1 SUMMARY

This subroutine **finds row and column permutations** that reorder an $m \times n$ sparse matrix to a **bordered block triangular form with full diagonal blocks**, or a **nested bordered block triangular form** with each block itself in bordered block triangular form.

ATTRIBUTES — Version: 1.0.0. (12 July 2004) **Types:** Real (single, double). **Calls:** MC59. **Original date:** February 1987. **Origin:** M. Arioli, I.S. Duff, N.P. Storer, Harwell.

2 HOW TO USE THE PACKAGE

2.1 Argument list

The single precision version:

```
      CALL MC33A(M,N,NZI,NZO,ITYPE,A,IRN,JCN,IP,IQ,IPROF,
     *           IFLAG,IW,IW1,IERR)
```

The double precision version:

```
      CALL MC33AD(M,N,NZI,NZO,ITYPE,A,IRN,JCN,IP,IQ,IPROF,
     *           IFLAG,IW,IW1,IERR)
```

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

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

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

NZO is an INTEGER variable. It will be set by the subroutine to the number of non-duplicate entries in the matrix. The difference NZI-NZO is the number of duplicate or out of range entries in the input data.

ITYPE is an INTEGER variable that must be set by the user. If ITYPE is negative, the matrix must be ordered by columns and the position of the first entry in each column set in the first N entries of IW. If ITYPE is positive, then the matrix must be input in coordinate form in the arrays IRN and JCN. If the absolute value of ITYPE is 3 then the P^3 algorithm (Preassigned Pivot Procedure), which produces the nested bordered block triangular form, is invoked. Any other value will employ the P^5 algorithm (Precautionary Partitioned Preassigned Pivot Procedure), which produces the ordinary bordered block triangular form, with the diagonal blocks completely full. This argument is not altered by the subroutine.

A is a REAL (DOUBLE PRECISION in the D version) array of length NZI that must be set by the user to the entries of the matrix. If ITYPE is positive then these entries can be in any order and, on exit, they will be reordered by columns. If ITYPE is negative, then the entries in A must be ordered so that the entries belonging to a single column are contiguous. The ordering of entries within each column is unimportant. The entries of column J must precede those of column J+1 (J=1, ..., N-1). If there are duplicate or out of range entries in the input data, A will be altered.

IRN is an INTEGER array of length NZI that must be set by the user to the row indices of the entries as ordered in A. If the entries in A are reordered or A is altered because of duplicate or out of range entries, IRN will be altered to correspond.

JCN is an INTEGER array of length NZI. If ITYPE is positive, JCN must hold the column indices of the matrix entries as ordered in A. If the user has preordered the input matrix by columns (ITYPE set negative), then this array need not be set. It is used as a work array and will be altered on exit.

IP is an INTEGER array of length M that need not be set by the user. It will be set by the subroutine to the row permutation matrix, held in packed form so that row IP(I) of the original matrix is row I of the permuted form (I=1, ..., M).

IQ is an INTEGER array of length N that need not be set by the user. It will be set by the subroutine to the column permutation matrix, held in packed form so that column IQ(J) of the original matrix is column J of the permuted form (J=1, ..., N).

IPROF is an INTEGER array of length N that need not be set by the user. It will be set by the subroutine so that IPROF(J) holds the row index of the first entry in column J of the permuted matrix.

IFLAG is an INTEGER array of length 3 that need not be set by the user. Any null rows and columns will be placed at the end of the permuted matrix. IFLAG(1) will be set to the position (in the permuted matrix) of the last non-null row, IFLAG(2) will be set to the position (in the permuted matrix) of the last non-null column, and IFLAG(3) will be set to the dimension of the final border of the permuted form.

IW is an INTEGER array of length M+N that is used as workspace. If ITYPE has been set negative then the first N entries of IW must be set by the user so that IW(J) is the position in IRN of the first entry in column J of the original matrix. If there are duplicate or out of range entries in the input data, IW will be altered. This argument is not altered by the subroutine. **Restriction:** If $\mathrm{ITYPE} < 0$ then $\mathrm{IW(I)} \le \mathrm{IW(I+1)}$, I=1, ..., N.

IW1 is an INTEGER array of length 9*N+3*M that is used as workspace.

IERR is an INTEGER variable that need not be set by the user. On output, a negative value indicates that an error has been detected. If $\mathrm{IERR} > 0$ than the value of IERR is the number of out of range entries in the input data IRN and JCN. See §2.3 for a list of possible values and their meanings.

2.2 Common

No common areas are used by the subroutine.

