AN IMPROVED METHOD OF CONSTRUCTIVE ENUMERATION OF GRAPHS

V. KVASNIČKA and J. POSPÍCHAL
Department of Mathematics, Slovak Technical University, 81237 Bratislava, Czechoslovakia

Received 9 June 1991; revised 23 September 1991

Abstract

A canonical numbering of molecular graphs based on the maximum code assigned to the lower-triangle part of the adjacency matrix is used for a recurrent constructive enumeration of molecular graphs. The underlying theoretical concept – the so-called semicanonical numbering – is much more restrictive than the originally used concept of cooperative numbering. The method does not involve very time-consuming isomorphism checks of currently constructed graphs against those already constructed.

1. Introduction

The problem of constructive enumeration of molecular graphs is one of the very attractive problems of mathematical chemistry. In particular, it is not important only for computer-assisted structure elucidation [1,2], but also for computer-assisted synthesis design [3,4]. Many effective methods [5–14] for constructive enumeration have been elaborated and suggested. The common problem of all methods has been how to overcome the typical problem of almost all combinatorial algorithms, called *combinatorial explosion*. This problem is here manifested by the appearance of graphs in a current stage that have already been constructed in the previous stages of enumeration. Therefore, the most time-consuming process usually is checking the isomorphism of the currently constructed graph with those already constructed. From this point of view, in each well-established method of constructive enumeration it is very important to carefully study its combinatorial and graph-theoretical properties; then, as a result of these theoretical considerations, the necessary conditions of nonredundancy of created graphs are formulated.

In our recent communications [15,16], we have elaborated a recurrent constructive enumeration of molecular graphs and a canonical numbering based on the maximum code of the lower-triangle part of the adjacency matrix. It was proved that this canonical numbering of graphs is also cooperative [15]. It considerably restricts a huge number of graphs being created in the course of the so-called extension process (where from a canonically numbered graph composed of n vertices,

cooperatively numbered graphs composed of $n+1$ vertices are created). Applying the above-mentioned theoretical tools, we have arrived at an efficient method of constructive enumeration of graphs in which it is not necessary to carry out isomorphism checks of a currently constructed graph with the previously constructed graphs.

The purpose of this communication is to modify our constructive enumeration of graphs by a stronger concept, that of the so-called *semicanonical numbering* of graphs, which was initially suggested by Faradzhev [17–19] fifteen years ago. His method is based on a canonical numbering of graphs, which is produced by the maximum code of the whole adjacency matrices. Recently, we have demonstrated [20] that a similar concept may be also used when the canonical numbering is based on the maximum code of the upper-triangle part of the adjacency matrices. The resulting algorithm is surprisingly fast and simple. We believe it might be of value in modifying our original approach [15] of constructive enumeration of graphs with canonical numbering employing the lower-triangle part of the adjacency matrices by the concept of semicanonical numbering. Although the algorithm was initially introduced for canonical numbering with the upper-triangle part or the whole adjacency matrix, we will show that it may be used also for constructive enumeration based on a canonical numbering employing the lower-triangle part of the adjacency matrix. However, the whole theory may be presented in a more simplified and transparent form than its original formulation [15]. For example, it is no longer necessary to use the notion of saturated and/or unsaturated vertices, which has played an important role, and also the so-called *extension process* will now be conceptually simpler.

2. Basic concepts

Let $\mathcal{F}_{pq}$ (for $p > 0$ and $q \geq 0$) be a family of adjacency matrices $\mathbf{A} = (a_{ij})$. These matrices are symmetric, with p rows and columns (we say that their types are $t(\mathbf{A}) = (p, p)$), their entries are zero and/or unit digits, the diagonal entries are zero, and their nondiagonal entries are restricted by

$$\sum_{i<j} a_{ij} = q. \tag{1}$$

Each adjacency matrix $\mathbf{A} \in \mathcal{F}_{pq}$ unambiguously represents a numbered graph [15] G composed of p vertices and q edges. In our forthcoming considerations, we shall always assume that its vertex set $V(G)$ is composed of the first p natural integers, i.e. $V(G) = \{1, 2, \ldots, p\}$. An edge set $E(G)$ is composed of unordered pairs $[i, j]$ of integers from $V(G)$, where $[i, j] \in E(G) \Leftrightarrow a_{ij} = 1$.

A permutation $R = (r_1, r_2, \ldots, r_p)$ of p objects $(1, 2, \ldots, p)$ is uniquely represented by the so-called permutation matrix [21] composed of zero and unit entries, and each row/column contains just one unit entry. A set (called the symmetry group) of all permutation matrices of p objects will be denoted by S_p.

DEFINITION 1

Two adjacency matrices $\mathbf{A}_1, \mathbf{A}_2 \in \mathcal{F}_{pq}$ are called *equivalent*, $\mathbf{A}_1 \cong \mathbf{A}_2$, if there exists a permutation matrix $\mathbf{R} \in \mathcal{S}_p$ such that

$$\mathbf{A}_1 = \mathbf{R}^{\mathrm{T}}\mathbf{A}_2\mathbf{R}. \tag{2}$$

Two equivalent matrices $\mathbf{A}_1$ and $\mathbf{A}_2$ are represented by isomorphic graphs G_1 and G_2 ($G_1 \cong G_2$). The family $\mathcal{F}_{pq}$ may be decomposed into disjoint subfamilies that are composed of equivalent matrices, $\mathcal{F}_{pq} = \mathcal{F}_{pq}^{(1)} \cup \mathcal{F}_{pq}^{(2)} \cup \ldots$, where $\mathbf{A}_1, \mathbf{A}_2 \in \mathcal{F}_{pq}^{(i)} \Leftrightarrow \mathbf{A}_1 \cong \mathbf{A}_2$.

