# A Complete Grammar for Decomposing a Family of Graphs into 3-connected Components

Guillaume Chapuy[1], Éric Fusy[2], Mihyun Kang[3] and Bilyana Shoilekova[4]

Submitted: Sep 17, 2008; Accepted: Nov 30, 2008; Published: Dec 9, 2008
Mathematics Subject Classification: 05A15

**Abstract**

Tutte has described in the book "Connectivity in graphs" a canonical decomposition of any graph into 3-connected components. In this article we translate (using the language of symbolic combinatorics) Tutte's decomposition into a *general grammar* expressing any family $\mathcal{G}$ of graphs (with some stability conditions) in terms of the subfamily $\mathcal{G}_3$ of graphs in $\mathcal{G}$ that are 3-connected (until now, such a general grammar was only known for the decomposition into 2-connected components). As a byproduct, our grammar yields an explicit system of equations to express the series counting a (labelled) family of graphs in terms of the series counting the subfamily of 3-connected graphs. A key ingredient we use is an extension of the so-called dissymmetry theorem, which yields negative signs in the grammar and associated equation system, but has the considerable advantage of avoiding the difficult *integration steps* that appear with other approaches, in particular in recent work by Giménez and Noy on counting planar graphs.

As a main application we recover *in a purely combinatorial way* the analytic expression found by Giménez and Noy for the series counting labelled planar graphs (such an expression is crucial to do asymptotic enumeration and to obtain limit laws of various parameters on random planar graphs). Besides the grammar, an important ingredient of our method is a recent bijective construction of planar maps by Bouttier, Di Francesco and Guitter.

Finally, our grammar applies also to the case of *unlabelled* structures, since the dissymetry theorem takes symmetries into account. Even if there are still difficulties in counting unlabelled 3-connected planar graphs, we think that our grammar is a promising tool toward the asymptotic enumeration of unlabelled planar graphs, since it circumvents some difficult integral calculations.

[1]: LIX, École Polytechnique, Paris, France. chapuy@lix.polytechnique.fr
[2]: Dept. Mathematics, UBC, Vancouver, Canada. fusy@lix.polytechnique.fr
[3]: Institut für Informatik, Humboldt-Universität zu Berlin, Germany. kang@math.tu-berlin.de
[4]: Department of Statistics, University of Oxford, UK. shoileko@stats.ox.ac.uk

# 1 Introduction

Planar graphs and related families of structures have recently received a lot of attention both from a probabilistic and an enumerative point of view [1, 6, 10, 15, 19]. While the probabilistic approach already yields significant qualitative results, the enumerative approach provides a complete solution regarding the asymptotic behaviour of many parameters on random planar graphs (limit law for the number of edges, connected components), as demonstrated by Giménez and Noy for planar graphs [15] building on earlier work of Bender, Gao, Wormald [1]. Subfamilies of labelled planar graphs have been treated in a similar way in [4, 6].

The main lines of the enumerative method date back to Tutte [27, 28], where graphs are decomposed into components of higher connectivity: A graph is decomposed into connected components, each of which is decomposed into 2-connected components, each of which is further decomposed into 3-connected components. For planar graphs every 3-connected graph has a unique embedding on the sphere, a result due to Whitney [31], hence the number of 3-connected planar *graphs* can be derived from the number of 3-connected planar *maps*. This already makes it possible to get a polynomial time method for exact counting (via recurrences that are derived for the counting coefficients) and uniform random sampling of labelled planar graphs, as described by Bodirsky et al [5]. This decomposition scheme can also be exploited to get asymptotic results: asymptotic enumeration, limit laws for various parameters. In that case, the study is more technical and relies on two main steps: symbolic and analytic. In the symbolic step, Tutte's decomposition is translated into an equation system satisfied by the counting series. In the analytic step, a careful analysis of the equation system makes it possible to locate and determine the nature of the (dominant) singularities of the counting series; from there, transfer theorems of singularity analysis, as presented in the forthcoming book by Flajolet and Sedgewick [9], yield the asymptotic results.

In this article we focus on the symbolic step: how to translate Tutte's decomposition into an equation system in an automatic way. Our goal is to use a formalism as general as possible, which works both in the labelled and in the unlabelled framework, and works for a generic family of graphs (however under a certain stability condition), not only planar graphs. Our output is a generic decomposition grammar—the grammar is shown in Figure 6—that corresponds to the translation of Tutte's decomposition. Getting such a grammar is however nontrivial, as Tutte's decomposition is rather involved; we exploit the *dissymmetry theorem* (Theorem 3.1) applied to trees that are naturally associated with the decomposition of a graph. Similar ideas were recently independently described by Gagarin et al in [13], where they express a species of 2-connected graphs in terms of the 3-connected subspecies. Translating the decomposition into a grammar as we do here is very transparent and makes it possible to easily get equation systems in an automatic way, both in the *labelled* case (with generating functions) and in the *unlabelled* case (with Pólya cycle index sums). Let us also mention that, when performing the symbolic step in [15], Giménez and Noy also translate Tutte's decomposition into a positive equation system, but they do it only *partially*, as some of the generating functions in the system they obtain have to be *integrated*; therefore they have to deal with complicated analytic integrations, see [15] and more recently [14] for a generalized presentation. In contrast, in the equation system derived from our grammar, no integration step is needed; and as expected, the only terminal series are those counting the 3-connected subfamilies (indeed, 3-connected graphs are the terminal bricks in Tutte's decomposition). In some way, the dissymmetry theorem used to write down the grammar allows us to do the integrations *combinatorially*.

In addition to the grammar, an important outcome of this paper is to show that the analytic (implicit) expression for the series counting labelled planar graphs can be found in a completely combinatorial way (using also some standard algebraic manipulations), thus providing an alternative more direct way compared to the method of Giménez and Noy, which requires integration steps. Thanks to our grammar, finding an analytic expression for the series counting planar graphs reduces to finding one for the series counting 3-connected planar graphs, which is equivalent to the series counting 3-connected maps by Whitney's theorem. Some difficulty occurs here, as only an expression for the series counting *rooted* 3-connected maps is accessible in a direct combinatorial way. So it seems that some integration step is needed here, and actually that integration was analytically solved by Giménez and Noy in [15]. In contrast we aim at finding an expression for the series counting *unrooted* 3-connected maps in a more direct combinatorial way. We show that it is possible, by starting from a bijective construction of vertex-pointed maps—due to Bouttier, Di Francesco, and Guitter [7]—and going down to vertex-pointed 3-connected maps; then Euler's relation makes it possible to obtain the series counting 3-connected maps from the series counting vertex-pointed and rooted ones. In some way, Euler's relation can be seen as a generalization of the dissymmetry theorem that applies to maps and allows us to integrate "combinatorially" a series of rooted maps.

Concerning unlabelled enumeration, we prefer to stay very brief in this article (the counting tools are cycle index sums, which are a convenient refinement of ordinary generating functions). Let us just mention that our grammar can be translated into a generic equation system relating the cycle index sum (more precisely, a certain refinement w.r.t. edges) of a family of graphs to the cycle index sum of the 3-connected subfamily. However such a system is very complicated. Indeed the relation between 3-connected and 2-connected graphs involves edge-substitutions, which are easily addressed by exponential generating functions for labelled enumeration (just substitute the variable counting edges) but are more intricate when it comes to unlabelled enumeration (the computation rule is a specific multivariate substitution). We refer the reader to the recent articles by Gagarin et al [12, 13] for more details. And we plan to investigate the unlabelled case in future work, in particular to recover (and possibly extend) in a unified framework the few available results on counting asymptotically unlabelled subfamilies of planar graphs [25, 3].

**Outline.** After the introduction, there are four preliminary sections to recall important results in view of writing down the grammar. Firstly we recall in Section 2 the principles of the symbolic method, which makes it possible to translate systematically combinatorial decompositions into enumeration results, using generating functions for labelled classes and ordinary generating functions (via cycle index sums) for unlabelled classes. In Section 3 we recall the dissymmetry theorem for trees and state an extension of the theorem to so-called tree-decomposable classes. In Section 4 we give an outline of the necessary graph theoretic concepts for the decomposition strategy. Then we recall the decomposition of connected graphs into 2-connected components and of 2-connected graphs into 3-connected components, following the description of Tutte [28]. We additionally give precise characterizations of the different trees resulting from the decompositions.

In the last three sections, we present our new results. In Section 5 we write down the grammar resulting from Tutte's decomposition, thereby making an extensive use of the dissymmetry theorem. The complete grammar is shown in Figure 6. In Section 6, we discuss applications to labelled enumeration; the grammar is translated into an equation system—shown in Figure 7—expressing a series counting a graph family in terms of the series counting the 3-connected subfamily. Finally, building on this and on enumeration techniques for maps, we explain in Section 7 how to get an (implicit) analytic expression for the series counting labelled planar graphs.

# 2 The symbolic method of enumeration

In this section we recall important concepts and results in symbolic combinatorics, which are presented in details in the book by Flajolet and Sedgewick [9] (with an emphasis on analytic methods and asymptotic enumeration) and the book by Bergeron, Labelle, and Leroux [2] (with an emphasis on unlabelled enumeration). The symbolic method is a theory for enumerating *decomposable* combinatorial classes in a systematic way. The idea is to find a recursive decomposition for a class $\mathcal{C}$, and to write this decomposition as a *grammar* involving a collection of *basic classes* and *combinatorial constructions*. The grammar in turn translates to a recursive equation-system satisfied by the associated generating function $C(x)$, which is a formal series whose coefficients are formed from the counting sequence of the class $\mathcal{C}$. From there, the counting coefficients of $\mathcal{C}$ can be extracted, either in the form of an estimate (asymptotic enumeration), or in the form of a counting process (exact enumeration).

## 2.1 Labelled/unlabelled structures

A *combinatorial class* $\mathcal{C}$ (also called a species of combinatorial structures) is a set of *labelled* objects equipped with a size function; each object of $\mathcal{C}$ is made of $n$ atoms (typically, vertices of graphs) assembled in a specific way, the atoms bearing distinct labels in $[1..n] := \{1, \ldots, n\}$ (in the general theory of species, any system of labels is allowed). The number of objects of each size $n$, denoted $\mathcal{C}_n$, is finite. The classes we consider are stable under isomorphism (two structures are called *isomorphic* if one is obtained from the other by relabelling the atoms). Therefore, the labels on the atoms only serve to distinguish them, which means that no notion of order is used for the labels. The class of objects in $\mathcal{C}$ taken up to isomorphism is called the *unlabelled class* of $\mathcal{C}$ and is denoted by $\widetilde{C} = \cup_n \widetilde{C}_n$.

## 2.2 Basic classes and combinatorial constructions

We introduce the basic classes and combinatorial constructions, as well as the rules to compute the associated counting series. The neutral class $\mathcal{E}$ is made of a single object of size 0. The atomic class $\mathcal{Z}$ is made of a single object of size 1. Further basic classes are the Seq-class, the Set-class, and the Cyc-class, each object of the class being a collection of $n$ atoms assembled respectively as an ordered sequence, an unordered set, and an oriented cycle.

Next we turn to the main constructions of the symbolic method. The *sum* $\mathcal{A}+\mathcal{B}$ of two classes $\mathcal{A}$ and $\mathcal{B}$ refers to the disjoint union of the classes. The *partitional product* (shortly product) $\mathcal{A} * \mathcal{B}$ of two classes $\mathcal{A}$ and $\mathcal{B}$ is the set of labelled objects that are obtained as follows: take a pair $(\gamma \in \mathcal{A}, \beta \in \mathcal{B})$, distribute distinct labels on the overall atom-set (i.e., if $\beta$ and $\gamma$ are of respective sizes $n_1, n_2$, then the set of labels that are distributed is $[1..(n_1+n_2)]$), and forget the original labels on $\beta$ and $\gamma$. Given two classes $\mathcal{A}$ and $\mathcal{B}$ with no object of size 0 in $\mathcal{B}$, the *composition* of $\mathcal{A}$ and $\mathcal{B}$, is the class $\mathcal{A}\circ\mathcal{B}$ —also written $\mathcal{A}(\mathcal{B})$ if $\mathcal{A}$ is a basic class—of labelled objects obtained as follows. Choose an object $\gamma \in \mathcal{A}$ to be the *core* of the composition and let $k=|\gamma|$ be its size. Then pick a k-set of elements from $\mathcal{B}$. Substitute each atom $v \in \gamma$ by an object $\gamma_v$ from the k-set, distributing distinct labels to the atoms of the composed object, i.e., the atoms in $\cup_{v\in\gamma}\gamma_v$. And forget the original labels on $\gamma$ and the $\gamma_v$. The composition construction is very powerful. For instance, it allows us to formulate the classical Set, Sequence, and Cycle constructions from basic classes. Indeed, the class of sequences (sets, cycles) of objects in a class $\mathcal{A}$ is simply the class $\mathrm{Seq}(\mathcal{A})$ ($\mathrm{Set}(\mathcal{A})$, $\mathrm{Cyc}(\mathcal{A})$, resp.). Sets, Cycles, and Sequences with a specific range for the number of components are also readily handled. We use the subscript notations $\mathrm{Seq}_{\geq k}(\mathcal{A})$, $\mathrm{Set}_{\geq k}(\mathcal{A})$, $\mathrm{Cyc}_{\geq k}(\mathcal{A})$, when the number of components is constrained to be at least some fixed value $k$.

## 2.3 Counting series

For labelled enumeration, the counting series is the exponential generating function, shortly the EGF, defined as

$$C(x) := \sum_n \frac{1}{n!}|\mathcal{C}_n|x^n \tag{1}$$

whereas for unlabelled enumeration the counting series is the *ordinary generating function*, defined as

$$\widetilde{C}(x) := \sum_n |\widetilde{C}_n|x^n. \tag{2}$$

In general, *cycle index sums* are used for unlabelled enumeration as a convenient refinement of ordinary generating functions. Cycle index sums are multivariate power series that preserve information on symmetries. A *symmetry* of size $n$ on a class $\mathcal{C}$ is a pair $(\sigma \in \mathfrak{S}_n, \gamma \in \mathcal{C}_n)$ such that $\gamma$ is stable under the action of $\sigma$ (notice that $\sigma$ is allowed to be the identity). The corresponding weight is defined as $\prod_{i=1}^n s_i^{c_i}$, where $s_i$ is a formal variable and $c_i$ is the number of cycles of length $i$ in $\sigma$. The cycle index sum of $\mathcal{C}$, denoted

| Basic classes | Notation | EGF | Cycle index sum |
|---|---|---|---|
| **Neutral Class** | $\mathcal{C} = 1$ | $C(z) = 1$ | $Z[\mathcal{C}] = 1$ |
| **Atomic Class** | $\mathcal{C} = \mathcal{Z}$ | $C(z) = z$ | $Z[\mathcal{C}] = s_1$ |
| **Sequence** | $\mathcal{C} = \mathrm{Seq}$ | $C(z) = \frac{1}{1-z}$ | $Z[\mathcal{C}] = \frac{1}{1-s_1}$ |
| **Set** | $\mathcal{C} = \mathrm{Set}$ | $C(z) = \exp(z)$ | $Z[\mathcal{C}] = \exp\left(\sum_{r\geq 1}\frac{1}{r}s_r\right)$ |
| **Cycle** | $\mathcal{C} = \mathrm{Cyc}$ | $C(z) = \log\left(\frac{1}{1-z}\right)$ | $Z[\mathcal{C}] = \sum_{r\geq 1}\frac{\phi(r)}{r}\log\left(\frac{1}{1-s_r}\right)$ |
| **Construction** | **Notation** | **Rule for EGF** | **Rule for Cycle index sum** |
| **Union** | $\mathcal{C} = \mathcal{A} + \mathcal{B}$ | $C(z) = A(z) + B(z)$ | $Z[\mathcal{C}] = Z[\mathcal{A}] + Z[\mathcal{B}]$ |
| **Product** | $\mathcal{C} = \mathcal{A} * \mathcal{B}$ | $C(z) = A(z) \cdot B(z)$ | $Z[\mathcal{C}] = Z[\mathcal{A}] \times Z[\mathcal{B}]$ |
| **Composition** | $\mathcal{C} = \mathcal{A} \circ \mathcal{B}$ | $C(z) = A(B(z))$ | $Z[\mathcal{C}] = Z[\mathcal{A}] \circ Z[\mathcal{B}]$ |

