

Balaban index of some nanotubes by GAP program

A. TEHRANIAN, A. IRANMANESH^{a*}, Y. ALIZADEH^a

Science and Research Branch, Islamic Azad University, Tehran, Iran

^aDepartment of Mathematics, Tarbiat Modares University, Tehran, Iran

In this paper, we give an algorithm for computing the Balaban index of any graph. Also we compute the Balaban index of $HAC_5C_7[p,q]$ and $VC_5C_7[p,q]$ nanotubes by GAP program.

(Received March 17, 2010; accepted July 14, 2010)

Keywords: Balaban index, Gap program, $HAC_5C_7[p,q]$ nanotubes, $VC_5C_7[p,q]$ Nanotube

1. Introduction

A topological index is a single unique number characteristic of the molecular graph and is mathematically known as the graph invariant. The topological index of a molecule is a non-empirical numerical quantity that quantifies the structure and the branching pattern of the molecule. Therefore, the topological analysis of a molecule involves translating its molecular structure into a characteristic unique number (or index) that may be considered a descriptor of the molecule under examination. Usage of topological indices in biology and chemistry began in 1947 when chemist Harold Wiener [1] introduced Wiener index to demonstrate correlations between physicochemical properties of organic compounds and the index of their molecular graphs. The Balaban index of a molecular graph G is defined by Balaban [2]. The Balaban index appears to be a very useful molecular descriptor with attractive properties[3,4].

Let G be a connected graph. The vertex-set and edge-set of G denoted by $V(G)$ and $E(G)$ respectively. The distance between the vertices u and v , $d(u,v)$, in a graph is the number of edges in a shortest path connecting them. Two graph vertices are adjacent if they are joined by a graph edge. The degree of a vertex $i \in V(G)$ is the number of vertices joining to i and denoted by δ_i .

The Balaban index of a graph G is denoted by $J(G)$ and defined as $J(G) = \frac{m}{\mu+1} \sum_{ij \in E(G)} \frac{1}{\sqrt{d(i)d(j)}}$, where m is the number of edges of G and $\mu(G)$ is the cyclomatic number of G . Note that the cyclomatic number is the minimum number of edges that must be removed from G in order to transform it to an acyclic graph; it can be calculated using $\mu(G) = m - n + 1$ where n is the number of vertices and $d(i)$ is the sum of distances between vertex i and all other vertices of G , and the summation goes over all edges from the edge set $E(G)$. In a series of papers some topological indices are computed for some other nanotubes[5-18].

In this paper, we give an algorithm for computing the Balaban index of any graph. Also we compute the Balaban index of $HAC_5C_7[p,q]$ and $VC_5C_7[p,q]$ nanotubes by GAP program [19].

2. An algorithm for computing the Balaban index of any graph

In this section we present an algorithm for obtaining the Balaban index of any graph. For this purpose the following algorithm is presented:

- 1- We assign to any vertex of the graph one number.
- 2- We determine all of adjacent vertices set of the vertex u , $u \in V(G)$ and this set denoted by $N(u)$.

The set of vertices that their distance to vertex u is equal to t ($t \geq 0$) is denoted by $D_t(u)$ and consider $D_0(u) = \{u\}$. We have the following relations:

- $V(G) = \bigcup_{t \geq 0} D_t(u), \forall u \in V(G)$
- $d(u) = \sum_{v \in V(G)} d(u,v) = \sum_{t \geq 1} t \times |D_t(u)|, \forall u \in V(G)$
- $\delta_u = |N(u)|$

According to the above relations, we can obtain the Balaban index by determining $D_t(u)$ for every vertex u .

- 3- The distance between vertex i and its adjacent vertices is equal to 1, therefore $D_{i,1} = N(i)$. For each $j \in D_{i,t}, t \geq 1$, the distance between each vertex of set $N(j) \setminus (D_{i,t} \cup D_{i,t-1})$ and the vertex i is equal to $t+1$, thus we have :

$$D_{i,t+1} = \bigcup_{j \in D_{i,t}} (N(j) \setminus (D_{i,t} \cup D_{i,t-1})), t \geq 1.$$

According to the above equation we can obtain $D_{i,t}, t \geq 2$, for each $i \in V(G)$.

3. Results and discussion

A C_5C_7 net is a trivalent decoration made by alternating C_5 and C_7 .

