Analysis of Algorithms (AofA):
Part II: 1998 – 2000 ("Princeton–Barcelona–Gdańsk")

May 5, 2003

Michael Drmota
Department of Geometry
Vienna University of Technology
Vienna, A-1040
Austria

Wojciech Szpankowski*
Department of Computer Science
Purdue University
W. Lafayette, IN 47907
U.S.A.

Abstract

This article is a continuation of our previous *Algorithmic Column* [54] (*EATCS*, 77, 2002) dedicated to activities of the *Analysis of Algorithms* group during the "Dagstuhl–Period" (1993–1997). The first three meetings took place in Schloss Dagstuhl, Germany. The next three meetings of AofA were in Princeton (1998), Barcelona (1999), and Krynica Morska (near Gdańsk, 2000). We shall present here some research problems that have been the *highlights* of these three meetings. Three special issues [42, 31, 43] were also published after these meetings and we briefly summarize them.

1 Introduction

The area of *analysis of algorithms* was born on July 27, 1963, when D. E. Knuth wrote his "Notes on Open Addressing" about hashing tables with linear probing. Since then the area has been undergoing substantial changes; we now use various methods from different branches of mathematics: combinatorics, probability theory, graph theory, real and complex analysis, number theory and occasionally algebra, geometry, operations research, and so forth.

In 1993 the first meeting entirely devoted to the *analysis of algorithms* was organized by P. Flajolet, R. Kemp and H. Prodinger at Schloss Dagstuhl (Germany). After that there have been two further meetings in Dagstuhl (1995, 1997). Some of the research activities of that time have been described in the first *Algorithmic Column* [54].

The emergence of AofA as an organized field of research, which began with the Dagstuhl seminars and continues till nowadays, started a transformation from a collection of results on individual problems to a study of methods of general applicability, to an understanding of relationships to classical methods of analysis, combinatorics, and discrete probability, to a web of knowledge that applies in a broad context.

In this second *Algorithmics Column* on analysis of algorithms we concentrate on activities of the next tree meetings: Princeton (1998), Barcelona (1999) and Krynica Morska (near Gdańsk, 2000). Most of the material we outline here is published in three special issues: *Algorithmica*, 29, 2001, [42], *Algorithmica*, 31, 2001 [31], and *Random Structures & Algorithms*, 19, 2001, [43] (dedicated to Don Knuth on the occasion of his $(100)_8$th birthday).

*This research was supported in part by the NSF Grant CCR-0208709.

2 The Contraction Method for Recursive Algorithms

Recursive algorithms are popular tools in computer science. Quicksort is one of the most prominent one. Recursive structures are often subject of precise mathematical analysis since usually a parameter of interest can be translated into recurrences (e.g. the number of comparisons in Quicksort). Assuming that a properly normalized version of such a parameter has a limiting distribution (under a probabilistic model), the above recurrence may further translate into a fixed point equation for the distribution. The main thrust of the *contraction method* introduced by Rösler and Rüschendorf [51] is to solve such a fixed point equation using Banach's fixed point equation.

In what follows we describe the contraction method when applied to the number of comparisons L_n of Quicksort sorting n items. The recursive description of Quicksort translates to[1]

$$\mathcal{L}(L_n) = \mathcal{L}\left(L_{Z_n-1} + \overline{L}_{n-Z_n} + n - 1\right), \qquad n \geq 2, \tag{1}$$

where $L_0 = L_1 = 0$, $L_2 = 1$, Z_n is uniformly distributed on $\{1, 2, \ldots, n\}$, $\mathcal{L}(L_j) = \mathcal{L}(\overline{L}_j)$, and Z_n, L_j, $\overline{L}_j$ $(1 \leq j \leq n)$ are independent. For example, it is an easy exercise to obtain explicit representations for the expected value $\mathbf{E}L_n$. From (1) we find the recurrence

$$\mathbf{E}L_n = n - 1 + \frac{1}{n}\sum_{j=1}^{n}\left(\mathbf{E}L_{j-1} + \mathbf{E}L_{n-j}\right)$$

that can be explicitly solved yielding

$$\begin{aligned}
\mathbf{E}L_n &= 2(n+1)\sum_{k=1}^{n+1}\frac{1}{k} - 4(n+1) + 2 \\
&= 2n\log n + n(2\gamma - 4) + 2\log n + 2\gamma + 1 + O(\log n)/n)
\end{aligned}$$

with $\gamma = 0.57721...$ being Euler's constant.

Let us now consider the random variable $Y_n = (L_n - \mathbf{E}L_n)/n$ that satisfies the following equation

$$\mathcal{L}(Y_n) = \mathcal{L}\left(Y_{Z_n-1}\frac{Z_n - 1}{n} + \overline{Y}_{n-Z_n}\frac{n - Z_n}{n} + c_n(Z_n)\right), \quad n \geq 2,$$

where $Y_0 = Y_1 = 0$, Z_n is uniformly distributed on $\{1, 2, \ldots, n\}$, and $\mathcal{L}(Y_j) = \mathcal{L}(\overline{Y}_j)$, and Z_n, Y_j, $\overline{Y}_j$ $(1 \leq j \leq n)$ are independent. Furthermore,

$$c_n(j) = \frac{n-1}{n} + \frac{1}{n}\left(\mathbf{E}L_{j-1} + \mathbf{E}L_{n-j} - \mathbf{E}L_n\right).$$

Thus if Y_n has a limiting distribution Y, then it has to satisfy

$$\mathcal{L}(Y) = \mathcal{L}\left(UY + (1-U)\overline{Y} + c(U)\right), \tag{2}$$

where U is uniformly distributed on $[0, 1]$, $\mathcal{L}(\overline{Y}) = \mathcal{L}(Y)$, $U, \overline{Y}, Y$ are independent, and

$$c(x) = 2x\log x + 2(1-x)\log(1-x) + 1.$$

[1] We denote by $\mathcal{L}(X)$ the distribution function of X.

The first step is to show that (2) has actually a unique solution with $\mathbf{E}Y = 0$.

