Analytic Combinatorics— A Calculus of Discrete Structures

Philippe Flajolet*

Abstract

The efficiency of many discrete algorithms crucially depends on quantifying properties of large structured combinatorial configurations. We survey methods of *analytic combinatorics* that are simply based on the idea of associating numbers to atomic elements that compose combinatorial structures, then examining the geometry of the resulting functions. In this way, an operational calculus of discrete structures emerges. Applications to basic algorithms, data structures, and the theory of random discrete structures are outlined.

1 Algorithms and Random Structures

A prime factor in choosing the best algorithm for a given computational task is efficiency with respect to the resources consumed, for instance, auxiliary storage, execution time, amount of communication needed. For a given algorithm $\mathcal{A}$, such a complexity measure being fixed, what is of interest is the relation

$$\text{Size of the problem instance } (n) \longrightarrow \text{Complexity of the algorithm } (C),$$

which serves to define the complexity function $C(n) \equiv C_{\mathcal{A}}(n)$ of algorithm $\mathcal{A}$. Precisely, this complexity function can be specified in several ways.

(*i*) *Worst-case analysis* takes $C(n)$ to be the maximum of C over all inputs of size n. This corresponds to a pessimistic scenario, one which is of relevance in critical systems and real-time computing.

(*ii*) *Average-case analysis* takes $C(n)$ to be the expected value (average) of C over inputs of size n. The aim is to capture the "typical" cost of a computational task observed when the algorithm is repeatedly applied to various kinds of data.

(*iii*) *Probabilistic analysis* takes $C(n)$ to be an indicator of the most likely values of C. Its more general aim is to obtain fine estimates on the probability distribution of C, beyond average-case analysis.

*Algorithms Project, INRIA-Rocquencourt, F78153 Le Chesnay, France.

In cases (*ii*) and (*iii*), a *model* supposed to reflect the distribution of inputs is assumed.

In the period 1945–1970, both worst-case and average-case were considered relevant. For instance, in their report to the U.S. Army Ordnance Department in 1946, Burk, Goldstine, and von Neumann [5] conduct a thorough discussion of the comparative merits of binary versus decimal computer arithmetics, proving in the course of the discussion that *"for a sum of binary words, each of length n, the length of the largest carry sequence is on the average not in excess of $\log_2 n$"*. (Their conclusion, based on this and other analyses, is worthy of note: *"In spite of the long-standing tradition of building digital machines in the decimal system, we feel strongly in favor of the binary system for our device."*)

Knuth in the first three volumes of *The Art of Computer Programming (TAOCP)* [34] published in the period 1968–1973 brilliantly demonstrated many basic algorithms to be amenable to a thorough mathematical analysis, leading to a highly valuable classification of their merits based on the average-case criterion. Knuth proved the point that precise analysis is both feasible and fruitful, but his attention to detail was viewed as excessive by many. A large fraction of the theoretical computer science research community reverted to worst-case analysis based on simple tools from computational complexity. In all too many cases, this has resulted in an excess of its own, with works culminating in teratological constructions both devoid of mathmematical simplicity and elegance and bearing little relevance to the practice of computing. At the same time, average-case and probabilistic analyses have proven to have spectacular impact on the practice of computing, from Hoare's use of randomization in quicksort in 1960 to simulated annealing to Markov models for web search: see Rabin's historic paper [44] and the book by Motwani and Raghavan [37] for an attractive introduction. Such algorithms, being based on (pseudo) random choices, offer a framework in which modelling is both mathematically pleasing (simply consider the choices as being perfectly random) and practically adequate (since we know of very good ways to emulate perfect randomness). In addition,

the algorithms being based on randomness, their design and justification inherently require a non-negligible amount of probabilistic analysis. Another equally important motivation for departing from the worst-case scenario has been Sedgewick's influential books, *Algorithms* (see, e.g., [46]). His algorithms aim at being at the same time conceptually simple and surprisingly efficient; their design is based on a fascinating mix of experimental analysis (supported by illuminating visual traces) and Knuthian analysis (purposely kept hidden to the reader). In Sedgewick's own terms [46, p. 27]: *"Analysis is the key to be able to understand algorithms sufficiently well that we can apply them effectively to practical problems."*

This paper offers a partial review of methods principally elaborated in the past decade, whose aim is to develop a calculus meant to "measure" properties of large combinatorial objects. A fair fragment of this research is motivated by algorithmic efficiency. At the same time, it is deeply rooted in a mathematical tradition that goes back to Pólya in the 1930's; cf [42] and [28, 43]. As we shall see, the basic ideas are simple, though some amount of [easy] complex analysis (mostly going back to Cauchy) is needed at some stage. Signal processing depends on Fourier transforms, coding theory and cryptography make a deep usage of finite fields and number fields; there is no decent reason why the metric study of algorithms could be kept "free" of mathematics of sorts. As we propose to demonstrate, going from the discrete to the continuous realm, and back, provides a wealth of metric properties of discrete structures.

The prime reference for this survey is the forthcoming book *Analytic Combinatorics* [21]. For analysis of algorithms *per se* examined in a similar spirit, we refer to treatises by Hofri [30], Mahmoud [36], Szpankowski [48], as well as our own elementary *Introduction* [47].

Plan. A skeleton of the theory of analytic combinatorics starts with core constructions together with their translations into generating functions (§2), continues with a geometric interpretation of generating functions as transformations of the complex plane (§3), and concludes with perturbative methods that yield distributional information on properties of large random strctures (§5). Boltzmann models that graft naturally on the core theory are discussed in §4. A sample of important results in average-case and probabilistic analysis of algorithms and data structures is discussed in §6.

2 The Formal Game of Constructions

Our first goal is to develop an effective theory of (elementary) combinatorial enumerations, which is based on *constructions*.

We regard combinatorial objects as being formed of *atoms*. The *size* (noted $|\cdot|$) of an object is the number of atoms it comprises. For instance, a binary string is composed of letters, each an atom of $\{\bullet, \circ\} \equiv \{\mathsf{a}, \mathsf{b}\}$, in which case size is the usual length; a tree is made of nodes that are also atoms (e.g., $\bullet$ for a generic node, $\downarrow\bullet$ for the root node, $\circ$ for a leaf), and so on. A *combinatorial class* $\mathcal{C}$ is a finite or denumerable collection of combinatorial objects, and we let $\mathcal{C}_n$ represent the subset (assumed to be finite) of objects that have size n. We denote by $\mathcal{Z} = \{\bullet\}$ the "atomic" class comprised of a single atomic object of size 1 and by $\mathbf{1} = \{\epsilon\}$ the neutral class composed of a single object of size 0 (analogous to the "empty" word). Since we are interested in enumerative properties, we freely identify two classes $\mathcal{C}, \mathcal{D}$ that are isomorphic, in the sense that there exists a size-preserving bijection between them. In this way, details in the actual representation of objects become immaterial[1]. Given a class $\mathcal{C}$, we systematically denote by C_n the number of objects in $\mathcal{C}$ having size n.

