Trends in Mathematics, © 2002 Birkhäuser Verlag Basel/Switzerland

A Note on Random Suffix Search Trees

Luc Devroye and Ralph Neininger

ABSTRACT: *A random suffix search tree is a binary search tree constructed for the suffixes $X_i = 0.B_iB_{i+1}B_{i+2}\ldots$ of a sequence $B_1, B_2, \ldots$ of independent identically distributed random b-ary digits B_j. Let D_n denote the depth of the node for X_n in this tree when B_1 is uniform on $\mathbb{Z}_b$. We show that for any value of $b > 1$, $\mathbb{E}\, D_n = 2\log n + O(\log^2 \log n)$, just as for the random binary search tree. We also show that $D_n / \mathbb{E}\, D_n \to 1$ in probability.*

1 Introduction

Current research in data structures and algorithms is focused on the efficient processing of large bodies of text (encyclopedia, search engines) and strings of data (DNA strings, encrypted bit strings). For storing the data such that string searching is facilitated, various data structures have been proposed. The most popular among these are the suffix tries and suffix trees (Weiner, 1973; McCreight, 1976), and suffix arrays (Manber and Myers, 1990). Related intermediate structures such as the suffix cactus (Karkkainen, 1995) have been proposed as well. Apostolico (1985), Crochemore and Rytter (1994) and Stephen (1994) cover most aspects of these data structures, including their applications and efficient construction algorithms (Ukkonen 1995, Weiner 1973, Giegerich and Kurtz, 1997, and Kosaraju, 1994). If the data are thought of as strings $B_1, B_2, \ldots$ of symbols taking values in an alphabet $\mathbb{Z}_b = \{0, 1, \ldots, b-1\}$ for fixed finite b, then the suffix trie is an ordinary b-ary trie for the strings $X_i = (B_i, B_{i+1}, \ldots)$, $1 \le i \le n$. The suffix tree is a compacted suffix trie. The suffix array is an array of lexicographically ordered strings X_i on which binary search can be performed. Additional information on suffix trees is given in Farach (1997), Farach and Muthukrishnan (1996, 1997), Giancarlo (1993, 1995), Giegerich and Kurtz (1995), Gusfield (1997), Sahinalp and Vishkin (1994), Szpankowski (1993). The suffix search tree we are studying in this paper is the search tree obtained for $X_1, \ldots, X_n$, where again lexicographical ordering is used. Care must be taken to store with each node the position in the text, so that the storage comprises nothing but pointers to the text. Suffix search trees permit dynamic operations, including the deletion, insertion, and alteration of parts of the string. Suffix arrays on the other hand are clearly only suited for off-line applications.

The analysis of random tries has a long history (see Szpankowski, 2001, for references). Random suffix tries were studied by Jacquet, Rais and Szpankowski (1995) and Devroye, Szpankowski and Rais (1992). The main model used in these studies is the independent model: the B_i's are independent and identically distributed. Markovian dependence has also been considered. If $p_j = \mathbb{P}\{B_1 = j\}$, $0 \le j < b$, then it is known that the expected depth of a typical node in an n-node suffix trie is close in probability to $(1/\mathcal{E})\log n$, where $\mathcal{E} = \sum_j p_j \log(1/p_j)$ is the entropy of B_1. The height is in probability close to $(b/\xi)\log n$, where $\xi = \log(1/\sum_j p_j^b)$. If ξ or $\mathcal{E}$ are small, then the performance of these structures deteriorates to the point that perhaps more classical structures such as the binary search tree are

B. Chauvin et al. (eds.), *Mathematics and Computer Science II*
© Springer Basel AG 2002

preferable.

In this paper, we prove that for first order asymptotics, random suffix search trees behave roughly as random binary search trees. If D_n is the depth of X_n, then

$$\mathbb{E}\, D_n = 2\log n + O(\log^2 \log n)$$

and $D_n/\log n \to 2$ in probability, just as for the random binary search tree constructed as if the X_i's were independent identically distributed strings (Knuth, 1973, and Mahmoud, 1992, have references and accounts). We prove this for $b = 2$ and $p_0 = p_1 = 1/2$. The generalization to $b > 2$ is straightforward as long as B_1 is uniform on $\mathbb{Z}_b$.

