 diagonal blocks and T is upper triangular as returned, for example, by F08XAF (DGGES), or F08XEF (DHGEQZ) with JOB = 'S'.

If S and T are the result of a generalized Schur factorization of a matrix pair (A,B)

$$A = QSZ^{\mathrm{T}}, \qquad B = QTZ^{\mathrm{T}}$$

then, optionally, the matrices Q and Z can be updated as $Q\hat{Q}$ and $Z\hat{Z}$.

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, Philadelphiahttp://www.netlib.org/lapack/lug

5 Parameters

1: WANTQ – LOGICAL *Input*

On entry: if WANTQ = .TRUE., update the left transformation matrix Q.

If WANTQ = .FALSE., do not update Q.

2: WANTZ – LOGICAL *Input*

On entry: if WANTZ = .TRUE., update the right transformation matrix Z.

If WANTZ = .FALSE., do not update Z.

3: N – INTEGER *Input*

On entry: n, the order of the matrices S and T.

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

4: A(LDA,∗) – ***double precision*** array *Input/Output*

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

On entry: the matrix S in the pair (S, T).

On exit: the updated matrix $\hat{S}$.

5: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08YFF (DTGEXC) is called.

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

6: B(LDB,∗) – ***double precision*** array *Input/Output*

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

On entry: the matrix T, in the pair (S, T).

On exit: the updated matrix $\hat{T}$

7: LDB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F08YFF (DTGEXC) is called.

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

8: Q(LDQ,∗) – ***double precision*** array *Input/Output*

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

On entry: if WANTQ = .TRUE., the orthogonal matrix Q.

On exit: if WANTQ = .TRUE., the updated matrix $Q\hat{Q}$.

If WANTQ = .FALSE., Q is not referenced.

9: LDQ – INTEGER *Input*

On entry: the first dimension of the array Q as declared in the (sub)program from which F08YFF (DTGEXC) is called.

Constraints:

if WANTQ = .TRUE., $\mathrm{LDQ} \geq \max(1, \mathrm{N})$;
otherwise $\mathrm{LDQ} \geq 1$.

10: Z(LDZ,∗) – ***double precision*** array *Input/Output*

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

On entry: if WANTZ = .TRUE., the orthogonal matrix Z.

On exit: if WANTZ = .TRUE., the updated matrix $Z\hat{Z}$.

If WANTZ = .FALSE., Z is not referenced.

11: LDZ – INTEGER *Input*

On entry: the first dimension of the array Z as declared in the (sub)program from which F08YFF (DTGEXC) is called.

Constraints:

if WANTZ = .TRUE., LDZ $\geq \max(1, \mathrm{N})$;
otherwise LDZ ≥ 1.

12: IFST – INTEGER *Input/Output*
13: ILST – INTEGER *Input/Output*

On entry: the indices i_1 and i_2 that specify the reordering of the diagonal blocks of (S, T). The block with row index IFST is moved to row ILST, by a sequence of swapping between adjacent blocks.

On exit: if IFST pointed on entry to the second row of a 2 by 2 block, it is changed to point to the first row; ILST always points to the first row of the block in its final position (which may differ from its input value by $+1$ or -1).

Constraint: $1 \leq$ IFST $\leq$ N and $1 \leq$ ILST $\leq$ N.

14: WORK$(\max(1, \mathrm{LWORK}))$ – ***double precision*** array *Workspace*

On exit: if INFO $= 0$, 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)program from which F08YFF (DTGEXC) is called.

If LWORK $= -1$, a workspace query is assumed; the routine only calculates the minimum 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.

Constraints: if LWORK $\neq -1$,

if N ≤ 1, LWORK ≥ 1;
otherwise LWORK $\geq 4 \times \mathrm{N} + 16$.

.

16: INFO – INTEGER *Output*

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

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

INFO < 0

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

INFO $= 1$

The transformed matrix pair $\left(\hat{S}, \hat{T}\right)$ would be too far from generalized Schur form; the problem is ill-conditioned. (S, T) may have been partially reordered, and ILST points to the first row of the current position of the block being moved.

7 Accuracy

The computed generalized Schur form is nearly the exact generalized Schur form for nearby matrices $(S+E)$ and $(T+F)$, where

$$\|E\|_2 = O\,\epsilon\|S\|_2 \qquad \text{and} \qquad \|F\|_2 = O\,\epsilon\|T\|_2,$$

and ϵ is the ***machine precision***. See Section 4.11 of Anderson *et al.* (1999) for further details of error bounds for the generalized nonsymmetric eigenproblem.

