Inter*net* Electronic Journal of Molecular Design

August 2002, Volume 1, Number 8, Pages 388–400

Editor: Ovidiu Ivanciuc

Special issue dedicated to Professor Milan Randić on the occasion of the 70th birthday
Part 4

Guest Editor: Mircea V. Diudea

On Walk Counts and Complexity of General Graphs

István Lukovits,[1] Ante Miličević,[2] Sonja Nikolić,[2] and Nenad Trinajstić[2]

[1] Chemical Research Center, Hungarian Academy of Sciences, P. O. Box 17, H–1525 Budapest, Hungary
[2] The Rugjer Bošković Institute, P.O. Box 180, HR–10002 Zagreb, Croatia

Received: April 10, 2002; Revised: July 4, 2002; Accepted: July 10, 2002; Published: August 31, 2002

Citation of the article:
I. Lukovits, A. Miličević, S. Nikolić, and N. Trinajstić, On Walk Counts and Complexity of General Graphs, *Internet Electron. J. Mol. Des.* **2002**, *1*, 388–400, http://www.biochempress.com.

Inter*net* Electronic Journal of Molecular Design

BioChem Press
http://www.biochempress.com

On Walk Counts and Complexity of General Graphs[#]

István Lukovits,[1] Ante Miličević,[2] Sonja Nikolić,[2] and Nenad Trinajstić[2,*]

[1] Chemical Research Center, Hungarian Academy of Sciences, P. O. Box 17, H–1525 Budapest, Hungary

[2] The Rugjer Bošković Institute, P.O. Box 180, HR–10002 Zagreb, Croatia

Received: April 10, 2002; Revised: July 4, 2002; Accepted: July 10, 2002; Published: August 31, 2002

Internet Electron. J. Mol. Des. **2002**, *1* (8), 388–400

Abstract

Motivation. This report was motivated by a recent work of Gutman, Rücker and Rücker on walks in simple molecular graphs, *i.e*., graphs without multiple edges and loops.

Method. Three methods for counting walks in general graphs, *i.e*., graphs with multiple bonds and loops, are presented: (*i*) graphical method based on the Morgan summation procedure, (*ii*) method based on augmented adjacency matrices of higher orders and (*iii*) method based on eigenvalues and eigenvectors of augmented adjacency matrices of higher orders. They represent extensions of the methods discussed previously in the literature for simple graphs.

Results. The total walk count (*twc*) was used as a measure for complexity of general graphs. It is shown that *twc* indices increase with size, branching, cyclicity, the number of loops and multiple bonds, and decrease with symmetry of general graphs.

Conclusions. The total walk count appears to be a valuable tool to account for complexity for several types of molecular graphs.

Keywords. Augmented adjacency matrix; complexity; general graph; total walk count; vertex–degree; walk; walk count; topological index; structural descriptor.

1 INTRODUCTION

This report was motivated by a recent work of Gutman, Rücker and Rücker [1] on walks in simple molecular graphs, *i.e*., graphs without multiple edges and loops, and by papers of Randić [2,3] and others [4–6] on complexity of (molecular) graphs. Since in these papers there is hardly any discussion about complexity of general graphs, we decided to report our work on complexity of these graphs using the graph theoretical concept of walks. General graphs [7] are graphs in which

[#] Dedicated to Professor Milan Randić on the occasion of the 70th birthday.

* Correspondence author; E–mail: trina@rudjer.irb.hr.

multiple edges and/or loops are allowed. These graphs can be used to represent molecules with multiple bonds and a variety of heteroatoms. Note that a loop is an edge with both of its vertices identical. A loop is also called a 1–cycle.

General graphs are sometimes called non–simple graphs [8]. Harary [9] calls graphs with multiple edges multigraphs and graphs with multiple edges and loops presudographs. Pseudographs are also called loop–multigraphs [10]. Cvetković, Doob and Sachs [11] use the term multigraphs for general graphs. In both of these terms, *i.e.*, multigraphs and general graphs, the misleading term "presudograph" or unwieldy term "loop–multigraph" does not appear. An illustrative example of a general molecular graph is given in Figure 1.

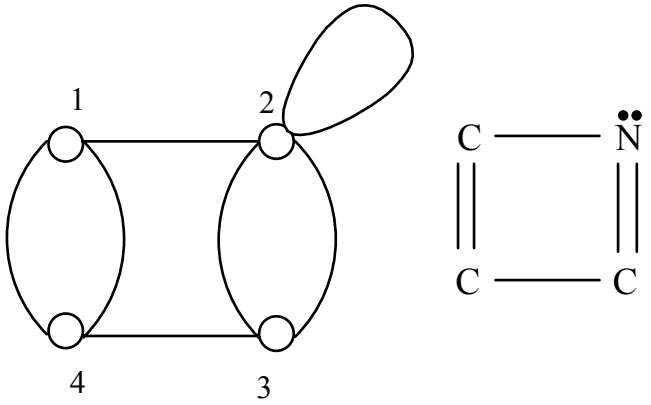


Figure 1. A labeled general molecular graph *G* corresponding to azacyclobutadiene.

A walk in a general graph is any sequence of consecutive edges and loops. The length (k) of the walk is the number of edges (E) and loops (L) in it. For example, there are many possible walks between vertices 1 and 3 in *G* given in Figure 1: 1–2–3 (k = 2), 1–2–2–3 (k = 3), 1–4–3 (k = 2), 1–4–1–2–3 (k = 4), etc.

