資 料

〔衛 生 化 学〕
〔EISEI KAGAKU〕
〔37 (4) 307-314 (1991)〕

—Data—

A BASIC Program to Calculate Molecular Connectivity Index

YOSHITADA YOSHIOKA

*Gifu College of Medical Technology, 897-5 Aza-nagamine,
Ichihiraga, Seki 501-32, Japan*

(Received January 11, 1991)

A BASIC program was developed to calculate the molecular connectivity index (MCI). This program can be characterized as follows: 1) Simple data input method, 2) Automatic calculation of delta value, 3) Calculation of all types of MCIs (path, cluster, path & cluster and chain) up to 50 order, and 4) Fast calculation using the relational table of the atoms in the chemical structure.

Keywords — molecular connectivity index ; program ; BASIC ; algorithm

Introduction

Pollution of the environment must be continuously monitored. Since several tens of thousands of chemicals are on the market and some thousands of new ones will enter the market every year, it is urgent to establish a mean of identifying hazardous chemicals and estimating their toxicity to the environment. Quantitative structure activity relationships (QSAR) help in the toxicity assessment, although in a QSAR study, it is difficult to find an adequate descriptor which can show the chemical and structural characteristics. The descriptor of chemical characteristics used in assessing toxicity is usually *n*-octanol/water partition coefficient (logP) based on the theory of transportation through the membranes.¹⁾ But it is not easy to obtain logP value from the literature because the chemicals detectable in the environment are of a wide variety and many of them are comparatively new. Many researchers have made efforts to find and develop simple descriptors which correlate to the toxicity and

can be easily calculated from the chemical structure. The molecular connectivity index (MCI) is a non-parametric descriptor developed by Hall and Kier.²⁾ This has been widely used in QSAR analysis because of the simple definition of the calculating method, the good differentiation ability of similar chemical structures and the good correspondence with toxicity.³⁾ Although the low order MCI value is easily calculated for a small molecule, it is hard to calculate a high order MCI value, particularly for a large molecule. For this reason, low order MCIs (zero and first order) have frequently been used in QSAR. But in some cases higher order MCI may correlate to the toxicity. Therefore, there is a demand for computer calculation of MCIs for efficient investigations. The author has developed a program with BASIC which will be a help to researchers.

Theory

The molecular connectivity index (MCI ; χ)¹⁾ is calculated from a non-hydrogen chemical structure. A delta value (δ) and a valence

delta value (δ^v) are assigned to each atom in the structure. δ is defined as the number of adjacent atoms. δ^v is calculated from $\delta^v = (Z^v - h) / (Z - Z^v - 1)$, where Z , Z^v and h mean the atomic number, the number of the valence electrons and the number of the conjunct hydrogen, respectively. The chemical structure is then divided into four types of sub-structure, allowing a duplicate selection; these are path, cluster, path & cluster and chain. Path type is a linear combination of the atoms (for example, o-o-o-o-o). Cluster type is a linear combination of the simplest cluster unit ($\text{o} - \text{O} - \text{o}$) which has three branched bonds with four atoms (for example, $\text{o} - \text{O} - \text{O} - \text{O} - \text{o}$). Path & cluster type is a combination of a path type and one simplest cluster unit (for example, $\text{o} - \text{o} - \text{O} - \text{o}$). Chain is a ring type. The order is the number of bonds included in a sub-structure. MCI (χ and χ^v) is calculated from the following definitions for each of the types (t) and the order (m), using δ and δ^v values.

$${}^m\chi_t = \sum_{s=1}^n \left(\prod_{i=1}^{m+1} \delta_i \right)^{-1/2} \quad {}^m\chi_t^v = \sum_{s=1}^n \left(\prod_{i=1}^{m+1} \delta_i^v \right)^{-1/2}$$

The problem is how to find all of the combinations of the atoms. The algorithm

first employed was a one-line-drawing. Tracing the bonds to all directions every time there was a new atom, MCI values were calculated according to the path. This algorithm required a long calculation time. Then, the algorithm of relational data was employed. Table I shows how to make a relational table with this algorithm.

