Constructing fullerene graphs from their eigenvalues and angles

Dragoš Cvetković a,*, Patrick Fowler b, Peter Rowlinson c, Dragan Stevanović d

aFaculty of Electrical Engineering, University of Belgrade, P.O. Box 35-54, 11120 Belgrade, Yugoslavia
bSchool of Chemistry, University of Exeter, Stocker Road, Exeter EX4 4QD, UK
cDepartment of Computing Science and Mathematics, University of Stirling, Stirling FK9 4LA, Scotland, UK
dDepartment of Mathematics, Faculty of Sciences, Cirila i Metodija 2, 18000 Niš, Yugoslavia

Received 4 November 2001; accepted 17 December 2001
Submitted by W. Haemers

Abstract

We discuss means of constructing fullerene graphs from their eigenvalues and angles. An algorithm for such a construction is given.
© 2002 Elsevier Science Inc. All rights reserved.
Keywords: Fullerene graphs; Graph eigenvalues; Graph angles

0. Introduction

Let G be a graph on n vertices with adjacency matrix A. Let {e1, e2, ..., en} constitute the standard orthonormal basis for Rn. Then A has spectral decomposition A = µ1P1 + µ2P2 + ... + µmPm, where µ1 > µ2 > ... > µm and Pi represents the orthogonal projection of Rn onto ῦ(µi) (moreover, P2 = Pi = Pi T, i = 1, ..., m; and PiPj = O, i ≠ j). The non-negative quantities αij = cos βij, where βij is the angle between ῦ(µi) and ej, are called angles of G. Since Pi represents the orthogonal projection of Rn onto ῦ(µi) we have αij = ∥Pi ej∥. The sequence

*Corresponding author.
E-mail addresses: ecvetkod@etf.bg.ac.yu (D. Cvetković), p.w.fowler@exeter.ac.uk (P. Fowler), p.rowlinson@cs.stir.ac.uk (P. Rowlinson), dragance@pmf.pmf.ni.ac.yu (D. Stevanović).
$\alpha_{ij}$ $(j = 1, 2, \ldots, n)$ is the $i$th eigenvalue angle sequence; $\alpha_{ij}$ $(i = 1, 2, \ldots, m)$ is the $j$th vertex angle sequence. For further properties of angles, see Section 6 and the monograph [7].

Eigenvalues and angles do not constitute a complete set of graph invariants, but they carry a great deal of information about the graph. Cvetković in [2] presented an algorithm for constructing all trees with given eigenvalues and angles, and in [3] constructed a graph which is the supergraph of all graphs with given eigenvalues and angles. Stevanović [20] presented a branch-and-bound algorithm based on this supergraph for the construction of all graphs with given eigenvalues and angles.

Fullerenes have attracted much attention in the chemical and mathematical literature. Here we mention some results on eigenvalues of fullerene graphs from chemical literature. Definitions of the chemical terms used in this paper may be found in [13,21].

Approximations to the energy levels of the $\pi$-system represented by the molecular graph of the fullerene are found by diagonalisation of the adjacency matrix (see, for example, [21]). With the eigenvalues $\lambda_1, \ldots, \lambda_n$ in non-increasing order, the possible configurations for neutral C$_n$( $\pi$-electrons) are [12]: properly closed ($\lambda_n/2 \neq \lambda_n/2+1$, $\lambda_n/2 > 0$, $\lambda_n/2+1 \leq 0$); pseudo-closed ($\lambda_n/2 \neq \lambda_n/2+1$, $\lambda_n/2+1 > 0$); meta-closed ($\lambda_n/2 \neq \lambda_n/2+1$, $\lambda_n/2 \leq 0$); open ($\lambda_n/2 = \lambda_n/2+1$). Overwhelmingly most fullerenes have the pseudo-closed $\pi$-configuration. Examples of fullerenes with $\lambda_n/2 < 0$ are known at high $n$ (see [11]). The rare properly closed shells are ideal for the $\pi$-electrons: all electrons are in bonding levels, and no bonding capacity is ‘wasted’ in unfilled, potentially bonding levels. Three series with properly closed $\pi$-shells are as follows [13].

(i) **Leapfrogs.** All leapfrog fullerenes have closed $\pi$-shells [16,18]. A leapfrog is obtained by omnicapping and then dualising a parent fullerene, giving an equisymmetric new fullerene with disjoint pentagons and three times the vertex count [16]: leapfrog fullerenes occur at $n = 60 + 6k$ ($k \neq 1$) with the number of leapfrogs at $n$ equal to the total number of fullerene isomers at $n/3$. Leapfrog fullerenes have non-zero HOMO–LUMO gaps, $\lambda_{n/2} - \lambda_{n/2+1} = 0$. As the size of the fullerene increases, the typical gap is expected on physical grounds to decrease, as graphite itself has a zero gap. The result for leapfrog fullerenes is part of a classification for leapfrogs of cubic maps [8,14,18]: leapfrogs of cubic polyhedra with all faces of sizes divisible by three have open shells with $\lambda_{n/2} = \lambda_{n/2+1} = 0$ [14,18]; analogues of leapfrog fullerenes on the torus, Klein-bottle and elliptic plane have four, two and zero eigenvalues $\lambda = 0$, respectively [8].

(ii) **Carbon cylinders.** Carbon cylinder fullerenes, formed by tubular extension along the fivefold axis of the truncated icosahedron, or the sixfold axis of the unique C$_{72}$ isolated-pentagon fullerene. They have closed shells with positive $\lambda_{n/2}$ and $\lambda_{n/2+1} = 0$ [10] at every third member of the series, i.e. at vertex numbers $n = 70 + 30k$ ($k = 0, 1, 2, \ldots$) and $n = 84 + 36k$ ($k = 0, 1, 2, \ldots$).

