b. Parentheses are replaced by nondirectional single quotes.

c. Subscripts are changed to on-line numbers.

In the context of this notation true carbides, carbonates, and carbon oxides are considered to be inorganic compounds.

Rule 47. Inorganic Hydrogen-Replacing Groups. When complex inorganic groups appear as hydrogen-replacing groups in organic compounds, it is necessary to show the internal structure of the groups. The last symbol of the group is the bond attaching it to the carbon skeleton. These are the usual symbols Y, =, and no entry for a single bond. The next to the last symbol indicates the atom involved in this bond, and the remaining symbols are then listed in order of attachment from right to left. The entire group is set off by nondirectional single quotes, and any side chains within the group are set off in a similar fashion. A common feature in such groups is an oxygen atom attached to the group by a double bond or by a coordinate covalent bond. Such an atom is indicated by the symbol U (for unsaturated) appearing in front of the atom to which it is attached. Two such groups attached to the same atom are indicated by the symbol W (for double U). If instead of oxygen, the atom is sulfur or some other element, the U is preceded by the symbol of the element and the group is set off by quotes to indicate a side chain.

Some common inorganic hydrogen-replacing groups are "UN" (nitroso), "WN" (nitro), "UNO" (nitrito), "HOWSO" (sulfuric acid), and "HOWS" (sulfonic acid).

Rule 48. Inorganic Hetero Groups. Some nonmetal heteroatoms, particularly sulfur and phosphorus, can form additional coordinate covalent bonds. In such cases the entire group enclosed in quotes is used as a hetero symbol. The sulfoxide group is therefore "SU", and the sulfone group is "SW". Dimethyl sulfoxide has the notation T′SU′ (propane sulfoxide).

Rule 49. Organic Salts. If one of the ions of an organic salt is inorganic, it is listed as in an inorganic salt. In general positive organic ions are formed by gaining hydrogen ions and negative ions are formed by losing them. A positive ion is indicated by placing an H after the symbol of the group gaining the hydrogen, and a negative ion is indicated by placing an H before the symbol of the group losing the hydrogen. The name of the positive ion is obtained by adding the suffix onium to the name of the group involved. "Amine" becomes "ammonium" and "nitride" becomes "nitronium". The name of the negative ion is obtained by deleting "hydroxo" from the name of the group involved. "Hydroxoxo" becomes "oxoxo".

The notations of the two ions are separated by a nondirectional quote. If subscripts are needed, they appear as arabic numerals set off from the notations of the ions by as many single quotes as are needed.

Thus lead acetate is P-B′HO,O=E′2 (lead oxoxoethane), sodium ethoxide is N-A′HOE (sodium oxoethane), and tetramethylammonium chloride is 2-MTNH′L (dimethylpropanenitronium chloride).

CONNECTIVITY MATRICES

A notation or its name furnishes explicity all the information needed to construct a connectivity matrix for the compound. Starting with the central feature of the notation and moving to the left, each segment of the notation is entered along the matrix diagonal from left to right. For each segment the index number of its first atom and its locant are the coordinates of the matrix element which is to be used to indicate its attachment to the portion already entered.

At the end of the notation the unsaturation symbols and locants show how and where the matrix is to be modified to show unsaturation, and the heteroatom data indicate which carbon atoms along the diagonal are to be replaced with heteroatoms.

Using the convention that the digit 1 indicates a single bond; 2, a double bond; and 0, no bond, the matrix for 1N2MV=5O3 (1-amino-2-methylpentene-5 oxide-3) is as shown in (I).

$$\begin{array}{c|ccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
\hline
1 & \mathrm{C} & 1 & 0 & 0 & 0 & 0 & 1 \\
2 & 1 & \mathrm{C} & 1 & 0 & 0 & 1 & 0 \\
3 & 0 & 1 & \mathrm{O} & 1 & 0 & 0 & 0 \\
4 & 0 & 0 & 1 & \mathrm{C} & 2 & 0 & 0 \\
5 & 0 & 0 & 0 & 2 & \mathrm{C} & 0 & 0 \\
6 & 0 & 1 & 0 & 0 & 0 & \mathrm{C} & 0 \\
7 & 1 & 0 & 0 & 0 & 0 & 0 & \mathrm{N}
\end{array} \qquad (\mathrm{I})$$

A Fortran IV Program for Finding the Smallest Set of Smallest Rings of a Graph

BURGHARDT SCHMIDT and JÖRG FLEISCHHAUER*

Department of Theoretical Chemistry, RWTH Aachen, Templergraben, D-5100 Aachen, West Germany

Received December 20, 1977

A Fortran IV program to find the set of smallest rings is described. The algorithm consists of two parts. First the nodes, which are on rings, are found; then in the second part the smallest set of smallest rings (SSSR) is found considering these nodes only.

