Invited lecture given at the 19th International Colloquium ICALP'92, Vienna, July 1992. Proceedings published as *Automata, Languages and Programming*, in *Lecture Notes in Computer Science* **623**, W. Kuich Editor (1992), pp 186–210.

Analytic Analysis of Algorithms

Philippe FLAJOLET*
Algorithms Project
INRIA, Rocquencourt
F–78150 Le Chesnay (France)

Abstract. *The average case analysis of algorithms can avail itself of the development of synthetic methods in combinatorial enumerations and in asymptotic analysis. Symbolic methods in combinatorial analysis permit to express directly the counting generating functions of wide classes of combinatorial structures. Asymptotic methods based on complex analysis permit to extract directly coefficients of structurally complicated generating functions without a need for explicit coefficient expansions.*

Three major groups of problems relative to algebraic equations, differential equations, and iteration are presented. The range of applications includes formal languages, tree enumerations, comparison–based searching and sorting, digital structures, hashing and occupancy problems.

These analytic approaches allow an abstract discussion of asymptotic properties of combinatorial structures and schemas while opening the way for automatic analysis of whole classes of combinatorial algorithms.

Introduction

In elementary cases, the average case analysis of a combinatorial algorithm follows a simple pattern. First, set up *recurrences* depending upon the structure of the algorithm that relate the complexity on the collection of all inputs of size n to the complexity on inputs of a smaller size $k < n$. Next solve the recurrences explicitly by algebraic manipulations, whenever possible. Conclude by an asymptotic evaluation often based on basic real analysis, for instance approximating discrete sums by integrals. These classical techniques are reviewed for instance in [24, 46, 51, 54, 55, 79].

In this section, we re-examine the analysis of Quicksort. First we recall the usual analysis by means of recurrences. Next, we sketch an alternate derivation by means of generating functions. This provides a simple illustration of the leading theme of this paper: *Generating functions are central to combinatorial enumerations and the average–case analysis of algorithms.*

Part I deals with general methods. Various theories have been developed that furnish direct correspondences between combinatorial structures and generating functions, as explained in Section 1. Asymptotic methods based on complex analysis discussed in Section 2, then permit to extract coefficients directly from the generat-

*Author's electronic mail address: flajolet@inria.fr

ing function itself. In this way, wide classes of problems receive satisfactory solutions in asymptotic form.

Part II presents a panorama of some recent investigations where generating functions have been instrumental in arriving at results barely accessible to elementary methods. Three major groups of problems relating to algebraic equations, differential equations, and functional equations are presented. Applications in the average–case analysis of algorithms concern a variety of domains: formal languages, tree enumerations, comparison based searching and sorting, digital structures, hashing and occupancy.

Part III surveys some recent approaches to the analysis of combinatorial schemas, as well as related studies in the automated analysis of some well defined classes of combinatorial problems.

Quicksort and recurrences. The traditional example of Quicksort, the sorting algorithm used in the Unix system, is now discussed. The structure of the algorithm (see [70, Chap. 9]) is well known.

Sorting n elements reduces partioning the n elements with respect to the first element in the file and then sorting the two resulting subgroups of sizes K and $n-1-K$, with K depending on the actual data input to the algorithm. The structure of the algorithm is as follows:

```
procedure QuickSort(l,r : integer);
{sorts the part T[l..r] of a global array T[1..n]}
if r>l then
     i:=Partition(l,r);
     {a[l] is to be placed at position i}
     Quicksort(l,i-1);
     Quicksort(i+1,r)
```

When applied to data in random order, i.e. to a *random* permutation of size n, the random variable K assumes each of its possible values $K \in [0..n-1]$ with equal likelihood. Let $\bar{Q}_n$ be the expected number of comparisons of Quicksort when applied to n random data. The recurrence is based on the recursive structure of the algorithm,

$$\bar{Q}_n = p_n + \sum_{k=0}^{n-1} \pi_{n,k}[\bar{Q}_k + \bar{Q}_{n-1-k}]. \tag{1}$$

There $\pi_{n,k}$ is the probability that the partitioning stage splits the file into two subfiles of sizes k and $n-1-k$, so that $\pi_{n,k} = \frac{1}{n}$ because of our earlier observation on the random variable K. The quantity p_n represents a 'toll' incurred at each recursive call which is the cost (measured in the number of comparisons) for partitioning, and with some standard implementations, we may take $p_n = (n-1)$. Knuth [55, p. 120] explains how to manipulate such recurrences. Reducing summations, and solving a linear equation of order 1, we arrive at an exact solution,

$$\bar{Q}_n = 2(n+1)H_{n+1} - 4n - 2 \qquad \text{with} \qquad H_n = \sum_{j=1}^{n} \frac{1}{j}.$$

Approximating the discrete sum by an integral,

$$H_n \equiv \sum_{j=1}^{n} \frac{1}{j} \sim \int_1^n \frac{dt}{t} \equiv \log n,$$

produces the asymptotic form of the cost,

$$\bar{Q}_n \sim 2n \log n. \tag{2}$$

In this way, Quicksort is proved to be about 40% off from the information theoretic optimum of $n \log_2 n$, which is the final conclusion sought.

Quicksort and generating functions. There is an alternative approach to this problem which in such a simple case seems to be a mere variant of the analysis, but nonetheless reveals some important features of the approaches we plan to discuss here.

Introduce the *generating function* (GF) of the mean values

$$Q(z) = \sum_{n=0}^{\infty} \bar{Q}_n z^n, \tag{3}$$

and set similarly $p(z) = \sum_{n \geq 0} p_n z^n$. Then, the equation corresponding to the recurrence (1) is

$$Q(z) = p(z) + 2 \int_0^z Q(t) \frac{dt}{1-t}. \tag{4}$$

(This could be checked by multiplying in Eq. (1) by z^n and summing over n, employing the usual rules of generating function manipulations [44].) The differential equation, corresponding to (4),

$$\frac{d}{dz} Q(z) = \frac{d}{dz} p(z) + 2 \frac{Q(z)}{1-z},$$

is solved by the usual techniques,

$$Q(z) = \frac{1}{(1-z)^2} \int_0^z \frac{d}{dt} \{p(t)\} (1-t)^2 dt. \tag{5}$$

This integral transform expresses the global cost of the algorithm in terms of the local cost incurred at each recursive call. In the particular case of Quicksort, this leads to the solution: we have $p(z) = z^2/(1-z)^2$, and, carrying out the integration,

$$Q(z) = 2 \frac{\log(1-z)^{-1}}{(1-z)^2} - \frac{2z}{(1-z)^2}. \tag{6}$$

If we expand $Q(z)$, we retrieve again the form of $\bar{Q}_n$ that involves the harmonic numbers.

The solution expressed by (6) can be used to produce direct asymptotic results from the generating function itself, without any need for explicit expansions. The key observation is that it suffices to examine the generating function *locally* near its *singularity* at $z = 1$ and apply systematic translation mechanisms described in Section 2. Letting $[z^n]f(z)$ denote the coefficient of z^n in the generating function $f(z)$, a single rule

$$[z^n]\ \frac{1}{(1-z)^2}(\log(1-z)^{-1}) \sim n \log n$$

will give us the $2n \log n$ result directly.

The generating function approach is the one that leads to higher level generalizations applicable to more complicated algorithms and cost measures.

First, in a suitable framework, the structure of the equation is seen to be a *direct translation* of the specification of the algorithm, and we discuss such aspects in Section 1. The rules exemplified here concern sequential execution and recursive descent into smaller subfiles (with the suitable probability distribution $\pi_{n,k} = 1/n$). Informally, the two rules are

$$\begin{array}{lcl} \text{Sequential execution: } \texttt{F; G} & \Longrightarrow & F(z) + G(z) \\ \text{Recursive descent with } \texttt{F} & \Longrightarrow & \displaystyle\int_0^z F(t)\,\frac{dt}{1-t}. \end{array}$$

Such correspondences cover a wide range of problems: *Generating functions of wide classes of combinatorial structures and algorithms can be determined from formal specifications.* The character of these correspondences is systematic enough that we may even use computer algebra programmes to compute the generating function equations automatically, a fact that is explored in Section 1 and further discussed in Part III.

Next, the translation from the local singular behaviour of a function to the asymptotics of its coefficients is a powerful mechanism. General rules valid under simple conditions (analytic continuation) apply, like for instance, the relation

$$[z^n]\ \frac{1}{(1-z)^\alpha}(\log(1-z)^{-1})^k \sim \frac{n^{\alpha-1}}{\Gamma(\alpha)}(\log n)^k.$$

If the partitioning cost of Quicksort becomes of the form $\sqrt{n}$ (due to parallel execution perhaps) a direct asymptotic analysis is still feasible (and easy!), despite the fact that the explicit coefficient expressions become more involved. The general principle is the following: *Generating functions need only be studied locally near their singularities.* Again, this systematic process presented in Section 2 can be subjected to automatic analysis.

