1 SUMMARY

This subroutine provides estimates of the classical condition number, $\|\mathbf{A}\|_\infty \|\mathbf{A}^{-1}\|_\infty$, and Skeel's condition number, $\| |\mathbf{A}^{-1}| \, |\mathbf{A}| \|_\infty$, of an $n \times n$ sparse matrix $\mathbf{A}$. We denote by $|\mathbf{A}|$ and $|\mathbf{A}^{-1}|$ the matrices whose entries are the absolute values of the entries in $\mathbf{A}$ and $\mathbf{A}^{-1}$.

ATTRIBUTES — **Version:** 1.1.0. (4 September 2007) **Types:** Real (single, double). **Calls:** MA48, MC71, I_AMAX. **Original date:** September 2001. **Remark:** MC75 is a threadsafe version of MC45. **Origin:** M. Arioli, I.S. Duff, Harwell.

2 HOW TO USE THE PACKAGE

2.1 Initialization

The optional user controls passed in the ICNTL argument to the main entry MC75A/AD should be initialized by calling the MC75I/ID entry.

The single precision version:

        CALL MC75I(ICNTL)

The double precision version:

        CALL MC75ID(ICNTL)

ICNTL is the INTEGER array of length 5 used to pass user options to the MC75A/AD entry.

2.2 Argument list

The single precision version:

        CALL MC75A(N,NZ,LA,A,IRN,JCN,COND,LIW,IW,LW,W,ICNTL,INFO)

The double precision version:

        CALL MC75AD(N,NZ,LA,A,IRN,JCN,COND,LIW,IW,LW,W,ICNTL,INFO)

N   is an INTEGER variable that must be set by the user to the order of the matrix. This argument is not altered by the subroutine. **Restriction:** N > 0;

NZ   is an INTEGER variable that must be set by the user to the number of entries of the matrix. This argument is not altered by the subroutine. **Restriction:** NZ > 0.

LA   is an INTEGER variable that must be set by the user to the length of arrays A, IRN and JCN. Since the arrays are used to store the **LU** factors of $\mathbf{A}$, LA must be large enough to accommodate this. Typically, a value between 4*NZ and NZ*$\log_2$N will suffice. If the specified value is too small, recommended values are returned in INFO(3) and INFO(4) (Section 2.3). This argument is not altered by the subroutine. **Restriction:** LA ≥ NZ.

A   is a REAL (DOUBLE PRECISION in the D version) array of length LA, that must be set by the user to the values of the entries in the matrix $\mathbf{A}$. The entries may be in any order. Entry K has value in A(K), row index in IRN(K) and column index in JCN(K). On output the array holds the values of the entries of the factors of the matrix $\mathbf{A}$.

IRN   is an INTEGER array of length LA. It must be set by the user to hold the row indices of the corresponding entries in the array A. It is used as workspace.

JCN   is an INTEGER array of length LA. On entry, JCN must hold the column indices of the corresponding entries in the array A. It is used as workspace.

`COND` is a `REAL` (`DOUBLE PRECISION` in the D version) array of length 2 that need not be set by the user. On return `COND(1)` holds the value of the classical condition number in the ∞–norm, and `COND(2)` holds the value of Skeel's condition number (but see Section 2.5 for the case when **A** is singular). In the case of an error return, `COND(1)` and `COND(2)` are set to zero.

`LIW` is an `INTEGER` variable which must be set to the length of the integer workspace array `IW`. The length must be at least `19*N+7`.

`IW` is an `INTEGER` array of length `LIW` that is used as workspace.

`LW` is an `INTEGER` variable which must be set to the length of the real workspace array `W`. The length must be at least `8*N`.

`W` is a `REAL` (`DOUBLE PRECISION` in the D version) array of length `LW` that is used as workspace.

`ICNTL` is an `INTEGER` array of length 5 which is used to pass optional control values, see Section 2.3 for details.