Figure 1: Basic classes and constructions, with their translations to generating functions for labelled classes and to cycle index sums for unlabelled classes. For the composition construction, the notation $Z[\mathcal{A}]\circ Z[\mathcal{B}]$ refers to the series $Z[\mathcal{A}] \circ Z[\mathcal{B}](s_1, s_2, \ldots) = Z[\mathcal{A}](Z[\mathcal{B}](s_1, s_2, \ldots), Z[\mathcal{B}](s_2, s_4, \ldots), Z[\mathcal{B}](s_3, s_6, \ldots), \ldots)$.

by $Z[\mathcal{C}](s_1, s_2, \ldots)$, is the multivariate series defined as the sum of the weight-monomials over all symmetries on $\mathcal{C}$. The ordinary generating function is obtained by substitution of $s_i$ by $x^i$:

$$\widetilde{C}(x) = Z[\mathcal{C}](x, x^2, \ldots).$$

### 2.4 Computation rules for the counting series

The symbolic method provides for each basic class and each construction an explicit simple rule to compute the EGF (labelled enumeration) and the cycle index sum (unlabelled enumeration), as shown in Figure 1. These rules will allow us to convert our decomposition grammar into an enumerative strategy in an automatic way. As an example, consider the class $\mathcal{T}$ of nonplane rooted trees. Such a tree is made of a root vertex and a collection of subtrees pending from the root-vertex, which yields

$$\mathcal{T} = \mathcal{Z} * \mathrm{Set} \circ \mathcal{T}.$$

For labelled enumeration, this is translated into the following equation satisfies by the EGF:

$$T(x) = x \exp(T(x)).$$

For unlabelled enumeration, this is translated into the following equation satisfied by the OGF (via the computation rules for cycle index sums):

$$\widetilde{T}(x) = x \exp\left(\sum_{r\geq 1}\frac{1}{r}\widetilde{T}(x^r)\right).$$

In general, if a class $\mathcal{C}$ is found to have a decomposition grammar, the rules of Figure 1 allow us to translate the combinatorial description of the class into an equation-system satisfied by the counting series *automatically* for both labelled and unlabelled structures. The purpose of this paper is to completely specify such a grammar to decompose any family of graphs into 3-connected components. Therefore we have to specify how the basic classes, constructions, and enumeration tools have to be defined in the specific case of graph classes.

## 2.5 Graph classes

Let us first mention that the graphs we consider are allowed to have multiple edges but no loops (multiple edges are allowed in the first formulation of the grammar, then we will explain how to adapt the grammar to simple graphs in Section 5.4). In the case of a class of graphs, we will need to take both vertices and edges into account. Accordingly, we consider a class of graphs as a species of combinatorial structures with two types of labelled atoms: vertices and edges. In general we imagine that if there are $n$ *labelled* vertices and $m$ *labelled* edges, then these labelled vertices carry distinct blue labels in $[1..n]$ and the edges carry distinct red labels in $[1..m]$ [1]. Hence, graph classes have to be treated in the extended framework of species with several types of atoms, see [2, Sec 2.4] (we shortly review here how the basic constructions and counting tools can be extended).

For labelled enumeration the exponential generating function (EGF) of a class of graphs is

$$G(x,y) = \sum_{n,m} \frac{1}{n!m!} |\mathcal{G}_{n,m}| x^n y^m,$$

where $\mathcal{G}_{n,m}$ is the set of graphs in $\mathcal{G}$ with $n$ vertices and $m$ edges. For unlabelled enumeration (i.e., graphs are considered up to relabelling the vertices and the edges), the ordinary generating function (OGF) is

$$\widetilde{G}(x,y) = \sum_{n,m} |\widetilde{\mathcal{G}}_{n,m}| x^n y^m,$$

where $\widetilde{\mathcal{G}}_{n,m}$ is the set of unlabelled graphs in the class that have $n$ vertices and $m$ edges. Cycle index sums can also be defined similarly as in the one-variable case, (as a sum of weight-monomials) but the definition is more complicated, as well as the computation rules, see [30]. In this article we restrict our attention to labelled enumeration and postpone to future work the applications of our grammar to unlabelled enumeration.

We distinguish three types of graphs: unrooted, vertex-pointed, and rooted. In an unrooted graph, all vertices and all edges are labelled. In a vertex-pointed graph, there is one distinguished vertex that is unlabelled, all the other vertices and edges are labelled. In a rooted graph, there is one distinguished edge—called the root—that is oriented, all the vertices are labelled except the extremities of the root, and all edges are labelled except the root. A class of unrooted graphs is typically denoted by $\mathcal{G}$, and the associated vertex-pointed and rooted classes are respectively denoted $\mathcal{G}'$ and $\overrightarrow{\mathcal{G}}$. Notice that $\mathcal{G}'_{n,m} \simeq \mathcal{G}_{n+1,m}$.

[1] If the graphs are simple, there is actually no need to label the edges, since two distinct edges are distinguished by the labels of their extremities.

The generating functions $G'$ of $\mathcal{G}'$ and $\overrightarrow{G}$ of $\overrightarrow{\mathcal{G}}$ satisfy:

$$G'(x,y) = \partial_x G(x,y), \qquad \overrightarrow{G}(x,y) = \frac{2}{x^2}\partial_y G(x,y).$$

A class of vertex-pointed graphs is called a *vertex-pointed class* and a class of rooted graphs is called a rooted class. In this article, all vertex-pointed classes will be of the form $\mathcal{G}'$, but we will consider rooted classes that are not of the form $\overrightarrow{\mathcal{G}}$; for such classes we require nevertheless that the class is stable when reversing the direction of the root-edge.

The basic graph classes are the following:

- The *vertex-class* $v$ stands for the class made of a unique graph that has a single vertex and no edge. The series is $(x,y) \mapsto x$.
- The *edge-class* $e$ stands for the class made of a unique graph that has two unlabelled vertices connected by one directed labelled edge. The series is $(x,y) \mapsto y$.
- The *ring-class* $\mathcal{R}$ stands for the class of *ring-graphs*, which are cyclic chains of at least 3 edges. The series of $\mathcal{R}$ is $(x,y) \mapsto \frac{1}{2}(-\log(1-xy) - xy - \frac{1}{2}x^2y^2)$.
- The *multi-edge-class* $\mathcal{M}$ stands for the class of *multi-edge graphs*, which consist of 2 labelled vertices connected by $k \geq 3$ edges. The series of $\mathcal{M}$ is $(x,y) \mapsto \frac{1}{2}x^2(\exp(y)-1-y-\frac{y^2}{2})$.

The constructions we consider for graph classes are the following: *disjoint union*, *partitional product* (defined similarly as in the one-variable case), and now two types of substitution:

- *Vertex-substitution*: Given a graph class $\mathcal{A}$ (which might be unrooted, vertex-pointed, or rooted) and a vertex-pointed class $\mathcal{B}$, the class $\mathcal{C} = \mathcal{A} \circ_v \mathcal{B}$ is the class of graphs obtained by taking a graph $\gamma \in \mathcal{A}$, called the core graph, and attaching at each labelled vertex $v \in \gamma$ a graph $\gamma_v \in \mathcal{B}$, the vertex of attachment of $\gamma_v$ being the distinguished (unlabelled) vertex of $\gamma_v$. We have

  $$C(x,y) = A(xB(x,y), y),$$

  where $A$, $B$ and $C$ are respectively the exponential generating functions of $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$.

- *Edge-substitution*: Given a graph class $\mathcal{A}$ (which might be unrooted, vertex-pointed, or rooted) and a rooted class $\mathcal{B}$, the class $\mathcal{C} = \mathcal{A} \circ_e \mathcal{B}$ is the class of graphs obtained by taking a graph $\gamma \in \mathcal{A}$, called the core graph, and substituting each labelled edge $e = \{u,v\}$ (which is implicitly given an orientation) of $\gamma$ by a graph $\gamma_e \in \mathcal{B}$, thereby identifying the origin of the root of $\gamma_e$ with $u$ and the end of the root of $\gamma_e$ with $v$. After the identification, the root edge of $\gamma_e$ is deleted. We have

  $$C(x,y) = A(x, B(x,y)),$$

  where $A$, $B$ and $C$ are respectively the generating functions of $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$.

# 3 Tree decomposition and dissymmetry theorem

The dissymmetry theorem for trees [2] makes it possible to express the class of unrooted trees in terms of classes of rooted trees. Precisely, let $\mathcal{A}$ be the class of tree, and let us define the following associated rooted families: $\mathcal{A}_\circ$ is the class of trees where a node is marked, $\mathcal{A}_{\circ-\circ}$ is the class of trees where an edge is marked, and $\mathcal{A}_{\circ\to\circ}$ is the class of trees where an edge is marked and is given a direction. Then the class $\mathcal{A}$ is related to these three associated rooted classes by the following identity:

$$\mathcal{A} + \mathcal{A}_{\circ\to\circ} \simeq \mathcal{A}_\circ + \mathcal{A}_{\circ-\circ}. \tag{3}$$

The theorem is named after the dissymmetry resulting in a tree rooted anywhere other than at its centre, see [2]. Equation (3) is an elegant and flexible counterpart to the dissimilarity equation discovered by Otter [22]; as we state in Theorem 3.1 below, it can easily be extended to classes for which a tree can be *associated* with each object in the class.

A *tree-decomposable* class is a class $\mathcal{C}$ such that to each object $\gamma \in \mathcal{C}$ is associated a tree $\tau(\gamma)$ whose nodes are distinguishable in some way (e.g., using the labels on the vertices of $\gamma$). Denote by $\mathcal{C}_\circ$ the class of objects of $\mathcal{C}$ where a node of $\tau(\gamma)$ is distinguished, by $\mathcal{C}_{\circ-\circ}$ the class of objects of $\mathcal{C}$ where an edge of $\tau(\gamma)$ is distinguished, and by $\mathcal{C}_{\circ\to\circ}$ the class of objects of $\mathcal{C}$ where an edge of $\tau(\gamma)$ is distinguished and given a direction. The principles and proof of the dissymmetry theorem can be straightforwardly extended to any tree-decomposable class, giving rise to the following statement.

**Theorem 3.1.** *(Dissymmetry theorem for tree-decomposable classes) Let $\mathcal{C}$ be a tree-decomposable class. Then*

$$\mathcal{C} + \mathcal{C}_{\circ\to\circ} \simeq \mathcal{C}_\circ + \mathcal{C}_{\circ-\circ}. \tag{4}$$

Note that, if the trees associated to the graphs in $\mathcal{C}$ are bipartite, then $\mathcal{C}_{\circ\to\circ} \simeq 2\mathcal{C}_{\circ-\circ}$. Hence, Equation (4) simplifies to

$$\mathcal{C} \simeq \mathcal{C}_\circ - \mathcal{C}_{\circ-\circ}. \tag{5}$$

(At the upper level of generating functions, this reflects the property that the number of vertices in a tree exceeds the number of edges by one.)

# 4 Tutte's decomposition and beyond: Decomposing a graph into 3-connected components

In this section we recall Tutte's decomposition [28] of a graph into 3-connected components, which we will translate into a grammar in Section 5. The decomposition works in three levels: (i) standard decomposition of a graph into connected components, (ii) decomposition of a connected graph into 2-connected blocks that are articulated around vertices, (iii) decomposition of a 2-connected into 3-connected components that are articulated around (virtual) edges.

A nice feature of Tutte's decomposition is that the second and third level are "tree-like" decompositions, meaning that the "backbone" of the decomposition is a tree. The tree associated with (ii) is called the Bv-tree, and the tree associated with (iii) is called the RMT-tree (the trees are named after the possible types of the nodes). The tree-property of the decompositions will enable us to apply the dissymmetry theorem—Theorem 3.1—in order to write down the grammar. As we will see in Section 5.2, writing the grammar will require the canonical decomposition of *vertex-pointed* 2-connected graphs. It turns out that a smaller backbone-tree (smaller than for *unrooted* 2-connected graphs) is more convenient in order to apply the dissymmetry theorem, thereby simplifying the decomposition process for vertex-pointed 2-connected graphs. Thus in Section 4.4 we introduce these smaller trees, called restricted RMT-trees (to our knowledge, these trees have not been considered before).

## 4.1 Graphs and connectivity

We give here a few definitions on graphs and connectivity, following Tutte's terminology [28]. The vertex-set (edge-set) of a graph $G$ is denoted by $V(G)$ ($E(G)$, resp.). A subgraph of a graph $G$ is a graph $G'$ such that $V(G') \subset V(G)$, $E(G') \subset E(G)$, and any vertex incident to an edge in $E(G')$ is in $V(G')$. Given an edge-subset $E' \subset E(G)$, the corresponding *induced graph* is the subgraph $G'$ of $G$ such that $E(G') = E'$ and $V(G')$ is the set of vertices incident to edges in $E'$; the induced graph is denoted by $G[E']$.

A graph is *connected* if any two of its vertices are connected by a path. A *1-separator* of a graph $G$ is given by a partition of $E(G)$ into two nonempty sets $E_1, E_2$ such that $G[E_1]$ and $G[E_2]$ intersect at a unique vertex $v$; such a vertex is called *separating*. A graph is *2-connected* if it has at least two vertices and no 1-separator. Equivalently (since we do not allow any loop), a 2-connected graph $G$ has at least two vertices and the deletion of any vertex does not disconnect $G$. A *2-separator* of a graph is given by a partition of $E(G)$ into two subsets $E_1, E_2$ each of cardinality at least 2, such that $G[E_1]$ and $G[E_2]$ intersect at two vertices $u$ and $v$; such a pair $\{u, v\}$ is called a *separating vertex pair*. A graph is *3-connected* if it has no 2-separator and has at least 4 vertices. (The latter condition is convenient for our purpose, as it prevents any ring-graph or multiedge-graph from being 3-connected.) Equivalently, a 3-connected graph $G$ has at least 4 vertices, no loop nor multiple edges, and the deletion of any two vertices does not disconnect $G$.

## 4.2 Decomposing a connected graph into 2-connected ones

There is a well-known decomposition of a graph into 2-connected components, which is described in several books [16, 8, 20, 28].

Given a connected graph $C$, a *block* of $C$ is a maximal 2-connected induced subgraph of $C$. The set of blocks of $C$ is denoted by $\mathfrak{B}(C)$. A vertex $v \in C$ is said to be *incident* to a block $B \in \mathfrak{B}(C)$ if $v$ belongs to $B$. The *Bv-tree* of $C$ describes the incidences between vertices and blocks of $C$, i.e., it is a bipartite graph $\tau(C)$ with node-set $V(C) \cup \mathfrak{B}(C)$, and edge-set given by the incidences between the vertices and the blocks of $C$, see Figure 2.

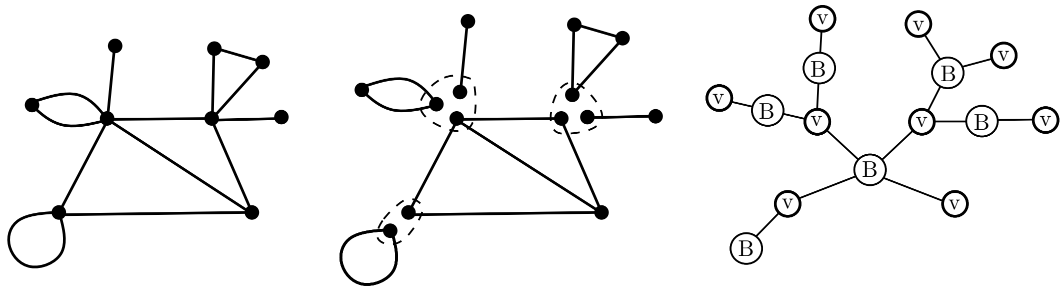



Figure 2: Decomposition of a connected graph into blocks, and the associated Bv-tree.

