Partial Fillup and Search Time in LC Tries*

Svante Janson† Wojciech Szpankowski‡

Abstract

Andersson and Nilsson introduced in 1993 a *level-compressed trie* (in short: LC trie) in which a full subtree of a node is compressed to a single node of degree being the size of the subtree. Recent experimental results indicated a "dramatic improvement" when full subtrees are replaced by "partially filled subtrees". In this paper, we provide a theoretical justification of these experimental results showing, among others, a rather moderate improvement of the search time over the original LC tries. For such an analysis, we assume that n strings are generated independently by a binary memoryless source (a generalization to Markov sources is possible) with p denoting the probability of emitting a "1" (and $q = 1 - p$). We first prove that the so called α-fillup $F_n(\alpha)$ (i.e., the largest level in a trie with α fraction of nodes present at this level) is concentrated on two values whp (with high probability); either $F_n(\alpha) = k_n$ or $F_n(\alpha) = k_n + 1$ where $k_n = \log_{\frac{1}{\sqrt{pq}}} n - \frac{|\ln(p/q)|}{2\ln^{3/2}(1/\sqrt{pq})}\Phi^{-1}(\alpha)\sqrt{\ln n} + O(1)$ is an integer and $\Phi(x)$ denotes the normal distribution function. This result directly yields the typical depth (search time) $D_n(\alpha)$ in the α-LC tries with $p \neq 1/2$, namely we show that whp $D_n(\alpha) \approx C_1 \log\log n$ where $C_1 = 1/|\log(1 - h/\log(1/\sqrt{pq}))|$ and $h = -p\log p - q\log q$ is the Shannon entropy rate. This should be compared with recently found typical depth in the original LC tries which is $C_2 \log\log n$ where $C_2 = 1/|\log(1 - h/\log(1/\min\{p, 1-p\}))|$. In conclusion, we observe that α affects only the lower term of the α-fillup level $F_n(\alpha)$, and the search time in α-LC tries is of the same order as in the original LC tries.

1 Introduction

Tries and suffix trees are the most popular data structures on words [7]. A *trie* is a digital tree built over, say n, strings (the reader is referred to [12, 14, 24] for an in depth discussion of digital trees.) A string is stored in an external node of a trie and the path length to such a node is the shortest prefix of the string that is not a prefix of any other strings (cf. Figure 1). Throughout, we assume a binary alphabet. Then each branching node in a trie is a binary node. A special case of a trie structure is a *suffix trie* (tree) which is a trie built over suffixes of a *single* string.

Since 1960 tries were used in many computer science applications such as searching and sorting, dynamic hashing, conflict resolution algorithms, leader election algorithms, IP addresses lookup, coding, polynomial factorization, Lempel-Ziv compression schemes, and molecular biology. For example, in the internet IP addresses lookup problem [15, 22] one needs a fast algorithm that directs an incoming packet with a given IP address to its destination. As a matter of fact, this is the *longest matching prefix* problem, and standard tries are well suited for it. However, the search time is too large. If there are n IP addresses in the database, the search time is $O(\log n)$, and this is not acceptable. In order to improve the search time, Nilsson [15] introduced a novel data structure called the *level compressed trie* or in short LC trie (cf. Figure 1). In the LC trie we replace the root with a node of degree equal to the size of the largest *full subtree* emanating from the root (the depth of such a subtree is called the *fillup level*). This is further carried on recursively throughout the whole trie.

Some recent experimental results reported in [8, 18, 17] indicated a "dramatic improvement" in the search time when full subtrees are replaced by "partially fillup subtrees". In this paper, we provide a theoretical justification of these experimental results by considering α-LC tries in which one replaces a subtree with the last level only α% filled by a node of degree equal to the size of such a subtree (and we continue recursively). In order to understand theoretically the α-LC trie behavior, we study here the so called α-*fillup* level $F_n(\alpha)$ and the *typical depth* or the search time $D_n(\alpha)$. The α-fillup level is the last level in a trie that is α% filled up (e.g., in a binary trie level k is α% filled if it contains $\alpha 2^k$ nodes). The typical depth is the length of a path from the root to a randomly selected external node; thus it represents the typical search time.

*The work was supported by NSF Grants CCR-0208709 and DMS-02-02815, NIH Grant R01 GM068959-01, and NSA Grant MDA 904-03-1-0036

†Dept. Mathematics, Uppsala University, P.O. Box 480, SE-751 06 Uppsala, Sweeden.

