communications in mathematical
match
and in computer chemistry

no. 41, March 2000

ISSN 0340-6253

MATCDY (41) 145-149 (2000)

Automatic Enumeration of All Connected Subgraphs

Gerta Rücker and Christoph Rücker*

Institut für Organische Chemie und Biochemie, Universität Freiburg,

Albertstrasse 21, D-79104 Freiburg, Germany

A computer program for generating all connected subgraphs of a connected undirected simple graph is introduced, which is based on a path-tracing algorithm in the graph's edge adjacency matrix.

INTRODUCTION

The number of all connected subgraphs was first proposed as a measure of a graph's complexity by Bertz and Herndon in 1986.[1] These authors also cursorily mentioned a computer program written by them for the enumeration of all such subgraphs. We were not able to obtain more information on this program.

Bertz and Sommer in 1997 advocated the use of the number of all connected subgraphs to evaluate the complexity of synthetic intermediates and thus of synthetic strategies.[2]

Bonchev in the same year proposed the number of connected subgraphs and the sum of their total adjacencies ("Topological Complexity", TC), as measures of a graph's complexity.[3] It is not written in references 2 and 3 how the numbers of subgraphs were obtained. In fact, the authors constructed the subgraphs of their examples by hand.[4]

Again in 1997, Bone and Villar for estimating molecular diversity came up with a computer program to find all connected subgraphs in a molecular structure.[5] These authors, however, defined a connected subgraph simply as a set of connected graph vertices, thus missing several subgraphs in which the same vertex set may be joined by different edge sets. In contrast, since a graph is defined by its vertex set **and** its edge set, the same should hold for a subgraph.

We had introduced in 1993 the number of all walks in a graph (of n vertices and m edges) up to length n-1, the total walk count, *twc*, as an extremely easily obtained graph invariant.[6] We feel *twc* to be a far better complexity measure than the number of all subgraphs.[7] When detailing this idea and comparing *twc* to other complexity measures,[8] we required a computer program for the enumeration of all connected subgraphs. Since such a program was not available, we undertook to write one ourselves, and here we report on the resulting program SUBGRAPH.

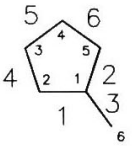
RESULTS AND DISCUSSION

It is clear from the beginning that the connected subgraphs of even a modest-sized graph are numerous. The number has as an upper bound 2^m, the number of all possible sets of edges. This however is not a serious problem. For comparison, the walks in most cases are even more numerous, but their number is obtained easily by matrix multiplication or by the Morgan algorithm without knowledge of every single walk. In contrast, there is no simple mathematical procedure known to obtain the number of connected subgraphs, therefore the individual subgraphs have to be constructed systematically and then counted.

Since a connected subgraph is uniquely described by the set of its edges, the program essentially performs a depth-first path-tracing procedure in the graph's edge adjacency matrix. The procedure is illustrated by the example of methylcyclopentane (Scheme 1), which was chosen in order to allow comparison with reference 5. The graph is entered as its usual (vertex) adjacency matrix, with the vertices numbered arbitrarily, e.g. as given by the smallprint numbers in the drawing in Scheme 1. The edge adjacency matrix then is easily constructed, resulting in the edge numbering given in larger print in the drawing. Now starting with every edge in turn, the program lines up adjacent edges in strings. Scheme 1 (to be read from left to right and from top to bottom) gives the proper order in which new subgraphs are found by extension of an existing subgraph.

As in every path algorithm, there are forward steps and back steps. A step forward is done if a given connected subgraph can be extended by addition of edge k, that is if edge k is not already part of the given subgraph, if k is adjacent to at least one edge of the given subgraph, and if addition of edge k is not forbidden by some restrictions given below.

1	**12**	**123**	**1234**	**12345**	**123456**
				12346	
			1236	**12365**	
		124	**1245**	**12456**	
			1246		
		126	**1265**		
	13	**134**	**1345**	**13456**	
	14	**145**	**1456**		
2	**23**	**236**	**2365**	**23654**	
	26	**265**	**2654**		
3					
4	**45**	**456**			
5	**56**				
6					



6 7 8 9 6 1 Sum = 37

Scheme 1

A step back is done as soon as a given connected subgraph cannot be further elongated. In this case the edge added last is removed from the string, it is temporarily given the status "forbidden", and any other edges which were forbidden by backtracking from a previous longer string are simultaneously "allowed" again. In contrast, an edge which is forbidden by being removed from a string shorter than the present one remains forbidden, thus assuring that every connected subgraph is constructed once and only once.