The second application area of our analysis is related directly to random binary search trees. We may consider the X_i's as real numbers on $[0,1]$ by considering the b-ary expansions

$$X_i = 0.B_iB_{i+1}\ldots, \quad 1 \le i \le n\ .$$

In that case, we note that $X_{i+1} = \{bX_i\} := (bX_i) \bmod 1$. If we start with X_1 uniform on $[0,1]$, then every X_i is uniform on $[0,1]$, but there is some dependence in the sequence $X_1, X_2, \ldots$. The sequence generated by applying the map $X_{i+1} = \{bX_i\}$ resembles the way in which linear congruential sequences are generated on a computer, as an approximation of random number sequences. In fact, all major numerical packages in use today use linear congruential sequences of the form $x_{n+1} = (bx_n + a) \bmod M$, where a, b, x_n, x_{n+1}, M are integers. The sequence x_n/M is then used as an approximation of a truly random sequence. Thus, our study reveals what happens when we replace i.i.d. random variables with the multiplicative sequence. It is reassuring to note that the first order behavior of binary search trees is identical to that for the independent sequence.

The study of the behavior of random binary search trees for dependent sequences in general is quite interesting. For the sequence $X_n = (nU) \bmod 1$, with U uniform on $[0,1]$, a detailed study by Devroye and Goudjil (1998) shows that the height of the tree is in probability $\Theta(\log n \log\log n)$. The behavior of less dependent sequences $X_n = (n^\alpha U) \bmod 1$, $\alpha > 1$, is largely unknown. The present paper shows of course that $X_n = (2^n U) \bmod 1$ is sufficiently independent to ensure behavior as for an i.i.d. sequence. Antos and Devroye (2000) looked at the sequence $X_n = \sum_{i=1}^n Y_i$, where the Y_i's are i.i.d. random variables and showed that the height is in probability $\Theta(\sqrt{n})$. Cartesian trees (Devroye 1994) provide yet another model of dependence with heights of the order $\Theta(\sqrt{n})$.

This extended abstract is organized as follows. In section 2 we introduce a perturbed version of the random suffix search tree on which we will draw back troughout our analysis. Section 3 provides a rough bound for the mean of the height of the random suffix search tree, which will be used later in the analysis of $\mathbb{E}\, D_n$. In the following two sections we present a key lemma on which our expansion of the mean and a weak law of large numbers for D_n is based, and give a detailed proof for $\mathbb{E}\, D_n = 2\log n + O(\log^2 \log n)$. From section 6 on we approach the tree from a different path, the spacings formed by $X_1, \ldots, X_n$ on $[0,1]$. First we show a limit law for the scaled length of a randomly chosen spacing, convergence of all moments and a related limit law when the spacings are chosen with probability according to their length. These results could also be used to find the dominant term in the expansion of $\mathbb{E}\, D_n$. We will derive asymptotic information on the size of the subtree rooted at X_j for a large range of j. In the last section we state some lemmas which were used in the analysis. Complete proofs can be found in Devroye and Neininger (2002).

2 Notation and perturbed tree

Denote the uniform distribution on $[0,1]$ by $U[0,1]$ and the Bernoulli(p) distribution by $Be[p]$. We have given a $U[0,1]$ distributed random variable X_1 and define $X_k := T(X_{k-1})$ for $k \geq 2$, with the map $T : [0,1] \to [0,1], x \mapsto \{2x\} = 2x \mod 1$. In the binary representation $X_1 = 0.B_1B_2\ldots$, the B_k are independent $Be[1/2]$ bits. Then we have

$$X_k = 0.B_kB_{k+1}B_{k+2}\cdots$$

for all $k \geq 1$. For $m \geq 1$ we introduce the corresponding perturbed random variates

$$Y_k^{\langle m \rangle} := 0.B_kB_{k+1}\ldots B_{k+m-1}B_1^{(k)}B_2^{(k)}\ldots, \quad k = 1, \ldots, n,$$

where $\{B_j^{(k)} : k, j \geq 1\}$ is a family of independent $Be[1/2]$ distributed bits, independent of X_1. Then we have for all $k \geq 1$,

$$|X_k - Y_k^{\langle m \rangle}| \leq \frac{1}{2^m}.$$

and $Y_i^{\langle m \rangle}, Y_j^{\langle m \rangle}$ are independent if $|i - j| \geq m$.

