CONSTRUCTIVE ENUMERATION OF ACYCLIC MOLECULES

Vladimír KVASNIČKA and Jiří POSPÍCHAL

Department of Mathematics, Slovak Technical University, 812 37 Bratislava

Received June 14, 1990
Accepted December 20, 1990

Dedicated to Ivar Ugi, in honor of his 60th birthday.

Simple combinatorial theory of constructive enumeration of rooted trees and trees is suggested. As a byproduct of this approach very simple recursive formulae for numerical (i.e. nonconstructive) enumeration are obtained. The method may be simply generalized for (rooted) trees with edges evaluated by multiplicities and vertices evaluated by alphabetic — atomic symbols. In the process of constructive enumeration the (rooted) trees are represented by unambiguous linear code composed of valences of vertices, edge multiplicities, and atomic symbols assigned to vertices. The elaborated theory may serve as a simple algorithmic background of computer programs for constructive enumeration of acyclic molecular structures containing heteroatoms and multiple bonds.

1. INTRODUCTION

The problem of constructive enumeration[1-3] of acyclic molecular structures (or graph-theoretically, trees) was initially studied and successfully solved by Joshua Lederberg[4-5] for purposes of his famous DENDRAL project[6]. He devised a very simple procedure for coding acyclic molecules, the procedure produced a linear string composed of alphanumerical entries. These linear strings unambiguously represent acyclic molecules in a manner closely related to the usual chemical notation. Applying basic ideas of Henze and Blair[7-8] for numerical (i.e. nonconstructive) enumeration of alkanes and their simple analogs and derivatives, Lederberg was

able to formulate effective algorithm for constructive enumeration of acyclic molecules. In the first step the so-called radicals (rooted trees) were constructed, and then (second step) linked together to form alkanes. This method is based on the well-known Jordan theorem[9-10], (see Theorem 1) stating that each tree has uniquely determined centroid (or bicentroid, but not simultaneously both of them) represented by a vertex (edge) used for the above mentioned linkage of radicals producing alkanes.

The problem of linear codes of (rooted) trees has frequently been studied in the literature[11]. The interest is justified, since these linear codes solve correctly the problem of isomorphism of (rooted) trees, i.e. if a couple of (rooted) trees have the same code, then they are isomorphic. We shall use the linear codes initially elaborated by Read[11] (cf. also Knop et al.[12]). This approach will be generalized towards the possibility to reflect the multiplicities of edges as well as the atomic symbols assigned to vertices.

Recent literature[1-3,13-18] presented many different approaches for constructive enumeration of general molecular structures, i.e. not restricted only to cyclic structures. One approach[6,19], though with implanted heuristics but likely most effective one, is based on the constructive enumeration of trees; some special vertices (treated as the so-called superatoms) are expanded in cyclic structures. It means that almost all chemically relevant structures may be constructed in this simple way initially suggested by Lederberg[19]. The diversity of the produced structures depend entirely on the class of used superatoms. The class of superatom determine the kind of cyclic system, in which the superatom can be expanded. Therefore, we believe that there is very important theoretical as well as computational task to formulate the basic principles of constructive enumeration of acyclic structures. The purpose of this communication is to formulate an effective (theoretical) method, a method employing only simple graph-theoretical and combinatorial notions for constructive enumeration of acyclic molecular structures, and equipped with flexible theoretical tools for incorporation of additional requirements, characterizing more deeply their topology.

2. BASIC CONCEPTS

Let us consider a *tree*[10] (connected graph without cycles) $\mathbb{T} = (V, E)$, where $V = \{v_1, v_2, \ldots, v_N\}$ is a nonempty (i.e. $N \geq 1$) *vertex set* and $E = \{e_1, e_2, \ldots, e_M\}$ an *edge set*. The cardinalities of sets V and E, corresponding to the integers N and M, respectively, are mutually related by

$$M = N - 1 . \tag{1}$$

Let $v \in V$ be a vertex of the tree $\mathbb{T}$, the *valence* of this vertex, denoted by val(v), is the number of edges that are incident with the vertex v. The sum of valences of all

vertices satisfies[10]

$$\sum_{v \in V} \mathrm{val}(v) = 2M = 2(N - 1) . \tag{2}$$

A *rooted tree*[10] has one vertex, called the *root*, which is especially distinguished from the vertices of V. Formally, the rooted tree will be determined as an ordered triple

$$\mathbb{T}(v) = (V, E, v) , \tag{3}$$

where $v \in V$ is the root of the rooted tree $\mathbb{T}(v)$. The two rooted trees $\mathbb{T}(v) = (V, E, v)$ and $\mathbb{T}'(v') = (V', E', v')$ are *isomorphic*[10] ($\mathbb{T}(v) \approx \mathbb{T}'(v')$) if and only if (iff) there exists a 1–1 mapping ϕ: $V \to V'$ which saves the adjacency of vertices and maps the root of $\mathbb{T}(v)$ on the root of $\mathbb{T}'(v')$, i.e. $\phi(v) = v'$.

For every tree there exists a unique vertex or an edge. The unique vertex may be classified as a root, in the case of a unique edge a vertex incident with it may be classified as a root.

Theorem 1. Let $\mathbb{T}$ be a tree composed of N vertices, the following three different cases should be separately considered:

(1) For odd $N = 2k + 1$ there exists a unique vertex, called the *centroid*, such that all (two or more) incident subtrees are composed, at most, of k vertices (see Fig. 1, graph A).

For even $N = 2k$ there exists either

(2) a unique vertex – *centroid* such that all (three or more) incident subtrees are composed of less than k vertices (see Fig. 1, graph B), or

(3) a unique edge, called the *bicentroid*, such that the incident two subtrees are composed exactly of k vertices (see Fig. 1, graph C).

This theorem[10] was initially proved by Jordan[9] in 1869. It states that for trees

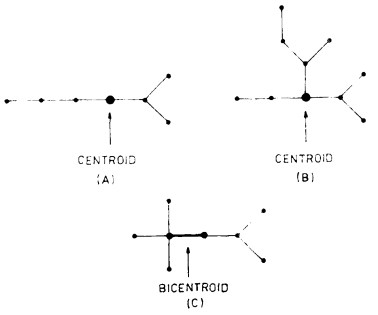

FIG. 1

The tree A composed of seven vertices contains the centroid denoted by heavy dot. The trees B and C are composed of even number of vertices (ten and eight, respectively), the tree B contains centroid whereas the tree C contains the bicentroid denoted by bold edge

with an odd number of vertices the centroid is unambiguously determined, whereas for trees with even number of vertices there exists either a centroid or a bicentroid (an edge of $\mathbb{T}$). If a tree $\mathbb{T}$ has a bicentroid, then this tree has a pair of vertices (incident with the edge – bicentroid) and one of them may play a role of the centroid; this potential ambiguity will be simply removed by making use of the lexicographical ordering of codes assigned to the rooted subtrees (see Section 3).

The above theorem claims nothing about the finding of the centroid or bicentroid in a tree, it ensures only that one of them does exist. Since the concept of centroid/bicentroid is of great importance for our forthcoming considerations, it is necessary to have an algorithm for finding of centroid/bicentroid[11]. This algorithm will be formulated in a recurrent manner: for each step the vertex set V is divided into two disjoint subsets,

$$V = V_e \cup V_{ne} , \tag{4}$$

where the subset V_e and V_{ne} is composed of the so-called *evaluated* and *nonevaluated vertices*, respectively. Every vertex $v \in V_e$ is evaluated by an integer denoted by $\chi(v)$. A vertex $v \in V_{ne}$ is called the *candidate* if it will satisfy the following two conditions: (1) The vertex v is adjacent to one or more evaluated vertices, and (2) the vertex is adjacent to just one nonevaluated vertex. A candidate $v \in V_{ne}$ is evaluated by

$$\chi(v) = 1 + \sum_{v'} \chi(v') , \tag{5}$$

where the summation runs over all already evaluated vertices incident with the vertex v. The algorithm is applicable for trees composed of three or more vertices. For trees with one or two vertices a determination of centroid/bicentroid is a trivial task.

Algorithm 1.

Step 1. (Initialization) The marginal vertices of $\mathbb{T}$ (i.e. the vertices with unit valences) are evaluated by 1.

Step 2. If the number of nonevaluated vertices is equal to 1 (2), then go to step 4 (step 5).

Step 3. The candidate with the minimal value of potential evaluation is evaluated. If there exists more than one such candidate, then all of them are evaluated. Go to step 2.

Step 4. The nonevaluated vertex is centroid – go to step 6.

Step 5. If values of potential evaluations of the two remaining vertices are equal, then an edge simultaneously incident with both candidates is bicentroid. Otherwise the centroid is the vertex with greater potential evaluation.

Step 6. The end of algorithm.

Simple illustrative examples of this algorithm are shown in Fig. 2. We see that the evaluations of vertices are equal to numbers of vertices of the corresponding subtrees going successively from the centroid/bicentroid to marginal vertices. It is easy to understand that in the case of last two still nonevaluated vertices with equal potential evaluation (see step 5) these should be adjacent and form an edge called bicentroid. Moreover, the algorithm may be taken as an alternative constructive proof of the Theorem 1.

