NAG C Library Function Document

nag_ztrsna (f08qyc)

1 Purpose

nag_ztrsna (f08qyc) estimates condition numbers for specified eigenvalues and/or right eigenvectors of a complex upper triangular matrix.

2 Specification

```c
void nag_ztrsna (Nag_OrderType order, Nag_JobType job, Nag_HowManyType how_many,
    const Boolean select[], Integer n, const Complex t[], Integer pdt,
    const Complex vl[], Integer pdvl, const Complex vr[], Integer pdvr,
    double s[],
    double sep[], Integer mm, Integer *m, NagError *fail)
```

3 Description

nag_ztrsna (f08qyc) estimates condition numbers for specified eigenvalues and/or right eigenvectors of a complex upper triangular matrix \( T \). These are the same as the condition numbers of the eigenvalues and right eigenvectors of an original matrix \( A = ZTZ^H \) (with unitary \( Z \)), from which \( T \) may have been derived.

nag_ztrsna (f08qyc) computes the reciprocal of the condition number of an eigenvalue \( \lambda_i \) as

\[
s_i = \frac{|v^H u|}{\|u\|_E \|v\|_E},
\]

where \( u \) and \( v \) are the right and left eigenvectors of \( T \), respectively, corresponding to \( \lambda_i \). This reciprocal condition number always lies between zero (i.e., ill-conditioned) and one (i.e., well-conditioned).

An approximate error estimate for a computed eigenvalue \( \lambda_i \) is then given by

\[
\frac{\epsilon \|T\|}{s_i},
\]

where \( \epsilon \) is the machine precision.

To estimate the reciprocal of the condition number of the right eigenvector corresponding to \( \lambda_i \), the function first calls nag_ztrexc (f08qtc) to reorder the eigenvalues so that \( \lambda_i \) is in the leading position:

\[
T = Q \begin{pmatrix} \lambda_i & * \\ 0 & T_{22} \end{pmatrix} Q^H.
\]

The reciprocal condition number of the eigenvector is then estimated as \( \text{sep}_i \), the smallest singular value of the matrix \((T_{22} - \lambda_i I)\). This number ranges from zero (i.e., ill-conditioned) to very large (i.e., well-conditioned).

An approximate error estimate for a computed right eigenvector \( u \) corresponding to \( \lambda_i \) is then given by

\[
\frac{\epsilon \|T\|}{\text{sep}_i}.
\]

4 References

5 Parameters

1: \textbf{order} – Nag_OrderType \hspace{1cm} \textit{Input}

\textit{On entry:} the \textbf{order} parameter specifies the two-dimensional storage scheme being used, i.e., row-major ordering or column-major ordering. C language defined storage is specified by \textbf{order} = \texttt{Nag_RowMajor}. See Section 2.2.1.4 of the Essential Introduction for a more detailed explanation of the use of this parameter.

\textit{Constraint:} \textbf{order} = \texttt{Nag_RowMajor} or \texttt{Nag_ColMajor}.

2: \textbf{job} – Nag_JobType \hspace{1cm} \textit{Input}

\textit{On entry:} indicates whether condition numbers are required for eigenvalues and/or eigenvectors, as follows:

- if \textbf{job} = \texttt{Nag_EigVals}, then condition numbers for eigenvalues only are computed;
- if \textbf{job} = \texttt{Nag_EigVecs}, then condition numbers for eigenvectors only are computed;
- if \textbf{job} = \texttt{Nag_DoBoth}, then condition numbers for both eigenvalues and eigenvectors are computed.

\textit{Constraint:} \textbf{job} = \texttt{Nag_EigVals}, \texttt{Nag_EigVecs} or \texttt{Nag_DoBoth}.

3: \textbf{how_many} – Nag_HowManyType \hspace{1cm} \textit{Input}

\textit{On entry:} indicates how many condition numbers are to be computed, as follows:

- if \textbf{how_many} = \texttt{Nag_ComputeAll}, then condition numbers for all eigenpairs are computed;
- if \textbf{how_many} = \texttt{Nag_ComputeSelected}, then condition numbers for selected eigenpairs (as specified by \textbf{select}) are computed.

