

NAG Library Routine Document

F08YYF (ZTGSNA)

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

F08YYF (ZTGSNA) estimates condition numbers for specified eigenvalues and/or eigenvectors of a complex matrix pair in generalized Schur form.

2 Specification

```

SUBROUTINE F08YYF (JOB, HOWMNY, SELECT, N, A, LDA, B, LDB, VL, LDVL, VR, &
                  LDVR, S, DIF, MM, M, WORK, LWORK, IWORK, INFO)
INTEGER          N, LDA, LDB, LDVL, LDVR, MM, M, LWORK, IWORK(*), &
                  INFO
REAL (KIND=nag_wp) S(*), DIF(*)
COMPLEX (KIND=nag_wp) A(LDA,*), B(LDB,*), VL(LDVL,*), VR(LDVR,*), &
                  WORK(max(1,LWORK))
LOGICAL          SELECT(*)
CHARACTER(1)     JOB, HOWMNY

```

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

3 Description

F08YYF (ZTGSNA) estimates condition numbers for specified eigenvalues and/or right eigenvectors of an n by n matrix pair (S, T) in generalized Schur form. The routine actually returns estimates of the reciprocals of the condition numbers in order to avoid possible overflow.

The pair (S, T) are in generalized Schur form if S and T are upper triangular as returned, for example, by F08XNF (ZGGES) or F08XPF (ZGGESX), or F08XSF (ZHGEQZ) with $JOB = 'S'$. The diagonal elements define the generalized eigenvalues (α_i, β_i) , for $i = 1, 2, \dots, n$, of the pair (S, T) and the eigenvalues are given by

$$\lambda_i = \alpha_i / \beta_i,$$

so that

$$\beta_i S x_i = \alpha_i T x_i \quad \text{or} \quad S x_i = \lambda_i T x_i,$$

where x_i is the corresponding (right) eigenvector.

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

$$A = Q S Z^H, \quad B = Q T Z^H$$

then the eigenvalues and condition numbers of the pair (S, T) are the same as those of the pair (A, B) .

Let $(\alpha, \beta) \neq (0, 0)$ be a simple generalized eigenvalue of (A, B) . Then the reciprocal of the condition number of the eigenvalue $\lambda = \alpha / \beta$ is defined as

$$s(\lambda) = \frac{\left(|y^H A x|^2 + |y^H B x|^2 \right)^{1/2}}{\left(\|x\|_2 \|y\|_2 \right)},$$

where x and y are the right and left eigenvectors of (A, B) corresponding to λ . If both α and β are zero, then (A, B) is singular and $s(\lambda) = -1$ is returned.

If U and V are unitary transformations such that

$$U^H(A, B)V = (S, T) = \begin{pmatrix} \alpha & * \\ 0 & S_{22} \end{pmatrix} \begin{pmatrix} \beta & * \\ 0 & T_{22} \end{pmatrix},$$

where S_{22} and T_{22} are $(n-1)$ by $(n-1)$ matrices, then the reciprocal condition number is given by

$$\text{Dif}(x) \equiv \text{Dif}(y) = \text{Dif}((\alpha, \beta), (S_{22}, T_{22})) = \sigma_{\min}(Z),$$

where $\sigma_{\min}(Z)$ denotes the smallest singular value of the $2(n-1)$ by $2(n-1)$ matrix

$$Z = \begin{pmatrix} \alpha \otimes I & -1 \otimes S_{22} \\ \beta \otimes I & -1 \otimes T_{22} \end{pmatrix}$$

and \otimes is the Kronecker product.

See Sections 2.4.8 and 4.11 of Anderson *et al.* (1999) and Kågström and Poromaa (1996) for further details and information.

