Canonical Indexing and Constructive Enumeration of Molecular Graphs

VLADIMÍR KVASNIČKA* and JIŘÍ POSPÍCHAL

Department of Mathematics, Slovak Technical University, 81237 Bratislava, Czechoslovakia

Received July 21, 1989

A canonical indexing of molecular graphs based on the maximal digital code corresponding to the lower triangle part of the adjacency matrix is suggested. Graph-theoretical properties of this indexing make possible formulation of an exhaustive and nonredundant constructive enumeration of connected graphs with prescribed numbers of vertices and edges. The correctness of the concept is confirmed by a series of theorems.

INTRODUCTION

The problem of canonical indexing of molecular graphs, which is equivalent to the problem of isomorphism[1-3] between molecular graphs, is of great importance to chemical informatics as well as to graph-theoretical models of chemistry. In the early 1960s Morgan[4] suggested a method based on the concept of extended connectivities for unique canonical indexing of atoms in molecules. He also introduced the method of *cooperative indexing* of atoms (see the following section). This substantially reduced the total number of all possible indexings (in fact $n!$ problem) to a much smaller, more manageable number.

Many different methods[5-18] for canonical indexing of molecular graphs have been suggested. Some of them are plagued by the lack of rigorous proof that the indexing produced correctly solves the problem of isomorphism (i.e., an analogue of theorem 3). A singular position in the studies of canonical indexing is occupied by approaches[3,5,6,15,16,18] based on the production of minimal/maximal codes constructed from the lower triangle/upper triangle/whole adjacency matrix. All these approaches are conceptually simple and solve the problem of graph isomorphism correctly.

We choose from the above six possibilities the approach of canonical indexing based on the maximal code produced by the lower triangle part of the adjacency matrix. The main reason for this choice is the suitable graph-theoretical properties of the canonical indexing that is used. They enable us to formulate an effective backtrack-searching algorithm for constructive enumeration of all possible connected graphs with the prescribed number of vertices and edges. Further constraints or generalizations of the method could be simply implanted. In other aspects, the chosen canonical indexing has neither advantages nor disadvantages in comparison with canonical indexings based on maximal codes produced by the upper triangle adjacency matrix or the whole adjacency matrix.[6,18]

The problem of constructive enumeration of molecular graphs was successfully solved only for acyclic molecules.[19-21] Use was made of the well-known fact that for acyclic connected graphs (trees) the notion of *centroid*[22] is uniquely determined. The situation for cyclic molecular graphs is much more difficult. There does not exist a general canonical indexing that would solve the problem of graph isomorphism better than in a nonpolynomial time. Therefore, the constructive enumerations of molecular graphs are usually solved[23-26] by making use of heuristics which more or less effectively overcome the aforementioned theoretical difficulty. The approaches[15-17] that do not use any heuristic are usually of a "brute-force" type. Such approaches involve a huge number of isomorphism checks to assess whether a currently constructed graph was already constructed in the previous steps.

A novel approach surmounting these problems was suggested by Faradzhev et al.[26] in the form of an effective algorithm for the nonredundant and exhaustive constructive enumeration of graphs with prescribed distribution of vertex valences. The method is based on the canonical indexing employing the maximal code produced by whole adjacency matrix. It uses matrix formalism, and graph theory notions and concepts are applied only marginally.

The main requirements for constructive enumeration are as follows: The algorithm should be exhaustive, nonredundant, and efficient. The main problem of most constructive enumeration algorithms lies in production of a great number of redundant solutions, which have to be found and excluded. To do that, it is necessary to determine whether the generated solution is not isomorphic with previously generated solutions; it includes the necessity of some kind of canonical indexing. This process of canonical indexing and subsequent searching for a redundant solution is very time-consuming. Algorithms that produce less redundancy are generally better. In our approach based on a backtrack search algorithm, most potentially redundant solutions are excluded at the higher level of the search tree. Moreover, there is no necessity to search for redundant solutions; only canonically indexed solutions are taken into account.

The purpose of this paper is to elaborate a graph-theoretical method suitable for the constructive enumeration of molecular graphs which employs the canonical indexing based on the maximal code produced by the lower triangle part of adjacency matrix. This graph-theoretical approach has properties that considerably facilitate its effective implementation on computers.

BASIC CONCEPTS

Let $\mathbf{G}$ be a *graph*[22] with a nonempty *vertex set* $V(\mathbf{G})$ and an *edge set* $E(\mathbf{G})$. In our forthcoming considerations we shall always assume that graph $\mathbf{G}$ is connected and that it does not contain multiple edges and loops. An edge $e \in E(\mathbf{G})$ incident with two distinct vertices $v,v' \in V(\mathbf{G})$ is denoted $\{v,v'\}$. An *indexing* of the graph $\mathbf{G}$, composed of $n = |V(\mathbf{G})|$ vertices, consists of a one-to-one mapping

$$\phi: V(\mathbf{G}) \rightarrow \{1, 2, ..., n\} \quad (1)$$

An integer $\phi(v) \in \{1, 2, ..., n\}$ assigned to a vertex $v \in V(\mathbf{G})$ is called the *index* of the vertex v. Graph $\mathbf{G}$ together with the mapping ϕ is called an *indexed graph*, and it is denoted $\mathbf{G}_\phi$.

In general, there exist $n!$ distinct indexings of $\mathbf{G}$; if this graph has a group $\Gamma(\mathbf{G})$ of automorphisms, then the total number of nonequivalent indexings is equal to $n!/|\Gamma(\mathbf{G})|$. The de-

0095-2338/90/1630-0099$02.50/0

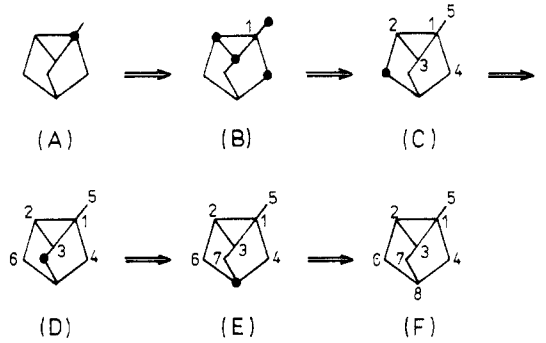

Figure 1. Illustrative example of cooperative indexing of a graph. The heavy dots correspond to those vertices that are still nonindexed and are "hot" candidates for indexing in the next step. In graph A the heavy dot is selected in an arbitrary manner. The indexing in graphs B–F is made in such a way that we index in the same manner as for the heavy dots of the previous graph. Graph F is cooperatively indexed.