\textit{Constraint:} \textbf{how_many} = \texttt{Nag_ComputeAll} or \texttt{Nag_ComputeSelected}.

4: \textbf{select}[\textit{dim}] – const Boolean \hspace{1cm} \textit{Input}

\textit{Note:} the dimension, \textit{dim}, of the array \textbf{select} must be at least \text{max}(1, n) when \textbf{how_many} = \texttt{Nag_ComputeSelected} and at least 1 otherwise.

\textit{On entry:} \textbf{select} specifies the eigenpairs for which condition numbers are to be computed if \textbf{how_many} = \texttt{Nag_ComputeSelected}. To select condition numbers for the eigenpair corresponding to the eigenvalue \( \lambda_j \), \textbf{select}[j] must be set to \texttt{TRUE}.

\textbf{select} is not referenced if \textbf{how_many} = \texttt{Nag_ComputeAll}.

5: \textbf{n} – Integer \hspace{1cm} \textit{Input}

\textit{On entry:} \textit{n}, the order of the matrix \( T \).

\textit{Constraint:} \textbf{n} \geq 0.

6: \textbf{t}[\textit{dim}] – const Complex \hspace{1cm} \textit{Input}

\textit{Note:} the dimension, \textit{dim}, of the array \textbf{t} must be at least \text{max}(1, \text{pdt} \times \textbf{n})

If \textbf{order} = \texttt{Nag_ColMajor}, the \((i,j)\)th element of the matrix \( T \) is stored in \( \textbf{t}[(j-1) \times \text{pdt} + i - 1] \) and if \textbf{order} = \texttt{Nag_RowMajor}, the \((i,j)\)th element of the matrix \( T \) is stored in \( \textbf{t}[(i-1) \times \text{pdt} + j - 1] \).

\textit{On entry:} the \textit{n} by \textit{n} upper triangular matrix \( T \), as returned by \texttt{nag_zhseqr} (f08psc).

7: \textbf{pdt} – Integer \hspace{1cm} \textit{Input}

\textit{On entry:} the stride separating matrix row or column elements (depending on the value of \textbf{order}) in the array \textbf{t}.

\textit{Constraint:} \textbf{pdt} \geq \text{max}(1, \textbf{n}).
8:  \( \text{vl}[\text{dim}] \) – const Complex  

Note: the dimension, \( \text{dim} \), of the array \( \text{vl} \) must be at least  
\[
\max(1, \text{pdvl} \times \text{mm}) \quad \text{when} \quad \text{job} = \text{Nag_EigVals} \quad \text{or} \quad \text{Nag_DoBoth} \quad \text{and} \quad \text{order} = \text{Nag_ColMajor};
\]
\[
\max(1, \text{pdvl} \times n) \quad \text{when} \quad \text{job} = \text{Nag_EigVals} \quad \text{or} \quad \text{Nag_DoBoth} \quad \text{and} \quad \text{order} = \text{Nag_RowMajor};
\]
1 when \( \text{job} = \text{Nag_EigVecs} \).

If \( \text{order} = \text{Nag_ColMajor} \), the \((i, j)\)th element of the matrix is stored in \( \text{vl}[(j - 1) \times \text{pdvl} + i - 1] \) and  
if \( \text{order} = \text{Nag_RowMajor} \), the \((i, j)\)th element of the matrix is stored in \( \text{vl}[(i - 1) \times \text{pdvl} + j - 1] \).

On entry: if \( \text{job} = \text{Nag_EigVals} \) or \( \text{Nag_DoBoth} \), \( \text{vl} \) must contain the left eigenvectors of \( T \) (or of any matrix \( QTQ^H \) with \( Q \) unitary) corresponding to the eigenpairs specified by \( \text{how_many} \) and \( \text{select} \). The eigenvectors must be stored in consecutive rows or columns (depending on the value of \( \text{order} \) of \( \text{vl} \), as returned by \( \text{nag_ztrevc} \) (f08qxc) or \( \text{nag_zhsein} \) (f08pxc).

