NAG Library Routine Document

F11MGF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

F11MGF computes an estimate of the reciprocal of the condition number of a sparse matrix given an LU factorization of the matrix computed by F11MEF.

2 Specification

```
SUBROUTINE F11MGF (NORM, N, IL, LVAL, IU, UVAL, ANORM, RCOND, IFAIL)

INTEGER             N, IL(*), IU(*), IFAIL
REAL (KIND=nag_wp) LVAL(*), UVAL(*), ANORM, RCOND
CHARACTER(1)        NORM
```

3 Description

F11MGF estimates the condition number of a real sparse matrix A, in either the 1-norm or the ∞-norm:

$$\kappa_1(A) = \|A\|_1 \|A^{-1}\|_1 \qquad \text{or} \qquad \kappa_\infty(A) = \|A\|_\infty \|A^{-1}\|_\infty.$$

Note that $\kappa_\infty(A) = \kappa_1(A^{\mathrm{T}})$.

Because the condition number is infinite if A is singular, the routine actually returns an estimate of the **reciprocal** of the condition number.

The routine should be preceded by a call to F11MLF to compute $\|A\|_1$ or $\|A\|_\infty$, and a call to F11MEF to compute the LU factorization of A. The routine then estimates $\|A^{-1}\|_1$ or $\|A^{-1}\|_\infty$ and computes the reciprocal of the condition number.

4 References

None.

5 Parameters

1: NORM – CHARACTER(1) *Input*

On entry: indicates whether $\kappa_1(A)$ or $\kappa_\infty(A)$ is to be estimated.

NORM = '1' or 'O'
: $\kappa_1(A)$ is estimated.

NORM = 'I'
: $\kappa_\infty(A)$ is estimated.

Constraint: NORM = '1', 'O' or 'I'.

2: N – INTEGER *Input*

On entry: n, the order of the matrix A.

Constraint: $\mathrm{N} \geq 0$.

3: IL(∗) – INTEGER array *Input*

Note: the dimension of the array IL must be at least as large as the dimension of the array of the same name in F11MEF.

On entry: records the sparsity pattern of matrix L as computed by F11MEF.

4: LVAL(∗) – REAL (KIND=nag_wp) array *Input*

Note: the dimension of the array LVAL must be at least as large as the dimension of the array of the same name in F11MEF.

On entry: records the nonzero values of matrix L and some nonzero values of matrix U as computed by F11MEF.

5: IU(∗) – INTEGER array *Input*

Note: the dimension of the array IU must be at least as large as the dimension of the array of the same name in F11MEF.

On entry: records the sparsity pattern of matrix U as computed by F11MEF.

6: UVAL(∗) – REAL (KIND=nag_wp) array *Input*

Note: the dimension of the array UVAL must be at least as large as the dimension of the array of the same name in F11MEF.

On entry: records some nonzero values of matrix U as computed by F11MEF.

7: ANORM – REAL (KIND=nag_wp) *Input*

On entry: if NORM = '1' or 'O', the 1-norm of the matrix A.

If NORM = 'I', the ∞-norm of the matrix A.

ANORM may be computed by calling F11MLF with the same value for the parameter NORM.

Constraint: ANORM ≥ 0.0.

8: RCOND – REAL (KIND=nag_wp) *Output*

On exit: an estimate of the reciprocal of the condition number of A. RCOND is set to zero if exact singularity is detected or the estimate underflows. If RCOND is less than ***machine precision***, A is singular to working precision.

9: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this parameter, the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

On exit: IFAIL $= 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

6 Error Indicators and Warnings

If on entry IFAIL $= 0$ or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, NORM $\neq$ '1', 'O' or 'I',
or N < 0,
or ANORM < 0.0.

IFAIL $= 301$

Unable to allocate required internal workspace.

7 Accuracy

The computed estimate RCOND is never less than the true value ρ, and in practice is nearly always less than 10ρ, although examples can be constructed where RCOND is much larger.

8 Further Comments

A call to F11MGF involves solving a number of systems of linear equations of the form $Ax = b$ or $A^{\mathrm{T}}x = b$.

9 Example

This example estimates the condition number in the 1-norm of the matrix A, where

$$A = \begin{pmatrix} 2.00 & 1.00 & 0 & 0 & 0 \\ 0 & 0 & 1.00 & -1.00 & 0 \\ 4.00 & 0 & 1.00 & 0 & 1.00 \\ 0 & 0 & 0 & 1.00 & 2.00 \\ 0 & -2.00 & 0 & 0 & 3.00 \end{pmatrix}.$$

Here A is nonsymmetric and must first be factorized by F11MEF. The true condition number in the 1-norm is 20.25.

9.1 Program Text

```
    Program f11mgfe

!     F11MGF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: f11mdf, f11mef, f11mgf, f11mlf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Real (Kind=nag_wp), Parameter    :: one = 1.E0_nag_wp
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: anorm, flop, rcond, thresh
      Integer                          :: i, ifail, n, nnz, nnzl, nnzu, nzlmx, &
                                          nzlumx, nzumx
      Character (1)                    :: norm, spec
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: a(:), lval(:), uval(:)
      Integer, Allocatable             :: icolzp(:), il(:), iprm(:),          &
                                          irowix(:), iu(:)
```

```
!     .. Executable Statements ..
      Write (nout,*) 'F11MGF Example Program Results'
!     Skip heading in data file
      Read (nin,*)

!     Read order of matrix

      Read (nin,*) n

      Allocate (icolzp(n+1),iprm(7*n))

!     Read the matrix A

      Read (nin,*) icolzp(1:n+1)
      nnz = icolzp(n+1) - 1

      Allocate (a(nnz),lval(8*nnz),uval(8*nnz),il(7*n+8*nnz+4),irowix(nnz), &
        iu(2*n+8*nnz+1))

      Do i = 1, nnz
        Read (nin,*) a(i), irowix(i)
      End Do

!     Calculate COLAMD permutation

      spec = 'M'

!     ifail: behaviour on error exit
!             =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
      ifail = 0
      Call f11mdf(spec,n,icolzp,irowix,iprm,ifail)

!     Factorise

      thresh = one
      ifail = 0
      nzlmx = 8*nnz
      nzlumx = 8*nnz
      nzumx = 8*nnz

      Call f11mef(n,irowix,a,iprm,thresh,nzlmx,nzlumx,nzumx,il,lval,iu,uval, &
        nnzl,nnzu,flop,ifail)

!     Calculate norm

      norm = '1'

      ifail = 0
      Call f11mlf(norm,anorm,n,icolzp,irowix,a,ifail)

!     Calculate condition number

      ifail = 0
      Call f11mgf(norm,n,il,lval,iu,uval,anorm,rcond,ifail)

!     Output result

      Write (nout,*)
      Write (nout,*) 'Norm  ,Condition number'
      Write (nout,'(F7.3,A1,F7.3)') anorm, ',', 1.0E0_nag_wp/rcond

    End Program f11mgfe
```

9.2 Program Data

```
F11MGF Example Program Data
  5  N
 1
 3
 5
```

```
7
9
12   ICOLZP(I) I=1,..,N+1
 2.   1
 4.   3
 1.   1
-2.   5
 1.   2
 1.   3
-1.   2
 1.   4
 1.   3
 2.   4
 3.   5    A(I), IROWIX(I) I=1,NNZ
```

9.3 Program Results

```
F11MGF Example Program Results

Norm  ,Condition number
 6.000, 20.250
```