NAG Toolbox

nag_lapack_dtrrfs (f07th)

1 Purpose

nag_lapack_dtrrfs (f07th) returns error bounds for the solution of a real triangular system of linear equations with multiple right-hand sides, $AX = B$ or $A^{T}X = B$.

2 Syntax

```
[ferr, berr, info] = nag_lapack_dtrrfs(uplo, trans, diag, a, b, x, 'n', n,
'nrhs_p', nrhs_p)
```

```
[ferr, berr, info] = f07th(uplo, trans, diag, a, b, x, 'n', n, 'nrhs_p', nrhs_p)
```

3 Description

nag_lapack_dtrrfs (f07th) returns the backward errors and estimated bounds on the forward errors for the solution of a real triangular system of linear equations with multiple right-hand sides $AX = B$ or $A^{T}X = B$. The function handles each right-hand side vector (stored as a column of the matrix B) independently, so we describe the function of nag_lapack_dtrrfs (f07th) in terms of a single right-hand side b and solution x.

Given a computed solution x, the function computes the *component-wise backward error* β. This is the size of the smallest relative perturbation in each element of A and b such that x is the exact solution of a perturbed system

$$(A + \delta A)x = b + \delta b$$
$$|\delta a_{ij}| \leq \beta |a_{ij}| \quad \text{and} \quad |\delta b_i| \leq \beta |b_i|.$$

Then the function estimates a bound for the *component-wise forward error* in the computed solution, defined by:

$$\max_i |x_i - \hat{x}_i| / \max_i |x_i|$$

where $\hat{x}$ is the true solution.

For details of the method, see the F07 Chapter Introduction.

4 References

Golub G H and Van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

5.1 Compulsory Input Parameters

1: **uplo** – CHARACTER(1)

Specifies whether A is upper or lower triangular.

uplo = 'U'
A is upper triangular.

uplo = 'L'
A is lower triangular.

Constraint: **uplo** = 'U' or 'L'.

2: **trans** – CHARACTER(1)

Indicates the form of the equations.

trans = 'N'
The equations are of the form $AX = B$.

trans = 'T' or 'C'
The equations are of the form $A^{\mathrm{T}}X = B$.

Constraint: **trans** = 'N', 'T' or 'C'.

3: **diag** – CHARACTER(1)

Indicates whether A is a nonunit or unit triangular matrix.

diag = 'N'
A is a nonunit triangular matrix.

diag = 'U'
A is a unit triangular matrix; the diagonal elements are not referenced and are assumed to be 1.

Constraint: **diag** = 'N' or 'U'.

4: $\mathbf{a}(lda,:)$ – REAL (KIND=nag_wp) array

The first dimension of the array **a** must be at least $\max(1,\mathbf{n})$.

The second dimension of the array **a** must be at least $\max(1,\mathbf{n})$.

The n by n triangular matrix A.

If **uplo** = 'U', a is upper triangular and the elements of the array below the diagonal are not referenced.

If **uplo** = 'L', a is lower triangular and the elements of the array above the diagonal are not referenced.

If **diag** = 'U', the diagonal elements of a are assumed to be 1, and are not referenced.

5: $\mathbf{b}(ldb,:)$ – REAL (KIND=nag_wp) array

The first dimension of the array **b** must be at least $\max(1,\mathbf{n})$.

The second dimension of the array **b** must be at least $\max(1,\mathbf{nrhs\_p})$.

The n by r right-hand side matrix B.

6: $\mathbf{x}(ldx,:)$ – REAL (KIND=nag_wp) array

The first dimension of the array **x** must be at least $\max(1,\mathbf{n})$.

The second dimension of the array **x** must be at least $\max(1,\mathbf{nrhs\_p})$.

The n by r solution matrix X, as returned by nag_lapack_dtrtrs (f07te).

5.2 Optional Input Parameters

1: **n** – INTEGER

Default: the first dimension of the arrays **a**, **b**, **x** and the second dimension of the array **a**.

n, the order of the matrix A.

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

2: **nrhs_p** – INTEGER

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

r, the number of right-hand sides.

Constraint: **nrhs_p** ≥ 0.

5.3 Output Parameters

1: **ferr**(**nrhs_p**) – REAL (KIND=nag_wp) array

ferr(j) contains an estimated error bound for the jth solution vector, that is, the jth column of X, for $j = 1, 2, \ldots, r$.

2: **berr**(**nrhs_p**) – REAL (KIND=nag_wp) array

berr(j) contains the component-wise backward error bound β for the jth solution vector, that is, the jth column of X, for $j = 1, 2, \ldots, r$.

3: **info** – INTEGER

info $= 0$ unless the function 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 bounds returned in **ferr** are not rigorous, because they are estimated, not computed exactly; but in practice they almost always overestimate the actual error.

8 Further Comments

A call to nag_lapack_dtrrfs (f07th), for each right-hand side, involves solving a number of systems of linear equations of the form $Ax = b$ or $A^{\mathrm{T}}x = b$; the number is usually 4 or 5 and never more than 11. Each solution involves approximately n^2 floating-point operations.

The complex analogue of this function is nag_lapack_ztrrfs (f07tv).

9 Example

This example solves the system of equations $AX = B$ and to compute forward and backward error bounds, where

$$A = \begin{pmatrix} 4.30 & 0.00 & 0.00 & 0.00 \\ -3.96 & -4.87 & 0.00 & 0.00 \\ 0.40 & 0.31 & -8.02 & 0.00 \\ -0.27 & 0.07 & -5.95 & 0.12 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} -12.90 & -21.50 \\ 16.75 & 14.93 \\ -17.55 & 6.33 \\ -11.04 & 8.09 \end{pmatrix}.$$

9.1 Program Text

```
function f07th_example

fprintf('f07th example results\n\n');

% Solve AX=B and compute error bounds, where A is Lower triangular
a = [ 4.30,  0,     0,    0;
     -3.96, -4.87,  0,    0;
      0.40,  0.31, -8.02, 0;
     -0.27,  0.07, -5.95, 0.12];
b = [-12.90, -21.50;
      16.75,  14.93;
     -17.55,   6.33;
     -11.04,   8.09];

uplo = 'L';
trans = 'N';
diag = 'N';

% Solve
[x, info] = f07te( ...
                   uplo, trans, diag, a, b);

% Error bounds
[ferr, berr, info] = f07th( ...
                            uplo, trans, diag, a, b, x);

% Display solution
[ifail] = x04ca( ...
                 'Gen', diag, x, 'Solution(s)');

fprintf('\nBackward errors (machine-dependent)\n   ')
fprintf('%11.1e', berr);
fprintf('\nEstimated forward error bounds (machine-dependent)\n   ')
fprintf('%11.1e', ferr);
fprintf('\n');
```

9.2 Program Results

```
f07th example results

 Solution(s)
             1          2
 1     -3.0000    -5.0000
 2     -1.0000     1.0000
 3      2.0000    -1.0000
 4      1.0000     6.0000

Backward errors (machine-dependent)
       6.9e-17    0.0e+00
Estimated forward error bounds (machine-dependent)
       8.3e-14    2.6e-14
```