Since we will switch in our analysis between the random suffix search tree built from $X_1, \ldots, X_n$ and its perturbed counterpart generated by $Y_1^{\langle m \rangle}, \ldots, Y_n^{\langle m \rangle}$ we have to control the probability that they coincide. We denote by $\|\lfloor x \rfloor\| := 2\lfloor x/2 \rfloor$ the largest even integer not exceeding x. For a vector $(a_1, \ldots, a_n)$ of distinct real numbers, let $\pi(a_1, \ldots, a_n)$ be the permutation given by the vector.

Lemma 2.1. *If $m := 18\|\lfloor \log_2 n \rfloor\|$, then for all $n \geq 16$,*

$$\mathbb{P}\left(\pi(X_1, \ldots X_n) \neq \pi(Y_1^{\langle m \rangle}, \ldots, Y_n^{\langle m \rangle})\right) \leq \frac{8}{n^2}.$$

The perturbed tree and the original tree are thus identical with high probability. In the perturbed tree, note that $Y_i^{\langle m \rangle}$ and $Y_j^{\langle m \rangle}$ are independent whenever $|i-j| \geq m$. Unfortunately, it is not true that random binary search trees constructed on the basis of identically distributed m-dependent sequences behave as those for i.i.d. sequences, even when m is as small as 1. For example, the depth of a typical node and the height may increase by a factor of m when m is small and positive.

3 A rough bound for the height

We will need a rough upper bound for the mean of the height of the random suffix search tree.

Lemma 3.1. *Let a binary search tree $\mathcal{T}$ be built up from distinct numbers $x_1, \dots, x_n$ and denote its height by H. We assume that the set of indices $\{1, \dots, n\}$ is decomposed into k nonempty subsets $\mathcal{I}_1, \dots, \mathcal{I}_k$ of cardinalities $|\mathcal{I}_j| = n_j$. Assume that $\mathcal{I}_j$ consists of the indices $n(j,1) < \cdots < n(j,n_j)$ and denote the height of the binary search tree $\mathcal{T}_j$ built up from $x_{n(j,1)}, \dots, x_{n(j,n_j)}$ by H_j for $j = 1, \dots, k$. Then we have*

$$H \leq k - 1 + \sum_{j=1}^{k} H_j.$$

This can be turned into a rough estimate for the height using the fact the mean of the height is known to be of the order $\log n$ for the binary search tree in the random permutation model, where each permutation of the keys inserted is equally likely (Devroye 1987). Lemma 3.2 below is valid for our model, but also for any random binary search tree constructed on the basis of $U[0,1]$ random variables that are m-dependent, with $m = O(\log n)$.

Lemma 3.2. *Let H_n denote the height of the random suffix search tree with n nodes. Then $\mathbb{E}\, H_n = O(\log^2 n)$.*

4 A key lemma

We introduce the events $A_j = \{X_j \text{ is ancestor of } X_n \text{ in the tree}\}$. Then we have the representations

$$D_n = \sum_{j=1}^{n-1} \mathbf{1}_{A_j}, \quad \mathbb{E}\, D_n = \sum_{j=1}^{n-1} \mathbb{P}(A_j).$$

We use the notation $\alpha, \beta \rhd \gamma_1, \dots, \gamma_n$, if there does not exist k with $1 \leq k \leq n$ for which $\alpha < \gamma_k < \beta$ or $\beta < \gamma_k < \alpha$, i.e., α, β are neighbors in $\{\gamma_1, \dots, \gamma_n\}$. Note that $A_j = \{X_j, X_n \rhd X_1, \dots, X_{j-1}\}$. We use $A_j^{\langle m \rangle}$ for the corresponding event involving the $Y_k^{\langle m \rangle}$: $A_j^{\langle m \rangle} = \{Y_j^{\langle m \rangle}, Y_n^{\langle m \rangle} \rhd Y_1^{\langle m \rangle}, \dots, Y_{j-1}^{\langle m \rangle}\}$. Thoughout we abbreviate $m = 18 \| \lfloor \log_2 n \rfloor \|$.

Our key lemma consists of an analysis of the depth of the n-th inserted node X_n conditioned on its location. For $x \in [0,1]$ and $1 \leq i \leq n-1$, define

$$p_i(x) := \mathbb{P}\Big(Y_i^{\langle m \rangle}, x \rhd Y_1^{\langle m \rangle}, \dots, Y_{i-1}^{\langle m \rangle} \Big).$$

We use the following *bad set*:

$$B_n(\xi) := \bigcup_{k=1}^{m} \{x \in [0,1] : |x - T^k(x)| < \xi\}, \quad \xi > 0,$$

where T is the map $T(x) := \{2x\}$ and T^k its k-th iteration, see Figure 1.

