HAL
archives-ouvertes.fr

On the lempel-ziv parsing algorithm and its digital tree representation

Philippe Jacquet, Wojciec Szpankowski

▶ **To cite this version:**

Philippe Jacquet, Wojciec Szpankowski. On the lempel-ziv parsing algorithm and its digital tree representation. [Research Report] RR-1833, INRIA. 1993. <inria-00074838>

HAL Id: inria-00074838
https://hal.inria.fr/inria-00074838

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

On the lempel-ziv parsing algorithm and its digital tree representation

Philippe JACQUET
Wojciech SZPANKOWSKI

N° 1833
Janvier 1993

PROGRAMME 2

Calcul Symbolique,
Programmation
et Génie lociciel

ON THE LEMPEL-ZIV PARSING ALGORITHM AND ITS DIGITAL TREE REPRESENTATION

Philippe Jacquet*
INRIA
Rocquencourt
78153 Le Chesnay Cedex
France

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

Abstract

Consider the parsing algorithm due to Ziv and Lempel that partitions a sequence of length n into variable phrases (blocks) such that a new block is the shortest substring not seen in the past as a phrase. In universal data compression schemes the following parameters are of interest: number of phrases, the size of a phrase, the number of phrases of given size, etc. These parameters can be efficiently analyzed through a digital search tree representation of the algorithm. In particular, using this representation and a recent result of Kirschenhofer, Prodinger and Szpankowski, we solve the problem left open in Aldous and Shields, namely: we show that the variance of the number of phrases becomes asymptotically $(C+\delta(n))n/\log_2^3 n$, where $\delta(\cdot)$ is a fluctuating function with a small amplitude, and C is a constant that have been found in Kirschenhofer *et al.* We also present one result concerning the length of a phrase. Finally, we formulate several open problems concerning second-order properties of the Ziv-Lempel scheme which we envision can be solved by using appropriate tools from the digital trees arsenal. All of our results are formulated in a probabilistic framework.

NOTE SUR L'ALGORITHME DE COMPRESSION DE LEMPEL-ZIV ET SA REPRÉSENTATION EN ARBRE DIGITAL

Résumé

Nous considérons l'algorithme de compression de Ziv et Lempel. Cet algorithme consiste à séparer une séquence de longueurs n en blocs successifs tous différents, avec la règle que tout nouveau bloc est identique à un des blocs précédent ajouté d'un symbole. Nous nous intéressons aux paramètres suivants : nombre de blocs, tailles des blocs, nombre de blocs d'une taille donnée, etc. Pour analyser ces paramètres il est avantageux d'utiliser la représentation en arbre digital de cet algorithme. En particulier, grâce à un résultat récent de Kirschenhofer, Prodinger and Szpankowski, nous résolvons le problème laissé ouvert par Aldous et Shields, à savoir que la variance du nombre de blocs quand n croît, se comporte asymptotiquement en $(C+\delta(n))n/\log_2^3 n$, où $\delta(\cdot)$ est une fonction fluctuante de faible amplitude, et C est une constante déterminée par Kirschenhofer *et al.* Nous présentons aussi un résultat sur la longueur des blocs. Enfin nous formulons quelques problèmes ouverts au sujet des propriétés au second ordre de l'algorithme de Lempel et Ziv. Tous nos résultats sont exprimés sous un modèle probabiliste.

*This research was supported by NATO Collaborative Grant 0057/89.

†This research was primary done while the author was visiting INRIA in Rocquencourt, France. The author wishes to thank INRIA (projects ALGO, MEVAL and REFLECS) for a generous support. In addition, support was provided by NSF Grants NCR-9206315 and CCR-9201078 and INT-8912631, and from Grant AFOSR-90-0107, and in part by NATO Collaborative Grant 0057/89

1. INTRODUCTION

The heart of the Ziv-Lempel compression scheme is a method of parsing a string into blocks of different words. The precise scheme of parsing a string of length n is complicated and can be found in [23] (cf. [15]). Two important features of such a parsing are: (i) the blocks are pairwise distinct; (ii) each block that occurs in the parsing is the shortest phrase not yet seen to the left. For example, the string 110010100010001000... is parsed into (1)(10)(0)(101)(00)(01)(000)(100...). There is also another possibility of parsing, as already noticed in [23], and explored by Grassberger [9] and Szpankowski [21] that allows overlapping in the course of creating the partition. For example, for the above sequence the latter parsing gives (1)(10)(0)(101)(00)(01)(000100...). In this paper, we only consider the former parsing algorithm. These parsing algorithms play crucial rôle in universal data compression schemes. The interested reader is referred to [4], [15], and [23] for more details.