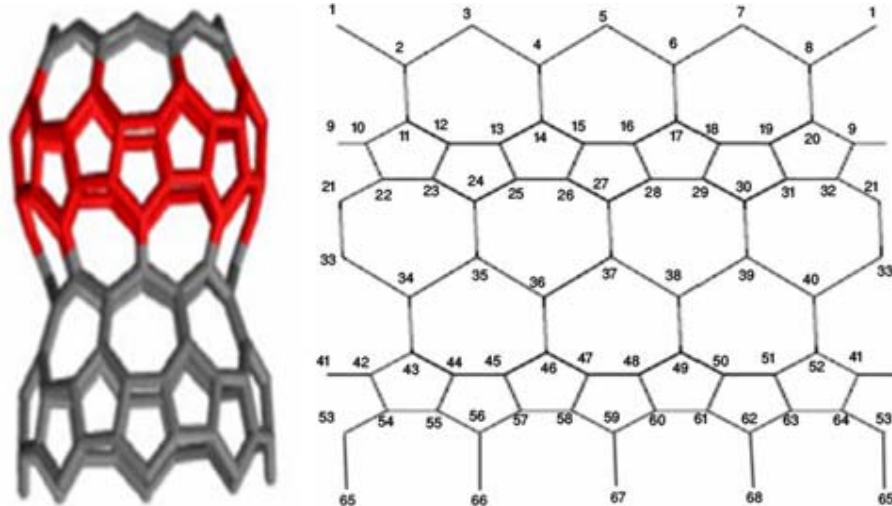


Fig. 1. $HAC_5C_7[4, 2]$ nanotube.

We denote the number of heptagons in one row by p . In this nanotube, the three first rows of vertices and edges are repeated alternatively, and we denote the number of this repetition by q . In each period there are $8p$ vertices and p vertices which are joined to the end of the graph and hence the number of vertices in this nanotube is equal to $8pq + p$. We partition the vertices of this graph to following sets:

K_1 : The vertices of first row whose number is $2p$.

K_2 : The vertices of the first row in each period except the first one whose number is $2p(q-1)$.

K_3 : The vertices of the second rows in each period whose number is $3pq$.

K_4 : The vertices of the third row in each period whose number is $3pq$.

K_5 : The last vertices of the graph whose number is p .

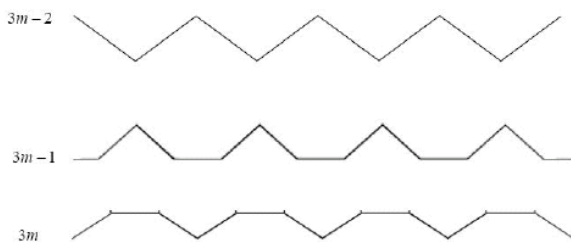


Fig. 2. Rows of m -th period.

It can cover either a cylinder or a torus. In this section we compute the Balaban index of $HAC_5C_7[p, q]$ nanotubes by GAP program.

We write a program to obtain the adjacent vertices set to each vertex in the sets K_i , $i=1..5$. We can obtain the adjacent vertices set to each vertex by the joint of these programs. In this program, the value of x is the assign number of vertex i in that period.

The following program computes the Balaban index of $HAC_5C_7[p, q]$ nanotubes for arbitrary p and q .

```

p:=7; q:=7;#(for example)
n:=8*p*q+p;
N:=[];
K1:=[1..2*p];
V1:=[2..2*p-1];
N[1]:=[2,2*p]; N[2*p]:=[2*p-1,5*p,1];
for i in V1 do
  if i mod 2=0 then N[i]:=[i-1,i+1,3/2*i+2*p];
  else N[i]:=[i-1,i+1];fi;
od;
k:=[2*p+1..8*p*q];
k2:=Filtered(k,i->i mod (8*p) in [1..2*p]);;
for i in k2 do
  x:= i mod (8*p);
  if x mod 2 =1 then N[i]:=[i-1,i+1,(x-1)*(3/2) +1+i-x-3*p];
  else N[i]:=[i-1,i+1,x*(3/2) +2*p+i-x];fi;
  if x=1 then N[i]:=[i+1,i-1+2*p,i-3*p];fi;
  if x=2*p then N[i]:=[i-1,i+3*p,i-2*p+1];fi;
od;
k3:=Filtered(k,i->i mod (8*p) in [2*p+1..5*p]);;
for i in k3 do
  x:=i mod (8*p);
  if (x-(2*p)) mod 3 =1 then N[i]:=[i-1,i+1,i+3*p-1];
  elif (x-(2*p)) mod 3 =2 then N[i]:=[i-1,i+1,i+3*p];