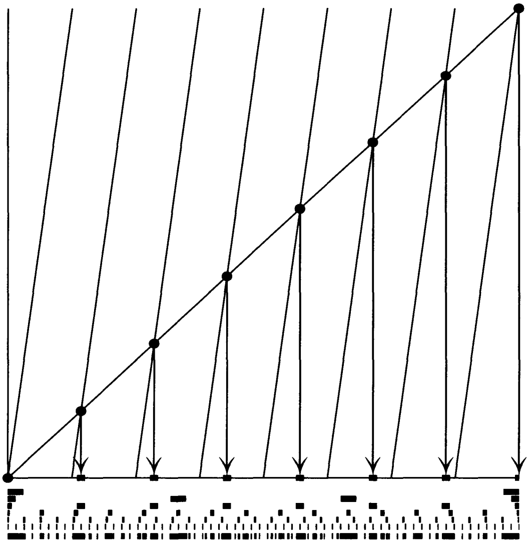

Figure 1: *The last line shows the bad set $B_n(\xi)$ for $m = 6$ and $\xi = 3/50$. The six lines above show the sets $\{|x - T^k(x)| \le \xi\}$ for $k = 1, \ldots, 6$. In the square, for the case $k = 3$, it is shown how these sets emerge.*

Lemma 4.1. *For all n sufficiently large, all $x \in [0,1]$, and $1 \le i < n$, we have*

$$\begin{aligned} p_i(x) &= \mathbf{1}_{[m^2/i, 1-m^2/i]}(x) \left(\frac{2}{i} + R_1(n,i) + \mathbf{1}_{B_n(2m^2/\sqrt{i})}(x) R_2(n,i) \right) \\ &\quad + \left(1 - \mathbf{1}_{[m^2/i,1-m^2/i]}(x)\right) R_3(n,i), \end{aligned}$$

where for appropriate constants $C_1, C_2, C_3 > 0$,

$$\begin{aligned} |R_1(n,i)| &\le C_1 \frac{\log^6 n}{i^{3/2}}, \\ |R_2(n,i)| &\le C_2 \frac{\log^3 n}{i}, \\ |R_3(n,i)| &\le C_3 \frac{\log n}{i}. \end{aligned}$$

5 Analysis of the depth

Based on Lemma 4.1, we obtain an expansion for the mean of the depth D_n as well as a weak law of large numbers. For a random binary search tree based on i.i.d. random variables, it is well-known that $\mathbb{E}\, D_n = 2 \log n + O(1)$, where D_n is the depth of the n-th node (see, e.g., Knuth 1973 or the references in Mahmoud 1992).

Theorem 5.1. *The depth D_n of the n-th node inserted into a random suffix search tree satisfies*

$$\mathbb{E}\, D_n = 2\log n + O(\log^2 \log n).$$

Proof: We define the events $A_j = \{X_j \text{ is ancestor of } X_n \text{ in the tree}\}$ and the representations

$$D_n = \sum_{j=1}^{n-1} \mathbf{1}_{A_j}, \quad \mathbb{E}\, D_n = \sum_{j=1}^{n-1} \mathbb{P}(A_j).$$

For the estimate of $\mathbb{P}(A_j)$ we distinguish three ranges for the index j, namely $1 \le j \le \lceil \log_2^{12} n\rceil$, $\lceil \log_2^{12} n\rceil < j \le n-m$, and $n-m < j < n$, where we choose $m = 18\|\lfloor \log_2 n\rfloor\|$.

The range $1 \le j \le \lceil \log_2^{12} n\rceil$: Note that $\sum_{j=1}^{\lceil \log_2^{12} n\rceil} \mathbf{1}_{A_j}$ is bounded from above by the height of the random suffix search tree with $\lceil \log_2^{12} n\rceil$ nodes. Thus, by Lemma 3.2, we obtain

$$\sum_{j=1}^{\lceil \log_2^{12} n\rceil} \mathbb{P}(A_j) \le \mathbb{E}\, H_{\lceil \log_2^{12} n\rceil} = O(\log_2^2 \log_2^{12} n)) = O(\log^2 \log n).$$