Chemical structural data is put in the matrix. The atoms are numbered serially. The relations between the atoms are tabulated and the dummy atom is ignored. In the table, the serial number of the atom accords to the column number and the column data includes the serial number of the atom connected to the atom of the column. The search of the path (a series of the connected atoms) begins from the first column. MCI values are calculated when the search of the new path is successful. The path becomes longer or shorter by the search. For example, starting from the serial number one (column 1), the path becomes 1-2-5-8. If the path comes to the edge, backing up to the previous column (5), another path is examined (1-2-5-6-7-3). After examination of all the paths from the column (1), the next column (2) is selected as the

TABLE I. Relational Table of 2-Methylcyclopentanol

Structure ^{a)}	Serial number ^{b)}	Relational table ^{c)}								
		Row/Col.	1	2	3	4*	5	6	7	8
O	1	1	2	1	2	0	2	5	3	5
3	/	2	0	5	7	0	8	7	6	0
C 1 C 1 9	2-3-4*	3	0	3	0	0	6	0	0	0
1		4	0	0	0	0	0	0	0	0
C 1 C 1 C	5-6-7	5	0	0	0	0	0	0	0	0
4	\	6	0	0	0	0	0	0	0	0
C	8	7	0	0	0	0	0	0	0	0
		8	0	0	0	0	0	0	0	0

a) The atoms in the chemical structure of 7×5 matrix should be replaced by the delta - V values. Input 9 for a dummy atom. The number of the direction is 8 in this case because two oblique lines (O-3-C and C-4-C) are employed. The number of the direction defines the number of rows in a relational table. If four is inputted as the direction, the maximum row number is 4.

b) Serial number assigned to the atoms. 4* is a dummy atom.

c) The serial number of the atom is in accord with the column number of the table. The dummy atom is ignored in making the relational table. These save calculation time.

```

10 REM >>> Programed by Yoshitada Yoshioka, December 1990
20 OPTIONBASE=1:DEFINT I-N:DUMMY=9
30 DIM TYPE$(4),GB(2),GC(2),ANS(2,4,50),NANS(3,50),WA(50),QQ$(23)
40 TYPE$(1)=" PATH      ":TYPE$(2)=" CLUSTER ":TYPE$(3)=" P AND CL"
50 TYPE$(4)=" CHAIN      ":LPW=80:WIDTH "LPT1:",LPW
60 REM >>> If width of printer is 132, change LPW=132
70 REM >>> *****
80 CLS:LOCATE 1,1:PRINT SPC(5) "*" * "*" * "*" Calculation of molecular connectivity
  index(MCI) * * * * *
90 PRINT SPC(20) "Order of MCI is limited to 50.":PRINT
100 REM >>> Data input from keyboard.
110 GOSUB 2510:DIM Y(JMAX,IMAX),M(JMAX,IMAX):GOSUB 2610
