Comput. Math. Applic. Vol. 15, No. 4, pp. 303–309, 1988
Printed in Great Britain.

0097-4943/88 $3.00 + 0.00

GRAPH CODES AND A DEFINITION OF STRUCTURAL SIMILARITY

W. C. HERNDON
Department of Chemistry, The University of Texas at El Paso, El Paso, TX 79968, U.S.A.

Abstract—A succinct linear notation system to encode the structure of a graph is exemplified. The notation requires a prior canonical numbering of the graph nodes based on the lengths of a longest path and path branches, and uses locants to designate branch positions and cyclicity. An algorithm and computer program to obtain the longest paths and a spanning tree containing a longest path is described. An index which measures the similarity of a pair of graphs is defined based on a comparison of their linear codes.

1. INTRODUCTION

The quantitative elucidation of molecular structural similarity is an important topic of research in chemistry. A basic premise is that molecules with similar chemical structures will exhibit similar physical and chemical properties. Perhaps even more important, they could also exhibit similar biological or pharmacological activities. Practical use of the molecular similarity concept falls within the subfield of chemistry denoted by the acronym QSAR, standing for quantitative structure activity relationships. QSAR methodology usually involves multivariate linear regression where a correlation is sought between an observed biological or chemical activity and a set of arbitrarily selected chemical or molecular descriptors. Molecular descriptors used in the past have included various kinds of physical properties, both theoretical and experimental reactivity parameters, and many kinds of structural descriptors [1–7].

Several studies have made use of graph theoretical concepts to define molecular structural descriptors [2, 8]. The possibility to use graph theory arises from the fact that the structure of any molecule can be represented by a molecular graph. The nodes of this graph are labeled and normally correspond to (the nuclei of) atoms while the graph edges usually represent electronic chemical bonds. Two recent series of QSAR papers exemplify work in this area. In the first, abstract graphical quantities, i.e. path counts of varying sizes, have been used as a basis for structure-activity analyses [9–14]. In the second, labeled molecular graph paths selected by a discriminant-type analysis served as the structural descriptors [15–19].

In the present work a fundamentally different approach to the structural similarity problem has devolved, primarily based on molecular graph representations of chemical structures, and the use of these representations to give rise to a linear molecular coding system [20, 21]. The procedure encompasses the following steps:

(a) A unique graph code is derived for the underlying unlabelled graph of the molecular graph.
(b) The graph code is converted into the molecular graph code which is composed of a linear list of atom and bond symbols and of locants for particular structural features.
(c) The similarity between two molecular structures is evaluated by a comparison of the two sequences of symbols in their molecular graph codes using standard string comparison techniques [22–23].

We will address principally the problems subsumed under step (a), and the further use of graph codes to define graph structural similarity analogous to the procedures used in step (c). The conversion of the graph code to a complete molecular code, step (b), is a problem particular to chemistry which is examined in detail elsewhere [24].

2. LINEAR NOTATIONS FOR GRAPHS

Notation systems to represent molecular systems are an important component of the nomenclature of chemistry [25–32]. Many of the chemical notation systems require a canonicalization and unique numbering of the nodes of the molecular graph. Three distinct types of algorithms have been developed to obtain a unique numbering. The first, which is used in one of the generally accepted substitutive chemical nomenclature systems [33], employs the longest path and largest cycle as a basis for numbering acyclic and cyclic molecular graphs respectively. The remaining two approaches use either standard rearrangements of the adjacency matrix of the molecular graph or particular definitions of extended connectivity to obtain a hierarchal numbering [34–45]. The results of any one of these numbering systems can be cast into a linear notation format, for example using a well-known linear representation of the graph adjacency matrix [20].

The procedure to be used here takes a longest path in a graph as the single structural element to initiate canonical numbering. Path branches, which are paths that emanate from previously numbered paths, are subsequently numbered in order of decreasing path branch size. Then the notation is completed by adding locants for the "path branches", and double locants for single edge "path bridges" that define the graph cyclicity. The components of the graph notation are written in a prescribed order to be illustrated below. When two or more derived alternative notations can be obtained for the same graph, then the notation is chosen that is lowest at the first point of difference.

Three examples are presented in Fig. 1, hopefully sufficient to clarify the system for assigning a notation. Example G1 illustrates the hierarchal assignment of nodal numbers based on the lengths of the longest path and the path branches. Note that all locants are enclosed in parentheses, and the paths are listed in the order of decreasing length. The alternative numbering of the 10 node longest path would give a notation 10(06)04(07)02(12)01 which is lexicographically inferior to the preferred notation.