The total walk count (*twc*) is the sum of (molecular) graph walk counts of different lengths (*mwc*):

$$twc = \sum_{k=1}^{k_{\max}} mwc_k \tag{1}$$

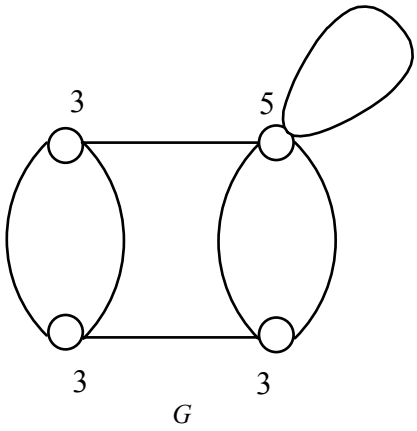
where mwc_k is the number of all walks of length k and $k_{\max}$ is the length of the longest walk agreed to consider in a given case.

The total walk count was found to be very useful as a molecular descriptor in QSPR modeling [12,13]. Similarly, *twc* is also used as a criterion for molecular complexity [6,14,15].

2 METHODS FOR THE ENUMERATION OF WALKS IN GENERAL GRAPHS

There are at least three methods to compute the walk counts and total walks counts in simple graphs [1]. However, in order to extend them to general graphs we need to use the augmented (weighted) adjacency matrix [16], *i.e.*, the adjacency matrix in which the non–zero elements appear on the diagonal, reflecting loops in a graph and some off–diagonal elements may have values greater than one, reflecting multiple edges in a graph. Randić and Basak utilized the augmented adjacency matrix in their work on the variable connectivity index [17].

The degree of a vertex in a (general) graph is equal to the number of edges meeting at this vertex. However, the degree of a vertex with a loop is taken to be the number of edges meeting at this vertex plus two for the loop, because the loop contributes twice to the number of edges incident at the vertex. Observe that the loop can be traversed in clockwise or anticlockwise fashion. Consider the vertex–degrees *G* given in Figure 1:



G

It can be seen that with these rules the handshaking lemma remains valid; and therefore the choice of weights is not arbitrary at all:

$$\sum_{i=1}^{V} d(i) = 3+3+3+5 = 2(E+L) = 2\cdot(6+1) \qquad (2)$$

where *d(i)* is the degree of vertex *i* and *V* is the number of vertices in *G*.

The following methods are used for counting walks in general graphs: (1) Graphical approach based on the Morgan summation procedure (this approach was used for simple graphs by Rücker and Rücker [12]); (2) The use of the augmented adjacency matrices of higher orders [16] (the number of elements in the *k*–th adjacency matrix is equal to *mwc* of the length *k*, mwc_k); and (3) using the eigenvectors and eigenvalues of augmented adjacency matrices for various values of *k*. Below we will illustrate each of these three methods.

2.1 Graphical Method

The graphical method for counting walks in a graph is described by Rücker and Rücker [12]. It is based on the Morgan extended connectivities (EC) [18], which are obtained from the degrees of all vertices in *G* by iterative summation over all neighbours. Rücker and Rücker [12] proved that the *EC*'s and mwc_k's are identical. This was first observed by Razinger [19].

The graphical procedure can be summarized in the following steps [12]. For a walk of length *1* starting at vertex *i*, there are *d*(*i*) possibilities. If one arrives to a particular vertex *j* by a walk of length *k* from vertex *i*, then the number of possibilities to extend this walk by one step is equal to degree of *j*. The sum of the degrees of all end–vertices *j* of walks of length *k* gives the number of walks of length *k* + 1. This can be formalized by the following expression:

$$mwc_{k+1}(i) = \sum_{j(i)} mwc_k(j) \tag{3}$$

where *j*(*i*) denotes a first neighbor of vertex *i* and the summation is over all such neighbors. The application of this graphical method to general graphs is presented in Figure 2.

