An Estimation of the Size of Non-Compact Suffix Trees

Bálint Vásárhelyi*

*Szegedi Tudományegyetem, TTIK, Szeged, 6720, Hungary. E-mail: mesti@math.u-szeged.hu
Abstract

A suffix tree is a data structure used mainly for pattern matching. It is known that the space complexity of suffix trees is $O(n^2)$. By a slight modification of the simple suffix trees one gets the compact suffix trees, which have $O(n)$ space complexity. The motivation of this paper is the question of whether the space complexity simple suffix trees is $O(n^2)$ not only in special cases, but also in expectation.
1 Introduction

The suffix tree is a special data structure which is mainly used for a large number of combinatorial problems involving strings. The notion was introduced by Weiner [6], while Gusfield gave a detailed explanation of suffix trees [3].

Let $S$ be a string over the alphabet $\Sigma$. The size of $\Sigma$ is denoted by $\sigma$. Let $S[i, j]$ denote the substring of $S$ from position $i$ to position $j$. The suffix tree of $S$ is a rooted directed tree with $n$ leaves, where $n$ is the length of $S$. Each edge $e$ has a label $\ell(e)$. On a path from the root to leaf $j$ the suffix $S[j, n]$ can be found with a $\$$ sign at the end. Such a path is denoted by $\mathcal{P}(j)$. By concatenating the edge labels on the path $\mathcal{P}(j)$ we get the path label of $j$, which is denoted by $L(j)$. For an example, see Figure 1.

![Suffix tree of string aabccb](image)

Figure 1: Suffix tree of string aabccb

By compressing the long branches of the suffix tree we get the so-called compact suffix tree in which the label of an edge may consist of more than one character and each internal node has at least two children. The compact suffix tree has $O(n)$ nodes, compared to the (non-compact) suffix tree, which may have $O(n^2)$ nodes. Through this paper, when suffix tree is mentioned without further distinction, always non-compact suffix tree should be meant.

The compact suffix trees can be constructed in linear time. Ukkonen [5], Weiner [6] and McCreight [4] gave linear time algorithms.

An important question is the height of suffix trees, which was answered by Devroye, Szpankowski and Rais [2]. The average size of compact suffix trees was also examined by Blumer, Ehrenfeucht and Haussler [1]. In this paper we show that for a random string the simple or non-compact suffix tree indeed contains $O(n^2)$ nodes in expectation.
2 Main results

Let $S$ be a string of length $n$. We say that the growth of $S$ ($\gamma(S)$) is the minimal depth of leaf 1 from its least common ancestor with any leaf $j$. Obviously, $j$ is such that $S[j, n]$ has the longest common prefix with $S[1, n]$. Let $\Omega(n, k)$ be the number of strings of length $n$ with growth $k$.

Observe that the growth is $k$ if $S[1, n - k] = S[j, n - k + j - 1]$ for a certain $j > 1$.

Our main results are formulated in the following theorems.

**Theorem 1.** On an alphabet $\Sigma$ of size $\sigma$ for all $n \geq 2k - 1$, $\Omega(n, k) = \phi(k)$ for some $\phi(k)$, where $\phi(k)$ depends only on $\sigma$ and $k$.

**Theorem 2.** Let $S'$ be a string of length $n - 1$, and $S$ be a string obtained from $S'$ by adding a character to its beginning chosen uniformly random from $\Sigma$. Then the expected growth of $S$ is at least $\frac{n}{2} + 1$.

**Theorem 3.** On an alphabet $\Sigma$ the simple suffix tree of a random string has at least $\frac{n(n+5)}{4}$ nodes.

Theorem 3 immediately follows from Theorem 2 by the following inequality:

$$\sum_{m=1}^{n} \mathbb{E}(\gamma(S_m)) \geq \sum_{m=1}^{n} \frac{m}{2} = \frac{n(n+1)}{4} = \Theta(n^2).$$  \hspace{1cm} (1)

3 Proofs

Let $\mu(j)$ be the number of $j$-length aperiodic strings. The following lemmas show us some important properties of $\mu$.