(iii) **‘Sporadic’ Closed Shells.** Extra properly closed isomers occur for large values of $n$ [13]. The known isomer counts for $n \leq 140$ are 112(1), 116(1), 120(1),
122(1), 124(3), 128(3), 130(3), 132(4), 134(7), 136(9), 138(4), 140(12). All members of this set have small eigenvalues \( \lambda_{n/2+1} \sim -10^{-3} \) and in a chemical sense appear to be only ‘accidentally’ closed-shell [13].

A numerical survey of the distribution of Ramanujan graphs amongst small fullerenes, i.e. graphs that have one eigenvalue +3 and all others in the range \( 2\sqrt{2} \geq \lambda \geq -2\sqrt{2} \), has also been published [15].

In this paper we discuss means for constructing fullerene graphs from their eigenvalues and angles using two approaches. The first approach is a continuation of the above mentioned research. An algorithm for such a construction using this approach is given. The possibility of designing an improved algorithm using results of this paper is indicated. The second approach stems from the experience of chemists; it uses the so-called topological coordinates of fullerene (cf., e.g., [13, pp. 101–105]). Further research is necessary to clarify details of this approach.

The plan of the paper is as follows. In Section 1 we explain what information about the fullerene can be obtained from its eigenvalues. Section 2 shows how vertices can be classified according to their distances from pentagons. Some auxiliary results concerning the infinite hexagonal net are derived in Section 3. The study of a single pentagon surrounded by hexagons in Section 4 enables us to reconstruct further details of the fullerene. Section 5 shows how it is possible for vertices lying in pentagons to be allocated to pentagons. In Section 6 we summarize the information we have obtained and give an algorithm for the construction of fullerene graphs. The possibility of an improvement of this algorithm which is based on the results of Sections 3–5 is indicated as well. Finally, Section 7 outlines the idea of constructing a fullerene graph on the basis of topological coordinates.

1. Information derived from eigenvalues

From the point of view of spectral graph theory, the most important question regarding fullerene graphs is whether they are characterized by their spectra. An exhaustive search of small fullerene graphs shows that no two with \( n \leq 100 \) have the same spectrum. If this is universally true then it is possible to determine any graph invariant of a fullerene from its eigenvalues only.

Based on classic results from spectral graph theory, given only the eigenvalues of a fullerene, we can

- confirm that the graph with such eigenvalues is connected and regular of degree 3 [5, p. 94],
- determine the girth \( g = 5 \) for fullerenes and the number of circuits of length \( g \) (12 for fullerenes) [5, p. 95, Theorem 3.26],
- determine the number of circuits of lengths 6, 7, 8 and 9 [5, p. 97, Theorem 3.27]. (For a fullerene graph the number of 7-circuits is 0.)

From this last item, we conclude that we can recognize whether the fullerene has disjoint pentagons. We shall assume this condition in the sequel.
If we know the angles of a fullerene in addition to its eigenvalues, we can ask the weaker question of whether fullerenes are characterized by their eigenvalues and angles. As a first step, we address the following question in this paper.

**Question.** Given the eigenvalues and angles of a fullerene $F$, how much of its structure can we reconstruct.

A different question is whether it is possible to tell from the eigenvalues (and angles) of a graph $G$ only that the graph is indeed a fullerene? Here we offer an answer to this question in an algorithmic way. In our considerations we do not use any information on $G$ beyond that contained in the spectrum and angles of $G$. If the given spectrum and angles do not belong to a fullerene, our algorithm will detect this and stop. For example, if from the eigenvalues we determine that the graph is not regular, further considerations will be abandoned. However, if the input data do belong to a fullerene, all fullerene graphs having the given eigenvalues and angles will be constructed.

2. Angles, closed walks and pentagons

From eigenvalues and angles we can obtain the generating function

$$H_j(t) = \sum_{k=0}^{\infty} N_k(j)t^k,$$

where $N_k(j)$ is the number of closed walks of length $k$ starting and terminating at vertex $j$ [7, pp. 82, 83].

For any vertex $j$ of a fullerene $F$ we have

$$N_0(j) = 1, \quad N_1(j) = 0, \quad N_2(j) = 3, \quad N_4(j) = 15.$$  

If $j$ is a vertex of a pentagon, then $N_5(j) = 2$ and, otherwise, $N_5(j) = 0$. If we allow pentagons to have common vertices, then $N_5(j)$ is twice the number of pentagons to which the vertex $j$ belongs.

If $j$ does not belong to a pentagon, then we can find its distance from the nearest pentagon. Suppose that the nearest pentagon is at distance $s$ from $j$. The $(s + 2)$-neighborhood of $j$ does not contain odd circuits and it is bipartite, so that $N_{2s+5}(j) = 0$ for $k < s$. The distance of the vertex $j$ to the nearest pentagon is called the pentadistance of a vertex. The pentadistance of a vertex $j$ is obtained from the eigenvalues and angles as the smallest $s$ for which $N_{2s+5}(j) > 0$. Let $P_s$ be the set of vertices at the distance $s$ from the nearest pentagon. Let $t$ be the largest pentadistance of a vertex. The vertex set is then partitioned into subsets $P_0, P_1, \ldots, P_t$. From the eigenvalues and angles we can obtain this partition and the numbers $|P_0|, |P_1|, \ldots, |P_t|$ which can tell us a good deal about the structure of the fullerene. Obviously, when
the pentagons are disjoint, we have $|P_0| = 60$. In Section 4 we will show how this sequence can be used to establish the minimal distance between pentagons.

The sequences $N_i(j)$ ($j = 1, 2, \ldots, n$) will be used in the sequel as a basic tool for detecting details of the structure of a fullerene graph.

3. Enumeration of shortest paths and closed walks in hexagonal nets

The $(s + 2)$-neighbourhood of a vertex $j$ at distance $s$ from the nearest pentagon is isomorphic to a subgraph of the infinite, 3-regular, hexagonal net. In the current section we will study this net in order to obtain results on the number of the shortest paths between a vertex $j$ and the vertices in its $(s + 2)$-neighbourhood. The number of closed walks starting and terminating at a vertex in the hexagonal net will also be useful.

