Sequences and Digital Trees: A Symbiosis

Wojciech Szpankowski
Purdue University, spa@cs.purdue.edu

Report Number:
88-749

https://docs.lib.purdue.edu/cstech/645

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.
SEQUENCES AND DIGITAL TREES: A SYMBIOSIS

Wojciech Szpankowski

CSD TR-749
March 1988
SEQUENCES AND DIGITAL TREES: A SYMBIOSIS

(Extended Abstract)

Wojciech Szpankowski*
Department of Computer Science
Purdue University
West Lafayette, IN 47907, USA

Abstract

This paper studies in a probabilistic framework several topics concerning the way words (strings) can overlap. A word is defined as a random sequence of (possible infinite) symbols over a $V$-ary alphabet. A key notion of alignment (or common) matrix $\{C_{ij}\}_{i,j=1}^n$ is introduced, where $C_{ij}$ is the length of the longest string that is prefix of the $i$-th and the $j$-th word. This matrix plays a crucial role in estimating some periodicities and correlations on words such as detecting squares and other repetitions, computing substring statistics, evaluating longest substring common to a set of words, estimating the total length of a code needed to transmit a set of words, and so forth. On the other hand, the alignment matrix is a "bridge" between string characteristics and some parameters of digital trees built over these strings (e.g., radix tries, suffix trees, position trees, etc.). We explore this relationship and show how such a symbiosis can be used to evaluate expected complexity of string algorithms.

1. INTRODUCTION

Periodicities, correlations and related phenomena in words (sequences, strings) are known to play a central role in many areas of computer science and telecommunications [CR, GO, LO, LZ, ZL]. In this paper, we focus on a class of problems that share the following common feature: given a set of words what is the length of the longest, the shortest and the average prefix of all of these words. We do not impose any special restrictions on the set of words, that is, the words may be generated by independent sources or they might be dependent, e.g., each word in the set is a suffix of the previous word [AA, AH, AS, MC, WE]. We note here that several efficient algorithmic constructions have been set up to date to detect and exploit the presence of correlated (repeated) subpatterns and other kinds of more or less unavoidable regularities in words [AP, AP1, ML]. For all of these problems, however, algorithms were mostly finalized in the optimization of the worst-case behavior. We reconsider some of the above problems from the

* Supported in part by NSF under grant NCR-8702115.
probabilistic viewpoint and concentrate on the average complexity of the above constructions. We do not, however, restrict our investigations to a given algorithm, and rather we try to provide a methodological tool to analyze periodicities and correlation in words [GL, LI, NI].

Correlation in words, on the other hand, are usually associated with some data structures built over these words, such as radix tries, subword-trees, suffix trees, position trees, etc. [AA, AH, KN]. In this paper we explore in depth a mutual relationship (e.g. symbiosis) between these digital trees and some characteristics in words. This symbiosis is abstracted in a key notion of alignment matrix \( C = \{C_{ij}\}_{i,j=1}^{n} \) where \( n \) is the number of words and \( C_{ij} \) measures the overlap on the first symbols between the \( i \)-th and the \( j \)-th word. We shall show that some important parameters of the digital search trees can be expressed in terms of some functions of the alignment matrix. This allows us to study string characteristic by exploring properties of the associated tree and vice versa. We shall see that in some cases it is of our advantage to concentrate on digital trees instead of strings, and in some other cases, string characteristics are easier to analyze and they can provide some useful information about the associated trees.

This paper is organized as follows. In the next section, we formulate the problem and our probabilistic model. In particular, we define the alignment matrix and show how one can obtain some information about digital trees from this matrix. In Section 3 we present our main results in the form of two propositions. The first one deals with the so called independent model, that is, strings are generated by independent sources, and therefore, radix tries are studied. The other case discusses dependent strings, that is, each string is a suffix of the previous strings. Suffix tree is associated with the latter case. This paper is based on the recent research of the author [SZ1, SZ2, SZ3, SZ4, AS, KPS1, KPS2]. Nevertheless, we give new interpretations of these results, and we establish some new results which stress the symbiosis between digital trees and string characteristics.
2. MODEL FORMULATION

Let $X_1, X_2, \ldots, X_n$ be words (strings) of (possible) unbounded lengths formed by symbols from an alphabet $A = \{\alpha_1, \alpha_2, \ldots, \alpha_V\}$ of cardinality $V$. For every pair $(i, j)$, $i \neq j$, $i, j = 1, 2, \ldots, n$ we define alignment $C_{ij}$ as the length of the longest string that is a prefix of both $X_i$ and $X_j$. Thus, $C_{ij} = k$ iff $X_i$ and $X_j$ agree exactly on their first $k$ symbols, but differ on their $(k + 1)$-st.