The range $\lceil \log_2^{12} n\rceil < j \le n-m$: We start, using Lemma 2.1, with the representation

$$\begin{aligned}\mathbb{P}(A_j) &= \mathbb{P}(X_j, X_n \rhd X_1, \dots, X_{j-1})\\ &= \mathbb{P}(Y_j^{\langle m\rangle}, Y_n^{\langle m\rangle} \rhd Y_1^{\langle m\rangle}, \dots, Y_{j-1}^{\langle m\rangle}) + O(1/n^2)\\ &= \mathbb{P}(A_j^{\langle m\rangle}) + O(1/n^2).\end{aligned}$$

Note that $Y_n^{\langle m\rangle}$ is independent of $Y_1^{\langle m\rangle}, \dots, Y_j^{\langle m\rangle}$, since $j \le n-m$. Thus for the calculation of $\mathbb{P}(A_j^{\langle m\rangle})$ we may condition on $Y_n^{\langle m\rangle}$. With the notation of Lemma 4.1 and using the fact that $Y_n^{\langle m\rangle}$ is $U[0,1]$ distributed this yields for all $1 \le j \le n-m$,

$$\mathbb{P}(A_j^{\langle m\rangle}) = \mathbb{E}\,[p_j(Y_n^{\langle m\rangle})] = \frac{2}{j} + R_{n,j}, \quad |R_{n,j}| \le C\frac{\log^6 n}{j^{3/2}},$$

for some constant $C > 0$. When summing note that

$$\sum_{j=\lceil \log_2^{12} n\rceil}^{\infty} \frac{\log^6 n}{j^{3/2}} \le \log^6 n \int_{\lceil \log_2^{12} n\rceil - 1}^{\infty} \frac{1}{x^{3/2}}\, dx = O(1).$$

We obtain

$$\begin{aligned}\sum_{j=\lceil \log_2^{12} n\rceil}^{n-m} \mathbb{P}(A_j) &= \sum_{j=\lceil \log_2^{12} n\rceil}^{n-m} \left(\frac{2}{j} + R_{n,j} + O\Big(\frac{1}{n^2}\Big)\right)\\ &= 2\log n + O(\log\log n).\end{aligned}$$

Hence, this range gives the main contribution.
The range $n-m<j<n-1$: With $q:=\lfloor j/m \rfloor -1$ we have

$$\begin{aligned}\mathbb{P}(A_j) &= \mathbb{P}(X_j, X_n \rhd X_1, \ldots, X_{j-1}) \\ &\leq \mathbb{P}(X_j, X_n \rhd X_{j-m}, \ldots, X_{j-qm}) \\ &= \mathbb{P}(Y_j^{\langle m\rangle}, Y_n^{\langle m\rangle} \rhd Y_{j-m}^{\langle m\rangle}, \ldots, Y_{j-qm}^{\langle m\rangle}) + O(1/n^2).\end{aligned}$$

We have, using Lemma 9.2, for n sufficiently large,

$$\begin{aligned}&\mathbb{P}(Y_j^{\langle m\rangle}, Y_n^{\langle m\rangle} \rhd Y_{j-m}^{\langle m\rangle}, \ldots, Y_{j-qm}^{\langle m\rangle}) \\ &\leq \mathbb{P}\Big(\{|Y_j^{\langle m\rangle} - Y_n^{\langle m\rangle}| \geq m^2/j\} \cap \{Y_j^{\langle m\rangle}, Y_n^{\langle m\rangle} \rhd Y_{j-m}^{\langle m\rangle}, \ldots, Y_{j-qm}^{\langle m\rangle}\}\Big) \\ &\quad + \mathbb{P}(|Y_j^{\langle m\rangle} - Y_n^{\langle m\rangle}| < m^2/j) \\ &\leq \left(1-\frac{m^2}{j}\right)^{j/m-2} + 8\frac{m^2}{j} \\ &\leq 4\exp(-m) + 8\frac{m^2}{j} \\ &\leq O\left(\frac{1}{n^{18}}\right) + 8\frac{m^2}{j}.\end{aligned}$$

The summation yields

$$\sum_{j=n-m}^{n-1} \mathbb{P}(A_j) = O(1),$$

so that the third range makes an asymptotically negligible contribution. Collecting the estimates of the three ranges, we obtain the assertion. ■

Theorem 5.2. *We have $D_n/\mathbb{E}\,D_n \to 1$ in probability as $n \to \infty$.*

