MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **74** (2015) 505-520

ISSN 0340 - 6253

Enumeration of Generalized Fusenes

Gunnar Brinkmann, Pieter Goetschalckx

Applied Mathematics, Computer Science and Statistics
Ghent University Krijgslaan 281-S9
9000 Ghent, Belgium

`Gunnar.Brinkmann@UGent.be` , `Pieter.Goetschalckx@UGent.be`

(Received April 30, 2015)

Abstract

Generalized fusenes are structures that are similar to the hexagon based benzenoids and fusenes, except that also non-hexagonal faces are allowed. In this paper, a fast algorithm to enumerate generalized fusenes with given faces is given. The algorithm is fast enough to generate millions of non-isomorphic structures per second and is based on a two-step approach using the canonical construction path method and the homomorphism principle.

Introduction

Between 1968 [1] and 1998 [5] a large number of increasingly fast algorithms have been proposed for the enumeration of benzenoids and fusenes , see [3] for a survey. Finally, in [2] an algorithm was proposed that could generate tens of millions of nonisomorphic fusenes and benzenoids per second. The bottleneck in this algorithm is in fact writing the structures to a file or a pipe.

The restriction to hexagonal rings was due to the fact that hexagonal rings are energetically ideal for carbon. The discovery of fullerenes [7, 12], that need pentagonal faces for their curvature, led to an increasing interest in structures also containing pentagons. Not much later also heptagons were detected in carbon structures [11]. Nanojoins [14] joining two nanotubes with different parameters or even more than two nanotubes also need heptagonal faces. To this end also carbon structures including (a limited amount of) pentagons and heptagons in addition to hexagonal rings could not be neglected any more. See e.g. [6, 8, 15].

This motivates the development of algorithms specializing in the generation of structures with mostly hexagons and few pentagons and heptagons allowed. In this article we will solve this problem by giving a more general algorithm that can generate all these chemically relevant structures extremely efficiently, but can in addition also generate structures that are of less chemical interest, like e.g., structures with only octagons or even larger faces.

Our algorithm is based on the algorithm described in [2] and we will describe here only the parts that are essentially different from [2]. For the parts following closely the lines of the algorithm generating fusenes and benzenoids, the reader is referred to [2].

An efficient implementation of the algorithm can be obtained from the authors or – embedded in an easily usable user interface – as part of the program package CaGe [9].

The algorithm generates generalized fusenes from their inner duals. While for fusenes each vertex in the inner dual represents a hexagon, in generalized fusenes the vertices may represent faces of different size. This makes an additional step necessary that assigns face sizes to the vertices. In case of several different face sizes, this additional step can increase the efficiency of the algorithm, but the characterization of inner duals is more complicated.

The structures we will generate are defined as follows:

Definition 1 *A* generalized fusene *with face vector* $(n_3, n_4, \ldots, n_k)$ *is a plane graph* F *with the following properties:*

(a) F *is simple.*

(b) F *is 2-connected.*

(c) *All vertices* v *not in the boundary of the outer face have degree* $d(v) = 3$ *and all vertices* w *in the boundary of the outer face have degree* $d(w) \in \{2, 3\}$.

(d) *For* $3 \leq i \leq k$ *there are exactly* n_i *bounded faces of degree* i *and there are no bounded faces with other degrees.*

We will denote these structures as $(n_3, \ldots, n_k)$*-fusenes.*

The algorithm

We assume that the order of edges around a vertex of a plane graph describes the clockwise rotation around that vertex.

There is a one-to-one correspondence between the dual of a plane graph and the graph itself. Removing the vertex corresponding to the outer face, we get the *inner dual.* Non-isomorphic plane graphs can have the same inner duals. An example can be seen in Figure 1.

Traversing the boundary of the outer face of an inner dual in anticlockwise order (seen from inside the face) we get a cyclic sequence of (not necessarily pairwise disjoint) vertices. A subsequence u, v, w of this sequence is called an angle of the outer face. Such a subsequence corresponds to two edges $e_1(u, v, w)$ with endpoints u, v and $e_2(u, v, w)$ with endpoints v, w that were traversed when generating the subsequence. The inner duals of fusenes are simple graphs, but inner duals of generalized fusenes can be multigraphs (without loops), so describing edges just by their endpoints could be ambiguous. If an angle u, v, w in the outer face of an inner dual of a 2-connected plane graph G is labeled with the number of edges that were deleted between $e_1(u, v, w)$ and $e_2(u, v, w)$ in the order around v, we get *the angle labeled inner dual* of G. Note that all labels are strictly positive as a label 0 would correspond to a face of size at least 4 in the dual and therefore a vertex of degree at least 4 in the generalized fusene.