The graph $\tau(C)$ is actually a tree, as shown for instance in [28, 20]. Conversely, take a collection $\mathfrak{B}$ of 2-connected graphs, called blocks, and a vertex-set $V$ such that every vertex in $V$ is in at least one block and the graph of incidences between blocks and vertices is a tree $\tau$. Then the resulting graph is connected and has $\tau$ as its Bv-tree. Consequently, connected graphs can be *identified* with their tree-decompositions into blocks, which will be very useful for deriving decomposition grammars.

## 4.3 Decomposing a 2-connected graph into 3-connected ones

In this section we recall Tutte's decomposition of a 2-connected graph into 3-connected components [27]. A similar decomposition has also been described by Hopcroft and Tarjan [17], however they use a split-and-remerge process, whereas Tutte's method only involves (more restrictive) split operations. We follow here the presentation of Tutte.

First, one has to define connectivity modulo a pair of vertices. Let $G$ be a 2-connected graph and $\{u, v\}$ a pair of vertices of $G$. Then $G$ is said to be *connected modulo* $[u, v]$ if there exists no partition of $E(G)$ into two nonempty sets $E_1, E_2$ such that $G[E_1]$ and $G[E_2]$ intersect only at $u$ and $v$. Being non-connected modulo $[u, v]$ means either that $u$ and $v$ are adjacent or that the deletion of $u$ and $v$ disconnects the graph.

Consider a 2-separator $E_1, E_2$ of a 2-connected graph $G$, with $u, v$ the corresponding separating vertex-pair. Then $E_1, E_2$ is called a *split-candidate*, denoted by $\{E_1, E_2, u, v\}$, if $G[E_1]$ is connected modulo $[u, v]$ and $G[E_2]$ is 2-connected. Figure 3(a) gives an example of a split-candidate, where $G[E_1]$ is connected modulo $[u, v]$ but not 2-connected, while $G[E_2]$ is 2-connected but not connected modulo $[u, v]$.

As described below, split candidates make it possible to completely decompose a 2-connected graph into 3-connected components. We consider here only 2-connected graphs with at least 3 edges (graphs with less edges are degenerated for this decomposition). Given a split candidate $S = \{E_1, E_2, u, v\}$ in a 2-connected graph $G$ (see Figure 3(b)), the corresponding *split operation* is defined as follows, see Figure 3(b)-(c):

- an edge $e$, called a *virtual edge*, is added between $u$ and $v$,
- the graph $G[E_1]$ is separated from the graph $G[E_2]$ by cutting along the edge $e$.

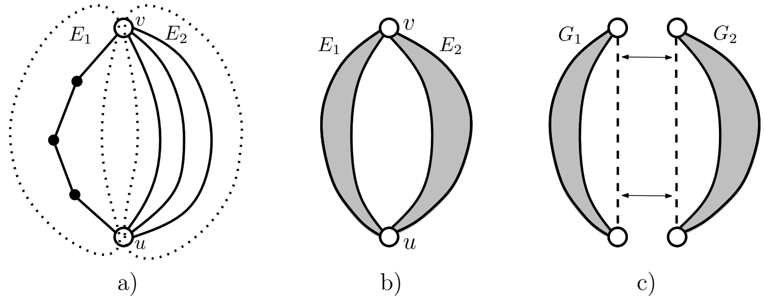



Figure 3: (a) Example of a split candidate. (b) Splitting the edge-set. (c) Splitting a graph along a virtual edge.

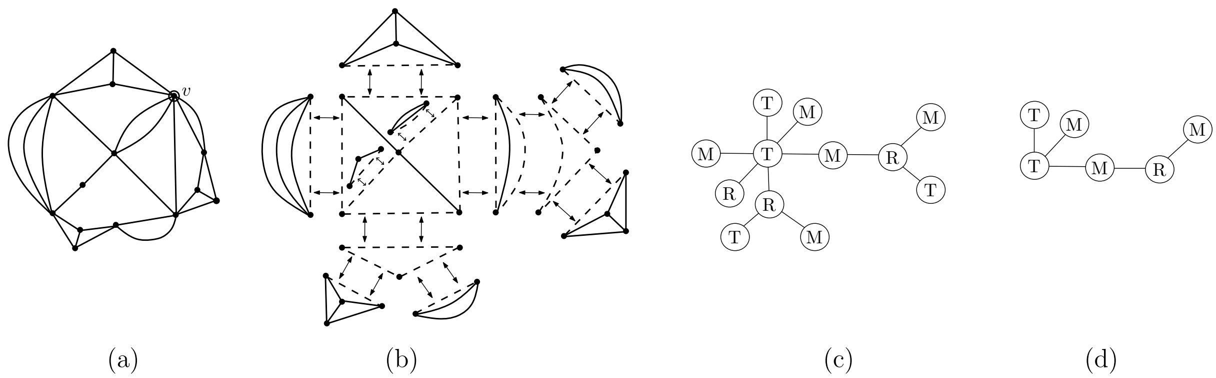



Figure 4: (a) A 2-connected graph, (b) decomposed into bricks. (c) The associated RMT-tree. (d) The associated restricted RMT-tree if the graph is pointed at $v$.

Such a split operation yields two graphs $G_1$ and $G_2$, see Figure 3(d), which correspond respectively to $G[E_1]$ and $G[E_2]$ together with $e$ as a real edge. The graphs $G_1$ and $G_2$ are said to be *matched* by the virtual edge $e$. It is easily checked that $G_1$ and $G_2$ are 2-connected (and have at least 3 edges). The splitting process can be repeated until no split candidate remains left.

As shown by Tutte in [28], the structure resulting from the split operations is independent of the order in which they are performed. It is a collection of graphs, called the *bricks* of $G$, which are articulated around virtual edges, see Figure 4(b). By definition of the decomposition, each brick has no split candidate; Tutte has shown that such graphs are either multiedge-graphs (M-bricks) or ring-graphs (R-bricks), or 3-connected graphs with at least 4 vertices (T-bricks).

The *RMT-tree* of $G$ is the graph $\tau(G)$ whose nodes are the bricks of $G$ and whose edges correspond to the virtual edges of $G$ (each virtual edge matches two bricks), see Figure 4. The graph $\tau(G)$ is indeed a tree [28]. By maximality of the decomposition, it is easily checked that $\tau(G)$ has no two R-bricks adjacent nor two M-bricks adjacent.

Call a *brick-graph* a graph that is either a ring-graph or a multi-edge graph or a 3-

connected graph, with again the letter-triple $\{R, M, T\}$ to refer to the type of the brick. The inverse process of the split decomposition consists in taking a collection of brick-graphs and a collection of edges, called virtual edges, so that each virtual edge belongs to two bricks, and so that the graph $\tau$ with vertex-set the bricks and edge-set the virtual edges (each virtual edge matches two bricks) is a tree avoiding two R-bricks or two M-bricks being adjacent. Then the resulting graph, obtained by matching the bricks along virtual edges and then erasing the virtual edges, is a 2-connected graph that has $\tau$ as its RMT-tree. Hence, 2-connected graphs with at least 3 edges can be identified with their RMT-tree, which again will be useful for writing down a decomposition grammar.

### 4.4 The restricted RMT-tree

The grammar to be written in Section 5 requires to decompose not only unrooted 2-connected graphs, but also *vertex-pointed* 2-connected graphs. It turns out that these vertex-pointed 2-connected graphs are much more convenient to decompose using a sub-tree of the RMT-tree.

The *restricted RMT-tree* of a 2-connected graph $G$ (with at least 3 edges) rooted at a vertex $v$, is defined as the subgraph $\tau'(G)$ of the RMT-tree $\tau(G)$ induced by the bricks containing $v$ and by the edges of $\tau(G)$ connecting two such bricks.

**Lemma 4.1.** *The restricted RMT-tree of a vertex-pointed 2-connected graph with at least 3 edges is a tree.*

*Proof.* Let $G$ be a vertex-pointed 2-connected graph with at least 3 edges. Let $\tau(G)$ be the RMT-tree of $G$ and let $\tau'(G)$ be the restricted RMT-tree of $G$. The pointed vertex is denoted by $v$. As $\tau'(G)$ is a subgraph of the tree $\tau(G)$, it is enough to show that $\tau'(G)$ is connected for it to be a tree. Recall that a virtual edge $e$ corresponds to splitting $G$ into two graphs $G_1 = G[E_1] + e$ and $G_2 = G[E_2] + e$, where $E_1, E_2$ is a 2-separator of $G$. The two subtrees $T_1$ and $T_2$ attached at each extremity of the virtual edge correspond to the split-decomposition of $G_1$ and $G_2$, respectively. Hence, the pointed vertex $v$, if not incident to the virtual edge $e$, is either a vertex of $G_1 \backslash e$ or is a vertex of $G_2 \backslash e$. In the first (second) case, $\tau'(G)$ is contained in $T_1$ ($T_2$, respectively). Hence, if an edge of $\tau(G)$ is not in $\tau'(G)$, then $\tau'(G)$ does not overlap simultaneously with the two subtrees attached at each extremity of that edge. This property ensures that $\tau'(G)$ is connected. $\square$

Having proved that the restricted RMT-tree is indeed a tree and not a forest, we will be able to use the dissymmetry theorem—Theorem 3.1—in order to write a decomposition grammar for the class of vertex-pointed 2-connected graphs. The restricted RMT-tree turns out to be much better adapted for this purpose than the RMT-tree.

## 5 Decomposition Grammar

In this section we translate Tutte's decomposition into an explicit grammar. Thanks to this grammar, counting a family of graphs reduces to counting the 3-connected subfamily, which turns out to be a fruitful strategy in many cases, in particular for planar graphs, as we will see Section 7.

Given a graph family $\mathcal{G}$, our grammar corresponds at the first level to the connected components, at the second level to the decomposition of a connected graph into 2-connected blocks, and at the third level to the decomposition of a 2-connected graph into 3-connected components. The first level is classic, the second level already makes use of the dissymmetry theorem, it is implicitly used by Robinson [23], and appears explicitly in the work by Leroux [18, 2]. The third level is new (though Leroux et al [13] have recently independently derived general equation systems relating the series of 2-connected graphs and 3-connected graphs of a given class). As we will see, it makes an even more extensive use of the dissymmetry theorem than the second level.

We define the following subfamilies of $\mathcal{G}$:

- The class $\mathcal{G}_1$ is the subfamily of graphs in $\mathcal{G}$ that are connected and have at least one vertex.
- The class $\mathcal{G}_2$ is the subfamily of graphs in $\mathcal{G}$ that are 2-connected and have at least two vertices. Multiple edges are allowed. (The smallest possible such graph is the link-graph that has two vertices connected by one edge.)
- The class $\mathcal{G}_3$ is the subfamily of graphs in $\mathcal{G}$ that are 3-connected and have at least four vertices. (The smallest possible such graph is the tetrahedron.)

A class $\mathcal{G}$ of graphs is said to be *stable under Tutte's decomposition* if it satisfies the following property:

"any graph $G$ is in $\mathcal{G}$ iff all 3-connected components of $G$ are in $\mathcal{G}$".

Notice that a class of graphs stable under Tutte's decomposition satisfies the following properties:

- a graph $G$ is in $\mathcal{G}$ iff all its connected components are in $\mathcal{G}_1$,
- a graph $G$ is in $\mathcal{G}_1$ iff all its 2-connected components are in $\mathcal{G}_2$,
- a graph $G$ is in $\mathcal{G}_2$ iff all its 3-connected components are in $\mathcal{G}_3$.

## 5.1 General from connected graphs

The first level of the grammar is classic. A graph is simply the collection of its connected components, which translates to:

$$\mathcal{G} = \mathrm{Set}(\mathcal{G}_1). \tag{6}$$

## 5.2 Connected from 2-connected graphs

In order to write down the second level, i.e., decompose the connected class $\mathcal{C} := \mathcal{G}_1$, we define the following classes: $\mathcal{C}_B$ is the class of graphs in $\mathcal{G}_1$ with a distinguished block, $\mathcal{C}_v$ is the class of graphs in $\mathcal{G}_1$ with a distinguished vertex, and $\mathcal{C}_{Bv}$ is the class of graphs in $\mathcal{G}_1$ with a distinguished incidence block-vertex. In other words, $\mathcal{C}_B$, $\mathcal{C}_v$, and $\mathcal{C}_{Bv}$ correspond to graphs in $\mathcal{G}_1$ where one distinguishes in the associated Bv-tree, respectively, a v-node, a B-node, and an edge. The generalized dissymmetry theorem yields the following relation between $\mathcal{C} = \mathcal{G}_1$ and the auxiliary rooted classes:

$$\mathcal{C} + \mathcal{C}_{Bv} = \mathcal{C}_v + \mathcal{C}_B,$$

which can be rewritten as

$$\mathcal{C} = \mathcal{C}_v + \mathcal{C}_B - \mathcal{C}_{Bv}. \tag{7}$$

Clearly the class $\mathcal{C}'$ is related to $\mathcal{C}_v$ by $\mathcal{C}_v = v * \mathcal{C}'$. To decompose $\mathcal{C}'$, we observe that the pointed vertex gives a starting point for a recursive decomposition. Precisely, from the block decomposition described in Section 4.2, any vertex-pointed connected graph is obtained as follows: take a collection of vertex-pointed 2-connected graphs attached together at their marked vertices, and attach a vertex-pointed connected graph at each non-pointed vertex of these 2-connected graphs. (Clearly the 2-connected graphs correspond to the blocks incident to the pointed vertex in the resulting graph.)

This recursive decomposition translates to the equation

$$\mathcal{C}' = \mathrm{Set}(\mathcal{G}_2' \circ_v \mathcal{C}'). \tag{8}$$

Similarly, each graph in $\mathcal{C}_B$ is obtained in a unique way by taking a block in $\mathcal{G}_2$ and attaching at each vertex of the block a vertex-pointed connected graph in $\mathcal{C}'$, which yields

$$\mathcal{C}_B = \mathcal{G}_2 \circ_v \mathcal{C}'. \tag{9}$$

Finally, each graph in $\mathcal{C}_{Bv}$ is obtained from a vertex-pointed block in $\mathcal{G}_2'$ by attaching at each vertex of the block —even the root vertex— a vertex-pointed connected graph, which yields

$$\mathcal{C}_{Bv} = (v * \mathcal{G}_2') \circ_v \mathcal{C}'. \tag{10}$$

The grammar to decompose a class of connected graphs into 2-connected components results from the concatenation of Equations (7), (8), (9), and (10).

A similar grammar is given in the book of Bergeron, Labelle and Leroux [2]. Notice that there are two terminal classes in this grammar, the class $\mathcal{G}_2$ and the class $\mathcal{G}_2'$.

## 5.3 2-Connected from 3-connected graphs

In this section we start to describe the new contributions of this article, namely the decomposition grammars for $\mathcal{G}_2$ and $\mathcal{G}_2'$.