Quadtrees. The generating function approach therefore allows for a unified discussion of a whole range of related problems. As a further illustration, consider an analogous two–dimensional problem, namely the analysis of path length in standard quadtrees [26, 49]. In such a tree, there are two successive descents (the first one in a half plane based on the x–coordinate, the second one in a quadrant determined by the y–coordinate). Accordingly, the single integral of (5) gets replaced by a double integral,

$$Q(z) = p(z) + 4\int_0^z \left[\int_0^t Q(u)\,\frac{du}{(1-u)}\right]\,\frac{dt}{t(1-t)}. \tag{7}$$

The associated differential equation is now of order 2. Thanks to a relation to special hypergeometric functions, we still have an explicit solution available,

$$Q(z) = \frac{(1+2z)}{(1-z)^2} \int_0^z \frac{(1-t)^3}{t(1+2t)^2} \left[\int_0^t \frac{(1+2v)}{(1-v)^2} \frac{d}{dv} \{ v(1-v) \frac{d}{dv} p(v) \} \, dv \right] \, dt.$$

Given the principles of singularity analysis, it suffices to examine locally near $z = 1$ the effect of this integral expression, viewing it as "singularity transformer".

Complex analysis asymptotics render the analysis of such GF's really simple although coefficients soon turn out to have intractable expressions. For quadtrees in higher dimensions $d > 2$, the integral equation is of order d and does not even admit of closed form solutions any more. However, singularities of such direct and inverse operators can still be studied by appealing to the classification of singularities of linear differential systems, and complete asymptotic solutions are available. This illustrates a further feature of the theory: *Generating functions may be analyzed even in cases where they admit of no closed form.*

In this way, the cost of partial match and exact match queries in quadtrees of all dimensions has been precisely quantified [26]. It is found that an exact match query in a quadtree of size n and dimension d has cost asymptotic to $\frac{2}{d} \log n$.

Part I: Methods

1 Symbolic Methods in Combinatorial Analysis

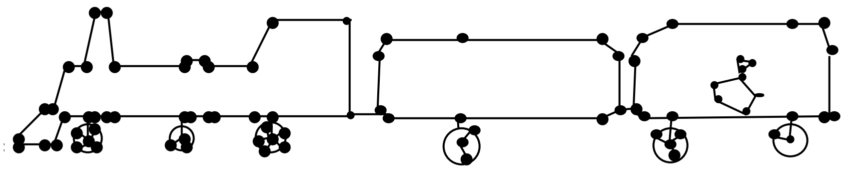
Laplace discovered the remarkable correspondence between set theoretic operations and operations on formal power series and put it to use with great success to solve a variety of combinatorial problems.

— G.–C. Rota

Early practitioners of combinatorial analysis often realized that certain types of counting problems would invariably lead to definite types of generating functions. The systematization of these scattered observations had to wait a bit, however. First, in the late 1950's, Chomsky and Schützenberger discovered that enumerative problems described by regular languages or finite automata lead to rational generating functions, while algebraic functions correspond to (unambiguous) context free languages. Then, Rota and his school on one side, Foata and Schützenberger on an other side, came to general frameworks that would 'explain' such correspondences. Later Joyal [50] with the theory of species, as well as Goulden and Jackson [43] produced frameworks of comparable power. In a remarkable thesis, Greene [45] developed a notion of labelled grammars with a focus on order constraints and analysis of algorithms. Flajolet and Steyaert developed rules initially specialized to trees from which a 'complexity calculus' could be derived for a wide class of algorithms [22, 24, 36, 37, 75]. This was later extended into a much more general system [32] to be discussed in Part III.

We propose now to explain the major principle of a symbolic approach to the derivation of generating functions.

1. Trains.



2. Formal specification.

$$\left\{\begin{array}{l}
\mathit{train} = (\mathit{locomotive} * \mathit{wagons}) \\
\mathit{wagons} = \text{sequence}(\mathit{wagon}) \\
\mathit{locomotive} = \text{sequence}(\mathit{slice}) \\
\mathit{slice} = (\mathit{upper} * \mathit{lower}) \text{ union } (\mathit{upper} * \mathit{lower} * \mathit{wheel}) \\
\mathit{wagon} = (\mathit{locomotive} * \mathit{passengers}) \\
\mathit{passengers} = \text{set}(\mathit{passenger}) \\
\mathit{passenger} = (\mathit{head} * \mathit{belly}) \\
\mathit{wheel} = \text{cycle}(\mathit{wheel_element}) \\
\mathit{head} = \mathit{belly} = \text{cycle}(\mathit{passenger_element}) \\
\mathit{upper} = \mathit{lower} = \mathit{wheel_element} = \mathit{passenger_element} = \mathit{point}.
\end{array}\right.$$

3. Generating function equations (excerpts).

```
train(z):=locomotive(z)*wagons(z);
wagons(z):=Q(wagon);                    % Q(u):=1/(1-u);
locomotive(z):=slice*Q(slice);
slice(z):=upper*lower+upper*lower*wheel;
wheel(z):=center*L(wheel_element);     % L(u):=log(1/(1-u));
..........................
wheel_element(z):=z;
passenger_element(z):=z;
```

4. Explicit generating function.

$$\frac{z^2 - z^3 \ln(1 - z)}{(1 - z^2 + z^3 \ln(1 - z))\left(1 - \dfrac{(z^2 - z^3 \ln(1 - z)) \exp(\ln(1 - z)^2)}{1 - z^2 + z^3 \ln(1 - z)}\right)}$$

Figure 1. The example of 'random trains' illustrates the power of symbolic methods in combinatorial analysis. We define a complex combinatorial structure [§1] that is formed with sequences, cycles, and sets. A formal specification [§2] is easily set up. From it, generating functions are computable systematically, and a system of equations is compiled from the specification [§3]. The generating function is then solved explicitly [§4].
Currently, the analysis of this problem can be achieved automatically. A system, Lambda–Upsilon–Omega ($\Lambda\Upsilon\Omega$), has been designed by B. Salvy and P. Zimmermann jointly with the author [32, 69, 84]. It does the analysis and via an implementation of complex asymptotic methods and singularity analysis, it is also able to find automatically the asymptotic form of the coefficients: The number of trains of size n satisfies the estimate

$$\frac{\text{train}_n}{n!} \sim 0.07097007911 \cdot 1.930298068^n .$$

Principle. *A number of set–theoretic constructions like union, cartesian product, sequence set, cycle set, power set, substitution have direct tanslation into generating function equations. Thus, a counting problem which is expressible in the language of these constructions can be translated systematically (and automatically) into generating function equations.*

Combinatorial structures to be discussed here fall into two types; the *well–labelled* structures which are graph complexes in which nodes are labelled by distinct integers (from 1 to n when the structure comprises n nodes) and unlabelled ones. Examples of labelled structures are labelled trees, permutations (when viewed as collections of labelled cyclic graphs), etc. Unlabelled trees, formal languages are examples of unlabelled objects.

Given a class $\mathcal{F}$ of combinatorial structures, we let $\mathcal{F}_n$ denote the collection of objects of size n, and set $F_n = \mathrm{card}(\mathcal{F}_n)$. The *ordinary generating function* (OGF) and *exponential generating function* (EGF) are defined respectively to be

$$F(z) = \sum_{n\geq 0} F_n z^n \qquad \text{and} \qquad \hat{F}(z) = \sum_{n\geq 0} F_n \frac{z^n}{n!}. \tag{8}$$

A combinatorial construction is *admissible* if it admits a translation into generating functions.

The following two theorems are well known under one form or the other. They embody a powerful collection of combinatorial constructions. For detailed definitions, the reader is referred to modern treatments of the subject [15, 42, 43, 72, 74, 81] or to the paper [32] where a similar system of notations is developed.