3. LINEAR CODES OF TREES

Recently, the concept of linear codes for the constructive enumeration of trees was studied by Read[11] and by Knop et al.[12] They demonstrated the uniqueness, effectiveness, and compactness of this approach. First, we shall present their recursive construction for rooted trees and then the method will be generalized also for trees.

The linear code for rooted trees consists of a string of digits which corresponds either to the valence of the root or to the valence decreased by 1 for other vertices (see Fig. 3). We have to emphasize that the linear code approach is applicable only for a rooted tree $\mathbb{T}(v) = (V, E, v)$ with the valences of vertices restricted by $1 \leqq \leqq \mathrm{val}(v) \leqq 9$ and $1 \leqq \mathrm{val}(v') \leqq 10$, $\forall v' \in V \setminus \{v\}$. The linear code assigned to the rooted tree $\mathbb{T}(v)$ will be denoted by code $(\mathbb{T}(v))$; this code may be simply interpreted as a decimal number. In case of valences greater than 10 there is a possibility to replace the code given by a p-digit number (where p is number of vertices) by a p dimensional vector.

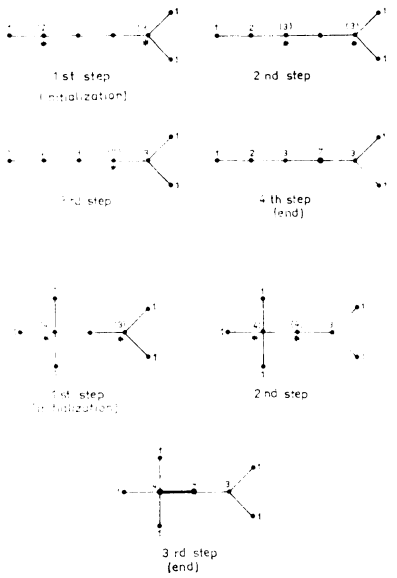

FIG. 2

Illustrative examples of Algorithm 1 applied to graphs A and C in Fig. 1. The vertices labeled by stars are the candidates, potential evaluations of vertices are given in parentheses

The linear code assigned to the rooted tree $\mathbb{T}(v)$ composed of one vertex (i.e. $V = \{v\}$) is a zero digit, code$(\mathbb{T}(v))$ = ‚0'. Let us now study a rooted tree $\mathbb{T}(v)$ composed of two or more vertices and assume that its root is incident with q $(1 \leqq q \leqq 9)$ subtrees, i.e. deleting the root from the tree we get q, the so-called *first-generation* subtrees. These first-generation subtrees will again be formally treated as rooted trees, the roots of which are identified by vertices that were adjacent to the original root of $\mathbb{T}(v)$. The formed rooted trees are denoted by $\mathbb{T}_1(v_1), \mathbb{T}_2(v_2), \ldots, \mathbb{T}_q(v_q)$. The linear code of the original root $\mathbb{T}(v)$ is determined as:

$$\text{code}\,(\mathbb{T}(v)) = \text{'}q\text{'} + \text{code}\,(\mathbb{T}_{i_1}(v_{i_1})) + \ldots + \text{code}\,(\mathbb{T}_{i_q}(v_{i_q}))\,, \tag{6}$$

wherein the operation '+' means the concatenation of "subcodes", and indices $(i_1, i_2, \ldots, i_q)$ correspond to a permutation of $(1, 2, \ldots, q)$ such that the codes are lexicographically ordered,

$$\text{code}\,(\mathbb{T}_{i_1}(v_{i1})) \leqq \text{code}\,(\mathbb{T}_{i_2}(v_{i_2})) \leqq \ldots \leqq \text{code}\,(\mathbb{T}_{i_q}(v_{i_q}))\,. \tag{7}$$

If in the r.h.s. of Eq. (6) a rooted tree code $(\mathbb{T}_{i_j}(v_{i_j}))$ is composed of more than one vertex, then its code is determined by an analog of Eq. (6); this procedure is recursively repeated until the appearing rooted subtrees are composed of a single vertex (their codes are the unit strings composed of zero digit, '0'. The construction of linear codes for rooted trees is illustrated in Fig. 3.

Theorem 2. Two rooted trees $\mathbb{T}(v)$ and $\mathbb{T}'(v')$ are isomorphic iff their linear codes are equal,

$$\mathbb{T}(v) \approx \mathbb{T}'(v') \Leftrightarrow \text{code}\,(\mathbb{T}(v)) = \text{code}\,(\mathbb{T}'(v'))\,. \tag{8}$$

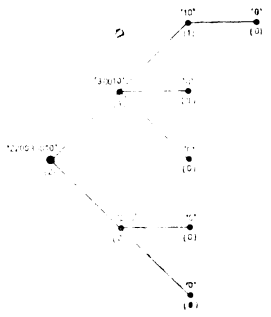

FIG. 3

Finding of linear code of a rooted tree. In parentheses are given initial evaluations of vertices, all vertices are evaluated by their valences decreased by 1, except for the root, evaluated by its valence. The linear code is constructed successively going from the marginal vertices to the root. For a given vertex, different of marginal vertices, a linear subcode is constructed by the concatenation of its evaluation with the subcodes (lexicographically ordered) of its predecessors

This very important theorem was proved by Knop et al.[12]; it states that the rooted tree is unambiguously represented by a string of decimal digits.

The above described linear code approach for rooted trees may be straightforwardly enlarged also for trees. Following Theorem 1, each tree has unique either a centroid or a bicentroid, but not simultaneously both of them. If the tree $\mathbb{T} = (V, E)$ has a centroid corresponding to the vertex $v \in V$, then the given tree is formally considered as a rooted tree $\mathbb{T}(v)$, and the linear code of the tree $\mathbb{T}$ is set equal to code $(\mathbb{T}(v))$, i.e. code $(\mathbb{T})$ = code $(\mathbb{T}(v))$. Second, if the tree $\mathbb{T}$ has a bicentroid equal to an edge $e = [v, v'] \in E$, then the linear code of $\mathbb{T}$ is lesser linear code of code $(\mathbb{T}(v))$ and code $(\mathbb{T}(v'))$.

$$\text{code}(\mathbb{T}) = \begin{cases} \text{code}(\mathbb{T}(v)), & (\text{code}(\mathbb{T}(v)) \leqq \text{code}(\mathbb{T}(v'))) \\ \text{code}(\mathbb{T}(v')), & (\text{code}(\mathbb{T}(v')) < \text{code}(\mathbb{T}(v))) \end{cases} \tag{9}$$

It means also, that for trees with bicentroid the linear codes are unambiguously determined. Construction of linear codes is illustrated by simple examples in Fig. 4.

Theorem 3. Two trees $\mathbb{T}$ and $\mathbb{T}'$ are isomorphic iff their linear codes are equal,

$$\mathbb{T} \approx \mathbb{T}' \Leftrightarrow \text{code}(\mathbb{T}) = \text{code}(\mathbb{T}') . \tag{10}$$

This theorem is an immediate consequence of the above Theorems 1 and 2.

Algorithm 1, presented in the previous section, for finding of a centroid/bicentroid, may be simply modified in such a way that it gives linear code of the tree. One only needs to substitute the integer evaluations of single vertices by the corresponding

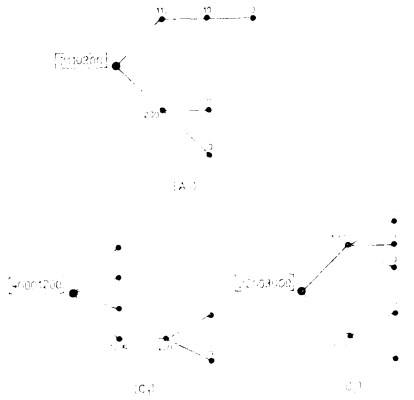

FIG. 4

Finding the linear codes of trees A and C in Fig. 1. The root in A_1 corresponds to the centroid of the tree, the linear code of A in Fig. 1 is equal to the linear code of rooted subtree A_1. The tree C in Fig. 1 has a bicentroid, the vertices incident with the edge-bicentroid are used as roots, then we construct linear codes of the corresponding rooted trees C_1 and C_2, lesser code determines the linear code of the tree C in Fig. 1. The correct linear codes are placed at a block

linear codes of subtrees, constructed according to the formulae (*6*) and (*7*). The marginal vertices are initially evaluated by the unit string '0'. Some small formal difficulties may arise when the studied tree contains bicentroid. In that case the linear codes assigned recursively to the vertices from the bicentroid would not be immediately used for construction of the linear code of the whole tree. We then select the vertex with a smaller linear code as a centroid and linear code of the tree is constructed from codes of vertices belonging to the bicentroid. We increase the first digit of linear code of centroid by 1 and concatenate the obtained code with the code of the other vertex of the bicentroid, see Fig. 5.

