# Applications of Artificial Intelligence for Chemical Inference. XII. Exhaustive Generation of Cyclic and Acyclic Isomers[1,2]

**L. M. Masinter, N. S. Sridharan, J. Lederberg, and D. H. Smith***

*Contribution from the Departments of Computer Science, Genetics and Chemistry, Stanford University, Stanford, California 94305.* *Received March 22, 1974*

**Abstract:** A systematic method of identification of all possible structural isomers consistent with a given empirical formula is described. The method, embodied in a computer program, generates a complete list of isomers. Duplicate structures are avoided prospectively.

Problems of structural isomerism in chemistry have received much attention. But only occasional inroads have been made toward a systematic solution of the underlying graph theoretical problems of structural isomerism. Solutions in the past have been partial, with acyclic and cyclic structures being treated independently. Recently the "boundaries, scope, and limits" [3] of the subject of structural isomerism of acyclic molecules have been defined by the DENDRAL algorithm.[3] This algorithm permits an enumeration and representation of all possible acyclic molecular structures with a given empirical formula.

Acyclic molecules represent only a subset of molecular structures, however, and it may be argued that cyclic structures (including those possessing acyclic chains) are of more general interest and importance to modern chemistry from both a practical and theoretical standpoint. An approach to cyclic structure generation has appeared in a previous paper in this series.[4] That approach, which operates on a set of previously generated acyclic forms by labeling hydrogen atoms pairwise and connecting the atoms to which they are attached with a new bond, has one serious drawback. The approach cannot make efficient use of the symmetry properties of cyclic graphs; hence an inordinate amount of computer time must be spent in retrospective checking of each candidate structure with existing structures to remove duplicates. For this reason, an alternative approach to construction of cyclic molecules has been developed. This approach is designed to take advantage of the underlying graph theoretic considerations, primarily symmetry, to arrive at a method for more efficient construction of a complete and irredundant list of isomers for a given empirical formula. Central to the successful solution of this problem is the generation of all positional isomers obtained by substitutions on a given ring system. This topic has received attention for nearly 100 years, with limited success.[5] Its more general ramifications go far beyond organic chemistry. Graph theoreticians have considered various aspects of this topic, frequently, but not necessarily, in the context of organic molecules. Polya has presented a theorem[6] which permits calculation of the number of structural isomers for a given ring system. Hill[7a,b] has applied this theorem to enumeration of isomers of simple ring compounds, and Hill[7c] and Taylor[8] have pointed out that Polya's theorem permits enumeration of geometrical and optical isomers in addition to structural isomers. More recently, formulas for enumeration of isomers of monocyclic aromatic compounds based on graph theory, permutation groups, and Polya's theorem have been presented.[9a] This history of interest and results provides only marginal benefit to the organic chemist. Although the number of isomers may be interesting, these methods[5-9a] do not display the structure of each isomer. Also, these methods do not provide information on the more general case where the ring system is embedded in a more complex structure. Even for simple cases the task of specifying each structure by hand, without duplication, is an onerous one.

Balaban has published a series of papers[9] addressed, in part, to the problem of specification of isomeric structures. Although his method, which differs substantially from our own, involves significant manual effort and does not appear to encompass a mechanism for prospective avoidance of duplicate structures, his compilations of isomers of annulenes[9b,c] represent an important contribution as extensions to the compilations of Lederberg.[10]

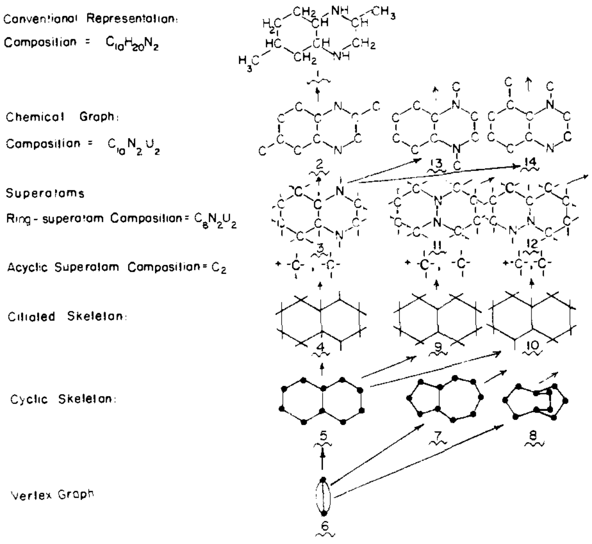
## Method

**Overview. Framework.** The framework for this method is that chemical structures consist of some combination of acyclic chains and rings or ring systems.[10,11] The problem of construction of acyclic isomers (and radicals) has been solved previously.[3] If all possible ring systems can be constructed from all or part of the atoms in the empirical formula, and all possible acyclic parts are available from the acyclic generator, the combination of ring systems with acyclic parts in all unique ways would yield the complete list of isomers. The method for construction of ring systems is described below. This description employs some terms which require definition. The definitions also serve to illustrate the taxonomic principles which underlie the operation of the structure generator. The generator's view of molecular structure differs in some respects from the chemist's. A chemist, for example, may view structures possessing the same functional group or ring as related. The generator works at the more fundamental level of the vertex-graph,[10] as described below.

**Chemical Graph.** A molecular structure may be viewed as a graph, termed the *chemical graph*, or skeleton. A chemical graph consists of *nodes*, with associated atom names, and *edges*, which correspond to chemical bonds. Consider as an example the substituted piperazine, **1**, whose chemical graph is illustrated in Chart I as **2**. Note that hydrogen atoms are ignored by convention, while the symbol "U" is used to specify the unsaturation. The degree (primary, secondary, . . .) of a node in the chemical graph has its usual meaning, *i.e.*, the number of (nonhydrogen) edges connected to it. The valence of each atom determines its maximum degree in the graph. As usually displayed by chemists in planar representation, the chemical graph describes the connectivity rather than the geometric configuration of a molecular structure.

**Superatom.** In general, a chemical graph can be separated into cyclic and acyclic parts. Each cyclic structural subunit may be deemed a *superatom* possessing any number of *free valences.*[12] The chemical graph **2** arises from a combination of two carbon atoms with ring-superatom **3.** Ring-superatom **3** possesses the indicated free valences to which the remaining hydrogen and two methyl radicals will be attached (Chart I).

**Chart I**


Conventional Representation:
Composition = $C_{10}H_{20}N_2$
Chemical Graph:
Composition = $C_{10}N_2U_2$
2
13
14
Superatoms
Ring-superatom Composition= $C_8N_2U_2$
3
11
12
Acyclic Superatom Composition= $C_2$
Ciliated Skeleton:
4
9
10
Cyclic Skeleton:
5
7
8
Vertex Graph
6

**Ciliated Skeleton.** A *ciliated skeleton* is a skeleton with free valences but without atom names. Ring-superatom **3** arises from the ciliated skeleton **4** by associating the atom names of eight carbon and two nitrogen atoms with the skeleton (Chart I).

**Cyclic Skeleton.** A chemical graph whose nodes are not associated with atom names and which contains no acyclic parts and no free valences is termed a *cyclic skeleton.* Ciliated skeleton **4** arises from one way of associating 16 free valences with the nodes on the cyclic skeleton **5** (Chart I).

**Vertex-Graph.** Vertex-graphs[10] are cyclic skeletons from which nodes of degree less than three have been deleted. The vertex-graph of the cyclic skeleton **5** is the regular trivalent graph[10] of two nodes, **6.** Note that the remaining nodes of the cyclic skeleton **5** are of degree two. Removal of these secondary nodes from **5** while retaining the interconnections of the two tertiary nodes yields **6** (Chart I).

As an illustration of the variety of structures which may be constructed from a given vertex-graph and empirical formula, for example, $C_{10}H_{20}N_2$, consider that graph **6** is the vertex-graph for all bicyclic ring systems (excluding spiro forms). Cyclic skeletons **7** and **8** (Chart I), for example, may be constructed from eight secondary nodes and **6.** There are many ways of associating 16 free valences with each cyclic skeleton, resulting in a larger number of ciliated skeletons. For example, **9** and **10** arise from different allocations of 16 free valences to **5** (Chart I). There is only one way to associate eight carbon atoms and two nitrogen atoms with each ciliated skeleton to yield superatoms (*e.g.*, **11** and **12,** Chart I). However, several structures are obtained by associating the remaining two carbon atoms (in this example) with each superatom, as an ethyl or two methyl groups. Chemical graphs **13** and **14,** for example, arise from two alternative ways of associating two methyl groups with superatom **3.**

**Multiple Bonds.** For the purposes of this program, we adopt the formalism that all multiple bonds (double, triple, . . .) are considered to be small rings by the program. Previous versions[3] (acyclic generator) differ from this program in that double and triple bonds are regarded as specially labeled edges.

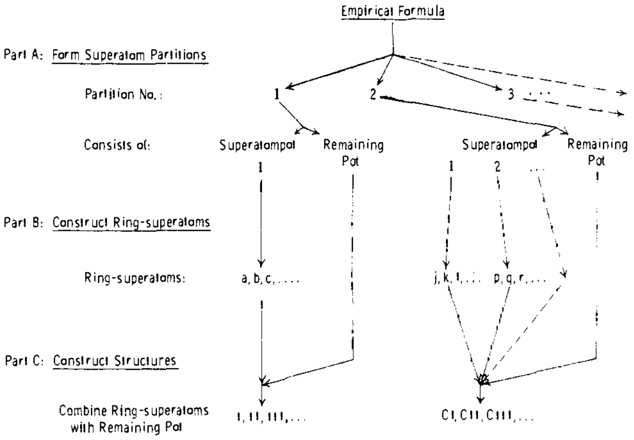


**Figure 1.** Outline of the strategy for structure generation.

## Aims

The structure generator must produce a complete list of structures without duplication. By duplicate structures we mean structures which are equivalent in some well-defined sense. The class of isomers generated by the program includes only connectivity isomers. Transformations (utilized to determine equivalence) allowed under connectivity symmetry preserve the valence and bond distribution of every atom. Connectivity symmetry does not consider bond lengths or bond angles. This choice of symmetry results in construction of a set of topologically unique, structural isomers. A more detailed discussion of equivalence is discussed in Appendix A and in the accompanying paper;[13] a discussion of isomerism and symmetry is presented in Appendix B.