In a geometric representation of the hexagonal net, each edge in the net has the same unit length and one of three directions. Denote the unit vectors having these directions with $u$, $v$ and $w$, as shown on Fig. 1. Note that there are two kinds of vertices: vectors $u$, $v$ and $w$ leave the vertices of the first kind and enter the vertices of the second kind. Note also that edges in the hexagonal net always connect vertices of different kinds.

Consider an arbitrary shortest path $P$: $v_0, v_1, \ldots, v_m$. If there exist numbers $k$ and $l$ ($0 \leq k < l < m$) such that $\overrightarrow{v_kv_{k+1}} = u$ and $\overrightarrow{v_lv_{l+1}} = -u$, then we may obtain a shorter path $P'$ between $v_0$ and $v_m$ of length $m - 2$ if, instead of the path $v_k, v_{k+1}, \ldots, v_l, v_{l+1}$, we take the path between $v_k$ and $v_{l+1}$ which is symmetric to the path between $v_{k+1}$ and $v_l$ w.r.t. the intersection point of the line between vertices $v_k$ and $v_l$ and the line between vertices $v_{k+1}$ and $v_{l+1}$. If we say that a path contains a vector if there is an edge of the path equal to the vector, then the above consideration shows that no shortest path between any two vertices in a hexagonal net contains both $u$ and $-u$. Similar restrictions hold for $v$ and $-v$, and $w$ and $-w$.

In addition, there can be no shortest path containing vectors $u$, $v$ and $w$, since when we traverse the edge along vector $u$, the next edge must be traversed either along $-v$ or along $-w$. For the same reason, there is no shortest path containing vectors $-u$, $-v$ and $-w$.

All shortest paths from a fixed vertex $A$ to all other vertices may be divided into classes based on the set of vectors they contain:

![Fig. 1. Two kinds of vertices in hexagonal net.](image-url)
These classes divide the hexagonal net into six hexagonal subnets starting from $A$, as shown in Fig. 2. There are two kinds of hexagonal subnets. We will call the hexagonal subnet having only one edge with an end in $A$ the \textit{hex subnet of the first kind}, while the hexagonal subnet having two edges with ends in $A$ will be called the \textit{hex subnet of the second kind}.

Suppose that $A$ is a vertex of first kind, so that $u$, $v$ and $w$ leave the vertex $A$. Consider the vertex $B$ from the hex subnet of the first kind that corresponds to the set of vectors $\{-u, -v, w\}$. Let $d$ be the distance between $A$ and $B$. Along any shortest path between $A$ and $B$, positive and negative vectors appear consecutively. Since the first edge is traversed along the vector $w$, this means that every odd edge is traversed along the vector $w$, and every even edge is traversed along one of $u$ and $v$. Let $a_x$ be the number of edges on the shortest path from $A$ to $B$ along the vector $x \in \{-u, -v, w\}$. From the above considerations, we can write the following equations:

\[
\overrightarrow{AB} = a_{-u}(-u) + a_{-v}(-v) + a_w w,
\]

\[
\left\lfloor \frac{d}{2} \right\rfloor = a_{-u} + a_{-v}, \quad \left\lceil \frac{d}{2} \right\rceil = a_w.
\]
Since \( w = -u - v \) we have
\[
\begin{pmatrix} a_{-u} + \left\lceil \frac{d}{2} \right\rceil \end{pmatrix} u + \begin{pmatrix} a_{-v} + \left\lceil \frac{d}{2} \right\rceil \end{pmatrix} v = -\vec{AB}.
\]

The vectors \( u \) and \( v \) are linearly independent, and therefore \( a_{-u} \) and \( a_{-v} \) are uniquely determined. For each order of the \( a_{-u} \) vectors \( -u \) and \( a_{-v} \) vectors \( -v \) there is exactly one shortest path between \( A \) and \( B \), and so we conclude that the number of shortest paths is equal to
\[
\left\lfloor \frac{d}{2} \right\rfloor a_{-u} = \left\lfloor \frac{d}{2} \right\rfloor a_{-v}.
\]

Returning to Fig. 2, let us coordinatisate the hex subnet: vertex \( B \) (as well as the number of shortest paths from \( A \) to \( B \)) is uniquely determined by the distance \( d \) and \( \min\{a_{-u}, a_{-v}\} \) up to the symmetry of the hex subnet. Thus \( d \) and \( \min\{a_{-u}, a_{-v}\} \) may be viewed as the coordinates of \( B \) in this hex subnet. Other hex subnets may be coordinatised in the same manner, and our coordinatisation ensures that vertices on the border between two neighboring hex subnets have the same coordinates in both hex subnets.

By similar reasoning, in the case of a hex subnet of the second kind we find that the number of shortest paths is equal to \( \left\lceil \frac{d}{2} \right\rceil a_{-u} \), where \( a \) is equal to the smaller member of the pair \( \{a_u, a_{-u}\}, \{a_v, a_{-v}\} \) or \( \{a_w, a_{-w}\} \), depending on the type of vertex \( A \) and the signs of the vectors occurring in shortest paths in particular hex subnet. This subnet may be coordinatised in the same manner as above, and we see that if vertex \( B \) is on the border line between two hex subnets then the second coordinate is equal to 0.

Thus we have proved:

**Proposition 1.** Let \( d \) and \( a \) be the coordinates of \( B \) in the hex subnet starting from \( A \). If \( B \) belongs to a hex subnet of the first kind starting from \( A \), then the number of shortest paths between \( A \) and \( B \) is equal to \( \left\lfloor \frac{d}{2} \right\rfloor a_{-u} \). If \( B \) belongs to a hex subnet of the second kind starting from \( A \), then the number of shortest paths between \( A \) and \( B \) is equal to \( \left\lceil \frac{d}{2} \right\rceil a_{-u} \).