4. CONSTRUCTIVE ENUMERATION OF ROOTED TREES

In our approach the constructive enumeration of rooted trees is based on a recursive process of construction of linear codes from the already constructed rooted trees with smaller number of vertices than that one just constructed. The approach will automatically ensure that the produced codes correspond to nonisomorphic rooted trees (see Theorem 2) and that it is exactly counting (enumerating) their appearance.

A rooted tree $\mathbb{T}(v) = (V, E, v)$ is called *q-nary* if $\mathrm{val}(v) = q$, i.e. the root v is adjacent to q vertices. Deleting the root v from $\mathbb{T}(v)$ we get q subtrees that may be again formally considered as rooted trees. In particular, let the root of $\mathbb{T}(v)$ be adjacent to vertices $\{v_1, v_2, \ldots, v_q\} \subseteq V$, deleting the root v we obtain the so-called *first-generation* rooted trees $\mathbb{T}_1(v_1), \mathbb{T}_2(v_2), \ldots, \mathbb{T}_q(v_q)$, where $|V| = 1 + |V_1| +$

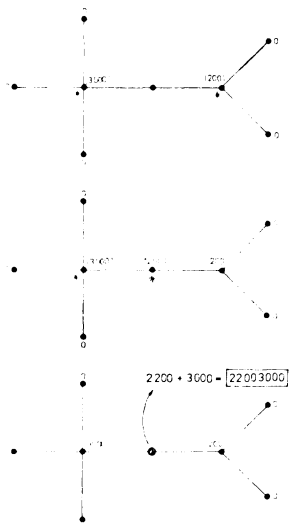

FIG. 5

Construction of linear code by making use of the modified form of Algorithm 1 for a tree with bicentroid, the resulting code is placed in a block

$|V_2| + \ldots + |V_q|$. This process may be recursively repeated until the produced rooted trees are composed of only one vertex, see Fig. 6.

The above simple considerations, carried out in a reverse way and combined with the construction of linear codes of rooted trees, offer an effective method for constructive enumeration of rooted trees (initially used by Henze and Blair[7,8] for enumeration of monosubstituted alkanes, e.g. alcohols, and by Joshua Lederberg[4,5] for constructive enumeration of acyclic molecules in the framework of his famous DENDRAL project[6]).

Let $\mathscr{R}_p$ be a class of all mutually nonisomorphic rooted trees composed of p vertices, its cardinality, (i.e. number of elements – rooted trees) is denoted by r_p, $|\mathscr{R}_p| = r_p$. We say that a rooted tree $\mathbb{T}(v)$ belongs to the class $\mathscr{R}_p$, formally $\mathbb{T}(v) \in \mathscr{R}_p$, if it is isomorphic to a rooted tree from $\mathscr{R}_p$ (i.e. $\mathbb{T}(v)$ has p vertices). The goal of our theoretical considerations is to construct the class $\mathscr{R}_p$ under an assumption that the classes $\mathscr{R}_1, \mathscr{R}_2, \ldots, \mathscr{R}_{p-1}$ were already constructed. A q-nary rooted tree $\mathbb{T}(v)$ from $\mathscr{R}_p$ (i.e. the vertex v is adjacent to q vertices that form again the roots of the first-generation rooted trees, see second paragraph of this section) may be formally expressed as follows

$$\mathbb{T}(v) = v \oplus \mathbb{T}_1(v_1) \oplus \mathbb{T}_2(v_2) \oplus \ldots \oplus \mathbb{T}_q(v_q)\,, \tag{11}$$

where the r.h.s. is interpreted as a "gluing" of the vertex v (a root of the produced tree) with the (first generation) rooted trees $\mathbb{T}_1(v_1)$, $\mathbb{T}_2(v_2)$, ..., $\mathbb{T}_q(v_q)$, the resulting rooted tree is isomorphic to $\mathbb{T}(v)$. The vertex and edge sets of $\mathbb{T}(v)$ are determined by

$$V = \{v\} \cup V_1 \cup V_2 \cup \ldots \cup V_q\,, \tag{12a}$$

$$E = \{[v, v_i]; 1 \leqq i \leqq q\} \cup E_1 \cup E_2 \cup \ldots \cup E_q\,. \tag{12b}$$

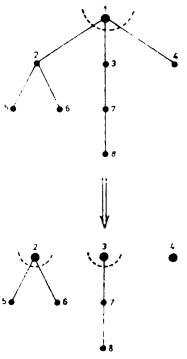

FIG. 6

Deleting the root (indexed by 1) from the rooted tree we get the first generation rooted trees with roots indexed by 2, 3, and 4

The only restriction imposed on the rooted trees $\mathbb{T}_1(v_1), \mathbb{T}_2(v_2), \ldots, \mathbb{T}_q(v_q)$ is that their number of vertices should be restricted by

$$|V| = 1 + |V_1| + |V_2| + \ldots + |V_q| . \tag{12c}$$

Summarizing our considerations, a rooted tree $\mathbb{T}(v)$ is unambiguously reconstructed (up to an isomorphism) from its first-generation rooted trees restricted by the condition (*12c*).

Theorem 4. Let $\mathbb{T}_1(v_1), \mathbb{T}_2(v_2), \ldots, \mathbb{T}_q(v_q)$ be rooted trees restricted by $p = 1 + |V_1| + |V_2| + \ldots + |V_q|$, and v be a vertex not belonging to these rooted trees, then the expression (*11*) determines unambiguously (up to an isomorphism) a q-nary rooted tree $\mathbb{T}(v)$ composed of p vertices, i.e. $|V| = p$ and $\mathrm{val}(v) = q$.

The above theorem offers attractive and straightforward possibilities for constructing all rooted trees with the prescribed number of vertices and given valence of their roots. The thus obtained rooted trees will be represented by their linear codes, constructed by the way outlined in the previous section.

Let us consider a subclass $\mathscr{R}_{p,q} \subset \mathscr{R}_p$ composed of all q-nary rooted trees with p vertices. Following the above considerations, the rooted trees from $\mathscr{R}_{p,q}$ can be exhaustively constructed by making use of their first-generation rooted trees which belong to $\mathscr{R}_1, \mathscr{R}_2, \ldots, \mathscr{R}_{p-q}$. Here we have to note that the set $\mathscr{R}_1$ contains only one isolated vertex — root, i.e. $r_1 = 1$. The integer $(p - 1)$ (where $p \geqq 2$) may be decomposed into positive integers $1 \leqq a < b < \ldots$ as follows

$$\alpha a + \beta b + \ldots = p - 1 , \tag{13a}$$

$$\alpha + \beta \ldots = q , \tag{13b}$$

where positive integers $\alpha, \beta, \ldots$ determine "multiplicities" of appearance integers $a, b, \ldots$ Such a decomposition will be formally abbreviated in the form of $a^\alpha b^\beta \ldots$. For instance, if we put $p - 1 = 5$ and $q = 3$, then we get two distinct decompositions $1^1 2^2$ and $1^2 3^1$. The decomposition $a^\alpha b^\beta \ldots$, restricted by Eqs (*13a–13b*), means that in the process of construction of q-nary rooted trees with p vertices we shall use as the first-generation α rooted trees from the class $\mathscr{R}_a$, β rooted trees from $\mathscr{R}_b$, and so on. Here it is very important to emphasize that from the class $\mathscr{R}_x$ (for $x = a, b, \ldots$) we take into account ξ-tuples of rooted trees in which some of them may be identical. In terminology of combinatorics we say that all combinations of ξ elements (i.e. rooted trees) with allowed repetitions are considered. Their number is determined by

$$[r_x^\xi] = \binom{r_x + \xi - 1}{\xi} , \tag{14}$$

where $\binom{p}{q} = p!/[q!(p-q)!]$ is the so-called binomial number. Let $\mathbb{T}_1(v_1)$, $\mathbb{T}_2(v_2)$, ..., $\mathbb{T}_q(v_q)$ be rooted trees assigned to the decomposition $a^\alpha b^\beta \ldots$, i.e. the first α trees are from the class $\mathscr{R}_a$, the next β trees are from the class $\mathscr{R}_b$, etc., then the resulting rooted tree $\mathbb{T}(v)$ and its linear code are determined by Eqs (*11*) and (*6*), respectively. Repetition of this process for all decompositions $a^\alpha b^\beta \ldots$ will give us all possible q-nary rooted trees composed of p vertices. If this construction is successively repeated for fixed p and $q = 1, 2, \ldots, p-1$, then we get all rooted trees from $\mathscr{R}_p$. Simple illustrative example of the construction of class $\mathscr{R}_p$ is given in Fig. 7. Finally, the above theory gives also a very simple formula for the number of rooted trees in $\mathscr{R}_p$, i.e. the number $r_p = |\mathscr{R}_p|$,

$$r_p = \sum_{q=1}^{p-1} \sum_{a^\alpha b^\beta} [r_a^\alpha]\,[r_b^\beta] \ldots , \qquad (15)$$

where the symbols $[\cdot]$ were defined by Eq. (*14*) and the second summation run over all decompositions of the integer $(p-1)$ restricted by Eqs (*13a–13b*).

