ELSEVIER

Computers and Chemistry 26 (2002) 119–123

Computers & Chemistry

www.elsevier.com/locate/compchem

The study on automorphism group of ESESOC

Jun-Feng Hao, Lu Xu *

Changchun Institute of Applied Chemistry, Academia Sinica, Changchun 130022, People's Republic of China

Received 28 February 2001; received in revised form 22 March 2001; accepted 1 May 2001

Abstract

It is necessary to generate the automorphism group of a chemical graph in computer-aided structure elucidation. In this paper, an algorithm was developed by the all-paths topological symmetry algorithm to build the automorphism group of a chemical graph. A comparison of several topological symmetry algorithms reveals that the all-paths algorithm (APA) could yield the correct class of a chemical graph. It lays a foundation for the ESESOC system in computer-aided structure elucidation. © 2002 Elsevier Science Ltd. All rights reserved.

Keywords: Isomorphism; Automorphism; Automorphism group

1. Introduction

For a structure elucidation expert system, whether the stereoisomers can be interpreted or not is very important, because most of the natural organic compounds and medicines possess stereoisomers. For the existing systems such as DENDRAL, CHEMICS, SEMAMI as well as our system, ESESOC, etc. the methods for the interpretations of stereoisomers used may be different, but the consideration of molecular symmetry cannot be avoided.

Chemists have paid much attention to molecular symmetry because it has an important impact on the properties of compounds. For the elucidation of some experimentally derived information, e.g. X-ray diffraction patterns or IR spectra, molecular symmetry must be considered, in which, the group theory is an effective tool. For a molecule, the first step is to decide which group it belongs to, and the next step is to get its property. This method has been widely applied in quantum chemistry for many years. The shortcomings of this method are that symmetry cannot be determined without knowing which group a structure belongs to, and it needs a large amount of calculation. In some other areas, like computer-enhanced structure elucidation or the enumeration of ^{13}C-NMR signals, partial solutions are possible using topological symmetry (connectivity) instead of three-dimensional geometry. For such cases, a different concept of symmetry based on the graph theory can be introduced.

This article studies the algorithm for automorphism group, based on the graph theory and the partition of the all-paths topological equivalence algorithm developed by our laboratory (Hu and Xu, 1999). The speed of calculation of the automorphism group was improved by the use of the matrix technique and properties of the group.

2. Basic concept

A graph can be represented in several different ways, but all have an identical topological symmetry. That is because topological symmetry is based on connectivity, an invariant of a graph. From the point of the graph theory, all organic molecular structures can be drawn as graphs in which atoms and bonds are represented by vertices and edges, respectively. Structural symmetry is related to the automorphism group of a vertex, and the

* Corresponding author.
E-mail address: luxu@ns.ciac.jl.cn (L. Xu).