## Strategy

The strategy behind the structure generator is strongly tied to the framework described above. The strategy is summarized in greatly simplified form in Figure 1. The vertex-graphs from which structures are constructed can be specified for a given problem by a series of calculations. Thus part A of the program (Figure 1) partitions the pot of atoms in all possible ways; each partition consists of those atoms assigned to one or more "superatompots" and a "remaining pot." Each superatompot is a collection of atoms from which all possible, unique ring-superatoms[12] can be constructed based on the appropriate vertex-graphs (part B, Figure 1). Each ring-superatom will be a ring system in completed structures. The atoms in the remaining pot will form acyclic parts of the final structures when combined in all possible, unique ways with the ring-superatoms from the corresponding initial partition (part C, Figure 1).

## Description

We are faced with the difficulty of describing a complex computer program in the traditional mode of presentation in a scientific journal. The narrative form is not the ideal medium for this description; simple examples do not always indicate all essential aspects of a program. A deeper understanding of a program could be engendered through the use of a large number of well-chosen examples, but the length of such a presentation would be excessive and would tax the patience of even the most interested reader.

We are thus aware of the insufficiency of considering only one example in the following written description. We have adopted the strategy of presenting essential aspects of the procedure for structure generation in the main body of the text. Details of the description which might obscure the principal concepts are placed in Appendexes C and D.

Mathematical details are available elsewhere.[14,15] We hope this serves the purpose of providing the casual reader with a deeper understanding of the method without having to contend with details which, on the other hand, are important to others who wish to make use of our approach.

The example chosen to illustrate each step of the method is $C_6H_8$ (or $C_6U_3$ as there are three degrees of unsaturation). This example does not contain bivalent or trivalent atoms (*e.g.*, oxygen and nitrogen, respectively) or atoms of valence greater than four, nor any univalent atoms other than hydrogen (*e.g.*, chlorine and fluorine).

**Partitioning and Labeling.** The mechanism for structure generation involves a series of "partitioning" steps followed by a series of "labeling" steps. Partitions are made of items which must be assigned to objects (usually graph structures or parts thereof) as the molecular structures are built up from the vertex-graphs. The process by which items are assigned to the graphs is termed labeling.[13,14]

Examination of Chart I reveals the different types of items involved. For example, nodes are partitioned among and labeled upon the edges of the vertex-graphs to yield the cyclic skeletons. Free valences are partitioned among and labeled upon the nodes of cyclic skeletons to yield ciliated skeletons, and so forth.

Partitioning steps in the subsequent discussion are carried out assuming that objects among which items are partitioned are indistinguishable. Distinguishability of objects (edges, nodes, . . .) is specified during labeling and will be discussed in a subsequent section. The partitioning steps performed by the program are outlined in Table I. Each step is described in more detail below.

Table I. Partitioning Steps Performed by the Structure Generator

| Step Number | Partition | Among |
|---|---|---|
| 1 | Atoms and unsaturations in empirical formula | Superatompots and remaining pot |
| 2 | Free valence | Atoms in superatompot |
| 3 | Secondary nodes | Loops/nonloops |
| 4 | Nonloop secondary nodes | Edges of graph |
| 5 | Loop secondary nodes | Loops |
| 6 | Ring superatoms and remaining pot | Efferent links (see Appendix D) |

## Part A. Superatom Partitions

Ring-superatoms are "two-connected" structures, *i.e.*, the ring-superatom cannot be split into two parts by scission of a single bond. The atoms in an empirical formula may be distributed among from one to several such two-connected ring-superatoms. A distribution which allots atoms to two or more superatompots will yield (respectively) structures containing two or more ring-superatoms linked together by single bonds (or acyclic chains).[16] In the generation process, one must find all possible ways of partitioning the given formula into superatompots and a remaining pot, such that molecules can be constructed. The considerations in forming superatom partitions deal primarily with valence and unsaturation. This procedure is summarized in Appendix C, Superatom Partitions. The partitions which result are summarized in Table II.

## Part B. Ring-Superatom Construction

Each partition (Table II) must now be treated in turn. The complete set of ring-superatoms for each superatompot in a given partition must be constructed. The major steps in the procedure are outlined in Figure 2.

**Table II.** Allowed Partitions of $C_6U_3$ into Superatompots and Remaining Pot

| Partition no. | Number of superatompots | Superatompot number 1 | 2 | 3 | Remaining pot |
|---|---|---|---|---|---|
| 1 | 1 | $C_6U_3$ | | | |
| 2 | 1 | $C_5U_3$ | | | $C_1$ |
| 3 | 1 | $C_4U_3$ | | | $C_2$ |
| 4 | 1 | $C_3U_3$ | | | $C_3$ |
| 5 | 2 | $C_4U_2$ | $C_2U_1$ | | |
| 6 | 2 | $C_3U_2$ | $C_2U_1$ | | $C_1$ |
| 7 | 2 | $C_2U_2$ | $C_2U_1$ | | $C_2$ |
| 8 | 2 | $C_4U_1$ | $C_2U_2$ | | |
| 9 | 2 | $C_3U_1$ | $C_2U_2$ | | $C_1$ |
| 10 | 2 | $C_3U_2$ | $C_3U_1$ | | |
| 11 | 3 | $C_2U_1$ | $C_2U_1$ | $C_2U_1$ | |

**Valence List.** The first step in part B is to strip the superatompot of atom names, while retaining the valence of each atom. The numbers of each type of atom are saved for later labeling of the ciliated skeletons (Chart I). A valence list may then be specified, giving in order the number of bi-, tri-, tetra-, and *n*-valent nodes which will be incorporated into the superatom. Thus the superatompot $C_6U_3$ is transformed into the valence list 0 bivalents, 0 trivalents, 6 tetravalents (0,0,6), and $C_4U_2$ becomes (0,0,4) (Figure 2).

**Calculation of Free Valence.** From the valence list and the associated unsaturation count, the number of free valences of each superatompot is determined uniquely (see Calculation of Free Valence, Appendix C). For $C_6U_3$, the free valence is eight (Figure 2). The free valence of a superatom represents the number of bonding sites which can connect to hydrogen atoms, other superatoms or atoms in the remaining pot.

**Partitioning of Free Valence.** The free valences are then partitioned among the nodes in the valence list in all possible, unique ways (see Appendix C, Partitioning of Free Valence).

**Degree List.** Each partition of free valences alters the affective valence of the nodes in the original valence list with respect to the ring-superatom. In the example, assignment of one or two free valences to a tetravalent node transforms this node into a tri- or bivalent node, respectively. As the ring-superatom is constructed, those tetravalent nodes which have been assigned, say, two free valences, have then only two valences remaining for attachment to the ring-superatom. These nodes are then of degree[17] two and may be termed secondary nodes. Thus the partition of free valences 2,2,2,2,0,0 on six tetravalent nodes yields the degree list (4,0,2) (Figure 2) as four of the tetravalent nodes receive two free valences each, yielding four nodes of degree two (secondary) and leaving two nodes of degree four (quaternary). The program keeps track of the number of free valences assigned to all nodes for use in a subsequent step.

**Loops.** As will be clarified in the subsequent discussion, there are several general types of ring-superatoms which cannot be constructed from the vertex-graphs available in the Catalog (described below). These are all cases of multiple extended unsaturations either in the form of double bonds or rings. Examples are the following: (1) bi-, tri-, . . . , *n*-cyclics with exocyclic double bonds; (2) some types of spiro ring systems; and (3) allenes extended by additional double bonds, *e.g.*, C=C=C=C. The concept of a loop, each loop consisting of a single unsaturation and at least one bivalent node, must be utilized for these cases. Examples of loops containing one, two, and three bivalent nodes are shown in Chart II. Note that the two remaining "ends" of the unsaturation will yield a "looped structure" when attached to a single node in a graph (shown as X, Chart II).

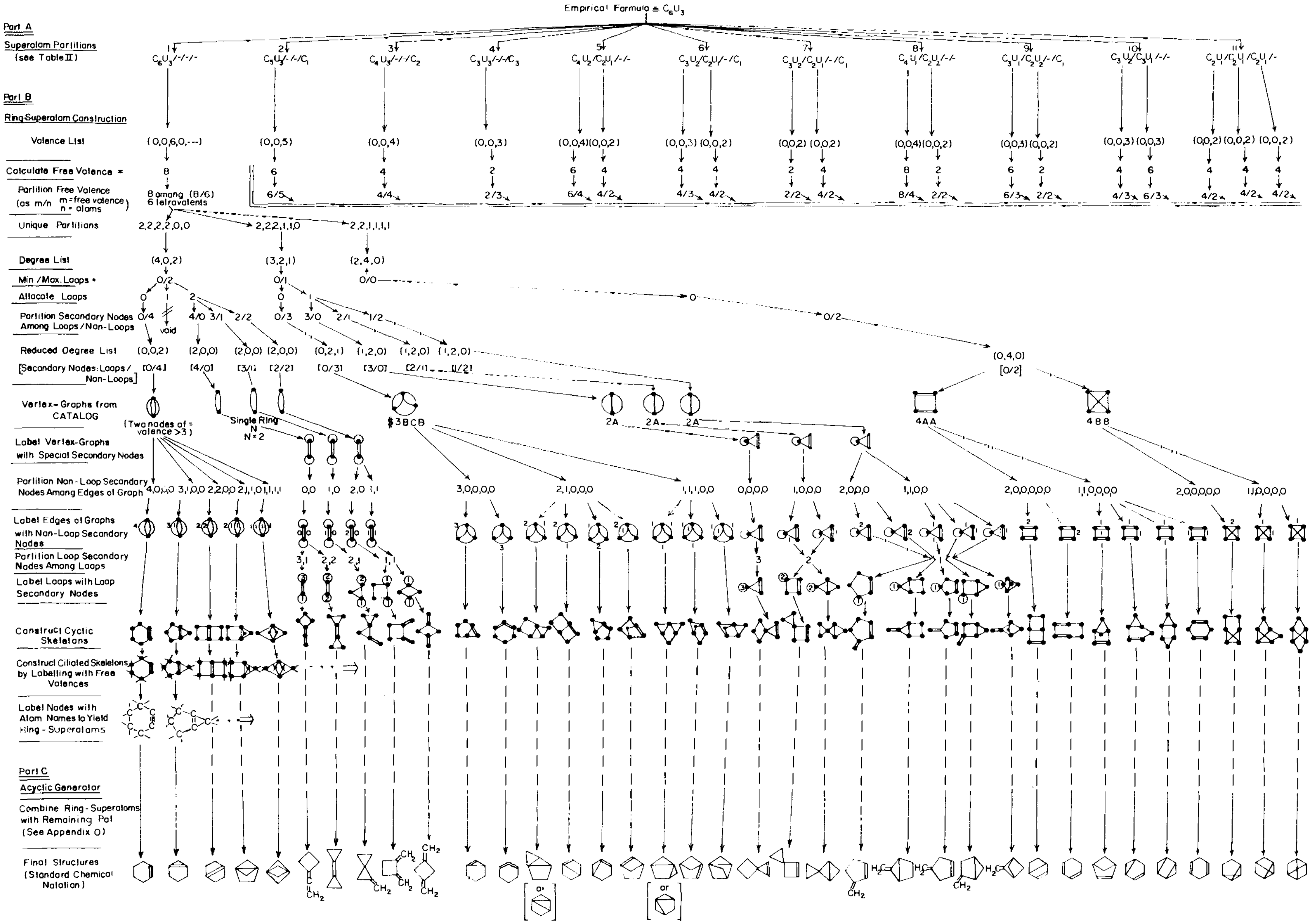
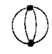
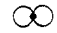
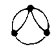