Let D denote the space of distribution functions with finite second moment and zero first moment. Then the Wasserstein metric d_2 is defined as

$$d_2(F,G) = \inf \|X - Y\|_2,$$

where $\|\cdot\|_2$ denotes the L_2-norm and the infimum is taken over all random variables X with distributions function F and all Y with distribution function G. It is well known that (D, d_2) constitutes a Polish space.[2] Let $S : D \to D$ be a map defined by

$$S(F) := \mathcal{L}(UX + (1-U)\overline{X} + c(U)),$$

where $X, \overline{X}, U$ are independent, $\mathcal{L}(\overline{X}) = \mathcal{L}(X) = F$, and U is uniformly distributed on $[0,1]$. Then one can show that S is a contraction with respect to the Wasserstein metric d_2 and, thus, there is a unique fixed point $F \in D$ with $S(F) = F$.

Indeed, let $F, G \in D$ and suppose that $\mathcal{L}(\overline{X}) = \mathcal{L}(X) = F$, $\mathcal{L}(\overline{Y}) = \mathcal{L}(Y) = G$, and U is uniformly distributed on $[0,1]$ such that $U, \overline{X}, X$ and $U, \overline{Y}, Y$ are independent. Then $S(F) = \mathcal{L}(UX + (1-U)\overline{X} + c(U))$ and $S(G) = \mathcal{L}(UY + (1-U)\overline{Y} + c(U))$ and consequently

$$\begin{aligned}
d_2^2(S(F), S(G)) &\le \|UX + (1-U)\overline{X} - UY - (1-U)\overline{Y}\|_2^2 \\
&= \|U(X-Y) + (1-U)(\overline{X} - \overline{Y})\|_2^2 \\
&= \mathbf{E}(X-Y)^2 \cdot \mathbf{E}U^2 + \mathbf{E}(\overline{X} - \overline{Y})^2 \cdot \mathbf{E}(1-U)^2 \\
&= \frac{2}{3}\mathbf{E}(X-Y)^2.
\end{aligned}$$

Taking the infimum over all possible X, Y we obtain

$$d_2(S(F), S(G)) \le \sqrt{\frac{2}{3}}\, d_2(F,G),$$

which completes the proof that S is a contraction.

The final step is to show that Y_n actually converges to Y. We refer to [49] for details, but it is sufficient to show that $d_2(\mathcal{L}(Y_n), \mathcal{L}(Y)) \to 0$. In fact, Rösler [49] showed that

$$d_2^2(\mathcal{L}(Y_n), \mathcal{L}(Y)) \le \frac{2}{n} \sum_{j=1}^{n} \left(\frac{j-1}{n}\right)^2 d_2^2(\mathcal{L}(Y_{j-1}), \mathcal{L}(Y)) + O\left(\frac{\log^2 n}{n}\right)$$

which implies $d_2(\mathcal{L}(Y_n), \mathcal{L}(Y)) \to 0$. This completes the proof that the normalized number of comparisons $(L_n - \mathbf{E}L_n)/n$ has a limiting distribution.

From the fixed point equation (2) it also possible to calculate all moments; e.g. the variance of Y is given by

$$\mathbf{Var}\, Y = 7 - \frac{2}{3}\pi^2.$$

Note that the existence of a limiting distribution (the *Quicksort distribution*) was first observed by Régnier [45] via a martingale approach, whereas the characterization of Y with

[2] A sequence F_n converges to F in D if and only if F_n converges weakly to F and if the second moments of F_n converge to the second moment of F.

a fixed point equation is due to Rösler [49]. It is now also known that there exists a density ([55]), which is a bounded C^∞ function, tail estimates are available, and orders of convergence are estimated (compare with [21, 22, 23, 32]). However, no explicit representations for the limiting distribution are known.

In passing, we should add that the contraction method has developed into to a very powerful tool in the analysis of (recursive algorithms), see [11, 15, 29, 37, 38, 36, 39, 40, 50].

Finally, we also mention that Dobrow and Fill [17] used a similar approach to analyze the path length of the so-called *recursive trees* (this unfortunate term is due to Meir and Moon). These are labelled non-plane trees whose labels increase away from the root. The number of such trees is plainly $(n-1)!$ as can be seen from the fact that their exponential generating function $Y(x)$ satisfies

$$Y(z) = \int_0^z e^{Y(t)}\, dt.$$

Symbolically one can read this as: "A tree is a root of minimal label (the $\int$) to which is attached a set (the e^Y) of similar trees." Taking inspiration of Hennequin's and Rösler's methods, Dobrow and Fill were able to show the existence of a limit distribution that has interesting features not unlike the quicksort distribution. The structure of recursive trees is also of interest as one of the early examples of a priority queue (i.e., a data structure based on unbalanced heap-like trees).

3 The Height of Binary Search Trees

A *binary search tree* is a binary tree in which each node contains a key, where the keys are drawn from some totally ordered set, say $\{1, 2, \ldots, n\}$. The first key is stored in the root. The next key is placed either in the left child of the root if its value smaller than the key stored in the root or otherwise in the right child. We repeat this procedure recursively until all n keys are inserted into the tree. Observe that Quicksort can be viewed as building a binary search tree. In fact parameter L_n discussed in the previous section is also equal to the total path length in the associated binary search tree.

There are many interesting parameters of a binary search tree built over randomly selected permutation of $\{1, 2, \ldots, n\}$. We mention here the depth of a key, the height (maximum depth), the total path length, and others. The distribution of the height H_n of such a binary search tree turns out to be an interesting (and difficult) problem. We briefly describe such an analysis, but we start with some history.