PII: S0097-8485(01)00089-4

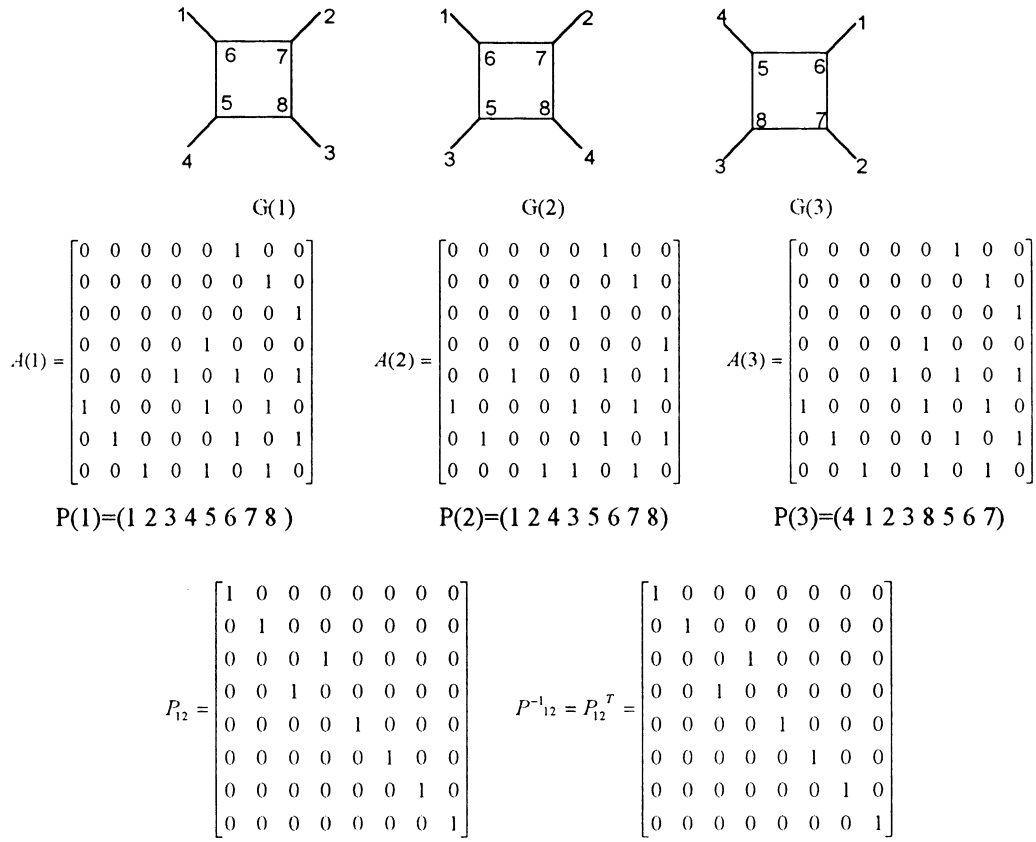

Fig. 1. A representation of graph isomorphism and automorphism.

automorphism group of the vertex is a subgroup of the vertex permutation group. These relations can be expressed by the adjacency matrix **A** and permutation matrix **P** (Razinger et al., 1993).

$$A_{ij} = \begin{cases} 1 & \text{if vertices } i \text{ and } j \text{ are neighbors} \\ 0 & \text{otherwise} \end{cases}$$

$$P_{ij} = \begin{cases} 1 & \text{if there exists a mapping of vertices } i\text{–}j \\ 0 & \text{otherwise} \end{cases}$$

A graph with n vertices has $n!$ possible labelings. Each of them can be represented by a vector $\boldsymbol{P}$, and the transformation between two different labelings is given by the permutation matrix **P**.

In Fig. 1, three of the 8! possible labelings of a graph are shown together with the corresponding A and P. Though $G(1)$ and $G(2)$ represent the same structure, they are not identical because they represent different connectivities. For all graphs, if there exists the following relation, we call it *isomorphism*.

$$P_{12}^{-1} \cdot A(G_1) \cdot P_{12} = A(G_2) \tag{1}$$

where P_{12} is a mapping of P_1–P_2. Considering that the permutation matrix is orthogonal, the previous relation can be further simplified.

$$P_{12}^{T} \cdot A(G_1) \cdot P_{12} = A(G_2) \tag{2}$$

If a transformation can transform itself, we call it *automorphism* using the following equation:

$$P^{T} \cdot A(G) \cdot P = A(G) \tag{3}$$

Every graph has at least one automorphism, i.e. the one made by identity permutation. Excluding the identity permutation, G_1 of Fig. 2(b) has seven other automorphisms, and G_3 is one of them. It can be reached by calculating from formula (3) or from the value of the adjacency matrix.