The general theory of constructive enumeration of rooted trees is straightforwardly applicable also in chemistry for enumeration of acyclic monovalent functional groups (radicals). There is only necessary to bound from above the valence of carbon roots by 3, then we automatically arrive at results initially obtained by Henze and Blair[7,8] and Joshua Lederberg[4,5].

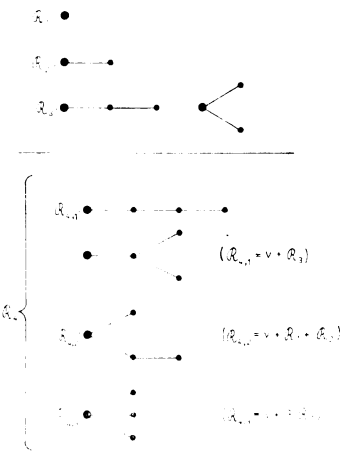

FIG. 7

Bottom part of this figure illustrates the construction of rooted trees composed of four vertices. In parentheses are formally specified the ways of construction of rooted trees from the same line

5. CONSTRUCTIVE ENUMERATION OF TREES

The constructive enumeration of trees shares many common features with the constructive enumeration of rooted trees outlined in the previous section. The main problem emerging here concerns the problem of which vertex of a tree is to be classified as a root. Fortunately, the problem with the selection of root is unambiguously solved by the Theorem 1. The centroids are declared as roots; if a tree has a bicentroid, then one of its vertices may play the role of a root. We shall study separately the trees composed of odd and even number of vertices.

5.1 *Trees with Odd Number of Vertices*

According to Theorem 1, for trees composed of $p = 2k + 1$ vertices the centroid is unambiguously determined in such a way that it is incident with two or more rooted trees composed of at most k vertices. It means, for trees with an odd number of vertices the centroids may be declared as roots, and the whole approach to constructive enumeration of rooted trees is applicable, with minor modification, also to the enumeration of trees.

Let $\mathscr{T}_p$ be a class of all mutually nonisomorphic trees with p vertices, its cardinality (i.e. number of elements – trees) is denoted by t_p. We say that a tree $\mathbb{T}$ belongs to the class $\mathscr{T}_p$, formally $\mathbb{T} \in \mathscr{T}_p$, if it is isomorphic to a tree of $\mathscr{T}_p$ (i.e. $\mathbb{T}$ has p vertices). The class $\mathscr{T}_1$ is composed of one tree represented by single isolated vertex, $t_1 = 1$. A tree $\mathbb{T} = (V, E) \in \mathscr{T}_p$, with centroid $v \in V$, where p is an odd integer determined by $p = 2k + 1$, may be formally treated as a rooted tree $\mathbb{T}(v) = (V, E, v)$. The root v should be adjacent to $q \geqq 2$ vertices $\{v_1, v_2, \ldots, v_q\} \subseteq V$, deleting the vertex v we get the first-generation rooted trees $\mathbb{T}_1(v_1), \mathbb{T}_2(v_2), \ldots, \mathbb{T}_q(v_q)$ restricted by

$$|V_1| + |V_2| + \ldots + |V_q| + 1 = p = 2k + 1\,, \tag{16a}$$

$$1 \leqq |V_i| \leqq k\,, \quad (\text{for } i = 1, 2, \ldots, q)\,, \tag{16b}$$

where the last inequalities (*16b*) reflect the fact that the subtrees are composed of at most k vertices. Since the root v assigned to the centroid of $\mathbb{T}$ is unambiguously determined we get an analog of Theorem 4 for trees.

Theorem 5. Let $\mathbb{T}_1(v_1), \mathbb{T}_2(v_2), \ldots, \mathbb{T}_q(v_q)$, where $q \geqq 2$, be rooted trees restricted by Eqs (*16a* to *16b*) and v be a vertex not belonging to these rooted trees, then the expression (*11*) determines unambiguously (up to an isomorphism) a tree $\mathbb{T}$ composed of $p = 2k + 1$ vertices and its centroid is equal to the vertex v.

In a way similar to that for rooted trees, this theorem offers simple method for constructive enumeration of trees with odd number of vertices. In order to construct

the class $\mathscr{T}_p$, where $p = 2k + 1$, we need to know the classes of rooted trees $\mathscr{R}_1$, $\mathscr{R}_2$, ..., $\mathscr{R}_k$; hence, the construction of trees should be preceded by the construction of rooted trees composed of at most k vertices.

The class $\mathscr{T}_p$ is divided into disjoint subclasses $\mathscr{T}_{p,1}$, $\mathscr{T}_{p,2}$, ..., $\mathscr{T}_{p,q}$, ..., where $\mathscr{T}_{p,q}$ is composed of all possible trees with $p = 2k + 1$ vertices and with q-nary centroid – root. Following Theorem 5, a tree $\mathbb{T} \in \mathscr{T}_{p,q}$ can be constructed by making use of the so-called first generation trees $\mathbb{T}_1(v_1)$, $\mathbb{T}_2(v_2)$, ..., $\mathbb{T}_q(v_q)$ that are restricted by Eqs (*16a–16b*). The integer $(p - 1)$ (for $p \geqq 2$) is decomposed into positive integers (cf. Eqs (*13a–13b*)),

$$1 \leqq a < b < \ldots \leqq k\,, \tag{17a}$$

$$\alpha a + \beta b + \ldots = p - 1\,, \tag{17b}$$

$$\alpha + \beta + \ldots = q\,, \tag{17c}$$

its abbreviated form is $a^\alpha b^\beta \ldots$. This decomposition determines the form of rooted trees $\mathbb{T}_1(v_1)$, $\mathbb{T}_2(v_2)$, ..., $\mathbb{T}_q(v_q)$ in Eq. (*11*). The process of constructive enumeration of trees with odd number of vertices is illustrated in the upper part of Fig. 8.

The same approach may be also used for simple numerical (i.e. nonconstructive)

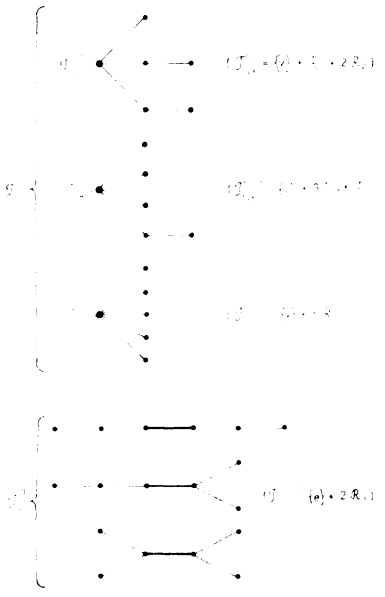

FIG. 8

Construction of all trees with six vertices. These trees are divided into two disjoint subsets composed of trees with centroid ($\mathscr{T}_6^{(c)}$) and trees with bicentroid ($\mathscr{T}_6^{(bc)}$). In the upper part trees with a centroid are given, they are constructed by three different ways, formally determined by expressions in parentheses. The bottom part illustrates trees with a bicentroid, they are formally determined by introduction of an edge – bicentroid between roots from $\mathscr{R}_3$. If we restrict the maximal valence of vertices to 4, then the tree from the third line should be omitted

enumeration of trees from $\mathscr{T}_p$, we get

$$t_p = \sum_{q=2}^{p-1} \sum_{a^\alpha b^\beta \ldots} [r_a^\alpha] [r_b^\beta] \ldots , \tag{18}$$

where the second summation runs over all decompositions $a^\alpha b^\beta \ldots$ restricted by Eqs (17a–17c). Assuming that the first summation in Eq. (18) runs only up to 4, and that classes $\mathscr{R}_1, \mathscr{R}_2, \ldots, \mathscr{R}_k$ are composed of rooted trees with at most four-valence vertices, then the formula (18) provides the results initially obtained by Henze and Blair[7,8]. Moreover, the constructive enumeration of trees restricted by these constraints produces the same trees — alkanes as the method initially suggested by Joshua Lederberg[4,5].

5.2 *Trees with Even Number of Vertices*

The trees with even number ($p = 2k$) of vertices may have either a centroid or a bicentroid. The centroid is incident with three or more first generation rooted trees composed of less than k vertices; the bicentroid is incident with two rooted trees composed of exactly k vertices. This "dichotomy" centroid/bicentroid causes some formal difficulties in the constructive enumeration of trees with even number of vertices. The trees from $\mathscr{T}_p$ (for $p = 2k$) have two distinct origins, the first ones (with centroid) are those trees constructed in close analogy to the trees with odd number of vertices; the second ones (with bicentroid) are constructed by making a simple "linkage" of two rooted trees, both containing exactly k vertices. The class $\mathscr{T}_p$ may be decomposed into two disjoint subclasses of trees with centroid or bicentroid,

$$\mathscr{T}_p = \mathscr{T}_p^{(c)} \cup \mathscr{T}_p^{(bc)} , \tag{19}$$

their cardinality is denoted by $t_p^{(c)}$ and $t_p^{(bc)}$, respectively, i.e. $t_p = t_p^{(c)} + t_p^{(bc)}$. The symbol $\mathscr{T}_{p,q}^{(c)}$ denotes a subclass of $\mathscr{T}_p^{(c)}$ composed of the trees with a centroid — root adjacent to $q \geqq 3$ rooted trees that are containing less than k vertices.