120 REM >>> Print out the input data on CRT and by printer
130 GOSUB 210:GOSUB 350
140 REM >>> Relational data base, Calculation of MCI
150 GOSUB 500
160 REM >>> Print out the result by printer
170 GOSUB 2380
180 GOTO 80
190 REM >>> *****
200 REM >>> --- Print out the input data on CRT
210 CLS:PRINT USING "Name=&                                     & Data matrix #"
  ### Directions=###";NMAE$,JMAX,IMAX,MD48
220 PRINT "delta-V value is multiplied by 100,  #=dummy, *=bond":PRINT
230 FOR J=1 TO JMAX STEP 2:FOR I=1 TO IMAX STEP 2
240   IF Y(J,I)>0 AND Y(J,I)<DUMMY THEN PRINT USING "###";Y(J,I)*100;
250   IF Y(J,I)<=0 THEN PRINT " ";
260   IF Y(J,I)=DUMMY THEN PRINT " # ";
270   IF I<IMAX THEN IF Y(J,I+1)>0 THEN PRINT " * "; ELSE PRINT " ";
280 NEXT I:PRINT:IF J=JMAX THEN 320
290 FOR I=1 TO IMAX
300   IF Y(J+1,I)>0 THEN PRINT " * "; ELSE PRINT " ";
310 NEXT I:PRINT
320 NEXT J
330 RETURN
340 REM >>> --- Print out the input data by printer
350 LPRINT:LPRINT
360 LPRINT "Chemical name:";NMAE$:LPRINT USING " Data matrix= ### No. of d
  irections=###";JMAX,IMAX,MD48:LPRINT
370 LPRINT "delta-V value is multiplied by 100,  #=dummy *=bond":LPRINT
380 LPRINT "delta-V value";STRING$(20,"-")
390 FOR J=1 TO JMAX STEP 2:FOR I=1 TO IMAX STEP 2
400   IF Y(J,I)>0 AND Y(J,I)<DUMMY THEN LPRINT USING "###";Y(J,I)*100;
410   IF Y(J,I)<=0 THEN LPRINT " ";
420   IF Y(J,I)=DUMMY THEN LPRINT " # ";
430   IF I+1<=IMAX THEN IF Y(J,I+1)>0 THEN LPRINT " * "; ELSE LPRINT " ";
440 NEXT I:LPRINT:IF J=JMAX THEN GOTO 480
450 FOR I=1 TO IMAX
460   IF J+1<=JMAX THEN IF Y(J+1,I)>0 THEN LPRINT " * "; ELSE LPRINT " ";
470 NEXT I:LPRINT
480 NEXT J
490 RETURN
500 REM >>> --Initialization,relational data base,calculation of delta value
510 FOR JA=1 TO 2:GB(JA)=0:GC(JA)=0:NEXT JA
520 FOR JA=1 TO 50:WA(JA)=0
530   FOR JB=1 TO 3:NANS(JB,JA)=0:NEXT JB
540   FOR JB=1 TO 4:FOR JC=1 TO 2:ANS(JC,JB,JA)=0:NEXT JC,JB
550 NEXT JA:KZ=0
560 FOR J=1 TO JMAX STEP 2:FOR I=1 TO IMAX STEP 2
570   IF Y(J,I)>0 THEN KZ=KZ+1:M(J,I)=KZ
580 NEXT I,J
590 DIM IH(KZ,MD48),XS(KZ,2)
600 M1=0
610 FOR J=1 TO JMAX STEP 2:FOR I=1 TO IMAX STEP 2
620   IF Y(J,I)<=0 THEN 810 ELSE L=0:M1=M1+1
630   IF Y(J,I)>=DUMMY THEN XS(M1,1)=-1000! ELSE XS(M1,1)=Y(J,I)
640   FOR M2=1 TO MD48
650     J1=J:J2=J:I1=I:I2=I:IH(M1,M2)=0
660     ON M2 GOTO 670,680,690,700,710,720,730,740
670     J1=J-1:J2=J1-1:GOTO 750
680     I1=I-1:I2=I1-1:GOTO 750

