Decomposing, Counting, and Generating Unlabeled Cubic Planar Graphs

Manuel Bodirsky[1,3], Clemens Gröpl[2], and Mihyun Kang[1,4]

[1] Humboldt-Universität zu Berlin, Institut für Informatik, Unter den Linden 6, D-10099 Berlin, Germany
{bodirsky,kang}@informatik.hu-berlin.de

[2] Freie Universität Berlin, Institut für Informatik, Takustraße 9, D-14195 Berlin, Germany
groepl@inf.fu-berlin.de

Abstract. We present an expected polynomial time algorithm to generate an unlabeled connected cubic planar graph uniformly at random. We first consider *rooted* cubic planar graphs, where rooting means that certain symmetries are broken by specifying one edge as a root and assigning a direction to it. We derive recurrence formulas for counting the exact number of rooted cubic planar graphs, based on decompositions along the connectivity structure. This leads to 3-connected planar graphs, which have a unique embedding on the sphere; but special care has to be taken for *rooted* graphs, which can have a *sense-reversing* and a *pole-exchanging automorphism*. Therefore we introduce the concept of a *colored network* which is used to count rooted graphs with given symmetries via bijections. Colored networks can again be decomposed along their connectivity structure. For rooted 3-connected cubic planar graphs embedded in the plane, we switch to the dual and count rooted triangulations. All these numbers can be evaluated in polynomial time by dynamic programming. We can use them to generate rooted cubic planar graphs uniformly at random. To generate connected cubic planar graphs *without a root* uniformly at random, we apply rejection sampling and obtain an expected polynomial time algorithm.

1 Introduction

The number of planar graphs, and various subclasses thereof, has been investigated for a long time (see e.g. [14] for a survey). Significant progress has been achieved only recently [2, 4, 18], but the picture is still far from complete. When the graphs are counted up to isomorphisms, i. e., we consider *unlabeled* planar graphs, the situation is even more complex.

A *maximal* planar graph is a planar graph where we cannot add an edge without destroying planarity. These graphs are *triangulations*, and Tutte gave an exact formula for the labeled case [22]. Since almost all triangulations do not have a nontrivial automorphism, this also yields the asymptotic number of unlabeled triangulations [21]. Every triangulation is three-connected, and thus has a unique embedding on the sphere (see e.g. [9]). The dual of a triangulation is a three-connected *cubic* planar graph, i.e., all vertices have degree three. Conversely, every three-connected cubic planar graph is the dual of a triangulation.

If a cubic planar graph is not three-connected, it might have several non-equivalent embeddings. The dual of a cubic (planar) *map*, i.e., a connected cubic graph embedded in

[3] Supported by the Deutsche Forschungsgemeinschaft within the European graduate program 'Combinatorics, Geometry, and Computation' (DFG grant no. GRK 588/2).

[2] Supported by the German Federal Ministry of Education and Research within the Berlin Center for Genome Based Bioinformatics (BMBF grant no. 031 2705A).

[4] Supported by the Deutsche Forschungsgemeinschaft (DFG FOR 413/1-1).

the plane, is still a triangulation, but might contain multiple edges and self-loops. Using generating functions it is possible to determine the asymptotic growth of the number of cubic planar maps [1]. Brinkmann and McKay [5] developed an algorithm that produces a list of non-isomorphic planar cubic graphs.

Many difficulties in the enumeration of general unlabeled planar graphs already show up even if the graph is cubic. On the other hand cubic graphs have properties that simplify many arguments. A powerful strategy to count them is to decompose the graphs along their connectivity structure [23]. From the decomposition we can derive recursive formulas, and these formulas can be used to sample such objects from the uniform distribution, that is, to generate them *uniformly at random*. This approach is an instance of the so-called *recursive method* for sampling; an early reference for this method is [17]. The approach was systematized e.g. in [11].

Another technique for sampling is based on Markov chains. It can be successful even when it is difficult to apply any deterministic method to analyze the combinatorial objects. Markov chains on combinatorial objects having a uniform stationary distribution can be used for *Monte Carlo* algorithms to sample random instances *approximately* uniformly at random. These algorithms are efficient if the Markov chain is *rapidly mixing*, that is, the number of steps until the Markov chain is “close” to the uniform distribution (which is called the *mixing time*) is polynomial [13]. There are such efficient algorithms for the generation of certain random structures, for instance, triangulations of convex polygons [15, 16]. However it is not known whether the Markov chain in [8] for the uniform generation of planar graphs is rapidly mixing. This crucial question is open even for the analogous Markov chain for outerplanar graphs.

The recursive method for random sampling via the decomposition along the connectivity structure has several advantages. The sampled objects are *exactly* uniformly distributed because each generation procedure is branching into subroutines with the right probabilities evaluated by counting formulas. The counting formulas are interesting in their own right. Finally, the running time for the generation improves considerably if we allow precomputation to compute the values of the dynamic programming arrays.

Sketch of the strategy. In order to count unlabeled cubic planar graphs, we first have to *root* them. Here the root is a distinguished directed edge, and rooted cubic planar graphs are counted up to isomorphisms that fix the vertices of this edge. Clearly, generating a random rooted cubic planar graph and then simply ignoring the root edge does not yield the uniform distribution, since unlabeled graphs generally correspond to different numbers of rooted graphs. But this imbalance can be compensated by *rejection sampling*, i.e., the generation algorithm is restarted with a probability that is inverse proportional to the size of the orbit of the root (which is linear). In this way we can generate unlabeled connected cubic planar graphs in expected polynomial time.

Then we *decompose* rooted cubic planar graphs along their connectivity structure. Our approach here is motivated by [14], and in principle similar to the one described in [3]; but for unlabeled structures we need several new techniques. A classical theorem of Whitney (see e.g. [9]) says that a rooted 3-connected planar graph can have either one or two

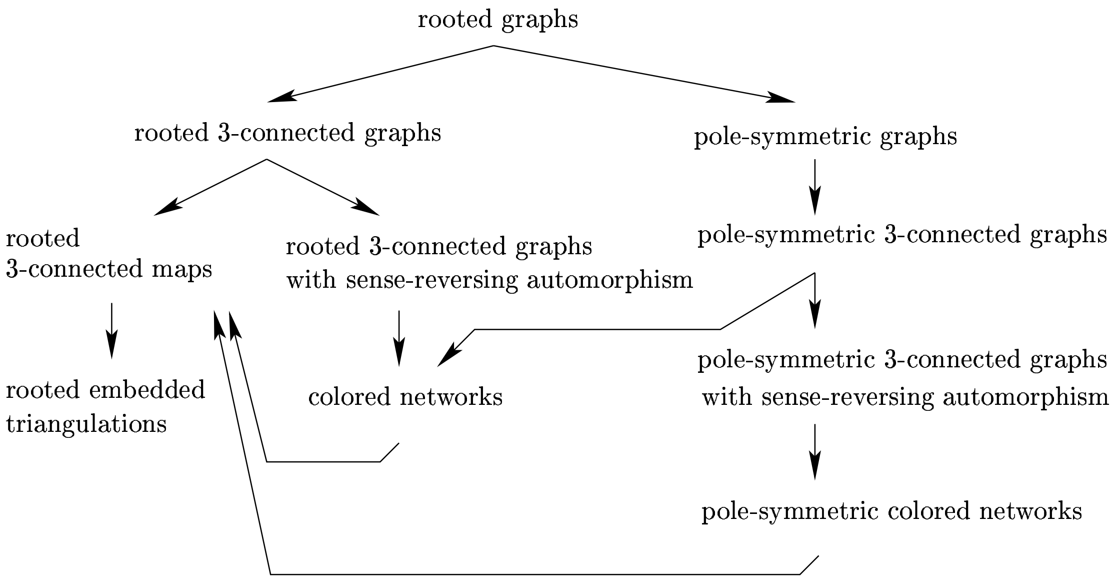

Fig. 1. Dependencies of the sections and concepts in this paper.

embeddings in the plane where the root edge is embedded on the outer face. In the first case we say that it has a *sense-reversing automorphism* or it is *symmetric*. We deal with the symmetric and the asymmetric case separately. For 3-connected planar graphs it is difficult to deal with the asymmetric case explicitly. Therefore we only provide a decomposition strategy, counting formulas, and a generation procedure for the symmetric case and for rooted 3-connected maps. Having this, we can also generate the asymmetric case: We first generate a rooted 3-connected map uniformly at random, and test whether the generated object has a symmetry or not. Then we can use rejection sampling, since we know the number of symmetric and asymmetric objects. This yields an expected polynomial time algorithm, since the number of asymmetric objects is much larger than the number of symmetric objects. In the case of three-connected cubic planar graphs this was shown for the dual objects [21], which are planar triangulations. To count and generate triangulations, we decompose the more general rooted *near-triangulations*, i.e., embedded graphs where every face except the outer face is a triangle. For these objects an exact formula was given by Brown [6]. In our decomposition we control the number of vertices at the outer face and additionally the degree of one of the poles, as we need to have control over this parameter for the earlier steps of the decomposition.

In the decomposition of rooted graphs we also have to deal with automorphisms that exchange the poles. In this case we say that the rooted graphs are *pole-symmetric*, and we decompose and count them along their connectivity structure until we end up with pole-symmetric 3-connected graphs. We are left with the task to count and generate symmetric 3-connected graphs and pole-symmetric 3-connected graphs. For that we show new bijective correspondences to *colored networks*, a new concept that we introduce in Section 5, and decompositions of these objects. We believe that planar graphs with various symmetries and their enumeration are interesting in their own right. Recently they also attracted attention in graph drawing [10].

Plan of the paper. We first introduce important concepts for the decomposition of rooted graphs. Figure 1 shows the dependencies between the concepts in the decomposition and the counting formulas, and also gives an overview of the dependencies of the remaining sections in this paper. In the conclusion we show how to apply these decomposition results and the exact counting formulas to generate connected cubic planar graphs uniformly at random.