Let us begin with $\mathcal{G}_2$. Again we have to define auxiliary classes that correspond to the different ways to distinguish a node or an edge in the RMT-tree. Let $\mathcal{B}$ be the class of graphs in $\mathcal{G}_2$ with at least 3 edges (i.e., those whose RMT-tree is not empty). Since we consider graph classes stable under Tutte's decomposition, the link-graph $\ell_1$ and the double-link graph $\ell_2$ (which have counting series $x^2y/2$ and $x^2y^2/2$, respectively) are in $\mathcal{G}_2$, hence

$$\mathcal{G}_2 = \ell_1 + \ell_2 + \mathcal{B}. \tag{11}$$

Next we decompose $\mathcal{B}$ using the RMT-tree. Let $\mathcal{B}_\circ$ ($\mathcal{B}_{\circ-\circ}$, $\mathcal{B}_{\circ\to\circ}$) be the class of graphs in $\mathcal{B}$ such that the RMT-tree carries a distinguished node (edge, directed edge, resp.). Theorem 3.1 yields

$$\mathcal{B} = \mathcal{B}_\circ + \mathcal{B}_{\circ-\circ} - \mathcal{B}_{\circ\to\circ}. \tag{12}$$

The class $\mathcal{B}_\circ$ is naturally partitioned into 3 classes $\mathcal{B}_R$, $\mathcal{B}_M$, and $\mathcal{B}_T$, depending on the type of the distinguished node (R-node, M-node, or T-node). Similarly, the class $\mathcal{B}_{\circ-\circ}$ is partitioned into 4 classes $\mathcal{B}_{R-M}$, $\mathcal{B}_{R-T}$, $\mathcal{B}_{M-T}$, and $\mathcal{B}_{T-T}$ (recall that a RMT-tree has no two adjacent R-bricks nor two adjacent M-bricks); and $\mathcal{B}_{\circ\to\circ}$ is partitioned into 7 classes $\mathcal{B}_{R\to M}$, $\mathcal{B}_{M\to R}$, $\mathcal{B}_{R\to T}$, $\mathcal{B}_{T\to R}$, $\mathcal{B}_{M\to T}$, $\mathcal{B}_{T\to M}$, and $\mathcal{B}_{T\to T}$. Notice that $\mathcal{B}_{R\to M} \simeq \mathcal{B}_{M\to R} \simeq \mathcal{B}_{R-M}$, $\mathcal{B}_{R\to T} \simeq \mathcal{B}_{T\to R} \simeq \mathcal{B}_{R-T}$, and $\mathcal{B}_{M\to T} \simeq \mathcal{B}_{T\to M} \simeq \mathcal{B}_{M-T}$. Hence, Equation (12) is rewritten as

$$\mathcal{B} = \mathcal{B}_R + \mathcal{B}_M + \mathcal{B}_T - \mathcal{B}_{R-M} - \mathcal{B}_{R-T} - \mathcal{B}_{M-T} - \mathcal{B}_{T\to T} + \mathcal{B}_{T-T}. \tag{13}$$

### 5.3.1 Networks

In order to decompose the classes on the right-hand-side of Equation (13), we first have to decompose the class of rooted 2-connected graphs in $\mathcal{G}$, more precisely we need to specify a grammar for a class of objects closely related to $\overrightarrow{\mathcal{G}_2}$, which are called networks. A network is defined as a connected graph arising from a graph in $\overrightarrow{\mathcal{G}_2}$ by deleting the root-edge; the origin and end of the root-edge are respectively called the 0-pole and the $\infty$-pole of the network. The associated class is classically denoted by $\mathcal{D}$ in the literature [29]. Observe that the only rooted 2-connected graph disconnected by root-edge deletion is the rooted link-graph. Hence

$$\overrightarrow{\mathcal{G}_2} = 1 + \mathcal{D},$$

where the rooted link-graph has weight 1 instead of $e$ because, in a rooted class, the rooted edge is considered as unlabelled, i.e., is not counted in the size parameters. (We will see in Section 5.4 that the link between $\mathcal{D}$ and $\overrightarrow{\mathcal{G}_2}$ is a bit more complicated if multiple edges are forbidden.)

As discovered by Tracktenbrot [26] a few year's before Tutte's book appeared, the class of networks with at least 2 edges (recall that the root-edge has been deleted) is naturally partitioned into 3 subclasses: $\mathcal{S}$ for *series networks*, $\mathcal{P}$ for *parallel networks*, and $\mathcal{H}$ for *polyhedral networks*:

$$\mathcal{D} = e + \mathcal{S} + \mathcal{P} + \mathcal{H}. \tag{14}$$

With our terminology of RMT-tree, the three situations correspond to the root-edge belonging to a R-brick, M-brick, or T-brick, respectively [2]. In a similar way as for the class $\mathcal{G}_1'$ in Section 5.2, the root-edge gives a starting point for a recursive decomposition. Clearly, as there is no edge R-R in the RMT-tree, each series network is obtained as a collection of at least two non-series networks connected as a chain (the $\infty$-pole of a network is identified with the 0-pole of the following network in the chain):

$$\mathcal{S} = (\mathcal{D} - \mathcal{S}) * v * \mathcal{D}. \tag{15}$$

Similarly, as there is no edge M-M in the RMT-tree, each parallel-network is obtained as a collection of at least two non-parallel networks sharing the same 0- and $\infty$-poles:

$$\mathcal{P} = \mathrm{Set}_{\geq 2}(\mathcal{D} - \mathcal{P}). \tag{16}$$

Finally, each polyhedral network is obtained as a rooted 3-connected graph where each non-root edge is substituted by a network, which yields:

$$\mathcal{H} = \overrightarrow{\mathcal{G}_3} \circ_e \mathcal{D}. \tag{17}$$

The resulting decomposition grammar for $\mathcal{D}$ is obtained as the concatenation of Equations (14), (15), (16), and (17). This grammar has been known since Walsh [29]. Notice that the only terminal class is the 3-connected class $\overrightarrow{\mathcal{G}_3}$.

### 5.3.2 Unrooted 2-connected graphs

We can now specify the decompositions of the families on the right-hand-side of (13). Recall that $\mathcal{R}$ is the class of ring-graphs (polygons) and $\mathcal{M}$ is the class of multiedge graphs with at least 3 edges. Given a graph in $\mathcal{B}_R$, each edge $e$ of the distinguished R-brick is either a real edge or a virtual edge; in the latter case the graph attached on the other side of $e$ (i.e., the side not incident to the rooted R-brick) is naturally rooted at $e$; it is thus identified with a network (upon choosing an orientation of $e$), precisely it is a non-series network, as there are no two R-bricks adjacent. Hence

$$\mathcal{B}_R = \mathcal{R} \circ_e (\mathcal{D} - \mathcal{S}). \tag{18}$$

Similarly we obtain

$$\mathcal{B}_M = \mathcal{M} \circ_e (\mathcal{D} - \mathcal{P}) = (v^2 * \mathrm{Set}_{\geq 3}(\mathcal{D} - \mathcal{P}))/\bullet \leftrightarrows \bullet, \tag{19}$$

where the last notation means "up to exchanging the two pole-vertices of the multiedge component", and

$$\mathcal{B}_T = \mathcal{G}_3 \circ_e \mathcal{D}, \tag{20}$$

to be compared with $\mathcal{H} = \overrightarrow{\mathcal{G}_3} \circ_e \mathcal{D}$. Next we decompose 2-connected graphs with a distinguished edge in the RMT-tree. Consider the class $\mathcal{B}_{R-M}$. The distinguished edge of

[2] Actually Trackhtenbrot's decomposition can be seen as Tutte's decomposition restricted to rooted 2-connected graphs.

the RMT-tree corresponds to a virtual edge $\{u,v\}$ matching one R-brick and one M-brick, such that the two bricks are attached at $\{u,v\}$. Upon fixing an orientation of the virtual edge $\{u,v\}$, there is a series-network on one side of $\{u,v\}$ and a parallel-network on the other side. Notice that such a construction has to be considered up to orienting $\{u,v\}$, i.e., up to exchanging the two poles $u$ and $v$ (notation $/\bullet \leftrightarrows \bullet$). We obtain

$$\mathcal{B}_{R-M} = (\mathcal{S} * \mathcal{P})/\bullet \leftrightarrows \bullet. \tag{21}$$

Similarly,

$$\mathcal{B}_{M-T} = (\mathcal{P} * \mathcal{H})/\bullet \leftrightarrows \bullet, \quad \mathcal{B}_{R-T} = (\mathcal{S} * \mathcal{H})/\bullet \leftrightarrows \bullet, \quad \mathcal{B}_{T\to T} = (\mathcal{H} * \mathcal{H})/\bullet \leftrightarrows \bullet,$$
$$\mathcal{B}_{T-T} = (\mathcal{H} * \mathcal{H})/(\bullet \leftrightarrows \bullet, H \leftrightarrows H),$$

where the very last notation means "up to orienting the distinguished virtual edge".

### 5.3.3 Vertex-pointed 2-connected graphs

Now we decompose the class $\mathcal{G}_2'$ of vertex-pointed 2-connected graphs (recall that $\mathcal{G}_2'$ is, with $\mathcal{G}_2$, one of the two terminal classes in the decomposition grammar for connected graphs into 2-connected components). We proceed in a similar way as for $\mathcal{G}_2$, with the important difference that we use the restricted RMT-tree instead of the RMT-tree. Observe that the class of graphs in $\mathcal{G}_2'$ with at least 3 edges (those whose RMT-tree is not empty) is the derived class $\mathcal{V} := \mathcal{B}'$. By deriving the identity (11), we get

$$\mathcal{G}_2' = \ell_1' + \ell_2' + \mathcal{V} = v * e + v * \mathrm{Set}_2(e) + \mathcal{V}. \tag{22}$$

We denote by $\mathcal{V}_\circ$ ($\mathcal{V}_{\circ-\circ}$, $\mathcal{V}_{\circ\to\circ}$) the class of graphs in $\mathcal{G}_2'$ where the associated restricted RMT-tree carries a distinguished node (edge, oriented edge, resp.). Theorem 3.1 yields

$$\mathcal{V} = \mathcal{V}_\circ + \mathcal{V}_{\circ-\circ} - \mathcal{V}_{\circ\to\circ}. \tag{23}$$

Notice that $\mathcal{V}_\circ \neq (\mathcal{B}_\circ)'$, $\mathcal{V}_{\circ-\circ} \neq (\mathcal{B}_{\circ-\circ})'$, and $\mathcal{V}_{\circ\to\circ} \neq (\mathcal{B}_{\circ\to\circ})'$, due to the fact that the associated tree is not the same for $\mathcal{V} = \mathcal{B}'$ and for $\mathcal{B}$ (actually, taking the derivative of the grammar for $\mathcal{B}$ would produce a much more complicated grammar for $\mathcal{B}'$ than the one we will obtain using the restricted RMT-tree). We partition the classes $\mathcal{V}_\circ$, $\mathcal{V}_{\circ-\circ}$, and $\mathcal{V}_{\circ\to\circ}$ according to the types of the bricks incident to the root, and proceed with a decomposition in each case; the arguments are very similar as for the decomposition of $\mathcal{G}_2$. Take the example of $\mathcal{V}_T$. Since the pointed vertex of the graph is incident to the marked T-brick (this is where it is very nice to consider the restricted RMT-tree instead of the RMT-tree), we have

$$\mathcal{V}_T = \mathcal{G}_3' \circ_e \mathcal{D},$$

to be compared with $\mathcal{B}_T = \mathcal{G}_3 \circ_e \mathcal{D}$. Similarly we obtain

$$\mathcal{V}_R = \mathcal{R}' \circ_e (\mathcal{D} - \mathcal{S}), \qquad \mathcal{V}_M = \mathcal{M}' \circ_e (\mathcal{D} - \mathcal{P}) = v * \mathrm{Set}_{\geq 3}(\mathcal{D} - \mathcal{P}),$$
$$\mathcal{V}_{R-M} = v * \mathcal{S} * \mathcal{P}, \qquad \mathcal{V}_{R-T} = v * \mathcal{S} * \mathcal{H}, \qquad \mathcal{V}_{M-T} = v * \mathcal{P} * \mathcal{H},$$
$$\mathcal{V}_{T\to T} = v * \mathcal{H} * \mathcal{H}, \qquad \mathcal{V}_{T-T} = (v * \mathcal{H} * \mathcal{H})/H \leftrightarrows H.$$

We have now a generic grammar to decompose a family of graphs into 3-connected components; the complete grammar is shown in Figure 6, and the main dependencies are shown in Figure 5. Observe that, except for the easy basic classes $v$, $e$, $\ell_1$, $\ell_2$, $\mathrm{Set}_{\geq k}$, $\mathcal{R}$, $\mathcal{M}$, $\mathcal{R}'$, $\mathcal{M}'$, the only terminal classes are the 3-connected classes $\mathcal{G}_3$, $\mathcal{G}_3'$, and $\overrightarrow{\mathcal{G}_3}$.

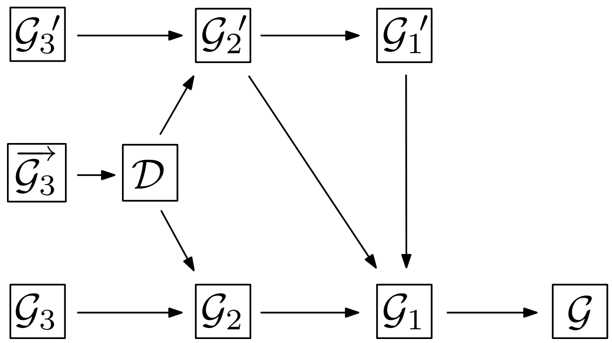



Figure 5: The main dependencies in the grammar.

## 5.4 Adapting the grammar for families of simple graphs

The grammar has been described for a family $\mathcal{G}$ satisfying the stability condition under Tutte's decomposition, and where multi-edges are allowed. It is actually very easy to adapt the grammar for the corresponding *simple* family of graphs. Call $\underline{\mathcal{G}}$ the subfamily of graphs in $\mathcal{G}$ that have no multiple edges and $\underline{\mathcal{G}_1}$, $\underline{\mathcal{G}_2}$, $\underline{\mathcal{G}_3}$ the corresponding subfamilies of connected, 2-connected, and 3-connected graphs.

To write down a grammar for $\underline{\mathcal{G}}$ we just need to trace where the multiple edges might appear in the decomposition grammar for $\mathcal{G}$. Clearly a graph is simple iff all its 2-connected components are simple, so we just need to look at the last part of the grammar: 2-connected from 3-connected. Among the two 2-connected graphs with less than 3 edges (the family $\epsilon$), we have to forbid the double-link graph, i.e., we have to take $\epsilon = \ell_2$ instead of $\epsilon = \ell_1 + \ell_2$. Among the 2-connected graphs with at least 3 edges—those giving rise to a RMT-tree—we have to forbid those where some M-brick has at least 2 components that are edges (indeed all representatives of a multiple edge are components of a same M-brick, so we can characterize the absence of multiple edges directly on the RMT-tree). Accordingly we have to change the specifications of the classes involving the decomposition at an M-brick, i.e., the classes $\mathcal{P}$, $\mathcal{B}_M$, and $\mathcal{V}_M$. In each case we have to distinguish if there is one edge component or zero edge-component incident to the M-brick, so there are two terms for the decomposition of each these families. For the class $\mathcal{P}$ of parallel networks, we have now $\mathcal{P} = e * \mathbf{Set}_{\geq 1}(\mathcal{D} - \mathcal{P} - e) + \mathbf{Set}_{\geq 2}(\mathcal{D} - \mathcal{P} - e)$ instead of $\mathcal{P} = \mathbf{Set}_{\geq 2}(\mathcal{D} - \mathcal{P})$. For the class $\mathcal{B}_M$, we have now $\mathcal{B}_M = (v^2 * e * \mathrm{Set}_{\geq 2}(\mathcal{D} - \mathcal{P} - e) + v^2 * \mathrm{Set}_{\geq 3}(\mathcal{D} - \mathcal{P} - e))/\bullet \leftrightarrows \bullet$ instead of $\mathcal{B}_M = (v^2 * \mathrm{Set}_{\geq 3}(\mathcal{D} - \mathcal{P}))/\bullet \leftrightarrows \bullet$. And for the class $\mathcal{V}_M$, we have now $\mathcal{V}_M = v * e * \mathbf{Set}_{\geq 2}(\mathcal{D} - \mathcal{P} - e) + v * \mathbf{Set}_{\geq 3}(\mathcal{D} - \mathcal{P} - e)$ instead of $\mathcal{V}_M = v * \mathbf{Set}_{\geq 3}(\mathcal{D} - \mathcal{P})$. These are indicated in the grammar (Figure 6).