It has been testified that, for a graph, all permutations that meet automorphism are composed of a group, which is called the *automorphism group* (Balaban, 1976). In Fig. 1, the automorphism group of graph G_1 comprises eight vectors, i.e. $\boldsymbol{P}_1 = (1\ 2\ 3\ 4\ 5\ 6\ 7\ 8)$, $\boldsymbol{P}_2 = (4\ 1\ 2\ 3\ 8\ 5\ 6\ 7)$, $\boldsymbol{P}_3 = (3\ 2\ 1\ 4\ 5\ 8\ 7\ 6)$, $\boldsymbol{P}_4 = (3\ 4\ 1\ 2\ 7\ 8\ 5\ 6)$, $\boldsymbol{P}_5 = (2\ 1\ 4\ 3\ 8\ 7\ 6\ 5)$, $\boldsymbol{P}_6 = (1\ 4\ 3\ 2\ 7\ 6\ 5\ 8)$, $\boldsymbol{P}_7 = (2\ 3\ 4\ 1\ 6\ 7\ 8\ 5)$ and $\boldsymbol{P}_8 = (4\ 3\ 2\ 1\ 6\ 5\ 8\ 7)$.

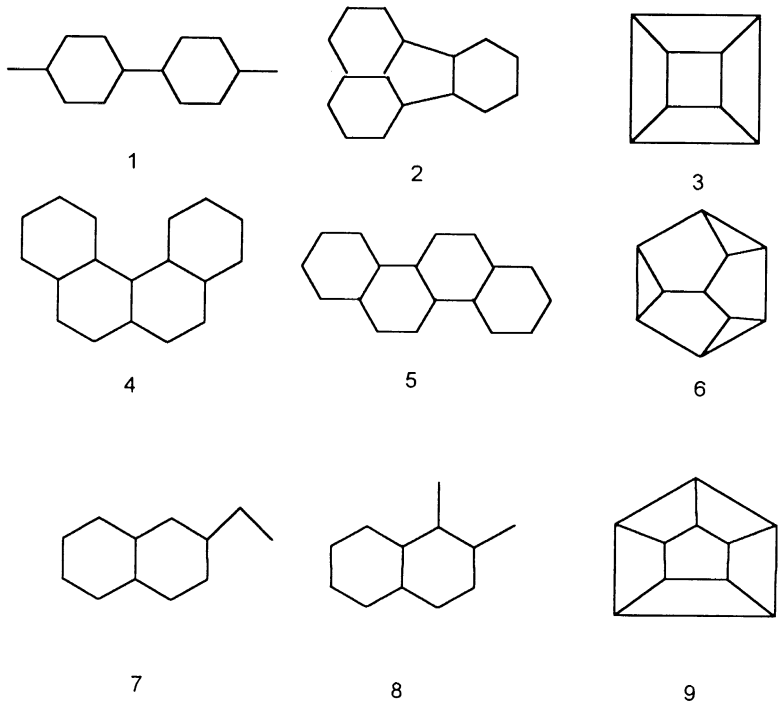

Fig. 2. Test graph.

Since each labeling of a graph corresponds to a vertex permutation, a symmetry manipulation in a graph can be represented by an automorphism, i.e. every automorphism corresponds to a symmetry operation of a graph. Therefore, all symmetry operations of a graph can be represented by automorphisms.

3. Algorithm

3.1. Partitioning of vertex

Generally speaking, the automorphism group of a graph is a subgroup of S_n, of which, n represents the number of a vertex, and S_n is the set of the total permutation of the vertex. The automorphism group might be equal to S_n only when the graph is a regular graph.

The calculation of automorphism group is time-consuming, as it needs to check all the vectors of S_n, in principle, the time complexity is $O(n^3)$. As shown by Razinger et al. (1993), first, we partitioned the vertex of a graph into equivalence class by the topological equivalence algorithm; second, we permuted every topological equivalence class and multiplied the number of permutation between classes, then obtained the final number of vectors that needed checking. For instance, tetramethylcyclobutane can be divided into two equivalence classes in Fig. 1, among which the vertices 1, 2, 3 and 4 are in one class, and the rest in another class. The number of vectors that needs checking is originally 8!, but it reduces to $4! \times 4!$ by using the topological classification.