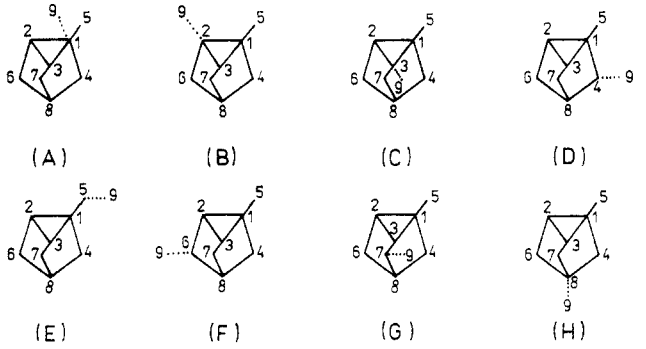

Figure 2. Illustrative example of finding the saturated and/or unsaturated vertices for the cooperatively indexed graph F in Figure 1. Graphs A–C are not cooperatively indexed, whereas graphs D–H are cooperatively indexed. Hence, the vertices indexed by 1–3 are saturated, and the vertices indexed by 4–8 are unsaturated; i.e., $V_{sat}(\mathbf{G}_\phi) = \{1, 2, 3\}$ and $V_{unsat}(\mathbf{G}_\phi) = \{4, 5, 6, 7, 8\}$. The virtual vertex indexed 9 serves for better understanding of the check whether the given vertex (connected by dashed line with virtual vertex) is saturated or not.

nominator $|\Gamma(\mathbf{G})|$ corresponds to the order of group $\Gamma(\mathbf{G})$ (often called the *symmetry number*[27] of graph $\mathbf{G}$). A set of all possible indexings ϕ of graph $\mathbf{G}$ is denoted $\Phi(\mathbf{G})$

$$\Phi(\mathbf{G}) = \{\phi;\ \phi \text{ is an indexing of } \mathbf{G}\} \tag{2}$$

To substantially restrict the enormous number of indexings of graph $\mathbf{G}$, we use the so-called *cooperative indexing*, initially suggested by Morgan[4] for his famous canonical indexing of atoms in structural formulas. The cooperative indexing is determined by the following convention: An arbitrary vertex of $\mathbf{G}$ is indexed by 1. If this vertex is p-ternary (i.e., p vertices are adjacent with the vertex), then all adjacent vertices are indexed by 2, 3, ..., $p + 1$ in whatever combination. In the forthcoming step we select the vertex already indexed by 2, and all its adjacent still nonindexed vertices are indexed by $p + 2$, $p + 3$, ..., $p + q$ in whatever combination. In this way one proceeds indexing each successive vertex until all vertices are indexed. The process of cooperative indexing is illustrated in Figure 1.

The vertices of the cooperatively indexed graph $\mathbf{G}_\phi$ may be classified as follows: A vertex $v \in V(\mathbf{G}_\phi)$ is called *saturated* (*unsaturated*) if an appendage of an additional vertex $\tilde{v} \notin V(\mathbf{G}_\phi)$ indexed by $n + 1$ to the vertex v produces a new graph (composed of $n + 1$ vertices) which is *also* (*not*) cooperatively indexed (see Figure 2). One can easily show that the vertex set $V(\mathbf{G}_\phi)$ of a cooperatively indexed graph $\mathbf{G}_\phi$ can be divided into two disjoint subsets composed of saturated and unsaturated vertices, respectively.

$$V(\mathbf{G}_\phi) = V_{sat}(\mathbf{G}_\phi) \cup V_{unsat}(\mathbf{G}_\phi) \tag{3}$$

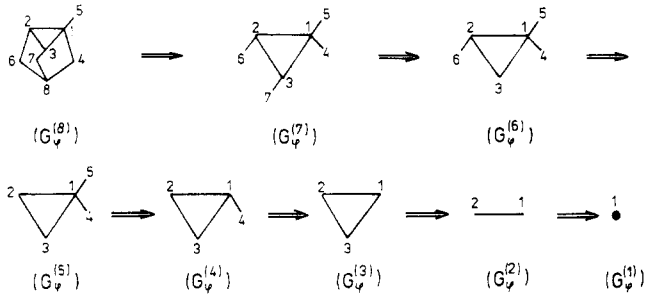

Figure 3. Construction of subgraphs $\mathbf{G}_\phi^{(p)}$ (for $1 \le p \le n - 1$) from the cooperatively indexed graph $\mathbf{G}_\phi$. This is an illustration of theorem 1; that is, if the initial graph was cooperatively indexed, then all its subgraphs $\mathbf{G}_\phi^{(p)}$ made by successive deleting of the highest indexed vertex are also cooperatively indexed.

Moreover, the subset $V_{unsat}(\mathbf{G}_\phi)$ is always nonempty, and it is composed of those vertices $V(\mathbf{G}_\phi)$ that are indexed by integers q, $q + 1$, ..., n, where $1 \le q \le n$; i.e.

$$V_{sat}(\mathbf{G}_\phi) = \{v \in V(\mathbf{G}_\phi);\ 1 \le \phi(v) \le q - 1\} \tag{4a}$$

$$V_{unsat}(\mathbf{G}_\phi) = \{v \in V(\mathbf{G}_\phi);\ q \le \phi(v) \le n\} \tag{4b}$$

A subgraph $\mathbf{G}_\phi^{(p)}$ (for $p = 1, 2, ..., n - 1$) of $\mathbf{G}_\phi$ is induced by a vertex subset $V(\mathbf{G}_\phi^{(p)}) \subset V(\mathbf{G}_\phi)$ determined by

$$V(\mathbf{G}_\phi^{(p)}) = \{v \in V(\mathbf{G}_\phi),\ 1 \le \phi(v) \le p\} \tag{5a}$$

$$|V(\mathbf{G}_\phi^{(p)})| = p \tag{5b}$$

This means that the subgraph $G_\phi^{(p)}$ contains p vertices of $\mathbf{G}_\phi$ indexed 1, 2, ..., p and the edges incident with these vertices. For an arbitrary pair of indices $i, j \in \{1, 2, ..., p\}$ we have

$$\{\phi^{-1}(i), \phi^{-1}(j)\} \in E(\mathbf{G}_\phi) \Leftrightarrow \{\phi^{-1}(i), \phi^{-1}(j)\} \in E(\mathbf{G}_\phi^{(p)}) \tag{6}$$

The concept of the subgraph $\mathbf{G}_\phi^{(p)}$ is illustrated in Figure 3.

Theorem 1. If $\mathbf{G}_\phi$ is a cooperatively indexed graph, then for each $1 \le p \le n$ the subgraph $\mathbf{G}_\phi^{(p)}$ is also cooperatively indexed.

The proof of this theorem is simple, following immediately from the definition of cooperative indexing.