Empirical Formula = $C_6U_3$
Part A
Superatom Partitions (see Table II)
Part B
Ring-Superatom Construction
Valence List
Calculate Free Valence *
Partition Free Valence (as m/n m = free valence n = atoms)
8 among (8/6) 6 tetravalents
Unique Partitions
2,2,2,2,0,0
2,2,2,1,1,0
2,2,1,1,1,1
Degree List
(4,0,2)
(3,2,1)
(2,4,0)
Min /Max Loops •
Allocate Loops
Partition Secondary Nodes Among Loops/Non-Loops
void
Reduced Degree List
[Secondary Nodes: Loops / Non-Loops]
(0,4,0)
[0/2]
Vertex-Graphs from CATALOG
(Two nodes of valence > 3)
Single Ring N N=2
$38CB
2A
4AA
4BB
Label Vertex-Graphs with Special Secondary Nodes
Partition Non-Loop Secondary Nodes Among Edges of Graph
Label Edges of Graphs with Non-Loop Secondary Nodes
Partition Loop Secondary Nodes Among Loops
Label Loops with Loop Secondary Nodes
Construct Cyclic Skeletons
Construct Ciliated Skeletons by Labelling with Free Valences
Label Nodes with Atom Names to Yield Ring-Superatoms
Part C
Acyclic Generator
Combine Ring-Superatoms with Remaining Pot (See Appendix O)
Final Structures (Standard Chemical Notation)
or
$CH_2$
$H_2C$

**Figure 2.** Major steps in the generation of isomers as illustrated for $C_6H_8$. This example outlines the method for one superatom partition, that which allocates all atoms to a single superatompot with no atoms in the remaining pot.

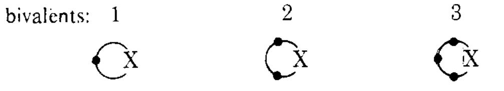
Table III. Vertex-Graphs Necessary for Construction of Isomers of $C_6H_8$ (Partial Listing of the Catalog)[a]

| Planar representation | Name[b] | No. of nodes of degree Three | Four | Remarks |
|---|---|---|---|---|
| | 2A (hosahedron) | 2 | 0 | Regular trivalent graph of two nodes |
| | 4AA | 4 | 0 | Regular trivalent graphs of four nodes |
| | 4BB (tetrahedron) | 4 | 0 | |
| | "Singlering k" | 0 | 0 | A single ring composed of *k* secondary nodes |
|  | Tetravelent dihedron | 0 | 2 | Two nodes of degree four |
|  | "Daisy" | 0 | 1 | A single quaternary node |
|  | **$3BCB** | 2 | 1 | |

[a] The listing of ref 10 has been expanded to include vertex-graphs of other combinations of nodes of degree three and four.[18a,b] The completeness of the Catalog has been verified where possible by independent graph construction methods[18] and by comparison with Balaban's compilations[9b,c] where appropriate. [b] Names, except those in quotation marks, taken from Lederberg.[10]

The method for specification of loops is discussed in Calculation of Loops, Appendix C.

Chart II



**Partitioning of Secondary Nodes among Loops and Nonloops.** The secondary nodes in the degree list are partitioned between the loops (if any) calculated in the previous step and the remaining nonloop portion of the eventual graph. Aspects of this partitioning step are presented in Partitioning of Secondary Nodes Among Loops and Nonloops, Appendix C. Results for the example are indicated in Figure 2.

**Reduced Degree List.** This procedure yields the reduced degree list which contains none of the secondary nodes originally present in the degree list. Any secondary nodes appearing in the reduced degree list are termed "special" secondary nodes as these nodes will have loops attached in subsequent steps.

**Vertex-Graphs.** The reduced degree lists are used to specify a set of vertex-graphs for the eventual ring-superatoms. All two-connected structures can be described by their vertex-graphs, which are, for many structures, regular trivalent graphs. This concept has been described in detail by Lederberg,[10] who has also presented a generation and classification scheme for such graphs. Given a set of all vertex-graphs, the set of all ring-superatoms may be specified.[15] The vertex-graphs are maintained by the program in the Catalog. Catalog entries for regular trivalent graphs possessing two and four nodes are presented in Table III. This list must be supplemented by additional vertex-graphs to cover several special cases required for generation of all structures for the example. These are also presented in Table III. With the reduced degree list of a superatompot, the program requests the appropriate Catalog entries. In the example (Figure 2), the reduced degree list (0,0,2) specifies vertex-graphs containing two quaternary nodes (tetravalent dihedron). The reduced degree list (0,4,0) specifies regular trivalent graphs of four nodes, of which there are two: **4AA** and **4BB** (Table III). When *only* secondary nodes are present in the reduced degree list, the graph "Singlering" (Table III) is utilized.

**Interlude.** Up to this point, the program has effectively decomposed the problem into a series of subproblems, working down from the total pot of atoms through a series of partitions and subpartitions to the set of possible vertex-graphs. In subsequent steps the vertex-graphs are expanded to the final structures by a series of constructive graph labelings (Table IV).

Table IV. The Six Graph Labeling Steps Performed by the Labeling Algorithm

| Labeling step | Function |
|---|---|
| 1 | Label edges of vertex-graphs with special secondary nodes |
| 2 | Label edges of resulting graphs with non-loop secondary nodes |
| 3 | Label loops of resulting graphs with loop secondary nodes |
| 4 | Label nodes of cyclic skeletons with free valences |
| 5 | Label nodes of ciliated skeletons with atom names |
| 6 | Label free valences of superatoms with radicals (see Appendix D) |

**Labeling Edges of Vertex-Graphs with Special Secondary Nodes.** Special secondary nodes are those that will have loops attached. The specification of the possible attachments of the nodes to the graph is a "labeling" procedure. This is the first of six such graph labeling steps performed by the program (Table IV). All of these labeling steps involve the same combinatorial problem, that of associating a set of *n* labels, not necessarily distinct, with a set of objects with arbitrary symmetry.[13] The same labeling algorithm is utilized for each of the six labeling steps. A description of the underlying mathematics and proof of completeness and irredundancy appears separately.[14]

Some aspects of the first labeling step indicate how equivalent labelings (which would eventually yield duplicate structures) may be avoided prospectively, by recognition of the symmetry properties of the graph in the first labeling, the vertex-graph. These symmetry properties are expressed in terms of the permutation group (see Appendix A and ref 13 and 14) on the edges of the vertex-graph. This permutation group, which defines the equivalence of the edges, may be specified in the Catalog or, alternatively, calculated as needed by a separate part of the structure generator. As subsequent steps are executed, a new permutation group (*e.g.*, on the nodes for labeling step four, Table IV) is derived as necessary.[13] Thus, only labelings which result in unique expansions of the structure are permitted. The reader examining Figure 2 may note that for this simple example the symmetries of the vertex-graph and subsequent skeletons can be discerned easily by eye. For example, all edges of the tetravalent dihedron are equivalent, as are all the edges of the regular trivalent graphs **2A** and also **4BB.** The **$3BCB** graph (Table II, Figure 2) has four equivalent edges and one other edge, and so forth. In the general case, however, the symmetries of the vertex-graphs and subsequent expansions thereof are not always obvious.

With the group on the edges specified, the labeling of the vertex-graphs with special secondary nodes is carried out. The results of this procedure for partitons containing loops are indicated in Figure 2.

**Labeling with Nonloop Secondary Nodes.** The graphs which resulted from the previous labeling are now labeled with the partitions of nonloop secondary nodes (see Partitioning of Nonloop Secondary Nodes among Edges, Appendix C). Each of the five partitions for the tetravalent dihedron in Figure 2 results in a single labeling, as all four edges of the graph are equivalent. When edges are distinguishable there may be several ways to label a graph with a single partition. There are, for example, for the **$3BCB** graph, two ways to label with the partition 3,0,0,0,0, four ways with the partition 2,1,0,0,0, and three ways with the partition 1,1,1,0,0 (Figure 2).

