

# An algorithm for computing symmetry of fullerenes

A. R. GILANI, M. FAGHANI<sup>a\*</sup>

Department of Mathematics, Islamic Azad University, Tehran South Branch, Tehran, Iran

<sup>a</sup>Department of Mathematics, Payam-e-Noor University, Tehran, Iran

Suppose  $M$  is a molecule,  $G$  is its molecular graph and  $A = [a_{ij}]$  is the adjacency matrix of  $G$ . An Euclidean graph associated to  $M$  is defined by a weighted graph with the adjacency matrix  $D = [d_{ij}]$ , where for  $i \neq j$   $d_{ij}$  is the Euclidean distance between the nuclei  $i$  and  $j$ . In this matrix  $d_{ii}$  can be taken as zero if all the nuclei are equivalent. Otherwise, one may introduce different weights for different nuclei. In this work a simple method is described, by means of which it is possible to calculate the automorphism group of weighted graphs. We apply this method to compute the symmetry of some fullerenes.

(Received February 28, 2010; accepted April 11, 2010)

**Keywords:** Topological symmetry, Fullerene

## 1. Introduction

Fullerenes are carbon-cage molecules in which a large number of carbon atoms are bonded in a nearly spherically symmetric configuration. Let  $F$  be a fullerene molecule with exactly  $p$  pentagons,  $h$  hexagons,  $n$  carbon atoms and  $m$  bonds. Since each atom lies in exactly 3 faces and each edge lies in 2 faces, the number of atoms is  $n = (5p+6h)/3$ , the number of edges is  $m = (5p+6h)/2 = 3/2n$  and the number of faces is  $f = p + h$ . By the Euler's formula  $n - m + f = 2$ , one can deduce that  $(5p+6h)/3 - (5p+6h)/2 + p + h = 2$ , and therefore  $p = 12$ ,  $v = 2h + 20$  and  $e = 3h + 30$ . This implies that such molecules made up entirely of  $n$  carbon atoms and having 12 pentagonal and  $(n/2 - 10)$  hexagonal faces, where  $n \neq 22$  is a natural number equal or greater than 20.<sup>1,2</sup>

In mathematics, groups are often used to describe symmetries of objects. This is formalized by the notion of a group action: every element of the group "acts" like a bijective map (or "symmetry") on some set. To clarify this notion, we assume that  $G$  is a group and  $X$  is a set.  $G$  is said to act on  $X$  when there is a map  $\phi : G \times X \rightarrow X$  such that all elements  $x \in X$ , (i)  $\phi(e, x) = x$  where  $e$  is the identity element of  $G$ , and, (ii)  $\phi(g, \phi(h, x)) = \phi(gh, x)$  for all  $g, h \in G$ . In this case,  $G$  is called a transformation group,  $X$  is called a  $G$ -set, and  $\phi$  is called the group action. For simplicity we define  $gx = \phi(g, x)$ .

Randic<sup>3,4</sup> showed that a graph can be depicted in different ways such that its point group symmetry or three dimensional perception may differ, but the underlying connectivity symmetry is still the same as characterized by the automorphism group of the graph. However, the molecular symmetry depends on the coordinates of the various nuclei which relate directly to its three dimensional geometry. Although the symmetry as perceived in graph theory by the automorphism group of the graph and the molecular group are quite different, it showed by Balasubramanian<sup>5-10</sup> that the two symmetries are connected.

## 2. Main results

Detecting topological symmetry of molecules is a well-studied problem with applications in a large number of areas. The Euclidean matrix of a molecular graph  $G$  is a

matrix  $D(G) = [d_{ij}]$ , where for  $i \neq j$ ,  $d_{ij}$  is the Euclidean distance between the nuclei  $i$  and  $j$ . In this matrix  $d_{ii}$  can be taken as zero if all the nuclei are equivalent. Otherwise, one may introduce different weights for different nuclei. In this paper a new algorithm for computing topological symmetry of molecules is presented. We apply our algorithm on a fullerene  $C_{60}$ , Fig. 1.

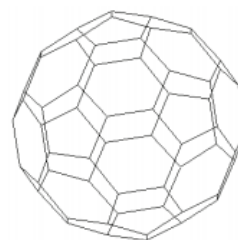


Fig. 1. The  $C_{60}$  fullerene molecule.

Our computations of the symmetry properties of molecules were carried out with the use of GAP.<sup>11</sup> GAP stands for Groups, Algorithms and Programming. The name was chosen to reflect the aim of the system, which is a group theoretical software for solving computational problems in computational group theory. This software was constructed by GAP's team in Aachen. GAP is a free and extendable software package. The term extendable means that you can write your own programs in the GAP language, and use them in just the same way as the programs which form part of the system (the "library"). More information on the motivation and development of GAP to date can be found on GAP web page on <http://www.gap-system.org>. GAP contains a large library of functions, which are important for the calculations of this paper.