As duals of 2-connected plane graphs do not have loops, we can reconstruct the duals and finally the graphs from angle labeled inner duals of 2-connected plane graphs, or with other words: there is a one-to-one correspondence between 2-connected plane graphs and their angle labeled inner duals. To this end it is sufficient to generate the angle labeled inner duals of the graphs we want to generate.

During the construction we will also use *vertex labeled inner duals.* The vertex labeled inner dual of a generalized fusene is the inner dual with each vertex v of the inner dual labeled with the size of the face that corresponds to v. So the vertex labeled inner dual of a generalized fusene with face vector $(n_3, n_4, \ldots, n_k)$ has exactly n_i vertices labeled i for $3 \leq i \leq k$ and no other vertices.

We will generate angle labeled inner duals of generalized fusenes with face vector $(n_3, n_4, \ldots, n_k)$ in three steps: first we will generate inner duals, from these we will generate vertex labeled inner duals and from these we will generate the angle labeled inner duals. In each step it is important to generate only those structures that will later actually correspond to a $(n_3, \ldots, n_k)$-fusene. To ensure this, we will give some lemmas characterizing these labeled and unlabeled inner duals.

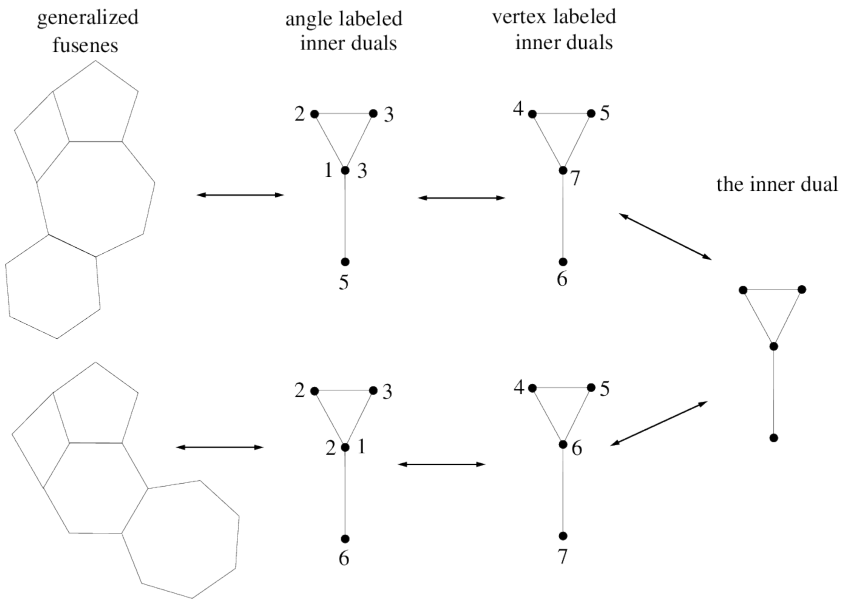

Figure 1: The figure shows two different generalized fusenes with the same inner dual represented as angle labeled inner duals. The vertex labeled inner dual is shown as an intermediate step. Only two vertex labeled inner duals are given, but note that there are more non-equivalent ways to label the vertices of the inner dual with $4, 5, 6$ and 7 and more angle labeled inner duals for each of the vertex labeled inner duals in the figure.

The total degree $d_t(v)$ of a vertex in an angle labeled multigraph is the sum of the degree $d(v)$ and all labels of angles centered at v. So for vertices w not in the boundary of the outer face we have $d_t(w) = d(w)$ and for vertices w in the boundary of the outer face we have $d_t(w) > d(w)$.