**(1). General from Connected (folklore).**

$$\mathcal{G} = \mathbf{Set}(\mathcal{G}_1)$$

**(2). Connected from 2-Connected (Bergeron, Labelle, Leroux).**

$$\begin{aligned}
\mathcal{G}_1 &= \mathcal{C} = \mathcal{C}_v + \mathcal{C}_B - \mathcal{C}_{Bv} \text{ [dissymmetry theorem]}\\
\mathcal{C}_v &= v * \mathcal{C}'\\
\mathcal{C}' &= \mathbf{Set}(\mathcal{G}_2' \circ_v \mathcal{C}')\\
\mathcal{C}_B &= \mathcal{G}_2 \circ_v \mathcal{C}'\\
\mathcal{C}_{Bv} &= (v * \mathcal{G}_2') \circ_v \mathcal{C}'
\end{aligned}$$

**(3). 2-Connected from 3-Connected.**

(i) **Networks**

$$\begin{aligned}
\mathcal{D} &= e + \mathcal{S} + \mathcal{P} + \mathcal{H}\\
\mathcal{S} &= (\mathcal{D} - \mathcal{S}) * v * \mathcal{D}\\
\mathcal{P} &= \begin{cases} \mathbf{Set}_{\geq 2}(\mathcal{D} - \mathcal{P}), & \text{[Multi-edges allowed]}\\ e * \mathbf{Set}_{\geq 1}(\mathcal{D} - \mathcal{P} - e) + \mathbf{Set}_{\geq 2}(\mathcal{D} - \mathcal{P} - e), & \text{[No Multi-edges]} \end{cases}\\
\mathcal{H} &= \overrightarrow{\mathcal{G}_3} \circ_e \mathcal{D}
\end{aligned}$$

(ii) **Unrooted 2-Connected**

$$\begin{aligned}
\mathcal{G}_2 &= \epsilon + \mathcal{B}, \qquad \epsilon = \begin{cases} \ell_1 + \ell_2 & \text{[Multi-edges allowed]}\\ \ell_1 & \text{[No multi-edges]} \end{cases}\\
\mathcal{B} &= \mathcal{B}_R + \mathcal{B}_M + \mathcal{B}_T - \mathcal{B}_{R-M} - \mathcal{B}_{R-T} - \mathcal{B}_{M-T} - \mathcal{B}_{T\to T} + \mathcal{B}_{T-T}\\
\mathcal{B}_R &= \mathcal{R} \circ_e (\mathcal{D} - \mathcal{S})\\
\mathcal{B}_M &= \begin{cases} \mathcal{M} \circ_e (\mathcal{D} - \mathcal{P}) = (v^2 * \mathrm{Set}_{\geq 3}(\mathcal{D} - \mathcal{P}))/\bullet \leftrightarrows \bullet, & \text{[Multi-edges allowed]}\\ (v^2 * e * \mathrm{Set}_{\geq 2}(\mathcal{D} - \mathcal{P} - e) + v^2 * \mathrm{Set}_{\geq 3}(\mathcal{D} - \mathcal{P} - e))/\bullet \leftrightarrows \bullet, & \text{[No multi-edges]} \end{cases}\\
\mathcal{B}_T &= \mathcal{G}_3 \circ_e \mathcal{D}\\
\mathcal{B}_{R-M} &= (v^2 * \mathcal{S} * \mathcal{P})/\bullet \leftrightarrows \bullet, \qquad \text{[Up to pole exchange, denoted by } /\bullet \leftrightarrows \bullet\text{]}\\
\mathcal{B}_{R-T} &= (v^2 * \mathcal{S} * \mathcal{H})/\bullet \leftrightarrows \bullet\\
\mathcal{B}_{M-T} &= (v^2 * \mathcal{P} * \mathcal{H})/\bullet \leftrightarrows \bullet\\
\mathcal{B}_{T\to T} &= (v^2 * \mathcal{H} * \mathcal{H})/\bullet \leftrightarrows \bullet\\
\mathcal{B}_{T-T} &= (v^2 * \mathcal{H} * \mathcal{H})/(\bullet \leftrightarrows \bullet, H \leftrightarrows H), \qquad \text{[Up to pole and component exchange]}
\end{aligned}$$

(iii) **Vertex-pointed 2-Connected**

$$\begin{aligned}
\mathcal{G}_2' &= \epsilon' + \mathcal{B}', \qquad \epsilon' = \begin{cases} \ell_1' + \ell_2' = v * (e + \mathrm{Set}_2(e)) & \text{[Multi-edges allowed]}\\ \ell_1' = v * e & \text{[No multi-edges]} \end{cases}\\
\mathcal{B}' &= \mathcal{V} = \mathcal{V}_R + \mathcal{V}_M + \mathcal{V}_T - \mathcal{V}_{R-M} - \mathcal{V}_{R-T} - \mathcal{V}_{M-T} - \mathcal{V}_{T\to T} + \mathcal{V}_{T-T}\\
\mathcal{V}_R &= \mathcal{R}' \circ_e (\mathcal{D} - \mathcal{S})\\
\mathcal{V}_M &= \begin{cases} v * \mathbf{Set}_{\geq 3}(\mathcal{D} - \mathcal{P}), & \text{[Multi-edges allowed]}\\ v * e * \mathbf{Set}_{\geq 2}(\mathcal{D} - \mathcal{P} - e) + v * \mathbf{Set}_{\geq 3}(\mathcal{D} - \mathcal{P} - e), & \text{[No multi-edges]} \end{cases}\\
\mathcal{V}_T &= \mathcal{G}_3' \circ_e \mathcal{D}\\
\mathcal{V}_{R-M} &= v * \mathcal{S} * \mathcal{P}\\
\mathcal{V}_{R-T} &= v * \mathcal{S} * \mathcal{H}\\
\mathcal{V}_{M-T} &= v * \mathcal{P} * \mathcal{H}\\
\mathcal{V}_{T\to T} &= v * \mathcal{H} * \mathcal{H}\\
\mathcal{V}_{T-T} &= (v * \mathcal{H} * \mathcal{H})/H \leftrightarrows H
\end{aligned}$$

Figure 6: The grammar to decompose a graph family $\mathcal{G}$ stable under Tutte's decomposition. The non-basic terminal classes of the grammar are the 3-connected classes $\mathcal{G}_3$, $\mathcal{G}_3'$, and $\vec{\mathcal{G}_3}$.

**(1). General from Connected (folklore).**

$$G(z,y) = \exp(G_1(z,y))$$

**(2). Connected from 2-Connected (Bergeron, Labelle, Leroux).**

[The variables $z$ and $x$ are related by $x = zG_1{}'(z,y)$.]

$$\begin{aligned} G_1(z,y) &= C(z,y) = C_v(z,y) + C_B(z,y) - C_{vB}(z,y) \\ C_v(z,y) &= zC'(z,y) \\ C'(z,y) &= \exp\left(G_2{}'(x,y)\right) \\ C_B(z,y) &= G_2(x,y) \\ C_{vB}(z,y) &= xG_2{}'(x,y) \end{aligned}$$

**(3). 2-Connected from 3-Connected.**

[The variables $y$ and $w$ are related by $w = D(x,y)$.]

[We use the notations $\exp_{\geq k}(t) = \exp(t) - \sum_{i=1}^{k-1} \frac{t^i}{i!}$ and $\mathrm{loga}_{\geq k}(t) = \log(1/(1-t)) - \sum_{i=1}^{k-1} \frac{t^i}{i}$.]

(i) **Networks**

$$\begin{aligned} D(x,y) &= y + S(x,y) + P(x,y) + H(x,y) \\ S(x,y) &= (D(x,y) - S(x,y))xD(x,y) \\ P(x,y) &= \begin{cases} \exp_{\geq 2}(D(x,y) - P(x,y)) & \text{[Multi-edges]} \\ y\exp_{\geq 1}(D(x,y) - P(x,y) - y) + \exp_{\geq 2}(D(x,y) - P(x,y) - y) & \text{[No Multi-edges]} \end{cases} \\ H(x,y) &= \overrightarrow{G_3}(x,w) \end{aligned}$$

(ii) **Unrooted 2-Connected**

$$\begin{aligned} G_2(x,y) &= \epsilon(x,y) + B(x,y), \qquad \epsilon(x,y) = \begin{cases} x^2(y/2 + y^2/4) & \text{[Multi-edges]} \\ x^2y/2 & \text{[No multi-edges]} \end{cases} \\ B(x,y) &= B_R(x,y) + B_M(x,y) + B_T(x,y) - B_{R-M}(x,y) - B_{R-T}(x,y) - B_{M-T}(x,y) - B_{T-T}(x,y) \\ B_R(x,y) &= \mathrm{loga}_{\geq 3}(x \cdot (D(x,y) - S(x,y)))/2 \\ B_M(x,y) &= \begin{cases} x^2\exp_{\geq 3}(D(x,y) - P(x,y))/2, & \text{[Multi-edges]} \\ x^2\left(y\exp_{\geq 2}(D(x,y) - P(x,y) - y) + \exp_{\geq 3}(D(x,y) - P(x,y) - y)\right)/2, & \text{[No multi-edges]} \end{cases} \\ B_T(x,y) &= G_3(x,w) \\ B_{R-M}(x,y) &= x^2S(x,y) \cdot P(x,y)/2 \\ B_{R-T}(x,y) &= x^2S(x,y) \cdot H(x,y)/2 \\ B_{M-T}(x,y) &= x^2P(x,y) \cdot H(x,y)/2 \\ B_{T-T}(x,y) &= x^2H(x,y)^2/4 \end{aligned}$$

(iii) **Vertex-pointed 2-Connected**

$$\begin{aligned} G_2{}'(x,y) &= \epsilon'(x,y) + B'(x,y), \qquad \epsilon'(x,y) = \begin{cases} x(y + y^2/2) & \text{[Multi-edges]} \\ xy & \text{[No multi-edges]} \end{cases} \\ B'(x,y) &= V_R(x,y) + V_M(x,y) + V_T(x,y) - V_{R-M}(x,y) - V_{R-T}(x,y) - V_{M-T}(x,y) - V_{T-T}(x,y) \\ V_R(x,y) &= x^2(D(x,y) - S(x,y))^2D(x,y)/2 \\ V_M(x,y) &= \begin{cases} x\exp_{\geq 3}(D(x,y) - P(x,y)), & \text{[Multi-edges]} \\ xy\exp_{\geq 2}(D(x,y) - P(x,y) - y) + x\exp_{\geq 3}(D(x,y) - P(x,y) - y), & \text{[No multi-edges]} \end{cases} \\ V_T(x,y) &= G_3{}'(x,w) \\ V_{R-M}(x,y) &= xS(x,y) \cdot P(x,y) \\ V_{R-T}(x,y) &= xS(x,y) \cdot H(x,y) \\ V_{M-T}(x,y) &= xP(x,y) \cdot H(x,y) \\ V_{T-T}(x,y) &= xH(x,y)^2/2 \end{aligned}$$

Figure 7: The equation-system to express the series counting a family of graphs in terms of the series counting the 3-connected subfamilies (the terminal series are $G_3(x,w)$, $G_3{}'(x,w)$, $\vec{G}_3(x,w)$) is obtained by translating the grammar shown in Figure 6.

# 6 Application: counting a graph family reduces to counting the 3-connected subfamily

At first notice that our grammar makes it possible to get an *exact enumeration* procedure for a graph family $\mathcal{G}$ (stable under Tutte's decomposition) from the counting coefficients of the 3-connected subfamily, whether in the labelled or in the unlabelled setting. For instance, Figure 7 shows the translation of the grammar into an equation system for the corresponding exponential generating functions (for labelled enumeration). To get the counting coefficients, one can either turn this system into recurrences (see [5] for the precise recurrences) or simply compute Taylor developments of increasing degree in an inductive way. For doing all this, a partial decomposition grammar (allowing integrals) is actually enough, since formal integration just means dividing the $n$th coefficient by $n$.

The purpose of this section is to stress the relevance of our grammar in order to get *analytic expressions* for series counting families of graphs, thereby making it possible—via the techniques of singularity analysis—to get precise asymptotic results. (The latter point, singularity analysis, is not treated in this article, see [15, 14] for a detailed study.) Precisely, we show that finding an (implicit) analytic expression for a series counting a family of graphs stable under Tutte's decomposition reduces to finding an analytic expression for the series counting the 3-connected subfamily, a task that is easier in many cases, in particular for planar graphs, see Section 7.

Let us mention that Giménez, Noy, and Rué have recently shown in [14] that their method, which involves integrations, also makes it possible to have general equation systems relating the series counting a family of graphs and the series counting the 3-connected subfamilies. Hence we do not claim originality here, we only point out that the equation system can be derived automatically from a grammar, without using analytic integrations.

**Definition 6.1.** *A multivariate power series $C := C(\mathbf{z})$ is said to be* analytically specified *if it is of the following form (notice that the definition is inductive and relies heavily on substitutions).*

Initial level. *Rational generating functions (in any number of variables), the* log *function, and the* exp *function are analytically specified.*

Incremental definition. *Let $C = F_1$ be a power series that is uniquely specified by a system of the form*

$$\left\{ \begin{array}{rcl} F_1 & = & \text{Expression}_1, \\ & \vdots & \\ F_m & = & \text{Expression}_m, \end{array} \right. \tag{24}$$

*each expression being of the form $G(H_1, \ldots, H_k)$, where $G$ is a series* already known *to be analytically specified, and where each $H_i$ is either equal to one of the $F_j$'s or is a series already known to be analytically specified. Then $C$ is declared to be analytically specified as well.*

Analytically specified series are typically amenable to singularity analysis techniques in order to obtain precise asymptotic informations (enumeration, limit laws of parameters on a random instance); let us review briefly how the method works. Given a series $C$ specified by a system of the form (24), one has to consider the dependency graph between the $F_i$'s, in particular the strongly connected components of that graph. Notice that some of the $F_i$'s are expressed in (24) as a substitution in terms of functions already known to be analytically specified; in that case one would also include the dependency graph for the specification of this series, at a lower level. Therefore one gets a global dependency graph for $C$ that is a hierarchy of dependency graphs (one for each specifying system).

Then one treats the strongly components of this global dependency graph "from bottom to top", tracing the singularities on the way. In general, singularities are either due to the Jacobian of the system vanishing (for "tree-like" singularities, see the Drmota-Lalley-Woods theorem [9, sec VII.6]) or stem from a singularity of a function in a lower strongly connected component. Recently the techniques of singularity analysis have been successfully applied to the class of planar graphs by Giménez and Noy [15], building on earlier results by Bender, Gao, and Wormald [1]. One important difficulty in their work was to actually obtain a system of equations specifying planar graphs without integration operator, since integrations make it difficult to trace the singularities (in particular when several variables are involved).

We prove here that our grammar provides a direct combinatorial way to obtain such a system of equations for planar graphs. Thanks to our grammar, finding an (implicit) expression for the series counting a graph family $\mathcal{G}$ (stable under Tutte's decomposition) reduces to finding an (implicit) expression for the series counting the 3-connected subfamily $\mathcal{G}_3$.

**Theorem 6.1.** *Let $\mathcal{G}$ be a family of graphs satisfying the stability condition (i.e., a graph is in $\mathcal{G}$ iff its 3-connected components are also in $\mathcal{G}$). Call $\mathcal{G}_3$ the 3-connected subfamily of $\mathcal{G}$, and denote respectively by $G(x,y)$ and $G_3(x,y)$ the (exponential) counting series for the classes $\mathcal{G}$ and $\mathcal{G}_3$, where $x$ marks the vertices and $y$ marks the edges. Assume that $G_3(x,y)$ is analytically specified. Then $G(x,y)$ is also analytically specified via the equation-system shown in Figure 7 and the analytic specification for $G_3(x,y)$.*

*Proof.* Assume that $G_3$ is analytically specified. At first let us mention the following simple lemma: if a series $C$ is analytically specified then any of its partial derivatives is also analytically specified; this is due to the fact that one can "differentiate" the specification for $C$. (For the sake of illustration, take the univariate series for binary trees: $B(x) = x+B(x)^2$. Then the derivative $B'(x)$ satisfies the system $\{B'(x) = 1+2B'(x)B(x), B(x) = x + B(x)^2\}$.) Hence, the series $\underline{G_3}'$ and $\overrightarrow{G_3}$ are also analytically specified.

Now notice that $G_3$, $\underline{G_3}'$ and $\overrightarrow{G_3}$ are (together with elementary functions such as log) the only terminal series in the equation-system shown in Figure 7. In addition, this system clearly satisfies the requirements of an analytic specification. One shows successively "from bottom to top" that the series for the following graph classes are analytically specified: networks, unrooted 2-connected, vertex-pointed 2-connected, vertex-pointed connected, connected, and general graphs in the family one considers. $\square$

# 7 The series counting labelled planar graphs can be specified analytically

This section is dedicated to proving the following result.