GAP contains several functions for working with finite groups. For the sake of completeness, we describe some of these functions which are useful throughout. Let  $a_1, a_2, \dots, a_r$  are permutations of  $\{1, 2, \dots, n\}$ . The command "Group( $a_1, a_2, \dots, a_r$ )" computes the group generated by permutations  $a_1, a_2, \dots, a_r$ . For two groups  $A$  and  $B$ , the commands "Size( $A$ )", "GeneratorsOfGroup( $A$ )" and "Intersection( $A, B$ )" compute the cardinality of the set  $A$ , a generator set for  $A$  and intersection of  $A$  and  $B$ , respectively. Finally the command "IsSimple( $A$ )"

determines whether or not  $A$  has a non-trivial proper normal subgroup. In this paper, we use freely these functions and the reader is encouraged to consult the manual of GAP, as well as papers by Ashrafi and his co-workers.<sup>12-16</sup>

Consider the equation  $(P_\sigma)^t A P_\sigma = A$ , where  $A$  is the adjacency matrix of the weighted graph  $G$ . Suppose

$\text{Aut}(G) = \{\sigma_1, \sigma_2, \dots, \sigma_m\}$ . The matrix  $S_G = [s_{ij}]$ , where  $s_{ij} = \sigma_i(j)$  is called a solution matrix for  $G$ . Clearly, for computing the automorphism group of  $G$ , it is enough to calculate a solution matrix for  $G$ . Using [Lemma 1, 15] and its corollary, we present a MATLAB program for computing a solution matrix for the automorphism group of Euclidean graphs.

### A MATLAB program for computing the symmetries of molecules

```

n=length(a);
for i=1:n-1
    for j=i+1:n
        b(i,j)=norm(a(i,:)-a(j,:));
    end
end
b(n,n)=0;
b=b+b';

function y=halat(s,a)
t=length(a);
m=length(s);
t(s)=[];
j=0;
for i=t
    if min(min(a(1:m+1,1:m+1))==a([s,i],[s,i]))==1
        j=j+1;
        y(j)=i;
    end
end

function s=hazf(s)
m=size(s);
for i=m(1):-1:1
    if min(s(i,:))==0
        s(i,:)=[];
    end
end

function s=jaigasht(a)
m=length(a);
for i=1:m
    s(i,1)=i;
end
for j=2:m
    n=size(s);
    k=0;
    for i=1:n(1)
        y=[halat(s(i,:),a)];
        for r=1:length(y)
            b(r+k,1:n(2)+1)=[s(i,:),y(r)];
        end
        k=k+length(y);
    end
    s=b;
    s=hazf(s);
end
b=0;
n=size(s);
for i=1:n(1)
    for j=1:n(2)
        b(i,s(i,j))=j;
    end
end
s=b;

```

Our program needs the Cartesian coordinates of the atoms to determine the Euclidean distances in the molecule under consideration. If we calculate these distances by HyperChem<sup>17</sup> then for computing the symmetry of molecule under consideration, it is enough to

delete the first eight lines of the program and load the distance matrix of the molecule under consideration. In Table 1, the Cartesian coordinates the the fullerene molecule  $C_{60}$  is given.

Table 1. Cartesian coordinates of  $C_{60}$  molecule.