In many string algorithms the following four parameters are used to establish complexity of the construction: the height $H_n$, the shallowness $h_n$, the depth $D_n$ and the total length $L_n$. They are defined through the alignment matrix as follows:

$$H_n = \max_{1 \leq i < j \leq n} \{C_{ij}\} \quad (2.1a)$$
$$h_n = \min_{1 \leq i \leq n} \{\max_{1 \leq j \leq n} \{C_{ij}\}\} \quad (2.1b)$$
$$L_n = \sum_{i=1}^{n} \max_{1 \leq j \leq n} \{C_{ij}\} \quad (2.1c)$$
$$D_n = \frac{L_n}{n} \quad (2.1d)$$

The interpretation of these parameters are rather obvious. For example, if $X_2, X_3, \ldots, X_n$ are suffixes of $X_1$, then the height $H_n$ measures the length of the longest substring $Z$ of $X_1$ that starts at some position $j \leq n$ of $X_1$ and such that the occurrence of $Z$ that starts at $j$ can be fully recopied from some previous occurrence of $Z$ in $X_1$; the depth $D_n$ represents the average of the string $Z$ which can be recopied, and so forth. If $X_1, X_2, \ldots, X_n$ are (unbounded) strings generated by $n$ independent sources, then the total length $L_n$ reflects the (minimum) number of symbols necessary to distinguish all of the strings $X_1, X_2, \ldots, X_n$ (e.g., when $X_1, \ldots, X_n$ are sent through a communication channel). For illustrating examples, see below.

With each set of strings $X_1, X_2, \ldots, X_n$, we can associate a digital tree. Such a tree is a $V$-ary digital search tree with edges labeled by elements from the alphabet $A$ and leaves (external
nodes) contain the strings [KN]. The access path from the root to a leaf is a minimal prefix of the information contained in the leaf. A brute force construction of such a tree is simple. For example, for binary alphabet \( A = \{a, b\} \), symbol \( a \) means "go left", and \( b \) means "go right". This process is continued until all strings \( X_1, X_2, \ldots, X_n \) can be separated (distinguished) [KN, AH]. Such a tree is characterized by the height of the tree \( H_n^T \), the depth of a randomly chosen leaf \( D_n^T \), the shortest path from the root to a randomly selected leaf, \( h_n^T \) and by the external path length \( L_n^T \), which is defined as the sum of all depths. It turns out that the above tree parameters are simple related to the appropriate string parameters defined in (2.1a)-(2.1d), e.g., \( H_n^T = H_n + 1 \), and \( L_n^T = L_n + n \). This is illustrated below on a suffix tree.

EXAMPLE 2.1. Illustrating definitions

We assume here dependent strings, that is, for a binary alphabet \( A = \{a, b\} \) let \( X = abbabaa \ldots \) be a word. Then, we define five strings \( X_1 = X, X_2 = bbabaa \ldots, X_3 = babaa \ldots, X_4 = abaa \ldots, \) and \( X_5 = baa \ldots \), which are suffixes of \( X \). The corresponding (self)-alignment matrix \( C = \{C_{ij}\} \) is as follows:

\[
C = \begin{bmatrix}
* & 0 & 0 & 2 & 0 \\
0 & * & 1 & 0 & 1 \\
0 & 1 & * & 0 & 2 \\
2 & 0 & 0 & * & 0 \\
0 & 1 & 2 & 0 & *
\end{bmatrix}
\]

From \( C \) and the expressions (2.1), we obtain \( H_n = 2, h_n = 1, D_n = 9/5, \) and \( L_n = 9 \). The appropriate digital tree (e.g., suffix tree) is shown below.
We note that the height, the depth, the shallowness, and the external path length of the above tree are equal to $H_n + 1$, $D_n + 1$, $h_n + 1$ and $L_n + n$, respectively.

The above equivalence between strings and trees parameters can be used in the analysis, either of string algorithms or tree characterizations. Before entering the discussion, we describe the framework of probabilistic assumptions within which we plan to pursue our study.