**Lemma 1.** $\mu(j) = \sigma^j - \sum_{d|j \atop d \neq j} \mu(d)$

**Proof.** $\mu(1) = \sigma$ is trivial.

There are $\sigma^j$ strings of length $j$. Suppose that a string is periodic with period exactly $d$. This implies that its first $d$ characters form an aperiodic string of length $d$, and there are $\mu(d)$ such strings. This finishes the proof.

Specially, if $p$ is prime, then $\mu(p) = \sigma^p - \sigma$.

**Lemma 2.** If $p$ is prime and $t \in \mathbb{N}$, then $\mu(p^t) = \sigma^{p^t} - \sigma^{p^t-1}$.

**Proof.** We count the aperiodic strings of length $p^t$. There are $\sigma^{p^t}$ strings. Consider the minimal period of the string, i.e. the period which is aperiodic. If we exclude all minimal periods of length $k$, we exclude $\mu(k)$ strings. This yields the following equality:

$$\mu(p^t) = \sigma^{p^t} - \sum_{1 \leq s < t} \mu(p^s).$$  \hspace{1cm} (2)
With a few transformations, we get

\[
\sigma^{p^t} - \mu(p^{t-1}) - \sum_{1 \leq s < t-1} \mu(p^s) = \sigma^{p^t} - \sigma^{p^{t-1}} + \sum_{1 \leq s < t-1} \mu(p^s) - \sum_{1 \leq s < t-1} \mu(p^s)
\]

\[
= \sigma^{p^t} - \sigma^{p^{t-1}}.
\]

**Lemma 3.** For all \( j > 1 \), \( \mu(j) \leq \sigma^j - \sigma \).

**Proof.** \( \mu(j) = \sigma^j - \sum_{d|j} \mu(d) \)

Considering \( \mu(d) \geq 0 \) and \( \mu(1) = \sigma \), we have \( \mu(j) \geq \sigma^j - \sigma \).

**Lemma 4.** For all \( j > 1 \), \( \mu(j) \geq \sigma^{j-1} \).

**Proof.** Given a string of length \( j - 1 \), there is at most one character which makes it periodic if we append it to the end. Therefore at least one character makes it aperiodic (as \( \sigma > 1 \)).

**Proof.** (Theorem [1])

For any value of \( k \) there are only a few possibilities to create a string with \( \gamma(S) = k \). First, we decide how the overlap will occur in the string, i.e., we choose an index \( j \leq k \) such that \( S[1, n - k] = S[j + 1, j + 1 + n - k] \). We call \( 1, \ldots, j \) **first-type quasi-free** while \( j + n - k + 2, \ldots, n \) **second-type quasi-free** positions.

Obviously, if the first-type quasi-free positions are given, \( S[j + 1, j + 1 + n - k] \) is determined due to the condition \( S[1, n - k] = S[j + 1, j + 1 + n - k] \). The first of the second-type quasi-free characters cannot be arbitrary: \( S[1, j + 1 + n - k] \) can be continued in exactly one way to make the overlap longer, therefore we have \( \sigma - 1 \) ways to ensure that the overlap has length exactly \( k \). The rest of the second-type quasi-free characters can be chosen arbitrarily, which gives \( (\sigma - 1)\sigma^{k-j-2} \) possibilities for a fixed \( j \).

The possibilities for the first-type quasi-free characters are slightly more difficult to count. We should be aware of the periods which can occur in this part of the string; if periodicity appears, the overlap becomes longer. This gives us the idea to deal with aperiodic strings.

Observe that \( S[j + 1, n - k + j] \) is determined by \( S[1, j] \) and \( S[n - k + j + 1, n] \).

Let \( j \) be fixed. If \( j = k \), then the growth is \( k \) if and only if \( S[1, k] \) is aperiodic, otherwise the overlap would be larger than \( n - k \). In this case we have \( \mu(k) \) strings with growth \( k \).