In 1986 Devroye [12] proved that the expected value $\mathbf{E}H_n$ satisfies the asymptotic relation $\mathbf{E}H_n \sim c \log n$ (as $n \to \infty$), where $c = 4.31107\ldots$ is the (largest real) solution of the equation $c \log\left(\frac{2e}{c}\right) = 1$. (Earlier Pittel [41] had shown that $H_n/\log n \to \gamma$ almost surely as $n \to \infty$, where $\gamma \le c$, compare also with Robson [46]. Later Devroye [13] provided a first bound for the error term, he proved $H_n - c\log n = O(\sqrt{\log n\, \log\log n})$ in probability.) Based on numerical data Robson conjectured that the variance $\mathbf{Var}H_n$ is bounded. In fact, he could prove (see [47]) that there is an infinite subsequence for which

$$\mathbf{E}|H_n - \mathbf{E}H_n| = O(1),$$

and that his conjecture is equivalent to the assertion that the expected value of the number of nodes at level $k = H_n$ is bounded (see [48]). The best bounds were given using two completely different methods by Devroye and Reed [16] and later by Drmota [18]. They (both) proved

$$\mathbf{E}H_n = c \log n + O(\log \log n) \tag{3}$$

and

$$\mathbf{Var}H_n = \mathbf{E}(H_n - \mathbf{E}H_n)^2 = O((\log \log n)^2).$$

Eventually, Reed [44][3] settled Robson's conjecture by showing that

$$\mathbf{Var}H_n = O(1) \qquad (n \to \infty).$$

His approach is related to that of [16], but he also showed that

$$\mathbf{E}H_n = c \log n - \frac{3c}{2(c-1)} \log \log n + O(1). \tag{4}$$

Reed's approach is purely probabilistic. An analytic proof of Robson's conjecture was given (independently) by Drmota [19].[4]

The analytic proof of Drmota pays off since some time later he was able to extend his analysis and obtain the limiting distribution for the height. In [20] he uses a sequence of functions $y_k(x)$ defined as

$$y_k(x) = \sum_{n \geq 0} \mathbf{Pr}[H_h \leq k] \cdot x^n.$$

Then $y_0(x) \equiv 1$ and

$$y_{k+1}(x) = 1 + \int_0^x y_k(t)^2 \, dt. \tag{5}$$

Obviously, $y_k(x)$ are polynomials of degree $2^k - 1$ and have a limit $y(x) = 1/(1-x)$ (for $0 \leq x < 1$). The main result of [20] states

$$\mathbf{Pr}[H_n \leq k] = \Psi(n/y_k(1)) + o(1) \quad (n \to \infty), \tag{6}$$

where the $o(1)$-error term is uniform for all $k \geq 0$ and $\Psi(y)$, $y \geq 0$, is a monotonically decreasing function with $\Psi(0) = 1$ and $\lim_{y \to \infty} \Psi(y) = 0$ that satisfies the integral equation

$$y\Psi(y/e^{1/c}) = \int_0^y \Psi(z)\Psi(y-z) \, dz. \tag{7}$$

Furthermore, there exist constants $C, \eta > 0$ such that

$$\mathbf{Pr}[|H_n - \mathbf{E}H_n| \geq y] \leq Ce^{-\eta y}, \qquad (y > 0). \tag{8}$$

Drmota's method is based on a careful analysis of (5). In particular, if one sets

$$\tilde{y}_k(x) := \int_0^\infty \Psi(ye^{-k/c})e^{-y(1-x)} \, dy, \tag{9}$$

[3] Reed has also presented his result in Barcelona, 1999.

[4] Drmota talked on this topic in Princeton, 1998, and in Krynica Morska, 2000.

then $\tilde{y}_k(0) = 1 - o(1)$ and (7) translates to

$$\tilde{y}_{k+1}(x) = \tilde{y}_{k+1}(0) + \int_0^x \tilde{y}_k(t)^2\, dt.$$

Thus, the functions $\tilde{y}_k(x)$ emulate the original functions $y_k(x)$. The idea is to approximate $y_k(x)$ by $\tilde{y}_k(x)$ Observe that

$$\begin{aligned} \tilde{y}_k(x) &= \sum_{n\geq 0} \left(\frac{1}{n!} \int_0^\infty y^n e^{-y} \Psi(ye^{-k/c})\, dy \right) x^n. \\ &= \sum_{n\geq 0} \left(\Psi\left(\frac{n}{\tilde{y}_k(1)} \right) + o(1) \right) x^n \end{aligned}$$

and then the resulting relation (6) is not unexpected any more.

4 Random LC Tries

The primary purpose of a *trie* [28, 33, 34, 52, 53]) is to store a set $\mathcal{C}$ of strings (words, sequences), say $\mathcal{C} = \{X^1, \ldots, X^n\}$. Each string is a finite or infinite sequence of symbols taken from a finite alphabet $\mathcal{A} = \{\omega_1, \ldots, \omega_V\}$ of size $V = |\mathcal{A}|$. Strings are stored in leaves of the trie. The trie over $\mathcal{C}$ is built recursively as follows: For $|\mathcal{C}| = 0$, the trie is, of course, empty. For $|\mathcal{C}| = 1$, $trie(\mathcal{C})$ is a single node. If $|\mathcal{C}| > 1$, $\mathcal{C}$ is split into V subsets $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_V$ so that a string is in $\mathcal{C}_j$ if its first symbol is ω_j. The tries $trie(\mathcal{C}_1), trie(\mathcal{C}_2), \ldots, trie(\mathcal{C}_V)$ are constructed in the same way except that at the kth step, the splitting of sets is based on the kth symbol. These subtrees are then connected from their respective roots to a single node to create $trie(\mathcal{C})$. When a new string is inserted, the search starts at the root and proceeds down the tree as directed by the input symbols.

There are many possible variations of the trie. One such variation is the *b-trie*, in which a leaf is allowed to hold as many as b strings. The b-trie is particularly useful in algorithms for extendible hashing in which the capacity of a page or other storage unit is b. A second variation of the trie, the PATRICIA trie (*P*ractical *A*lgorithm *T*o *R*etrieve *I*nformation *C*oded *I*n *A*lphanumeric) eliminates the waste of space caused by nodes having only one branch. This is done by collapsing one-way branches into a single node.