**Labeling with Loop Secondary Nodes.** There remain unassigned to the graphs at this point only secondary nodes which were assigned to loops. These nodes are first partitioned among the loops. (see Partitioning of Loop Secondary Nodes among Loops, Appendix C). For example, following the path from the degree list (4,0,2) through labeling with nonloop secondary nodes (Figure 2), there are two ways of labeling the two equivalent loops with four secondary nodes. There is one way to label the two loops of the adjacent graph with three secondary nodes and one way of labeling the two loops of each of the two remaining graphs in this section of Figure 2 with two secondary nodes. In this example ($C_6U_3$) the loops in every case are equivalent or there is only one loop to be labeled. In the general case, loops may not be equivalent, resulting in a greater number of ways to label loops with a given partition of secondary nodes.

**Cyclic Skeletons.** The previous labeling steps specified the number of secondary nodes on each edge of and loop attached to the vertex-graphs. All atoms in the original superatompot are thus accounted for. A representation of the result is the cyclic skeleton, where nodes and their connections to one another are specified. (These skeletons begin to resemble conventional representations of chemical structures.)

**Labeling with Free Valences.** The nodes in a cyclic skeleton are then labeled with free valences, yielding ciliated skeletons. This labeling is trivial in the example, as all atoms are of the same valence (four) (Figure 2). Free valence labeling is performed with knowledge of how many atoms of each valence were present in the original superatompot, but independent of the *identities* of the aoms. The combinatorial complexity of this labeling problem follows from the possible occurrence of atoms with differing valences. In the general case, there may be several ways to perform this labeling on a single cyclic skeleton, whereas in the $C_6U_3$ example there is only one way.

**Labeling with Atom Names.** The nodes of a ciliated skeleton are then labeled with atom names to yield the ring-superatom(s). Again this labeling is trivial in the example, as only one type of atom is present (carbon), yielding in each case only a single superatom (Figure 2). If there is more than one type of atom with the same valence (*e.g.*, silicon and carbon), the labeling problem is more complex. Each node of appropriate valence may be labeled with either type of atom. Duplicate structures are avoided by calculations involving the group pertaining to the set of nodes of equal valence.[13]

## Part C. Acyclic Generator

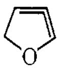
The superatom partition expanded in the example had no atoms assigned to acyclic chains (remaining pot). The set of ring-superatoms on completion of part B, above, thus yields the set of 36 structures on placement of a hydrogen atom on each free valence (Figure 2). If the superatom partition (partitions 2-11, Table II) contained more than one superatompot or any atoms in the remaining pot, the acyclic generator must be used to connect the segments of the structure in all ways. This procedure is described in detail in Appendix D.

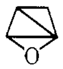
## Discussion

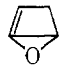
**Completion of $C_6H_8$.** The example (Figure 2) has considered only expansion of a single superatom partition. It might be instructive for the reader to attempt to generate all, or at least the remaining, structures for $C_6H_8$. The number of solutions is presented in a subsequent section. If the algorithm as outlined in Figure 2 is followed, it is suggested that the initial superatom partitions in Table II be examined carefully. These partitions yield some indication of the types of structures which will result from each partition. For example, partition 4, $C_3U_3$ in a single superatompot, plus three carbons in the remaining pot, should yield all structures containing a three-membered ring possessing two double bonds or a triple bond. As there are only two free valences, the remaining atoms can be in a single chain (as a propyl or isopropyl radical) or as a methyl and an ethyl group, but not as three methyl groups.

**Completeness and Irredundancy.** Although a mathematical proof of the completeness and irredundancy of the method exists,[15] there is no guarantee that the implementation of the algorithm in a computer program maintains these desired characteristics. Until it is possible to "prove" large programs, the possibility of undetected software or hardware errors will always remain. Confidence in the completeness and irredundancy of a program of this complexity can be engendered in the following ways.

(1) Verification of the program's performance by another, completely independent approach. An independent method has been developed which enumerates, but does not construct, all isomers of compositions containing C, H, N, and O.[18c] It is interesting that the program for simple counting of the solutions is significantly slower than construction of all of the solutions, despite some effort to improve the efficiency of the former program. Thus, due to limitations of computer time, we have been limited to compositions containing only five or fewer nonhydrogen atoms. For these cases, however, the numbers of isomers obtained by both programs agree. Balaban has presented lists of isomers of $C_4H_4$, $C_6H_6$, $C_5H_8$, and $C_4H_4O$.[9d] These lists were derived from his tables[9d] of graphs of degrees 2-4 and orders (numbers of nodes) 1-5. Although we agree with his lists of hydrocarbon isomers, the list of isomers of $C_4H_4O$ is incomplete. The structure generator provides 62 structures (as opposed to 59). The three missing structures are:







These structures should have been produced following Balaban's method.[9d] The fact that they were not points out the difficulties inherent in any procedure for isomer generation in which manual steps are involved (see below).

(2) Testing by manual generation of structures. Several chemists, all without knowledge of the algorithm described above, have been given several test cases, including $C_6U_3$, from which structures were generated by hand. Familarity with chemistry is no guarantee of success, as evidenced by the performance of three chemists for the superficially simple case of $C_6U_3$ ($C_6H_8$, Table V).

This example indicates that for more than very trivial cases, it is extremely difficult to avoid duplicates (tricyclics,

Table V. Performance of Three[a] Chemists in Manual Generation of Isomers of $C_6H_8$ ($C_6U_3$) (159 Isomers)

| | No. generated | Type of error |
|---|---|---|
| Chemist 1 | 161 | 4 duplicates; 4 omissions;<br>2 with 7 carbon atoms |
| Chemist 2 | 168 | 16 duplicates; 7 omissions |
| Chemist 3 | 160 | 2 duplicates; 1 omission |

[a] One Ph.D. and two graduate students.

for example, are difficult to visualize when testing for duplicates) and omissions. Omissions appear to result from both carelessness and neglect of ring systems that are implausible or unfamiliar. The program seems better at testing the chemist than *vice versa.* In every instance of manual structure generation, no one has been able to construct a legal structure that the program failed to construct. No one has been able to detect an instance of duplication by the program. This performance builds some confidence, but manual verification of more complicated cases is extremely tedious and difficult. Isomers for many empirical formulas have been generated, and some results are tabulated in Table VI.

Table VI. Number of Isomers for Several Empirical Formulas

| Empirical formula | Example compd | No. of isomers | Manually verified |
|---|---|---|---|
| $C_6H_6$ | Benzene | 217 | Yes |
| $C_6H_8$ | 1,3-Cyclohexadiene | 159 | Yes |
| $C_6H_{10}$ | Cyclohexene | 77 | Yes |
| $C_6H_{12}$ | Cyclohexane | 25 | Yes |
| $C_6H_{14}$ | Hexane | 5 | Yes |
| $C_6H_6O$ | Phenol | 2237 | No |
| $C_6H_{10}O$ | Cyclohexanone | 747 | No |
| $C_6H_{12}O$ | 2-Hexanone | 211 | Yes |
| $C_3H_4N_2$ | Pyrazole | 155 | No |
| $C_3H_6N_2$ | 2-Pyrazoline | 136 | Yes |
| $C_3H_8N_2$ | Tetrahydropyrazole | 62 | No |
| $C_3H_{10}N_2$ | Propylenediamine | 14 | Yes |
| $C_3H_9P_1$ | (Pentavalent P) | 110 | No |

The choice of examples has been motivated by a desire to test all parts of the program where errors may exist while keeping the number of isomers small enough to allow verification. In this manner all obvious sources of error have been checked, for example, construction of loops on loops, multiple types of atoms of the same valence (*e.g.*, Cl, Br, I), and examples containing atoms of several different valences including penta- and hexavalent atoms.

(3) Varying the order of generation. The structure of the program permits additional tests by doing some operations in a different order. For example, one variation allowed is to leave hydrogens associated with the atoms in each partition rather than to strip them away initially and place them on the remaining free valences in the last step. Each such test has resulted in the same set of isomers.

(4) Using Polya enumeration[6] at the various labeling steps of the procedure to verify the correctness of subparts of the program. Using various combinatorial formulas, one can ensure that the results of at least parts of the program are consistent with independent calculations. This approach was used extensively in the development of the labeling algorithm.

In summary, the verification procedures utilized have all indicated absence of errors in the computer implementation of the algorithm. Also, there is no clear reason why generation of larger sets of isomers should not also proceed correctly. The final verdict, however, must await development of new mathematical tools for verification by enumeration (see above) or an alternative algorithm.

**Constraints.** The structure generator is designed to produce a list of all possible graph isomers (Appendix B). This list contains many structures whose existence seems unlikely based on present chemical knowledge. In addition, the program may be called on to generate possible structures for an unknown in the presence of a body of data on the unknown which specify various features, *e.g.*, functional groups, of the molecule. In such instances, mechanisms are required for constraining the generator to produce only structures conforming to specified rules. The implementation of the acyclic generator possessed such a mechanism in the form of GOODLIST (desired features) and BADLIST (unwanted features)[3] which could be utilized during the course of structure generation.

The complete structure generator is less tractable. As in prospective avoidance of duplicate structures, it is important that unwanted structures, or portions thereof, be filtered out as early in the generation process as possible. It is relatively easy to specify certain general types of constraints in chemical terms, for example, the number of each of various types of rings or ring systems in the final structure, ring fusions, functional groups, substructures, and so forth. It is not always so easy to devise an efficient scheme for utilizing a constraint in the algorithm, however. As seen in the above example (Figure 2), the expanded superatom partition results in what would be viewed by the chemist as several very different ring systems.

The design of the program facilitates some types of constraints. For example, the program may be entered at the level of combining superatoms to generate structures from a set of known substructures. If additional atoms are present in an unknown configuration, they can be treated as a separate generation problem, the results of which are finally combined in all ways with the known superatoms. This approach will not form additional two-connected structures, however. Constraints which disallow an entire partition may be easily included. For example, it is possible to generate only pure ring isomers by "turning off" the appropriate initial superatom partitions.