Lemma 2 *A plane multigraph G (without loops) with all angles of the outer face labeled is the angle labeled inner dual of a $(n_3, \ldots, n_k)$-fusene F, if and only if*

(i$_2$) *For $3 \leq i \leq k$ there are exactly n_i vertices with total degree i and no other vertices.*

(ii$_2$) *G is connected.*

(iii$_2$) *All bounded faces are triangles.*

(iv$_2$) *All labels are strictly positive.*

(v$_2$) *The sum of all labels is at least 3.*

Proof: The necessity of (i_2),(iii_2) and (iv_2) follows directly from the properties of a dual and the fact that F is a $(n_3, \ldots, n_k)$-fusene. Property (v_2) follows from the fact that F is simple – so the outer face is no 2-gon. It is well known that the dual of a 2-connected multigraph (without loops) is also 2-connected. After removing the vertex corresponding to the outer face, the inner dual is at least 1-connected, proving property (ii_2).

Assume now that an angle labeled multigraph G with properties (i_2) to (v_2) is given. In each angle of the outer face we add as many edges as given by the label and connect them to a new vertex o representing the outer face. This graph $\bar{F}$ has only faces of size 2 and 3 and all 2-gons are adjacent with o. We will show that the dual F of $\bar{F}$ is an $(n_3, \ldots, n_k)$-fusene.

If F had a double edge, $\bar{F}$ had two faces sharing at least two edges. As all faces in $\bar{F}$ are 2- or 3-gons, this would imply the existence of a vertex with degree 2 in $\bar{F}$, which could neither be o (property (v_2)) nor one of the other vertices that have degrees between 3 and k. This implies that F is a simple graph (property (a)).

We will now show that $\bar{F}$ is a 2-connected graph (without loops). This implies that F is also 2-connected (property (b)). G is a connected graph, but may have cutvertices. As all bounded faces are triangles and G has no loops, all cutvertices are in the boundary of the outer face. As o is connected to all vertices in the boundary of the outer face, each path between vertices of G using a cutvertex c in G can be modified to use neighbours of c in the boundary of the outer face and the new vertex o. So the removal of a single vertex cannot separate two vertices from G. It remains to be shown that no vertex v in G can be separated from o by removing just one vertex. If v is in the boundary of the outer face, it is connected to o, so v and o cannot be separated. After the removal of an arbitrary vertex, the component of an internal vertex v in G still contains at least one other boundary vertex (G has no loops), so that there is still a path from v to o via this vertex.

Property (c) follows from the fact that all bounded faces of $\bar{F}$ are 2- or 3-gons and that all 2-gons contain the vertex o.

Property (d) finally follows from (i_2). ■

Also for the next step we can characterize the structures that finally lead to $(n_3, \ldots, n_k)$-

fusenes. We define the degree bound $d_b(v)$ of a vertex v to be the degree plus the number of times it occurs in the boundary cycle. The degree bound of a vertex v gives a lower bound for the label of v.

Lemma 3 *A plane multigraph $G = (V, E)$ (without loops) together with a labeling function $l : V \to \mathbb{N}$ is the vertex labeled inner dual of a $(n_3, \ldots, n_k)$-fusene F, if and only if*

(i$_3$) *For $3 \leq i \leq k$ there are exactly n_i vertices labeled i and no other vertices.*

(ii$_3$) *G is connected.*

(iii$_3$) *All bounded faces are triangles.*

(iv$_3$) *For all vertices v we have $l(v) \geq d_b(v)$. For interior vertices we have $l(v) = d_b(v) \quad (= d(v))$.*

(v$_3$) $\sum_{v \in V} l(v) \geq (\sum_{v \in V} d(v)) + 3$.

Proof: It is easy to see that a vertex labeled plane multigraph with these properties can be transformed into an angle labeled graph G' with the properties of Lemma 2 by labeling angles with positive numbers so that the total degree equals the vertex label. Applying Lemma 2 we get that G' corresponds to a $(n_3, \ldots, n_k)$-fusene F and it is easy to see that G is the vertex labeled inner dual of F. ■

Finally we can exactly characterize the class of graphs that are the starting point of our labelings leading to $(n_3, \ldots, n_k)$-fusenes.

