SIAM J. COMPUT.
Vol. 21, No. 1, pp. 48-53, February 1992

005

A NOTE ON THE HEIGHT OF SUFFIX TREES*

LUC DEVROYE†, WOJCIECH SZPANKOWSKI‡, AND BONITA RAIS§

Abstract. Consider a random word in which the individual symbols are drawn from a finite or infinite alphabet with symbol probabilities p_i, and let H_n be the height of the suffix tree constructed from the first n suffixes of this word. It is shown that H_n is asymptotically close to $2 \log n / \log (1/\sum_i p_i^2)$ in many respects: the difference is $O(\log \log n)$ in probability, and the ratio tends to one almost surely and in the mean.

Key words. suffix tree, height, trie hashing, analysis of algorithms, strong convergence algorithms on words

AMS(MOS) subject classifications. 68Q25, 68P05

C.R. classifications. 3.74, 5.25, 5.5

1. Introduction.

Tries are efficient data structures that were developed and modified by Fredkin [14]; Knuth [19]; Larson [21]; Fagin, Nievergelt, Pippenger, and Strong [10]; Litwin [23], [24]; Aho, Hopcroft, and Ullman [2]; and others. Multidimensional generalizations were given in Nievergelt, Hinterberger, and Sevcik [26] and Régnier [30]. One kind of trie, the suffix tree, is of particular utility in a variety of algorithms on strings (Aho, Hopcroft, and Ullman [1]; McCreight [25]; Apostolico [3]). However, except for the results in Apostolico and Szpankowski [5], who give an upper bound on the expected height (see also Szpankowski [32]), very little is known about the expected behavior of suffix trees. Also noteworthy is a result by Blumer, Ehrenfeucht, and Haussler [6] who obtained asymptotics for the expected size of the suffix tree under an equal probability model. The difficulty arises from the interdependence between the keys, which are suffixes of one string. In this note, we study the height of the suffix tree. The results of our analysis find applications in many areas (Aho, Hopcroft, and Ullman [1]; Apostolico [3]). For example, suffix trees are used to find the longest repeated substring (Weiner [33]), to find all squares or repetitions in strings (Apostolico and Preparata [4]), to compute string statistics (Apostolico and Preparata [4]), to perform approximate string matching (Landau and Vishkin [20]; Galil and Park [15]), to compress text (Lempel and Ziv [22]; Rodeh, Pratt, and Even [29]), to analyze genetic sequences, to identify biologically significant motif patterns in DNA (Chung and Lawler [8]), to perform sequence assembly (Chung and Lawler [8]), and to detect approximate overlaps in strings (Chung and Lawler [8]). Consequences of our findings for an efficient design of algorithms are extensively discussed in Apostolico and Szpankowski [5].

* Received by the editors July 13, 1989; accepted for publication (in revised form) March 31, 1991.

† School of Computer Science, McGill University, 805 Sherbrooke Street West, Montreal, Canada H3A 2K6. This research was performed while the author was visiting Institut National de Recherche en Informatique et en Automatique, Domaine de Voluceau, Rocquencourt, B.P. 105, 78153 Le Chesnay, France. This research was sponsored by National Sciences and Engineering Research Council of Canada grant A3456 and by Fonds Le Chercheurs et d'Actions Concertées grant EQ-1678.

‡ Department of Computer Science, Purdue University, West Lafayette, Indiana 47907. This research was performed while the author was visiting Institut National de Recherche en Informatique et en Automatique, Domaine de Voluceau, Roquencourt, B.P. 105, 78153 Le Chesnay, France. This research was supported in part by North Atlantic Treaty Organization Collaborative grant 0057/89; by National Science Foundation grants NCR-8702115, CCR-8900305, and INT-8912631; by Air Force grant 90-0107; and by National Library of Medicine grant R01 LM05118.

§ Department of Computer Science, Purdue University, West Lafayette, Indiana 47907. This research was partially supported by National Science Foundation grant CCR-8900305 and Air Force grant 90-0107.