We assign to each adjacency matrix $\mathbf{A} \in \mathcal{F}_{pq}$ a *string* composed of its lower-triangle entries:

$$[\mathbf{A}] = (a_{21}a_{31}a_{32} \ldots a_{p1}a_{p2} \ldots a_{p,p-1}); \tag{3}$$

the string contains $p(p-1)/2$ characters "0" and/or "1", and is formally considered a binary number, i.e. we assign to each adjacency matrix a binary number. According to this interpretation of strings, they may be mutually related by the following three relations: "equal to", "smaller than", and "greater than", i.e. $[\mathbf{A}_1] = [\mathbf{A}_2]$, $[\mathbf{A}_1] < [\mathbf{A}_2]$, and $[\mathbf{A}_1] > [\mathbf{A}_2]$, respectively.

Each subfamily $\mathcal{F}_{pq}^{(i)} \subset \mathcal{F}_{pq}$ is represented by its *code* determined as the maximum value of all strings of matrices from $\mathcal{F}_{pq}^{(i)}$,

$$\mathrm{code}_i = \max_{\mathbf{A} \in \mathcal{F}_{pq}^{(i)}} [\mathbf{A}]. \tag{4}$$

This type of code was initially introduced by the present authors [15, 16] as a valid theoretical tool for constructive enumeration of graphs. Definition (4) may be alternatively rewritten as follows:

$$\mathrm{code}_i = \max_{\mathbf{R} \in \mathcal{S}_p} [\mathbf{R}^{\mathrm{T}}\mathbf{A}\,\mathbf{R}], \tag{5}$$

where $\mathbf{A}$ is an arbitrary adjacency matrix of $\mathcal{F}_{pq}^{(i)}$.

DEFINITION 2

Adjacency matrix $\bar{\mathbf{A}} \in \mathcal{F}_{pq}$ with the code determined by

$$\mathrm{code}_i = [\bar{\mathbf{A}}] = \mathrm{code}\,(\bar{\mathbf{A}}) \tag{6}$$

is called the *canonical adjacency matrix*. A graph determined by a canonical adjacency matrix is called the *canonically numbered graph*.

For a fixed s, bounded by $1 \leq s < p$, an adjacency matrix $\mathbf{A} \in \mathcal{F}_{pq}$ can be decomposed into block matrices as follows:

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{21}^{\mathrm{T}} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix}, \tag{7}$$

where $\mathbf{A}_{11}$ ($\mathbf{A}_{22}$) corresponds to the left-up (right-bottom) corner submatrix of $\mathbf{A}$, its type is $t(\mathbf{A}_{11}) = (s, s)$ [$t(\mathbf{A}_{22}) = (p - s, p - s)$]. A rectangular submatrix $\mathbf{A}_{21}$ of the type $t(\mathbf{A}_{21}) = (p - s, s)$ will be expressed via its s-dimensional row vectors,

$$\mathbf{A}_{21} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{p-s} \end{pmatrix}. \tag{8}$$

Let $\boldsymbol{a} = (a_i)$ and $\boldsymbol{b} = (b_i)$ be two s-dimensional row vectors; they are equal, $\boldsymbol{a} = \boldsymbol{b}$, if $a_i = b_i$, for $i = 1, 2, \ldots, s$. These vectors may also be related by $\boldsymbol{a} > \boldsymbol{b}$ ($\boldsymbol{a} < \boldsymbol{b}$) if there exists an integer $1 \leq i \leq s$ such that $a_j = b_j$, for $j = 1, 2, \ldots, i - 1$ and $a_i > b_i$ ($a_i < b_i$).

DEFINITION 3

An adjacency matrix $\mathbf{A} \in \mathcal{F}_{pq}$ is called *semicanonical*, if for each $1 \leq s < p$ the row vectors $\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_{p-s}$ of the submatrix $\mathbf{A}_{21}$ satisfy

$$\boldsymbol{c}_1 \geq \boldsymbol{c}_2 \geq \ldots \geq \boldsymbol{c}_{p-s}. \tag{9}$$

A graph determined by a semicanonical matrix is called the *semicanonically numbered graph.*

The above notion of a semicanonical adjacency matrix represents a crucial point of our constructive enumeration of graphs. It enables the formulation of a necessary condition of canonicity of adjacency matrices.

THEOREM 1

A canonical adjacency matrix is semicanonical.

Let us consider an adjacency matrix $\mathbf{A} \in \mathcal{F}_{pq}$ and let us assume that for a fixed $1 \leq s < p$ the row vectors $\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_{p-s}$ of the rectangular submatrix $\mathbf{A}_{21}$ do not satisfy condition (9). For instance, the vectors $\boldsymbol{c}_i$ and $\boldsymbol{c}_j$ (for $1 \leq i < j \leq s$) satisfy $\boldsymbol{c}_i < \boldsymbol{c}_j$; then there should exist a permutation matrix $\mathbf{R}$ assigned to a transposition of indices i and j such that $[\mathbf{R}^{\mathrm{T}}\mathbf{A}\,\mathbf{R}] > [\mathbf{A}]$. We have proved that an adjacency matrix which is not semicanonical cannot be canonical; a reverse form of this implication is given by theorem 1.

Theorem 1 represents a strong restrictive necessary condition of the canonicity of adjacency matrices. If we are looking for a canonical matrix in the subfamily $\mathcal{F}_{pq}^{(i)}$, then it is entirely sufficient to consider only those matrices of $\mathcal{F}_{pq}^{(i)}$ that are semicanonical; all others are omitted.

Let us assume that for some fixed $1 \leq s < p$ the adjacency matrix will be written in the form (7). What is now very important is that the submatrix $\mathbf{A}_{11}$ is canonical if the original adjacency matrix is also canonical. If the adjacency matrix $\mathbf{A}$ corresponds to a graph G, then the submatrix $\mathbf{A}_{11}$ corresponds to a subgraph $G_{11} \subset G$ created from G by deleting all vertices numbered by $s+1, s+2, \ldots, p$.

THEOREM 2

If an adjacency matrix $\mathbf{A} \in \mathcal{F}_{pq}$ is canonical, then for each $1 \leq s < p$ the corresponding submatrix $\mathbf{A}_{11}$ is also canonical.