Let $A = \{\alpha_1, \alpha_2, \ldots, \alpha_V\}$ be an alphabet of $V$ symbols, and let $S = \{X_1, X_2, \ldots, X_n\}$ be a set of $n$ (possibly infinite) strings (keys, sequences) over the alphabet $A$. To set up a stochastic model, we need to characterize the probabilistic features of the set $S$. In information theory terminology, we can look at $S$ as a set of $n$ sources of information generating (possible infinite) codes (strings, keys) from the alphabet $A$. Then, the following features of the model must be specified:

- **Characteristics of the source**, that is, how (according to what distribution) the symbols are selected from the alphabet to form a key,
• **Statistical dependency between sources**, that is, whether or not the keys are statistically dependent;

• **The number of sources**, that is, whether the number $n$ of keys is a constant or itself a random variable.

In the basic model, we assume:

(i) A key $X_k = x_1^k x_2^k \cdots$ is an infinite sequence of symbols from $A$, such that it forms an independent sequence of Bernoulli trials with $\Pr \{ x_i^k = \alpha_i \} = p_i$, $i = 1, 2, \ldots, V$ and $\sum_{i=1}^{V} p_i = 1$. If $p_1 = p_2 = \cdots = p_V = 1/V$, then the model is called *symmetric*, otherwise it is *asymmetric*.

(ii) The keys $X_1, X_2, \ldots, X_n$ are statistically independent.

(iii) The number of keys is fixed and equal to $n$.

These three assumptions formulate the so called *Bernoulli model*. A modification of the model can be obtained by replacing (iii) by a more general assumption:

(iii1) The number of keys is a random variable $N$ with a probability distribution function $p(n) = \Pr \{ N = n \}$.

If $p(n)$ is Poisson distributed, then the model (i), (ii) and (iii1) is called the *Poisson model*. In most cases, a solution for the Poisson model is easy to obtain from the corresponding solution of the Bernoulli model.

In some circumstances, the assumption (i) is too unrealistic. For example, if the alphabet $A$ consists of English letters, then there is a dependency between the occurrence of two consecutive letters. In a more elaborate random model, the assumption (i) is replaced by [KO].
(i1) There is a Markovian dependency between neighboring symbols in a key $X_k = \alpha_1^k \alpha_2^k \alpha_3^k \cdots$, that is, the probability $p_{ij} = Pr \{ \alpha_j \mid \alpha_{k+1} = \alpha_i \}$, prescribes the conditional probability of sampling symbol $\alpha_j$ following symbol $\alpha_i$.

The model (i1), (ii) and (iii) or (iii1) is called Markovian model. A more sophisticated dependency may also occur.

Note that the models discussed so far are very suitable for the analysis of digital search tries [FS, KN, KP, PI, SZ1, SZ3], since it is reasonable to assume that keys are independent (assumption (ii)). This is not the case, however, for the suffix tree, because the keys $X_2, X_3, \ldots, X_n$ are suffixes of the first key, hence strongly dependent [AS, AH]. Therefore, we modify the assumption (ii) as follows.

(ii1) The keys $X_1, X_2, \ldots, X_n$ are dependent.

To the best of our knowledge, the model with assumption (ii1) has never been discussed before.

In this paper we assume that the strings are suffixes of the basic string $X_1$, and we propose an analytical technique which can be used to evaluate some characteristics of such a dependent model.

3. MAIN RESULTS

In this section, we present our main results in the form of two propositions. The first proposition deals with our basic model, that is, under assumptions (i) – (iii). We call this independent model. The second proposition discusses the same characteristics, but assumption (iii) is replaced by a more sophisticated assumption (iii1). More precisely, in this model which we shall call dependent model, each string is a suffix of a given generic string $X$. We shall see that in the independent model it is more convenient to study string characteristics through the investigations of the associated digital tree, while in the dependent model the reverse is true (e.g., parameters of
suffix trees can be obtained in a simpler manner from the study of string characteristics. This is an example of a symbiosis between sequences and digital trees.

Our main results are summarized in the following two propositions.