Now we turn our attention to an inverse problem—how to generate from the cooperatively indexed graph $\mathbf{G}_\phi$ a larger graph, composed of $n + 1$ vertices (where $n = |V(\mathbf{G}_\phi)|$), which is also cooperatively indexed. A prototype of such a process was already outlined in the framework of the above determination of saturated/unsaturated vertices of the cooperatively indexed graph $\mathbf{G}_\phi$. Let $V_{unsat}(\mathbf{G}_\phi)$ be a subset composed of the unsaturated vertices of $\mathbf{G}_\phi$. We select a nonempty subset $V_{ext}(\mathbf{G}_\phi) \subset V_{unsat}(\mathbf{G}_\phi)$ composed of some preselected unsaturated vertices of $\mathbf{G}_\phi$. The *extension* of the graph $\mathbf{G}_\phi$ with respect to the subset $V_{ext}(\mathbf{G}_\phi)$ consists of the graph $\mathbf{G}_{ext,\phi}$ determined by

$$V(\mathbf{G}_{ext,\phi}) = V(\mathbf{G}_\phi) \cup \{\tilde{v}\} \tag{7a}$$

$$E(\mathbf{G}_{ext,\phi}) = E(\mathbf{G}_\phi) \cup \{\{v, \tilde{v}\};\ v \in V_{ext}(\mathbf{G}_\phi)\} \tag{7b}$$

where $\tilde{v}$ is an additional vertex indexed by $n + 1$. Loosely speaking, the extension $\mathbf{G}_{ext,\phi}$ is constructed from the original cooperatively indexed graph $\mathbf{G}_\phi$ in such a way that a new vertex $\tilde{v}$ indexed by $n + 1$ is connected by edges with all preselected unsaturated vertices from the subset $V_{ext}(\mathbf{G}_\phi)$ (see Figure 4). It implies, e.g., that the subgraph $\mathbf{G}_{ext,\phi}^{(n)}$ (when the vertex $\tilde{v}$ was deleted from the extension $\mathbf{G}_{ext,\phi}$) is identical with graph $\mathbf{G}_\phi$.

Theorem 2. If graph $\mathbf{G}_\phi$ is cooperatively indexed, then its extension $\mathbf{G}_{ext,\phi}$ is also cooperatively indexed.

The proof of this theorem suggests itself immediately as a direct consequence of the definition of extension of graph $\mathbf{G}_\phi$.

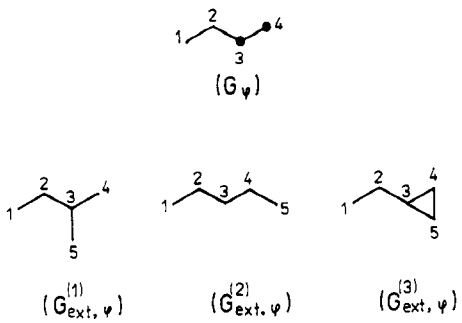

Figure 4. Construction of extensions from the cooperatively indexed graph G_ϕ; its unsaturated vertices are denoted by heavy dots. The extension of graph G_ϕ cannot include any edges {1, 5} or {2, 5} because vertices 1 and 2 are saturated.

CANONICAL INDEXING

An *adjacency matrix* $\mathbf{A}_\phi = (a_{ij})$ assigned to the indexed graph $\mathbf{G}_\phi$ is a symmetric matrix of the type (n, n).

$$a_{ij} = 1, \text{ for } \{\phi^{-1}(i), \phi^{-1}(j)\} \in E(\mathbf{G}_\phi)$$
$$= 0, \text{ for } \{\phi^{-1}(i), \phi^{-1}(j)\} \notin E(\mathbf{G}_\phi) \quad (8)$$

A *code* of the indexed graph $\mathbf{G}_\phi$, denoted $[\mathbf{G}_\phi]$, is a string composed of $n(n+1)/2$ digits of the lower triangle of the adjacency matrix $\mathbf{A}_\phi$.

$$[\mathbf{G}_\phi] = (a_{11}a_{21}a_{22}a_{31}a_{32}a_{33}\ldots a_{n1}\ldots a_{nn}) \quad (9)$$

We see that the code $[\mathbf{G}_\phi]$ unambiguously determines the adjacency matrix $\mathbf{A}_\phi$.

Two indexings ϕ and ϕ' of graph G are called equivalent ($\phi = \phi'$) if the corresponding adjacency matrices are identical.

$$\phi = \phi' \Leftrightarrow \mathbf{A}_\phi = \mathbf{A}_{\phi'} \Leftrightarrow [\mathbf{G}_\phi] = [\mathbf{G}_{\phi'}] \quad (10)$$

The codes are mutually ordered according to the lexicographical (or simply numerical) relation between these strings. In the same way we relate also indexings ϕ' of graph G.

$$\phi \leq \phi' \Leftrightarrow [\mathbf{G}_\phi] \leq [\mathbf{G}_{\phi'}] \quad (11)$$

We will call a *canonical indexinng* ϕ_{can} of graph G such an indexing of $\Phi(\mathbf{G})$ that satisfies

$$\forall\, \phi \in \Phi(\mathbf{G}) : \phi \leq \phi_{can} \quad (12)$$

This means that the canonical indexing ϕ_{can} provides the *maximal code* of graph G. Recently, Hendrickson et al. have published[6] a method of canonical indexing of molecular graphs based on the maximal code assigned to the upper triangle of the adjacency matrix. It is possible to demonstrate simply (see Figure 5) that the present approach of canonical indexing (based on the lower triangle of the adjacency matrix) is not equivalent to that of Hendrickson et al. In the framework of our approach special properties of the used canonical indexing will appear (e.g., see theorem 5 in the following section).

A subset of $\Phi(\mathbf{G})$ composed of all possible canonical indexings of graph G will be denoted $\Phi_{can}(\mathbf{G})$. The subset is composed of at least one indexing ϕ which provides the maximal code of graph G; if the subset $\phi_{can}(\mathbf{G})$ contains more than one mapping, then graph G has a nontrivial automorphism (one-to-one mapping)

$$\omega: V(\mathbf{G}) \rightarrow V(\mathbf{G}) \quad (13)$$

which conserves the adjacency of vertices in G

$$[v, v'] \in E(\mathbf{G}) \Leftrightarrow \{\omega(v), \omega(v')\} \in E(\mathbf{G}) \quad (14)$$

Let us study two canonical indexings $\phi, \phi' \in \Phi_{can}(\mathbf{G})$. The corresponding automorphism (eq 13) is determined as a composition.

$$\omega = \phi \circ \phi'^{-1} \quad (15)$$

This means that the group of automorphisms $\Gamma(\mathbf{G})$ may be constructed from the canonical indexings as follows:

$$\Gamma(\mathbf{G}) = \{\omega = \phi \circ \phi'^{-1}; \text{ for a fixed } \phi \in \Phi_{can}(\mathbf{G}) \text{ and all } \phi' \in \Phi_{can}(\mathbf{G})\} \quad (16)$$

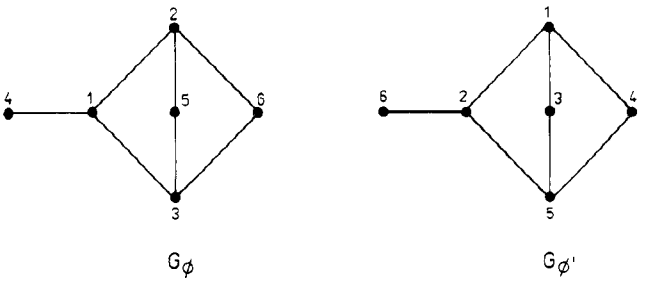

Figure 5. Illustrative example of the fact that "lower triangle" (right graph) and "upper triangle" (left graph) approaches to the construction of maximal code offer different canonical indexings.