The proof [15] may be carried out in a similar fashion as the proof of theorem 1.

3. Canonical numbering of graphs

The concept of canonical numbering of a graph G, represented by an adjacency matrix $\mathbf{A} \in \mathcal{F}_{pq}^{(i)}$, requires one to find a permutation matrix $\mathbf{R} \in \mathcal{S}_p$ such that, cf. (6),

$$\text{code}_i = [\mathbf{R}^T \mathbf{A}\, \mathbf{R}]. \tag{10}$$

If the graph G is canonically numbered, then the above relation is achieved automatically for an identical permutation $\mathbf{R} = \mathbf{E}$; all other permutation matrices that satisfy (10) correspond to automorphisms of the graph. For our constructive enumeration of graphs, we only need to know whether a graph is canonically numbered or not. If we find a matrix $\mathbf{R}$ such that $[\mathbf{R}^T \mathbf{A}\, \mathbf{R}] > [\mathbf{A}]$, then the graph G is not canonically numbered. The main effort in finding a canonical numbering of graphs should be concentrated on a process of achieving condition (10) by making use of only those permutation matrices $\mathbf{R} \in \mathcal{S}_p$ that lead as quickly as possible to the required result. A vertex of G which in this process will be numbered by 1 should belong to a vertex subset $V_{\text{prior}}(G) \subseteq V(G)$ composed of the vertices that induce a maximal clique (or cliques) in G (see theorem 6 in [15]); all other vertices will give adjacency matrices with the first few rows smaller than the ones of the matrix $\bar{\mathbf{A}}$ corresponding to the maximum code$(\bar{\mathbf{A}})$, see (6). This process is considerably accelerated by theorem 1; a canonical numbering may be successfully constructed by the trial and error method (implemented as a backtrack searching algorithm with a branch and bound modification [22]) in such a way that the corresponding block matrices $\mathbf{A}_{11}$ are canonical. Moreover, only those permutations $\mathbf{R} \in \mathcal{S}_p$ (see theorem 1) that will produce semicanonical adjacency matrices are used. Combining these two observations, we obtain a very

efficient method of canonical numbering of graphs which, in the process of traversing the searching tree, skips all situations where there is no chance of achieving the canonical numbering.

3.1. COOPERATIVE NUMBERING [15,24,25]

In our previous communication [15], we have used the concept of *cooperative numbering* of molecular graphs as a proper tool restricting the process of construction of canonical numbering. In its original formulation, the numbering has been done in a slightly vague manner. Therefore, for the purpose of our forthcoming considerations (mainly for a comparison of its efficiency with the semicanonical numbering), it will be worthwhile to specify this numbering by an exact set-theoretical formalism.

The method of cooperative numbering is determined by the following recurrent procedure:

(1) In the initialization (zero) step, two sets Y_1 and Y_2 are constructed, $Y_1 = \{v\}$ and $Y_2 = V(G) \setminus Y_1$, where $v \in V(G)$ is an arbitrary vertex of G.

(2) In the ith step ($i = 1, 2, \ldots, p$), an arbitrary vertex $v \in Y_1$ is numbered by i; formally, we put $r_i = v$. From the current sets $Y_1, Y_2, \ldots, Y_t$, we form auxiliary sets

$$Y_1 = Y_1 \setminus \{v\}, \tag{11a}$$

$$Y_i' = Y_i, \qquad (\text{for } i = 1, 2, \ldots, t-1), \tag{11b}$$

$$Y_t' = Y_t \cap \Gamma(v), \tag{11c}$$

$$Y_{t+1}' = Y_t \setminus Y_t', \tag{11d}$$

where $\Gamma(v) \subseteq V(G)$ is a vertex subset composed of all vertices that are adjacent to the vertex $v \in V(G)$. Finally, from the auxiliary sets $Y_1', \ldots, Y_{t+1}'$ we form new sets $Y_1, Y_2, \ldots, Y_t$, in such a way that we take into account only nonempty ones and order them in the same manner as they appear.

The second step is recurrently repeated until all vertices are numbered (when it is impossible to create nonempty sets Y); this is achieved after p repetitions for graphs composed of p vertices. The set Y_1 serves as a stack of vertices that may be potentially numbered by i in the ith step; if this set is emptied, then the forthcoming set Y_2 will be used as Y_1, and so on. We see that sets Y_i (for $i = 2, \ldots, t-1$) remain constant until the set Y_1 is emptied, then the set Y_2 is pushed to Y_1. The set Y_t is used for the creation of new sets Y_t' and Y_{t+1}' by splitting the set according to whether its vertices are connected with the currently numbered vertex or not. The process of this numbering carried out by the present set-theoretic formalism is illustrated in fig. 1. The suggested method is summarized in the form of a pseudo-PASCAL algorithm.