Let us start our considerations by the construction of trees belonging to the subclass $\mathscr{T}_{p,q}^{(c)}$. A tree $\mathbb{T} \in \mathscr{T}_{p,q}^{(c)}$ has a centroid — root v incident with q vertices $\{v_1, v_2, \ldots, v_q\}$, deleting the vertex v we get the first-generation rooted trees $\mathbb{T}_1(v_1)$, $\mathbb{T}_2(v_2), \ldots, \mathbb{T}_q(v_q)$ restricted by

$$|V_1| + |V_2| + \ldots + |V_q| + 1 = p = 2k , \tag{20a}$$

$$1 \leqq |V_i| < k , \quad (\text{for } i = 1, 2, \ldots, k) . \tag{20b}$$

Since the centroid v is unambiguously determined (see Theorem 1) we may formulate a theorem which ensures the uniqueness of $\mathbb{T} \in \mathscr{T}_{p,q}^{(c)}$ determined by the first generation rooted trees.

Theorem 6. Let $\mathbb{T}_1(v_1)$, $\mathbb{T}_2(v_2)$, ..., $\mathbb{T}_q(v_q)$, where $q \geqq 3$, be rooted trees restricted by Eqs (*20a–20b*) and v be a vertex not belonging to these trees, then the expression (*11*) determines unambiguously (up to an isomorphism) a tree $\mathbb{T}$ composed of $p = 2k$ vertices with a centroid equal to the vertex v.

This theorem offers, similarly as in the previous cases, a simple way how to construct exhaustively the trees from the subclass $\mathscr{T}^{(c)}_{p,q}$, where $p = 2k$ and $q \geqq 3$. Repeating this approach for all $\mathscr{T}^{(c)}_{p,q}$, $q = 3, 4, \ldots, p - 1$, we get the whole subclass $\mathscr{T}^{(c)}_p$. We shall not repeat again the details of this construction, it is quite similar to the previous constructions. The decomposition of the integer $(p - 1)$ onto positive integers $1 \leqq a < b < \ldots \leqq k - 1$, represented by $a^\alpha b^\beta \ldots$ is restricted by Eqs (*17b–17c*). These decompositions specify the first-generation rooted trees $\mathbb{T}_1(v_1)$, $\mathbb{T}_2(v_2)$, ..., $\mathbb{T}_q(v_q)$ in Eq. (*11*). The number $t^{(c)}_p$ of trees in $\mathscr{T}^{(c)}_p$ is determined by

$$t^{(c)}_p = \sum_{q=3}^{p-1} \sum_{a^\alpha b^\beta \ldots} [r_a^\alpha]\,[r_b^\beta] \ldots . \tag{21}$$

The trees of $\mathscr{T}^{(bc)}_p$ with bicentroid are constructed in the same way as it was already outlined (see comment above Eq. (*9*) and the last paragraph). Let us now consider two rooted trees $\mathbb{T}_1(v_1)$ and $\mathbb{T}_2(v_2)$ taken from the class $\mathscr{R}_k$. Formally, if we connect the roots v_1 and v_2 by an edge $[v_1, v_2]$, we obtain the following tree

$$\mathbb{T} = (V, E) = \mathbb{T}_1(v_1) \oplus \mathbb{T}_2(v_2)\,, \tag{22a}$$

where

$$V = V_1 \cup V_2\,, \tag{22b}$$

$$E = \{[v_1, v_2]\} \cup E_1 \cup E_2\,. \tag{22c}$$

The resulting tree $\mathbb{T}$ belongs to the class $\mathscr{T}^{(bc)}_p$ ($p = 2k$), its bicentroid is formed by the edge $[v_1, v_2]$.

Theorem 7. Let $\mathbb{T}_1(v_1)$ and $\mathbb{T}_2(v_2)$ be rooted trees from the class $\mathscr{R}_k$, then the expression (*22a*) determines unambiguously (up to an isomorphism) a tree composed of even number of vertices, $p = 2k$, and its bicentroid is equal to the edge $[v_1, v_2]$.

Following this theorem we can easily construct all trees from $\mathscr{T}^{(bc)}_p$. Here it must be emphasized that each pair of rooted trees, irrespective of whether they are isomorphic or not, should be in Eq. (*22*) counted exactly once. The number $t^{(bc)}_p$ of trees in $\mathscr{T}^{(bc)}_p$ is simple determined by

$$t^{(bc)}_p = [r_k^2]\,. \tag{23}$$

The construction of trees with even number of vertices and with centroid or bicentroid is illustrated in Fig. 8.

As was already mentioned, the present theory of constructive enumeration of trees is straightforwardly applicable for constructive enumeration of alkanes. It is only necessary to restrict the maximal valence of vertices – carbon atoms to 4; the obtained result exactly agreed with those obtained by Henze and Blair[7,8] and by Joshua Lederberg[4,5].

6. CONSTRUCTIVE ENUMERATION OF TREES WITH MULTIEDGES

To extend the possibilities and scope of the constructive enumeration of trees towards actual acyclic chemical systems it is vital to take into account also multiple edges (which correspond to multiple bonds in molecules). We say that a tree $\mathbb{T} = (V, E)$ is *edge-evaluated* if its edges are evaluated by positive integers called *multiplicities*. Formally, the evaluation of edges consists in a mapping

$$\varphi : E \rightarrow \{1, 2, 3, \ldots\} , \tag{24}$$

where the integer $\varphi(e)$ is the multiplicity of an edge $e \in E$. A tree $\mathbb{T}$ with edges evaluated by Eq. (24) is represented by an ordered triple

$$\mathbb{T}(\varphi) = (V, E, \varphi) . \tag{25}$$

The difference between $\mathbb{T}$ and $\mathbb{T}(\varphi)$ is well chemically assessed by the so-called *unsaturation*[6],

$$\text{unsat}\,(\mathbb{T}(\varphi)) = \sum_{e \in E} [\varphi(e) - 1] . \tag{26}$$

To some extent, unsaturation indicates the appearance of edges with multiplicities greater than one, its zero value means that all edges of $\mathbb{T}(\varphi)$ are of unit multiplicity. For example, if $\text{unsat}\,(\mathbb{T}(\varphi)) = 2$, then the tree has two double edges or one triple edge, both these cases correspond to the same value of unsaturation.

Two edge-evaluated trees $\mathbb{T}(\varphi) = (V, E, \varphi)$ and $\mathbb{T}'(\varphi') = (V', E', \varphi')$ are *isomorphic* iff there exists a 1–1 mapping

$$\psi : V \rightarrow V' \tag{27}$$

which saves simultaneously the adjacencies of vertices and the evaluation of edges,

$$\varphi([v_1, v_2]) = \varphi'([\psi(v_1), \psi(v_2)]) , \tag{28}$$

for each $[v_1, v_2] \in E$ and the corresponding "mapped" edge $[\psi(v_1), \psi(v_2)] \in E'$.

In a similar way we introduce also the notion of edge-evaluated rooted trees and their isomorphism.

Let us consider an edge-evaluated rooted tree $\mathbb{T}(v, \varphi) = (V, E, v, \varphi)$. Deleting from $\mathbb{T}(v, \varphi)$ its root incident with q vertices $\{v_1, v_2, \ldots, v_q\} \subseteq V$, we get the first-generation edge-evaluated rooted trees $\mathbb{T}_1(v_1, \varphi_1)$, $\mathbb{T}_2(v_2, \varphi_2)$, ..., $\mathbb{T}_q(v_q, \varphi_q)$. The evaluations – mappings $\varphi_1, \varphi_2, \ldots, \varphi_q$ are simple restrictions of the original mapping φ with respect to the edge sets $E_1, E_2, \ldots, E_q$.

$$\varphi_i(e) = \begin{cases} \varphi(e), & (\text{for } e \in E_i) \\ 0, & (\text{for } e \notin E_i) \end{cases} \tag{29}$$

The linear codes of edge-evaluated trees may be constructed by the way already outlined, see Eqs (6–7), enlarged by "subcodes" which determine the multiplicities of edges. In particular, the linear code code $(\mathbb{T}(v_i, \varphi_i))$ is enlarged at the first position by an integer – digit specifying the multiplicity of edge $[v, v_i] \in E$,

$$\overline{\text{code}}\,(\mathbb{T}_i(v_i, \varphi_i)) = \text{'}\varphi([v, v_i])\text{'} + \text{code}\,(\mathbb{T}_i(v_i, \varphi_i))\,. \tag{30}$$

The symbol '+' should be interpreted as concatenation of "subcodes". Then the linear code of $\mathbb{T}(v, \varphi)$ is determined as follows:

$$\text{code}\,(\mathbb{T}(v, \varphi)) = \text{'}q\text{'} + \overline{\text{code}}\,(\mathbb{T}_u(v_u, \varphi_u)) + \ldots + \overline{\text{code}}\,(\mathbb{T}_w(v_w, \varphi_w))\,, \tag{31}$$

where the indices $(u, \ldots, w)$ correspond to a permutation of $(1, \ldots, q)$ such that the enlarged codes are lexicographically ordered,

$$\overline{\text{code}}\,(\mathbb{T}_u(v_u, \varphi_u)) \leqq \ldots \leqq \overline{\text{code}}\,(\mathbb{T}_w(v_w, \varphi_w))\,. \tag{32}$$

The above procedure of construction of enlarged linear code is recursively repeated until the "first-generation" rooted trees appear to be composed of only one vertex, see Fig. 9.