There is a useful tree representation of such a parsing scheme. Initially this tree, called also the *digital search tree* [13], consists of a single node, the root. All phrases are stored in internal nodes, excluding the root. When a new phrase is created, the search starts at the root and proceeds down the tree as directed by the input symbols. For example, for the binary alphabet, "0" in the input string means to move to the right and "1" means to proceed to the left. The search is completed when a branch is taken from an existing tree node to a new node that has not been visited before. Then, the edge and the new node are added to the tree. The phrases created in such a way are stored directly into the nodes of the tree. An example is shown in Figure 1. In passing, we note that the second parsing algorithm discussed above leads to another digital tree called the suffix tree (cf. [9], [21], [20]).

There are several parameters of Ziv-Lempel algorithm that are of significant importance for universal data compression schemes. We mention here a few: the number of phrases M_n, the number of phrases of given length, the length of the mth phrase ℓ_m, the length of the longest phrase, etc. We shall argue that these parameters are closely related to digital tree parameters such as: the number of internal nodes in the associated digital tree built from n independent strings, the number of internal nodes at given level, the depth of the mth internal node, the hight of the tree, the depth on insertion, and so forth.

Let us now concentrate on the number of phrases M_n that the algorithm returns from a single stationary and ergodic sequence of length n. Ziv and Lempel [23] proved that $\lim_{n\to\infty} M_n n^{-1} \log n = 1$ almost surely (a.s.) (cf. also [4], [22]). This kind of result is known as the first-order property. However, one would like to know the limiting distribution of M_n, which will be considered as a second-order property. For instance, such information can be used to evaluate the performance of an optimal off-line data compression algorithm (we shall propose one). To

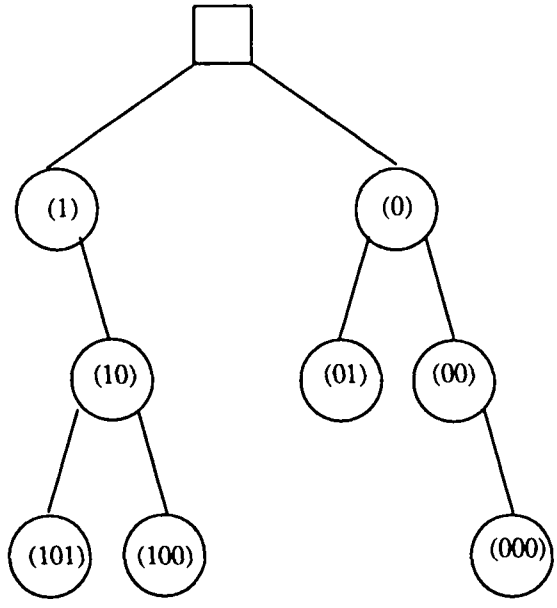

Figure 1: A digital tree representation of Ziv's parsing for the string 11001010001000100...

the best of our knowledge, the only result in this endouver is due to Aldous and Shields [1] who proved that in the *Bernoulli symmetric model* (i.e., symbols are generated independently and with equal probability) M_n weakly converges to the normal distribution with the mean $EM_n \sim n/\log_2 n$ and the variance $varM_n = O(n/\log_2^3 n)$. The coefficient at $n/\log_2^3 n$ was not obtained in [1], and in fact the authors of [1] indicated that determining it might be a complicated problem.