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>

Kågström B and Poromaa P (1996) LAPACK-style algorithms and software for solving the generalized Sylvester equation and estimating the separation between regular matrix pairs *ACM Trans. Math. Software* **22** 78–103

5 Parameters

- 1: JOB – CHARACTER(1) *Input*
On entry: indicates whether condition numbers are required for eigenvalues and/or eigenvectors.
 JOB = 'E'
 Condition numbers for eigenvalues only are computed.
 JOB = 'V'
 Condition numbers for eigenvectors only are computed.
 JOB = 'B'
 Condition numbers for both eigenvalues and eigenvectors are computed.
Constraint: JOB = 'E', 'V' or 'B'.
- 2: HOWMNY – CHARACTER(1) *Input*
On entry: indicates how many condition numbers are to be computed.
 HOWMNY = 'A'
 Condition numbers for all eigenpairs are computed.
 HOWMNY = 'S'
 Condition numbers for selected eigenpairs (as specified by SELECT) are computed.
Constraint: HOWMNY = 'A' or 'S'.
- 3: SELECT(*) – LOGICAL array *Input*
Note: the dimension of the array SELECT must be at least $\max(1, N)$ if HOWMNY = 'S', and at least 1 otherwise.
On entry: specifies the eigenpairs for which condition numbers are to be computed if HOWMNY = 'S'. To select condition numbers for the eigenpair corresponding to the eigenvalue λ_j , SELECT(j) must be set to .TRUE..

- If HOWMNY = 'A', SELECT is not referenced.
- 4: N – INTEGER *Input*
On entry: n , the order of the matrix pair (S, T) .
Constraint: $N \geq 0$.
- 5: A(LDA, *) – COMPLEX (KIND=nag_wp) array *Input*
Note: the second dimension of the array A must be at least N.
On entry: the upper triangular matrix S .
- 6: LDA – INTEGER *Input*
On entry: the first dimension of the array A as declared in the (sub)program from which F08YYF (ZTGSNA) is called.
Constraint: $LDA \geq N$.
- 7: B(LDB, *) – COMPLEX (KIND=nag_wp) array *Input*
Note: the second dimension of the array B must be at least N.
On entry: the upper triangular matrix T .
- 8: LDB – INTEGER *Input*
On entry: the first dimension of the array B as declared in the (sub)program from which F08YYF (ZTGSNA) is called.
Constraint: $LDB \geq N$.
- 9: VL(LDVL, *) – COMPLEX (KIND=nag_wp) array *Input*
Note: the second dimension of the array VL must be at least $\max(1, MM)$ if JOB = 'E' or 'B', and at least 1 otherwise.
On entry: if JOB = 'E' or 'B', VL must contain left eigenvectors of (S, T) , corresponding to the eigenpairs specified by HOWMNY and SELECT. The eigenvectors must be stored in consecutive columns of VL, as returned by F08WNF (ZGGEV) or F08YXF (ZTGEVC).
 If JOB = 'V', VL is not referenced.
- 10: LDVL – INTEGER *Input*
On entry: the first dimension of the array VL as declared in the (sub)program from which F08YYF (ZTGSNA) is called.
Constraints:
 if JOB = 'E' or 'B', $LDVL \geq \max(1, N)$;
 otherwise $LDVL \geq 1$.
- 11: VR(LDVR, *) – COMPLEX (KIND=nag_wp) array *Input*
Note: the second dimension of the array VR must be at least $\max(1, MM)$ if JOB = 'E' or 'B', and at least 1 otherwise.
On entry: if JOB = 'E' or 'B', VR must contain right eigenvectors of (S, T) , corresponding to the eigenpairs specified by HOWMNY and SELECT. The eigenvectors must be stored in consecutive columns of VR, as returned by F08WNF (ZGGEV) or F08YXF (ZTGEVC).
 If JOB = 'V', VR is not referenced.