Example G2 is a regular graph that has been drawn in such a way as to illustrate that the nodes can be assigned to 3 different equivalence classes. This particular graph has figured in arguments regarding the efficacy of molecular symmetry perception algorithms [21], and the symmetry is reflected in the fact that the graph possesses 204 distinct Hamiltonian paths. Six of these Hamiltonian paths provide the preferred graph notations, and one of these is exhibited in G2 along with the optimal numbering. The cylomatic number of the graph is 6, obtained by summing the number of path bridges (double locants appearing in parentheses). Example G3 displays both path

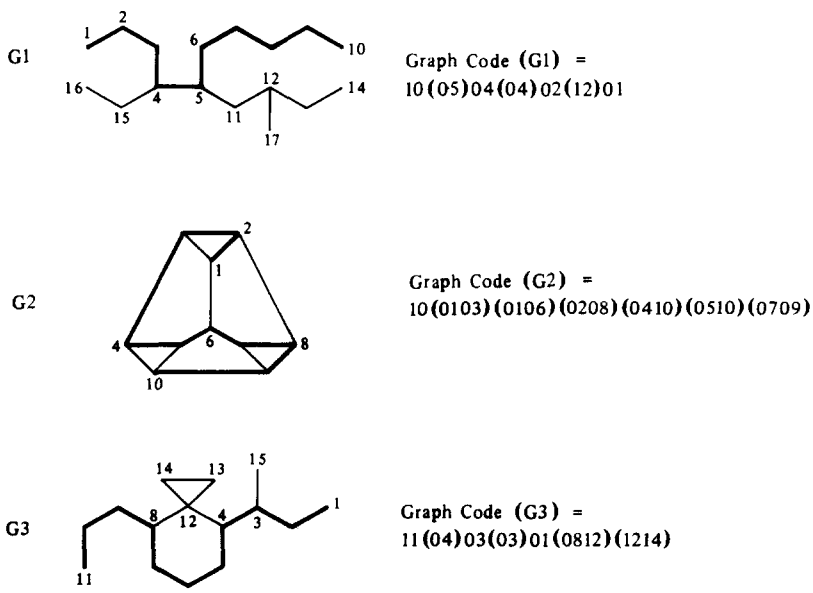


Fig. 1. Three graphs and their codes.

branches and cyclicity. The assigned numbering of the main path with 11 nodes is determined by the path branch at node 3 and the cyclicity is delimited by the path bridges (0812) and (1214).

3. ALGORITHM FOR LONGEST PATHS

The notation assignment must be preceded by determination of all of the longest paths in a graph, a well-known NP-complete problem. The longest path algorithm given below is patterned after a method published by Kaufman [46, 47] which involves the successive symbolic matricial multiplication of particular defined matrices derived from the adjacency matrix of the graph. However, the present algorithm, which involves successive multiplications of a single matrix by a column vector, runs in *ca* 1/n of the time of the Kaufman procedure, where n is the number of graph nodes. By obvious extensions the results of the revised algorithm can be processed to also yield all paths and/or all cycles including Hamiltonian cycles. An advantage of this method is that paths are obtained and enumerated *without any redundancy* so auditing of the results for duplications is unnecessary. The algorithmic steps are as follows, illustrated with Graph G4 in Fig. 2.

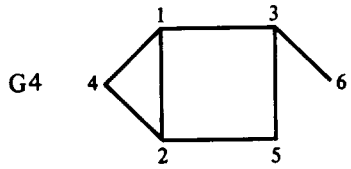


Fig. 2. Graph for illustration of longest paths algorithm.

Step 1

(a) Define the row-labeled $n \times n$ string matrix $\{\mathbf{B}\}$ from the adjacency matrix of the graph $\{\mathbf{A}\}$ by exchanging each non-zero off-diagonal element of $\{\mathbf{A}\}$ for the string representation of the respective row number. All other elements of $\{\mathbf{B}\}$ consist of null strings.

(b) Define a column vector $\{\mathbf{V}\}^0$, consisting of the numbers one through n in string form.