In this paper, we use the digital tree representation and some known results for such trees to obtain some new characteristics of the the Ziv-Lempel scheme. In our main result, we derive the missing coefficient in the variance of M_n, thus proving the problem posed by Aldous and Shields [1]. We show that $varM_n \sim (C+\delta(n))n/\log_2^3 n$ where $\delta(\cdot)$ is a fluctuating function with a small amplitude, and $C = 0.26000\ldots$ is a constant. The coefficient $C+\delta(n)$ turned out to be the same as the one appearing in the derivation of the variance of the internal path length of a digital search tree, as was recently shown by by Kirschenhofer, Prodinger and Szpankowski [12] (cf. also [11]) who also derived an explicit, although complicated, formula for C. We use the results of Aldous and Shields [1], and Kirschenhofer, Prodinger and Szpankowski [12] to prove our main result which will complete the work of [1].

Finally, we use Pittel's result [18] to establish almost sure behavior of the nth (last) phrase length ℓ_n (ℓ_{M_n}). More precisely, in the Bernoulli asymmetric model (in fact, the result is true for a more general probabilistic model) we show that $\ell_n/\log n$ does *not* converge almost surely

to any constant even if it converges in probability (pr.) to $1/entropy$.

In passing, we mention that our ultimate goal is to use the digital tree representation to study second-order properties of the Ziv-Lempel scheme for other parameters of interest. In particular, we are working on extending the Aldous-Shields result to Bernoulli asymmetric model. In our concluding remarks, we formulate several new problems in this area, and formulate some conjectures.

2. MAIN RESULTS

A *digital search tree* is a digital tree constructed from n (statistically independent) strings $X_1, \ldots, X_n$, each of which is possibly an infinite sequence of symbols over a finite alphabet. For simplicity of presentation, we assume that the alphabet is binary and we denote the two symbols as "1" and "0". We leave the root empty, and then the first string X_1 is stored in the left node or the right one depending whether the first symbol of X_1 is "1" or "0". More generally, every string is stored directly in a tree node, and the branching policy at level k is based on the kth symbol of a string to store. An example of a digital search tree is shown in Figure 1. For more details concerning digital tree the reader is referred to Knuth [13] and Mahmoud [17].

In practice (e.g., data structures [13], [17], algorithms on strings [2], [21], data compression, [1], [15] [21], etc.), several parameters of digital search trees are of interest, as discussed in the Introduction. We concentrate here on the depth of the mth node, $D_n(m)$ (the length of the path from the root to the mth node), the typical depth D_n (i.e., length of the path from the root to a *randomly* selected node), and the the internal path length L_n (the sum of all depths of nodes), the size (the number of nodes), the height (the longest path from the root to a node), and so forth (cf. [6], [7], [19]).

These parameters can be analyzed in various probabilistic frameworks. The simplest is the *Bernoulli model* in which one assumes that every string is an independent sequence of i.i.d. random variables (symbols) taken over a finite set (i.e., the alphabet). If every symbol occurs with the same probability, then such a model is called *symmetric*, otherwise it is *asymmetric*. The most general model assumes that a string is a realization of a stationary and ergodic sequence. Several first-order property results (i.e., convergence in probability and/or almost surely convergence) are known for the stationary ergodic model (cf. [18], [23]), however, the second order property (i.e., limiting distributions) are known only for the Bernoulli model (cf. [1], [16], [22]).

In the data compression scheme by Ziv and Lempel [23] a *single* string of length n is given. It is parsed into phrases as described above, and the parsing can be performed efficiently by

building the associated digital search tree. Note that such a digital tree has M_n nodes. In fact, we can alternatively study the properties of the parsing algorithm by analyzing the digital tree built over M_n *independent* strings, and then take the advantages of many already known results for digital search trees (cf. [12], [13], [17], [19], etc.).

Let us first concentrate on the number of phrases M_n. There is a basic relationship between M_n and the length of the internal path in the associated digital search tree. Note that the event $\{M_n > m\}$ is equivalent to the event that the internal path length in such a tree built from m strings is smaller equal than the length of the underlying sequence, that is, n. In other words,

$$L_{M_n-1} < n \leq L_{M_n} \tag{1}$$

or

$$\Pr\{M_n > m\} = \Pr\{L_m \leq n\} \ . \tag{2}$$

We use these relationships to characterize M_n.