2 Graph Decompositions

In this section we introduce concepts that we need for the decomposition of graphs. But first we recall some graph-theoretical concepts. A graph is *simple* if it does not contain multiple-edges or self-loops. A graph is *planar* if it can be embedded in the plane, and it is *cubic* if every vertex has degree three. A graph G is called k*-connected* if $|G| > 2$ and $G - K$ is connected for every set $K \subset G$ with $|K| < k$.

Let G be a simple graph with distinguished vertices s, t and let $e = st$ be a separate directed edge from s to t (called the *root-edge*) such that if we insert the edge e into G the resulting multigraph G^* is connected. If s and t were already connected by an edge in G, we introduce a multi-edge in G^*. We also allow the case that e is a self-loop. We call such graphs *rooted*, and say that a rooted graph G is planar, cubic, or k-connected if and only if G^* has the corresponding property. A k-point set K of a graph G is called a k*-cut* of G if $G - K$ is disconnected. A single point that forms a 1-cut is called a *cut-vertex*. A 2-cut is also called a *split pair*. Two adjacent cut-vertices form a *cut-edge*, and we say that a cut-edge *separates* two vertices u, v if u, v are contained in two different connected components of G after deleting the cut-edge (but not its ends) from G.

For cubic graphs we often make use of the following operation: We *replace* an edge uv by a graph rooted at st by connecting u to s and v to t. If the original graph and the rooted graph were cubic, the resulting graph will again be cubic. Then we have the following structure theorem for rooted cubic graphs:

Theorem 1. *A rooted cubic graph G is of precisely one of the following types:*

- *d: The vertices s and t are contained in two different connected components of G.*
- *s: There is a cut-edge in G that separates s and t.*
- *p: There is no cut-edge in G separating s and t, and either s and t are adjacent in G or they form a split pair of G.*
- *h: The graph G is built from a uniquely determined 3-connected rooted cubic graph H, where the root is not a self-loop or a multi-edge in H^*, by replacing some edges of H (in the above sense) with rooted s-, p-, h-, or g-graphs.*
- *g: The root is a self-loop, i.e., $s = t$.*

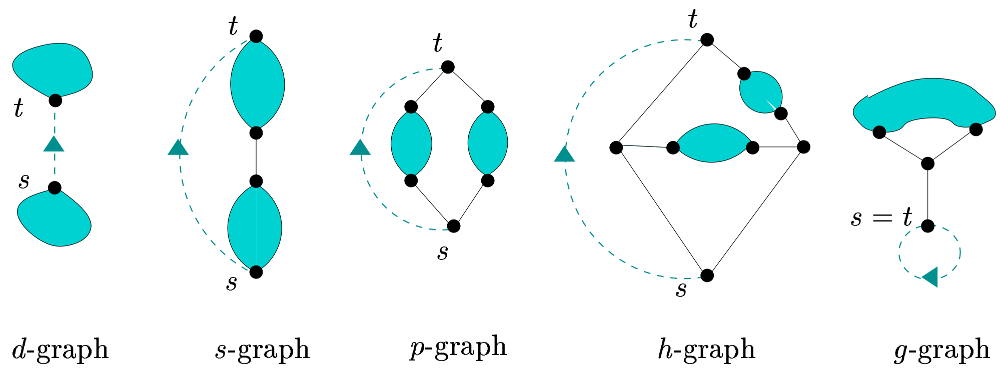

Fig. 2. Types of rooted cubic graphs

Proof. The cases that G is a g- or d-graph are obviously disjoint from the other cases. If G is an s-graph it cannot be a p-graph, and vice versa. If G contains a split pair, one of the corresponding components does not contain a root vertex. We replace this component by an edge if we can preserve three-regularity. If we iterate this process, the graph will eventually be 3-connected, or an s- or p-graph. Since the replacement operations are confluent and always terminating, the resulting 3-connected cubic graph H will be unique. □

The corresponding rooted graphs are called d- (for "disconnected"), s- (for "serial"), p- (for "parallel"), h-, or *g-graphs*, respectively. The five types of cubic rooted graphs are shown in Figure 2. A *network* N is a rooted graph such that N^* is 2-connected and the root is not a self-loop. The unique network with 2 vertices only is called the *trivial* network. Note that for a network N, the rooted graph N^* cannot be a d- or a g-graph. The networks corresponding to the remaining cases we call s-, p-, or h-networks, respectively. In networks, the vertices s and t of the root are also called the *poles* of the network. Every split pair $\{k_1, k_2\}$ in N induces a partition of the edge set, and each of these parts is again a network, where k_1k_2 are the poles. These networks are called *subnetworks* of N. If $k_1, k_2 \notin \{s, t\}$ the split pair $\{k_1, k_2\}$ induces two other networks N_1 and N_2 with the poles st and k_1k_2, respectively, in the sense that we can obtain N by replacing an edge uv of N_1 by N_2.

All the main results of this paper are proven for *planar* graphs only. Thus, if we speak about graphs, we always assume that they are planar, and do not mention this explicitly. However, planarity will only be used for the geometric arguments for three-connected graphs. The other decomposition arguments extend to arbitrary graph classes. For technical reasons, we allow the cases that the root of a rooted graph might be a self-loop or a multi-edge. If this is not the case for a rooted graph G we call G a *rooted simple graph*, which corresponds to a "standard" rooted cubic graph that has neither a multi-edge nor a self-loop, but a distinguished directed edge. We will also enumerate these objects at the end of the next section.

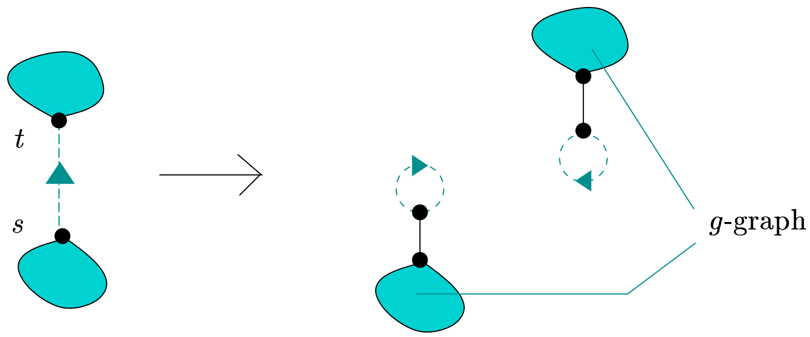

Fig. 3. Decomposition of a d-graph

3 Counting Rooted Cubic Graphs

In this section we present a decomposition of rooted cubic graphs, and derive recurrence formulas for counting them. Let $r(n)$ be the number of rooted cubic graphs on n vertices. According to Theorem 1 we have $r(n) = d(n) + s(n) + p(n) + h(n) + g(n)$, for $n \geq 0$, where the functions $d(n), s(n), p(n), h(n), g(n)$ count the number of d-, s-, p-, h- and g-graphs on n vertices, respectively. It will be convenient to use the numbers $c(n) = r(n) - d(n)$, $n \geq 1$, $c(0) = 1$, and call the corresponding rooted graphs *c-graphs* (c for "connected"). Because of three-regularity, for odd n, there is no rooted cubic graph, that is, these functions take values zero. One general remark concerning the recurrences in this article: We often give only the most important initial cases. But in all cases an exhaustive list of initial cases can be deduced easily from the definitions of the counting functions.

Counting d-graphs. Since a d-graph G without the root edge is disconnected, we can decompose a d-graph G uniquely into two g-graphs (see Fig. 3): To obtain the first g-graph, we shrink the connected component containing t in G into the vertex t, connect s and t, and think of a self-loop at t as a root-edge. To obtain the second g-graph, we do the same for the connected component containing s. For $0 \leq n \leq 11$, $d(n) = 0$, and for $n \geq 12$,

$$d(n) \;=\; \sum_{i=6}^{n-4} g(i)g(n+2-i)\,.$$

Counting s-graphs. Note that every s-graph S has a unique cut-edge uv separating s, t such that u is closest to the vertex s (in this paper, *closest* is meant with respect to the length of a shortest connecting path). It could be the case that $u = s$, or $v = t$. An s-graph can be split into a p-, h-, or g-graph rooted at su, and an s-, p-, h-, or g-graph (i.e., a c-graph) rooted at vt (see Fig. 4). Initially, $s(n) = 0$ for $0 \leq n \leq 7$. For $n \geq 8$,

$$s(n) \;=\; \sum_{i=4}^{n-4} (p(i) + h(i) + g(i))c(n-i)\,.$$

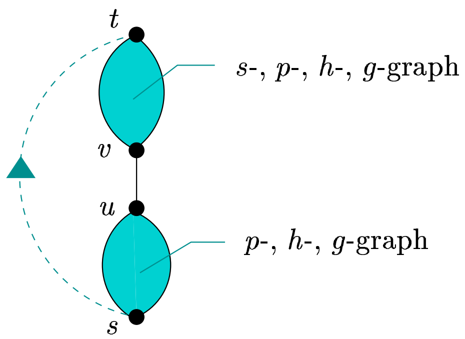

Fig. 4. Decomposition of an s-graph

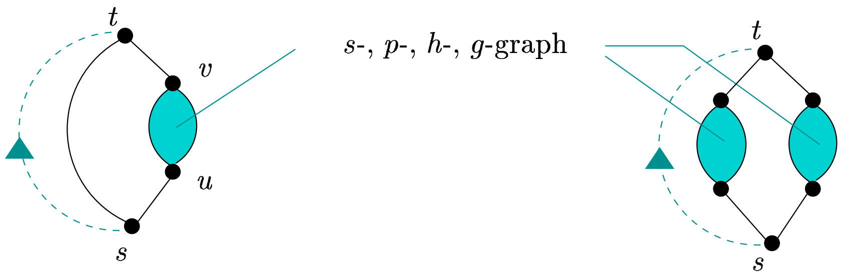

Fig. 5. Decomposition of a p-graph

Counting p-graphs. A p-graph G could contain a non-root edge between s and t. If this is the case (see Figure 5, left part), then due to the three-regularity of G^*, there are unique non-pole neighbors u and v for s and t, respectively, and $\{u, v\}$ induces a smaller graph rooted at uv that is a c-graph. These graphs are counted by $c(n-2)$. If the poles are not adjacent (see Figure 5, right part), a p-graph can be split into two c-graphs, of size i and $n-i-2$. These subgraphs are counted by $c(i)$ and $c(n-i-2)$. We must not count twice the case where the two subgraphs are isomorphic, and this explains the following formula: for $0 \leq n \leq 3$, $p(n) = 0$, and for $n \geq 4$,

$$p(n) \;=\; c(n{-}2) \;+\; \sum_{i=4}^{(n-4)/2} c(i)c(n{-}i{-}2) \;+\; \binom{c((n{-}2)/2)+1}{2}.$$