‡Department of Computer Science, Purdue University, West Lafayette, IN 47907-2066 U.S.A.

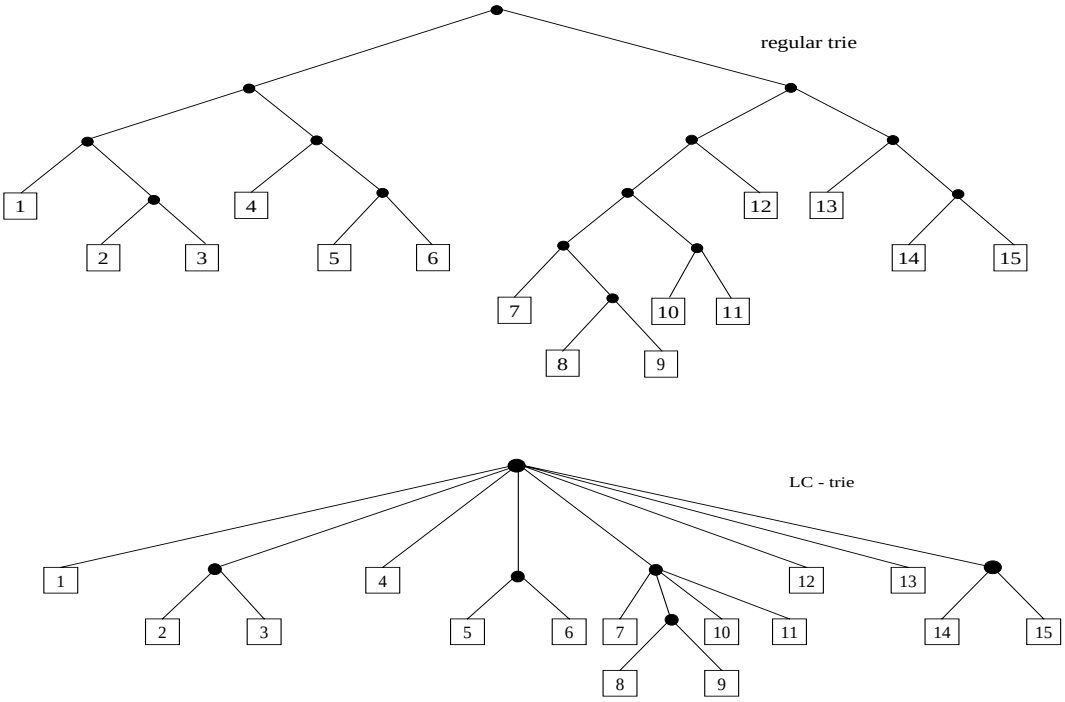

Figure 1: A trie and its associated full LC trie.

In this paper we analyze the α-fillup and the typical depth in an α-LC trie in a probabilistic framework when all strings are generated by a memoryless source with $\mathbb{P}(1) = p$ and $\mathbb{P}(0) = q := 1 - p$. Among other results, we prove that the α-LC trie shows a rather moderate improvement over the original LC tries. We shall quantify this statement below.

Tries were analyzed over the last thirty years for memoryless and Markov sources (cf. [2, 9, 11, 12, 14, 19, 20, 23, 24]). Pittel [19, 20] found the typical value of the fillup level F_n (i.e., $\alpha = 1$) in a trie built over n strings generated by mixing sources; for memoryless sources with high probability (whp)

$$F_n \overset{\text{p}}{\sim} \frac{\log n}{\log(1/p_{\min})} = \frac{\log n}{h_{-\infty}}$$

where $p_{\min} = \min\{p, 1-p\}$ is the smallest probability of generating a symbol and $h_{-\infty} = \log(1/p_{\min})$ is the Rényi entropy of infinite order (cf. [24]). We let $\log := \log_2$.

This was further extended by Devroye [2], and Knessl and Szpankowski [11] who, among other results, proved that the fillup F_n is concentrated on two points k_n and $k_n + 1$, where k_n is an integer

$$\frac{1}{\log p_{\min}^{-1}} (\log n - \log\log\log n) + O(1) \tag{1.1}$$

for $p \neq 1/2$. The depth in regular tries was analyzed by many authors who proved that whp the depth is about $(1/h)\log n$ (where $h = -p\log p - (1-p)\log(1-p)$ is the Shannon entropy rate of the source) and that it is normally distributed when $p \neq 1/2$ [20, 24].