We now review some results concerning the internal path length L_m of a digital search tree built from m independent strings within the *symmetric* Bernoulli model. We start with a result of Kirschenhofer, Prodinger and Szpankowski [12] concerning the variance of the internal path length L_m (cf. also [11]).

Theorem 1. (i) (Konheim and Newman [14], Knuth [13]) *The average $EL_m = \mu_m$ of the internal path length in a digital search tree in the Bernoulli symmetric model becomes*

$$\begin{aligned} EL_m &= m\log_2 m + m\left((\gamma-1)/\log 2 + 0.5 - \alpha + \delta_1(\log_2 m)\right) + \log_2 m \\ &+ 0.5(2\gamma-1)/\log 2 + 2.5 - \alpha + \delta_2(\log_2 m) + O(\log m/m) \ , \end{aligned} \tag{3}$$

where $\gamma = 0.57721\ldots$ is the Euler constant and $\alpha = \sum_{n\geq 1} 1/(2^n-1) = 1.60669\ldots$, and $\delta_1(x)$ and $\delta_2(x)$ are continous periodic functions of periof 1, mean 0 and very small amplitude ($< 10^{-6}$). More precisely,

$$\delta_1(x) = \frac{1}{\log 2}\sum_{k\neq 0}\Gamma\left(-1-\frac{2k\pi i}{\log 2}\right)e^{2k\pi i x} \ ,$$

where $\Gamma(x)$ is the gamma function.

(ii) (Kirschenhofer, Prodinger and Szpankowski [12]) *The variance $varL_m = \sigma_m^2$ of L_m is asymptotically equal to*

$$varL_m = m(C + \delta(\log_2 m)) + O(\log^2 m/m) \tag{4}$$

where

$$\begin{aligned} C &= -\frac{28}{2L} - \frac{39}{4} - 2\beta + \frac{2\alpha}{L} + \frac{\pi^2}{2L^2} + \frac{2}{L^2} - \frac{2w'(3)}{L} \\ &- \frac{2}{L}\sum_{k\geq 3} \frac{(-1)^{k+1}(k-5)}{(k+1)k(k-1)(2^k-1)} \\ &+ \frac{2}{L}\sum_{r\geq 1} b_{r+1}\left(\frac{L(1-2^{-r+1})/2-1}{1-2^{-r}} - \sum_{k\geq 2}\frac{(-1)^{k+1}}{k(k-1)(2^{r+k}-1)}\right) \end{aligned}$$

with $L = \log 2$, $\beta = \sum_{k\geq 1} k2^k/(2^k-1)^2$, *and* $b_{r+1} = (-1)^r 2^{-\binom{r+1}{2}}$. *The fluctuating function* $\delta(x)$ *is continous with period 1, mean zero and amplitude smaller than* 10^{-6}. *Finally,* $w(z)$ *satisfies*

$$\begin{aligned} \frac{w(z+1)}{Q_{z-1}} &= -2zQ_\infty + \frac{\xi(z+2)}{2^z Q_z} + \frac{\xi(z+3)}{2^{z+1}Q_{z+1}} \\ &+ \sum_{j\geq 2}\left(\frac{\xi(z+j+2)}{2^{z+j}Q_{z+j}} - \frac{\xi(j+2)}{2^j Q_j}\right) \end{aligned}$$

with $Q_z = Q_\infty/Q(2^{-z})$ *where* $Q(t) = \prod_{i\geq 1}(1-t/2^i)$, $Q_\infty = Q(1)$, *and*

$$\begin{aligned} \xi(z+1) = \sum_{r\geq 0}\frac{b_{r+1}}{Q_r} \cdot \frac{Q_\infty}{Q(2^{3-z-r})}\left(2^z - \frac{2}{1-2^{1-z-r}} - \frac{2z}{1-2^{2-z-r}}\right. \\ \left. + 2\sum_{k\geq 2}\binom{z}{k}\frac{1}{2^{r+k-1}-1}\right) . \end{aligned}$$

Numerical evaluation reveals that $C = 0.26600\ldots$ *(with five significant digits after the decimal point).* ■

The second result we review is due to Aldous and Shields [1] who – after some slight modifications - proved the following finding concerning the limiting distribution of L_m.