The indexed graph $\mathbf{G}_\phi$ with the canonical indexing ϕ [i.e., $\phi \in \Phi_{can}(\mathbf{G})$] will be called the *canonically indexed graph*. The following three theorems are satisfied for canonically indexed graphs.

Theorem 3. Two graphs G and G' are isomorphic ($\mathbf{G} \approx \mathbf{G}'$) iff their canonical codes are identical.

Theorem 4. If graph $\mathbf{G}_\phi$ is canonically indexed, then the indexing is cooperative.

Theorem 5. If graph $\mathbf{G}_\phi$ is canonically indexed, then for each $1 \leq p \leq n$ the subgraph $\mathbf{G}_\phi^{(p)} \subseteq \mathbf{G}_\phi$ is canonically indexed.

To prove theorem 3, we have to remember that an alternative definition of isomorphism between two graphs G and G' may be done in such a way that there exist indexed graphs $\mathbf{G}_\phi$ and $\mathbf{G}_{\phi'}$, with identical adjacency matrices. In other words, their maximal codes must also be identical. This theorem serves as a well-founded theoretical basis for finding an isomorphism between graphs.

The proof of theorem 4 is very simple. Since the canonical indexing is defined in such a way that it produces the maximal code constructed from the lower triangle of the adjacency matrix, the vertices which are adjacent to a labeled vertex and are still unlabeled by indices should be indexed successively by the lowest possible integers. In the opposite case the adjacency matrix does not produce a maximal code.

To prove theorem 5, let us assume that a subgraph $\mathbf{G}_\phi^{(p)}$ (for $1 \leq p \leq n$) is not canonically indexed. This means that there exists a restricted mapping $\tilde{\phi}$: $V(\mathbf{G}_\phi^{(p)}) \rightarrow \{1, 2, \ldots, p\}$ that produces a larger code than the original mapping ϕ restricted to the vertex subset $\mathbf{G}_\phi^{(p)}$. Since this property is also transferred to larger induced subgraphs of $\mathbf{G}_\phi$, there must exist a mapping (eq 1) that fully incorporates the restricted mapping $\tilde{\phi}$ and gives a larger code than the one produced by the original mapping ϕ. Hence, graph $\mathbf{G}_\phi$ could not have been canonically indexed.

Theorems 4 and 5 represent very important *necessary* conditions for canonical indexing of graphs. In particular, the finding of canonical indexing may be restricted entirely to the cooperative indexings (see theorem 4) of the graph. Theorem 5 provides an effective theoretical tool for constructive enumeration of graphs (see the following section).

For an effective construction of canonical indexing it is very important to know which vertices of graph G may potentially be indexed by 1. This problem is solved by the following two theorems.

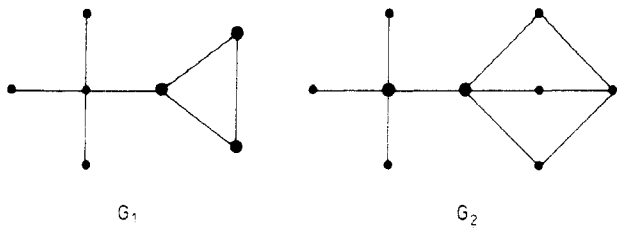

Figure 6. Examples of application of theorems 6 (left graph) and 7 (right graph) for selecting $V_{prior}(G)$; the vertices belonging to this vertex subset are denoted by heavy dots. The left graph contains maximal clique identical with the triangle. The right diagram does not contain triangles; therefore, its set $V_{prior}(G)$ is composed only of two vertices of the valence equal to 4.

Theorem 6. If G_ϕ is a canonically indexed graph, then its vertex indexed by 1 belongs to a vertex subset $V_{prior}(G) \subseteq V(G)$ composed of the vertices that induce a maximal clique (or cliques) in G.

Theorem 7. If G_ϕ is a canonically indexed graph and it does not contain a triangle (i.e., maximal cliques are only single edges), then the vertex indexed by 1 belongs to the vertex subset $V_{prior}(G) \subset V(G)$ composed of vertices which are of maximal valence.

Both these theorems are illustrated in Figure 6. Theorem 6 follows immediately from the fact that if a vertex belonging to a maximal clique is indexed by 1, then the cooperative indexing generates in the adjacency matrix A_ϕ a submatrix placed in its left top corner occupied merely by unit off-diagonal entries. Similarly (theorem 7), if the graph does not contain triangles, then a vertex of maximal valence should be indexed by 1. This choice and cooperative indexing ensure that in the first column of the adjacency matrix A_ϕ the maximal possible number of unit entries, going from top to bottom, is generated. Of course, the choice of a vertex potentially indexed by 1 based on theorem 6 is of higher priority than an alternative choice based on theorem 7; i.e., this second possibility may be used only when graph G does not contain triangles.

A generalization of the suggested method of canonical indexing for pseudographs (graphs with multiple edges and loops) and multigraphs (graphs with multiple edges but without loops) may be carried out simply. For pseudographs and multigraphs analogues of theorems 6 and 7 are straightforwardly formulated.

Theorem 8. If G_ϕ is a canonically indexed pseudograph, then the vertex indexed by 1 belongs to the vertex subset $V_{prior}(G) \subset V(G)$ composed of the vertices incident with the loops of largest multiplicity.

Theorem 9. If G_ϕ is a canonically indexed multigraph, then the vertex indexed by 1 belongs to the vertex subset $V_{prior}(G) \subset V(G)$ composed of vertices incident with the edges of largest multiplicity.

Since the notion of a pseudograph is more general than the notion of a multigraph (and this again is more general than the notion of a graph), theorem 8 is of a higher priority than theorem 9. That is, the vertices of highest priority in algorithm 1 (see rows 1 and 12) are determined for pseudographs according to theorem 8. If the indexed graph is only a multigraph, then these vertices are selected by theorem 9. Finally, for "simple" graphs (which are not pseudographs or multigraphs) the vertices of highest priority are determined by either theorem 6 or theorem 7, depending on the existence of a triangle in the graph.