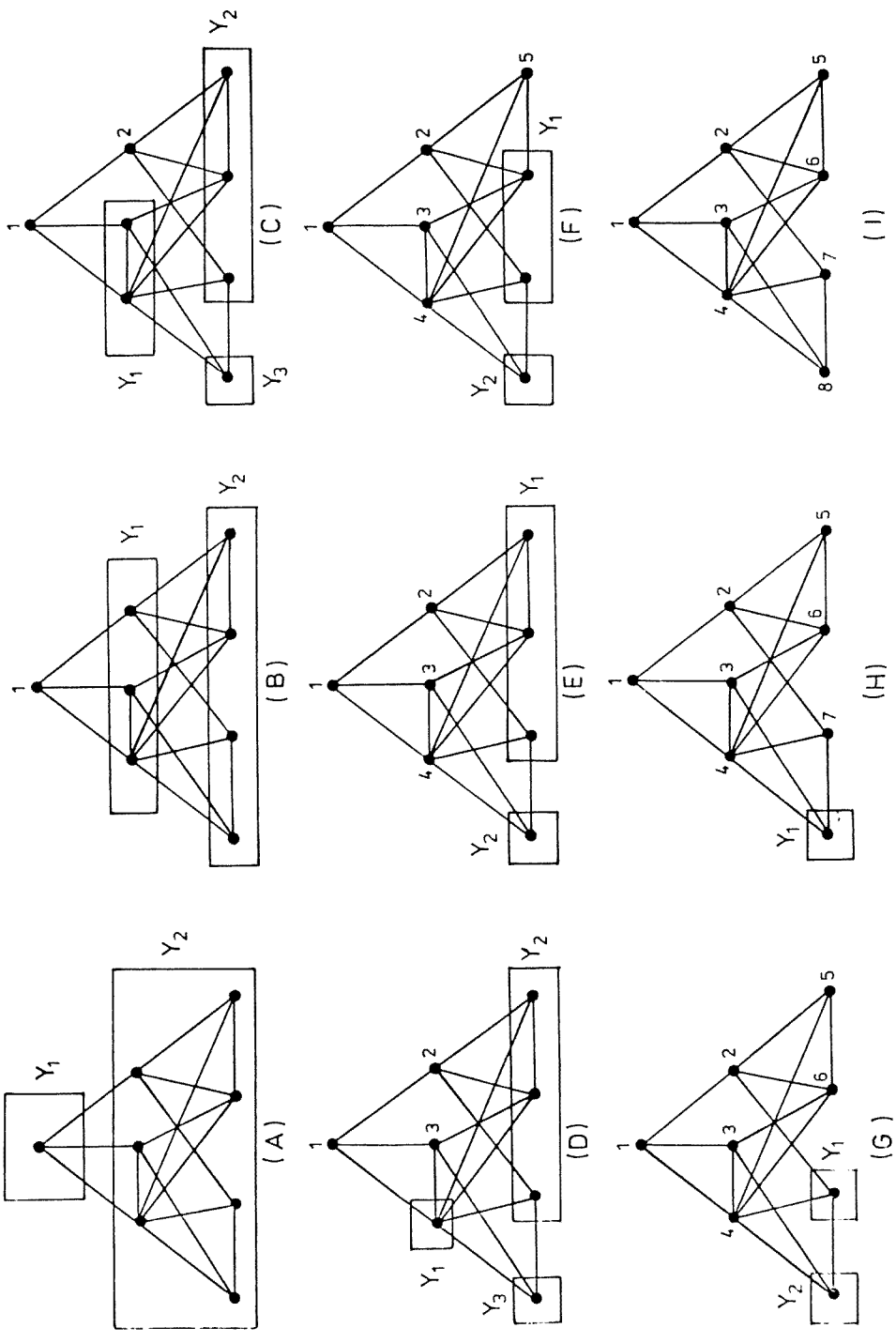


Fig. 1. An illustrative example of cooperative numbering of a graph composed of eight vertices.

ALGORITHM 1

1. $v :=$ any element of $V(G)$;
2. $Y_1 := \{v\}$; $Y_2 := V(G) \setminus Y_1$; $t := 1$; $i := 0$;
3. **WHILE** $t > 0$ **DO**
4. **BEGIN** $i := i + 1$;
5. $r_i :=$ any element of Y_1;
6. $Y_1 := Y_1 \setminus \{r_i\}$;
7. **FOR** $i := 1$ **TO** $t - 1$ **DO** $Y_i' := Y_i$;
8. $Y_t' := Y_t \cap \Gamma(r_i)$; $Y_{t+1}' := Y_t \setminus Y_t'$;
9. $t' := 0$;
10. **FOR** $i := 1$ **TO** $t + 1$ **DO**
11. **IF** $Y_i' \neq 0$ **THEN BEGIN** $t' := t' + 1$; $Y_{t'} := Y_i'$ **END**;
12. $t := t'$;
13. **END**.

3.2. SEMICANONICAL NUMBERING

Semicanonical numbering of graphs with codes formed from the upper-triangle part of adjacency matrices is discussed in our recent communication [20] devoted to an approach of constructive enumeration of graphs similar to those initially suggested by Faradzhev [17, 18]. The concept of semicanonical numbering has been introduced (see definition 3) through adjacency matrices which have to be semicanonical. This definition is not constructive; by its application, we may only check whether an adjacency matrix is semicanonical or not. For our constructive enumeration of graphs, the availability of a method which is able to construct a semicanonical numbering in a similar way as for the cooperative numbering described in the previous subsection (see algorithm 1) is of paramount importance.

Let $v \in V(G)$ be an arbitrary vertex of G chosen to be numbered by 1, i.e. we put $r_1 = v$. The vertex set $V(G)$ is divided into two disjoint subsets Y_1 and Y_2,

$$Y_1 = \Gamma(v), \quad Y_2 = (V(G) \setminus \{v\}) \setminus Y_1. \tag{12}$$

Let us assume that $(i-1)$ vertices have already been numbered by $1, 2, \ldots, i-1$ and that the sets $Y_1, Y_2, \ldots, Y_t$ have been formed. An arbitrary vertex $v \in Y_1$ is numbered by i, i.e. $r_i = v$; this is accompanied by the creation of the following auxiliary sets:

$$Y_k' = \Gamma(r_i) \cap Y_k, \qquad (\text{for } k = 1, 2, \ldots, t), \tag{13a}$$

$$Y_k'' = \begin{cases} (Y_k \setminus \{v\}) \setminus Y_k', & (\text{for } k = 1), \\ Y_k \setminus Y_k', & (\text{for } k = 2, \ldots, t). \end{cases} \tag{13b}$$