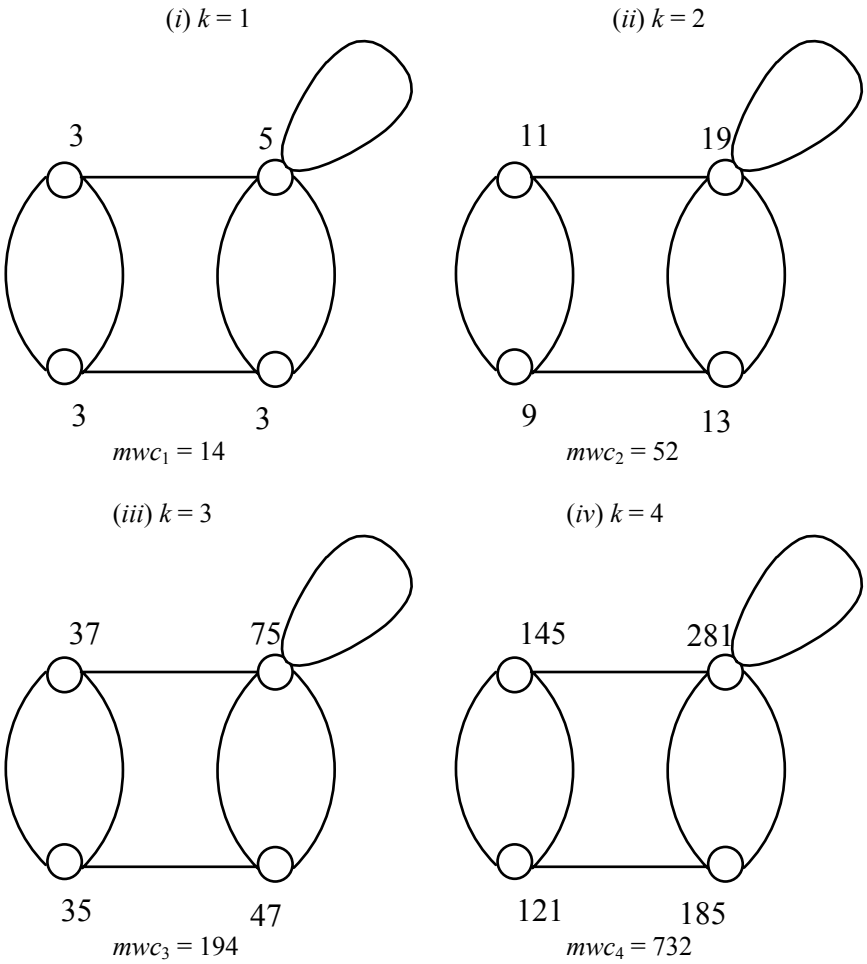


Figure 2. Application of the graphical method to counting walks with up to *k* = 6 in the graph *G* from Figure 1.

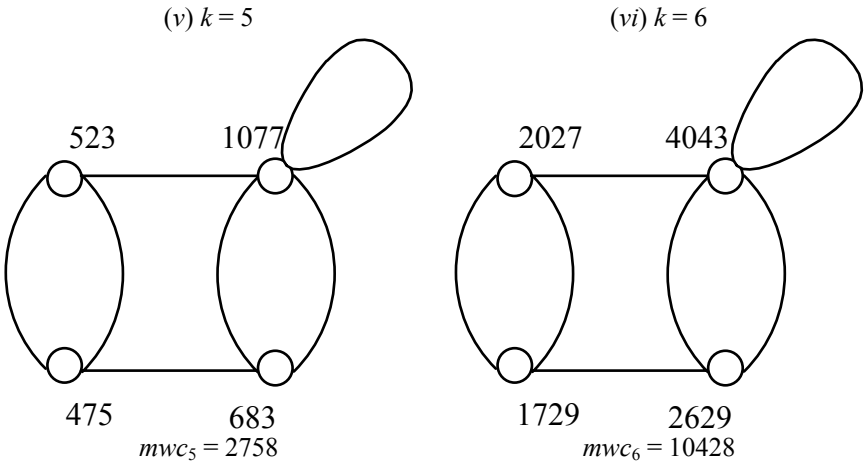


Figure 2. (Continued).

2.2 Method Based on the Augmented Adjacency Matrix

The use of the (augmented) adjacency matrix is based on the relationship between the matrix–elements and the number of walks, that is, $(\mathbf{A}^k)_{ij}$ is the number of walks of length k in G starting at vertex i and ending at vertex j. Thus mwc_k is given by:

$$mwc_k = \sum_{i=1}^{V}\sum_{j=1}^{V}(\mathbf{A}^k)_{ij} \tag{4}$$

The use of this method is illustrated in Table 1. The input data for our computer program are the vertex connectivities and k_{max}. The program automatically sets up the augmented adjacency matrix and produces matrices of higher orders (k = 2, 3,..., k_{max}) by straightforward matrix multiplications. The sums of the matrix–elements are equal to mwc_k for given k.

Table 1. Augmented Adjacency Matrix, Matrices of Higher Orders and Molecular Walk Counts with up to k = 6

(*i*) k = 1

$$\mathbf{A}^1 = \begin{bmatrix} 0 & 1 & 0 & 2 \\ 1 & 2 & 2 & 0 \\ 0 & 2 & 0 & 1 \\ 2 & 0 & 1 & 0 \end{bmatrix} \quad \begin{matrix} mwc_1 \\ 3 \\ 5 \\ 3 \\ \underline{3} \\ 14 \end{matrix}$$

(*ii*) k = 2

$$\mathbf{A}^2 = \begin{bmatrix} 5 & 2 & 4 & 0 \\ 2 & 9 & 4 & 4 \\ 4 & 4 & 5 & 0 \\ 0 & 4 & 0 & 5 \end{bmatrix} \quad \begin{matrix} mwc_2 \\ 11 \\ 19 \\ 13 \\ \underline{9} \\ 52 \end{matrix}$$

(*iii*) k = 3

$$\mathbf{A}^3 = \begin{bmatrix} 2 & 17 & 4 & 14 \\ 17 & 28 & 22 & 8 \\ 4 & 22 & 8 & 13 \\ 14 & 8 & 13 & 0 \end{bmatrix} \quad \begin{matrix} mwc_3 \\ 37 \\ 75 \\ 47 \\ \underline{35} \\ 194 \end{matrix}$$

(*iv*) k = 4

$$\mathbf{A}^4 = \begin{bmatrix} 45 & 44 & 48 & 8 \\ 44 & 117 & 64 & 56 \\ 48 & 64 & 57 & 16 \\ 8 & 56 & 16 & 41 \end{bmatrix} \quad \begin{matrix} mwc_4 \\ 145 \\ 281 \\ 185 \\ \underline{121} \\ 732 \end{matrix}$$

Table 1. (Continued)

(*v*) $k = 5$

$$\mathbf{A}^5 = \begin{bmatrix} 60 & 229 & 96 & 138 \\ 229 & 406 & 290 & 152 \\ 96 & 290 & 144 & 153 \\ 138 & 152 & 153 & 32 \end{bmatrix} \quad \begin{matrix} mwc_5 \\ 523 \\ 1077 \\ 683 \\ \underline{475} \\ 2758 \end{matrix}$$

