OPTOELECTRONICS AND ADVANCED MATERIALS – RAPID COMMUNICATIONS Vol. 4, No. 1, January 2010, p. 56 -58

On computing Merrifield–Simmons index

M. B. AHMADI*, Z. SEIF
Department of Mathematics, College of Sciences, Shiraz University, Iran

The Merrifield–Simmons index, $i(G)$, of a molecular graph G is important in structural chemistry and defined as the number of subsets of the vertex set, in which any two vertices are non-adjacent, i.e., the number of independent-vertex sets of G. In this paper for calculating Merrifield–Simmons index, we formulate an integer linear programming problem to compute the k-independent set of G and then we solve it by *AIMMS* software.

(Received November 05, 2009; accepted January 19, 2010)

Keywords: Merrifield–Simmons index, Molecular graph, *AIMMS* software, Integer programming

1. Introduction

A graph is a pair $G=(V,E)$ of sets such that, the elements of V are the vertices of the graph G and the elements of E are 2-element subset of V. The elements of E are called edges. A simple graph is a graph that does not have more than one edge between any two vertices and no edge starts and ends at the same vertex. Two vertices are adjacent if there is an edge between them. The adjacency matrix of a simple graph is a matrix with rows and columns labeled by graph vertices, with a 1 or 0 in position (i,j) according to whether vertices v_i and v_j are adjacent or not. Two vertices of G are said to be independent if they are not adjacent in G.

Topological indices are numerical parameters of a graph which characterize its topology and are usually graph invariant. Topological indices are used for example in the development of quantitative structure-activity relationships (QSARs) in which the biological activity or other properties of molecules are correlated with their chemical structure [1, 2, 3].

Merrifield-Simmons index is one of the most popular topological indices in chemistry, which was extensively studied in a monograph [4]. Chemical applications of this index were demonstrated. Merrifield and Simmons were able to show the correlation between this index and boiling points [4]. It was further investigated in various papers. In [5], Li et al. gave its other properties and applications.

Let G be a graph with n vertices. A *k*-independent set of G is a set of *k* mutually independent vertices. Denote by $i(G,k)$ the number of the *k*-independent sets of G. By definition, the empty vertex set is an independent set. Then $i(G,0)=1$, for any graph G. The Merrifield-Simmons index of G, denoted by $i(G)$, is defined by

$$i(G)=\sum_{k=0}^{n} i(G,k). \qquad (1)$$

So $i(G)$ is equal to the total number of the independent sets of G.

The goal of this article is to get a way for calculating $i(G,k)$ the finite molecular graphs for different amount of *k*. For this purpose, we formulate an integer linear programming problem (ILP) [6] whose number of optimal solutions is $i(G,k)$. An integer programming problem is a mathematical optimization problem in which all of the variables can take only the integer values. *AIMMS* Software helps us in solving our ILP and computing $i(G,k)$. *AIMMS* was introduced by Paragon Decision Technology as a new type of mathematical modeling tool. It is far more than just another mathematical modeling language [7].

2. Main results and discussion

In this section we show an integer linear programming for computing $i(G,k)$ of finite molecular graph G for different amount of k. In fact, the value of $i(G,k)$ is obtained by the number of optimal solutions of our ILP.

Let $G(V,E)$ be a graph with n vertices labeled with $v_1,\ldots,v_n$. Assumes that c_{ij} is the element of its adjacency matrix placed in ith row and jth column. The problem is to find *k*-independent set A of G.

To transcribe the problem into a formal ILP, let us denote the binary decision variables as follows:

$$x_i = \begin{cases} 1 & v_i \in A \\ 0 & v_i \notin A \end{cases} \quad (2)$$

We should now define the constraints of the problem, which are the restrictions imposed upon the values of the decision variables by the characteristics of the problem under study. First, since we hope that the set A has k elements, we should make the summation of decision variables exactly equal to k . i.e.

$$\sum_{i=1}^{n} x_i = k \quad (3)$$

Next, it should be noticed that the elements of A mustn't be adjacent. This restriction can be expressed as follows:

$$x_i + x_j \le 2 - c_{ij}, \quad i = 0,1,...,n, \quad i < j \quad (4)$$

If $c_{ij} = 1$, we have $x_i + x_j \le 1$. So, either x_i or x_j can be equal to 1, i.e., A can include just one of the vertices v_i and v_j. Otherwise, If $c_{ij} = 0$, we have $x_i + x_j \le 2$. Since the decision variables are binary, it is obvious that this constraint is always confirmed.

Now we should find x_i, $i = 1,...,n$ such that

$$\sum_{i=1}^{n} x_i = k$$
$$x_i + x_j \le 2 - c_{ij}, \quad i = 0,1,...,n, \quad i < j.$$
$$x_i \in \{0,1\}, \; i = 1,..,n$$

For finding the value of x_j, $j = 1,...,n$, we convert the above problem to an ILP. For this purpose, we add an artificial variable R to the constraint (3). The desired constraint will be:

$$\sum_{i=1}^{n} x_i + R = k \quad (5)$$

An artificial variable is required to be zero for feasible solution.