```

```

690 J1=J+1:J2=J1+1:GOTO 750
700 I1=I+1:I2=I1+1:GOTO 750
710 J1=J-1:J2=J1-1:I1=I+1:I2=I1+1:GOTO 750
720 J1=J-1:J2=J1-1:I1=I-1:I2=I1-1:GOTO 750
730 J1=J+1:J2=J1+1:I1=I-1:I2=I1-1:GOTO 750
740 J1=J+1:J2=J1+1:I1=I+1:I2=I1+1
750 IF J2<=0 OR J2>JMAX OR I2<=0 OR I2>IMAX THEN 800
760 IF Y(J1,I1)<.01 OR Y(J2,I2)<=0 THEN 800
770 IF M2=5 OR M2=7 THEN IF Y(J1,I1)><3 OR Y(J2,I2)<=0 THEN 800
780 IF M2=6 OR M2=8 THEN IF Y(J1,I1)><4 OR Y(J2,I2)<=0 THEN 800
790 L=L+1:IH(M1,L)=M(J2,I2)
800 NEXT M2
810 NEXT I,J
820 ERASE Y,M
830 DIM NK(KZ+1),NI(KZ+1),GA(KZ),ID(KZ+1),KA(KZ,MD48)
840 FOR J=1 TO KZ:II=0
850 IF XS(J,1)<-999! THEN XS(J,2)=-1000!:GOTO 900
860 FOR I=1 TO MD48
870 IF IH(J,I)>0 THEN II=II+1 ELSE GOTO 890
880 NEXT I
890 XS(J,2)=II
900 NEXT J
910 FOR J=1 TO KZ:IF XS(J,1)>-999! THEN 970
920 JK1=IH(J,1):IH(J,1)=-1000:JK2=IH(J,2):IH(J,2)=-1000
930 FOR JJ=1 TO MD48
940 IF IH(JK1,JJ)=J THEN IH(JK1,JJ)=JK2
950 IF IH(JK2,JJ)=J THEN IH(JK2,JJ)=JK1
960 NEXT JJ
970 NEXT J
980 REM >>> ===== Calculation of MCI =====
990 MG=0:PRINT:PRINT "Calculation of Cluster(order=3)"
1000 FOR J1=1 TO KZ
1010 IF XS(J1,2)<3! THEN 1080 ELSE KOSU=XS(J1,2):MG=4:PRINT J1;
1020 FOR J3=1 TO KOSU-2:FOR J4=J3+1 TO KOSU-1:FOR J5=J4+1 TO KOSU
1030 FOR J2=1 TO 2
1040 A=XS(J1,J2)*XS(IH(J1,J3),J2)*XS(IH(J1,J4),J2)*XS(IH(J1,J5),J2)
1050 ANS(J2,2,4)=ANS(J2,2,4)+1!/SQR(A)
1060 NEXT J2:NANS(2,4)=NANS(2,4)+1
1070 NEXT J5,J4,J3
1080 NEXT J1
1090 PRINT:PRINT "Calculation of Cluster(order>3)"
1100 FOR J=1 TO KZ
1110 IF XS(J,2)<3! THEN 1550 ELSE PRINT J;
1120 KBAN=1:ID(1)=J
1130 NI(KBAN)=0
1140 IF KBAN=0 THEN 1550
1150 NI(KBAN)=NI(KBAN)+1
1160 IF NI(KBAN)>MD48 THEN KBAN=KBAN-1:GOTO 1140
1170 IF IH(ID(KBAN),NI(KBAN))<=0 THEN KBAN=KBAN-1:GOTO 1140
1180 IF XS(IH(ID(KBAN),NI(KBAN)),2)<3! THEN GOTO 1150
1190 ID(KBAN+1)=IH(ID(KBAN),NI(KBAN)):KBAN=KBAN+1
1200 FOR JJ=1 TO KBAN-1
1210 IF ID(KBAN)=ID(JJ) THEN KBAN=KBAN-1:GOTO 1150
1220 NEXT JJ
1230 IF ID(KBAN)<ID(1) THEN GOTO 1130
1240 FOR JJ=1 TO KBAN:FOR II=1 TO MD48
1250 KA(JJ,II)=IH(ID(JJ),II)
1260 NEXT II,JJ
1270 FOR JA=1 TO KBAN:FOR JJ=1 TO MD48:FOR JB=1 TO KBAN
1280 IF KA(JA,JJ)=ID(JB) THEN KA(JA,JJ)=-1000
1290 NEXT JB,JJ,JA
1300 FOR JA=1 TO KBAN
1310 L=0:FOR JJ=1 TO MD48-1
1320 IF KA(JA,JJ)<KA(JA,JJ+1) THEN SWAP KA(JA,JJ),KA(JA,JJ+1):L=1
1330 NEXT JJ:IF L=1 THEN 1310
1340 NEXT JA
1350 FOR JA=1 TO KBAN-1:JB=0
1360 JB=JB+1:IF JB>MD48 THEN 1430
1370 IF KA(JA,JB)<=0 THEN 1430
1380 FOR JC=JA+1 TO KBAN:JD=0
1390 JD=JD+1:IF JD>MD48 THEN 1420