(*vi*) $k = 6$

$$\mathbf{A}^6 = \begin{bmatrix} 505 & 710 & 596 & 216 \\ 710 & 1621 & 964 & 748 \\ 596 & 964 & 733 & 336 \\ 216 & 748 & 336 & 429 \end{bmatrix} \quad \begin{matrix} mwc_6 \\ 2027 \\ 4043 \\ 2629 \\ \underline{1729} \\ 10428 \end{matrix}$$

2.3 Method Based on Eigenvectors and Eigenvalues of Augmented Adjacency Matrices for Various Values of *k*

This method is based on the relationship between walks of length *k*, eigenvectors (c_{ri}; $i = 1, 2, ..., N$) and eigenvalues λ_r^k of the augmented adjacency matrices with various values of exponent *k* [1]:

$$mwc_k = \sum_{r=1}^{V} \sigma_r \lambda_r^k \tag{5}$$

where

$$\sigma_r = (c_{r1} + c_{r2} + ... + c_{rV})^2 \tag{6}$$

Note that if λ_r is a degenerate eigenvalue, then the sum of σ_r over all degenerate eigenvectors is a true graph invariant [1].

Our computer program mentioned above gives also eigenvectors and eigenvalues of the corresponding augmented matrices. Therefore, it contributes walk counts by using equations (5) and (6). This is shown in Table 2.

Table 2. Walk Counts Calculated up to $k = 6$ using Graph Eigenvectors and Eigenvalues for the *G* from Figure 1. Below Each Set of Data the *mwc* Values are Given

(*i*) $k = 1$
Eigenvalues
3.7792 1.4906 –0.5980 –2.6718
Coefficients of Eigenvectors
0.3630 0.5753 0.4628 –0.5684
0.7368 –0.4562 0.3827 0.3203
0.4739 –0.1715 –0.7285 –0.4640
0.3175 0.6569 –0.3297 0.5992
$mwc_1 = 14.0000000$

(ii) $k = 2$
Eigenvalues
14.2821 7.1385 2.2219 0.3576
Coefficients of Eigenvectors
0.3630 –0.5684 0.5753 0.4628
0.7368 0.3203 –0.4562 0.3827
0.4739 –0.4640 –0.1715 –0.7285
0.3175 0.5992 0.6569 –0.3297
$mwc_2 = 52.0000000$

Table 2. (Continued)

(*iii*) $k = 3$
Eigenvalues
53.9743 3.3120 –0.2138 –19.0724
Coefficients of Eigenvectors
–0.3630 0.5753 0.4628 0.5684
–0.7368 –0.4562 0.3827 –0.3203
–0.4739 –0.1715 –0.7285 0.4640
–0.3175 0.6569 –0.3297 –0.5992
$mwc_3 = 194.0000000$

(*iv*) $k = 4$
Eigenvalues
203.9778 50.9576 4.9368 0.1279
Coefficients of Eigenvectors
–0.3630 0.5684 0.5753 0.4628
–0.7368 –0.3203 –0.4562 0.3827
–0.4739 0.4640 –0.1715 –0.7285
–0.3175 –0.5992 0.6569 –0.3297
$mwc_4 = 732.0000000$

(*v*) $k = 5$
Eigenvalues
770.8655 7.3588 –0.0765 –136.1478
Coefficients of Eigenvectors
–0.363 0.5753 0.4628 0.5684
–0.7368 –0.4562 0.3827 –0.3203
–0.4739 –0.1715 –0.7285 0.4640
–0.3175 0.6569 –0.3297 –0.5992
$mwc_5 = 2758.0000000$

(*vi*) $k = 6$
Eigenvalues
2913.2270 363.7583 10.9690 0.0457
Coefficients of Eigenvectors
–0.3630 0.5684 0.5753 0.4628
–0.7368 –0.3203 –0.4562 0.3827
–0.4739 0.4640 –0.1715 –0.7285
–0.3175 –0.5992 0.6569 –0.3297
$mwc_6 = 10428.0000000$

All three methods reviewed above give, of course, identical *mwc* and *twc* values. However, of all three methods exposed above, the most convenient method to calculate these indices automatically is the third method. Our computer program based on this method consists of the following steps: (*a*) Input data – vertex connectivities and the value of k_{max}; (*b*) setting up the augmented adjacency matrices with various exponents corresponding to lengths of walks we wish to calculate and getting their eigenvectors and eigenvalues; (*c*) calculation of molecular walk counts using equations (5) and (6); and (*d*) calculation of total walk counts using equation (1). Printouts list the values of *mwc* and *twc* for walks of various lengths.

3 COMPLEXITY OF GENERAL (MOLECULAR) GRAPHS

The notion of complexity and its antonym, simplicity, are topics which belong basically to the realm of philosophy. Nevertheless, complexity of chemical structures has been studied by means of graph invariants (topological indices) [2–6]. Most authors agree that any measure used to study complexity of structures encountered in chemistry should satisfy the homology principle, *i.e.*, the index should increase with the graph (molecular) size and reflect the principal structural features such as branching, cyclicity, multiple edges, loops and symmetry. These structural features are interconnected. Therefore, if we wish to study the influence of one particular structural feature on molecular complexity, it will be practically impossible to keep other structural features constant, though we may minimize their influence on the complexity of a series of molecules under the considerations. This is what we will do here.