The original LC tries were analyzed by Andersson and Nilsson [1] for unbiased memoryless source and by Devroye [3] for memoryless sources (cf. also [21]). The typical depth (search time) for regular LC tries was only studied recently by Devroye and Szpankowski [4] who proved that for memoryless sources with $p \neq 1/2$

$$D_n \overset{\text{p}}{\sim} \frac{\log\log n}{-\log(1 - h/h_{-\infty})}.$$

In this paper we shall prove some rather surprising results. First of all, for $0 < \alpha < 1$ we show that the α-fillup $F_n(\alpha)$ is whp equal either to k_n or $k_n + 1$ where

$$k_n = \log_{\frac{1}{\sqrt{pq}}} n - \frac{|\ln(p/q)|}{2\ln^{3/2}(1/\sqrt{pq})}\Phi^{-1}(\alpha)\sqrt{\ln n} + O(1). \tag{1.2}$$

As a consequence, we find that if $p \neq 1/2$, the depth $D_n(\alpha)$ of the α-LC is for large n typically about

$$\frac{\log\log n}{-\log\left(1 - h/\log(1/\sqrt{pq})\right)}.$$

The (full) 1-fillup F_n shown in (1.1) should be compared to the α-fillup $F_n(\alpha)$ presented in (1.2). Observe that the leading term of $F_n(\alpha)$ is *not* the same as the leading term of F_n when $p \neq 1/2$. Furthermore, α contributes only to the second term asymptotics. When comparing the typical depths D_n and $D_n(\alpha)$ we conclude that both grow like $\log\log n$ with two constants that do not differ by much. This comparison led us to a statement in the abstract that the improvement of α-LC tries over the regular LC tries is rather moderate. We may add that for relatively slowly growing functions such as $\log\log n$ the constants in front of them do matter (even for large values of n) and perhaps this led the authors of [8, 17, 18] to their statements.

The paper is organized as follows. In the next section we present our main results which are proved in the next two sections. We first consider a poissonized version of the problem for which we establish our findings. Then we show how to depoissonize our results completing our proof.

2 Main Results

Consider tries created by inserting n random strings of 0 and 1. We will always assume that the strings are (potentially) infinite and that the bits in the strings are independent random bits, with $\mathbb{P}(1) = p$ and thus $\mathbb{P}(0) = q := 1 - p$; moreover we assume that different strings are independent.

We let $X_k := \#\{\text{internal nodes filled at level } k\}$ and $\overline{X}_k := X_k/2^k$, i.e. the proportion of nodes filled at level k. Note that X_k may both increase and decrease as k grows, while

$$1 \geq \overline{X}_k \geq \overline{X}_{k+1} \geq 0.$$

Recall that the fillup level of the trie is defined as the last full level, i.e. $\max\{k : \overline{X}_k = 1\}$, (and for example the height is the last level with any nodes at all, i.e. $\max\{k : \overline{X}_k > 0\}$). Similarly, if $0 < \alpha \leq 1$, the α-fillup level is the last level where at least a proportion α of the nodes are filled, i.e. $\max\{k : \overline{X}_k \geq \alpha\}$.

We will in this paper study the α-fillup level for a given α with $0 < \alpha < 1$ and a given p with $0 < p < 1$.

We have the following result, where whp means with probability tending to 1 as $n \to \infty$, and Φ denotes the normal distribution function. Theorem 2.1 is proved in Section 4, after first considering a Poissonized version in Section 3.

THEOREM 2.1. *Let α and p be fixed with $0 < \alpha < 1$ and $0 < p < 1$, and let $F_n(\alpha)$ be the α-fillup level for the trie formed by n random strings as above. Then, for each n*

there is an integer

$$k_n = \log_{\frac{1}{\sqrt{pq}}} n - \frac{|\ln(p/q)|}{2\ln^{3/2}(1/\sqrt{pq})}\Phi^{-1}(\alpha)\sqrt{\ln n} + O(1)$$

such that whp $F_n(\alpha) = k_n$ *or* $k_n + 1$. *Moreover,* $\mathbf{E}\,\overline{X}_{k_n} = \alpha + O(1/\sqrt{\log n})$ *for* $p \neq 1/2$.

Thus the α-fillup $F_n(\alpha)$ is concentrated on at most two values; as in many similar situations (cf. [11]), it is easily seen from the proof that in fact for most n it is concentrated on a single value k_n, but there are transitional regimes, close to the values of n where k_n changes, where $F_n(\alpha)$ takes two values with comparable probabilities.