If \( j < k \), then \( S[1, j] \) must be aperiodic. This gives \( \mu(j) \) possibilities. The first character of \( S[n - k + j + 1, n] \) can be arbitrary, but one character, the others can be arbitrary, which gives \( (\sigma - 1)\sigma^{k-j-2} \) ways. Hence we get \( \mu(j)(\sigma - 1)\sigma^{k-j-2} \) more possibilities.

If we put these cases together, we have that

\[
\phi(k) = \sum_{j=1}^{k-1} \mu(j)(\sigma - 1)\sigma^{k-j-1} + \mu(k).
\]

This completes the proof. \qed
A few examples for the number of aperiodic strings are given in Table 1.

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 2 | 2  | 6 | 12 | 30 | 54 | 126 | 240 | 504 |
| 3 | 3  | 6 | 24 | 72 | 240 | 696 | 2184 | 6480 |
| 4 | 4  | 12 | 60 | 240 | 1020 | 4020 | 16380 | 65280 |
| 5 | 5  | 20 | 120 | 600 | 3120 | 15480 | 78120 | 390000 |

Table 1: Number of aperiodic strings for small alphabets. \( \sigma \) is the size of the alphabet. \( \mu(x) \) is the number of aperiodic strings of that length.

Proof. (Theorem 2)

According to Lemma 3, \( \mu(j) \leq \sigma^j - \sigma \) (if \( j > 1 \)).

By Theorem 1, for \( k \geq 1 \) and \( n \geq 2k - 1 \)

\[
\phi(k) = \mu(k) + \sum_{j=1}^{k-1} \mu(j)(\sigma - 1)\sigma^{k-j-1} \\
\leq \sigma^k - \sigma + \sum_{j=2}^{k-1} (\sigma^j - \sigma)(\sigma - 1)\sigma^{k-j-1} \\
= (k-1)(\sigma - 1)\sigma^{k-1} + \sigma. 
\]

Using Theorem 1 and 3

\[
\phi(k) = \mu(k) + \mu(1) + \sum_{j=2}^{k-1} \mu(j)(\sigma - 1)\sigma^{k-j-1} \\
\geq \sigma^k - \sigma + \sum_{j=2}^{k-1} (\sigma^j - \sigma)(\sigma - 1)\sigma^{k-j-1} \\
= \sigma^k + \sum_{j=2}^{k-1} (\sigma - 1)\sigma^{k-2} \\
= (k-2)(\sigma - 1)\sigma^{k-2} + \sigma^k. 
\]

Let \( m = \lceil \frac{n}{2} \rceil \).

Now

\[
\sum_{k=1}^{m} \phi(k) \leq \frac{(m-1)\sigma^{m+1} - m\sigma^m + m\sigma^2 - m\sigma - \sigma^2 + 2\sigma}{\sigma - 1} + \sigma \leq m\sigma^m. 
\]

As \( \sigma^n \gg \frac{\sigma}{2} \), this implies that in most cases the suffix tree will be expanded with more than \( \frac{n}{2} \) new internal nodes.
A lower bound on the expectation of $\gamma(S)$ is

$$
E(\gamma(S)) \geq \frac{1}{\sigma^n} \left( \frac{n}{2} \sigma^\frac{3}{2} + \left( \sigma^n - \frac{n}{2} \sigma^\frac{3}{2} \right) \left( \frac{n}{2} + 1 \right) \right)
\geq \frac{n}{2} + 1.
$$

(7)
References

[1] A. Blumer, A. Ehrenfeucht, and D. Haussler. Average sizes of suffix trees and DAWGs. *Discrete Applied Mathematics*, 24:37–45, 1989.

[2] L. Devroye, W. Szpankowski, and B. Rais.

[3] D. Gusfield. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press, 1997.

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

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

[6] P. Weiner. Linear pattern matching algorithms. In *Proceedings of the 14th IEEE Symposium on Switching and Automata Theory*, pages 1–11, 1973.