A *(combinatorial) construction* is an operation that associates a combinatorial class to one or several classes. The core set of constructions, which we consider here are

$$+, \quad \times, \quad \textsc{Seq}, \quad \textsc{PSet}, \quad \textsc{MSet}, \quad \textsc{Cyc}, \tag{2.1}$$

corresponding respectively to disjoint union (union of disjoint copies, systematically obtained by using different "colours"), cartesian product (the formation of ordered pairs), the building in all possible ways of finite sequences ("lists"), multisets ("heaps", sets with repetitions allowed), plain sets (multiplicities are 0 or 1), and directed cycles. Subscripts to constructions indicate a corresponding restriction on the number of components, e.g., $\textsc{Set}_{\geq 5}$ denotes sets of at least 5 elements. This gives us a systematic way to express *specifications*. For instance, the equations

$$\mathcal{W} = \textsc{Seq}(\mathcal{Z} + \mathcal{Z}), \qquad \mathcal{G} = \mathcal{Z} \times \textsc{Seq}(\mathcal{G}),$$

specify the class of binary words and the class of rooted plane trees respectively. Nonplane trees are described by $\mathcal{H} = \mathcal{Z} \times \textsc{MSet}(\mathcal{H})$, since subtrees stemming from the root form a multiset (order does not count; repetitions are allowed). As illustrated by such examples, both recursive and nonrecursive (or "iterative') specifications are allowed.

It is a programing exercise to describe familiar combinatorial structures in the language of constructions. Examples from [21] include words containing or excluding a pattern (as a factor or subword), degree-

[1] Readers interested in foundational issues should study Joyal's elegant theory of *species* exposed in [4], which places itself within the framework of category theory.

constrained trees, trees of bounded height (either balanced or not), functional graphs, necklaces, lattice paths of various sorts, noncrossing geometric graphs in the plane, coupon collector and birthday sequences, to name a few. This framework can be viewed as an extension of context-free grammars enriched by commutation/association rules. In particular, all languages defined by regular expressions and finite-state automata as well as derivation trees of context-free languages are *a priori* specifiable.

DEFINITION 2.1. *For a class $\mathcal{C}$, with counting sequence (C_n), the* ordinary generating function *(OGF) is the formal power series,*

$$C(z) := \sum_{n\geq 0} C_n z^n = \sum_{\gamma\in\mathcal{C}} z^{|\gamma|}. \tag{2.2}$$

Generating functions (GFs) are the central objects of the theory, rather than a mere artefact to solve recurrences, as it is still often believed. Conceptually, by the second form in (2.2), the GF of a class is nothing but the combinatorial class itself, after internal structural details are "forgotten":

atom (letter, node)	$\bullet, \circ, \mathsf{a}, \mathsf{b}, \ldots \mapsto z$
object (word, tree)	$\gamma \mapsto z^{\|\gamma\|} = z^n$
class (language, tree class)	$\mathcal{C} \mapsto C(z) \equiv \sum_{\gamma\in\mathcal{C}} z^{\|\gamma\|}.$

(This vision originates in pioneering works of Schützenberger, Foata, Rota, Stanley, Joyal, Goulden, Jackson, and a few others.) Constructions are found to admit of systematic translations as operators over generating functions according to rules summarized by the symbolic dictionary sketched in Fig. 1.

THEOREM 2.1. (SYMBOLIC METHOD) *For specifiable classes, generating functions are automatically computable from specifications by the symbolic dictionary of Fig. 1.*

Proof. By simple algebra and the fact that GFs are a reduced form of the combinatorial classes themselves. For instance, if $\mathcal{C} = \mathcal{A} \times \mathcal{B}$,

$$C(z) \equiv \sum_{(\alpha,\beta)\in\mathcal{C}} z^{|\alpha|+|\beta|} = \sum_{\alpha\in\mathcal{A}} z^{|\alpha|} \cdot \sum_{\beta\in\mathcal{B}} z^{|\beta|} \equiv A(z)\cdot B(z),$$

which yields the cartesian product rule. □

The symbolic method summarizes the basic laws of counting in a way that parallels Wilf's *Generatingfunctionology* [55]. It permits us to derive a large number of explicit counting results. Typically, for plane trees, we find

$$\mathcal{G} = \mathcal{Z} \times \textsc{Seq}(\mathcal{G}) \quad \Longrightarrow \quad G(z) = \frac{z}{1-G(z)}$$
$$\rightsquigarrow \quad G(z) = \frac{1-\sqrt{1-4z}}{2} \quad \rightsquigarrow \quad G_n = \frac{1}{n}\binom{2n-2}{n-1}.$$

Construction	*Operation on* GF*s*
$+$	$+$
$\times, \star$	$\times$
SEQ	$\frac{1}{1-\cdot}$
SET, MSET, PSET	$\exp, \mathrm{Exp}, \overline{\mathrm{Exp}}$
CYC	$\log, \mathrm{Log}$

Figure 1: A table summarizing the translation of constructions as operations over generating functions (GFs): $\mathrm{Exp}, \overline{\mathrm{Exp}}, \mathrm{Log}$ known as Pólya operators are variants of the usual $\exp, \log$.

THEOREM 2.2. (COUNTING & RANDOM GENERATION) *Consider a combinatorial class $\mathcal{C}$, which is specifiable by the constructions of* (2.1)*. Then, the counts $\{C_j\}_{j=0}^n$ can be computed in $O(n^{1+\epsilon})$ arithmetic operations. In addition, it is possible to draw an element of size n uniformly at random in $O(n \log n)$ arithmetic operations in the worst case.*

Proof. The equations deriving from Theorem 2.1 and Fig. 1 are normalized as a quadratic differential (Riccati) system. Induced convolutions then provide an elementary algorithm that determines the first n terms of the counting sequence (C_n) in $O(n^2)$ arithmetic operations [20, 24], the improvement in complexity being due to van der Hoeven [52]. Random generation uses a recursive method that extends those of [39], combined with a so-called boustrophedonic search based on properties of Friedman-Knuth recurrences. (Note: for anything specifiable in (2.1), a bound of the form $C_n < K \cdot L^n$ holds, see below, so that the corresponding boolean complexities are $O(n^{2+\epsilon})$.) □

Labelled classes. There is a parallel universe where atoms composing objects are now taken to be all different, say they bear distinct integer labels. For instance, a permutation 4 1 3 5 2 is a labelled object of size 5 and the class $\mathcal{P}$ of all permutations is a labelled class. One defines classically a labelled product, where labels are distributed in an order-preserving fashion on components, and, from there, labelled sequences, sets, and cycles. Labelled trees, permutations with cycle constraints, allocations of (distinguishable) balls into urns, and mappings of a finite set to itself are examples of labelled classes that are specifiable. One then operates with the *exponential generating function* (EGF), which for a class $\mathcal{C}$ with counting sequence (C_n) is defined by

$$\widehat{C}(z) = \sum_{n=0}^{\infty} C_n \frac{z^n}{n!}. \tag{2.3}$$

The dictionary is then analogous to the plain (unlabelled) case, with the standard exponential replacing the pseudo-exponentials $\mathrm{Exp}, \overline{\mathrm{Exp}}$, etc. Analogues of

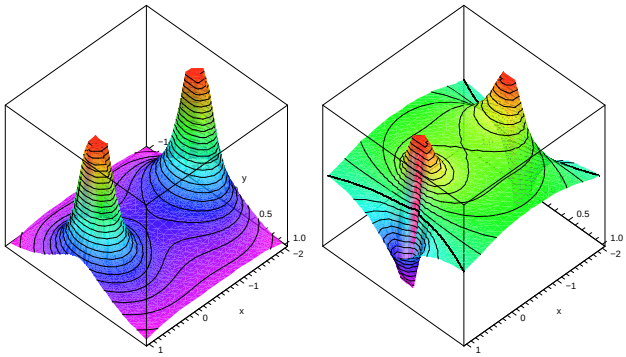


Figure 2: The modulus and real part of the OGF of coverings of an interval by sticks of lengths 1 and 2. (The specification is $\mathcal{F} = \textsc{Seq}(\mathcal{Z} + \mathcal{Z}\mathcal{Z})$, with OGF $F(z) = \frac{1}{1-z-z^2}$, which has polar singularities at $-\frac{1}{2} \pm \frac{1}{2}\sqrt{5}$.)