Counting g-graphs. A rooted cubic g-graph G has a unique neighbor u of the pole $s = t$, which in turn is adjacent to two other distinct vertices w_1 and w_2 in G (see Figure 6). Note that the orientation of the root of the remainder does not matter. Thus we obtain the same g-graph in two ways, unless there is an automorphism mapping w_1 to w_2 and w_2 to w_1. The number of rooted graphs that have such a *pole-exchanging* automorphism will be counted by $\tilde{r}(n)$ in Section 6. Therefore for $0 \leq n \leq 5$, $g(n) = 0$, and for $n \geq 6$,

$$g(n) \;=\; \big(r(n{-}2) \;-\; g(n{-}2) \;+\; \tilde{r}(n{-}2)\big)/2\,.$$

Counting h-graphs. Let G be an h-graph. Theorem 1 asserts that there is a unique 3-connected rooted cubic graph H, such that we can derive G from H by replacing edges

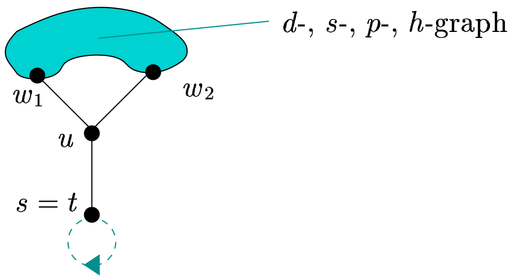

Fig. 6. Decomposition of a g-graph

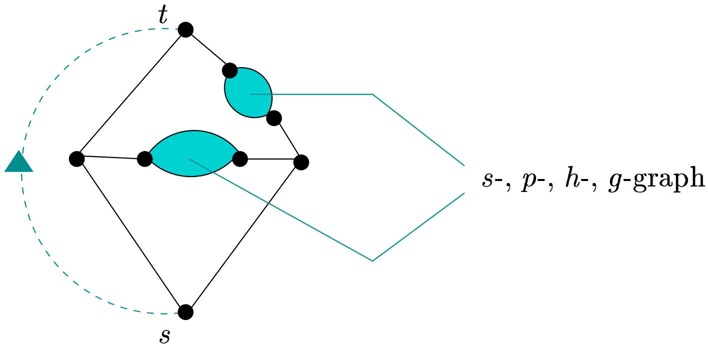

Fig. 7. Decomposition of an h-graph

in H with rooted c-graphs. We call H the *core* of G and denote $H = \text{core}(G)$. We call G *symmetric* if it has a *sense-reversing automorphism* φ, i. e., $\varphi \neq \text{id}$, but $\varphi(s) = s$ and $\varphi(t) = t$, and *asymmetric* otherwise. In the remainder of this section we treat the symmetric and the asymmetric case separately. We describe how to construct them from symmetric and from asymmetric 3-connected networks. If the core of G is asymmetric, we can uniquely order the edges of the core of G. For that, we define two sequences, for each of the two possible embeddings of the core in the plane that have the root at the outer face. To define the sequence for one of these embeddings, we do a depth-first search traversal of the core, beginning with the root edge and visiting the neighbors of a vertex in clockwise order with respect to the embedding. We label the vertices with numbers according to the order of their occurrence in the traversal. We can now denote edges by pairs of such vertex labels. The sequence we are interested in is the sequence of edges represented in that way, in the order as we traversed them in the depth-first search. If the core is asymmetric, the two sequences are distinct, since we can reconstruct the graph from such a sequence. We can now order the edges of the core uniquely according to the lexicographically smaller sequence.

If the graph has a symmetric core, clearly both edge sequences are the same. In this case we order the edges of the core in the following way. Let φ be the nontrivial isomorphism. We start with the edges uv where $u = \varphi(u)$ and $v = \varphi(v)$ according to the traversal: we say such edges are *blue*. Then we list the edges uv where $v = \varphi(u)$ and $u = \varphi(v)$ according to the traversal: we say such edges are *red*. We continue with the edges that are not fixed by the φ, and order them according to the traversal. Edges and their images, sometimes called *corresponding edges*, are ordered arbitrarily.

Let $b_{b,r,l}(n)$ denote the number of symmetric h-graphs B on n vertices, where core(B) has b blue and r red edges, and the first l edges of core(B) are edges of B. To generate such a symmetric h-graph B, we have to start from a symmetric core. We look at the l+1-st edge of core(B). It might also be an edge in B. If not, it is either blue (for $l{+}1 \leq b$), red (for $b < l{+}1 \leq b{+}r$), or uncolored (for $l{+}1 > b{+}r$). If it is blue we can replace it by an arbitrary s-, p-, h-, or g-graph (i.e., c-graph). If it is red we can replace it by a graph with a pole-exchanging automorphism (counted by $\tilde{c}(n)$, again see Section 6). Otherwise we can replace corresponding uncolored edges in pairs by the same copy of a c-graph. When l equals the number of edges of B, i.e., $l = 3n/2 - 1$, we have a 3-connected symmetric network with b blue and r red edges, which will be counted in Section 5. Thus for $0 \leq n \leq 3$, $b_{b,r,l}(n) = 0$, and for $n \geq 4$,

$$b_{b,r,l}(n) = \begin{cases} \sum_{i\geq 0} c(i) b_{b,r,l+1}(n-i) & \text{for } l{+}1 \leq b \\ \sum_{i\geq 0} \tilde{c}(i) b_{b,r,l+1}(n-i) & \text{for } b < l{+}1 \leq b{+}r \\ \sum_{i\geq 0} c(i) b_{b,r,l+2}(n-2i) & \text{for } l{+}1 > b{+}r\,. \end{cases}$$

Let $a_l(n)$ denote the number of asymmetric h-graphs A on n vertices, where the first l edges of core(A) are edges of B. To generate such an asymmetric h-graph A, we could first take an h-graph which is already asymmetric (counted by a_{l+1}) and replace the l+1-st core edge by a c-graph. Or we could take a symmetric h-graph whose core has b blue and r red edges and the $l+1$-st edge is red, and replace this edge by a c-graph that has no automorphism exchanging the poles; for the enumeration of graphs with and without a pole-exchanging automorphism we refer to Section 6. Finally, we could take such a symmetric h-graph, where the $l+1$-st edge is not fixed by the automorphism, and substitute two different c-graphs for the corresponding two core edges. Again when l equals the number of edges of A, we have to count the number of 3-connected asymmetric networks; for that we refer to Section 5. Hence for $0 \leq n \leq 5$, $a_l(n) = 0$, and for $n \geq 6$,

$$\begin{aligned} a_l(n) = & \sum_{i\geq 0} c(i) a_{l+1}(n-i) + \\ & \begin{cases} \sum_{b,r\geq 1, i\geq 0} (c(i) - \tilde{c}(i)) b_{b,r,l+1}(n-i) & \text{for } b < l{+}1 \leq b{+}r \\ \sum_{b,r\geq 1, i,j\geq 0} c(i)c(j) b_{b,r,l+2}(n-i-j) - c(i) b_{r,l+2}(n-2i) & \text{for } l{+}1 > b{+}r\,. \end{cases} \end{aligned}$$

With these numbers we can compute $h(n) = a_0(n) + \sum_{b,r\geq 1} b_{b,r,0}(n)$.

Counting simple graphs. We can also compute the number of rooted simple cubic planar graphs with the functions already introduced. Note that a g- graph together with the root edge is not simple, since the root edge creates a self-loop. Every d-, or h- graph together with the root edge is simple. But some s-graphs and some p-graphs together with the root edge are not simple: An s-graph that can be split into two g-graphs, together with the root edge, is not simple, because the root edge forms a multi-edge between the poles.

A p-graph that can be split into a single edge and a c-graph is not simple, because the root edge again forms a multi-edge. Thus the number of rooted simple cubic planar graphs on n vertices is

$$d(n) + h(n) + s(n) - \sum_i g(i)g(n-i) + p(n) - c(n-2).$$

4 Networks with a Sense-reversing Automorphism

We want to count 3-connected cubic planar graphs with a distinguished directed edge up to isomorphisms that fix this root edge. If we additionally specify a face incident to the root edge, then there is a unique embedding in the plane where this face is the outer face. Thus we can as well count the dual objects. These are *face-rooted triangulations*, i.e. maximal planar graphs with a distinguished directed edge and an incident face. A closed counting formula for face-rooted triangulations was derived by Tutte [20], see also Section 9.

In general, there are two ways how to obtain a face-rooted 3-connected planar graph from an edge-rooted one. But if the graph is *symmetric* then there is only one corresponding face-rooted graph. Therefore we need to compute the number of the symmetric graphs; then we can also compute the number of asymmetric graphs. Moreover, we can also *generate* asymmetric graphs, using *rejection sampling*: First we generate a face-rooted 3-connected planar graph. Then we check whether it is symmetric or not. This can be done in linear time using known techniques for planar graph isomorphism [12]. If the graph is symmetric, we reject it and restart the procedure. The procedure will terminate after a constant number of rounds in expectation, since there are more asymmetric than symmetric graphs. For 3-connected planar graphs, there are even asymptotically more asymmetric than symmetric graphs and one round will suffice in most cases. This follows from a result formulated for the dual objects, namely triangulations [21].