6 Weak convergence of a random spacing

The lengths of the spacings formed by $X_1, \ldots, X_n$ on $[0,1]$ are denoted by $S_j^n := X_{(j+1)} - X_{(j)}$ for $j = 1, \ldots, n-1$ and $S_0^n := X_{(1)}$, $S_n^n := 1 - X_{(n)}$, where $X_{(j)}$ denotes the j-th order statistic of $X_1, \ldots, X_n$. In this section we provide a limit law for the rescaled length of a spacing chosen uniformly from $S_0^n, \ldots, S_n^n$, where by uniform we mean that we choose one of the indices $j = 0, \ldots, n$ uniformly at random. Later we will choose an index by into which spacing an $U[0,1]$ random variable, independent of X_1, falls.

Lemma 6.1. *We have*

$$nS_{I_n}^n \xrightarrow{\mathcal{L}} E, \quad (n \to \infty),$$

where E is exp(1)-distributed, i.e., has Lebesgue-density e^{-x} on $[0,\infty)$ and I_n is uniformly distributed on $\{0,\ldots,n\}$ and independent of X_1.

This can be reduced to the following result on the spacings between fractional parts of lacunary sequences due to Rudnick and Zaharescu (2002). A *lacunary sequence* is a sequence $(a_j)_{j\geq 1}$ of integers such that we have $\liminf_{j\to\infty} a_{j+1}/a_j > 1$. The primary example is $a_j = 2^j$. Now, for an $\alpha \in \mathbb{R}$ we define $S_j^n(\alpha)$ for $j = 0, \dots, n$ as the spacings between the fractional parts of αa_j, $j = 1, \dots, n$, in the unit interval $[0,1]$. More precisely, for $\vartheta_j^n := \{\alpha a_j\}$ we define $S_j^n(\alpha) := \vartheta_{(j+1)} - \vartheta_{(j)}$ for $j = 1, \dots, n-1$ as well as $S_0^n(\alpha) := \vartheta_{(1)}$ and $S_n^n(\alpha) := 1 - \vartheta_{(n)}$. Then Rudnick and Zaharescu (2002) prove:

Theorem 6.2. *Let (a_j) be a lacunary sequence. Then we have for almost all $\alpha \in \mathbb{R}$ and all $0 \leq a < b$,*

$$\lim_{n\to\infty} \frac{1}{n+1} \#\{0 \leq j \leq n : nS_j^n(\alpha) \in [a,b]\} = \int_a^b e^{-x} dx.$$

For background, see also Kurlberg and Rudnick (1999, Appendix A). This can directly be turned into a proof of Lemma 6.1.

7 Uniform integrability

In this section we show that the convergence in Corollary 6.1 holds for all moments.

Lemma 7.1. *For all fixed $p > 0$*

$$\sup_{n\in\mathbb{N}} \mathbb{E}\,(nS_{I_n}^n)^p < \infty,$$

where the random index I_n is unif$\{0, \dots, n\}$ distributed and independent of X_1.

The limit law of Theorem 6.1 together with the uniform integrability of Lemma 7.1 implies convergence of all moments (Billingsley 1979, Theorem 25.12). Thus we have

$$\lim_{n\to\infty} \mathbb{E}\,(nS_{I_n}^n)^\ell = \int_0^\infty x^\ell e^{-x}\, dx = \ell!, \quad \ell = 0, 1, 2, \dots. \qquad (1)$$

We turn to the analysis of the rescaled length of a spacing chosen according to into which spacing an indepedendent $U[0,1]$ random variable falls. For this we define the conditional distribution of the index J_n chosen by

$$\mathbb{P}(J_n = k \mid S_0^n, \dots, S_n^n) = S_k^n, \quad k = 0, \dots, n.$$

Then we have the following limit law:

Lemma 7.2. *We have*

$$nS_{J_n}^n \xrightarrow{\mathcal{L}} G_2, \quad (n \to \infty),$$

where G_2 is Gamma(2)-distributed, i.e., has Lebesgue density xe^{-x} on $[0,\infty)$.

8 Applications of spacings

The analysis of the random spacings generated by $X_1, \ldots, X_n$ can be used for the asymptotic analysis of parameters of the random suffix search tree. The leading order term of $\mathbb{E}\, D_n$ can be rediscovered using (1) with $\ell = 2$. This provides an alternative path to that followed in Theorem 5.1. The limit law for the size $N_{n,j}$ of the subtree rooted at X_j can be found for a large range of values j. This result is rooted in the lemmas of section 7.