We consider an independently and identically distributed (i.i.d.) sequence $X_1, X_2, \cdots$ of integer-valued nonnegative random variables with $\mathbf{P}(X_1 = i) = p_i$ for $i = 0, 1, 2, \cdots$ and $\sum_i p_i = 1$. The X_i's should be considered as symbols in some alphabet. Together, they form a word $X = X_1X_2X_3 \cdots$. We do not assume that the alphabet is finite, but we will assume that no p_i is one, for otherwise all the symbols are identical with probability one. The *suffixes* Y_i of X are obtained by forming the sequences $Y_i = (X_iX_{i+1} \cdots)$. The *suffix* tree based upon $Y_1, \cdots, Y_n$ is the trie obtained when the Y_i's are used as words (for a definition of tries, see Knuth [19]; for a survey of recent results, see Szpankowski [31], [32]). Note, however, that we do not compress the trie as in a PATRICIA trie, i.e., no substrings are collapsed into one node.

In this note we study the *height* H_n of the suffix tree, which is given by

$$H_n = \max_{i \neq j, 1 \leqq i, j \leqq n} C_{ij},$$

where C_{ij} is the length of the longest common prefix of Y_i and Y_j, i.e., $C_{ij} = k$ if

$$(X_i, \cdots, X_{i+k-1}) = (X_j, \cdots, X_{j+k-1}) \quad \text{and} \quad X_{i+k} \neq X_{j+k}.$$

In the discussions to follow, we will use the standard notations for the L_r-metric: $\|p\|_r = (\sum_i p_i^r)^{1/r}$, where $0 < r < \infty$, and $\|p\|_\infty = \max_i p_i$. We write $f(n) \sim g(n)$ if $\lim_{n \to \infty} f(n)/g(n) = 1$, and we will reserve the symbol Q to stand for $1/\|p\|_2$.

THEOREM. *For a random suffix tree, $H_n / \log_Q n \to 1$ in probability. Also, for all $m \geqq 1$, $\mathbf{E}H_n^m \sim (\log_Q n)^m$.*

We will prove this result using only elementary probability theoretical tools, such as the second moment method. Nevertheless, we will in fact be able to show that for any $\varepsilon > 0$ and any sequence $\omega_n \uparrow \infty$,

$$\text{(1)} \qquad \lim_{n \to \infty} \mathbf{P}(H_n > \log_Q n + \omega_n) = 0$$

and

$$\text{(2)} \qquad \lim_{n \to \infty} \mathbf{P}(H_n < \log_Q n - (1+\varepsilon) \log_Q \log n) = 0.$$

Thus, the variations of H_n are at most of the order of $\log \log n$. In § 4, we will show that the convergence in the theorem is in the almost sure sense as well.

It is interesting to note that the first asymptotic term ($\log_Q n$) is of the same order of magnitude as for the asymmetric trie when the words $Y_1, \cdots, Y_n$ are i.i.d. (Pittel [27], [28]; Szpankowski [32]). In [27], Pittel showed that $H_n / \log_Q n \to 1$ almost surely, and in [28], he showed that $H_n - \log_Q n = O(1)$ in probability. Other properties of the height of a trie under the independent model can be found in Yao [34]; Régnier [30]; Flajolet [11]; Devroye [9]; Pittel [27], [28]; Jacquet and Régnier [16]; and Szpankowski [32], who presents a survey of recent results. The reader is also referred to some other related papers, such as Kirschenhofer and Prodinger [18], Flajolet and Puech [12], Flajolet and Sedgewick [13], and Szpankowski [31].

2. Preliminary results.

We present four simple lemmas. The first two are trivial. The third one is due to Apostolico and Szpankowski [5].