Note that when $p = 1/2$, the second term on the right hand side disappears, and thus simply $k_n = \log n + O(1)$; in particular, two different values of $\alpha \in (0,1)$ have their corresponding k_n differing by $O(1)$ only. When $p \neq 1/2$, changing α means shifting k_n by $\Theta(\log^{1/2} n)$. By Theorem 2.1, whp $F_n(\alpha)$ is shifted by the same amounts.

To the first order, we thus have the following simple result.

COROLLARY 2.1. *For any fixed* α *and* p *with* $0 < \alpha < 1$ *and* $0 < p < 1$,

$$F_n(\alpha) = \log_{\frac{1}{\sqrt{pq}}} n + O_p(\sqrt{\ln n});$$

in particular, $F_n(\alpha)/\log_{1/\sqrt{pq}} n \xrightarrow{\mathrm{p}} 1$ *as* $n \to \infty$.

Surprisingly enough, the fill up level for $\alpha = 1$ and $\alpha < 1$ are quantitatively different for $p \neq 1/2$. It is well known, as explained in the introduction, that the regular fillup F_n is concentrated on two points around $\log n/\log(1/p_{\min})$, while the partial fillup $F_n(\alpha)$ concentrates around $k_n \sim \log n/\log(1/\sqrt{pq})$. Secondly, the leading term of $F_n(\alpha)$ does not depend on α and the second term is proportional to $\sqrt{\log n}$, while for the regular fillup F_n the second term is of order $\log\log\log n$.

Theorem 2.1 yields several consequences for the behavior of α-LC tries. In particular, it implies the typical behavior of the depth, that is, the search time. Below we formulate our main second result concerning the depth for α-LC tries delaying the formal proof (that will follow the footsteps of [4]) till the final (journal) version of the paper. However, after the statement of theorem we provide a brief heuristics justification.

THEOREM 2.2. *For any fixed* $0 < \alpha < 1$ *and* $p \neq 1/2$ *we have*

$$D_n(\alpha) \overset{\mathrm{p}}{\sim} \frac{\log\log n}{-\log\left(1 - \frac{h}{\log(1/\sqrt{pq})}\right)} \tag{2.3}$$

as $n \to \infty$ *where* $h = -p\log p - (1-p)\log(1-p)$ *is the entropy rate of the source.*

First, let us explain heuristically our estimate for $D_n(\alpha)$. By the Asymptotic Equipartition Property (cf. [24]) at level k_n there are about $n2^{-hk_n}$ strings where h is the entropy. That is, $n2^{-hk_n} \approx n^{1-h/b}$ where for simplicity $b = \log(1/\sqrt{pq})$. In the next level, we shall have about $n^{(1-h/b)^2}$ nodes, and so on. In particular, at level $D_n(\alpha)$ we have approximately

$$n^{(1-h/b)^{D_n(\alpha)}} = O(1)$$

nodes. This leads to our estimate (2.3) of Theorem 2.2.

As a direct consequence of Theorem 2.2 we can numerically quantify experimental results observed by Nilsson and Karlsson who reported in [18] a "dramatic improvement" in the search time of α-LC tries over the regular LC tries. In a regular LC trie the search time is $O(\log\log n)$ with the constant in front of $\log\log n$ being $1/\log(1 - h/\log(1/p_{\min}))^{-1}$ [4]. For α-LC tries this constant decreases to $1/\log(1 - h/\log(1/\sqrt{pq}))^{-1}$. While it is hardly a "dramatic improvement", the fact that we deal with a slowly growing leading term $\log\log n$, may indeed lead to experimentally observed significant changes in the search time.

3 Poissonization

In this section we consider a Poissonized version of the problem, where there are $\mathrm{Po}(\lambda)$ strings inserted in the trie. We let $\tilde{F}_\lambda(\alpha)$ denote the α-fillup level of this trie.