First, we deal with a "relaxed problem" where the artificial variable R is allowed to be greater than zero. If we can make $R = 0$, the solution will be feasible for the original problem. So, our goal is to minimize the amount of R. We can determine the objective function for ILP as follows:

$$Minimize \;\; z = R. \quad (6)$$

Then we have the following ILP

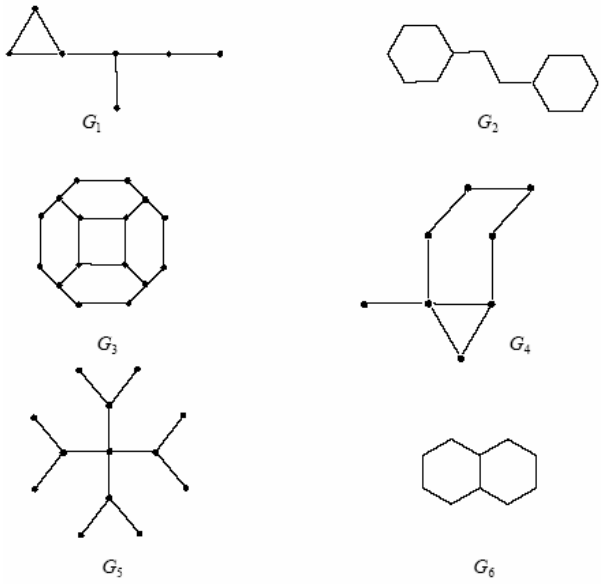
$$Minimize \;\; z = R$$
$$Subject\ to:$$
$$\sum_{i=1}^{n} x_i + R = k$$
$$x_i + x_j \le 2 - c(i,j) \quad for \; i = 1,\ldots,n \; and \; i < j \quad (7)$$
$$x_i \in \{0,1\} \quad for \; i = 1,\ldots,n$$
$$R \in \{0,1,...,n\}.$$

If the above ILP has feasible solutions, then after solving it, the amount of R will become zero. In this case, $i(G,k)$ will be equal to the number of optimal solutions of our ILP. Otherwise, $i(G,k)$ will be zero.

We solve our mathematical program by *AIMMS* software. This software is able to find all of the optimal solutions of our program.

3. Numerical examples

Let us consider the following graphs:



For these graphs, by solving ILP (7), one obtains the results shown in Table 1.

Table 1. Computing Merrifield–Simmons index for G_1 - G_6.

	i(G,0)	i(G,1)	i(G,2)	i(G,3)	i(G,4)	i(G,5)	i(G,6)	i(G,7)	i(G,8)	i(G,9)	i(G)
G_1	1	7	14	8	0	0	0	0	0	0	30
G_2	1	14	76	202	273	176	46	4	0	0	792
G_3	1	16	101	322	552	502	223	38	2	0	1757
G_4	1	8	19	15	4	0	0	0	0	0	47
G_5	1	13	66	172	251	214	114	40	9	1	881
G_6	1	10	34	46	21	2	0	0	0	0	114

References

[1] H. González-Díaz, S. Vilar, L. Santana, E. Uriarte, Current Topics in Medicinal Chemistry **7**(10), 1015 (2007).

[2] H. González-Díaz, Y. González-Díaz, L.Santana, F. M. Ubeira, E.Uriarte, Proteomics **8**(4), 750 (2008).

[3] Hall, H. Lowell, Kier, B. Lemont, Molecular connectivity in chemistry and drug research, Boston, Academic Press, 1976.

[4] R. E Merrifield, H. E. Simmons, Topological methods in chemistry, John Wiley & Sons, New York, 1989.

[5] X. Li, Z. Li, L. Wang, J. Comput. Biol. **10**, 47 (2003).

[6] M. S. Bazara, J. J. Jarvis, H. D. Sherali, Linear programming and network flows, Third edition, John Wiley & Sons 2005.

[7] J. Bisschop, AIMMS optimization modeling, Publisher: Lulu.com, 2006.

[8] H. Prodinger, R. F. Tichy, Fibonacci Quarterly, **20**, 16 (1982).

[9] Y.Wang, X. Li, I. Gutman, Inst. Math. **69**, 41 (2001).

[10] I. Gutman, Inst. Math. **52**, 5 (1992).

[11] H. Timmerman, T. Roberto, V. Consonni, R. Mannhold, H. Kubinyi, Handbook of Molecular Descriptors, Weinheim: Wiley-VCH, 2002.

[12] H. Ren, F, Zhang, Journal of Mathematical Chemistry **42**(4), 679 (2007).

*Corresponding author: mbahmadi.shirazu.ac.ir