Much additional work remains, however, before a reasonably complete set of constraints can be included. The implementation of each type of constraint must be examined and tested in detail to ensure that the generator remains thorough and irredundant.[28]

## Conclusions

The algorithm summarized in this paper permits the substantial realization of the graphical structures that constitute the domain of organic chemistry. The version of the algorithm presented here ignores the tetrahedral symmetry of the valences of the carbon atom. However, the topological framework readily admits of systematic tests for asymmetric centers which can then be assigned to the dichotomous categories of the alternating group $A_4$. This framework also provides a simple, systematic weighting of radicals for assignment of precedence that proves to be, if anything, even more straightforward, comprehensive, and free from ambiguity than the Cahn-Ingold-Prelog conventions.[19]

The mathematical framework of our analysis is a mapping of chemical bonds onto the edges of topological graphs. This simplification can lead to disparities, for example, in the description of coordination complexes, the bonds of which are nonequivalent. The symmetries of such complexes are similar to those of certain superatoms, suggesting an obvious and easy way to extend the system. Likewise, the system does not now accommodate isomerism based on steric hindrance, or the association of molecules by secondary forces, or by noncovalent constraints. For example, from a topological standpoint, threaded molecules, or catenanes, are disjoint graphs. Nor do we attempt to display the geometric conformations of molecules; indeed, some topologically plausible structures may be chemically unrealizable.

Conversely, implausible constructs, such as carbon atoms possessing "inverted" tetrahedral geometry,[20] may become reality by empirical discovery. The constraints on chemically plausible structures depend on the domain specified by the chemist. A DENDRAL[3] system for molecular structure elucidation[21,28] (based on the structure generator described in this work) of molecules in frozen hydrogen matrices would have different constraints from a version useful to biochemists.

Chemists hitherto have been able to explore the *de facto* boundaries of their domain without explicit maps. The exhaustive and efficient study of all possible structures can now be facilitated with assistance of computer programs that can help assure that no possible construction has been overlooked.[22]

## Experimental Section

The structure generator program is written in the INTERLISP language[23] (with a small section for drawing structures written in Fortran) and is mounted on a DEC PDP-KI-10 computer at Stanford. Here it constructs isomers at the rate of 100-500/min, depending on the list of atoms. Further programming effort might be expected to improve its efficiency by a factor of perhaps 10 for demanding applications, but only at a substantial cost in recoding and debugging a version in a lower level language.

A listing of the program source text is available from the authors. However, the complexity of this and similar programs makes the mere reading of such texts an uninviting and perhaps unrewarding task absent a convenient mechanism for actually running and testing the program on an operational facility which maintains the INTERLISP language.

The Stanford University Medical Experimental Computer Resource (SUMEX) has in fact been initiated and funded in cooperation with the Biotechnology Resources Branch of the National Institutes of Health with the view of facilitating higher level communication of complex programs of this kind, particularly those that relate to applications of artificial intelligence to medical research. To this end, we will endeavor (within limits of available resources) to enable interested investigators to secure updated versions of the programs described in this paper, or to run them at SUMEX from their own remote terminals *via* a computer communications network. Those desiring access to the structure generator to solve specific questions of isomerism should contact either J. Lederberg (Department of Genetics) or D. H. Smith (Department of Chemistry) for additional details.

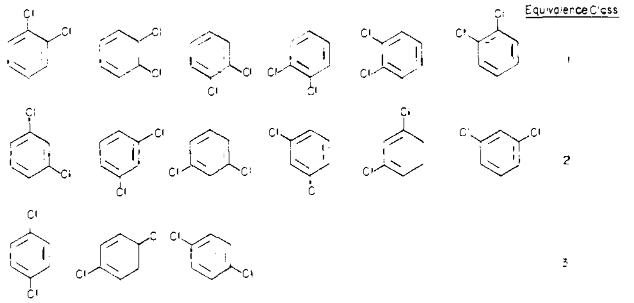
Also available *via* this route are the compendia of vertex-graphs which comprise the Catalog. Although partial listings are available,[18b] the Catalog has recently been extended to include many additional sets of graphs including multiple nodes of higher valence. These extensions are based on modifications of previously described procedures[18a] and will be described separately.

**Acknowledgments.** We wish to express our thanks to Professor Harold Brown, who provided valuable assistance in the formulation of the mathematical principles underlying the structure generator. We also wish to thank Raymond Carhart, William Fitch, and Jan Simek for their efforts on manual verification of the structure generator.

## Appendix A. Equivalence Classes and Finite Permutation Groups

Two members of a set of possible isomers may be defined to be equivalent if a specified transformation of one member causes it to be superimposable upon another member of the set. For example, there are 15 possible ways of attaching two chlorine and four hydrogen atoms to a spatially "frozen" benzene ring (Chart III).

**Chart III**



If rotations by multiples of 60° are specified as allowed transformations, the 15 structures fall logically into three classes, termed "equivalence classes" (Chart III). Within each equivalence class, structures may be made superimposable by the rotational transformation. If one element (in this case a molecular structure) is chosen from each equivalence class, the complete set of possible structures is determined, without duplication. It is the task of the labeling algorithm to produce one and only one graph labeling corresponding to one member of each equivalence class.

The set of transformations which define an equivalence class is termed a "finite permutation group." This permutation group may be calculated based on the symmetry properties of a graph (or chemical structure in the example of Chart III). This calculation provides the mechanism for prospective avoidance of duplication. These procedures are described more fully in the accompanying paper.[13]

## Appendix B. Isomerism and Symmetry

Appendix A introduced the concept of equivalence classes and finite permutation groups. The selection of transformation (Appendix A) directs the calculation of the permutation group and thus defines the equivalence classes. Different types of transformation may be allowed depending on the symmetry properties of the class of isomers considered. This Appendix discusses several of the possible types of isomerism, most of which are familiar to chemists. The reader seeking a more thorough discussion of some types of isomerism discussed below is referred to an exposition of molecular symmetry in the context of chemistry and mathematics.[24]

Isomers are most often defined as chemical structures possessing the same molecular formula but different structures. Different concepts of symmetry give rise to different classes of isomers, some of which are described below.

**Permutational Isomers.** Permutational isomers are isomers which have in common the same skeleton and set of ligands. They differ in the distribution of ligands about the skeleton. Gillespie, *et al.*,[25] and Klemperer[26] have used the concept of permutational isomers to probe into rearrangement or isomerization reactions.

**Stereoisomers.** Ugi, *et al.*,[24] have defined the "chemical constitution" of an atom to be its bonds and bonded neighbors. Those permutational isomers which differ only by permutations of ligands at constitutionally equivalent positions form the class of stereoisomers.

**Isomers under Rigid Molecular Symmetry.** If one perceives molecular structures as having rigid skeletons, the physical rotational (three-dimensional) symmetries and transformations may be readily defined. Each transformation causes each atom (and bond) to occupy the position of another or same atom (and bond) so that the rotated structure can physically occupy its former position and at the same time be indistinguishable from it in any way. This is the most familiar form of symmetry. Under this type of symmetry, conformers are distinguishable and belong in distinct equivalence classes. Every transformation is orthogonal and preserves bond angles and bond lengths as well as maintaining true chirality.

If one allows other orthogonal transformations that alter chiral properties of structures, equivalence classes result that treat both the left-handed and right-handed forms of chiral molecules to be the "same." Thus a "mirror image" transformation when suitably defined permits the left-handed form to exactly superimpose the right-handed form and *vice versa*.

**Isomers under Total Molecular Symmetry.** If in addition to the above-mentioned rigid molecular transformations one recognizes the flexional movements of a nonrigid skeleton, a dynamic symmetry group may be defined. Under this definition, different conformers now are grouped together. Thus the "chair" and "boat" conformations of cyclohexane belong to the same equivalence class under dynamic symmetry. The permutation group of skeletal flexibility is computable separately and independently of rigid molecular symmetry. One can then view total molecular symmetry as the product of the two finite permutation groups.

**Isomers under Connectivity Symmetry.** The concept of connectivity symmetry was introduced previously (Method section). Every permutation of atoms and bonds onto themselves is a symmetry transformation for connectivity symmetry if (a) each atom is mapped into another of like species, *e.g.*, N to N, C to C, O to O; and (b) for every pair of atoms, the connectivity (none, single, double, triple, . . .) is preserved in the mapping; *i.e.*, the connectivity of the two atoms is identical with the connectivity of the atoms they are mapped into.

One can readily recognize that transformations as defined automatically preserve the valence and bond distribution of every atom. It is very probable that readers accustomed to three-dimensional rotational and reflectional symmetries will tend to equate them with the symmetries of connectivity. It is emphasized again that connectivity symmetry does not consider bond lengths or bond angles, and it includes certain transformations that are conceivable but have no physical interpretation save that of permuting the atoms and bonds.

## Appendix C

**Superatom Partitions.** The first step is to replace the hydrogen count with the degree of unsaturation. The number of unsaturations (rings plus double bonds) is determined from the empirical formula in the normal way, as given in eq 1, where $U$ = unsaturation, $i$ = valence, $n$ = maximum

$$U = \tfrac{1}{2}\left(2 + \sum_{i=1}^{n}(i-2)a_i\right) \qquad (1)$$

valence in composition, and $a_i$ = number of atoms with valence $i$. If the unsaturation count is zero, the formula is passed immediately to the acyclic generator. Specifying the unsaturations as $U$'s, the example $C_6H_8$ becomes $C_6U_3$ (hydrogen atoms are omitted by convention).

There are several rules which are used during the partitioning scheme, as follows:

(I) The resulting formula is stripped of other univalent atoms (*e.g.*, chlorine) as such atoms cannot be part of two-connected ring-superatoms. These univalent atoms are relegated to the pot of remaining atoms.

(II) The remaining pot in a given partition (those atoms not allocated to superatompots) can contain *no* unsaturations. Thus *all* rings and/or multiple bonds will be generated from the superatompots.

(III) It follows that every superatompot in the partition must contain at least two atoms of valence 2 or higher plus at least one unsaturation. If there are no unsaturations then no rings could be built. In addition, an unsaturation cannot be placed on a single atom. This rule defines the minimum number of atoms and unsaturations in a superatompot.