Note that by this procedure edge set 1235 in our example (not a connected subgraph, digits are edge numbers) is not constructed and thus not counted, while 12356 (a connected subgraph) is constructed as the extension 12365 of subgraph 1236. Similarly, subgraphs 2356 and 23456 are found as extensions 2365 and 23654 of subgraph 236.

The program was tested on all graphs appearing in references 1-3, and the number of subgraphs found was in each case identical to that reported in the references.[9,10]

Obviously, this is a brute force algorithm, and as a path-tracing procedure its dependence on problem size is notoriously exponential.

The CPU time required to process all (hydrogen-suppressed) alkane graphs of up to 10 vertices (the acyclic decanes) was ≤0.01 sec each, the pentacyclic octanes cubane (2433 connected subgraphs), cuneane (2237) and octabisvalene (1852) used ca. 0.1 sec each, as did tetracyclic decanes. Dodecahedrane (an undecacycloicosane) or fullerene-20 (145168228 subgraphs) required 73 min (all runs on a SG Indigo workstation, 150 HMz, R5000 coprocessor). For graphs that are not amenable to full treatment by the program within reasonable time, e.g. the fullerenes higher than C_{20}, a program version is available which constructs all subgraphs up to a certain size limit only. Thus in the graph of I_h-fullerene-60 (buckminsterfullerene, C_{60}) the numbers of connected subgraphs of size 1, 2, ..., 12 edges are 90, 180, 420, 1080, 2922, 8120, 23040, 66480, 193780, 569082, 1681560, 4990090. 8.4 min CPU time were used to construct these subgraphs.

A copy of the program SUBGRAPH, written in FORTRAN 77, is available from the authors.

REFERENCES AND NOTES

(1) Bertz, S.H.; Herndon, W.C. Similarity of Graphs and Molecules. In *Artificial Intelligence Applications in Chemistry*, (Pierce, T.H.; Hohne, B.A., Eds.), ACS, Washington, 1986, pp. 169-175.

(2) Bertz, S.H.; Sommer, T.J. Rigorous Mathematical Approaches to Strategic Bonds and Synthetic Analysis Based on Conceptually Simple New Complexity Indices. *Chem. Commun.* **1997**, 2409-2410.

(3) Bonchev, D. Novel Indices for the Topological Complexity of Molecules. *SAR and QSAR in Environm. Res.* **1997**, *7*, 23-43.

(4) S. H. Bertz and D. Bonchev, private communications. D. Bonchev also informed us on the existence of a computer program for the present purpose written by Dr. K. Gordeeva at MDL, San Leandro, CA.

(5) Bone, R.G.A.; Villar, H.O. Exhaustive Enumeration of Molecular Substructures. *J. Comput. Chem.* **1997**, *18*, 86-107.

(6) Rücker, G.; Rücker, C. Counts of All Walks as Atomic and Molecular Descriptors. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 683-695.

(7) Rücker, G.; Rücker, C. Walk Counts, Labyrinthicity, and Complexity of Acyclic and Cyclic Graphs and Molecules. *J. Chem. Inf. Comput. Sci.*, submitted.

(8) For recent surveys on complexity measures see (a) Bonchev, D.; Seitz, W.A., The Concept of Complexity in Chemistry. Chapter 11 in *Concepts in Chemistry: Contemporary Challenge* (Rouvray, D.H., Ed.), Research Studies Press, Taunton, U.K., 1997, pp.353-381. (b) Bonchev, D. Overall Connectivity and Topological Complexity: A New Tool for QSPR/QSAR. Chapter 8 in *Topological Indices and Related Descriptors in QSAR and QSPR* (Devillers, J.; Balaban, A.T., Eds.), Gordon & Breach, 1999, pp. 361-401. (c) Bertz, S.H.; Zamfirescu, C. New Complexity Indices Based on Edge Covers, *MATCH*, in press. (d) Randic, M. On the Concept of Molecular Complexity, *Croat. Chem. Acta*, in press.

(9) The smallest structural units counted as subgraphs in our procedure are single edges, while the authors of references 1-3 counted single vertices also. Thus, to be exact, the numbers obtained by our procedure are smaller by n than those reported in refs 1-3.

(10) As an exception, one error in a previously published subgraph number was detected: For the graph of methylcyclobutane reference 3 reported 28 subgraphs, which after subtraction of 5 (for the single vertices[9]) corresponds to 23 subgraphs, whereas our program found (correctly) 24 subgraphs.