Next, we turn to the similar problem of calculating the number of closed walks of given length in hexagonal net. Since the graph of this net is bipartite, the number of closed walks of odd length is equal to 0. Let \( N_k \) be the number of closed walks of length \( k \) starting and terminating at a given vertex. We shall prove:

**Proposition 2.** The number of closed walks of length \( 2m \) starting and terminating at a vertex in the hexagonal net is given by
\[
N_{2m} = \sum_{l=0}^{m} \binom{m}{l}^2 \binom{2l}{l}.
\]
Proof. Any closed walk $T$ of length $2m$ consists of edges traversed along one of the six vectors: $u$, $-u$, $v$, $-v$, $w$, and $-w$. Let $a_x$ be the number of edges of $T$ traversed along the vector $x$ for $x \in \{u, -u, v, -v, w, -w\}$. Since the walk $T$ is closed, we have

$$a_u u + a_{-u} (-u) + a_v v + a_{-v} (-v) + a_w w + a_{-w} (-w) = 0,$$

i.e.

$$(a_u - a_{-u}) u + (a_v - a_{-v}) v + (a_w - a_{-w}) w = 0.$$  

Since the vectors $u, v, w$ satisfy $u + v + w = 0$, and any pair of these vectors is linearly independent, it must be true that

$$a_u - a_{-u} = a_v - a_{-v} = a_w - a_{-w}.$$  

(1)

Positive and negative vectors appear consecutively along $T$. Since $T$ has length $2m$, we conclude that $m$ edges are traversed along positive vectors and $m$ edges are traversed along negative vectors. This means that

$$a_u + a_v + a_w = m,$$

(2)

$$a_{-u} + a_{-v} + a_{-w} = m.$$  

(3)

From Eqs. (1)-(3) we obtain

$$a_u - a_{-u} = a_v - a_{-v} = a_w - a_{-w} = 0,$$

i.e.

$$a_u = a_{-u}, \quad a_v = a_{-v}, \quad a_w = a_{-w}.$$  

The closed walk $T$ may be identified by the order of positive and negative vectors along the walk. Since the walk may leave a vertex in any direction, we see that any alternating order of positive and negative vectors corresponds to a closed walk. The number of closed walks of length $2m$ with fixed $a_u, a_v, a_w$ is equal to $[m!/(a_u!a_v!a_w!)]^2$, and the total number of closed walks of length $2m$ in the hexagonal net is given by

$$N_{2m} = \sum_{a_u+a_{-u}+a_w=a} \left( \frac{m!}{a_u!a_v!a_w!} \right)^2.$$  

By simple combinatorial transformations we find that

$$N_{2m} = \sum_{l=0}^{m} \left( \begin{array}{c} m \\ l \end{array} \right)^2 \left( \frac{2l}{l} \right).$$  

□

The sequence $N_{2m}$ may be found in [19]. It has appeared in a number of references [1,9,17] in connection with different problems. There seems to be no closed form for $N_{2m}$, for if we look at the first few numbers ($m = 0, 1, \ldots$)

1, 3, 15, 93, 639, 4653, 35169, 272835, 2157759, 17319837, 140668065, \ldots
then we see that they contain large prime factors (for example, $140668065 = 3^2 \cdot 5 \cdot 3125957$), which makes it unlikely that there is a “nice” formula for $N_{2m}$.

For any vertex $j$ of a fullerene graph we can compare the sequences $N_k(j)$ and $N_k$. The smallest index $k$ for which $N_k(j) \neq N_k$ indicates the presence of an object foreign to the hexagonal net in the $\lfloor k/2 \rfloor$-neighborhood of $j$. In our case, it is of course a pentagon. More details can be obtained in a similar way.

4. Penta-hex subnets and the nearest pentagons

In order to obtain the distance between the nearest pentagons in the fullerene, we need to study the 3-regular, plane _penta-hexagonal net_, consisting of a central pentagon $p$ surrounded by an infinite number of hexagons, shown in Fig. 3.

The infinite sequence of hexagons $h_1, h_2, \ldots$ such that $h_1$ has a common edge with $p$, $h_{i+1}$ has a common edge with $h_i$ for $i \in N$ and the centres of the pentagon $p$ and all hexagons $h_1, h_2, \ldots$ are collinear is called a _pentagonal band_. The subnets consisting of hexagons lying between two pentagonal bands for which the initial hexagons have a common edge are called _penta-hex subnets_. There are five pentagonal bands and five penta-hex subnets. Notice that penta-hex subnets are disjoint and taken together contain all vertices of the penta-hexagonal net. A penta-hex subnet is isomorphic to a hexagonal subnet and can be coordinatised in the same manner (cf. Fig. 3).

Let $H_k$ be the number of closed walks of length $k$ starting and terminating at a vertex of the pentagon in the penta-hexagonal net of Fig. 3. We do not know a general explicit expression for $H_k$ but actual values for a finite number of values of $k$, which

![Fig. 3. Part of a penta-hexagonal net with pentagonal bands and penta-hex subnets.](image-url)
we might need, can be computed. Accordingly we shall assume that the sequence $H_k$ is known. We have $H_0 = 1$, $H_1 = 0$, $H_2 = 3$, $H_3 = 0$, $H_4 = 15$, $H_5 = 2$, \ldots

**Lemma 1.** The number of vertices at distance $d$ from the pentagon in a penta-hexagonal net is given by $5 \left\lfloor d/2 \right\rfloor + 5$.

**Proof.** In a fixed penta-hex subnet, each vertex at distance $2m$ ($m \geq 0$) from the pentagon is adjacent to exactly one vertex at distance $2m + 1$. Also, each vertex at distance $2m + 1$ is adjacent to exactly two vertices at distance $2m + 2$. On the other hand, each of two vertices at distance $2m + 2$, that belong to pentagonal bands, is adjacent to exactly one vertex in this penta-hex subnet at distance $2m + 1$, while the remaining vertices at distance $2m + 2$ are adjacent to exactly two vertices at distance $2m + 1$. If we denote the number of vertices in the penta-hex subnet at distance $d$ from the pentagon by $n_d$, we conclude that

$$n_{2m+1} = n_{2m} \quad \text{and} \quad n_{2m+2} = n_{2m+1} + 1.$$  

Since $n_0 = 1$, we have by induction that $n_d = \lceil d/2 \rceil + 1$.  \qed

Next we turn our attention from pentahexagonal nets to fullerenes. Since there are 12 pentagons in the fullerene, the number of vertices with pentadistance $d$ is equal to $60 \left\lfloor d/2 \right\rfloor + 60$, in the case that $d$-neighbourhoods of all pentagons are disjoint.

