

NAG Toolbox

nag_lapack_ztgsna (f08yy)

1 Purpose

nag_lapack_ztgsna (f08yy) estimates condition numbers for specified eigenvalues and/or eigenvectors of a complex matrix pair in generalized Schur form.

2 Syntax

```
[s, dif, m, info] = nag_lapack_ztgsna(job, howmny, select, a, b, vl, vr, mm, 'n', n)
```

```
[s, dif, m, info] = f08yy(job, howmny, select, a, b, vl, vr, mm, 'n', n)
```

3 Description

nag_lapack_ztgsna (f08yy) estimates condition numbers for specified eigenvalues and/or right eigenvectors of an n by n matrix pair (S, T) in generalized Schur form. The function 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 nag_lapack_zgges (f08xn) or nag_lapack_zggesx (f08xp), or nag_lapack_zhgeqz (f08xs) 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

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

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 KÖgström 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>

KÖgström 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

5.1 Compulsory Input Parameters

1: **job** – CHARACTER(1)

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)

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

The dimension of the array **select** must be at least $\max(1, \mathbf{n})$ if **howmny** = 'S', and at least 1 otherwise

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: **a**(*lda*, :) – COMPLEX (KIND=nag_wp) array

The first dimension of the array **a** must be at least **n**.

The second dimension of the array **a** must be at least **n**.

The upper triangular matrix S .

- 5: **b**(*ldb*,:) – COMPLEX (KIND=nag_wp) array

The first dimension of the array **b** must be at least **n**.

The second dimension of the array **b** must be at least **n**.

The upper triangular matrix T .

- 6: **vl**(*ldvl*,:) – COMPLEX (KIND=nag_wp) array

The first dimension, *ldvl*, of the array **vl** must satisfy

if **job** = 'E' or 'B', $ldvl \geq \max(1, \mathbf{n})$;
otherwise $ldvl \geq 1$.

The second dimension of the array **vl** must be at least $\max(1, \mathbf{mm})$ if **job** = 'E' or 'B', and at least 1 otherwise.

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 nag_lapack_zggeev (f08wn) or nag_lapack_ztgevc (f08yx).

If **job** = 'V', **vl** is not referenced.

- 7: **vr**(*ldvr*,:) – COMPLEX (KIND=nag_wp) array

The first dimension, *ldvr*, of the array **vr** must satisfy

if **job** = 'E' or 'B', $ldvr \geq \max(1, \mathbf{n})$;
otherwise $ldvr \geq 1$.

The second dimension of the array **vr** must be at least $\max(1, \mathbf{mm})$ if **job** = 'E' or 'B', and at least 1 otherwise.

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 nag_lapack_zggeev (f08wn) or nag_lapack_ztgevc (f08yx).

If **job** = 'V', **vr** is not referenced.

- 8: **mm** – INTEGER

The number of elements in the arrays **s** and **dif**.

Constraints:

if **howmny** = 'A', $\mathbf{mm} \geq \mathbf{n}$;
otherwise $\mathbf{mm} \geq$ the number of selected eigenvalues.

5.2 Optional Input Parameters

- 1: **n** – INTEGER

Default: the first dimension of the arrays **a**, **b** and the second dimension of the arrays **a**, **b**. (An error is raised if these dimensions are not equal.)

n, the order of the matrix pair (S, T) .

Constraint: $\mathbf{n} \geq 0$.

5.3 Output Parameters

1: **s**(:) – REAL (KIND=nag_wp) array

The dimension of the array **s** will be $\max(1, \mathbf{mm})$ if **job** = 'E' or 'B' and 1 otherwise

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.

2: **dif**(:) – REAL (KIND=nag_wp) array

The dimension of the array **dif** will be $\max(1, \mathbf{mm})$ if **job** = 'V' or 'B' and 1 otherwise

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.

3: **m** – INTEGER

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**.

4: **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: **job**, 2: **howmny**, 3: **select**, 4: **n**, 5: **a**, 6: **lda**, 7: **b**, 8: **ldb**, 9: **vl**, 10: **ldvl**, 11: **vr**, 12: **ldvr**, 13: **s**, 14: **dif**, 15: **mm**, 16: **m**, 17: **work**, 18: **lwork**, 19: **iwork**, 20: **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.

7 Accuracy

None.

8 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 function is nag_lapack_dtgsna (f08yl).

9 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 `nag_lapack_ztgevc` (f08yx).

9.1 Program Text

```
function f08yy_example

fprintf('f08yy example results\n\n');

% Generalized Schur form pair (S,T)
n = nag_int(4);
S = [ 4 + 4i, 1 + 1i, 1 + 1i, 2 - 1i;
      0 + 0i, 2 + 1i, 1 + 1i, 1 + 1i;
      0 + 0i, 0 + 0i, 2 - 1i, 1 + 1i;
      0 + 0i, 0 + 0i, 0 + 0i, 6 - 2i];
T = [2, 1 + 1i, 1 + 1i, 3 - 1i;
      0 + 0i, 1 + 0i, 2 + 1i, 1 + 1i;
      0 + 0i, 0 + 0i, 1 + 0i, 1 + 1i;
      0 + 0i, 0 + 0i, 0 + 0i, 2 + 0i];

% Obtain scaled eigenvectors from Schur form
job = 'Both';
howmny = 'All';
select = [false];
Q = complex(eye(n));
Z = Q;

[VL, VR, m, info] = f08yx( ...
    job, howmny, select, S, T, Q, Z, n);

% Estimate condition numbers for eigenvalues and right eigenvectors
[rconde, rcondv, m, info] = ...
    f08yy( ...
    job, howmny, select, S, T, VL, VR, n);

disp('Reciprocal condition numbers for eigenvalues of (S,T)');
fprintf('%11.1e', rconde);
fprintf('\n\n');
disp('Reciprocal condition numbers for right eigenvectors of (S,T)');
fprintf('%11.1e', rcondv);
fprintf('\n\n');

% Calculate approximate error estimates

snorm = norm(S,1);
tnorm = norm(T,1);
stnorm = sqrt(snorm^2 + tnorm^2);

disp('Approximate error estimates for eigenvalues of (S,T)')
erre = x02aj*stnorm./rconde;
fprintf('%11.1e', erre);
```

```
fprintf('\n\n');  
disp('Approximate error estimates for right eigenvectors of (S,T)')  
errv = x02aj*stnorm./rcondv;  
fprintf('%11.1e',errv);  
fprintf('\n');
```

9.2 Program Results

f08yy example results

Reciprocal condition numbers for eigenvalues of (S,T)
1.0e+00 8.2e-01 7.2e-01 8.2e-01

Reciprocal condition numbers for right eigenvectors of (S,T)
3.2e-01 3.6e-01 5.5e-01 2.8e-01

Approximate error estimates for eigenvalues of (S,T)
1.5e-15 1.9e-15 2.1e-15 1.9e-15

Approximate error estimates for right eigenvectors of (S,T)
4.8e-15 4.3e-15 2.8e-15 5.5e-15