A backtrack searching[28] for the canonical indexing may be considerably accelerated by the following simple consequence of theorem 5: Let us study two indexed graphs G_ϕ and $G_{\phi'}$; for a fixed $1 \leq p \leq n$ their induced subgraphs are denoted $G_\phi^{(p)}$ and $G_{\phi'}^{(p)}$, respectively. We assign to these subgraphs the adjacency matrices $A_\phi^{(p)}$ and $A_{\phi'}^{(p)}$, respectively. Then the inequality $[G_\phi^{(p)}] < [G_{\phi'}^{(p)}]$ implies that graph G_ϕ is not canonically indexed. This simple condition will be advantageously used as a *branch and bound* test in our backtrack searching of the canonical indexing of graph G. The search tree of finding the canonical indexing is substantially pruned by this simple property of canonical indexing. We use a substring assigned to the mentioned submatrix of the adjacency matrix for each level of backtrack searching. If the substring is lexicographically (i.e., numerically) smaller than its previous value for the given level of search, then the procedure is stopped and directed back to the higher level of searching.

We present here the suggested algorithm in a symbolic Pascal-like form.

Algorithm 1

```
1.  i:=1; U_1:=A_1:=set of vertices of highest priority;
    RESTORE(1);
2.  repeat if |U_i| > 0 then
3.  begin φ^-1(i):=any element of U_i;
4.    remove the vertex φ^-1(i) from U_i and mark it as tried
    for level i;
5.    V:=set of vertices that are adjacent with φ^-1(i) and
    are not marked as tried for level i;
6.    if |V|>0 then
7.    begin mark all vertices of V as tried for level i;
8.      for k:=n_i + 1 to n_i + |V| do A_k:=V; n_i:=n_i + |V|;
9.    end else
10.   if i=n_i and i<n then
11.   begin n_i:=n_i + 1; k:=n_i;
12.     A_k:=set of highest priority vertices that are not
    marked as tried;
13.   end;
14.   if TEST(i) then
15.   begin if i=n then STORE else
16.     begin i:=i+1; U_i:=A_i;
17.       remove from U_i all vertices φ^-1(1), φ^-1(2),
    ...φ^-1(i-1);
18.       RESTORE(i);
19.     end;
20.   end else RESTORE(i);
21. end else
22. begin i:=i-1; RESTORE(i) end;
23. until i=0;
```

Procedure RESTORE(i) redefines the value of n_i and the marking of vertices to be "tried" or "untried" for level i on the basis of level $i-1$. If $i = 1$, then we put $n_1 = 1$ and all vertices are marked as untried. Boolean function TEST(i) tests whether the code $[G_\phi^{(i)}]$, for level i and temporary mapping ϕ, is lexicographically (numerically) greater than or equal to an older code for the same level i. If so, then the value of the function is "true"; in the opposite case its value is "false". In the case when the temporary code is greater than an older code, the older code is refined by the former one. The procedure STORE takes note of the mapping ϕ. These noted mappings are indexed by a counter N_mapping (this integer variable describes the serial number of mappings already noted). If an actual mapping provides greater code than the previous mappings already noted, then the counter N_mapping is refined by N_mapping = 1. Rows 11–13 are activated if the indexed graph is disconnected.

CONSTRUCTIVE ENUMERATION

The theory of canonical indexing elaborated in the previous section represents a very effective tool for the constructive enumeration of graphs. Let us consider a canonically indexed graph G_ϕ composed of n vertices; its induced subgraph $G_\phi^{(n-1)}$ is also canonically indexed (see theorem 5). This approach

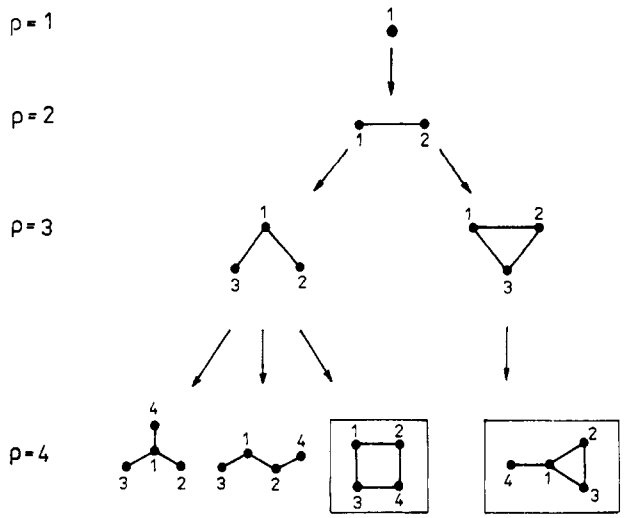

Figure 7. Construction of all canonically indexed connected graphs with four vertices and four edges. The graphs situated at pth level are canonically indexed extensions of the corresponding "parent" graphs from the level $p - 1$.

may be reversed; we have then a canonically indexed graph $\mathbf{G}_\phi$ and would like to construct all canonically indexed graphs for which graph $\mathbf{G}_\phi$ is the induced subgraph formed from them by deleting the vertex with the highest index. Such a construction of the "successors" from the "predecessor" $\mathbf{G}_\phi$ is nothing less than an extension of the latter. Unfortunately, the produced extensions of $\mathbf{G}_\phi$ are not automatically canonically indexed. They are only cooperatively indexed, and their induced subgraphs (e.g., the original graph $\mathbf{G}_\phi$) are canonically indexed. Therefore, we have to check whether a produced extension is canonically indexed. If it is not, then this extension is rejected from our forthcoming considerations. These ideas are summarized by the following theorem.

Theorem 10. For a given canonically indexed graph $\mathbf{G}_\phi$ (composed of n vertices) all its canonically indexed extensions $\mathbf{G}_{\text{ext},\phi}^{(1)}$, $\mathbf{G}_{\text{ext},\phi}^{(2)}$, $\mathbf{G}_{\text{ext},\phi}^{(3)}$, ... (composed of $n + 1$ vertices) represent all possible canonically indexed nonisomorphic graphs which have as the induced subgraph (formed from them by deleting the vertex indexed by $n + 1$) the original graph $\mathbf{G}_\phi$.

This theorem enables us to devise a very simple and simultaneously effective method for constructive enumeration of all possible graphs with the prescribed number of vertices and edges. Its strongest asset lies in the fact that only nonisomorphic graphs are constructed; i.e., it is not necessary to check whether a produced graph was already constructed. According to theorem 10, an exhaustive constructive enumeration of graphs with prescribed number of vertices and edges may be formulated in a recurrent manner. We start from the simplest graph composed of one vertex indexed by 1. From this graph we construct its canonically indexed extension with two vertices. In general, for the pth step in which we have already constructed all canonically indexed graphs with $p - 1$ vertices, we shall construct all their canonically indexed extensions composed of p vertices. This simple recurrent procedure is stopped when p becomes equal to the prescribed number of vertices; from the produced graphs we then select those having the prescribed number of edges. To reduce the total number of produced graphs, the excess of edges in each step of the above procedure may be checked; graphs with a greater number of edges than its prescribed value are not extended in the forthcoming step. The basic principles of the method are illustrated in Figure 7, where a tree of construction of connected graphs with four vertices and four edges is presented. An algorithmic outline of the proposed method will be presented in the backtrack search form.