Theorem 1 (Admissible constructions for OGF's) *For unlabelled structures, the constructions of union, cartesian product, sequence, cycle, set, multiset, substitution are admissible. The translations into ordinary generating functions are given by the following table*

Construction	Translation (OGF)
$\mathcal{F} = \mathcal{G} \cup \mathcal{H}$	$F(z) = G(z) + H(z)$
$\mathcal{F} = \mathcal{G} \times \mathcal{H}$	$F(z) = G(z) \cdot H(z)$
$\mathcal{F} = \mathtt{sequence}(\mathcal{G}) = \mathcal{G}^*$	$F(z) = \frac{1}{1-G(z)}$
$\mathcal{F} = \mathtt{set}(\mathcal{G})$	$F(z) = \exp(G(z) - \frac{1}{2}G(z^2) + \frac{1}{3}G(z) - \cdots)$
$\mathcal{F} = \mathtt{multiset}(\mathcal{G})$	$F(z) = \exp(G(z) + \frac{1}{2}G(z^2) + \frac{1}{3}G(z) + \cdots)$
$\mathcal{F} = \mathtt{cycle}(\mathcal{G})$	$F(z) = \log(1 - G(z))^{-1} + \cdots$
$\mathcal{F} = \mathcal{G}[\mathcal{H}]$	$F(z) = G(H(z))$

Theorem 2 (Admissible constructions for EGF's) *For labelled structures, the constructions of union, partitional product, sequence, cycle, set, substitution are admissible. The translations into exponential generating functions are given by the following table*

Construction	Translation (EGF)
$\mathcal{F} = \mathcal{G} \cup \mathcal{H}$	$\hat{F}(z) = \hat{G}(z) + \hat{H}(z)$
$\mathcal{F} = \mathcal{G} * \mathcal{H}$	$\hat{F}(z) = \hat{G}(z) \cdot \hat{H}(z)$
$\mathcal{F} = \mathtt{sequence}(\mathcal{G}) = \mathcal{G}^*$	$\hat{F}(z) = \frac{1}{1-\hat{G}(z)}$
$\mathcal{F} = \mathtt{set}(\mathcal{G})$	$\hat{F}(z) = \exp(\hat{G}(z))$
$\mathcal{F} = \mathtt{cycle}(\mathcal{G})$	$\hat{F}(z) = \log(1 - \hat{G}(z))^{-1}$
$\mathcal{F} = \mathcal{G}[\mathcal{H}]$	$\hat{F}(z) = \hat{G}(\hat{H}(z))$

2 Complex Analysis and Asymptotics

Es ist eine Tatsache, daß die genauere Kenntnis des Verhaltens
einer analytischen Funktion in der Nähe ihrer singulären Stellen
eine Quelle von arithmetischen Sätzen ist.

— E. Hecke

Complex analytic methods permit to represent coefficients of generating functions and many combinatorial sums as integrals of an analytic function in the complex plane. The choice of a suitable contour of integration often leads to highly non trivial asymptotic results. A thorough review of theses techniques appears in [38]. Other excellent references are [5, 16, 48, 66].

The first part of this section is devoted to singularity analysis techniques which make it possible to derive estimates on the coefficients of generating functions starting from Cauchy's formula,

$$f_n \equiv [z^n]f(z) = \frac{1}{2i\pi} \oint f(z) \frac{dz}{z^{n+1}}. \tag{9}$$

Singularity analysis. Most functions occurring in combinatorial enumeration problems are built by operators from standard functions that exist over the whole of the complex plane. They thus tend to exist in larger areas of the complex plane. The method of *singularity analysis* is well suited to extracting coefficients of functions lying in a class that enjoys interesting closure properties.

Definition 1 A function analytic at the origin is *star continuable* iff it has a finite number of singularities $\zeta_j = \rho e^{i\theta_j}$ on its circle of convergence $|z| = \rho$ and if for some $\epsilon < \frac{\pi}{2}$ and $\eta > 0$ it is continuable in $|\mathrm{Arg}(ze^{-i\theta_j} - \rho)| > \epsilon$ and $|z| \le \rho + \eta$.

An *algebraic–logarithmic* element is a formal series

$$F(u) = \sum_{k=0}^{\infty} c_k(\log(\frac{1}{u}))u^{\alpha_k},$$

where the α_k satisfy $\Re(\alpha_1) \le \Re(\alpha_2) \le \cdots$, $\Re(\alpha_k) \to \infty$ and each $c_k(x)$ is a polynomial.

A function is *algebraic–logarithmic* iff it is star continuable and near each singularity ζ_j it admits an asymptotic expansion $f(z - \zeta_j) \sim F_j(z - \zeta_j)$, where $F_j(u)$ is an algebraic–logarithmic element.

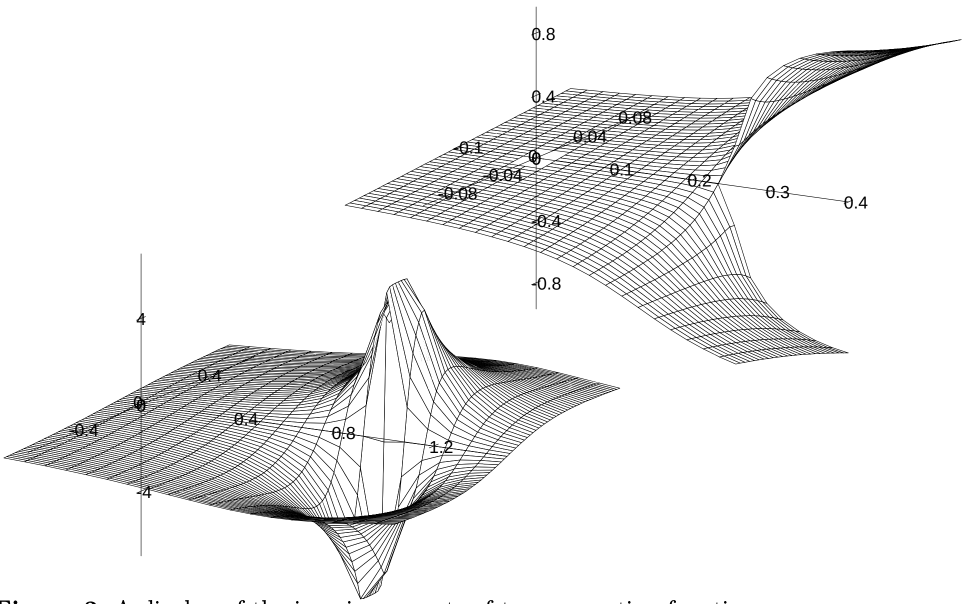


Figure 2. A display of the imaginary parts of two generating functions,

$$f(z) = \frac{1-\sqrt{1-4z}}{2z} \qquad \text{and} \qquad g(z) = \frac{1}{1-z}.$$

The function $f(z)$ [top] is the ordinary generating function of binary trees with a singularity at $\rho = 1/4$ which is a branch point of the $\sqrt{\ }$ type. The function $g(z)$ [bottom] is the exponential generating function of permutations with a singularity at $\rho = 1$ of a polar type. The singularities are reflected at the level of coefficients,

$$[z^n]f(z) \sim \frac{4^n}{\sqrt{\pi n^3}} \qquad \text{and} \qquad [z^n]g(z) = 1.$$

The first theorem summarizes a variety of results known since about the time of Hadamard [19, 78].

Theorem 3 (Analytic Closure Theorem) *Star continuable functions and algebraic-logarithmic functions are closed under sum, product, Hadamard product*[1]*, integration, and differentiation.*

Star continuable functions and algebraic–logarithmic functions thus enjoy rich closure properties. There is a direct relation between singular expansions and coefficient expansions (see Fig. 2 for an illustration), and the coefficients of algebraic–logarithmic functions can be determined systematically.

Theorem 4 (Singularity analysis) *An asymptotic form of coefficients of algebraic–logarithmic functions is obtained by termwise translation of coefficients*

[1] The Hadamard product of $f(z)$ and $g(z)$ is their term by term product $f \odot g(z) = \sum_n f_n g_n z^n$.

of elements, using the rules ($\alpha \notin \{0, -1, -2, \ldots\}$)

$$[z^n](1-z)^{-\alpha}(\log \frac{1}{1-z})^\beta \sim \frac{n^{\alpha-1}(\log n)^\beta}{\Gamma(\alpha)}\left[1 + \frac{C_1}{1!}\frac{\beta}{\log n} + \frac{C_2}{2!}\frac{\beta(\beta-1)}{\log^2 n} + \cdots\right],$$

$$\textit{where} \qquad C_j = \Gamma(\alpha)\left.\frac{d^j}{ds^j}\frac{1}{\Gamma(s)}\right|_{s=\alpha}, \quad \Gamma(s) = \int_0^\infty e^{-t}t^{s-1}dt.$$

This theorem [29] allows for a vast number of generalizations based on terminating expansions, partial expansions involving $O(.)$ or $o(.)$ error terms, as well as iterated logarithms or functions of slow variation. It is inspired by the classical method of Darboux in asymptotic analysis and by Tauberian theorems. Its principle however relies on integration using contours of the Hankel type.

The applications are numerous. Most of the analysis carried out by Darboux's method can usually be conducted in a transparent way using this method. Instances are uniform tree models and the so–called simple families of trees [64], and generally enumeration problems expressible in terms of context–free languages.

Saddle point integrals. Given an analytic function $f(z)$, a *saddle point* is a point ζ which cancels the derivative, $f'(\zeta) = 0$. This terminology is due to the topography of the modulus of the function near ζ which resembles the inside part of a saddle. When computing a complex integral

$$\frac{1}{2i\pi}\int_{\mathcal{C}} e^{g(z)}\,dz,$$

it is often the case that the contour $\mathcal{C}$ can be deformed so as to traverse a saddle point, with the value at the saddle point being a local maximum for the modulus. Under suitable conditions, the integral is concentrated near the saddle point. In that case, a local expansion of the analytic integrand holds, and one finds the approximation

$$\frac{1}{2i\pi}\int_{\mathcal{C}} e^{g(z)}\,dz \sim \frac{1}{\sqrt{2\pi g''(\zeta)}}e^{g(\zeta)}.$$

When the method applies, it is often said that the integral is (saddle point) *admissible.* Most notably, the method is useful for the computation of coefficients of whole classes of entire functions [67]. In that case, we should take $g(z) = \log f(z) - (n+1)\log z$, where $f(z)$ is the function to be analyzed.