**Theorem 7.1.** *The series counting labelled planar graphs w.r.t. vertices and edges is analytically specified.*

Theorem 7.1 has already been proved by Giménez and Noy [15] (as a crucial step to enumerate planar graphs asymptotically). Our contribution is a more straightforward proof of this result, which uses only combinatorial arguments and elementary algebraic manipulations; the main ingredients are our grammar and a bijective construction of vertex-pointed maps. In this way we avoid the technical integration steps addressed in [15].

Let us give a more precise outline of our proof. At first we take advantage of the grammar; the class of planar graphs satisfies the stability condition (planarity is preserved by taking the 3-connected components), hence proving Theorem 7.1 reduces (by Theorem 6.1) to finding an analytic specification for the series $G_3(x, w)$ counting 3-connected planar graphs, which is the task we address from now on.

By a theorem of Whitney [32], 3-connected planar graphs have a unique embedding on the sphere (up to reflections), hence counting 3-connected planar graphs is equivalent to counting 3-connected maps (up to reflections). To solve the latter problem, we combine several tools. Firstly, the Euler relation makes it possible to reduce the enumeration of 3-connected maps to the enumeration of vertex-pointed 3-connected maps; secondly a bijective construction due to Bouttier et al [7] allows us to count *unrestricted* vertex-pointed maps; finally a suitable adaptation of our grammar used in a top-to-bottom way yields the enumeration of vertex-pointed 3-connected maps via vertex-pointed 2-connected maps.

## 7.1 Maps

A *map* is a connected graph planarly embedded on the sphere up to continuous deformation. All maps considered here have at least one edge. As opposed to graphs, loops are allowed on arbitrary maps (as well as multi-edges). Thus, under Tutte's definition, a 2-connected map is either the loop-map, or is a loopless map with no separating vertex. Similarly as for graphs, a map is rooted if one of its edges is marked and oriented, and a map is vertex-pointed if one of its vertices is marked. An important remark is that rooting a map is equivalent to marking one of its corners (indeed there is a one-to-one correspondence between the corners and the half-edges of a map). We use the following notations:

- $\overrightarrow{M}(x, s)$ and $M'(x, s)$ are the series counting respectively rooted maps and vertex-pointed maps, where $x$ marks the number of vertices minus one (say, the root vertex for $\overrightarrow{M}$, the pointed vertex for $M'$), and $s$ marks the number of half-edges. Here vertices are unlabelled and half-edges are labelled (this is enough to avoid any symmetry), hence the series are exponential w.r.t. half-edges and ordinary w.r.t. vertices.

- Similarly, $\overrightarrow{L}(x,s)$, and $L'(x,s)$ are the series counting rooted 2-connected maps and vertex-pointed 2-connected maps respectively, where $x$ marks the number of (unlabelled) vertices minus one, and $s$ marks the number of (labelled) half-edges.
- Finally, $K(x,w)$ ($\overrightarrow{K}(x,w)$, $K'(x,w)$) are the series of unrooted (rooted, vertex-pointed, resp.) 3-connected maps with at least 4 vertices. Contrary to maps and 2-connected maps, the conventions are exactly the same as for graphs, i.e., $x$ marks labelled vertices (which carry blue labels) and $w$ marks labelled edges (which carry red labels); the marked vertex is unlabelled for maps counted by $K'(x,w)$, and the root-edge and its ends are unlabelled for maps counted by $\overrightarrow{K}(x,w)$.

## 7.2 Reduction to counting vertex-pointed 3-connected maps

At first we recall Whitney's theorem [31]: every 3-connected planar graph has a unique embedding on the sphere up to reflection. Since a 3-connected map with at least 4 vertices differs from its mirror-image, each planar graph in $\mathcal{G}_3$ gives rise to exactly two maps, which yields

$$G_3(x,w) = \frac{1}{2}K(x,w). \tag{25}$$

Next, the task of finding an expression for $K(x,w)$ is reduced to finding an expression for the series counting vertex-pointed 3-connected maps. Let $K^V(x,w)$, $K^E(x,w)$, $K^F(x,w)$ be the series counting respectively vertex-pointed, edge-pointed, and face-pointed 3-connected maps (here all vertices and all edges are labelled). The Euler relation (i.e., $1 = 1/2 \cdot (|\text{vertices}| - |\text{edges}| + |\text{faces}|)$) yields

$$K(x,w) = \frac{1}{2}(K^V(x,w) - K^E(x,w) + K^F(x,w)).$$

The series $K^E$ corresponds to half the series counting rooted 3-connected planar maps, which has been obtained by Mullin and Schellenberg [21] (we review briefly the computation scheme in Section 7.5.1) and more recently in a bijective way in [11]:

$$K^E(x,w) = \frac{1}{2}x^2 w\overrightarrow{K}(x,w) = \frac{1}{2}x^2\left(w - \frac{xw^2}{1+xw} - \frac{w^2}{1+w} - \frac{\gamma_1\gamma_2}{xw(1+\gamma_1+\gamma_2)^3}\right),$$

where $\gamma_1 = \gamma_1(x,w)$ and $\gamma_2 = \gamma_2(x,w)$ are specified by the system

$$\gamma_1 = xw(1+\gamma_2)^2, \qquad \gamma_2 = w(1+\gamma_1)^2.$$

Moreover, as the class of 3-connected maps is well known to be stable by duality, we have (using again the Euler relation)

$$K^F(x,w) = x^2 K^V(1/x, xw).$$

We conclude that finding an analytic expression for $G_3(x,w)$ reduces to finding an analytic expression for $K^V(x,w)$, i.e., reduces to finding an analytic expression for $K'(x,w) = K^V(x,w)/x$.

## 7.3 Expression for the series of vertex-pointed maps

Our goal is now to get an analytic expression for the series $K'(x,w)$ counting vertex-pointed 3-connected maps. At first we perform this task for the series counting vertex-pointed maps in this section (we will go subsequently from vertex-pointed maps to vertex-pointed 3-connected maps), taking advantage of a recent bijective construction.

### 7.3.1 Bijection between vertex-pointed maps and mobiles

The Bouttier, Di Francesco, Guitter bijection, which is described in [7], relates the problem of counting vertex-pointed maps to the enumeration of a certain class of trees, called *mobiles*. We state here the bijection between mobiles and vertex-pointed maps in a slightly reformulated way.

**Definition 7.1.** *A* mobile *is a plane tree such that:*

- *vertices are of two types: white vertices ($\circ$) and black vertices ($\bullet$),*
- *edges are of two types: $\bullet - \bullet$ and $\bullet - \circ$,*
- *additional "legs" are pending from black vertices; these legs are not considered as edges,*
- *each black vertex has exactly as many pending legs as it has white neighbours.*

*The $2m$ half-edges (not counting the legs) carry distinct labels in $[1..2m]$.*

We define $\mathcal{T}[n,2m]$ as the set of mobiles with $n$ (unlabelled) white vertices and $2m$ (labelled) half-edges, and $\mathcal{M}'[n,2m]$ as the set of vertex-pointed maps with $n+1$ (unlabelled) vertices and $2m$ (labelled) half-edges. The following theorem is a reformulation of [7]:

**Theorem 7.2 (Bouttier, Di Francesco and Guitter).** *For $n \geq 0$ and $m \geq 1$, there is an explicit bijection between $\mathcal{M}'[n,2m]$ and $\mathcal{T}[n,2m]$.*

The reader familiar with [7] may be surprised by our definition of mobiles: in [7], the white vertices of the mobiles and the edges of type $\bullet - \bullet$ are labelled by additional positive integers constrained by certain conditions[3]. Moreover, the original mobiles of [7] have no legs. Indeed, the role of the legs is precisely to encode the variations of the constrained labels. To wit, interpreting legs and white neighbours around a black vertex as, respectively, increments and decrements of the constrained labels, one easily reconstructs a valide mobile, in the sense of [7]. We leave the details to the reader.

[3]These labels correspond to a distance from a reference vertex, hence they are of a different nature from the labels considered here, which are just used to distinguish the atoms.

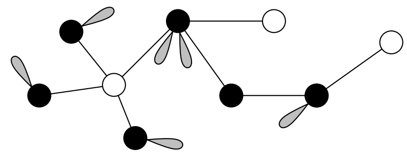


Figure 8: A mobile with 6 black vertices, 3 white vertices and 8 edges.

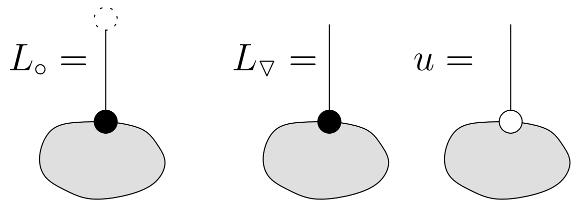



Figure 9: The series $L_\circ$, $L_\triangledown$ and $u$.

### 7.3.2 Families of Motzkin paths

In order to perform the enumeration of mobiles, we first need to compute some series related to Motzkin paths. We consider paths starting at position 0 and with steps 0, +1, and −1. A bridge is a path ending at position 0, and an excursion is a bridge that stays nonnegative. We denote respectively by $E$, $B$, and $B^{(+1)}$ the ordinary generating functions of excursions, bridges, and paths ending at position +1. These series are in two variables, $t$ and $u$, which count respectively the number of steps 0 and steps −1. Decomposing these paths at their last passage at 0, we obtain the following equations:

$$\begin{aligned} E &= 1 + tE + uE^2, \\ B &= 1 + (t + 2uE)B, \\ B^{(+1)} &= EB. \end{aligned}$$

We also need a variant of the series of bridges. Let $\widehat{B}(t,u)$ be the generating function of bridges counted with a weight divided by the total number of steps[4]. It is easily seen that $\widehat{B}$ is also the series of excursions counted with a weight divided by the number of returns to 0. Hence we have

$$\widehat{B}(t,u) = \log \frac{1}{1 - (t + uE(t,u))} = \log E(t,u).$$

### 7.3.3 Counting families of mobiles

We now compute the generating functions of some classes of mobiles. All the series considered here are exponential in the variable $s$, which counts the number of half-edges (not counting the legs), and ordinary in the variables $x$ and $y$, which count respectively the number of white and black vertices. Following [7], we define (see Figure 9):

[4]Formally, $\hat{B}$ is obtained from the series expansion of $B$ by the substitution $u^k t^\ell \to \frac{u^k t^\ell}{2k+\ell}$

- $L_\circ(x, y, s)$ the series counting mobiles rooted at an univalent white vertex, which is not counted in the series.
- $L_\triangledown(x, y, s)$ the series counting mobiles where an additional edge is attached to some black vertex. This edge plays the role of a root and is counted in the series (multiplication by $s^2$).
- $u(x, y, s)$ the series counting mobiles where an additional edge is attached to some white vertex. This edge plays the role of a root; it is counted in the series (multiplication by $s^2$) and it has no vertex at its other extremity.

Observe that one sees exactly a Motzkin bridge when walking around a black vertex (starting at a given position) and interpreting white vertices, black vertices, and legs respectively as steps $-1$, steps 0, and steps $+1$. Thus a mobile in each type we consider is naturally decomposed as a Motzkin bridge where each step is substituted by a mobile. Precisely we obtain the following equations (see [7]):

$$L_\circ(x, y, s) \;=\; ys^2 B^{(+1)}\Big(L_\triangledown(x, y, s), u(x, y, s)\Big), \tag{26}$$

$$L_\triangledown(x, y, s) \;=\; ys^2 B\Big(L_\triangledown(x, y, s), u(x, y, s)\Big). \tag{27}$$

Moreover, one has

$$u(x, y, s) \;=\; \frac{s^2 x}{1 - L_\circ(x, y, s)}. \tag{28}$$

All the series involved in the previous equations can be expressed in terms of the two simple algebraic series $\beta_1 = \beta_1(x, y, s)$ and $\beta_2 = \beta_2(x, y, s)$ specified by the equation system:

$$\left\{ \begin{array}{lcl} \beta_1 & = & xs^2 + \beta_1^2 + 2\beta_1\beta_2, \\ \beta_2 & = & ys^2 + \beta_2^2 + 2\beta_1\beta_2. \end{array} \right. \tag{29}$$

Then, elementary algebraic manipulations imply

$$usE\Big(L_\triangledown(x, y, s), u(x, y, s)\Big) \;=\; \beta_1, \tag{30}$$

$$L_\triangledown(x, y, s) \;=\; \beta_2, \tag{31}$$

$$E\Big(L_\triangledown(x, y, s), u(x, y, s)\Big) \;=\; \frac{1}{1 - \beta_1 - \beta_2}, \tag{32}$$

$$\frac{1}{1 - L_\circ\Big(L_\triangledown(x, y, s), u(x, y, s)\Big)} \;=\; \frac{\beta_1(1 - \beta_1 - \beta_2)}{s^2 x}. \tag{33}$$

#### 7.3.4 Counting mobiles

The enumeration of (unrooted) mobiles is obtained—from the dissymmetry theorem—by counting mobiles pointed in several ways. We introduce the following series in the three variables $x$, $y$, $s$:

- $T_\bullet$ ($T_\circ$) is the series of mobiles with a distinguished black vertex (white vertex, respectively).

- $T_{\bullet-\bullet}$ ($T_{\bullet-\circ}$) is the series of mobiles with a distinguished edge of type $\bullet - \bullet$ ($\bullet - \circ$, respectively).

- $T$ is the generating function of all mobiles.

The dissymetry theorem implies

$$T(x,y,s) = T_\bullet(x,y,s) + T_\circ(x,y,s) - T_{\bullet-\bullet}(x,y,s) - T_{\bullet-\circ}(x,y,s). \tag{34}$$

Moreover, each of the four series above can be expressed in terms of $\beta_1$ and $\beta_2$, thanks to a decomposition at the pointed vertex or edge:

$$\begin{aligned}
T_\bullet &= y\widehat{B}\Big(L_\nabla(x,y,s), u(x,y,s)\Big) = y \log \frac{1}{1-\beta_1-\beta_2} \\
T_\circ &= x \log \frac{1}{1 - L_\circ\Big(L_\nabla(x,y,s), u(x,y,s)\Big)} = x \log \frac{\beta_1(1-\beta_1-\beta_2)}{s^2 x} \\
T_{\bullet-\bullet} &= \frac{1}{2s^2} L_\nabla\Big(L_\nabla(x,y,s), u(x,y,s)\Big)^2 = \frac{1}{2s^2}\beta_2^2 \\
T_{\bullet-\circ} &= \frac{u}{s^2} L_\circ\Big(L_\nabla(x,y,s), u(x,y,s)\Big) = \frac{\beta_1\beta_2}{s^2}
\end{aligned}$$

In view of Theorem 7.2 and Equation (34), we obtain (taking $y=1$):

**Lemma 7.1.** *The series $M'(x,s)$ counting vertex-pointed maps w.r.t. vertices (except one) and half-edges satisfies the analytic expression*

$$M'(x,s) = T(x,1,s) = \log \frac{1}{1-\beta_1-\beta_2} + x \log \frac{\beta_1(1-\beta_1-\beta_2)}{s^2 x} - \frac{\beta_2}{2s^2} + \frac{1}{2}, \tag{35}$$

*where $\beta_1$ and $\beta_2$ are specified by* (29) *(with $y=1$).*

## 7.4 Expression for the series of vertex-pointed 2-connected maps

In this section we take advantage of the block-decomposition to relate the enumeration of vertex-pointed maps to the enumeration of vertex-pointed 2-connected maps. The strategy here differs from the one for graphs (from 2-connected to connected, Section 5.2) in two points. Firstly we use the decomposition in a top-to-bottom manner, i.e., we *extract* (instead of expand) the enumeration of vertex-pointed 2-connected maps from the enumeration of vertex-pointed maps obtained in Section 7.3. Secondly, the tree associated here with the decomposition is quite different from the Bv-tree: it takes the embedding into account (the corners play the role of vertices) and it is "completely incident" to a specific pointed vertex (similarly as the restricted RMT-tree in Section 5.3.3, which applies to 2-connected graphs).