```

```

1400 IF KA(JC,JD)=<0 THEN 1420
1410 IF KA(JA,JB)=KA(JC,JD) THEN 1540 ELSE 1390
1420 NEXT JC:GOTO 1360
1430 NEXT JA
1440 FOR J2=1 TO KBAN:NK(J2)=0:FOR J1=1 TO MD48
1450 IF KA(J2,J1)>0 THEN NK(J2)=NK(J2)+1 ELSE 1470
1460 NEXT J1
1470 NEXT J2
1480 IF NK(1)<2 OR NK(KBAN)<2 THEN 1130
1490 IF KBAN<3 THEN 1530
1500 FOR J1=2 TO KBAN-1
1510 IF NK(J1)<1 THEN 1130
1520 NEXT J1
1530 GOSUB 1810:IF MG<KBAN*2+2 THEN MG=KBAN*2+2
1540 GOTO 1130
1550 NEXT J
1560 PRINT:PRINT "Calculation of path,chain and path & cluster"
1570 FOR J1=1 TO KZ:IF IH(J1,1)=<0 THEN 1780 ELSE PRINT J1;
1580 ID(1)=J1:NK(1)=0:NX=1:GOTO 1760
1590 NK(NX)=NK(NX)+1
1600 IF NK(NX)>MD48 THEN NX=NX-1:IF NX=<0 THEN 1780 ELSE 1590
1610 IF IH(ID(NX),NK(NX))<=0 THEN NX=NX-1:IF NX=<0 THEN 1780 ELSE 1590
1620 IF NX>=2 AND IH(ID(NX),NK(NX))=ID(NX-1) THEN 1590
1630 ID(NX+1)=IH(ID(NX),NK(NX)):NX=NX+1:NK(NX)=0
1640 IF ID(1)<>ID(NX) THEN 1720
1650 REM >>> --Calculation of chain
1660 IF MG<NX THEN MG=NX
1670 NX=NX-1
1680 FOR M1=1 TO 2:Q=1
1690 FOR M2=1 TO NX:Q=Q*XS(ID(M2),M1):NEXT M2
1700 ANS(M1,4,NX+1)=ANS(M1,4,NX+1)+1!/SQR(Q)/NX/2!
1710 NEXT M1:WA(NX+1)=WA(NX+1)+1!/NX/2!:GOTO 1590
1720 IF NX<4 THEN GOTO 1760
1730 FOR J2=2 TO NX-1
1740 IF ID(NX)=ID(J2) THEN NX=NX-1:GOTO 1590
1750 NEXT J2
1760 REM >>> -- Sub. for path and path & cluster
1770 GOSUB 2110:GOTO 1590
1780 NEXT J1:PRINT
1790 RETURN
1800 REM >>> --- Sub. for cluster(edge)
1810 FOR J1=1 TO 2:GA(J1)=1
1820 FOR J2=1 TO KBAN:GA(J1)=GA(J1)*XS(ID(J2),J1)
1830 NEXT J2,J1
1840 FOR J1=1 TO NK(1)-1:FOR J2=J1+1 TO NK(1)
1850 FOR J3=1 TO NK(KBAN)-1:FOR J4=J3+1 TO NK(KBAN):FOR J5=1 TO 2
1860 GB(J5)=GA(J5)*XS(KA(1,J1),J5)*XS(KA(1,J2),J5)*XS(KA(KBAN,J3),J5)*XS(KA(KBAN,J4),J5)
1870 GC(J5)=GB(J5):NEXT J5
1880 IF KBAN=<2 THEN GOSUB 2060:GOTO 2030
1890 REM >>> --- SUB. for cluster(intermediate)
1900 FOR J5=2 TO KBAN-1:NK(J5)=1:NEXT J5
1910 GOTO 1970
1920 NFG=2
1930 NK(NFG)=NK(NFG)+1
1940 IF KA(NFG,NK(NFG))<=0 THEN NFG=NFG+1:IF NFG>KBAN-1 THEN 2030 ELSE 1930
1950 IF NFG=2 THEN 1970
1960 FOR J5=2 TO NFG-1:NK(J5)=1:NEXT J5
1970 J5=0
1980 J5=J5+1:GC(J5)=GB(J5):J6=1
1990 J6=J6+1:GC(J5)=GC(J5)*XS(KA(J6,NK(J6)),J5)
2000 IF J6<KBAN-1 THEN 1990
2010 IF J5<2 THEN 1980
2020 GOSUB 2060:GOTO 1920
2030 NEXT J4,J3,J2,J1
2040 RETURN
2050 REM >>> --- Sub. memo of result(cluster)
2060 FOR J7=1 TO 2
2070 ANS(J7,2,KBAN*2+2)=ANS(J7,2,KBAN*2+2)+1!/SQR(GC(J7))
2080 NEXT J7:NANS(2,KBAN*2+2)=NANS(2,KBAN*2+2)+1
2090 RETURN