Theorem 5 (Saddle point coefficient asymptotics) *For an admissible function $f(z)$, one has*

$$[z^n]f(z) \sim \frac{f(\zeta)}{\sqrt{2\pi C}\zeta^{n+1}} \quad \textit{where} \;\; C = \left.\frac{d^2}{dz^2}\log\frac{f(z)}{z^{n+1}}\right|_{z=\zeta},$$

where $\zeta = \zeta_n$ is the smallest real root of $\frac{d}{dz}\log\frac{f(z)}{z^{n+1}}$.

This classical method originating in applied mathematics is discussed in de Bruijn's book [16], and in the context of coefficient extraction in [67, 81]. A review appears in [69]. Typical applications are to fast growing functions like

$$e^{z/(1-z)},\ e^{z},\ e^{e^{z}-1},\ e^{z+z^2/2},\ldots.$$

This covers problems related to increasing subsequences in permutations, set partitions, involutions, etc.

Saddle point methods also serve to analyze coefficients of large powers of functions. This is one way of establishing the central limit theorem (together with error bounds), a fact well explained in [46]. There are also numerous applications to hashing and occupancy problems. Difficult questions present themselves with higher dimensional saddle point problem. Gardy has obtained several general results on urn models [39, 40] and attained a precise quantification of the size of relational algebra operations applied to collections of random points, using two–dimensional saddle point techniques. McKay and his collaborators have pushed the analysis to situations where a counting problem of a large size n leads to an n–dimensional complex integral. For instance,

$$\frac{1}{(2i\pi)^n}\oint\oint\cdots\oint\prod_{1\le i<j\le n}(1+z_iz_j)\,\frac{dz_1dz_2\cdots dz_n}{z_1^{k+1}z_2^{k+2}\cdots z_n^{k+1}},$$

gives the number of k–regular graphs, and the integral can be estimated using an n–dimensional saddle point integral while the dimension of the space n tends to infinity! See [63] for these promising techniques.

Part II: Classes of Applications

In this part, we explain how the methods introduced can be put to use in order to analyze whole *classes* of problems relative to combinatorial structures and algorithms. We limit ourselves to a descriptive inventory that tries to put in perspective a vast body of literature.

3 Algebraic Functions and Implicit Functions

Regular languages can be specified either by regular expressions or by finite automata. The corresponding GF's either appear as built from the variable z by means of rational operations ($+$, $\times$, quasi–inverse $Q(y)=(1-y)^{-1}$) or as components of linear systems of equations (over $\mathbf{Z}[x]$). At any rate, they are rational [7]. Examples are

$$\cfrac{z}{1-\cfrac{z}{1-\cfrac{z}{1-\cfrac{z}{1-z}}}},\quad \frac{1-z^{k+1}}{1-2z+z^{k+1}},\quad \frac{z^4}{1-4z+4z^2+z^3-3z^4+2z^5},$$

representing the OGF of plane unlabelled trees (Dyck words) of height at most 4, of binary strings with no runs of more than k consecutive ones, and of binary strings containing the pattern 0100.

An immediate consequence of the partial fraction decomposition of rational functions is the following.

Theorem 6 (Rational Asymptotics) *The coefficients of a rational function of* $\mathbf{Q}(z)$ *are a finite linear combination of 'exponential polynomials' of the form*

$$\lambda \omega^n n^k, \tag{10}$$

with λ, ω *algebraic numbers and* k *an integer.*

By grouping the ω's in order of decreasing modulus, this has the character of an asymptotic expansion. In this way, a counting problem relative to a single regular language normally poses no difficulty and falls into a decidable class of problems.

Interesting problems arise from consideration of parameterized sets of rational functions, like trees of bounded height [17], longest runs of ones and carry propagation [56], or occurrences of prefixes of an infinite pattern sequence [65]. The analysis of the clustering of dominant roots, which will have accumulation points, has made it possible to analyze the expected height of trees, and the expected time for carry propagation in binary adders.

Context free languages lead to polynomial nonlinear equations, provided the grammar is unambiguous or we count words with their multiplicities. Thus, the generating function of a context free language is algebraic. This is the famous Chomsky–Schützenberger theorem. By standard elimination theory, such a function satisfies a single polynomial equation,

$$P(z, f(z)) = 0. \tag{11}$$

Near a singularity, an algebraic function admits an expansions into fractional powers of the form $(1 - z/\rho)^{p/q}$, which is also called a Puiseux expansion. The method of singularity analysis applies well to Puiseux elements.

Theorem 7 (Algebraic Asymptotics) *The coefficients of a* $\mathbf{Q}(z)$*–algebraic function are asymptotic to a sum of 'algebraic elements' of the form*

$$\frac{\lambda}{\Gamma(r/s+1)} \omega^n n^{r/s}, \tag{12}$$

where λ, ω *are algebraic numbers, and the exponent* r/s *is a rational number.*

This furnishes a generalized density theorem for context free languages and was used in [23] in order to establish the inherent ambiguity of several context free languages.

Similar singular expansions involving fractional powers also hold for functions implicitly defined by equations of the form $\Phi(z, f(z)) = 0$, where Φ, analytic function of two complex variables, need no longer be a polynomial. By the implicit function theorem, singularities are almost invariably of the $\sqrt{\ }$ type, so that coefficients involve the rational exponent $r/s = -3/2$. Examples are

$$y = z(1 + y^t), \quad y = ze^y, \quad e^y - 2y - 1 + z = 0.$$

The first equation counts regular (plane, unlabelled) t–ary trees; the second one labelled non plane trees whose number is n^{n-1}, a famous result due to Cayley; the third one Schröder's partition systems [15, p. 224].

Implicit functions. Meir and Moon have developed in a series of papers (see, e.g., [64]) a general theory of statistics on 'simple families of trees' corresponding to equations of the form $y - z\phi(y) = 0$. Under general conditions, the singularities are again of the $\sqrt{\ }$ type, which leads again to asymptotic forms

$$\lambda \omega^n n^{-3/2}.$$

Using this theory, it is found that various families of trees share common features; for instance, path length is of order $n^{3/2}$ on average.

Variations around this theme have led to the analysis of a large number of tree algorithms in the context of symbolic manipulations. We shall cite here: pattern–matching [76, 2], simplification [10], unification [1], common subexpression factorization [34], and term rewriting techniques [11]. See also [9] for an interesting survey.

Another important case of application is to the Cayley function $y = ze^y$. This function shows up in the enumeration of labelled trees, random mappings [28], in the analysis of hashing with linear probing [55], in union find problems [57], in caching algorithms, etc.

A further extension concerns the enumeration of unlabelled non plane trees that involves the operators of Theorem 1 (see the set and multiset constructions). The GF of such graphical trees satisfies a functional relation known to Cayley,

$$f(z) = z \exp\left(f(z) + \frac{1}{2} f(z^2) + \frac{1}{3} f(z^3) + \cdots \right).$$

Although no closed form is available for this GF, it can still be subjected to the same treatment as implicitly defined functions, a general fact discovered by Pólya in his famous 1937 paper [68]: Once it has been recognized that $f(z)$ has radius of convergence $\rho < 1$, the terms in the sum involving $f(z^2), f(z^3)$, etc, being analytic near $z = \rho$, can be treated as 'known' perturbations for all asymptotic purposes. In this way, it is also possible to analyze the GF's

$$f(z) = z + \frac{1}{2} f^2(z) + \frac{1}{2} f(z^2), \quad f(z) = \cfrac{1}{1 - \cfrac{z}{1 - \cfrac{z^2}{1 - \cfrac{z^4}{1 - \cdots}}}},$$

relative to non plane unlabelled binary trees (Otter, 1948) and to structurally isomeric alcohols $C_nH_{2n+1}OH$ without asymmetric carbon atoms. The asymptotic forms found are

$$2.9557 \cdot 0.4399^n n^{-3/2}, \quad 0.3187 \cdot 2.4832^n n^{-3/2}, \quad 0.3067 \cdot 1.6813^n,$$

for Cayley's graphical trees, binary trees and alcohols, respectively [47].

In summary, functions defined implicitly tend to have singularities like those of algebraic functions, involving fractional exponents. This is reflected by the asymptotics of their coefficients of the form $\omega^n n^{-r/s}$. Such a property also holds for many functions satisfying finite and infinite functional equations involving terms like $f(z^2), f(z^3)$ provided that their radius of convergence is < 1.