Next, we count face-rooted 3-connected cubic planar graphs with a *sense-reversing automorphism*, i.e., an automorphism that fixes the root edge, but maps one incident face to the other. Let H be a symmetric edge-rooted 3-connected cubic planar graph, and φ be its sense-reversing automorphism. Throughout this section we assume that the root edge st is not a self-loop, i.e., $s \neq t$. A vertex v of H is called *blue* if $\varphi(v) = v$, and *red* if v is connected to $\varphi(v)$ by an edge. Thus a vertex is either red, blue, or uncolored, and the poles are blue. We can think of H as being embedded in such a way that φ corresponds to a reflection, the blue vertices are aligned on the reflection axis, and the red vertices are connected by an edge crossing this axis perpendicularly (see Fig. 8, left part). Our arguments do not rely on such a representation, however.

If we remove from H the blue vertices and the edges between red vertices (i. e., we cut H along the symmetry axis), then the resulting graph has exactly two connected components. The graphs induced by these components and the blue vertices are isomorphic and will be called H_1 and H_2 (see Fig. 8, right part). We claim that H_1^* is 2-connected, and hence H_1 is a network: Suppose there was a cut-vertex v in H_1^*. Then this cut-vertex together with the cut-vertex $\varphi(v)$ in H_2^* forms a 2-cut in H^*, contradicting the 3-connectivity of H^*.

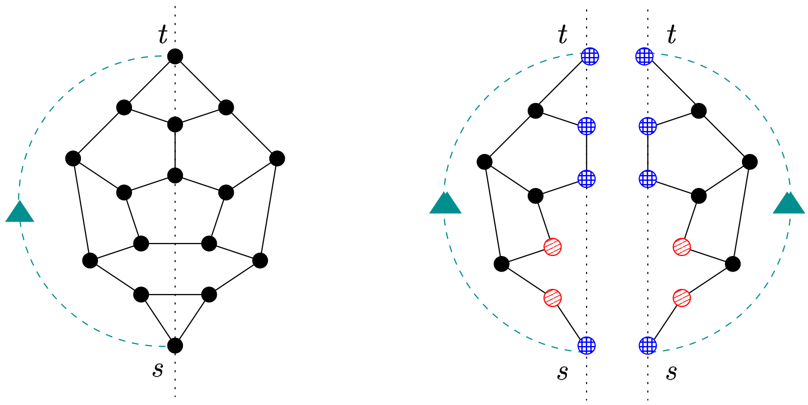

Fig. 8. Decomposition of a symmetric 3-connected network

Now we extract some more properties of the graphs H_1 and H_2 and define *colored networks*. They are defined in such a way that we can recursively decompose them, and that we can establish a bijection between symmetric 3-connected networks and colored networks.

Definition 1. *A* colored network *is a network N in which some vertices are colored* red *and* blue *such that:*

(P1) *The colored vertices have degree 2 in N^*, and all other vertices have degree 3.*
(P2) *N^* has an embedding such that all colored vertices and the poles lie on the outer face.*
(P3) *N and all subnetworks of N contain at least one colored vertex.*
(P4) *No nontrivial subnetwork N' of N has two blue poles.*

The bijection to symmetric 3-connected networks looks as follows.

Theorem 2. *For $3, b, r < n$ there is a bijection between the following two sets of objects:*

(i) colored networks with $(n + b)/2$ vertices and blue poles, where b is the number of blue vertices and r the number of red vertices, and
(ii) 3-connected networks on n vertices that have a nontrivial automorphism that fixes b vertices and exchanges r pairs of adjacent vertices.

Proof. Given a symmetric 3-connected network H, we first check that both H_1 and H_2, constructed as described above, are networks and satisfy properties (P1) – (P4). They are 2-connected, since if there was a cut-vertex in H_1, we would also have a cut-vertex in H_2, and together they would form a 2-cut in H^*. No nontrivial subnetwork has two blue pole vertices, since these blue pole vertices would be a nontrivial 2-cut in H^*. This contradicts the assumption that H is 3-connected. (P1) and (P1) are immediate from the definition of H_1 and H_2. (P3): Every subnetwork contains a colored vertex, since otherwise its poles would be a 2-cut in H^*. (P4): No subnetwork has two blue pole vertices, since these blue pole vertices would be a 2-cut in H^*.

Conversely, we have to (re-)construct for every colored network H_1 with blue poles the corresponding symmetric 3-connected network H. First we make an isomorphic copy H_2

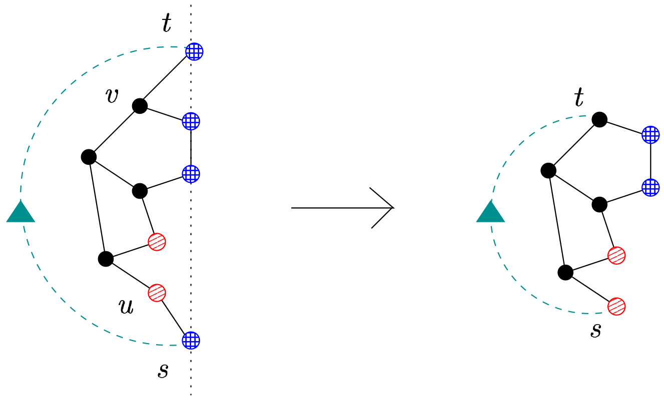

Fig. 9. Colored network

of H_1. Then we identify corresponding blue vertices in H_1 and H_2, and connect by edges corresponding red vertices in H_1 and H_2. The constructed graph H is clearly a symmetric, by (P2) planar, and by (P1) a cubic network. Suppose H^* was not 3-connected. Then there is a split pair $\{k_1, k_2\}$ in H^* that determines two subnetworks N_1 and N_2. We distinguish four cases:

(1) Both of k_1, k_2 are blue. – This is impossible because then H_1 or H_2 would contain a subnetwork with two blue poles k_1, k_2, contradicting (P4).
(2) Exactly one of k_1, k_2 is blue. – Then wlog. k_2 is blue and k_1 is in $H_1 \setminus H_2$. Let N_1' and N_2' be those (non-empty) parts of N_1 and N_2 that also lie in H_1. By (P3) there are colored vertices $v_1 \in N_1'$ and $v_2 \in N_2'$. Since H_2 is 2-connected, there is a path from v_1 to v_2 passing through H_2 and avoiding k_2 (and k_1), which contradicts the assumption that k_1, k_2 is a split pair in H.
(3) None of k_1, k_2 is blue, and either both lie in H_1 or both lie in H_2. – Suppose wlog. both vertices lie in H_1. Then k_1 and k_2 define a nontrivial subnetwork in H_1. But since every such subnetwork contains a colored vertex, this contradicts that k_1, k_2 is a 2-cut in H^*.
(4) Again none of k_1, k_2 is blue, but this time k_1 is in $H_1 \setminus H_2$, and k_2 is in $H_2 \setminus H_1$. – It cannot be that H_1 contains vertices from both N_1 and N_2 because of 2-connectivity; the same for H_2. Thus H_1 either equals N_1 or N_2, which is impossible by (P3), since a colored vertex in H_1 again contradicts that k_1, k_2 is a 2-cut in H^*. □

5 Colored Networks

In this section we show how to decompose and count colored networks that bijectively correspond to 3-connected networks with a sense-reversing automorphism.

Because we consider cubic networks, a colored network with blue poles s and t (see Fig. 9, left side) has a unique colored subnetwork with poles u, v where u is adjacent to s and v is adjacent to t, and both u and v are not blue (see Fig. 9, right hand side). Let $n'_{b,r,k}(n)$ denote the number of colored networks, with b blue vertices, r red vertices, and n the total number of vertices, that have no blue cut-vertex and whose both poles are

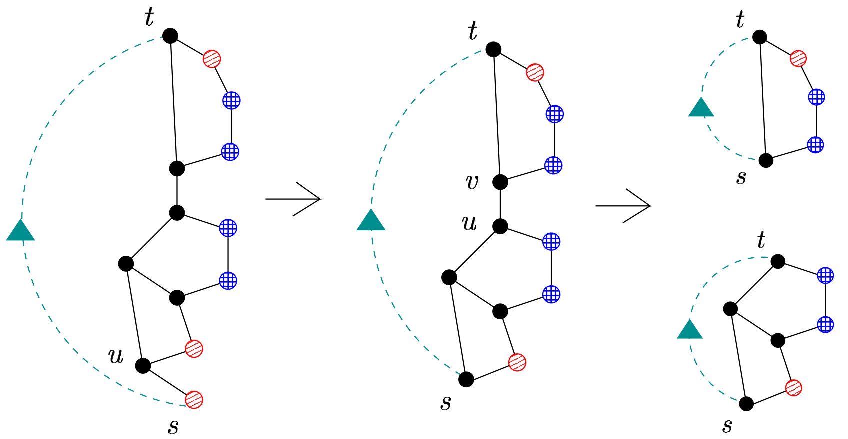

Fig. 10. Colored s-networks with no blue cut-vertex

not blue. Moreover we require for flexibility (used in later sections) that the first colored vertex has distance k to the s-pole. If this distance does not matter we use the function $n'_{b,r}(n)$, which equals $\sum_k n'_{b,r,k}(n)$. By Theorem 2 we can express the number of symmetric 3-connected networks as $b_{b,r,3n/2-1}(n) = n'_{b,r}((n+b)/2-2)$ for $n \geq 4$.

To count colored networks with non-blue poles we introduce the functions $s'_{b,r,k}(n)$, $p'_{b,r,k}(n)$, and $h'_{b,r,k}(n)$ that count the number of colored s-, p, and h-networks where no cut-vertex and no pole is blue, and let $n'_{b,r,k}(n) = s'_{b,r,k}(n) + p'_{b,r,k}(n) + h'_{b,r,k}(n)$. To drive the recurrence we need one more function: Let $s''_{b,r,k}(n)$ be the number of colored s-networks such that both poles are not blue, but that might have a blue cut-vertex, and let $n''_{b,r,k}(n) = s''_{b,r,k}(n) + p'_{b,r,k}(n) + h'_{b,r,k}(n)$. We also introduce the corresponding functions $s'_{b,r}(n)$, $s''_{b,r}(n)$, $p'_{b,r}(n)$, and $h'_{b,r}(n)$ for the case that we do not require that the first colored vertex has a distance k to the s-pole. The functions defined here might take non-zero values only if $n \geq 3$ and $b, r, k \geq 0$ and $b + r \geq 1$.

