NAG Toolbox

nag_lapack_dgttrs (f07ce)

1 Purpose

nag_lapack_dgttrs (f07ce) computes the solution to a real system of linear equations $AX = B$ or $A^{\mathrm{T}}X = B$, where A is an n by n tridiagonal matrix and X and B are n by r matrices, using the LU factorization returned by nag_lapack_dgttrf (f07cd).

2 Syntax

```
[b, info] = nag_lapack_dgttrs(trans, dl, d, du, du2, ipiv, b, 'n', n, 'nrhs_p',
nrhs_p)
```

```
[b, info] = f07ce(trans, dl, d, du, du2, ipiv, b, 'n', n, 'nrhs_p', nrhs_p)
```

3 Description

nag_lapack_dgttrs (f07ce) should be preceded by a call to nag_lapack_dgttrf (f07cd), which uses Gaussian elimination with partial pivoting and row interchanges to factorize the matrix A as

$$A = PLU,$$

where P is a permutation matrix, L is unit lower triangular with at most one nonzero subdiagonal element in each column, and U is an upper triangular band matrix, with two superdiagonals. nag_lapack_dgttrs (f07ce) then utilizes the factorization to solve the required equations.

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

5 Parameters

5.1 Compulsory Input Parameters

1: **trans** – CHARACTER(1)

Specifies the equations to be solved as follows:

trans = 'N'
Solve $AX = B$ for X.

trans = 'T' or 'C'
Solve $A^{\mathrm{T}}X = B$ for X.

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

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

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

Must contain the $(n - 1)$ multipliers that define the matrix L of the LU factorization of A.

3: **d**(:) – REAL (KIND=nag_wp) array

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

Must contain the n diagonal elements of the upper triangular matrix U from the LU factorization of A.

4: **du**(:) – REAL (KIND=nag_wp) array

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

Must contain the $(n - 1)$ elements of the first superdiagonal of U.

5: **du2**(:) – REAL (KIND=nag_wp) array

The dimension of the array **du2** must be at least $\max(1, \mathbf{n} - 2)$

Must contain the $(n - 2)$ elements of the second superdiagonal of U.

6: **ipiv**(:) – INTEGER array

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

Must contain the n pivot indices that define the permutation matrix P. At the ith step, row i of the matrix was interchanged with row **ipiv**(i), and **ipiv**(i) must always be either i or $(i + 1)$, **ipiv**$(i) = i$ indicating that a row interchange was not performed.

7: **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 matrix of right-hand sides B.

5.2 Optional Input Parameters

1: **n** – INTEGER

Default: the first dimension of the array **b** and the dimension of the arrays **d**, **ipiv**.

n, the order of the matrix A.

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

2: **nrhs_p** – INTEGER

Default: the second dimension of the array **b**.

r, the number of right-hand sides, i.e., the number of columns of the matrix B.

Constraint: $\mathbf{nrhs\_p} \geq 0$.

5.3 Output Parameters

1: **b**$(ldb, :)$ – REAL (KIND=nag_wp) array

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

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

The n by r solution matrix X.

2: **info** – INTEGER

$\mathbf{info} = 0$ 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 computed solution for a single right-hand side, $\hat{x}$, satisfies an equation of the form

$$(A + E)\hat{x} = b,$$

where

$$\|E\|_1 = O(\epsilon)\|A\|_1$$

and ϵ is the ***machine precision***. An approximate error bound for the computed solution is given by

$$\frac{\|\hat{x} - x\|_1}{\|x\|_1} \leq \kappa(A)\frac{\|E\|_1}{\|A\|_1},$$

where $\kappa(A) = \left\|A^{-1}\right\|_1\|A\|_1$, the condition number of A with respect to the solution of the linear equations. See Section 4.4 of Anderson *et al.* (1999) for further details.

Following the use of this function nag_lapack_dgtcon (f07cg) can be used to estimate the condition number of A and nag_lapack_dgtrfs (f07ch) can be used to obtain approximate error bounds.

8 Further Comments

The total number of floating-point operations required to solve the equations $AX = B$ or $A^{\mathrm{T}}X = B$ is proportional to nr.

The complex analogue of this function is nag_lapack_zgttrs (f07cs).

9 Example

This example solves the equations

$$AX = B,$$

where A is the tridiagonal matrix

$$A = \begin{pmatrix} 3.0 & 2.1 & 0 & 0 & 0 \\ 3.4 & 2.3 & -1.0 & 0 & 0 \\ 0 & 3.6 & -5.0 & 1.9 & 0 \\ 0 & 0 & 7.0 & -0.9 & 8.0 \\ 0 & 0 & 0 & -6.0 & 7.1 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 2.7 & 6.6 \\ -0.5 & 10.8 \\ 2.6 & -3.2 \\ 0.6 & -11.2 \\ 2.7 & 19.1 \end{pmatrix}.$$

9.1 Program Text

```
    function f07ce_example

fprintf('f07ce example results\n\n');

% Tridiagonal matrix A stored as diagonals:
du = [        2.1    -1.0      1.9     8.0];
d  = [3.0     2.3    -5.0     -0.9     7.1];
dl = [3.4     3.6     7.0     -6.0        ];
n  = numel(d);

% Factorize A.
[dlf, df, duf, du2f, ipiv, info] = ...
  f07cd(dl, d, du);
```

```
% Rhs B
b = [ 2.7,   6.6;
     -0.5,  10.8;
      2.6,  -3.2;
      0.6, -11.2;
      2.7,  19.1];

% Solve AX = B
trans = 'No transpose';
[x, info] = f07ce( ...
                   trans, dlf, df, duf, du2f, ipiv, b);

disp('Solution(s)');
disp(x);
```

9.2 Program Results

```
     f07ce example results

Solution(s)
   -4.0000    5.0000
    7.0000   -4.0000
    3.0000   -3.0000
   -4.0000   -2.0000
   -3.0000    1.0000
```