Level Compression (LC) tries were introduced by Andersson and Nilsson [5]. They are further compacted versions of tries or PATRICIA tries. The following operation is repeated recursively: at the root of the trie (or PATRICIA trie) T, find the highest complete subtree C (of height h). Let T_i $(1 \leq i \leq 2^h)$ denote the subtrees rooted at level h. Replace T by the root of T and the 2^h subtrees T_i. Repeat the above path compression procedure recursively for every T_i. The resulting trie is called LC trie (or LC PATRICIA trie). Note that the number of children of each node is a power of 2.

To analyze LC tries we assume throughout that data $X^1, \ldots, X^n$ are drawn independently and uniformly from $[0,1]$ (and the keys are just the binary expansions of X^i). The quantities of interest in a trie (or LC trie) are D_n, the depth of the n-th string, A_n, the *typical* depth defined as $A_n = \frac{1}{n}\sum_{i=1}^n D_i$, and H_n, the height of the trie. Andersson and Nilsson [5] showed that for such probabilistic model (i.e., unbiased memoryless source) the typical depth in LC tries is $A_n = \Theta(\log^* n)$, where $\log^* n$ is the log-star function, defined as the minimum positive integer i such that i-th iterate $\log_2 \log_2 \cdots \log_2 n \le 1$.

Devroye [14] substantially improved results of Andersson and Nilsson. He showed that for LC tries and LC PATRICIA tries we have (under the uniform model), as $n \to \infty$,

$$\mathbf{E}\, A_n \sim \mathbf{E}\, D_n \sim \log^* n.$$

Furthermore, he showed that

$$\frac{D_n}{\log^* n} \to 1$$

in probability,

$$\frac{H_n}{\log_2 n} \to 1$$

in probability for the height of LC tries, and

$$\frac{H_n}{\sqrt{2 \log_2 n}} \to 1$$

in probability for the height of LC PATRICIA tries.

The proof is based on the property that in a random (PATRICIA) trie the fill-up-level (the number of consecutive full levels starting at the root) is about $\log_2 n - \log_2 \log_2 n$ in probability (cf. [53]). Thus, all these levels are *compressed* into one node in the corresponding LC trie. The remaining subtrees are now of size about $\log_2 n$. Hence, the fill-up-level of these subtrees is about $\log_2 \log_2 n - \log_2 \log_2 \log_2 n$ and so on. This heuristics shows that the number of levels in the LC trie is approximately $\log^* n$.

These considerations also show that random tries in the uniform model constitute very well balanced binary trees, even if we look at the *typical* structure after the fill-up-level. The only puzzling thing is that the height of LC-tries is relatively large. For random tries the height is about $2 \log_2 n$ and for random LC tries about $\log_2 n$. For random PATRICIA tries the height is about $\log_2 n + \sqrt{2 \log_2 n}$ and for random LC PATRICIA tries about $\sqrt{2 \log_2 n}$. This means that the LC-construction for tries only *compresses* the first $\log_2 n$ levels to $\log^* n$ new levels whereas the remaining levels are not really effected by this procedure. There are relatively few nodes at these higher levels because the average depth is not affected but the height is.

5 Lopsided Trees

In this section, we briefly describe a remarkable contribution of Choi and Golin [9] on lopsided trees. *Lopsided trees* are ordered rooted r-ary trees in which the length of the edge from a parent to its i-th child is c_i (where $c_1 \le c_2 \le \cdots \le c_r$). These kinds of trees model prefix codes, where different letters may have different costs. The total cost of such prefix codes corresponds to the external path length of the corresponding lopsided tree. Especially, one is interested in Varn codes. Varn codes for n symbols are the minimal prefix codes. Equivalently, a Varn code of n words corresponds to a lopsided tree with n external nodes and minimal external path length.

The main contribution of this paper is the classification of the optimal structure and analysis of such trees T_n with n external nodes. We first describe the optimal construction: One starts by labeling the nodes of an infinite lopsided tree in order of increasing depths. Now, for any set V of nodes we denote by $LEAF(V)$ the set of nodes that are not in V but their immediate ancestor is in V. Furthermore, for $n \le |LEAF(V)|$ let $LEAF_n(V)$ be the n smallest labeled nodes in $LEAF(V)$ and set

$$T_n^m = \{1, 2, \ldots, m\} \cup LEAF_n(\{1, 2, \ldots, m\}),$$

where $\lceil (n-1)/(r-1) \rceil \le m \le n-1$. Next, let $l_0, l_1, l_2, \ldots$ denote the consecutive levels upon which nodes appear, i.e. $l_0 = 0$ and $l_i = \min\{depth(v) : depth(v) > l_{i-1}\}$, and let m_j be the number of nodes v with $depth(v) \le l_j$. Finally, set $x_m = \left(\sum_{i=1}^m c_i\right)/(m-1)$ (for $r = 2, \ldots, r$) and let $k \ge 2$ be defined by

$$x_2 \ge x_3 \ge \cdots \ge x_{k-1} \ge x_k < x_{k+1} < \cdots < x_r.$$

With help of this notation we set

$$A_j = \{v \in LEAF(V_{m_j}) : depth(v) \le l_j + x_k\},$$

$a_j = |A_j|$,

$$B_j = A_j \cup \{v \in LEAF(V_{m_j}) : l_j + x_k < depth(v) \le l_{j+1} + x_k\}$$

and $b_j = |B_j|$.

The classification of optimal lopsided trees of size n is as follows:

1. If $n = a_j$ for some $j \ge 0$ then $T_{a_j}^{m_j} = V_{m_j} \cup A_j$ is an optimal lopsided tree.

2. If $a_j < n \le b_j$ for some $j \ge 0$ then $T_n^{b_j}$ and $T_{b_j}^{m_j} = V_{m_j} \cup B_j$ are optimal lopsided trees.

3. If $b_j < n \le a_{j+1}$ for some $j \ge 0$ then $T_n^{m_j+p}$ is optimal if $n = b_j + p(k-1)$ and $T_n^{m_j+p}$ or $T_n^{m_j+p+1}$ is optimal if $n = b_j + p(k-1) + q$ for $q < k-1$.