**Proposition 3.** Let $s$ be the smallest number such that $|P_s| \neq 60 \left\lfloor s/2 \right\rfloor + 60$. The distance $D_F$ between the nearest pentagons in a fullerene $F$ is equal to either $2s - 1$ or $2s$. For each pair of pentagons at distance $D_F$ it is the case that if $D_F = 2s - 1$ then one of the pentagons is in the penta-hex subnet of the other, and if $D_F = 2s$ then one of the pentagons is in the pentagonal band of the other.

**Proof.** Consider an arbitrary vertex $v$ of an arbitrary pentagon $p$ in $F$. We show that we can determine whether there is any pentagon at distance $2s - 1$ from $v$ in the penta-hex subnet corresponding to $v$. In the case that there is no pentagon at distance $2s - 1$ in this penta-hex subnet, we can then determine whether there is any pentagon at distance $2s$ from $v$, not necessarily in this penta-hex subnet. In this way, by checking for existence of pentagons at distance $2s - 1$ and, if needed, at distance $2s$, we can determine the correct value of $D_F$.

First, we want to determine whether there is any pentagon at distance $2s - 1$ from $v$ in the corresponding penta-hex subnet (see Fig. 4). There are $s$ vertices at distance $2s - 1$ and $s + 1$ vertices at distance $2s$ from $v$ in its penta-hex subnet. Thus, if there are two pentagons at distance $2s - 1$ from $v$ in its penta-hex subnet, then the maximum distance between them is $2s - 2$: this is a contradiction and we conclude that there can be at most one pentagon at distance $2s - 1$ from $v$. To determine
whether there exists a pentagon at distance $2s - 1$ we compare the number of closed walks of length $2(2s - 1) + 5$, starting and terminating at $v$, in $F$ and in the penta-hex net. If there is no such pentagon, then

$$N_{4s+3}(v) - H_{4s+3} = 0,$$

otherwise there is one pentagon at distance $2s - 1$.

If there is no pentagon at distance $2s - 1$ from $v$, then by comparing $N_{4s+5}(v) - H_{4s+5}$ with 0, we can tell if there is at least one pentagon at distance $2s$ from $v$. In this case we cannot claim that such pentagons reside in the same penta-hex subnet of $v$, because it may happen that they occur in penta-hex subnets corresponding to the neighbours of $v$ in the pentagon $p$, and that they are at distance $2s - 1$ from the pentagon $p$. Then by taking for $v$ its relevant neighbour in the pentagon $p$ we reduce the situation to the case already considered. In the case that there is no pentagon at distance $2s - 1$ from neighbours of $v$ in $p$, we may claim that pentagons at distance $2s$ from $v$ reside in pentagonal bands attached to $v$. This case is shown in Fig. 5. □

The largest $k$ for which $|P_k| = 60\lfloor k/2 \rfloor + 60$ is called the width of the fullerene $F$.

**Examples.** The leapfrog [13] of C$_{60}$, i.e. icosahedral C$_{180}$ has 60 vertices in pentagons, 60 vertices at distance 1 and 60 at distance 2 from pentagons, so it has width 1.

Any leapfrog of any fullerene where the parent had at least one pentagon adjacency has width 0. For example, there are 40 isomers of C$_{120}$ that are leapfrogs of C$_{40}$ isomers.

The leapfrog of the icosahedral C$_{80}$, i.e. C$_{240}$ has 60 vertices in pentagons, 60 vertices at distance 1 and 120 vertices at distance 2, so it has width 2.
If $W$ is the width of $F$, then the $W$-neighbourhoods of pentagons are clearly reconstructible. In particular, we have the following proposition.

**Proposition 4.** If $W$ is the width of $F$, then coordinates of vertices in the corresponding penta-hex subnets are reconstructible from eigenvalues and angles of $F$ for all vertices in $W$-neighborhoods of pentagons.

**Proof.** In $W$-neighborhoods of pentagons each vertex has a unique closest pentagon. The coordinates of a vertex $x$ in corresponding penta-hex subnet are determined by the pentadistance of $x$ and by the number of shortest paths $N$ between $x$ and the closest vertex on the pentagon. If the pentadistance of $x$ is $d$, then we have $N_{2d+5}(x) = 2N^2$. Since $d$ and $N_{2d+5}(x)$ can be determined from eigenvalues and angles, the proof is completed. \[\square\]

5. **Individualization of vertices on pentagons**

We have partitioned the vertex set of $F$ into sets $P_0, P_1, \ldots, P_t$, but have not yet individualized the vertices. The criterion for membership of $P_t$ was the distance $t$ to the closest pentagon. Now we shall partition the set $P_0$ into individual pentagons. This will be done by using the distance to the closest pentagon to which the vertex does not belong.

As in Section 2, for any $j \in P_0$ we determine the sequence $N_k(j)$ ($k = 0, 1, 2, \ldots$). We shall classify vertices in $P_0$ according to $N_k(j)$.

It is useful to compare $H_k$ (defined in Section 4) with $N_k(j)$. The smallest index $k$ for which $N_k(j) \neq H_k$ is called the *first jump* of $j$.

In Proposition 3 we have estimated the distance between the closest pentagons. Now we can be more precise.
Proposition 5. Let \( j \in P_0 \) have the smallest first jump \( h \). Then the smallest distance between \( j \) and a pentagon to which \( j \) does not belong is equal to \( \frac{1}{2}(h - 5) \).