- 12: LDVR – INTEGER *Input*
On entry: the first dimension of the array VR as declared in the (sub)program from which F08YYF (ZTGSNA) is called.
Constraints:
 if JOB = 'E' or 'B', $LDVR \geq \max(1, N)$;
 otherwise $LDVR \geq 1$.
- 13: S(*) – REAL (KIND=nag_wp) array *Output*
Note: the dimension of the array S must be at least $\max(1, MM)$ if JOB = 'E' or 'B', and at least 1 otherwise.
On exit: if JOB = 'E' or 'B', the reciprocal condition numbers of the selected eigenvalues, stored in consecutive elements of the array.
 If JOB = 'V', S is not referenced.
- 14: DIF(*) – REAL (KIND=nag_wp) array *Output*
Note: the dimension of the array DIF must be at least $\max(1, MM)$ if JOB = 'V' or 'B', and at least 1 otherwise.
On exit: if JOB = 'V' or 'B', the estimated reciprocal condition numbers of the selected eigenvectors, stored in consecutive elements of the array. If the eigenvalues cannot be reordered to compute $DIF(j)$, $DIF(j)$ is set to 0; this can only occur when the true value would be very small anyway.
 If JOB = 'E', DIF is not referenced.
- 15: MM – INTEGER *Input*
On entry: the number of elements in the arrays S and DIF.
Constraints:
 if HOWMNY = 'A', $MM \geq N$;
 otherwise $MM \geq$ the number of selected eigenvalues.
- 16: M – INTEGER *Output*
On exit: the number of elements of the arrays S and DIF used to store the specified condition numbers; for each selected eigenvalue one element is used.
 If HOWMNY = 'A', M is set to N.
- 17: 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.
- 18: LWORK – INTEGER *Input*
On entry: the dimension of the array WORK as declared in the (sub)program from which F08YYF (ZTGSNA) 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 JOB = 'V' or 'B', $LWORK \geq \max(1, 2 \times N \times N)$;
 otherwise $LWORK \geq \max(1, N)$.

- 19: IWORK(*) – INTEGER array *Workspace*
Note: the dimension of the array IWORK must be at least $(N + 2)$.
 If JOB = 'E', IWORK is not referenced.
- 20: 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

None.

8 Parallelism and Performance

F08YYF (ZTGSNA) is not threaded by NAG in any implementation.

F08YYF (ZTGSNA) 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

An approximate asymptotic error bound on the chordal distance between the computed eigenvalue $\tilde{\lambda}$ and the corresponding exact eigenvalue λ is

$$\chi(\tilde{\lambda}, \lambda) \leq \epsilon \|(A, B)\|_F / S(\lambda)$$

where ϵ is the *machine precision*.

An approximate asymptotic error bound for the right or left computed eigenvectors \tilde{x} or \tilde{y} corresponding to the right and left eigenvectors x and y is given by

$$\theta(\tilde{z}, z) \leq \epsilon \|(A, B)\|_F / \text{Dif}.$$

The real analogue of this routine is F08YLF (DTGSNA).

10 Example

This example estimates condition numbers and approximate error estimates for all the eigenvalues and right eigenvectors of the pair (S, T) given by

$$S = \begin{pmatrix} 4.0 + 4.0i & 1.0 + 1.0i & 1.0 + 1.0i & 2.0 - 1.0i \\ 0 & 2.0 + 1.0i & 1.0 + 1.0i & 1.0 + 1.0i \\ 0 & 0 & 2.0 - 1.0i & 1.0 + 1.0i \\ 0 & 0 & 0 & 6.0 - 2.0i \end{pmatrix}$$

and

$$T = \begin{pmatrix} 2.0 & 1.0 + 1.0i & 1.0 + 1.0i & 3.0 - 1.0i \\ 0 & 1.0 & 2.0 + 1.0i & 1.0 + 1.0i \\ 0 & 0 & 1.0 & 1.0 + 1.0i \\ 0 & 0 & 0 & 2.0 \end{pmatrix}.$$

The eigenvalues and eigenvectors are computed by calling F08YXF (ZTGEVC).