This characterization can be also applied to formulate an algorithm to construct an optimal tree T_n in $O(n \log r)$ time which is better than previous algorithms.

After building the optimal trees, the authors of [9] analyze lopsided trees. In particular, they present asymptotic analysis of $F(x)$ (the number of nodes in $A_x = \{v : depth(v) \le x\}$), $L(x)$ (the number of leaves in $A_x = \{v : depth(v) \le x\}$), and the minimum height of a tree with n leaves and the cost $C(T_n)$ (resp. the cost of Varn codes of n words).

Let us describe the analysis of $F(x)$, that is, the number of nodes of depth no bigger than x. It is easy to see that $F(x)$ satisfies the following equation

$$F(x) = \begin{cases} 1 + F(x - c_1) + \cdots + F(x - c_r) & \text{if} \quad x \ge c_1 \\ 1 & \text{if} \quad 0 \le x \le c_1 \\ 0 & \text{if} \quad x < 0. \end{cases}$$

This functional equation can be solved either by using Laplace's transform or the Mellin transform. The authors of [9] set $x = \ln t$ and $d_i = \ln c_i$ to reduce the above equation to the one on $f(t) = F(\ln t)$ for $t > 1$ that is accessible by the Mellin transform approach. Indeed, the Mellin transform $f^*(s) = \int_1^\infty f(t) t^{s-1} dt$ becomes

$$f^*(s) = \frac{1}{s(1 - d_1^s - \cdots - d_r^s)}$$

for $\Re(s) < -1$. Using the inverse Mellin transform, one can extract the asymptotics of $F(x)$ as $x \to \infty$. In particular, it is proved in [9] that

- if $(c_1, \ldots, c_r)$ are rationally related (i.e., for all $1 \le i, j \le r$ the ratio c_i/c_j is rational), then

  $$F(x) = D(x)\varphi^x + O(\rho^x), \qquad x \to \infty$$

  where $1/\varphi$ is the smallest positive solution of $1 - z^{c_1} - \cdots - z^{c_r} = 0$ and $\rho < \varphi$, and

  $$D(x) = \frac{d}{c}(1 - \varphi^{-d})\varphi^{-d\{x/d\}}$$

  with $d = \gcd(c_1, \ldots, c_r)$, $c = \sum_{i=1}^r c_i \varphi^{-c_i}$ and $\{a\} = a - \lfloor a \rfloor$ is the fractional part of a.

- if $(c_1, \ldots, c_r)$ are irrationally related (i.e., for some $1 \le i, j \le r$ the ratio c_i/c_j is irrational), then

  $$F(x) = \frac{1}{c \ln \varphi}\varphi^x + o(\varphi^x)$$

  as $x \to \infty$.

6 Dynamical Sources and Algorithms

It is a quite natural idea to consider an algorithm together with its possible inputs as a dynamical system. The (discrete) time is related to the number of iterations. In what follows we shortly review on the *realization* of this idea by B. Vallée and her collaborators [1, 7, 8, 10, 56, 58, 57].

One considers a dynamical systems (or sources S) on a finite or denumerable alphabet $\mathcal{M}$. Let $T : (0,1) \to (0,1)$ be a mapping of the kind that there is a partition $(I_m : m \in \mathcal{M})$ of $(0,1)$ such that the restriction of $T : I_m^\circ \to (0,1)$ is a bijection (satisfying certain analytic properties). Then each $x \in (0,1)$ is associated with an infinite sequence (word)

$$M(x) = (M_1(x), M_2(x), \ldots),$$

where $M_j(x) = m \in \mathcal{M}$ if $T^{j-1}(x) \in I_m$. Furthermore there is a probability distribution on $(0,1)$ so that one can consider statistical properties of such dynamical systems.

The key element of the whole analysis is a function $\lambda(s)$ (where s in a suitable complex neighborhood of the real interval $I = [0,1]$) which is the largest eigenvalue of an appropriate bounded compact operator such that an analog of the Perron-Frobenius theory can be applied. These operators are called *classical* $\mathcal{G}_s$ (resp. *generalized) Ruelle operators*. For $s = 1$ the classical Ruelle operator $\mathcal{G}_1$ is just the density transform operator on $(0,1)$ with respect to the mapping $T : (0,1) \to (0,1)$ (see [57]).

Two parameters are of particular interest, namely the entropy $h(S)$ and the coincidence probability (related to the second order Rényi entropy) $c(S)$. They are are related to $\lambda(s)$ via $h(S) = -\lambda'(1)$ and $c(S) = \lambda(2)$. For example, one asserts that the number $B(x)$ of finite prefixes of $M(x)$ with probability $\geq x$ is asymptotically given by

$$B(x) \sim \frac{-1}{\lambda'(1)} \frac{1}{x} = \frac{1}{h(S)\, x}$$

as $x \to \infty$ (see [57]).

One can apply dynamic sources and this new methodology to the analysis of tries. In such a case, it is assumed that $M(x)$ determines the infinite strings of the data keys (see [10]). One obtains that the height H_n of these random tries satisfies

$$\mathbf{E}\, H_n \sim \frac{2}{|\log c(S)|} \log n$$

and

$$\mathbf{Pr}[H_n \leq k] = \exp\left(-\rho\, c(S)^k n^2\right) + o(1)$$

uniformly for all integers $k \geq 0$ as $n \to \infty$ (where $\rho > 0$ is a constant depending on the source and the initial density f). Furthermore, the average size of such a trie is approximately $n/h(S)$ and the average path length (the sum of all depth of leaves) is approximately $n \log n / h(S)$.

Another application of this concept is the analysis of generalized pattern matchings ("hidden patterns", see [7, 24]) where the words are generated according to a dynamical source. The authors of [7] determine the mean and the variance of the number of occurrences in this generalized pattern matching problem, and establish a property of concentration of distributions. The motivation to study this problem comes from an attempt at finding a reliable threshold for intrusion detections, from textual data processing applications, and from molecular biology.

