NAG Toolbox

nag_lapack_dsytrs (f07me)

1 Purpose

nag_lapack_dsytrs (f07me) solves a real symmetric indefinite system of linear equations with multiple right-hand sides,

$$AX = B,$$

where A has been factorized by nag_lapack_dsytrf (f07md).

2 Syntax

```
[b, info] = nag_lapack_dsytrs(uplo, a, ipiv, b, 'n', n, 'nrhs_p', nrhs_p)
[b, info] = f07me(uplo, a, ipiv, b, 'n', n, 'nrhs_p', nrhs_p)
```

3 Description

nag_lapack_dsytrs (f07me) is used to solve a real symmetric indefinite system of linear equations $AX = B$, this function must be preceded by a call to nag_lapack_dsytrf (f07md) which computes the Bunch–Kaufman factorization of A.

If **uplo** = 'U', $A = PUDU^{T}P^{T}$, where P is a permutation matrix, U is an upper triangular matrix and D is a symmetric block diagonal matrix with 1 by 1 and 2 by 2 blocks; the solution X is computed by solving $PUDY = B$ and then $U^{T}P^{T}X = Y$.

If **uplo** = 'L', $A = PLDL^{T}P^{T}$, where L is a lower triangular matrix; the solution X is computed by solving $PLDY = B$ and then $L^{T}P^{T}X = Y$.

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 how A has been factorized.

uplo = 'U'
$A = PUDU^{T}P^{T}$, where U is upper triangular.

uplo = 'L'
$A = PLDL^{T}P^{T}$, where L is lower triangular.

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

2: **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 $\max(1,\mathbf{n})$.

Details of the factorization of A, as returned by nag_lapack_dsytrf (f07md).

3: **ipiv**$(:)$ – INTEGER array

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

Details of the interchanges and the block structure of D, as returned by nag_lapack_dsytrf (f07md).

4: **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.

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**, **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.

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

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

For each right-hand side vector b, the computed solution x is the exact solution of a perturbed system of equations $(A + E)x = b$, where

if **uplo** = 'U', $|E| \leq c(n)\epsilon P|U||D||U^{\mathrm{T}}|P^{\mathrm{T}}$;

if **uplo** = 'L', $|E| \leq c(n)\epsilon P|L||D||L^{\mathrm{T}}|P^{\mathrm{T}}$,

$c(n)$ is a modest linear function of n, and ϵ is the ***machine precision***.

If $\hat{x}$ is the true solution, then the computed solution x satisfies a forward error bound of the form

$$\frac{\|x-\hat{x}\|_\infty}{\|x\|_\infty} \le c(n)\,\mathrm{cond}(A,x)\epsilon$$

where $\mathrm{cond}(A,x) = \big\||A^{-1}||A||x|\big\|_\infty / \|x\|_\infty \le \mathrm{cond}(A) = \big\||A^{-1}||A|\big\|_\infty \le \kappa_\infty(A)$.

Note that $\mathrm{cond}(A,x)$ can be much smaller than $\mathrm{cond}(A)$.

Forward and backward error bounds can be computed by calling nag_lapack_dsyrfs (f07mh), and an estimate for $\kappa_\infty(A)$ ($=\kappa_1(A)$) can be obtained by calling nag_lapack_dsycon (f07mg).

8 Further Comments

The total number of floating-point operations is approximately $2n^2r$.

This function may be followed by a call to nag_lapack_dsyrfs (f07mh) to refine the solution and return an error estimate.

The complex analogues of this function are nag_lapack_zhetrs (f07ms) for Hermitian matrices and nag_lapack_zsytrs (f07ns) for symmetric matrices.

9 Example

This example solves the system of equations $AX = B$, where

$$A = \begin{pmatrix} 2.07 & 3.87 & 4.20 & -1.15 \\ 3.87 & -0.21 & 1.87 & 0.63 \\ 4.20 & 1.87 & 1.15 & 2.06 \\ -1.15 & 0.63 & 2.06 & -1.81 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} -9.50 & 27.85 \\ -8.38 & 9.90 \\ -6.07 & 19.25 \\ -0.96 & 3.93 \end{pmatrix}.$$

Here A is symmetric indefinite and must first be factorized by nag_lapack_dsytrf (f07md).

9.1 Program Text

```
     function f07me_example

fprintf('f07me example results\n\n');

% Indefinite matrix A (lower triangular part stored)
uplo = 'L';
a = [ 2.07,  0,    0,     0;
       3.87, -0.21, 0,     0;
       4.20,  1.87, 1.15,  0;
     -1.15,  0.63, 2.06, -1.81];

% RHS
b = [-9.50, 27.85;
     -8.38,  9.90;
     -6.07, 19.25;
     -0.96,  3.93];

% Factorize
[af, ipiv, info] = f07md( ...
                          uplo, a);

% Solve
[x, info] = f07me( ...
                   uplo, af, ipiv, b);

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

9.2 Program Results

```
     f07me example results

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