\( \text{vl} \) is not referenced if \( \text{job} = \text{Nag_EigVecs} \).

9:  \( \text{pdvl} \) – Integer  

On entry: the stride separating matrix row or column elements (depending on the value of \( \text{order} \) in the array \( \text{vl} \).

Constraints:

if \( \text{order} = \text{Nag_ColMajor} \),  
if \( \text{job} = \text{Nag_EigVals} \) or \( \text{Nag_DoBoth} \), \( \text{pdvl} \geq \max(1, \text{n}) \);  
if \( \text{job} = \text{Nag_EigVecs} \), \( \text{pdvl} \geq 1 \);

if \( \text{order} = \text{Nag_RowMajor} \),  
if \( \text{job} = \text{Nag_EigVals} \) or \( \text{Nag_DoBoth} \), \( \text{pdvl} \geq \max(1, \text{mm}) \);  
if \( \text{job} = \text{Nag_EigVecs} \), \( \text{pdvl} \geq 1 \).

10:  \( \text{vr}[\text{dim}] \) – const Complex  

Note: the dimension, \( \text{dim} \), of the array \( \text{vr} \) must be at least  
\[
\max(1, \text{pdvr} \times \text{mm}) \quad \text{when} \quad \text{job} = \text{Nag_EigVals} \quad \text{or} \quad \text{Nag_DoBoth} \quad \text{and} \quad \text{order} = \text{Nag_ColMajor};
\]
\[
\max(1, \text{pdvr} \times n) \quad \text{when} \quad \text{job} = \text{Nag_EigVals} \quad \text{or} \quad \text{Nag_DoBoth} \quad \text{and} \quad \text{order} = \text{Nag_RowMajor};
\]
1 when \( \text{job} = \text{Nag_EigVecs} \).

If \( \text{order} = \text{Nag_ColMajor} \), the \((i, j)\)th element of the matrix is stored in \( \text{vr}[(j - 1) \times \text{pdvr} + i - 1] \) and  
if \( \text{order} = \text{Nag_RowMajor} \), the \((i, j)\)th element of the matrix is stored in \( \text{vr}[(i - 1) \times \text{pdvr} + j - 1] \).

On entry: if \( \text{job} = \text{Nag_EigVals} \) or \( \text{Nag_DoBoth} \), \( \text{vr} \) must contain the right eigenvectors of \( T \) (or of any matrix \( QTQ^H \) with \( Q \) unitary) corresponding to the eigenpairs specified by \( \text{how_many} \) and \( \text{select} \). The eigenvectors must be stored in consecutive rows or columns (depending on the value of \( \text{order} \) of \( \text{vr} \), as returned by \( \text{nag_ztrevc} \) (f08qxc) or \( \text{nag_zhsein} \) (f08pxc).

\( \text{vr} \) is not referenced if \( \text{job} = \text{Nag_EigVecs} \).

11:  \( \text{pdvr} \) – Integer  

On entry: the stride separating matrix row or column elements (depending on the value of \( \text{order} \) in the array \( \text{vr} \).

Constraints:

if \( \text{order} = \text{Nag_ColMajor} \),  
if \( \text{job} = \text{Nag_EigVals} \) or \( \text{Nag_DoBoth} \), \( \text{pdvr} \geq \max(1, \text{n}) \);  
if \( \text{job} = \text{Nag_EigVecs} \), \( \text{pdvr} \geq 1 \);

if \( \text{order} = \text{Nag_RowMajor} \),  
if \( \text{job} = \text{Nag_EigVals} \) or \( \text{Nag_DoBoth} \), \( \text{pdvr} \geq \max(1, \text{mm}) \);  
if \( \text{job} = \text{Nag_EigVecs} \), \( \text{pdvr} \geq 1 \).
12:  \( s[\text{dim}] \) – double  
   \( \text{Output} \)
   
   \text{Note:} the dimension, \( \text{dim} \), of the array \( s \) must be at least \( \max(1, \text{mm}) \) when \( \text{job} = \text{Nag\_EigVals} \) or \( \text{Nag\_DoBoth} \) and at least 1 when \( \text{job} = \text{Nag\_EigVecs} \).