### 7.4.1 From rooted maps to rooted 2-connected maps

Let us first briefly review how a top-to-bottom use of the block-decomposition combined with suitable algebraic manipulations makes it possible to count rooted 2-connected maps from rooted maps, as demonstrated by Tutte [27]. Let $\overrightarrow{M}(x,s)$ and $\overrightarrow{L}(x,t)$ be the series counting rooted maps and rooted 2-connected maps w.r.t. vertices (unlabelled) and half-edges (labelled). Schaeffer [24] has shown in a bijective way that

$$\overrightarrow{M}(x,s) = \frac{1}{s^4 x}\beta_1\beta_2(1-2\beta_1-2\beta_2)-1, \tag{36}$$

where $\beta_1 = \beta_1(x,s)$ and $\beta_2 = \beta_2(x,s)$ are specified by the system

$$\beta_1 = xs^2 + \beta_1^2 + 2\beta_1\beta_2, \qquad \beta_2 = s^2 + \beta_2^2 + 2\beta_1\beta_2. \tag{37}$$

Given a rooted map $\mu$, its 2-connected core is the maximal 2-connected submap of $\mu$ that contains the root. Clearly, all maps with a given core $\kappa$ are obtained by inserting at each corner $c$ of $\kappa$ either a rooted map $\mu_c$ or nothing. Hence $\overrightarrow{M}(x,s)$ and $\overrightarrow{L}(x,t)$ are related by the equation

$$\overrightarrow{M}(x,s) = \overrightarrow{L}(x, s\cdot(1+\overrightarrow{M}(x,s))).$$

In other words,

$$\overrightarrow{M}(x,s) = \overrightarrow{L}(x,t), \quad \text{where} \;\; t = s\cdot(1+\overrightarrow{M}(x,s)).$$

In view of (36), we obtain

$$t = \frac{1}{s^3 x}\beta_1\beta_2(1-2\beta_1-2\beta_2),$$

hence

$$xt^2 = \frac{1}{xs^6}\beta_1^2\beta_2^2(1-2\beta_1-2\beta_2)^2 = \frac{\beta_1^2}{xs^2}\frac{\beta_2^2(1-2\beta_1-2\beta_2)^2}{s^4} = \frac{\beta_1}{1-\beta_1-2\beta_2}\left(\frac{1-2\beta_1-2\beta_2}{1-\beta_2-2\beta_1}\right)^2,$$

and

$$t^2 = \frac{1}{x^2s^6}\beta_1^2\beta_2^2(1-2\beta_1-2\beta_2)^2 = \frac{\beta_2^2}{s^2}\frac{\beta_1^2(1-2\beta_1-2\beta_2)^2}{x^2s^4} = \frac{\beta_2}{1-\beta_2-2\beta_1}\left(\frac{1-2\beta_1-2\beta_2}{1-\beta_1-2\beta_2}\right)^2.$$

Equivalently:

$$xt^2 = \eta_1(1-\eta_2)^2, \qquad t^2 = \eta_2(1-\eta_1)^2,$$

where

$$\eta_1 := \frac{\beta_1}{1-\beta_1-2\beta_2}, \qquad \eta_2 := \frac{\beta_2}{1-\beta_2-2\beta_1}. \tag{38}$$

The equation-system (38) is readily inverted, giving the following expressions for $\beta_1$ and $\beta_2$ in terms of $\eta_1$ and $\eta_2$:

$$\beta_1 := \frac{\eta_1(1-\eta_2)}{1+\eta_1+\eta_2-3\eta_1\eta_2}, \qquad \beta_2 := \frac{\eta_2(1-\eta_1)}{1+\eta_1+\eta_2-3\eta_1\eta_2}. \tag{39}$$

Rearranging (36), we get the following expression for $\overrightarrow{M}$ involving only $\beta_1$ and $\beta_2$:

$$\overrightarrow{M}(x,s) = \frac{1-2\beta_1-2\beta_2}{(1-\beta_1-2\beta_2)(1-\beta_2-2\beta_1)} - 1, \tag{40}$$

hence $\overrightarrow{L}(x,t) = \overrightarrow{M}(x,s)$ also satisfies this expression. Replacing $\beta_1$ and $\beta_2$ by their expression in (39), we obtain the following expression for $\overrightarrow{L}(x,t)$ in terms of $\eta_1$ and $\eta_2$:

$$\overrightarrow{L}(x,t) = \eta_1 + \eta_2 - 3\eta_1\eta_2, \tag{41}$$

where $\eta_1 = \eta_1(x,t)$ and $\eta_2 = \eta_2(x,t)$ are specified by the system

$$\eta_1 = xt^2(1-\eta_2)^{-2}, \qquad \eta_2 = t^2(1-\eta_1)^{-2}. \tag{42}$$

More generally, if two series $F(x,s)$ and $G(x,t)$ are related by the equation

$$F(x,s) = G(x,t) \quad \text{ where } \; t = s\cdot(1+\overrightarrow{M}(x,s)),$$

and if $F(x,s)$ has an expression in terms of $\beta_1$, $\beta_2$, and $x$, then one gets an expression for $G(x,t)$ in terms of $\eta_1$, $\eta_2$, and $x$ (replacing each occurence of $\beta_1$ or $\beta_2$ according to (39)). This change of variable rule will be useful later on.

### 7.4.2 The tree associated to a vertex-pointed map

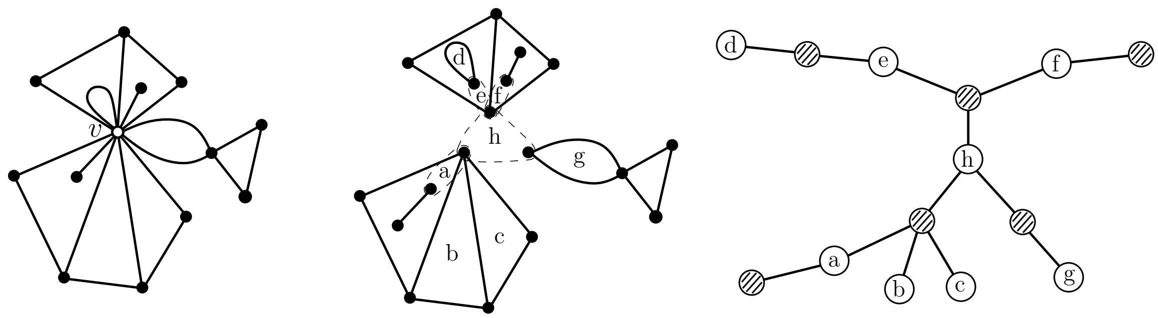
Let $\mu$ be a vertex-pointed map (as usual, on the sphere), with $v$ the pointed vertex. Let $\mathfrak{B}_v(\mu)$ be the set of (embedded) 2-connected blocks of $\mu$ that contain the vertex $v$, and let $\mathfrak{F}_v(\mu)$ be the set of faces of $\mu$ that are incident to $v$. A face $f \in \mathfrak{F}_v(\mu)$ is said to be incident to a block $b \in \mathfrak{B}_v(\mu)$ if there exists a corner $c$ of $\mu$ incident to $v$, such that the face incident to $c$ is $f$ and at least one side of $c$ belongs to $b$. Call $T$ the bipartite graph with vertex-set $\mathfrak{F}_v(\mu) \cup \mathfrak{B}_v(\mu)$ and edge-set corresponding to the incidences faces/blocks. From the Jordan curve theorem and an inductive argument (e.g., on the number of blocks attached at $v$), it is easily checked that $T$ is a tree, see Figure 10.

### 7.4.3 From vertex-pointed maps to vertex-pointed 2-connected maps

We call respectively ${M'}_B$, ${M'}_f$, and ${M'}_{Bf}$ the series counting vertex-pointed maps where, respectively, a block-node of $T$ is marked, a face-node of $T$ is marked, and an edge of $T$ is marked. By the dissymmetry theorem (or simply, by the fact that each tree has one more vertex than edge), the series $M'$ counting vertex-pointed maps satisfies

$$M' = {M'}_B + {M'}_f - {M'}_{Bf}. \tag{43}$$

As detailed next, the series $M'$ (obtained in Section 7.3), and the series ${M'}_f$ and ${M'}_{Bf}$ have explicit expressions; hence Equation (43) will yield an expression for ${M'}_B$, and thus (by the change of variable rule) an expression for the series $L'$ counting vertex-pointed 2-connected maps.




(a) (b) (c)

Figure 10: (a) A vertex-pointed map, (b) decomposed according to the blocks and faces incident to the pointed vertex $v$, (c) the tree resulting from the decomposition (a shaded node for each block incident to $v$, a letter-node for each face incident to $v$).

Consider a map $\mu$ counted by ${M'}_f$, i.e., a map with a distinguished incidence face-vertex $(f, v)$. As $\mu$ might not be 2-connected, the incidence $(f, v)$ can be realised by several corners. Let $c_1, \ldots, c_k$ be these corners cyclically ordered in ccw order around $v$, and let $\mu_1, \ldots, \mu_k$ be the components of $\mu$ delimited between successive corners, each of these maps being naturally rooted at the corner of separation from the other components. Clearly the maps $\mu_i$ are exactly constrained to have the incidence root-face/root-vertex realised by a unique corner, which is their root corner. Such rooted maps are said to have a simple root corner and their series is denoted by $S$. From the above discussion and the computation rule associated with the cycle construction (i.e., $\mathcal{C} = \mathrm{Cyc}(\mathcal{A}) \Rightarrow C(x) = \log(1/(1-A)))$ we get

$$ {M'}_f = \log\left(\frac{1}{1-S}\right). $$

Moreover the series $S$ is easy to obtain from $\overrightarrow{M}$. Indeed each rooted map is obtained from a rooted map with a single corner where one may insert another rooted map in the root corner; therefore

$$ \overrightarrow{M} = S \cdot (1 + \overrightarrow{M}), $$

i.e., $S = \overrightarrow{M}/(1+\overrightarrow{M})$. Hence

$$ {M'}_f = \log(1 + \overrightarrow{M}). \tag{44} $$

Finding an expression for ${M'}_{Bf}$ in terms of $\overrightarrow{M}$ is easier. From Figure 10 one easily sees that marking an incidence face-vertex $(f, v)$ + a block incident to $(f, v)$ is the same as marking a corner of the map. Hence

$$ {M'}_{Bf} = \overrightarrow{M}. \tag{45} $$

Plugging the expression (40) of $\overrightarrow{M}$ into (44) and (45), we obtain explicit expressions for ${M'}_f$ and ${M'}_{Bf}$ in terms of $\{\beta_1, \beta_2, x\}$. We can also obtain from (35) (replacing $s^2$ by $\beta_2(1-\beta_2-2\beta_1)$) an expression for $M'$ in terms of $\{\beta_1, \beta_2, x\}$.

Therefore, we now have expressions for $M'$, ${M'}_f$ and ${M'}_{Bf}$ in terms of $\beta_1$, $\beta_2$, and $x$. Plugging these into ${M'}_B = M' - {M'}_f + {M'}_{Bf}$, we obtain:

$$\begin{aligned} {M'}_B &= \log\left(\frac{(1-\beta_1-2\beta_2)(1-\beta_2-2\beta_1)}{(1-2\beta_1-2\beta_2)(1-\beta_1-\beta_2)}\right) + x\log\left(\frac{1-\beta_1-\beta_2}{1-\beta_1-2\beta_2}\right) \\ &\quad + \frac{1-3\,\beta_1-2\,\beta_2}{2\,(1-\beta_1-2\,\beta_2)\,(1-\beta_2-2\,\beta_1)} - \frac{1}{2}, \end{aligned} \tag{46}$$

where $\beta_1$ and $\beta_2$ are specified by (37).

Clearly each map counted by ${M'}_B$ arises from a vertex-pointed 2-connected map (the marked 2-connected block) where a rooted connected map may be inserted in each corner:

$${M'}_B(x,s) = L'(x, s(1+\overrightarrow{M}(x,s))).$$

In other words,

$${M'}_B(x,s) = L'(x,t), \qquad \text{where} \quad t = s\cdot(1+\overrightarrow{M}(x,s)).$$

As described at the end of Section 7.4.1, to get an expression for $L'$ in terms of $\eta_1$, $\eta_2$, and $x$, we have to take the expression (46) of ${M'}_B(x,s)$, and then replace each occurence of $\beta_1$ and $\beta_2$ by their expressions given in (39). All simplifications done, we obtain the following expression for the series counting vertex-pointed 2-connected maps:

$$L'(x,t) = -\log(1-\eta_1\,\eta_2) + x\log\left(\frac{1-\eta_1\,\eta_2}{1-\eta_2}\right) + \eta_1 + \eta_2 - 3\eta_1\eta_2 - \frac{2\eta_1+\eta_2-3\eta_1\eta_2}{2(1-\eta_1)}, \tag{47}$$

where $\eta_1 = \eta_1(x,t)$ and $\eta_2 = \eta_2(x,t)$ are specified by (42).

## 7.5 Expression for the series of vertex-pointed 3-connected maps

Our aim in this section is to obtain an analytic specification for $K'$ from the one for $L'$. The strategy is to use the restricted RMT-tree in a top-to-bottom manner (extract the enumeration of 3-connected from the enumeration of 2-connected), while taking the embedding into account.

### 7.5.1 From rooted 2-connected to rooted 3-connected maps

Similarly as in Section 7.4, we first review how rooted 3-connected maps can be extracted from rooted 2-connected maps, following the approach of Mullin and Schellenberg [21]. We consider here networks that are *embedded* in the plane, with the two poles in the outer face. The series of networks, series-networks, parallel-networks, and polyhedral networks are still denoted by $D = D(x,y)$, $S = S(x,y)$, $P = P(x,y)$, $H = H(x,y)$ ($x$ marks vertices not incident to the root, $y$ marks edges different from the root) but in all this section the embedding is taken into account, i.e., we are counting maps. Notice that a network is obtained from a rooted 2-connected map with at least 2 edges as follows: distribute distinct blue labels on the vertices not incident to the root, distribute red labels on the edges different from the root, erase the root edge and erase the labels on the half-edges. Therefore

$$xt^2 D(x,t^2) = \overrightarrow{L}(x,t) - (1+x)t^2.$$

In other words (with $y = \sqrt{t}$)

$$D(x,y) = \frac{\eta_1 + \eta_2 - 3\eta_1\eta_2}{xy} - 1/x - 1, \tag{48}$$

where $\eta_1 = \eta_1(x,y)$ and $\eta_2 = \eta_2(x,y)$ are specified by

$$\eta_1 = xy(1-\eta_2)^{-2}, \qquad \eta_2 = y(1-\eta_1)^{-2}. \tag{49}$$

Next, Trakhtenbrot's decomposition for rooted 2-connected graphs is readily adapted so as to take the embedding into account; the main difference is that the components of a parallel network are ordered (say from left to right if the axis passing by the poles is viewed as vertical). We get the system (equivalent to the one in [21], where everything is formulated on quadrangulations):

$$\left\{ \begin{array}{rcl} D &=& y + S + P + H, \\ S &=& (D-S)\cdot x \cdot D, \\ P &=& (D-P)\cdot D, \\ H &=& \overrightarrow{K}(x, D(x,y)). \end{array} \right. \tag{50}$$

Hence

$$H = D - S - P - y = D - \frac{xD^2}{1+xD} - \frac{D^2}{1+D} - y,$$

which yields

$$\overrightarrow{K}(x,w) = w - \frac{xw^2}{1+xw} - \frac{w^2}{1+w} - y, \quad \text{where} \ \ w = D(x,y). \tag{51}$$

Replacing $y$ by $\eta_2(1-\eta_1)^2$ and $x$ by $\eta_1(1-\eta_2)^2/(\eta_2(1-\eta_1)^2)$ in the expression (48) of $D$, we get

$$w = \frac{\eta_2(1-\eta_1-\eta_2)}{(1-\eta_2)^2}.$$

and

$$xw = \frac{\eta_1(1-\eta_1-\eta_2)}{(1-\eta_1)^2}.$$

Equivalently, if we introduce

$$\gamma_1 = \frac{\eta_1}{1-\eta_1-\eta_2}, \quad \gamma_2 = \frac{\eta_2}{1-\eta_1-\eta_2}, \tag{52}$$

we have

$$xw = \frac{\gamma_1}{(1+\gamma_2)^2}, \qquad w = \frac{\gamma_2}{(1+\gamma_1)^2}. \tag{53}$$

Now we want to express $\overrightarrow{K}(x,w)$ in terms of $x$, $w$, $\gamma_1$ and $\gamma_2$. In view of Equation (51), this task is clearly equivalent to finding an expression for $y$ in terms of $x$, $w$, $\gamma_1$ and $\gamma_2$. Inverting the system (52), we get

$$\eta_1 = \frac{\gamma_1}{1+\gamma_1+\gamma_2}, \qquad \eta_2 = \frac{\gamma_2}{1+\gamma_1+\gamma_2}. \tag{54}$$

Substituting these expressions in $y = \eta_2(1-\eta_1)^2$, we obtain

$$y = \frac{\gamma_2(\gamma_2+1)^2}{(1+\gamma_1+\gamma_2)^3} = \frac{\gamma_1\gamma_2}{xw(1+\gamma_1+\gamma_2)^3}.$$

This yields the following expression for $\overrightarrow{K}(z,w)$ (see [21] for more details):

$$\overrightarrow{K}(z,w) = w - \frac{xw^2}{1+xw} - \frac{w^2}{1+w} - \frac{\gamma_1\gamma_2}{xw(1+\gamma_1+\gamma_2)^3}, \tag{55}$$

where $\gamma_1 = \gamma_1(x,w)$ and $\gamma_2 = \gamma_2(x,w)$ are specified by the system

$$\gamma_1 = xw(1+\gamma_2)^2, \qquad \gamma_2 = w(1+\gamma_1)^2. \tag{56}$$

More generally, if two series $F(x,y)$ and $G(x,w)$ are related by the equation

$$F(x,y) = G(x,w), \quad \text{where } w = D(x,y),$$

and if $F(x,y)$ has an expression in terms of $\{x,\eta_1,\eta_2\}$, then an expression for $G(x,w)$ in terms of $\{x,\gamma_1,\gamma_2\}$ is obtained upon replacing each occurence of $\eta_1$ or $\eta_2$ according to (54).