| No | x        | y        | z        |
|----|----------|----------|----------|
| 1  | 2.22673  | .59492   | 2.68420  |
| 2  | 3.13408  | .16181   | 1.63544  |
| 3  | 1.34213  | -.31185  | 3.25682  |
| 4  | 3.11765  | -1.15956 | 1.20440  |
| 5  | 3.21693  | -1.46205 | -.21326  |
| 6  | 3.25025  | 1.24013  | .66829   |
| 7  | 3.34529  | .95071   | -.68805  |
| 8  | 3.32838  | -.43039  | -1.13854 |
| 9  | -.45219  | 1.37427  | 3.23023  |
| 10 | .47229   | 2.32191  | 2.63187  |
| 11 | -.02700  | .08654   | 3.53589  |
| 12 | 1.78217  | 1.94095  | 2.36496  |
| 13 | 2.41456  | 2.33964  | 1.11912  |
| 14 | -.26422  | 3.11848  | 1.66508  |
| 15 | .34088   | 3.49999  | .47317   |
| 16 | 1.70997  | 3.10198  | .19425   |
| 17 | -2.14150 | -.84650  | 2.68609  |
| 18 | -2.58587 | .49930   | 2.36675  |
| 19 | -.89047  | -1.04841 | 3.25775  |
| 20 | -1.75993 | 1.58519  | 2.63288  |
| 21 | -1.64375 | 2.66299  | 1.66555  |
| 22 | -3.33263 | .44215   | 1.12147  |
| 23 | -3.22155 | 1.47344  | .19603   |
| 24 | -2.35845 | 2.60849  | .47424   |
| 25 | -.50667  | -2.99822 | 1.80380  |
| 26 | -1.81414 | -2.78732 | 1.20648  |
| 27 | -.05497  | -2.14818 | 2.80677  |
| 28 | -2.61360 | -1.73524 | 1.63792  |
| 29 | -3.34959 | -.93889  | .67091   |
| 30 | -1.71453 | -3.09035 | -.21119  |
| 31 | -2.41884 | -2.32850 | -1.13637 |
| 32 | -3.25455 | -1.22898 | -.68546  |
| 33 | 2.19300  | -2.10729 | 1.80256  |
| 34 | 1.72091  | -2.99576 | .75444   |
| 35 | 1.32481  | -1.69291 | 2.80612  |
| 36 | .40093   | -3.43150 | .75507   |
| 37 | -.34546  | -3.48836 | -.49018  |
| 38 | 2.35389  | -2.59709 | -.49129  |
| 39 | 1.63971  | -2.65148 | -1.68271 |
| 40 | .26019   | -3.10713 | -1.68211 |
| 41 | -2.23068 | -.58428  | -2.70132 |
| 42 | -1.78597 | -1.93008 | -2.38210 |
| 43 | -.47591  | -2.31071 | -2.64897 |
| 44 | -1.34628 | .32285   | -3.27364 |
| 45 | -2.19799 | 2.11835  | -1.81962 |
| 46 | -3.12230 | 1.17037  | -1.22153 |
| 47 | -3.13838 | -.15109  | -1.65266 |
| 48 | -1.32963 | 1.70406  | -2.82302 |
| 49 | .50149   | 3.01012  | -1.82049 |
| 50 | -.40603  | 3.44313  | -.77188  |
| 51 | -1.72599 | 3.00716  | -.77147  |
| 52 | .04997   | 2.15987  | -2.82339 |
| 53 | 2.13705  | .85879   | -2.70269 |
| 54 | 2.60899  | 1.74742  | -1.65466 |
| 55 | 1.80927  | 2.79941  | -1.22329 |
| 56 | .88593   | 1.06039  | -3.27422 |
| 57 | .44851   | -1.36265 | -3.24697 |
| 58 | 1.75612  | -1.57334 | -2.64974 |
| 59 | 2.58182  | -.48717  | -2.38366 |
| 60 | .02300   | -.07494  | -3.55242 |

Using these coordinates and MATLAB programs given above, one can see that the symmetry group of the fullerene is isomorphic to the group  $Z_2 \times A_5$ , where  $Z_2$  is a cyclic group of size 2 and  $A_5$  is the alternating group on five symbols.

## References

- [1] H. W. Kroto, J. R. Heath, S. C. O'Brien, R. F. Curl, R. E. Smalley, *Nature* **318**, 162 (1985).
- [2] P. W. Fowler, D. E. Manolopoulos, *An Atlas of Fullerenes*, Oxford Univ. Press, Oxford, 1995.
- [3] M. Randic, *Chem. Phys. Letters* **42**, 283 (1976).
- [4] M. Randic, *J. Chem. Phys.* **60**, 3920 (1974).
- [5] K. Balasubramanian, *Chem. Phys. Letters* **232**, 415 (1995).
- [6] K. Balasubramanian, *J. Chem. Phys.* **72**, 665 (1980).
- [7] K. Balasubramanian, *Intern. J. Quantum Chem.* **21**, 411 (1982).
- [8] K. Balasubramanian, *Chem. Rev.* **85**, 599 (1985).
- [9] K. Balasubramanian, *Studies Phys. Theor. Chem.* **23**, 149 (1983).
- [10] K. Balasubramanian, *J. Chem. Phys.* **75**, 4572 (1981).
- [11] M. Schönert, H. U. Besche, Th. Breuer, F. Celler, B. Eick, V. Felsch, A. Hulpke, J. Mnich, W. Nickel, G. Pfeiffer, U. Polis, H. Theißen, A. Niemeyer, *GAP, Groups, Algorithms and Programming, Lehrstuhl De für Mathematik, RWTH, Aachen*, 1995.
- [12] A. R. Ashrafi, M. Hamadani, *Croat. Chem. Acta* **76**, 299 (2003).
- [13] M. Hamadani, A. R. Ashrafi, *Croat. Chem. Acta* **76**, 305 (2003).
- [14] A. R. Ashrafi, *Chem. Phys. Letters* **406**, 75 (2005).
- [15] A. R. Ashrafi, M. R. Ahmadi, *Cent. Eur. J. Chem.* **3**, 647 (2005).
- [16] A. R. Ashrafi, *MATCH Commun. Math. Comput. Chem.* **53**, 161 (2005).
- [17] A. R. Ashrafi, M. Ghorbani, *MATCH Commun. Math. Comput. Chem.* **59**, 595 (2008).
- [18] A. R. Ashrafi, M. Ghorbani, M. Jalali, *J. Theor. Comput. Chem.* **7**, 221 (2008).
- [19] H. Yousefi-Azari, B. Manoochehrian, A. R. Ashrafi, *Appl. Math. Lett.* **21**, 624 (2008).
- [20] M. H. Khalifeh, H. Yousefi-Azari, A. R. Ashrafi, *Discrete Appl. Math.* **156**, 1780 (2008).

---

\*Corresponding author: m\_faghani@pnu.ac.ir.