Lemma 4 *A plane multigraph $G = (V, E)$ (without loops) is the inner dual of a $(n_3, \ldots, n_k)$-fusene F, if and only if*

(i$_4$) *G has $\sum_{i=3}^{k} n_i$ vertices.*

(ii$_4$) *G is connected.*

(iii$_4$) *All bounded faces are triangles.*

(iv$_4$) *For $3 \leq i \leq k$ there are at most n_i vertices of degree i that are not in the boundary.*

(v$_4$) *If m_i is the number of vertices v with $d_b(v) = i$, then for $3 \leq i \leq k$ we have $\sum_{j=3}^{i} m_j \geq \sum_{j=3}^{i} n_j$*

(vi$_4$) $\sum_{j=3}^{k} (n_j \cdot j) \geq (\sum_{v \in V} d(v)) + 3$

Proof: It is easy to see that disregarding the labels of a vertex labeled inner dual of a $(n_3, \ldots, n_k)$-fusene – that is an inner dual with the properties from Lemma 3 – one has a plane multigraph with the given properties.

In order to prove that a plane multigraph G with these properties is the inner dual of a $(n_3, \ldots, n_k)$-fusene, we prove that the vertices can be labeled in a way that the resulting vertex labeled multigraph has the properties from Lemma 3. Properties (ii_3) and (iii_3) are immediate as they are independent of the labeling.

We label the vertices as follows:

1.) Label all interior vertices with their degree.

Let $\bar{n}_j$ denote the number of interior vertices labeled j.

Property (iv_4) makes sure that after this step at most n_i vertices have label i for all $3 \le i \le k$

In the following loop we will start with $i = 3$ and proceed to $i = k$.

2.) For i from 3 to k choose $n_i - \bar{n}_i$ still unlabeled vertices with $d_b(v) \le i$ and label them i.

Note that at the beginning of step j only interior vertices are labeled j.

It is clear that if we can perform this step for all $3 \le i \le k$, at the end properties (i_3), (iv_3) and (v_3) are fulfilled. At the beginning of step i, for $3 \le j < i$ exactly n_j vertices are labeled j. Due to property (v_4) there are $\sum_{j=3}^{i} m_j - \sum_{j=3}^{i-1} n_j \ge n_i$ vertices with $d_b(v) \le i$ that are still unlabeled or internal vertices labeled i. So there are at least $n_i - \bar{n}_i$ vertices that can still be labeled i. ■

The construction of the inner dual

Once we have the inner duals of $(n_3, \ldots, n_k)$-fusenes, it is straightforward how to label the vertices in every possible way leading to a vertex labeled inner dual of a $(n_3, \ldots, n_k)$-fusene and in the next step label the angles in every possible way leading to an angle labeled inner dual of a $(n_3, \ldots, n_k)$-fusene. The construction of the unlabeled inner dual must be described though.

The construction is based on the construction in [2]. Some changes are necessary, as for generalized fusenes different face sizes are allowed and double edges can occur in inner duals.

Definition 5 *A plane multigraph $G = (V, E)$ (without loops) is called an $(n_3, \ldots, n_k)$-dual with deficit d, if and only if*

(i$_5$) *G has $(\sum_{i=3}^{k} n_i) - d$ vertices.*

(ii$_5$) *G is connected.*

(iii$_5$) *All bounded faces are triangles.*

(iv$_5$) *For $3 \leq i \leq k$ there are at most n_i vertices of degree i that are not in the boundary.*

(v$_5$) *If m_i is the number of vertices v with $d_b(v) = i$, then for $3 \leq i \leq k$ we have $\sum_{j=3}^{i} m_j \geq (\sum_{j=3}^{i} n_j) - d$*

(vi$_5$) $\sum_{j=3}^{k} n_j \geq (\sum_{v \in V} d(v)) + 3$

Note that except for deficit 0 property (vi$_5$) is implicit by the other properties.

For $d = 0$ this exactly characterizes the inner duals of $(n_3, \ldots, n_k)$-fusenes and for $d = (\sum_{i=3}^{k} n_i) - 1$ a single vertex is a $(n_3, \ldots, n_k)$-dual for all $(n_3, \ldots, n_k)$ that are not all 0. Removing a boundary vertex of a $(n_3, \ldots, n_k)$-dual with deficit d that is not a cutvertex, we get a $(n_3, \ldots, n_k)$-dual with deficit $d + 1$, so all $(n_3, \ldots, n_k)$-duals can be recursively constructed by the inverse operations of such removals. The inverse operations can – just like in [2] – be described by adding a new vertex in the outside face and connecting it to all occurrences of vertices in a boundary segment of the smaller dual. For the extension of a $(n_3, \ldots, n_k)$-dual with deficit $d+1$ only operations are considered that lead to $(n_3, \ldots, n_k)$-duals with deficit d. Though not difficult, the details of how boundary segments are tested for this property are quite technical. We refer the reader to [10] for details.