Proposition 1. In the independent model (assumptions (i) - (iii)) the following holds

(i) The average $ED_n$ and the variance $\text{var} D_n$ of the depth $D_n$ asymptotically satisfy

$$ED = \frac{1}{h_1} \log n + \frac{1}{h_1} \left[ \gamma + \frac{h_2}{2h_1} \right] + F_1(n) + O(n^{-1})$$

$$\text{var} D_n = \frac{h_2 - h_1^2}{h_1^3} \log n + c + F_2(n) + O(n^{-1})$$

where $h_i = (-1)^i \sum_{k=1}^v p_k \log^i p_k$ for $i=1,2$ ($h_1$ is called the entropy of the alphabet),

$\gamma = 0.577..$ is the Euler constant, $F_1(n)$ and $F_2(n)$ are periodic functions with very small amplitude. Moreover, $D_n$ converges in probability to $ED_n$, that is,

$$D_n - ED_n(1 + o(1)) \text{ in probability}$$

In the symmetric case ($p_1 = p_2 = \cdots = p_v = 1/V$) the variance reduces to

$$(h_2 = h_1^2 = \log^2 V)$$

$$\text{var} D_n = \frac{\pi^2}{6 \log^2 V} + \frac{1}{12} + F(n) + O(n^{-1})$$

and a stronger result than (3.3) holds, namely $D_n$ is asymptotically equal to $ED_n$ with probability one, hence

$$D_n - ED_n(1 + o(1)) \text{ with probability one}$$

(ii) The average external path length is $EL_n = n ED_n$ where $ED_n$ is given in (3.1). In the symmetric case the following holds
\[ \text{var } L_n = n(A + F_3(n)) + O(\log^2 n) \]  
(3.6) 
where 
\[ A = 1 + \frac{1}{2 \log 2} - \frac{1}{\log^2 2} + \frac{2}{\log 2} (\mu + v) + \tau \]  
(3.7) 
with 
\[ \mu = \sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{k(2^k - 1)} ; \quad v = \sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{2^k - 1} \]  
(3.8) 
\[ \tau = \frac{4\pi^2}{\log^3 2} \sum_{k=1}^{\infty} \frac{k}{\sinh(2k\pi/\log 2)} \]  
(3.9) 
Moreover, in that case
\[ L_n \sim E L_n (1 + o(1)) \sim n \log n \text{ with probability one} \]  
(3.10) 

(iii) The average \( EH_n \) and the variance \( \text{var } H_n \) of the height asymptotically satisfy the following

\[ EH_n = \langle 1 + o(1) \rangle \frac{2}{R} \log n \]  
(3.11) 
\[ \text{var } H_n = o(1) \log^2 n \]  
(3.12) 

where \( R \) \( \overset{\text{def}}{=} \) \(-\log \sum p_k^2 \). Moreover
\[ H_n \sim (1 + o(1))EH_n \text{ in probability} \]  
(3.13) 

Our second result concerns dependent keys, that is, suffix trees. It is assumed that a generic (random) word \( X \) is given and \( X_1 = X, X_2, \ldots, X_n \) are \( n \) suffixes of \( X \). Then

Proposition 2. For the dependent model the following holds

(i) the \( t \)-th moment of the height \( EH_n^t \) is bounded by
\[ EH_n^t \leq \frac{2^t}{\log^t p_{\max}^{-1}} \log^t n + c \]  
(3.14) 

For symmetric model
\[ EH_n^i = (1 + o(1))2^i \log \sqrt{n} \] (3.15)

and, the following convergence in probability holds

\[ H_n \sim (1 + o(1))EH_n - 2 \log \sqrt{n} \text{ in probability} \] (3.16)

where \( p_{\max} = \max_{1 \leq i \leq V} p_i \), \( c \) is a constant.

(ii) The following inequalities hold, respectively, for the average depth and shallowness

\[ ED_n \leq \frac{1}{\log p_{\max}^{-1}} \log n + c' \] (3.17)

\[ Eh_n \leq \frac{1}{\log p_{\max}^{-1}} \log n + c'' \] (3.18)

In the symmetric case,

\[ ED_n \sim \log \sqrt{n} \] (3.19)

\[ Eh_n \sim \log \sqrt{n} \] (3.20)

and, \( D_n \sim ED_n, h_n \sim Eh_n \) in probability.

(iii) The average value of the total length \( EL_n \) is upper bounded by \( nED_n \), where \( ED_n \) is given in (3.17), (3.19).

Before we discuss implications of the above results, we first remark that in the dependent model we conjecture that \( EH_n \sim (2/\log p_{\max}^{-1}) \log n \). The constant \( 2/\log p_{\max}^{-1} \) becomes very large in strongly asymmetric cases, that is, for \( p_{\max} \) close to one. On the other hand, the constant at \( \log n \) in \( Eh_n \) is too large. Later, we indicate that this constant can be reduced to \( 1/\log p_{\max}^{-1} \), where \( p_{\min} = \min_{1 \leq i \leq V} p_i \) and this seems to be asymptotically correct.