Algorithm 2

1. p:=1; $\mathcal{U}_1$:=set composed of the graph with one vertex indexed by 1;
2. **repeat if** $|\mathcal{U}_p|>0$ **then**
3. **begin** $\mathbf{G}_p$:=an arbitrary graph from $\mathcal{U}_p$;
4. $\mathcal{U}_p$:=$\mathcal{U}_p \setminus \{\mathbf{G}_p\}$
5. **if** $p=p_{\max} \wedge |E(\mathbf{G}_p)|=q_{\max}$ **then** write $(\mathbf{G}_p)$ **else**
6. **if** $p<p_{\max}$ **then**
7. **begin** p:=p+1;
8. $\mathcal{U}_p$:=set of all possible canonically indexed extensions of $\mathbf{G}_{p-1}$;
9. **end**
10. **end else** p:=p−1;
11. **until** p=0;

The constants $p_{\max}$ and $q_{\max}$ are prescribed numbers of vertices and edges, respectively, of the graphs to be constructed. The fourth line of the algorithm represents a statement of removing graph $\mathbf{G}_p$ from the set $\mathcal{U}_p$. The most critical part of the algorithm is the eighth line. It corresponds to construction of all canonically indexed extensions of graph $\mathbf{G}_{p-1}$ collected at the set $\mathcal{U}_p$. For all permissible subsets $V_{\text{ext}}(\mathbf{G}_{p-1})$ the corresponding extensions are constructed and simultaneously checked whether they are canonically indexed; if so, then they are accounted for in the set $\mathcal{U}_p$. The check of canonical indexing is carried out by algorithm 1.

APPLICATION: RECONSTRUCTION OF GRAPHS FROM TOPOLOGICAL INDICES

The method of constructive enumeration of graphs described in the previous section offers very simple and straightforward methods for a reconstruction of graphs from topological indices. In general, the *topological index*[30] $\chi(\mathbf{G})$ assigned to graph $\mathbf{G}$ is a real number

$$\chi: \mathcal{G} \rightarrow R \tag{17}$$

where $\mathcal{G}$ is a universe of connected graphs and R is the set of nonnegative real numbers; if graph $\mathbf{G}$ is composed of only one vertex, then we put $\chi(\mathbf{G}) = 0$. At present, many different topological indices have been defined and widely used for the correlations of molecular properties/activities vs their structure.[31] We shall focus our attention on the following two topological indices: (1) the Wiener topological index[32]

$$\chi_W(\mathbf{G}) = \tfrac{1}{2} \sum_{\substack{v,v' \in V(\mathbf{G}) \\ (v \neq v')}} d(v, v') \tag{18a}$$

where the summation runs over all distinct pairs of vertices $v, v' \in V(\mathbf{G})$ and $d(v, v')$ denotes the graph distance between the vertices v and v', and (2) the Randić topological index[33]

$$\chi_R(\mathbf{G}) = \sum_{e \in E(\mathbf{G})} [1/\sqrt{\text{val}(v)\ \text{val}(v')}] \tag{18b}$$

where the summation runs over all edges $e = [v, v'] \in E(\mathbf{G})$ and the symbols $\text{val}(v)$ and $\text{val}(v')$ denote the valences of the vertices v and v', respectively.

Both these topological indices belong to the most frequently used topological indices for structure vs property/activity correlations.[31] Therefore, it might be of interest to solve the inverse problem: Construct for a given value of the topological index all nonisomorphic graphs with topological indices equal to the prescribed value (alternatively, the resulting topological indices are from an interval centered around the prescribed value). This problem was initially solved by Skvorcova et al.,[34] Zefirov et al.,[35] and Gordeeva et al.[36] for the Randić and Wiener topological indices. In the forthcoming part of this section we describe an alternative method of solving the inverse problem in the framework of our constructive enumeration of graphs.

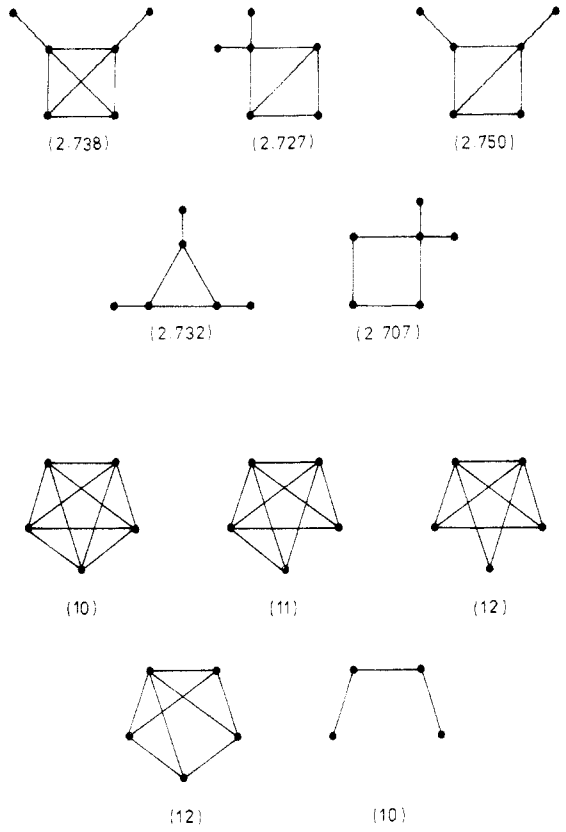

Figure 8. All graphs with Wiener and Randić topological indices ranged by $10 \leq \chi_W \leq 12$ and $2.7 \leq \chi_R \leq 2.75$, respectively.

Conjecture 1. For any canonically indexed graph $\mathbf{G}_\phi$ composed of $n \geq 2$ vertices the following two inequalities are satisfied

$$\chi_W(\mathbf{G}_\phi^{(n-1)}) < \chi_W(\mathbf{G}_\phi) \tag{19a}$$

$$\chi_R(\mathbf{G}_\phi^{(n-1)}) < \chi_R(\mathbf{G}_\phi) \tag{19b}$$

where $\mathbf{G}_\phi^{(n-1)}$ is the subgraph of $\mathbf{G}_\phi$ created from them by deleting the vertex indexed by n (or, in other words, graph $\mathbf{G}_\phi$ is an extension of graph $\mathbf{G}_\phi^{(n-1)}$).