INTRODUCTION

To use the computer for solving chemical problems, particularly for the design of organic synthesis, analysis, and the storage of chemical information, it is necessary to translate the chemist's formulas of molecules into a form which can be manipulated by the computer. To handle molecules in a computer, graphs and their adjacency matrices[1] are very well suited. (Other well-known representations of chemical formulas in a computer program are connection-tables or the WLN notation.[2]) The nodes of the graphs represent the atoms and the edges the bonds. By definition the entries $a_{ij} = a_{ji}$ of the adjacency matrix **A** are only unequal to zero if the nodes I and J are connected by an edge.[1] Among other things a computer program must now recognize special structural features of the graphs. Very important is the finding of cyclic structures, because they differ strongly from acyclic ones with regard to their chemical and physical behavior. Especially for chemical synthesis work the so-called smallest set of smallest rings (SSSR) of a chemical structure must be recognized. This was pointed out by Corey and Wipke[3] in their fundamental work about computer-assisted design of organic synthesis. The SSSR is the set of those rings which are not the envelopes of smaller rings. This is equivalent to Corey's and Petersson's[4] definition of the synthetically important ring.

In this paper a program is described for finding the indices of the nodes which form the set of the smallest rings of the

0095-2338/78/1618-0204$01.00/0

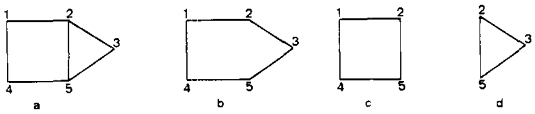


Figure 1. Graph a can be decomposed in the cyclic subgraphs b, c, and d.

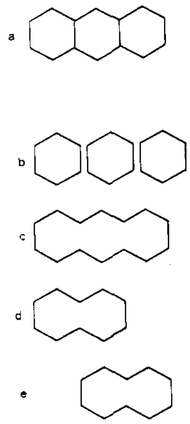


Figure 2. The 3 rings b are real; the rings c, d, and e are pseudo rings of the graph a.

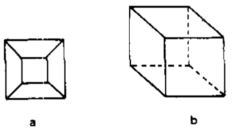


Figure 3. Graph a is the planar form of the spherical graph b.

graph. In Figure 1 this smallest set consists of the two subgraphs 1-2-5-4 and 2-3-5.

Corey and Wipke[3] distinguish between real and pseudo rings. Real rings have to be as small as possible. In Figure 2 only the three six-rings are real rings. Pseudo rings are the two 10- and the one 14-membered ring. It is clear that the real rings are identical with the smallest rings.

The number ICYC of the smallest rings for M planar graphs is given by[5]

$$\mathrm{ICYC} = \mathrm{M} + \mathrm{NBIND} - \mathrm{NELEM}$$

NBIND and NELEM are the number of edges and nodes, respectively. This formula is valid only for the planar forms of spherical graphs. For example, a cube contains in its spherical form six smallest rings according to Euler's polyhedron formula:[5]

$$\mathrm{ICYC} = 2 + \mathrm{NBIND} - \mathrm{NELEM}$$

In its planar form, however, ICYC is 5 (Figure 3). There exist several algorithms to find the SSSR other than the one which is described in this paper. Paton,[6] Gotlieb and Corneil,[7] Welch,[8] and Gibbs[9] proposed methods to find all rings of a graph. Plotkin[10] starts from the set of all rings and finds the SSSR by comparing rings of equal length. Wipke and Dyott[11] represent an algorithm called the Welch–Assembly–Gibbs algorithm. It finds the basis set of rings by Welch's algorithm, then ring assemblies, and finally all rings by Gibb's algorithm. Zamora[12] finally distinguishes between different types of rings and uses for each type special algorithms to find the SSSR.

I. SEARCH OF NODES ON RINGS

In this section the method for finding for a given adjacency matrix **A** the indices of the nodes which are on rings is described. To solve this problems we will use some algorithms which work only with oriented graphs or the corresponding adjacency matrices.

For an oriented graph a path exists between two nodes only in one direction. If the final node of such a path is identical with the starting node, then one will have an oriented cycle.

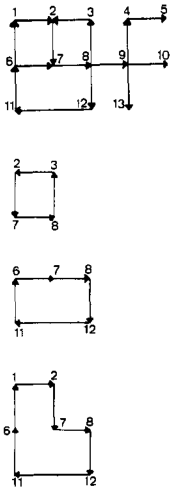
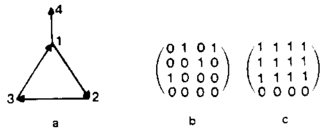


Figure 4. Nodes which are on rings are situated on at least one oriented cyclic subgraph.



$$\begin{pmatrix} 0&1&0&1 \\ 0&0&1&0 \\ 1&0&0&0 \\ 0&0&0&0 \end{pmatrix} \quad \begin{pmatrix} 1&1&1&1 \\ 1&1&1&1 \\ 1&1&1&1 \\ 0&0&0&0 \end{pmatrix}$$

Figure 5. The two matrices are the oriented adjacency matrix b and the path matrix c of the oriented graph a.