2.3 Errors and diagnostic messages

A negative entry for IERR indicates that some problem has been found with the user's input. Possible values and their meanings are:

–1 M out of range. $\mathrm{M} \le 0$

–2 N out of range. $\mathrm{N} \le 0$

–3 NZI out of range. $\mathrm{NZI} \le 0$

–4 $\mathrm{ITYPE} < 0$ and $\mathrm{IW(I)} > \mathrm{IW(I+1)}$ for some I.

$\mathrm{IERR} > 0$, IERR is equal to the number of entries out of range.

3 GENERAL INFORMATION

Workspace: Provided by user, see arguments IW, IW1, and JCN.

Use of common: None.

Other routines called directly: MC33A/AD calls MC59A/AD and MC33B/BD which in turn calls either MC33C/CD or MC33D/DD. None of these subroutines need be called directly by the user.

Input/output: None.

Restrictions: $\mathrm{M} > 0$, $\mathrm{N} > 0$, $\mathrm{NZI} > 0$.

4 METHOD

The method used is based on that developed by Hellerman and Rarick (1971) for the P^3 version, and on work of Erisman, Grimes, Lewis, and Poole (1985) for the P^5 version. Both are described by Duff, Erisman, and Reid (1986). In the permuted matrix, the number of columns that project above the diagonal in P^3 is equal to the number in P^5, but some may have been moved forward so that the total projection above the diagonal is reduced.

References

Duff, I. S., Erisman, A. M., and Reid, J. K. (1986). *Direct Methods for Sparse Matrices.* Clarendon Press, Oxford.

Erisman, A. M., Grimes, R. G., Lewis, J. G., and Poole, W. G. Jr. (1985). A structurally stable modification of Hellerman-Rarick's P^4 algorithm for reordering unsymmetric sparse matrices. *SIAM J Numer. Anal.* **22**, 369-385.

Hellerman E., and Rarick, D. C. (1971). Reinversion with preassigned pivot procedure. *Math. Programming* **1**, 195-216.

5 EXAMPLE OF USE

The following example shows the use of the subroutine MC33A. When the program

```
      DOUBLE PRECISION A(25)
      INTEGER IW(20),IW1(90),IFLAG(3)
      INTEGER JCN(25),IRN(25),IP(9),IQ(9)
      INTEGER IPROF(9)
      INTEGER ITYPE,N,M,NZI,I,NZO,IERR
      READ(5,*) ITYPE
      READ(5,*) N,M,NZI
      READ(5,*) (IRN(I),I=1,NZI)
      IF (ITYPE .GT. 0) THEN
      READ(5,*) (JCN(I),I=1,NZI)
      READ(5,*) (A(I),I=1,NZI)
      CALL MC33AD(M,N,NZI,NZO,ITYPE,A,IRN,JCN,IP,IQ,IPROF,IFLAG,IW,IW1,
     *            IERR)
      ELSE
      READ(5,*)(IW(I),I=1,N)
      READ(5,*) (A(I),I=1,NZI)
      CALL MC33AD(M,N,NZI,NZO,ITYPE,A,IRN,JCN,IP,IQ,IPROF,IFLAG,IW,IW1,
     *            IERR)
      ENDIF
      IF (IERR .LT. 0) GO TO 1000
      WRITE(6,300)'ITYPE',ITYPE
      WRITE(6,300)'IPROF',(IPROF(I),I=1,N)
      WRITE(6,300)'IP',(IP(I),I=1,N)
      WRITE(6,300)'IQ',(IQ(I),I=1,N)
      WRITE(6,300)'IFLAG', IFLAG(1),IFLAG(2),IFLAG(3)
 300  FORMAT(A8,10I4)
      STOP
1000  WRITE(6,*) 'IERR',IERR
      STOP
      END
```

is run on the input

```
-5
5 5 20
1 2 3 4 5 1 2 3 4 5 1 2 3 5 2 3 4 5 4 5
1 6 11 15 19
 1.  2.  3.  4.  5.  6.  7.  8.  9. 10.
11. 12. 13. 14. 15. 16. 17. 18. 19. 20.
```

it produces the output

```
ITYPE  -5
IPROF   1   2   3   1   1
   IP   1   2   4   3   5
   IQ   3   4   5   1   2
IFLAG   5   5   2
```

The pattern of the input matrix is:

$$\begin{array}{ccccc} \times & \times & \times & \cdot & \cdot \\ \times & \times & \times & \times & \cdot \\ \times & \times & \times & \times & \cdot \\ \times & \times & \cdot & \times & \times \\ \times & \times & \times & \times & \times \end{array}$$

and the pattern of the permuted matrix is:

$$\begin{array}{ccccc} \times & \cdot & \cdot & \times & \times \\ \times & \times & \cdot & \times & \times \\ \cdot & \times & \times & \times & \times \\ \times & \times & \cdot & \times & \times \\ \times & \times & \times & \times & \times \end{array}$$