3.1 Influence of Size

To study the influence of the size on complexity of general graphs we have selected four acyclic general graphs shown in Figure 3. The increase in the size is reflected in the increase of the number of vertices and edges, and consequently in the lengths of walks and in the values of *twc* indices. We computed walks with up to the largest self–returning walk (a walk that starts and ends at the same vertex) which does not repeat itself unnecessarily *in toto* or in parts, the length of which is denoted by k_{srw} (in this case k_{srw} is k_{max}). For example, such a self–returning walk in graph **1** from Figure 3 is 1–2–3–3–2–1 (k_{srw} = 5). There are, of course, longer self–returning walks possible, such as in **1** 1–2–1–2–3–3–2–1, but this walk has a part that repeats itself. Thus, in our example given in Figure 3, *twc* is computed in **1** for $k = 1 - k_{srw}$ (= 5), in **2** for $k = 1 - k_{srw}$ (= 7), in **3** for $k = 1 - k_{srw}$ (= 9) and in **4** for $k = 1 - k_{srw}$ (= 11).

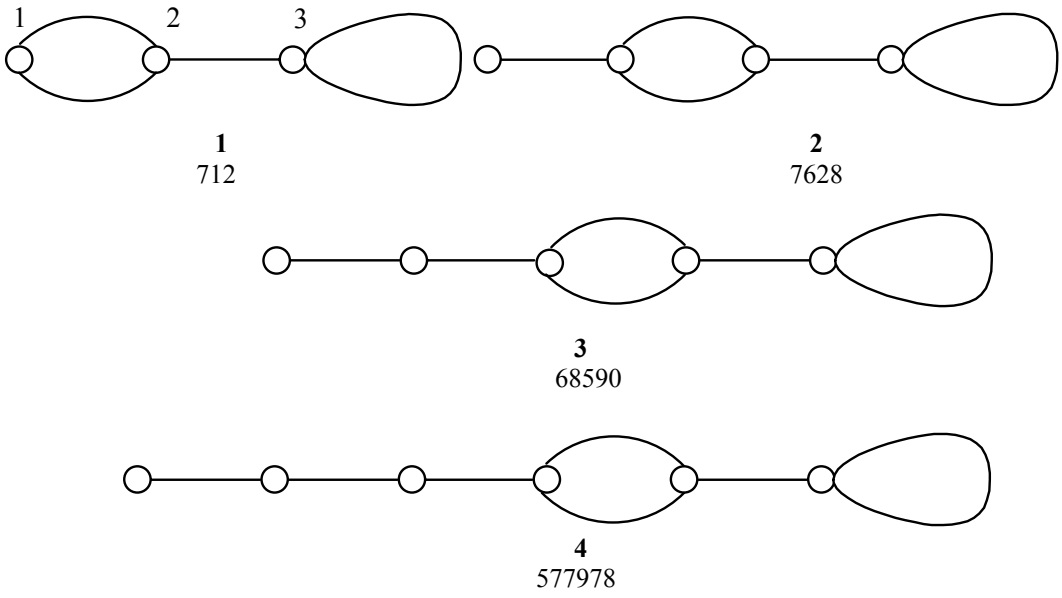


Figure 3. Four acyclic general graphs and the values of their *twc* indices.

Of course, it could be argued, for example, that two–edge connection between adjacent vertices is a 2–cycle, therefore the graphs that we consider are not acyclic. This is true in mathematical sense. However, a two–edge connection corresponds to a double bond which is not considered as a cycle in chemistry. The *twc* index ordered these graphs as expected **1** < **2** < **3** < **4**, indicating the enormous increase in complexity on going from **1** to **4**. This is the same order that is given by the size in terms of either edges or vertices or both. Therefore, the increase in the general graph size causes the increase in the graph complexity. This agrees with studies on simple graphs.

3.2 Influence of Branching

We again consider acyclic general graphs. Two branched trees and the corresponding unbranched tree are given in Figure 4. The same computation as above was carried out. Thus, *twc* indices for graphs shown in Figure 4 were computed in **5** for $k = 1– k_{srw}$ (= 13), in **6** for $k = 1– k_{srw}$ (= 13) and in **7** for $k = 1– k_{srw}$ (= 13).