LEMMA 1.

$$\|p\|_\infty^2 \leqq \|p\|_2^2 \leqq \|p\|_\infty.$$

LEMMA 2. *For every $r \geqq 2$, $\|p\|_r \leqq \|p\|_2$.*

Proof. Let $f(x) = \{\sum_i p_i^x\}^{1/x}$ for $x > 0$. It is easy to show that the first derivative of $f(x)$ is negative for all $x > 0$, and hence f is a decreasing function. For details, see Szpankowski [32] and Karlin and Ost [17]. □

LEMMA 3. *For $0<|i-j|=d$, we have*

$$\mathbf{P}(C_{ij}\geqq k)=\left(\sum_s p_s^{l+2}\right)^r\left(\sum_s p_s^{l+1}\right)^{d-r},$$

where $l=\lfloor k/d\rfloor$ and $r=k-dl=k \bmod d$. In particular, for $|i-j|\geqq k$, we have $\mathbf{P}(C_{ij}\geqq k)=\|p\|_2^{2k}$.

LEMMA 4. *For $0<|i-j|=d<k$, we have $\mathbf{P}(C_{ij}\geqq k)\leqq\|p\|_2^{k+d}$.*

Proof. From Lemmas 2 and 3 we immediately obtain

$$\mathbf{P}(C_{ij}\geqq k)=\left(\sum_s p_s^{l+2}\right)^r\left(\sum_s p_s^{l+1}\right)^{d-r}\leqq\|p\|_2^{(l+2)r+(l+1)(d-r)}=\|p\|_2^{k+d}. \qquad \square$$

3. Proof of the theorem. We prove our theorem by showing two tight bounds for the height H_n. Roughly speaking, we shall show that for every $\varepsilon>0$ and large n the following holds: $\mathbf{P}(H_n>(1+\varepsilon)\cdot\log_Q n)\to 0$ as $n\to\infty$ (upper bound), and $\mathbf{P}(H_n<(1-\varepsilon)\cdot\log_Q n)\to 1$ as $n\to\infty$ (lower bound).

We start with an easier part of our proof, namely, the upper bound. Assume that $2\leqq k\leqq n-1$. We have, from Lemmas 2 and 4 and Bonferroni's inclusion-exclusion inequality for the probability of the union of events,

$$\begin{aligned}\mathbf{P}(\max_{i\neq j} C_{ij}\geqq k)&\leqq 2n\left(\sum_{d=1}^{k-1}\mathbf{P}(C_{1,1+d}\geqq k)+\sum_{d=k}^{n-1}\mathbf{P}(C_{1,1+d}\geqq k)\right)\\ &\leqq 2n\left(\sum_{d=1}^{k-1}\|p\|_2^{k+d}+\sum_{d=k}^{n-1}\|p\|_2^{2k}\right) \qquad (3)\\ &\leqq 2n\left(\frac{\|p\|_2^{k+1}}{1-\|p\|_2}+n\|p\|_2^{2k}\right).\end{aligned}$$

This tends to zero provided that $\|p\|_2<1$ (this is always true) and that $n\|p\|_2^k\to 0$ (for this, it suffices that $k=(\log n+\omega_n)/(-\log\|p\|_2)$, with $\omega_n\to\infty$). This establishes (1). Let u_+ be defined as $\max(u,0)$. Clearly, $\mathbf{E}H_n\leqq\log_Q n+\mathbf{E}(H_n-\log_Q n)_+$. We will show that the second term in this upper bound is $O(1)$. Indeed, by (3),

$$\begin{aligned}\mathbf{E}(H_n\log(1/\|p\|_2)-\log n)_+^m&=\int_0^\infty\mathbf{P}(H_n\log(1/\|p\|_2)-\log n>u^{1/m})\,du\\ &\leqq\int_0^\infty\left(\frac{2e^{-u^{1/m}}}{1-\|p\|_2}+\frac{2e^{-2u^{1/m}}}{\|p\|_2}\right)du<\infty.\end{aligned}$$

A matching lower bound is obtained by the second moment method. We will use a form due to Chung and Erdös [7], which states that for events A_i, we have

$$\mathbf{P}(\cup_i A_i)\geqq\frac{(\sum_i\mathbf{P}(A_i))^2}{\sum_i\mathbf{P}(A_i)+\sum_{i\neq j}\mathbf{P}(A_i\cap A_j)}.$$