Since the concept of centroid/bicentroid is determined for trees irrespective whether the edges are evaluated or not, they may be also assigned to the edge-evaluated rooted trees. If a tree $\mathbb{T}(\varphi)$ has a centroid, then it is classified as a root of $\mathbb{T}(\varphi)$. Let v be the centroid of $\mathbb{T}(\varphi)$, then this tree may be formally treated as an edge-evaluated rooted tree $\mathbb{T}(v, \varphi)$, the linear code of $\mathbb{T}(\varphi)$ is determined as follows:

$$\text{code}\,(\mathbb{T}(\varphi)) = \text{code}\,(\mathbb{T}(v, \varphi))\,. \tag{33}$$

For an edge-evaluated tree with bicentroid $[v_1, v_2]$, the linear code of $\mathbb{T}(\varphi)$ is equal to lesser linear code of $\mathbb{T}_1(v_1, \varphi_1)$ and $\mathbb{T}_2(v_2, \varphi_2)$, see Eq. (9). Analogs of Theorems 2

and 3 are also satisfied for edge-evaluated trees and rooted trees. That is, if a pair of these entities has the same linear codes, then they are isomorphic. Algorithm 1 may be simply modified in such a form that it provides not only a "localization" of centroid/bicentroid but also the corresponding linear codes.

Let $\mathscr{R}_p^{[i]}$ be the class of all possible mutually nonisomorphic edge-evaluated rooted trees with p vertices and with unsaturation equal to i; its cardinality – number of elements will be denoted by $r_p^{[i]}$. The approach of reconstruction of a rooted tree from its first generation rooted trees described previously is not fully applicable for edge-evaluated rooted trees. In particular, the expression (*11*) does not contain necessary information about the multiplicities of edges incident with the root v. Hence, in order to use also an analog of Eq. (*11*) for the reconstruction of edge-evaluated rooted trees, the multiplicities should be prescribed in a proper form. Let $s_1, s_2, \ldots, s_q$ be the prescribed multiplicities, then the edge-evaluated rooted tree $\mathbb{T}(v, \varphi)$ reconstructed by Eq. (*11*) has vertex and edge sets determined by Eqs (*12a–12c*), and moreover, the mapping φ and the unsaturation are determined by

$$\varphi(e) = \begin{cases} \varphi_i(e), & (\text{for } e \in E_i) \\ s_i, & (\text{for } e = [v, v_i]) \end{cases} \tag{34a}$$

$$\text{unsat}\,(\mathbb{T}(v, \varphi)) = \sum_{i=1}^{q} [\text{unsat}\,(\mathbb{T}_i(v_i, \varphi_i)) + s_i - 1]\,. \tag{34b}$$

The relation (*34a*) means that the evaluation of $\mathbb{T}(v, \varphi)$ is formed as extensions of $\varphi_1, \varphi_2, \ldots, \varphi_q$, and the created edges $[v, v_1]$, $[v, v_2] \ldots, [v, v_q]$ are evaluated by the prescribed values $s_1, s_2, \ldots, s_q$. The relation (*34b*) determines the unsaturation of $\mathbb{T}(v, \varphi)$ as the sum of unsaturations of the first-generation subtrees plus the unsaturations of created edges.

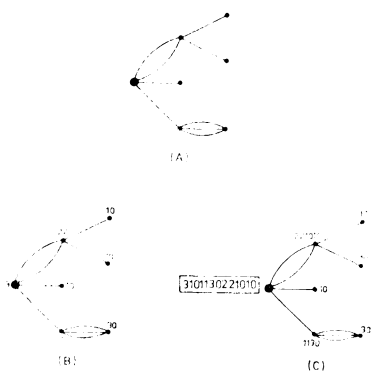

FIG. 9

Illustrative example of the construction of linear code for rooted trees with multiedges. The rooted tree A is composed of seven vertices, one double edge, and one triple edge. The evaluation of its vertices is displayed in B. Each evaluation is composed of two entries, the first one corresponds to the multiplicity of an edge incident with the vertex and the second one is determined in the same way as for simple trees with unevaluated edges. The resulting linear code (placed at the block) is constructed successively going from the marginal vertices to the root

Theorem 8. Let $\mathbb{T}_1(v_1, \varphi_1)$, $\mathbb{T}_2(v_2, \varphi_2)$, ..., $\mathbb{T}_q(v_q, \varphi_q)$ be edge evaluated rooted trees, v be a vertex not belonging to these rooted trees, and $s_1, s_2, \ldots, s_q$ a sequence of positive integers. Then the expression (*11*) determines unambiguously (up to an isomorphism) an edge-evaluated q-nary rooted tree $\mathbb{T}(v, \varphi)$ with mapping φ and unsaturation determined by Eqs (*34a–34b*).

This theorem represents a generalization of Theorem 4; it is suitable for a reconstruction of edge-evaluated trees from smaller rooted trees with specified unsaturations and multiplicities of created edges.

Let us consider a subclass $\mathscr{R}_{p,q}^{[i]} \subseteq \mathscr{R}_p^{[i]}$ composed of q-nary edge-evaluated rooted trees with p vertices and with unsaturation equal to i. Following the Theorem 8, the rooted trees of $\mathscr{R}_{p,q}^{[i]}$ are exhaustively constructed by making use of their first generation predecessors, i.e. edge-evaluated rooted trees with smaller number of vertices than p. The integer $(p - 1)$ (where $p \geqq 2$) and i are simultaneously decomposed at the form of *distinct* ordered triples,

$$(a, j, r)^{\alpha} (b, k, s)^{\beta} \ldots , \tag{35}$$

where the first entries $1 \leqq a < b < \ldots < p - 1$ correspond to the number of vertices, the second entries $0 \leqq j, k, \ldots$ determine the unsaturations, and the third entries $1 \leqq r, s, \ldots$ are the prescribed multiplicities of the created edges. The "exponents" $1 \leqq \alpha, \beta, \ldots$ determine the number of appearance of single triples. All these entities are restricted by the following set of conditions:

$$\alpha a + \beta b + \ldots = p - 1 , \tag{36a}$$

$$\alpha + \beta + \ldots = q , \tag{36b}$$

$$(j + r - 1)\,\alpha + (k + s - 1)\,\beta + \ldots = i . \tag{36c}$$

The first two conditions are obvious, they mean that an edge-evaluated q-nary rooted tree with p vertices is constructed. The last condition (*36c*) restricts the possible unsaturations of subtrees and multiplicities of the created edges in a way that the constructed rooted tree is of the prescribed unsaturation i. For instance, if $p - 1 = 5$, $q = 3$, and $i = 1$ we get six decompositions: $(1, 0, 1)^2\,(3, 0, 2)^1$, $(1, 0, 1)^1\,(1, 0, 2)^1\,(3, 0, 1)^1$, $(1, 0, 1)^2\,(3, 1, 1)^1$, $(1, 0, 1)^1\,(2, 0, 1)^1\,(2, 0, 2)^1$, $(1, 0, 2)^1\,(2, 0, 1)^2$, and $(1, 0, 1)^1\,(2, 0, 1)^1\,(2, 1, 1)^1$. The first decomposition $(1, 0, 1)^2\,(3, 0, 2)^1$ means that edge-evaluated ternary rooted trees are constructed from three subtrees composed of 1, 1, and 3 vertices, respectively, these subtrees are of zero unsaturation, and finally, the first two subtrees (with 1 vertex) are connected with the root by single edges whereas the third subtree (with 3 vertices) is connected with the root by a double edge. The decomposition $(a, j, r)^{\alpha} (b, k, s)^{\beta} \ldots$ restricted by Eqs (*36a–36c*)

means that an edge-evaluated q-nary rooted tree is constructed such that we use α subtrees with a vertices and with unsaturation equal to j; their roots are adjacent to the root v by r-tuple edges. The remaining terms in the decomposition are interpreted similarly. The number of all possible contributions from this first term is equal to $[r_a^{[j]\alpha}]$. The total number of produced rooted trees, corresponding to the integer $r_p^{[i]}$, is determined by

$$r_p^{[i]} = \sum_{q=1}^{p-1} \sum_{a^{[j]\alpha} b^{[k]\beta} \ldots} [r_a^{[j]\alpha}] [r_b^{[k]\beta}] \ldots , \tag{37}$$

where the second summation runs over all possible distinct decompositions $(a, j, r)^\alpha (b, k, s)^\beta \ldots$ that are restricted by Eqs (*36a–36c*). Hence, in order to enumerate the rooted trees with p vertices and with unsaturation i, i.e. the number $r_p^{[i]}$, we have to know their preceding values $r_{p'}^{[i']}$, for $1 \leqq p' < p$ and $0 \leqq i' \leqq i$. Simple illustrative examples of the constructive enumeration of edge-evaluated rooted trees are given in Fig. 10. If we restrict the valences of vertices from above by 4, then the produced rooted trees correspond to monosubstituted acyclic hydrocarbons (or radicals) with multiple bonds.