THEOREM 3.1. *Let* α *and* p *be fixed with* $0 < \alpha < 1$ *and* $0 < p < 1$, *and let* $\tilde{F}_\lambda(\alpha)$ *be the* α*-fillup level for the trie formed by* $\mathrm{Po}(\lambda)$ *random strings as above. Then, for each* $\lambda > 0$ *there is an integer*

$$k_\lambda = \log_{\frac{1}{\sqrt{pq}}} \lambda - \frac{|\ln(p/q)|}{2\ln^{3/2}(1/\sqrt{pq})}\Phi^{-1}(\alpha)\sqrt{\ln\lambda} + O(1) \tag{3.4}$$

such that whp (as $\lambda \to \infty$*)* $\tilde{F}_\lambda(\alpha) = k_\lambda$ *or* $k_\lambda + 1$.

We shall prove Theorem 3.1 through a series of lemmas. Observe first that a node at level k can be labelled by a binary string of length k, and that the node is filled if and only if at least two of the inserted strings begin with this label. For $r \in \{0,1\}^k$, let $N_1(r)$ be the number of ones in r, and let $P(r) = p^{N_1(r)}q^{k-N_1(r)}$ be the probability that a random string begins with r. Then, in the Poissonized version, the number of inserted strings beginning with $r \in \{0,1\}^k$ has a Poisson distribution $\mathrm{Po}(\lambda P(r))$, and these numbers are independent for different strings r of the same length.

Consequently,

$$X_k = \sum_{r\in\{0,1\}^k} I_r \tag{3.5}$$

where I_r are independent indicators with

$$\mathbb{P}(I_r = 1) = \mathbb{P}(\mathrm{Po}(\lambda P(r)) \ge 2) = 1-(1+\lambda P(r))e^{-\lambda P(r)}. \tag{3.6}$$

Hence,

$$\mathbf{Var}\,(X_k) = \sum_{r\in\{0,1\}^k} P(I_r = 1)\big(1 - P(I_r = 1)\big) < 2^k$$

so $\mathbf{Var}\,(\overline{X}_k) < 2^{-k}$ and, by Chebyshev's inequality,

$$\mathbb{P}(|\overline{X}_k - \mathbf{E}\,\overline{X}_k| > 2^{-k/3}) \to 0. \tag{3.7}$$

Consequently, $\overline{X}_k$ is sharply concentrated, and it is enough to study its expectation. (It is straightforward to calculate $\mathbf{Var}\,(X_k)$ more precisely, and to obtain a normal limit theorem for X_k, but we do not need that.)

Assume first $p > 1/2$.

LEMMA 3.1. *If $p > 1/2$ and*

$$k = \log_{\frac{1}{\sqrt{pq}}} \lambda - \frac{\ln(p/q)}{2\ln^{3/2}(1/\sqrt{pq})}\Phi^{-1}(\alpha)\sqrt{\ln\lambda} + O(1), \tag{3.8}$$

then $\mathbf{E}\,\overline{X}_k = \alpha + O(k^{-1/2})$.

Proof. Let $\rho = p/q > 1$ and define γ by $\lambda p^\gamma q^{k-\gamma} = 1$, i.e.,

$$\rho^\gamma = \left(\frac{p}{q}\right)^\gamma = \lambda^{-1}q^{-k},$$

which leads to

$$\gamma = \frac{k\ln(1/q) - \ln\lambda}{\ln(p/q)}. \tag{3.9}$$

Let $\mu_j = \lambda p^j q^{k-j} = \rho^{j-\gamma}$. By (3.5) and (3.6),

$$\mathbf{E}\,\overline{X}_k = 2^{-k}\sum_{j=0}^{k}\binom{k}{j}\mathbb{P}(\mathrm{Po}(\mu_j) \ge 2). \tag{3.10}$$

If $j < \gamma$, then $\mu_j < 1$ and

$$\mathbb{P}(\mathrm{Po}(\mu_j) \ge 2) < \mu_j^2 < \mu_j.$$

If $j \ge \gamma$, then $\mu_j \ge 1$ and

$$1 - \mathbb{P}(\mathrm{Po}(\mu_j) \ge 2) = (1+\mu_j)e^{-\mu_j} \le 2\mu_j e^{-\mu_j} < 4\mu_j^{-1}.$$

Hence (3.10) yields, using $\binom{k}{j} \le \binom{k}{\lfloor k/2\rfloor} = O(2^k k^{-1/2})$,

$$\begin{aligned}\mathbf{E}\,\overline{X}_k &= 2^{-k}\sum_{j<\gamma}\binom{k}{j}O(\mu_j) + 2^{-k}\sum_{j\ge\gamma}\binom{k}{j}(1-O(\mu_j^{-1})) \\ &= 2^{-k}\sum_{j\ge\gamma}\binom{k}{j} + 2^{-k}\sum_{j=0}^{k}\binom{k}{j}O(\rho^{-|j-\gamma|}) \\ &= \mathbb{P}\big(\mathrm{Bi}(k,1/2)\ge\gamma\big) + O(k^{-1/2}).\end{aligned} \tag{3.11}$$