Let S be the collection of pairs of indices (i,j) with $1\leqq i,j\leqq n$, and $|i-j|\geqq k$. Let A_{ij} be the event that $C_{ij}\geqq k$. Then

$$\mathbf{P}(\max_{i\neq j} C_{ij}\geqq k)\geqq\mathbf{P}(\cup_{(i,j)\in S}A_{ij})\geqq\frac{\mathcal{P}^2}{\mathcal{P}+\mathcal{Q}},$$

where

$$\mathcal{P} \overset{\text{def}}{=} \sum_{(i,j)\in S} \mathbf{P}(A_{ij})$$

and

$$\mathcal{Q} \overset{\text{def}}{=} \sum_{(i,j),(l,m)\in S;(i,j)\neq(l,m)} \mathbf{P}(A_{ij} \cap A_{lm}).$$

To prove our lower bound it is enough to show that the probability on the right-hand side (RHS) of the above tends to 1 for k slightly smaller than $\log_Q n$ ($k = \log_Q n - \omega_n$). First we note that when $k = o(n)$, then by Lemma 3,

$$\mathcal{P} = \sum_{(i,j)\in S} \mathbf{P}(A_{ij}) = |S| \|p\|_2^{2k} \in [(n^2-(2k+1)n)\|p\|_2^{2k}, n^2\|p\|_2^{2k}].$$

We decompose the collection of pairs of pairs of indices

$$\{((i,j),(l,m)) : (i,j)\in S, (l,m)\in S, (i,j)\neq(l,m)\}$$

as follows into $I_1 \cup I_2 \cup I_3$: I_1 captures all members with $\min(|l-i|, |l-j|) \geqq k$ and $\min(|m-i|, |m-j|) \geqq k$. I_2 holds all members with either $\min(|l-i|, |l-j|) \geqq k$ and $\min(|m-i|, |m-j|) < k$, or $\min(|l-i|, |l-j|) < k$ and $\min(|m-i|, |m-j|) \geqq k$. Finally, I_3 collects all members with $\min(|l-i|, |l-j|) < k$ and $\min(|m-i|, |m-j|) < k$. By Lemmas 1 and 2,

$$\sum_{((i,j),(l,m))\in I_1} \mathbf{P}(A_{ij} \cap A_{lm}) \leqq n^4 \|p\|_2^{4k},$$

$$\sum_{((i,j),(l,m))\in I_2} \mathbf{P}(A_{ij} \cap A_{lm}) \leqq 8kn^3 \|p\|_2^{2k} \|p\|_\infty^k \leqq 8kn^3 \|p\|_2^{3k},$$

$$\sum_{((i,j),(l,m))\in I_3} \mathbf{P}(A_{ij} \cap A_{lm}) \leqq (4k)^2 n^2 \|p\|_2^{2k}.$$

Thus,

$$\mathcal{Q} \leqq n^4/Q^{4k} + 8kn^3/Q^{3k} + 16k^2n^2/Q^{2k}.$$

If we choose k such that $n\|p\|_2^k/k \to \infty$, then

$$\mathcal{Q} = \sum_{(i,j),(l,m)\in S;\ (i,j)\neq(l,m)} \mathbf{P}(A_{ij} \cap A_{lm}) \sim n^4 \|p\|_2^{4k}.$$