4 Holonomic Functions and Differential Equations

When discussing the analysis of Quicksort, we have encountered a particular case of the general *probabilistic divide and conquer* schema

$$f_n = e_n + \sum_{k=0}^{n-1} \pi^*_{n,k} f_k. \tag{13}$$

There f_n is the sequence to be analyzed, e_n is a fixed toll sequence, and the $\pi^*_{n,k}$ are proportional to the splitting probabilities $\pi_{n,k}$ that express the chances that a task of size n involve a subtask of size $k < n$.

In comparison based sorting and searching, it is often the case that the $\pi_{n,k}$ involves some rational combination of n and k, a fact also well accounted for by Greene's treatment of 'min-rooting' operators [45]. Naturally occurring examples of $\pi_{n,k}$'s are

$$\frac{1}{n}, \ \frac{2(n-k)}{n(n+1)}, \ \ \frac{1}{n}[H_n - H_k], \ \ \frac{2(k-1)(n-k)}{n(n-1)(n-2)},$$

that arise in Quicksort (or binary search trees), fully specified search in 2-d quadtrees, partial match queries in 2-d quadtrees, and median–of–three Quicksort [55, p. 609]. In that case the translation into generating functions leads to integral equations of which (4,7) are typical. This in turn leads to generating functions satisfying differential equations with rational (or equivalently polynomial) coefficients. (Only in simpler cases is the equation an Euler equation that admits elementary solutions.)

Functions satisfying differential equations with polynomial coefficients are sometimes called $\mathcal{D}$–finite and their coefficient sequences which satisfy recurrences with polynomial (in n) coefficients are then called $\mathcal{P}$–recursive. These notions are formalized by the concept of *holonomy* introduced in this range of problems by Zeilberger.

Definition 2 A series $f(z_1, z_2, \ldots, z_r) \in \mathbf{C}[[z_1, z_2, \ldots, z_r]]$ is said to be *holonomic* iff the infinite collection of its partial derivatives

$$\frac{\partial^{j_1}}{\partial z^{j_1}} \frac{\partial^{j_2}}{\partial z^{j_2}} \cdots \frac{\partial^{j_r}}{\partial z^{j_r}} f(z_1, z_2, \ldots, z_r)$$

span a finite dimensional vector space over the field of rational fractions $\mathbf{C}(z_1, z_2, \ldots, z_r)$.

A sequence $f_{n_1,n_2,\ldots,n_r}$ is holonomic iff its generating function $f(z_1, z_2, \ldots, z_r) = \sum_{n_1,n_2,\ldots,n_r} f_{n_1,n_2,\ldots,n_r} z_1^{n_1} z_2^{n_2} \cdots z_r^{n_r}$ is holonomic.

The major closure theorem here is due to Stanley, Lipschitz, and Zeilberger [59, 60, 73, 83].

Theorem 8 (Holonomic Closure) *Holonomic functions are closed under sums, products, Hadamard products, diagonals, algebraic substitutions, integration, differentiation, direct and inverse Laplace transforms.*

Coefficients sequences enjoy the corresponding closure properties. For instance, closure under sum, product, convolution, summation, multiplication and division by polynomials in n. Many combinatorial quantities that are expressible as multiple summations of multinomial coefficients with linear constraints (these are sometimes called 'multihypergeometric') are in particular holonomic. Using a theory of holonomic symmetric functions, Gessel [41] has established that the generating functions of k–regular graphs and k–Latin rectangles are holonomic. With the closure theorem, Massazza [62] has shown that context-free languages with linear constraints on the number of occurrences of letters have holonomic generating functions. Finally, the cost sequences for usual variants of Quicksort and comparison–based search are clearly holonomic.

A major interest of holonomic sequences is that the identities they satisfy form a decidable class. In the 1–dimensional case, the corresponding asymptotic properties are also in essence decidable. The spirit of the available result is captured by the following informally stated theorem.

Theorem 9 (Holonomic Asymptotics) *A holonomic sequence f_n is asymptotic to a sum of elements of the form*

$$\lambda(n!)^{r/s} e^{Q(n^{1/m})} \omega^n n^\alpha (\log n)^k,$$

where r, s, m, k are integers, Q is a polynomial and λ, ω, α are complex numbers.

This theorem is originally due to Birkhoff and his students, and we refer to the useful discussion that Wimp and Zeilberger gave in [82]. The original proof is based on a direct treatment of difference equations in the complex plane.

In our perspective, this theorem relates to the classification of singularities of linear differential equations. The theory of linear differential equations with analytic coefficients [80], distinguishes for solutions of such equations two cases, the regular case and the irregular case. Singular expansions of solutions are then locally composed of elements of one of two types,

$$(1 - z/\rho)^{\alpha'} (\log(1 - z/\rho))^{k'}, \quad (1 - z/\rho)^{\alpha'} (\log(1 - z/\rho))^{k'} \exp(\tilde{Q}(1 - z/\rho)^{1/m'}).$$

The method of singularity analysis and the method of saddle point integrals are applicable each in one of the two cases. The resulting forms found for coefficients are exactly the ones stated in the theorem on holonomic asymptotics.

The analysis of regular singularities has given results on various multidimensional search problems in k-d trees and quadtrees [30, 26]. For instance the expected cost of a partial match query in a quadtree (alternatively a k-d-tree) when a proportion of $\frac{1}{2}$ or $\frac{2}{3}$, of the coordinates is of the order of

$$n^{(\sqrt{17}-3)/2} \qquad \text{and} \qquad n^{\theta-1} \text{ with } \theta = \left(\frac{109}{27} + \sqrt{\frac{1320}{81}}\right)^{1/3} + \left(\frac{109}{27} - \sqrt{\frac{1320}{81}}\right)^{1/3}.$$

Such algebraic numbers in the exponents (!) are typical of $\mathbf{Q}(z)$ holonomic functions.

Irregular singularities occur more seldom, a clear example being the number of increasing subsequences of a random permutation of n elements which was determined by Pittel and Lifschitz [58] and is asymptotic to

$$\frac{1}{2\sqrt{e\pi}} n^{-1/4} e^{2\sqrt{n}}.$$

In particular, the longest increasing subsequence of the random permutation is proved to have length $O(\sqrt{n})$. (It is actually known to be asymptotic to $2\sqrt{n}$.)

In summary recurrences involving rational functions and summations can usually be treated by means of the theory of singularity of linear differential equations applied to GF's and combined with singularity analysis or saddle point techniques. In essence, such problems fall into a decidable class.

5 Functional Equations and Iteration

We confine our discussion to *linear functional equations* of the form

$$f(z) = a(z) + b(z) f(\sigma(z)), \tag{14}$$

where $f(z)$ is the unknown function, and a, b, σ are explicitly known. The behaviour depends on the iteration structure of $\sigma(z)$. The formal solution to (14) is found by iteration,

$$f(z) = \sum_{k=0}^{\infty} a(\sigma^{\langle k \rangle}(z)) \frac{B(\sigma^{\langle k \rangle}(z))}{B(z)} \qquad \text{with } B(z) = \prod_{j=0}^{\infty} B(\sigma^{\langle j \rangle}(z)), \tag{15}$$

$\sigma^{\langle k \rangle}(z)$ being the k-th iterate of σ.

In the functional equation of (14), everything depends crucially on the dynamics of the iterates of σ. In a few important cases, the iterates are explicit, and one general method available relies on the Mellin transform, some of whose uses are recalled below. In the case of non explicit iterates, singularity analysis or saddle point techniques have to be applied.

Mellin transforms. Mellin transforms constitute another set of techniques based on complex analysis methods. The Mellin transform of a real function $f(t)$ is defined by

$$f^*(s) = \int_0^{\infty} f(t) t^{s-1} dt.$$

A harmonic sum is a function of the form $\sum_k \lambda_k \varphi(\mu_k x)$, where $\varphi(x)$ is a base function, and the λ_k, μ_k play the rôles of amplitudes and frequencies. Using the inversion theorem for Mellin transforms and simple functional properties, we arrive at a method to evaluate harmonic sums asymptotically.

Theorem 10 (Mellin summation of harmonic sums) *Under conditions of meromorphic continuation and smallness at $\pm i\infty$,*

$$\sum_k \lambda_k \varphi(\mu_k x) \sim \pm \sum_{\zeta \in H} \operatorname{Res} \left[\sum_k \frac{\lambda_k}{\mu_k^s} \cdot \int_0^\infty \varphi(t)\, t^{s-1} dt \right]_{s=\zeta},$$

where H is a left half plane (resp. a right half plane), the sum of residues is over poles ζ in the half plane, the sign is $+$ (resp. $-$), and the expansion applies as $x \to 0^+$ (resp. $x \to +\infty$).