Theorems 2.1 and 2.2 hold. This framework is implemented in the library `Combstruct` of the computer algebra package MAPLE. The core language of constructions (2.1) can be enriched so as to include pointing operations, substitutions, and order constraints [21]. In summary, what we have done at this stage is *embed a fragment of combinatorics into the algebraic domain of power series.*

3 Complexification

Assigning *complex* values to the variable z that figures in a generating function turns out to have serendipitous consequences. When we do so, a generating function becomes a geometric transformation of the complex plane. This transformation is very regular near the origin—it is *analytic*. In other words, near 0, it only effects a smooth distortion of the complex plane. Farther away from the origin, some cracks start appearing in the picture. These cracks—the dignified name is *singularities*—correspond to the disappearance of smoothness (Fig. 2). It turns out that a function's singularities provide a wealth of information regarding the function's coefficients, and especially their asymptotic rate of growth. Adopting a geometric point of view has a large pay-off.

First, we recall:

DEFINITION 3.1. *A function $f(z)$ is* analytic *(or* holomorphic*) at z_0 if the limit,* $\lim \frac{f(z)-f(z_0)}{z-z_0} =: f'(z_0)$*, exists as $z - z_0 \to 0$ in the complex domain $\mathbb{C}$. Equivalently, $f(z)$ is analytic if, near z_0 it admits a convergent series representation in powers of $(z - z_0)$.*

For instance, all generating functions of classes having at most exponential growth ($C_n \leq K \cdot L^n$) are analytic at the origin.

Integral calculus in the complex domain is appreciably easier[2] than on the real line, since the integral of an analytic function along a path only depends on the end points of the path. As a consequence (via the "residue theorem"): *Coefficients of an analytic function at the origin can be recovered from values of the function* away *from the origin, by means of Cauchy's integral formula:*

$$\text{coeff.}\,[z^n]f(z) = \frac{1}{2i\pi}\int_\gamma f(z)\,\frac{dz}{z^{n+1}}. \tag{3.4}$$

(There, the contour γ can be chosen freely, provided it encircles the origin and stays within the domain of analyticity of f.) The relevance of these notions to combinatorial counting is due to the fact that the GFs of all specifiable classes (in the sense of (2.1)), whether labelled or unlabelled, are analytic at 0, so that Cauchy's formula (3.4) is *a priori* appplicable to them.

Like in analytic number theory (see the classical proofs of the Prime Number Theorem), it turns out that singularities of generating functions play a crucial rôle in estimating coefficients. In essence, a *singularity* is a point on the boundary of the region of analyticity, beyond which the function cannot be continued analytically. (E.g., it ceases to be differentiable). The technology of *singularity analysis* developed by Flajolet and Odlyzko [21, 40], is a central ingredient of analytic combinatorics.

THEOREM 3.1. (SINGULARITY ANALYSIS) *Let $f(z)$ be analytic at 0, have an isolated singularity at σ, and be suitably continuable in a "Δ–domain". If $f(z)$ can be expanded as $z \to \sigma$ in the scale of functions of the form $(1 - z/\sigma)^{-\alpha}$, with $\alpha \in \mathbb{C}$, then an asymptotic expansion of its coefficient $[z^n]f(z)$ results from a formal translation:*

$$\text{function}: C\cdot(1-\frac{z}{\sigma})^{-\alpha} \quad \bullet\!\!\longrightarrow \quad \text{coeff.}: C\cdot\frac{n^{\alpha-1}}{\Gamma(\alpha)}\sigma^{-n}.$$

(Proof: Start from (3.4) and deform the contour so as to pass at distance $1/n$ of the singularity, then steer away from it.)

The singularity analysis theorem has many avatars allowing for logarithms, iterated logarithms in asymptotic expansions, as well as $O(\cdot)$ and $o(\cdot)$ error terms: see [12] for extensions. For "low-complexity" combinatorial classes like the specifiable ones, its effect is the following: *Asymptotic counting is reduced to a purely* local analysis *of generating functions at a few points, their singularities.* This principle brings much clarity

[2] Cf Jacques Hadamard's aphorism: "*The shortest path between two truths in the real domain passes through the complex plane.*"

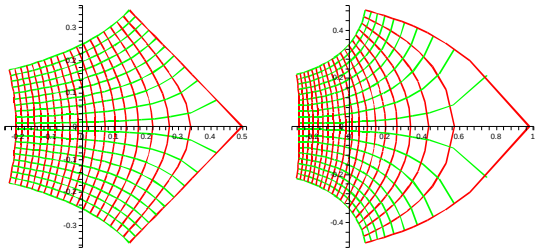


Figure 3: Universality of the square-root singularity: the images by the GFs $G(z)$ (plane trees) and $H(z)$ (nonplane trees) of two regular grids with corners at $\pm\rho(1 \pm i)$ (with $\rho = 1/4$ for G and $\rho \doteq 0.33$ for H).

in the task of estimating the growth of combinatorial classes and, to a large extent, it reduces it to a systematic process. As a consequence, it makes possible a jump in generality: instead of solving isolated combinatorial counting problems, we can consider solving whole families of related problems. We call *schema* a collection of combinatorial conditions, possibly supplemented by side analytic assumptions, which imply a common set of asymptotic properties.

The regular-language schema. This is the easiest instance. It is determined by the combinatorial condition that a class belonging to it is specifiable non-recursively from finite sets by the constructions of $\{+, \times, \textsc{Seq}\}$ alone (this is equivalent to the regular expression framework) or by a linear set of combinatorial equations (this is equivalent to a deterministic automaton). The corresponding generating functions are rational (a well known fact that also derives from Th. 2.1), hence their singularities are poles only. Then, a simplified form of singularity analysis applies (with $\alpha \in \mathbb{Z}_{\geq 1}$ in the notations of Th. 3.1). Thus, the enumeration sequence is described by a finite linear combination of elements of the form

$$\kappa \sigma^{-n} n^{\ell}, \qquad \ell \in \mathbb{Z}_{\geq 0}.$$

For the subclass satisfying the stronger condition that the defining automaton is strongly connected, one has the *irreducible regular-language schema*, and the exponent ℓ is bound to satisfy $\ell = 0$, the proof being a consequence of Perron-Frobenius theory.

The simple-tree schema. A class of rooted unlabelled trees is said to be *simple* if it is determined by a finite or cofinite set Ω of allowable node degrees. Any such class admits a specification of the form either $\mathcal{Y} = \mathcal{Z} \times \textsc{Seq}_{\Omega}(\mathcal{Y})$ (planar case) or $\mathcal{Y} = \mathcal{Z} \times \textsc{Set}_{\Omega}(\mathcal{Y})$ (nonplane case). The implicit function theorem then guarantees a locally linear relation between z and the OGF $Y(z)$, as long as the latter remains analytic. Then, it fails to apply, in which case the relation between z and $Y(z)$ becomes quadratic. This means that the OGF $Y(z)$ has a singularity that is "universally" of the square root type, which can be exploited by singularity analysis:

$$Y(z) \sim \kappa_0 - \kappa_1 \sqrt{1 - z/\rho} \bullet\!\!\blacktriangleright Y_n \sim \kappa_1 \frac{\rho^{-n}}{2\sqrt{\pi n^3}}, \tag{3.5}$$

for a computable number ρ. We express (3.5) as follows: *For any class of trees resorting to the simple-tree schema, asymptotic counts involve an exponential term (ρ^{-n}) modulated by a* universal $n^{-3/2}$ *subexponential factor.* This notion of universality of properties, independently of details of a model, has its origin in statistical physics. Here, universality of the square-root singularity and its companion $n^{-3/2}$ term is related to such trees invariably having height and width $O(\sqrt{n})$, both on average and in probability. See Fig. 3 for an illustration (the similarities of shape in the apex on the right of each diagram correspond to a square-root singularity).