Proof. The deviation from \( H_k \) can occur only in the presence of a pentagon \( P \) in the penta-hexagonal net which is different from the basic one \( B \), to which \( j \) belongs. Let \( s \) be the distance between \( P \) and \( B \). Then we have \( h = 2s + 5 \), hence the assertion. □

Suppose now that \( j \) does not necessarily have the smallest first jump. If \( P \) belongs to the penta-hex subnet corresponding to \( j \), then we have \( h = 2s + 5 \). If \( P \) belongs to the penta-hex subnet corresponding to a neighbour of \( j \) in \( P \), then we have \( h = 2s + 7 \). Finally, if \( P \) belongs to the penta-hex subnet corresponding to a vertex of \( P \) not adjacent to \( j \), then we have \( h = 2s + 9 \).

Of course, since the distance between the nearest pentagons in \( F \) is equal to \( D_F \), the smallest first jump of vertices from \( P_0 \) is equal to \( 2D_F + 5 \). If there are just two pentagons at a smallest distance \( D_F \), then there will be two vertices \( j_1, j_2 \) in these two pentagons with the same first jump \( 2D_F + 5 \). If they are unique they belong to different pentagons. If one of pentagons is in the pentagonal band of the other, there will be two vertices in each pentagon with this property.

The above discussion gives some information on how the vertices may be grouped into pentagons. Let \( j_1, j_2 \in P_0 \) and let \( h_1, h_2 \) be their first jumps, respectively. Then \( j_1 \) and \( j_2 \) may be adjacent (and belong to the same pentagon) only if \( |h_1 - h_2| \leq 2 \). Also, \( j_1 \) and \( j_2 \) may belong to the same pentagon (without being adjacent) only if \( |h_1 - h_2| \leq 4 \).

In order to obtain more information on the graph structure consider the set \( J \) of vertices \( j \) from \( P_0 \) with the smallest first jump \( h \). For any \( j \in J \), let \( n_j = N_b(j) \). Consider separately the situations in which \( D_F \) is even and odd.

(1) If \( D_F = D \) is even, closest pentagons lie in the pentagonal bands of each other, as in Fig. 5. Let \( m_D \) be the number of closed walks of length \( 2D + 5 \) which start and terminate at a vertex \( j \) from \( J \) and contain edges of a fixed closest pentagon. Then we have \( n_j = m_D \) or \( n_j = 2m_D \) depending on whether \( j \) has one or two pentagons at distance \( D \). In the first case we say that \( j \) belongs to a subset \( J_1 \) of \( J \) and in the second case to \( J_2 \). So we have a partition \( J = J_1 \cup J_2 \). Now it is not difficult to construct all possible structures of pentagons at the mutual distance \( D \). In particular, one can recognize “triangles” of pentagons at distance \( D \), as in Fig. 5. In some situations, as in Fig. 5, there is a “central” vertex \( c \) of the triangle at the same distance from the three pentagons which can be clearly identified by \( N_{2D+5}(c) \).

(2) If \( D_F = D \) is odd, two closest pentagons \( Q_1, Q_2 \) each lie in a penta-hex subnet of the other. Let \( j_1 \in Q_1, j_2 \in Q_2 \) and \( j_1, j_2 \in J \). Then \( j_1 \) has the same coordinates w.r.t. \( Q_2 \) as \( j_2 \) w.r.t. \( Q_1 \). These coordinates are obtainable from basic spectral information if there is no third pentagon \( Q_3 \) at the same distance from \( Q_1 \) or/and \( Q_2 \). For different pairs of pentagons at distance \( D \) these coordinates
my be different. Although the situation is a little more complex compared with case (1), certainly one can easily reconstruct all possible structures of closest pentagons.

The above observations make it possible not only to individulise vertices on pentagons but also to reconstruct the local structure of the fullerene graph for vertices with pentadistance equal to and a little larger than the width.

6. Reconstruction of a fullerene

In this section we give a backtracking-based algorithm for constructing all fullerenes with given eigenvalues and angles. We then indicate how the basic form of the algorithm can be improved using results from Sections 3–5.

We have seen in previous sections that the \((DF+2)\)-neighbourhood of each pentagon in a fullerene \(F\) coincides with the \((DF+2)\)-neighbourhood of a pentagon in the penta-hexagonal net. However, these neighbourhoods may overlap and, at the moment, we cannot determine the relation between them. Nevertheless, the \((s-1)\)-neighbourhoods are disjoint, and we can use them to start the construction of the fullerene \(F\).

To construct the remaining part of the fullerene, we apply the edge condition and the fuzzy image of a graph from [3].

Let \(P_H(x)\) be the characteristic polynomial of a graph \(H\). From the eigenvalues of a graph \(G\) we can, of course, determine its characteristic polynomial \(P_G(x)\). From the eigenvalues and angles of \(G\) we can determine characteristic polynomials \(P_{G-u}(x)\) of all vertex-deleted subgraphs \(G-u\) of \(G\) (cf. [7, p. 83]).

**Theorem 1** [3]. Let \(G\) be a graph with \(n\) vertices and \(m\) edges, and let \(uv\) be an edge of \(G\). Then there exists a polynomial \(q(x)\) of degree at most \(n-3\) such that

\[
(x^n - (m-1)x^{n-2} + q(x))P_G(x) + P_{G-u}(x)P_{G-v}(x) \text{ is a square.}
\]

The necessary condition (4) for two vertices \(u\) and \(v\) to be adjacent is called the edge condition. The quasi-graph \(Q(G)\) of the graph \(G\) is defined as the graph with the same vertices as \(G\), with two vertices adjacent if and only if they fulfil the edge condition. Obviously, any graph is a spanning subgraph of its quasi-graph.