Colored s-networks. We first compute the number s' of colored s-networks S with no blue poles and no blue cut-vertex. A colored s-network might have a red pole s. If this is the case (for $k = 0$), then there is a unique neighbor u which together with s forms a cut-edge. We remove s and the cut-edge su from S and take u as a new pole to obtain a new colored network with non-blue poles u, t and no blue cut-vertex. Such networks are counted by $n'_{b,r-1}(n-1)$ (compare for the left arrow in Fig. 10). If s is not red (for $k \geq 1$), then we consider the cut-edge uv in S which is closest to s (compare for the right arrow in Fig. 10). Due to our assumption, v cannot be blue. The vertex v might equal t, in which case su is the root of a p- or an h-network. The cut-edge uv induces a split colored p- or h-network with non-blue poles s, u (counted by $p'_{b',r',k} + h'_{b',r',k}$), and a remaining part with non-blue poles v, t, which is again an arbitrary colored network that has no blue cut-vertex (counted by $n'_{b,r}$), or a red vertex when $v = t$. This explains the first formula for $s'_{b,r,k}(n)$.

Analogously we can also compute s''. The split colored network induced by the cut-edge uv is again counted by $p'_{b',r',k} + h'_{b',r',k}$. But we have to consider the case that v might be

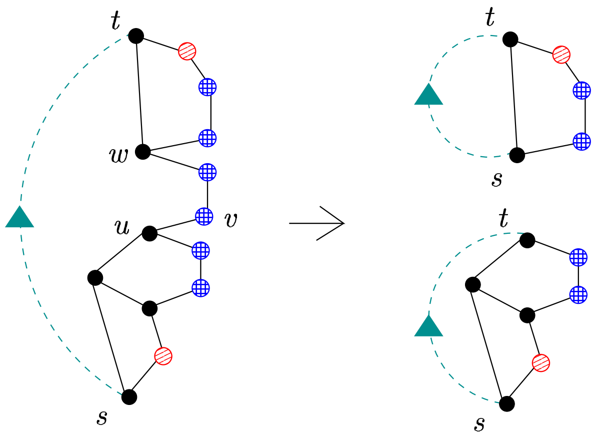

Fig. 11. Colored s-networks

blue. If v is blue, it is adjacent to another blue vertex, which in turn has another unique non-blue neighbor w in S, by 3-regularity (see Fig. 11). The colored network with the non-blue poles w and t should not contain a blue cut-vertex to satisfy (P4), and therefore those networks are counted by $n'_{b,r}$. If v is not blue, v and t form another colored network that might have a blue cut-vertex (counted by $n''_{b,r}$), or a red vertex if $v = t$. This explains the formula for s'' below. Since every subnetwork contains a colored vertex by (P3) the parameter k is only relevant for the subnetwork closest to s.

$$\begin{aligned}
s'_{b,r,0}(n) &= n'_{b,r-1}(n-1) \\
s'_{b,r,k}(n) &= p'_{b,r-1,k}(n-1) + h'_{b,r-1,k}(n-1) \\
&\quad + \sum_{i \geq 3, b', r'} \left(p'_{b',r',k}(i) + h'_{b',r',k}(i)\right) n'_{b-b',r-r'}(n-i) \\
s''_{b,r,0}(n) &= n''_{b,r-1}(n-1) \\
s''_{b,r,k}(n) &= p'_{b,r-,k}(n-1) + h'_{b,r-1,k}(n-1) \\
&\quad + \sum_{i \geq 3, b', r'} \left(p'_{b',r',k}(i) + h'_{b',r',k}(i)\right) \left(n'_{b-b'-2,r-r'}(n-i-2) + n''_{b-b',r-r'}(n-i)\right).
\end{aligned}$$

Colored *p*-networks. The colored vertices of a colored p-network must all lie in one of its two parts, by property (P2). Since every subnetwork has to contain at least one colored vertex (P3), the other part of the p-network must be a single edge. Moreover the poles s, t are not colored, and thus $p'_{b,r,0}(n) = 0$. One (but not both) of the unique neighbors of s and t in this subnetwork might be blue (P4). If both neighbors u and v are not blue, we remove the edges st, su, and vt to obtain a new colored network with poles u and v, which might have a blue cut-vertex; see the left hand side of Fig. 12. The number of networks in that case is $n''_{b,r,k-1}$.

If there is a blue neighbor u, either for s or for t, then u has a unique blue neighbor w, which in turn has a unique neighbor w'. We remove the edges st, su, vt, uw, and

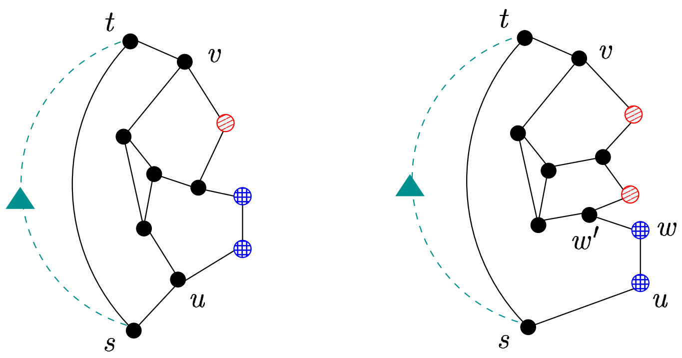

Fig. 12. Colored p-networks

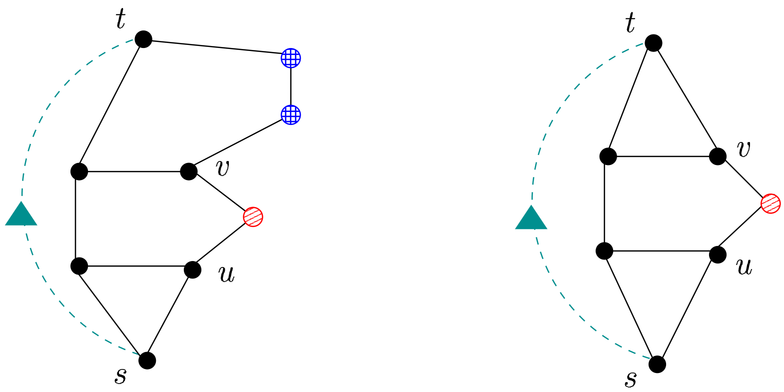

Fig. 13. Colored h-networks

ww' to obtain a new colored network with poles w', v. These networks cannot have a blue cut-vertex and are therefore counted by $n'_{b-2,r,k-3}$; see the right hand side of Fig. 12.

$$p'_{b,r,k}(n) \; = \; n''_{b,r,k-1}(n-2) \; + \; 2n'_{b-2,r,k-3}(n-4)\,.$$

Colored *h*-networks. The core of a colored h-network H has a unique embedding into the plane where the root edge and the core edges that are replaced by colored networks lie on the outer face. Let $h'_{b,r,k,l}(n)$ be the number of colored h-networks such that the first colored vertex has distance k to the s-pole and the l closest core edges (on the outer face) to vertex s are also edges in H. Thus $h'_{b,r,k}(n) = \sum_{l \le k-1} h'_{b,r,k,l}(n)$. We decrease l by one in each step of the decomposition.

Given a colored h-network such that the first colored vertex has distance k to the s-pole and the l closest core edges (on the outer face) to vertex s are also edges in H. We look at the l+1st closest edge uv on the outer face of the core. It might also be an edge in H. If not, uv is a split pair in H and determines a subnetwork S (see Fig. 13, left). We split off S from the h-network. If we insert an edge between u and v in S, it becomes a colored p-network. The remaining graph after splitting off S is again an h-network. It might be the case that all colored vertices lie in S (see Fig. 13, right). Then the remaining graph after splitting off S is a 3-connected network with at least $l+1$ vertices on the outer face (which is counted by $\sum_{k'\geq l+1} t_{k',3}$ using its dual, see Section 9).

$$\begin{aligned} h'_{b,r,k,l}(n) &= \sum_{i\geq 1} p'_{b,r,k-l-1}(i{+}2) \sum_{k'\geq l+1} t_{k',3}((n{-}i)/2{+}2) \\ &\quad + \sum_{b',r',i\geq 0} p'_{b',r',k-l-1}(i{+}2) \sum_{k'>l+1} h'_{b-b',r-r',k',l+1}(n{-}i)\,. \end{aligned}$$

Summary. We give a summary of the formulas for the decomposition of colored networks for the simpler case where we do not specify the distance k of the first colored vertex to the s-pole.

$$\begin{aligned} s'_{b,r}(n) &= n'_{b,r-1}(n{-}1) \;+\; p'_{b,r-1}(n{-}) + h'_{b,r-1}(n{-}1) \\ &\quad + \sum_{i\geq 3,b',r'} \left(p'_{b',r'}(i) \;+\; h'_{b',r'}(i)\right) n'_{b-b',r-r'}(n{-}i) \\ s''_{b,r}(n) &= n''_{b,r-1}(n{-}1) \;+\; p'_{b,r-1}(n{-}1) \;+\; h'_{b,r-1}(n{-}1) \\ &\quad + \sum_{i\geq 3,b',r'} \left(p'_{b',r'}(i) + h'_{b',r'}(i)\right)\left(n'_{b-b'-2,r-r'}(n{-}i{-}2) + n''_{b-b',r-r'}(n{-}i)\right) \\ p'_{b,r}(n) &= n''_{b,r}(n{-}2) \;+\; 2n'_{b-2,r}(n{-}4) \\ h'_{b,r}(n) &= h'_{b,r,0}(n) \\ h'_{b,r,l}(n) &= \sum_{i\geq 1} p'_{b,r}(i{+}2) \sum_{k'\geq l+1} t_{k',3}((n{-}i)/2{+}2) \;+\; \sum_{b',r',i\geq 0} p'_{b',r'}(i{+}2) h'_{b-b',r-r',l+1}(n{-}i) \end{aligned}$$

6 Pole-symmetric Graphs

At two points in Section 3 we already needed to count *pole-symmetric* cubic planar graphs. The first point we met them was at g-graphs, where we needed an orientation of the two vertices adjacent to the poles for further decomposition. The second point was that the edges in the core of an h-network fixed by a sense-reversing automorphism must be replaced by pole-symmetric rooted graphs.