To partition vertices, a topological equivalence algorithm is necessary. The final automorphism group is related to the partitioning algorithm. There are some topological equivalence algorithms, among which the Morgan algorithm (Morgan, 1965) is perhaps the first. Because of its simplicity, it is still popular at present. Other algorithms, such as the SEMA algorithm suggested by Wipke (Wipke and Dyott, 1974), the S–M algorithm of the SEMAMI system (Shelley and Munk, 1979), and the IMA algorithm suggested by Ouyang et al. (1999) are all based on Morgan's concept of extended connectivity or a small modification.

There are other kinds of algorithms based on the graph theory. Because of the shortcoming of extended connectivity, especially its vibration,[1] some authors have suggested other algorithms based on other properties of a graph, such as HOC-1 suggested by Balaban (Balaban et al., 1985) and our laboratory's all-paths

[1] The connectivity of extended connectivity largely relies on the connectivity of the neighbors, and it had changed a lot recently. Shortcomings may still exist, in the light of its specialty.

algorithm (APA), which will be given briefly in the next paragraph.

3.2. All-paths algorithm

A molecular structure can be represented by a graph (Hu and Xu, 1999) in which nodes are interpreted as atoms and edges as bonds. In a graph, a 'path' between two vertices i and j is any connected subgraph that starts and ends in those two vertices and traverses each of the intermediate vertices and edges only once. Chemical graphs are chromatic graphs, so the path that starts at vertex i and ends at vertex j ($\mathrm{Path}_{i;j}$) may not be identical with the path that starts at vertex j and ends at vertex i ($\mathrm{Path}_{j;i}$).

To characterize the paths in a molecular graph, a new path identifier (PI) was introduced by Hu and Xu, 1997.

$$\mathrm{PI} = \prod_{k=2}^{n_{ij}} \sqrt{\frac{b_{(k,k-1)}}{k} \times \frac{1}{p_k \times p_{k-1}}} \tag{4}$$

$$p = \frac{\delta}{\sqrt{Z}} \tag{5}$$

where k is the sequence number of the nodes along the path from node i to node j, n_{ij} the total nodes in the path, $b_{(k,k-1)}$ the code of the bond between node k and node $k-1$ ($b = 1, 2, 3$ and 1.5, respectively, for single, double, triple, and aromatic bonds), p_k the atom's property of atom k, δ the atom's connectivity (the atom of non-hydrogen atoms attached to it), and Z the atomic number.

This PI contains the factor, $b_{(k,k-1)}/k$ to characterize the multiple bonds and indicate the positions of the atoms and multiple bonds along the path. So it can discriminate the chromatic path.

To characterize the topological environment of the atom in a molecule, the Atomic IDentification number (AID) is defined by adding the PIs of all paths starting at that vertex.

$$\mathrm{AID} = \sum \mathrm{PI} \tag{6}$$

For each atom, all paths starting at that atom must be evaluated.

A rigorous test of the all-paths topological equivalence algorithm has been done, and it was used in the ESESOC system. In this study, we partitioned vertexes by the all-paths topological equivalence algorithm.

3.3. Matrix representation of triad

In the operation, we adopted a triad for matrix storage and manipulation. In formula (3), the multiplication of the matrix is a bottleneck in the algorithm, as its time complexity is $O(n^3)$. Hence, the key to speed up calculation is matrix manipulation.

As we know, adjacency and permutation matrices are sparse matrices, especially the permutation matrix; furthermore, the value of two matrix elements is either 1 or 0. Therefore, we adopted a triad to store and calculate. For every non-zero element of the matrix, it was represented by a structure which included the value of the non-zero element, its row and column number in the matrix. All the non-zero elements were represented by a one-dimensional array, which is called a triad.

By using the previous technique, a large amount of useless manipulation was avoided and the calculation was speeded up.