If \(G\) is regular and both \(G\) and \(\overline{G}\) are connected, then from the eigenvalues and angles of \(G\) we also know the eigenvalues and angles of \(\overline{G}\) [6]. The edge condition in \(\overline{G}\) is a necessary condition for non-adjacency in \(G\), and any two distinct vertices of \(G\) are adjacent either in \(Q(G)\) or in \(Q(\overline{G})\). If they are adjacent in one and not adjacent in the other, then their status coincides with that in \(Q(G)\). Thus, the fuzzy image \(FI(G)\) is defined as the graph with the same vertex set as \(G\) and two kinds of edges, solid and fuzzy. Vertices \(u\) and \(v\) of \(FI(G)\) are joined by a fuzzy edge if they are adjacent in both \(Q(G)\) and \(Q(\overline{G})\), otherwise they are joined by a solid
edge if they are adjacent in $Q(G)$ and they are non-adjacent if they are non-adjacent in $Q(G)$. The graph $G$ contains all solid edges and some of the fuzzy edges of $FI(G)$.

To decrease the number of fuzzy edges in the fuzzy image we can use the lower bounds on the distance between vertices. Let $d(j, k)$ be the distance between vertices $j$ and $k$ in $G$.

**Lemma 2** [4]. If $g = \min \{ s : \sum_{i=1}^{m} |\mu_{i}^{s}| \alpha_{ij} \alpha_{ik} \geq 1 \}$, then $d(j, k) \geq g$.

**Lemma 3** [20]. If $g = \min \{ s : \sum_{i=1}^{m} |\mu_{i}^{s+2}| \alpha_{ij} \alpha_{ik} \geq d_{j} + d_{k} + \delta_{s-1} - s \}$, where $\delta_{s-1}$ is the sum of $s-1$ smallest degrees of vertices other than $j$ and $k$, then $d(j, k) \geq g$.

The lower bounds given by these lemmas are mutually independent and rarely greater than 3. They are also independent of the edge condition, so that if for any two vertices $u$ and $v$, either Lemma 2 or Lemma 3 gives a lower bound on distance at least 2, then we can remove the fuzzy edge $uv$ from the fuzzy image.

**The Basic Fullerene Construction Algorithm**

**Input:** Eigenvalues and angles of a fullerene.

**Output:** All fullerene graphs with given eigenvalues and angles.

**Step 1a** Find the fuzzy image $F^*$ and lower bounds on distance (Lemmas 2 and 3). Remove the fuzzy edges from $F^*$ for all pairs of vertices for which the lower bound on distance is at least 2.

**Step 1b** Find the pentadistance of each vertex, and the sets $P_{s}$ for each $s = 0, 1, \ldots, t$, where $t$ is the largest pentadistance.

**Step 1c** Find the value of the width $W$. If $W = 0$, output the graph of the fullerene $C_{60}$ and go to Step 8. Otherwise set $k = w - 1$. The subgraph $F_{k}$ induced by $\bigcup_{i=0}^{k} P_{i}$ is the disjoint union of 12 pentagons together with their $k$-neighborhoods in a penta-hexagonal net.

**Step 2** Increment $k$ and let $F_{k}$ be the graph obtained from $F_{k-1}$ by adding isolated vertices of $P_{k}$. In the fullerene, there are adjacencies between vertices in $P_{k}$ and there are edges from $P_{k}$ to $P_{k-1}$. Using the modified fuzzy image $F^*$, construct all possible sets of edges $E_{k,1}, E_{k,2}, \ldots, E_{k,e_{k}}$ which join vertices of $P_{k}$ and which join vertices of $P_{k}$ to vertices of $P_{k-1}$.

**Step 3** Set $j_{k} = 1$ and add the edges of $E_{k,1}$ to $F_{k}$.

**Step 4** If the subgraph $F_{k}$ is planar with (i) all or all but one of its faces five-sided or six-sided, (ii) maximal degree at most 3, (iii) eigenvalues satisfying the Interlacing Theorem, then go to Step 2 in case $k < t$, otherwise output the graph $F_{t}$.

**Step 5** If $j_{k} < e_{k}$, then delete the edges of $E_{k,j_{k}}$ from $F_{k}$, increment $j_{k}$ and add the edges of $E_{k,j_{k}}$ to $F_{k}$. Go to Step 4.
Step 6  While $j_k = e_k$ decrement $k$. If $k < w - 1$ then go to Step 8, else go to Step 5.

Step 7  For each output graph, obtained in Step 4, check whether it has the given eigenvalues and angles. If not, delete it from the list.

Step 8  Stop

Note that it may happen that vertex angle sequences given as input data of the algorithm are actually assigned to vertices of an output graph only at the end of the algorithm (Step 7). For example, at the beginning of the execution of the algorithm we certainly know what the subgraph $F_{w-1}$ looks like although we do not know the distribution of vertices from $P_0$ among pentagons.

It is difficult to estimate the complexity of this algorithm. On the other hand, the question of complexity has second order significance. Certainly, the algorithm is much better than the constructing all fullerenes with the given number of vertices and checking for each one whether it has given eigenvalues and angles. The point of the algorithm is to show what information on the structure of a fullerene is contained in eigenvalues and angles and how this information can be used.

One might expect that automorphisms present potential pitfalls for the execution of this algorithm. Equivalent vertices have the same angles, making it impossible for the algorithm to distinguish between them. This could lead to exponentially large numbers of edge sets found in Step 2, and also to exponentially long running times. In particular, a large automorphism group can cause many fuzzy edges in the fuzzy image of the graph. For example, the automorphism group of $C_{60}$ is transitive on vertices and the fuzzy image of $C_{60}$ is a complete graph containing only fuzzy edges. (In this basic case, however, reconstruction is immediate.)

In many cases it can happen that a great number of solid edges in the fuzzy image forces the reconstruction by diminishing the number of alternatives, or that some special distribution of pentagons over the fullerene surface can lead to lower execution times of the algorithm.