$\{A\}=$

	1	2	3	4	5	6
1		1	1	1		
2	1			1	1	
3	1				1	1
4	1	1				
5		1	1			
6			1			

$\{B\}=$

	01	01	01		
02			02	02	
03				03	03
04	04				
	05	05			
		06			

$\{V\}^0=$

01
02
03
04
05
06

Step 2

(a) Define multiplication $\{\mathbf{B}\}L\{\mathbf{v}\}^m = \{\mathbf{V}\}^{l+m}$ where L is the symbol for "Latin" multiplication as prescribed by Kaufman. Latin multiplication is performed according to steps (b)–(e).

(b) Null multiplied by anything equals null.

(c) String multiplication is defined as string concatenation.

(d) String addition can also be defined as string concatenation. Each element of the string sum is processed separately in subsequent Latin multiplication.

(e) Any string that has a repeated number is set equal to null.

$$\{\mathbf{V}\}^1 = \begin{vmatrix} 0102 & 0103 & 0104 \\ 0201 & 0204 & 0205 \\ 0301 & 0305 & 0306 \\ 0401 & 0403 & \\ 0502 & 0503 & \\ 0603 & & \end{vmatrix} \quad \{\mathbf{V}\}^2 = \begin{vmatrix} 010204 & 010205 & 010305 & 010306 \\ 020103 & 020104 & 020401 & 020503 \\ 030102 & 030102 & 030502 & \\ 040102 & 040103 & 040201 & 040205 \\ 050201 & 050204 & 050301 & 050306 \\ 060301 & 060305 & & \end{vmatrix}.$$

Step 3

(a) The entries in the vector $\{\mathbf{V}\}^k$ give the non-redundant paths of length k.

(b) $\{\mathbf{V}\}^n$ gives the Hamiltonian paths.

$$\{\mathbf{V}\}^6 = \begin{vmatrix} 010402050306 & & \\ & & \\ 040102050306 & & \\ 050204010306 & & \\ 060301040205 & 060305020104 & 060305020401 \end{vmatrix}$$

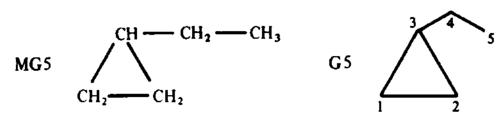
A computer program to implement this algorithm actually creates the vector $\{\mathbf{V}\}^k$ as a single list, the length of the list being the number of paths of length k. For a successive multiplication, the first term of an entry in the vector list is read, and the appropriate column of $\{\mathbf{B}\}$ is multiplied by that entry using the rules for multiplication given in Step 2. If a path of length $k+1$ is found, the paths of length $k-1$ are discarded to conserve space. Structure-based heuristics that allow some of the elements of $\{\mathbf{V}\}^0$ to be set equal to null are also incorporated in the program. In particular, nodes with a valence of two adjacent to terminal nodes, or adjacent to two other nodes with valence two, cannot initiate a longest path, and their string numerical symbols are therefore omitted from $\{\mathbf{V}\}^0$. Symmetry perceived by the user can also be employed to reduce computational time by eliminating the symbols for redundant nodes in $\{\mathbf{V}\}^0$ if desired.

The computer program is written in the Basic language, and a compiled version is used running on standard microcomputers. After randomization of the graph numbering for G1, G2 and G3, the optimal graph codes are obtained in 45, 565 and, 88 s, respectively. Use of the symmetry of G2 lowers the required time to 241 s. Cyclicity is of course the main factor that increases the number of longest paths that must be tested, thereby increasing the running time for obtaining the graph code. One notes that the code up to the double locants defining cyclicity is simply a notation for a spanning tree containing a longest path. As far as can be ascertained no general faster algorithm to obtain such a tree is extant.

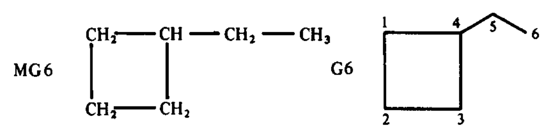
4. GRAPH SIMILARITY

Two different graphs are represented by two different linear codes, i.e. two different strings of numerical symbols. The similarity of two graphs will be defined by developing a quantitative comparison of the string notations. One of the general approaches used in sequence or string comparisons is to seek the number of insertions and/or deletions that are required to convert one string to the other [22, 23, 48, 49]. The larger this number, called the distance between the two strings, the less similar are the two sequences. This type of definition will be used in this work. However, before giving the quantitative definition of similarity, a digression to consider the molecular graph code is necessary which will help to justify the final form of the defined graph similarity.