In this section we consider the decomposition of rooted cubic graphs with such a pole symmetry and derive recurrence formulas to count them. Let $\tilde{r}(n)$ be the number of pole-symmetric rooted cubic graphs on n vertices, where the poles are distinct. According to Theorem 1 we have $\tilde{r}(n) = \tilde{d}(n) + \tilde{s}(n) + \tilde{p}(n) + \tilde{h}(n)$, where these functions count the number of d-, s-, p-, and h-graphs with a pole symmetry on n vertices, respectively. Note that in $\tilde{r}(n)$ we did not count the rooted graphs where the poles are not distinct. By definition, a g-graph is pole-symmetric, $\tilde{g}(n) = g(n)$. Finally, let $\tilde{c}(n) = \tilde{s}(n) + \tilde{p}(n) + \tilde{h}(n) + \tilde{g}(n)$, and call the corresponding graphs pole-symmetric c-graphs.

If a d-graph G has a pole symmetry, then G without the root edge has two identical components. This component together with the old root edge st forms a g-graph rooted at the self-loop ss. Thus there is a bijection between g-graphs on $n/2+1$ vertices and pole-symmetric d-graphs on n vertices: $\tilde{d}(n) = g(n/2+1)$.

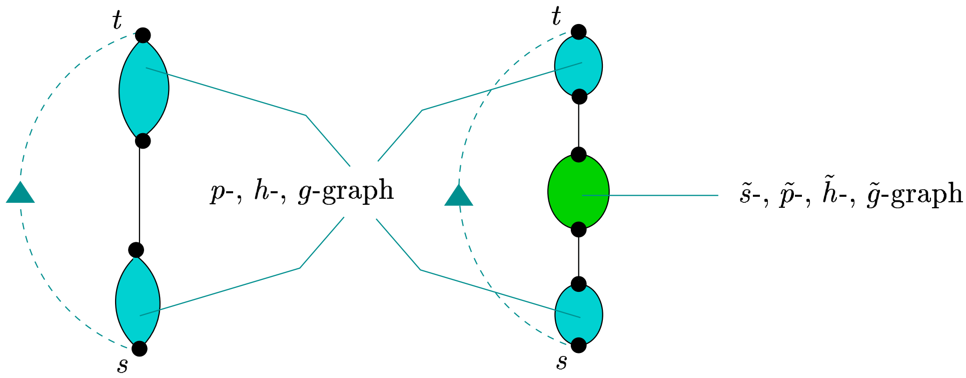

Fig. 14. Pole symmetric s-graphs

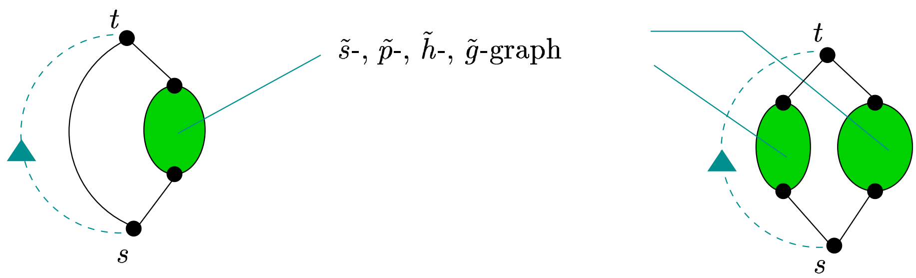

Fig. 15. Pole symmetric p-graphs

Pole-symmetric s-graphs. Here we split off a p-, h- or g-graph at both poles simultaneously. The remaining graph is either a pole-symmetric c-graph, or an edge (see Fig. 14). For $0 \le n \le 7$, $s(n) = 0$. For $n \ge 8$,

$$\tilde{s}(n) \;=\; \sum_{i=4}^{(n-4)/2} (c(i) - s(i))\tilde{c}(n-2i) \;+\; p(n/2) \;+\; h(n/2)\,.$$

Pole-symmetric p-graphs. The root edge st induces two components. One of them might be a single edge (see Fig. 15, left part). Otherwise the two components, rooted at the corresponding neighbors of s and t, are pole-symmetric c-graphs (see Fig. 15, right part). The two components are not ordered, and thus we count every pair of c-graphs twice, except pairs having the same pole-symmetric c-graph. This explains the formula: for $0 \le n \le 5$, $p(n) = 0$, and for $n \ge 6$,

$$\tilde{p}(n) \;=\; \tilde{c}(n-2) \;+\; \frac{1}{2}\Big(\sum_{i=4}^{(n-4)/2} \tilde{c}(i)\tilde{c}(n-i-2) \;+\; \tilde{c}((n-2)/2)\Big)\,.$$

Pole-symmetric h-graphs. Let ψ be a pole-exchanging automorphism of an h-graph $\tilde{H}$ rooted at st, i.e. $\psi(s) = t$ and $\psi(t) = s$. As in the paragraph on h-graphs in Section 3 we need to control the number of such h-graphs that also have a sense-reversing automorphism

φ with $\varphi(s) = s$ and $\varphi(t) = t$. In the case where we do not have a sense-reversing automorphism we order the edges of the core of H in such a way that corresponding edges with respect to the pole-symmetry are consecutive. Blue edges come first, which are followed by red edges, and then the uncolored edges. This can be done uniquely.

In the case where we have a sense-reversing automorphism φ, we order the edges of the core in such a way that we start with the blue edges with respect to φ and then the blue edges with respect to ψ. Next we list the red edges with respect to φ and then the red edges with respect to ψ. Finally we list corresponding edges with respect to φ, followed by their corresponding edges with respect to ψ, consecutively in groups of four.

Let $\tilde{b}_{b,\tilde{b},r,\tilde{r},l}(n)$ be the number of pole-symmetric h-graphs $\tilde{B}$ with a sense-reversing automorphism φ on n vertices, where $\text{core}(\tilde{B})$ has $b, \tilde{b}$ blue and $r, \tilde{r}$ red edges with respect to φ and ψ, respectively, and the first l edges of $\text{core}(\tilde{B})$ are also edges in $\tilde{B}$. Analogously we introduce $\tilde{a}_{\tilde{b},\tilde{r},l}(n)$ for the number of pole-symmetric h-graphs $\tilde{A}$ without a sense-reversing automorphism.

To produce $\tilde{B}$, we take an h-graph with both φ and ψ. The $l{+}1$-st edge of $\text{core}(\tilde{B})$ might be an edge of $\tilde{B}$. If not, we could replace corresponding blue (with respect to φ and ψ) edges $l{+}1$ and $l{+}2$ by an arbitrary c-graph (for $l{+}1 \leq b{+}\tilde{b}$), or we could replace corresponding red (with respect to φ and ψ) edges $l{+}1$ and $l{+}2$ by a pole-symmetric c-graph (for $b{+}\tilde{b} < l{+}1 \leq b{+}\tilde{b}{+}r{+}\tilde{r}$). Finally we could replace corresponding uncolored edges $l{+}1, \ldots, l{+}4$ by the same c-graph (for $l{+}1 > b{+}\tilde{b}{+}r{+}\tilde{r}$). The initial case is that all edges of $\text{core}(\tilde{B})$ are also edges of $\tilde{B}$, where we have a 3-connected pole-symmetric graph with a sense-reversing automorphism (see Section 8).

$$\tilde{b}_{b,\tilde{b},r,\tilde{r},l}(n) \;=\; \begin{cases} \sum_{i\geq 0} c(i)\tilde{b}_{b,\tilde{b},r,\tilde{r},l+2}(n{-}2i)\,, & l{+}1 \leq b{+}\tilde{b} \\ \sum_{i\geq 0} \tilde{c}(i)\tilde{b}_{b,\tilde{b},r,\tilde{r},l+2}(n{-}2i)\,, & b{+}\tilde{b} < l{+}1 \leq b{+}\tilde{b} + {+}r{+}\tilde{r} \\ \sum_{i\geq 0} c(i)\tilde{b}_{b,\tilde{b},r,\tilde{r},l+4}(n{-}4i)\,, & l{+}1 > b{+}\tilde{b}{+}r{+}\tilde{r}\,. \end{cases}$$

To produce $\tilde{A}$, we could take either a pole-symmetric h-graph with no sense-reversing automorphism or an h-graph with both a pole-exchanging and a sense-reversing automorphism. In the first case we might replace the $l{+}1$st edge of $\text{core}(\tilde{A})$ by a c-graph. In the second case, we could take an h-graph with both a pole-exchanging automorphism ψ and a sense-reversing automorphism φ, whose core has additional b blue and r red edges with respect to φ. Then we might replace two corresponding blue (with respect to ψ) edges $l{+}1$ and $l{+}2$ edges by two different c-graphs (for $b < l{+}1 \leq b{+}\tilde{b}$), or two corresponding red edges with respect to φ by a c-graph without a pole-symmetry (for $b{+}\tilde{b} < l{+}1 \leq b{+}\tilde{b}{+}r$), and then red edges with respect to ψ by two different pole-symmetric c-graphs (for $b{+}\tilde{b}{+}r < l{+}1 \leq b{+}\tilde{b}{+}r{+}\tilde{r}$). If there is no red nor blue edge left (for $l{+}1 > b{+}\tilde{b}{+}r{+}\tilde{r}$), we could replace the $l{+}1$st and $l{+}3$-rd core edges with a different c-graph than the $l{+}2$-nd and $l{+}4$-th core edge. The initial case is that all edges of $\text{core}(\tilde{A})$ are also edges of $\tilde{A}$, which is counted by 3-connected pole-symmetric graphs with no sense-reversing automorphism (see Section 8).