Because

$$\mathcal{Q} - \mathcal{P}^2 \leqq 8kn^3/Q^{3k} + 16k^2n^2/Q^{2k} + 2(2k+1)n^3/Q^{4k}$$

and $\mathcal{P} \leqq n^2/Q^{2k}$, we have

$$\mathbf{P}(\max_{i\neq j} C_{ij} < k) \leqq \frac{\mathcal{P} + \mathcal{Q} - \mathcal{P}^2}{\mathcal{P} + \mathcal{Q}}$$

$$\leqq \frac{n^2/Q^{2k} + 8kn^3/Q^{3k} + 16k^2n^2/Q^{2k} + 2(2k+1)n^3/Q^{4k}}{n^2/Q^{2k} + (1+o(1))n^4/Q^{4k}} \tag{4}$$

$$\sim \frac{8kQ^k}{n}.$$

Collecting all these terms shows that $\mathbf{P}(H_n \geqq k) \to 1$ when $n \to \infty$. The lower bound in (2) follows by setting $k = \lfloor (\log n - (1+\varepsilon)\log\log n)/(-\log\|p\|_2) \rfloor$ for $\varepsilon > 0$. Also,

$$\mathbf{E}H_n^m \geqq k^m \mathbf{P}(H_n \geqq k) \sim k^m$$

if k is chosen as indicated. This concludes the proof of the lower bound and of the theorem. □

4. Strong convergence.

PROPOSITION. *For the suffix tree, $H_n/\log_Q n \to 1$ almost surely.*

Proof. We observe that H_n is monotone ↑. Thus, if a_n is a monotone ↑ sequence, we have $H_n > a_n$ finitely often if $H_{2^i} > a_{2^{i-1}}$ finitely often in i. Similarly, $H_n < a_n$ finitely often if $H_{2^i} < a_{2^{i+1}}$ finitely often in i. By the Borel-Cantelli lemma, the proposition is proved if we can show that for all $\varepsilon > 0$,

$$\sum_{i=1}^{\infty} \mathbf{P}\{H_{2^i} > (1+\varepsilon) i \log_Q 2\} < \infty \tag{5}$$

and

$$\sum_{i=1}^{\infty} \mathbf{P}\{H_{2^i} < (1-\varepsilon) i \log_Q 2\} < \infty. \tag{6}$$

To show (5), we can use the inequality (3) with $n = 2^i$ and $k = \lceil (1+\varepsilon) i \log_Q 2 \rceil$. Note that $Q^k \geqq 2^{(1+\varepsilon)i}$. The ith term in (5) is not larger than

$$\frac{2n}{(Q-1)Q^k} + 2\left(\frac{n}{Q^k}\right)^2 \leqq \frac{2}{(Q-1)2^{\varepsilon i}} + \frac{2}{2^{2\varepsilon i}},$$

which is summable in i. Similarly, to verify (6), we use (4) with $n = 2^i$ and $k = \lfloor (1-\varepsilon) i \log_Q 2 \rfloor$. The ith term in (6) does not exceed

$$(1+o(1))\frac{8kQ^k}{n} \leqq (8+o(1)) i(1-\varepsilon)(\log_Q 2) 2^{-\varepsilon i},$$

which is summable in i, as required. □

REFERENCES

[1] A. V. AHO, J. E. HOPCROFT, AND J. D. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1975.

[2] ———, *Data Structures and Algorithms*, Addison-Wesley, Reading, MA, 1983.

[3] A. APOSTOLICO, *The myriad virtues of suffix trees*, in Combinatorial Algorithms on Words, Springer-Verlag, Berlin, New York, 1985, pp. 85-96.

[4] A. APOSTOLICO AND F. P. PREPARATA, *Optimal off-line detection of repetitions in a string*, Theoret. Comput. Sci., 22 (1983), pp. 297-315.

[5] A. APOSTOLICO AND W. SZPANKOWSKI, *Self-alignments in words and their applications*, J. Algorithms, 1991.

[6] A. BLUMER, A. EHRENFEUCHT, AND D. HAUSSLER, *Average sizes of suffix trees* and DAWGs, Discrete Appl. Math., 24 (1989), pp. 37-45.

[7] K. L. CHUNG AND P. ERDÖS, *On the application of the Borel-Cantelli lemma*, Trans. Amer. Math. Soc., 72 (1952), pp. 179-186.

[8] W. CHUNG AND E. LAWLER, *Approximate string matching in sublinear expected time*, in Proc. 32nd IEEE Conference on the Foundations of Computer Science, 1990, pp. 116-124.