Finally, Vallée and her collaborators applied dynamic sources to various versions of the Euclidean algorithm (e.g. the binary Euclidean algorithm [56], the Lehmer-Euclid algorithm, the α-Euclidean algorithm [8]). Again the entropy $h(S)$ governs the analysis of these algorithms. For example, one obtains that the average number of iterations P_n in the Euclidean algorithm is given by

$$P_n \sim \frac{2}{h(S)} \log n$$

and the average bit complexity C_n becomes

$$C_n \sim \frac{\rho}{h(S)} \log^2 n$$

as $n \to \infty$, where the constant ρ is related to the mean value of the digits.

7 The Random Assignment Problem

In this section, we report on the solution of a long standing conjecture concerning the average value of the random assignment problem due to David Aldous.[5] In the *linear assignment*

[5] Aldous outlined his proof in his talk in Krynica Morska, 2002, on "Zeta(2) and the random assignment problem".

problem (LAP) a matrix $\{a_{ij}\}_{i,j=1}^n$ is given and one asks for the best permutation σ such that

$$A_n = \min_\sigma \sum_{i=1}^n a_{i,\sigma(i)}.$$

In the *random* assignment problem the elements a_{ij} are uniformly distributed in $[0,1]$. The long standing open problem was to evaluate the average value $\mathbf{E}A_n$.

There is another model of the LAP problem. In this representation, a complete bipartite graph $K_{n,n}$ is given with random weights on edges that obey an exponential law of parameter 1. Let A_n be the cost of a random assignment which is the same as the cost of LAP. It has long been conjectured that

$$\mathbf{E}A_n \underset{n\to\infty}{\to} \zeta(2) = \sum_{k=1}^\infty \frac{1}{k^2}. \tag{10}$$

There is indeed a finite version of the conjecture, namely,

$$\mathbf{E}A_n = \sum_{k=1}^n \frac{1}{k^2}. \tag{11}$$

In fact, this problem has been open for some 20 years: Karp [30] proved in 1983 that $\mathbf{E}A_n < 2$; Aldous [2] (1992) proved the existence of the limit $\alpha = \lim \mathbf{E}A_n$ and Goemans and Kodialam [27] (1993) established that $\mathbf{E}A_n$ is a little over $1 + e^{-1}$. Mezard and Parisi [35] have a non rigorous argument based on ideas from statistical mechanics that $\mathbf{E}A_n \to \pi^2/6$. Aldous developed the ideas of an approach to proving the infinite n conjecture–this by viewing it as an infinite matching problem. This gives already the improved upper bound $\mathbf{E}A_n \le \zeta(2)$ and there was good hope that the infinite n conjecture will succumb. Indeed, it did. After our seminar Aldous submitted a complete proof and it was recently published in [4].

There are several interesting points in Aldous' lecture commented by Philippe Flajolet in his post-conference *Research Notes.*[6] First, the general approach of the probabilistic methods consists in designing an infinite (continuous) model in which the finite scale models are immersed; see Aldous' continuum random tree [3]. This is dual to analytic-combinatorial methods that aim at an exact modeling by generating function complemented by subsequent asymptotic analysis: "First approximate, then analyze!" versus "First analyze then approximate!" Second, Aldous spent quite some time during his talk advocating "pure thought" proofs: this is the way he envisions the probabilistic approach. This made Flajolet wonders, however, as to the amount of technology that is needed. Flajolet's impression was that everything is in the eye of the beholder, and perhaps what is "pure thought" for some is hard work for others? Conversely, perhaps, analysts should devote more time structuring proofs by taking the "pure thought" motto as an inspiration?

A last fact regarding this motivating lecture. One may consider the analogous problem of the cost of a minimal spanning tree of K_n with edge weights that are uniform (0,1). Frieze [26] showed in 1985 that the expected cost tends to $\zeta(3)$ as $n \to \infty$. Is there a finite n version of Frieze's result?

[6] They were published in August 2000 on the AofA web page `http://pauillac.inria.fr/algo/AofA/Research/index.html`.

8 Coalescing Saddle Points

We finally comment on an analytic method that has appeared in several applications, namely on coalescing saddle points and the Airy function.[7] For many years, there had been good reason to suspect that Airy functions play a role in quantifying certain transition regions of random combinatorics. The Airy function can be defined either as a solution of the differential equation $y'' - zy = 0$ or by the integral representation

$$\mathrm{Ai}(z) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} e^{i(zt+t^3/3)}dt = \frac{1}{\pi 3^{2/3}}\sum_{n=0}^{\infty}\frac{\Gamma((n+1)/3)}{n!}\sin\left(\frac{2(n+1)\pi}{3}\right)\left(3^{1/3}z\right)^n. \tag{12}$$

It is thus the prototype of integrals involving the exponential of a cubic.

Many limit distributions of analytic combinatorics are known to be attainable through perturbation of a singularity analysis or a saddle point analysis. The approximations are of an exponential quadratic form, e^{-x^2}, which usually leads to Gaussian laws. However, when there is some confluence of singularities or some "coalescence" of saddle points, approximations of a more complicated form should be sought. Precisely, coalescence of two saddle points is known in applied mathematics to lead to expressions involving the Airy function.

We first observe that some complications may arise with straightforward saddle point method. For example, imagine that the integral $I(n,\alpha)$ defined

$$I(n,\alpha) = \int f(z)e^{-nh(z,\alpha)}dz.$$

depends on the parameter α such that for $\alpha \neq \alpha_0$ there are two distinct saddle points z_+ and z_- of multiplicity one. For $\alpha = \alpha_0$ these two points coincide to a single saddle point z_0 of multiplicity two. Therefore, (under appropriate assumptions) for $\alpha \neq \alpha_0$

$$I(n,\alpha) \sim f(z_+)e^{-nh(z_+)}\left[\frac{2\pi}{nh''(z_+)}\right]^{1/2} + f(z_-)e^{-nh(z_-)}\left[\frac{2\pi}{nh''(z_-)}\right]^{1/2}.$$

For $\alpha = \alpha_0$ the asymptotic behavior of $I(n,\alpha_0)$ differs radically since $h''(z_0) = 0$. Then one arrives at

$$I(n,\alpha_0) \sim Af(z_0)e^{-nh(z_0)}\Gamma\left(\frac{4}{3}\right)\left[\frac{3!}{nh'''(z_+)}\right]^{1/3},$$

where A is a constant that depend son the contour of the integration. Thus the order of n changes discontinuously from $\frac{1}{2}$ to $\frac{1}{3}$. The interested reader is refereed to Wong [59] and [6] for more in depth discussion.