The basic form of the algorithm does not use any result from Sections 3–5 concerning details of the fullerene graph under reconstruction. The algorithm can be improved by appropriate use of information from these sections. This could lead to the reduction in number of possible sets $E_{k,j_k}$ for the construction of the subgraph $F_k$. We omit a formal description of such an improved algorithm but indicate its spirit. Namely, it would contain many heuristics each applicable if certain conditions are met. The use of heuristics should be guided by the principles of building artificial intelligence programs, and we could then speak about an expert system for constructing fullerenes on the basis of their eigenvalues and angles.

Although we have presented an algorithm for constructing all fullerene graphs with given eigenvalues and angles, we have not yet answered the question of whether the construction ends with a unique graph. Further research is necessary to clarify this and related questions.
7. Topological coordinates

In this section we describe an alternative approach to the problem of constructing fullerene graphs from their eigenvalues and angles.

It is well known that a graph is determined by its eigenvalues and eigenvectors (cf., e.g., [7, p. 22]). Of course, it is sufficient to know eigenvalues and a basis of corresponding eigenvectors. Our position is less favourable since from eigenvalues and angles the eigenvectors cannot be uniquely reconstructed in general.

For simple (i.e. non-degenerate) eigenvalues the eigenvalue angle sequence is equal to the sequence of moduli of the coordinates of the corresponding eigenvector. Consequently to determine the eigenvector in this case we should only determine correct signs for the coordinates. If we have a lot of simple eigenvalues the orthogonality between eigenvectors reduces the number of possibilities substantially. In particular, the largest eigenvalue is simple, with an eigenvector whose all coordinates are equal to 1 which implies that the sum of coordinates of eigenvectors belonging to other eigenvalues is equal to 0.

Calculations show that typical fullerene graphs usually have many simple eigenvalues; perhaps one can say almost all eigenvalues are simple. This fact opens a possibility to design an algorithm for constructing fullerene graphs from their eigenvalues and angles via eigenvectors. However, there is something more to be said; namely we shall see that it is usually sufficient to construct just three mutually orthogonal eigenvectors.

In what follows we need the eigenvectors of the Laplacian matrix $L = D - A$ of a graph where $A$ is the adjacency matrix and $D$ the diagonal matrix of vertex degrees. However, fullerene graphs are regular and the matrices $A$ and $L$ have the same eigenvectors, so we can deal equally well with either.

The first relevant fact is that for a graph that represents a spherical polyhedron, the Laplacian eigenvectors are approximate solutions of the particle-on-a-sphere wave equation

$$\nabla^2 \psi_{\ell m}(\theta, \phi) = \ell(\ell + 1)\psi_{\ell m}(\theta, \phi),$$

where $\nabla^2$ is the operator $\sum_i \partial^2 / \partial x_i^2$ which can be transformed to polar coordinates $\theta, \phi$ on the unit sphere.

Exact solutions of this equation are spherical harmonics (waves on a spherical planet), characterised by an angular momentum quantum number $\ell = 0, 1, \ldots$. For each value of $\ell$ there is a $(2\ell + 1)$-fold degenerate set of harmonics with quantum number $m$ describing the $z$-component of the total angular momentum.

The significant properties of the spherical harmonics are their nodal behaviour and, at least for low $\ell$, their simple dependence on $(x, y, z)$ or $(\theta, \phi)$. The lowest solution $\ell = 0$ is a constant over the sphere i.e. nodeless. For $\ell = 1$ there are three orthogonal solutions which in real form can be chosen as exactly $x, y, z$ in a Cartesian system (i.e. $\cos \theta$ and two rotated forms).
Solutions for $\ell = 2$ have two angular nodes and degeneracy 5, i.e. five independent solutions $3z^2 - r^2, x^2 - y^2, xy, xz, yz$.

Thus the series for the spherical harmonic continuum functions has a monotonic increase of eigenvalue, node count and multiplicity.

The claim is that in the eigenvectors of $L$ it will be possible to find analogous node—eigenvalue correlations so that there will be an “S-like” vector, three “P-like” vectors, and so on. Two factors intervene in a graph with finite point group less than the spherical $R_{3h}$: (i) the multiplicities split (the maximum symmetry-enforced $\ell$ multiplicity is 5 in $I_h$, 3 in cubic graphs, ...) (ii) the ordering of the levels will change, certainly at high nodal count, but perhaps also at low count e.g. if the graph when embedded is highly non-spherical, a wave such as

may have lower energy than a wave such as

(for straightforward physical reasons based on wavelength), and so the ‘natural’ sequence

may be perturbed.

Note that for all cubic polyhedra, the 0 eigenvalue of $L$ has a uniform vector matching the S harmonic—it is constant everywhere. For sufficiently dense coverings of the sphere, we should always be able to find three vectors that correspond to the triple of P spherical harmonics, and cut the graph into two connected parts of opposite sign, possibly separated by some vertices with zero coefficient.

The suggested procedure then is to use the coefficients of these three vectors as literally $x$- or $y$- or $z$-coordinates of graph vertices in $\mathbb{R}^3$. The vertices will be embedded in a closed surface whose inside domain is a convex set. Pentagon and hexagon faces of the fullerene graph on this surface will easily be recognized.

If we take eigenvectors for $\lambda_2, \lambda_3$ and $\lambda_4$ we will usually get a faithful embedding, but can occasionally be caught out by intrusion of a more highly noded vector, giving a non-convex set of coordinates. If, however, we take the first three uninodal vectors we apparently always get a faithful embedding for a fullerene—certainly for the many thousands of cases tested.
It is conceivable that the three vectors would have more than one allowed solution of signs, but it is hard to believe that more than one such solution would give a fullerene adjacency matrix on assigning three edges to each vertex on the basis of physical proximity. Thus topological coordinates offer a promising alternative method of reconstructing fullerenes, of heuristic value, which should be investigated and sharpened further, in parallel with the implementation of the algorithm described in the earlier part of the paper.

Acknowledgement

We thank Felix Goldberg, Nevin Kapur, Helmut Prodinger, Volker Strehl and Christian Krattenthaler who replied to our message in DMANET mailing list and pointed us to the references [1,9,17].

References