[9] L. DEVROYE, *A probabilistic analysis of the height of tries and of the complexity of triesort*, Acta Inform., 21 (1984), pp. 229-237.

[10] R. FAGIN, J. NIEVERGELT, N. PIPPENGER, AND H. R. STRONG, *Extendible hashing—a fast access method for dynamic files*, ACM Trans. Database Systems, 4 (1979), pp. 315-344.

[11] P. FLAJOLET, *On the performance evaluation of extendible hashing and trie search*, Acta Inform., 20 (1983), pp. 345-369.

[12] P. FLAJOLET AND C. PUECH, *Tree structure for partial match retrieval*, J. Assoc. Comput. Mach., 33 (1986), pp. 371-407.

[13] P. Flajolet and R. Sedgewick, *Digital search trees revisited*, SIAM J. Comput., 15 (1986), pp. 748-767.

[14] E. H. Fredkin, *Trie memory*, Comm. ACM, 3 (1960), pp. 490-500.

[15] Z. Galil and K. Park, *An improved algorithm for approximate string matching*, SIAM J. Comput., 19 (1990), pp. 989-999.

[16] P. Jacquet and M. Régnier, *Trie partitioning process: Limiting distributions*, Lecture Notes in Computer Science, 214, Springer-Verlag, Berlin, 1986, pp. 196-210.

[17] S. Karlin and F. Ost, *Some monotonicity properties of Schur powers of matrices and related inequalities*, Linear Algebra Appl., 68 (1985), pp. 47-65.

[18] P. Kirschenhofer and H. Prodinger, *Further results on digital trees*, Theoret. Comput. Sci., 58 (1988), pp. 143-154.

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

[20] G. Landau and U. Vishkin, *Introducing efficient parallelism into approximate string matching*, in Proc. 18th Annual ACM Symposium on the Theory of Computing, 1986, pp. 220-230.

[21] P. A. Larson, *Dynamic hashing*, BIT, 18 (1978), pp. 184-201.

[22] A. Lempel and J. Ziv, *On the complexity of finite sequences*, IEEE Trans. Inform. Theory, IT-22 (1976), pp. 75-81.

[23] W. Litwin, *Trie hashing*, in Proc. ACM-SIGMOD Conference on Management of Data, Ann Arbor, MI, 1981.

[24] ———, *Trie hashing: Further properties and performances*, in Proc. Internat. IEEE Conference on Foundations of Data Organization, Kyoto, 1985.

[25] E. M. McCreight, *A space-economical suffix tree construction algorithm*, J. Assoc. Comput. Mach., 23 (1976), pp. 262-272.

[26] J. Nievergelt, H. Hinterberger, and K. C. Sevcik, *The grid file: An adaptable, symmetric multikey file structure*, ACM Trans. Database Systems, 9 (1984), pp. 38-71.

[27] B. Pittel, *Asymptotical growth of a class of random trees*, Ann. Probab., 13 (1985), pp. 414-427.

[28] ———, *Path in a random digital tree: Limiting distributions*, Adv. in Appl. Probab., 18 (1986), pp. 139-155.

[29] M. Rodeh, V. Pratt, and S. Even, *Linear algorithm for data compression via string matching*, J. Assoc. Comput. Mach., 28 (1981), pp. 16-24.

[30] M. Régnier, *On the average height of trees in digital searching and dynamic hashing*, Inform. Process. Lett., 13 (1981), pp. 64-66.

[31] W. Szpankowski, *Some results on V-ary asymmetric tries*, J. Algorithms, 9 (1988), pp. 224-244.

[32] ———, *On the height of digital trees and related problems*, Algorithmica, 6 (1991), pp. 256-277.

[33] P. Weiner, *Linear pattern matching algorithms*, in Proc. 14th ACM Annual Symposium on Switching and Automata Theory, 1973, pp. 1-11.

[34] A. Yao, *A note on the analysis of extendible hashing*, Inform. Process. Lett., 11 (1980), pp. 84-86.