(IV) The maximum number of unsaturations in a superatompot is given by eq 2, where $U_{\max}$ = maximum unsatura-

$$U_{\max} = \tfrac{1}{2}\left(\sum_{i=3}^{n}(i-2)a_i\right) \qquad (2)$$

tion of a superatompot, $n$ = maximum valence in composition, $i$ = valence, and $a_i$ = number of atoms with valence $i$. Superatoms must possess at least one free valence,[12] so that superatompots with no free valences, *e.g.*, $O_2U_1$ or $C_2U_3$, are not allowed, unless the superatompot contains all atoms in the empirical formula (since no univalents, and thus no hydrogens, are allowed in a superatompot, this is indeed a rare occurence).

(V) The maximum number of superatompots for a given formula is defined by eq 3, where $n$ = maximum valence in

$$S_{\max} = \tfrac{1}{2}\sum_{i=2}^{n}a_i \qquad (3)$$

composition, $S_{\max}$ = maximum number of superatompots in a superatom partition, and $a_i$ = number of atoms with valence $i$. (Note: the summation is over all atoms of valence >2; univalents are not considered).

Rules I–V define the allowed partitions of a group of atoms into superatompots. These rules do not, however, prevent generation of equivalent partitions, which would eventually result in duplicate structures. By defining a canonical ordering scheme to govern partitioning, we prevent equivalent partitions. One such canonical ordering is as follows.

(a) Partition in order of increasing number of superatompots.

(b) For each entry in each part of (a), partition in order of decreasing size of superatompot by allocation of atoms one at a time to the remaining pot.

(c) Each individual partition containing two or more superatompots must be in order of equal or decreasing size of the superatompot. In other words, the number of atoms and unsaturations in superatompot $n + 1$ must be equal to or less than the number in superatompot $n$. The program notes the equality of superatompots in a partition to avoid repetition.

The application of rules I–V is best illustrated through reference to the example of $C_6U_3$. The maximum number of superatompots for this example is three (eq 3). There is one way to partition $C_6U_3$ into one superatompot with no remaining pot (partition 1, Table II). Subsequent assignment of carbon atoms one at a time to the remaining pot results in partitions 2–4, Table II. The next partition following the sequence 1–4 would be $C_2U_3$ with $C_4$ assigned to the remaining pot. This partition is forbidden as $C_2U_3$ has no free valences. The three ways to partition $C_6U_3$ into two superatompots are indicated along with the corresponding partitions following assignment of atoms to the remaining pot, as partitions 5–10, Table II. There is only one unique way of partitioning $C_6U_3$ into three superatompots, partition 11, Table II.

**Calculation of Free Valence.** The expression for the free valence of a superatompot is given by eq 4, where $U$ = un-

$$\mathrm{FV} = \left(2 + \sum_{i=3}^{n}(i-2)a_i\right) - 2U \qquad (4)$$

saturation of superatompot, $i$ = valence, $n$ = maximum valence in composition, $a_i$ = number of atoms with valence $i$, and FV = free valence.

**Partitioning of Free Valence.** Because ring-superatoms are two-connected structures, two valences of each atom of a superatompot must be used to connect the atom to the ring-superatom. Thus no free valences can be assigned to bivalent nodes in the valence list, a maximum of one to each trivalent, a maximum of two to each tetravalent, and so forth. The example (Figure 2) is further simplified in that there are only tetravalent nodes in the valence list. Inclusion of trivalent nodes (*e.g.*, nitrogen atoms) merely extends the number of possible partitions. The free valences are partitioned among the tetravalent nodes in all ways, as illustrated in Figure 2. It is important to note that removal of atom names makes all $n$-valent ($n = 2$ or 3 or . . .) nodes in the valence list equivalent at this stage. Thus the partitions (of eight free valences among six tetravalent nodes) 222200, 222020, 222002, . . . . . . , 002222 are all equivalent. Only one of these partitions is considered to avoid eventual duplication of structures.

**Calculation of Loops.** There are several rules which must be followed in consideration of loop assignment to ring-superatoms. The minimum (MINLOOPS) and maximum (MAXLOOPS) numbers of loops for a given valence list are designated by eq 5 and 6, where MINLOOPS = minimum

$$\text{MINLOOPS} = \max\left\{0, a_2 + 1\middle/\left[2\left(2n - \sum_{j=2}^{n} ja_j\right)\right]\right\} \quad (5)$$

$$\text{MAXLOOPS} = \min\left\{a_2, 1\middle/\left[2\sum_{j=4}^{n}(j-2)a_j\right]\right\} \quad (6)$$

number of loops, MAXLOOPS = maximum number of loops, $a_j$ = number of nodes with degree $j$, $j$ = degree, and $n$ = highest degree in list ($a_n \neq 0$). The form of the equations results from the following considerations.

(1) Only secondary nodes may be assigned to loops. Nodes of higher degree will always be in the nonloop portion of the ring-superatom.

(2) A loop, by definition, must be attached by two bonds to a single node in the resulting ring-superatom. The loop cannot be attached through the free valences. Thus the degree list must possess a sufficient number of quaternary or higher degree nodes to support the loop(s).

(3) Each loop must have at least one secondary node, which is the reason MAXLOOPS is restricted to be at most the number of secondary nodes in the degree list (eq 6).

(4) There must be available one unsaturation for each loop (this is implicit in the calculation of MINLOOPS and MAXLOOPS) as each loop effectively forms a new ring.

**Partitioning of Secondary Nodes between Loops and Nonloops.** For each of the possible numbers of loops (0,1, . . .), the secondary nodes are removed from the degree list and partitioned among the loops, remembering that the loops are at present indistinguishable and each loop must receive at least one secondary node. In the example (Figure 2), starting with the degree list (4,0,2), there are three ways of partitioning the four secondary nodes between two loops and the remaining nonloop portion. Removal of the four secondary nodes from the degree list and assignment of two, three, or four of them to two loops result in the list specified in Figure 2 as the "reduced degree list." Specification of two loops transforms the two quaternary nodes in the degree list into two secondary nodes. This results from the fact that two valences of a quaternary or higher degree node must be used to support each loop. These are "special" secondary (or higher, for atoms with valence >4) nodes, however, as these particular nodes will have loops attached as the structure is built up. Thus, in the example, any secondary nodes which are found in the reduced degree list will have a loop attached in a subsequent step. The degree list (4,0,2) thus becomes the reduced degree list (2,0,0) in the partition specifying two loops (Figure 2). Similarly, the partition of one loop for the degree list (3,2,1) results in a reduced degree list of (1,2,0) with the three original secondary nodes partitioned between loop and nonloop portions (Figure 2).

If, after the first, second, . . . , $n$th loop partition, there remain one or more quaternary or higher degree nodes in the reduced degree list, the list must be tested again for the possibility of additional loops. Each loop partition will result in an additional set of structures. The second pass will yield those structures possessing loops on loops, and so forth. One such superatom which would be generated in this manner from a composition of (at least) $C_6U_5$ is **15.**

$$C{=}C{=}C{=}C{=}C{=}C$$

**15**

**Partitioning of Nonloop Secondary Nodes among Edges.** The secondary nodes which were not assigned to loops ("nonloop secondary nodes") are partitioned among the edges of the graphs after labeling with special secondary nodes, or loops. Loops are not counted as edges. There are, for example, five ways to partition four nonloop secondary nodes among the edges of the vertex-graph possessing two quaternary nodes (Figure 2).

**Partitioning of Loop Secondary Nodes among Loops.** This partitioning step is carried out assuming indistinguishability of the loops. Each loop must receive at least one secondary node, which limits the number of possible partitions. Results are presented in Figure 2.

## Appendix D. Acyclic Generator

A method of construction of structures similar to the method for acyclic isomers is utilized to join multiple ring-superatoms and remaining atoms. The DENDRAL algorithm for construction of acyclic isomers[3,10,21] relied on the existence of a unique central atom (or bond) to every molecule. The present acyclic generator uses the same idea. The present algorithm, though simpler in not having to treat interconnection of atoms or ring-superatoms through multiple bonds, is more complex because of the necessity to deal with the symmetries of the ring-superatoms.

**D1. Method for the Case With Even Number of Total Atoms.** The superatom partition $C_2U_2/C_2U_1/-/C_2$ (partition 7, Table II, and Figure 2) will be used here to illustrate this procedure. The superatompots $C_2U_2$ and $C_2U_1$ have exactly one possible ring-superatom for each (see Table VII).

Table VII

| Superatompot | Superatom |
|---|---|
| $C_2U_2$ | -C≡C- |
| $C_2U_1$ | >C=C< |

Thus acyclic structures are to be built with -C≡C-, >C=C<, and two C's.

There are an even number of atoms and ring-superatoms. The structures to be generated fall into two categories: (a) those with bond centroid; and (b) those with an atom centroid.[21]

**Category A. Bond Centroid (See Figure 3). Step 1. Partition into Two Parts.** The atoms and ring-superatoms in the list of superatoms are partitioned into two parts, with each part having exactly half the total number of items. Each atom or ring-superatom is a single item. Each part has to satisfy eq 7, called the Restriction on Univalents.

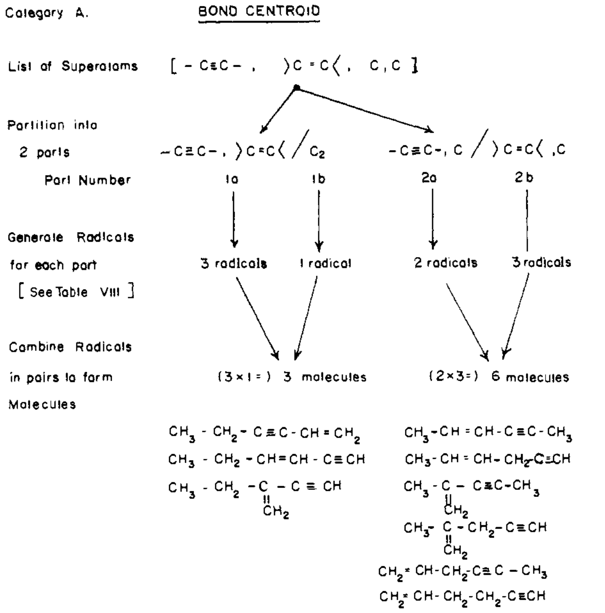


**Figure 3.** Operation of the acyclic generator for the case of a bond as a centroid for the structures.

Restriction on Univalents

$$a_1 \leq \left[\sum_{i=2}^{n}(i-2)a_i\right] - 1 \qquad (7)$$

$i$ = valence, $a_i$ = number of atoms or superatoms of valence $i$, and $n$ = maximum valence in composition.