Furthermore, if it happens to include a big equivalence class, we can also use the properties of the group, such as inversion and close, to simplify calculation.

Table 1
Automorphism in ESESOC

Structure	Type of nodes					Time taken with automorphism group CPU (s)	Number of automorphism group
	ECA	SEMA	S–M	HOC-1	APA		
1	5	5	5	5	5	0.33	8
2	9	9	9	9	9	0.05	2
3	1	1	1	1	1	1.70	48
4	10	10	10	10	10	0.11	2
5	9	9	9	9	9	0.05	2
6	3	3	3	3	3	0.28	6
7	9	9	12	12	12	0.02	1
8	11	10	12	12	12	0.02	1
9	1	1	1	1	1	148	20

APA was developed by the authors. The hardware: Intel Celeron 300, 64 MB RAM. The operating system: Windows 95, BorlandC-5.02.

4. Results and conclusions

To verify the correctness of our algorithm, we have compared our system with other systems by using several structure graphs. The results are represented in Table 1.

We can draw some conclusions from Table 1. For all graphs in Fig. 2, the number of vertex equivalence classes predicted by APA developed by the authors is the same as the HOC-1 algorithm of Balaban and the S–M algorithm. The correctness for HOC-1 has been tested from theory (Mekenyan et al., 1985); thus, the APA algorithm is correct.

On generating the automorphism group, the speed is fast for graphs with low symmetry, such as graphs 1, 2, 4, 5, 6, 7 and 8 in Fig. 2. For example, the CPU time for graph 1 was 90 s on a VAX workstation 3100 in the SEMAMI system (Razinger et al., 1993), while on our system it was only 0.33 s on Celeron 300. For high symmetry compounds, especially, there is only one equivalent class, and the time needed to achieve the automorphism group is long, but these compounds are not commonly found in structure elucidation problems. Therefore, our algorithm for generating the automorphism group is effective and its reliability is verified by the generation of stereoisomers of ESESOC.

Acknowledgements

The authors acknowledge the financial support of the National Natural Science Foundation of China.

References

Balaban, A.T., 1976. Chemical Applications of Graph Theory. Academic Press, New York.

Balaban, A.T., Mekenyan, O., Bonchev, D., 1985. Unique description of chemical structures based on hierarchically ordered extended connectivities (HOC Procedures). I. Algorithms for finding graph orbit and canonical numbering of atoms. J. Comput. Chem. 6, 538.

Hu, C.Y., Xu, L., 1997. Developing molecular identification numbers by an all-paths method. J. Chem. Inf. Comput. Sci. 37, 311.

Hu, C.Y., Xu, L., 1999. Computer perception of topological symmetry by all-paths algorithm. Chemom. Intell. Lab. Syst. 45, 318.

Mekenyan, O., Bonchev, D., Balaban, A.T., 1985. Unique description of chemical structures based on hierarchically ordered extended connectivities (HOC Procedures). II. Mathematical proofs for the HOC algorithm. J. Comput. Chem. 6, 552.

Morgan, H.L., 1965. The generation of a unique machine description for chemical structures — a technique developed at chemical abstracts service. J. Chem. Doc. 5, 107.

Ouyang, Z., Yuan, S.G., Brandit, J., Zheng, C.Z., 1999. An effective topological symmetry perception and unique numbering algorithm. J. Chem. Inf. Comput. Sci. 39, 299.

Razinger, M., Balasubramanian, A., Munk, M.E., 1993. Graph automorphism perception algorithms in computer-enhanced structure elucidation. J. Chem. Inf. Comput. Sci. 33, 197.

Shelley, C.A., Munk, M.E., 1979. Computer perception of topological symmetry. J. Chem. Inf. Comput. Sci. 19, 247.

Wipke, W.T., Dyott, T.M., 1974. Stereochemically unique naming algorithm. J. Am. Chem. Soc. 96, 4834.