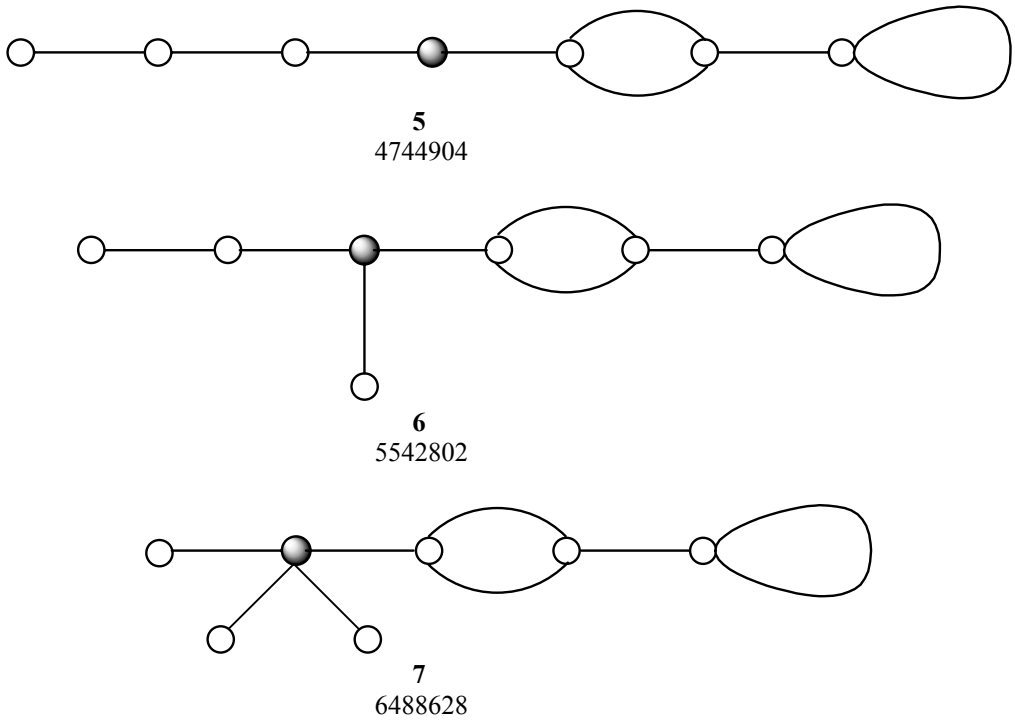


Figure 4. Unbranched acyclic graph **5**, two branched graphs **6** and **7** and their *twc* indices, computed for walks with up $k_{srw} = 13$. The protobranched vertex in **5** and branched vertices in **6** and **7** are denoted by black dots.

The *twc* index ordered these graphs as **5** < **6** < **7**, indicating that graph complexity increases with branching. Branching is an intuitive concept not uniquely defined, though it can be identified through the appearance of vertices of degrees three and higher. Thus, the same order is obtained if we consider the degree of the branched vertex.

3.3 Influence of Cyclicity

In studying the influence of cyclicity on the general graphs complexity we will discard 1– and 2–cycles from the consideration in agreement with the above. We selected three general cyclic graphs, shown in Figure 5.

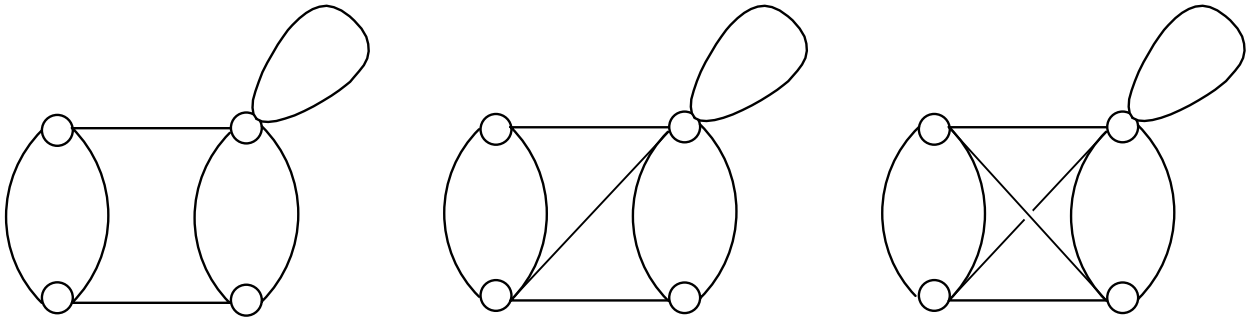


Figure 5. Three general cyclic graphs with their *twc* indices, given below each graph, computed for walks with up to $k_{max} = 10$.

There is a dramatic increase in the values of *twc* with the increase of the number of cycles. Graph **8** has four 4–cycles (in Figure 6 we indicate these four cycles), graph **9** has four 3–cycles and four 4–cycles and graph **10** has eight 3–cycles and four 4–cycles. Thus, the complexity of these graphs increases with the number cycles (and also with their size in terms of edges): **8** < **9** < **10**. This is so because with the increase in the number of cycles the number of walks also increases. It should also be noted that the three cyclic graphs considered differ in their sizes in terms of edges. Therefore, considering the conclusion reached in section 3.1 the same complexity order can be reached by considering only the number of edges in these graphs. Gratifying is that the conclusions reached by considering cyclicity and graph sizes are not contradictory.

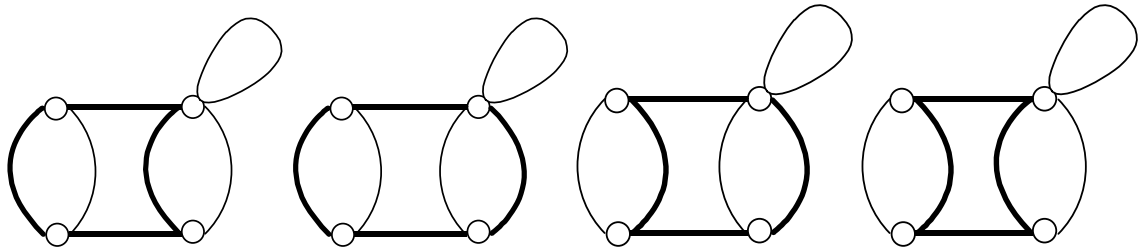


Figure 6. Four underlying 4-cycles in graph **8**.

3.4 Influence of Loops

We have investigated the influence of loops on the graph complexity considering a graph with multiple bonds, but without a loop (**11**) and the related graph with a loop (**12**), see Figure 7.

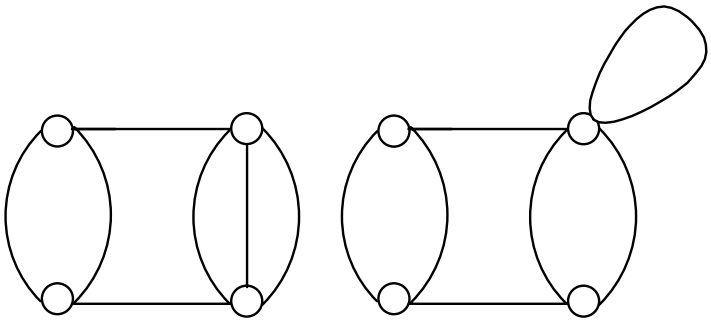


Figure 7. General graphs with and without loop. Their *twc* indices are given beneath each graph and are computed for walks with up to $k = 8$ (selected as k_{max} in this case).