10.1 Program Text

```

Program f08yyfe

!      F08YYF Example Program Text

!      Mark 25 Release. NAG Copyright 2014.

!      .. Use Statements ..
Use nag_library, Only: f06bnf, f06uaf, nag_wp, x02ajf, ztgevc, ztgsna
!      .. Implicit None Statement ..
Implicit None
!      .. Parameters ..
Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
Real (Kind=nag_wp)         :: eps, snorm, stnrm, tnorm
Integer                    :: i, info, lda, ldb, ldvl, ldvr,      &
                             lwork, m, n
!      .. Local Arrays ..
Complex (Kind=nag_wp), Allocatable :: a(:,,:), b(:,,:), vl(:,,:), vr(:,,:), &
                             work(:)
Real (Kind=nag_wp), Allocatable  :: dif(:), rwork(:), s(:)
Integer, Allocatable             :: iwork(:)
Logical                          :: select(1)
!      .. Executable Statements ..
Write (nout,*) 'F08YYF Example Program Results'
Write (nout,*)
!      Skip heading in data file
Read (nin,*)
Read (nin,*) n
lda = n
ldb = n
ldvl = n
ldvr = n
lwork = 2*n*n
Allocate (a(lda,n),b(ldb,n),vl(ldvl,n),vr(ldvr,n),work(lwork),dif(n), &
         rwork(2*n),s(n),iwork(n+2))

!      Read A and B from data file

Read (nin,*)(a(i,1:n),i=1,n)
Read (nin,*)(b(i,1:n),i=1,n)

!      Calculate the left and right generalized eigenvectors of the
!      pair (A,B).

!      The NAG name equivalent of ztgevc is f08yxf
Call ztgevc('Both','All',select,n,a,lda,b,ldb,vl,ldvl,vr,ldvr,n,m,work, &
         rwork,info)

!      Estimate condition numbers for all the generalized eigenvalues
!      and right eigenvectors of the pair (A,B)

!      The NAG name equivalent of ztgsna is f08yyf
Call ztgsna('Both','All',select,n,a,lda,b,ldb,vl,ldvl,vr,ldvr,s,dif,n,m, &
         work,lwork,iwork,info)

!      Print condition numbers of eigenvalues and right eigenvectors

Write (nout,*) 'S'
Write (nout,99999) s(1:m)

```

```

      Write (nout,*)
      Write (nout,*) 'DIF'
      Write (nout,99999) dif(1:m)

!      Calculate approximate error estimates

!      Compute the 1-norms of A and B and then compute
!      SQRRT(snorm**2 + tnorm**2)

      eps = x02ajf()
      snorm = f06uaf('1-norm',n,n,a,lda,rwork)
      tnorm = f06uaf('1-norm',n,n,b,ldb,rwork)
      stnrm = f06bnf(snorm,tnorm)
      Write (nout,*)
      Write (nout,*) 'Approximate error estimates for eigenvalues of (A,B)'
      Write (nout,99999)(eps*stnrm/s(i),i=1,m)
      Write (nout,*)
      Write (nout,*) &
      'Approximate error estimates for right eigenvectors of (A,B)'
      Write (nout,99999)(eps*stnrm/dif(i),i=1,m)

99999 Format ((3X,1P,7E11.1))
      End Program f08yyfe

```

10.2 Program Data

F08YYF Example Program Data

```

  4                                     :Value of N
( 4.0, 4.0) ( 1.0, 1.0) ( 1.0, 1.0) ( 2.0,-1.0)
( 0.0, 0.0) ( 2.0, 1.0) ( 1.0, 1.0) ( 1.0, 1.0)
( 0.0, 0.0) ( 0.0, 0.0) ( 2.0,-1.0) ( 1.0, 1.0)
( 0.0, 0.0) ( 0.0, 0.0) ( 0.0, 0.0) ( 6.0,-2.0)      :End of matrix A
( 2.0, 0.0) ( 1.0, 1.0) ( 1.0, 1.0) ( 3.0,-1.0)
( 0.0, 0.0) ( 1.0, 0.0) ( 2.0, 1.0) ( 1.0, 1.0)
( 0.0, 0.0) ( 0.0, 0.0) ( 1.0, 0.0) ( 1.0, 1.0)
( 0.0, 0.0) ( 0.0, 0.0) ( 0.0, 0.0) ( 2.0, 0.0)      :End of matrix B

```

10.3 Program Results

F08YYF Example Program Results

```

S
  1.0E+00   8.2E-01   7.2E-01   8.2E-01

DIF
  3.2E-01   3.6E-01   5.5E-01   2.8E-01

Approximate error estimates for eigenvalues of (A,B)
  1.5E-15   1.9E-15   2.1E-15   1.9E-15

Approximate error estimates for right eigenvectors of (A,B)
  4.8E-15   4.3E-15   2.8E-15   5.5E-15

```