   \text{On exit:} the reciprocal condition numbers of the selected eigenvalues if \( \text{job} = \text{Nag\_EigVals} \) or \( \text{Nag\_DoBoth} \), stored in consecutive elements of the array. Thus \( s[j] \), \( \text{sep}[j] \) and the \( j \)th rows or columns of \( vl \) and \( vr \) all correspond to the same eigenpair (but not in general the \( j \)th eigenpair unless all eigenpairs have been selected).

   \( s \) is not referenced if \( \text{job} = \text{Nag\_EigVecs} \).

13:  \( \text{sep}[\text{dim}] \) – double  
   \( \text{Output} \)
   
   \text{Note:} the dimension, \( \text{dim} \), of the array \( \text{sep} \) must be at least \( \max(1, \text{mm}) \) when \( \text{job} = \text{Nag\_EigVecs} \) or \( \text{Nag\_DoBoth} \) and at least 1 when \( \text{job} = \text{Nag\_EigVals} \).

   \text{On exit:} the estimated reciprocal condition numbers of the selected right eigenvectors if \( \text{job} = \text{Nag\_EigVecs} \) or \( \text{Nag\_DoBoth} \), stored in consecutive elements of the array.

   \( \text{sep} \) is not referenced if \( \text{job} = \text{Nag\_EigVals} \).

14:  \( \text{mm} \) – Integer  
   \( \text{Input} \)
   
   \text{On entry:} the number of elements in the arrays \( s \) and \( \text{sep} \), and the number of rows or columns (depending on the value of \( \text{order} \)) in the arrays \( vl \) and \( vr \) (if used). The precise number required, \( \text{required\_rowcol} \), is \( n \) if \( \text{how\_many} = \text{Nag\_ComputeAll} \); if \( \text{how\_many} = \text{Nag\_ComputeSelected} \), \( \text{required\_rowcol} \) is the number of selected eigenpairs (see \( \text{select} \)), in which case \( 0 \leq \text{required\_rowcol} \leq n \).

   \text{Constraint:} \( \text{mm} \geq \text{required\_rowcol} \).

15:  \( m \) – Integer  
   \( \text{Output} \)
   
   \text{On exit:} \( \text{required\_rowcol} \), the number of selected eigenpairs. If \( \text{how\_many} = \text{Nag\_ComputeAll} \), \( m \) is set to \( n \).

16:  \( \text{fail} \) – NagError  
   \( \text{Output} \)
   
   The NAG error parameter (see the Essential Introduction).

6   \textbf{Error Indicators and Warnings}

\textbf{NE\_INT}

On entry, \( n = \langle \text{value} \rangle \).

Constraint: \( n \geq 0 \).

On entry, \( \text{mm} = \langle \text{value} \rangle \).

Constraint: \( \text{mm} \geq \text{required\_rowcol} \), where \( \text{required\_rowcol} \) is the number of selected eigenpairs.

On entry, \( \text{pdt} = \langle \text{value} \rangle \).

Constraint: \( \text{pdt} > 0 \).

On entry, \( \text{pdvl} = \langle \text{value} \rangle \).

Constraint: \( \text{pdvl} > 0 \).

On entry, \( \text{pdvr} = \langle \text{value} \rangle \).

Constraint: \( \text{pdvr} > 0 \).

\textbf{NE\_INT\_2}

On entry, \( \text{pdt} = \langle \text{value} \rangle \), \( n = \langle \text{value} \rangle \).

Constraint: \( \text{pdt} \geq \max(1, n) \).
On entry, \( \text{job} = \langle \text{value} \rangle \), \( \text{n} = \langle \text{value} \rangle \), \( \text{pdvl} = \langle \text{value} \rangle \).
Constraint: if \( \text{job} = \text{Nag_EigVals} \) or \( \text{Nag_DoBoth} \), \( \text{pdvl} \geq \max(1, \text{n}) \);
if \( \text{job} = \text{Nag_EigVvecs} \), \( \text{pdvl} \geq 1 \).