Theorem 8.1. *The size $N_{n,j}$ of the subtree of the random suffix search tree of size n rooted at X_j satisfies for $j = j(n)$ with $j = o(n/\log^2 n)$ and $j/\log^5 n \to \infty$,*

$$\mathbb{E}\, N_{n,j} \sim \frac{2n}{j}, \quad \frac{j}{n} N_{n,j} \xrightarrow{\mathcal{L}} G_2,$$

as $n \to \infty$, where G_2 denotes the Gamma(2)-distribution.

It can be shown that in the case $j \sim \alpha n$ with $\alpha \in (0,1)$ the size $N_{n,j}$ tends in distribution to the negative binomial distibution with parameters $(2, \alpha)$, given by its generating function $s \mapsto (\alpha/(1-(1-\alpha)s))^2$.

9 Appendix

Lemma 9.1. *Let I be an interval in $[0,1]$ of length $|I|$. Then for all $1 \le i \le -\log_2 |I|$ we have*

$$\mathbb{P}(X_1, X_{1+i} \in I) \le \frac{|I|}{2^i}.$$

Lemma 9.2. *For all integer $1 \le i < j$, $t \ge 1$ and real $\varepsilon > 0$,*

$$\mathbb{P}(|X_i - X_j| \le \varepsilon) \le 2\varepsilon, \quad \mathbb{P}(|Y_i^{\langle t \rangle} - Y_j^{\langle t \rangle}| \le \varepsilon) \le 8\varepsilon.$$

Lemma 9.3. *For all integer $1 \le i < j$, $t \ge 1$ and real $\varepsilon > 0$, and U being $U[0,1]$ distributed and independent of $X_1, Y_i^{\langle t \rangle}, Y_j^{\langle t \rangle}$ we have*

$$\mathbb{P}(X_i, X_j \in [U, U+\varepsilon]) \le 2\varepsilon^2, \quad \mathbb{P}(Y_i^{\langle t \rangle}, Y_j^{\langle t \rangle} \in [U, U+\varepsilon]) \le 8\varepsilon^2.$$

Lemma 9.4. *For any Borel set $A \subseteq [0,1]$, real $\varepsilon, \delta > 0$, integer $i \ge 0$, and U being $U[0,1]$ distributed we have*

$$\mathbb{P}(\lambda(T^{-i}((U, U+\varepsilon)) \cap A) \ge \delta) \le \frac{\varepsilon \lambda(A)}{\delta},$$

where $\lambda(\,\cdot\,)$ denotes Lebesgue measure.

Lemma 9.5. *For all $n \ge 1$, $a \in [0,1)$, and $\Delta \in (0, 1/\sqrt{n})$ with $a + \Delta \le 1$, we have*

$$\mathbb{P}\Big(Y_1^{\langle m \rangle}, \ldots, Y_{L/2}^{\langle m \rangle} \notin [a, a+\Delta]\Big) \le 1 - \frac{L\Delta}{4} + \frac{2L}{n},$$

where $L = \|\lfloor \log_2 n \rfloor\|$ and $m = 18L$.

References

[1] Antos, A. and Devroye, L. (2000) Rawa Trees. *Mathematics and Computer Science (Versailles, 2000)*, 3–15, Birkhäuser, Basel.

[2] Apostolico, A. (1985) The myriad virtues of suffix trees. *Combinatorial Algorithms on Words*, 85–96, Springer-Verlag.

[3] Billingsley, P. (1979) *Probability and Measure.* John Wiley, New York-Chichester-Brisbane.

[4] Chung, K. L. and Erdös, P. (1952) On the application of the Borel-Cantelli lemma. *Trans. Amer. Math. Soc.* **72**, 179–186.

[5] Crochemore, M. and Rytter, W. (1994) *Text Algorithms.* Oxford University Press, New York,

[6] Devroye, L. (1986) A note on the height of binary search trees. *Journal of the ACM* **33**, 489–498.

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

[8] Devroye, L. (1994) On random Cartesian trees. *Random Structures Algorithms* **5**, 305–327.

[9] Devroye, L. and Goudjil, A. (1998) A study of random Weyl trees. *Random Structures Algorithms* **12**, 271–295.

[10] Devroye, L. and Neininger, R. (2002) Random suffix search trees. Technical Report, McGill University.