Isomorphism rejection

The isomorphism rejection techniques closely follow the lines in [2] and will not be described in detail here. We apply McKay's canonical construction path method [13] and for testing isomorphism of the vertex labeled inner duals and angle labeled inner duals in addition the homomorphism principle is applied.

Vertex labeled inner duals are constructed from inner duals by assigning vertex labels, so isomorphic vertex labeled inner duals come as labelings of the same inner dual and each such isomorphism induces an automorphism of the inner dual. For inner duals with a nontrivial automorphism group we apply the canonical construction path method, but only with respect to the (always small) group of automorphisms of the inner dual. If the automorphism group of the inner dual is trivial (which is in by far most of the cases the case), all vertex labeled inner duals are non-isomorphic (and have a trivial group themselves) – so no tests are necessary. If the group is nontrivial, a code – very similar to the one in [2] – is constructed to decide on canonicity.

Angle labeled inner duals are constructed by distributing the differences between the vertex label and the degree of boundary vertices over the boundary angles. Isomorphic angle labeled inner duals come from the same vertex labeled inner dual and each such isomorphism induces an automorphism of the vertex labeled inner dual, so that again no isomorphism tests are necessary in case the automorphism group of the vertex labeled inner dual is trivial, which is – unless only one of the n_i is positive – is even more often the case than for unlabeled inner duals. If the automorphism group of the vertex labeled inner dual is nontrivial, again the canonical construction path method is applied in a way that is very similar to [2].

Kekuléan structures and structures embeddable in a lattice.

As an additional feature we implemented the option to restrict the generation to generalized fusenes with a Kekulé structure (or in mathematical terms: a perfect matching). In [4] an algorithm was given that added essential modifications to the algorithm from [2] in order to get optimal performance. We did not go that far here, but implemented just a filter combined with a look ahead deciding whether the fusene would have an even number of vertices, which is a necessary prerequisite for a Kekulé structure. This look ahead uses the fact described in the following lemma that the number of vertices is already fixed by the inner dual, so that inner duals that lead to structures with an odd number of vertices don't have to be labeled. The filter is a straightforward combination of a greedy approach and an exhaustive algorithm searching for Tutte paths increasing the size of a matching. As the structures dealt with are relatively small, more complicated matching algorithms with a better asymptotic complexity are not needed.

Lemma 6 *Let b be the length of the boundary cycle of an inner dual D of a $(n_3, \dots, n_k)$-fusene G. Furthermore let f denote the number of bounded faces of D. Then G has $\sum_{i=3}^{k}(i \cdot n_i) - b - 2f$ vertices.*

Proof: Counting vertices by summing up all face sizes – that is: computing $\sum_{i=3}^{k}(i \cdot n_i)$

– vertices that are contained in 3 bounded faces (internal vertices) are counted 3 times and vertices contained in 2 bounded faces are counted twice. The number of internal vertices is equal to the number of bounded faces in D – so we have to subtract $2f$ in order to count these vertices only once. Vertices contained in 2 faces are vertices in the boundary with degree 3. Each of these vertices has 2 edges in the boundary and 1 internal directed edge starting in the vertex that is not in the boundary. There are one-to one correspondences between these vertices and the internal directed edges on one hand and between the internal directed edges and directed edges corresponding to them in the inner dual. In the inner dual these edges are exactly the directed edges in the boundary cycle, so their number is b and we have to subtract b. ■

Benzenoids are fusenes that are subgraphs of the euclidean hexagonal lattice (or tiling). For constant vertex degree 3, triangles, squares and pentagons give a tiling of the sphere (the tetrahedron, cube and dodecahedron) and faces with size $s \geq 7$ give a regular tiling of the hyperbolic plane. This means that there is a natural generalization of the concept of benzenoids (although of more theoretical interest than actual chemical relevance): generalized fusenes with only one size of face that can be embedded into the corresponding (spherical or hyperbolic) regular tiling. As the spherical tilings are finite, there are obviously only few such generalized benzenoid-like fusenes for face size $s < 6$, while for the hyperbolic case ($s > 6$) the numbers grow fast. We implemented a filter for embeddability that also follows the lines of [2]: the corresponding lattice is built in the computer and it is tested whether the boundary of the generalized fusene is a simple closed curve in the tiling. There is only one additional optimization. That optimization is based on the following two easy lemmas:

Lemma 7 *Each generalized fusene with only one face size $k \geq 6$ has at least k vertices of degree 2 in the boundary.*

Proof: This result is a consequence of the Euler formula. Let f denote the number of bounded faces of the generalized fusene, v denote the number of vertices and e denote the number of edges. Furthermore let b denote the size of the outer face and b_2 denote the number of vertices in the boundary with degree 2.