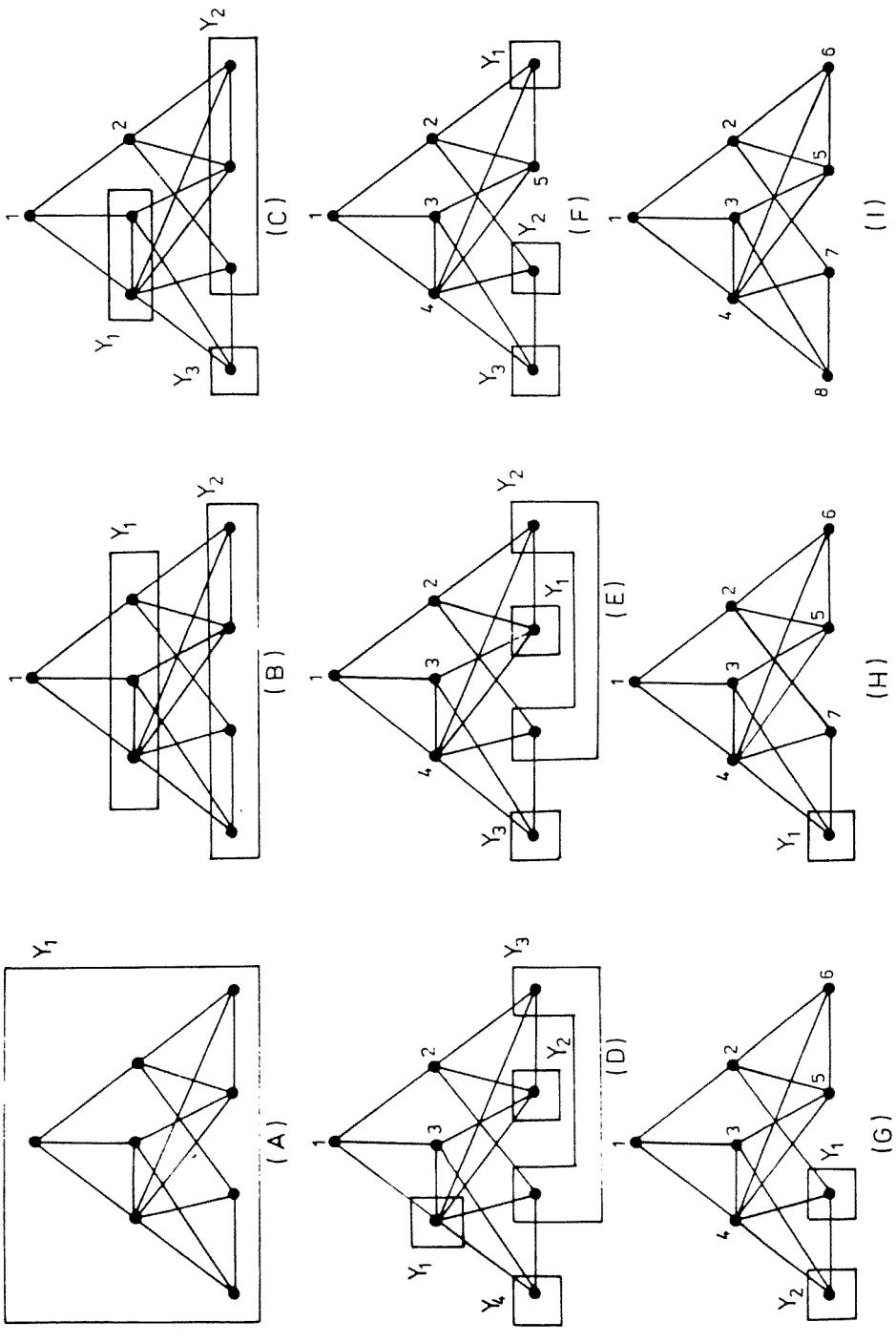


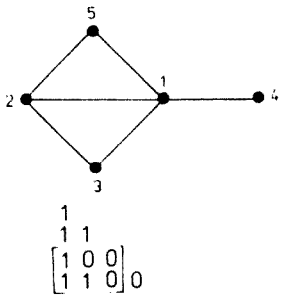
Fig. 2. An illustrative example of semicanonical numbering of the same graph as in fig. 1.

Each subset Y_k is divided into two disjoint subsets Y_k' and Y_k''. A sequence of sets $Y_1', Y_1'', Y_2', Y_2'', \ldots, Y_t', Y_t''$ is compressed into a new sequence $Y_1, Y_2, \ldots, Y_t$ in such a way that only nonempty sets are considered. The above procedure is repeated until all vertices are numbered.

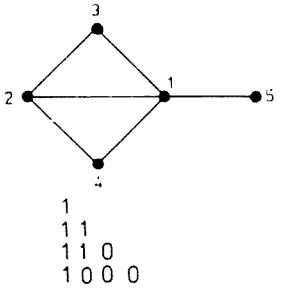
The sets $Y_1, Y_2, \ldots, Y_t$ constructed in the second part of the ith step have a simple interpretation: a set Y_i (for $i = 1, 2, \ldots, t'$) is composed of those still nonnumbered vertices that are similar [23] from the standpoint of already numbered vertices when edges between nonnumbered vertices are not considered, see fig. 2.

A verification of the above recurrent method as to whether it produces a semicanonical numbering of graphs or not can be carried out simply by tracing what its single steps are doing. The sets $Y_1, Y_2, \ldots, Y_t$ are formed according to (13a–b) in such a way that condition (9) should be fulfilled.

For a comparison of the above-introduced semicanonical and cooperative numberings of graphs, it is very important to recognize that the sets $Y_2, Y_3, \ldots, Y_{t-1}$ remain constant in the course of cooperative numbering, in contrast to sets $Y_1, Y_2, \ldots, Y_{t-1}$ in the semicanonical numbering that are in each step reconstructed with respect to the just numbered vertex. Moreover, according to the reconstruction process, the whole number of Y sets can increase more for semicanonical numbering. The reconstruction causes the set Y_1, used as a stack of the currently numbered vertices, to become progressively smaller in the forthcoming steps and also to more adequately reflect the bonding situation than the set Y_1 in the cooperative numbering. It can be easily proved that each semicanonical numbering should also be cooperative, but a reverse statement is not necessarily true, see fig. 3.



Cooperatively numbered graph



Semicanonically numbered graph

Fig. 3. An example of the statement that not each cooperatively numbered graph is also a semicanonically numbered graph. On the left-hand side, it is easy to see that the part of the adjacency matrix placed in a rectangle is not semicanonical.