```

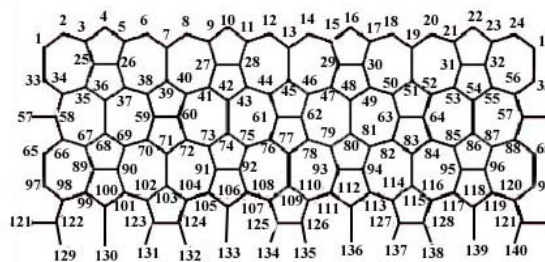
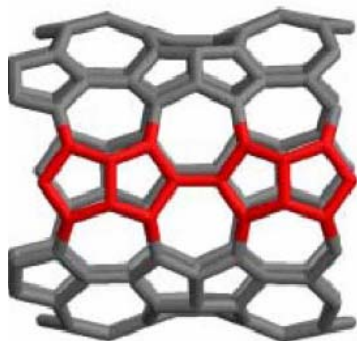
```

    elif (x-(2*p)) mod 3 =0 then N[i]:=[i-1,i+1,(2/3)*(x-
2*p)+i-x];fi;
    if x=2*p+1 then N[i]:=[i-1+3*p,i-1+6*p,i+1];fi;
    if x=5*p then N[i]:=[i-3*p,i-3*p+1,i-1];fi;
    od;
    k4:=Filtered(k,i->i mod (8*p) in Union([5*p+1..8*p-
1],[0]));
    for i in k4 do
    x:=i mod (8*p);
    if (x-(5*p)) mod 3 =1 then N[i]:=[i-1,i+1,(x-(5*p)-
1)*(2/3)+1+(i-x)+8*p];
    elif (x-(5*p)) mod 3 =2 then N[i]:=[i-1,i+1,i-3*p];
    elif (x-(5*p)) mod 3 =0 then N[i]:=[i-1,i+1,i-3*p+1];fi;
    if x=5*p+1 then N[i]:=[i+3*p-1,i+1,i+3*p];fi;
    if x=0 then N[i]:=[i-1,i-3*p+1,i-6*p+1];fi;
    od;
    K5:=[8*p*q+1 ..8*p*q+p];
    for i in K5 do
    x:=i-8*p*q;
    y:=8*p*(q-1)+5*p+3*x-2;
    N[i]:=[y];
    N[y][3]:=i;
    od; v:=[]; D:=[];
    for i in [1..n] do
    D[i]:=[]; u:=[i]; D[i][1]:=N[i]; v[i]:=Size(N[i]);
    u:=Union(u,D[i][1]);
    r:=1; t:=1;
    while r<>0 do
    D[i][t+1]:=[];
    for j in D[i][t] do
    for m in Difference (N[j],u) do
    AddSet(D[i][t+1],m);
    od; od;
    u:=Union(u,D[i][t+1]);
    if D[i][t+1]=[] then r:=0;fi;
    t:=t+1;
    od;od;
    m:=(1/2)*Sum(v)-n+1;
    d:=[];
    for i in [1..n] do
    d[i]:=0;
    for t in [1..Size(D[i])] do
    d[i]:=d[i]+t*Size(D[i][t]);
    od; od;
    b:=0;
    for i in [1..n] do
    for j in N[i] do
    b:=b+ER(1/((d[i]*d[j])););
    od; od;
    b:=m*b/2;#(this value is equal to Balaban index of the
graph)

```

Table 1. The Balaban index of $HAC_5C_7[p,q]$ nanotubes.

p	q	Balaban index
2	2	8.757
3	2	8.13913
4	3	5.62539
5	4	4.27636
3	5	3.66101
5	5	3.53814
6	6	2.94685
5	7	2.60890
7	7	2.52439

Fig. 3. $VC_5C_7[p,q]$ nanotube.

In continue, we compute the Balaban index of $VC_5C_7[p,q]$ Nanotubes by GAP program.

We denote the number of pentagons in the first row by p , in this nanotube the four first rows of vertices and edges are repeated alternatively, we denote the number of this repetition by q . In each period there are $16p$ vertices and $3p$ vertices which are joined to the end of the graph and hence the number of vertices in this nanotube is equal to $16pq+3p$.

