NAG Toolbox for MATLAB

f04fa

1 Purpose

f04fa calculates the approximate solution of a set of real symmetric positive-definite tridiagonal linear equations.

2 Syntax

```
[d, e, b, ifail] = f04fa(job, d, e, b, 'n', n)
```

3 Description

f04fa is based on the LINPACK routine SPTSL (see Dongarra *et al.* 1979) and solves the equations

$$Tx = b,$$

where T is a real n by n symmetric positive-definite tridiagonal matrix, using a modified symmetric Gaussian elimination algorithm to factorize T as $T = MKM^{\mathrm{T}}$, where K is diagonal and M is a matrix of multipliers as described in Section 8.

When the input parameter **job** is supplied as 1, then the function assumes that a previous call to f04fa has already factorized T; otherwise **job** must be supplied as 0.

4 References

Dongarra J J, Moler C B, Bunch J R and Stewart G W 1979 *LINPACK Users' Guide* SIAM, Philadelphia

5 Parameters

5.1 Compulsory Input Parameters

1: **job – int32 scalar**

Specifies the job to be performed by f04fa as follows:

job $= 0$

The matrix T is factorized and the equations $Tx = b$ are solved for x.

job $= 1$

The matrix T is assumed to have already been factorized by a previous call to f04fa with **job** $= 0$; the equations $Tx = b$ are solved for x.

2: **d**(**n**) **– double array**

If **job** $= 0$, **d** must contain the diagonal elements of T.

If **job** $= 1$, **d** must contain the diagonal matrix K, as returned by a previous call of f04fa with **job** $= 0$.

3: **e**(**n**) **– double array**

If **job** $= 0$, **e** must contain the superdiagonal elements of T, stored in **e**(2) to **e**(n).

If **job** $= 1$, **e** must contain the off-diagonal elements of the matrix M, as returned by a previous call of f04fa with **job** $= 0$. **e**(1) is not used.

4: **b(n) – double array**

The right-hand side vector b.

5.2 Optional Input Parameters

1: **n – int32 scalar**

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

n, the order of the matrix T.

Constraint: $\mathbf{n} \geq 1$.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **d(n) – double array**

If $\mathbf{job} = 0$, **d** contains the diagonal matrix K of the factorization.

If $\mathbf{job} = 1$, **d** is unchanged.

2: **e(n) – double array**

If $\mathbf{job} = 0$, $\mathbf{e}(2)$ to $\mathbf{e}(n)$ are overwritten by the off-diagonal elements of the matrix M of the factorization.

If $\mathbf{job} = 1$, **e** is unchanged.

3: **b(n) – double array**

The array contains the solution vector x.

4: **ifail – int32 scalar**

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

6 Error Indicators and Warnings

Errors or warnings detected by the function:

$\mathbf{ifail} = 1$

On entry, $\mathbf{n} < 1$,
or $\mathbf{job} \neq 0$ or 1.

$\mathbf{ifail} = 2$

The matrix T is either not positive-definite or is nearly singular. This failure can only occur when $\mathbf{job} = 0$ and inspection of the elements of **d** will give an indication of why failure has occurred. If an element of **d** is close to zero, then T is probably nearly singular; if an element of **d** is negative but not close to zero, then T is not positive-definite.

overflow

If overflow occurs during the execution of this function, then either T is very nearly singular or an element of the right-hand side vector b is very large. In this latter case the equations should be scaled so that no element of b is very large. Note that to preserve symmetry it is necessary to scale by a transformation of the form $\left(PTP^{\mathrm{T}}\right)b = Px$, where P is a diagonal matrix.

underflow

Any underflows that occur during the execution of this function are harmless.

7 Accuracy

The computed factorization (see Section 8) will satisfy the equation

$$MKM^{\mathrm{T}} = T + E$$

where $\|\mathbf{e}\|_p \leq 2\epsilon\|T\|_p$, $p = 1, F, \infty$,

ϵ being the ***machine precision***. The computed solution of the equations $Tx = b$, say $\bar{x}$, will satisfy an equation of the form

$$(T + F)\bar{x} = b,$$

where F can be expected to satisfy a bound of the form

$$\|F\| \leq \alpha\epsilon\|T\|,$$

α being a modest constant. This implies that the relative error in $\bar{x}$ satisfies

$$\frac{\|\bar{x} - x\|}{\|x\|} \leq c(T)\alpha\epsilon,$$

where $c(T)$ is the condition number of T with respect to inversion. Thus if T is nearly singular, $\bar{x}$ can be expected to have a large relative error.

8 Further Comments

The time taken by f04fa is approximately proportional to n.

The function eliminates the off-diagonal elements of T by simultaneously performing symmetric Gaussian elimination from the top and the bottom of T. The result is that T is factorized as

$$T = MKM^{\mathrm{T}},$$

where K is a diagonal matrix and M is a matrix of the form

$$M = \begin{pmatrix} 1 & 0 & 0 & .. & 0 & 0 & 0 & .. & 0 & 0 & 0 \\ m_2 & 1 & 0 & .. & 0 & 0 & 0 & .. & 0 & 0 & 0 \\ 0 & m_3 & 1 & .. & 0 & 0 & 0 & .. & 0 & 0 & 0 \\ . & . & . & .. & . & . & . & .. & . & . & . \\ . & . & . & .. & . & . & . & .. & . & . & . \\ 0 & 0 & 0 & .. & 1 & 0 & 0 & .. & 0 & 0 & 0 \\ 0 & 0 & 0 & .. & m_{j+1} & 1 & m_{j+2} & .. & 0 & 0 & 0 \\ 0 & 0 & 0 & .. & 0 & 0 & 1 & .. & 0 & 0 & 0 \\ . & . & . & .. & . & . & . & .. & . & . & . \\ . & . & . & .. & . & . & . & .. & . & . & . \\ 0 & 0 & 0 & .. & 0 & 0 & 0 & .. & 1 & m_{n-1} & 0 \\ 0 & 0 & 0 & .. & 0 & 0 & 0 & .. & 0 & 1 & m_n \\ 0 & 0 & 0 & .. & 0 & 0 & 0 & .. & 0 & 0 & 1 \end{pmatrix}$$

j being the integer part of $n/2$. (For example when $n = 5, j = 2$.) The diagonal elements of K are returned in $\mathbf{d}$ with k_i in the ith element of $\mathbf{d}$ and m_i is returned in the ith element of $\mathbf{e}$.

The function fails with **ifail** $= 2$ if any diagonal element of K is nonpositive. It should be noted that T may be nearly singular even if all the diagonal elements of K are positive, but in this case at least one element of K is almost certain to be small relative to $\|T\|$. If there is any doubt as to whether or not T is nearly singular, then you should consider examining the diagonal elements of K.

9 Example

```
job = int32(0);
d = [4;
     10;
     29;
     25;
     5];
e = [-1.893585205078129;
     -2;
     -6;
     15;
     8];
b = [6;
     9;
     2;
     14;
     7];
[dOut, eOut, bOut, ifail] = f04fa(job, d, e, b)
```

```
dOut =
    4.0000
    9.0000
    6.5574
   12.2000
    5.0000
eOut =
   -1.8936
   -0.5000
   -0.6667
    1.2295
    1.6000
bOut =
    2.5000
    2.0000
    1.0000
   -1.0000
    3.0000
ifail =
           0
```