By the Berry–Esseen theorem [6, Theorem XVI.5.1],

$$\mathbb{P}(\mathrm{Bi}(k,1/2)\ge\gamma) = 1 - \Phi\Big(\frac{\gamma - k/2}{\sqrt{k/4}}\Big) + O(k^{-1/2}). \tag{3.12}$$

By (3.9) and the assumption (3.8),

$$\begin{aligned}\gamma - \frac{k}{2} &= \frac{1}{\ln(p/q)}\left(k\ln\frac{1}{q} - \ln\lambda - \frac{k}{2}\ln\frac{p}{q}\right) \\ &= \frac{1}{\ln(p/q)}\left(k\ln\frac{1}{\sqrt{pq}} - \ln\lambda\right) \\ &= \frac{\ln(1/\sqrt{pq})}{\ln(p/q)}\left(k - \log_{1/\sqrt{pq}}\lambda\right) \\ &= -\tfrac{1}{2}(\ln(1/\sqrt{pq}))^{-1/2}\Phi^{-1}(\alpha)\sqrt{\ln\lambda} + O(1) \\ &= -\tfrac{1}{2}\Phi^{-1}(\alpha)k^{1/2} + O(1).\end{aligned} \tag{3.13}$$

This finally implies

$$1 - \Phi\Big(\frac{\gamma - k/2}{\sqrt{k/4}}\Big) = 1 - \Phi(-\Phi^{-1}(\alpha)) + O(k^{-1/2}) =$$

$$= \alpha + O(k^{-1/2}),$$

and the lemma follows by (3.11) and (3.12).

LEMMA 3.2. *Fix $p > 1/2$. For every $A > 0$, there exists $c > 0$ such that if $|k - \log_{1/\sqrt{pq}}\lambda| \le Ak^{1/2}$, then $\mathbf{E}\,\overline{X}_k - \mathbf{E}\,\overline{X}_{k+1} > ck^{-1/2}$.*

Proof. A string $r \in \{0,1\}^k$ has two extensions $r0$ and $r1$ in $\{0,1\}^{k+1}$. Clearly, $I_{r0}, I_{r1} \le I_r$, and if there are exactly 2 (or 3) of the inserted strings beginning with r, then $I_{r0} + I_{r1} \le 1 < 2I_r$. Hence

$$\begin{aligned}\mathbf{E}\,(2X_k - X_{k+1}) &= \sum_{r\in\{0,1\}^k}\mathbf{E}\,(2I_r - I_{r0} - I_{r1}) \\ &\ge \sum_{r\in\{0,1\}^k}\mathbb{P}\big(\mathrm{Po}(\lambda P(r)) = 2\big).\end{aligned} \tag{3.14}$$

Let ρ and γ be as in the proof of Lemma 3.1, and let $j = \lceil\gamma\rceil$. Then $\mu_j = \rho^{j-\gamma} \in [1,\rho]$ and thus

$\mathbb{P}(\mathrm{Po}(\mu_j) = 2) \ge \frac{1}{2}e^{-\rho}$. Moreover, by (3.13) and the assumption,

$$|j - k/2| \le \frac{\ln(1/\sqrt{pq})}{\ln(p/q)} Ak^{1/2} + 1 = O(k^{1/2}).$$

Thus, if k is large enough, we have by the standard normal approximation of the binomial probabilities (which follows easily from Stirling's formula, as found already by de Moivre [5])

$$2^{-k}\binom{k}{j} = \frac{1+o(1)}{\sqrt{2\pi k/4}} e^{-2(j-k/2)^2/k} \ge c_1 k^{-1/2}$$

for some $c_1 > 0$. Hence, by (3.14),

$$\begin{aligned} \mathbf{E}\,\overline{X}_k - \mathbf{E}\,\overline{X}_{k+1} &= 2^{-k-1}\mathbf{E}\,(2X_k - X_{k+1}) \\ &\ge 2^{-k-1}\binom{k}{j}\mathbb{P}(\mathrm{Po}(\mu_j) = 2) \\ &\ge \frac{c_1 e^{-\rho}}{4} k^{-1/2} \end{aligned}$$

as needed.