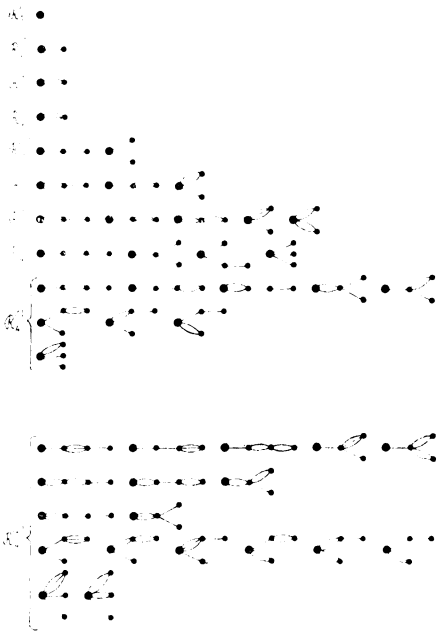

FIG. 10

The rooted trees composed of up to four vertices and with the unsaturation of up to two

Now we are ready to pay our attention to the constructive enumeration of trees with multiedges. We have to distinguish between two different cases, represented by trees with odd or even number of vertices (see Section 5). We shall not repeat all the details of this process, it has already been outlined in Section 5, it is only necessary to introduce some additional specifications of rooted trees (e.g. their unsaturation and multiplicities of created edges, see the first part of this section). Analogs of Theorems 5, 6, and 7 are simply constructed, the concept of unsaturation is implanted into the theory in the same way as for rooted trees in the first part of this section. Moreover, the formulae (*18*) and (*21*) for numerical (i.e. nonconstructive) enumeration of trees with centroids and multiedges are formally correct with small modification in the second summation, which is now running over all possible distinct decompositions of the type (*35*). The formula (*23*) counting the trees with bicentroid should be slightly generalized as follows: We shall separately consider two distinct decompositions $(a, j, s)^2$ and $(a, j, s)^1\,(a, j', s)^1$, where in the second decomposition the first and the third component must be equal (since these components determine the number of vertices in subtrees and the multiplicity of edge which was created by linking the roots of corresponding rooted trees, respectively). The above decompositions should be restricted by $2a = p$, $2j + s - 1 = i$ and $2a = p$, $j \neq j'$, $j + j' + s - 1 = i$, respectively. Then the number of trees composed of $p = 2k$ vertices with bicentroid is

$$t_{\mathrm{p}}^{[\mathrm{i}](\mathrm{bc})} = \sum_{s \geq 1} \left[r_{\mathrm{a}}^{[\mathrm{j}]2} \right] + \sum_{s \geq 1} \sum_{j \neq j'} r_{\mathrm{a}}^{[\mathrm{j}]} r_{\mathrm{a}}^{[\mathrm{j}']} , \qquad (39)$$

where the first (second) term on the r.h.s determines the number of trees with a bicentroid corresponding to the first (second) decomposition. The second summation in the second term runs over all possible nonnegative integers $j \neq j'$ restricted by $j + j' = 1 + i - s$; these indices determine the unsaturation of rooted subtrees used for the construction of trees with a bicentroid. The present approach for enumeration of trees with multiedges is illustrated in Fig. 11. Simple numerical enumeration of hydrocarbons with multiple bonds was carried out initially by Read[2].

7. FURTHER GENERALIZATIONS

The method outlined in the first part of this communication allows us to construct both rooted trees and trees. Its generalization towards accounting of the multiedges was also already discussed in the previous section. The multiedges were covered in the concept of unsaturation. This nonnegative integer corresponds to the sum of edge multiplicities decreased by 1, i.e. an actual value of unsaturation corresponds to several different cases with specific numbers of double, triple, ... edges. For some special application it is more important to construct trees with prescribed distribution of edge multiplicities. Of course, such a version of our constructive enumeration

requires deeper description of trees which are to be constructed. Therefore, the trees should be now described by some additional entries specifying the number of double, triple, ... edges. The edge multiplicity distribution is simply determined as an ordered n-tuple of nonnegative integers,

$$\boldsymbol{i} = (i_1, i_2, \ldots), \tag{40}$$

where the u-th entry i_u corresponds to the number of edges with multiplicity equal to $(u + 1)$. For instance, let us consider a rooted tree composed of p vertices and with the edge multiplicity distribution determined by $\boldsymbol{i}$, then an analog of Eq. (35) may now look as follows:

$$(a, \boldsymbol{j}, r)^{\alpha} (b, \boldsymbol{k}, s)^{\beta} \ldots . \tag{41}$$

The n-tuples $\boldsymbol{j}$, $\boldsymbol{k}$, ... specify the edge multiplicity distributions of the corresponding subtrees and are restricted by integers $r, s, \ldots$ in such a way that the overall edge multiplicity distribution $\boldsymbol{i}$ remains unchanged. The concept of edge multiplicity distributions is very appropriate for a generalization of the constructive enumeration of tree with multiedges presented in the previous section, the used theoretical tools need only slight modification without the necessity to introduce any additional theoretical concepts.

Next generalization of the present theory consists in an evaluation of vertices by alphabetic symbols (which correspond to atomic symbols). We shall also assume that this vertex evaluation determines maximal valences of vertices depending on the used alphabetic symbol. Such generalization gives, in fact, the trees with straightforward 1–1 correspondence to heteroatomic acyclic molecules composed not only of carbon atoms but also of the heteroatoms (e.g. nitrogen, oxygen, etc.). A class

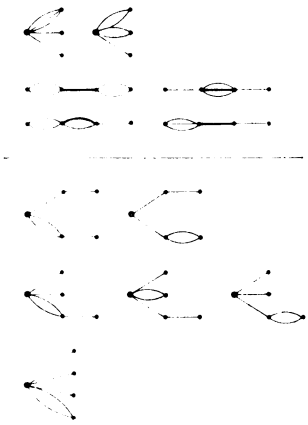

FIG. 11

The upper part of the figure contains trees composed of four vertices and with unsaturation equal to 2. The bottom part contains trees composed of five vertices and unsaturation 1. The second and third rows contain trees with bicentroids (bold line), other rows contain trees with a centroid (heavy dot)

of such trees is determined simultaneously by an edge multiplicity distribution as well as by a distribution which determines how many vertices are evaluated by the same alphabetic atomic symbol with predetermined maximal valence. First problem which should be solved is an extension of the linear code approach to trees with evaluated vertices (and edges). Let us consider a tree $\mathbb{T}(V, E)$, its vertices and edges are evaluated by

$$\varphi : E \to \{1, 2, 3, \ldots\} , \tag{42a}$$

$$\omega : V \to \mathscr{V} , \tag{42b}$$

where the mapping φ evaluates the edges by multiplicities and the mapping ω evaluates the vertices by atomic symbols from the *vocabulary* $\mathscr{V}$. We assume that this second evaluation also determines the maximal valences of evaluated vertices. For simplicity, we shall postulate that the vocabulary $\mathscr{V}$ is composed of one-term alphabetic symbols, e.g. C, N, O, ..., where the maximal valence of vertex evaluated by C (N, O, ...) is four (tree, two, ...). The tree $\mathbb{T}(V, E)$ with vertices and edges evaluated by mappings φ and ω, respectively, is expressed by an ordered quadruple,

$$\mathbb{T}(\varphi, \omega) = (V, E, \varphi, \omega) . \tag{43}$$

In a similar way we introduce the notion of rooted tree with evaluated vertices and edges. Let $v \in V$ be a vertex of $\mathbb{T}(V, E)$. If we classify this vertex as the root, then the corresponding rooted tree is

$$\mathbb{T}(v, \varphi, \omega) = (V, E, v, \varphi, \omega) . \tag{44}$$

Let us consider a rooted tree $\mathbb{T}(v, \varphi, \omega)$. Deleting from this rooted tree its root v, which is incident with q vertices $\{v_1, v_2, \ldots, v_q\} \subseteq V$, we get the first generation vertex and edge-evaluated rooted trees $\mathbb{T}_1(v_1, \varphi_1, \omega_1)$, $\mathbb{T}_2(v_2, \varphi_2, \omega_2)$, ..., $\mathbb{T}_q \cdot (v_q, \varphi_q, \omega_q)$. The mappings $\varphi_1, \varphi_2, \ldots, \varphi_q$ and $\omega_1, \omega_2, \ldots, \omega_q$ are simple restrictions of the original mapping φ and ω, respectively, with respect to the vertex and edge sets $V_1, V_2, \ldots, V_q$ and $E_1, E_2, \ldots, E_q$ (cf. Eq. (29)).