Unfortunately, we did not succeed in proving this conjecture. For many particular cases its proof may be done, but for general graphs there are many obstacles in the theoretical considerations needed by the correct proof. We have successfully checked this conjecture, however, for all canonically indexed graphs up to nine vertices. It states that on going from a canonically indexed graph $\mathbf{G}_\phi^{(n-1)}$ to its extension $\mathbf{G}_\phi^{(n)} = \mathbf{G}_\phi$ the Wiener and the Randić topological indices are increased. This property may be simply implanted in a modified form of algorithm 2 for constructive enumeration of graphs. The searching tree of the algorithm is terminated if the current graph has a topological index greater than a prescribed maximal value.

Algorithm 3

1. p:=1; $\mathcal{U}_1$:=set composed of the graph with one vertex indexed by 1;
2. **repeat if** $|\mathcal{U}_p|>0$ **then**
3. **begin** $\mathbf{G}_p$:=an arbitrary graph from $\mathcal{U}_p$;
4. $\mathcal{U}_p$:=$\mathcal{U}_p \setminus [\mathbf{G}_p]$;
5. **if** $\chi_{min} \leq \chi(\mathbf{G}_p)$ **then** write $(\chi(\mathbf{G}_p), \mathbf{G}_p)$ **else**
6. **if** $p<p_{max}$ **then**
7. **begin** p:=p+1;
8. $\mathcal{U}_p$:=set of all possible canonically indexed extensions of $\mathbf{G}_{p-1}$ restricted by $\chi(\mathbf{G}) \leq \chi_{max}$;
9. **end**

Table I: Total Number of Connected Canonically Indexed Graphs with p Vertices and q Edges[a]

	p					
q	3	4	5	6	7	8
2	1 [<0.1]					
3	1 [<0.1]	2 [<0.1]				
4		2 [0.1]	3 [0.2]			
5		1 [0.1]	5 [0.3]	6 [0.8]		
6		1 [0.1]	5 [0.4]	13 [1.3]	11 [4.0]	
7			4 [0.6]	19 [2.3]	33 [5.4]	23 [25.6]
8			2 [0.5]	22 [3.4]	67 [10.7]	89 [24.7]
9			1 [0.4]	20 [4.2]	107 [18.5]	236 [51.4]
10			1 [0.4]	14 [4.2]	132 [27.6]	486 [100.2]
11				9 [4.5]	138 [36.4]	814 [174.7]
12				5 [3.9]	126 [44.8]	1169 [280.8]
13				2 [3.1]	95 [47.5]	1454 [418.3]
14				1 [2.9]	64 [47.0]	1579 [556.1]
15				1 [3.5]	40 [44.9]	1515 [682.0]

[a] In brackets are given computing times in seconds.

10. **end else** p:=p−1;
11. **until** p=0;

The constant p_{max} corresponds to the maximal number of vertices of the graphs to be constructed with the topological index (Wiener or Randić) ranged by χ_{min} and χ_{max}. At the eighth line are constructed only those canonically indexed extensions of graph $\mathbf{G}_p$ that have the topological index equal to or smaller than the upper bound χ_{max}. In Figure 8 are displayed all graphs with Wiener topological index ranged by $10 \leq \chi_W \leq 12$ and all graphs with Randić topological index ranged by $2.7 \leq \chi_R \leq 2.75$.

SUMMARY

The main purposes of the present paper have been (1) to specify the proper graph-theoretical formalism of canonical indexing for nonredundant and exhaustive constructive enumeration of graphs and (2) to formulate the principal ideas of this enumeration. The initial formulation was done only for simple connected graphs (i.e., multiedges and loops are not permitted). But, as was already demonstrated, the formalism can easily be generalized to include multigraphs or pseudographs as well, so that no new concepts are required. The resulting constructive enumeration shares many common features with the constructive enumeration of connected acyclic graphs (trees) initially elaborated by Joshua Lederberg[19] for purposes of the DENDRAL project.[22] In particular, both methods are based on a proper canonical indexing and produce only nonredundant canonically indexed graphs. This means that in both methods one does not need to check whether the produced graphs have already been constructed in the previous steps. Moreover, the present method may be also used for the reconstruction of graphs from topological indices.

The proposed method of constructive enumeration of graphs was implemented in Pascal on a PC-AT compatible computer (with 20-MHz clock). In Table I are presented results of the program for enumeration of connected graphs with the prescribed number of vertices and edges. The results fully agree with the values given by Harary and Palmer.[29]

REFERENCES

(1) Read, R. C.; Corneil, D. G. The Graph Isomorphism Disease. *J. Graph Theory* **1977**, *1*, 339–352.
(2) Unger, S. H. GIT—A Heuristic Program for Testing Pairs of Directed Line Graphs for Isomorphism. *Commun. ACM* **1964**, *7*, 26–34.
(3) Randić, M. On Canonical Numbering of Atoms in a Molecule and Graph Isomorphism. *J. Chem. Inf. Comput. Sci.* **1977**, *17*, 171–180.
(4) Morgan, H. L. The Generation of a Unique Description for Chemical Structures—A Technique Developed at Chemical Abstracts Service. *J. Chem. Doc.* **1965**, *5*, 107–113.
(5) Nagle, J. F. On Ordering and Identifying Undirected Linear Graphs. *J. Math. Phys.* **1966**, *4*, 1588–1592.
(6) Hendrickson, J. B.; Toczko, A. G. Unique Numbering and Cataloguing