The general context-free schema. This schema comprises all specifications, possibly recursive, that make use of the constructions $\{+, \times, \textsc{Seq}\}$ only (SEQ is redundant). By the symbolic method, the corresponding generating functions are algebraic (this is the Chomsky-Schützenberger theorem). Now algebraic functions have expansions at singularities characterized by the Newton-Puiseux theorem, to the effect that the exponents at singularities are rational numbers. Singularity analysis is then systematically applicable, which yields the property that coefficients are described asymptotically as finite linear combination of elements of the form

$$\kappa \sigma^{-n} n^{\gamma}, \qquad \gamma = \frac{p}{q} \in \mathbb{Q} \setminus \mathbb{Z}_{<0}. \tag{3.6}$$

This last property constitutes a generalized density theorem for unambiguous context-free languages.

The irreducible context-free schema. This is a subschema defined by the fact that the dependency graph between classes entering the specification is strongly connected. The important Drmota-Lalley-Woods (DLW) theorem (see [21] and references) asserts the universality of the square-root singularity, so that for such a class $\mathcal{C}$:

$$C_n \sim K \sigma^{-n} n^{-3/2}, \tag{3.7}$$

a vast generalization of the simple tree case. For instance, given the n vertices of a convex polygon, consider *noncrossing graphs* defined by the fact that no two edges cross. An explicit construction shows that the class $\mathcal{N}$ of these geometric graphs belongs to the irreducible context-free schema (Fig. 4). The DLW theorem then leads to precise asymptotic counts as well as probabilistic estimates (number of components,

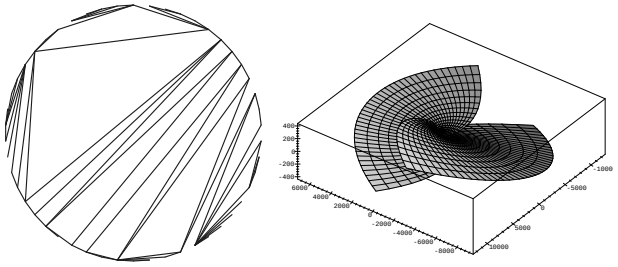


Figure 4: A random non-crossing graph of size 50 [left] and a blowup of the imaginary part of its GF [right] revealing a square-root singularity, in accordance with universality predicted by the irreducible context-free schema.

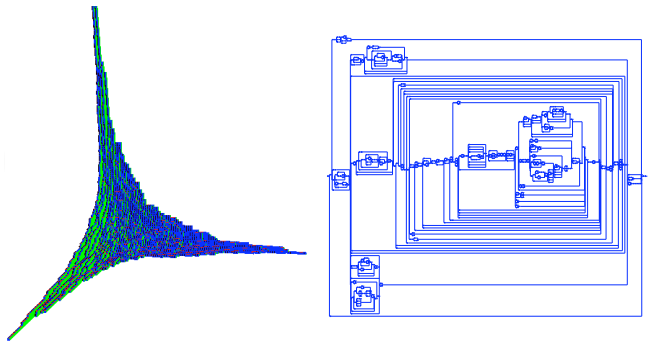


Figure 5: A random plane partition and a random series-parallel graph obtained by Boltzmann sampling.

number of edges) that, in a sense, quantify percolation on the non-crossing graph.

Many more schemas are developed in [21]. Singularity analysis and related complex asymptotic methods—the *saddle-point method* and the *Mellin transform*[3], principally—are widely applicable to combinatorial structures amenable to the symbolic method, as well as to many other structures for which one has a handle on the corresponding GFs.

4 Real Numbers and Boltzmann models

In the previous section we have gone directly from the formal to the complex domain of z–values. As a matter of fact, regarding asymptotic enumeration, the real domain is usually not sufficient to derive interesting properties: classical Tauberian theory, which deals with such questions, necessitates too strong side conditions and usually provides too weak estimates. There is however a benefit in introducing probabilistic models based on positive real values of z as regards efficient algorithms for *random generation*, a classical area of combinatorial mathematics [39], which makes the simulation of discrete models effective.

DEFINITION 4.1. *Given a combinatorial class $\mathcal{C}$, the (ordinary)* Boltzmann model *assigns to an element $\gamma \in \mathcal{C}$ a probability proportional to an exponential of its size: $\mathbb{P}(\gamma) \propto x^{|\gamma|}$, or equivalently,*

$$\mathbb{P}(\gamma) = \frac{x^{|\gamma|}}{C(x)},$$

where x is a positive control parameter.

[3]The saddle-point method consists in choosing in (3.4) a contour that crosses saddle-points of the integrand. The Mellin transform of a function $f(x)$ is $f^\star(s) = \int_0^\infty f(x)x^{s-1}\,dx$; it maps asymptotic properties of $f(x)$ to singularities of $f^\star(s)$ and enjoys properties analogous to Th. 3.1.

A parallel notion exists for labelled classes. By definition, two objects of the same size are assigned equal probabilities, so that a Boltzmann sampler conditioned upon the size of the object it produces is a *bona fide* random generator. It turns out that, for classes that are specifiable in the core constructions of (2.1), random generators according to this distribution can be systematically built [10, 15].

THEOREM 4.1. (BOLTZMANN SAMPLERS) *Given a specification for a class $\mathcal{C}$, there exists a direct translation, which produces a Boltzmann sampler $\Gamma C(x)$ for $\mathcal{C}$. That sampler has an expected linear-time complexity (under a real-arithmetic model) in the size of the object it produces.*

Under a Boltzmann model, the size of the object produced is a random variable S (depending on x) such that

$$\mathbb{P}(S = n) = \frac{C_n x^n}{C(x)}. \tag{4.8}$$

Let $\rho \le +\infty$ be the radius of convergence of $C(x)$: larger structures can only be obtained by letting $x \to \rho$. The distribution (4.8) can then be studied by precisely the same complex methods that give information on the sequence (C_n). In particular, for each of the major schema, one can *tune* the Boltzmann sampler so that it will tend to produce objects near a target size n: the resulting samplers (based on tuning, rejection, and possibly specification transformations based on pointing) then produce in expected *linear time* an object whose size belongs to any prescribed interval $[(1-\epsilon)n, (1+\epsilon)n]$. If exact size n is imposed, then the generator so obtained has *quadratic* or subquadratic time complexity. Fig. 5 displays a random plane partition of size about 15,000 and a random series-parallel graph of size about 500 as drawn by Boltzmann samplers (courtesy of Ms Carine Pivoteau).

5 Random Structures

So far we have been discussing purely enumerative results. The study of *parameters* of large discrete structures can also be conducted within the framework of analytic combinatorics.