On entry, \( \text{job} = \langle \text{value} \rangle \), \( \text{n} = \langle \text{value} \rangle \), \( \text{pdvr} = \langle \text{value} \rangle \).
Constraint: if \( \text{job} = \text{Nag_EigVals} \) or \( \text{Nag_DoBoth} \), \( \text{pdvr} \geq \max(1, \text{n}) \);
if \( \text{job} = \text{Nag_EigVvecs} \), \( \text{pdvr} \geq 1 \).

On entry, \( \text{job} = \langle \text{value} \rangle \), \( \text{mm} = \langle \text{value} \rangle \), \( \text{pdvl} = \langle \text{value} \rangle \).
Constraint: if \( \text{job} = \text{Nag_EigVals} \) or \( \text{Nag_DoBoth} \), \( \text{pdvl} \geq \max(1, \text{mm}) \);
if \( \text{job} = \text{Nag_EigVvecs} \), \( \text{pdvl} \geq 1 \).

On entry, \( \text{job} = \langle \text{value} \rangle \), \( \text{mm} = \langle \text{value} \rangle \), \( \text{pdvr} = \langle \text{value} \rangle \).
Constraint: if \( \text{job} = \text{Nag_EigVals} \) or \( \text{Nag_DoBoth} \), \( \text{pdvr} \geq \max(1, \text{mm}) \);
if \( \text{job} = \text{Nag_EigVvecs} \), \( \text{pdvr} \geq 1 \).

Memory allocation failed.

On entry, parameter \( \langle \text{value} \rangle \) had an illegal value.

An internal error has occurred in this function. Check the function call and any array sizes. If the call is correct then please consult NAG for assistance.

The computed values \( \text{sep}_i \) may over estimate the true value, but seldom by a factor of more than 3.

The real analogue of this function is \text{nag_dtrsna (f08qlc)}.

To compute approximate error estimates for all the eigenvalues and right eigenvectors of the matrix \( T \), where
\[
T = \begin{pmatrix}
-6.0004 - 6.9999i & 0.3637 - 0.3656i & -0.1880 + 0.4787i & 0.8785 - 0.2539i \\
0.0000 + 0.0000i & -5.0000 + 2.0060i & -0.0307 - 0.7217i & -0.2290 + 0.1313i \\
0.0000 + 0.0000i & 0.0000 + 0.0000i & 7.9982 - 0.9964i & 0.9357 + 0.5359i \\
0.0000 + 0.0000i & 0.0000 + 0.0000i & 0.0000 + 0.0000i & 3.0023 - 3.9982i
\end{pmatrix},
\]

/* \text{nag_ztrsna (f08qyc)} Example Program.  *
 * Copyright 2001 Numerical Algorithms Group.  *
 * Mark 7, 2001.  */
#include <stdio.h>
#include <nag.h>
#include <nag_stdlib.h>
#include <naga02.h>
#include <nagf08.h>
#include <nagf16.h>
#include <nagx02.h>

int main(void)
{
    /* Scalars */
    Integer i, j, m, n, pdt, pdvl, pdvr;
    Integer select_len, s_len;
    Integer exit_status=0;
    double  eps, tnorm;
    NagError fail;
    Nag_OrderType order;
    /* Arrays */
    double *s = 0, *sep = 0;
    Complex *t = 0, *vl = 0, *vr = 0;
    Boolean *select = 0;

    #ifdef NAG_COLUMN_MAJOR
    #define T(I,J) t[(J-1)*pdt+I-1]
    order = Nag_ColMajor;
    #else
    #define T(I,J) t[(I-1)*pdt+J-1]
    order = Nag_RowMajor;
    #endif