To find all the oriented cycles the oriented adjacency matrix is constructed by the program as follows: Any node I is chosen as starting point. If I is adjacent to the node J ($a_{ij} \neq 0$ and $a_{ji} \neq 0$), the entry a_{ji} will be set equal to zero. Then a_{lj} ($l \neq i$) will be set equal to zero if L is connected to node J, and so on. If there exists for any node L on the considered path more than one entry $\neq 0$, the index L will be stored. This is continued until a node M is reached, which has no further adjacent node. Then the program returns to the last stored node with several branches and this node is regarded as a new starting point. This procedure is continued until all the stored nodes are treated. Then we have constructed an oriented adjacency matrix which represents an oriented graph. On this graph each ring-node is placed at least on one cyclic oriented subgraph (Figure 4). We then can use Warshall's algorithm to find the nodes which are on rings.

In the next step the program constructs, starting from the oriented adjacency matrix, the corresponding path matrix **KW** using an algorithm developed by Warshall.[13] Let W1 be a path from I to J ($kw_{ij} = 1$) and W2 be a path from J to K ($kw_{jk} = 1$); then there is also a path from I to K ($kw_{ik} = 1$). At the beginning the matrix **KW** is set equal to the oriented adjacency matrix. The statements by which the final **KW** matrix will be constructed are:

```
C NN is the number of the nodes of the graph
      DO 20 J = 1, NN
      DO 20 I = 1, NN
      IF (KW(I,J)) 10, 20, 10
   10 DO 30 K = 1, NN
   30 KW (I,K) = (KW(I,K) +KW(J,K) + 1)/2
   20 CONTINUE
```

In Figure 5, an example of an oriented graph, the oriented adjacency matrix **A** and the path matrix **KW** are given.

The indices I of the nodes which are on rings are then easily found by investigating the diagonal elements kw_{ii}. If kw_{ii} is not equal to zero, then the node I will be situated on a cycle.

II. FINDING THE SMALLEST SET OF SMALLEST RINGS

To find the smallest rings, clearly only those nodes have to be examined which are situated on rings. In the first step, the

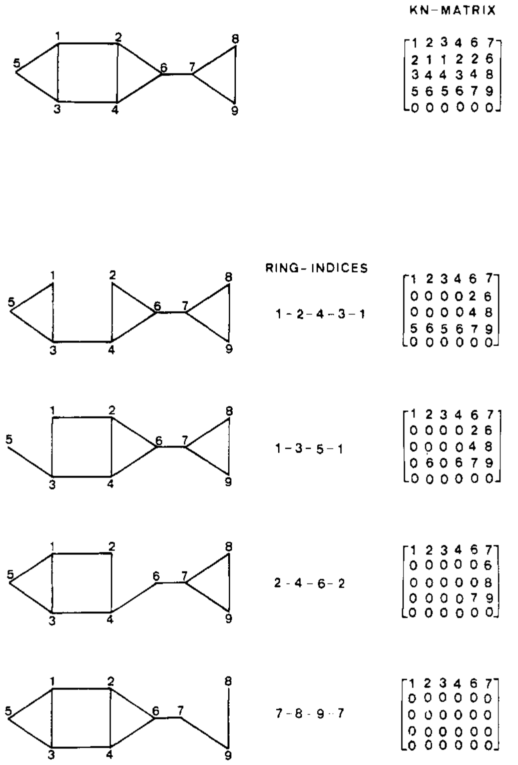


Figure 6. The scheme demonstrated how the **KN** matrix is used to find the indices of the smallest rings (the edge 6–7 is not cut because there will be no path from 6 to 7).

indices of the N nodes which are connected to at least 3 nodes are found and stored in a 5 × N matrix **KN** together with the indices of the neighboring nodes. The entry kn_{1i} is the index of the branching node, and kn_{Ji} ($J = 2, \ldots, M + 1$) are the indices of the M nodes which are adjacent to I ($M = 3$ or 4). The program proceeds now as follows: Starting with the first of the N nodes I, the edge to that node J_1, whose index is given by kn_{2i} is "cut". Then for each node the shortest path of length l (l = number of edges on the path) to the starting node I is computed. This is done by building a tree starting from node I. If the distance of the node J_1 to the starting node I is known, the procedure stops. If there exists a shortest path of length $l > 1$ between J_1 and I, then these two nodes will be on a smallest ring of length $l + 1$. If a node J_2 is adjacent to J_1 and there exists a path of length $l - 1$ to I, then J_2 will also be situated on this smallest ring of length $l + 1$. This is continued until all the nodes of the considered ($l + 1$) cycle are found. If there are two shortest paths, the choice between them is arbitrary. During this procedure all the entries kn_{lk} are set equal to zero if the nodes $J_k = kn_{1k}$ and $J_l = kn_{lk}$ are adjacent to each other in the ring. This is continued until all the entries of the **KN** matrix except those in the first row are zero. The different steps are demonstrated in the example in Figure 6. Finally those cycles in which the nodes have the connectivity 2 are easily found by examining the remaining nodes.