Flajolet's talk in Krynica Morska focuses on the case of *random maps*. Recall that a map is a connected planar graph given together with a rigid embedding on the plane or the Riemann sphere. Consider now the core which is the largest 2-connected component of a map (this is in essence the largest submap obtained by breaking the original map at its articulation points). Then core size admits a limiting distribution that has several surprising features: the

[7] Philippe Flajolet talked in Krynica Morska, 2000, about "Random Maps and Airy Phenomena", based on joint work with Cyril Banderier, Michele Soria, and Gilles Schaeffer [6] published in the post-conference special issue of *Random Structures & Algorithms*. His talk was followed by talks of Michele Soria and Gilles Schaeffer on related subjects.

tails are highly dissymmetric, decaying like $x^{-5/2}$ on the left and like e^{-x^3} on the right. The authors of [6] propose to call this distribution the *map-Airy distribution*: it arises precisely from a confluence of two saddle points (as seen via Lagrange inversion) or, equivalently, from a certain type of confluence of singularities (in the realm of the original generating functions) and it involves the Airy function–whence the name given to the distribution. Indeed, the map-Airy distribution is found to have density

$$\mathcal{A}(x) = 2\exp\left(-\frac{2}{3}x^3\right)\left(x\mathrm{Ai}(x^2) - \mathrm{Ai}'(x^2)\right), \tag{13}$$

and is, in disguise, a stable law of index $\frac{3}{2}$.

The next talk in Krynica Morska by Soria put these results into the more general framework of composition of singularity schemes. The final talk in this series by Schaeffer made explicit the generality of the approach. In fact a dozen or so types of maps exhibit the distribution (13)) and this has implication in the fast random generation of maps with higher connectivity indices. Finally, readers of these pages have already heard about the Airy function, e.g., in the context of linear probing hashing [54]. As a matter of fact, there is good hope to attack the evolution of the random graph $G_{n,m}$ (n vertices and m edges) and of linear probing hashed tables by means of coalescing saddle points [25].

References

[1] A. Akhavi and B. Vallée, Average bit-complexity of Euclidean algorithms, *Lecture Notes in Comput. Sci.*, 1853, Springer, Berlin, 2000, 373–387.

[2] D. Aldous, Asymptotics in the random assignment problem *Probab. Theory Related Fields* **93** (1992), 507–534.

[3] D. Aldous, Recursive Self-Similarity for Random Trees, Random Triangulations and Brownian Excursion, *Ann. Probab.* **22** (1994), 527-545.

[4] D. Aldous, The zeta(2) Limit in the Random Assignment Problem, *Random Struc. Alg.* **18** (2001), 381-418.

[5] A. Andersson and S. Nilsson, Improved behaviour of tries by adaptive branching, *Inform. Process. Lett.* **46** (1993), 295–300.

[6] C. Banderier, P. Flajolet, G. Schaeffer, and M. Soria, Random Maps, Coalescing Saddles, Singularity Analysis, and Airy Phenomena, *Random Struc. Alg.* **19** (2001), 194–246.

[7] J. Bourdon and B. Vallée, Generalized pattern matching statistics. *Mathematics and computer science, II* (Versailles, 2002), 249–265, Trends Math., Birkhäuser, Basel, 2002.

[8] J. Bourdon, B. Daireaux and B. Vallée, Dynamical analysis of α-Euclidean algorithms. *J. Algorithms* **44** (2002), 246–285.

[9] V. Choi and M. Golin, Lopsided Trees, I: Analyses, *Algorithmica* **31** (2001), 240–290.

[10] J. Clément, P. Flajolet, and B. Vallée, Dynamical sources in information theory: a general analysis of trie structures. *Algorithmica* **29** (2001), 307–369.

[11] M. Cramer and L. Rüschendorf, Convergence of two-dimensional branching recursions. *J. Comput. Appl. Math.* **130** (2001), 53–73.

[12] L. Devroye, A note on the height of binary search trees, *J. Assoc. Comput. Mach.* **33** (1986), 489–498.

[13] L. Devroye, Branching processes in the analysis of the height of trees, *Acta Inform.* **24** (1987), 277–298.

[14] L. Devroye, Analysis of random LC tries. *Random Struc. Alg.* **19** (2001), 359–375.

[15] L. Devroye and R. Neininger, Density approximation and exact simulation of random variables that are solutions of fixed-point equations. *Adv. Appl. Probab.* **34** (2002), 441–468.

[16] L. Devroye and B. Reed, On the variance of the height of random binary search trees, *SIAM J. Comput.* **24** (1995), 1157–1162.

[17] R. P. Dobrow and J. A. Fill, Total path length for random recursive trees. *Combin. Probab. Comput.* **8** (1999), 317–333.

[18] M. Drmota, An analytic approach to the height of binary search trees, *Algorithmica* **29** (2001), 89-119.

[19] M. Drmota, The variance of the height of binary search trees, *Theoret. Comput. Sci.* **270** (2002), 913–919.

[20] M. Drmota, An analytic approach to the height of binary search trees II, *J. Assoc. Comput. Mach.,* to appear 2003.

[21] J. A. Fill and S. Janson, Smoothness and decay properties of the limiting Quicksort density function. *Mathematics and Computer Science* (Versailles 2000), 53–64, Trends Math., Birkhäuser, Basel, 2000.