The following program computes the Balaban index of $VC_5C_7[p,q]$ nanotubes for arbitrary p and q .

```

p:=10; q:=3; # (for example)
n:=16*p*q+3*p;
N:=[]; K1:=[1..6*p]; V1:=[2..6*p-1];
for i in V1 do
    if i mod 6=1 then N[i]:=[i-1,i+1,i+8*p];
    elif i mod 6 in [0,2,4] then N[i]:=[i-1,i+1];
    elif i mod 6=3 then N[i]:=[i-1,i+1,(1/3)*(i-3)+6*p+1];

```

```

    elif i mod 6=5 then N[i]:=[i-1,i+1,(1/3)*(i-5)+6*p
+2];fi;
od;
N[1]:=[2,6*p,8*p+1];          N[6*p]:=[6*p-1,1];
K:=[6*p+1..16*p*q];
K2:=Filtered(K,i->i mod (16*p) in [1..6*p]);
for i in K2 do x:= i mod (16*p);
  if x mod 6=1 then N[i]:=[i-1,i+1,i+8*p];
  elif x mod 6=2 then N[i]:=[i-1,i+1,(1/3)*(x-2)+2+i-x-
2*p];
  elif x mod 6=3 then N[i]:=[i-1,i+1,(1/3)*(x-3)+1+i-
x+6*p];
  elif x mod 6=4 then N[i]:=[i-1,i+1,i-8*p];
  elif x mod 6=5 then N[i]:=[i-1,i+1,(1/3)*(x-5) +2+i-
x+6*p];
  elif x mod 6=0 then N[i]:=[i-1,i+1,(1/3)*x +1+i-x-
2*p];fi;
  if x=1 then N[i]:=[i+1,i-1+6*p,i+8*p];fi; if x=6*p then
N[i]:=[i-1,i- 6*p+1,i- 8*p+1];fi;
od;
K3:=Filtered(K,i->i mod (16*p) in [6*p+1..8*p]);
for i in K3 do x:=(i-6*p) mod (16*p);
  if x mod 2=0 then N[i]:=[i-1,3*(x-2)+5+i-x- 6*p,3*(x-
2)+5+i-x+2*p];
  else N[i]:=[i+1,2*x+i- 6*p,2*x+i+2*p];fi;
od;
K4:=Filtered(K,i->i mod (16*p) in [8*p+1..14*p]);
for i in K4 do x:=(i- 8*p) mod (16*p);
  if x mod 6=1 then N[i]:=[i-1,i+1,i- 8*p];
  elif x mod 6=2 then N[i]:=[i-1,i+1,(1/3)*(x-2)+2+i-
x+6*p];
  elif x mod 6=3 then N[i]:=[i-1,i+1,(1/3)*(x-3)+1+i-x-
2*p];
  elif x mod 6=4 then N[i]:=[i-1,i+1,i+8*p];
  elif x mod 6=5 then N[i]:=[i-1,i+1,(1/3)*(x-5)+2+i-x-
2*p];
  elif x mod 6=0 then N[i]:=[i-1,i+1,(1/3)*x+1+i-
x+6*p];fi;
  if x=1 then N[i]:=[i-8*p,i+1,i+6*p-1];fi; if x=6*p then
N[i]:=[i-1,i-6*p+1,i+1];fi;
od;
K5:=Filtered(K,i->i mod (16*p) in Union([14*p+1..16*p-
1],[0]));
for i in K5 do x:=(i-14*p) mod (16*p);
  if x mod 2=1 then N[i]:=[i+1,3*(x-1)+i-x-6*p,3*(x-1)+i-
x+2*p];
  else N[i]:=[i-1,3*(x-2)+2+i-x-6*p,3*(x-2)+2+i-
x+2*p];fi;
  if x=1 then N[i]:=[i+1,i-1,i-1+8*p];fi;
  if x=2*p then N[i]:=[i-1,3*(x-2)+2+i-x-6*p,3*(x-
2)+2+i-x+2*p];fi;
od;
K6:=[16*p*q+1..n];
for i in K6 do x:=i mod (16*p);
  if x mod 3=1 then y:=(2/3)*(x-1)+2+i-x- 2*p;
  elif x mod 3=2 then y:=i+x- 8*p;
  elif x mod 3=0 then y:=(2/3)*(x- 3)+3+i-x-2*p];fi;
  if x=3*p then y:=i- 5*p+1;fi;
  N[i]:=[y]; N[y][3]:=i;
od;