### 7.5.2 Getting an expression for $K'(x,w)$

To get an expression for the series $K'(x,w)$ counting vertex-pointed 3-connected maps, the restricted RMT-tree introduced in Section 4.4 proves very convenient; it keeps the calculations as simple as possible. Here we are dealing with maps, so we have to take the embedding into account; this means that the equations relating vertex-pointed 2-connected maps and vertex-pointed 3-connected maps are the same as for graphs, except that parallel components (M-nodes) are cyclically ordered for maps, due to the embedding. This yields the following adaptation—taking the embedding into account—of the subsystem of equations in Figure 6 related to vertex-pointed 2-connected graphs (here $V$ denotes the series counting vertex-pointed 2-connected maps with at least 3 edges, i.e., $V(x,y) = L'(x,\sqrt{y}) - xy - y/2 - xy^2/2$):

$$V = V_R + V_M + V_T - V_{R-M} - V_{R-T} - V_{M-T} - V_{T-T},$$

$$V_R = \frac{1}{2}x^2(D-S)^2D, \qquad V_M = x\,\mathrm{loga}_{\geq 3}(D-P), \qquad V_T = K'(x, D(x,y)),$$

where $\mathrm{loga}_{\geq 3}(Z) := \log(1/(1-Z)) - Z - Z^2/2$; and

$$V_{R-M} = x \cdot S \cdot P, \quad V_{R-T} = x \cdot S \cdot H, \quad V_{M-T} = x \cdot P \cdot H, \quad V_{T-T} = \frac{1}{2} x \cdot H^2.$$

Hence

$$K'(x,w) = V - V_R - V_M + V_{R-M} + V_{R-T} + V_{M-T} + V_{T-T}, \quad \text{where } w = D(x,y).$$

All the series on the right-hand side admit an explicit expression in terms of $\{x, \eta_1, \eta_2\}$. Hence $K'(x,w)$ also admits an expression in terms of $\{x, \eta_1, \eta_2\}$, which is turned into an expression in terms of $\{x, \gamma_1, \gamma_2\}$ when substituting each occurence of $\eta_1$ or $\eta_2$ by the corresponding expression given in (54). (In fact, it is better to allow the presence of $w = D(x,y)$ in all these expressions to have simpler forms, similarly as we have done to get an expression for $\overrightarrow{K}(x,w)$ in Section 7.5.) All calculations and simplifications done, we obtain the following analytic expression for $K'(x,w)$:

$$\begin{aligned} K'(x,w) \;\; = \;\; & -\ln\left(1 - \frac{\gamma_1\,\gamma_2}{(1+\gamma_1+\gamma_2)^2}\right) + x\ln\left(1 + \frac{\gamma_2\,(1+\gamma_2)}{(1+\gamma_1+\gamma_2)\,(1+\gamma_1)}\right) \\ & - \frac{1+\gamma_2/2+xw\,(1+\gamma_1)\,(1+\gamma_2)}{1+\gamma_1+\gamma_2} - \frac{3\gamma_1\,\gamma_2}{(1+\gamma_1+\gamma_2)^2} - \frac{\gamma_1\,\gamma_2\,(1+2\,x+2\,xw)}{2xw\,(1+\gamma_1+\gamma_2)^3} \\ & - \frac{1}{2x} + 1 + \frac{w}{2} + xw - \frac{xw^2}{2} + \frac{1}{2\,(1+xw)\,x} - x\ln(1+w)\,, \end{aligned}$$

where $\gamma_1 = \gamma_1(x,w)$ and $\gamma_2 = \gamma_2(x,w)$ are specified by (56).

## 7.6 Expression for the series of 3-connected planar graphs

As we have seen in Section 7.2, having an expression for the series $K'(x,w)$ (and $\overrightarrow{K}(x,w)$) allows us to obtain analytic expressions for the series $G_3(x,w)$, $G_3'(x,w)$, $\overrightarrow{G_3}(x,w)$ counting 3-connected planar graphs, which are the terminal series of the equation-system shown in Figure 7 (assuming we apply this generic equation-system to the family of planar graphs).

**Proposition 7.3 (expressions for the series counting 3-connected planar graphs).** *The series $\overrightarrow{G_3}(x,w)$, $G_3'(x,w)$, and $G_3(x,w)$ that count respectively rooted, vertex-pointed, and unrooted 3-connected planar graphs w.r.t. vertices and edges (recall that one vertex is discarded in $G_3'$, and two vertices and one edge are discarded in $\overrightarrow{G_3}$) satisfy the following analytic expressions (the one for $\overrightarrow{G_3}(x,w)$ was first given in [21], and alternative expressions and computation methods for $G_3(x,w)$ are given in [15]):*

$$\overrightarrow{G_3}(x,w) = \frac{1}{2}\left(w - \frac{xw^2}{1+xw} - \frac{w^2}{1+w} - \frac{\gamma_1\gamma_2}{xw(1+\gamma_1+\gamma_2)^3}\right), \tag{57}$$

$$
\begin{aligned}
G_3'(x,w) = &-\frac{1}{2}\ln\left(1-\frac{\gamma_1\,\gamma_2}{\left(1+\gamma_1+\gamma_2\right)^2}\right)+\frac{x}{2}\ln\left(1+\frac{\gamma_2\,\left(1+\gamma_2\right)}{\left(1+\gamma_1+\gamma_2\right)\left(1+\gamma_1\right)}\right)\\
&-\frac{1+\gamma_2/2+xw\left(1+\gamma_1\right)\left(1+\gamma_2\right)}{2(1+\gamma_1+\gamma_2)}-\frac{3\gamma_1\,\gamma_2}{2\left(1+\gamma_1+\gamma_2\right)^2}-\frac{\gamma_1\,\gamma_2\,\left(1+2\,x+2\,xw\right)}{4xw\left(1+\gamma_1+\gamma_2\right)^3}\\
&-\frac{1}{4x}+\frac{1}{2}+\frac{w}{4}+\frac{xw}{2}-\frac{xw^2}{4}+\frac{1}{4\left(1+xw\right)x}-\frac{x}{2}\ln\left(1+w\right), \qquad (58)
\end{aligned}
$$

$$
\begin{aligned}
G_3(x,w) = &\frac{x}{4}\Bigg(-2\ln\left(1-\frac{\gamma_1\gamma_2}{\left(1+\gamma_1+\gamma_2\right)^2}\right)+x\ln\left(1+\frac{\gamma_2\left(1+\gamma_2\right)}{\left(1+\gamma_1+\gamma_2\right)\left(1+\gamma_1\right)}\right)\\
&+\frac{1}{x}\ln\left(1+\frac{\gamma_1\left(1+\gamma_1\right)}{\left(1+\gamma_1+\gamma_2\right)\left(1+\gamma_2\right)}\right)\Bigg)\\
&+\frac{x}{4}\Bigg(-\frac{1}{2}-\frac{3}{2}\frac{1}{1+\gamma_1+\gamma_2}-\frac{w\left(1+x\right)\left(1+\gamma_1\right)\left(1+\gamma_2\right)}{1+\gamma_1+\gamma_2}-\frac{6\gamma_1\gamma_2}{\left(1+\gamma_1+\gamma_2\right)^2}\\
&-\frac{3}{2}\frac{\gamma_1\gamma_2\left(x+1+xw\right)}{xw\left(1+\gamma_1+\gamma_2\right)^3}\Bigg)\\
&+\frac{x}{4}\left(-\frac{1}{2}xw^2+xw+w+2-x\ln\left(1+w\right)-\frac{1}{x}\ln\left(1+xw\right)\right), \qquad (59)
\end{aligned}
$$

*where $\gamma_1=\gamma_1(x,w)$ and $\gamma_2=\gamma_2(x,w)$ are specified by*

$$\gamma_1 = xw(1+\gamma_2)^2, \quad \gamma_2 = w(1+\gamma_1)^2.$$

*Proof.* According to Whitney's theorem, $\overrightarrow{G_3}=\overrightarrow{K}/2$, ${G_3}'=K'/2$, $G_3=K/2$, where $\overrightarrow{K}$, $K'$ and $K$ are the series counting rooted, vertex-pointed, and unrooted 3-connected maps. We have already showed how to compute the expressions of $\overrightarrow{K}$ and $K'$ in Section 7.5.1 and Section 7.5.2, respectively. Moreover, as discussed in Section 7.2, the series $K$ satisfies

$$K(x,w)=\frac{1}{2}x\left(K'(x,w)-\frac{1}{2}xw\overrightarrow{K}(x,w)+K'(1/x,xw)\right).$$

Notice that the effect on $(\gamma_1,\gamma_2)$ of replacing $(x,w)$ by $(1/x,xw)$ is simply to swap $\gamma_1$ and $\gamma_2$. All simplifications done, one obtains the expression of $G_3$ given above. $\square$

The expressions of Proposition 7.3, together with Theorem 6.1, finish the proof of Theorem 7.1.

## Acknowledgements.

The authors thank Pierre Leroux and Konstantinos Panagiotou for very interesting discussions.

# References


[1] E. Bender, Z. Gao, and N. Wormald. The number of labeled 2-connected planar graphs. *Electron. J. Combin.*, 9:1–13, 2002.

[2] F. Bergeron, G. Labelle, and P. Leroux. *Combinatorial Species and Tree-like Structures*. Cambridge University Press, 1997.

[3] M. Bodirsky, É. Fusy, M. Kang, and S. Vigerske. Enumeration and asymptotic properties of unlabeled outerplanar graphs. *Elect. J. Comb.*, 14, R66, Sep 2007.

[4] M. Bodirsky, O. Giménez, M. Kang, and M. Noy. Enumeration and limit laws for series-parallel graphs. *Eur. J. Comb.*, 28(8):2091–2105, 2007.

[5] M. Bodirsky, C. Groepl, and M. Kang. Generating labeled planar graphs uniformly at random. *Theoretical Computer Science*, 379:377–386, 2007.

[6] M. Bodirsky, M. Kang, M. Löffler, and C. McDiarmid. Random cubic planar graphs. *Random Structures and Algorithms*, 30:78–94, 2007.

[7] J. Bouttier, P. Di Francesco, and E. Guitter. Planar maps as labeled mobiles. *Electr. J. Comb.*, 11(1), 2004.

[8] R. Diestel. *Graph Theory*. Springer-Verlag, 3rd edition edition, 2005.

[9] P. Flajolet and R. Sedgewick. Analytic combinatorics. Preliminary version available at `http://algo.inria.fr/flajolet/Publications`.

[10] É. Fusy. Quadratic exact size and linear approximate size random generation of planar graphs. *Discrete Mathematics and Theoretical Computer Science*, AD:125–138, 2005.

[11] É. Fusy, D. Poulalhon, and G. Schaeffer. Dissections, orientations, and trees, with applications to optimal mesh encoding and to random sampling. *Transactions on Algorithms*, 4(2):Art. 19, April 2008.

[12] A. Gagarin, G. Labelle, and P. Leroux. Counting unlabelled toroidal graphs with no $k_{3,3}$-subdivisions. *Advances in Applied Mathematics*, 39:51–74, 2007.

[13] A. Gagarin, G. Labelle, P. Leroux, and T. Walsh. Two-connected graphs with prescribed three-connected components. arXiv:0712.1869, 2007.

[14] O. Giménez, M. Noy, and J. Rué. Graph classes with given 3-connected components: asymptotic counting and critical phenomena. *Electronic Notes in Discrete Mathematics*, 29:521–529, 2007.

[15] Omer Giménez and Marc Noy. Asymptotic enumeration and limit laws of planar graphs. arXiv:math/0501269, 2005.

[16] F. Harary and E. Palmer. *Graphical Enumeration*. Academic Press, New York, 1973.

[17] J. E. Hopcroft and R. E. Tarjan. Dividing a graph into triconnected components. *SIAM J. Comput.*, 2(3):135–158, 1973.

[18] P. Leroux. Methoden der Anzahlbestimmung fur einige Klassen von Graphen. *Bayreuther Mathematische Schriften*, 26:1–36, 1988.

[19] C. McDiarmid, A. Steger, and D. Welsh. Random planar graphs. *J. Comb. Theory, Ser. B*, 93:187–205, 2005.

[20] B. Mohar and C. Thomassen. *Graphs on Surfaces.* Johns Hopkins University Press, 2001.

[21] R.C. Mullin and P.J. Schellenberg. The enumeration of c-nets via quadrangulations. *J. Combin. Theory*, 4:259–276, 1968.

[22] R. Otter. The number of trees. *Annals of Mathematics*, 49(3):583–599, 1948.

[23] R.W. Robinson. Enumeration of non-separable graphs. *J. Comb. Theory*, 9:327–356, 1970.

[24] G. Schaeffer. Bijective census and random generation of Eulerian planar maps with prescribed vertex degrees. *Electron. J. Combin.*, 4(1):# 20, 14 pp., 1997.

[25] B. Shoilekova. Unlabelled enumeration of cacti graphs. Manuscript, 2007.

[26] B. A. Trakhtenbrot. Towards a theory of non–repeating contact schemes (russian). In *Trudi Mat. Inst. Akad. Nauk SSSR 51*, pages 226–269, 1958.

[27] W. T. Tutte. A census of planar maps. *Canad. J. Math.*, 15:249–271, 1963.

[28] W.T. Tutte. *Connectivity in graphs.* Oxford U.P, 1966.

[29] T. R. S. Walsh. Counting labelled three-connected and homeomorphically irreducible two-connected graphs. *J. Combin. Theory*, 32(B):1–11, 1982.

[30] T. R. S. Walsh. Counting unlabelled three-connected and homeomorphically irreducible two-connected graphs. *J. Combin. Theory*, 32(B):12–32, 1982.

[31] H. Whitney. Congruent graphs and the connectivity of graphs. *Amer. J. Math.*, 54:150–168, 1932.

[32] H. Whitney. 2-isomorphic graphs. *Amer. J. Math.*, 54:245–254, 1933.