Summing up the number of edges in each face we count each edge twice and get $e = \frac{k \cdot f + b}{2}$.

Summing up the number of vertices in each face, we count each vertex 3 times except for the vertices of degree 2. Adding b_2 to this sum we get $v = \frac{k \cdot f+b+b_2}{3}$.

Inserting this into the Euler formula we get

$\frac{k \cdot f+b+b_2}{3} - \frac{k \cdot f+b}{2} + (f+1) = 2$

which can be simplified to $2b_2 - b = 6 + (k-6)f$.

As $b_2 \leq b$ we get $b_2 \geq 6 + (k-6)f$ and as $f \geq 1$ and $(k-6) \geq 0$ finally $b_2 \geq k$. ■

With this lemma we can prove the following result, which makes it possible to decide that for some inner duals all corresponding generalized fusenes can be embedded. This look ahead speeds up the tests especially for large face sizes.

Lemma 8 *The inner dual of a generalized fusene G with only one face size $k \geq 6$ that can not be embedded into the corresponding lattice contains two vertices at distance at least $k-1$.*

Proof: Assume that a generalized fusene G with only one face size $k \geq 6$ can not be embedded into the corresponding lattice. Then there are faces f, f' so that different vertices in the boundary of these faces are mapped onto the same vertex in the lattice. For each such f, f' there is some shortest path connecting the corresponding vertices in the inner dual. This path corresponds to a smallest set $S = \{f = f_1, f_2, \ldots, f_m = f'\}$ of faces so that for $1 \leq i \leq m-1$ face f_i shares an edge with f_{i+1} in G. We choose f, f' so that m is as small as possible, which implies that no faces in $\{f_2, \ldots, f_{m-1}\}$ share vertices that are not on the connecting edges – otherwise choosing those as f, f' if the common edge is not in G, resp. a shortcut from f to f' through this edge if it is in G would give a corresponding set with fewer elements. In the dual lattice, this path of length m together with the edge connecting the centers of f and f' forms a cycle. The interior of this cycle must contain faces as otherwise one can easily prove that the two overlapping vertices also agree in G. There is only one component of such faces as otherwise faces in $\{f_2, \ldots, f_{m-1}\}$ would share vertices. This component is a generalized fusene and has at least k vertices of degree 2 in the boundary. At each such vertex the third edge sticking out is an edge shared by two k-gons in S, so S induces a cycle of length at least k in the dual lattice. One of the edges is the overlapping edge, but the others are also edges in the fusene, so the inner dual of the fusene contains a shortest path with at least k vertices and these are at distance at least $k-1$. ■

Testing

In order to test the program based on this algorithm, we implemented a second, very simple (and very inefficient) program. For a given set of n faces, this program first generates all generalized fusenes with one of this faces. When a list of all generalized fusenes with $m < n$ of the given faces is generated, each of the graphs in the list is read and each of the remaining faces are added in every possible way to the graph. The new graphs are piped through an isomorphism tester to make the new list with $m + 1$ faces. This approach is as simple as inefficient, but well suited for testing.

We compared the results for all combinations of $n \leq 8$ faces where each face had a size of at most n (so e.g., all combinations of 7 faces where each face is at least a triangle and at most a heptagon). Furthermore we compared the lexicographically first 1099 combinations of 9 faces with maximum face size 9. The first one had the face vector $(9, 0, 0, 0, 0, 0, 0)$ and the last one had the face vector $(2, 2, 1, 1, 1, 2, 0)$. All results agreed.

The internal routine for restricting the structures to those with a Kekulé-structure was tested by once applying the internal routine and once applying an independent external filter. All combinations of $n \leq 9$ faces with maximum size 9 were tested. There was complete agreement.