There are two ways of partitioning the four items into two parts (Figure 3). The restriction on univalents is satisfied in each case. The restriction will disallow certain partitions that have "too many" [27] univalents other than hydrogens and therefore is essential only in partitioning compositions that contain any number of nonhydrogen univalents.

**Step 2. Generate Radicals from Each Part.** By use of a procedure described in section D3, radicals are generated from each part in each partition. The result of application of this procedure to the example is shown in Table VIII.

**Table VIII.** Radicals Generated from Given Parts

| Part | Radicals |
|---|---|
| (1a) $-C{\equiv}C-$, $>C{=}C<$ | → $-C{\equiv}CCH{=}CH_2$<br>$-CH{=}CHC{\equiv}CH$<br>$-\underset{\underset{\displaystyle CH_2}{\|}}{C}C{\equiv}CH$ |
| (1b) $C_2$ | → $-CH_2CH_3$ |
| (2a) $-C{\equiv}C-$,C | → $-C{\equiv}CCH_3$<br>$-CH_2C{\equiv}CH$ |
| (2b) $>C{=}C<$, C | → $-CH{=}CHCH_3$<br>$-\underset{\underset{\displaystyle CH_2}{\|}}{C}CH_3$<br>$-CH_2CH{=}CH_2$ |

**Step 3. Form Molecules from Radicals.** The radicals are combined in unique pairs, within each initial partition. Each pair gives rise to a unique molecule, for each of which the centroid is a bond. There are nine such molecules for the example chosen (Fig. 3).

**Category B. Atom Centroid (see Figure 4). Step 1. Selection of Centroid.** One must consider every unique atom or ring-superatom that has a free valence of 3 or higher as an atom centroid.[10,21] In the example, of three candidates available, $-C{\equiv}C-$, $>C{=}C<$, and C, the first is not chosen for it has a free valence of only 2.

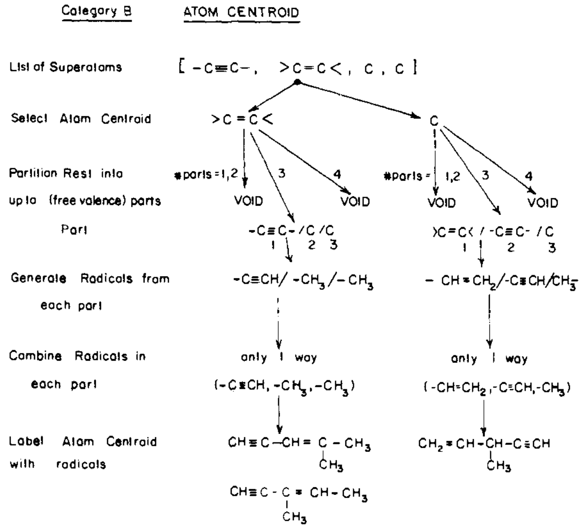


**Figure 4.** Operation of the acyclic generator for the case of an atom or superatom as a centroid for the structures.

**Step 2. Partition the Rest of the Atoms.** The atom or ring-superatom chosen for the centroid is removed from the set, and the rest are partitioned into a number of parts less than or equal to the valence of the central atom. Each part must have less than half the total number of items being partitioned (again a ring-superatom is a single item). Each part must satisfy the restriction on univalents (eq 7).

Thus, for the case where a carbon is the centroid, four partitions are attempted. The condition that each part has less than or equal to one-half the number of superatoms remaining after selection of the central atom must be satisfied, or at most one for this example. There is exactly one partition for three parts, *i.e.*, one in each. The partitions are shown in Figure 4.

**Step 3. Generate Radicals.** Once again, using the procedure described in section D3, radicals are constructed for each part in each partition. For example, the partition $-C{\equiv}C-$ gives rise to exactly one possible radical $-C{\equiv}CH$ (Figure 4).

**Step 4. Combine Radicals.** Although in the example shown every part generates only one radical, in the general case there will be many radicals for each part. If so, the radicals must be combined to give all unique combinations of radicals within each part.

**Step 5. Form Molecules from Central Atom and Radicals.** If the centroid is not a ring-superatom but is a simple atom, then each combination of radicals derived in step 4 defines a single molecule that is unique. Thus, for example, when C is chosen as the centroid, step 4 gives one combination of radicals which determines a single molecule when connected to the central C (see Figure 4).

If the centroid is a ring-superatom, and the valences of the ring-superatom are not identical, then different ways of distributing the radicals around the center may yield different molecules. Labeling of the free valences of the central ring-superatom with radicals treated as labels (supplemented with adequate numbers of hydrogens to make up the total free valence of the ring superatom) generates a complete and irredundant list of molecules. Thus $>C{=}C<$ is labeled with the label set: one of $-C{\equiv}CH$, two of $-CH_3$, and one of $-H$. There are two unique labelings as shown in Figure 4.

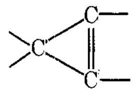
**D2. Method for Odd Number of Total-Atoms.** With an odd number of total atoms, no structures can be generated with a bond centroid. Only atom centroids are possible.[10,21] However, it is possible for structures to be built with a bivalent atom at the centroid. Thus the procedure outlined in category B above is followed, in this case also allowing a bivalent atom as the centroid.

**D3. Generation of Radicals.** The goal of this procedure is to generate all radicals from a list of atoms and ring-superatoms. A radical is defined to be an atom or superatom with a single free valence. When a composition of atoms and ring-superatoms is presented, from which radicals are to be constructed, two special cases are recognized.

**Special Case 1. Only One Atom in List of Atoms.** When only one atom which is not a ring-superatom is in the list, only one radical is possible. For example, with one C, the radical $CH_3$ is the only possibility.

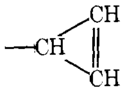
**Special Case 2. Only One Ring-Superatom in List of Ring-Superatoms.** In this case, depending upon the symmetry of the ring-superatom, several radicals may be possible. This is determined by labeling the free valences of the ring-superatom with one label of a special type, a "radical valence."

Example: A list of ring-superatoms consists of one ring superatom, **16**

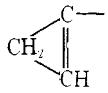


**16**

Two radicals result from labeling with one radical valence



**17**



**18**

**General Case.** Radicals have uniquely defined centroids as well.[10,21] The centroid is always an atom of valence 2 or higher. The steps for construction of radicals are as follows.

**Step 1. Selection of Atom Centroid.** Any bivalent or higher valent atom or ring-superatom is a valid candidate to be the centroid of a radical. Thus, for example, for the composition -C≡C- and >C=C< (see part 1a in Figure 3), both are valid centroids (Figure 5).

**Step 2. Partition the Rest of the Atoms.** The atom chosen for the centroid is removed from the list of superatoms. One of the valences of the centroid is to remain free (the radical valence). Therefore, the rest of the atoms in the list are partitioned into less than or equal to (valence of centroid − 1) parts. Of course, each part should satisfy the restriction on univalents (eq 7), but for constructing radicals there is no restriction on the size of the parts.

**Step 3. Form Radicals from Each Part.** The procedure to construct radicals is freshly invoked on each part thus generating radicals. Each part in Figure 5 gives rise to only one radical, each arising from special case 2.

**Step 4. Combine Radicals in Each Part.** For the example in Figure 5, each part yields only one radical. In a more general situation, where the rest of the list of superatoms after selection of a centroid is partitioned into several parts, and where each part yields several radicals, the radicals are combined to determine all unique combinations of radicals.

**Step 5. Label Central Atom with Radicals.** If the center is an atom (not a ring-superatom), then each unique combination defines a single unique molecule.

If the center is a ring-superatom, the radicals are determined by labeling the center with a set of labels which includes (i) the radicals, (ii) a leading radical-valence, and (iii) an adequate number of hydrogens to make up the remaining free valences of the ring-superatom. One selection of center gives one radical and the other gives two more, to complete a list of three radicals for the example chosen (Figure 5).

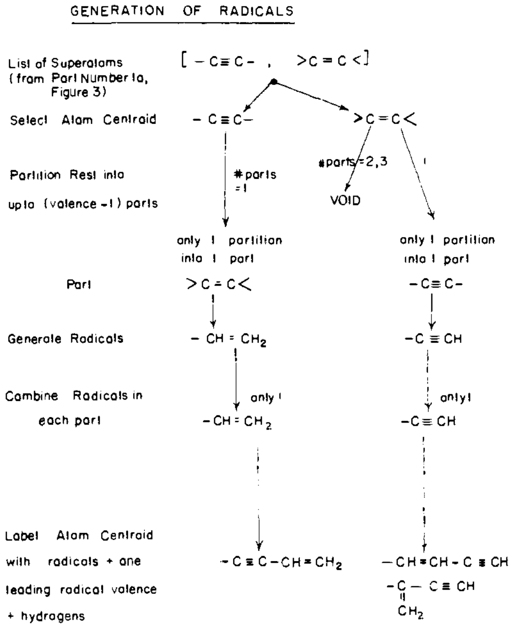

GENERATION OF RADICALS
List of Superatoms (from Part Number 1a, Figure 3)
[ −C≡C− , >C=C< ]
Select Atom Centroid
−C≡C−
>C=C<
Partition Rest into upto (valence −1) parts
#parts =1
#parts=2,3
VOID
1
only 1 partition into 1 part
only 1 partition into 1 part
Part
>C=C<
−C≡C−
Generate Radicals
−CH=CH2
−C≡CH
Combine Radicals in each part
only 1
only1
−CH=CH2
−C≡CH
Label Atom Centroid with radicals + one leading radical valence + hydrogens
−C≡C−CH=CH2
−CH=CH−C≡CH
−C−C≡CH
CH2

**Figure 5.** Outline of the method for generation of radicals which are eventually combined by the acyclic generator to yield final structures.

