Automatic average-case analysis of algorithms

Philippe Flajolet, Paul Zimmermann, Bruno Salvy

Rapports de Recherche

N°1233

Programme 2

Calcul symbolique, Programmation et Génie logiciel

AUTOMATIC AVERAGE-CASE ANALYSIS OF ALGORITHMS

Philippe FLAJOLET
Bruno SALVY
Paul ZIMMERMANN

Août 1990

Automatic Average–case Analysis of Algorithms

Philippe Flajolet, Bruno Salvy, Paul Zimmermann

Abstract. Many probabilistic properties of elementary discrete combinatorial structures of interest for the average–case analysis of algorithms prove to be decidable. This paper presents a general framework in which such decision procedures can be developed: It is based on a combination of generating function techniques for counting, and complex analysis techniques for asymptotic estimations.

We expose here the theory of exact analysis in terms of generating functions for four different domains: the iterative/recursive and unlabelled/labelled data type domains. We then present some major components of the associated asymptotic theory and exhibit a class of naturally arising functions that can be automatically analyzed.

A fair fragment of this theory is also incorporated into a system called Lambda–Upsilon-Omega. In this way, using computer algebra, one can produce automatically non–trivial average–case analyses of algorithms operating over a variety of "decomposable" combinatorial structures.

At a fundamental level, this paper is part of a global attempt at understanding why so many elementary combinatorial problems tend to have elementary asymptotic solutions. In several cases, it proves possible to relate entire classes of elementary combinatorial problems whose structure is well–defined with classes of elementary "special" functions and classes of asymptotic forms relative to counting, probabilities, or average–case complexity.

L'analyse automatique d'algorithmes

Résumé. De nombreuses propriétés probabilistes de structures discrètes intéressant l'analyse d'algorithmes s'avèrent être décidables. Cet article présente un cadre général dans lequel il est possible de développer de telles procédures de décision: ce cadre repose d'une part sur l'utilisation de séries génératrices de dénombrement, d'autre part sur des méthodes d'analyse asymptotique complexe.

On expose ici la théorie de l'analyse exacte en termes de séries génératrices pour quatre domaines différents: les domaines itératif/récursif et étiqueté/non–étiqueté. On présente ensuite quelques composantes principales en montrant une classe de fonctions dont l'analyse asymptotique peut être rendue automatique.

Une partie de cette théorie est incorporée dans un analyseur automatique, appelé Lambda–Upsilon–Omega. De la sorte, grâce au calcul formel, il est possible de produire automatiquement des analyses automatiques non–triviales d'algorithmes qui opèrent sur une variété de structures combinatoires "décomposables".

A un niveau plus fondamental, cet article fait partie d'une tentative plus générale visant à comprendre pourquoi tant de problèmes combinatoires élémentaires tendent à avoir une solution asymptotique de forme simple. Dans plusieurs cas, il s'avère possible de relier des classes entières de problèmes combinatoires élémentaires de structure bien définie avec des classes de fonctions "spéciales" élémentaires et des classes de formes asymptotiques exprimant des propriétés de dénombrement, de probabilités ou de complexité moyenne.

This report is to appear in the journal *Theoretical Computer Science*, special issue on "Algebraic and computing treatment of noncommutative power series", guest edited by G. Jacob and C. Reutenauer.
Scheduled date of publication is February 1991.

Automatic Average–case Analysis of Algorithms

Philippe Flajolet
INRIA Rocquencourt
F-78153 Le Chesnay
(France)

Bruno Salvy
INRIA and LIX,
Ecole Polytechnique
F-91128 Palaiseau
(France)

Paul Zimmermann
INRIA Rocquencourt
F-78153 Le Chesnay
(France)

June 2, 1993

Abstract

Many probabilistic properties of elementary discrete combinatorial structures of interest for the average–case analysis of algorithms prove to be decidable. This paper presents a general framework in which such decision procedures can be developed: It is based on a combination of generating function techniques for counting, and complex analysis techniques for asymptotic estimations.

We expose here the theory of exact analysis in terms of generating functions for four different domains: the iterative/recursive and unlabelled/labelled data type domains. We then present some major components of the associated asymptotic theory and exhibit a class of naturally arising functions that can be automatically analyzed.

A fair fragment of this theory is also incorporated into a system called Lambda–Upsilon-Omega. In this way, using computer algebra, one can produce automatically non–trivial average–case analyses of algorithms operating over a variety of "decomposable" combinatorial structures.

At a fundamental level, this paper is part of a global attempt at understanding why so many elementary combinatorial problems tend to have elementary asymptotic solutions. In several cases, it proves possible to relate entire classes of elementary combinatorial problems whose structure is well–defined with classes of elementary "special" functions and classes of asymptotic forms relative to counting, probabilities, or average–case complexity.