Summarizing our considerations, the semicanonical numbering is much stronger than its cooperative counterpart. Its application in a process of finding canonical

numbering should lead to a faster achievment of (10) than a numbering based on the cooperative numbering.

Algorithmically, the semicanonical method in pseudo-PASCAL is formulated as follows.

ALGORITHM 2

1. $Y_1 := V(G)$; $i := 0$; $t := 1$;
2. **WHILE** $t > 0$ **DO**
4. **BEGIN** $i := i + 1$; $v :=$ an arbitrary vertex of Y_1;
5. $r_i := v$; $Y_1 := Y_1 \setminus \{v\}$;
6. **FOR** $k = 1$ **TO** t **DO**
7. **BEGIN** $Y'_k := \Gamma(r_i) \cap Y_k$; $Y''_k := Y_k \setminus Y'_k$ **END**;
8. $t' := 0$;
9. **FOR** $k := 1$ **TO** t **DO**
10. **BEGIN IF** $Y'_k \neq 0$ **THEN BEGIN** $t' := t' + 1$; $Y_{t'} := Y'_k$ **END**;
11. **IF** $Y''_k \neq 0$ **THEN BEGIN** $t' := t' + 1$; $Y_{t'} := Y''_k$ **END**;
12. **END**;
13. $t := t'$;
14. **END**.

3.3. CANONICAL NUMBERING

The method of semicanonical numbering of graphs, as outlined above, will now be advantageously used as an efficient approach for the canonical numbering. For the purpose of our constructive enumeration of graphs, we will need to know only whether an adjacency matrix is canonical or not; it means that the method may be formulated in a simpler version.

Let us consider an adjacency matrix $\mathbf{A} = (A_{ij})$ from the family $\mathcal{F}_{pq}$, and let us further look for a permutation $R = (r_1, r_2, \ldots, r_p)$ which produces a maximum code. To do this, we have to consider only those permutations (see theorem 1) that give, a priori, the semicanonical adjacency matrices, all other permutations may be automatically rejected. This means that algorithm 2 may be simply modified in such a way that it produces an algorithm for checking the canonicity of an adjacency matrix.

ALGORITHM 3

1. $Y_1^{(1)} := \{1, 2, \ldots, p\}$; $U_1 :=$ {vertices of highest priority};
   $i := 1$; $t_1 := 1$; canonicity := true;
2. **REPEAT IF** $U_i \neq \emptyset$ **THEN**
3. **BEGIN** $r_i := \min(U_i)$; $U_i := U_i \setminus \{r_i\}$; $Y_1^{(i)} := Y_1^{(i)} \setminus \{r_i\}$;

```
4.          FOR k := 1 TO t_i DO
5.          BEGIN Y'_k := Γ(r_i) ∩ Y_k^(i); Y''_k := Y_k^(i) \ Y'_k END;
6.          Y_1^(i) := Y_1^(i) ∪ {r_i}; j := 0;
7.          FOR k := 1 TO t_i DO
8.          BEGIN IF Y'_k ≠ 0 THEN BEGIN j := j + 1; Y_j^(i+1) := Y'_k END;
9.                IF Y''_k ≠ 0 THEN BEGIN j := j + 1; Y_j^(i+1) := Y''_k END
10.         END; t_{i+1} := j;
13.         IF row_i = ROW_i THEN
14.         BEGIN IF i = p THEN
15.               BEGIN i := min{k; r_k ≠ k};
16.                     FOR j := 1 TO i - 1 DO FOR k := 1 TO p DO
17.                     IF r_k < k THEN U_j := U_j \ {k};
18.               END ELSE BEGIN i := i + 1; U_i := Y_1^(i) END
19.         END ELSE IF row_i > ROW_i THEN canonicity := false
20.   END ELSE i := i - 1;
21.   UNTIL (i = 0) or (not canonicity);
```

The value of the Boolean variable "canonicity" is true (false) if the adjacency matrix is canonical (not canonical). The set U_1 (see row 1) is composed of indices that correspond to vertices of the highest priority, i.e. these vertices form a vertex subset which induces a maximal clique (or cliques). The symbol ROW_i denotes the ith row of the lower-triangle part of the checked adjacency matrix, whereas the symbol row_i denotes the same row of transformed matrix $\mathbf{R}^{\mathrm{T}}\mathbf{A}\,\mathbf{R}$ (i.e. this row is composed of the entries $A_{r_i,r_1}, A_{r_i,r_2}, \ldots, A_{r_i,r_{i-1}}$). Rows 15 to 17 contain the very efficient speeding method based on the existence of automorphisms in the verified graph. The vertices (indices) that are similar [23] to a vertex already used are removed from sets U_i, in which the index i is bounded by $i < k$, and where k is the minimum value of an index set for which $r_k \neq k$. Here, it is very important to note that this approach changes the value of index i (determining the depth of the searching tree) to a lower value, that is, a huge part of the searching tree is pruned by this approach: many branches which may give only results already achieved are skipped. In other words, the acceleration is made by pruning the searching tree, using created automorphisms. When some corresponding vertices are found, those with the lower index are removed from the set of potential candidates in levels where the sequence of indices of vertices remained unchanged from the outset. After this, we can jump to the level where for the first time the index of the vertex does not match its sequence number. By that, we have omitted the branches going from the lower levels.

4. Constructive enumeration

Let us consider a canonically numbered graph G determined by a canonical adjacency matrix $\mathbf{A} \in \mathcal{T}_{pq}$. A subgraph $G_s \subseteq G$ (for $s = 1, 2, \ldots, p$) is induced [23] by a vertex subset $V(G_s) \subseteq V(G)$ composed of the first s natural integers, $V(G_s) = \{1, 2, \ldots, s\}$.

THEOREM 3

If the graph G is canonically numbered, then all its subgraphs $G_s \subseteq G$ (for $1 \leq s \leq p$) are also canonically numbered.