In the graph code, the terms outside of parentheses represent the number of graph nodes in the longest path and in any additional path branches. In the molecular code, each non-parenthetical term is replaced by the string of labeled molecular graph nodes along with the intervening labeled edges. Examples are shown in Fig. 3, where C, H and S stand for carbon atoms, hydrogen atoms, and molecular single bonds, respectively.



Graph Code 05(0103)
Molecular Code CH2–S–CH2–S–CH–S–CH2–S–CH3(0103)–S–



Graph Code 06(0104)
Molecular Code CH2–S–CH2–S–CH2–S–CH–S–CH2–S–CH3(0104)–S–

Fig. 3. Two molecular graphs and their codes.

The identical parts of the two molecular graph codes are underlined. Three insertions and one deletion must be made in the code for MG5 to convert to the code for MG6. The total number of terms in the two codes is $12 + 14 = 26$. The similarity is then calculated as unity minus the number of insertions/deletions required for conversion divided by the total number of terms

$$S(MG5, MG6) = 1 - 4/26 = 0.846. \tag{1}$$

The similarity of two molecular graphs is thus highly dependent upon molecular structure connectedness, in particular upon the ordering of labeled nodes and edges in the longest paths, and upon the total number of terms in the molecular graph codes. A simple definition of graph similarity should retain this order of magnitude of structural dependence, and one also expects that a calculated similarity for two graphs (where nodes and edges are not labeled) should be equal to or larger than the calculated similarity of two corresponding molecular graphs. The required degree of structural dependence can be obtained by postulating that each path and path branch in the graph code contributes $2 \times p$ unlabeled structural terms to the description of the graph where p is the number of nodes in the path or path branch. To exemplify, the similarities of pairs G1, G3 and G5, G6 are now calculated

$$S(G1, G3) = 1 - 17/(37 + 36) = 0.767, \tag{2}$$

$$S(G5, G6) = 1 - 4/26 = 0.846. \tag{3}$$

The hand calculation of graph similarity is facilitated by drawing an alignment of the two codes as illustrated below for the codes of G1 and G3.

```
Graph code (G1)        10(05)     04(04)02(12)      01
Graph code (G3)        11      (04)03            (03)01(0812)(1214)
Insertions/deletions    2 1     1  2  1  4  1     1       1 1   1 1.
```

This illustration demonstrates that, in part, the similarity definition depends upon a correspondence between notations that preserves the order of terms in the codes. Alignments are used for similarity analysis in many practical fields of application, particularly in polymer chemistry and biochemistry [22, 23, 48, 49]. This use of alignments is adopted to facilitate extensions of the present results to chemical problems.

5. SUMMARY

The definition of graph and molecular graph similarity given here has the practical advantages of simplicity [50]. If two molecular graphs have every structural element dissimilar, a condition easily fulfilled for molecules, their calculated similarity is zero. Identical pairs of molecular graphs and graphs will both have similarities of unity. The range of defined similarities, from zero to one, is easily interpreted and may therefore be useful in practical comparisons. This similarity definition also allows one to derive general formulas for comparison of specified types of graphs, an application that is being explored in detail in work on chemical systems presently under investigation.

However, the question of the structural similarity of a pair of graphs is of course a concept open to a variety of intepretations. It can surely be stated that similarity, like beauty, is in the eye of the beholder. The desirability of a particular graph and chemical coding system is also a matter of preference. In some eyes the simplicity and concise nature of the two derived constructs presented here may confer a modicum of beauty.

Acknowledgements—The financial support of the Robert A. Welch Foundation of Houston, Texas is gratefully acknowledged. The author also thanks Frank Harary for informal lessons and helpful arguments about graph theory.