(*i*) Moments of simple parameters can be subjected to the symbolic and analytic methods already discussed. For instance, in order to estimate the mean number of leaves of trees in a family it suffices to enumerate both the basic tree family $\mathcal{T}$ and a derived family $\mathcal{T}^\star$ which is comprised of all trees with a single leaf pointed. The expected number of leaves is then the ratio $[z^n]T(z)/([z^n]T^\star(z))$. This extends to higher moments. Limit distributions can then often be identified thanks to *moment convergence theorems* of classical probability theory (these provide conditions under which convergence of moments implies convergence in law).

(*ii*) Extremal parameters can be investigated by introducing a collection of combinatorial classes corresponding to fixed values of the parameter. For instance, binary trees and height are characterized by the collection of specifications,

$$\mathcal{B} = \mathcal{Z} + (\mathcal{B}\times\mathcal{B}), \quad \mathcal{B}^{[h+1]} = \mathcal{Z} + (\mathcal{B}^{[h]}\times\mathcal{B}^{[h]}), \tag{5.9}$$

where $\mathcal{B}^{[h]}$ is the class of trees of height bounded by h. The corresponding OGFs satisfy the recurrence $y_{h+1} = z + y_h^2$, which is none other than what gives rise to the Mandelbrot set: the limit distribution of height[4], an elliptic theta function, is actually encoded by the behaviour of the recurrence near the singularity $z = \frac{1}{4}$ of the fixed point $B(z)$.

(*iii*) Complete distributional information on simple parameters can be obtained by means of a multivariate extension of the univariate paradigm, as we explain below.

Consider a class $\mathcal{F}$ on which a parameter $\chi : \mathcal{F} \mapsto \mathbb{Z}_{\geq 0}$ is defined. The *bivariate generating function* (BGF) of $\mathcal{F}, \chi$ is in the ordinary (unlabelled) case:

$$F(z,u) = \sum_{n,k} F_{n,k} z^n u^k = \sum_{\varphi\in\mathcal{F}} z^{|\varphi|} u^{\chi[\varphi]}.$$

Naturally, the BGF becomes a classical counting GF when one sets $u = 1$, namely, $F(z,1) = F(z)$, For "inherited" parameters, the symbolic calculus of Fig. 1 can be extended. (Conceptually, the BGF is a refined reduction, which keeps track of atoms as well as another characteristic of combinatorial objects.) For instance, for the class $\mathcal{G}$ of plane trees with χ being the number of leaves, one has available the enriched specification

$$\mathcal{G} = \widehat{\mathcal{Z}} + \mathcal{Z}\times \textsc{Seq}_{\geq 1}(\mathcal{G}) \quad \Longrightarrow \quad G = zu + z\frac{G}{1-G}, \tag{5.10}$$

where $G \equiv G(z,u)$ is the corresponding BGF. The problem of extracting asymptotic distributional information is *a priori* one of extracting coefficients of a function of several variables.

The road to distributional information regarding parameters goes as follows. Parameters of combinatorial structures induce, via the auxiliary variable u considered near 1, local deformations of the univariate (counting) generating functions. Under fairly general conditions, such deformations are amenable to *perturbation theory* and admit of uniform expansions near singularities. In this way the deep and difficult theory of functions of several variables is bypassed. (See however the work by Pemantle and Wilson for exciting uses of this theory in combinatorics [41].) In the perturbative approach, two frequently occurring scenarios then emerge.

MS: *Movable singularity.* In this case, the perturbation by $u \approx 1$ induces a smooth (i.e., analytic) displacement of the dominant singularity of the BGF, viewed as a function of z alone. One has $F(z,u) \approx (1 - z/\rho(u))^{-\alpha_0}$, where the singular exponent remains constant.

VE: *Variable exponent.* In this case, the singularity remains fixed at some value ρ, for u in a small neighbourhood of 1, but the singular exponent is allowed to vary smoothly. One has $F(z,u) \approx (1 - z/\rho)^{-\alpha(u)}$.

One has:

THEOREM 5.1. (GAUSSIAN LIMITS) *In the movable singularity case, the mean and variance of the parameter χ are asymptotically linear in the value n of size. In the variable exponent case, the mean and variance are asymptotically logarithmic in n. In both cases, the distribution of χ is asymptotically normal.*

Proof. We only sketch it in case *MS*. The usual *probability generating function* (PGF) of χ over $\mathcal{F}_n$ is exactly given by coefficient extraction as

$$\phi_n(u) = \frac{[z^n]F(z,u)}{[z^n]F(z,1)}.$$

[4] Since order constraints correspond symbolicly to integral operators, the height problem of binary search trees and the recursion depth of Quicksort translate into an integral quadratic recurrence of the form $y_{h+1} = 1 + \int y_h^2$. The difficult problem of quantifying these parameters has been recently advanced by Drmota [8], whose analytic estimates refine the ingenious probabilistic approach of Devroye [6].

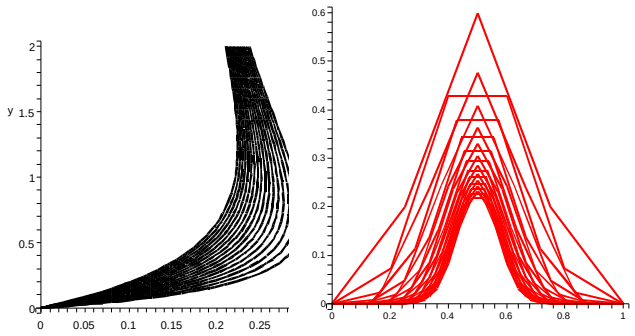


Figure 6: The BGF of trees $G(z, u)$ of (5.10), for z real as u varies around 1, has a constant singular exponent $(\frac{1}{2})$ and a movable singularity at $\rho(u)$ [left], corresponding to a limit normal distribution of the number of leaves in trees[right].

By the moving singularity hypothesis, and since singularity analysis preserves *uniformity*, we obtain an estimate

$$\phi_n(u) \approx \left(\frac{\rho(1)}{\rho(u)}\right)^n. \tag{5.11}$$

The PGFs of sums of independent identically distributed random variables are exact power representations of a form comparable to (5.11), whose precise form is known as a *quasi-powers* approximation. An important theorem[5] due to H. K. Hwang [31] then implies the normal approximation. □

This theorem applied to (5.10) implies that the number of leaves in a random plane tree is asymptotically normal; see Fig. 6. The property extends to the simple-tree schema, as well as to several other important schemas, as shown by Drmota [7].

6 Applications and Extensions

The analytic combinatorial theory provides a powerful conceptual framework that could be summarized (optimistically) by the motto: *"If you can specify it, you can analyse it!"*. It makes it possible to organize discrete models and algorithms into broad categories and discuss the functional equations that arise in a highly synthetic fashion. (This is a bit like physics with its wave equations, heat equation, laws of optics and electrostatics, etc.) An important economy of technical detail then results from a moderate use of mathematical abstraction. We illustrate this point by some examples from algorithmic theory and practice.

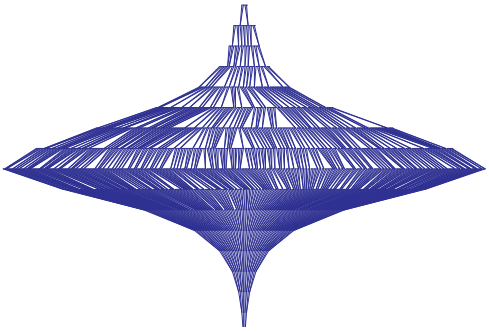
[5]The proof bases itself on the continuity theorem for characteristic functions (like in the classical Central Limit Theorem), on the Berry-Esseen inequalities, and on strong differentiability properties of analytic functions.