```

```

v:=[]; D:=[];
for i in [1..n] do
  D[i]:=[]; u:=[i]; D[i][1]:=N[i]; v[i]:=Size(N[i]);
  u:=Union(u,D[i][1]);
  r:=1; t:=1;
  while r<>0 do
    D[i][t+1]:=[];
    for j in D[i][t] do
      for m in Difference (N[j],u) do
        AddSet(D[i][t+1],m);
      od; od;
    u:=Union(u,D[i][t+1]);
    if D[i][t+1]=[] then r:=0;fi;
    t:=t+1;
  od;od;
  m:=(1/2)*Sum(v)-n+1;
  d:=[];
  for i in [1..n] do
    d[i]:=0;
    for t in [1..Size(D[i])] do
      d[i]:=d[i]+t*Size(D[i][t]);
    od;od;
    b:=0;
    for i in [1..n] do
      for j in N[i] do
        b:=b+ER(1/((d[i]*d[j])));
      od;od;
    b:=m*b/2;#(this value is equal to Balaban index of the
graph)

```

Table 2. The Balaban index of $VC_5C_7[p,q]$ nanotubes.

p	q	Balaban index
2	2	0.91465
3	2	0.74748
4	3	0.52975
5	4	0.40985
3	4	0.51844
5	5	0.36691

4. Conclusions

An algorithm has been presented for computing the Balaban index of any connected simple graph. According to this algorithm and using the GAP program, we can write a program to compute this index quickly. This method has been used here for the first time. We tested the algorithm to calculate the Balaban index of $HAC_5C_7[p,q]$ and $VC_5C_7[p,q]$ nanotubes.

References

- [1] H. Wiener, J. Amer. Chem. Soc. **69**, 17 (1947).
- [2] A. T. Balaban, Chem. Phys. Letters. **89**, 399 (1982).
- [3] N. Trinajsti, Chemical Graph Theory, CRC Press, Boca Raton, FL. 246 (1992).
- [4] M. V. Diudea, M. S. Florescu, P. V. Khadikar,

- Molecular Topology and Its Applications, EfiCon Press, Bucarest, 2006.
- [5] A. Iranmanesh, Y. Alizadeh. *Int. J. Mol. Sci.* **9**, 131 (2008).
- [6] A. Iranmanesh, Y. Alizadeh. *MATCH Commun. Math. Compute. Chem.* **60**, 883 (2008).
- [7] A. Iranmanesh, Y. Alizadeh. *MATCH Commun. Math. Compute. Chem.* **62**, 285 (2009).
- [8] A. Iranmanesh, B. Soleimani, A. Ahmadi. *J. Comput. Theor. Nanosci.* **4**, 147 (2004).
- [9] A. Iranmanesh, B. Soleimani. *MATCH Commun. Math Comput. Chem.* **57**, 251(2007).
- [10] A. Iranmanesh, O. Khormali. *J. Comput. Theor. Nanosci.* **5**, 131 (2008).
- [11] A. Iranmanesh, Y. Pakraves. *Ars Combinatoria* **7**, 3606 (2007).
- [12] A. Iranmanesh, A. R. Ashrafi, *Journal of Computational and Theoretical Nanoscience* **4**, 514 (2007).
- [13] A. Iranmanesh, Y. Pakraves, *Ars Combinatorics* **84**, 247 (2007).
- [14] A. Iranmanesh, Y. pakraves, *Utilitas Mathematica* **75**, 89 (2008).
- [15] Anehgaldi Mahmiani, Omid Khormali, A. Iranmanesh, *Optoelectron. Adv. Mater.-Rapid Comm.* **4**(2), 252 (2010).
- [16] Anehgaldi Mahmiani, Omid Khormali, A. Iranmanesh, Ali Ahmadi, *Optoelectron. Adv. Mater.- Rapid Comm.* **4**(2), 256 (2010).
- [17] A. Iranmanesh, Y. Pakraves, *Optoelectron. Adv. Mater.- Rapid Comm.* **4**(2), 264 (2010).
- [18] A. Iranmanesh, O. Khormali, *J. Comput. Theor. Nanosci.* **5**(1), 131(2008).
- [19] M. Schonert, et al., *GAP, Groups, Algorithms and Programming; Lehrstuhl D fuer Mathematik, RWTH: Achen, 1992.*

*Corresponding author: Iranmanesh@modares.ac.ir