Consequences of Propositions 1 and 2 are discussed in length in [SZ1, SZ2, SZ3, AS, KPS1, KPS2]. In short, both propositions imply that brute force algorithms and related data
structures (digital tree), are attractive alternatives (from the average complexity viewpoint) for
more complicated constructions optimized with respect to the worst case analysis. In particular
in [SZ1, SZ3], we have argued that index search files in the form of radix tries or Patricia tries are
really not need to be additionally balanced, in order to reach fast access time [FN, KPS1, KPS2].
In other words, digital search trees built over independent keys are well balanced in practice. The
same conclusion is true by Proposition 2 for suffix trees. However, the consequences in the case
of dependent strings are even more important. They are discussed in detail in [AS]. For exam-
ple, we have shown there that square detection by brute force takes \( O(n \log n) \) as much as the
best known, however, more complicated algorithm [AP, CR, ML]. More consequences of our
main result will be discussed in the final version of the paper.

Finally, we present a sketch of the proofs. To prove Propositions 1 (i) – (ii) we consider the
associated digital tree and study properties of such a tree. The advantage of such an approach
comes from the fact that we can reduce the analysis of the depth \( D_n \), length \( L_n \), etc., to the inves-
tigation of some (general) recurrences. Let \( x_n \) denote a property of the digital trees with \( n \) exter-
nal nodes, that is, \( x_n \) is either \( ED_n \) or \( \text{var } D_n \) or \( EL_n \) or \( \text{var } L_n \), etc. Let also \( a_n \) denote the
amount of the property \( x_n \) contained in the root. Then, within the independent model and assum-
ing for simplicity binary case with \( p_1 = p, p_2 = q = 1 - p \), we can consider two subtrees of the
root, each with \( x_k \) and \( x_{n-k} \) amount of the property \( x_n \), where \( k \) is the actual number of strings
(keys) which are in the left subtree. Noting that, in fact, \( k \) is a random variable with the Bernoulli
distribution (see assumption (i)), we need to study the following general recurrence

\[
x_0 = x_1 = 0
\]

\[
x_n = a_n + \sum_{k=0}^{n} \binom{n}{k} p^k q^{n-k}(x_k + x_{n-k}) \quad n \geq 2
\]

where \( a_n \) is called additive term of (3.22). This recurrence models most of our quantities of
interests (the so called additive properties [SZ1]), except the height of the tree. To solve (3.22),
we define a sequence \( d_n \) (binomial inverse relations [KN]) as

\[
\begin{align*}
\hat{d}_n &= \sum_{k=0}^{n} (-1)^k \left( \binom{n}{k} a_k \right) \\
a_n &= \sum_{k=0}^{n} (-1)^k \left( \binom{n}{k} \hat{d}_k \right)
\end{align*}
\] (3.23)

Note that the exponential generating functions of \( \hat{d}_n \) and \( a_n \) are related by \( \hat{A}(-z) = A(z)e^{-z} \).

Using this, in [SZ1] it is proved that

**Lemma 1:** (i) The recurrence (3.22) possesses the following solution

\[
x_n = \sum_{k=2}^{n} (-1)^k \left( \binom{n}{k} \right) \frac{\hat{d}_k + n a_1 - a_0}{1 - p^k - q^k}
\] (3.24)

(ii) The inverse relation \( \check{x}_n \) of \( x_n \) satisfies

\[
\check{x}_n = \frac{\hat{d}_n + n a_1 - a_0}{1 - p^n - q^n}, \quad n \geq 2
\] (3.25)

Finally, to find asymptotic approximations for \( x_n \), we apply a general approach proposed either in [FS] (Rice’s method) or in [SZ4] (Mellin like approach, see also Knuth [KN]). Namely, we consider an alternating sum of the form \( \sum_{k=2}^{n} (-1)^k \left( \binom{n}{k} \right) f(k) \) where \( f(k) \) is any sequence.