Patterns in random strings. We are here in a direct case of application of the standard paradigm. Let Ω represent the total number of occurrences of a fixed finite set of patterns in a random string over some finite alphabet. For either the Bernoulli model (letters are independent and identically distributed) or the Markov model, the corresponding BGF is rational as it corresponds to an effectively constructible finite-state device, which belongs to the irreducible regular-language schema. Perturbation methods imply that the BGF viewed as a function of z has a simple dominant pole at some $\rho(u)$ that depends in an analytic manner on u. Hence the limit law of Ω in random strings of large size n is Gaussian. Many extensions are possible [22, 38], and the book [48] offers a systematic discussion. Such results are of relevance to the analysis of biological sequences, where it is needed to separate signal (a meaningful observation of occurrences) from noise (a statistically unavoidable phenomenon).

Polynomials over finite fields. This example illustrates best the synthetic reasoning afforded by the analytic combinatorics calculus. It is relative to the class $\mathcal{P}$ of all polynomials with coefficients in a finite field $\mathbb{F}_q$, which are of interest in coding theory, computational number theory, cryptography, and symbolic manipulation systems [54].

A polynomial being determined by its *sequence* of coefficients, the GF $P(z)$ of all polynomials has a polar singularity. By the unique factorization property, $\mathcal{P}$ is also isomorphic to the class of all *multisets* of irreducible polynomials $\mathcal{I}$: $\mathcal{P} \simeq \text{MSET}\{\mathcal{I}\}$. Since the multiset construction corresponds to exponentiation of GFs, one has $P \approx e^I$, that is, the singularity of the OGF $I(z)$ is logarithmic. By singularity analysis, the number of irreducible polynomials is thus found to be asymptotic to q^n/n: this is an analogue for polynomials of the Prime Number Theorem (known to Gauß). By multivariate symbolic methods, the BGF of the number of irreducible factors in polynomials is of the singular type $(1 - qz)^{-u}$, showing a smoothly variable exponent. Then, Th. 5.1 gives: *the number of irreducible factors of a random polynomial over $\mathbb{F}_q$ is asymptotically Gaussian.* (The integer version is the celebrated Erdős-Kac theorem.) Refinements of this method lead to a complete analysis of a major polynomial factorization chain [18].

Hashing. Hashing techniques are archetypes of randomized algorithms. They depend on properties of random allocations and at the same time lead to nice combinatorial developments. For instance, the formation of "islands" in linear probing hashing represents a generic type of coalescence, which can be approached in a variety of fashions. A combinatorial decomposition



$$\frac{\partial F(z,u)}{\partial z} = F(pz,u)F(qz,u)$$

Figure 7: A random trie of size 500 and the Jacquet-Szpankowski equation that models Lempel-Ziv compression.

leads to a nonlinear difference-differential equation,

$$\frac{\partial F}{\partial z} = F \cdot \Delta_q[F], \qquad \Delta_q[f(z)] := \frac{f(qz) - f(z)}{z(q-1)},$$

out of which moments can be "pumped". There result fascinating connections with path length in trees, area under the Brownian excursion, depth-first search traversal of graphs, and the formation of the giant component in the Erdős-Rényi graph [19].

Digital trees (tries). Digital trees are an efficient and highly versatile data structure [3, 34, 36, 47, 48, 53] and an abstract process on which many algorithms are based (Fig. 7). A finite set of words ω can be decomposed according to the leading letter of each word in ω, giving rise to a tree structure, the *trie*, where common prefixes are factored and stored only once. In the abstract, we have a recursive Bernoulli splitting process. The first average-case analysis, due to Knuth and de Bruijn around 1965, was based on what was later to be recognized as a Mellin transform technology. It revealed the existence of periodic fluctuations[6] in the expected cost of the algorithm, which are of a tiny amplitude (about 10^{-5}), seemingly a theoretician's curiosity, but ones without which are understanding of the trie model proves severely incomplete. Recently, Fill and Janson [13] have shown by these techniques the bit-complexity of binary tree search to be $\sim C \cdot (\log n)^2$.

The abstract trie model is the one that underlies a communication protocol, the Capetanakis-Tsybakov-Mikhailov *tree protocol* that is a competitor to the ETHERNET. The latter was proved to be strongly unstable in the long term by Aldous [1], whereas the tree protocol remains stable till arrival rates of $\lambda_{\max} \doteq 0.36$, corresponding to an efficiency that is about 70% of the information-theoretic optimum. The stability result of the tree protocol strongly depends on the analytic methodology: for instance the variance of collision resolution intervals can be quantified thanks to identities of Ramanujan, as noted by Kirschenhofer and Prodinger [33].

The trie model also underlies *data compression schemes* like the Lempel-Ziv algorithms. Jacquet, Louchard, and Szpankowski (see, e.g., [32, 35]) could make a deep use of analytic methods so as to quantify precisely the *redundancy* rate of such algorithms. (The problem is one of second-order asymptotics, not easily attacked by probabilistic methods, with fluctuations that are in the way!) In the course of their investigations, they were led to developing a powerful de-Poissonization technique of great generality, based on generating functions and the saddle-point method [48].

Search trees and the holonomic framework. Splitting a file according to a random pivot or its largest element or its first element leads to generating functions that satisfy differential equations and systems. To these, singularity analysis is applicable under broad conditions. A typical instance is the *quadtree* data structure of Finkel and Bentley, for which the most important parameters are now well understood: for example, the expected cost of a partial match query in a random quadtree is found to satisfy [17]

$$Q_n \sim \kappa \cdot n^{\beta}, \qquad \beta = \frac{-3+\sqrt{17}}{2} \doteq 0.56155, \tag{6.12}$$

with a curious occurrence of a "critical exponent" that is an algebraic number. Such analytic considerations are closely intertwined with algorithmic design: see for instance the randomized k-dimensional tree of Martínez *et al.* [9].

Asymptotic phenomena like (6.12) are typically attached to recursive combinatorial structures that involve order constraints. The generating functions resort to what Zeilberger [56] has named the *holonomic framework*[7], which lies at the crossroads of differential algebra, combinatorics, special function theory, and complex analysis, and has great descriptive powers [26].

Arithmetic algorithms. Methods akin to the ones presented here have provided some deep results in the analysis of *Euclidean algorithms* and *continued fractions* [29, 49, 51]. They depend on the introduction of *transfer operators* from dynamical systems theory,

[6] Similar fluctuations, often of a fractal nature, are otherwise known to be systematically present in the behaviour of deterministic divide-and-conquer algorithms [16].