The linear codes of vertex and edge-evaluated rooted trees are constructed by a similar way as in Section 6 for the edge-evaluated rooted tree, the so-called enlarged code of the rooted tree $\mathbb{T}_i(v_i, \varphi_i, \omega_i)$ is determined by

$$\overline{\text{code}}\,(\mathbb{T}_1(v_i, \varphi_i, \omega_i)) = \text{'}\omega(v_i)\text{'} + \overline{\text{code}}\,(\mathbb{T}_i(v_i, \varphi_i, \omega_i)) , \tag{45}$$

where $\overline{\text{code}}\,(\mathbb{T}_i(v_i, \varphi_i, \omega_i))$ is the enlarged code of rooted tree $\mathbb{T}_i(v_i, \varphi_i, \omega_i)$ determined by Eq. (30). It means that this enlargement of linear code consists in an addition of a two-term substring, composed of the atomic symbol of root and the multi-

plicity of edge $[v, v_i]$. The overall linear code of rooted trees is then formed by an analog of Eq. (*31*), where the substring 'q' is now substituted by '$\omega(v)\, q$', i.e. it is extended by the atomic symbol of the root. The procedure of construction of linear codes is recursively repeated until the produced first generation rooted trees become composed of only one vertex (marginal vertices). We assign to these marginal vertices a two-term linear code '$\omega(v)\, 0$', where $\omega(v)$ is atomic symbol of a marginal vertex v. Table I lists all possible trees composed of six vertices and with special vertex evaluation (hydrogens are added subsequently).

8. CONCLUSIONS

The general theory of constructive enumeration of rooted trees and trees may be simply generalized and/or modified in such a way that it handles acyclic molecular structures possessing multiple bonds and heteroatoms. As a byproduct of the theory simple formulae for numerical enumeration of (rooted) trees were obtained. Their

TABLE I

All acyclic molecules with empirical formula $C_3H_3N_3$ and with unsaturation equal 4. The constructed molecules contain either four double bonds, or two double bonds and one triple bond, or finally, two triple bonds

No.	Formula	No.	Formula
1	NH=C=C=C=N—NH_2	21	NH=N—N=CH—C≡CH
2	NH=N—N=C=C=CH_2	22	NH=C=C=CH—N=NH
3	CH≡C—N=C=N—NH_2	23	CH≡C—N=CH—N=NH
4	NH=C=N—N=C=CH_2	24	N≡C—CH=CH—N=NH
5	NH=C=N—C≡C—NH_2	25	N≡C—N=C=CH—NH_2
6	NH=C=C=N—N=CH_2	26	NH=C=N—CH=C=NH
7	NH=N—C≡C—N=CH_2	27	N≡C—NH—CH=C=NH
8	CH≡C—N=N—N=CH_2	28	CH_2=N—CH=N—C≡N
9	N≡C—CH=N—N=CH_2	29	NH=CH—CH=N—C≡N
10	N≡C—CH=C=N—NH_2	30	N≡C—N=N—CH=CH_2
11	NH=C=C=N—CH=NH	31	CH_3—N(—C≡N)$_2$
12	NH=N—C≡C—CH=NH	32	CH_2=N—C(—C≡N)=NH
13	CH≡C—N=N—CH=NH	33	NH=CH—C(—C≡N)=NH
14	N≡C—CH=N—CH=NH	34	NH_2—N(—C≡N)—C≡CH
15	CH_2=C=N—NH—C≡N	35	CH=C(—C≡N)—N=NH
16	NH_2—C≡C—NH—C≡N	36	NH=C=C(—C≡N)—NH_2
17	N≡C—N=C=N—CH_3	37	NH=N—C(—C≡CH)=NH
18	NH=C=N—NH—C≡CH	38	NH=C=N—CH_2—C≡N
19	N≡C—NH—NH—C≡CH	39	N≡C—NH—CH_2—C≡N
20	N≡C—C≡C—NH—NH_2	40	NH_2—CH(—C≡N)$_2$

derivation is based purely on simple combinatorial considerations. Of course, they may be derived very elegantly by making use of the Polya theorem[20,21]. But their forthcoming generalization to cover more complicated tree systems of chemical interest (in particular, multiple edges, heteroatoms, restrictions of different kind imposed on valences of vertices, etc.) usually involves many formal difficulties. Furthermore, the enumeration methods based on Polya theorem give only numbers of isomers, a result of very limited importance and little impact to chemistry.

The theory of constructive enumeration of trees presented in this communication represents a unified approach for exhaustive construction of acyclic molecular systems. The approach can be simply modified and/or extended to produce trees with specifically restricted topology. It may be understood as a graph-theoretical and combinatorial formulation of the Lederberg's[4,5] intuitively formulated constructive enumeration of acyclic molecules. In this approach the central role plays the concept of centroid and bicentroid unambiguously determined according to Jordan theorem (see Theorem 1). Then, having an appropriate method for construction of linear codes of (rooted) trees, the constructive enumeration is relatively simple and straightforward task. It means, that under the term "constructive enumeration" we understand a sequential construction of linear codes of (rooted) trees from the already known linear codes of rooted trees containing substantially less vertices than those ones just constructed. This construction may be formally omitted and give us thus a simple method for numerical enumeration of (rooted) trees restricted by specific requirements.

The mentioned restrictions imposed on the produced (rooted) trees are of simple nature, e.g. number of double and/or triple edges, maximal valence of vertices, etc. For constructive enumeration of acyclic molecules with importance for structure elucidation there are much more complex and diverse restrictions and requirements specifying acyclic subgraphs that are either forbidden or required for constructed molecules. The only known method[6] that is able to implant these restrictions and requirements, involves a check in the course of constructive enumeration of rooted trees as well as trees, whether the forbidden/required structures are or are not subgraphs of just constructed (rooted) tree. If the answer is positive in the case of forbidden structures, then the given checked (rooted tree) is eliminated from the process of constructive enumeration. In the case of a required structure we take a note on the appearance of the substructure and we continue the process. We see that this straightforward approach may be, for larger molecular systems, picturesquely ineffective, since it needs a very fast method for finding of subgraphs in acyclic molecular structures. Above the general graph-theoretical formulation of the problem of elimination of forbidden substructures and/or accounting for only the molecules with required substructures (together with their number of appearance) though conceptually very simple, represents a very serious problem of all constructive enumerations and remains yet to be solved.

REFERENCES

1. Sheley C. A., Woodruff H. B., Snelling C. R., Munk M. E. in: *Computer Assisted Structure Elucidation. A.C.S. Symposium Series* (D. H. Smith, Ed.), Vol. 54, p. 92. American Chemical Society, Washington, D.C. 1977.
2. Balaban A. T. (Ed.): *Chemical Applications of Graph Theory.* Academic Press, London 1976.
3. Gray N. A. B.: *Computer Assisted Structure Elucidation.* Wiley, New York 1986.
4. Lederberg J.: *Computation of Molecular Formulas for Mass Spectrometry.* Holden-Day, San Francisco 1964.
5. Lederberg J., Sutherland G. L., Buchanan B. G., Feigenbaum E. A., Robertson A. V., Duffield A. M., Djerassi C. J.: J. Am. Chem. Soc. *91*, 2973 (1969).
6. Lindsay R. K., Buchanan B. G., Feigenbaum E. A., Lederberg J.: *Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project*, p. 48. McGraw-Hill, New York 1980.
7. Henze H. R., Blair C. M.: J. Am. Chem. Soc. *53*, 3042 (1931).
8. Henze H. R., Blair C. M.: J. Am. Chem. Soc. *53*, 3077 (1931).
9. Jordan C.: Reine Angew. Math. *70*, 185 (1869).
10. Harary F.: *Graph Theory.* Addison Wesley, Reading 1969.
11. Read R. C. in: *Graph Theory and Computing* (R. C. Read, Ed.), p. 153. Academic Press, New York 1972.
12. Knop J. V., Müller W. R., Szymanski K., Trinajstić N.: *Computer Generation of Certain Classes of Molecules.* SKTH/Kemija u industrii, Zagreb 1985.
13. Sheley C. A., Hays T. R., Roman R. V., Munk M. E.: Anal. Chim. Acta *103*, 121 (1978).
14. Carhart R. E., Smith D. H., Brown H., Djerassi C.: J. Am. Chem. Soc. *97*, 5755 (1975).
15. Carhart R. E., Smith D. H., Gray N. A. B., Nourse J. G., Djerassi C.: J. Org. Chem. *46*, 1708 (1981).
16. Munk M. E., Farkas M., Lipkus A. H., Christie B. D.: Mikrochim. Acta II 1986, 199.
17. Kudo Y., Sasaki S.: J. Chem. Inf. Comput. Sci. *16*, 43 (1976).
18. Kvasnička V., Pospíchal J.: J. Chem. Inf. Comput. Sci. *30*, 99 (1990).
19. Lederberg J.: Proc. Natl. Acad. Sci. U.S.A. *53*, 134 (1965).
20. Pólya G.: Acta Math. *68*, 145 (1937).
21. Pólya G., Read R. C.: *Combinatorial Enumeration of Groups, Graphs and Chemical Compounds.* Springer, New York 1987.

Translated by the author (V.K.).