Theorem 2. (Aldous and Shields [1]) *In the symmetric Bernoulli model*

$$\frac{L_m - \mu_m}{\sigma_m} \to N(0,1) \qquad as \quad m \to \infty \tag{5}$$

where $N(0,1)$ *is standard normal distribution, and* μ_m *and* σ_m^2 *are given in Theorem 1.* ■

Using these two facts, we shall establish in this paper a precise characterization of the variance of the number of phrases M_n in the Ziv-Lempel parsing algorithm. Our main results can be formulated as follows.

Theorem 3. *The mean and variance of the number of phrases M_n in the Ziv-Lempel schemes become*

$$EM_n \sim \frac{n}{\log_2 n} \tag{6}$$

$$varM_n \sim (C + \delta(n))\frac{n}{\log_2^3 n} . \tag{7}$$

as $n \to \infty$, where C and $\delta(x)$ are given in Theorem 1.

Proof. The proof of Theorem 3 directly follows from Theorems 1 and 2, and the following general *renewal recurrence* that was kindly pointed to us by D. Aldous (we refer to Billingsley [3] for the proof, while here we give a short summary of this result). In terms of our notation, let us consider depths $D_n(k)$ of kth nodes where $k = 1, \ldots, n$. Note that $L_n = \sum_{k=1}^{n} D_n(k)$. By (1) and (2), we can define M_n as

$$M_n = \max\{m : \ L_m = \sum_{k=1}^{m} D_m(k) \le n\} . \tag{8}$$

The above equation is called the *renewal equation* (cf. Billingsley [3], Chap. 17). We note that $D_m(k)$ may be *dependent* random variables, and we only need that they are positive, which holds in our case. Theorem 17.3 of Billingsley [3] proves (with some trivial modifications) that if

$$\frac{L_n - \mu_n}{\sigma_n} \to N(0,1) \tag{9}$$

where μ_n and σ_n are as in Theorem 1, then the following holds too

$$\frac{M_n - n/(\mu_n/n)}{\sigma_n(\mu_n/n)^{-3/2}} = \frac{M_n - n/\log_2 n}{\sqrt{(C + \delta(\log_2 n))n/\log_2^3 n}} \to N(0,1) . \tag{10}$$

Our results (6) and (7) follow directly from the above by the standard uniform integrability argument. ■

Remark 1. Actually, we believe the following stronger results are true: $EM_n = n/\log_2 n + 1/2 + O(1/\log^2 n)$, and $varM_n = (C + \delta(n))n/\log_2^3 n + O(\sqrt{n})$. Also, it can be showed that the kth moment of M_n behaves asymptotically as $(n/\log_2 n)^k$. These results can be obtained from (2) by direct computations. In particular,

$$EM_n^{k+1} = (k+1)\sum_{m \ge 0} m^k \Pr\{M_n > m\} = (k+1)\sum_{m \ge 0} m^k \Pr\{L_m \le n\} ,$$

and appling Theorem 2 to the right-hand side of the above one obtains the above estimates. □

Finally, we present one surprising result concerning the almost sure behavior of the length of the last full phrase $\ell_{M_n - 1} = \tilde{\ell}_n$ in the Ziv-Lempel parsing scheme. We use Pittel's result [18] to prove the following finding.

Theorem 4. *Consider the asymmetric Bernoulli model with $p_{\min}$ and $p_{\max}$ denoting the smallest and the largest probability of a symbol occurrence. Then,*

$$\liminf_{n\to\infty} \frac{\widetilde{\ell}_n}{\log n} = \frac{-1}{\log p_{\min}} \qquad (a.s.) \qquad \limsup_{n\to\infty} \frac{\widetilde{\ell}_n}{\log n} = \frac{-1}{\log p_{\max}}$$