[7] A function is holonomic or ∂-finite if the vector space (over the field of rational functions) of all its partial derivatives is finite-dimensional.

which, for a transformation T, are defined by

$$\mathbf{G}_s[f](x) = \sum_{h \in T^{-1}} h(x)^s f \circ h(x).$$

This has led to a discovery of the fact that *"Euclid's algorithm is Gaussian"* and to the solution of the long-standing open problem of analysing the *binary GCD* algorithm. Such analyses are also of practical relevance, since it is estimated that many dedicated systems in cryptography and symbolic computation spend a large fraction of their time in gcd computations (e.g., each time two rational numbers are added). There are also surprising connections with classical mathematics. For instance, it is established in [23] that *the average complexity of sorting real numbers by means of continued fractions involves fluctuations whose order of growth is dictated by the Riemann hypothesis.*

Planar maps and graphs. Planar *maps* are (planar) graphs embedded in the plane. In the 1960's, Tutte showed many varieties of maps to have an algebraic GF, whose analysis a priori resorts to the estimates (3.6), with a universal exponent $\gamma = \frac{5}{2}$. Recently, Schaeffer [45] has given simplified constructions for maps, leading to fast random generation algorithms and opening access to new combinatorial parameters like distances and embedding characteristics. Such results are of interest in the areas of graph drawing, computational geometry (the information content of discretized surfaces), and the design of implicit data structures. Gimenez and Noy [27] succeeded recently in solving the long-standing open problem of enumerating *planar graphs*: the growth is of the form $\kappa K^n n^{-7/2}$ (the exponential rate K was unknown), and the methods involve a difficult blend of combinatorial decompositions (building on works of Bender *et al.* [2]) and complex-analytic methods. Based on Boltzmann models, Fusy [25] could then derive a random generation algorithm for planar graphs that is of linear time complexity for approximate size and of quadratic complexity for exact size sampling.

Probabilistic stream algorithms. This is an area of intense activity in the data mining and networking communities. Due to space limitations, we only discuss *cardinality estimators*. Given a stream of data (of several gigabytes), it is required to determine its cardinality (the number of distinct elements). No a priori probabilistic model is available to account for replications and permutations amongst the data items. The basic idea consists in hashing elements (so as to randomize values) and examine "observables" which are simple functions of the hashed values, independent of replications. (For instance, observing the initial pattern `01010101` is a likely indication that a file's cardinality is $> 2^8$.) The design of an algorithm then consists in the following steps: (*i*) choose an observable; (*ii*) analyse its behaviour under the random binary string model; (*iii*) invert the expected value of the observable to deduce a plausible estimate of the (unknown) cardinality n; (*iv*) perform another combinatorial-probabilistic analysis in order to derive an unbiased estimator. Random allocations and their EGFs, de-Poissonization, and Mellin transforms are central ingredients of this programme. This has led to the best currently known algorithm [11] for estimating cardinalities: $m \log_2 \log n$ bits of information suffice to estimate cardinalities till n, with a typical accuracy of $\approx \frac{1}{\sqrt{m}}$. The equivalent of four lines of printed text is enough to estimate, to a few percents, the size of the vocabulary of all of Shakespeare's works!

7 Perspective

Stochastic versus analytic approaches. The differences between the two approaches can largely be summarized by the following diagram:

$$\begin{array}{ccc} \mathcal{M}_n & \longrightarrow & \mathcal{M}_\infty \\ \downarrow & & \downarrow \\ M(z) & \longrightarrow & \text{asymp}(\mathcal{M}_n). \end{array}$$

Roughly, the probabilistic approach (East, then South) aims at coming up with a continuous stochastic process ($\mathcal{M}_\infty$) that describes the limit behaviour of the whole family of models ($\mathcal{M}_n$), then return to finite models of large index. (For instance, discrete random walks are approximated by Brownian motion.) By contrast, the analytic approach (South, then East) encapsulates exact information into a function ($M(z)$), from which asymptotic information is extracted by means of complex-analytic methods. In the current state of our knowledge, analytic methods mostly apply to objects of low structural complexity, the ones that are often relevant for basic algorithms and data structures. As summarized by Odlyzko in [40]: *"Analytic methods are extremely powerful and when they apply they often yield estimates of unparalleled precision."*

Complexity lower bounds. There is a limited but perhaps interesting intersection between analytic methods and complexity lower bounds. Roughly, we associate to a combinatorial class a characteristic analytic object (its GF, viewed as a transformation of the complex plane). If it can be recognized that a given class $\mathcal{C}$ has a GF that does not belong to a family Ξ of functions corresponding to a formal specification mechanism $\mathbf{M}$, then, there results a *structural complexity lower bound*: the class $\mathcal{C}$ cannot be specified (described) within $\mathbf{M}$. For instance, trees cannot be encoded by any regular language nor recognized by any finite automaton since their GF is an algebraic irrational func-

tion. (The classical proof is based on combinatorial "pumping lemmas".) It has proved possible to obtain purely language-theoretic results in this way [14]. What however currently limits the efficacy of this method is the fact that usual complexity classes have GFs, which are hard to characterize. Perhaps, Vallée's methodology [50, 51] that interprets atoms as *transfer operators* of dynamical system theory has a richer potential as it is less "reductionistic" and can better take into account noncommutativity of letters in words. The intermediate approach of replacing z by complex matrices would deserve investigation in this perspective.

A calculus of discrete structures. A final conclusion to this brief guided tour is the possibility of building a synthetic theory of metric properties of large discrete structures, based on a dedicated operational calculus. The central chain of analytic combinatorics based on

$$\text{constructions} \rightarrow \text{complexification} \rightarrow \text{perturbation theory}$$

proves effective in describing combinatorial classes that "decompose well". This fact in turn implies that many fundamental algorithms and data structures can be precisely analysed and tuned for optimal performance. The corresponding calculus, largely motivated by considerations of algorithmic efficiency, is also of some mathematical interest *per se*.

References

[1] David J. Aldous. Ultimate instability of exponential back-off protocol for acknowledgement-based transmission control of random access communication channels. *IEEE Transactions on Information Theory*, 33(2):219–223, March 1987.

[2] Edward A. Bender, Zhicheng Gao, and Nicholas C. Wormald. The number of labeled 2-connected planar graphs. *Electronic Journal of Combinatorics*, 9(1):Research Paper 43, 13 pp., 2002.

[3] Jon Bentley and Robert Sedgewick. Fast algorithms for sorting and searching strings. In *Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM Press, 1997.

[4] F. Bergeron, G. Labelle, and P. Leroux. *Combinatorial species and tree-like structures*. Cambridge University Press, Cambridge, 1998.

[5] Arthur W. Burks, Herman H. Goldstine, and John von Neumann. Preliminary discussion of the logical design of an electronic computing instrument. Report to the U.S. Army Ordnance Department, 1946. Reprinted in *Computer Structures: Readings and Examples*, C. Gordon Bell and Allen Newell editors, McGraw-Hill, 1971, pp. 92–119.

[6] Luc Devroye. Branching processes in the analysis of the heights of trees. *Acta Informatica*, 24:277–298, 1987.

[7] Michael Drmota. Systems of functional equations. *Random Structures & Algorithms*, 10(1–2):103–124, 1997.

[8] Michael Drmota. An analytic approach to the height of binary search trees. *Algorithmica*, 29:89–119, 2001.

[9] A. Duch, V. Estevill-Castro, and C. Martínez. Randomized k-dimensional binary search trees. In K.-Y. Chwa and O.H. Ibarra, editors, *Proc. of the 9th Int. Symp. on Algorithms and Computation (ISAAC)*, volume 1553 of *Lecture Notes in Computer Science*, pages 327–338, 1998.

[10] Philippe Duchon, Philippe Flajolet, Guy Louchard, and Gilles Schaeffer. Boltzmann samplers for the random generation of combinatorial structures. *Combinatorics, Probability and Computing*, 13(4–5):577–625, 2004. Special issue on Analysis of Algorithms.

[11] Marianne Durand and Philippe Flajolet. LogLog counting of large cardinalities. In G. Di Battista and U. Zwick, editors, *Annual European Symposium on Algorithms (ESA03)*, volume 2832 of *Lecture Notes in Computer Science*, pages 605–617, 2003.