```

```

2100 REM >>> --- Sub. for path
2110 IF ID(1)>ID(NX) THEN 2170
2120 FOR M1=1 TO 2:GA(M1)=1!
2130 FOR M2=1 TO NX:GA(M1)=GA(M1)*XS(ID(M2),M1)
2140 NEXT M2:ANS(M1,1,NX)=ANS(M1,1,NX)+1!/SQR(GA(M1))
2150 NEXT M1:NANS(1,NX)=NANS(1,NX)+1
2160 IF MG<NX THEN MG=NX
2170 IF NX<=2 OR XS(ID(1),2)=<2! THEN RETURN
2180 FOR M1=1 TO 2:GA(M1)=1!
2190 FOR M2=1 TO NX:GA(M1)=GA(M1)*XS(ID(M2),M1)
2200 NEXT M2,M1
2210 REM >>> --- Sub. for path & cluster
2220 FOR M1=1 TO MD48:KA(1,M1)=-1000:NEXT M1:MSU=0
2230 FOR M1=1 TO MD48:IF IH(ID(1),M1)=<0 THEN 2260
2240 FOR M2=2 TO NX:IF IH(ID(1),M1)=ID(M2) THEN 2260
2250 NEXT M2:MSU=MSU+1:KA(1,MSU)=IH(ID(1),M1)
2260 NEXT M1
2270 IF MSU<=1 THEN RETURN
2280 M3=1:M4=2
2290 FOR M1=1 TO 2
2300 GB(M1)=GA(M1)*XS(KA(1,M3),M1)*XS(KA(1,M4),M1)
2310 ANS(M1,3,NX+2)=ANS(M1,3,NX+2)+1!/SQR(GB(M1))
2320 NEXT M1:NANS(3,NX+2)=NANS(3,NX+2)+1
2330 IF MG<NX+2 THEN MG=NX+2
2340 M4=M4+1
2350 IF M4>MSU THEN M3=M3+1:M4=M3+1
2360 IF M4>MSU THEN RETURN ELSE 2290
2370 REM >>> --- Print out the result by printer
2380 NS=1
2390 NE=NS+INT(LPW/10-2):IF NE>MG THEN NE=MG
2400 LPRINT:LPRINT "order ";
2410 FOR I=NS TO NE:LPRINT USING "      ###";I-1;:NEXT I
2420 LPRINT:LPRINT " X-V value"
2430 FOR J1=1 TO 2:FOR J2=1 TO 4:LPRINT TYPE$(J2);
2440 FOR J3=NS TO NE:LPRINT USING "####.####";ANS(J1,J2,J3);
2450 NEXT J3:LPRINT:NEXT J2
2460 IF J1=1 THEN LPRINT " X value "
2470 NEXT J1
2480 IF NE<MG THEN NS=NE+1:GOTO 2390
2490 ERASE IH,XS,NK,NI,GA,ID,KA
2500 RETURN
2510 PRINT SPC(6) "* * Input of structural data(delta-V and bond) from keyboard
* * "
2520 PRINT SPC(12) "Line input. Each value should be separated by a comma(,)."
2530 PRINT SPC(19) "Delta-V value(9) is a dummy atom.":PRINT
2540 INPUT "Name of chemical";NMAE$:PRINT
2550 INPUT "No. of Directions(4 OR 8)";MD48:IF MD48<>4 AND MD48<>8 THEN 2550 ELS
E PRINT
2560 INPUT "Max. of row numbers in matrix(odd number:If =0 then stop)";JMAX:IF
JM AX=0 THEN STOP ELSE PRINT
2570 IF JMAX=INT(JMAX/2)*2 THEN 2560
2580 INPUT "Max. of column numbers in matrix(odd number)";IMAX:PRINT
2590 IF IMAX=INT(IMAX/2)*2 THEN 2580
2600 RETURN
2610 H1$="          1 - - - - - 2 - - - - - 3 - -
- - -"
2620 H2$="          1,2,3,4,5,6,7,8,9,0,1,2,3,4,5,6,7,8,9,0,1,2
,3,4,5"
2630 H3$="0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0"
2640 CLS:LOCATE 1,1:PRINT LEFT$(H1$,IMAX*2+6):LOCATE 2,1:PRINT LEFT$(H2$,IMAX*2+
6)
2650 FOR J=1 TO JMAX:QQ$(J)=LEFT$(H3$,IMAX*2)
2660 LOCATE J+2,1:PRINT USING "No.==##:";J;:PRINT QQ$(J)
2670 NEXT J
2680 LOCATE 22,1:INPUT "Push RETURN key after data input.":A$:A$=""
2690 FOR J=1 TO JMAX:NK=0:QQ$(J)=""
2700 FOR I=8 TO 80
2710 B$=CHR$(SCREEN(J+2,I,0))
2720 IF B$=" " THEN 2760
2730 IF B$="," THEN NK=NK+1:Y(J,NK)=VAL(A$):QQ$(J)=QQ$(J)+A$+",":A$="":IF NK=I
MAX THEN 2760 ELSE 2750