`INFO` is an `INTEGER` array of length 5 in which `MC75A/AD` returns information to the user, see Section 2.3 for details.

2.3 The control and information arrays

The array `ICNTL` can be used to set control parameters and should be initialized using the `MC75I/ID` entry.

`ICNTL(1)` is used by the subroutine as the unit number for its warning and error messages. The default value is 6. The user can either reset it to a different stream number or suppress the output by setting it to zero.

`ICNTL(2)` is used by the subroutine as the unit number for warning and error messages from the `MA48` package. The default is to suppress `MA48` output with the value zero.

`ICNTL(3)` to `ICNTL(5)` are not used.

The array `INFO` is used by `MC75A/AD` to return information.

`INFO(1)` is returned set to a negative value when there are errors and to zero or a positive value on a completed calculation. A positive value indicates a warning, see Section 2.4.

`INFO(2)` returns supplementary information when `INFO(1)` indicates an error.

`INFO(3)` in the event of a workspace failure due to an insufficient value of `LA` (`INFO(1) = -1`), `INFO(3)` gives the minimum value of `LA` which would be required for a successful decomposition up to the point at which the failure occurred.

`INFO(4)` in the event of a successful return (`INFO(1) ≥ 0`) `INFO(4)` gives the minimum value for `LA` which would enable another successful run on the matrix. In the event of a workspace failure with `INFO(1) = -1` `INFO(4)` gives the recommended value for success.

`INFO(5)` is an estimate of the rank of the matrix.

Note: `INFO(3)`, `INFO(4)` and `INFO(5)` are in fact set to the values returned by `MA48A/AD` and `MA48B/BD` in its `INFO(3)`, `INFO(4)` and `INFO(5)` entries.

2.4 Errors and diagnostic messages

A successful return from `MC75A/AD` is indicated by a value of `INFO(1)` equal to zero. Possible nonzero values for `INFO(1)` are given below. Negative values identify errors and positive values are warning messages.

–11 an expected error return from `MA48C/CD` (its return value is placed in `INFO(2)`).

–10 an expected error return from `MA48B/BD` (its return value is placed in `INFO(2)`).

–9 an expected error return from `MA48A/AD` (its return value is placed in `INFO(2)`).

–8 there is one or more row or column indices out-of-range.

–7 `N < 1`

−6 `NZ` ≤ 0

−5 `LA` $<$ `NZ`

−4 the length of the integer workspace `IW` is too small, the required length for `LIW` is given by `INFO(2)`.

−3 the length of the real workspace `W` is too small, the required length for `LW` is given by `INFO(2)`.

−2 the factorization step of `MA48` has found `LA` to be too small. Increase `LA` to at least the value given by `INFO(4)`, see Section 2.3

−1 the analyse step of `MA48` has found `LA` to be too small. Increase `LA` to at least the value given by `INFO(3)`, see Section 2.3

+1 The input has two or more entries `A(i)`, $\cdots$,`A(k)` with the same row and column indices. The action taken in this case is to proceed using a numerical value of `A(i)` $+ \cdots +$ `A(k)`.

+2 The matrix is rank deficient with estimated rank `INFO(5)`, see Section 2.5.

2.5 Singular or rectangular matrices

The subroutine can handle singular or rectangular matrices. A rectangular $m \times n$ matrix **B** can be embedded in a structurally singular square matrix of order equal to the maximum of m and n.

If `MA48` detects singularity, it identifies a non-singular submatrix **S** of order `INFO(5)` and in this case `COND(1)` is set to $\| \mathbf{A} \|_\infty \| \mathbf{S}^{-1} \|_\infty$ and `COND(2)` is set to zero.

3 GENERAL INFORMATION

Workspace: Provided by the user, see arguments `IW` and `W`.

Use of common: None.

Other routines called directly: `MC75A/AD` calls `MC75B/BD`, `MA48A/AD`, `MC71A/AD` and `ISAMAX/IDAMAX`.