This sum appears in our Lemma 1. Then

**Lemma 2:** (i) [Rice’s method, see [FS], [KP]]. Let \( C \) be a curve surrounding the points 2, 3, \ldots, \( n \), and \( f(z) \) be an analytical continuation of \( f(k) \) inside \( C \). Then

\[
S_n \overset{\text{def}}{=} \sum_{k=2}^{n} \left( \binom{n}{k} \right) (-1)^k f(k) = \frac{1}{2\pi i} \int_{C} [n; z] f(z) dz
\] (3.25)

with

\[
[n; z] = \frac{(-1)^{n-1} n!}{z(z-1) \cdots (z-n)}
\]

(ii) [Mellin like approach; see [SZ4]]. Under certain conditions on the growth of \( f(z) \) at infinity
(compare [SZ4])

\[
S_n = \frac{1}{2\pi i} -3\frac{1}{2} + i\infty \int_{-3\frac{1}{2} - i\infty}^{3\frac{1}{2} + i\infty} \Gamma(z)f(-z)n^z dz = \frac{1}{2\pi i} -3\frac{1}{2} + i\infty \int_{-3\frac{1}{2} - i\infty}^{3\frac{1}{2} + i\infty} B(n + 1,z)f(-z)dz,
\]

where \(n^z = \Gamma(n + 1)/\Gamma(n + 1 + z)\), and

\[
B(n + 1,z) = \frac{n!}{z(z + 1) \cdots (z + n)},
\]

and \(\Gamma(z)\) is the gamma function [HE]. Equivalently, (3.26a) can be simplified to

\[
S_n = \frac{1}{2\pi i} -3\frac{1}{2} + i\infty \int_{-3\frac{1}{2} - i\infty}^{3\frac{1}{2} + i\infty} \Gamma(z)f(-z)n^{-z} dz + e_n
\]

where

\[
e_n = O(n^{-1}) \int_{-3\frac{1}{2} - i\infty}^{3\frac{1}{2} + i\infty} z\Gamma(z)f(-z)n^{-z} dz
\]

that is, \(e_n = o(n)\).

**Proof:** Both formulas are a consequence of Cauchy's Theorem [HE]. The proof of (3.25) is given in [FS], while (3.26a) is established in [SZ4]. Formula (3.26b) follows directly from (3.26a) and \(n^z = n^{-z}[1 + z O(n^{-1})]\), [HE].

\[
\prod_{k=1}^{n} \left(\frac{n}{k}\right) f(k) = \sum_{k=-\infty}^{n} \text{res} \{\Gamma(z)f(z); k\} + O(n^{-M}) = \sum_{k=-\infty}^{n} \text{res} \{\Gamma(z)f(-z)n^{-z}\} + e_n + O(n^{-M})
\]

for any \(M > 0\) and the sums are taken over all poles, \(z_k, k = 0, \pm 1, \ldots\), of the functions under the integrals (3.25) and (3.26) in the appropriate regions respectively. By (3.27) the asymptotics
of the alternative sum of type (3.24) (Lemma 1) is reduced to compute the residues of the functions under the integrals, which is usually an easy task.

Using Lemma 1 and 2 we can prove Proposition 1(i) – (ii) formulas (3.1), (3.2) and (3.6). For example, it is not difficult to notice that the average length of the external path length $EL_n$ satisfies our general recurrence with $a_n = n$. To compute the second moment of the depth $D_n$ and the external path length $L_n$, we need to consider again our general recurrence (3.22), however, this time the additive term is much more complicated. For details the reader is refer to [SZ1], [KPS1], [KPS2]. To obtain convergence in probability (3.3), we use Chebyshev’s inequality and (3.1) – (3.2). Indeed, by Chebyshev’s inequality

$$\Pr\left\{ \left| \frac{D_n}{ED_n} - 1 \right| \geq \varepsilon \right\} \leq \frac{\text{var} D_n}{\varepsilon^2 (ED_n)^2} = O\left( \frac{1}{\log n} \right) \rightarrow 0$$

hence (3.3) holds. To prove almost surely convergence (3.5) and (3.10), we use our result (3.6) and Borel-Cantelli Lemma (see [KPS2]).