Now assume $p > 1/2$. Starting with any k as in (3.8), we can by Lemmas 3.1 and 3.2 shift k up or down $O(1)$ steps and find k_λ as in (3.4) such that, for a suitable $c > 0$, $\mathbf{E}\,\overline{X}_{k_\lambda} \ge \alpha + \frac{1}{2}ck_\lambda^{-1/2} > \mathbf{E}\,\overline{X}_{k_\lambda+1}$ and $\mathbf{E}\,\overline{X}_{k_\lambda+2} \le \mathbf{E}\,\overline{X}_{k_\lambda+1} - ck_\lambda^{-1/2} < \alpha - \frac{1}{2}ck_\lambda^{-1/2}$. It follows by (3.7) that whp $\overline{X}_{k_\lambda} \ge \alpha$ and $\overline{X}_{k_\lambda+2} < \alpha$, and hence $\tilde{F}_\lambda(\alpha) = k_\lambda$ or $k_\lambda + 1$.

This proves Theorem 3.1 in the case $p > 1/2$. The case $p < 1/2$ follows by symmetry, interchanging p and q.

In the remaining case $p = 1/2$, all $P(r) = 2^{-k}$ are equal. Thus, by (3.5) and (3.6),

$$\mathbf{E}\,\overline{X}_k = \mathbb{P}(\mathrm{Po}(\lambda 2^{-k}) \ge 2). \tag{3.15}$$

Given $\alpha \in (0,1)$, there is a $\mu > 0$ such that $\mathbb{P}(\mathrm{Po}(\mu) \ge 2) = \alpha$. We take $k_\lambda = \lfloor \log(\lambda/\mu) - 1/2 \rfloor$. Then, $\lambda 2^{-k_\lambda} \ge 2^{1/2}\mu$ and thus $\mathbf{E}\,\overline{X}_{k_\lambda} \ge \alpha_+$ for some $\alpha_+ > \alpha$. Similarly, $\mathbf{E}\,\overline{X}_{k_\lambda+2} \le \alpha_-$ for some $\alpha_- < \alpha$, and the result follows in this case too.

4 Depoissonization

To complete the proof of Theorem 2.1 we must depoissonize the results obtained in Theorem 3.1, which we do in this section.

Proof. [Proof of Theorem 2.1] Given an integer n, let k_n be as in the proof of Theorem 3.1 with $\lambda = n$, and let $\lambda_\pm = n \pm n^{2/3}$. Then $\mathbb{P}(\mathrm{Po}(\lambda_-) \le n)) \to 1$ and $\mathbb{P}(\mathrm{Po}(\lambda_+) \ge n)) \to 1$ as $n \to \infty$. By monotonicity, we thus have whp $\tilde{F}_{\lambda_-}(\alpha) \le F_n(\alpha) \le \tilde{F}_{\lambda_+}(\alpha)$, and by Theorem 3.1 it remains only to show that we can take $k_{\lambda_-} = k_{\lambda_+} = k_n$.

Let us now write $X_k(\lambda)$ and $\overline{X}_k(\lambda)$, since we are working with several λ.

LEMMA 4.1. *Assume $p \ne 1/2$. Then, for every k,*

$$\frac{d}{d\lambda}\mathbf{E}\,\overline{X}_k(\lambda) = O(\lambda^{-1}k^{-1/2}).$$

Proof. We have

$$\frac{d}{d\mu}\mathbb{P}(\mathrm{Po}(\mu) \ge 2) = \frac{d}{d\mu}((1 - (1+\mu)e^{-\mu}) = \mu e^{-\mu}$$

and thus, by (3.10) and the argument in (3.11),

$$\begin{aligned} \frac{d}{d\lambda}\mathbf{E}\,\overline{X}_k(\lambda) &= 2^{-k}\sum_{j=0}^{k}\binom{k}{j}\mu_j e^{-\mu_j}\frac{d\mu_j}{d\lambda} \\ &= \lambda^{-1}2^{-k}\sum_{j=0}^{k}\binom{k}{j}\mu_j^2 e^{-\mu_j} \\ &= O\Big(\lambda^{-1}\sum_{j=0}^{k}2^{-k}\binom{k}{j}\min(\mu_j, \mu_j^{-1})\Big) \\ &= O(\lambda^{-1}k^{-1/2}) \end{aligned}$$

which completes the proof.