This theorem (initially proved in [15]) is nothing but a rephrasing of the above theorem 2. It represents our basic idea of how to construct recurrently all possible graphs with prescribed numbers of vertices and edges. A dominant role will be played by the so-called *extension process*, specified as follows: Let $V_{\text{ext}}(G)$ be a vertex subset composed of the *extendable vertices* (in [15], where they are also determined, these vertices are called *unsaturated*). The subset may be constructed by $V_{\text{ext}}(G) = \{r, r+1, \ldots, n\}$, where the integer r gives the position of the first nonzero entry in the nth row of the adjacency matrix of G and n is equal to the number of vertices of the graph G, $|V(G)| = n$. We select a nonempty subset $V'_{\text{ext}}(G) \subseteq V_{\text{ext}}(G)$ composed of some preselected extendable vertices of G. The *extension* of G with respect to $V'_{\text{ext}}(G)$ consists of the graph G_{ext} determined by

$$V(G_{\text{ext}}) = V(G) \cup \{n+1\}, \tag{14a}$$

$$E(G_{\text{ext}}) = E(G) \cup \{[i, n+1];\ i \in V'_{\text{ext}}(G)\}, \tag{14b}$$

where $n+1$ is a new added vertex. We may say that the graph G_{ext} is constructed from the original canonically numbered graph G in such a way that a new vertex $n+1$ is connected by edges with all vertices from $V'_{\text{ext}}(G)$.

In general, the extended graph G_{ext} is not canonically numbered although its "parental" graph is. The concept of semicanonical adjacency matrices now ensures that the resulting graph G_{ext} will be semicanonically numbered. According to definition 3, the extension process should be carried out in such a way that an adjacency matrix of G_{ext} is semicanonical, i.e. the nth and $(n+1)$st rows of $\mathbf{A}_{\text{ext}}$ (assigned to G_{ext}) must satisfy $\boldsymbol{c}_n \geq \boldsymbol{c}_{n+1}$, where $\boldsymbol{c}_n$ ($\boldsymbol{c}_{n+1}$) is a row vector composed of the $(n-1)$ first entries of $\mathbf{A}_{\text{ext}}$. This simple criterion automatically ensures the semicanonicity of the created adjacency matrix $\mathbf{A}_{\text{ext}}$ and also gives an explanation why the above vertex subset $V_{\text{ext}}(G)$ was constructed only from vertices numbered by $r+1, r+2, \ldots, n$. If this subset contains a vertex numbered by i (where $1 \leq i < r$), then the adjacency matrix $\mathbf{A}_{\text{ext}}$ contains entries $a_{ni} = 0$ and $a_{n+1,i} = 1$, that is, it could not be semicanonical.

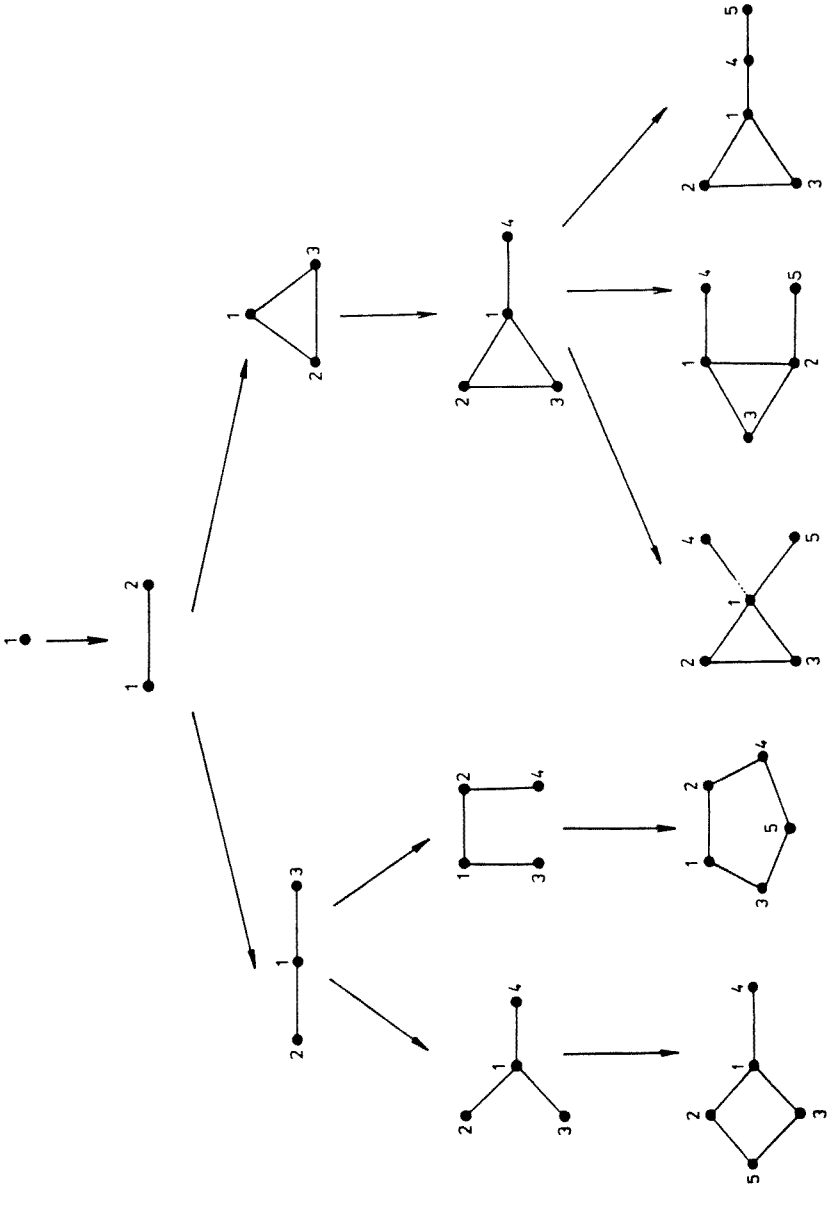


Fig. 4. An illustrative example of a recurrent construction of all connected graphs that are composed of five vertices and five edges. All graphs are numbered canonically.