Judging from the values of *twc* given in Figure 7, loops increase considerably the complexity of a graph. It is interesting to note that the position of a loop also influences the complexity of a graph. This is illustrated in Figure 8.

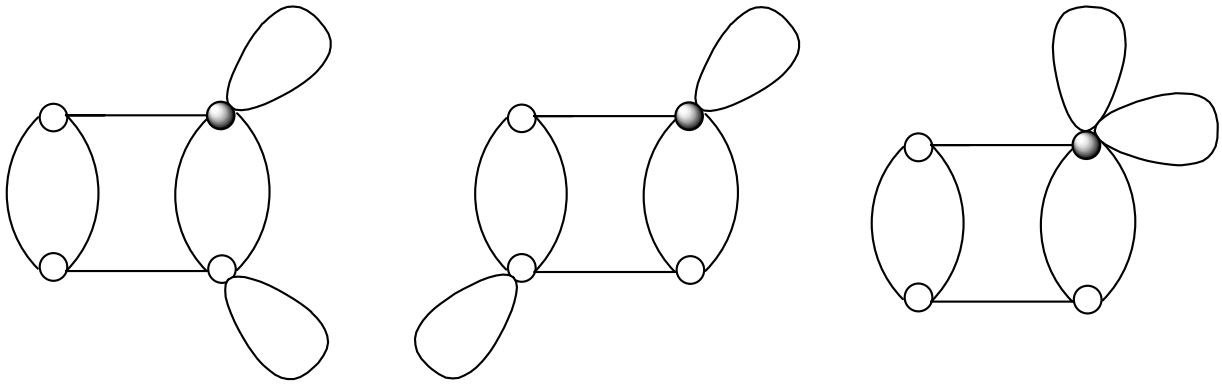


Figure 8. Three general graphs in which positions of two loops are *cis* (**13**), *trans* (**14**) and on the same vertex (**15**). Their *twc* indices are given below each graph and are computed for walks with up to $k = 9$ (k_{max}).

All these three graphs have the same size, but the positions of loops vary and strongly influence the complexity of the graphs. The most complex is graph **15** in which both loops are on the same vertex, while *cis*–position of loops leads to more complex graph than the *trans*–position; the complexity order for these graphs is **14** < **13** < **15**. The much greater value of the *twc* index for **15** may also be attributed in part to the increase of the degree of the vertex denoted by black dot in the figure.

3.5 Influence of Multiple Edges

Multiple edges increase complexity of general graphs. This is illustrated in Figure 9. The complexity order of these graphs: **16** < **17** < **18** is also predicted by the size increase in terms of edges. However, even if we keep the same size of graphs in terms of vertices and edges, as in graphs **8** and **18**, the graph with triple edges (**18**; *twc* = 117589258) is much more complex than the

graph with two double edges (**8**; *twc* = 2890274); their *twc* values were computed for walks with up to $k = 10$.

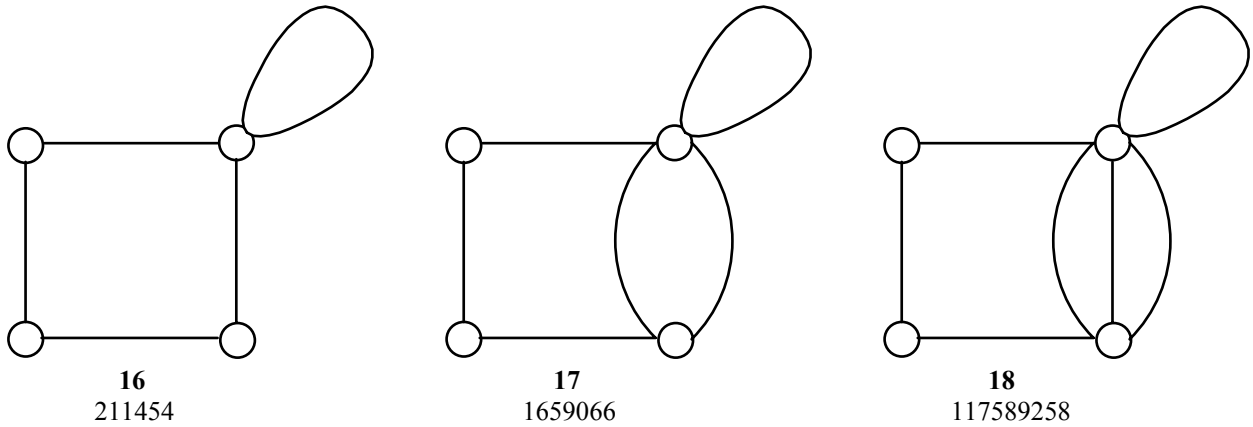


Figure 9. Graph **16** with all single edges between vertices, graph **17** with one double edge and graph **18** with one triple edge. Their *twc* indices are given below each graph and are computed for walks with up to $k = 10$ (k_{max}).

3.6 Influence of Symmetry

We considered two graphs: one asymmetric (**19**) and one symmetric (**20**). These graphs and the corresponding twc values, computed for walks with up to $k = 10$, are given in Figure 10. As expected asymmetric graph **19** is a more complex structure than symmetric graph **20**. The same observation was also made in the case of simple graphs [6].