The method of Mellin transforms in disguise lies at the heart of the proof of the prime number theorem (with additional difficulties resulting from the occurrence of the Riemann zeta function!). It is well suited to sums that occur in the analysis of algorithms, like

$$\sum_k 2^{-k} \varphi(x 2^{-k}), \quad \sum_k v_2(k) \varphi(kx), \quad \sum_k d(k) \varphi(kx),$$

where $\varphi(x)$ is often an exponential function e^{-x}, e^{-x^2}, and the coefficients are either powers of two or elementary arithmetic functions (e.g., the dyadic valuation $v_2(k)$ or the divisor function $d(k)$).

This transform was first introduced in our range of problems by de Bruijn, Knuth and Rice [17] for the purpose of analyzing the height of general plane trees (the expression involves the divisor function) which appears to be $\sim \sqrt{\pi n}$. It has also found many uses in the analysis of digital structures (prefix trees, tries, digital search trees, suffix trees), and in a large number of related areas (protocols, probabilistic counting, carry propagation, dynamic hashing).

Explicit iterations. The analysis of digital tries furnishes an example of the situation where the iteration of $\sigma(z)$ is explicit. The recurrence of expected path length in tries is of a new probabilistic divide–and–conquer type,

$$f_n = n - \delta_{n,1} + 2 \sum_{k=0}^{n} \pi_{n,k} f_k \qquad \text{with} \quad \pi_{n,k} = \frac{1}{2^n} \binom{n}{k}.$$

The corresponding EGF satisfies

$$f(z) = z(e^z - 1) + 2 e^{z/2} f(\frac{z}{2}).$$

The equation is solved by iteration, after which the solution can be expanded. The curious phenomenon occurring here is the presence of minute fluctuations in the behaviour of coefficients [55, p. 131]: The expected path length f_n of a trie of size n satisfies an estimate $n \log_2 n + n P(\log_2 n)$ where $P(u)$ has amplitude less than 10^{-5}. Such periodicities are traceable to complex poles in a Mellin transform. For instance, the transform of $\psi(x) = e^{-x} f(x)$ is

$$\psi^*(s) = \frac{\Gamma(s+1)}{1 - 2^{-s-1}},$$

which has rightmost poles at $s = -1 + 2ik\pi/\log 2$. The Mellin summation formula then leads to a Fourier series that expresses these fluctuations, and this in turn is reflected by corresponding fluctuations in the coefficients.

By now, a large number of applications have been given of this analysis technique and some of its variants (like Rice's integrals). Here, we only refer to [33, 55, 79]. Applications have been given in the area of trie searching and radix exchange sort [55], dynamic and extendible hashing [20], communication protocols [21], probabilistic counting and estimation algorithms [27], quadtries and multidimensional searching [30], suffix trees, Patricia trees and pattern matching in strings (see e.g. [77]), digital trees [31]. Surprising connections with identities of Ramanujan relative to modular forms were uncovered by Kirschenhofer and Prodinger on the occasion of variance analysis of digital structures [52, 53].

Divide–and–conquer algorithms also lead to explicitly solvable iterations, especially when Mellin transforms are used. The recurrence,

$$f_n = e_n + f_{\lfloor n/2 \rfloor} + f_{\lceil n/2 \rceil},$$

with e_n a known toll sequence, is typical. Let $f(z)$ be the corresponding OGF; the functional equations is

$$f(z) = e(z) + (2 + z + \frac{1}{z}) f(z^2).$$

Take the Mellin transform of $f(e^{-t})$. This is equivalent to considering the Dirichlet series $\phi(s) = \sum_n f_n n^{-s}$. The series involves a denominator which resembles that of tries, being $1 - 2^{-s}$. Again, this introduces complex poles and fluctuations. However, in this case, we are lead to some fractal function expressing these fluctuations. A typical case is the analysis of Mergesort [25], for which the expected cost is found to be of the form $\sim n \log_2 n - nQ(\log_2 n)$ for some fractal and periodic function $Q(u)$.

The method is applicable to wide classes of divide and conquer recurrences which are almost invariably found to give rise to periodic fluctuations involving fractals.

Implicit iterations. When the iterates $\sigma^{\langle j \rangle}(z)$ admit of no simple explicit form, one often has to resort to an analysis of individual terms in the sum (14), normally by the battery of complex analysis techniques examined so far. Odlyzko [66] considered the equation

$$f(z) = z + f(z^2 + z^3)$$

that arises when counting balanced 2-3-trees. What is needed is the behaviour of the iterates $\sigma(z) = z^2 + z^3$ near the dominant fixed point of σ, which is equal to the inverse of golden ration, $\varphi = (1 + \sqrt{5})/2$. A delicate analysis of this *singular iteration* problem eventually leads to the number of trees which is $\sim \frac{\varphi^n}{n} R(\log n)$ for some smooth periodic function $R(u)$.

A similar problem arises when analyzing the expected height of binary trees. The iteration

$$y_0(z) = 0; \quad y_h(z) = 1 + z(y_{h-1}(z))^2,$$

gives the OGF of trees whose height is bounded by h. The fixed point y_∞ is the OGF of all binary trees, i.e., the Catalan GF, $\frac{1-\sqrt{1-4z}}{2z}$. The y_h are polynomials of degree 2^{h-1}. Singularity analysis requires investigating the convergence of the y_h near the singularity $1/4$ of the fixed point y_∞. The iteration then becomes singular: for $|z| < \frac{1}{4}$, we have exponential convergence; for $z > 1/4$, there is a double exponential divergence; at $z = \frac{1}{4}$, there is slow convergence of order $O(h^{-1})$. A fine analysis then reveals a logarithmic singularity for the GF of heights, and the expected height of a binary tree with n nodes is found to be asymptotic to $2\sqrt{\pi n}$. Analogous results hold for any simple family of trees in the sense of Meir and Moon.

At the moment, a complete classification of the various cases of (14) is still lacking. Some cases appear to involve the theory of analytic iteration and some divergent series. We nonetheless have a number of useful and general tools available in the form of Mellin transforms and iteration theory of analytic functions.

Part III: Combinatorial Schemas and Automatic Analysis

We have exposed here a few general theorems in symbolic combinatorics and complex asymptotics. They make it possible to approach the analysis of entire classes of problems in combinatorial enumerations and the analysis of algorithms. We shall be brief by necessity here and refer to our paper [32] for a more detailed discussion of the implications of these general methods.

Structure theorems and schemas. A first observation suggested by the results of Part II is that *certain combinatorial and analytic mechanisms can only lead to certain designated types of asymptotic behaviours.* We have seen that regular and context–free structures only lead to exponential polynomials and algebraic asymptotic elements respectively. The probability of a gambler's ruin in $2n$ stages is associated to a context free language, and accordingly this probability is asymptotic to $1/\sqrt{\pi n}$, a typical algebraic element. In contrast, logarithms cannot occur in this range of problems.

The tools presented here are general enough that a large number of problems can be fitted into classes whose asymptotic properties are decidable. Our example of 'trains' in Fig. 1 falls into the category of elementary iterative structures a large subset of which has decidable asymptotic properties [32]. Structure theorems describe classes of syntactically specified combinatorial structures and algorithms whose asymptotic properties are decidable and expressible by a well characterized class of formulæ. Roughly speaking, such theorems seem to exist for all structures specified by the constructions of Theorems 1 and 2 excepting substitution. The full programme of making them explicit is however a delicate task that is yet to be completed.

The relation between structural combinatorics and asymptotic form can be pushed further in order to include results on probability distributions (and not simply counting as a function of the size n). The approach is to be contrasted to the 'stochastic' approach that deals directly with continuous limit models like branching processes or Brownian motion. The stochastic approach has been successful in solving a number of problems that had resisted a more analytic attack. Most notably in this category, we find works of Aldous (diffusion processes and the height of digital search trees [3]), Devroye (branching processes and the height of digital search trees [18]), Louchard (Brownian motion and interpolation search or dynamic analysis of algorithms, see e.g. [61]), as well as others.

Our perspective is different, and it attaches itself to the area of multivariate complex asymptotics. Multiple inversions are then needed in order to recover coefficients from functions, and the methods draw upon a combination of complex asymptotic techniques which have been presented here, as well as theorems in analytic probability, notably continuity theorems for characteristic functions or moment generating functions. For instance the analytic scheme $F(z,u) = \exp(uC(z))$ expresses the fact that an $\mathcal{F}$–structure is built of components of type $\mathcal{C}$. Under wide conditions, the number of $\mathcal{C}$–components in a random $\mathcal{F}$–structure will obey a law that is Gaussian in the asymptotic limit, as $n \to \infty$. Thus, a common schema covers a variety of seemingly unrelated phenomena. In this way, we find Gaussian limit laws for the number of cycles in a random permutation, the number of factors of a random polynomial over $GF(q)$, or the number of components in a random mapping of large size [35]. First results along these lines were derived by Bender, Canfield and Richmond [4, 6, 8]. A classification of some major schemas and their associated laws is given in Soria's thesis [71]. Even for a structure as complicated as random trains, it is the case that all probability distributions of various components can be characterized in their asymptotic form: Non–classical laws as well as standard laws like the Gaussian, geometric and Poisson laws appear in such a structure.