## Summary

For the example chosen to illustrate the operation of the acyclic generator, 12 isomers are generated; nine are shown in Figure 3 and three in Figure 4.

## References and Notes

(1) Part XI: R. E. Carhart and C. Djerassi, *J. Chem. Soc., Perkin Trans. 2*, 1753 (1973).

(2) Financial support for this work was provided by the National Institutes of Health (RR 00612-03) and the Advanced Research Projects Agency (SD-183).

(3) J. Lederberg, G. L. Sutherland, B. G. Buchanan, E. A. Feigenbaum, A. V. Robertson, A. M. Duffield, and C. Djerassi, *J. Amer. Chem. Soc.*, **91**, 2973 (1969).

(4) Y. M. Sheikh, A. Buchs, A. B. Delfino, G. Schroll, A. M. Duffield, C. Djerassi, B. G. Buchanan, G. L. Sutherland, E. A. Feigenbaum, and J. Lederberg, *Org. Mass Spectrom.*, **4**, 493 (1970).

(5) See, for example, A. C. Lunn and J. K. Senior, *J. Phys. Chem.*, **33**, 1027 (1929), and references cited therein.

(6) (a) G. Polya, *C. R. Acad. Sci.*, **201**, 1167 (1935); (b) *Helv. Chim. Acta*, **19**, 22 (1936); (c) *Z. Krystallogr.*, **92**, 415 (1936); (d) *Acta Math.*, **68**, 145 (1937).

(7) (a) T. L. Hill, *J. Phys. Chem.*, **47**, 253 (1943); (b) *ibid.*, 47, 413 (1943); (c) *J. Chem. Phys.*, **11**, 294 (1943).

(8) W. J. Taylor, *J. Chem. Phys.*, **11**, 532 (1943).

(9) (a) A. T. Balaban and F. Harary, *Rev. Roum. Chim.*, **12**, 1511 (1967); (b) *ibid.*, **11**, 1097 (1966); *Erratum, ibid.*, **12**, 103 (1967); (c) *ibid.*, **17**, 865 (1972); (d) *ibid.*, **18**, 635 (1973), and additional references cited therein.

(10) J. Lederberg, "DENDRAL-64, Part I. Notational Algorithm for Tree Structures," NASA Star No. N65-13158, NASA CR-57029; "Part II. Topology of Cyclic Graphs," NASA Star No. N66-14074, NASA CR-68898; "Part III. Complete Chemical Graphs: Embedding Rings in Trees," NASA Star No. N71-76061, NASA CR-123176.

(11) It is assumed that structures are completely connected by chemical bonds; thus catenates and threaded structures are viewed as consisting of separate molecules. It is further assumed that an atom with a given name has a constant valence. Cases involving, for example, charged or radical species or modified valence (*e.g.*, the nitrogen in nitro groups and *N*-oxides) can be treated separately by giving these species special names.

(12) A free valence is a bond with an unspecified terminus. Any substructure, cyclic or not, may be treated as a superatom; however, the term, in this paper, Is generally restricted to cyclic (termed ring-) superatoms.

(13) L. M. Masinter, N. S. Sridharan, R. E. Carhart, and D. H. Smith, *J. Amer. Chem. Soc.*, **96**, 7714 (1974).

(14) (a) H. Brown, L. M. Masinter, and L. Hjelmeland, *Discrete Math.*, **7**, 1 (1974); (b) Stanford Computer Science Memo STAN-CS-72-0318.

(15) (a) H. Brown and L. M. Masinter, *Discrete Math.*, **8**, 227 (1974); (b) Stanford Computer Science Memo STAN-CS-73-0361.

(16) Chemists are more familiar with terms such as rings or ring systems.

The term two-connected is used here in conjunction with ring-superatoms for a more precise description. For example, biphenyl may be viewed as a single ring system or two rings depending on the chemical context. In this work, however, biphenyl consists of two ring-superatoms (two phenyl rings) linked by a single bond.

(17) Use of the term degree with reference to the degree list refers to the number of bonds other than free valences, with double bonds being counted twice. A free valence may or may not eventually be attached to a hydrogen atom in the final structure.

(18) (a) N. S. Sridharan, Stanford Computer Science Memo STAN-CS-73-381, 1973; *(b) Stanford Computer Science Memo STAN-CS-*74-404, 1974; (c) L. M. Masinter, unpublished results.

(19) R. S. Cahn, C. K. Ingold, and V. Prelog, *Angew. Chem., Int. Ed. Engl.*, **5**, 385 (1966).

(20) (a) K. B. Wiberg and G. J. Burgmaier, *J. Amer. Chem. Soc.*, **94**, 7396 (1972); (b) K. B. Wiberg, G. J. Burgmaier, K. Shen, S. J. LaPlaca, W. C. Hamilton, and M. D. Newton, *J. Amer. Chem. Soc.*, **94**, 7402 (1972).

(21) B. G. Buchanan, A. M. Duffield, and A. V. Robertson in "Mass Spectrometry, Techniques and Applications," G. W. A. Milne, Ed., Wiley, New York, N.Y., 1971, p 121.

(22) (a) L. A. Paquette, S. Kirschner, and J. R. Malpass, *J. Amer. Chem. Soc.*, **92**, 4330 (1970); (b) S. Masamune and N. Darby, *Accounts Chem. Res.*, **5**, 272 (1972); (c) E. L. Allred and B. R. Beck, *J. Amer. Chem. Soc.*, **95**, 2393 (1973); (d) L. T. Scott and M. Jones, Jr., *Chem. Rev.*, **72**, 181 (1972).

(23) See D. G. Bobrow and B. Raphael, "New Programming Languages for AI Research," Report No. CSL-73-2, 1973, and W. Teitelman, "INTERLISP Reference Manual," Bolt Beranek and Newman and Xerox Corp., 1974, both available from Xerox PARC, 3180 Porter Dr., Palo Alto, California 94304.

(24) I. Ugi, D. Marquarding, H. Klusacek, G. Gokel, and P. Gillespie, *Angew. Chem., Int. Ed. Engl.*, **9**, 703 (1970).

(25) P. Gillespie, P. Hoffman, H. Klusacek, D. Marquarding, S. Pfohl, F. Ramirez, E. A. Tsolis, and I. Ugi, *Angew. Chem., Int. Ed. Engl.*, **10**, 687 (1971).

(26) (a) W. G. Klemperer, *J. Amer. Chem. Soc.*, **94**, 6940 (1972); (b) *ibid.*, **94**, 8360 (1972); (c) *ibid.*, **95**, 380 (1973); (d) *ibid.*, **95**, 2105 (1973).

(27) The form of eq 7 results from the fact that the number of univalents ($a_i$) cannot exceed the number of free valences necessary to connect the superatoms, leaving one valence free for the radical valence.

(28) NOTE ADDED IN PROOF. Since the acceptance of this manuscript major steps have been taken to effect a partial solution to the problem of constraints. The resulting program is also available *via* SUMEX (see Experimental Section).

# Applications of Artificial Intelligence for Chemical Inference. XIII. Labeling of Objects Having Symmetry[1,2]

**L. M. Masinter, N. S. Sridharan, R. E. Carhart, and D. H. Smith***

*Contribution from the Departments of Computer Science and Chemistry, Stanford University, Stanford, California 94305.*

*Received March 22, 1974*

**Abstract:** An algorithm for finding a complete set of nonequivalent labelings of a symmetric object and applications of the algorithm to problems in chemistry are presented.

Combinatorial problems which deal with finding a complete set of nonisomorphic objects under various constraints and based upon various concepts of isomorphism occur in many phases of chemistry. Solutions to some have been achieved using sophisticated applications of graph theory and group theory.[3,4] Perhaps the most common such problem involves attaching, in all unique ways, a fixed set of ligands to a given molecular skeleton.[5,6] This can be viewed as a *labeling* problem, in which all distinct assignments of a given set of labels to the parts of a symmetric object are sought. In developing the concepts set forth by Lederberg,[7] we have found that the labeling problem is common to many aspects of the generation of acyclic and cyclic isomers (see accompanying paper).[2] Herein, our solution to the labeling problem is presented.

Part A of this paper may be read as a brief tutorial on the nature of the problem and an introduction to the terminology found in more technical treatments. Part B is a textual description of a method for the solution of this type of problem. Part C is a summary of the procedure in a more algorithmic form; an even more formal description and a proof of correctness is available elsewhere.[8] In part D, certain generalizations of the basic algorithm are presented. Finally, in part E, a sample application of the method to a complex isomerism problem in organic chemistry is outlined.

The algorithm described here represents a concrete procedure for the solution of problems which previously were solvable only *via* "intuition." An intuitive approach to labeling problems is satisfactory for small cases, but can easily break down when applied to more complex problems, *e.g.*, the adamantane example in part D. It has been known how to compute the number of solutions,[4b,9] but an efficient method of actually constructing the solutions has not previously been published;[10] certainly the latter information is of substantially greater use to a chemist, as he is interested in the identity of each structure.

A computerized version of the labeling algorithm has been coded in the INTERLISP language. It is imbedded within the structure generator described in the accompanying paper,[2] and a listing of the source text of the program will be provided by the authors upon request. The program itself is available for use as described in the Experimental Section of ref 2.

The labeling program has been tested extensively using cases for which numbers of solutions have been published.[4b,9c] In each case, the program was used to construct the complete set of labelings, the solutions were counted, and the count was compared with the published value. No discrepancies have been found except in the largest case checked by the program of the node labeling of naphthalene (six labels of one type, two of a second, and two of a third). In this case, Balaban and Harary[4b] obtained a count of 329 solutions, while we obtained 330; further investigation showed that the expansion of the "generating function"[4b] for naphthalene was incorrect in this and at least one higher term, and that 330 was in fact the correct figure.

## Part A. Definitions

The object to be labeled may correspond to any finite collection of discrete parts (*e.g.*, faces of a cube or atoms of a molecule), as long as the symmetries of the object can be described as rearrangements of the parts which leave the object unchanged. The labels may be any properties or entities which can be associated with the parts (*e.g.*, colors, ligands, isotopic "tags"), as long as the labels themselves