The height $H^T_n$ of the digital tree in the independent model does not, however, satisfy our recurrence (3.22). Moreover, no parameters of suffix trees can be described by this recurrence. Therefore, another approach is necessary. It turns out that in these cases, it is better to study string characteristics defined in (2.1) (string-based approach) than to relate them to appropriate digital trees parameters. Then, however, we need to estimate maximum over a class of dependent variables $C_{ij}, i,j = 1, 2, \ldots, n$. Fortunately, using the results of [LR1, LR2], the following can be proved

Lemma 3: Let $Y_1, Y_2, \ldots, Y_m$ be a sequence of random variables with distribution function $F_1(y), F_2(y), \ldots, F_m(y)$, respectively. Let $R_i(y) = \Pr \{ Y_i \geq y \}$ be the complement function of the distribution function $F_i(y)$ (function $R$ is sometimes called the reliability function). Finally, let $\overline{M}_m = \max_{1 \leq i \leq m} Y_i$ and $\underline{M}_m = \min_{1 \leq i \leq m} Y_i$. Then:
(i) if \( a_m \) is a solution of

\[
\sum_{k=1}^{m} R_k(a_m) = 1,
\]

then

\[
EM_m \leq a_m + \sum_{k=1}^{m} \sum_{j=0}^{b_n} R_k(j). 
\]

(ii) If \( b_m \) is a solution of

\[
\sum_{k=1}^{m} F_k(b_m) = 1,
\]

then

\[
EM_m \geq b_m - \sum_{k=1}^{m} \sum_{j=0}^{b_n} F_k(j). 
\]

(iii) If \( Y_1, Y_2, \ldots, Y_m \) are identically distributed with distribution (reliability) function \( F(y) \) (\( R(y) \)), and, moreover,

\[
\lim_{y \to \infty} \frac{1 - F(cy)}{1 - F(y)} = 0 \text{ for } c > 1, 
\]

then

\[
\lim_{m \to \infty} \frac{EM_m}{a_m} = \lim_{m \to \infty} \frac{EM_m}{b_m} = 1. 
\]

That is, \( M_m \sim a_m \) and \( M_m \sim b_m \) where \( a_m \) and \( b_m \) solve

\[
mR(a_m) = 1 \quad \text{and} \quad mF(b_m) = 1, 
\]

respectively.

**Proof:** The proof generalizes the idea of [LR1] and can be found in [SZ2, AS].

To apply Lemma 3 we must determine the distribution function of the alignments \( C_{ij} \). Let us consider first the independent model. Then, one easily shows that \( C_{ij} \) are identically distri-
Pr \{C_{ij} = k \} = P^{k-1}(1 - P) \text{ for all } i, j = 1, 2, \ldots, n \quad (3.35)

where \( P = \sum_{k=1}^{V} p_k^2 \). Then, Proposition 1(iii) follows from Lemma 3(iii), noting that

Pr \{C_{ij} > k \} = P^{k+1} \text{ and } m = n^2 \text{ in that case. Formula (3.12) is a consequence of}

\( \bar{M}_m^k = \max\{Y_1^k, \ldots, Y_n^k\} \).

In the dependent model, the distribution function of \( C_{ij} \) is much more difficult to find. However, we note first that the distribution of \( C_{ij} \) does not directly depend on \( i \) and \( j \), but only on the difference \( d = j - i, j > i \). Let \( C_d \) denote the alignment \( C_{ij} \) for \( d = j - i \). Then, the following result is proved in [AS].

**Theorem.** Let \( d \) be any finite integer smaller than \( n \), and let \( l \) and \( r \) be the unique integers defined by \( k = dl + r \), where \( l = 0, 1, \ldots, \) and \( r < d \). Then,

\[
Pr \{C_d = k\} = Pr \{C_d = ld + r\} = \left( \sum_{i=1}^{V} p_i^{l+2} \right)^r \left( \sum_{i=1}^{V} p_i^{l+1}(1 - p_i) \right)^{d-r-l} \quad (3.36a)
\]

where \( k = 0, 1, \ldots \). For the symmetric distribution \( p_1 = p_2 = \cdots = p_V = 1/V \), expression (3.36a) simplifies to

\[
Pr \{C_d = k\} = \left( \frac{1}{V} \right)^k \left( 1 - \frac{1}{V} \right) \quad (3.36b)
\]

Using the above theorem and Lemma 3, we can prove Proposition 2. Details can be found in [AS].

As a final comment, we note that the symbiosis between digital trees and sequences can be used for our advantage to study complexity problems on strings and digital search trees.
ACKNOWLEDGEMENT

I thank Alberto Apostolico for showing me the beauty of stringology.

REFERENCES


[SZ3] W. Szpankowski, Patricia tries again revisited, Purdue University, CSD TR-625, (1986).

