NAG Toolbox for MATLAB

f07pa

1 Purpose

f07pa computes the solution to a real system of linear equations

$$AX = B,$$

where A is an n by n symmetric matrix stored in packed format and X and B are n by r matrices.

2 Syntax

```
[ap, ipiv, b, info] = f07pa(uplo, ap, b, 'n', n, 'nrhs_p', nrhs_p)
```

3 Description

f07pa uses the diagonal pivoting method to factor A as $A = UDU^{\mathrm{T}}$ if **uplo** = 'U' or $A = LDL^{\mathrm{T}}$ if **uplo** = 'L', where U (or L) is a product of permutation and unit upper (lower) triangular matrices, D is symmetric and block diagonal with 1 by 1 and 2 by 2 diagonal blocks. The factored form of A is then used to solve the system of equations $AX = B$.

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 URL: http://www.netlib.org/lapack/lug

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

Higham N J 2002 *Accuracy and Stability of Numerical Algorithms* (2nd Edition) SIAM, Philadelphia

5 Parameters

5.1 Compulsory Input Parameters

1: **uplo – string**

If **uplo** = 'U', the upper triangle of A is stored.

If **uplo** = 'L', the lower triangle of A is stored.

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

2: **ap**$(*)$ **– double array**

Note: the dimension of the array **ap** must be at least $\max(1, \mathbf{n} \times (\mathbf{n}+1)/2)$.

The n by n symmetric matrix A, packed by columns.

More precisely,

if **uplo** = 'U', the upper triangle of A must be stored with element A_{ij} in **ap**$(i + j(j-1)/2)$ for $i \leq j$;
if **uplo** = 'L', the lower triangle of A must be stored with element A_{ij} in **ap**$(i + (2n-j)(j-1)/2)$ for $i \geq j$.

3: **b**(**ldb**,$*$) **– double array**

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

The second dimension of the array 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 – int32 scalar**

n, the number of linear equations, i.e., the order of the matrix A.

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

2: **nrhs_p – int32 scalar**

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 Input Parameters Omitted from the MATLAB Interface

ldb

5.4 Output Parameters

1: **ap**$(*)$ **– double array**

Note: the dimension of the array **ap** must be at least $\max(1, \mathbf{n} \times (\mathbf{n}+1)/2)$.

The block diagonal matrix D and the multipliers used to obtain the factor U or L from the factorization $A = UDU^{\mathrm{T}}$ or $A = LDL^{\mathrm{T}}$ as computed by f07pd, stored as a packed triangular matrix in the same storage format as A.

2: **ipiv**$(*)$ **– int32 array**

Note: 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 determined by f07pd.

ipiv$(k) > 0$

Rows and columns k and **ipiv**(k) were interchanged, and $D(k,k)$ is a 1 by 1 diagonal block.

uplo = 'U' and **ipiv**$(k) =$ **ipiv**$(k-1) < 0$

Rows and columns $k-1$ and $-$**ipiv**(k) were interchanged and $D(k-1:k, k-1:k)$ is a 2 by 2 diagonal block.

uplo = 'L' and **ipiv**$(k) =$ **ipiv**$(k+1) < 0$

Rows and columns $k+1$ and $-$**ipiv**(k) were interchanged and $D(k:k+1, k:k+1)$ is a 2 by 2 diagonal block.

3: **b**(**ldb**,$*$) **– double array**

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

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

If **info** $= 0$, the n by r solution matrix X.

4: **info – int32 scalar**

info $= 0$ unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

info $= -i$

If **info** $= -i$, parameter i had an illegal value on entry. The parameters are numbered as follows:

1: **uplo**, 2: **n**, 3: **nrhs_p**, 4: **ap**, 5: **ipiv**, 6: **b**, 7: **ldb**, 8: **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.

info > 0

If **info** $= i$, d_{ii} is exactly zero. The factorization has been completed, but the block diagonal matrix D is exactly singular, so the solution could not be computed.

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 and Chapter 11 of Higham 2002 for further details.

f07pb is a comprehensive LAPACK driver that returns forward and backward error bounds and an estimate of the condition number. Alternatively, f04bj solves $Ax = b$ and returns a forward error bound and condition estimate. f04bj calls f07pa to solve the equations.

8 Further Comments

The total number of floating-point operations is approximately $\frac{1}{3}n^3 + 2n^2r$, where r is the number of right-hand sides.

The complex analogues of f07pa are f07pn for Hermitian matrices, and f07qn for symmetric matrices.

9 Example

```
uplo = 'U';
ap = [-1.81;
      2.06;
      1.15;
      0.63;
      1.87;
      -0.21;
      -1.15;
      4.2;
      3.87;
      2.07];
b = [0.96;
      6.07;
      8.380000000000001;
```

```
     9.5];
[apOut, ipiv, bOut, info] = f07pa(uplo, ap, b)
```

```
apOut =
    0.4074
    0.3031
   -2.5907
   -0.5960
    0.8115
    1.1500
    0.6537
    0.2230
    4.2000
    2.0700
ipiv =
           1
           2
          -2
          -2
bOut =
   -5.0000
   -2.0000
    1.0000
    4.0000
info =
           0
```