Automatic Analysis The approach of finding general decidable asymptotic properties of combinatorial structures has been prolonged. Flajolet, Salvy and Zimmermann [32] have designed a system called Lambda–Upsilon–Omega ($\Lambda\Upsilon\Omega$) that implements a number of decision procedures on combinatorial structures like the ones discussed here. The kernel specification language consists of the constructions of union, product, sequence, sets, multisets and cycles described in Section 1. The $\Lambda\Upsilon\Omega$ system also makes provisions for specifying traversal algorithms on the structures.

A first component of $\Lambda\Upsilon\Omega$ implements the automatic computation of counting generating functions and complexity descriptors that are cost generating functions. Zimmermann [84] has developed the necessary theory which builds on the principles of our Section 1, and he has also found a number of extensions most notably to boolean procedures [85], to some forms of composition, and to exact counting.

After a solving phase, the (usually complicated) generating functions produced need to be subjected to an automatic analysis of their coefficients. Salvy [69] has developed a collection of efficient decision algorithms in the style of computer algebra in order to manipulate general asymptotic scales in Hardy fields and apply transfers from functions to coefficients in the style of the methods of Section 2.

In its current stage, $\Lambda\Upsilon\Omega$ consists of some 20,000 instructions written largely in the computer algebra system Maple. It has provided so far more than 50 different analyses of combinatorial problems and algorithms related to regular languages, finite automata, random walks, term tree, rewriting systems, random mappings, and miscellaneous combinatorial problems. The system can assist experts in the analysis of combinatorial problems specifiable in this language and in a few cases, it has produced automatically results that had been published in the literature, so that its level of 'competence' is to be considered as reasonably good.

As a final conclusion, we have tried to present a global view of the analysis of classes of combinatorial objects, not unlike in spirit to what had been done earlier for formal languages and zero–one laws in combinatorics [12, 13, 14]. General results of an almost 'logical' nature relate *combinatorial structure* and *asymptotic form*. In a number of such cases, decision procedures can be found for asymptotic combinatorics. Their development and their implementation within computer algebra is a fascinating new area of investigation.

Acknowledgements. The author is grateful to Kevin Compton, Philippe Dumas, Helmut Prodinger, and Michèle Soria for detailed and constructive comments on the manuscript.
This work was supported in part by the ESPRIT III Basic Research Action Programme of the E.C. under contract ALCOM II (#7141).

References

[1] Albert, L., Casas, R., Fages, F., Torrecillas, A., and Zimmermann, P. Average case analysis of unification algorithms. In *Proceedings of STACS'91* (1991), C. Choffrut and M. Jantzen, Eds., Lecture Notes in Computer Science, pp. 196–213.

[2] Albert, L., and Fages, F. Average case analysis of the Rete pattern–matching algorithm. In *Automata, Languages and Programming* (1988), T. Lepistö and A. Salomaa, Eds., vol. 317 of *Lecture Notes in Computer Science*, Springer Verlag. Proceedings of 15th ICALP Colloquium, Tempere, Finland, July 1988.

[3] Aldous, D., and Shields, P. A diffusion limit for a class of randomly growing binary trees. *Probability Theory and Related Fields 79*, 4 (1988), 509–542.

[4] Bender, E. A. Central and local limit theorems applied to asymptotic enumeration. *Journal of Combinatorial Theory 15* (1973), 91–111.

[5] Bender, E. A. Asymptotic methods in enumeration. *SIAM Review 16*, 4 (Oct. 1974), 485–515.

[6] Bender, E. A., and Richmond, L. B. Central and local limit theorems applied to asymptotic enumeration II: Multivariate generating functions. *Journal of Combinatorial Theory,* Series A *34* (1983), 255–265.

[7] Berstel, J., and Reutenauer, C. *Les séries rationnelles et leurs langages.* Masson, Paris, 1984.

[8] Canfield, E. R. Central and local limit theorems for the coefficients of polynomials of binomial type. *Journal of Combinatorial Theory,* Series A *23* (1977), 275–290.

[9] Casas, R., Diaz, J., and Martinez, C. Statistics on random trees. In *Automata, Languages, and Programming* (1991), J. Leach Albert et al., Ed., vol. 510 of *Lecture Notes in Computer Science*, pp. 186–203. Proceedings of the 18th ICALP Conference, Madrid, July 1991.

[10] Casas, R., Fernández Camacho, M.-I., and Steyaert, J.-M. Algebraic simplification in computer algebra: An analysis of bottom-up algorithms. *Theoretical Computer Science 74*, 74 (1990), 273–298.

[11] Choppy, C., Kaplan, S., and Soria, M. Complexity analysis of term rewriting systems. *Theoretical Computer Science 67* (1989), 261–282.

[12] Compton, K. J. A logical approach to asymptotic combinatorics. I. First order properties. *Advances in Mathematics 65* (1987), 65–96.

[13] Compton, K. J. A logical approach to asymptotic combinatorics. II. Second–order properties. *Journal of Combinatorial Theory,* Series A *50* (1987), 110–131.

[14] Compton, K. J. 0–1 laws in logic and combinatorics. In *Proceedings NATO Advanced Study Institute on Algorithms and Order* (Dordrecht, 1988), I. Rival, Ed., Reidel, pp. 353–383.

[15] Comtet, L. *Advanced Combinatorics.* Reidel, Dordrecht, 1974.

[16] de Bruijn, N. G. *Asymptotic Methods in Analysis.* Dover, 1981. A reprint of the third North Holland edition, 1970 (first edition, 1958).

[17] de Bruijn, N. G., Knuth, D. E., and Rice, S. O. The average height of planted plane trees. In *Graph Theory and Computing* (1972), R. C. Read, Ed., Academic Press, pp. 15–22.

[18] Devroye, L. Branching processes in the analysis of the heights of trees. *Acta Informatica 24* (1987), 277–298.

[19] Dienes, P. *The Taylor Series.* Dover, New York, 1958. A reprint of the first Oxford University Press edition, 1931.

[20] Fagin, R., Nievergelt, J., Pippenger, N., and Strong, R. Extendible hashing: A fast access method for dynamic files. *A.C.M. Trans. Database Syst. 4* (1979), 315–344.

[21] Fayolle, G., Flajolet, P., and Hofri, M. On a functional equation arising in the analysis of a protocol for a multiaccess broadcast channel. *Advances in Applied Probability 18* (1986), 441–472.

[22] Flajolet, P. *Analyse d'algorithmes de manipulation d'arbres et de fichiers*, vol. 34–35 of *Cahiers du Bureau Universitaire de Recherche Opérationnelle.* Université Pierre et Marie Curie, Paris, 1981. 209 pages.

[23] Flajolet, P. Analytic models and ambiguity of context–free languages. *Theoretical Computer Science 49* (1987), 283–309.

[24] Flajolet, P. Mathematical methods in the analysis of algorithms and data structures. In *Trends in Theoretical Computer Science*, E. Börger, Ed. Computer Science Press, Rockville, Maryland, 1988, ch. 6, pp. 225–304. (Lecture Notes for *A Graduate Course in Computation Theory*, Udine, 1984).

[25] Flajolet, P., and Golin, M. Mellin transforms and asymptotics: The mergesort recurrence. Report, Institut National de Recherche en Informatique et en Automatique, January 1992. 11 pages.

[26] Flajolet, P., Gonnet, G., Puech, C., and Robson, J. M. The analysis of multidimensional searching in quad–trees. In *Proceedings of the Second Annual ACM–SIAM Symposium on Discrete Algorithms* (Philadelphia, 1991), SIAM Press, pp. 100–109.

[27] Flajolet, P., and Martin, G. N. Probabilistic counting algorithms for data base applications. *J. Comput. Syst. Sci. 31*, 2 (Oct. 1985), 182–209.

[28] Flajolet, P., and Odlyzko, A. M. Random mapping statistics. In *Advances in Cryptology* (1990), J.-J. Quisquater and J. Vandewalle, Eds., vol. 434 of *Lecture Notes in Computer Science*, Springer Verlag, pp. 329–354. Proceedings of Eurocrypt'89, Houtalen, Belgium, April 1989.

[29] Flajolet, P., and Odlyzko, A. M. Singularity analysis of generating functions. *SIAM Journal on Discrete Mathematics 3*, 2 (1990), 216–240.

[30] Flajolet, P., and Puech, C. Partial match retrieval of multidimensional data. *Journal of the ACM 33*, 2 (1986), 371–407.