Input/output: Error and warning messages only on units `ICNTL(1)` and `ICNTL(2)`.

Restrictions: `N` > 0, `NZ` > 0, `LA` $\geq$ `NZ`, and `LA` $\geq$ `NZ`;

4 METHOD

This subroutine is a simple interface that uses `MC71` in combination with `MA48` for the computation of the classical condition number and the condition number introduced by Skeel (1979). The method used by `MC71` is based on that developed by Hager (1984) and the version implemented is based on the modifications suggested by Higham (1987). More details can be obtained from the documentation of `MC71`.

References

Hager, W. W. (1984). Condition Estimates. *SIAM J. Sci. Stat. Comput.* **5**, 311-316.

Higham, N. J. (1987). Fortran codes for estimating the one-norm of a real or complex matrix, with applications to condition estimation. Numerical Analysis Report No. 135, University of Manchester, England.

Skeel, R. D. (1979). Scaling for numerical stability in Gaussian elimination. *JACM* **26**, *494-526.*

5 EXAMPLE OF USE

The example code of Figure 1 shows the use of the subroutine `MC75AD` when it is used to compute the condition numbers of the sparse matrix

$$\begin{pmatrix} 0.3\dot{3}\,10^4 & 0.1\,10^{-10} & & \\ 1 & 0.3\dot{3}\,10^4 & 0.1\,10^{-10} & \\ & 1 & 0.3\dot{3}\,10^4 & 0.1\,10^{-10} \\ & 10^4 & 10^4 & 0.3\dot{3}\,10^9 \end{pmatrix}.$$

When the program of Figure 1 is run on the input

```
    4   11
    1    1    2    2    2    3    3    3    4    4    4
    1    2    1    2    3    2    3    4    2    3    4
 0.33333333333333D+04 0.10000000000000D-10 0.10000000000000D+01
 0.33333333333333D+04 0.10000000000000D-10 0.10000000000000D+01
 0.33333333333333D+04 0.10000000000000D-10 0.10000000000000D+05
 0.10000000000000D+05 0.33333333333333D+09
```

it produces the output

```
        CLASSICAL CONDITION NUM. 1.000D+05
        SKEEL     CONDITION NUM. 7.000D+00
```

```
      INTEGER MAXN,MAXNZ,LA,LW,LIW
      PARAMETER( MAXN=10, MAXNZ=11)
      PARAMETER ( LA=4*MAXNZ, LIW=14*MAXN+7, LW=8*MAXN  )
      DOUBLE PRECISION  A(LA), W(LW), COND(2)
      INTEGER IRN(LA), JCN(LA), IW(LIW), INFO(5)
      INTEGER N,NZ,I,ICNTL(5)
C
C     READ SPARSE MATRIX
C
      READ(5,9300) N, NZ
      READ(5,9300) (IRN(I),I=1,NZ)
      READ(5,9300) (JCN(I),I=1,NZ)
      READ(5,9400) (A(I),I=1,NZ)
C
C      INITIALIZE
C
      CALL MC75ID(ICNTL)
C
C      COMPUTE CONDITION NUMBERS
C
      CALL MC75AD(N,NZ,LA,A,IRN,JCN,COND,LIW,IW,LW,W,ICNTL,INFO)
      IF (INFO(1) .GE. 0) THEN
        WRITE(6,9100) ' CLASSICAL CONDITION NUM. ', COND(1)
        WRITE(6,9100) ' SKEEL    CONDITION NUM. ', COND(2)
      ELSE
        WRITE(6,9200) ' INFO(1)     ', INFO(1)
      ENDIF
 9100 FORMAT(1X,A30,1P,D10.3)
 9200 FORMAT(1X,A10,I10)
 9300 FORMAT(11I5)
 9400 FORMAT(3D21.14)
      STOP
      END
```

Figure 1. Code to compute condition numbers.