Proof. We write

$$\frac{\ell_{M_n-1}}{\log n} = \frac{\ell_{M_n-1}}{\log M_n} \cdot \frac{\log M_n}{\log n}\ , \tag{11}$$

and now we deal only with the first factor of the right-hand side. Note that ℓ_{M_n-1} is equivalent to the depth of insertion $D_n(n)$ of (M_n-1)st phrase into the associated digital search tree, that is, $\ell_{M_n-1} = D_{M_n-1}(M_n-1)$. But, Pittel proved in [18] the following

$$\liminf_{n\to\infty} \frac{D_n(n)}{\log n} = \frac{-1}{\log p_{\min}} \qquad (a.s.) \qquad \limsup_{n\to\infty} \frac{D_n(n)}{\log n} = \frac{-1}{\log p_{\max}}\ ,$$

hence this, together with Ziv-Lempel result concerning the (a.s) behavior of M_n and (11), proves the theorem. ■

Remark 2. Theorem 4 can be generalized to the so called *mixing model* (i.e., when the underlying sequence is stationary satisfying some mixing condition) since Pittel's result is true for such a model. □

3. OPEN PROBLEMS AND CONCLUDING REMARKS

Our digital search tree representation can be used in many other ways to establish second-order properties of data compression schemes, hence also Ziv-Lempel parsing algorithms. In this concluding remarks we discuss three open problems of second-order properties within the *asymmetric* Bernoulli model (in fact, extensions to *Markovian model* seem to be possible). Below, we assume binary alphabet with p and q being the probability of "1" and "0" occurrence, respectively. We formulate also some conjectures.

In sequel we discuss: (a) the limiting distribution of M_n, or equivalently the limiting distribution of the internal path length L_n in a digital search tree; (b) the average number of phrases of size l, or equivalently the limiting distribution of the depth in a digital search tree; (c) the limiting distribution of the number phrases of size l, or equivalently the limiting distribution of the number of nodes at level l in a digital search tree.

A. Limiting Distribution for the Number of Phrases

Due to our relationship (2) and Billingsley's theorem concerning the renewal equation (cf. (8)-(10)), the limiting distribution of M_n will be known if one estimates the limiting distribution of the internal path length L_n in a digital tree built from fixed, say n, independent strings.

It is easy to establish a functional equation for the generating function of L_n. Let $L_n(u) = Eu^{L_n}$. Then, as in [12] we have $L_0(u) = 1$ and for $n \geq 1$

$$L_{n+1}(u) = u^n \sum_{k=0}^{n} \binom{n}{k} p^k q^{n-k} L_k(u) L_{n-k}(u) \ . \tag{12}$$

Define now the exponential generating function $L(z,u) = e^{-z} \sum_{n=0}^{\infty} L_n(u) z^n/n!$. Then, (12) translates into

$$\frac{\partial L(z,u)}{\partial z} = L(pzu,u)L(qzu,u) \tag{13}$$

with $L(z,0) = 1$.

The above differential-functional equation must be solved asymptotically to obtain the limiting distribution of L_n. Using a method similar to the one suggested in Jacquet and Régnier [10], we conclude that the following conjecture is very plausible.

Conjecture 1. *The internal path length L_n in a digital tree is normally distributed. More precisely:*

$$\frac{L_n - EL_n}{\sqrt{varL_n}} \to N(0,1) \tag{14}$$

where $EL_n \sim (n/h)\log n$ and $varL_n = \Theta(n \log n)$ where h is the entropy of the alphabet. ■

Note that this conjecture would automatically imply (by the renewal theory argument) that $(M_n - EM_n)/\sqrt{varM_n}$ is also asymptotically normally distributed with $EM_n \sim nh/\log n$ and $varM_n = \Theta(n/\log^2 n)$. The constant hidden in $\Theta(\cdot)$ of the variance seems to be difficult to estimate.

B. The Average Number of Phrases of Given Size.

Let $M_n(l)$ denote the number of phrases of size l in the Ziv-Lempel parsing algorithm. A quick look at Figure 1 suggests that this quantity is equivalent to the number of nodes at level l (where the root is at level zero) in a digital tree built from M_n nodes. We are interested in $EM_n(l)$ for any l and large n. Conditioning on M_n (which limiting distribution should be known from the solution to Conjecture 1), we can reduce the problem to the evaluation of the average number of nodes at level l in a digital tree built from a fixed number, say n, of strings. For such a model we shall evaluate the generating function $B_n(u) = \sum_{l=0}^{\infty} E\{\#\text{number of nodes at level } l\}u^l$.