REFERENCES

1. W. V. Valkenburg, *Biological Correlations—The Hansch Approach*. American Chemical Society, Washington D.C. (1972).
2. L. B. Kier and L. H. Hall, *Molecular Connectivity in Chemistry and Drug Research*. Academic Press, New York (1976).
3. C. Hansch and A. Leo, *Substituent Constants for Correlation Analysis in Chemistry and Biology*. Wiley, New York (1979).
4. J. K. Seydel, Ed., *QSAR and Strategies in the Design of Bioactive Compounds*. Verlag, Weinheim (1985).
5. R. Osman, H. Weinstein and J. P. Green, Parameters and methods in quantitative structure-activity relationships. In *Computer-Assisted Drug Design* (Eds E. C. Olson and R. E. Cristofferson) pp. 21–77. ACS Symp. Series 112, American Chemical Society, Washington D.C. (1979).
6. G. Klopman, Artificial intelligence approach to structure-activity studies. Computer automated structure evaluation of biological activity of organic molecules. *J. Am. chem. Soc.* **106,** 7315–7321 (1984).
7. P. C. Jurs, T. S. Stouch, M. Czerwinski and J. N. Narvaez, Computer-assisted studies of molecular structure-biological activity relationships. *J. chem. Inf. Comput. Sci.* **25,** (1985) 296–308.
8. A. T. Balaban, Applications of graph theory in chemistry. *J. chem. Inf. Comput. Sci.* **25,** 334–343 (1985).
9. C. L. Wilkins and M. Randić, A graph theoretical approach to structure-property and structure-activity correlations. *Theor. Chim. Acta* **58,** 45–68 (1980).
10. M. Randić and C. L. Wilkins, Graph-theoretical analysis of molecular properties. Isomeric variations in nonanes. *Int. J. Quantum. Chem.* **18,** 1005–1027 (1980).
11. C. L. Wilkins, M. Randić, S. M. Schuster, R. S. Markin, S. Steiner and L. Dorgan, A graph-theoretic approach to quantitative structure-activity/reactivity studies. *Anal. Chim. Acta* **133,** 637–645 (1981).
12. M. Randić, Nonempirical approach to structure-activity studies. *Int. J. Quantum Chem.* **11,** 137–153 (1984).
13. M. Randić, On molecular identification numbers. *J. chem. Inf. Comput. Sci.* **24,** 164–175 (1984).
14. B. Jerman-Blazic, I. Fabic and M. Randić, Comparison of sequences as a method for evaluation of the molecular similarity. *J. Comput. Chem.* **7,** 176–188 (1986).
15. G. Klopman, Artificial intelligence approach to structure-activity studies. Computer automated structure evaluation of biological activity of organic molecules. *J. Am. chem. Soc.* **106,** 7315–7321 (1984).
16. G. Klopman, K. Namboodiri and A. N. Kalos, Computer automated evaluation and prediction of the iball index of carcinogenicity of polycyclic aromatic hydrocarbons. In *Molecular Basis of Cancer, Part A: Macromolecular Structure, Carcinogens, and Oncogenes* pp. 287–298 (1985).
17. G. Klopman and O. T. Macina, Use of the computer automated structure evaluation program in determining quantitative structure-activity relationships within hallucinogenic phenylalkylamines. *J. Theor. Biol.* **113,** 637–648 (1985).
18. G. Klopman and A. N. Kalos, Quantitative structure-activity relationships of beta-adrenergic agents. Application of the computer automated structure evaluation (CASE) technique of molecular fragment recognition. *J. Theor. Biol.* **118,** 199–214 (1986).
19. G. Klopman, O. T. Macina, E. J. Simon and J. M. Hiller, Computer automated structure evaluation of opiate alkaloids. *J. Mol. Struct (Theochem)* **134,** 299–308 (1986).
20. W. C. Herndon and J. E. Leonard, Canonical numbering, stereochemical descriptors, and unique linear notations for polyhedral clusters. *Inorg. Chem.* **22,** 554–557 (1983).
21. W. C. Herndon, Canonical labelling and linear notation for chemical graphs. In *Chemical Applications of Topology and Graph Theory* (Ed. R. B. King) pp. 231–242. Elsevier, Amsterdam (1983).
22. D. Sankoff, Matching sequences under deletion-insertion constraints. *Proc. Nat. Acad. Sci. (U.S.A.)* **69,** 4–6 (1972).
23. D. Sandoff and J. B. Kruskal, Eds., *Time Warps, String Edits, and Macro-molecules; The Theory and Practice of Sequence Comparison*. Addison-Wesley, Reading, Mass. (1983).
24. W. C. Herndon and S. H. Bertz, Quantification of the Concept of Molecular Similarity, *J. Comput. Chem.* **8,** 367–374 (1987).