[31] Flajolet, P., and Richmond, B. Generalized digital trees and their difference–differential equations, Apr. 1991. 15 pages. INRIA Research Report 1423. To appear in *Random Structures and Algorithms.*

[32] Flajolet, P., Salvy, B., and Zimmermann, P. Automatic average–case analysis of algorithms. *Theoretical Computer Science, Series A 79*, 1 (February 1991), 37–109.

[33] Flajolet, P., and Sedgewick, R. Digital search trees revisited. *SIAM Journal on Computing 15*, 3 (Aug. 1986), 748–767.

[34] Flajolet, P., Sipala, P., and Steyaert, J.-M. Analytic variations on the common subexpression problem. In *Automata, Languages, and Programming* (1990), M. S. Paterson, Ed., vol. 443 of *Lecture Notes in Computer Science*, pp. 220–234. Proceedings of the 17th ICALP Conference, Warwick, July 1990.

[35] Flajolet, P., and Soria, M. Gaussian limiting distributions for the number of components in combinatorial structures. *Journal of Combinatorial Theory,* Series A *53* (1990), 165–182.

[36] Flajolet, P., and Steyaert, J.-M. A complexity calculus for classes of recursive search programs over tree structures. In *Proceedings of the 22nd Annual Symposium on Foundations of Computer Science* (1981), IEEE Computer Society Press, pp. 386–393.

[37] Flajolet, P., and Steyaert, J.-M. A complexity calculus for recursive tree algorithms. *Mathematical Systems Theory 19* (1987), 301–331.

[38] Gamkrelidze, R. V., Ed. *Analysis I, Integral Representations and Asymptotic Methods*, vol. 13 of *Encyclopedia of Mathematical Sciences.* Springer Verlag, 1989.

[39] Gardy, D. *Bases de données et allocations aléatoires: Quelques analyses de performance.* Doctorate in sciences, Université de Paris–Sud, 1989.

[40] Gardy, D. Méthode de col et lois limites en analyse combinatoire. *Theoretical Computer Science 92*, 2 (1992), 261–280.

[41] Gessel, I. M. Symmetric functions and P–recursiveness. *Journal of Combinatorial Theory,* Series A *53* (1990), 257–285.

[42] Gessel, I. M., and Stanley, R. P. Algebraic enumeration. Preprint, 1989. To appear as a chapter in the *Handbook of Combinatorics*,R. Graham, M. Grötschel and L. Lovász, Eds.

[43] Goulden, I. P., and Jackson, D. M. *Combinatorial Enumeration.* John Wiley, New York, 1983.

[44] Graham, R., Knuth, D., and Patashnik, O. *Concrete Mathematics.* Addison Wesley, 1989.

[45] Greene, D. H. *Labelled formal languages and their uses.* PhD thesis, Stanford University, June 1983.

[46] Greene, D. H., and Knuth, D. E. *Mathematics for the analysis of algorithms.* Birkhauser, Boston, 1981.

[47] Harary, F., and Palmer, E. M. *Graphical Enumeration.* Academic Press, 1973.

[48] Henrici, P. *Applied and Computational Complex Analysis.* John Wiley, New York, 1977. 3 volumes.

[49] Hoshi, M., and Flajolet, P. Page usage in quadtree indexes. Report 1434, Institut National de Recherche en Informatique et en Automatique, May 1991. 19 pages. Accepted for publication in *BIT.*

[50] Joyal, A. Une théorie combinatoire des séries formelles. *Advances in Mathematics 42*, 1 (1981), 1–82.

[51] Kemp, R. *Fundamentals of the Average Case Analysis of Particular Algorithms.* Wiley-Teubner, Stuttgart, 1984.

[52] Kirschenhofer, P., and Prodinger, H. On some applications of formulæ of Ramanujan in the analysis of algorithms. *Mathematika 38* (1991), 14–33.

[53] Kirschenhofer, P., Prodinger, H., and Szpankowski, W. On the variance of the external path length in a symmetric digital trie. *Discrete Applied Mathematics 25* (1989), 129–143.

[54] Knuth, D. E. *The Art of Computer Programming*, vol. 1: Fundamental Algorithms. Addison-Wesley, 1968. Second edition, 1973.

[55] Knuth, D. E. *The Art of Computer Programming*, vol. 3: Sorting and Searching. Addison-Wesley, 1973.

[56] Knuth, D. E. The average time for carry propagation. *Indagationes Mathematicae 40* (1978), 238–242.

[57] Knuth, D. E., and Schönhage, A. The expected linearity of a simple equivalence algorithm. *Theoretical Computer Science 6* (1978), 281–315.

[58] Lifschitz, V., and Pittel, B. The number of increasing subsequences of the random permutation. *Journal of Combinatorial Theory,* Series A *31* (1981), 1–20.

[59] Lipshitz, L. The diagonal of a D-finite power series is D-finite. *J. Algebra 113* (1988), 373–378.

[60] Lipshitz, L. D-finite power series. *J. Algebra 122* (1989), 353–373.

[61] Louchard, G. The Brownian motion: a neglected tool for the complexity analysis of sorted table manipulation. *RAIRO Theoretical Informatics 17* (1983).

[62] Massazza, P. Holonomic functions and their relation to linearly constrained languages. Manuscript, 1991. 14 pages. Based on the authors's Ph. D. Thesis, University of Milan, 1991.

[63] McKay, B. D. The asymptotic numbers of regular tournaments, eulerian digraphs and eulerian oriented graphs. *Combinatorica 10*, 4 (1990), 367–377.

[64] Meir, A., and Moon, J. W. On the altitude of nodes in random trees. *Canadian Journal of Mathematics 30* (1978), 997–1015.

[65] Odlyzko, A. M. Enumeration of strings. In *Combinatorial Algorithms on Words* (1985), A. Apostolico and Z. Galil, Eds., vol. 12 of *NATO Advance Science Institute Series.* Series F: Computer and Systems Sciences, Springer Verlag, pp. 205–228.

[66] Odlyzko, A. M. Asymptotic enumeration methods. Preprint, Mar. 1992. To appear as a chapter in the *Handbook of Combinatorics*,R. Graham, M. Grötschel and L. Lovász, Ed.

[67] Odlyzko, A. M., and Richmond, L. B. Asymptotic expansions for the coefficients of analytic generating functions. *Aequationes Mathematicae 28* (1985), 50–63.

[68] Pólya, G., and Read, R. C. *Combinatorial Enumeration of Groups, Graphs and Chemical Compounds.* Springer Verlag, New York, 1987.

[69] Salvy, B. *Asymptotique automatique et fonctions génératrices.* Ph. D. thesis, École Polytechnique, 1991.

[70] Sedgewick, R. *Algorithms*, second ed. Addison–Wesley, Reading, Mass., 1988.

[71] Soria-Cousineau, M. *Méthodes d'analyse pour les constructions combinatoires et les algorithmes.* Doctorat d'état, Université de Paris–Sud, Orsay, July 1990.

[72] Stanley, R. P. Generating functions. In *Studies in Combinatorics,* M.A.A. Studies in Mathematics, Vol. 17. (1978), G.-C. Rota, Ed., The Mathematical Association of America, pp. 100–141.

[73] Stanley, R. P. Differentiably finite power series. *European Journal of Combinatorics 1* (1980), 175–188.

[74] Stanley, R. P. *Enumerative Combinatorics*, vol. I. Wadsworth & Brooks/Cole, 1986.

[75] Steyaert, J.-M. *Structure et complexité des algorithmes.* Doctorat d'état, Université Paris VII, Apr. 1984.

[76] Steyaert, J.-M., and Flajolet, P. Patterns and pattern-matching in trees: an analysis. *Information and Control 58*, 1–3 (July 1983), 19–58.

[77] Szpankowski, W. Patricia tries again revisited. *Journal of the ACM 37*, 4 (1990), 691–711.

[78] Titchmarsh, E. C. *The Theory of Functions*, second ed. Oxford University Press, 1939.

[79] Vitter, J. S., and Flajolet, P. Analysis of algorithms and data structures. In *Handbook of Theoretical Computer Science*, J. van Leeuwen, Ed., vol. A: Algorithms and Complexity. North Holland, 1990, ch. 9, pp. 431–524.

[80] Wasow, W. *Asymptotic Expansions for Ordinary Differential Equations.* Dover, 1987. A reprint of the John Wiley edition, 1965.

[81] Wilf, H. S. *Generatingfunctionology.* Academic Press, 1990.

[82] Wimp, J., and Zeilberger, D. Resurrecting the asymptotics of linear recurrences. *Journal of Mathematical Analysis and Applications 111* (1985), 162–176.

[83] Zeilberger, D. A holonomic approach to special functions identities. *Journal of Computational and Applied Mathematics 32* (1990), 321–368.

[84] Zimmermann, P. *Séries génératrices et analyse automatique d'algorithmes.* Thèse de Doctorat, École Polytechnique, Palaiseau, France, 1991.

[85] Zimmermann, P. Analysis of functions with a finite number of return values, Feb. 1992. 12 pages.