```

```

2740   A$=A$+B$
2750   NEXT I
2760   NEXT J
2770   GOSUB 210
2780   LOCATE 23,35:INPUT "OK=0      NO=1 ";LM:IF LM=1 THEN 2640
2790   RETURN

```

Fig. 1. List of a Program for Molecular Connectivity Index Calculation

```

2610   HOR1$="          1 - - - - - 2 - - - - - 3 -
- - - - -"
2620   HOR2$="          1,2,3,4,5,6,7,8,9,0,1,2,3,4,5,6,7,8,9,0,1
,2,3,4,5"
2630   CLS:PRINT LEFT$(HOR1$,IMAX*2+6):PRINT LEFT$(HOR2$,IMAX*2+6)
2640   FOR J=1 TO JMAX
2650   PRINT USING "No. = ##: ";J;:LINE INPUT JX$:IF JX$="" THEN PRINT "Error;Lack o
f the data. Repeat!":GOTO 2650
2660   KS=1:KE=KS:JX$=JX$+",":KKK=LEN(JX$)
2670   FOR I=1 TO IMAX
2680   KE=KE+1:IF KE>KKK+1 THEN PRINT" Error;Lack of the data. Repeat!":GOTO 265
0
2690   IF MID$(JX$,KE,1)="," THEN XMCI(J,I)=VAL(MID$(JX$,KS,KE-KS)):KS=KE+1:KE=K
S ELSE 2680
2700   NEXT I,J
2710   RETURN

```

Fig. 2. List of a Program of Data Input Part for the Other Computer System

starting column. A duplicate selection of the path in path type (for example, 1-2-3 and 3-2-1) is avoided by ignoring the path where the serial number of the end atom is smaller than the first one. The calculated value for the chain type is divided by the number of atoms in the path to correct the duplicate calculation.

The cluster type path is effective when the end atom has more than three connected atoms and at least two of these are not common to the registered atoms through the path. In the process of finding the path of the path type, if the serial number of the end atom is that of the starting atom, the path is a ring. If the end atom has the characteristics of the cluster type, path & cluster type is calculated.

Use of the Program

The program list is shown in Fig. 1. Change LPW=80 in line 50 to LPW=132 if the printer has 132 column width.

Input the chemical name and the number of the direction of the bonds to be checked. The number of direction is usually 4 (up,

down, right and left). Use 8 when the user uses an oblique line for the bond (O-3-C and C-4-C in Table I). The maximum number of rows and columns in the data matrix should be odd numbers. Then the delta-V values and the bond number are inputted. It is assumed in this program that the delta-V values exist only at the crossing of a row and column with odd numbers. If a dummy atom is necessary to draw a chemical structure, put 9 instead of delta-V value. A series of 9 (9-9-9) is acceptable but a dummy atom has two bonds and no other branch is allowed. If the user draws a bond with an oblique line, the bond number should be 3 for the right-up line or 4 for the left-up line. Any bond number will do for the other directions. Place zero where there is no delta-V value and no bond. The program for the data input is specialized for the convenience of the user of an IBM-PS/55 system. Just draw the chemical structure with the delta-V value and the bond number separated by a comma, and push the return key. If the data input part of the program does not work for the user's computer, replace the line from

2610 to 2790 with the list in Fig. 2, although the program in Fig. 2 is rather inconvenient because it is written for general use.

The steps of the calculation will appear on the display. The maximum order of MCI is limited to 50 in this program. The user can expand the maximum order by changing the

dimension of the variables though the order of most of the chemicals, except polymers, is under 50. Delta value is automatically calculated from the structure inputted. All of the MCI values are printed out. If the user has a BASIC compiler, the program can be compiled as it is. It will save calculation time.

References

- 1) C. Hansch, R.M. Muir, T. Fujita, P.P. Maloney, F. Geiger and M. Streich, *J. Am. Chem. Soc.* **85**, 2817 (1963).
- 2) L.H. Hall and L.B. Kier, *Bull. Environ. Toxicol.* **32**, 354 (1984) ; L.B. Kier and L.H. Hall, *J. Pharm. Sci.* **64**, 1971 (1975).
- 3) H. Konemann, *Toxicology* **19**, 209 (1981) ; A. Sabljic and M. Protić, *Chem. Biol. Interactions*, **42**, 301 (1982) ; H. Govers, C. Ruepert and H. Aiking, *Chemosphere*, **13**, 227 (1984) ; Y. Yoshioka, T. Mizuno and T. Sato, *ibid.*, **15**, 195 (1986) ; M. Vighi and D. Calamari, *ibid.*, **16**, 1043 (1987).