[11] Devroye, L., Szpankowski, W. and Rais, B. (1992) A note on the height of suffix trees. *SIAM Journal on Computing* **21**, 48–53.

[12] Farach, M. (1997) Optimal suffix tree construction with large alphabets. *IEEE Symp. Found. Computer Science*, 137—143.

[13] Farach, M. and Muthukrishnan, S. (1996) Optimal logarithmic time randomized suffix tree construction. *Proc. 23rd ICALP*, 550–561.

[14] Farach, M. and Muthukrishnan, S. (1997) An optimal, logarithmic time, randomized parallel suffix tree contruction algorithm. *Algorithmica* **19**, 331–353.

[15] Giancarlo, R. (1993) The suffix tree of a square matrix, with applications. *Proc. of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, 402–411.

[16] Giancarlo, R. (1995) A generalization of the suffix tree to square matrices, with applications. *SIAM Journal on Computing*, 520–562.

[17] Giegerich, R. and Kurtz, S. (1995) A comparison of imperative and purely functional suffix tree constructions. *Science of Computer Programming* **25**, 187–218.

[18] Giegerich, R. and Kurtz, S. (1997) From Ukkonen to McCreight and Weiner: a unifying view of linear-time suffix tree construction. *Algorithmica* **19**, 331–353.

[19] Gusfield, D. (1997) *Algorithms on Strings, Trees, and Sequences. Computer Science and Computational Biology.* Cambridge University Press, Cambridge.

[20] Jacquet, P., Rais, B. and Szpankowski, B. (1995) Compact suffix trees resemble PATRICIA tries: limiting distribution of depth. Technical Report RR-1995, Department of Computer Science, Purdue University.

[21] Karkkainen, J. (1995) Suffix cactus : a cross between suffix tree and suffix array. *Combinatorial Pattern Matching, Proc. 6th Symposium on Combinatorial Pattern Matching, CPM 95* **937**, 191–204.

[22] Knuth, D. E. (1973) *The Art of Computer Programming, Vol. 1: Fundamental Algorithms.* Addison-Wesley, Reading, Mass., 2nd Ed.

[23] Knuth, D. E. (1973) *The Art of Computer Programming, Vol. 3: Sorting and Searching.* Addison-Wesley, Reading, Mass.

[24] Kosaraju, S. (1994) Real-time pattern matching and quasi-real-time construction of suffix trees. *Proc. of the 26th Ann. ACM Symp. on Theory of Computing*, 310–316, ACM.

[25] Kurlberg, P. and Rudnick, Z. (1999) The distribution of spacings between quadratic residues. *Duke Jour. of Math.* **100**, 211–242.

[26] Mahmoud, H. M. (1992) *Evolution of Random Search Trees.* John Wiley, New York.

[27] Manber, U. and Myers, G. (1990) Suffix arrays: a new method for on-line string searches. *Proceedings of the First Annual ACM-SIAM Symposium on Discrete Algorithms*, 319–327. SIAM, Philadelphia.

[28] McCreight, E. M. (1976) A space-economical suffix tree construction algorithm. *Journal of the ACM* **23**, 262–272.

[29] Rudnick, Z. and Zaharescu, A. (2002) The distribution of spacings between fractional parts of lacunary sequences. *Forum Math.*, to appear.

[30] Sahinalp, S. C. and Vishkin, U. (1994) Symmetry breaking for suffix tree construction. *Proc. 26th Symp. on Theory of Computing*, 300–309.

[31] Stephen, G. A. (1994) *String Searching Algorithms.* World Scientific, Singapore.

[32] Szpankowski, W. (1993) A Generalized Suffix Tree and its (Un)Expected Asymptotic Behaviors. *SIAM Journal on Computing* **22**, 1176–1198.

[33] Szpankowski, W. (2001) *Average-Case Analysis of Algorithms on Sequences.* John Wiley, New York.

[34] Ukkonen, E. (1995) On-line construction of suffix trees. *Algorithmica* **14**, 249–260.

[35] Weiner, P. (1973) Linear pattern matching algorithms. *Proceedings 14th Annual Symposium on Switching and Automata Theory*, 1–11. IEEE Press, New York.

Luc Devroye and Ralph Neininger

School of Computer Science
McGill University
3480 University Street
Montreal, H3A 2K6
Canada
{luc, neiningr}@cs.mcgill.ca