The routine testing whether the structure is embeddable in the corresponding regular tiling was tested by comparing the results for up to 21 hexagons with the known numbers for benzenoids. There was complete agreement.

Results

In this section we will give some results of the program for the probably most interesting parameter sets.

In the tables we will omit the case of only triangles, where there are unique structures for 1, 2 and 3 faces and otherwise none and the case of only 4-gons where there are 2 possible structures for 3, 4 and 5 faces and otherwise a unique structure. In these cases also embeddability and the existence of a Kekulé-structure can easily be seen.

Examples for generation rates on an Intel Xeon CPU E5-2690 0 with 2.90GHz running a large number of other jobs in parallel are 19.3 million graphs per second for face vector $(0, 0, 0, 0, 15)$, 16.8 million graphs per second for face vector $(0, 0, 2, 11, 2)$, and for face vector $(1, 1, 1, 1, 1, 1, 1, 1, 1)$ 26.9 million graphs per second.

faces	5-gons	7-gons	8-gons	9-gons	10gons
1	1	1	1	1	1
2	1	1	1	1	1
3	2	3	4	4	5
4	4	10	14	19	24
5	7	44	85	136	215
6	18	249	598	1226	2256
7	35	1513	4837	12259	26925
8	87	10002	41570	130435	339523
9	206	68455	372830	1441755	4456349
10	527	482571	3436289	16390266	60149731
11	1337	3470782	32373742	190484723	830492066
12	3524	25391403	310407548	2254489731	11681979493
13	9262	188353848	3020894755	27096464528	166929456330
14	24772	1414150643	29776585115	329995577356	2417719752083
15	66402	10728851795	296777846799	4065180800962	35429035715975
16	179589	82151290333	2986882651510	50584233942184	
17	487435	634203702080	30321553394791		
18	1330708	4931994064073			
19	3645754	38607981904156			
20	10031143	304032521364835			
21	27691894				
22	76708835				
23	213122892				
24	593851624				
25	1659104995				
26	4646951873				
27	13046181167				
28	36708619646				
29	103505545079				
30	292430771955				
31	827749791118				
32	2347201206583				
33	6667102499129				
34	18968089092814				

Table 1: Generalized fusenes with only one type of face.

For generalized Kekuléan fusenes on the same machine the generation rate for face vector $(0, 0, 1, 15)$ is 604.000 graphs per second and for $(0, 0, 2, 10, 2)$ it is 622.000 graphs per second. When testing embeddability in the lattice, a sample generation rate is 3.3 million graphs per second for face vector $(0, 0, 0, 0, 14)$.

To code the graphs (e.g. in *planarcode* format) and to write them to a pipe slows down the program. The factor depends on the parameters. In case the filter for Kekuléan structures is used, this factor can be close to 1, while in cases where the structures are not filtered the factor can be 10 or more. In each case the generation is so fast that for any possible application that actually has to deal with the graphs the construction of the graphs will not be the bottleneck.

faces	5-gons	7-gons	8-gons	9-gons	10gons
1	1	1	1	1	1
2	1	1	1	1	1
3	2	3	4	4	5
4	4	10	14	19	24
5	6	44	85	136	215
6	11	249	598	1226	2256
7	6	1512	4837	12259	26925
8	4	9990	41569	130435	339523
9	2	68279	372812	1441754	4456349
10	1	480508	3435918	16390242	60149730
11	1	3448337	32367494	190484033	830492034
12	0	25163082	310312435	2254474176	11681978319
13	0	186124663	3019551558	27096152001	166929422342
14	0	1393029504	29758541952	329989812650	2417718888714
15	0	10532933254	296543993549	4065080418871	
16	0	80362322284	2983929583401		
17	0	618059615229			
18	0	4787611202172			

Table 2: Generalized fusenes with only one type of face that are embeddable into the corresponding regular tiling.

hexagons	1 pentagon	1 heptagon	1 pentagon and 1 heptagon	2 pentagons and 2 heptagons
0	1	1	1	28
1	1	1	6	480
2	4	5	47	6220
3	15	20	315	65360
4	69	103	2063	609073
5	332	529	13063	5225332
6	1682	2849	81195	42303469
7	8682	15388	497640	328040556
8	45618	83967	3021681	2461330660
9	242111	459722	18224512	17994323957
10	1296961	2528448	109392728	128830158673
11	6998238	13953686	654339964	906650684335
12	38008177	77274705	3903951850	6289829698942
13	207593408	429300728	23247519162	
14	1139584340	2392234756	138238618771	
15	6284089975	13368323192	821140061024	
16	34794611081	74904618427	4873672024640	
17	193369623973	420750474440		
18	1078270552052	2368954994268		
19	6031209604370	13367204082634		