$$\tilde{a}_{\tilde{b},\tilde{r},l}(n) \;=\; \sum_{i\geq 0} c(i)\tilde{a}_{\tilde{b},\tilde{r},l+1}(n{-}i) \;+$$

$$\begin{cases} \sum_{b,r\geq 1,i,j\geq 0} c(i)c(j)\tilde{b}_{b,\tilde{b},r,\tilde{r},l+2}(n{-}i{-}j) - c(i)\tilde{b}_{b,\tilde{b},r,\tilde{r},l+2}(n{-}2i)\,, & b < l{+}1 \leq b{+}\tilde{b} \\ \sum_{b,r\geq 1,i\geq 0}(c(i) - \tilde{c}(i))\tilde{b}_{b,\tilde{b},r,\tilde{r},l+2}(n{-}2i)\,, & b{+}\tilde{b} < l{+}1 \leq b{+}\tilde{b}{+}r \\ \sum_{b,r\geq 1,i,j\geq 0} \tilde{c}(i)\tilde{c}(j)\tilde{b}_{b,\tilde{b},r,\tilde{r},l+2}(n{-}i{-}j) - \tilde{c}(i)\tilde{b}_{b,\tilde{b},r,\tilde{r},l+2}(n{-}2i)\,, & b{+}\tilde{b}{+}r < l{+}1 \leq b{+}\tilde{b}{+}r{+}\tilde{r} \\ \sum_{b,r\geq 1,i,j\geq 0} c(i)c(j)\tilde{b}_{b,\tilde{b},r,\tilde{r},l+4}(n{-}2i{-}2j) - c(i)\tilde{b}_{b,\tilde{b},r,\tilde{r},l+4}(n{-}4i)\,, & l{+}1 > b{+}\tilde{b}{+}r{+}\tilde{r}\,. \end{cases}$$

With these numbers we can compute $\tilde{h}(n) = \sum_{b,\tilde{b},\tilde{r},r\geq 1} \tilde{a}_{\tilde{b},\tilde{r},0}(n) + \tilde{b}_{b,\tilde{b},r,\tilde{r},0}(n)$.

7 Pole-symmetric 3-connected Graphs

To count the number of pole-symmetric 3-connected graphs we again want to use colored networks. This time the colors correspond to the vertices and edges that are fixed by the pole-exchanging automorphism.

Theorem 3. *There is a bijection between the following two sets of objects:*

- *Colored networks with b blue, r red, and $(n{+}b)/2{-}1$ vertices in total. Moreover between s and the first colored vertex there are $\lceil l/2\rceil - 2$ edges on the outer face.*
- *Face-rooted 3-connected graphs on n vertices with a pole-exchanging automorphism fixing b vertices and exchanging r pairs of adjacent vertices. Moreover there are l vertices on the outer face.*

Proof. Remove all vertices that are fixed - the *blue* vertices - and edges that are exchanged by the pole-exchanging automorphism - the *red* vertices. Call the two graphs induced by the vertices in the two resulting connected components together with the blue vertices H_1 and H_2. Consider the pair uv of the two neighbors of the old pole s in H_1. The pair uv is uniquely ordered because of the face-rooting, and u and v are the poles of the colored network H_1 where s is deleted. Conversely, given a colored network we can construct the corresponding face-rooted pole-symmetric 3-connected network by reversing the above decomposition. □

Hence $2\tilde{a}_{\tilde{b},\tilde{r},3n/2-1}(n) + \sum_{b,r} \tilde{b}_{b,\tilde{b},r,\tilde{r},3n/2-1}(n) = \sum_{k\geq 0} n'_{\tilde{b},\tilde{r},k}((n+\tilde{b})/2-1)$.

Next we have to compute the number of pole-symmetric 3-connected graphs with a sense-reversing automorphism. Again we use colored networks, but impose the additional constraint that the colored network has a pole-exchanging automorphism. Along the lines of Theorem 2 we have a bijection between these pole-symmetric colored networks and pole-symmetric networks with a sense-reversing automorphism.

8 Pole-symmetric Colored Networks

We write $\tilde{n}'(n)$ for the number of pole-symmetric colored networks that have no blue cut-vertex, which again can be computed via $\tilde{n}'(n) = \tilde{s}'(n) + \tilde{p}'(n) + \tilde{h}'(n)$, as the sum of the numbers for the corresponding s-, p-, and h-networks. Let $\tilde{s}'_{b,r}(n), \tilde{p}'_{b,r}(n), \tilde{h}'_{b,r}(n)$ denote the number of pole-symmetric colored s-, p-, and h-networks, with b blue, r red, and n vertices in total, and let $\tilde{n}'_{b,r}(n) = \tilde{s}'_{b,r}(n) + \tilde{p}'_{b,r}(n) + \tilde{h}'_{b,r}(n)$. Moreover we use functions $\tilde{n}'_{b,r,\tilde{r}}(n), \tilde{s}'_{b,r,\tilde{r}}(n), \tilde{p}'_{b,r,\tilde{r}}(n), \tilde{h}'_{b,r,\tilde{r}}(n)$ to specify that corresponding networks have the pole-exchanging automorphisms exchanging $\tilde{r}$ pairs of adjacent vertices. Due to Theorem 2 we have $\tilde{b}_{r,\tilde{r},3n/2-1}(n) = \sum_b \tilde{n}'_{b,r,\lceil\tilde{r}/2\rceil}((n+b)/2-2)$. The decomposition of pole-symmetric colored networks is a straightforward combination of the ideas in Section 6 and 5.

Pole-symmetric colored *s*-networks. Given a pole-symmetric colored s-network, we split off a p-, or h-graph at both poles simultaneously and count the remaining graphs, which are pole-symmetric colored s-, p-, or h-graphs, or an edge, or a red vertex.

$$\begin{aligned}\tilde{s}'_{b,r}(n) \;=\; &\sum_{i\geq 4,b',r'} (p'_{b',r'}(i) + h'_{b',r'}(i))\tilde{n}'_{b-2b',r-2r'}(n-2i) \\ &+\; p'_{b/2,r/2}(n/2) \;+\; h'_{b/2,r/2}(n/2) \;+\; p'_{b/2,(r-1)/2}((n-1)/2) \;+\; h'_{b/2,(r-1)/2}((n-1)/2)\,.\end{aligned}$$

Pole-symmetric colored *p*-networks. One of the two components induced by the poles of a colored p-network is a single edge. Because of the pole-symmetry and property (P4), none of the roots can be blue.

$$\tilde{p}'_{b,r}(n) \;=\; \tilde{n}'_{b,r}(n-2)\,.$$

Pole-symmetric colored *h*-networks. Let $\tilde{h}'_{b,r,l}(n)$ be the number of colored h-networks such that the l closest core edges (on the outer face) to vertex s are also edges in H.

$$\tilde{h}'_{b,r,l}(n) \;=\; \sum_{i\geq 0}\tilde{p}'_{b,r}(i+2)\sum_{k\geq l+1}\tilde{t}_k(n-2i) \;+\; \sum_{b',r',i\geq 0} p'_{b',r'}(i+2)h'_{b-2b',r-2r',l+1}(n-2i)\,.$$

When we have removed the last colored subnetwork in a pole-symmetric colored h-network, we have to count the number of face-rooted pole-symmetric networks with l vertices on the outer face, which is $\tilde{t}_l(n)$. Similar recurrences hold for $\tilde{s}'_{b,r,\tilde{r}}(n), \tilde{p}'_{b,r,\tilde{r}}(n), \tilde{h}'_{b,r,\tilde{r},l}(n)$ except that we use $\tilde{t}_{l,\tilde{r}}(n)$ in $\tilde{h}'_{b,r,\tilde{r},l}(n)$ instead of $\tilde{t}_l(n)$. To count face-rooted pole-symmetric 3-connected graphs we again use Theorem 3 and obtain $\tilde{t}_l(n) = \sum_r \tilde{t}_{l,r}(n)$ and $\tilde{t}_{l,r}(n) = \sum_b \sum_{k\geq\lceil l/2\rceil} n'_{b,r,k}((n+b)/2-1)$.

9 Cubic Planar Maps and Triangulations

The dual of a 3-connected cubic planar graph is a triangulation, i.e., a graph where we cannot add another edge without destroying planarity. If the 3-connected cubic planar graph is face-rooted, the triangulation is as well face-rooted, and vice versa: the root-face

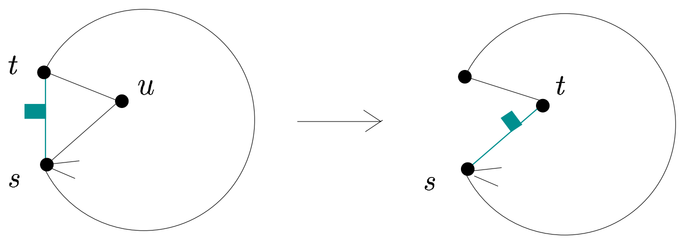

Fig. 16. Decomposition of a rooted near-triangulation

(incident to the root-edge) becomes the s-pole and the other face incident to the root-edge becomes the t-pole in the dual. In our drawings, the root-face incident to the root will always be the outer face. As already mentioned, for 3-connected face-rooted graphs there is only one such embedding.

To derive a recursion, we generalize the notion of a triangulation, as Tutte did [20]: We consider face-rooted 3-connected planar graphs where all the faces except the outer face are triangles, i.e., we do not require that the outer face is a triangle, but still assume that the graph does not contain a 2-cut. Then we distinguish between external and internal vertices and edges, where the external vertices and edges are defined to be the vertices and edges on the outer face. We call such objects *near-triangulations*. By 3-connectivity, in a near-triangulation there is no internal edge connecting two external vertices.

If the 3-connected face-rooted cubic planar graph has k vertices on the outer face, the s-pole of the rooted triangulation has degree k. To count the number of such graphs, we use the function $t_{k,l}(n)$ which denotes the number of rooted near-triangulations with n vertices, where the s-pole has degree k and there are l vertices on the outer face. When the unique internal vertex adjacent to the two poles, say u, has no internal edge connecting it to an external vertex except the poles, we remove the pole edge and move the t-pole to the unique internal vertex adjacent to the two former poles (see Figure 16).