The described algorithm fails in the following case: If a cyclic subgraph G of length l is surrounded by cyclic subgraphs G' of length $l' < 1$ (Figure 7), then G will not be found, because for every broken edge of G there always exists a shorter path via G'. It is valid for planar graphs. For example, for cubane, five rings are found. In contrast to Bersohn's[14] algorithm the represented algorithm has the advantage that only small parts of the graph are investigated by path-tracing.

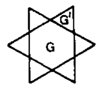


Figure 7. The described algorithm fails to find the cyclic subgraph G.

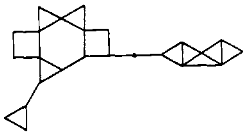


Figure 8.

The program consists of 250 statements. For graphs with about 40 nodes it uses less than 36 000 (octal) words, each containing 60 bits, on a Cyber 175 under NOS operating system. It is used as a subroutine in a program for the planing of organic synthesis.[15] For finding the SSSR in Figure 8 it needs less than 0.05 s.

ACKNOWLEDGMENT

We thank the Deutsche Forschungsgemeinschaft for financial support.

Supplementary Material Available: Program listing of SSSR-finding algorithm (2 pages). Supplementary material, which will appear in the microfilm edition of the journal, can be ordered from Business Operations, Books and Journals Division, American Chemical Society, 1155 16th St., N.W., Washington, D.C., or call (202) 872-4600. Orders must state photocopy or microfiche. Full bibliographic citation including names of all authors and prepayment are required: $3.00 for microfiche or $5.50 for photocopy (prices subject to change).

REFERENCES AND NOTES

(1) W. Dörfler and J. Mühlbacher, "Graphentheorie für Informatiker", Sammlung Göschen, de Gruyter, Berlin, 1973.
(2) E. G. Smith, "The Wiswesser Line Formula Chemical Notation", McGraw-Hill, New York, N.Y., 1968.
(3) E. J. Corey and W. T. Wipke, "Computer Assisted Design of Complex Organic Synthesis", *Science*, **166**, 178–92 (1969).
(4) E. J. Corey and G. A. Petersson, "An Algorithm for Machine Perception of Synthetically Significant Rings in Complex Cyclic Organic Structures", *J. Am. Chem. Soc.*, **94**, 460–465 (1972).
(5) R. Harary, "Graphentheorie", R. Oldenbourg-Verlag, München, Wien, 1974.
(6) K. Paton, "An Algorithm for Finding a Fundamental Set of Cycles for an Undirected Linear Graph", *Commun. Assoc. Comput. Mach.*, **12**, 594–8 (1969).
(7) C. C. Gotlieb and D. G. Corneil, "Algorithms for Finding a Fundamental Set of Cycles for an Undirected Linear Graph", *Commun. Assoc. Comput. Mach.*, **10**, 780–3 (1967).
(8) J. T. Welch, Jr., "A Mechanical Analysis of the Cyclic Structure of Undirected Linear Graphs", *J. Assoc. Comput. Mach.*, **13**, 205–10 (1966).
(9) N. E. Gibbs, "A Cycle Generation Algorithm for Finding a Fundamental Set of Cycles for an Undirected Linear Graph", *J. Assoc. Comput. Mach.*, **16**, 564–8 (1969).
(10) M. Plotkin, "Mathematical Basis of Ring-Finding Algorithms in CIDS", *J. Chem. Doc.*, **11**, 60–63 (1971).
(11) W. T. Wipke and T. M. Dyott, "Use of Ring Assemblies in a Ring Perception Algorithm", *J. Chem. Inf. Comput. Sci.*, **15**, 140–147 (1975).
(12) A. Zamora, "An Algorithm for Finding the Smallest Set of Smallest Rings", *J. Chem. Inf. Comput. Sci.*, **16**, 40–3 (1976).
(13) S. Warshall, "A Theorem on Boolean Matrices", *J. Assoc. Comput. Mach.*, **9**, 11 (1962).
(14) M. Bersohn, "An Algorithm for Finding the Synthetically Important Rings of a Molecule", *J. Chem. Soc., Perkin Trans. 1*, 1239–41 (1973).
(15) B. Schmidt, Diplomarbeit, RWTH Aachen, 1977.