25. R. S. Cahn and O. C. Dermer, *Introduction to Chemical Nomenclature* (5th edn). Butterworths, London (1979).
26. International Union of Pure and Applied Chemistry. Commission on the nomenclature of inorganic chemistry. *Nomenclature of Inorganic Chemistry* (2nd Edn). Butterworths, London (1971).
27. International Union of Pure and Applied Chemistry. Commission on the nomenclature of organic chemistry, *Nomenclature of Organic Chemistry* (1979 Edn). Pergamon Press, Oxford (1979).
28. Chemical Abstracts Service, Division of the American Chemical Society. *Chemical Substance Name Selection Manual* (1982 Edn), Vols I and II, American Chemical Society, Columbus, Ohio (1982).
29. J. E. Rush, Status of Notation and Topological Systems and Potential Future Trends, *J. chem. Inf. Comput. Sci.* **19,** 195–198 (1976).
30. J. A. Silk, Realistic vs Systematic Nomenclature. *J. chem. Inf. Comput. Sci.* **21,** 146–148 (1981).
31. W. C. Fernelius, Present status of inorganic chemical nomenclature. *J. chem. Inf. Comput. Sci.* **4,** 214–218 (1981).
32. W. J. Wiswesser, Historic development of chemical notations. *J. chem. Inf. Comput. Sci.* **25,** 258–263 (1985).
33. International Union of Pure and Applied Chemistry. Commission on the nomenclature of organic chemistry, Nomenclature of Organic Chemistry (1979 Edn), pp. 31–42. Pergamon Press, Oxford (1979).
34. C. Jochum and J. Gasteiger, Canonical numbering and constitutional symmetry, *J. chem. Inf. Comput. Sci.* **17,** 113–117 (1976).
35. M. Randić, On canonical numbering of atoms in a molecule and graph isomorphism. *J. chem. Inf. Comput. Sci.* **17,** 171–180 (1977).
36. W. Schubert and E. Ugi, Constitutional symmetry and unique descriptors of molecules. *J. Am. chem. Soc.* **100,** 37–41 (1977).
37. C. A. Shelley and M. E. Munk, Computer perception of topological symmetry. *J. chem. Inf. Comput. Sci.* **17,** 110–117 (1977).
38. C. A. Shelley and M. E. Munk, An approach to the assignment of canonical connection tables and topological symmetry perception. *J. chem. Inf. Comput. Sci.* **19,** 247–250 (1979).
39. G. Moreau, A topological code for molecular structures. A modified Morgan algorithm. *Nouv. J. Chim.* **4,** 17–21 (1979).
40. M. Randić, G. M. Brissey and C. L. Wilkins, Computer perception of topological symmetry via canonical numbering of atoms. *J. chem. Inf. Comput. Sci.* **21,** 52–59 (1980).
41. V. E. Golender, V. V. Drgoblav and A. B. Rosenblit, Graph potentials method and its application for chemical information processing. *J. chem. Inf. Comput. Sci.* **21,** 196–204 (1981).
42. M. Uchino, Algorithms for unique and unambiguous coding and symmetry perception of molecular structure diagrams—5. Unique coding by the method of "orbit graphs". *J. chem. Inf. Comput. Sci.* **22,** (1982) 201–206.
43. M. Razinger, Extended Connectivity in Chemical Graphs. *Theor. Chim. Acta* **61,** 581–586 (1982).
44. J. B. Hendrickson and A. G. Toczko, Unique numbering and cataloguing of molecular structures. *J. chem. Inf. Comput. Sci.* **23,** 171–177 (1983).
45. A. T. Balaban, Unique description of chemical structures based on hierarchically ordered extended connectivities (HOC Procedures)—I. Algorithms for finding graph orbits and canonical numbering of atoms. *J. Comput. Chem.* **6,** 538–551 (1985).
46. A. Kaufman, *Graphs, Dynamic Programming, and Finite Games*, pp. 270–280. Academic Press, New York (1967).
47. M. Gondran and M. Minoux, *Graphs and Algorithms* pp. 95–98. Wiley, Chichester (1984).
48. A. K. C. Wong, T. A. Reichert, D. N. Cohen and B. O. Aygun, A generalized method for matching informational macromolecular code sequences. *Comput. Biol. Med.* **4,** 43–57 (1974).
49. M. S. Westerman, General methods of sequence comparison. *Bull. Math. Biol.* **46,** 473–500 (1984).
50. Other definitions of molecular similarity are reviewed in Ref. [24] and in S. H. Bertz and W. C. Herndon, The similarity of graphs and molecules. In *Artificial Intelligence Applications in Chemistry* (Eds T. H. Pierce and B. A. Hohne). Chap. 15. ACS Symp. Series 306, American Chemical Society, Washington, D.C. (1986).