of Molecular Structures. *J. Chem. Inf. Comput. Sci.* **1983**, *23*, 171-177.
(7) Wipke, W. T.; Dyott, T. M. Stereochemically Unique Naming Algorithm. *J. Am. Chem. Soc.* **1974**, *96*, 4834-4842.
(8) Jochum, C.; Gasteiger, J. Canonical Numbering and Constitutional Symmetry. *J. Chem. Inf. Comput. Sci.* **1977**, *17*, 113-117.
(9) Schubert, W.; Ugi, I. Constitutional Symmetry and Unique Description of Molecules. *J. Am. Chem. Soc.* **1978**, *100*, 37-41.
(10) Masinter, L. M.; Sridharan, N. S.; Carhart, R. E.; Smith, D. H. Applications of Artificial Intelligence for Chemical Inference XIII. Labeling of Objects Having Symmetry. *J. Am. Chem. Soc.* **1974**, *96*, 7714-7723.
(11) Carhart, R. E. Erroneous Claims Concerning the Perception of Topological Symmetry. *J. Chem. Inf. Comput. Sci.* **1978**, *18*, 108-110.
(12) Dyott, T. M.; Hove, W. J. Canonical Numbering. *J. Chem. Inf. Comput. Sci.* **1979**, *19*, 187-187.
(13) Shelley, C. A.; Munk, M. J. An Approach to the Assignment of Canonical Connection Tables and Topological Symetry Perceptions. *J. Chem. Inf. Comput. Sci.* **1979**, *19*, 247-250.
(14) Bersohn, M. A Sum Algorithm for Numbering the Atoms of a Molecule. *Comput. Chem.* **1978**, *3*, 113-116.
(15) Heap, B. R. The Production of Graphs by Computer. In *Graph Theory and Computing*; Academic Press: New York, 1972; pp 47-62.
(16) Baker, H. H.; Dewdney, A. K.; Szilard, A. L. Generation the Nine-Point Graphs. *Math. Comp.* **1974**, *127*, 833-838.
(17) Bussemaker, F. S.; Cobejlič, S.; Cvetkovič, L. M.; Seidel, J. J. Computing Investigation of Cubic Graphs; Technical Report 76-WSK-01; Technical University Eindhowen: Eindhowen, 1976.
(18) Arlazarov, V. L.; Zuev, I. I.; Uskov, A. V.; Faradzhev, I. A. Algorithm for Transformation of Finite Nonoriented Graphs to Canonical Form. *Zn. Vychisl. Mat. Mat. Fiz.* **1974**, *14*, 737-743 (in Russian).
(19) Lederberg, J. *Computation of Molecular Formulas for Mass Spectroscopy*; Holden-Day: San Francisco, 1964.
(20) Read, R. C. The Enumeration of Acyclic Chemical Compounds. In *Chemical Application of Graph Theory*; Balaban, A. T., Ed.; Academic Press: London, 1976; pp 25-61.
(21) Trinajstič, N. *Chemical Graph Theory*; CRC Press: Boca Raton, FL, 1983; Vol. II.
(22) Harary, F. *Graph Theory*; Addison-Wesley: Reading, MA, 1969.
(23) Lindsay, R. K.; Buchanan, B. G.; Feigenbaum, E. A.; Lederberg, J. *Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project*; McGraw-Hill: New York, 1980.
(24) Balaban, A. T. Enumeration of Cyclic Graphs. In *Chemical Application of Graph Theory*; Balaban, A. T., Ed.; Academic Press: London, 1976; pp 63-105.
(25) Gray, N. A. B. *Computer Assisted Structure Elucidation*; Wiley: New York, 1986.
(26) Faradzhev, I. A. *Algorithmic Investigations in Combinatorics*; Nauka: Moscow, 1978 (in Russian).
(27) Essam, J. W.; Fisher, M. E. Supplement: Some Basic Definitions in Graph Theory. *Rev. Mod. Phys.* **1970**, *42*, 271-288.
(28) Lawler, E. L.; Wood, D. E. Branch and Bound Methods. *J. Oper. Res. Soc. Am.* **1966**, *14*, 217-245.
(29) Harary, F.; Palmer, E. M. *Graphical Enumeration*; Academic Press: New York, 1973.
(30) Rouvray, D. H. Topological Indices as Chemical Behaviour Descriptors. *Congr. Numerantium* **1985**, *49*, 161-179.
(31) Rouvray, D. H. The Modeling of Chemical Phenomena Using Topological Indices. *J. Comput. Chem.* **1987**, *8*, 470-480.
(32) Wiener, H. Structural Determination of Paraffin Boiling Points. *J. Am. Chem. Soc.* **1947**, *69*, 17-20.
(33) Randič, M. On Characterization of Molecular Branching. *J. Am. Chem. Soc.* **1975**, *97*, 6609-6615.
(34) Skvorcova, M. I.; Baskin, I. I.; Devdariani, R. O.; Zefirov, N. S. On the Problem of Generation of Structures of Organic Compounds with Prescribed Properties. *Proceedings of 8th All-Union Conference on Application of Computers in Molecular Spectroscopy and Chemical Research*; Novosibirsk Institute of Organic Chemistry, Academy of Sciences of USSR: Novosibirsk, USSR, 1989; pp 250-251 (in Russian).
(35) Zefirov, N. S.; Skvorcova, M. I.; Stankevitch, I. V. Generation of Structures of Polycondensed Benzenoid Hydrocarbons with Given Randič Index. *Proceedings of 8th All-Union Conference on Application of Computers in Molecular Spectroscopy and Chemical Research*; Novosibirsk Institute of Organic Chemistry, Academy of Sciences of USSR: Novosibirsk, USSR, 1989; pp 252-253 (in Russian).
(36) Gordeeva, E. V.; Zefirov, N. S. Solution of Inverse Problem for Wiener and Randič Topological Indices. Programs RING and WING. *Proceedings of 8th All-Union Conference on Application of Computers in Molecular Spectroscopy and Chemical Research*; Novosibirsk Institute of Organic Chemistry, Academy of Sciences of USSR: Novosibirsk, Novosibirsk, USSR, 1989; pp 254-255 (in Russian).

Enhanced Algorithm for Finding the Smallest Set of Smallest Rings

CHENG QIAN,* WILLIAM FISANICK, DALE E. HARTZLER, and STEVEN W. CHAPMAN
Chemical Abstracts Service, P.O. Box 3012, Columbus, Ohio 43210

Received August 21, 1989

The search algorithm for the smallest set of smallest rings (SSSR) has long been an important basic algorithm for processing chemical information. This paper describes the merits and limitations of one of the SSSR search algorithms used at Chemical Abstracts Service (CAS), provides the mathematical basis for the general approach, presents enhancements to this algorithm, and includes a new, more rigorous approach that extends the scope of the original algorithm while reducing its limitations.

I. INTRODUCTION

Isolated rings and isolated joined rings in a chemical structure are referred to as ring systems—an important part of structural topology used to identify and characterize structures. Because of their importance, the ring systems have been reported in *Chemical Abstracts* since 1907. In 1940, *The Ring Index*, a catalog of all known ring systems, was published by the American Chemical Society; a second edition was published in 1960.[1] The current catalog is the Chemical Abstracts Service (CAS) *Ring Systems Handbook*.[2]

In a ring system, all the possible rings, including the envelope rings, form the all-ring set of the system.[3] Although it is the most complete set and so provides exhaustive information about the ring system, the number of rings in a complex system usually makes the all-ring set unsuitable for practical use. Therefore, the topological features of the ring system are typically characterized by a subset of the all-ring set. However, the subset adopted to describe the ring system is not always the same under different implementations. The smallest set of smallest rings (SSSR)[4] is the ring set most commonly used, although some authors recommend SSSR+, i.e., the SSSR plus other rings.[5,6] [For example, Fujita has suggested a set of rings called the essential set of essential rings (ESER).[7]] All of these ring sets, however, are related: a ring in the all-ring set must be one of the SSSR rings or a linear combination of SSSR rings; SSSR+ sets are subsets of the all-ring set and supersets of an SSSR ring. Therefore, an SSSR or an SSSR+ can be generated by filtering unqualified members from the all-ring set (reduction strategy), or the all-ring set and SSSR+ can be generated by combining SSSR members with joint edges (expansion strategy).

Balaban presented an algorithm to generate the all-ring set of a ring system that uses a homomorphically reduced graph (HRG).[3] The use of the HRG greatly simplifies ring system graphs and makes the reduction strategy feasible even when limited computing resources are available. However, selecting

0095-2338/90/1630-0105$02.50/0