Table 3: Generalized fusenes with hexagons and few pentagons and heptagons.

hexagons	1 pentagon	1 heptagon	1 pentagon and 1 heptagon	2 pentagons and 2 heptagons
0	0	0	1	21
1	0	0	5	311
2	1	1	34	3747
3	4	5	204	37056
4	25	35	1245	333102
5	132	204	7454	2783184
6	722	1197	44625	22118058
7	3876	6785	265422	169075450
8	20910	38167	1574098	1254316891
9	112612	212687	9307150	9083594842
10	607677	1180305	54920628	64499257088
11	3283523	6529878	323535108	450517084307
12	17778908	36073732	1903402511	
13	96466121	199152674	11185215834	
14	524583808	1099465971	65664366284	
15	2859087380	6072329926	385153421764	
16	15617401965	33561177471		
17	85494407414	185657286262		
18	469015356870	1028104183734		

Table 4: Generalized Kekuléan fusenes with hexagons and few pentagons and heptagons.

References

[1] A. T. Balaban, F. Harary, Chemical graphs V. Enumeration and proposed nomenclature of benzenoid cata–condensed polycyclic aromatic hydrocarbons, *Tetrahedron* **24** (1968) 2505–2516.

[2] G. Brinkmann, G. Caporossi, P. Hansen, A constructive enumeration of fusenes and benzenoids, *J. Algorithms* **45** (2002) 155–166.

[3] G. Brinkmann, G. Caporossi, P. Hansen, A survey and new results on computer enumeration of polyhex and fusene hydrocarbons, *J. Chem. Inf. Comput. Sci.* **43** (2003) 842–851.

[4] G. Brinkmann, C. Grothaus, I. Gutman, Fusenes and benzenoids with perfect matchings, *J. Math. Chem.* **42** (2007) 909–924.

[5] G. Caporossi, P. Hansen, M. Zheng, Enumeration of fusenes to $h = 20$, in: P.Hansen, P. Fowler, M. Zheng (Eds.), *Discrete Mathematical Chemistry*, Am. Math. Soc., Providence, 2000, pp. 63–78.

[6] J. R. Dias, Indacenoid isomers of semibuckminsterfullerene (buckybowl) and heir topological characteristics, *J. Chem. Inf. Comput. Sci.* **35** (1995) 148–151.

[7] P. W. Fowler, D. E. Manolopoulos, *An Atlas of Fullerenes*, Oxford Univ. Press, Oxford, 1995.

[8] P. W. Fowler, D. Mitchell, Electronic and steric factors in the stability of a proto–fullerene framework: indacenoid isomers of C30H12, *J. Chem. Inf. Comput. Sci.* **35** (1995) 874–878.

[9] G. Brinkmann, O. Delgado–Friedrichs, S. Lisken, A. Peeters, N. Van Cleemput, CaGe – a virtual environment for studying some special classes of plane graphs – an update, *MATCH Commun. Math. Comput. Chem.* **63** (2010) 533–552.

[10] P. Goetschalckx, Constructie van veralgemeende fusenen, Master's thesis, Univ. Gent, Gent, 2015.

[11] S. Iijima, T. Ichihashi, Y. Ando, Pentagons, heptagons and negative curvature in graphite microtubule growth, *Nature* **356** (1992) 776–778.

[12] H. W. Kroto, J. R. Heath, S. C. O'Brien, R. F. Curl, R. E. Smalley, C_{60}: Buckminsterfullerene, *Nature* **318** (1985) 162–163.

[13] B. D. McKay, Isomorph–free exhaustive generation, block, *J. Algorithms* **26** (1998) 306–324.

[14] R. F. Service, Mixing nanotube structures to make a tiny switch, *Science* **271** (1996) 1232–1232.

[15] X. Zhang, H. Yao, Fixed bonds in a class of polygonal systems, *Polycyc. Arom. Comp.* **27** (2007) 95–105.