This problem is closely related to the evaluation of the limiting distribution of the typical depth D_n (i.e., the depth of a randomly selected node) in the associated digital tree. If $D_n(u) = Eu^{D_n}$ denotes the generating function of D_n, then $D_n(u) = B_n(u)/n$ (cf. [21]). The limiting distribution of D_n is only known for the symmetric Bernoulli model (cf. Louchard [16]).

The generating function $B_n(u)$ satisfies the following recurrence equation: $B_0(u) = 0$, $B_1(u) = 1$, and for $n \geq 2$

$$B_{n+1}(u) = u \sum_{j=0}^{n} \binom{n}{k} p^k q^{n-k} (B_k(u) + B_{n-k}(u)) . \tag{15}$$

Using the general solution proposed in Szpankowski [21] we can solve (15) to find that

$$B_n(u) = n - \sum_{k=2}^{n} (-1)^k \binom{n}{k} Q_k(u) \tag{16}$$

where

$$Q_k(u) = \prod_{j=2}^{k-1} (1 - up^j - uq^j) .$$

The solution (16) has the form of an alternating sum, and can be treated either by the Rice method or the Mellin-like method (cf. [6], [21]) to obtain the limiting distribution of D_n. The symmetric and asymmetric models lead to two different limiting distributions. The symmetric one was discussed in Louchard [16]. For the asymmetric model we propose the following solution.

Conjecture 2. *The depth of a node in the asymmetric Bernoulli model is asymptotically normal. More precisely:*

$$\frac{D_n - ED_n}{\sqrt{varD_n}} \to N(0,1) \tag{17}$$

where $ED_n = h^{-1} \log n + O(1)$ *and* $varD_n = (h_2 - h^2)/h^3 \log n + O(1)$ *(cf. [21]).* ■

The average number of phrases $EM_n(l)$ can be now estimated from Conjecture 1 and Conjecture 2. More precisely, we need a local version of the limiting distribution as in Conjecture 2, but this doable.

C. Limiting Distribution of the Number of Phrases of Given Size

Finally, we discuss one problem that leads to similar differential-functional equations as above, but for which we do not know at this time any solution.

Let, as before, $M_n(l)$ denote the number of phrases of size l in the Ziv-Lempel algorithm. We are now interested in the limiting distribution of $M_n(l)$. As argued above, this problem will be solved, provided we prove Conjecture 1, if we obtain the limiting distribution of the number of nodes at level l in a digital tree built from fixed, say n, strings. We denote the latter quantity as H_n^l.

Let $H_n^l(u) = Eu^{H_n^l}$ be the probability generating function of H_n^l. Note that it satisfies the following recurrence

$$H_{n+1}^l(u) = \sum_{k=0}^{n} \binom{n}{k} p^k q^{n-k} H_k^{l-1}(u) H_{n-k}^{l-1}(u) ,$$

with $H_0^0(u) = 1$.

The above recurrence translates into the following differential-functional equation for the exponential generating function $H^l(z,u) = e^{-z}\sum_{n=0}^{\infty} H_n^l(u)z^n/n!$

$$\frac{\partial H^l(z,u)}{\partial z} = H^{l-1}(pu,u)H^{l-1}(qu,u) \tag{18}$$

with $H^0(z,0) = 1$.

The differential-functional equation (18) is of similar type as the equation (13), but with one more degree of freedom, namely l. We believe that the same technique as the one used to solve (13) should work – with proper changes – in this case. At this moment of time, however, we refrain from making any conjecture.

ACKNOWLEDGEMENT

We would like to thank David Aldous (U. of California, Berkeley) for pointing to us the general renewal equation in Billingsley [3]

References

[1] D. Aldous and P. Shields, A Diffusion Limit for a Class of Random-Growing Binary Trees, *Probab. Th. Rel. Fields*, 79, 509-542 (1988).