By Lemma 4.1, $|\mathbf{E}\,\overline{X}_k(\lambda_\pm) - \mathbf{E}\,\overline{X}_k(n)| = O(n^{-1/3}k^{-1/2}) = o(k^{-1/2})$. Hence, by the proof of Theorem 3.1, for large n, $\mathbf{E}\,\overline{X}_{k_n}(\lambda_\pm) \ge \alpha + \frac{1}{3}ck_n^{-1/2}$ and $\mathbf{E}\,\overline{X}_{k_n+2}(\lambda_\pm) < \alpha - \frac{1}{3}ck_n^{-1/2}$, and thus whp $\tilde{F}_{\lambda_\pm}(\alpha) = k_n$ or $k_n + 1$. Moreover, the estimate $\mathbf{E}\,\overline{X}_{k_n} = \alpha + O(1/\sqrt{\log n})$ follows easily from the similar estimate for the Poisson version in Lemma 3.1; we omit the details. This completes the proof of Theorem 2.1 for $p > 1/2$. The case $p < 1/2$ is again the same by symmetry. The proof when $p = 1/2$ is similar, now using (3.15).

References

[1] A. Andersson and S. Nilsson, Improved behavior of tries by adaptive branching, *Information Processing Letters*, **46**, 295–300, 1993.

[2] L. Devroye, A Note on the Probabilistic Analysis of Patricia Tries, *Random Structures and Algorithms*, **3**, 203–214, 1992.

[3] L. Devroye, An analysis of random LC tries, *Random Structures and Algorithms*, **19**, 359–375, 2001.

[4] L. Devroye and W. Szpankowski, Probabilistic behavior of asymmetric level compressed tries, *Random Structures & Algorithms*, **26**, 2005

[5] A. de Moivre, *The Doctrine of Chances*, 2nd ed., H. Woodfall, London, 1738

[6] W. Feller, *An Introduction to Probability Theory and its Applications,* Vol. II. 2nd ed., Wiley, New York, 1971.

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

[8] P. Iivonen, S. Nilsson and M. Tikkanen, An experimental study of compression methods for functional tries, in: *Workshop on Algorithmic Aspects of Advanced Programming Languages (WAAAPL'99)*, 1999.

[9] P. Jacquet and W. Szpankowski, Analysis of Digital Tries with Markovian Dependency, *IEEE Trans. Information Theory*, **37**, 1470–1475, 1991.

[10] P. Jacquet and W. Szpankowski, Analytical depoissonization and its applications, *Theoretical Computer Science*, **201**, 1–62, 1998

[11] C. Knessl and W. Szpankowski, On the number of full levels in tries, *Random Structures and Algorithms*, **25**, 247–276, 2004.

[12] D. E. Knuth, *Fundamental Algorithms*, 3rd ed, Addison-Wesley, Reading, Massachusetts, 1997.

[13] D. E. Knuth, *Selected Papers on Analysis of Algorithms*, CSLI, Stanford, 2000.

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

[15] S. Nilsson, *Radix Sorting & Searching*, PhD Thesis, Lund University, 1996.

[16] S. Nilsson and G. Karlsson, Fast address look-up for Internet routers, in: *Proceedings IFIP 4th International Conference on Broadband Communications*, 11–22, 1998.

[17] S. Nilsson and G. Karlsson, IP-address lookup using LC-tries, *IEEE Journal on Selected Areas in Communications*, **17**(6), 1083–1092, 1999.

[18] S. Nilsson and M. Tikkanen, An experimental study of compression methods for dynamic tries, *Algorithmica*, **33**(1), 19–33, 2002.

[19] B. Pittel, Asymptotic Growth of a Class of Random Trees, *Annals of Probability*, **18**, 414–427, 1985.

[20] B. Pittel, Paths in a Random Digital Tree: Limiting Distributions, *Adv. in Applied Probability*, **18**, 139–155, 1986.

[21] Y. Reznik, Some Results on Tries with Adaptive Branching, *Theoretical Computer Science*, **289**, 1009–1026, 2002.

[22] V. Srinivasan and G. Varghese, Fast Address Lookups using Controlled Prefix Expansions, *ACM SIGMETRICS'98*, 1998.

[23] W. Szpankowski, On the Height of Digital Trees and Related Problems, *Algorithmica*, **6**, 256–277, 1991.

[24] W. Szpankowski *Average Case Analysis of Algorithms on Sequences*, John Wiley, New York, 2001.