    INIT_FAIL(fail);
    Vprintf("f08qyc Example Program Results\n");
    /* Skip heading in data file */
    Vscanf("%*[\n] ");
    Vscanf("%ld%*[\n] ", &n);
    #ifdef NAG_COLUMN_MAJOR
    pdt = n;
    pdvl = n;
    pdvr = n;
    #else
    pdt = n;
    pdvl = n;
    pdvr = n;
    #endif
    select_len = 1;
    s_len = n;
    /* Allocate memory */
    if ( !(t = NAG_ALLOC(n * n, Complex)) ||
        !(vl = NAG_ALLOC(n * n, Complex)) ||
        !(vr = NAG_ALLOC(n * n, Complex)) ||
        !(s = NAG_ALLOC(s_len, double)) ||
        !(sep = NAG_ALLOC(s_len, double)) ||
        (!select = NAG_ALLOC(select_len, Boolean)) )
    {
        Vprintf("Allocation failure\n");
        exit_status = -1;
        goto END;
    }
    /* Read T from data file */
    for (i = 1; i <= n; ++i)
    {
        for (j = 1; j <= n; ++j)
            Vscanf("( %lf , %lf ) ", &T(i,j).re, &T(i,j).im);
    }
    Vscanf("%*[\n] ");
    /* Calculate right and left eigenvectors of T */
    f08qxc(order, Nag_BothSides, Nag_ComputeAll, select, n, t, pdt,
           vl, pdvl, vr, pdvr, n, &m, &fail);
    if (fail.code != NE_NOERROR)
    {
        Vprintf("Error from f08qxc.\n%s\n", fail.message);
        exit_status = 1;
        goto END;
    }
}

f08qyc

NAG C Library Manual

f08qyc.6

[NP3645/7]
Estimate condition numbers for all the eigenvalues and right eigenvectors of T:

```c
f08qyc(order, Nag_DoBoth, Nag_ComputeAll, select, n, t, pdt,
   vl, pdvl, vr, pdvr, s, sep, n, &m, &fail);
if (fail.code != NE_NOERROR)
{
  Vprintf("Error from f08qyc.\n", fail.message);
  exit_status = 1;
  goto END;
}
```

Print condition numbers of eigenvalues and right eigenvectors:

```c
Vprintf("S\n");
for (i = 0; i < n; ++i)
  Vprintf("%11.1e", s[i]);
Vprintf("\n\nSep\n");
for (i = 0; i < n; ++i)
  Vprintf("%11.1e", sep[i]);
Vprintf("\n");
```

Calculate approximate error estimates (using the 1-norm):

```c
f16uac(order, Nag_OneNorm, n, n, t, pdt, &tnorm, &fail);
if (fail.code != NE_NOERROR)
{
  Vprintf("Error from f16uac.\n", fail.message);
  exit_status = 1;
  goto END;
}
eps = X02AJC;
Vprintf("Approximate error estimates for eigenvalues of T (machine dependent)\n");
for (i = 0; i < m; ++i)
  Vprintf("%11.1e", eps*tnorm/s[i]);
Vprintf("Approximate error estimates for right eigenvectors of T (machine dependent)\n");
for (i = 0; i < m; ++i)
  Vprintf("%11.1e", eps*tnorm/sep[i]);
Vprintf("\n");
```

Program Data:

```c
f08qyc Example Program Data
4
Value of N
(-6.0004, -6.9999) (0.3637, -0.3656) (-0.1880, 0.4787) (0.8785, -0.2539)
(0.0000, 0.0000) (-5.0000, 2.0060) (-0.0307, -0.7217) (-0.2290, 0.1313)
(0.0000, 0.0000) (0.0000, 0.0000) (7.9982, -0.9964) (0.9357, 0.5359)
(0.0000, 0.0000) (0.0000, 0.0000) (0.0000, 0.0000) (3.0023, -3.9998)
```

Program Results:

```c
f08qyc Example Program Results
S
9.9e-01 1.0e-00 9.8e-01 9.8e-01
Sep
8.4e+00 8.0e+00 5.8e+00 5.8e+00
```
Approximate error estimates for eigenvalues of $T$ (machine dependent)
1.0e-15  1.0e-15  1.1e-15  1.1e-15

Approximate error estimates for right eigenvectors of $T$ (machine dependent)
1.2e-16  1.3e-16  1.8e-16  1.8e-16