[2] A. Apostolico, The Myriad Virtues of Suffix Trees, *Combinatorial Algorithms on Words*, pp. 85-96, Springer-Verlag, ASI F12 (1985).

[3] P. Billingsley, *Convergence of Probability Measures*, John Wiley & Sons, New York 1968.

[4] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, John Wiley&Sons, New York (1991).

[5] Feller, W., *An Introduction to Probability Theory and its Applications*, Vol. II, John Wiley & Sons, New York (1971).

[6] P. Flajolet and R. Sedgewick, Digital Search Trees Revisited, *SIAM J. Computing*, 15, 748-767 (1986).

[7] P. Flajolet and B. Richmond, Generalized Digital Trees and Their Difference-Differential Equations, *Random Structures & Algorithms*, 3, 305-320 (1992).

[8] I. Gradshteyn and I. Ryznik, *Tables, Integrals, Series, and Products*, Academic press, New York 1980.

[9] P. Grassberger, Estimating the Information Content of Symbol Sequences and Efficient Codes, *IEEE Trans. Information Theory*, 35, 669-675 (1991).

[10] P. Jacquet and M. Régnier, Normal Limiting Distribution for the Size and the External Path Length of Tries, INRIA TR-827, (1988).

[11] P. Kirschenhofer, H. Prodinger and W. Szpankowski, On the Variance of the External Path in a Symmetric Digital Trie *Discrete Applied Mathematics*, 25, 129–143 (1989).

[12] P. Kirschenhofer, H. Prodinger and W. Szpankowski, Digital Search Trees Again Revisited: The Internal Path Length Perspective, Purdue University, CSD TR-989, 1990.

[13] D. Knuth, *The Art of Computer Programming. Sorting and Searching,* Addison-Wesley (1973).

[14] A. Konheim and D.J. Newman, A Note on Growing Binary Trees, *Discrete Mathematics*, 4, 57-63 (1973).

[15] A. Lempel and J. Ziv, On the Complexity of Finite Sequences, *IEEE Information Theory* 22, 1, 75-81 (1976).

[16] G. Louchard, Exact and Asymptotic Distributions in Digital and Binary Search Trees, *RAIRO Theoretical Inform. Applications*, 21, 479-495 (1987).

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

[18] B. Pittel, Asymptotic Growth of a Class of random Trees, *Annals of Probability*, 13, 414 - 427 (1985).

[19] W. Szpankowski, A characterization of digital search trees from the successful search viewpoint, *Theoretical Computer Science*, 85, 117-134 (1991).

[20] W. Szpankowski, A generalized suffix tree and its (un)expected asymptotic behaviors, *SIAM J. Computing*, to appear.

[21] W. Szpankowski, Asymptotic Properties of data Compression and Suffix Trees, *IEEE Trans. Information Theory*, submitted.

[22] A. Wyner and J. Ziv, Some Asymptotic Properties of the Entropy of a Stationary Ergodic Data Source with Applications to Data Compression, *IEEE Trans. Information Theory*, 35, 1250-1258 (1989).

[23] J. Ziv and A. Lempel, A Universal Algorithm for Sequential Data Compression, *IEEE Trans. Information Theory*, 23, 3, 337-343 (1977).

Unité de Recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 LE CHESNAY Cedex (France)

Unité de Recherche INRIA Lorraine Technopôle de Nancy-Brabois - Campus Scientifique
615, rue du Jardin Botanique - B.P. 101 - 54602 VILLERS LES NANCY Cedex (France)
Unité de Recherche INRIA Rennes IRISA, Campus Universitaire de Beaulieu 35042 RENNES Cedex (France)
Unité de Recherche INRIA Rhône-Alpes 46, avenue Félix Viallet - 38031 GRENOBLE Cedex (France)
Unité de Recherche INRIA Sophia Antipolis 2004, route des Lucioles - B.P. 93 - 06902 SOPHIA ANTIPOLIS Cedex (France)

EDITEUR
INRIA - Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 LE CHESNAY Cedex (France)

ISSN 0249 - 6399