



1 SUMMARY

Given the pattern of nonzeros of a **sparse matrix** **A**, finds a **symmetric permutation** that makes the matrix block lower triangular, i.e. finds **P** such that $\mathbf{L} = \mathbf{PAP}^T$ is block lower triangular.

The method is that of Tarjan, SIAM J. Computing, **1** (1972), 146-160, and is also described by Duff and Reid, ACM TOMS, **4** (1978), 137-147.

ATTRIBUTES — **Version:** 1.0.0. **Types:** Real (single, double). **Original date:** March 1976. **Origin:** I. S. Duff, Harwell.

2 HOW TO USE THE PACKAGE

2.1 Argument list

The single precision version

```
CALL MC13D(N, ICN, LICN, IP, LENR, IOR, IB, NUM, IW)
```

The double precision version

```
CALL MC13DD(N, ICN, LICN, IP, LENR, IOR, IB, NUM, IW)
```

N is an **INTEGER** variable which must be set by the user to the order of the matrix. This argument is not altered.

ICN is an **INTEGER** array of length **LICN**, which must be set by the user to contain the column indices of the nonzeros. Those belonging to a single row must be contiguous, but the ordering of column indices within each row is unimportant, and wasted space between rows is permitted. This argument is not altered.

LICN is an **INTEGER** variable which must be set by the user to the length of array **ICN**. This argument is not altered.

IP is an **INTEGER** array of length **N** which must be set by the user so that **IP(I)** contains the position in array **ICN** of the first column index of a nonzero in row **I**, for **I** = 1, 2, ..., **N**. This argument is not altered.

LENR is an **INTEGER** array of length **N**, which must be set by the user so that **LENR(I)** is equal to the number of nonzeros in row **I**, **I** = 1, 2, ..., **N**. This argument is not altered.

IOR is an **INTEGER** array of length **N** which need not be set by the user. It is set by the routine to the permutation matrix **P**, in packed form. Row or column **IOR(I)** of the original ordering is row or column **I** of the permuted form.

IB is an **INTEGER** array of length **N** which need not be set by the user. It is set by the routine so that **IB(I)** contains the row number in the permuted matrix of the beginning of block **I**, **I** = 1, 2, ..., **NUM**.

NUM is an **INTEGER** variable which need not be set by the user. It is set by the routine to the number of blocks in the permuted form.

IW is an **INTEGER** array, of length at least 3*N, that is used as workspace.

2.2 Errors and diagnostic messages

There are no error returns. However, if the user submits a matrix with zeros on the diagonal, MC13D might give a block triangular form which could be further reduced by unsymmetric permutations. To obtain the best results, the user is advised to first permute the matrix so that it has a zero-free diagonal. This can be done by subroutine MC21.

3 GENERAL INFORMATION

Workspace: Provided by user, see argument IW.

Use of common: None.

Other routines called directly: Internal to the package: MC13E/ED.

Input/output: None.

Restrictions: None.

4 METHOD

The method used is that of Tarjan (SIAM J. Computing (1972), **1**, pp. 146-160) and is described by Duff and Reid (in ACM Trans Math Software, **4** (1978), pp. 137-147).

5 EXAMPLE OF USE

An example to reduce a matrix to block triangular form

In the example code shown below, we read in the entries of a sparse matrix by rows. We then call the routine MC13D to attempt to reduce the input matrix into block triangular form, using a symmetric permutation.

```

      INTEGER MAXN,MAXNZ,LICN
      PARAMETER( MAXN = 5, MAXNZ = MAXN*MAXN, LICN = 4*MAXNZ )
      INTEGER ICN(LICN), N, I, J, NUM,
+         LENR(MAXN), IP(MAXN), IB(MAXN), IOR(MAXN), IW(3*MAXN)
      READ(5, *) N
C     IP(I) POINTS TO THE START OF THE I'TH ROW OF THE MATRIX
C     LENR(I) HOLDS THE NUMBER OF ENTRIES IN THE I'TH ROW
      IP(1) = 1
      DO 10 I=1,N
         READ(5, *) LENR(I), (ICN(J),J=IP(I),IP(I)+LENR(I)-1)
         IP(I+1) = IP(I) + LENR(I)
10    CONTINUE
      CALL MC13DD( N, ICN, LICN, IP, LENR, IOR, IB, NUM, IW)
      WRITE(6,20) NUM, (IOR(I),I=1,N)
20    FORMAT('/' THE NUMBER OF BLOCKS IN THE PERMUTED MATRIX IS ', I2 ,
+         / ' THE PERMUTATION ARRAY IP() IS:' , 4I2)
      STOP
      END

```

Thus if, in this example, we wish to reduce the following matrix:

$$\begin{pmatrix} 1 & 0 & 0 & 4 \\ 0 & 6 & 0 & 8 \\ 9 & 0 & 11 & 0 \\ 0 & 14 & 0 & 16 \end{pmatrix}$$

we could have as input

4		
2	1	4
2	4	2
2	4	1
2	2	4

and we would get the following output

```
THE NUMBER OF BLOCKS IN THE PERMUTED MATRIX IS 3
```

THE PERMUTATION ARRAY `IP()` IS: 2 4 1 3

The array `IP()` represents the following permutation matrices

$$\mathbf{P} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad \mathbf{P}^T = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

and the resulting triangularised matrix \mathbf{L} is

$$\mathbf{L} = \begin{pmatrix} 6 & 8 & 0 & 0 \\ 14 & 16 & 0 & 0 \\ 0 & 4 & 1 & 0 \\ 0 & 0 & 9 & 11 \end{pmatrix}.$$