Otherwise we remove the edge between the poles and decompose such triangulations along the edge, say uv, connecting to the first such external vertex, say v, according to a traversal of the outer face starting from the s-pole ending at the t-pole. Then one of the two split triangulations has the new t-pole at the vertex u, and the other one has the new s-pole at v (see Figure 17) except that it has the new s-pole at u when the number of edges on the outer face is 3 (see Figure 18). All these cases can be computed inductively using the value of $t_{k,l}(n)$ for lexicographically smaller arguments.

Initially, $t_{2,3}(3) = 1$ and $t_{k,l}(n) = 0$ if $k = 2$ and $l > 3$ or $n > 3$, or if $l+k-2 > n$. Otherwise

$$t_{k,l}(n) = t_{k-1,l+1}(n) + \sum_{k'+1,i\geq 3} t_{k-1,l}(i)t_{k',3}(n-i+2) + \sum_{k',l',i\geq 3} t_{k-1,l-l'+2}(i)t_{k',l'}(n-i+2) .$$

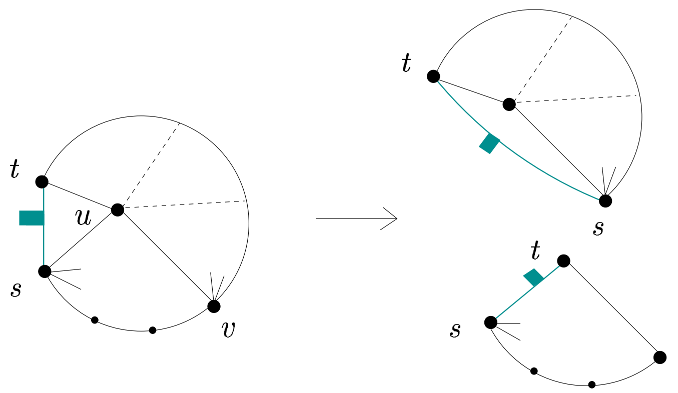

Fig. 17. Decomposition of a rooted near-triangulation

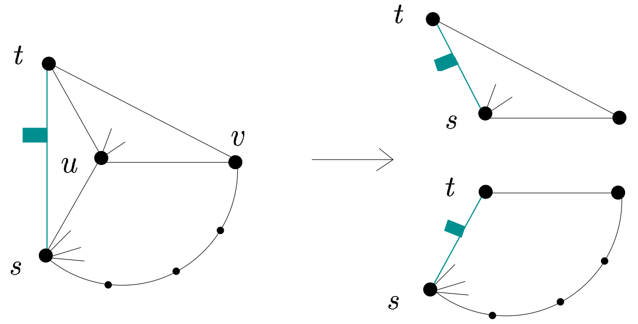

Fig. 18. Decomposition of a rooted near-triangulation

10 Conclusion

We presented a decomposition strategy for unlabeled connected cubic planar graphs along the connectivity structure. In order to count these objects we need a *unique* decomposition, and thus we used the well-known concept of an (edge-)root. For 3-connected rooted cubic planar graphs, however, we also had to control whether or not there is a sense-reversing automorphism, which in turn required to control networks that have a pole-exchanging automorphism, and finally networks with both a sense-reversing and a pole-exchanging automorphism. To count these objects we introduced the concept of *colored* networks, and proved several bijections. The decomposition together with the counting formulas immediately gives a polynomial time generation procedure for rooted connected cubic planar graphs. Using rejection-sampling we obtain our result:

Theorem 4. *There is an algorithm that generates an unlabeled connected cubic planar graph on n vertices uniformly at random, and whose expected running time is in $\tilde{O}(n^{10})$. If we allow for a preprocessing, the algorithm can generate such an object in $O(n^3)$.*

Proof. The algorithm first generates a rooted unlabeled cubic planar graph G with n vertices, using the recursive decomposition along the connectivity structure (Section 2-Section 9), and the values of the counting formulas that can be computed efficiently using dynamic programming. Note that the representation size of all the values in this paper is linear, since the logarithm of the number of *unlabeled* planar graphs is linear.

The longest computation time is due to the four-dimensional tables in Section 6, where the summation runs over at most four parameters, and where we have to perform multiplications with large numbers. Assuming an $O(n \log n \log \log n)$ multiplication algorithm (see e.g. [7]), the number of computation steps needed to fill the four-dimensional table is within $\tilde{O}(n^9)$, where $\tilde{O}(\cdot)$ denotes growth up to logarithmic factors.

If we do not charge for the costs of a precomputation step, the actual generation of a rooted cubic planar graph can be done in quadratic time: The decomposition tree is of linear size; computing the probabilistic decisions at each branch also takes linear time, if we assume that we have access to the values in the table and the partial sums of the formulae.

To obtain *unrooted* cubic planar graphs, the algorithm computes the size o of the orbit of the root in the automorphism group of the graph G, which can be done in linear time using e.g. well-known correspondingly adapted graph isomorphism algorithms for planar graphs [12], and outputs the graph G with probability $1/o$. Since the size o of the orbit of the root is at most linear, the expected number of restarts is also linear. Thus the overall expected running time is in $\tilde{O}(n^{10})$, and in $O(n^3)$ with precomputation. □

It is easy to see that we can count and generate random cubic planar *multi-graphs* by only changing the initial values. Another by-product is the enumeration and uniform generation of rooted near-triangulations with a given number of vertices on the outer face, and a given number of vertices on the outer face of the dual. For the special case of rooted triangulations a linear-time generation procedure was known [19]. The present paper also contains a series of new enumeration results and generation procedures of rooted cubic graphs with various sorts of symmetries and degrees of connectivity.

Many techniques presented in this paper also work for planar graphs. However, it is more difficult to count not necessarily cubic g-graphs, since there is no canonical way to decompose them into smaller rooted graphs. Another complication would be that we cannot decompose colored networks via networks where both poles are non-blue. The recurrences derived in this paper could serve as a starting point to compute the generating functions and the asymptotic behavior of the counting functions for the objects considered here.

References

1. A. Bender and Z. Gao. Enumeration of rooted cubic planar maps. *Annals of Combinatorics*, 6:313–325, 2002.
2. A. Bender, Z. Gao, and N. Wormald. The number of labeled 2-connected planar graphs. *Electronic Journal of Combinatorics*, 9(43), 2002.
3. M. Bodirsky, C. Gröpl, and M. Kang. Generating labeled planar graphs uniformly at random. In *Proceedings of Thirtieth International Colloquium on Automata, Languages and Programming (ICALP'03)*, pages 1095–1107, 2003.
4. N. Bonichon, C. Gavoille, and N. Hanusse. An information-theoretic upper bound of planar graphs using triangulation. In *20th Annual Symposium on Theoretical Aspects of Computer Science (STACS'03)*, pages 499–510, 2003.
5. G. Brinkmann and B. D. McKay. Fast generation of non-isomorphic planar cubic graphs. In preparation.
6. W. G. Brown. Enumeration of triangulations of the disk. *Proc. London Math. Soc.*, 14:746–768, 1964.

7. P. Bürgisser, M. Clausen, and M. Shokrollahi. *Algebraic Complexity Theory.* Number 315 in Grundlehren der mathematischen Wissenschaften. Springer Verlag, 1997.
8. A. Denise, M. Vasconcellos, and D. Welsh. The random planar graph. *Congressus Numerantium*, 113:61–79, 1996.
9. R. Diestel. *Graph Theory.* Springer–Verlag, New York, 1997.
10. P. Eades, S.-H. Hong, and B. McKay. Symmetric drawings of triconnected planar graphs. In *Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 356 – 365, 2002.
11. P. Flajolet, P. Zimmerman, and B. Van Cutsem. A calculus for the random generation of labelled combinatorial structures. *Theoretical Computer Science*, 132(1-2):1–35, 1994.
12. J. E. Hopcroft and J. K. Wong. Linear time algorithm for isomorphism of planar graphs. *Annual ACM Symposium on Theory of Computing*, pages 172–184, 1974.
13. M. Jerrum. Mathematical foundations of the Markov chain Monte Carlo method. In *Probabilistic Methods for Algorithmic Discrete Mathematics (M. Habib, C. McDiarmid, J. Ramirez-Alfonsin and B. Reed eds.)*, pages 116–165, Springer–Verlag, 1998.
14. V. A. Liskovets and T. Walsh. Ten steps to counting planar graphs. In *Eighteenth Southeastern International Conference on Combinatorics, Graph Theory, and Computing, Congr. Numer.*, volume 60, pages 269–277, 1987.
15. L. McShine and P. Tetali. On the mixing time of the triangulation walk and other Catalan structures. *DIMACS Series in Discrete Math. Theor. Comput. Sci.*, 43:147–159, 1999.
16. M. Molloy, B. Reed, and W. Steiger. On the mixing rate of the triangulation walk. *DIMACS Series in Discrete Math. Theor. Comput. Sci.*, 43:179–190, 1999.
17. A. Nijenhuis and H. Wilf. *Combinatorial algorithms.* Academic Press Inc., 1979.
18. D. Osthus, H. J. Prömel, and A. Taraz. On random planar graphs, the number of planar graphs and their triangulations. *J. Combinatorial Theory, Series B*, pages 119–143, 2003.
19. D. Poulalhon and G. Schaeffer. Optimal coding and sampling of triangulations. Thirtieth International Colloquium on Automata, Languages and Programming, 2003.
20. W. T. Tutte. A census of planar triangulations. *Canad. J. Math.*, 14:21–38, 1962.
21. W. T. Tutte. On the enumeration of convex polyhedra. *J. Combin. Th. Ser. B*, 28:105–126, 1980.
22. W. T. Tutte. *Graph Theory.* Cambridge University Press, 1984.
23. T. Walsh. Counting unlabelled three-connected and homeomorphically irreducible two-connected graphs. *J. Combin. Theory*, 32:12–32, 1982.