[12] James A. Fill, Philippe Flajolet, and Nevin Kapur. Singularity analysis, Hadamard products, and tree recurrences. *Journal of Computational and Applied Mathematics*, 174:271–313, February 2005.

[13] James Allen Fill and Svante Janson. The number of bit comparisons used by quicksort: An average-case analysis. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA04)*, pages 293–300, 2001.

[14] Philippe Flajolet. Analytic models and ambiguity of context–free languages. *Theoretical Computer Science*, 49:283–309, 1987.

[15] Philippe Flajolet, Éric Fusy, and Carine Pivoteau. Boltzmann sampling of unlabelled structures. Preprint. Submitted to *Analco'07*, New-Orleans, 14 pages.

[16] Philippe Flajolet and Mordecai Golin. Mellin transforms and asymptotics: The mergesort recurrence. *Acta Informatica*, 31:673–696, 1994.

[17] Philippe Flajolet, Gaston Gonnet, Claude Puech, and J. M. Robson. Analytic variations on quadtrees. *Algorithmica*, 10(7):473–500, December 1993.

[18] Philippe Flajolet, Xavier Gourdon, and Daniel Panario. The complete analysis of a polynomial factorization algorithm over finite fields. *Journal of Algorithms*, 40(1):37–81, 2001.

[19] Philippe Flajolet, Patricio Poblete, and Alfredo Viola. On the analysis of linear probing hashing. *Algorithmica*, 22(4):490–515, December 1998.

[20] Philippe Flajolet, Bruno Salvy, and Paul Zimmermann. Automatic average–case analysis of algorithms. *Theoretical Computer Science*, 79(1):37–109, February 1991.

[21] Philippe Flajolet and Robert Sedgewick. *Analytic Combinatorics*. April 2006. Chapters I–IX of a book to be published by Cambridge University Press, 717p.+x, available electronically from P. Flajolet's home page.

[22] Philippe Flajolet, Wojciech Szpankowski, and Brigitte Vallée. Hidden word statistics. *Journal of the ACM*, 53(1):147–183, January 2006.

[23] Philippe Flajolet and Brigitte Vallée. Continued fractions, comparison algorithms, and fine structure constants. In Michel Théra, editor, *Constructive, Experimental, and Nonlinear Analysis*, volume 27 of *Canadian Mathematical Society Conference Proceedings*, pages 53–82, Providence, 2000. American Mathematical Society.

[24] Philippe Flajolet, Paul Zimmerman, and Bernard Van Cutsem. A calculus for the random generation of labelled combinatorial structures. *Theoretical Computer Science*, 132(1-2):1–35, 1994.

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

[26] Ira M. Gessel. Symmetric functions and P–recursiveness. *Journal of Combinatorial Theory,* Series A, 53:257–285, 1990.

[27] Omer Giménez and Marc Noy. The number of planar graphs and properties of random planar graphs. *Discrete Mathematics and Theoretical Computer Science Proceedings*, AD:147–156, 2005.

[28] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics*. Addison Wesley, 1989.

[29] Doug Hensley. The number of steps in the Euclidean algorithm. *Journal of Number Theory*, 49(2):142–182, 1994.

[30] Micha Hofri. *Analysis of Algorithms: Computational Methods and Mathematical Tools*. Oxford University Press, 1995.

[31] Hsien-Kuei Hwang. On convergence rates in the central limit theorems for combinatorial structures. *European Journal of Combinatorics*, 19(3):329–343, 1998.

[32] Philippe Jacquet and Wojciech Szpankowski. Asymptotic behavior of the Lempel-Ziv parsing scheme and digital search trees. *Theoretical Computer Science*, 144(1–2):161–197, 1995.

[33] Peter Kirschenhofer and Helmut Prodinger. On some applications of formulæ of Ramanujan in the analysis of algorithms. *Mathematika*, 38:14–33, 1991.

[34] Donald E. Knuth. *The Art of Computer Programming*. Addison-Wesley, 1968-73. 3 volumes: *Fundamental Algorithms, Seminumerical Algorithms, Sorting and Searching.*

[35] Guy Louchard and Wojciech Szpankowski. On the average redundancy rate of the Lempel-Ziv code. *IEEE Transactions on Information Theory*, 43(1):2–8, 1997.

[36] Hosam M. Mahmoud. *Evolution of Random Search Trees*. John Wiley, New York, 1992.

[37] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[38] Pierre Nicodème, Bruno Salvy, and Philippe Flajolet. Motif statistics. *Theoretical Computer Science*, 287(2):593–617, 2002.

[39] Albert Nijenhuis and Herbert S. Wilf. *Combinatorial Algorithms*. Academic Press, second edition, 1978.

[40] A. M. Odlyzko. Asymptotic enumeration methods. In R. Graham, M. Grötschel, and L. Lovász, editors, *Handbook of Combinatorics*, volume II, pages 1063–1229. Elsevier, Amsterdam, 1995.

[41] Robin Pemantle and Mark C. Wilson. Twenty combinatorial examples of asymptotics derived from multivariate generating functions. Preprint, dec 2005. Arxiv:math.CO/0512548, 89 pages.

[42] G. Pólya. Kombinatorische Anzahlbestimmungen für Gruppen, Graphen und chemische Verbindungen. *Acta Mathematica*, 68:145–254, 1937.

[43] George Pólya, Robert E. Tarjan, and Donald R. Woods. *Notes on Introductory Combinatorics*. Progress in Computer Science. Birkhäuser, 1983.

[44] Michael O. Rabin. Probabilistic algorithms. In J. Traub, editor, *Algorithms and Complexity: New Directions and Recent Results*, pages 21–39, New York, 1976. Academic Press.

[45] Gilles Schaeffer. Random sampling of large planar maps and convex polyhedra. In *Proceedings of the thirty-first annual ACM symposium on theory of computing (STOC'99)*, pages 760–769, Atlanta, Georgia, May 1999. ACM press.

[46] Robert Sedgewick. *Algorithms in C, Parts 1–4*. Addison–Wesley, Reading, Mass., third edition, 1998.

[47] Robert Sedgewick and Philippe Flajolet. *An Introduction to the Analysis of Algorithms*. Addison-Wesley Publishing Company, 1996.

[48] Wojciech Szpankowski. *Average-Case Analysis of Algorithms on Sequences*. John Wiley, New York, 2001.

[49] Brigitte Vallée. Dynamics of the binary Euclidean algorithm: Functional analysis and operators. *Algorithmica*, 22(4):660–685, 1998.

[50] Brigitte Vallée. Dynamical sources in information theory: Fundamental intervals and word prefixes. *Algorithmica*, 29(1/2):262–306, 2001.

[51] Brigitte Vallée. Euclidean dynamics. *Discrete and Continuous Dynamical Systems*, 15(1):281–352, 2006.

[52] Joris van der Hoeven. Relax, but don't be too lazy. *Journal of Symbolic Computation*, 34(6):479–542, 2002.

[53] Jeffrey Scott Vitter and Philippe Flajolet. Analysis of algorithms and data structures. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume A: Algorithms and Complexity, chapter 9, pages 431–524. North Holland, 1990.

[54] Joachim von zur Gathen and Jürgen Gerhard. *Modern computer algebra*. Cambridge University Press, New York, 1999.

[55] Herbert S. Wilf. *Generatingfunctionology*. Academic Press, 1990.

[56] Doron Zeilberger. A holonomic approach to special functions identities. *Journal of Computational and Applied Mathematics*, 32:321–368, 1990.