THEOREM 4 [15]

For a given canonically numbered graph G (composed of n vertices), all its canonically numbered extensions $G^{(1)}_{ext}, G^{(2)}_{ext}, \ldots$ represent all possible canonically numbered graphs (composed of $n+1$ vertices) that have a subgraph induced by vertices $\{1, 2, \ldots, n\}$ which is identical with the graph G.

This theorem automatically follows from theorems 1 and 3 and from the process of extension. It enables us to suggest a very simple and effective method for constructive enumeration of all possible graphs with the prescribed numbers of edges and vertices. Its effectiveness is based on the fact that only nonequivalent adjacency matrices are formed, i.e. there is no need to check whether the currently constructed matrix has already been formed or not. According to theorem 4, the whole construction of canonical adjacency matrices from the family $\mathcal{F}_{pq}$ may be organized in a recurrent manner. We start from the simplest graph composed of one vertex numbered by 1; from this graph, we construct its canonically numbered extension composed of two vertices and one edge, etc. An algorithmic outline of the proposed method in the backtrack search form has been presented in our recent communication [20, algorithm 2] so, therefore, we shall not repeat it here. An illustrative example of our constructive enumeration is displayed in fig. 4.

5. Conclusions

The concept of semicanonical adjacency matrices (or semicanonically numbered graphs) represents a very powerful approach that allows us to considerably restrict the huge number of candidates whose canonicity has to be checked in the course of the recurrent enumeration of graphs with the prescribed number of vertices and edges. Its numerical efficiency is best documented by the fact that CPU times were about one or two orders smaller than those given in our earlier communication [15], where the cooperative numbering was used. The concept of semicanonical numbering has also been used in another recent communication [20], where we presented constructive enumeration of graphs based on the maximum codes produced by the upper-triangle part of adjacency matrices. It offers a constructive enumeration very similar to the approach initially suggested by Faradzhev [17,18] fifteen years ago. In our forthcoming communication [26], we would like to apply the present formalism for constructive enumeration of multigraphs with the prescribed distribution of valence states and empirical formulae (cf. ref. [16]).

References

[1] R.K. Lindsay, B.G. Buchanan, E.A. Feigenbaum and J. Lederberg, *Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project* (McGraw–Hill, New York, 1980).
[2] N.A.B. Gray, *Computer-Assisted Structure Elucidation* (Wiley, New York, 1986).
[3] J. Bauer, Tetrahedron Comput. Methodol. 2(1989)269.

[4] J.B. Hendrickson and C.A. Parks, J. Chem. Inf. Comput. Sci. 31(1991)101.
[5] B.R. Heap, in: *Graph Theory and Computing* (Academic Press, New York, 1972), p. 47.
[6] L.M. Masinter, N.S. Sridharan, J. Lederberg and D.H. Smith, J. Amer. Chem. Soc. 96(1974)7702.
[7] H. Baker, A.K. Dewdney and A.L. Szilard, Math. Comp. 127(1974)833.
[8] F.S. Bussemaker, S. Cobejlić and L.M. Cvetković, Technical Report 76-WSK-01, Technical University Eindhoven (1976).
[9] V.L. Arlazarov, I.I. Zuev, A.V. Uskov and I.A. Faradzhev, Zhur. Vychis. Mat. Mat. Fiz. 14(1974)737, in Russian.
[10] A.T. Balaban, in: *Chemical Application of Graph Theory*, ed. A.T. Balaban (Academic Press, London, 1976), p. 63;
R.C. Read, ibid., p. 25.
[11] J.V. Knop, W.R. Muller, Z. Jericević and N. Trinajstić, J. Chem. Inf. Comput. Sci. 21(1981)91;
J.V. Knop, W.R. Muller, K. Szymanski and N. Trinajstić, *Computer Generation of Certain Classes of Molecules* (SKTH, Zagreb, 1985);
N. Trinajstić, S. Nikolić, J.V. Knop, W.R. Muller and K. Szymanski, *Computational Chemical Graph Theory. Characterization, Enumeration and Generation of Chemical Structures by Computer Methods* (Ellis Horwood, 1991).
[12] M. Razinger, J. Župan and M. Nović, Microchimica Acta (1986)411.
[13] I.P. Bangov, J. Chem. Inf. Comput. Sci. 30(1990)277.
[14] A. Kerber, R. Laue and D. Moser, Anal. Chim. Acta 235(1990)221.
[15] V. Kvasnička and J. Pospíchal, J. Chem. Inf. Comput. Sci. 30(1990)99.
[16] V. Kvasnička and J. Pospíchal, Chemometrics Intell. Lab. Syst. 11(1991)137.
[17] I.A. Faradzhev, in: *Algorithmic Investigations in Combinatorics*, ed. I.A. Faradzhev (Nauka, Moscow, 1978), p. 11, in Russian.
[18] I.A. Faradzhev, in: *Constructive Enumeration of Combinatorial Objects*, Lecture, Orsay (July 8, 1976), Colloques internationaux C.N.R.S. No. 260 – Problèmes combinatoires et théorie des graphes, Paris (1978).
[19] S.G. Molodcov and V.N. Piottuch-Pelecky, Vychislitelnye Sistemy 103(1984)51, in Russian.
[20] J. Pospíchal and V. Kvasnička, Commun. Coll. Czech. Chem, in press.
[21] S. MacLane and G. Birkhoff, *Algebra* (MacMillan, New York, 1967).
[22] E.L. Lawler and D.E. Wood, J. ORSA 14(1966)217.
[23] F. Harary, *Graph Theory* (Addison–Wesley, Reading, MA, 1969).
[24] H.L. Morgan, J. Chem. Doc. 5(1965)107.
[25] J.B. Hendrickson and A.G. Toczko, J. Chem. Inf. Comput. Sci. 23(1983)171.
[26] V. Kvasnička and J. Pospíchal, Chemometrics Intell. Lab. Syst., submitted.