[22] J. A. Fill and S. Janson, Approximating the limiting Quicksort distribution. *Random Struc. Alg.* **19** (2001), 376–406.

[23] J. A. Fill and S. Janson, Quicksort asymptotics, *J. Algorithms,* **44** (2002), 4–28.

[24] P. Flajolet, Y. Guivarc'h, W. Szpankowski, and B. Vallée, Hidden Pattern Statistics, *ICALP 2001*, Crete, Greece, LNCS **2076**, 152-165, 2001

[25] P. Flajolet, B. Salvy, and G. Schaeffer, Airy Phenomena and Analytic Combinatorics of Connected Graphs,, *Electronic Journal of Combinatorics*, submitted.

[26] A. Frieze, On the value of a random minimum spanning tree problem *Discrete Appl. Math.* **10** (1985), 47–56.

[27] M. X. Goemans, and M. S. Kodialam, A lower bound on the expected cost of an optimal assignment *Math. Oper. Res.* **18** (1993), 267–274.

[28] G. H. Gonnet and R. Baeza-Yates, *Handbook of Algorithms and Data Structures: in Pascal and C,* Second edn. Addison-Wesley, Reading, MA, 1991.

[29] S. Janson, Ideals in a forest, one-way infinite binary trees and the contraction method. *Mathematics and computer science, II* (Versailles, 2002), 393–414, Trends Math., Birkhäuser, Basel, 2002.

[30] R. M. Karp, An upper bound on the expected cost of an optimal assignment. *Discrete algorithms and complexity* (Kyoto, 1986), 1–4, Perspect. Comput., 15, Academic Press, Boston, MA, 1987.

[31] R. Kemp and H. Prodinger (eds.), Mathematical Analysis of Algorithms, *Algorithmica* (special issue), **31** (3), 2001.

[32] C. Knessl and W. Szpankowski, Quicksort algorithms again revisited, *Discrete Math. Theor. Comput. Sci.* **3** (1999), 43–64.

[33] D. E. Knuth, *The Art of Computer Programming. Sorting and Searching*, Vol. 3, Second edn., Addison-Wesley, Reading, MA, 1998.

[34] H. Mahmoud, *Evolution of Random Search Trees*, John Wiley & Sons, New York, 1992.

[35] M. Mézard and G. Parisi, On the solution of the random link matching problem, *J. Phys.* **48** (1987), 1451–1459.

[36] R. Neininger and L. Rüschendorf, On the internal path length of d-dimensional quad trees. *Random Struc. Alg.* **15** (1999), 25–41.

[37] R. Neininger, Asymptotic distributions for partial match queries in K-d trees. *Random Struc. Alg.* **17** (2000), 403–427.

[38] R. Neininger, On a multivariate contraction method for random recursive structures with applications to Quicksort. *Random Struc. Alg.* **19** (2001), 498–524.

[39] R. Neininger and L. Rüschendorf, Limit laws for partial match queries in quadtrees. *Ann. Appl. Probab.* **11** (2001), 452–469.

[40] R. Neininger and L. Rüschendorf, Rates of convergence for Quicksort. *J. Algorithms* **44** (2002), 51–62.

[41] B. Pittel, On growing random binary trees, *J. Math. Anal. Appl.* **103** (1984), 461–480.

[42] H. Prodinger and W. Szpankowski (eds.), Average-Case Analysis of Algorithms (special issue), *Algorithmica*, **29** (1/2), 2001.

[43] H. Prodinger and W. Szpankowski (eds.), Analysis of Algorithms Dedicated to Don Knuth on the Occasion of his $(100)_8$th Birthday, *Random Struct. Alg.*, (special issue), **19** (3/4), 2001.

[44] B. Reed, The height of a random binary search tree, *J. Assoc. Comput. Mach.*, to appear 2003.

[45] M. Régnier, A limit distribution for Quicksort, *Informatique théorique et Applications/Theoretical Informatics and Applications* **23** (1989), 335–343.

[46] J. M. Robson, The height of binary search trees, *Austral. Comput. J.* **11** (1979), 151–153.

[47] J. M. Robson, On the concentration of the height of binary search trees. *ICALP 97* Proceedings, LNCS **1256** (1997), 441–448.

[48] J. M. Robson, Constant bounds on the moments of the height of binary search trees, *Theoret. Comput. Sci.* **276** (2002), 435–444.

[49] U. Rösler, A limit theorem for "Quicksort", *Informatique théorique et Applications/Theoretical Informatics and Applications* **25** (1991), 85–100.

[50] U. Rösler, On the analysis of stochastic divide and conquer algorithms. *Algorithmica* **29** (2001), 238–261.

[51] U. Rösler and L. Rüschendorf, The contraction method for recursive Algorithms, *Algorithmica* **29** (2001), 3–33.

[52] R. Sedgewick, *Algorithms in C: Fundamentals, Data Structures, Sorting, Searching*, Third edn., Addison-Wesley, Reading, MA, 1988.

[53] W. Szpankowski, *Average Case Analysis of Algorithms on Sequences*, John Wiley & Sons, New York, 2001.

[54] W. Szpankowski, Analysis of Algorithms, Part I, 1993–1998 ("Dagstuhl–Period"), *Bulletin of the European Association of Theoretical Computer Science*, **77** (2002), 43–62.

[55] K. H. Tan and P. Hadjicostas, Some properties of a limiting distribution of Quicksort, *Statistics Probab. Letters* **25** (1995), 87–94.

[56] B. Vallée, Dynamics of the Binary Euclidean Algorithm: Functional Analysis and Operators, *Algorithmica* **22** (1998), 660–685.

[57] B. Vallée, Dynamical sources in information theory: fundamental intervals and word prefixes, *Algorithmica* **29** (2001), 262–306.

[58] B. Vallée, Digits and continuants in Euclidean algorithms. Ergodic versus Tauberian theorems, *J. Théor. Nombres Bordeaux* **12** (2000), 531–570.

[59] R. Wong, *Asymptotic Approximations of Integrals*, Academic Press, Boston, 1989.