8 Further Comments

The complex analogue of this routine is F08YTF (ZTGEXC).

9 Example

This example exchanges blocks 2 and 1 of the matrix pair (S,T), where

$$S = \begin{pmatrix} 4.0 & 1.0 & 1.0 & 2.0 \\ 0 & 3.0 & 4.0 & 1.0 \\ 0 & 1.0 & 3.0 & 1.0 \\ 0 & 0 & 0 & 6.0 \end{pmatrix} \qquad \text{and} \qquad T = \begin{pmatrix} 2.0 & 1.0 & 1.0 & 3.0 \\ 0 & 1.0 & 2.0 & 1.0 \\ 0 & 0 & 1.0 & 1.0 \\ 0 & 0 & 0 & 2.0 \end{pmatrix}.$$

9.1 Program Text

```
*     F08YFF Example Program Text
*     Mark 21 Release. NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          LDQ, LDZ, NMAX
      PARAMETER        (LDQ=1,LDZ=1,NMAX=8)
      INTEGER          LDA, LDB, LWORK
      PARAMETER        (LDA=NMAX,LDB=NMAX,LWORK=4*NMAX+16)
      LOGICAL          WANTQ, WANTZ
      PARAMETER        (WANTQ=.FALSE.,WANTZ=.FALSE.)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, IFST, ILST, INFO, J, N
*     .. Local Arrays ..
      DOUBLE PRECISION A(LDA,NMAX), B(LDB,NMAX), Q(LDQ,1), WORK(LWORK),
     +                 Z(LDZ,1)
*     .. External Subroutines ..
      EXTERNAL         DTGEXC, X04CAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F08YFF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
*
*        Read A and B from data file
*
         READ (NIN,*) ((A(I,J),J=1,N),I=1,N)
         READ (NIN,*) ((B(I,J),J=1,N),I=1,N)
*
*        Read the row indices
*
         READ (NIN,*) IFST, ILST
*
*        Reorder A and B
*
         CALL DTGEXC(WANTQ,WANTZ,N,A,LDA,B,LDB,Q,LDQ,Z,LDZ,IFST,ILST,
     +               WORK,LWORK,INFO)
         IF (INFO.NE.0) THEN
            WRITE (NOUT,99999) INFO, ILST
            WRITE (NOUT,*)
         END IF
```

```
*
*        Print reordered generalized Schur form
*
         IFAIL = 0
         CALL X04CAF('General',' ',N,N,A,LDA,'Reordered Schur matrix A',
     +               IFAIL)
         WRITE (NOUT,*)
         IFAIL = 0
         CALL X04CAF('General',' ',N,N,B,LDB,'Reordered Schur matrix B',
     +               IFAIL)
*
      ELSE
         WRITE (NOUT,*) 'NMAX too small'
      END IF
*
99999 FORMAT (' Reordering could not be completed. INFO = ',I3,' ILST ',
     +       '= ',I5)
      END
```

9.2 Program Data

```
F08YFF Example Program Data
  4                     :Value of N
  4.0  1.0  1.0  2.0
  0.0  3.0  4.0  1.0
  0.0  1.0  3.0  1.0
  0.0  0.0  0.0  6.0    :End of matrix A
  2.0  1.0  1.0  3.0
  0.0  1.0  2.0  1.0
  0.0  0.0  1.0  1.0
  0.0  0.0  0.0  2.0    :End of matrix B
  2  1                  :Values of IFST and ILST
```

9.3 Program Results

```
 F08YFF Example Program Results

 Reordered Schur matrix A
           1          2          3          4
 1     -4.1926    -1.2591    -2.5578    -0.4520
 2     -0.8712     0.8627     2.7912     1.1383
 3      0.0000     0.0000     4.2426     2.1213
 4      0.0000     0.0000     0.0000     6.0000

 Reordered Schur matrix B
           1          2          3          4
 1     -1.7439     0.0000    -0.7533    -0.0661
 2      0.0000     0.5406     1.8972     1.7308
 3      0.0000     0.0000     2.1213     2.8284
 4      0.0000     0.0000     0.0000     2.0000
```