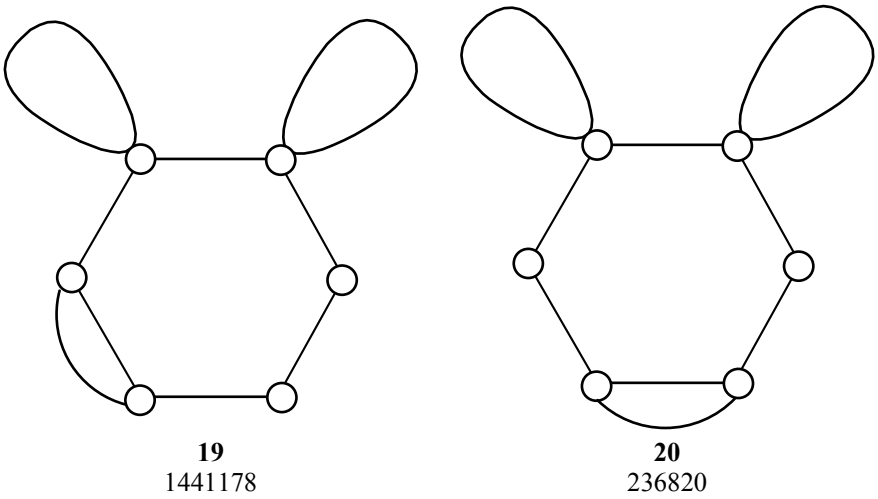


Figure 10. Asymmetric graph **19**, symmetric graph **20** and their *twc* values computed for walks with up to $k = 10$ (k_{max}).

4 CONCLUSIONS

We have reviewed methods for computing walks in simple graphs (graphs without multiple edges and loops) and have shown that these can be extended to general graphs (graphs with multiple edges and loops) using the concepts of augmented distance matrix and Morgan summation

procedure extended to embrace loops. We also developed a computer program for calculating the total walk count (the *twc* index). This index has been used in the literature among other things as a numerical criterion for molecular graph complexity. Here we applied this index to complexity of various general graphs. The following results were obtained: *twc* indices and molecular graph complexities increase with size, branching, cyclicity, loops and multiple bonds and with lower symmetry of general graphs. The same conclusions are reached for simple graphs. However, loops and multiple bonds increase profoundly the complexity of general graphs due to many more possibilities for walks that do not exist in simple graphs.

5 REFERENCES

[1] I. Gutman, C. Rücker, and G. Rücker, On Walks in Molecular Graphs, *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 739–745.
[2] M. Randić, On Complexity of Transitive Graphs Representing Degenerate Rearrangements, *Croat. Chem. Acta* **2001**, *74*, 683–705.
[3] M. Randić and D. Plavšić, On the Concept of Molecular Complexity, *Croat. Chem. Acta* **2002**, *75*, 107–116.
[4] D. Bonchev, Novel Indices for the Topological Complexity of Molecules, *SAR QSAR Environ. Res.* **1997**, *7*, 23–43.
[5] S. H. Bertz and C. M. Zamfirescu, New Complexity Indices Based on Edge Covers, *MATCH* (*Commun. Math. Comput. Chem.*) **2000**, *42*, 39–70.
[6] S. Nikolić, N. Trinajstić, I. M. Tolić, G. Rücker, and C. Rücker, On Molecular Complexity Indices; in: *Complexity in Chemistry*, Eds. D. Bonchev and D. H. Rouvray, Taylor and Francis, London, 2002, in press.
[7] R. J. Wilson, *Introduction to Graph Theory*, Oliver & Boyd, Edinburgh, 1972, p. 10.
[8] R. J. Wilson and J. J. Watkins, *Graphs – An Introductory Approach*, Wiley, New York, 1900, p. 10.
[9] F. Harary, *Graph Theory*, Addison–Wesley, 2nd printing, Reading, MA, 1971, p. 10.
[10] G. Chartrand, *Graphs as Mathematical Models*, Prindle, Weber & Schmidt, Boston, MA, 1977, p. 22.
[11] D. M. Cvetković, M. Doob, and H. Sachs, *Spectra of Graphs*, 3rd edition, Barth, Heidelberg, 1995, p. 11.
[12] G. Rücker and C. Rücker, Counts of All Walks as Atomic and Molecular Descriptors, *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 683–695.
[13] G. Rücker and C. Rücker, On Topological Indices, Boiling Points, and Cycloalkanes, *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 788–802.
[14] G. Rücker and C. Rücker, Walk Counts, Labyrinthicity, and Complexity, *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 99–106.
[15] G. Rücker and C. Rücker, Substructure, Subgraph, and Walk Counts as Measures of the Complexity of Graphs and Molecules, *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 1457–1462.
[16] N. Trinajstić, *Chemical Graph Theory*, 2nd revised ed., CRC Press, Boca Raton, FL, 1992, pp. 41–48.
[17] M. Randić and S. C. Basak, On Use of the Variable Connectivity Index $^{1}\chi^{f}$ in QSAR: Toxicity of Aliphatic Ethers, *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 614–618.
[18] H. L. Morgan, The Generation of a Unique Machine Description for Chemical Structures – A Technique Developed at Chemical Abstracts Service, *J. Chem. Doc.* **1965**, *5*, 107–113.
[19] M. Razinger, Extended Connectivities in Chemical Graphs, *Theor. Chim. Acta* **1982**, *61*, 581–586.