Contents

1 Introduction

This paper presents a systematic framework in which combinatorial enumerations and probabilistic properties of combinatorial structures can be studied formally.

The analysis system that we propose is "algebraically complete" with respect to a large category of so–called decomposable data types and their associated algorithms. It is also "asymptotically complete" with respect to several subclasses of decomposable problems. Thus a number of statistical and computational properties of combinatorial structures can be systematically decided, even in their precise asymptotic form.

A correlate of this is the possibility of designing an automatic analyzer of average case performance for several interesting classes of algorithms and programmes. Based on this theory, we have actually built a prototype system called *Lambda–Upsilon–Omega* ($\Lambda\Upsilon\Omega$) that is capable of producing rather non–trivial average–case analyses of algorithms.

Here is a small sample of properties amenable to automatic analysis in this context: (i) the average number of cycles in a random permutation of n elements is $\sim \log n$ and the probability that such a permutation has no 1–cycle is $\sim e^{-1}$; (ii) path length in a random heap–ordered tree of n elements is on average $\sim 2n \log n$, which represents also the comparison cost of Quicksort; (iii) path length in a random (uniform) plane tree is $\sim \frac{1}{2} n\sqrt{\pi n}$; (iv) the symbolic differentiation algorithms of computer algebra gain on average a factor of $O(\sqrt{n})$ if shared representations (i.e., dags) are used, etc.

The paper consists of two major parts that reflect the two components of the theory. The first one, the "algebraic" component, deals with exact counting through the algebra of generating functions. The second one, the "analytic" component, uses analytic properties of these generating functions in order to recover relevant asymptotic informations.

Algebraic enumeration. For the class of decomposable combinatorial structures under consideration, it is possible to compile automatically structural specifications into equations over *counting generating functions.* These equations represent in a compact format either explicit or else recursive forms of count sequences. For the associated algorithms, we introduce generating functions of average costs called *complexity descriptors*, and we provide similar translation mechanisms from programme specifications to these complexity descriptors.

The equations that one generates in this way are meaningful in the sense that all coefficients of generating functions—providing either the number of combinatorial structures of size n or the average case complexity of algorithms over random data of size n —are computable in time that is polynomial in n.

Asymptotic analysis. It is known from classical analysis and analytic number theory that the asymptotic growth of coefficients of a series is determined by analytic properties of the series (viewed then as an analytic function of a complex argument). In this domain, singularities and saddle points play an essential rôle.

One of the major benefits of the generating function approach is to associate well identified classes of special functions to well characterized classes of combinatorial structures and programmes. We can then systematically relate classes defined by special combinatorial constructions, classes of special functions with specific analytic properties, and asymptotic properties of structures.

We first illustrate the principles of our approach by discussing two examples drawn from the classical theory of formal languages and enumerations.

EXAMPLE 1. *Regular events and finite automata.* Combinatorial structures defined by regular languages and finite automata have rational generating functions [9, 27, 71]. The counting sequences accordingly satisfy linear recurrences with constant coefficients. From elementary analysis, we know that a rational generating function $f(z)$ admits a partial fraction decomposition, so that its coefficients f_n have an explicit form as "exponential polynomials",

$$f_n = \sum_k P_k(n)\omega_k^n, \tag{1}$$

for a finite family of polynomials $P_k(x)$ and a family of algebraic numbers ω_k.

In other words, for this restricted class of devices, we are able to predict in which class of formulæ, either exact or asymptotic, counting sequences and expected values of parameters are going to fall. For instance, *a priori*, the problem of run length statistics—What is the probability that a random binary string of length n contains no run of k consecutive 1's?—lies in this class for each fixed k. (See [28, XIII.7] for a classical introduction.) Further analytic properties are available; most notably the Perron–Frobenius theory [11, 51] predicts that the ω's of largest modulus in (1) have arguments that are commensurable with π, a fact that further restricts the range of fluctuations (due to complex ω's) to those that are asymptotically periodic. The whole theory [27] is a combinatorial analogue of the classical theory of Markov chains. ■

EXAMPLE 2. *Context–Free languages.* Trees and various types of lattice paths can be described by context–free grammars [9, 10, 26, 56, 71]. The corresponding generating functions are algebraic, as follows by the classical Chomsky–Schützenberger theorem [16]. Accordingly, the counting sequences satisfy linear recurrences with polynomial coefficients ("P–recursive" sequences). An algebraic function $f(z)$ has only algebraic singularities; from this fact, its coefficients f_n are found to be asymptotic to a sum of "algebraic" elements,

$$f_n \sim \sum_j c_j n^{r_j/s_j}\omega_j^n \qquad \text{with } r_j, s_j \in \mathbb{N} \text{ and } c_j, \omega_j \in \mathbb{C}. \tag{2}$$

This again characterizes the allowed types of probabilistic behaviours for all combinatorial processes that can be described by context–free languages. This asymptotic theory of context–free